بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Ministry of Higher Education and Scientific Research
University Kasdi Merbah Ouargla
Faculty The New Technologies of Information and Communication
Department of Computer Science and Information Technologies

Presentation of the project:

Implement an application detect and protect against MitM attack

Presented by :

Gherbi Borhan eddin
Beggari Ahmed said

supervised by :

Mr.Khaldi Amin

Session 2018/2019

PLAN

- Introduction
- Mechanism of attack
- Mechanism of protection
- Project Implementation
- Protect and detect methods
- Experimentation and Results
- Conclusion

Introduction

Information security has become a very important and indispensable element in people's life and in the work of different institutions, and it is now necessary to provide protection against various threats.

Among these well known threats is a kind of attack called the man in the middle attack which is considered one of the most dangerous attacks. Used by hacker in local networks allows the attacker to put himself between users and the router where he can intercept traffic and spy on various communications and control it,redirect, and steal sensitive information, ..etc

The goal of our project is to achieve an application that enables us to protect against these attacks so that detect the source, type of attack and cancel it if possible.

Mechanism of attack

Why exactly man-in-the-middle attack?

Man-in-the-middle attack statistics :

- **95%** of **HTTPS** servers vulnerable to **MitM** According to Netcraft, **MitM** attacks were thought to pose a threat to **95%** of **HTTPS** servers in 2016.
- **MitM** attacks were involved in **35%** of **exploitation** More than one-third of exploitation of inadvertent weaknesses involved **MitM** attacks, according to IBM's X-Force Threat Intelligence Index 2018.

Mechanism of attack

MitM attack

Man-in-the-middle (**MitM**) attacks happen at different levels and forms. However, its basic concept requires three key players : the **victim**, the **entity** which victim is trying to contact, and **the man in the middle**. The victim can be any user trying to access a website or a web application (the entity). On any typical connection, The “man in the middle” inserts himself between the connection of the user and the website server .

Impact of attack :

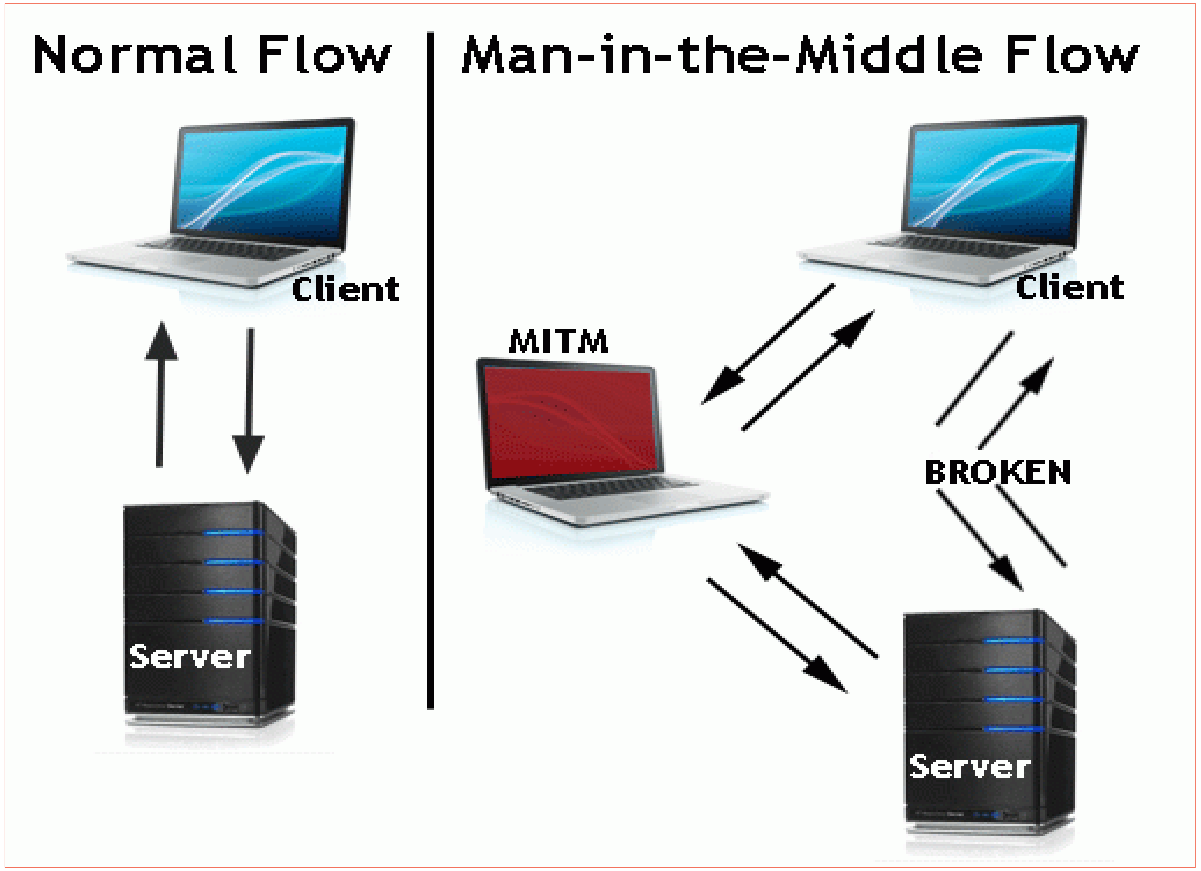
- Intercept network traffic and spy on different communications.
- Steal credential authentication(username and password, tokens, credit cards, ..etc).
- Control ,redirect and manipulating users traffic.
- scan if there Vulnerabilities infect system devices inside LAN and exploit them

Mechanism of attack

ARP poisoning attack

The mechanisms used to execute network **MitM** attacks include :

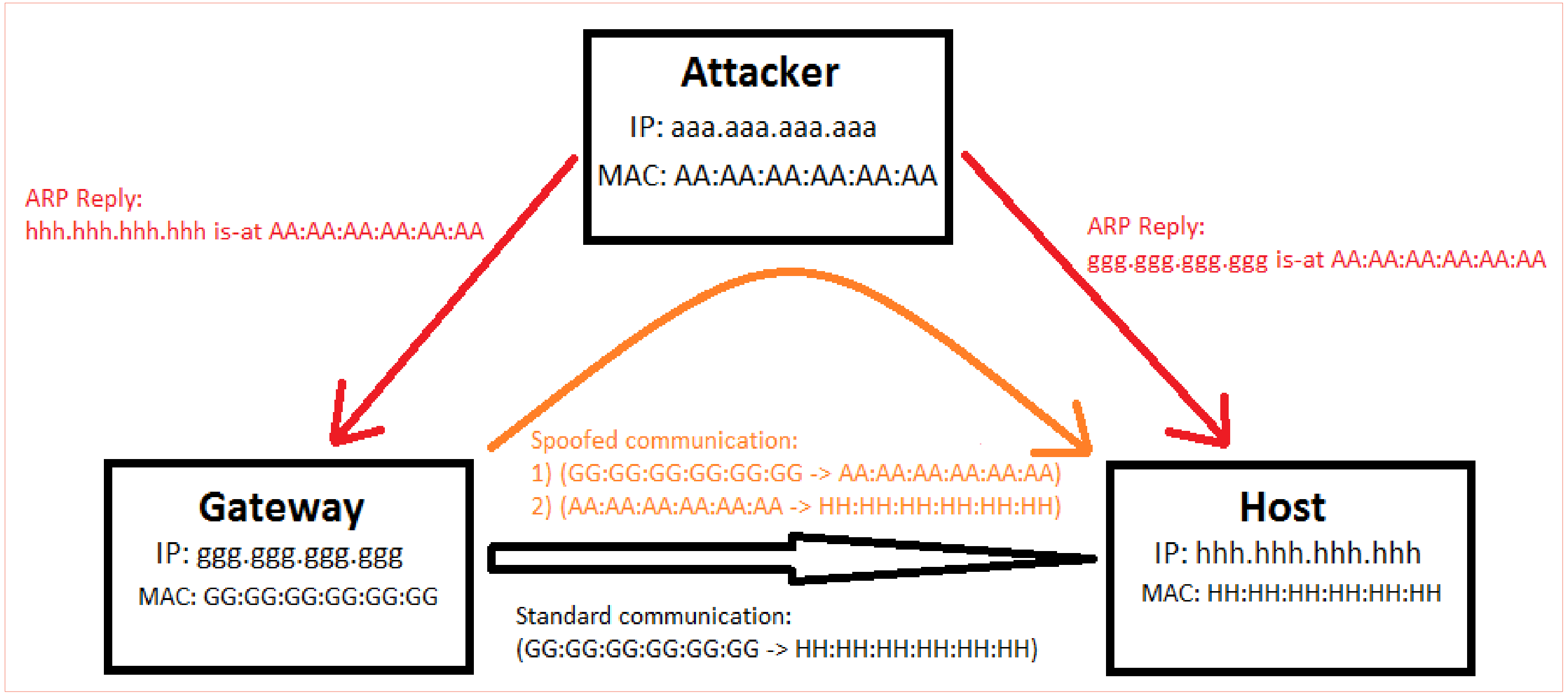
- **ARP** poisoning attack.
- **DNS** spoofing attack.
- **DHCP** spoofing attack.



Mechanism of attack

ARP poisoning attack

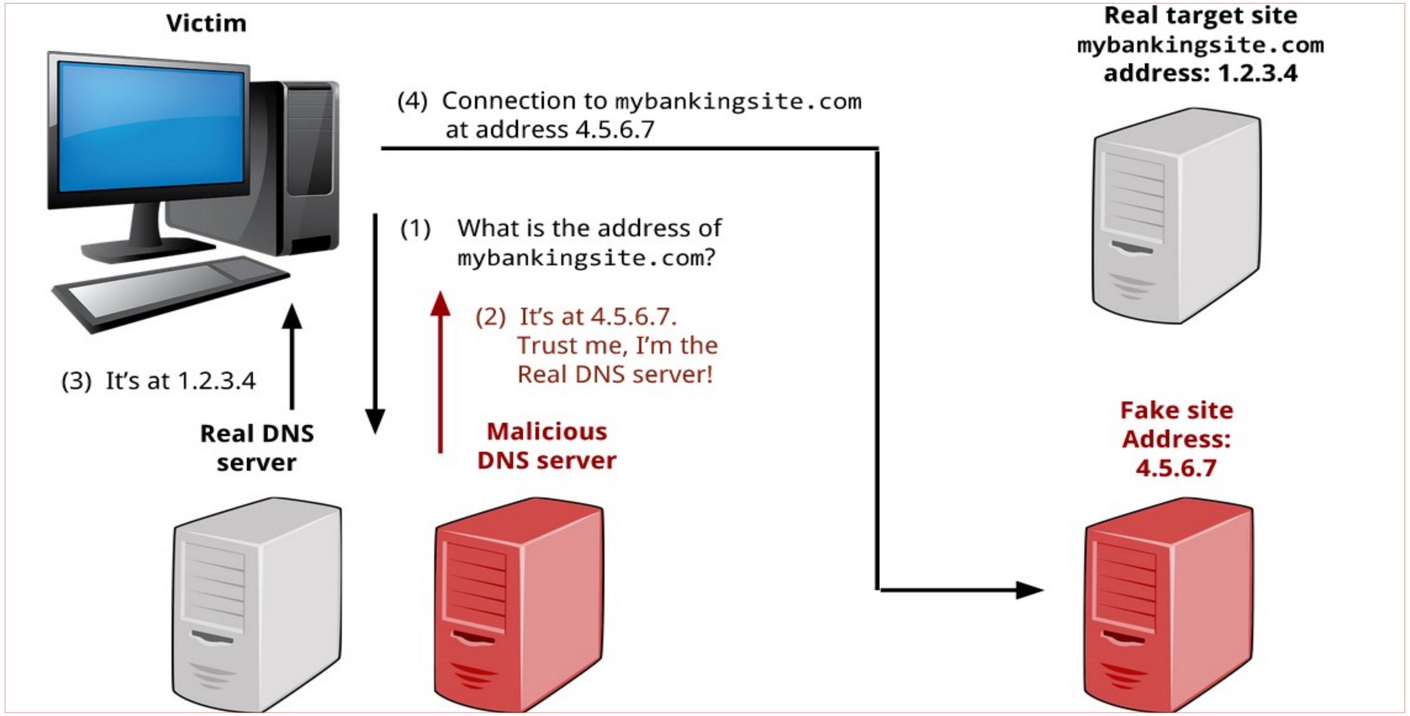
In order to execute the attack on the two victims, the attacker sends spoofed **ARP** packets to the local **victim** and the local **gateway**, to **ARP** poison these two caches. This way, all communication from the target node to internet (via the gateway) passes through the attacker.



Mechanism of attack

DNS spoofing attack

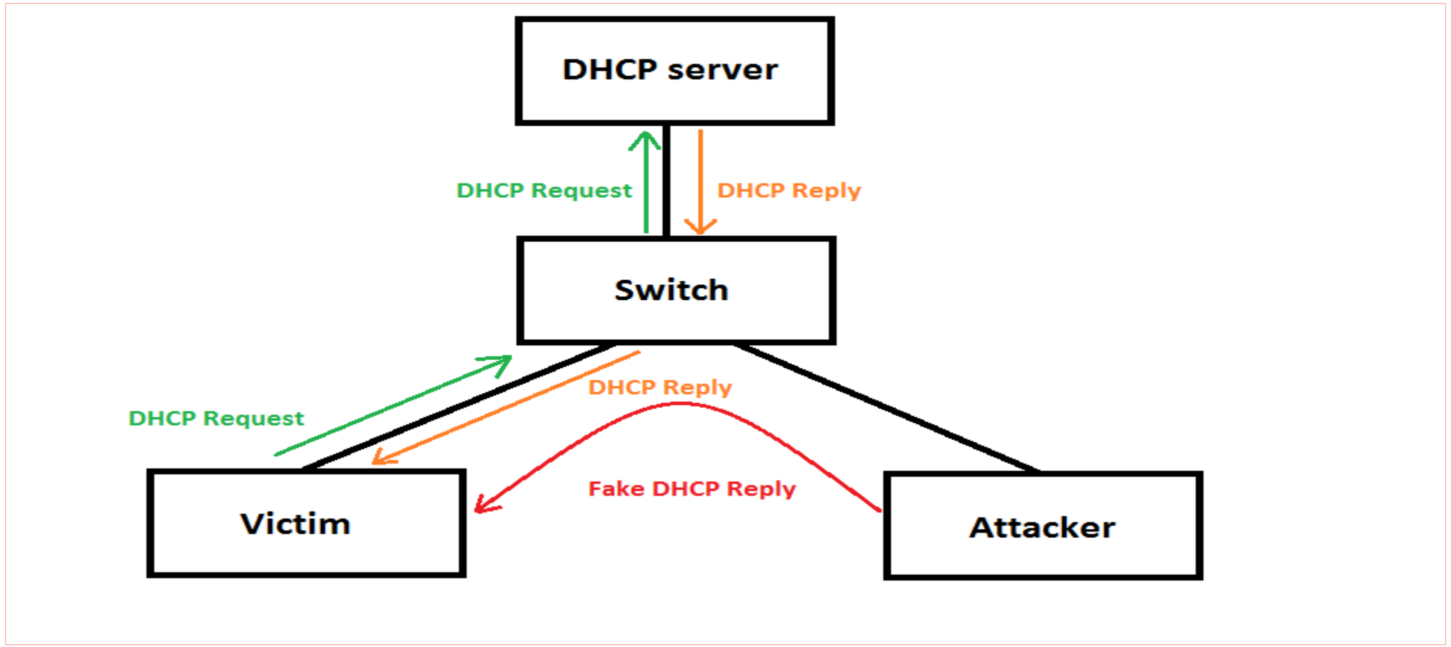
A **DNS** server provides the **IP** address associated with a given **domain/host** name. It is possible for an attacker to replace a valid domain name's **IP** with an attacker-controlled **IP** address.



Mechanism of attack

DHCP spoofing attack

A **DHCP** server provides **IP** information such as the default gateway **IP** address to network nodes that join a local network. An attacker can pose as a **DHCP** server and send forged **DHCP** acknowledgments to any connecting nodes.

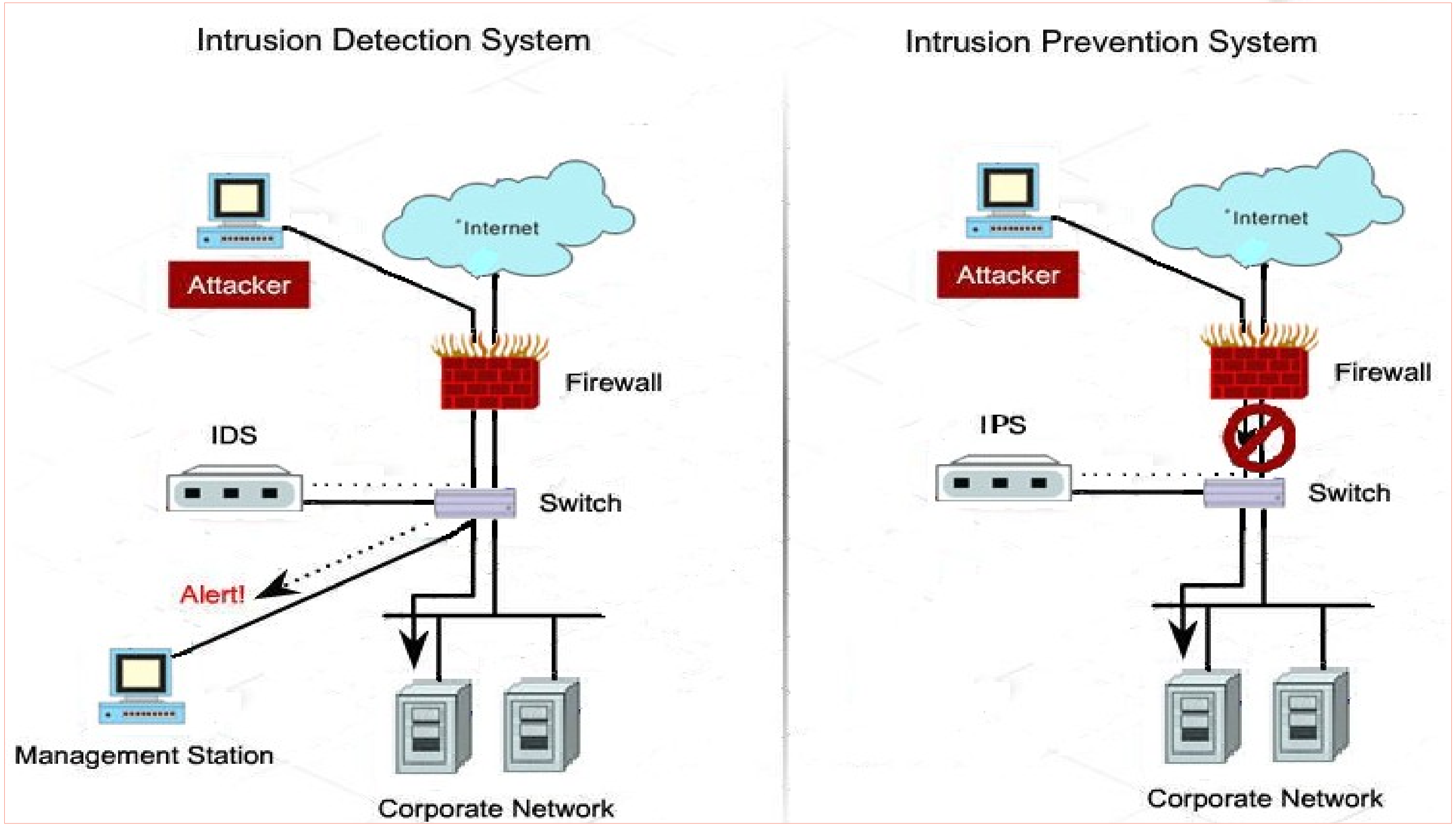


Mechanism of protection

IDS/IPS system

Intrusion Detection Systems **IDS** and Intrusion Prevention Systems **IPS** are both parts of the network infrastructure. **IDS/IPS** compare network packets to a Cyber threat database containing known signatures of Cyber attacks and flag any matching packets. The main difference between them is that **IDS** is a monitoring system, while **IPS** is a control system. **IDS** doesn’t alter the network packets in any way, whereas **IPS** prevents the packet from delivery based on the contents of the packet, much like how a firewall prevents traffic by **IP** address.

Mechanism of protection



Mechanism of protection

There are two major distinct families of **IDS** :

- Network Based IDS/IPS (**N-IDS/IPS**): they provide security at the network level.
- Host Based IDS/IPS (**H-IDS/IPS**): they provide security at the host level.

Detection techniques :

- Checking the protocol stack (packet inspection).
- Verification of application protocols (protocol analysis).
- Recognition of pattern matching attacks (signature search)

Mechanism of protection

Alert methods

- Reconfiguration of third-party equipment (firewall, ACL on routers).
- Sending a SNMP trap to a third-party hypervisor.
- Sending an email to one or more users.
- Log attack.
- Backup of suspicious packets.
- Starting an application.
- Sending a ResetKill.
- Visual notification of the alert.

Project Implementation

Python programming language

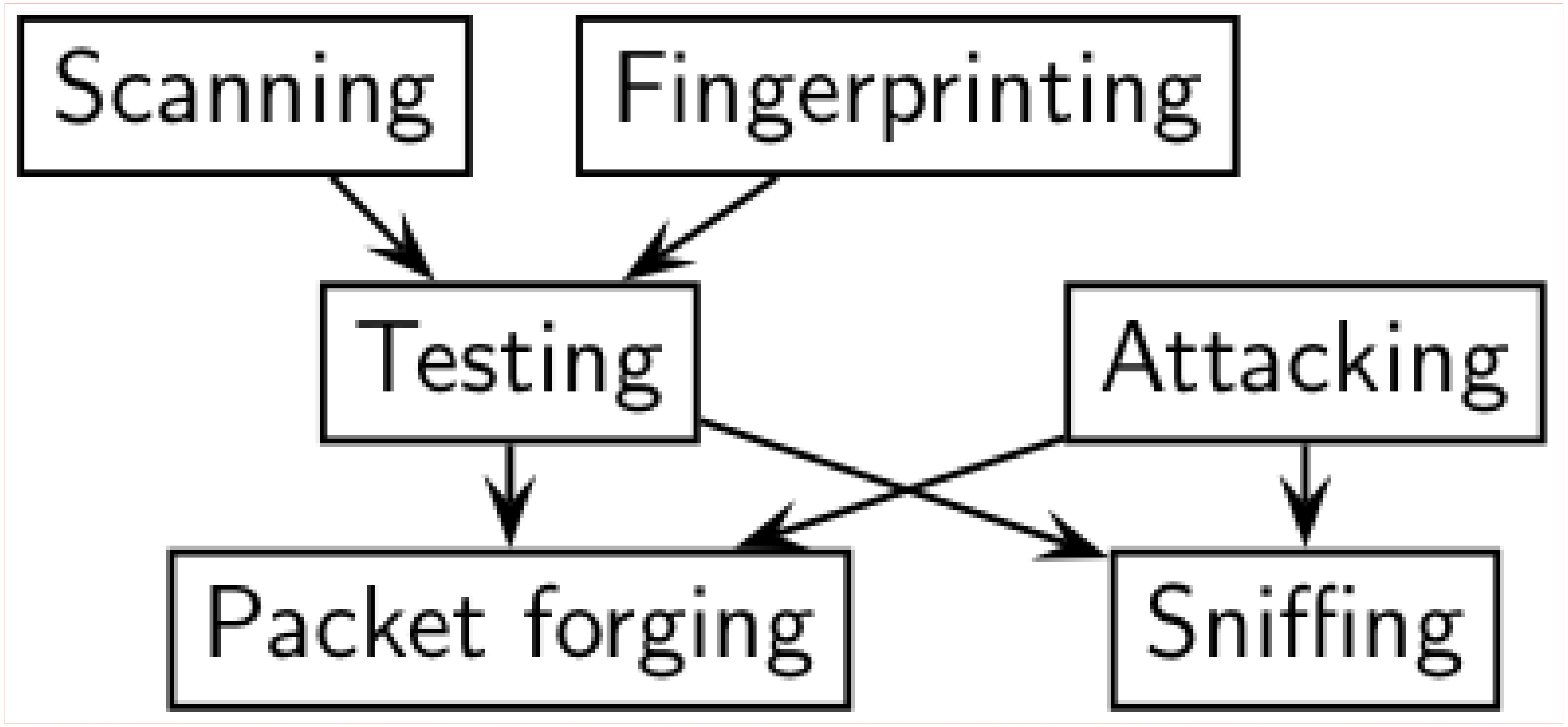
Python is an interpreted, object-oriented, high-level programming language with dynamic semantic. In (2017,2018 ,2019) Python is the most widely used in the world.

Used in artificial intelligence and by big company such as **NASA** and **IBM,Amazon**,etc..

Project Implementation

Scapy python programming

Scaby is a powerful python program for developing tools that work in the network, because it has many features that helped us to make our application more flexible and easy to control.



Project Implementation

Wireshark

Wireshark is the world’s foremost and widely-used network protocol analyzer.

- Deep inspection of hundreds of protocols, with more being added all the time.
- Live capture and offline analysis.
- The most powerful display filters in the industry.

Project Implementation

Kali linux test machine

Kali Linux is a popular penetration testing environment built on Debian distribution ,Contains hundreds of powerful tools such as Metasploit-framework and wireshark,maltego,Nmap,...ect

- Ettercap-framework.
- Arpspoof.
- Enable forwarding mode in kernel machine.
- Iptables firewall.

KALI LINUX
"the quieter you become, the more you are able to hear"

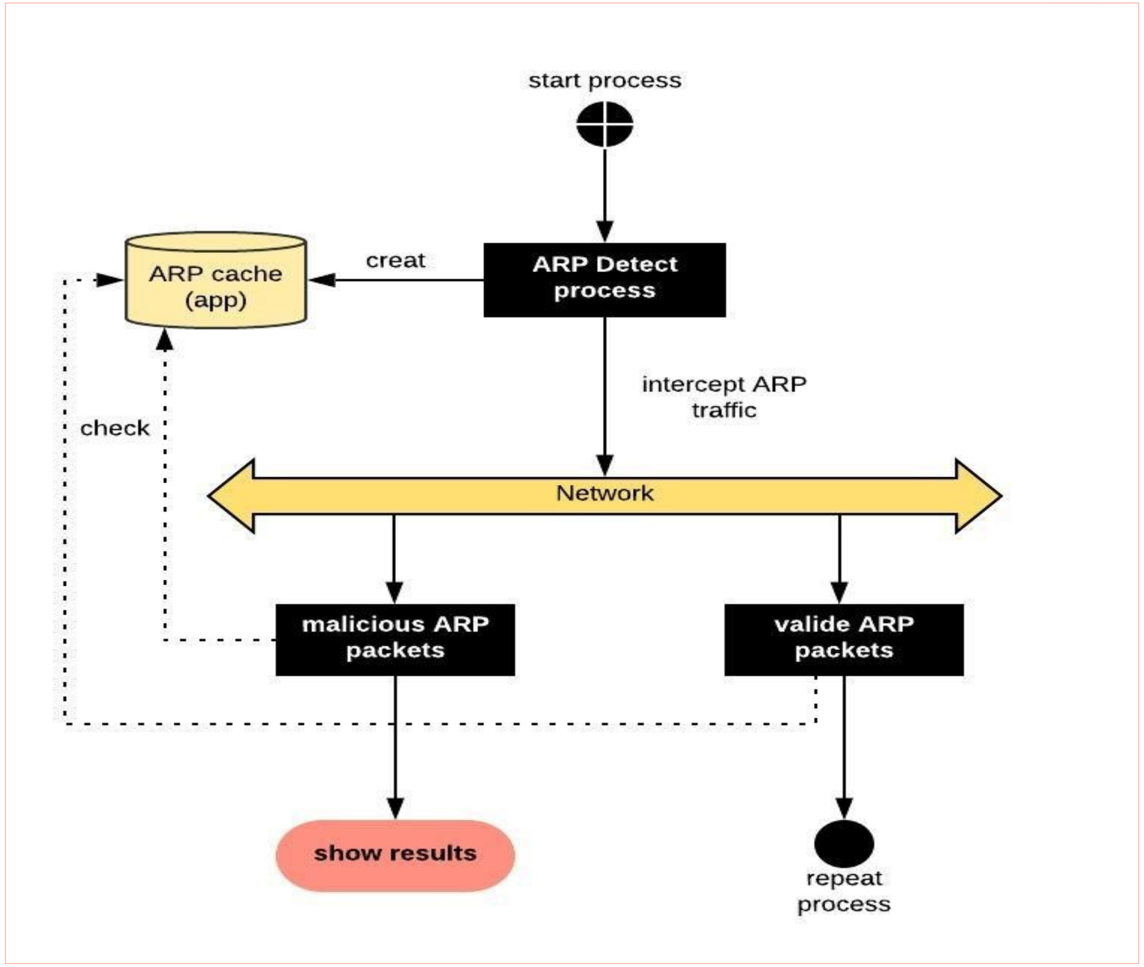
debian
GNU/Linux

ettercap

Protect and detect methods

ARP Poisoning Detect

Is an algorithm designed to detect Spoof **ARP** packets and scan network. The following Diagram is activity diagram that describe **ARP-Detect** method.

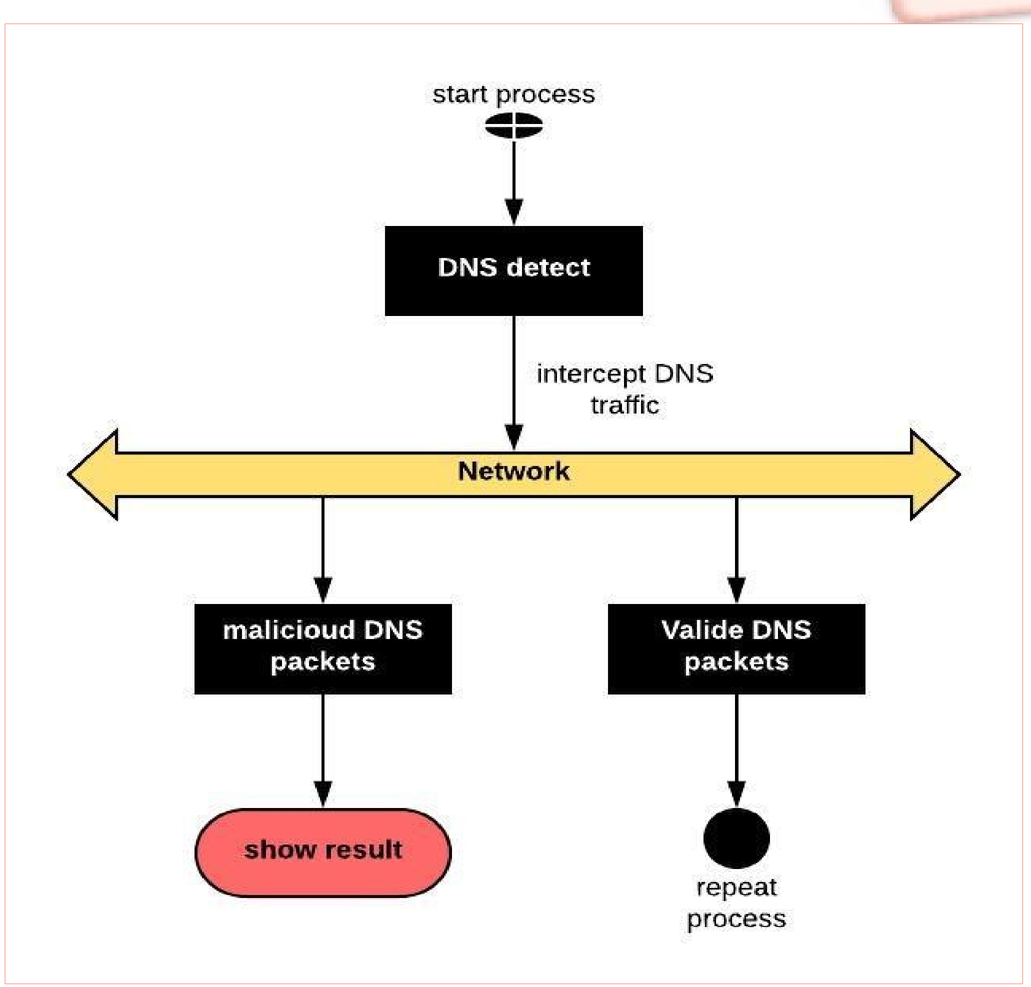


Protect and detect methods

DNS Spoof Detect

The algorithm captures **DNS** packets, and then inspect them through mechanisms to detect if there are malicious **DNS** response packets.

The following diagram describe **DNS-Detect** method.

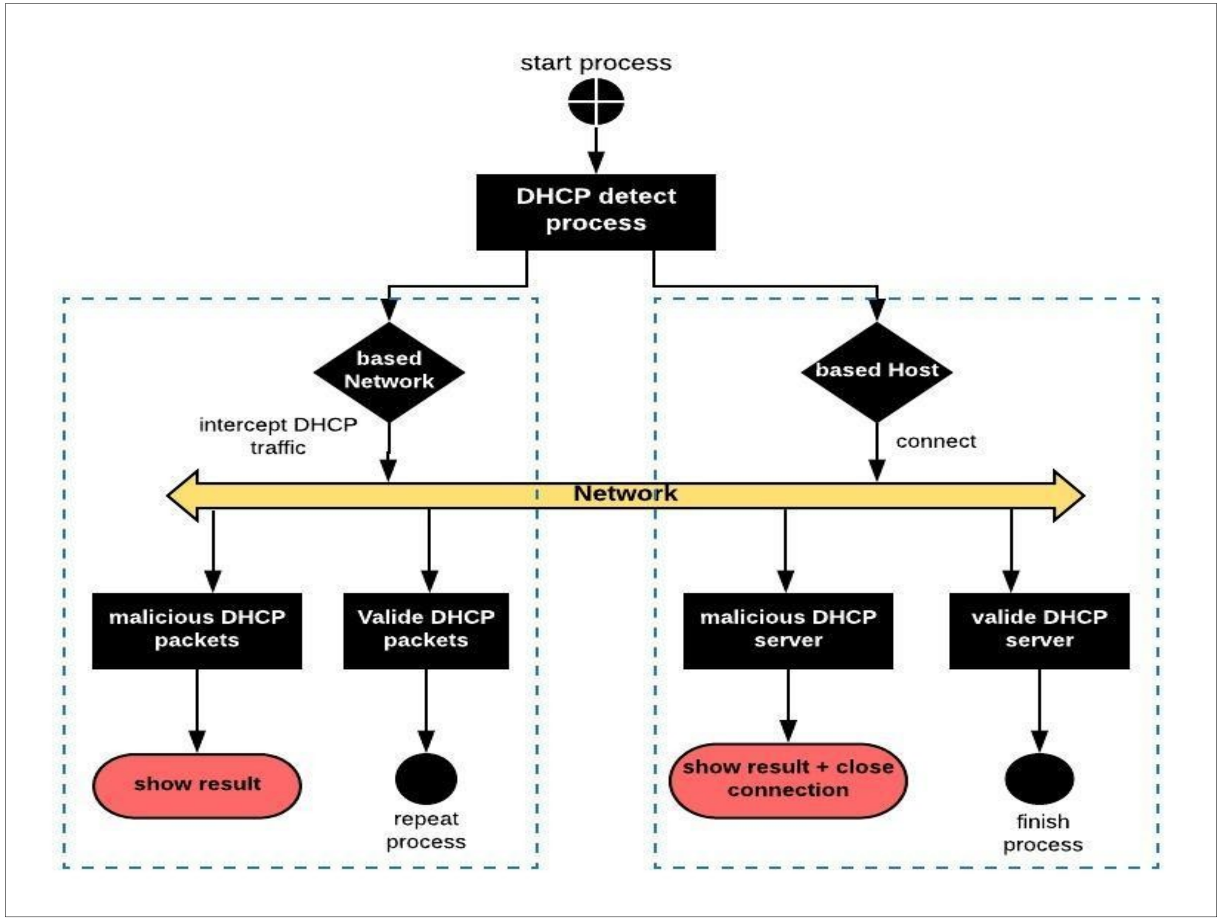


Protect and detect methods

DHCP Spoof Detect

This algorithm works in two different ways where intercept, examine, and analyze **DHCP** packets depending on the specific mode :

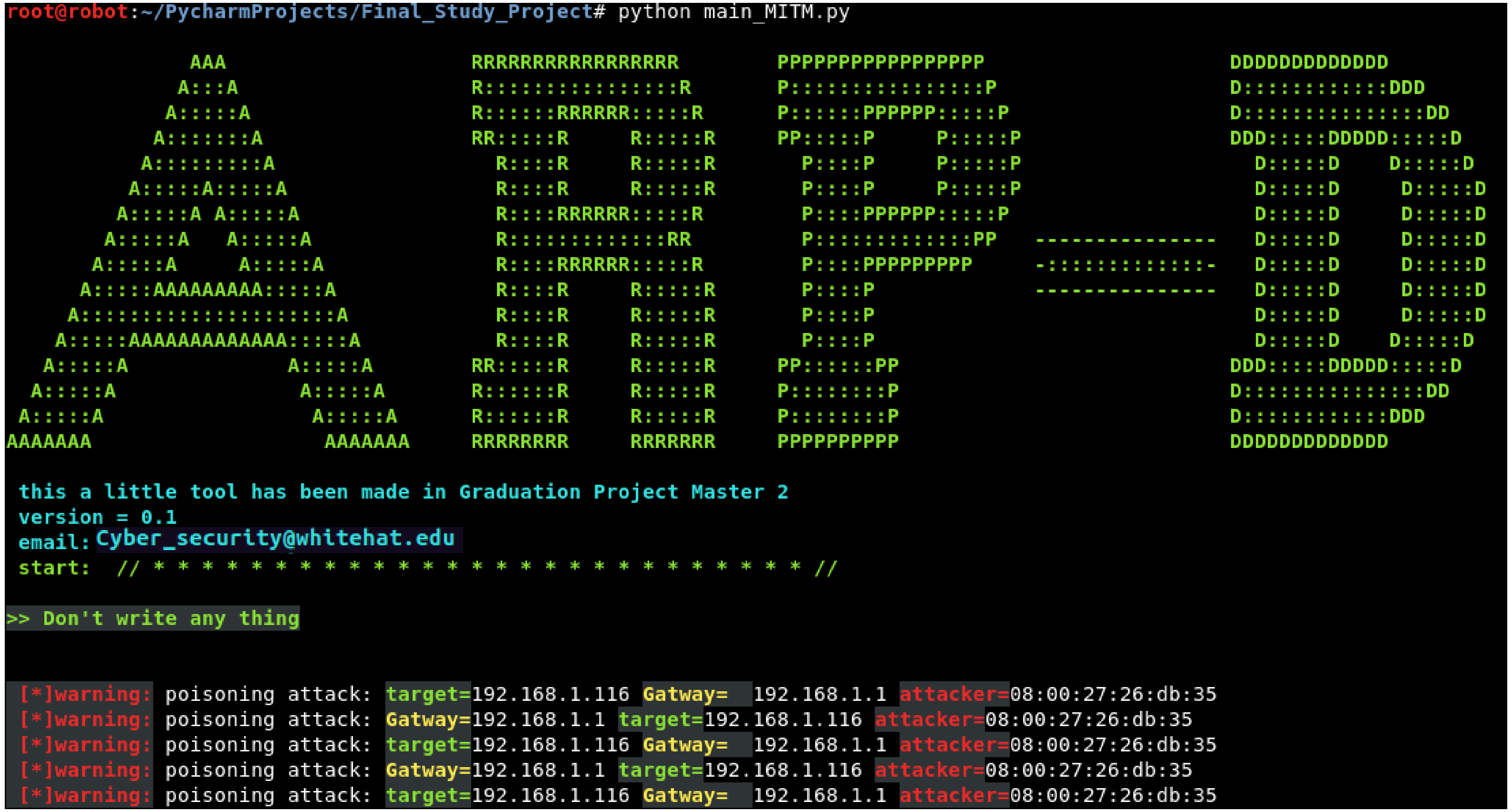
- Network **DHCP-Detect**.
- Host **DHCP-Detect**.



Experimentation and Results

Result of ARP Poisoning Detect

These are the results obtained by the application that means **ARP** spoofing attack currently happened.

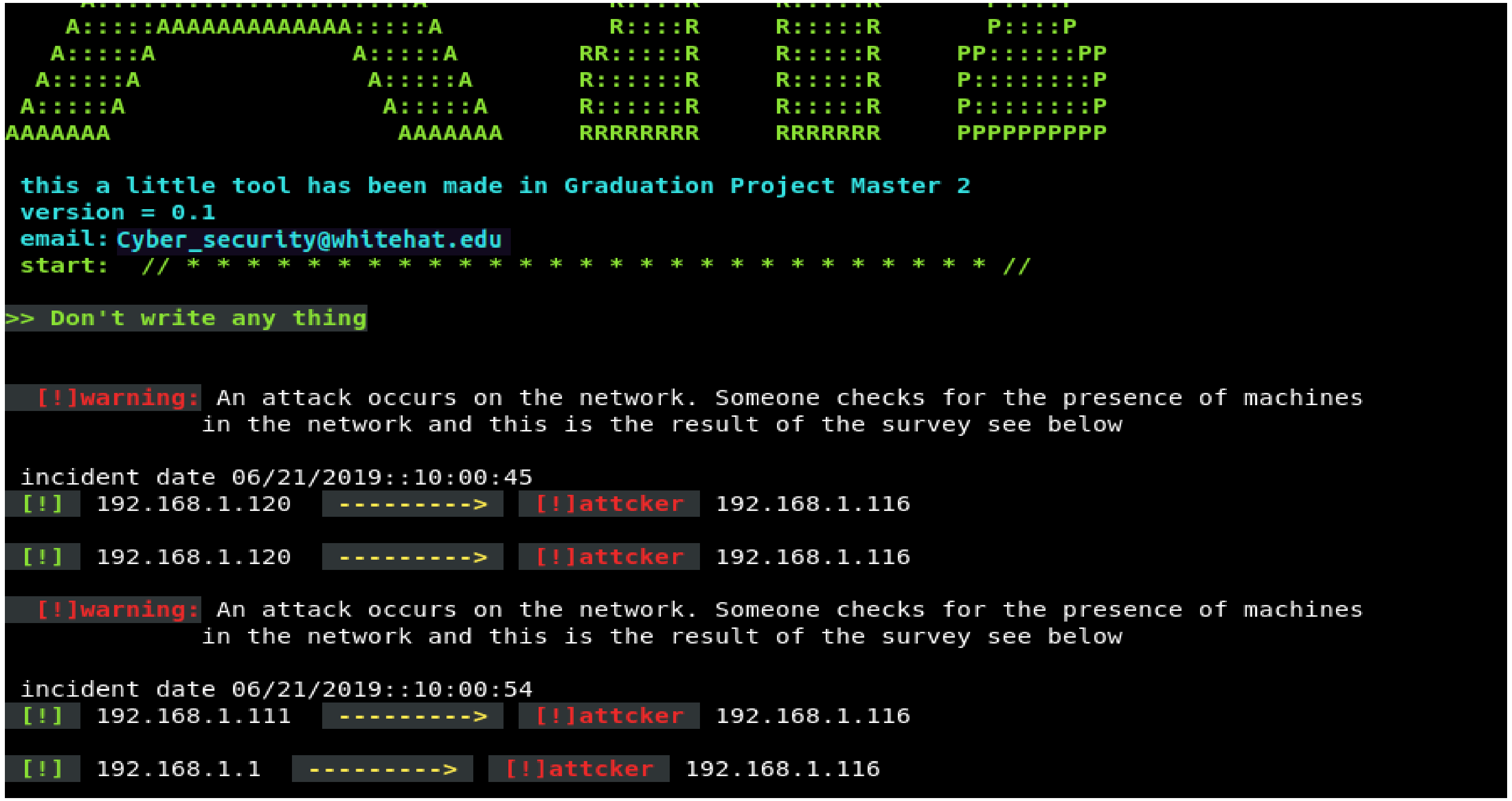


```
root@robot:~/PycharmProjects/Final_Study_Project# python main_MITM.py

               AAA                    RRRRRRRRRRRRRRRRR        PPPPPPPPPPPPPPPPP                    DDDDDDDDDDDDD
              A:::A                   R::::::::::::::::R       P::::::::::::::::P                   D::::::::::::DDD
             A:::::A                  R::::::RRRRRR:::::R      P::::::PPPPPP:::::P                  D:::::::::::::::DD
            A:::::::A                 RR:::::R     R:::::R     PP:::::P     P:::::P                 DDD:::::DDDDD:::::D
           A:::::::::A                  R::::R     R:::::R       P::::P     P:::::P                   D:::::D    D:::::D
          A:::::A:::::A                 R::::R     R:::::R       P::::P     P:::::P                   D:::::D     D:::::D
         A:::::A A:::::A                R::::RRRRRR:::::R        P::::PPPPPP:::::P                    D:::::D     D:::::D
        A:::::A   A:::::A               R:::::::::::::RR         P:::::::::::::PP   ---------------   D:::::D     D:::::D
       A:::::A     A:::::A              R::::RRRRRR:::::R        P::::PPPPPPPPP     -:::::::::::::-   D:::::D     D:::::D
      A:::::AAAAAAAAA:::::A             R::::R     R:::::R       P::::P             ---------------   D:::::D     D:::::D
     A:::::::::::::::::::::A            R::::R     R:::::R       P::::P                               D:::::D     D:::::D
    A:::::AAAAAAAAAAAAA:::::A           R::::R     R:::::R       P::::P                               D:::::D    D:::::D
   A:::::A             A:::::A        RR:::::R     R:::::R     PP::::::PP                           DDD:::::DDDDD:::::D
  A:::::A               A:::::A       R::::::R     R:::::R     P::::::::P                           D:::::::::::::::DD
 A:::::A                 A:::::A      R::::::R     R:::::R     P::::::::P                           D::::::::::::DDD
AAAAAAA                   AAAAAAA     RRRRRRRR     RRRRRRR     PPPPPPPPPP                           DDDDDDDDDDDDD

 this a little tool has been made in Graduation Project Master 2
 version = 0.1
 email: Cyber_security@whitehat.edu
 start:  // * * * * * * * * * * * * * * * * * * * * * * * * * * * //

>> Don't write any thing


 [*]warning: poisoning attack: target=192.168.1.116 Gatway=  192.168.1.1 attacker=08:00:27:26:db:35
 [*]warning: poisoning attack: Gatway=192.168.1.1 target=192.168.1.116 attacker=08:00:27:26:db:35
 [*]warning: poisoning attack: target=192.168.1.116 Gatway=  192.168.1.1 attacker=08:00:27:26:db:35
 [*]warning: poisoning attack: Gatway=192.168.1.1 target=192.168.1.116 attacker=08:00:27:26:db:35
 [*]warning: poisoning attack: target=192.168.1.116 Gatway=  192.168.1.1 attacker=08:00:27:26:db:35
```

Experimentation and Results

Discover the network scan process

Although this is not a serious process, it is classified as medium, while **ARP** Spoofing is critical.

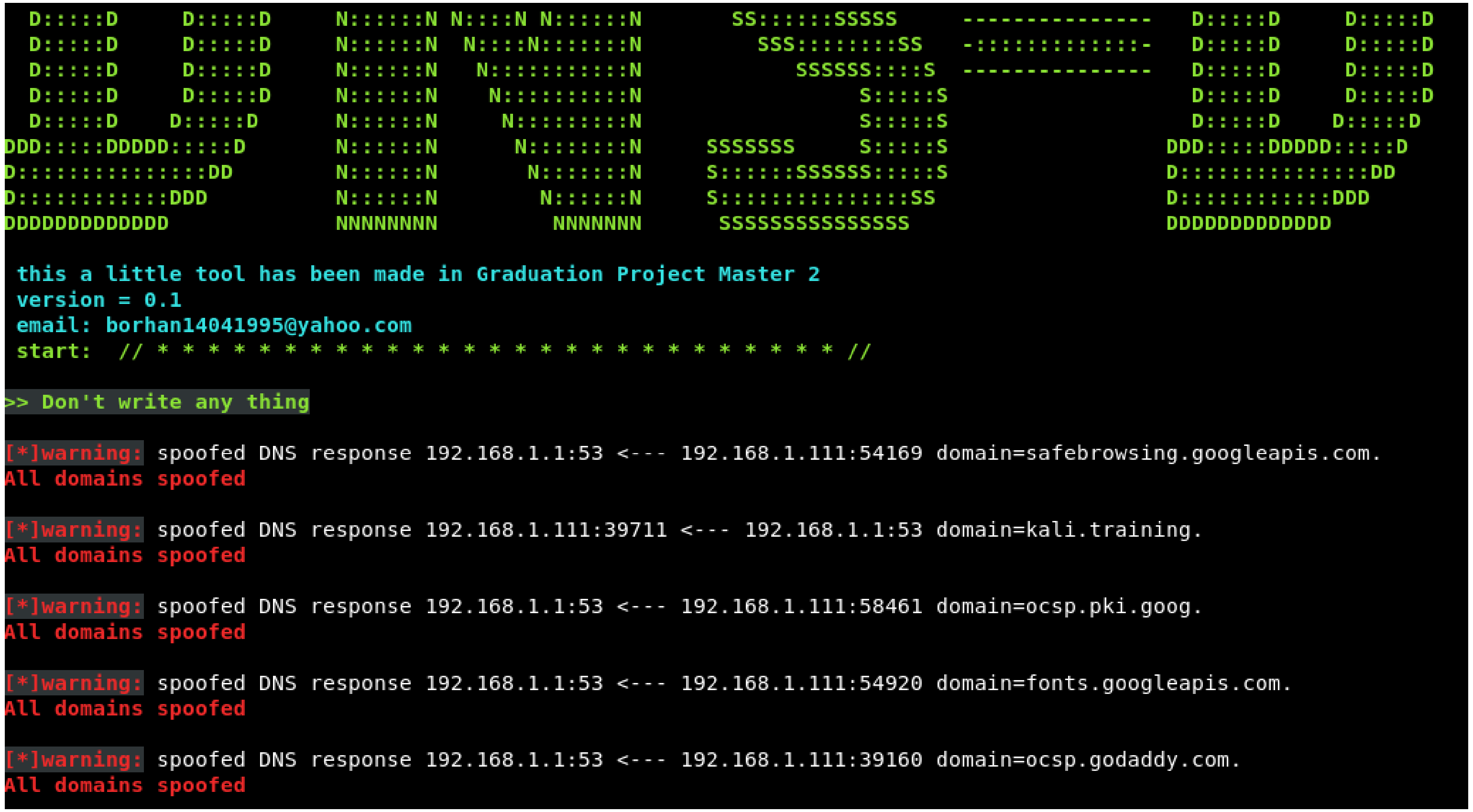


```
    A:::::AAAAAAAAAAAAA:::::A           R::::R     R:::::R       P::::P
   A:::::A             A:::::A        RR:::::R     R:::::R     PP::::::PP
  A:::::A               A:::::A       R::::::R     R:::::R     P::::::::P
 A:::::A                 A:::::A      R::::::R     R:::::R     P::::::::P
AAAAAAA                   AAAAAAA     RRRRRRRR     RRRRRRR     PPPPPPPPPP

 this a little tool has been made in Graduation Project Master 2
 version = 0.1
 email: Cyber_security@whitehat.edu
 start:  // * * * * * * * * * * * * * * * * * * * * * * * * * * * //

>> Don't write any thing


  [!]warning: An attack occurs on the network. Someone checks for the presence of machines
             in the network and this is the result of the survey see below

 incident date 06/21/2019::10:00:45
 [!]  192.168.1.120   ---------->  [!]attcker 192.168.1.116

 [!]  192.168.1.120   ---------->  [!]attcker 192.168.1.116

  [!]warning: An attack occurs on the network. Someone checks for the presence of machines
             in the network and this is the result of the survey see below

 incident date 06/21/2019::10:00:54
 [!]  192.168.1.111   ---------->  [!]attcker 192.168.1.116

 [!]  192.168.1.1   ---------->  [!]attcker 192.168.1.116
```

Experimentation and Results

Result of Detect DNS Spoof

This result is shown as the application works whatever mode, when run **DNS_Detect.py** file, here We have warning messages in red where each message corresponds to a captured malicious packet.



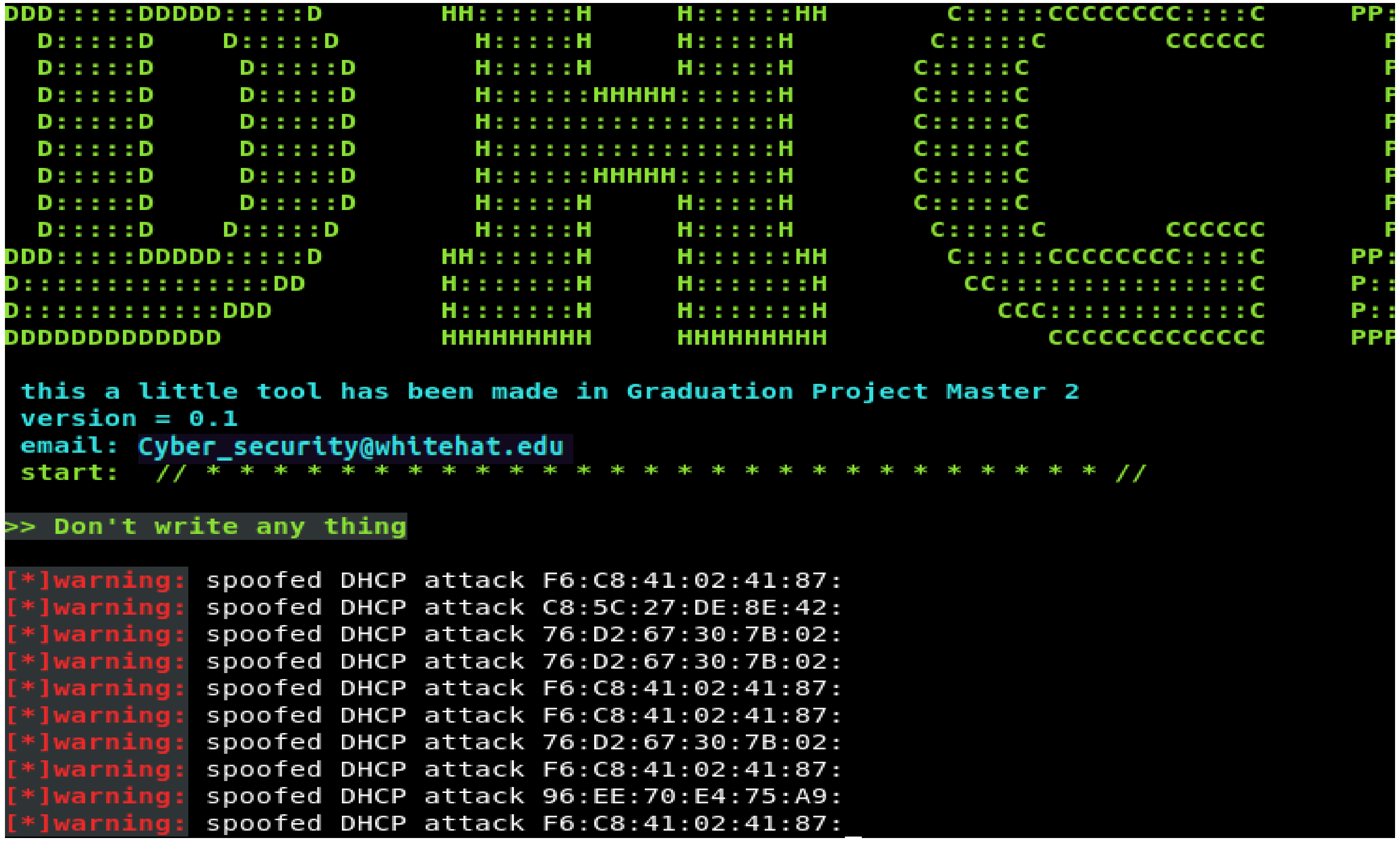
```
  D:::::D     D:::::D     N::::::N N::::N N::::::N       SS::::::SSSSS     ---------------   D:::::D     D:::::D
  D:::::D     D:::::D     N::::::N  N::::N:::::::N         SSS::::::::SS   -:::::::::::::-   D:::::D     D:::::D
  D:::::D     D:::::D     N::::::N   N:::::::::::N            SSSSSS::::S  ---------------   D:::::D     D:::::D
  D:::::D     D:::::D     N::::::N    N::::::::::N                 S:::::S                   D:::::D     D:::::D
  D:::::D    D:::::D      N::::::N     N:::::::::N                 S:::::S                   D:::::D    D:::::D
DDD:::::DDDDD:::::D       N::::::N      N::::::::N     SSSSSSS     S:::::S                 DDD:::::DDDDD:::::D
D:::::::::::::::DD        N::::::N       N:::::::N     S::::::SSSSSS:::::S                 D:::::::::::::::DD
D::::::::::::DDD          N::::::N        N::::::N     S:::::::::::::::SS                  D::::::::::::DDD
DDDDDDDDDDDDD             NNNNNNNN         NNNNNNN      SSSSSSSSSSSSSSS                    DDDDDDDDDDDDD

 this a little tool has been made in Graduation Project Master 2
 version = 0.1
 email: borhan14041995@yahoo.com
 start:  // * * * * * * * * * * * * * * * * * * * * * * * * * * * //

>> Don't write any thing

[*]warning: spoofed DNS response 192.168.1.1:53 <--- 192.168.1.111:54169 domain=safebrowsing.googleapis.com.
All domains spoofed

[*]warning: spoofed DNS response 192.168.1.111:39711 <--- 192.168.1.1:53 domain=kali.training.
All domains spoofed

[*]warning: spoofed DNS response 192.168.1.1:53 <--- 192.168.1.111:58461 domain=ocsp.pki.goog.
All domains spoofed

[*]warning: spoofed DNS response 192.168.1.1:53 <--- 192.168.1.111:54920 domain=fonts.googleapis.com.
All domains spoofed

[*]warning: spoofed DNS response 192.168.1.1:53 <--- 192.168.1.111:39160 domain=ocsp.godaddy.com.
All domains spoofed
```

Experimentation and Results

Result of Detect DHCP Spoof

Here We have warning messages in red where each message corresponds to a captured malicious packet and the information in the message is as follows :

*Random **MAC** addresses are drowned out by attacker to the router (**DHCP** server) in order to drain all **IP** addresses.



```
DDD:::::DDDDD:::::D       HH::::::H     H::::::HH       C:::::CCCCCCCC::::C     PP:
  D:::::D    D:::::D        H:::::H     H:::::H        C:::::C       CCCCCC       P
  D:::::D     D:::::D       H:::::H     H:::::H       C:::::C                     P
  D:::::D     D:::::D       H::::::HHHHH::::::H       C:::::C                     P
  D:::::D     D:::::D       H:::::::::::::::::H       C:::::C                     P
  D:::::D     D:::::D       H:::::::::::::::::H       C:::::C                     P
  D:::::D     D:::::D       H::::::HHHHH::::::H       C:::::C                     P
  D:::::D     D:::::D       H:::::H     H:::::H       C:::::C                     P
  D:::::D    D:::::D        H:::::H     H:::::H        C:::::C       CCCCCC       P
DDD:::::DDDDD:::::D       HH::::::H     H::::::HH       C:::::CCCCCCCC::::C     PP:
D:::::::::::::::DD        H:::::::H     H:::::::H        CC:::::::::::::::C     P::
D::::::::::::DDD          H:::::::H     H:::::::H          CCC::::::::::::C     P::
DDDDDDDDDDDDD             HHHHHHHHH     HHHHHHHHH             CCCCCCCCCCCCC     PPP

 this a little tool has been made in Graduation Project Master 2
 version = 0.1
 email: Cyber_security@whitehat.edu
 start:  // * * * * * * * * * * * * * * * * * * * * * * * * * * * //

>> Don't write any thing

[*]warning: spoofed DHCP attack F6:C8:41:02:41:87:
[*]warning: spoofed DHCP attack C8:5C:27:DE:8E:42:
[*]warning: spoofed DHCP attack 76:D2:67:30:7B:02:
[*]warning: spoofed DHCP attack 76:D2:67:30:7B:02:
[*]warning: spoofed DHCP attack F6:C8:41:02:41:87:
[*]warning: spoofed DHCP attack F6:C8:41:02:41:87:
[*]warning: spoofed DHCP attack 76:D2:67:30:7B:02:
[*]warning: spoofed DHCP attack F6:C8:41:02:41:87:
[*]warning: spoofed DHCP attack 96:EE:70:E4:75:A9:
[*]warning: spoofed DHCP attack F6:C8:41:02:41:87:_
```

Experimentation and Results

Statistics Result

	Based on Host wireless	Based on Network wireless	Based on Host wired	Based on Network wired
Test rate attack ARP	15	12	8	11
ARP Poisoning detection	98%	80%	98%	85%
Test rate attack DNS	20	15	5	10
DNS spoofing detection	75%	50%	75%	50%
Test rate attack DHCP	22	18	10	10
DHCP spoofing detection	45%	50%	60%	90%

Conclusion

Conclusion

- In this dissertation, we provided a good structure to limit the attack of the man in the middle, where we focused on the three most important factors where the vulnerability infects Internet protocols (**ARP, DNS, DHCP**) ,With our understanding the process of exploitation,we were able, through our application, to develop the necessary protection mechanism using **Scapy** programing.

- The solution to reduce attacks of a man in the middle is detection technique allowed us to have a deep knowledge of how the network protocols and data transmission work,through the continuous analysis with working of the application in addition to our study, we feel are able to develop application to work on different environments on infrastructure and adoption of smarter ways. The experimental results of our application show how to detect attacks with some study statistics and this is to ensure that users take preventive measures to deter the actual attack.

Conclusion

Perspectives

Continuing the project will to cover most offensive techniques for **MitM** attack. In the future :

- we going to add the prevention methods we are currently working on.
- It is also will more controllable by enabling remote control using the (client / server) method.
- event logging, and logs management.
- Enable the sending of (alerts / notifications) via email in detail.
- Add methods of statistics and deep learning if possible.
- Intelligent network analysis for various protocols and devices, for example, save the correct **HTTP** headers inside database as digital signature from which we can identify the worst attacks such as social engineering and phishing attacks.

Thank you for your attention