

DISCOVER, SCAN, AND MAP YOUR LAB NETWORK

In this activity, you will use **Kali Linux** to discover the live hosts on your lab network, identify their IP addresses, scan for open ports, gather service information, and build a simple asset and service map. This follows the same sequence as the lecture material: first identify hosts, then scan for ports and services, then use banner grabbing and enumeration to collect deeper details, and finally connect your findings to risks and countermeasures.

The activity is designed around the tools and virtual machines you already have: **Kali Linux**, **Windows 11**, **OWASP**, and **Metasploitable**. It directly supports the session learning outcomes, which include explaining scanning and enumeration, distinguishing common scanning techniques, using the right tools, and evaluating security risks and countermeasures.

Your aim

Your aim is to use Kali to:

1. find all live hosts on your lab subnet
2. identify their IP addresses without logging in to each target
3. scan for open ports
4. identify services and banners
5. perform simple enumeration where appropriate
6. create a small network asset and service map
7. explain the main security risks you discovered

This matches the lecture explanation that scanning is used to identify **live hosts, IP addresses, open ports, services, operating systems, and vulnerabilities**, while enumeration goes deeper by extracting details such as shares, services, and configurations.

Machines you will use

Use these VMs in the same isolated virtual network:

- **Kali Linux** as your scanning machine
- **Metasploitable** as a deliberately vulnerable target
- **Windows 11** as a workstation target
- **OWASP VM** as a web application target

Keep the work inside your own lab only. The lecture material stresses that scanning and enumeration should be practised ethically and in controlled environments.

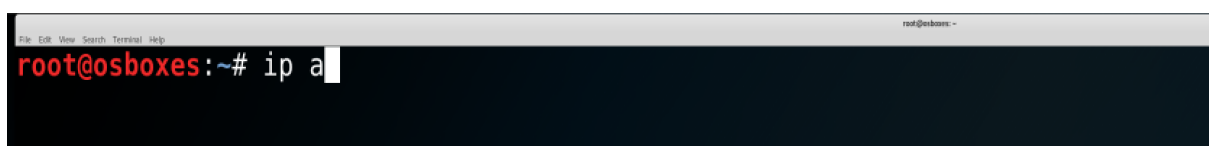
Step 1: Start the lab machines

Start Kali Linux, Metasploitable, Windows 11, and OWASP. Make sure they are connected to the same **host-only** or **internal** virtual network.

Do not check the IP addresses from the target machines. In this task, you must discover them from Kali.

Step 2: Identify your subnet from Kali

On Kali, open a terminal and run:



```
File Edit View Search Terminal Help
root@osboxes:~# ip a
```

- Look for Kali's IP address and subnet mask. For example, if Kali has an address like **192.168.56.101/24**, then your subnet is likely **192.168.56.0/24**.
- You need this because the lecture explains that ping sweeps and host discovery depend on knowing the subnet range you are scanning.
- Write down your subnet before moving on.

Step 3: Find all live hosts from Kali

From the Kali terminal, run:

```
File Edit View Search Terminal Help
root@osboxes: ~# nmap -sn 192.168.56.0/24
```

- Replace the subnet with the correct one for your lab.
- This performs a host discovery scan. Record every IP address that appears as live. Do not log in to the target machines to confirm their addresses. Your job is to discover them from Kali only.
- This step directly matches the lecture description of **ICMP scanning** and **ping sweeps**, which are used to identify active devices and create an inventory of live systems in a subnet.
- If a machine does not respond, note that this may be because it is powered off, on the wrong network, or blocking ICMP traffic with a firewall. The lecture notes that ICMP scanning can also help you determine whether a firewall is blocking replies.

Step 4: Create your initial host table

Using only the scan results from Kali, make a table like this:

Host Number	IP Address	Likely Identity	Reason for Your Guess
Host 1	192.168.56.X	Unknown	Live host discovered by Nmap
Host 2	192.168.56.Y	Unknown	Live host discovered by Nmap
Host 3	192.168.56.Z	Unknown	Live host discovered by Nmap

- At this stage, leave the identity as **unknown** or make only a cautious guess. You will confirm which machine is which later by looking at open ports, services, and banners.
- This reflects the lecture sequence: first identify the host, then scan it, then gather service and banner information, then map and document the network.

Step 5: Scan each live host for common TCP ports

For each live IP address you found, run:

```
File Edit View Search Terminal Help
root@osboxes:~# nmap -sS -Pn -T3 <target-ip>
```

- Run the scan once for every live host you discovered.
- Record the open ports for each system.
- This step practises **TCP scanning**, which the lecture describes as a way to identify open and closed ports. A SYN scan is a half-open scan, which stops before the full three-way handshake completes.

Make a table like this:

IP Address	Open Ports	First Impression
192.168.56.X	21, 22, 23, 80	Likely a server with several exposed services
192.168.56.Y	135, 139, 445	Likely a Windows system
192.168.56.Z	80, 443	Likely a web server or web application VM

Step 6: Run a full port scan on the most exposed host

Choose the host with the largest number of open services. This will often be **Metasploitable**.

Run:

```
File Edit View Search Terminal Help
root@osboxes:~# nmap -sS -Pn -p- -T3 <target-ip>
```

- This scans all TCP ports instead of only the most common ones.
- Record any additional ports that appear.
- This deepens your scan and supports the lecture theme that network scanning is used to discover not just hosts, but also the services running on them.

Step 7: Identify services and versions

For each live host, run:

```
File Edit View Search Terminal Help
root@osboxes:~# nmap -sV <target-ip>
```

Then, if needed, run:

```
File Edit View Search Terminal Help
root@osboxes:~# nmap -sV -sC <target-ip>
```

- This helps you identify the application or service running on each open port and may also reveal version information.
- This step is directly related to **banner grabbing**, which the lecture describes as identifying software and operating systems by reading service responses, like reading a name tag on a service.
- Now update your host table and replace **unknown** with your best confirmed identification.
- For example, a host showing services such as FTP, Telnet, SMB, and HTTP is likely **Metasploitable**. A host showing ports such as 135, 139, 445, or 3389 is likely **Windows 11**. A host mainly showing HTTP or HTTPS is likely the **OWASP VM**.

Make a table like this:

IP Address	Confirmed Host	Port	Service	Version or Banner
192.168.56.X	Metasploitable	21	FTP	Service banner shown by Nmap
192.168.56.Y	Windows 11	445	SMB	Microsoft service details
192.168.56.Z	OWASP VM	80	HTTP	Web server information

Step 8: Enumerate SMB or NetBIOS if present

If any host has port **139** or **445** open, run these commands from the Kali terminal:

```
File Edit View Search Terminal Help
root@osboxes:~# nbtscan <target-ip>
```

and

```
File Edit View Search Terminal Help
root@osboxes:~# smbclient -L //<target-ip> -N
```

- Record any hostnames, shares, workgroup names, or other useful information that appear.
- This matches the lecture discussion of **NetBIOS enumeration**, which can be used to identify systems and shared resources. The slides specifically mention listing systems and shares as part of enumeration.
- If anonymous access is blocked, simply record that access was restricted.

Step 9: Enumerate HTTP services if present

If a host has port **80** or **443** open, investigate it from **Kali**.

First, in the **terminal**, type:

```
File Edit View Search Terminal Help
root@osboxes:~# firefox http://<target-ip>
```

You type this command in the **terminal**, and it opens the website in the **Firefox browser**. Once the browser opens, look carefully at the page and note what you can identify, such as the application name, login page, or any visible version details.

Next, return to the **terminal** and run:

A terminal window with a dark background and a light gray title bar containing 'File Edit View Search Terminal Help'. The prompt is 'root@osboxes:~#' in red. The command 'curl -I http://<target-ip>' is entered in white text.

This shows the HTTP response headers. Look for things such as the status code and server header.

Then run:

A terminal window with a dark background and a light gray title bar containing 'File Edit View Search Terminal Help'. The prompt is 'root@osboxes:~#' in red. The command 'whatweb http://<target-ip>' is entered in white text.

This helps identify web technologies used by the target.

Record:

- the target IP
- what appeared in Firefox
- the headers returned by `curl -I`
- the technologies identified by `whatweb`
- how this information helps you identify the host or understand its exposure

This is a practical form of banner grabbing and service enumeration because you are moving beyond “`port open`” and collecting real information about the application and service stack.

That fits the lecture’s explanation of banner grabbing and deeper information gathering.

Step 10: Check for SNMP only if UDP 161 is present

If you have reason to believe a target is using SNMP, run a UDP scan from Kali:

```
File Edit View Search Terminal Help
root@osboxes:~# nmap -sU -p 161 <target-ip>
```

If UDP port **161** appears open, try this read-only query:

```
File Edit View Search Terminal Help
root@osboxes:~# snmpwalk -v1 -c public <target-ip>
```

Do this only inside your own lab.

In today's lecture, I explain that **SNMP enumeration** can reveal device and network configuration details, and it highlights the use of default community strings such as **public** for read access. If nothing is returned, record that SNMP was not exposed or did not allow access.

Step 11: Build your final asset and service map

Using everything you discovered from Kali, create a final table like this:

IP Address	Confirmed Host	Open Ports	Key Services	Main Exposure
192.168.56.X	Metasploitable	21, 23, 80, 139, 445	FTP, SSH, Telnet, HTTP, SMB	Many exposed legacy services
192.168.56.Y	Windows 11	135, 139, 445, 3389	RPC, SMB, RDP	Remote access and file sharing exposure
192.168.56.Z	OWASP VM	80, 443	HTTP, HTTPS	Web application attack surface

This reflects the lecture step of **mapping and documenting** the target network after host discovery, scanning, and banner grabbing.

Step 12: Write a short risk note for each host

For each machine, answer these questions in a few sentences:

- What services are exposed?
- Which services look risky, old, unnecessary, or overexposed?
- What should a defender do to reduce that exposure?

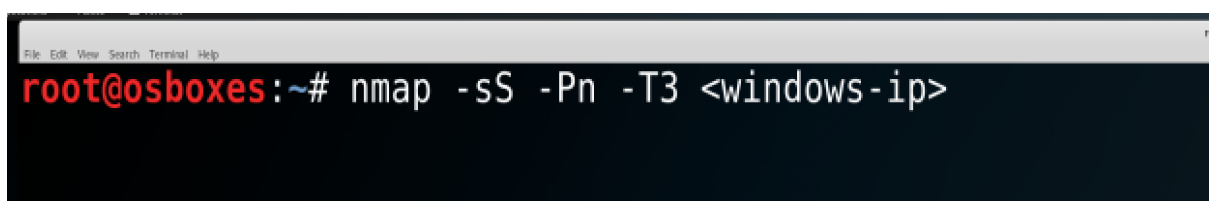
Use ideas from the lecture such as:

- close unused ports and services
- use stronger authentication and access controls
- hide or reduce unnecessary banner information
- monitor logs and unusual network queries
- use firewalls and filtering to reduce exposure

This part matters because the lecture is not only about how scanning works, but also about how understanding scanning helps defenders design stronger protections.

Optional extension: observe the effect of a firewall

On the Windows 11 VM, temporarily block an inbound rule such as **File and Printer Sharing** in Windows Defender Firewall. Then go back to Kali and rerun:

A terminal window with a dark background and a light gray title bar. The title bar contains the text "File Edit View Search Terminal Help". The terminal prompt is "root@osboxes:~#". The command being entered is "nmap -sS -Pn -T3 <windows-ip>".

```
File Edit View Search Terminal Help
root@osboxes:~# nmap -sS -Pn -T3 <windows-ip>
```

- Compare the new result with your earlier scan.
- Write down what changed. For example, note whether some ports disappeared, whether the machine still appeared live, and whether you could still identify the same services.
- This extension connects to the lecture explanation that firewalls filter traffic and affect what a scanner can see.

What you should hand in

Submit a short lab report that includes:

- A brief note of the subnet you scanned.
- A list of all live hosts discovered from Kali.
- Your port scan results for each host.
- Your service and banner findings.
- Any SMB, NetBIOS, HTTP, or SNMP enumeration results.
- Your final asset and service map.
- A short risk summary and recommended countermeasures for each host.

How this activity fits Week 6 lecture

- This activity follows the exact logic of the lecture content. You start with **host discovery**, then move to **TCP and service scanning**, then collect deeper information through **banner grabbing** and **enumeration**, and finally interpret the results from a **defender's point of view**.
- The key practical feature is that you must find the live hosts and IP addresses from **Kali**, rather than checking them on each machine. That makes the task more realistic and better aligned with how scanning is actually used to build an inventory of unknown systems on a network.