**Lab 3: Basic Networking Commands in Ubuntu**

**Objectives:**

This lab aims to provide hands-on experience with basic networking commands in Ubuntu. By the end of the lab, you will be familiar with key network configuration and troubleshooting tools such as `ifconfig`, `ping`, `traceroute`, and `netstat`.

**Aim:**

To study and utilize fundamental networking commands for network configuration and troubleshooting.

**Software Required:**

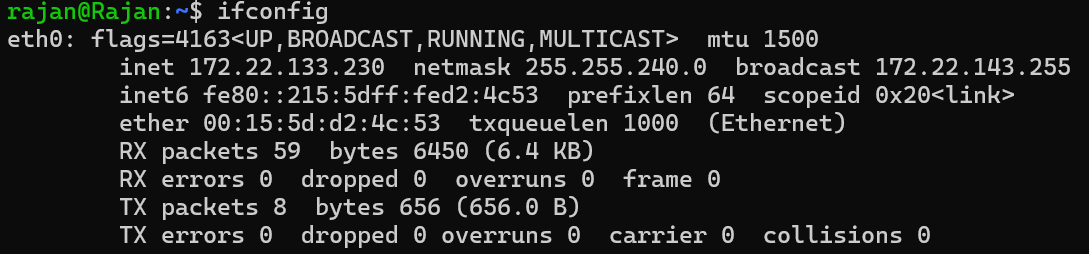
- Ubuntu Operating System

**Networking Commands:**

**1. ifconfig**

The `ifconfig` command is used to configure and display the status of network interfaces. It allows you to assign IP addresses, enable or disable interfaces, and configure multicast mode.

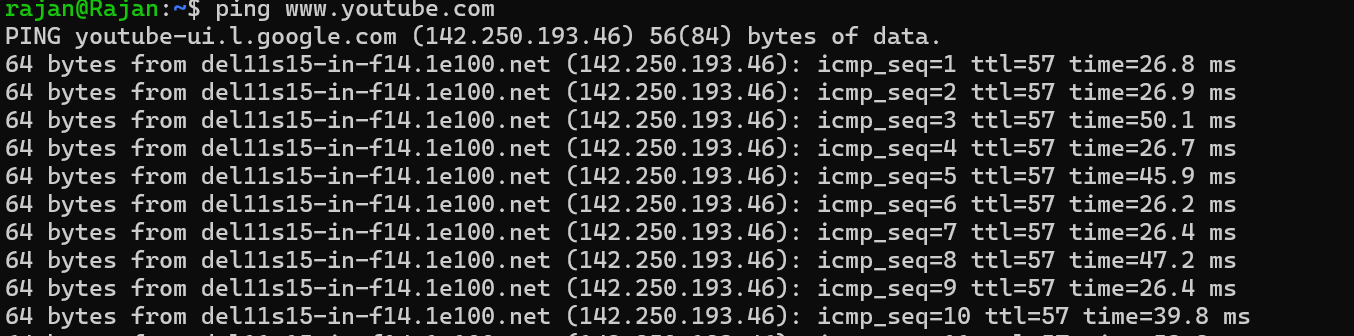
**Command: ifconfig**



**2. ping**

The `ping` command tests network connectivity between your machine and a target host by sending ICMP echo requests. It measures the response time, which helps identify network latency or connectivity issues.

**Command: ping [hostname or IP address]**



**3. tracepath:** The `tracepath` command traces the path packets take to reach a destination and shows the intermediate hops. It also provides information about the Maximum Transmission Unit (MTU), helping diagnose routing and performance issues.

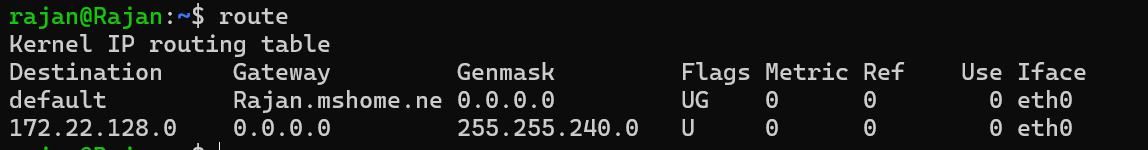
**Command:** **tracepath [hostname or IP address]**



**4. route**

The `route` command is used to view and manipulate the system’s IP routing table. It allows you to define how network packets are routed within your system or across networks.

**Command: route**



**5. netstat**

The netstat command displays detailed information about active network connections, routing tables, and interface statistics. It is a vital tool for network diagnostics, monitoring active connections, and identifying network bottlenecks.

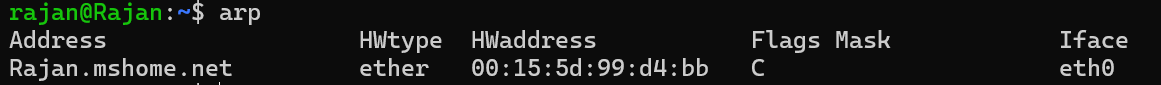
**Command: netstat**



**6. arp**

The arp command manages the ARP (Address Resolution Protocol) cache, which stores mappings between IP addresses and physical (MAC) addresses on the local network. This helps with resolving address conflicts and maintaining a clean ARP cache.

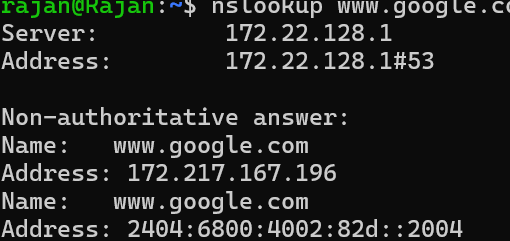
**Command: arp**



**7. nslookup**

The nslookup command queries DNS servers to obtain the IP address of a domain or the domain name associated with an IP. It’s useful for diagnosing DNS-related issues.

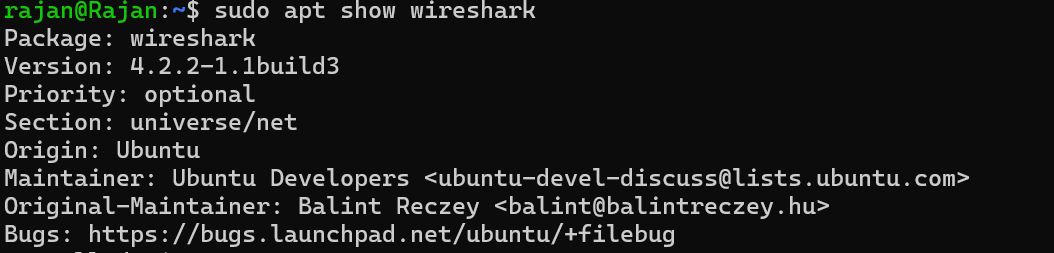
**Command: nslookup [hostname or IP address]**



**8. Wireshark**

Wireshark is a powerful network protocol analyzer that captures and inspects packets on a network in real-time. It provides deep analysis of network traffic, enabling you to diagnose issues or audit network security.

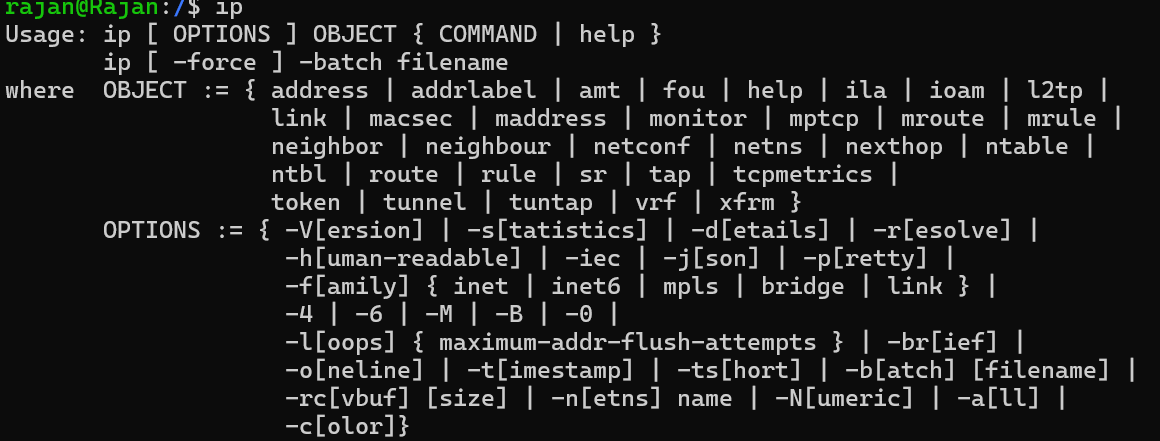
**Command: sudo apt show wireshark**



**9. ip**

The ip command is a versatile tool for managing IP addresses, routing, and network interfaces. It replaces older commands like ifconfig and provides more robust control over network configurations.

**Command: ip a**



**10. ss**

The ss command is used to display active socket connections and is more efficient than netstat. It can show detailed information about TCP, UDP, and other protocol connections, making it a great tool for monitoring network activity.

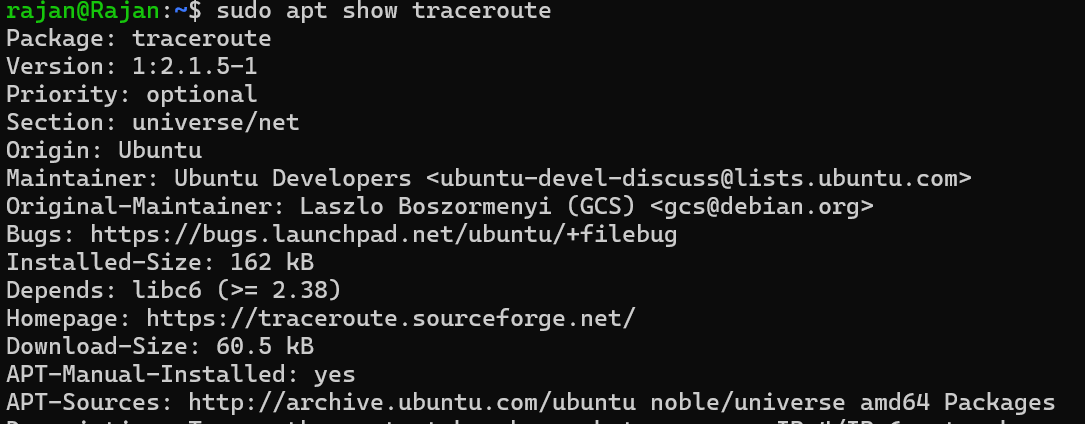
**Command: ss**



**11. traceroute**

The traceroute command is similar to tracepath and is used to trace the path packets take through the network to a specific destination. It lists the routers (hops) encountered along the way and helps identify delays or routing issues.

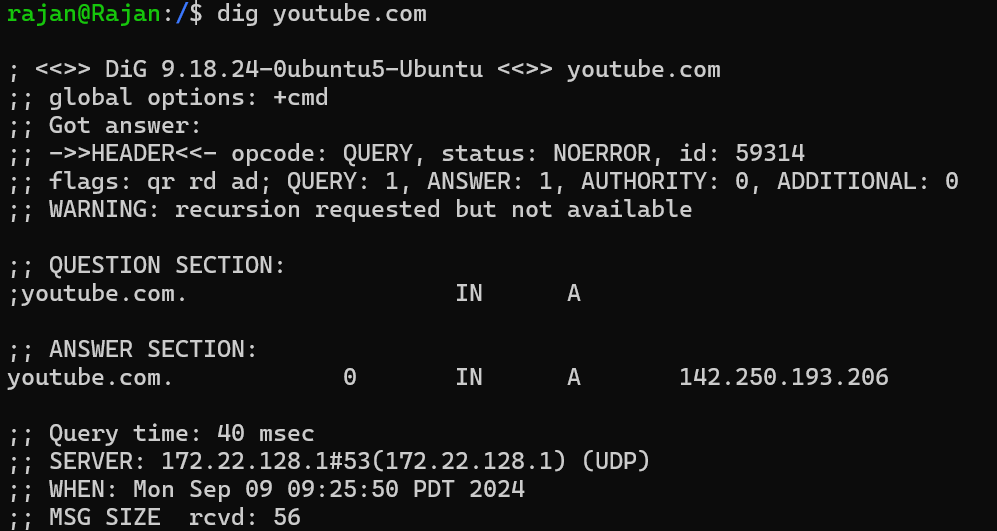
**Command: traceroute [hostname or IP address]**



**12. dig**

The dig command queries DNS name servers and is used for troubleshooting DNS issues. It allows you to retrieve various DNS records, such as A, MX, and TXT records.

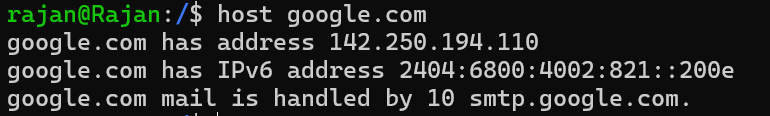
**Command: dig [domain]**



**13. host**

The host command performs DNS lookups, resolving domain names to IP addresses and vice versa. It's a simple and effective DNS query tool for quick verification of domain-to-IP mappings.

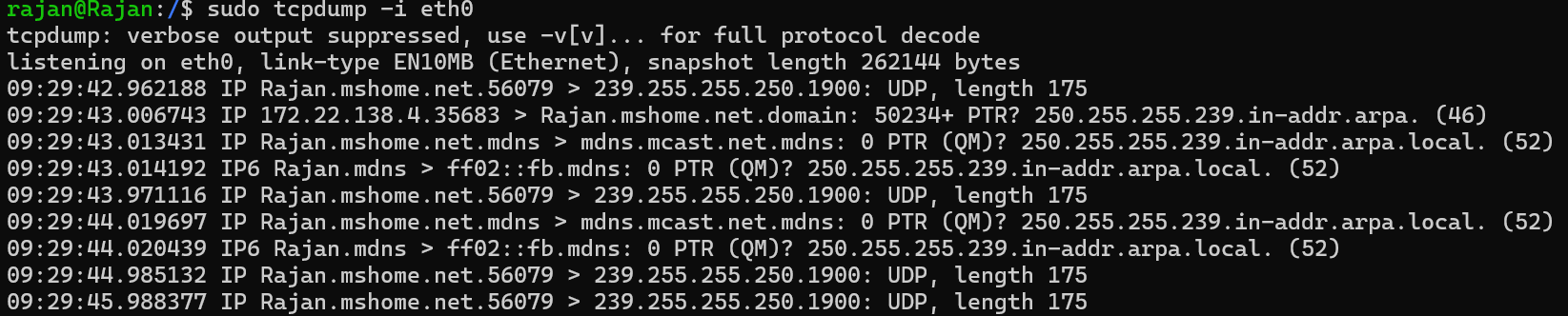
**Command: host [hostname or IP address]**



**14. tcpdump**

The tcpdump command is a packet analyzer that captures and displays packets transmitted over a network interface. It’s widely used for network traffic analysis and debugging.

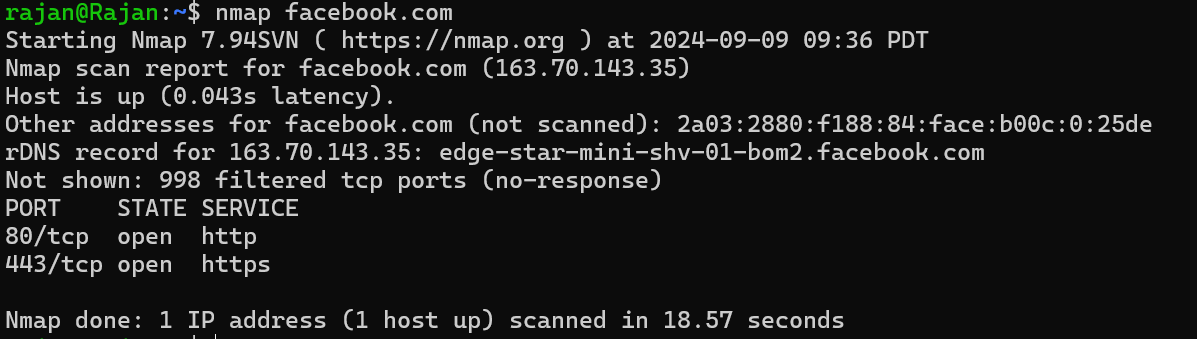
**Command: sudo tcpdump -i eth0**



**15. nmap**

The nmap (Network Mapper) tool is a powerful network scanner used for discovering hosts and services on a network. It is commonly used for port scanning, network mapping, and vulnerability assessment.

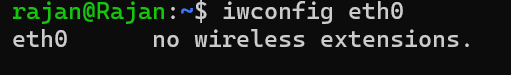
**Command: nmap [target]**



**16. iwconfig**

The iwconfig command is used to configure wireless network interfaces. It displays and modifies settings like SSID, frequency, and mode for Wi-Fi interfaces.

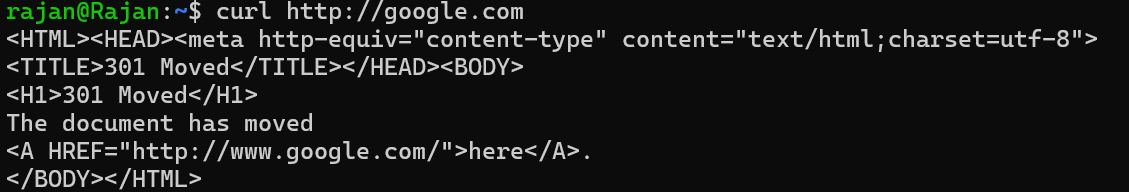
**Command: iwconfig [interface]**



**17. curl**

The curl command transfers data to and from servers using various protocols like HTTP, FTP, and more. It’s commonly used for testing APIs and downloading files from web servers.

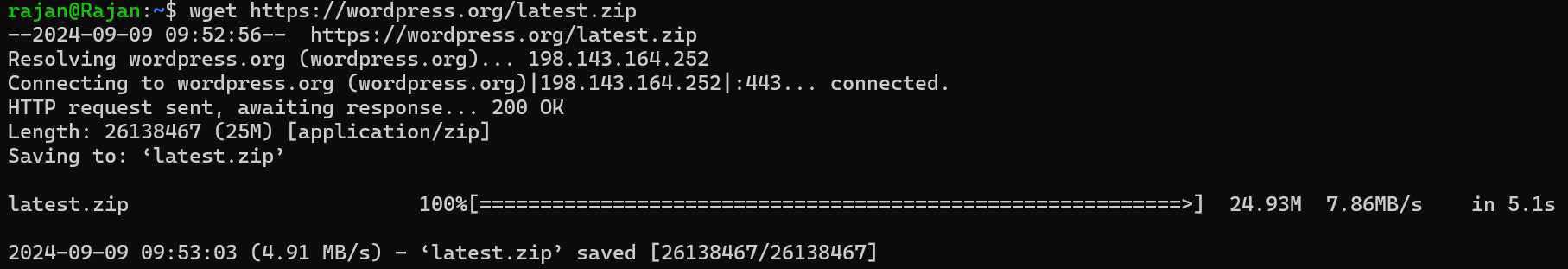
**Command: curl [URL]**



**18. wget**

The wget command is a non-interactive tool for downloading files from the web. It supports various protocols and can download large files or mirror entire websites.

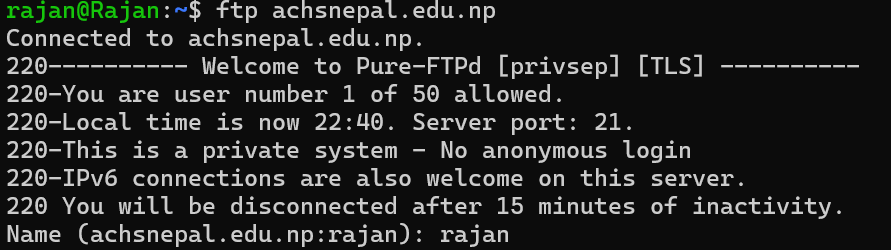
**Command: wget [URL]**



**19. ftp**

The ftp command is used for transferring files between a client and a server using the File Transfer Protocol. It provides an interactive session for file uploads and downloads.

**Command: ftp [hostname]**



**Conclusion**

This lab provides practical experience with essential networking commands in Ubuntu, enhancing your ability to manage and troubleshoot network interfaces. By understanding how to configure network settings, diagnose connectivity issues, and use various network analysis tools, you will gain a strong foundation in Linux networking.