

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
**SCHOOL OF INFORMATION AND
COMMUNICATION TECHNOLOGY**

LAB INSTRUCTIONS IT3080E COMPUTER NETWORKS

(FOR INTERNAL USE ONLY)

UPDATE: 29/10/2022

MỤC LỤC

| | |
|---|-----------|
| 1. INTRODUCTION..... | 3 |
| 1.1. GOAL AND SCOPE..... | 3 |
| 1.2. OBJECTIVES OF THE LAB..... | 3 |
| 1.3. GENERAL INFORMATION..... | 3 |
| 2. REGULATIONS FOR STUDENTS..... | 3 |
| 3. LAB 1: BUILDING AN ETHERNET CABLE..... | 4 |
| 3.1. PURPOSE OF WORK..... | 4 |
| 3.2. MAKE AND TEST YOUR CABLE..... | 6 |
| 4 LAB 2: LAYER 2 NETWORK DESIGN..... | 10 |
| 4.1. PURPOSE OF WORK..... | 10 |
| 4.1.3.1. BASIC SWITCH CONFIGURATION..... | 11 |
| 4.1.3.1.1. Hostname..... | 11 |
| 4.1.3.1.2. Turn Off Domain Name Lookups..... | 11 |
| 4.1.3.1.3. Set the Domain Name..... | 12 |
| 4.1.3.1.4. Configure console and other ports..... | 12 |
| 4.1.3.1.5. Usernames and Passwords..... | 12 |
| 4.1.3.1.6. Enabling login access for other devices..... | 13 |
| 4.1.3.1.7. Configure system logging..... | 13 |
| 4.1.3.1.8. Save the Configuration..... | 14 |
| 4.1.3.2. IP Address Configuration..... | 15 |
| 4.1.4. Internet Protocol..... | 15 |
| 4.1.5. Internet Control Message Protocol..... | 16 |
| 4.2. PRACTICE STEPS..... | 17 |
| 4.3. REPORT FOR LAB 2..... | 18 |
| 5. LAB 3: STATIC ROUTING IN IP NETWORK..... | 19 |
| 5.1 OBJECTIVES AND REQUIREMENTS..... | 19 |
| 5.2 CONTENTS OF THE LAB..... | 23 |

| | |
|------------------------------------|-----------|
| 1.3. REPORT FOR LAB 3..... | 31 |
| 6. LAB 4. UDP AND TCP..... | 34 |
| 6.1. LAB OVERVIEW..... | 34 |
| 6.2. LAB TASKS..... | 36 |
| 7. LAB 5: DNS AND HTTP..... | 45 |
| 7.1. LAB OVERVIEW..... | 45 |
| 7.2. LAB TASKS..... | 48 |

1. INTRODUCTION

1.1. GOAL AND SCOPE

1.2. OBJECTIVES OF THE LAB

1.3. GENERAL INFORMATION

Workload: 5 labs * 3 periods/ lab

2. REGULATIONS FOR STUDENTS

1. Comply with the regulations in the practice room
2. Print lab instructions (including manuals and report templates), read them carefully and review relevant knowledge contents
3. Bring the lab instructions when coming to practice classes.
4. Do the exercises according to the instructions in the document.
Do not perform contents other than the instructions, unless requested by the instructor
5. Submit the lab report and other results as required and instructed at the end of the practice session
6. Plagiarism in any form and for any reason will result in 0 score.

3. LAB 1: BUILDING AN ETHERNET CABLE

3.1. PURPOSE OF WORK

3.1.1. Objectives

- Study of technical data and designs of UTP Cat5 and Cat6 cables (twisted pair) used in computer networks.
- Exploring the EIA/TIA-568A/B standards for connecting the 8P8C connector (sometimes called RJ45) and the Twisted Pair cable.
- Get the initial skills of crimping and testing UTP Cat5 network cable.

3.1.2. Required Resources

- Bulk Ethernet Cable
- RJ45 Crimpable Connectors for CAT-5e (or for CAT-6)
- Crimping Tool for RJ-45

3.1.3. Task for the work

- Study of technical data and designs of EIA/TIA-568A/B, UTP/STP, RJ45.
- Make and test Your Cable UTP Cat5e.

3.1.4. Theoretical basis

3.4.1.1. Types of Cables

- Cable Types by Wired
Both for internal (building cables) and for external wiring, three classes of wired communication lines are most often used
 - o Twisted pair
 - o Coaxial cables
 - o Fiber optic cables

- Cable Types by UTP Categories

- Cable Types by Cross

There are two types of network cables

- o Direct cable (Direct Connection) - to connect a network card port to a switch or hub
- o A crossover cable with an inverted pinout of the connector for directly connecting two network cards installed in computers, as well as for connecting some older models of hubs and switches (uplink port).

There are some network cards that can automatically detect the type of cable and adapt to it.

- Cable Types by Crimping

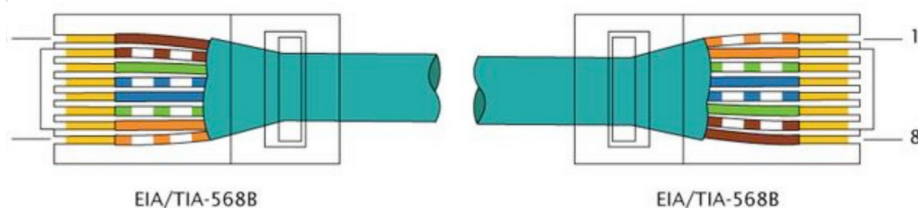
There are two types of crimping (preparation) of the plug. In practice, the EIA / TIA-568B connection scheme is more often used.

| UTP Category | Data Rate | Max. Length | Cable Type | Application |
|--------------|---------------|-------------|--------------|--|
| CAT1 | Up to 1Mbps | - | Twisted Pair | Old Telephone Cable |
| CAT2 | Up to 4Mbps | - | Twisted Pair | Token Ring Networks |
| CAT3 | Up to 10Mbps | 100m | Twisted Pair | Token Rink & 10BASE-T Ethernet |
| CAT4 | Up to 16Mbps | 100m | Twisted Pair | Token Ring Networks |
| CAT5 | Up to 100Mbps | 100m | Twisted Pair | Ethernet, FastEthernet, Token Ring |
| CAT5e | Up to 1 Gbps | 100m | Twisted Pair | Ethernet, FastEthernet, Gigabit Ethernet |
| CAT6 | Up to 10Gbps | 100m | Twisted Pair | GigabitEthernet, 10G Ethernet (55 meters) |
| CAT6a | Up to 10Gbps | 100m | Twisted Pair | GigabitEthernet, 10G Ethernet (55 meters) |
| CAT7 | Up to 10Gbps | 100m | Twisted Pair | GigabitEthernet, 10G Ethernet (100 meters) |

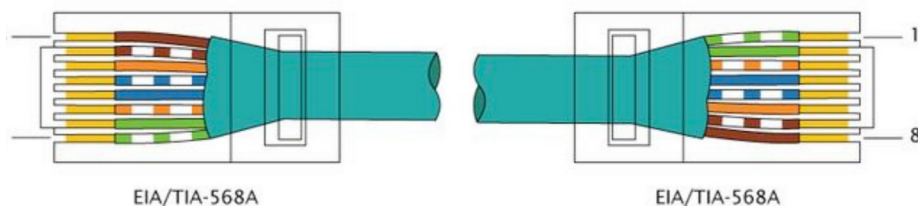
- o EIA/TIA-568A
- o EIA/TIA-568B

- Ethernet UTP Cables Schemes

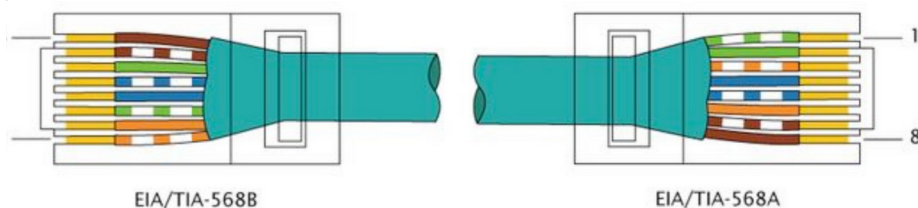
1 Direct Cable Cat5/5e/6 EIA/TIA-568B



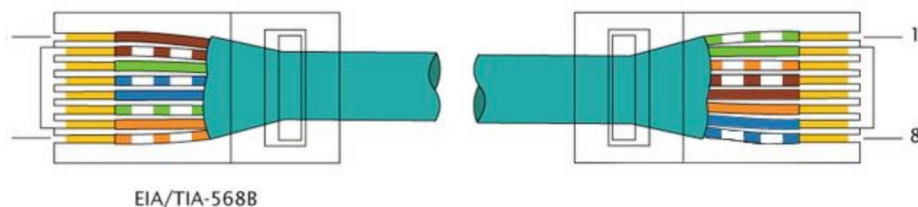
Direct Cable Cat5/5e/6 EIA/TIA-568A



3 Cross Cable Cat5/5e/6 EIA/TIA-568B/A (100MB/s)



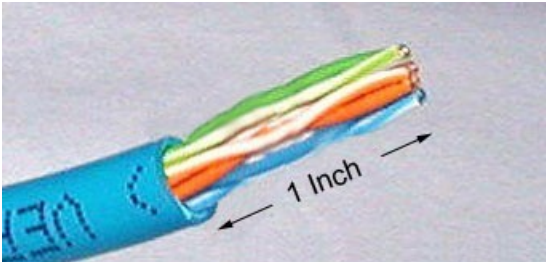
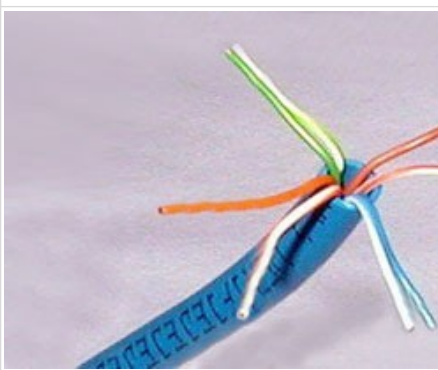


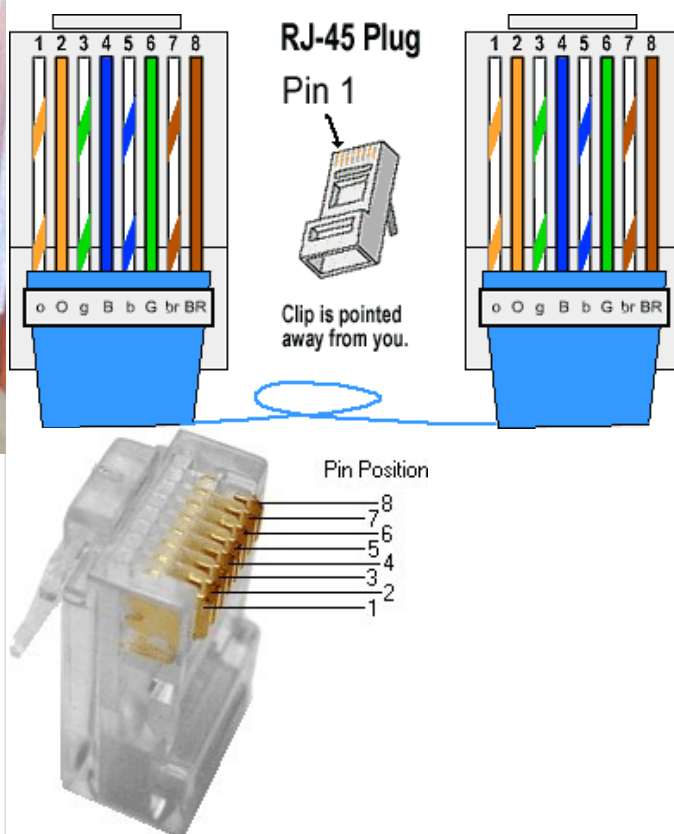
4 Cross Cable Cat5/5e/6 EIA/TIA-568B/x (1000MB/s)



- SHORT FILMS ABOUT HANDMADE ETHERNET UTP/STP CABLES.
 - o Terminating CAT5 Unshielded Cable with RJ45 Connector
<https://www.youtube.com/watch?v=WvP0D0jiyLg>
 - o Terminating CAT6 Unshielded Cable with EZ-RJ45 Connector
<https://www.youtube.com/watch?v=0vZs5oHyzgU>
 - o Terminating CAT6 Shielded Cable with RJ45 Connector
<https://www.youtube.com/watch?v=-bQjrDirT6g>

3.2. MAKE AND TEST YOUR CABLE

| | | |
|--------|--|---|
| Step 1 |  | Use Crimping Tool or Scissors, cut a cable 0.5 m long. |
| Step 2 | <p>or</p>  <p>or</p> | <p>Use Crimping Tool or Cable Stripper clean the outer shell about 1 Inch (2.5 cm) from the one end of the cable.</p>  |
| Step 3 |  | Unwind and pair the similar colors. |
| Step 4 | | Pinch the wires between your fingers and straighten them out as shown. The color order is important to get correct. |

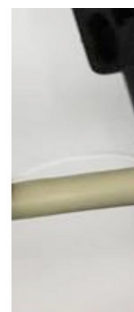
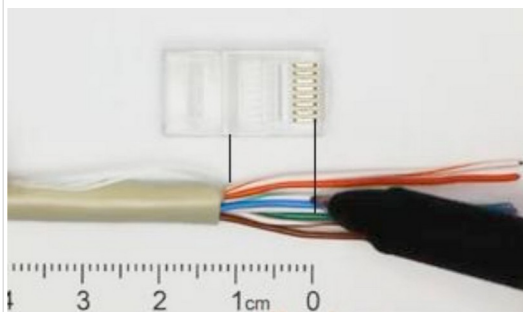



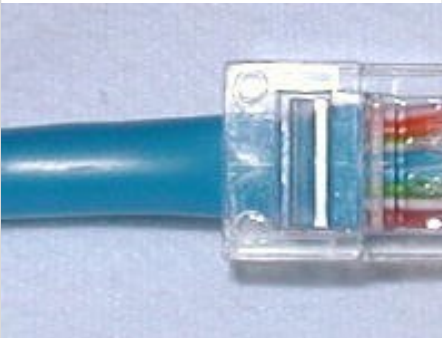
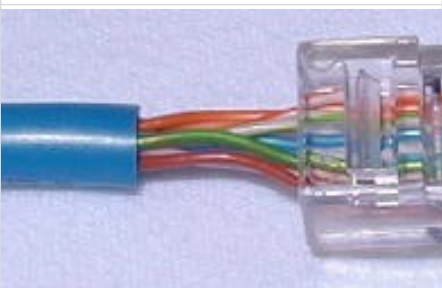
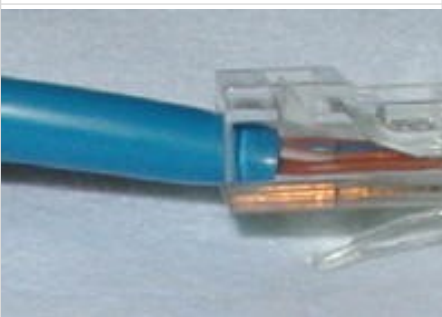
Step 5



or

Use Crimping Tool or Scissors to make a straight cut across the 8 wires to shorten them to **1/2 Inch** (1.3 cm) from the cut sleeve to the end of the wires.



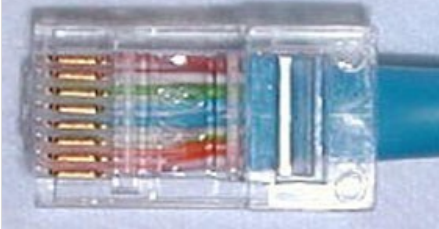
| | | |
|--------|---|---|
| Step 6 |  | <p>Carefully push all 8 unstripped colored wires into the connector.</p> <p>TRUE WAY - Note the position of the blue plastic sleeve.</p> |
| | | <p>Also note how the wires go all the way to the end.</p> |
| |  | <p>TRUE WAY - A view from the top. All the wires are all the way in. There are no short wires.</p> |
| |  | <p>WRONG WAY - Note how the blue plastic sleeve is not inside the connector where it can be locked into place. The wires are too long. The wires should extend only 1/2 inch from the blue cut sleeve.</p> |
| |  | <p>WRONG WAY - Note how the wires do not go all the way to the end of the connector.</p> |
| Step 7 | | <p>CRIMPING THE CABLE ... carefully place the connector into the Ethernet Crimper and cinch down on the handles tightly. The copper splicing tabs on the connector</p> |



will pierce into each of the eight wires. There is also a locking tab that holds the blue plastic sleeve in place for a tight compression fit. When you remove the cable from the crimper, that end is ready to use.

Step 8

REPEAT STEPS OTHER END



For a standard "Direct" cable, repeat steps 2-7.

(For a Cross-over cable, the other end will have a different color order as shown by the crossover pictures above.)

3.3. Test Your Cable

- Use the cable tester, test the crossover cable for functionality. If it fails, check with your instructor first as to whether you should re-cable the ends and re-test
- Connect two PCs together via NICs using your Ethernet cable

3.4. Reflection

1. Which part of making cables did you find the most difficult?
2. Why do you have to learn how to make a cable if you can easily buy pre-made cables?

4 LAB 2: LAYER 2 NETWORK DESIGN

4.1. PURPOSE OF WORK

4.1.1. Objectives

- Build Layer 2 (switched) networks

- Students will see how star topology, and Spanning Tree Protocol are put to work.

4.1.2. Required Resources

- Switch cisco 2960c
- Switch TP link 4 FE ports
- Console cable to config switch from CLI
- PCs
- Software: cisco_usbconsole_driver_3_1.zip and terminal emulator (Putty)

4.1.2. Task for the work

- Understand how to config a switch, assign IP to a switch
- Knowledge: IP, subnet mask
- Make a report

4.1.3. Theoretical basis

4.1.3.1. BASIC SWITCH CONFIGURATION

4.1.3.1.1. Hostname

Your switches should be given a basic configuration as follows:

```
Router> enable
Router# config terminal
Enter configuration commands, one per line. End
with CNTL/Z.
Router(config)# hostname dist1-b1.campusX
dist1-b1.campusX(config)#
```

4.1.3.1.2. Turn Off Domain Name Lookups

Cisco devices will always try to look up the DNS for any name or address specified in the command line. You can see this when doing a trace on a router with no DNS server or a DNS server with no in-addr.arpa entries for the IP addresses. We will turn this lookup off for the labs for the time being to speed up traceroutes.

```
dist1-b1.campusX(config)# no ip domain lookup
```

4.1.3.1.3. Set the Domain Name

We will now set the domain name of our campus devices, for future use in this workshop.

```
dist1-b1.campusX(config)# ip domain name  
ws.nsrc.org
```

4.1.3.1.4. Configure console and other ports

By default, Cisco devices will try all transports available if they don't recognise what is typed into the command line. This behaviour is annoying especially if making a typo during configuration work, so we will disable the behaviour completely. We will also set the idle-timeout on the console and other ports to 30 minutes - after 30 minutes of no activity on the port, the device will disconnect the connection.

```
dist1-b1.campusX(config)# line con 0  
dist1-b1.campusX(config-line)# transport  
preferred none  
dist1-b1.campusX(config-line)# exec-timeout 30 0  
dist1-b1.campusX(config-line)# line aux 0  
dist1-b1.campusX(config-line)# transport  
preferred none  
dist1-b1.campusX(config-line)# exec-timeout 30 0  
dist1-b1.campusX(config-line)# line vty 0 4  
dist1-b1.campusX(config-line)# transport  
preferred none  
dist1-b1.campusX(config-line)# exec-timeout 30 0
```

4.1.3.1.5. Usernames and Passwords

All router usernames should be **cndlab** with password being **lab-PW**. The enable password (which takes the operator into configuration mode) needs to be **lab-EN1**.

Please do not change the username or password to anything else, or leave the password unconfigured (access to vty ports is not possible if no password is set). It is essential for a smooth operating lab that all participants have access to all routers.

```
dist1-b1.campusX(config)# username cndlab secret
lab-PW
dist1-b1.campusX(config)# enable secret lab-EN
dist1-b1.campusX(config)# service password-
encryption
```

The service password-encryption directive tells the router to encrypt all passwords stored in the router's configuration (apart from enable secret2 which is already encrypted).

4.1.3.1.6. Enabling login access for other devices

In order to let you telnet into your device in future modules of this workshop, you need to configure a password for all virtual terminal lines.

```
dist1-b1.campusX(config)# aaa new-model
dist1-b1.campusX(config)# aaa authentication
login default local
dist1-b1.campusX(config)# aaa authentication
enable default enable
```

This series of commands tells the router to look locally for standard user login (the username password pair set earlier), and to the locally configured enable secret for the enable login. By default, login will be enabled on all vty's for other teams to gain access.

4.1.3.1.7. Configure system logging

A vital part of any Internet operational system is to record logs. The router by default will display system logs on the router console. However, this is undesirable for Internet operational routers, as the console is a 9600 baud connection, and can place a high processor interrupt load at the time of busy traffic on the network. However, the router logs can also be recorded into a buffer on the router - this takes no interrupt load and it also enables to operator to check the history of what events happened on the router.

```
dist1-b1.campusX(config)# no logging console
dist1-b1.campusX(config)# logging buffer 8192
debug
```

which disables console logs and instead records all logs in a 8192 byte buffer set aside on the router. To see the contents of this internal logging buffer at any time, the command `show log` should be used at the command prompt.

And we also want to set up improved time-stamping for the log messages as well:

```
service timestamps debug datetime msec localtime
show-timezone year
service timestamps log datetime msec localtime
show-timezone year
```

which will give resolution down to milliseconds, and include the year as well.

4.1.3.1.8. Save the Configuration.

With the basic configuration in place, save the configuration. To do this, exit from enable mode by typing `end` or `<ctrl>Z`, and at the command prompt enter `write memory`.

```
dist1-b1.campusX(config)# end
dist1-b1.campusX# write memory
Building configuration...
[OK]
dist1-b1.campusX#
```

It is highly recommended that the configuration is saved quite frequently to NVRAM. If the configuration is not saved to NVRAM, any changes made to the running configuration will be lost after a power cycle or virtual machine failure

Log off the switch:

```
dist1-b1.campusX# exit
```

and then log back in again. Notice how the login sequence has changed, prompting for a username and password from the user, like this:

```
dist1-b1.campusX con0 is now available
```

Press RETURN to get started.

User Access Verification

Username:

Note that at each checkpoint in the workshop, you should save the configuration to memory – remember that powering the device off will result in it reverting to the last saved configuration in NVRAM.

4.1.3.2. IP Address Configuration

Assign a switch an IPv4 address (and IPv6 address) as follows:

```
interface vlan 1
 ip address 172.2X.0.N 255.255.0.0
 ipv6 address 2001:DB8:X:1::N/64
 load-interval 30
 no shutdown
end
```

Replace the **X** with your group number, and **N** with the address according to the address plan earlier in the notes. Note the load-interval command which will calculate the average traffic load on the interface over a 30 second period (rather than over the default 5 minutes).

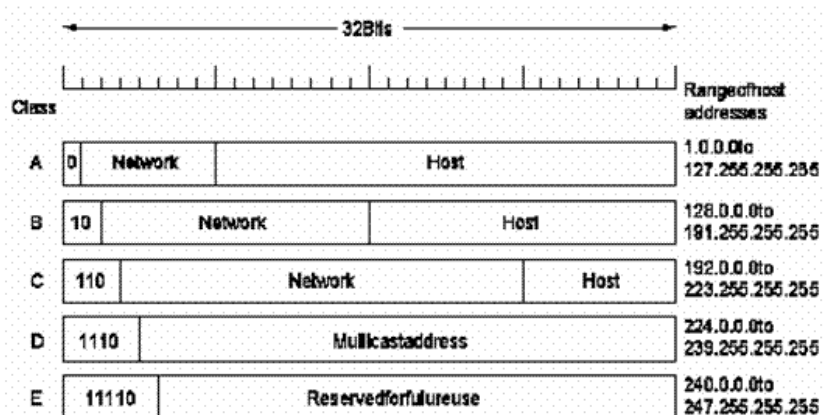
- For some unknown reason, the switch on rare occasions will add static MAC entries. These often point to the wrong interface. If you have entries like these in your configuration, trying shutting down the VLAN 1 interface mentioned, deleting the line by putting a no in front of the offending configuration, and then bringing the VLAN 1 interface back up again. If that doesn't work (the mac-address-table line is still there), ask your lab instructor.

4.1.4. Internet Protocol

- The Internet Protocol (IP) is a protocol, or set of rules, for routing and addressing packets of data so that they can travel across networks and arrive at the correct destination. Data traversing the networks is divided into smaller pieces, called packets. IP information is attached to each packet, and this information helps

routers to send packets to the right place. Every device or domain that connects to the Internet is assigned an IP address, and as packets are directed to the IP address attached to them, data arrives where it is needed.

- Once the packets arrive at their destination, they are handled differently depending on which transport protocol is used in combination with IP. The most common transport protocols are TCP and UDP.
- IP packets are created by adding an IP header to each packet of data before it is sent on its way. An IP header is just a series of bits (ones and zeros), and it records several pieces of information about the packet, including the sending and receiving IP address. In total there are 14 fields for information in IPv4 headers, although one of them is optional.
- IP provides mechanisms that enable different systems to connect to each other to transfer data. Identifying each machine in an IP network is enabled with an IP address. IPv4 provides a 32-bit IP addressing system that has four sections. For example, a sample IPv4 address might look like 192.168.0.1, which coincidentally is also commonly the default IPv4 address for a consumer router. IPv4 supports a total of 4,294,967,296 addresses. Traditionally, IPv4 has 5 categories namely A, B, C, D, E as shown in the following picture:



- Since the internet must accommodate networks of all sizes, an addressing scheme for a range of networks exists based on how the octets in an IP address are broken down. You can determine

based on the three high-order or left-most bits in any given IP address which of the five different classes of networks, A to E, the address falls within. Subnetting is the technique for logically partitioning a single physical network into multiple smaller sub-networks or subnets. A subnet mask is a 32-bit number created by setting host bits to all 0s and setting network bits to all 1s. In this way, the subnet mask separates the IP address into the network and host addresses. IPv4 addresses represented in CIDR notation are called network masks, and they specify the number of bits in the prefix to the address after a forward slash (/) separator

4.1.5. Internet Control Message Protocol

- The Internet Control Message Protocol (ICMP) is a network layer protocol used by network devices to diagnose network communication issues. ICMP is mainly used to determine whether or not data is reaching its intended destination in a timely manner. Commonly, the ICMP protocol is used on network devices, such as routers.
- The primary purpose of ICMP is for error reporting. When two devices connect over the Internet, the ICMP generates errors to share with the sending device in the event that any of the data did not get to its intended destination.
- A secondary use of ICMP protocol is to perform network diagnostics; the commonly used terminal utilities traceroute and ping both operate using ICMP. The traceroute utility is used to display the routing path between two Internet devices.

4.2. PRACTICE STEPS

1. The following diagram shows the flat network we have just built
2. Assign each computer a different IPv4 address as shown in the diagram.
3. On PC1 with the IP of 192.168.1.2 create a connection to control the switch using ssh tool.





4.3. REPORT FOR LAB 2

Student name:

Student ID:

Practice Course code:

Theory Course code:

IP of PC:

Experiment (3 điểm)

Student connects to a switch and collect information of MAC address that connect to the PC1 and PC3 at two points of time: before and after sending packets from PC1 to PC3 and vice versa.

The command to see the MAC address for port 1 (i.e., Gi0/3), the MAC address is stored in the MAC table in the switch, is as follow:

c2960#sh mac address-table interface Fa0/1

Mac Address Table

```

-----
Vlan  Mac Address      Type      Ports
---  -
32    68b5.99fc.d1df    DYNAMIC   Fa0/1
  
```

Total Mac Addresses for this criterion: 1

Use the diagram given in the section 4.2.1, fill in the following table in 3 cases:

1. PC1 ping PC2
2. PC2 ping PC3
3. PC3 ping PC4

| Port | Before pinging PC1->PC2 | After pinging PC1->PC2 |
|-------|-------------------------|---|
| Fa0/1 | Empty | MAC: 68b5.99fc.d1df Port: Fa0/1 MAC: 001b.10ae.7d00 |

| | | |
|--|--|-------------|
| | | Port: Fa0/2 |
| | | |

5. LAB 3: STATIC ROUTING IN IP NETWORKS

5.1 OBJECTIVES AND REQUIREMENTS

5.1.1. Objective

With this lab, students are equipped with practical skills about internal gateway routing mechanism. Students will be able to configure static routing for computer networks using IP routers so that the networks can transmit data to each other and to the Internet. Specifically, students practice about IP addressing, routing tables, using tools, configuration and testing statements.

5.1.2. Requirements for students

- Theoretical knowledge:
Students master routing principles in IP networks, principal operation of routing tables, setting up routing tables, assigning IP addresses.
- Practical skills:
 - o Students can connect network devices such as switches and routers to form interconnected subnets
 - o Students can set up a static routing table using the router function of Linux.
- **What to submit at the end of the practice session:**
 - o Demonstrate to teacher/TA the steps to check the connectivity at the end of sections 5.2.1, 5.2.2. Demo counts for 1 point/10.
 - o Report (on paper) using the provided form. The report includes answers to each questions. Report counts for 9 out of 10.

5.1.3. Backgrounds

5.1.3.1 IP addressing

To distinguish computers on the Internet, each machine is assigned an IP address. IP address (version 4) consists of 4 bytes, for example 10000010 10001010 00001000 00000001.

For convenience, the IP address is written as 4 decimal numbers separated by dots, for example, the above address is written as: 130.238.8.1.

Each IP address consists of two parts, the network identifier bits (network ID, on the left), which determine the network where the node is connected to, and the host identifier bits (hostID, on the right) identify a single station in the network.

The length of the network ID and host ID are not fixed. To define these lengths, one can apply one of two principles:

- Classify addresses into classes A, B, C, D, E (see the lectures), or
- Use netmasks: the mask is a number indicating how many leftmost bits belong to the network identifier.

For example, the network mask could be 24, it means the left-most 24 bits belong to network identifier. The network mask can also be written in 32-bit form as an IP address with the network identifier bits being 1 and the host identifier bits being 0.

For example, network mask 24 is written as 11111111 11111111 11111111 00000000,

or also can write as decimal as IP address to 255.255.255.0.

With a 24-bit mask, the host ID part contains $32-24=8$ bits. Thus, the network has maximally 256 distinct IP addresses. Excluding two special IP addresses: network address with all 0 bits in hostID and broadcast address with full bit 1 in hostID part, the remaining 254 addresses can be assigned to hosts.

5.1.3.2. Inter-connection and routing

The Internet consists of many small LANs connected together. To transfer data between these LANs, a data forwarding mechanism is required. That mechanism in IP networks, called IP forwarding mechanism, is implemented on IP routers that stay in between two LANs.

A router is a network node that basically has at least 2 interfaces

connecting to (belonging to) 2 different LANs. The router receives an IP packet from one interface and forwards the packet to the other interfaces depending on the packet's destination address, in order to redirect the packet to the destination network. To do so, it is first necessary to determine paths for packets from all sources to all destinations. The resulting paths are written to the routers in the form of a routing table.

The routing table must be built based on the topology of the network. The routing table must be updated regularly to reflect topological changes in the network. In small, simple networks, the routing table can be built manually (static routing), or built using routing protocols automatically. Some popular routing protocols are Routing Information Protocol (RIP) and Open Shortest Path First (OSPF).

5.1.3.3. Routing table and configuration commands

The routing table consists of many rows with the structure:

[Destination, netmask, cost, next hop, interface]

Example:

| | | | | |
|---------------|---------------|------|---------|------|
| 169.254.0.0 | 255.255.0.0 | 1000 | 0.0.0.0 | eth0 |
| 192.168.6.0 | 255.255.255.0 | 0 | 0.0.0.0 | eth1 |
| 192.168.122.0 | 255.255.255.0 | 0 | 0.0.0.0 | eth2 |
| 0.0.0.0 | 0.0.0.0 | 0 | 0.0.0.0 | eth2 |

When a packet arrives at the router with destination address Y, the router applies the network mask of each row of the routing table to the address Y to see if Y belongs to the corresponding destination network or not? If yes, then the route is considered as "match". If there are multiple matching route, the "Longest matching" principle is applied, whereby the route corresponding to the destination network with the longest number of matching bits is selected.

If no routes match, the default route is applied. The default route has the network address and the mask contain all 0. If there is no default route, the packet is dropped.

In Linux systems, Ethernet network interfaces are usually named ethX where X is an incrementing numbers from 0. For example, the first network interface is called eth0, the next is called eth1, and so on. In some cases the network interface may also have a different name.

To see the network interface name on a node

```
$ ifconfig
```

To activate, configuration IP address for a network interface.

```
$ ifconfig
```

Example:

```
$ ifconfig eth0 192.168.200.1 netmask 255.255.255.0 up
```

This command turns on network interface eth0 and assign it the address 192.168.200.1/24

To display and interact with routing table

```
$ route
```

Example, add a route to the network 192.168.205.0 by forwarding data to the next hop 192.168.200.1 which is connected directly to the current node.

```
$ sudo route add -net 192.168.205.0 netmask 255.255.255.0 gw  
192.168.200.1
```

To add a default route to going through next hop 10.1.0.1

```
$ sudo route add default gw 10.1.0.1
```

To show routing table

```
$ route -n
```

Command to print a route from the current host to another host

```
$ traceroute
```

Example:

```
$ traceroute -n -z 1 192.168.205.1
```

Activate IP forwarding function of a Linux host, making it working as a router

```
$ sudo sysctl net/ipv4/ip_forward
```

```
$ sudo sysctl -w net.ipv4.ip_forward=1
```

Install DHCP server program

```
$ sudo apt install isc-dhcp-server
```

Following commands need to be used to setup DHCP server service on a router

Students need to understand the configuration file `/etc/dhcp/dhcpd.conf`, the contents of the file may look like:

```
authoritative;
subnet 10.1.1.0 netmask 255.255.255.0 {
    range 10.1.1.101 10.1.1.200;
    option routers 10.1.1.2; }
```

Where the meaning of information are:

Network: 10.1.1.0

Subnet: 255.255.255.0

DCHP Range: 10.1.1.101 - 10.1.1.200

“option” item specify the information to be pushed to DHCP clients. For example, “option routers” specify the gateway/router to be set up on the host receiving the configuration. “option” item is optional.

Restart the DHCP server to activate the above configuration by the following command:

```
sudo systemctl restart isc-dhcp-server.service
```

Verify the list of the host receiving dynamic IP address from DHCP server

```
$ dhcp-lease-list
```

The result looks like:

To get manufacturer names please download

<http://standards.ieee.org/regauth/oui/oui.txt> to `/usr/local/etc/oui.txt`

Reading leases from `/var/lib/dhcp/dhcpd.leases`

| MAC | IP | hostname | valid until |
|-------------------|------------|----------------|-----------------------|
| manufacturer | | | |
| ===== | | | |
| 00:0c:29:45:ba:4d | 10.1.1.135 | DESKTOP-8UK989 | 2019-12-12 13:22:00 - |
| NA- | | | |

Command to turn on the DHCP service on a client (using interface eth0)

```
$ sudo dhcpcclient -r eth0
```

Students can use command “man” to see the manual about above commands.

Note: Gateway is generally the router where in/out data traffics of a network have to go through.

5.2 CONTENTS OF THE LAB

Students are divided into group of 4 members.

Each group is given 3 Raspberry PI to act as routers, 3 switches to form 3 LAN, 3 computer to act as workstations and 6 USB Ethernet to provide PI with network ports.

5.2.1. Connects two LANs using router

A company has 2 headquarters in Saigon and Hanoi (see picture). Each headquarters has a LAN. Each LAN has several workstations, but you can only access 2 hosts named hn-workstation in Hanoi, and sg-workstation in Saigon, and two routers hn-router and sg-router in each LAN.

Each LAN can be used to communicate within a headquarter but cannot communicate with the other one. To make the two headquarters to communicate with each other, a leased line was established between the two headquarters of Saigon and Hanoi.

Saigon network is provided with IP address range 10.1.0.0 with mask 255.255.0.0. Similarly, Hanoi network is provided with IP address range 10.2.0.0 and mask 255.255.0.0.

According to IP addressing principle, hosts in the same network must have the same NetID and can communicate directly with each other. With an IP address and a network mask, we can determine the address of the network containing this IP address.

5.2.1.1 Assign IP addresses

Question 1 (1 pt): Assign suitable IP address to sg-workstation, hn-workstation and network interfaces of routers. Fill these address into the network topology in Figure 1.

IP of hn-workstation:

Mask:.....

IP of sg-workstation:

Mask:.....

IP of Hanoi-eth0:

Mask:.....

IP of Hanoi-eth1:

Mask:.....

IP of Saigon-eth0:

Mask:.....

IP of Saigon-eth1:

Mask:.....

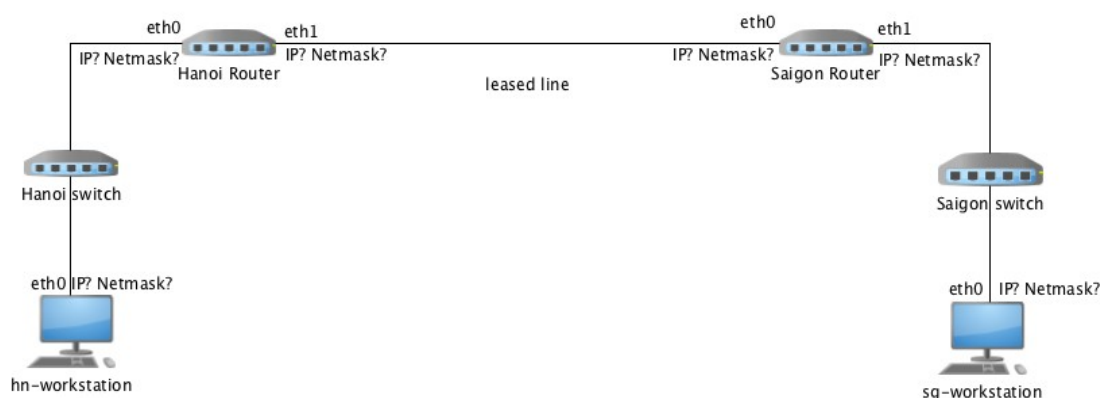


Figure 1: Sơ đồ mạng

5.2.1.2 Connect and Configure

Mục tiêu của phần thực hành là kết nối mạng theo sơ đồ Figure 1 và cấu hình sao cho các trạm có thể nói chuyện với nhau. Để làm được như vậy, sinh viên cần thực hiện cấu hình theo các bước như sau.

Lưu ý: Để làm được bài thực hành này, sinh viên cần có quyền quản trị khi thực hiện các câu lệnh (quyền root hoặc dùng lệnh sudo)

The goal of the this section is to create a network as shown in Figure 1 and configure it so that workstations can talk to each other. To do so, students need to make configuration according to the following steps.

Note: To do this exercise, students need to have administrative permissions when executing commands (root privileges or using sudo command).

Step 1: *Connects devices according to Figure 1.*

Step 2: *Configure the work stations according to following steps:*

- Assign IP address to the workstations.
- Set routing rules for workstations. We need to tell a workstation

that for destination addresses belonging to the its network, it is possible to transfer data directly without going through a routers. Usually this rule is automatically added to the routing table each time a network interface is enabled. The route -n command can be used to check the routing table

- For destination addresses that do not belong to the same network, it is necessary to forward packets to a router for forwarding further. There are two options: i) add a static route to each destination network, or ii) add a default route to all destination networks if the route to all destination networks is identical. In this lab, let's use a static route as we will expand the network in the next section.

Question 2 (1 pt): Enable the network interfaces of workstations and assign the IP address as specified in Question 1 using the ifconfig statement. The commands to use are:

On hn-workstation:

.....

On sg-workstation:

.....

Question 3 (1 pt): Setup routing rules on workstations

Use route -n to check the routing table

Use route add to add a new route to a distance network.

Write down the command to be executed on **hn-workstation** to forward data to Saigon network:

.....

Write down the command to be executed on **sg-workstation** to forward data to Hanoi network:

.....

Step 3: Configure routers:

- Set IP address for routers. Each router has two interfaces to configure: the interface to the LAN and the interface to the remote router.

The router interface connected to each LAN must have an IP address in the range of the LAN. Two interfaces of two routers connected on a leased line can have arbitrary addresses, but they must belong to the same network. That is, their IP addresses must have the same network address.

Question 4 (1 pt): Execute the command to set the IP address for the interfaces connected to the LAN of the Hanoi router:

.....

And Saigon router:

.....

Execute the command to set IP address for the interfaces connected to the leased line of the Hanoi router:

.....

And Saigon router:

.....

- Set up routing rules for routers so that they can forward packets between 2 LANs.

Question 5 (1 pt): Execute the command on the Hanoi router to add the routing rule to destinations in Saigon network:

.....

Execute the command on the Saigon router to add the routing rule to destinations in Hanoi networks:

.....

Note: In this lab, we are using Linux machines as routers, so we need to enable Linux IP forwarding by executing the following command on each router.

```
$ sysctl -w net.ipv4.ip_forward=1 - to enable IP forwarding.
```

Step 4 (0.5 pt): Validation (*to demonstrate to teacher/TA*)

Đến lúc này nếu các cấu hình đều đúng thì các máy ở các mạng đã có thể chuyển dữ liệu cho nhau. Sử dụng lệnh traceroute để kiểm tra tính thông suốt của các kết nối giữa máy trạm hn- workstation và sg-workstation. Kết quả có thể tương tự như sau:

At this point, if the configurations are correct, the hosts can transfer data to each other. Use traceroute command to check the connectivity of the connections between hn-workstation and sg-workstation. The results could be similar to the following:

```
sg-workstation:~# traceroute -n -z 1 10.2.0.10
traceroute to 10.2.0.10 (10.2.0.10), 30 hops max, 38 byte
packets
 1 10.1.0.1 2.600 ms 0.831 ms 0.802 ms
 2 10.10.0.2 3.517 ms 1.161 ms 1.156 ms
 3 10.2.0.10 7.695 ms 1.528 ms 1.514 ms
sg-workstation:~#
```

Make sure the connection is clear before proceeding to the next part of the lab.

5.2.2 DHCP service and connection to Internet

The company wants to connect to an Internet service provider (ISP) in Hanoi to provide Internet access to its machines. For some reason, the company just wants to maintain a single connection to this ISP. For both offices to access the ISP, the data streams must be routed through Hanoi. To implement that idea, at Hanoi router, eth2 network interface is added. This interface will connect directly to the ISP's Router and this router has been connected to the Internet by the ISP. See Figure 3.

To facilitate future expansion, the company wants the ISP Router to allocate dynamic IP addresses to all interfaces connected to it using the

DHCP service. So later whenever they connect a router to the ISP Router, that router will also receive an IP address automatically. The company also wants the IP addresses will be in the range:

IP address range : 192.168.N.0

Netmask: 255.255.255.0

IP address of the ISP router (the interface connecting to Hanoi network) is fixed as 192.168.N.1

Where N is the index of your lab group. For example, if your lab group is N03 then the IP address range is 192.168.3.0/24

The company asks you to configure the ISP Router to enable the DHCP service. Then, you need to help the company connect Hanoi router to the ISP router and make the necessary adjustments so that all data flows to the Internet from both Hanoi and Saigon networks are directed to the ISP Router.

The work to be done is as follows:

Step 1: Configure and enable the DHCP Server service on the ISP Router so that it allocates IPs in the range specified above. Besides, enable the DHCP client service on the interface of the Hanoi Router connected to the ISP Router

Question 6 (1 pt): Write down the necessary configuration in the file /etc/dhcp/dhcpd.conf

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Write down the command to activate DHCP service on ISP Router

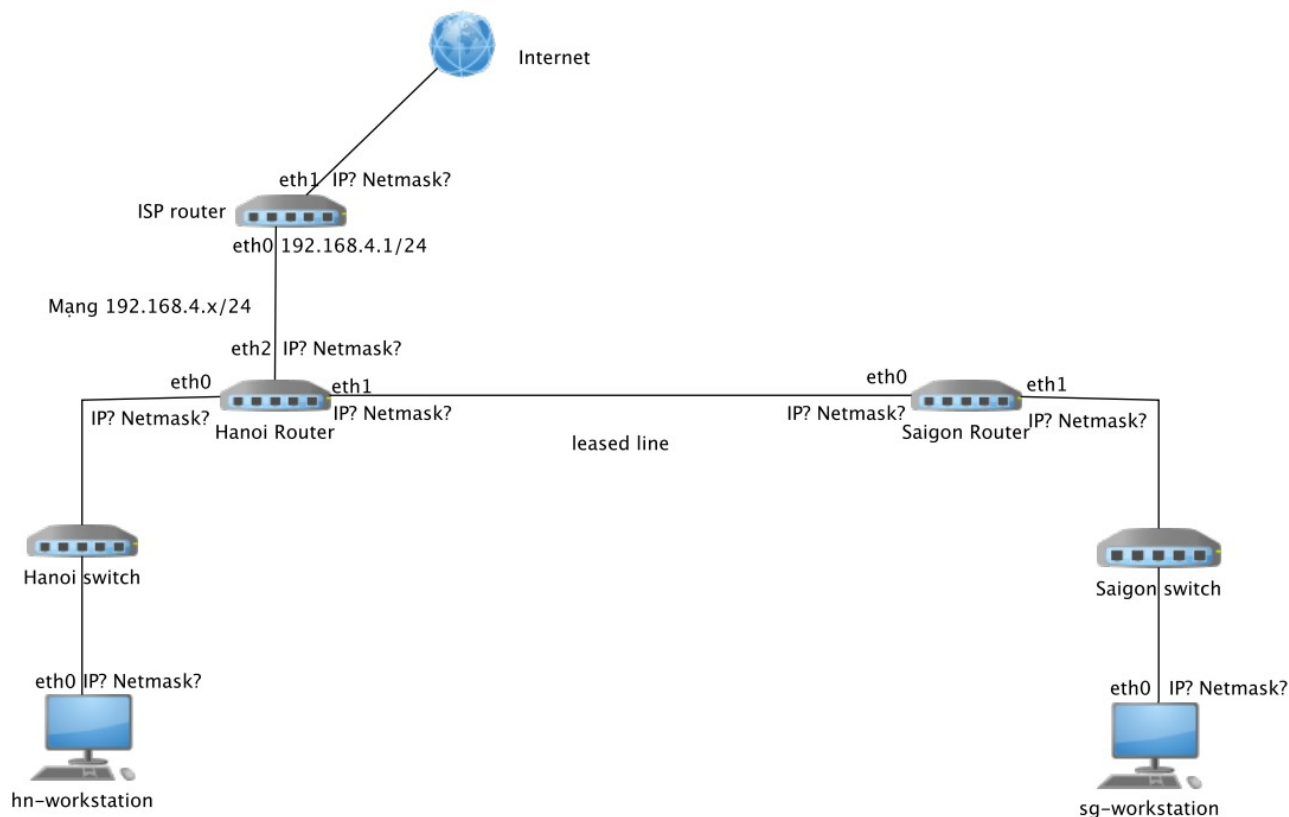


Figure 2: The network with Internet connection at Hanoi

Question 7 (1 pt): Write a command to enable DHCP service on Hanoi Router so that it automatically receives IP from ISP router

.....

What is the IP address that Hanoi Router receives on the eth2 interface

.....

Step 2: Configure Hanoi and Saigon routers

Question 8 (1 pt): What is the IP address that Hanoi Router receives on the eth2 interface

.....

Adjust the routing table of the Hanoi router to forward data that is not directed to the Hanoi and Saigon LANs to the Internet. Students should use the default route

.....

Question 9 (1 pt): Configure ISP router so that it can forward data to Hanoi, Saigon networks.

.....

Step 4 (0.5 pt): Validation (*Demo to teacher/TA*)

Đến lúc này nếu các cấu hình đều đúng thì các máy ở các mạng đã có thể chuyển dữ liệu cho nhau.

Gán địa chỉ IP 100.100.100.1 cho một giao diện ngoài của ISP Router (giao diện không nối với Router Hà nội, eth1 trên hình). Nếu các máy

trong mạng có thể truyền dữ liệu đến giao diện này thì coi như chúng truyền dữ liệu được ngoài Internet.

Sử dụng lệnh `traceroute` để kiểm tra tính thông suốt của các kết nối ra Internet (qua địa chỉ 100.100.100.1) từ các máy trạm tại Hà nội, Sài gòn.

At this point, if the configurations are correct, the machines on the networks can transfer data to each other.

Assign IP address 100.100.100.1 to an external interface of ISP Router (the interface that is not connected to Hanoi Router, `eth1` in the picture). If the computers in the network can transmit data to this interface, then they are able to transmit data outside the Internet.

Use the `traceroute` command to check the connections to the Internet (via address 100.100.100.1) from workstations in Hanoi, Saigon.

1.3. REPORT FOR LAB 3

Student fullname:

StudentID:

Lab Class ID:

Theoretical Class ID:

5.3.1 Connecting two LANs using routers

Question 1 (1 pt): Assign suitable IP address to sg-workstation, hn-workstation and network interfaces of routers. Fill these address into the network topology.

IP of hn-workstation:

Mask:.....

IP of sg-workstation:

Mask:.....

IP of Hanoi-eth0:

Mask:.....

IP of Hanoi-eth1:

Mask:.....

IP of Saigon-eth0:

Mask:.....

IP of Saigon-eth1:

Mask:.....

Fill the address to the bellow Figure.

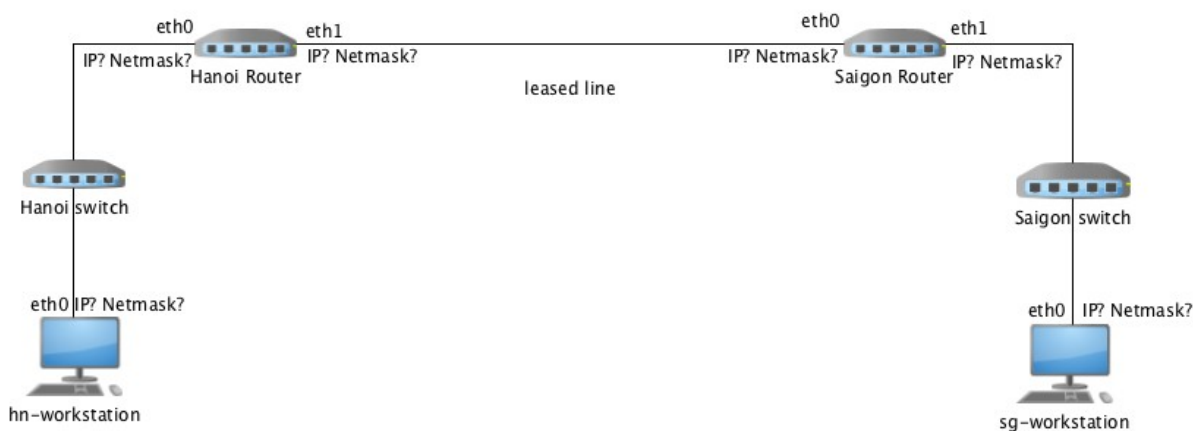


Figure 3: Network topology.

Question 2 (1 pt): Enable the network interfaces of workstations and assign the IP address as specified in Question 1 using the `ifconfig` statement. The commands to use are:

On hn-workstation:

.....

On sg-workstation:

.....

Question 3 (1 pt): Setup routing rules on workstations

Use `route -n` to check the routing table

Use `route add` to add a new route to a distance network.

Write down the command to be executed on **hn-workstation** to forward data to Saigon network:

.....

Write down the command to be executed on **sg-workstation** to forward data to Hanoi network:

.....

Question 4 (1 pt): Execute the command to set the IP address for the interfaces connected to the LAN of the Hanoi router:

.....

And Saigon router:

.....

Execute the command to set IP address for the interfaces connected to the leased line of the Hanoi router:

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....
.....
.....

What is the IP address that Hanoi Router receives on the eth2 interface

.....
.....

Question 8 (1 pt): What is the IP address that Hanoi Router receives on the eth2 interface

.....
.....

.....
.....

Adjust the routing table of the Hanoi router to forward data that is not directed to the Hanoi and Saigon LANs to the Internet. Students should use the default route

.....
.....

.....
.....

Question 9 (1 pt): Configure ISP router so that it can forward data to Hanoi, Saigon networks.

.....
.....

.....
.....

.....
.....

.....
.....

6. LAB 4. UDP AND TCP

6.1. LAB OVERVIEW

6.1.1. Objectives

In this lab, we'll take a quick look at the UDP transport protocol and investigate the behavior of the celebrated TCP protocol in detail.

- Analyze a trace of the UDP diagram sent and received
- Analyze a trace of the TCP segments sent and received
- Consider TCP connection setup and termination
- Study TCP's use of sequence and acknowledgement numbers for providing reliable data transfer

6.1.2. Requirements

- Lab environment:
 - OS: Microsoft Windows
 - Packet Sniffer: Wireshark
- You have to re-read about UDP and TCP protocol in learning materials before doing this lab
- Submission:
 - You need to submit a detailed lab report, with screenshots, to describe what you have done and what you have observed.
 - Store traffic file, called **lab04.pcapng** (maximum size: 2 MB) and report in folder, named **StudentName_ID_Lab04**; compress folder and send to your supervisor.

6.1.3. Brief description about UDP and TCP

6.1.3.1. UDP

The User Datagram Protocol, or UDP, is a communication protocol used across the Internet for especially time-sensitive transmissions such as video playback or DNS lookups. It speeds up communications by not formally establishing a connection before data is transferred. Like all networking protocols, UDP is a standardized method for transferring data between two computers in a network. Compared to other protocols, UDP accomplishes this process in a

simple fashion: it sends packets (units of data transmission) directly to a target computer, without establishing a connection first, indicating the order of said packets, or checking whether they arrived as intended. (UDP packets are referred to as 'datagrams'.)

6.1.3.2. TCP

TCP stands for Transmission Control Protocol. It is a transport layer protocol that facilitates the transmission of packets from source to destination. It is a connection-oriented protocol that means it establishes the connection prior to the communication that occurs between the computing devices in a network. This protocol is used with an IP protocol, so together, they are referred to as a TCP/IP.

The main functionality of the TCP is to take the data from the application layer. Then it divides the data into several packets, provides numbering to these packets, and finally transmits these packets to the destination. The TCP, on the other side, will reassemble the packets and transmits them to the application layer. As we know that TCP is a connection-oriented protocol, so the connection will remain established until the communication is not completed between the sender and the receiver.

The following are the features of a TCP protocol:

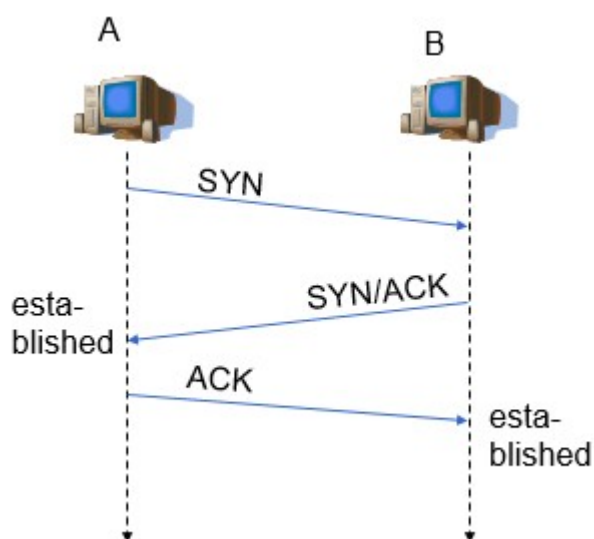
- Transport Layer Protocol: TCP is a transport layer protocol as it is used in transmitting the data from the sender to the receiver.
- Connection-oriented: It is a connection-oriented service that means the data exchange occurs only after the connection establishment. When the data transfer is completed, then the connection will get terminated.
- Reliable: TCP is a reliable protocol as it follows the flow and error control mechanism. It also supports the acknowledgment mechanism, which checks the state and sound arrival of the data. In the acknowledgment mechanism, the receiver sends either positive or negative acknowledgment to the sender so that the sender can get to know whether the data packet has been received or needs to resend.

- Order of the data is maintained: This protocol ensures that the data reaches the intended receiver in the same order in which it is sent. It orders and numbers each segment so that the TCP layer on the destination side can reassemble them based on their ordering.
- Full duplex: It is a full-duplex means that the data can transfer in both directions at the same time.
- Stream-oriented: TCP is a stream-oriented protocol as it allows the sender to send the data in the form of a stream of bytes and also allows the receiver to accept the data in the form of a stream of bytes. TCP creates an environment in which both the sender and receiver are connected by an imaginary tube known as a virtual circuit. This virtual circuit carries the stream of bytes across the internet.

Establishing a TCP connection

Establishing a TCP connection requires that both the computers participate in what is known as a three-way handshake. The process can be broken down as follows:

- Computer A sends the computer B a SYN packet—a connection request from its source port to a B's destination port.
- B responds with a SYN/ACK packet, acknowledging the receipt of the connection request.
- Computer A receives the SYN/ACK packet and responds with an ACK packet of its own.



After the connection is established, TCP works by breaking down transmitted data into segments, each of which is packaged into a datagram and sent to its destination. When a packet of data is sent over TCP, the recipient must always acknowledge what they received. The

first computer sends a packet with data and a sequence number. The second computer acknowledges it by setting the ACK bit and increasing the acknowledgement number by the length of the received data. Those two numbers help the computers to keep track of which data was successfully received, which data was lost, and which data was accidentally sent twice.

Either computer can close the connection when they no longer want to send or receive data. A computer initiates closing the connection by sending a packet with the FIN bit set to 1 (FIN = finish). The other computer replies with an ACK and another FIN. After one more ACK from the initiating computer, the connection is closed.

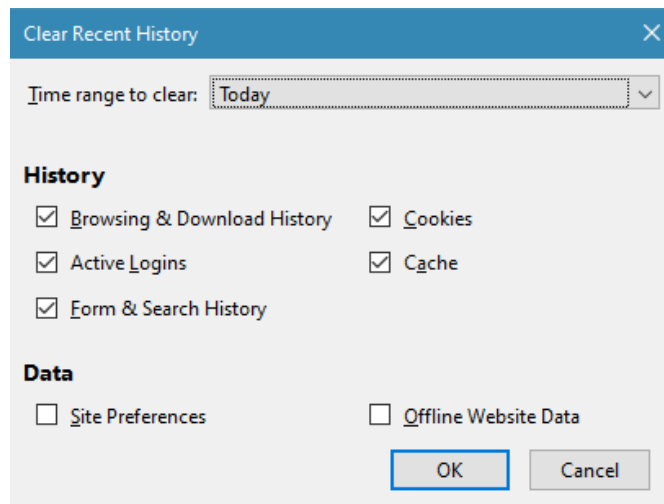
6.2. LAB TASKS

6.2.1. Identify IP address task

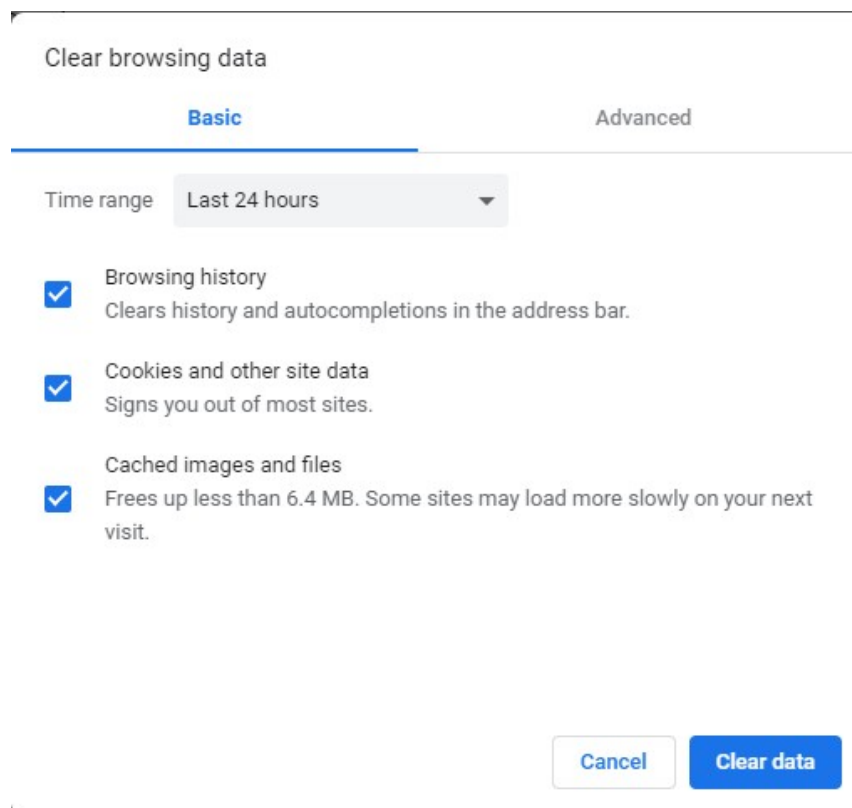
Student identify the IP address of computer and write on report.

6.2.2. Sniff network traffic task

- **Step 1:** Close all of programs, that transfer data on the network, except Web Browser
- **Step 2:** Start up your web browser. Go to the <http://nct.soict.hust.edu.vn/mmt/alice.txt> and retrieve an ASCII copy of "Alice in Wonderland". Store this file somewhere on your computer.
- **Step 3:** Next go to <http://nct.soict.hust.edu.vn/mmt/lab04/>
- **Step 4:** Clear Web cache on browser
 - Mozilla Firefox: Press **Ctrl + Shift + Del**. Select following options and press **OK**.



- Google Chrome: Press **Ctrl + Shift + Del**. Select following options and press **Clear Data**.



- **Step 5:** Open **Command Prompt** utilities on Windows OS, complete command **ipconfig /flushdns**
- **Step 6:** Open Wireshark and start capture network traffic.

- **Step 7:** Go back the Web browser, use the **Browse** button in this form to enter the name of the file (full path name) on your computer containing “Alice in Wonderland”. Press the “**Upload alice.txt file**” button to upload the file to the gaia.cs.umass.edu server.

Upload page for TCP Wireshark Lab

Computer Networking: A Top Down Approach, 6th edition

Copyright 2012 J.F. Kurose and K.W. Ross, All Rights Reserved

If you have followed the instructions for the TCP Wireshark Lab, you have *already* downloaded an ASCII copy of alice from <http://nct.soict.hust.edu.vn/mmt/alice.txt> and you also *already* have the Wireshark packet sniffer running and capturing packets on your computer.

Click on the Browse button below to select the directory/file name for the copy of alice.txt that is stored on your computer.

Browse... alice.txt **1**

Once you have selected the file, click on the "Upload alice.txt file" button below. This will cause your browser to send a copy of alice.txt over an HTTP connection (using TCP) to the web server at nct.soict.hust.edu.vn. After clicking on the button, wait until a short message is displayed indicating the the upload is complete. Then stop your Wireshark packet sniffer - you're ready to begin analyzing the TCP transfer of alice.txt from your computer to nct.soict.hust.edu.vn!!

Upload alice.txt file **2**

- **Bước 8:** Once the file has been uploaded, a short congratulations message will be displayed in your browser window. Stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.

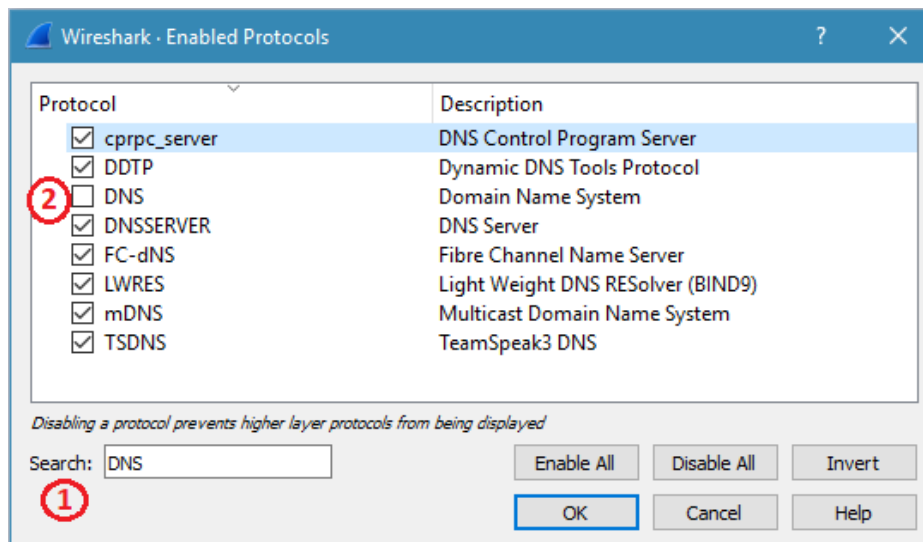
| | | | | | |
|----|----------|---------------|---------------|-----|---|
| 2 | 0.839954 | 192.168.1.176 | 8.8.8.8 | DNS | 81 Standard query 0x1c59 A nct.soict.hust.edu.vn |
| 3 | 0.906865 | 8.8.8.8 | 192.168.1.176 | DNS | 97 Standard query response 0x1c59 A nct.soict.hust.edu.vn A 202.191.56.66 |
| 4 | 0.908215 | 192.168.1.176 | 202.191.56.66 | TCP | 66 5729 → 80 [SYN] Seq=2221575575 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 |
| 5 | 0.908536 | 192.168.1.176 | 8.8.8.8 | DNS | 81 Standard query 0x6d03 A nct.soict.hust.edu.vn |
| 6 | 0.910222 | 202.191.56.66 | 192.168.1.176 | TCP | 66 80 → 5729 [SYN, ACK] Seq=943466327 Ack=2221575576 Win=29200 Len=0 MSS=1460 |
| 7 | 0.910378 | 192.168.1.176 | 202.191.56.66 | TCP | 54 5729 → 80 [ACK] Seq=2221575576 Ack=943466328 Win=131328 Len=0 |
| 8 | 0.913761 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 5729 → 80 [ACK] Seq=2221575576 Ack=943466328 Win=131328 Len=1460 |
| 9 | 0.913761 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 5729 → 80 [ACK] Seq=2221577036 Ack=943466328 Win=131328 Len=1460 |
| 10 | 0.913764 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 5729 → 80 [ACK] Seq=2221578496 Ack=943466328 Win=131328 Len=1460 |
| 11 | 0.913764 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 5729 → 80 [ACK] Seq=2221579956 Ack=943466328 Win=131328 Len=1460 |
| 12 | 0.913764 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 5729 → 80 [ACK] Seq=2221581416 Ack=943466328 Win=131328 Len=1460 |
| 13 | 0.913765 | 192.168.1.176 | 202.191.56.66 | TCP | 946 5729 → 80 [PSH, ACK] Seq=2221582876 Ack=943466328 Win=131328 Len=892 |
| 14 | 0.913936 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 5729 → 80 [ACK] Seq=2221583768 Ack=943466328 Win=131328 Len=1460 |

Notice: If you don't find DNS packet, please restart from step 3.

- **Bước 9:** Save file as **lab04.pcapng**

6.2.3. Analyze UDP datagrams task

- **Step 1:** On Wireshark menu, select **Analyze->Enabled Protocols**. Then uncheck the **DNS** box and select **OK**.



- **Step 2:** Set your packet filter as **udp** value so that Wireshark only displays the UDP packets sent and received at your host.

| udp | | | | | | | |
|-----|----------|---------------|---------------|----------|--------|--------------------|--|
| No. | Time | Source | Destination | Protocol | Length | Info | |
| 2 | 0.839954 | 192.168.1.176 | 8.8.8.8 | UDP | 81 | 55309 → 53 Len=39 | |
| 3 | 0.906865 | 8.8.8.8 | 192.168.1.176 | UDP | 97 | 53 → 55309 Len=55 | |
| 5 | 0.908536 | 192.168.1.176 | 8.8.8.8 | UDP | 81 | 54722 → 53 Len=39 | |
| 92 | 0.956218 | 8.8.8.8 | 192.168.1.176 | UDP | 97 | 53 → 54722 Len=55 | |
| 139 | 0.957318 | 192.168.1.176 | 8.8.8.8 | UDP | 81 | 52525 → 53 Len=39 | |
| 163 | 1.040953 | 8.8.8.8 | 192.168.1.176 | UDP | 132 | 53 → 52525 Len=90 | |
| 169 | 3.993781 | 192.168.1.144 | 192.168.1.255 | UDP | 85 | 5050 → 5050 Len=43 | |

- **Step 3:** Pick one of these UDP packets that are sent from your computer and expand the headers in the details window.

Question 1(1 point): Determine the value of following fields

No.:.....

Source IP address:.....Destination IP address:
.....

Source port:.....Destination port:
.....

What is network layer protocol?.....

- **Step 4:** Examine a pair of the above UDP packet in which your host sent and answer question 2.

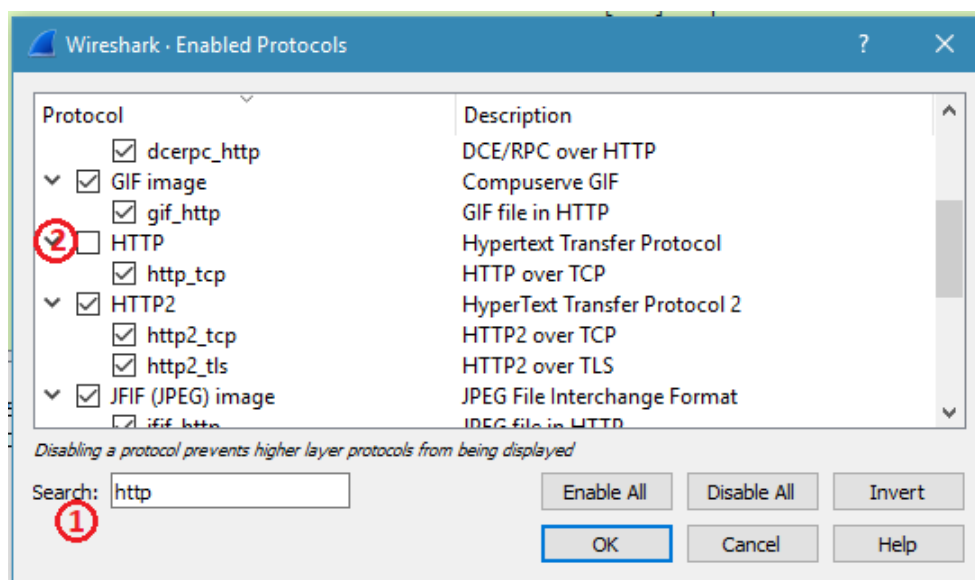
Question 2(1 point):

What is No. of packet?.....

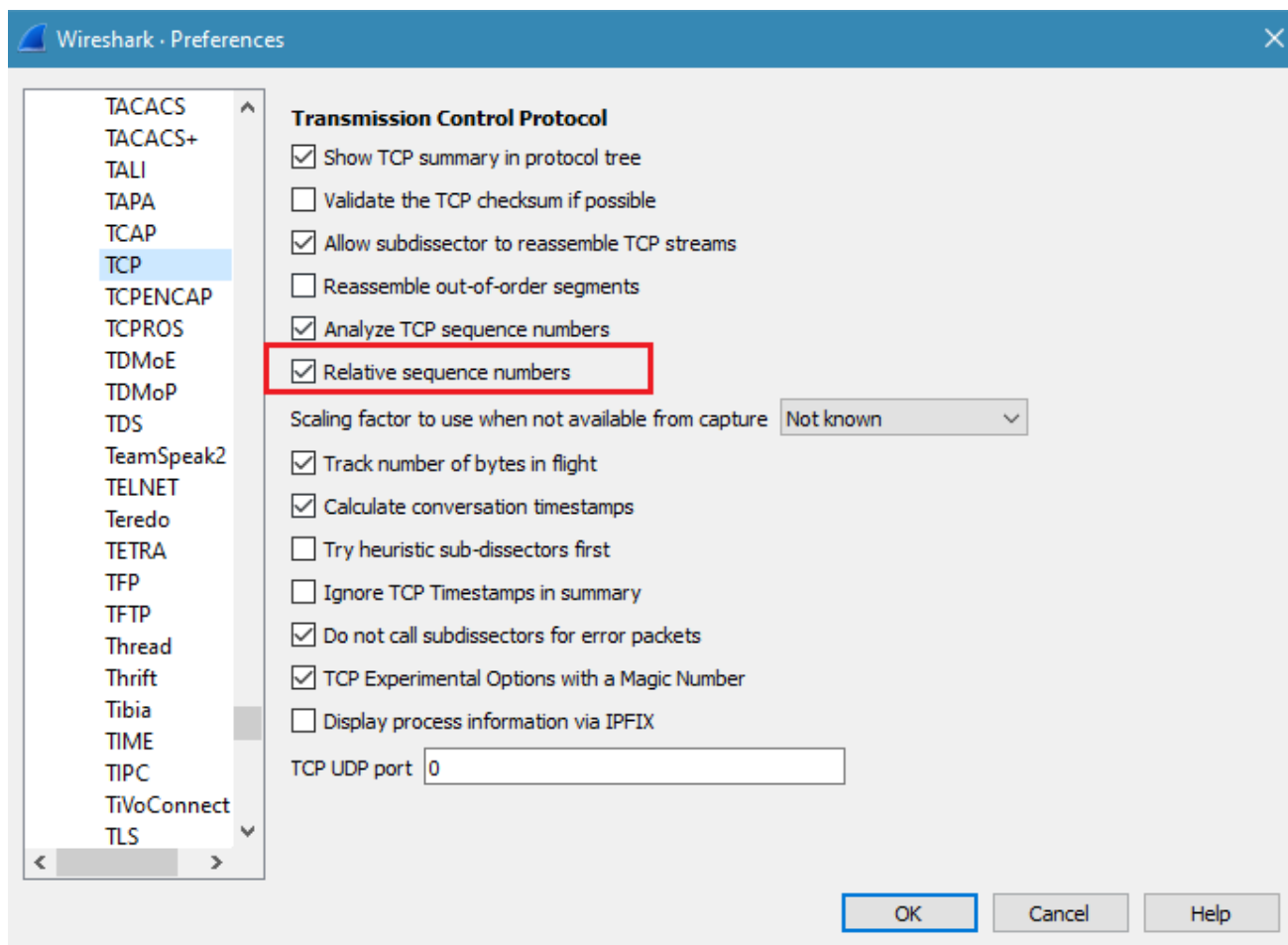
Describe the relationship between the port numbers in the two packets.
Can receiver know whether transmission is successful or not?

6.2.4. Analyze TCP segments task

- **Bước 1:** On Wireshark menu, select **Analyze->Enabled Protocols**. Then uncheck the **HTTP** box and select **OK**.



- **Step 2:** On Wireshark menu, select **Edit → Preferences...** Select **Protocol→TCP** on and check **Relative sequence numbers**.



- **Step 3:** Set your packet filter as the following value so that Wireshark only displays the TCP packets sent and received in upload file process.

tcp && ip.addr == 202.191.56.66

Your Wireshark window should look similar to the window shown below:

| tcp && ip.addr == 202.191.56.66 | | | | | | |
|---------------------------------|----------|---------------|---------------|----------|--------|---|
| No. | Time | Source | Destination | Protocol | Length | Info |
| 4 | 0.908215 | 192.168.1.176 | 202.191.56.66 | TCP | 66 | 5729 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 S |
| 6 | 0.910222 | 202.191.56.66 | 192.168.1.176 | TCP | 66 | 80 → 5729 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=14 |
| 7 | 0.910378 | 192.168.1.176 | 202.191.56.66 | TCP | 54 | 5729 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0 |
| 8 | 0.913761 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=1460 |
| 9 | 0.913761 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=1461 Ack=1 Win=131328 Len=1460 |
| 10 | 0.913764 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=2921 Ack=1 Win=131328 Len=1460 |
| 11 | 0.913764 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=4381 Ack=1 Win=131328 Len=1460 |
| 12 | 0.913764 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=5841 Ack=1 Win=131328 Len=1460 |
| 13 | 0.913765 | 192.168.1.176 | 202.191.56.66 | TCP | 946 | 5729 → 80 [PSH, ACK] Seq=7301 Ack=1 Win=131328 Len=892 |
| 14 | 0.913936 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=8193 Ack=1 Win=131328 Len=1460 |
| 15 | 0.913937 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=9653 Ack=1 Win=131328 Len=1460 |
| 16 | 0.913937 | 192.168.1.176 | 202.191.56.66 | TCP | 1514 | 5729 → 80 [ACK] Seq=11113 Ack=1 Win=131328 Len=1460 |

- **Step 3:** Observe the initial three-way handshake establishing the TCP connection between your client and server and answer question 3.

Question 3(2 point): What is the IP address and TCP port number used by parties?

- *The IP address of client:*
- *The IP address of server:*
- *The port number of client:*
- *The port number of server:*

Expand the TCP header of messages in initiation the TCP connection and determine the value of fields:

| No. | Flags (binary value) | TCP Flags | Sequence number | ACK number | Payload size |
|-----|-------------------------|-----------|-----------------|------------|--------------|
| | | | | | |

- **Step 4:** Find the segment containing the first bytes of alice.txt file (Hint: You should read the content of segment payload)

Question 4(1 point): Expand the headers of the segment and determine

- *No.:*

- *Source IP address:*
- *Destination IP address:*
- *Source port number:*
- *Destination port number:*
- *Sequence Number:*
- *ACK Number:*
- *TCP header size:*
- *Payload size:*
- *What are TCP flags set?*

- **Step 5:** Find the ACK segment for the above segment

Question 5(1 point): Expand the headers of the segment and determine

- *No.:*
- *Source IP address:*
- *Destination IP address:*
- *Source port number:*
- *Destination port number:*
- *Sequence Number:*
- *ACK Number:*
- *TCP header size:*
- *Payload size:*
- *What are TCP flags set?*

Question 6(1 point): What is the sequence number of the next TCP segment sent from Web browser on your computer?

- **Bước 6:** Observe the termination of the TCP connection.

Question 7(2 point): Expand the TCP header of messages in termination of the TCP connection and determine the value of fields:

| <i>No.</i> | <i>Flags (binary value)</i> | <i>TCP Flags</i> | <i>Sequence number</i> | <i>ACK number</i> | <i>Payload size</i> |
|------------|-------------------------------------|------------------|----------------------------|-----------------------|-------------------------|
| | | | | | |

| | | | | | | |
|--|--|--|--|--|--|--|
| | | | | | | |
|--|--|--|--|--|--|--|

Question 8(1 point): What is the throughput (bytes transferred per unit time) for the TCP connection during uploading file? Explain how you calculated this value.

7. LAB 5: DNS AND HTTP

7.1. LAB OVERVIEW

7.1.1. Objectives

In this lab, we'll take a quick look at the DNS application protocol and explore several aspects of the HTTP protocol:

- Analyze a trace of the DNS messages sent and received
- HTTP message format
- The basic HTTP GET/response interaction

7.1.2. Yêu cầu đối với sinh viên

- Lab environment:
 - OS: Microsoft Windows
 - Packet Sniffer: Wireshark
- You have to re-read about DNS and HTTP protocol in learning materials before doing this lab.
- Submission:
 - You need to submit a detailed lab report, with screenshots, to describe what you have done and what you have observed.
 - Store traffic file, called **lab05.pcapng** (maximum size: 2 MB) and report in folder, named **StudentName_ID_Lab05**; compress folder and send to your supervisor.

7.1.3. Brief description about DNS and HTTP

7.1.3.1. Domain name and DNS

The Domain Name System (DNS) is the phonebook of the Internet. Humans access information online through domain names, like `soict.hust.edu.vn`. Web browsers interact through Internet Protocol (IP) addresses. DNS translates domain names to IP addresses so browsers can load Internet resources.

Each device connected to the Internet has a unique IP address which other machines use to find the device. DNS servers eliminate the need for humans to memorize IP addresses such as 192.168.1.1 (in

IPv4), or more complex newer alphanumeric IP addresses such as 2400:cb00:2048:1::c629:d7a2 (in IPv6).

The process of DNS resolution involves converting a hostname (such as soict.hust.edu.vn) into a computer-friendly IP address (such as 202.191.56.65). An IP address is given to each device on the Internet, and that address is necessary to find the appropriate Internet device - like a street address is used to find a particular home. When a user wants to load a webpage, a translation must occur between what a user types into their web browser (soict.hust.edu.vn) and the machine-friendly address necessary to locate the soict.hust.edu.vn webpage.

In order to understand the process behind the DNS resolution, it's important to learn about the different hardware components a DNS query must pass between. For the web browser, the DNS lookup occurs "behind the scenes" and requires no interaction from the user's computer apart from the initial request.

7.1.3.2. HTTP

HTTP (HyperText Transfer Protocol) is the underlying protocol of the World Wide Web. It was designed for communication between web browsers and web servers, but it can also be used for other purposes. HTTP follows a classical client-server model, with a client opening a connection to make a request, then waiting until it receives a response. HTTP is a stateless protocol, meaning that the server does not keep any data (state) between two requests.

Developed by Tim Berners-Lee and his team between 1989-1991, HTTP has gone through many changes that have helped maintain its simplicity while shaping its flexibility.

HTTP/0.9 - The one-line protocol

The initial version of HTTP had no version number; it was later called 0.9 to differentiate it from later versions. HTTP/0.9 was extremely simple: requests consisted of a single line and started with the only possible method GET followed by the path to the resource. The full URL wasn't included as the protocol, server, and port weren't necessary once connected to the server. Unlike subsequent evolutions, there were

no HTTP headers. This meant that only HTML files could be transmitted. There were no status or error codes. If there was a problem, a specific HTML file was generated and included a description of the problem for human consumption.

HTTP/1.0 - Building extensibility

HTTP/0.9 was very limited, but browsers and servers quickly made it more versatile:

- Versioning information was sent within each request (HTTP/1.0 was appended to the GET line).
- A status code line was also sent at the beginning of a response. This allowed the browser itself to recognize the success or failure of a request and adapt its behavior accordingly. For example, updating or using its local cache in a specific way.
- The concept of HTTP headers was introduced for both requests and responses. Metadata could be transmitted and the protocol became extremely flexible and extensible.
- Documents other than plain HTML files could be transmitted thanks to the Content-Type header.

Between 1991-1995, these were introduced with a try-and-see approach. A server and a browser would add a feature and see if it got traction. Interoperability problems were common. In an effort to solve these issues, an informational document that described the common practices was published in November 1996. This was known as RFC 1945 and defined HTTP/1.0.

HTTP/1.1 - The standardized protocol

In the meantime, proper standardization was in progress. This happened in parallel to the diverse implementations of HTTP/1.0. The first standardized version of HTTP, HTTP/1.1, was published in early 1997, only a few months after HTTP/1.0.

HTTP/1.1 clarified ambiguities and introduced numerous improvements:

- A connection could be reused, which saved time. It no longer needed to be opened multiple times to display the resources embedded in the single original document.
- Pipelining was added. This allowed a second request to be sent before the answer to the first one was fully transmitted. This lowered the latency of the communication.
- Chunked responses were also supported.
- Additional cache control mechanisms were introduced.
- Content negotiation, including language, encoding, and type, was introduced. A client and a server could now agree on which content to exchange.
- Thanks to the Host header, the ability to host different domains from the same IP address allowed server collocation.

HTTP/1.1 was first published as RFC 2068 in January 1997.

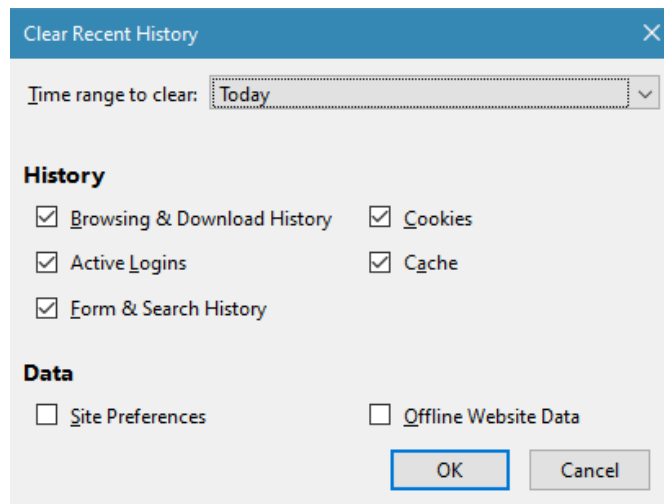
7.2. LAB TASKS

7.2.1. Identify IP address task

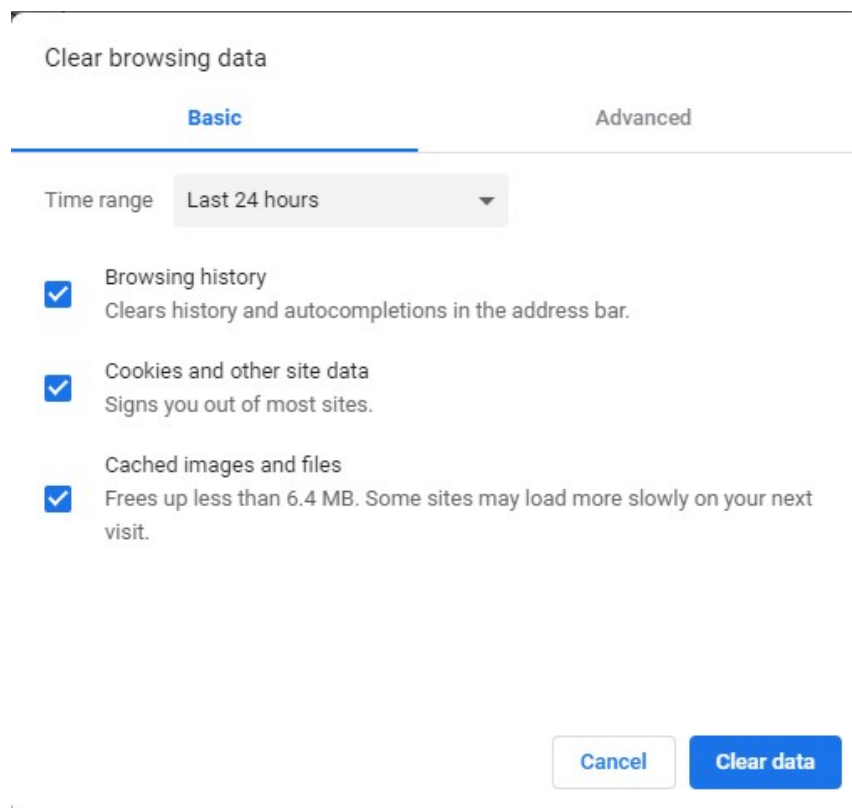
Student identify the IP address of computer and write on report.

7.2.2. Sniff network traffic task

- **Step 1:** Close all of programs, that transfer data on the network, except Web Browser.
- **Bước 2:** Clear Web cache on browser
 - Mozilla Firefox: Press **Ctrl + Shift + Del**. Select following options and press **OK**.



- Google Chrome: Press **Ctrl + Shift + Del**. Select following options and press **Clear Data**.



- **Step 3:** Open **Command Prompt** utilities on Windows OS, complete command **ipconfig /flushdns**
- **Step 4:** Open Wireshark and start capture network traffic.
- **Step 5:** Go back the Web browser, open Private Browsing tab

- **Mozilla Firefox:** Press Ctrl + Shift + P
- **Google Chrome:** Press Ctrl + Shift + N

Go to <http://nct.soict.hust.edu.vn/mmt/lab05/>

- **Step 6:** After browser display webpage, stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|---------------|---------------|----------|--------|--|
| 25 | 4.682835 | 192.168.1.176 | 8.8.8.8 | DNS | 81 | Standard query 0x54e1 A nct.soict.hust.edu.vn |
| 26 | 4.751908 | 8.8.8.8 | 192.168.1.176 | DNS | 97 | Standard query response 0x54e1 A nct.soict.hust.edu.vn A 202.191.56.66 |
| 27 | 4.753094 | 192.168.1.176 | 202.191.56.66 | TCP | 66 | 7253 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 |
| 28 | 4.753348 | 192.168.1.176 | 8.8.8.8 | DNS | 81 | Standard query 0xc89a A nct.soict.hust.edu.vn |
| 29 | 4.755770 | 202.191.56.66 | 192.168.1.176 | TCP | 66 | 80 → 7253 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128 |
| 30 | 4.755870 | 192.168.1.176 | 202.191.56.66 | TCP | 54 | 7253 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0 |
| 31 | 4.757913 | 192.168.1.176 | 202.191.56.66 | HTTP | 414 | GET /mmt/lab05/ HTTP/1.1 |
| 32 | 4.760169 | 202.191.56.66 | 192.168.1.176 | TCP | 60 | 80 → 7253 [ACK] Seq=1 Ack=361 Win=30336 Len=0 |
| 33 | 4.761982 | 202.191.56.66 | 192.168.1.176 | TCP | 1514 | 80 → 7253 [ACK] Seq=1 Ack=361 Win=30336 Len=1460 [TCP segment of a reassembled PDU] |
| 34 | 4.763646 | 202.191.56.66 | 192.168.1.176 | TCP | 1514 | 80 → 7253 [ACK] Seq=1461 Ack=361 Win=30336 Len=1460 [TCP segment of a reassembled PDU] |
| 35 | 4.763687 | 192.168.1.176 | 202.191.56.66 | TCP | 54 | 7253 → 80 [ACK] Seq=361 Ack=2921 Win=131328 Len=0 |
| 36 | 4.764615 | 202.191.56.66 | 192.168.1.176 | TCP | 1514 | 80 → 7253 [ACK] Seq=2921 Ack=361 Win=30336 Len=1460 [TCP segment of a reassembled PDU] |
| 37 | 4.765585 | 202.191.56.66 | 192.168.1.176 | TCP | 1514 | 80 → 7253 [ACK] Seq=4381 Ack=361 Win=30336 Len=1460 [TCP segment of a reassembled PDU] |
| 38 | 4.765622 | 192.168.1.176 | 202.191.56.66 | TCP | 54 | 7253 → 80 [ACK] Seq=361 Ack=5841 Win=131328 Len=0 |
| 39 | 4.766565 | 202.191.56.66 | 192.168.1.176 | TCP | 1514 | 80 → 7253 [ACK] Seq=5841 Ack=361 Win=30336 Len=1460 [TCP segment of a reassembled PDU] |
| 40 | 4.766913 | 202.191.56.66 | 192.168.1.176 | TCP | 1514 | 80 → 7253 [ACK] Seq=7301 Ack=361 Win=30336 Len=1460 [TCP segment of a reassembled PDU] |

Notice:

- On Wireshark menu, select **Analyze->Enabled Protocols**. Then check the **DNS** box, **HTTP** box and select **OK**.
- If you don't find DNS packet, please restart from step 2.
- **Bước 7:** Save file as **lab05.pcapng**

7.2.3. Tracing DNS task

- **Step 1:** Set your packet filter as **dns** value so that Wireshark only displays the DNS messages.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|---------------|---------------|----------|--------|--|
| 25 | 4.682835 | 192.168.1.176 | 8.8.8.8 | DNS | 81 | Standard query 0x54e1 A nct.soict.hust.edu.vn |
| 26 | 4.751908 | 8.8.8.8 | 192.168.1.176 | DNS | 97 | Standard query response 0x54e1 A nct.soict.hust.edu.vn A 202.191.56.66 |
| 28 | 4.753348 | 192.168.1.176 | 8.8.8.8 | DNS | 81 | Standard query 0xc89a A nct.soict.hust.edu.vn |
| 57 | 4.808481 | 192.168.1.176 | 8.8.8.8 | DNS | 92 | Standard query 0x175e AAAA ff.kis.v2.scr.kaspersky-labs.com |
| 60 | 4.810552 | 192.168.1.176 | 8.8.8.8 | DNS | 84 | Standard query 0x35f4 A www.lingosolutions.co.uk |
| 95 | 4.830542 | 8.8.8.8 | 192.168.1.176 | DNS | 97 | Standard query response 0xc89a A nct.soict.hust.edu.vn A 202.191.56.66 |
| 96 | 4.831702 | 192.168.1.176 | 8.8.8.8 | DNS | 81 | Standard query 0x3221 AAAA nct.soict.hust.edu.vn |
| 100 | 4.832592 | 8.8.8.8 | 192.168.1.176 | DNS | 174 | Standard query response 0x175e AAAA ff.kis.v2.scr.kaspersky-labs.com SOA d |
| 276 | 4.937883 | 8.8.8.8 | 192.168.1.176 | DNS | 132 | Standard query response 0x3221 AAAA nct.soict.hust.edu.vn SOA dns.hust.edu |
| 280 | 5.131132 | 8.8.8.8 | 192.168.1.176 | DNS | 100 | Standard query response 0x35f4 A www.lingosolutions.co.uk A 149.255.58.41 |
| 282 | 5.133981 | 192.168.1.176 | 8.8.8.8 | DNS | 84 | Standard query 0x8645 A www.lingosolutions.co.uk |
| 286 | 5.425351 | 8.8.8.8 | 192.168.1.176 | DNS | 100 | Standard query response 0x8645 A www.lingosolutions.co.uk A 149.255.58.41 |
| 287 | 5.427220 | 192.168.1.176 | 8.8.8.8 | DNS | 84 | Standard query 0xe374 AAAA www.lingosolutions.co.uk |
| 305 | 5.731849 | 8.8.8.8 | 192.168.1.176 | DNS | 147 | Standard query response 0xe374 AAAA www.lingosolutions.co.uk SOA ns1.unlim |

- **Step 2:** Locate the DNS query message that Web browser send to DNS server for resolving nct.soict.hust.edu.vn.

Question 1(1 point): Observe the packet detail that you need to answer

- No.:
- Are then sent over UDP or TCP?
- Source IP address:
- Destination IP address:
- Source port number:
- Destination port number:
- What “Type” of DNS query is it?

- **Step 3:** Locate and examine the DNS response message

Question 2(1 point): Observe the packet detail that you need to answer

- What field indicates that this message is the response of the above query message?
- No.:
- Are then sent over UDP or TCP?
- Source IP address:
- Destination IP address:
- Source port number:
- Destination port number:
- What “Type” of DNS query is it?
- What is queried domain name?
- How many “answers” are provided? What do each of these answers contain?

- **Step 4:** Observe all of DNS messages and answer question 3.

Question 3(1 point): Why web browser send other DNS queries when you go to <http://nct.soict.hust.edu.vn/mmt/lab05>? What are other domain names and their IP addresses?

7.2.4. Tracing HTTP task

- **Step 1:** Suppose you know that the IP mapping nct.soict.hust.edu.vn is **X**. Set your packet filter as **ip.addr == X** (e.g. if X is 192.168.1.1 then filter is ip.addr == 192.168.1.1) so that Wireshark only displays the packets transferred between your computer and Web server.

Question 4(1 điểm): Observe the set up TCP connection between your computer and the server before transferring HTTP messages.

- What are the No. of TCP packets?
- What is the port number of client?
- What is the port number of server?

- **Step 2:** Set your packet filter as **http** so that Wireshark only displays the HTTP messages.
- **Step 3:** Examine the messages transferred between your computer and the server.

Question 5(2 point): How many HTTP GET request messages did your browser send? Which packet number in the trace contains the status code and phrase associated with the response to the HTTP GET request?

| HTTP Request | | | HTTP Response | | |
|--------------|----------------|---------------------------|---------------|---------------|--------------------------|
| No. | Request method | What object is requested? | No. | Response code | Meaning of response code |
| | | | | | |

What HTTP request messages can be made back-to-back, without waiting for replies to pending requests?

- **Step 4:** Pick the first HTTP Request sent to nct.hust.edu.vn and answer question 6

Question 6(1 point): Observe the packet detail that you need to answer:

- What is transport protocol?

- *What is destination port number?*
- *What version of HTTP is your browser running?*
- *What is the value of **Connection** header?*

- **Step 5:** Find the HTTP Response messages

Câu hỏi 7(1 điểm): Observe the packet detail that you need to answer:

- *What version of HTTP is the server running?*
- *What is the value of **Connection** header?*
- *What is content of the HTTP message body? What is size in bytes?*
- *How many data-containing TCP segments were needed to carry the single HTTP response?*

After your browser received this response, is TCP connection persistent?

Question 8(1 point): What are other Web server that your browser also sent HTTP messages? Why did your browser send request to them? Expand these HTTP request messages and show the value of **Referer** header.

Question 9(1 point): The following paragraph demonstrate the access to a website. Fill the blanks.

When user go to a website, Web browser sends.....message to..... if it doesn't know the IP address of Web server. The browser will identify the IP address of the server in the.....message. After that, Web browser and server setup..... connection. On established connection, browser sends.....message to obtain objects of the webpage. Web server finds the objects and replies by.....messages with code.....if found, or code..... if not found. If two parties run HTTP version....., the connection is persistent.

