Cyber Project - ARP Poisoning Detection & Mitigation

Or Shani, Ori Katz

Under guidance of Fraier Sorana,

and supervision of Mitrany Roy

Technion – Israel Institute of Technology

Table of Contents

Table of Figures

1. Abstract

1.1. Motivation

ARP poisoning (or ARP spoofing) is a type of cyberattack that involves sending malicious ARP packets to one of the default gateways on a Local Area Network (LAN) in order to change the IP to MAC address pairings (ARP cache) of victim nodes. The ARP Protocol translates IP addresses into MAC addresses and was not designed with security in mind. Therefore, for an attacker inside - or with access to - the LAN, ARP poisoning attacks are extremely easy to carry out.

Although a simple attack itself, successfully completing it allows the attacker to implement more sophisticated and harmful attacks, such as “Man in the Middle”, and consecutively to read\alter private communications and to impersonate the victims. ARP poisoning attacks are a known and common problem for network security, plaguing LANs all over the world. Although many solutions exist today, ARP poisoning attacks are still a concern and new ways of detection and protection are still developed.

1.2. Objectives

In this project, we attempt to study the subject in depth as well as examine existing solutions, with the main objective of implementing our own mechanism for detecting and protecting a client from such an attack. The prescribed requirements also specified certain capabilities the solution should include: provide protection at end-station level, traffic listening mode, locked mode, enable populating the ARP cache with a known static list, and manage blacklisting hosts.

1.3. Conclusions

Our solution was implemented in a virtual environment we modeled as a network with a virtual subnet and traffic passing through a switch. The attack was implemented using “Bettercap” on a “Kali” machine. In our work we used several tools, e.g: Linux shell scripts, Linux system calls, Linux net libraries (as arping, nftable, etc.), python, scapy, wireshark and more.

Through our research we were exposed to many networking concepts and principles as well as different protection methods and strategies. This research led us to specify more requirements and capabilities to include in the solution, pushing it beyond the familiar basics. The process by which we came by the selected solution, is specified in this paper.

In the end, after examining its capabilities and testing its performance, we found that our implemented solution surpassed preliminary expectations, as on top of being able to accurately detect ARP-poisoning attacks and defend against them, it can even fight back.

The implemented features work as desired, including some novel ones, and can be used as part or even a basis of a much larger defense suite. The solution can be further developed to incorporate micro-segmentation security measures, and use each host and other network systems to create a variation of a zero-trust network.