Network Traffic Analysis and Attack Data

LAN-Based Attacks (DoS, SSL Stripping, ARP Poisoning)

Karthikeyan G
Roll Number: CB.SC.P2CYS24008

December 29, 2024

Introduction

This report aims to analyze and present the network traffic data related to three LAN-based attacks—Denial of Service (DoS), SSL Stripping, and ARP Poisoning—captured using Wireshark. For each attack, we will provide an overview, describe the simulation method used, and include screenshots of the data and analysis captured by Wireshark. Understanding these attacks helps in fortifying networks against similar threats in real-world scenarios.

Attack 1: Denial of Service (DoS)

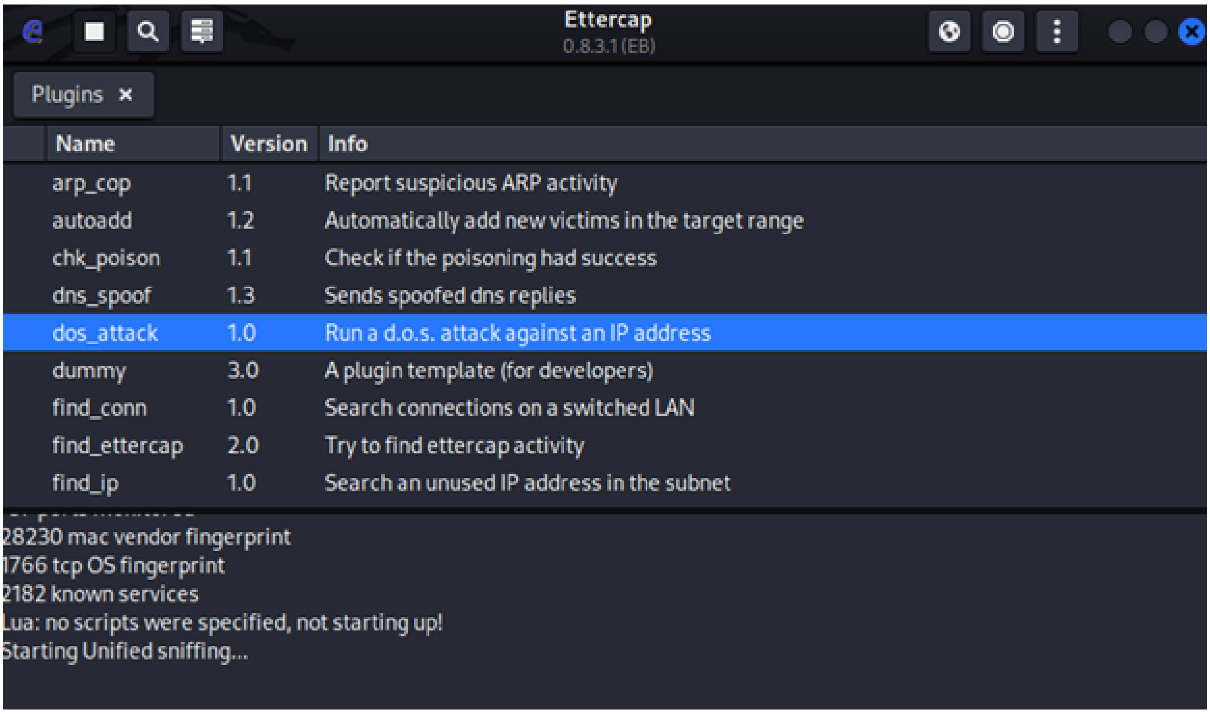
Overview

A Denial of Service (DoS) attack targets a network, server, or machine by flooding it with excessive traffic, which causes resource exhaustion and makes the system unavailable for legitimate users. The attack can result in downtime, loss of services, and a negative impact on business operations. DoS attacks are often used as a precursor to more sophisticated attacks, such as Distributed Denial of Service (DDoS) attacks.

Simulation Method

The DoS attack was simulated using `Ettercap` to generate a massive volume of traffic directed at a specific machine on the LAN. During the attack, Wireshark was used to capture and analyze the network traffic. The simulation involved sending a large number of ICMP (ping) requests to the target machine, overwhelming its resources.

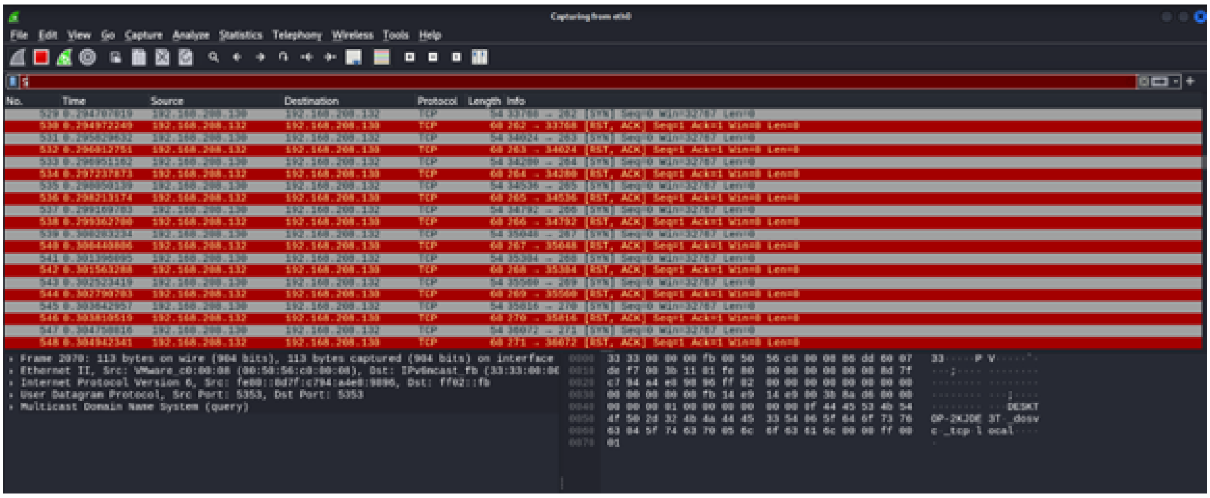
Screenshots



Name	Version	Info
arp_cop	1.1	Report suspicious ARP activity
autoadd	1.2	Automatically add new victims in the target range
chk_poison	1.1	Check if the poisoning had success
dns_spoof	1.3	Sends spoofed dns replies
dos_attack	1.0	Run a d.o.s. attack against an IP address
dummy	3.0	A plugin template (for developers)
find_conn	1.0	Search connections on a switched LAN
find_ettercap	2.0	Try to find ettercap activity
find_ip	1.0	Search an unused IP address in the subnet

```
28230 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!
Starting Unified sniffing...
```

Figure 1: DoS initialization


Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
No.   Time          Source            Destination       Protocol  Length  Info
529 0.294707019  192.168.208.130  192.168.208.132  TCP  54 33788 → 262 [SYN] Seq=0 Win=32787 Len=0
530 0.294972249  192.168.208.132  192.168.208.130  TCP  60 262 → 33788 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
531 0.295829632  192.168.208.130  192.168.208.132  TCP  54 34024 → 263 [SYN] Seq=0 Win=32787 Len=0
532 0.296012751  192.168.208.132  192.168.208.130  TCP  60 263 → 34024 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
533 0.296951162  192.168.208.130  192.168.208.132  TCP  54 34280 → 264 [SYN] Seq=0 Win=32787 Len=0
534 0.297237873  192.168.208.132  192.168.208.130  TCP  60 264 → 34280 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
535 0.298050139  192.168.208.130  192.168.208.132  TCP  54 34536 → 265 [SYN] Seq=0 Win=32787 Len=0
536 0.298213174  192.168.208.132  192.168.208.130  TCP  60 265 → 34536 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
537 0.299169783  192.168.208.130  192.168.208.132  TCP  54 34792 → 266 [SYN] Seq=0 Win=32787 Len=0
538 0.299362700  192.168.208.132  192.168.208.130  TCP  60 266 → 34792 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
539 0.300283234  192.168.208.130  192.168.208.132  TCP  54 35048 → 267 [SYN] Seq=0 Win=32787 Len=0
540 0.300440886  192.168.208.132  192.168.208.130  TCP  60 267 → 35048 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
541 0.301396095  192.168.208.130  192.168.208.132  TCP  54 35304 → 268 [SYN] Seq=0 Win=32787 Len=0
542 0.301563288  192.168.208.132  192.168.208.130  TCP  60 268 → 35304 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
543 0.302523419  192.168.208.130  192.168.208.132  TCP  54 35560 → 269 [SYN] Seq=0 Win=32787 Len=0
544 0.302790703  192.168.208.132  192.168.208.130  TCP  60 269 → 35560 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
545 0.303642957  192.168.208.130  192.168.208.132  TCP  54 35816 → 270 [SYN] Seq=0 Win=32787 Len=0
546 0.303810519  192.168.208.132  192.168.208.130  TCP  60 270 → 35816 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
547 0.304750816  192.168.208.130  192.168.208.132  TCP  54 36072 → 271 [SYN] Seq=0 Win=32787 Len=0
548 0.304942341  192.168.208.132  192.168.208.130  TCP  60 271 → 36072 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

› Frame 2070: 113 bytes on wire (904 bits), 113 bytes captured (904 bits) on interface
› Ethernet II, Src: VMware_c0:00:08 (00:50:56:c0:00:08), Dst: IPv6mcast_fb (33:33:00:00
› Internet Protocol Version 6, Src: fe80::8d7f:c794:a4e8:9896, Dst: ff02::fb
› User Datagram Protocol, Src Port: 5353, Dst Port: 5353
› Multicast Domain Name System (query)

0000  33 33 00 00 00 fb 00 50  56 c0 00 08 86 dd 60 07   33····P V·····`·
0010  de f7 00 3b 11 01 fe 80  00 00 00 00 00 00 8d 7f   ···;···· ········
0020  c7 94 a4 e8 98 96 ff 02  00 00 00 00 00 00 00 00   ········ ········
0030  00 00 00 00 00 fb 14 e9  14 e9 00 3b 8a d6 00 00   ········ ···;····
0040  00 00 00 01 00 00 00 00  00 00 0f 44 45 53 4b 54   ········ ···DESKT
0050  4f 50 2d 32 4b 4a 44 45  33 54 06 5f 64 6f 73 76   OP-2KJDE 3T·_dosv
0060  63 04 5f 74 63 70 05 6c  6f 63 61 6c 00 00 ff 00   c·_tcp·l ocal····
0070  01                                                 ·
```

Figure 2: DoS traffic visualization in Wireshark

Attack 2: SSL Stripping

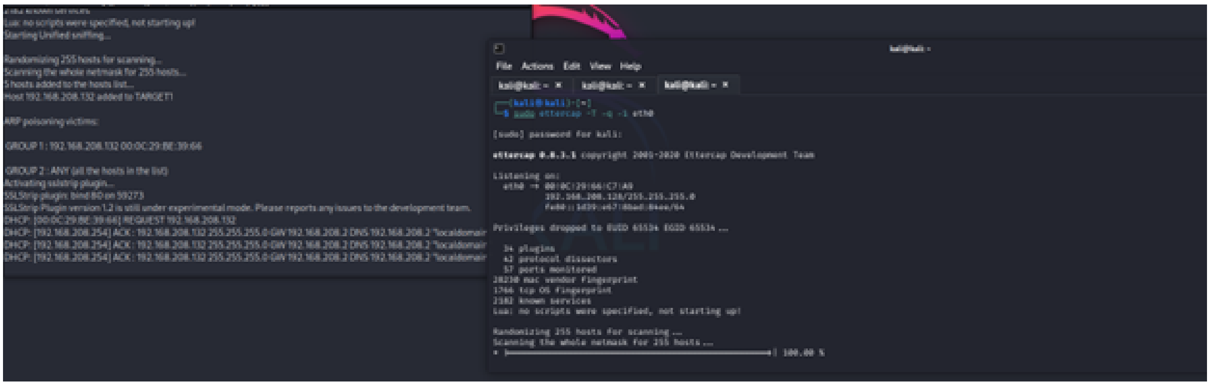
Overview

SSL Stripping is an attack that downgrades a secure HTTPS connection to an insecure HTTP connection. This allows attackers to intercept and manipulate data that would typically be encrypted, such as login credentials and sensitive information, compromising the integrity and confidentiality of communications. SSL Stripping is particularly dangerous in public Wi-Fi networks where users may not notice the downgrade.

Simulation Method

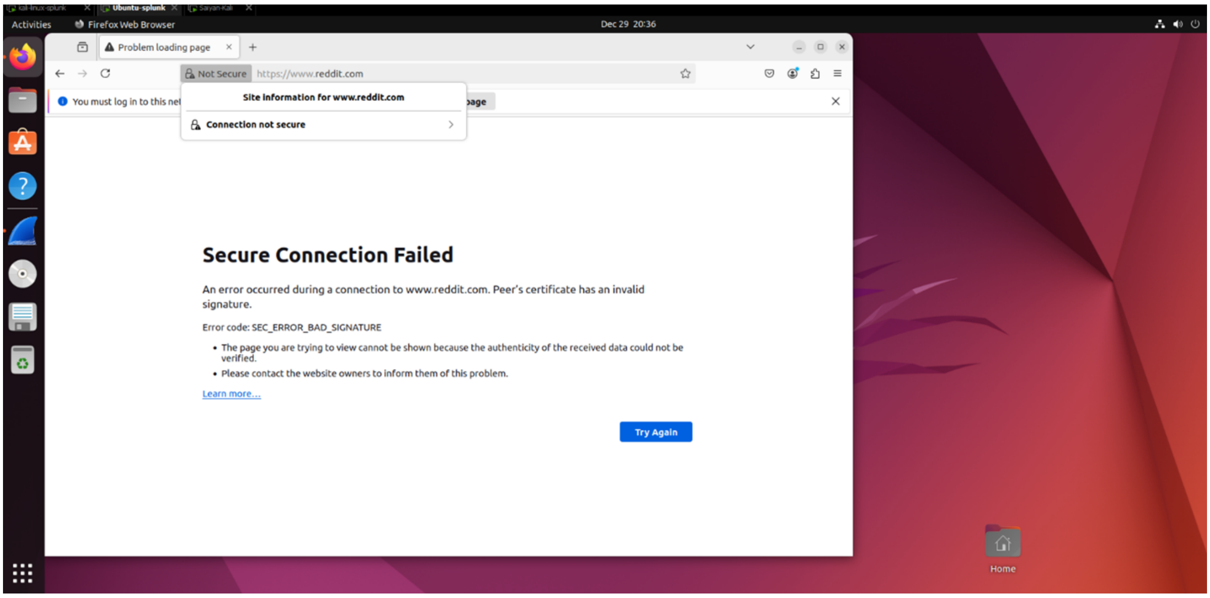
The SSL Stripping attack was simulated using `Ettercap`, which redirected HTTPS traffic to HTTP. This manipulation allowed Wireshark to capture the traffic, revealing the downgrading from HTTPS to HTTP and showing potential security vulnerabilities. The simulation involved intercepting traffic between a client and a server, modifying the HTTPS requests to HTTP, and observing the unencrypted data.

Screenshots


```
2182 known services
Lua: no scripts were specified, not starting up!
Starting Unified sniffing...

Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
5 hosts added to the hosts list...
Host 192.168.208.132 added to TARGET1

ARP poisoning victims:

 GROUP 1 : 192.168.208.132 00:0C:29:BE:39:66

 GROUP 2 : ANY (all the hosts in the list)
Activating sslstrip plugin...
SSLStrip plugin: bind 80 on 59273
SSLStrip Plugin version 1.2 is still under experimental mode. Please reports any issues to the development team.
DHCP: [00:0C:29:BE:39:66] REQUEST 192.168.208.132
DHCP: [192.168.208.254] ACK : 192.168.208.132 255.255.255.0 GW 192.168.208.2 DNS 192.168.208.2 "localdomain
DHCP: [192.168.208.254] ACK : 192.168.208.132 255.255.255.0 GW 192.168.208.2 DNS 192.168.208.2 "localdomain
DHCP: [192.168.208.254] ACK : 192.168.208.132 255.255.255.0 GW 192.168.208.2 DNS 192.168.208.2 "localdomain
```

```
┌──(kali㉿kali)-[~]
└─$ sudo ettercap -T -q -i eth0

[sudo] password for kali:

ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

Listening on:
  eth0 -> 00:0C:29:66:C7:A0
          192.168.208.128/255.255.255.0
          fe80::1d39:e67:8bad:84ee/64

Privileges dropped to EUID 65534 EGID 65534 ...

  34 plugins
  42 protocol dissectors
  57 ports monitored
28230 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!

Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
* |==================================================>| 100.00 %
```

Figure 3: SSL Stripping attack Initialization



Secure Connection Failed

An error occurred during a connection to www.reddit.com. Peer's certificate has an invalid signature.

Error code: SEC_ERROR_BAD_SIGNATURE

- The page you are trying to view cannot be shown because the authenticity of the received data could not be verified.
- Please contact the website owners to inform them of this problem.

Learn more…

Try Again

Figure 4: reddit page after SSL Stripping

Attack 3: ARP Poisoning

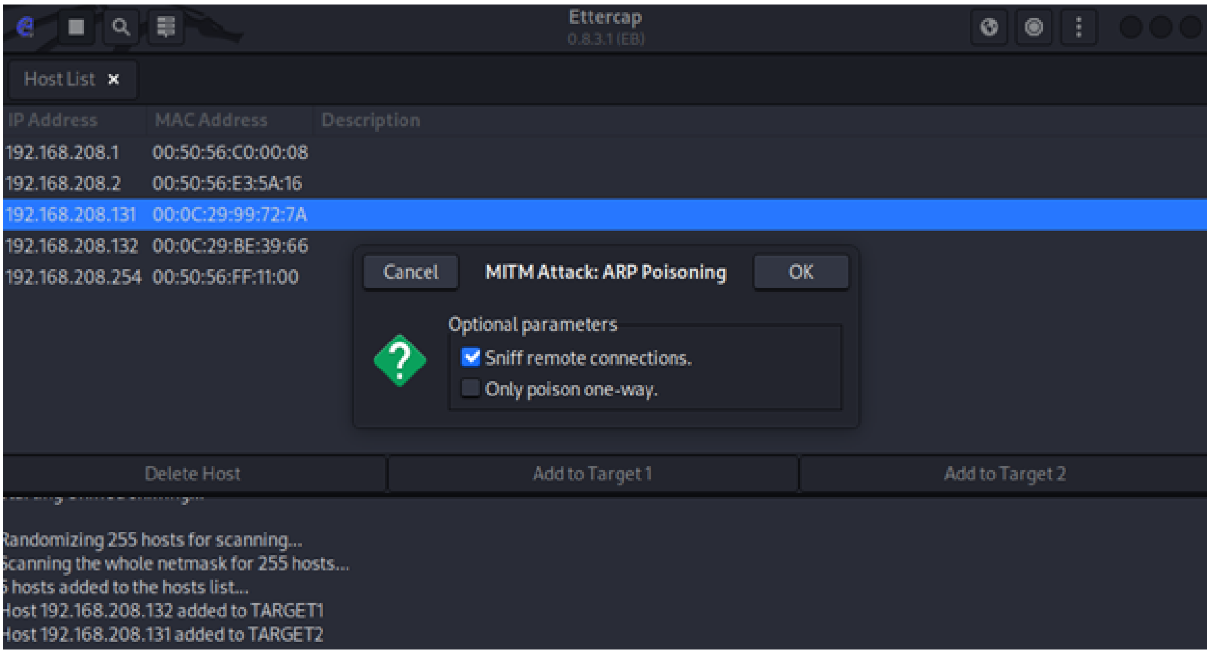
Overview

ARP Poisoning attacks exploit the Address Resolution Protocol (ARP) to associate the attacker's MAC address with the IP address of a legitimate machine in the network. This causes traffic meant for the legitimate machine to be intercepted by the attacker, enabling data theft, man-in-the-middle attacks, or denial of service. ARP Poisoning is a common technique used in local area networks (LANs) to eavesdrop on communications.

Simulation Method

ARP Poisoning was simulated using tools like `Ettercap` to manipulate ARP tables. Wireshark was used to capture the altered ARP entries and the resulting network traffic, revealing the attack's impact on communication between devices. The simulation involved sending spoofed ARP messages to the target machines, causing them to update their ARP tables with the attacker's MAC address.

Screenshots



IP Address	MAC Address	Description
192.168.208.1	00:50:56:C0:00:08	
192.168.208.2	00:50:56:E3:5A:16	
192.168.208.131	00:0C:29:99:72:7A	
192.168.208.132	00:0C:29:BE:39:66	
192.168.208.254	00:50:56:FF:11:00	

Cancel MITM Attack: ARP Poisoning OK

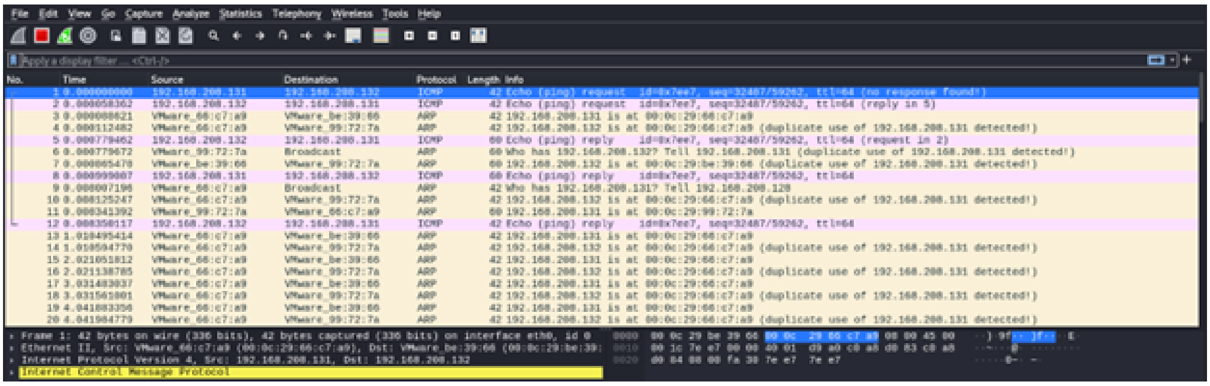
Optional parameters

☑ Sniff remote connections.

□ Only poison one-way.

```
Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
5 hosts added to the hosts list...
Host 192.168.208.132 added to TARGET1
Host 192.168.208.131 added to TARGET2
```

Figure 5: ARP Poisoning initialization


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time         Source             Destination        Protocol Length Info
   1 0.000000000  192.168.208.131    192.168.208.132    ICMP     42 Echo (ping) request  id=0x7ee7, seq=32487/59262, ttl=64 (no response found!)
   2 0.000058362  192.168.208.132    192.168.208.131    ICMP     42 Echo (ping) request  id=0x7ee7, seq=32487/59262, ttl=64 (reply in 5)
   3 0.000088621  VMware_66:c7:a9    VMware_be:39:66    ARP      42 192.168.208.131 is at 00:0c:29:66:c7:a9
   4 0.000112482  VMware_66:c7:a9    VMware_99:72:7a    ARP      42 192.168.208.132 is at 00:0c:29:66:c7:a9 (duplicate use of 192.168.208.131 detected!)
   5 0.000779462  192.168.208.132    192.168.208.131    ICMP     60 Echo (ping) reply    id=0x7ee7, seq=32487/59262, ttl=64 (request in 2)
   6 0.000779672  VMware_99:72:7a    Broadcast          ARP      60 Who has 192.168.208.132? Tell 192.168.208.131 (duplicate use of 192.168.208.131 detected!)
   7 0.000865478  VMware_be:39:66    VMware_99:72:7a    ARP      60 192.168.208.132 is at 00:0c:29:be:39:66 (duplicate use of 192.168.208.131 detected!)
   8 0.000999007  192.168.208.131    192.168.208.132    ICMP     60 Echo (ping) reply    id=0x7ee7, seq=32487/59262, ttl=64
   9 0.000007196  VMware_66:c7:a9    Broadcast          ARP      42 Who has 192.168.208.131? Tell 192.168.208.128
  10 0.000125247  VMware_66:c7:a9    VMware_99:72:7a    ARP      42 192.168.208.132 is at 00:0c:29:66:c7:a9 (duplicate use of 192.168.208.131 detected!)
  11 0.000341392  VMware_99:72:7a    VMware_66:c7:a9    ARP      60 192.168.208.131 is at 00:0c:29:99:72:7a
  12 0.000350117  192.168.208.132    192.168.208.131    ICMP     42 Echo (ping) reply    id=0x7ee7, seq=32487/59262, ttl=64
  13 1.010495414  VMware_66:c7:a9    VMware_be:39:66    ARP      42 192.168.208.131 is at 00:0c:29:66:c7:a9
  14 1.010594770  VMware_66:c7:a9    VMware_99:72:7a    ARP      42 192.168.208.132 is at 00:0c:29:66:c7:a9 (duplicate use of 192.168.208.131 detected!)
  15 2.021051812  VMware_66:c7:a9    VMware_be:39:66    ARP      42 192.168.208.131 is at 00:0c:29:66:c7:a9
  16 2.021138785  VMware_66:c7:a9    VMware_99:72:7a    ARP      42 192.168.208.132 is at 00:0c:29:66:c7:a9 (duplicate use of 192.168.208.131 detected!)
  17 3.031483037  VMware_66:c7:a9    VMware_be:39:66    ARP      42 192.168.208.131 is at 00:0c:29:66:c7:a9
  18 3.031561801  VMware_66:c7:a9    VMware_99:72:7a    ARP      42 192.168.208.132 is at 00:0c:29:66:c7:a9 (duplicate use of 192.168.208.131 detected!)
  19 4.041883356  VMware_66:c7:a9    VMware_be:39:66    ARP      42 192.168.208.131 is at 00:0c:29:66:c7:a9
  20 4.041984779  VMware_66:c7:a9    VMware_99:72:7a    ARP      42 192.168.208.132 is at 00:0c:29:66:c7:a9 (duplicate use of 192.168.208.131 detected!)
```

```
› Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
› Ethernet II, Src: VMware_66:c7:a9 (00:0c:29:66:c7:a9), Dst: VMware_be:39:66 (00:0c:29:be:39:
› Internet Protocol Version 4, Src: 192.168.208.131, Dst: 192.168.208.132
› Internet Control Message Protocol

0000  00 0c 29 be 39 66 00 0c  29 66 c7 a9 08 00 45 00
0010  00 1c 7e e7 00 00 40 01  d9 a0 c0 a8 d0 83 c0 a8
0020  d0 84 08 00 fa 38 7e e7  7e e7
```

Figure 6: Captured traffic during ARP Poisoning attack

Conclusion

In this report, we analyzed three significant LAN-based attacks: Denial of Service (DoS), SSL Stripping, and ARP Poisoning. Through simulation and Wireshark capture, we demonstrated the techniques used by attackers to disrupt network traffic and compromise communication security. Understanding these attacks helps in fortifying networks against similar threats in real-world scenarios. By implementing robust security measures, such as intrusion detection systems, secure protocols, and regular network monitoring, organizations can mitigate the risks associated with these attacks.