|  |  |
| --- | --- |
| **Experiment No: 14** | |
| **Name** | Vaibhav Sharma |
| **PRN** | 22070126125 |
| **Date of Performance** | 16th October 2024 |
| **Title** | To implement Network Troubleshooting using command line tools |
| **Theory (short)** | Networking commands are essential tools that allow users to interact with and diagnose network connections, resolve DNS issues, inspect routing tables, and capture data packets. These commands are universal in the sense that they can be used across different operating systems (Windows, macOS, Linux), though some minor variations in syntax may exist. Each of these commands serves a different function, but they all contribute to understanding the flow of data within a networked environment. |
| **Procedure** | 1. **Ping – Test Connectivity**    * **Objective**: Verify if a device can reach another device over the network.    * **Steps**:      1. Open a terminal/command prompt.      2. Type the following:         + **Windows/Linux/macOS**:   ping <hostname or IP address>   1. **Traceroute – Trace Path of Data**    * **Objective**: Determine the route data packets take from your device to the destination.    * **Steps**:      1. Open a terminal/command prompt.      2. Type the following:         + **Linux/macOS:**   traceroute <hostname or IP address>   * + - * **Windows:**   tracert <hostname or IP address>   * + 1. **IPConfig/Ifconfig – Display Network Configuration** * **Objective**: Display the current network settings of your system. * **Steps**:   1. Open a terminal/command prompt.   2. Type the following: |

|  |  |
| --- | --- |
|  | * **Linux**: ifconfig * **Mac:**   ifconfig   * **Windows:**   ipconfig   1. **Nslookup – Query DNS Information**    * **Objective**: Resolve domain names to IP addresses and vice versa.    * **Steps**:      1. Open a terminal/command prompt.      2. Type the following:         + **Linux/macOS/Windows:**   nslookup <hostname or domain>   1. **Netstat – View Network Connections**    * **Objective**: Display network connections and ports in use.    * **Steps**:      1. Open a terminal/command prompt.      2. Type the following:         + **Linux/macOS/Windows**: netstat -a 2. **ARP – View/Manage Address Resolution Protocol Table**    * **Objective**: View IP-to-MAC address mappings.    * **Steps**:      1. Open a terminal/command prompt.      2. Type the following:         + **Linux/macOS/Windows**: arp -a 3. **Route – Display/Modify Routing Table**    * **Objective: View or modify the IP routing table that governs data flow.**    * **Steps:**      1. **Open a terminal/command prompt.**      2. **To view the routing table, type the following:**         + **Linux:**   route -n   * + - * **MacOS:**   netstat -rn   * + - * **Windows:**   route print |

|  |  |
| --- | --- |
|  | 1. **Tcpdump – Capture Network Traffic (Linux/macOS)**    * **Objective**: Capture and analyze network packets.    * **Steps**:      1. Open a terminal.      2. Run the following command to start capturing network packets:         + **Linux/macOS:**   sudo tcpdump   * + - * **Windows**   Use Wireshark or WinDump |
| **Output Screenshots** | **Vaibhav Sharma**  **Fig 1- Pinging a website**    **Vaibhav Sharma**  **Fig 2- Tracing a route to a website** |

|  |  |
| --- | --- |
|  | Vaibhav Sharma  **Fig 3- ipconfig of my Wi-Fi**    **Fig 4- Finding DNS using nslookup** |

|  |  |
| --- | --- |
|  | Vaibhav Sharma  **Fig 5- Finding active network connections using netstat**    Vaibhav Sharma  **Fig 6- Mapping IP’s using arp command** |

|  |  |
| --- | --- |
|  | **Fig 7- Routing table in Windows**    **Fig 8- tcpdump for Windows(Taken from Wireshark since there is no command for native windows)** |

|  |  |
| --- | --- |
| **Observation** | 1. **Ping – Testing Connectivity**    * **Observations**:      + **Response time**: Measures how long it takes for packets to travel to the destination and back. High response times indicate network latency.      + **Packet loss**: Shows whether packets are being dropped along the path. Any packet loss suggests a problem with the network connection (e.g., poor link quality, misconfiguration, or congestion).      + **Unreachable Host**: If the ping fails, it indicates that the target is either down or unreachable due to routing issues, firewall settings, or host unavailability. 2. **Traceroute – Path Analysis**    * **Observations**:      + **Number of hops**: Displays the number of routers (hops) a packet passes through. A higher-than-expected number of hops can indicate suboptimal routing.      + **Response times at each hop**: Helps identify where delays are occurring in the network. If a particular hop shows a high delay or failure to respond, it might indicate congestion, a network bottleneck, or an outage at that point.      + **Path deviation**: The route should generally follow a known or expected path. If packets take unexpected routes, it could indicate a routing issue or misconfiguration. 3. **IPConfig/Ifconfig – Network Configuration**    * **Observations**:      + **IP Address**: Ensure that the device has the correct IP address assigned, either static or dynamically assigned by DHCP. An invalid or missing IP address could cause connectivity issues.      + **Subnet mask and gateway**: Check if the subnet mask and default gateway are correct. A wrong subnet or |

|  |  |
| --- | --- |
|  | gateway can prevent the device from communicating outside its local network.   * **MAC address**: Displays the hardware address of the network interfaces, useful for identifying devices on the network.  1. **Nslookup – DNS Resolution**    * **Observations**:      + **IP address resolution**: Nslookup should resolve the hostname into the correct IP address. If the resolution fails or returns the wrong IP, it indicates a DNS misconfiguration.      + **DNS server response**: If the DNS server is unreachable or returns an error, it suggests an issue with the DNS server configuration, or the server may be down.      + **Reverse lookup**: Using nslookup with an IP address should return the correct domain name if reverse DNS is configured correctly. If not, it could indicate a lack of reverse DNS records. 2. **Netstat – Network Connection Status**    * **Observations**:      + **Active connections**: Lists all active network connections. This is useful for identifying which services or applications are using the network and their associated IP addresses and ports.      + **Listening ports**: Observing open or listening ports helps to ensure that necessary services are running. Unexpected open ports could indicate a security risk (e.g., an open port vulnerable to attack).      + **Foreign addresses**: Displays the IP addresses and ports of remote systems connected to your device. Unrecognized connections may indicate malicious activity or unauthorized access. 3. **ARP – Address Mapping**    * **Observations**: |

|  |  |
| --- | --- |
|  | * **IP-to-MAC mapping**: The ARP table shows how IP addresses are mapped to MAC addresses within the local network. A missing or incorrect ARP entry could explain communication failures between devices. * **Suspicious entries**: Unexpected ARP entries (i.e., IP addresses or MAC addresses that don’t belong to known devices) may indicate an ARP spoofing attack, where a malicious actor is impersonating another device on the network.  1. **Route – Routing Table Inspection**    * **Observations**:      + **Default gateway**: Ensure that the default gateway is correctly configured. An incorrect or missing gateway could prevent access to other networks, including the internet.      + **Routing paths**: Verify that routes to other networks (such as internal subnets) are present and accurate. Missing or wrong routes may cause traffic to be misrouted, resulting in unreachable networks.      + **Metric**: The routing metric helps to determine the priority of a route. Lower metrics take precedence. Multiple routes to the same destination with different metrics could indicate load balancing or redundancy. 2. **Tcpdump – Packet Capture and Analysis**    * **Observations**:      + **Packet details**: View the data flowing through the network in real time. Analyzing the source and destination of packets helps in diagnosing issues like communication errors, protocol misconfigurations, or unauthorized traffic.      + **Traffic anomalies**: Unusual or excessive traffic from specific sources may indicate network misuse, a DDoS attack, or malware infection.      + **Protocol analysis**: By examining specific protocol traffic (e.g., HTTP, DNS, TCP), you can pinpoint issues with |

|  |  |
| --- | --- |
|  | specific services, such as web servers, DNS servers, or database applications.   * **Dropped packets**: Observing dropped packets or retransmissions can highlight network instability, congestion, or hardware failures. |
| **Self- assessment**  **Q&A** | Q: What is the purpose of the ping command in network troubleshooting?  Ans: ping checks the connectivity between your system and a remote host by sending ICMP Echo Requests and receiving Echo Replies.  Q: How does the traceroute command help in identifying network issues?  Ans: traceroute shows the path packets take to reach a destination, helping to identify where delays or failures occur in the network.  Q: What does the ipconfig command display?  Ans: ipconfig displays the IP configuration of a system, including IP addresses, subnet masks, and default gateways. |
| **Conclusion** | Networking commands like ping, traceroute, ipconfig/ifconfig, nslookup, netstat, arp, route, and tcpdump provide invaluable insights into the structure, health, and performance of a network. These tools enable users to test connectivity, resolve DNS issues, inspect routing tables, analyze network traffic, and detect security vulnerabilities. Regularly utilizing these commands allows network administrators and users alike to maintain optimal network performance, quickly identify and resolve problems, and ensure network security. Mastery of these tools is essential for anyone involved in managing or troubleshooting  networks, forming the foundation for effective network diagnostics and analysis. |