**DOMAIN NAME SERVER CACHE POISONING**

For

**NETWORK THREATS AND ATTACKS LABORATORY**

**B. E. Computer Engineering**

By

**Vidish Naik 60004140067**

**Pratik Parekh 60004140076**

**Vaishnavi Pendse 60004140081**

**Faculty-In-Charge**

Prof. Pranit Bari

Assistant Professor

**PAGE 1**

**Name of the Group Members with SAP Id with Batch No:**

Vidish Naik – 60004140067 B-1

Pratik Parekh – 60004140076 B-1

Vaishnavi Pendse - 60004140081 B-1

**Class with Division: Bachelor of Engineering (BE) –** B

**Name of the Supervisor/Guide:** Prof. Pranit Bari

**Title of Report:** Domain Name Server Cache Poisoning

**Field of Project:** Networking

**Area of the project:** Attacks

**INDEX:**

1. **INTRODUCTION**
2. **AIM & OBJECTIVE**
3. **THEORY OF OPERATION**
4. **PLATFORM USED**
5. **RESULT**
6. **CONCLUSIONS**
7. **REFERNCES**
8. **INTRODUCTION:**

The internet relies on the Domain Name System (DNS) to maintain an index of all public websites and their corresponding IP addresses. You can think of it like a phone book.

With a phone book, we don't have to memorize everyone's phone number, which is the only way phones can communicate: with a number. In the same way, DNS is used so we can avoid having to memorize every website's IP address, which is the only way network equipment can communicate with websites.

You type in a URL like *www.google.com* and your web browser asks your router for the IP address. The router has a DNS server address stored, so it asks the DNS server for the IP address of that hostname. The DNS server finds the IP address that belongs to *www.google.com* and then is able to understand what website you're asking for, after which your browser can then load the appropriate page.

The DNS cache attempts to speed up the process even more by handling the name resolution of recently visited addresses before the request is sent out to the internet.

1. **AIM AND OBJECTIVE:**

To perform a DNS cache poisoning attack and then apply preventive measures. The objective of this project is to avoid the poisoning of the DNS cache. This is done by encrypting the DNS cache file so that no one other than the DNS server can read the file.

1. **THEORY OF OPERATION**

A connection is established between the client and the DNS server using sockets. Further, the client takes the website input from the user and passes it to the DNS server. The server searches for the IP address of the requested website in its cache. If the IP address is found, it is returned to the user. If the IP address is not found, a request is made to other DNS servers and the IP address is obtained. This IP address is added to the cache for further reference, and then forwarded to the user.

When an attacker attacks the cache of the DNS server, he/she changes the IP addresses of some or all websites and redirects it to the malicious server.

In order to prevent such attacks, the DNS server encrypts its DNS cache. It encrypts both the website and its IP address. Now, even if the attacker attacks and changes the cache of the server, the DNS server implements checks to make sure that the decrypted text is a proper IP address.

1. **PLATFORM USED**

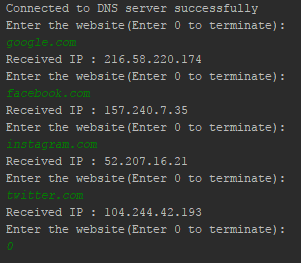
Operating System – Windows 10

Technology – Java

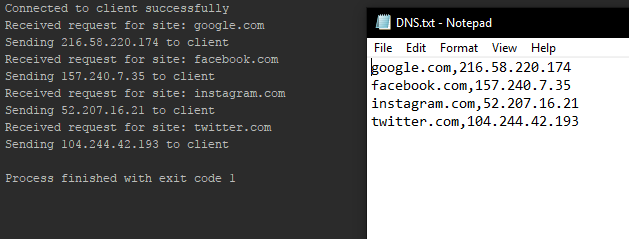
IDE – Intellij Idea

1. **RESULT**

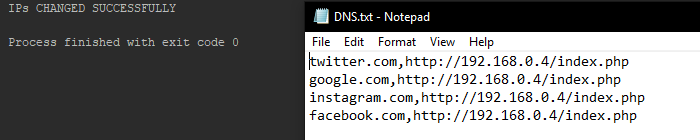
Client side:



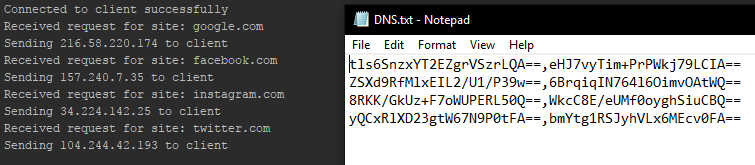
Server side and Cache file:



Attacker changes IP:



DNS server encrypted cache:



1. **CONCLUSION**

Thus, we have successfully implemented Domain Name Server Cache Poisoning.

1. **REFERENCES**

https://www.lifewire.com/what-is-a-dns-cache-817514

https://www.howtogeek.com/122845/htg-explains-what-is-dns/