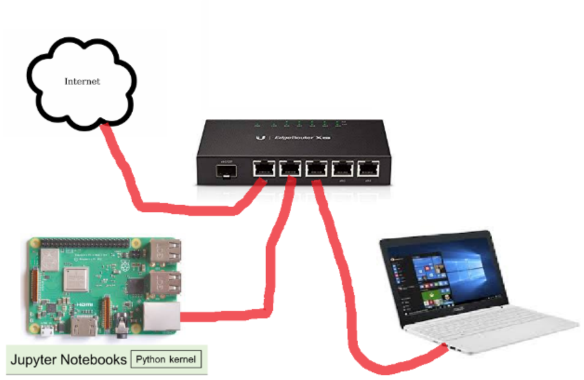
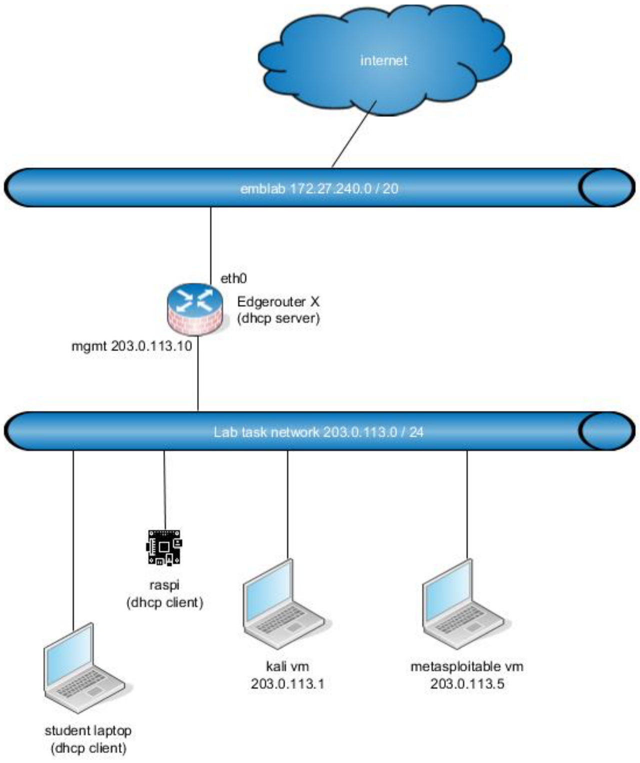
Lab - Port Scanning an IoT Device

Topology





Objectives

Part 1: Perform a Nmap network discovery scan

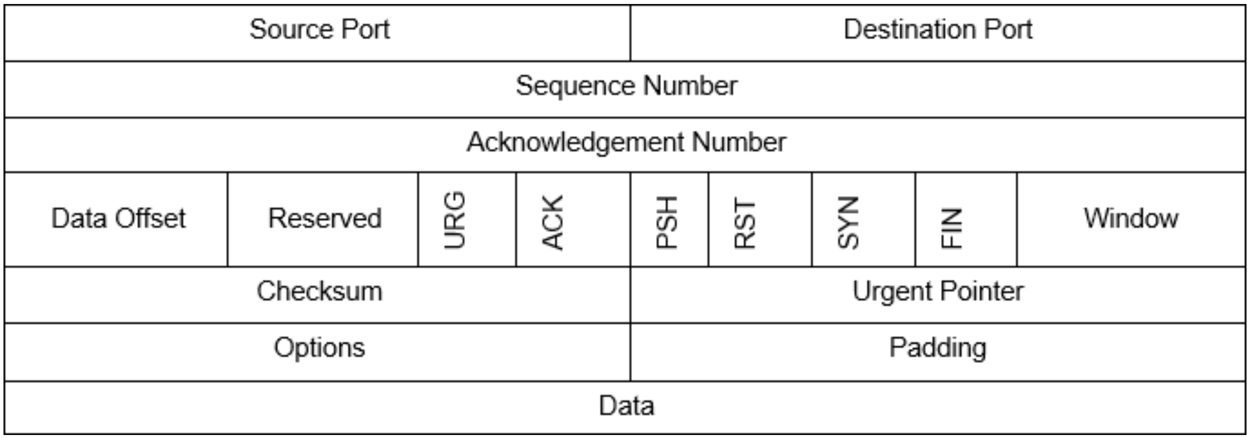
Part 2: Compare a Nmap TCP default port scan and full scan

Part 3: Perform a Nmap UDP Scan

Part 4: Perform Nmap OS and Service Foot Printing

Background / Scenario

Nmap, or “Network mapper,” is a free and open source network port scanner. It can be used to test IoT devices and firewalls for open, closed, or filtered ports and additionally can provide an inventory of devices and services available on a network. Nmap uses the TCP header flag fields URG, ACK, PSH, RST, SYN, FIN, shown in the diagram, to accomplish its scans. It also uses other protocols such as UDP, ICMP and a built-in database of known OS and application signatures. Nmap is included with Kali, but can also be downloaded from https://nmap.org/. A comprehensive reference about how Nmap works and how it is used is available at the Nmap website.



<table>
<tr><td colspan="4">Source Port</td><td colspan="5">Destination Port</td></tr>
<tr><td colspan="9">Sequence Number</td></tr>
<tr><td colspan="9">Acknowledgement Number</td></tr>
<tr><td>Data Offset</td><td>Reserved</td><td>URG</td><td>ACK</td><td>PSH</td><td>RST</td><td>SYN</td><td>FIN</td><td>Window</td></tr>
<tr><td colspan="4">Checksum</td><td colspan="5">Urgent Pointer</td></tr>
<tr><td colspan="4">Options</td><td colspan="5">Padding</td></tr>
<tr><td colspan="9">Data</td></tr>
</table>

Required Resources

- Raspberry Pi 3 Model B or later
- 8GB Micro SD card (minimum required)
- PC with IoTSec Kali VM
- Network connectivity between PC and Raspberry Pi

Part 1: Perform a Nmap Network Discovery Scan

Step 1: Use Kali to perform a host discovery scan.

The first step in network scanning is to discover the addresses of the hosts that are actually up on the network. This is because many network information gathering tasks, such as port scanning, are time-consuming. Nmap first scans the network to identify the addresses of hosts that should be scanned in greater depth. Doing in-depth scans of an entire network of 254 hosts, for example, is unnecessary if only a fraction of those hosts are actually on and able to respond.

By default, Nmap conducts a scan of the network to generate a list of hosts that are available for further scanning. It then conducts a port scan of those hosts.

a. Set up the topology by connecting the Raspberry Pi to the PC.

b. Start the IoTSec Kali VM and log in.

c. Verify that Kali VM is assigned an IP address on eth0.

```
└─$ ifconfig
```

d. Open the man page for nmap in Kali VM and review the options that are available in Nmap.

```
└─$ man nmap
```

Refer to the man page and complete the table below. There are many other resources available online to help you learn Nmap.

Option	Name	Notes
-p	Port scan	Only scan specified ports
-sV	Service/version detection	Probe open ports to determine service/version info
-sC	Script scan	Uses script to scan
-oN	Output	Output of the scan

e. Perform a scan on your network by specifying the network address and bit mask.

```
└─$ nmap 203.0.113.0/24
```

What is the IP address of the Raspberry Pi?

R: 203.0.113.40.

f. Sometimes a device would not reply to Nmap's initial network discovery scan because of a firewall, IDS/IPS system etc. Instead of relying on the initial scan to discover hosts that are alive for further scanning, we can use Nmap to scan the network by assuming all hosts are alive.

```
└─$ nmap -Pn 203.0.113.0/24
```

What is the IP address and MAC address of your Raspberry PI?

R: 203.0.113.40 and e4:5f:01:95:6c:f7.

Part 2: Compare a Nmap TCP Default Scan and Full Scan

Step 1: Using Wireshark to display Nmap scans

a. In a Kali VM terminal, start Wireshark. Wireshark is used to monitor the traffic while scanning the network using Nmap. Click **OK** for the warning message regarding running Wireshark as a root user.

```
└─$ wireshark
```

b. Select the eth0 interface and click **Capture** to start capturing packets.

c. There can be a lot of traffic on the network. A display filter is applied to limit the number of captured packets displayed to just those that you are interested in. The interesting traffic is between the IP address of the Kali VM and the Raspberry Pi. Replace IP address of the Raspberry Pi with the IP address

Apply the following display filter in Wireshark using IP address of Kali VM as the source address and IP address of your Raspberry Pi as the destination address.

```
ip.src == 203.0.113.1 && ip.dst == 203.0.113.13
```

As an example in this lab, the IP address for the Raspberry Pi is 203.0.113.13.

Step 2: Nmap TCP default scan

a. In the terminal, enter the following command to start the Nmap scanning.

```
└─$ nmap 203.0.113.13
```

After Nmap reports the result of the scan, stop the Wireshark capture. Click the arrow next to the display filter field to filter the results of the scan.

Review the Wireshark output. Which TCP flag is Nmap using to discover the open ports?

R: It is using SYN flags.

Notice the ports that are being tested. The default Nmap scan does not test all ports. How many ports does a default Nmap scan test? (Do a web search if necessary.)

R: A default nmap scan tests the 1000 most common ports.

Look at the results of the nmap scan in the terminal. What ports are identified?

R: Port 22 and 80.

Step 3: Nmap TCP full TCP port scan

a. Nmap by default only scan a limited number of TCP ports. We would like to scan all 65535 TCP ports.

b. Start a new Wireshark capture. Click **Continue without Saving** when prompted to save the capture. to start a new capture.

c. Enter the nmap command to scan all the TCP ports in Kali VM.

```
└─$ nmap -p 1-65535 203.0.113.13
```

d. Watch the Wireshark capture screen. Notice you are sending TCP packets just as before, but the number of ports being scanned has increased.

e. Stop the Wireshark capture when finished and clear the filter.

Part 3: Perform a Nmap UDP scan

UDP Header Format

Source Port	Destination Port
Length	Checksum

Step 1: UDP scan with a new Wireshark filter.

a. Start a new Wireshark capture. Click **Continue without Saving** when prompted to save the capture.

b. Apply the following filter in Wireshark:

```
ip.src == 203.0.113.13 && udp
```

This will allow us to see the UDP traffic generated by nmap to the Raspberry PI. Notice in the UDP header above there are no flags. We can only send UDP packets and receive a possible “port unreachable or destination unreachable” message meaning the port is closed.

c. In the Kali VM terminal, enter the command **nmap** with -sU option scanning the IP address of your Raspberry Pi.

```
└─$ nmap -sU -F 203.0.113.13
```

d. It will take a few minutes, but look at the Wireshark capture. As you scroll down the packets, you will see some “Destination unreachable” on different ports indicating the port is closed.

How many UDP ports are there? How many UDP ports does Nmap scan, by default? (Use web search as necessary.)

R: There are 65,535 ports and nmap only scans 1000, and 100 if we use -F, which we did.

What are the UDP ports that are open from the scan? What protocols use these UDP ports?

R: 68 for dhcpc and 5353 for zeroconf.

What is the meaning of the -F option? Try the same scan without the -F. What is the difference?

R: The -F only scans the top 100 ports and without it nmap scans 1000.

e. Stop the Wireshark capture when finished and clear the filter.

Part 4: Perform Nmap OS and Service Foot Printing

Step 1: Use Nmap to find a device operating system.

Nmap can guess the operating system of a device based on its response to a series of TCP and UDP packets.

a. Start a new Wireshark capture. Click **Continue without Saving** when prompted to save the capture.

b. Apply the following filter in Wireshark using IP address of Kali VM as the source address and IP address of your Raspberry Pi as the destination address.

```
ip.src == 203.0.113.1 && ip.dst == 203.0.113.13
```

c. In the Kali VM terminal, enter the command **nmap** using the -O option scanning the IP address of your Raspberry Pi.

```
└─$ nmap -O 203.0.113.13
```

What operating system did Nmap guess?

R: Linux 4.15 - 5.8.

Look at the packets in Wireshark. What protocols were used to determine the OS?

R: UDP.

d. We can sometimes identify a device by looking at the time to live (TTL) field of a local ping response, which varies by device OS. See the table below:

Operating System	TTL Response time
Cisco	255
Windows	128
Linux	64

In the Kali VM terminal, enter the command to ping your Raspberry Pi with 4 ICMP packets.

```
└─$ ping -c4 203.0.113.13
```

What is the response time? Is it consistent with the Nmap identification?

R: Response time is 64, which matches nmap identification.

e. Stop the Wireshark capture when finished.

Step 2: Use Nmap to find services versions

Nmap has a built-in database of about 2,200 well-known services that is used to help identify application service versions.

a. Start a new Wireshark capture. Click **Continue without Saving** when prompted to save the capture. Apply the same display filter as the previous step.

b. In the Kali VM, enter the **nmap** command with the -A option scanning the IP address of your Raspberry Pi.

```
└─$ nmap -A 203.0.113.13
```

What are the identified applications running on the ports? Complete the table.

Port Number	Application service identified
22	Openssh
80	Tornado server

What is different in the port identification from the Nmap TCP scan in Part 2 of the lab and this application service scan? What implications does this have for IoT security?

R: This scan uses UDP rather than TCP. It allows us to identify services running and versions, which may lead to someone finding vulnerabilities in our system.

What additional functions does this scan also perform?

R: It Enable OS detection, version detection, script scanning, and traceroute

c. Stop the Wireshark capture when finished.