Computer Networks
CL3001

LAB - 04
HTTP/HTTPS, DNS, Wireshark

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES,
KARACHI CAMPUS
FAST SCHOOL OF COMPUTING (AI & DS, CS, CY, SE)
SPRING 2025

Computer Networks Lab 04

Course: Computer Networks (CL3001) **Semester:** Spring 2025
Instructor: Sameer Faisal **T.A:** N/A

Note:

- Maintain discipline during the lab.
- Listen and follow the instructions as they are given.
- Just raise hand if you have any problem.
- Completing all tasks of each lab is compulsory.
- Get your lab checked at the end of the session.

Lab Objective

- Implementation & understanding of HTTP/HTTPS.
- Introduction to DNS & configuration of DNS in Cisco Packet Tracer.

HTTP/HTTPS

1. Hypertext Transfer Protocol (HTTP)

Hypertext Transfer Protocol (HTTP) is a protocol used in networking. When you type any web address in your web browser, your browser acts as a client, and the computer having the requested information acts as a server. When client requests for any information from the server, it uses HTTP protocol to do so. The server responds back to the client after the request completes. The response comes in the form of web page which you see just after typing the web address and press "Enter".

2. Hypertext Transfer Protocol Secure (HTTPS)

Hypertext Transfer Protocol Secure (HTTPS) is a combination of two different protocols. It is more secure way to access the web. It is combination of Hypertext Transfer Protocol (HTTPS) and SSL/TLS protocol. It is more secure way to sending request to server from a client, also the communication is purely encrypted which means no one can know what you are looking for. This kind of communication is used for accessing those websites where security is required. Banking websites, payment gateway, emails (Gmail offers HTTPS by default in Chrome browser), and corporate sector websites are some great examples where HTTPS protocols are used.

For HTTPS connection, public key trusted and signed certificate is required for the server. These certificates come either free or it costs few dollars depends on the signing authority. There is one other method for distributing certificates. Site admin creates certificates and loads in the browser of users. Now when user requests information to the web server, his identity can be verified easily.

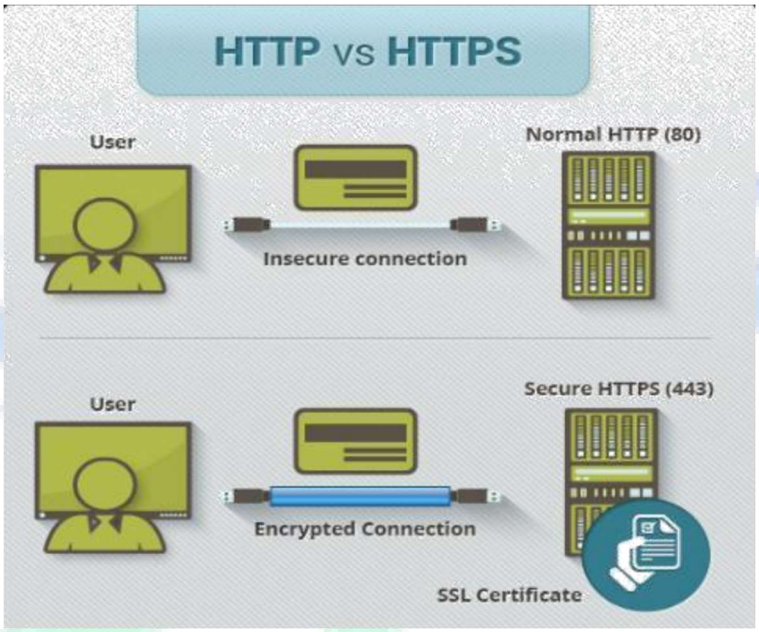

Figure 1

3. Difference between HTTP & HTTPS

Below given are some major differences between HTTP & HTTPS:

HTTP	HTTPS
URL begins with "http://"	URL begins with "https://"
It uses port 80 for communication.	It uses port 443 for communication.
Unsecured.	Secured.
Operates at Application Layer.	Operates at Transport Layer.
No encryption.	Encryption is present.
No certificates required.	Certificates are required.

4a. Client Error

The 4xx class of status code is intended for cases in which the client seems to have erred. Except when responding to a HEAD request, the server should include an entity containing an explanation of the error situation, and whether it is a temporary or permanent condition. These status codes are applicable to any request method. User agents should display any included entity to the user.

400 Bad Request

The server cannot or will not process the request due to something that is perceived to be a client error (e.g., malformed request syntax, invalid request message framing, or deceptive request routing).

401 Unauthorized (RFC 7235)

Similar to 403 Forbidden, but specifically for use when authentication is required and has failed or has not yet been provided. The response must include a WWW -Authenticate header field containing a challenge applicable to the requested resource. See Basic access authentication and Digest access authentication.

403 Forbidden
The request was a valid request, but the server is refusing to respond to it. Unlike a 401 unauthorized response, authenticating will make no difference.

404 Not Found
The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.

408 Request Timeout
The server timed out waiting for the request. According to HTTP specifications: "The client did not produce a request within the time that the server was prepared to wait. The client may repeat the request without modifications at any later time."

4b. Server Error
The server failed to fulfill an apparently valid request.
Response status codes beginning with the digit "5" indicate cases in which the server is aware that it has encountered an error or is otherwise incapable of performing the request. Except when responding to a HEAD request, the server should include an entity containing an explanation of the error situation, and indicate whether it is a temporary or permanent condition. Likewise, user agents should display any included entity to the user. These response codes are applicable to any request method.

500 Internal Server Error
A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.

501 Not Implemented
The server either does not recognize the request method, or it lacks the ability to fulfil the request. Usually this implies future availability (e.g., a new feature of a web-service API).

502 Bad Gateway
The server was acting as a gateway or proxy and received an invalid response from the upstream server.

503 Service Unavailable
The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.

5a. Implementation for HTTP

Design the given topology shown below. Assign IP address to PC using static through as done in pervious lab.

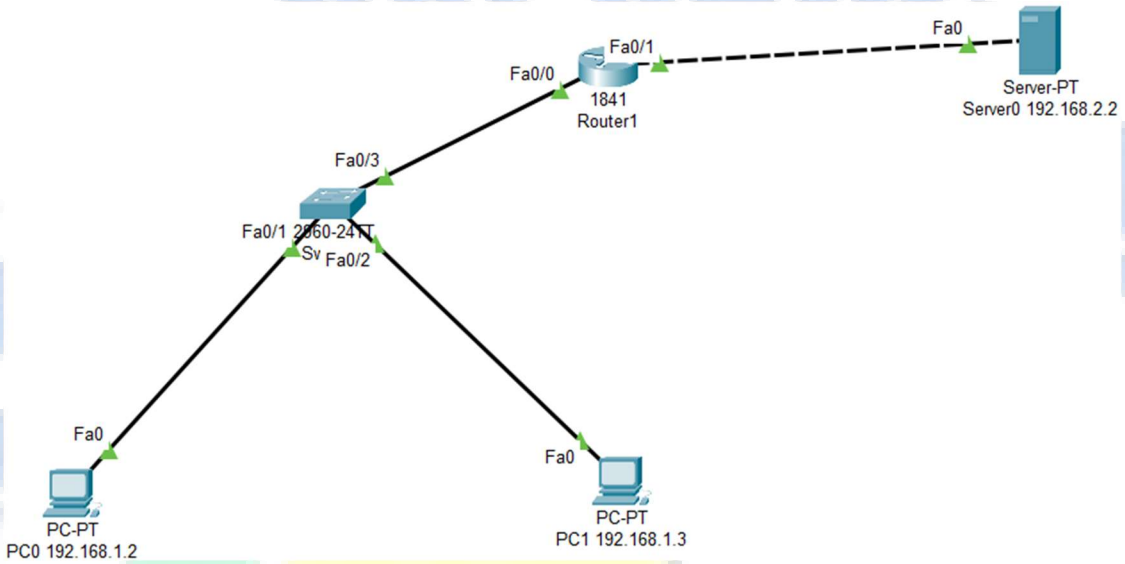

Figure 2

The above topology configured as “one server room”, “one IT room: and “Lab#01 environment having two systems”. On our server we have enabled web services as well as DNS services.

Click on the web server, go to services tab—HTTP

Here you can see HTTP & HTTPS services are on (turn them on if they are previously turned off).

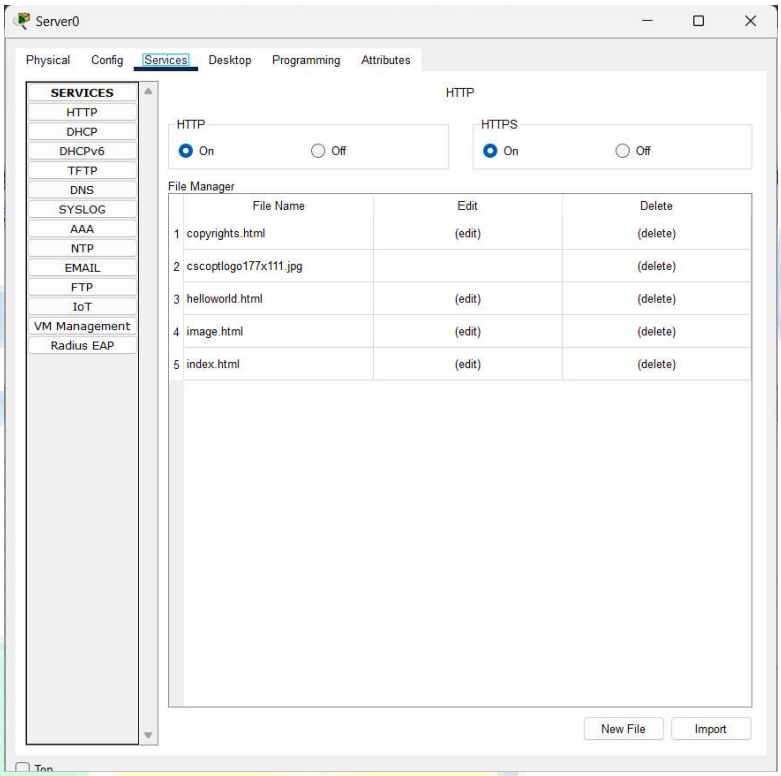

Figure 3

Now click on PC0 and go to Desktop -> Web Browser. Now type web-server IP which you have assign or the website name which you have store in the DNS server record.

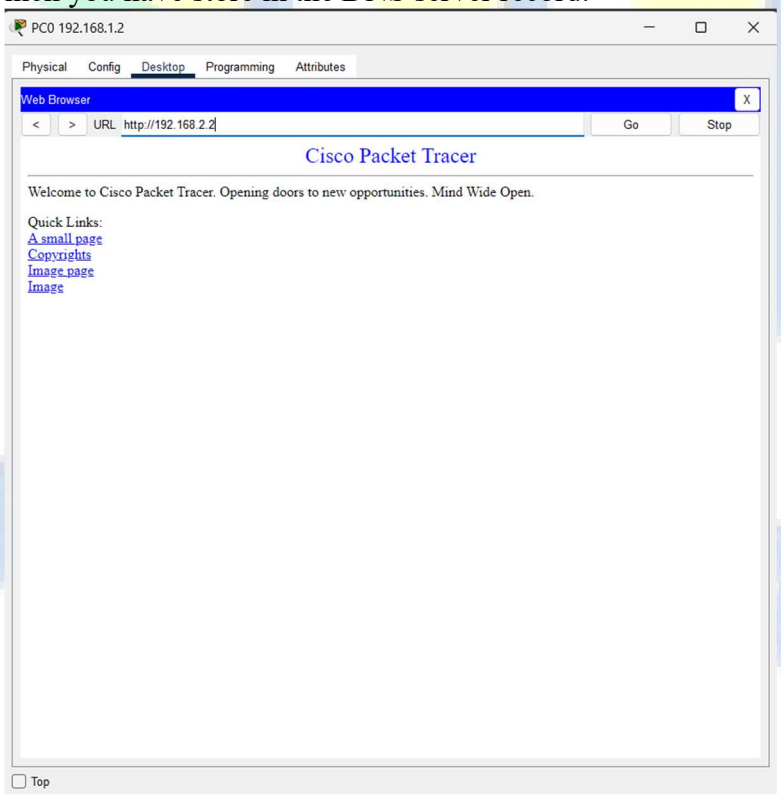

Figure 4

To note the http header format information, go to simulation mode edit filters and click on http check box then click on capture/forward button.

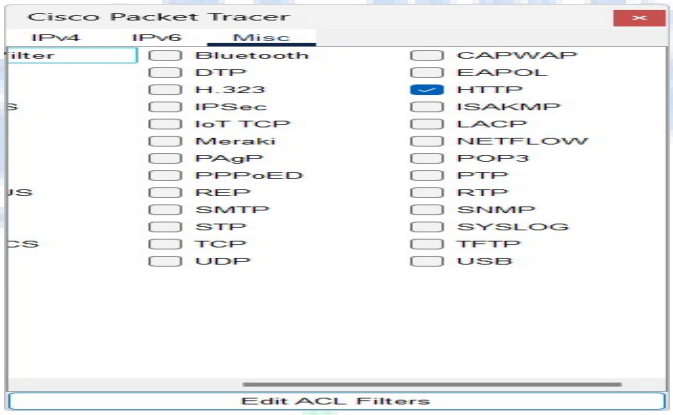

Figure 5

Now click on the http packet, you can note that the destination port is 80.

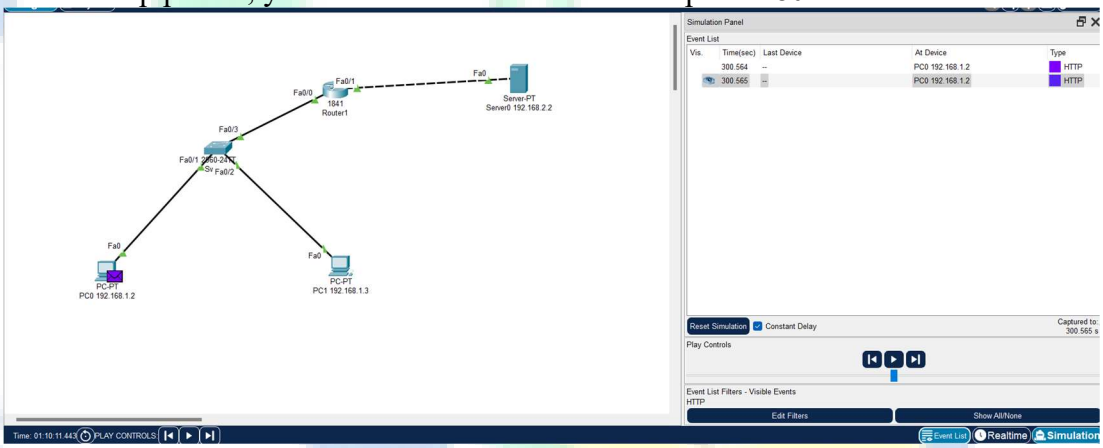

Figure 6

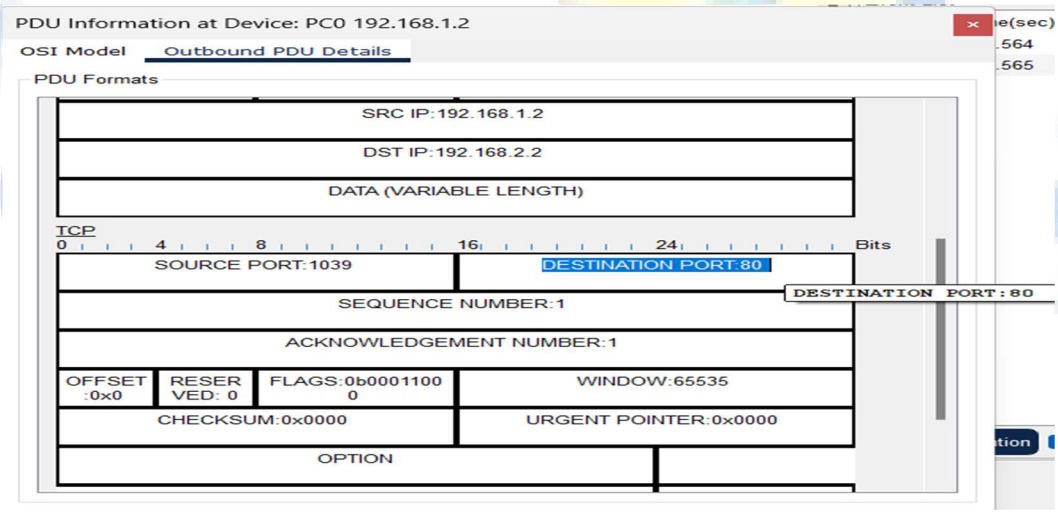

Figure 7

Now scroll the Outbound PDU Details, you can see the http protocol information.

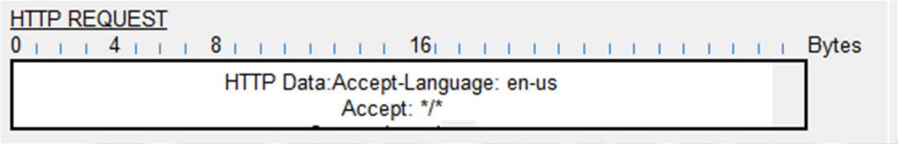

Figure 8

5b. Implementation for HTTPS

Now click on PC and go to Desktop---- >Web Browser. Now type web-server IP 192.168.2.2

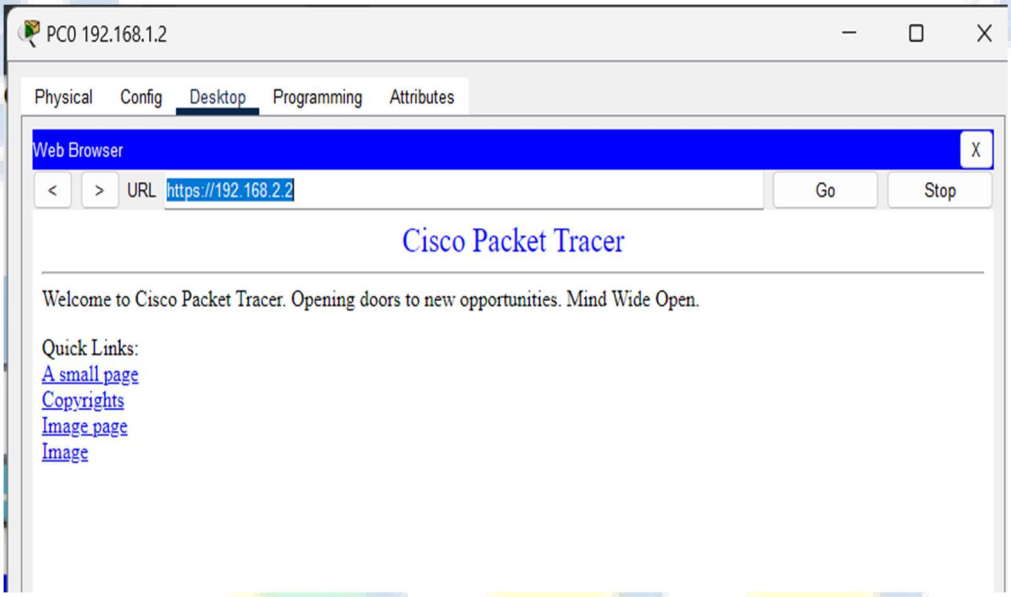

Figure 9

Now to note the https header format information go to simulation mode ----- > edit filters and click on https check box then click on capture/forward button.

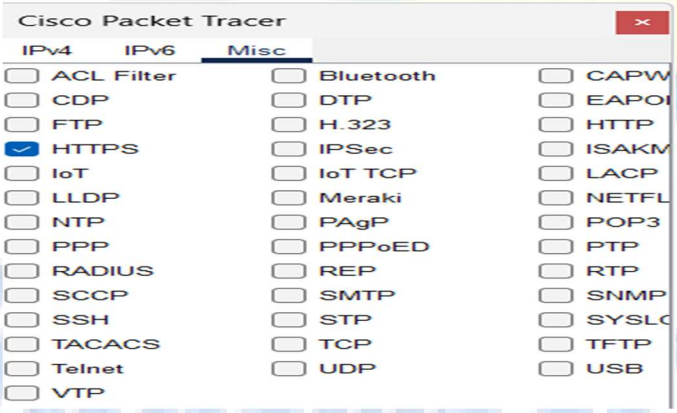

Figure 10

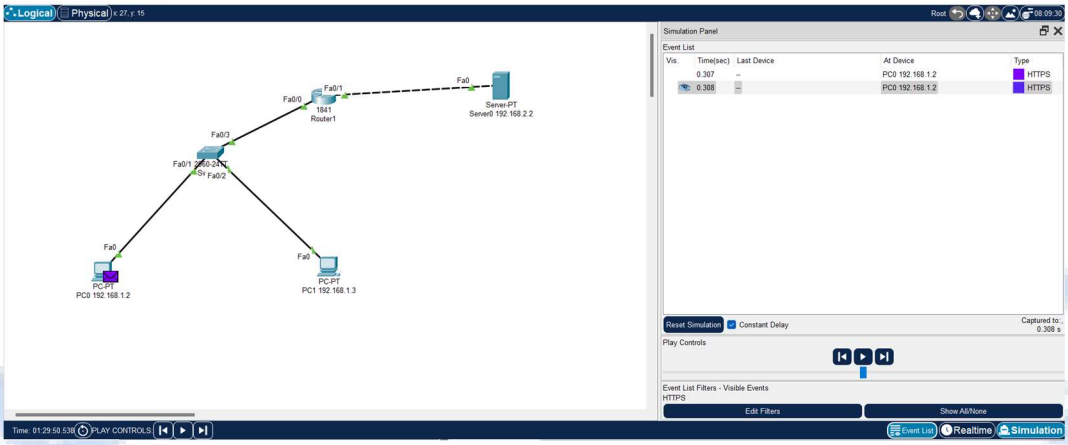

Figure 11

Now click on the https packet, you can note that the destination port is 443.

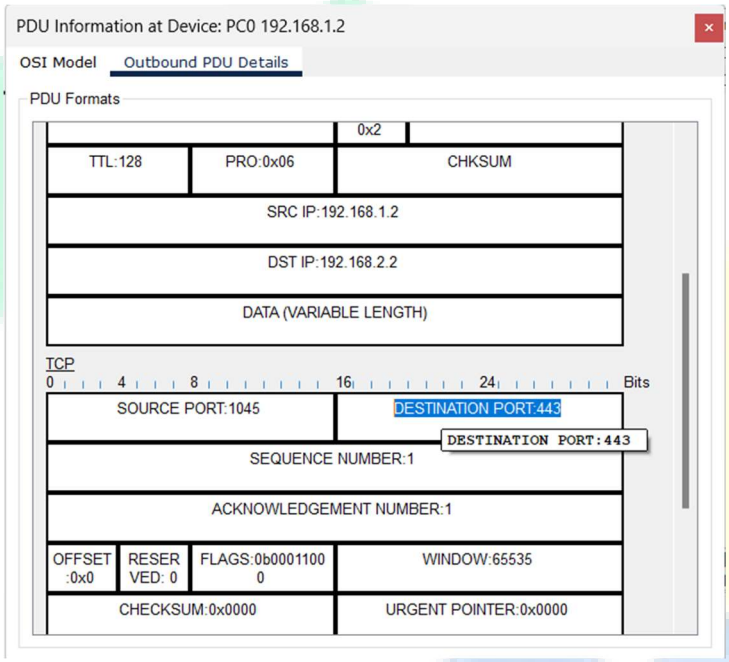

Figure 12

Now scroll the Outbound PDU Details, you can see the https PDU.

DNS IN CISCO PACKET TRACER

1. Introduction to DNS

The Domain Name System (DNS) is a hierarchical and distributed naming system for computers, services, and other resources in the Internet or other Internet Protocol (IP) networks. It associates various information with domain names assigned to each of the associated entities. Most prominently, it translates readily memorized domain names to the numerical IP addresses needed for locating and identifying computer services and devices with the underlying network protocols. The Domain Name System has been an essential component of the functionality of the Internet since 1985.

The Domain Name System distributes the responsibility of assigning domain names and mapping those names to IP addresses by designating authoritative name servers for each domain.

Authoritative name servers are assigned to be responsible for their supported domains, and may delegate authority over sub domains to other name servers. This mechanism provides distributed and fault tolerant service and was designed to avoid the need for a single central database.

Some common DNS record types are:

a) A Record:

The A record is one of the most commonly used record types in any DNS system. An A record is actually an address record, which means it maps a fully qualified domain name (FQDN) to an IP address. For example, an A record is used to point a domain name, such as "google.com", to the IP address of Google's hosting server, "74.125.224.147". This allows the end user to type in a human readable domain, while the computer can continue working with numbers. The name in the A record is the host for your domain, and the domain name is automatically attached to your name.

b) CNAME Record:

Canonical name records, or CNAME records, are often called alias records because they map an alias to the canonical name. When a name server finds a CNAME record, it replaces the name with the canonical name and looks up the new name. This allows pointing multiple systems to one IP without assigning a record to each host name. It means that if you decide to change your IP address, you will only have to change one A record.

c) NS Record:

An NS record identifies which DNS server is authoritative for a particular zone. The "NS" stands for "name server". NS records that do not exist on the apex of a domain are primarily used for splitting up the management of records on sub-domains.

d) SOA Record:

The SOA or Start of Authority record for a domain stores information about the name of the server that supplies the data for the zone, the administrator of the zone and the current version of the data. It also provides information about the number of seconds a secondary name server should wait before checking for updates or before retrying a failed zone transfer.

Implementation:

Consider the following topology:

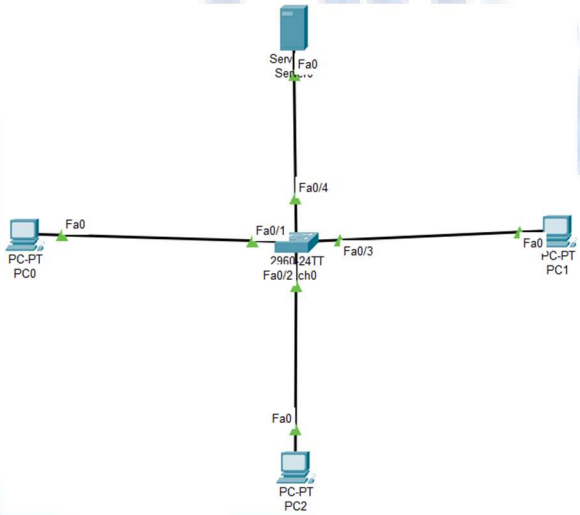

Figure 13

Assigning IP to DNS server:

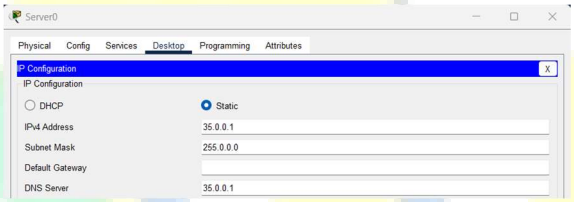

Figure 14

Assigning IP to PCs:

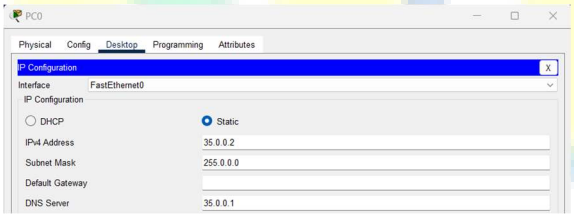

Figure 15

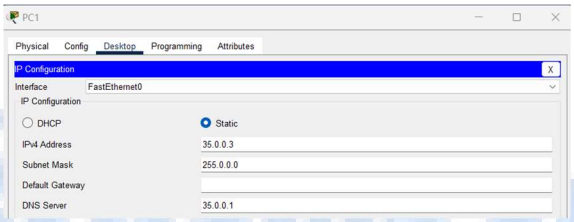

Figure 16

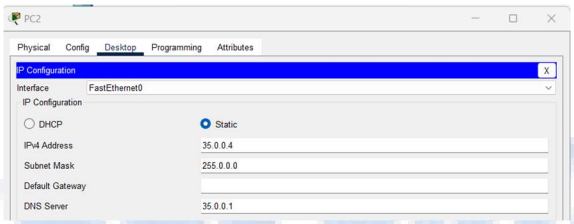

Figure 17

2. DNS Configuration & Simulation

Now using the DNS service on DNS Server. Go to server services DNS.

First, we add A record. We assign the web server IP against our Domain name. Now click on Add.

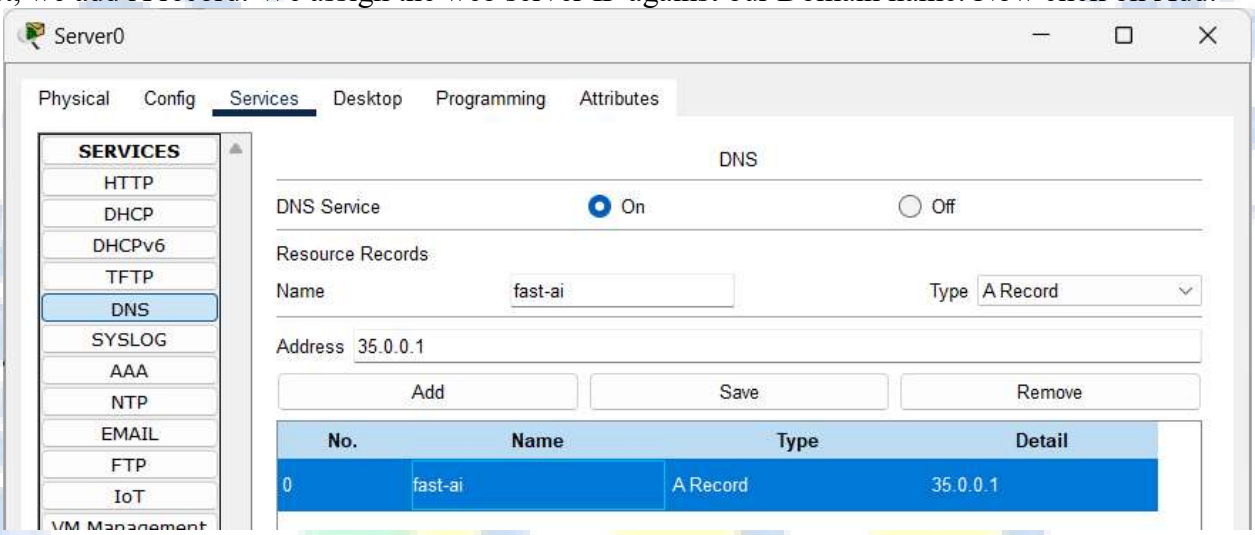

Figure 18

Now add CNAME record.

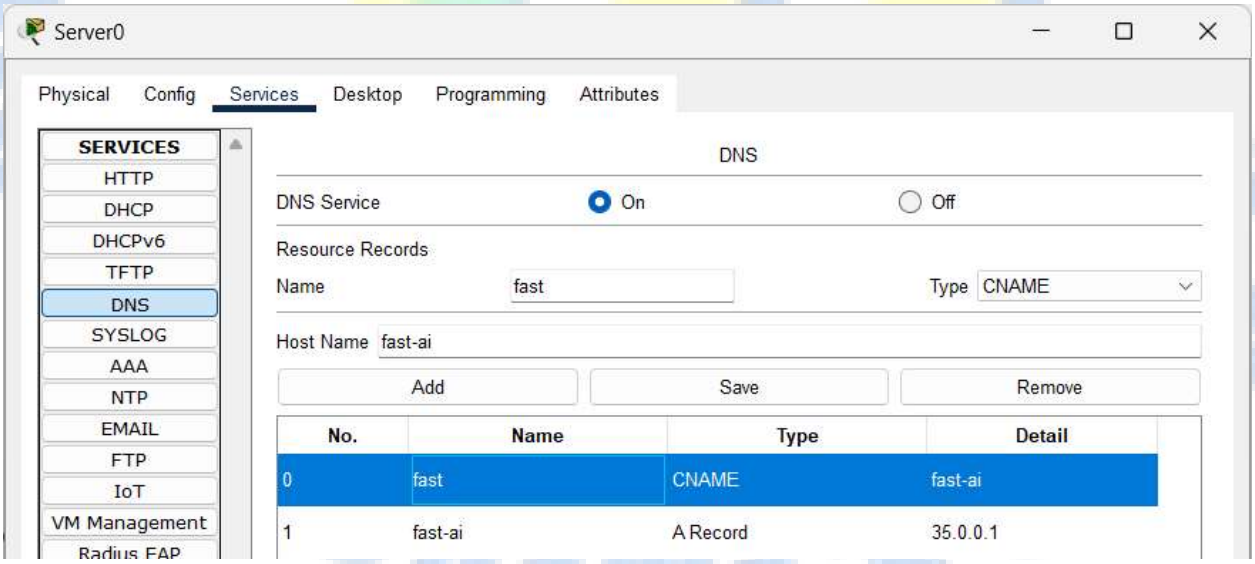

Figure 19

Now go to any PC →Desktop →web browser → type **"fast-ai"** and see how DNS works.

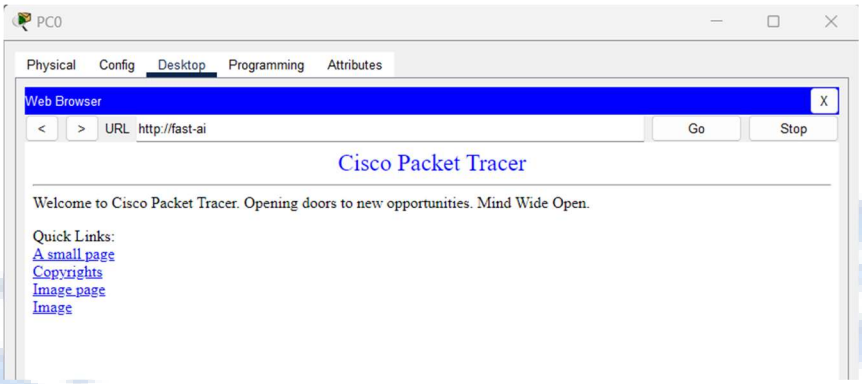

Figure 20

Start simulation.

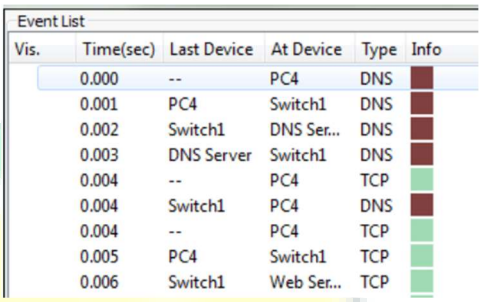

Figure 21

Click on DNS packet. See how DNS server resolved the name.

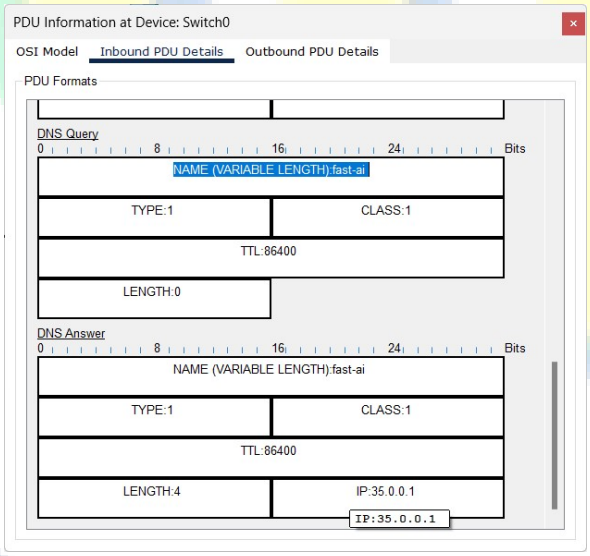

Figure 22

Shows OSI layers involved in transmission.

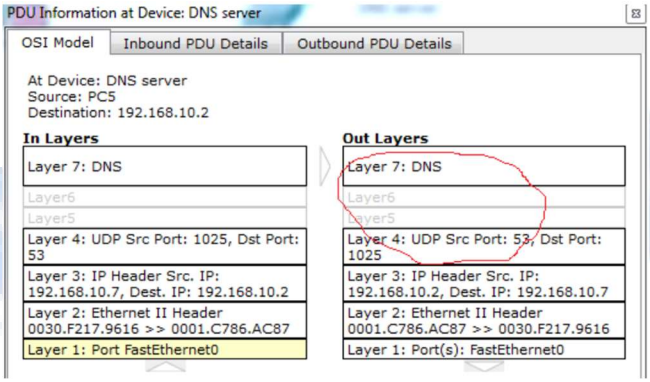

Figure 23

WIRESHARK

Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

Why we use Wireshark?

Wireshark has many uses, including troubleshooting networks that have performance issues. Cybersecurity professionals often use Wireshark to trace connections, view the contents of suspect network transactions and identify bursts of network traffic.

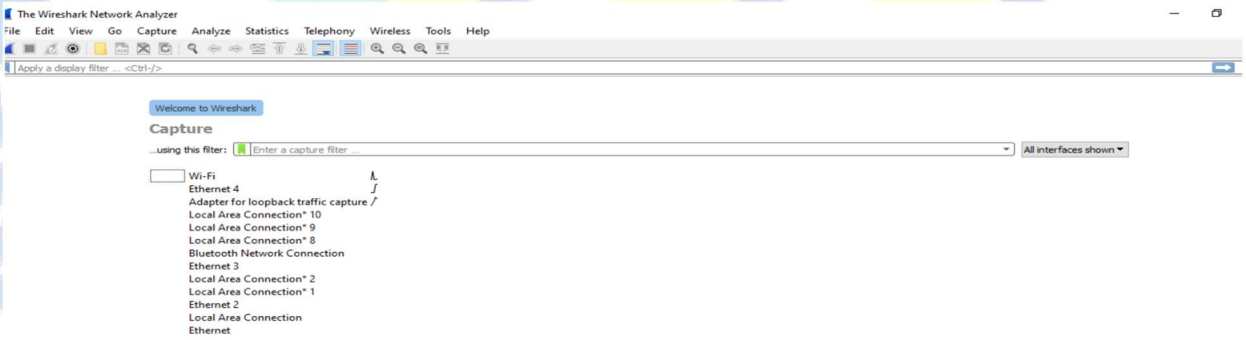

Figure 24

Open Wireshark

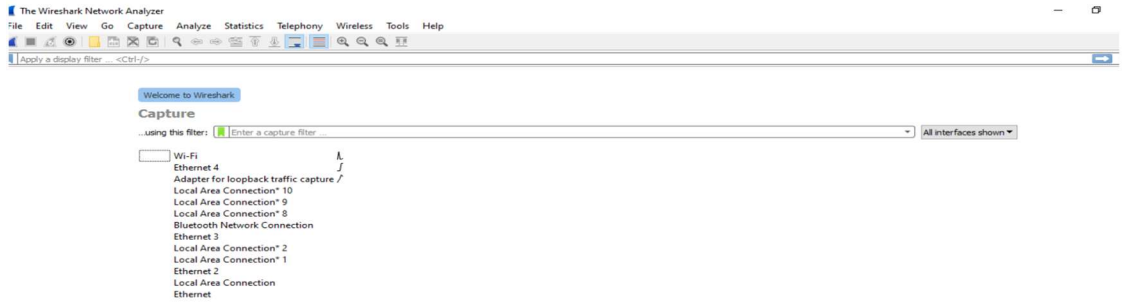

Figure 25

Select the technology you used for packet analysis

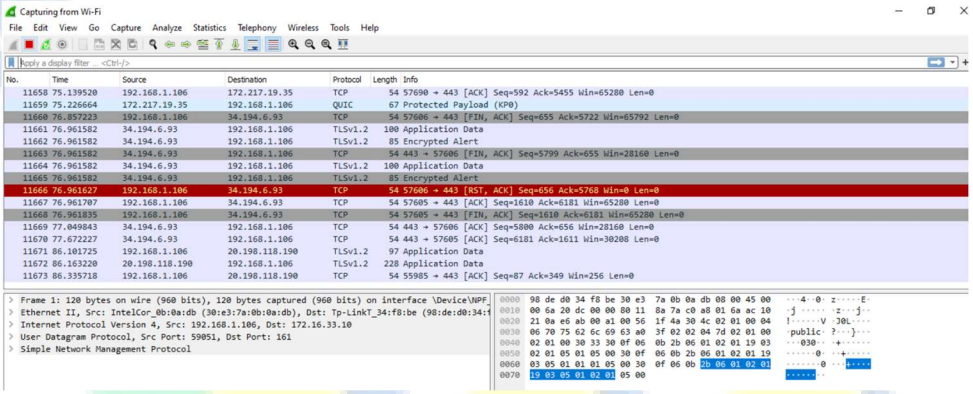

Figure 26

HTTP Packet Analysis

Goto website below:
http://testphp.vulnweb.com/login.php
Username: test
Password: test
Let`s take a look here what we can understand from here:

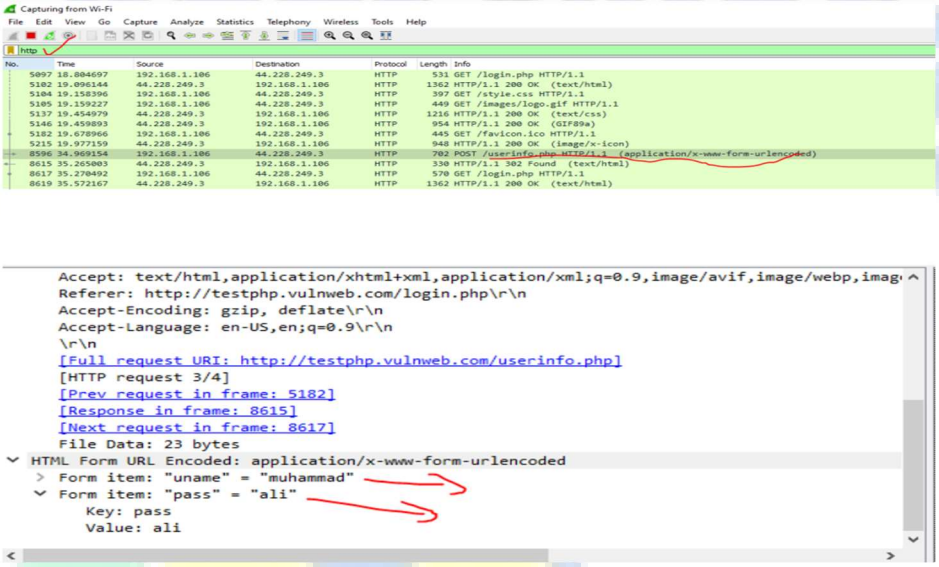

Figure 27

INTRODUCTION TO ROUTERS

They are small electronic devices that join multiple computer networks together via either wired or wireless connections.

1. How Routers Work?

In technical terms, a router is a Layer 3 network gateway device, meaning that it connects two or more networks and that the router operates at the network layer of the OSI model.
Routers contain a processor (CPU), several kinds of digital memory, and input-output (I/O) interfaces. They function as special-purpose computers, one that does not require a keyboard or display. The router's memory stores an embedded operating system (OS). Compared to general-purpose OS products like Microsoft Windows or Apple Mac OS, router operating systems limit what kind of applications can be run on them and also need much smaller amounts of storage space. Examples of popular router operating systems include Cisco Internetwork Operating System (IOS) and DD-WRT. These operating systems are manufactured into a binary firmware image and are commonly called router firmware.
By maintaining configuration information in a part of memory called the routing table, routers also can filter both incoming or outgoing traffic based on the addresses of senders and receivers.
Routers became mainstream consumer devices when households began to accumulate multiple computers and wanted to share the home Internet connection Home networks use Internet Protocol (IP) routers to connect computers to each other and to the Internet. Early generations of home routers supported wired networking with Ethernet cables while newer wireless routers supported Wi-Fi together with Ethernet. The term broadband router applies to any home wired or wireless router being used for sharing a broadband Internet connection.

2. Routing Table

A routing table contains the information necessary to forward a packet along the best path toward its destination. Each packet contains information about its origin and destination. When a packet is received, a network device examines the packet and matches it to the routing table entry providing the best match for its destination. The table then provides the device with instructions for sending the packet to the next hop on its route across the network.

A basic routing table includes the following information:

- **Destination**: The IP address of the packet's final destination.
- **Next hop:** The IP address to which the packet is forwarded.
- **Interface:** The outgoing network interface the device should use when forwarding the packet to the next hop or final destination.

There is much more information present in the routing table which will be discussed in the upcoming labs.

3. Simple Router Configuration

In this lab we will only have a look at how we can assign IPs to router interfaces using GUI (Graphical User Interface) and CLI (Command Line Interface). Further configurations will be done in later labs.

IP Assignment:

Following is the way for IP assignment to a router interface (port) using CLI.

```
Router>en
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface e 0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shut
%LINK-3-UPDOWN: Interface Ethernet0, changed state to up
Router(config-if)#exit
Router(config)#exit
```