Section 1: Nmap (Network Mapper)

1. What is Nmap, and what is its primary purpose in network monitoring?

- Nmap (Network Mapper) is an open-source network scanning tool used for network discovery and security auditing. Its primary purpose is to identify active devices, detect open ports, discover services running on a network and assess security vulnerabilities.

2. Explain the different types of scans available in Nmap. Provide examples.

- TCP Connect Scan (nmap -sT): Performs a full TCP three-way handshake to determine open ports.
- SYN Scan (nmap -sS): A stealthy scan that sends SYN packets and detects open ports without completing the handshake.
- UDP scan (nmap -sU): Scans for open UDP ports, useful for detecting services like DNS and SNMP.

APCO
Date: / /

3. How does Nmap help in identifying open ports and services on a network?
- Nmap sends specially crafted packets to target systems and analyzes the responses to determine which ports are open, closed or filtered. It also uses detection (-sV) to identify the specific applications running on open ports.

4. What is OS detection in Nmap and how does it work?
- OS detection (-O flag) allows Nmap to determine the operating system of a target device by analyzing factors such as TCP/IP stack responses, packet TTL values and specific fingerprinting techniques.

5. Discuss the significance of the -A option in Nmap scanning.
- The (-A) option enables aggressive scanning which includes OS detection, version detection, script scanning, and traceroute. It provide comprehensive information about the target network but is more intrusive.

6. How can Nmap be used for security auditing and penetration testing?
→ It identifies open ports and running services, exposing potential vulnerabilities.
→ It detects outdated software versions with known exploits.

→ Uncovers misconfigurations and weak security controls.

7. What are the limitations of Nmap in network Monitoring?

→ It cannot detect all vulnerabilities - relies on known signatures.

→ Slow performance on large network or when scanning with deep analysis options.

→ Limited effectiveness against encrypted traffic.

Section 2: Wireshark.

1. What is Wireshark, and how it used in network analysis?

→ Wireshark is powerful open-source packet analyser used for real-time network traffic analysis. It captures and inspects data packets travelling over a network, helping network administrators and security professionals, detect intrusions.

2. Explain the different components of a captured packet in Wireshark.

→ Frame Header - Contains metadata such as timestamp, frame length and source/destination MAC addresses.

APCO
Date: / /

→ Ethernet Header: Includes source and destination MAC addresses and the Ether Type field indicating the next protocol layer

→ IP Header: shows the source and destination IP addresses, protocol type (TCP, UDP, ICMP) and packet fragmentation details.

→ Transport layer: Display port numbers, sequence/acknowledgment numbers and flags.

3. How does Wireshark help in troubleshooting network issues?

→ It identifies latency and network congestion by analyzing packet delays.

→ Detects dropped packets, retransmissions and high error rates.

→ Analyzes DNS, HTTP traffic to pinpoint misconfigurations.

4. Discuss the importance of filtering packets in Wireshark?

→ Capture filters: Limit the data collected during packet capture.

→ Display filters: Narrow down displayed packets for easier analysis.

→ Filtering enhances performance, reduces noise, speeds up troubleshooting.

APCO
Date: / /

5. What security concerns should be considered when using Wireshark?

→ Unauthorised packet capture can expose sensitive data such as passwords or confidential communications.

→ Wireshark does not decrypt encrypted traffic but plaintext data remains visible.

→ Running Wireshark with administrative privileges can pose security risks.

6. How does Wireshark differentiate between various network protocols?

→ Wireshark use protocol dissectors, which analyze packet headers and payloads to classify traffic based on predefined rules. It recognizes protocols by inspecting fields such as Ethernet, port numbers and protocol identifiers (IP, ARP).

7. Why is Wireshark widely used in cybersecurity investigations?

→ It detects suspicious network traffic, such as malware communication and unauthorised access attempts.

→ Monitors encrypted traffic patterns for anomalies without decrypting data

→ It helps in analyse distributed attacks by identifying malicious packet floods.

Section 3: Comparitive Analysis

1. Compare and contrast Nmap and Wireshark in terms of their functionalities and use cases.

→

Nmap	Wireshark
(i) Network Scanning and discovery.	(i) Packet capture and analysis
(ii) It identifies open ports, services and hosts on a network.	(ii) It inspects the real-time network traffic at a granular level.
(iii) It sends packets to gather information.	(iii) It listens to network traffic without sending packets.
(iv) It identifies network structure, weak points and misconfigurations	(iv) It Analyses detailed network communications and detecting anomalies.

2. Which tool would you use for detecting open ports and why? → Nmap is the best tool because:

→ It actively probes target hosts and check port statuses (open, closed, filtered).

→ It provides detailed service

→ It is designed specifically for network scanning and security auditing

APCO
Date: / /

3. How can both Nmap and Wireshark be used together for network security analysis?

→ Step 1: Use Nmap to scan for vulnerabilities
- Identify active hosts.
- Detect open ports and services.
- Find OS details.

→ Step 2: Use Wireshark to analyze suspicious traffic
- Capture network traffic to examine packet behaviour.
- Filter specific traffic.
- Detect anomalies such as excessive failed login attempts or unusual data transfers.

→ Step 3: Correlate findings
- If Nmap detects an open port running a vulnerable service, Wireshark can confirm if it's being exploited.
- If Wireshark detects suspicious traffic, Nmap can scan for unprotected devices that may be the source.

4. Discuss a real-world scenario where Nmap and Wireshark would help in identifying a network attack.

APCO
Date: / /

— Scenario – Detecting a Brute-force Attack on a SSH Server

→ Step 1: Use Nmap to identify Open SSH Ports
- Run nmap -p 22 -sV target_ip to check if SSH is open.
- If detected, further analyse version details to check for vulnerabilities.

→ Step 2: Use Wireshark to Monitor SSH traffic
- Apply a filter (tcp.port == 22) to capture all SSH related packets
- Detect multiple failed login attempts from the same IP, indicating a brute-force attack.

→ Step 3: Take Action
- Block the attacker's IP using a firewall or intrusion prevention system (IPS)
- Enforce stronger authentication mechanisms
- Monitor further traffic for other attack patterns