**Packet Sniffing and Spoofing**

**SEED ATTACK LABS**

**Report**

Williams

# **Overview**

Packet sniffing and spoofing are two important concepts in network security; they are two major threats in network communication. Being able to understand these two threats is essential for understanding security measures in networking. There are many packet-sniffing and spoofing tools, such as Wireshark, Tcpdump, Netwox, Scapy, etc. Some of these tools are widely used by security experts, as well as by attackers. Being able to use these tools is important for students, but what is more important for students in a network security course is to understand how these tools work, i.e., how packet sniffing, and spoofing are implemented in software.

The objective of this lab is two-fold: learning to use the tools and understanding the technologies under- lying these tools. For the second object, students will write simple sniffer and spoofing programs, and gain an in-depth understanding of the technical aspects of these programs. This lab covers the following topics:   
• Scapy  
• Sniffing using the pcap library   
• Raw socket

# **Lab Task Set 1: Using Tools to Sniff and Spoof Packets**

A screenshot of a cell phone

Description automatically generated

# **Task 1.1: Sniffing Packets**

## **Task 1.1A**

## ***sudo python sniffer.py***

A screenshot of a cell phone

Description automatically generated

## ***python sniffer.py***

A screenshot of a cell phone

Description automatically generated

## **Task 1.1B**

## ***Capture only the ICMP packet***

A screenshot of a cell phone

Description automatically generated

## ***Capture any TCP packet that comes from a particular IP and with a destination port number 23.***

A screenshot of a computer

Description automatically generated

## ***Capture packets comes from or to go to a particular subnet. You can pick any subnet, such as 128.230.0.0/16; you should not pick the subnet that your VM is attached to.***

A screenshot of a computer

Description automatically generated

# **Task 1.2: Spoofing ICMP Packets**

A screenshot of a cell phone

Description automatically generated

**Observation:** In the screenshots above, we can see that the attacker sends a spoofed ICMP request to a host and the host sends back an ICMP reply which is evident in the Wireshark capture.

**Explanation:** The attacker creates an ICMP packet with source address as 99.99.99.99 and sends the request.

# **Task 1.3: Traceroute**

A screenshot of a computer

Description automatically generated

## **Explanation:**

1 - this is IP address of the first router.

\* \* \*

9 - this is IP address of the ninth router.

# **Task 1.4: Sniffing and-then Spoofing**

A screenshot of a cell phone

Description automatically generated

## **Observation:**

User pings a host 8.8.8.8 on the network, the attacker sniffs the ICMP request, immediately spoofs the ICMP reply to the source of the ICMP request. The user receives the ICMP reply from the attacker.

## **Explanation:**

Snoofing is sniffing for the request and immediately sending the reply. The user pings a host 8.8.8.8 the attacker on VM A receives the ICMP packet using pcap which listens to traffic (promiscuous mode on), spoofs an ICMP reply using raw socket by replacing the source ip as the destination ip and the destination ip as the source ip. The fields in the ip header and the icmp header are spoofed by the attacker. When the reply is sent to the User, it seems like he gets a normal reply from the host he pings to.