**PRACTICAL-1**

AIM:

Perform port scanning using nmap on a single port and capture the packets using wireshark and analyze the output

**THEORY:**

1. **Nmap:**

* Nmap is a free and open-source network scanner created by Gordon Lyon. Nmap is used to discover hosts and services on a computer network by sending packets and analyzing the responses.
* Nmap provides a number of features for probing computer networks, including host discovery and service and operating system detection.
* These features are extensible by scripts that provide more advanced service detection, vulnerability detection, and other features.
* Nmap can adapt to network conditions including latency and congestion during a scan.
* Nmap started as a Linux utility and was ported to other systems including Windows, macOS, and BSD. It is most popular on Linux, followed by Windows.

1. **Wireshark:**

* Wireshark is a free and open-source packet analyser.
* It is used for network troubleshooting, analysis, software and communications protocol development, and education.
* Originally named Ethereal, the project was renamed Wireshark in May 2006 due to trademark issues.
* Wireshark is cross-platform, using the Qt widget toolkit in current releases to implement its user interface, and using pcap to capture packets; it runs on Linux, macOS, BSD, Solaris, some other Unix-like operating systems, and Microsoft Windows.
* There is also a terminal-based (non-GUI) version called TShark. Wireshark, and the other programs distributed with it such as TShark, are free software, released under the terms of version 2 of the GNU General Public License.

1. **open**

* An application is actively accepting TCP connections, UDP datagrams or SCTP associations on this port.
* Finding these is often the primary goal of port scanning. Security-minded people know that each open port is an avenue for attack.
* Attackers and pen-testers want to exploit the open ports, while administrators try to close or protect them with firewalls without thwarting legitimate users.
* Open ports are also interesting for non-security scans because they show services available for use on the network.

1. **closed**

* A closed port is accessible (it receives and responds to Nmap probe packets), but there is no application listening on it.
* They can be helpful in showing that a host is up on an IP address (host discovery, or ping scanning), and as part of OS detection. Because closed ports are reachable, it may be worth scanning later in case some open up.
* Administrators may want to consider blocking such ports with a firewall. Then they would appear in the filtered state, discussed next.

1. **filtered**

* Nmap cannot determine whether the port is open because packet filtering prevents its probes from reaching the port.
* The filtering could be from a dedicated firewall device, router rules, or host-based firewall software.
* These ports frustrate attackers because they provide so little information. Sometimes they respond with ICMP error messages such as type 3 code 13 (destination unreachable: communication administratively prohibited), but filters that simply drop probes without responding are far more common.
* This forces Nmap to retry several times just in case the probe was dropped due to network congestion rather than filtering. This slows down the scan dramatically.

1. **Unfiltered**

* The unfiltered state means that a port is accessible, but Nmap is unable to determine whether it is open or closed.
* Only the ACK scan, which is used to map firewall rulesets, classifies ports into this state.
* Scanning unfiltered ports with other scan types such as Window scan, SYN scan, or FIN scan, may help resolve whether the port is open.

1. **open|filtered**

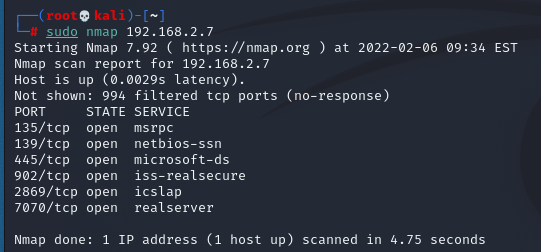
* Nmap places ports in this state when it is unable to determine whether a port is open or filtered. This occurs for scan types in which open ports give no response.
* The lack of response could also mean that a packet filter dropped the probe or any response it elicited.
* So Nmap does not know for sure whether the port is open or being filtered. The UDP, IP protocol, FIN, NULL, and Xmas scans classify ports this way.

1. **closed|filtered**

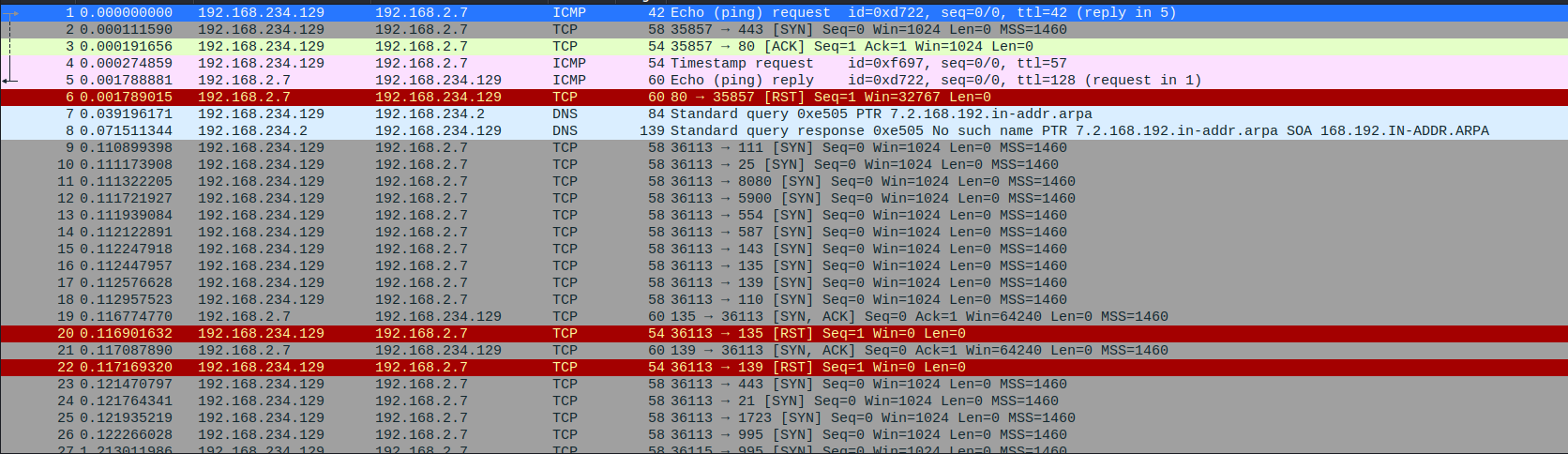
* This state is used when Nmap is unable to determine whether a port is closed or filtered.
* It is only used for the IP ID idle scan.

**IMPLEMENTATION:**

* Start the wireshark and start capturing the packets
* Firstly, we will write the following command:
* Write sudo nmap ip address of device
* This is the basic format for **Nmap**, and it will return information about the ports on that system.



* Below are the glimpses of the packets captured by wireshark when the above command was executed



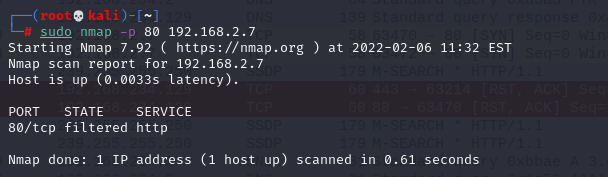
* Below are the packets captured for PORT 135



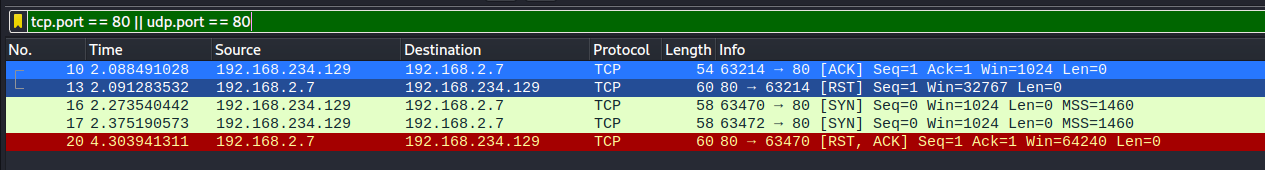
* If we want to scan for a range of ip address then, enter the following command
* Write sudo nmap ip address range



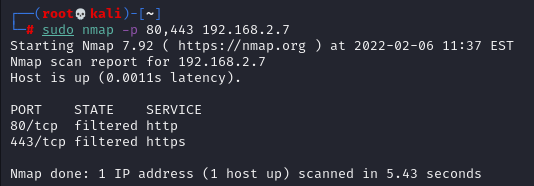
* You will get the result of scan for the whole range
* To know the status of a particular port, enter the following command



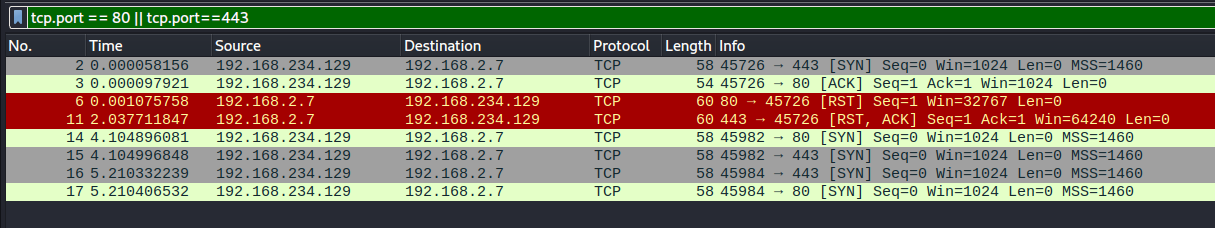
* Packets captured in wireshark



* For multiple ports, type the following command



* Packets captured by Wireshark



* To scan all the possible ports, write the following command



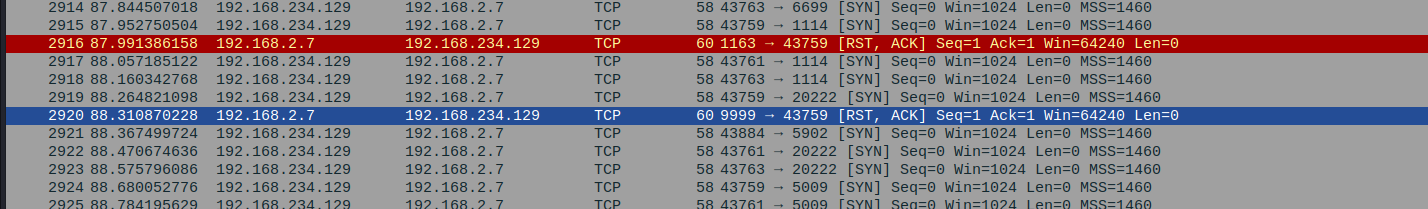
* To scan for all available TCP ports, enter the following command

E

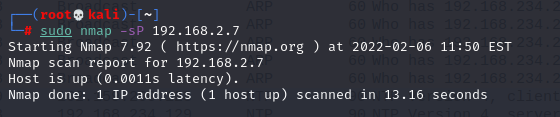
* To go for tcp syn scan, enter the following command

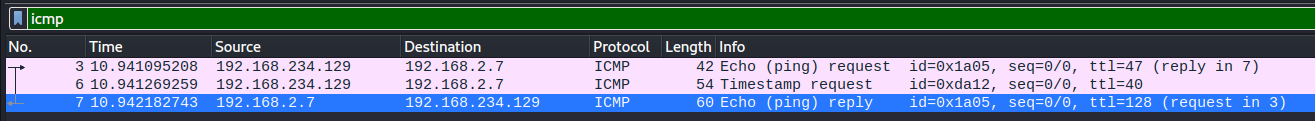


* Some of the packets captured in wireshark



* To scan for ping scan, enter the following command

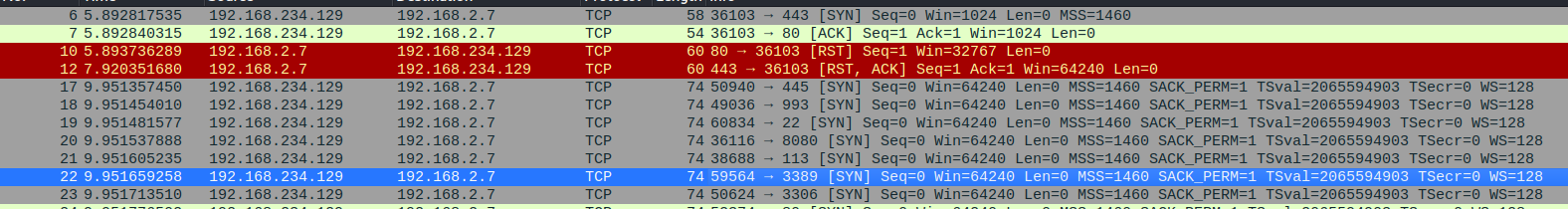




* For TCP Connect scan, enter the following command



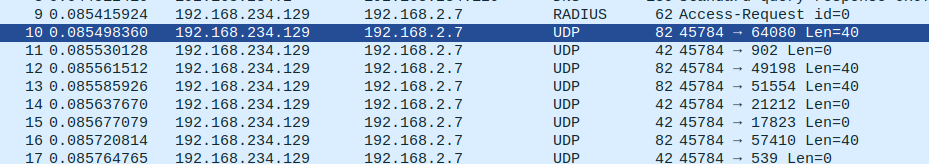
* Wireshark packets captured



* For UDP Scan, use the following command



* From the wireshark



**CONCLUSION:**

* By performing the above practical, I learnt about the basics of nmap and it’s functionalities