CSE406 Project 2025

Attack Tools Implementation

Design Report

ARP Cache Poisoning & Man-in-the-Middle Attack Tool

Group Members:

Name	Student ID
Sheikh Rahat Mahmud	2005048
Tusher Bhomik	2005046

Course: CSE406 - Computer Security

Department of Computer Science and Engineering

Bangladesh University of Engineering and Technology

Submission Date: Week 11, January 2025

Contents

1 Introduction

This design report presents a comprehensive approach to implementing an ARP (Address Resolution Protocol) cache poisoning attack combined with a man-in-the-middle (MITM) attack tool. The project aims to demonstrate the vulnerabilities inherent in the ARP protocol and how they can be exploited to intercept network communications in a controlled educational environment.

The tool will be implemented from scratch using Python, with custom packet crafting capabilities, avoiding the use of existing penetration testing frameworks. This approach ensures a deep understanding of the underlying protocols and attack mechanisms.

Disclaimer: This tool is developed solely for educational purposes and authorized security testing. It should never be used in unauthorized environments or for malicious activities.

2 Attack Definition and Network Topology

2.1 Attack Definition

ARP Cache Poisoning is a technique where an attacker sends falsified ARP messages onto a local area network. The goal is to associate the attacker's MAC address with the IP address of a legitimate computer or server on the network, causing any traffic meant for that IP address to be sent to the attacker instead.

Man-in-the-Middle (MITM) Attack is a form of eavesdropping where the attacker makes independent connections with the victims and relays messages between them, making them believe they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker.

2.2 Network Topology

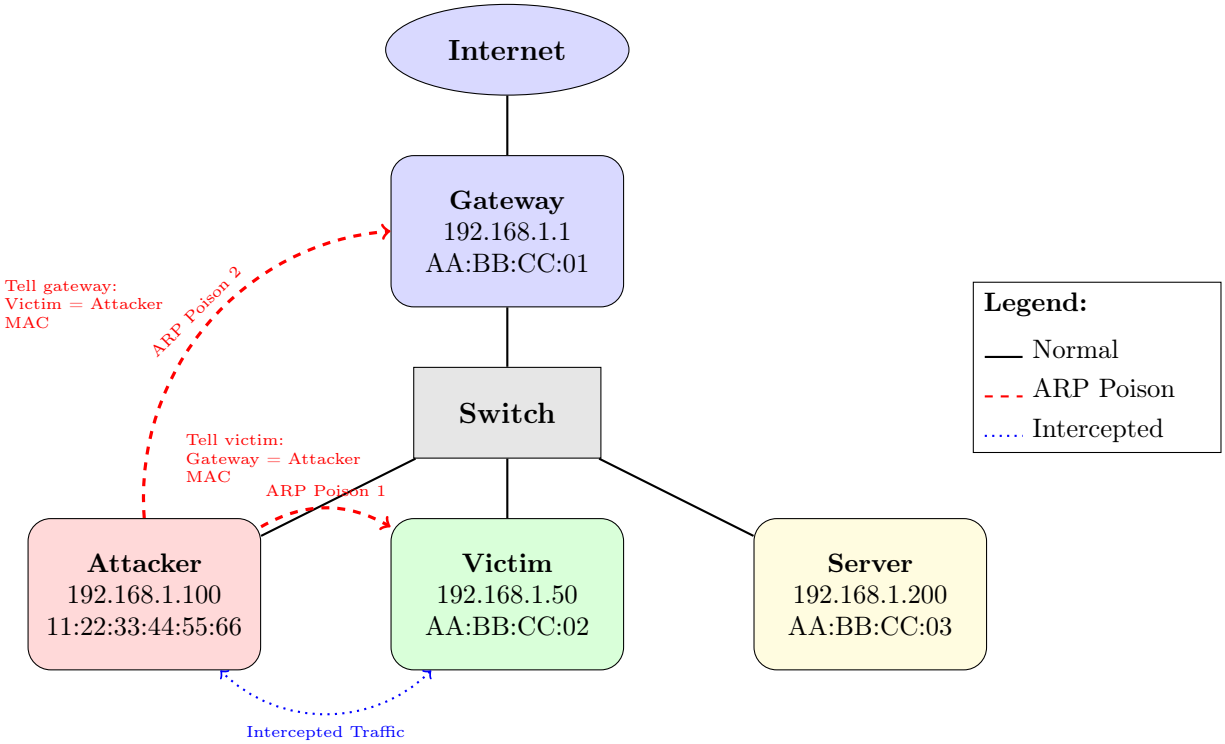

Figure 1: ARP Cache Poisoning Attack Network Topology

2.3 Network Components Description

- **Attacker PC:** The machine running our custom ARP poisoning tool (IP: 192.168.1.100)
- **Victim PC:** The target machine whose traffic will be intercepted (IP: 192.168.1.50)
- **Target Server:** The destination server that the victim is trying to communicate with (IP: 192.168.1.200)
- **Gateway/Router:** The network gateway providing internet access (IP: 192.168.1.1)
- **Switch/Hub:** Network infrastructure connecting all devices on the local segment

3 Timing Diagrams

3.1 Normal ARP Operation

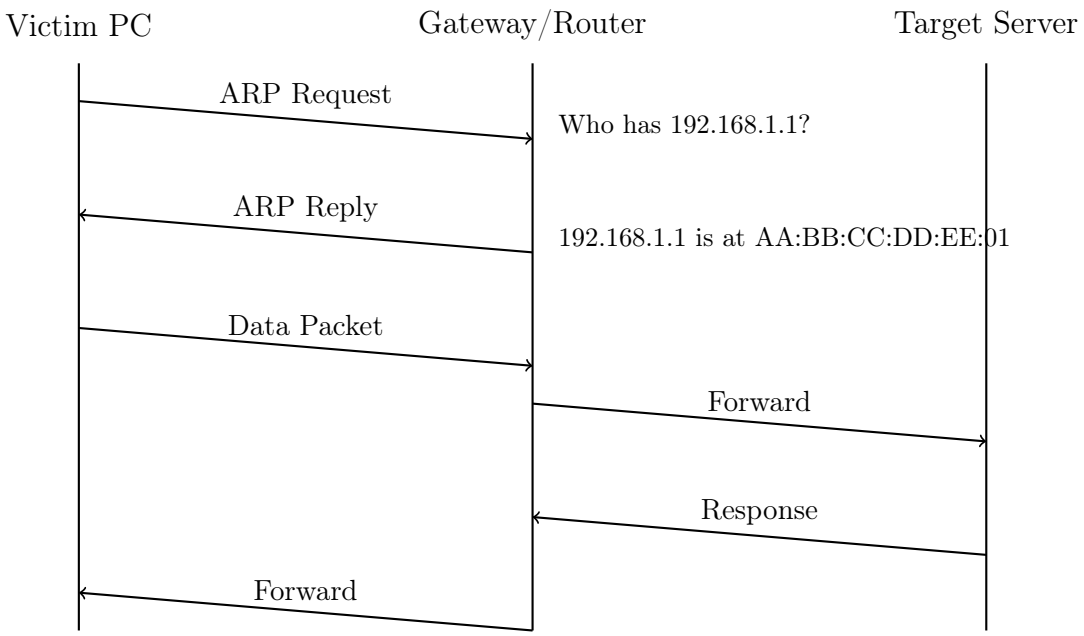

Figure 2: Normal ARP Operation Timing Diagram

3.2 ARP Cache Poisoning Attack Timing

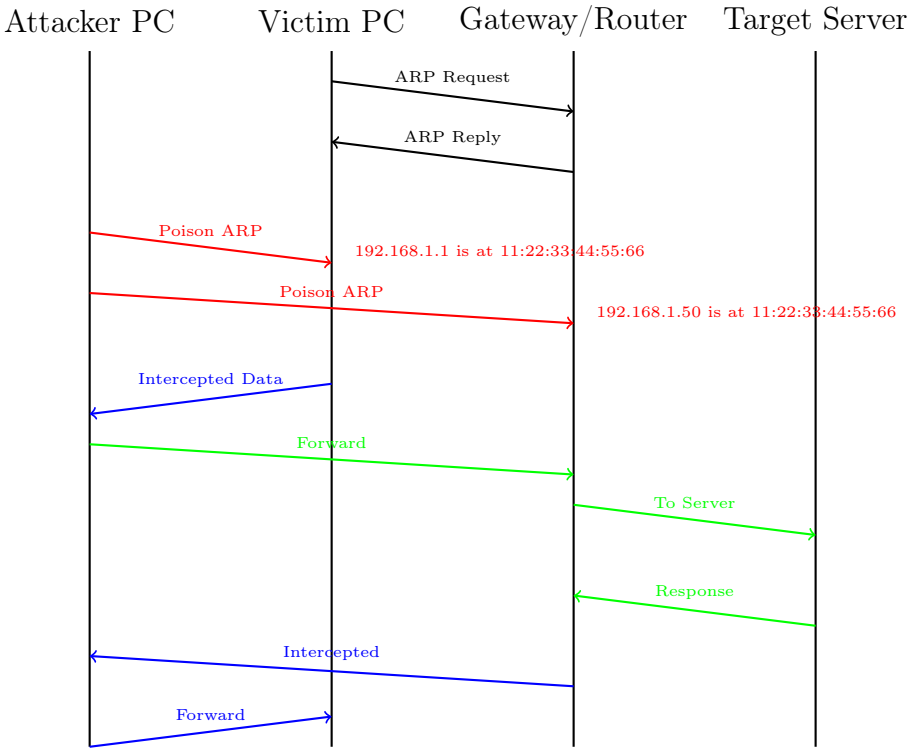

Figure 3: ARP Cache Poisoning Attack Timing Diagram

3.3 Attack Strategy Timeline

Table 1: Attack Phase Timeline

Phase	Duration	Activities
Network Discovery	0-10 seconds	Scan network for active hosts, identify gateway IP and MAC, select victim targets
Initial ARP Poisoning	10-15 seconds	Send malicious ARP replies to victim and gateway, verify ARP cache modification
Traffic Interception	15+ seconds	Start packet sniffing, intercept and log traffic, forward packets to maintain connectivity
Continuous Poisoning	Ongoing	Send poison packets every 2-3 seconds to maintain attack

4 Packet and Frame Details

4.1 Ethernet Frame Structure

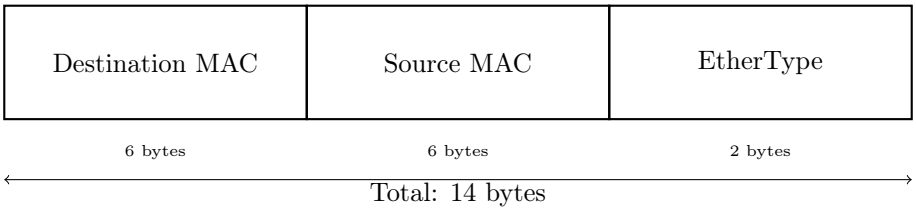

Figure 4: Ethernet Frame Header Structure

4.2 ARP Packet Structure

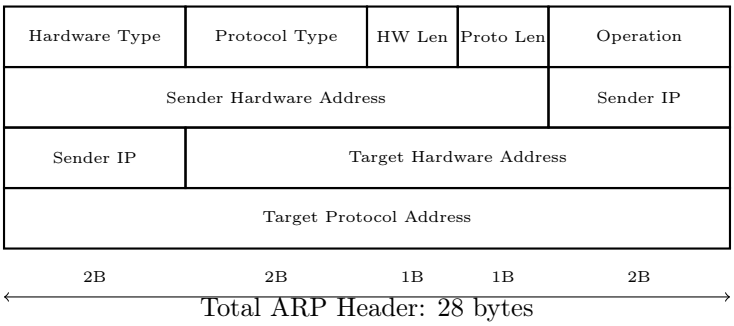

Figure 5: ARP Packet Header Structure

4.3 Packet Field Specifications

Table 2: ARP Packet Field Values

Field	Size	Normal Value	Attack Value
Hardware Type	2 bytes	0x0001 (Ethernet)	0x0001 (Ethernet)
Protocol Type	2 bytes	0x0800 (IPv4)	0x0800 (IPv4)
Hardware Length	1 byte	0x06	0x06
Protocol Length	1 byte	0x04	0x04
Operation	2 bytes	0x0002 (Reply)	0x0002 (Reply)
Sender MAC	6 bytes	Gateway MAC	Attacker MAC
Sender IP	4 bytes	Gateway IP	Gateway IP (Spoofed)
Target MAC	6 bytes	Victim MAC	Victim MAC
Target IP	4 bytes	Victim IP	Victim IP

4.4 Header Modifications for Attack

The key modifications made to create malicious ARP packets are:

1. **Sender MAC Address:** Changed from legitimate gateway MAC to attacker's MAC address
2. **Source MAC in Ethernet Header:** Modified to attacker's MAC address
3. **Gratuitous ARP:** Using unsolicited ARP replies to poison the cache
4. **Bidirectional Poisoning:** Creating separate packets for victim→gateway and gateway→victim poisoning

```
# Poison packet to victim (claiming to be gateway)
ethernet_dst = victim_mac            # Target: victim
ethernet_src = attacker_mac          # Source: attacker
arp_sender_mac = attacker_mac        # SPOOFED: claiming attacker is
    gateway
arp_sender_ip = gateway_ip           # SPOOFED: gateway's IP
arp_target_mac = victim_mac          # Victim's actual MAC
arp_target_ip = victim_ip            # Victim's actual IP
```

Listing 1: Packet Crafting Example

5 Design Justification

5.1 Technical Foundation

Our design leverages fundamental weaknesses in the ARP protocol:

- **No Authentication:** ARP has no built-in mechanism to verify the authenticity of ARP messages
- **Stateless Protocol:** ARP replies are accepted regardless of whether requests were made
- **Cache Update Policy:** Most systems automatically update their ARP cache upon receiving ARP replies
- **Trust Model:** Local network devices implicitly trust ARP messages from other devices