HackerFrogs Afterschool
What is Networking? /w TryHackMe

Class:
Network Hacking

Workshop Number:
AS-NET-01

Document Version:
1.75

Special Requirements:
Registered account
at tryhackme.com

Welcome to HackerFrogs Afterschool!

HackerFrogs Afterschool is a cybersecurity program for learning beginner cybersecurity skill across a wide variety of subjects.

This workshop is the intro class to network hacking.

What is Network Hacking?

In network hacking is the study of techniques that allow users to circumvent or abuse computer network services in order to gain access to data or user accounts that--

What is Network Hacking?

should not be available to them, or allow the elevation of access or user privileges via network services.

What is Network Hacking?

The first thing we'll learn regarding network hacking are the concepts of computer networking, common networking services and how to interact with them.

What are CTFs?

HackerFrogs Afterschool classes prefer to incorporate CTF games into classes for a more interactive experience, but what are CTFs?

What are CTFs?

Cybersecurity Capture The Flag (CTF) games are training exercises where the goal of the exercise is to “capture the flag” through use of cybersecurity skills.

What are CTFs?

In this context, “capture” means to gain access to a file or other resource, and “flag” refers to a secret phrase or password.

TryHackMe

The CTF platform we will be playing to learn network exploitation is called TryHackMe, which is a very popular and content-rich platform for learning cybersecurity skills across a wide variety of topics.

TryHackMe

TryHackMe learning modules (rooms) are split up across different topics, difficulty levels, as well as free / subscriber-only content.

TryHackMe

If we do not already have an account, we can register one at the following link:

https://tryhackme.com/signup

Let's Get Started

Let's sign into TryHackMe, then navigate to the following URL:

https://tryhackme.com/room/learncyberin25days

We want to look at **Task 10**

Network Ranges

```
Network Ranges (CIDR Notation)

192.168.10.0/24 (256 addresses)

192.168.0.0/16 (65536 addresses)

192.0.0.0/8 (16777216 addresses)
```

This is what network ranges look like in CIDR notation

Public (External) vs Private (Internal) IP Addresses

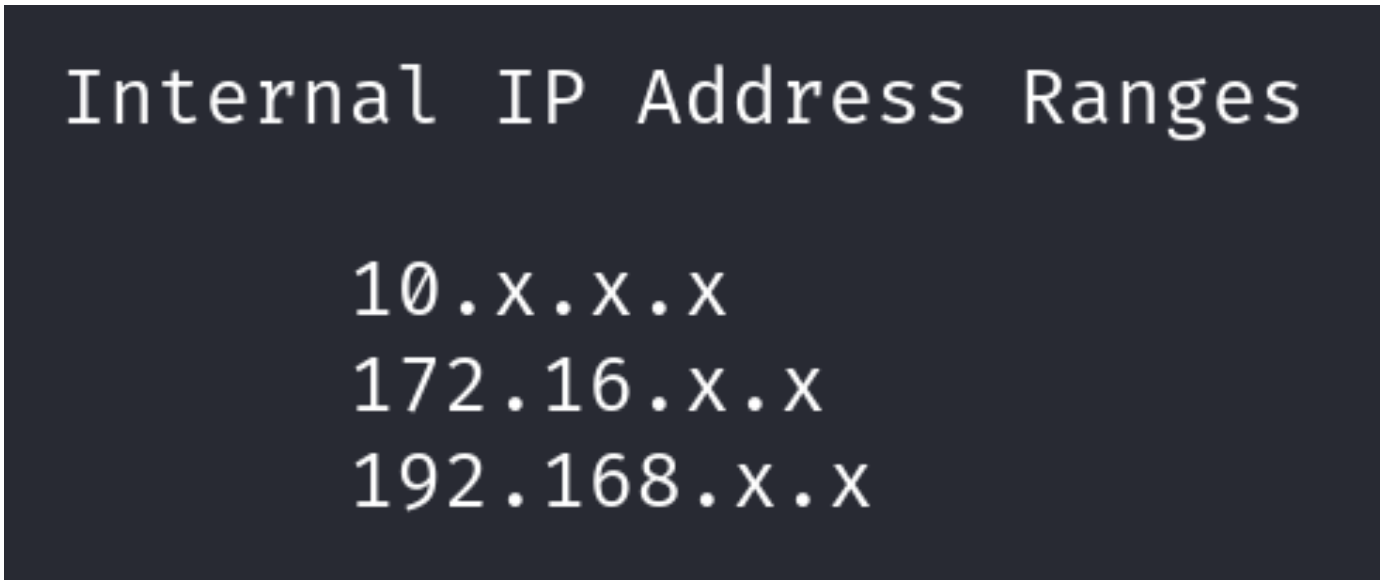
```
Example Public IP Address  8.8.8.8 (Google Dns)

Example Private IP Address 10.10.10.10

Example Special IP Address 127.0.0.1 (Localhost)
```

IP addresses are either internet-accessible, also called public or external IP addresses, or private, also called internal IP addresses.

Public (External) vs Private (Internal) IP Addresses



For the purposes of network CTF exercises, we'll need to recognize common private IP address ranges

Ping Sweeping

```
└─$ nmap -sn 192.168.10.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-04 16:32 PST
Nmap scan report for 192.168.10.1
Host is up (0.00017s latency).
MAC Address: 0A:00:27:00:00:05 (Unknown)
```

Ping sweeping is the act of sending ping packets to each address on a network range for the purpose of determining which IP addresses are online

Ping Sweeping

```
└─$ nmap -sn 192.168.10.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-04 16:32 PST
Nmap scan report for 192.168.10.1
Host is up (0.00017s latency).
MAC Address: 0A:00:27:00:00:05 (Unknown)
```

There are many different ways to do ping sweeping with different programs, including Netcat, Python, and even Bash scripting, but we'll use the Nmap method

Ping Sweeping

```
└─$ nmap -sn 192.168.10.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-04 16:32 PST
Nmap scan report for 192.168.10.1
Host is up (0.00017s latency).
MAC Address: 0A:00:27:00:00:05 (Unknown)
```

The Nmap ping sweep syntax looks like this:

`nmap -sn <IP_ADDRESS_RANGE>`

Ping Command

```
root@ip-10-10-88-142:~# ping -c 4 10.10.221.251
PING 10.10.221.251 (10.10.221.251) 56(84) bytes of data.
64 bytes from 10.10.221.251: icmp_seq=1 ttl=64 time=0.247 ms
64 bytes from 10.10.221.251: icmp_seq=2 ttl=64 time=0.235 ms
64 bytes from 10.10.221.251: icmp_seq=3 ttl=64 time=1.30 ms
64 bytes from 10.10.221.251: icmp_seq=4 ttl=64 time=0.302 ms
```

The Ping command is the most common way we determine connectivity between two network devices

Ping Command

```
root@ip-10-10-88-142:~# ping -c 4 10.10.221.251
PING 10.10.221.251 (10.10.221.251) 56(84) bytes of data.
64 bytes from 10.10.221.251: icmp_seq=1 ttl=64 time=0.247 ms
64 bytes from 10.10.221.251: icmp_seq=2 ttl=64 time=0.235 ms
64 bytes from 10.10.221.251: icmp_seq=3 ttl=64 time=1.30 ms
64 bytes from 10.10.221.251: icmp_seq=4 ttl=64 time=0.302 ms
```

The standard Ping syntax for Linux is:

`ping -c 4 <IP_ADDRESS>`

Where the -c indicates the number of packets to send

Looking Up Your Networking Info
Ip and Ifconfig commands

```
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
```

We'll also need to get our own networking info, which we can do with the Ip command. Ip command syntax:

```
ip a
```

Looking Up Your Networking Info
Ip and Ifconfig commands

```
└─$ ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST>
        inet 172.17.0.1  netmask 255.255.0.0
        ether 02:42:a1:be:76:28  txqueuelen
```

We can also get similar info by using the older Ifconfig command. Ifconfig command syntax:

`ifconfig`

Enumerating Open Ports /w Nmap

```
root@ip-10-10-88-142:~# nmap -vv -sCV -p- -T4 10.10.221.251
Starting Nmap 7.80 ( https://nmap.org ) at 2025-03-05 00:46 GMT
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 00:46
```

Nmap is very common network security tool that can be used to determine which networking ports and services are open on remote servers

Enumerating Open Ports /w Nmap

```
root@ip-10-10-88-142:~# nmap -vv -sCV -p- -T4 10.10.221.251
Starting Nmap 7.80 ( https://nmap.org ) at 2025-03-05 00:46 GMT
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 00:46
```

Nmap has a **lot** of different options, flags, and arguments that can be used with it, but we'll go over a typical method of using it. Let's go over these arguments:

Enumerating Open Ports /w Nmap

```
root@ip-10-10-88-142:~# nmap -vv -sCV -p- -T4 10.10.221.251
Starting Nmap 7.80 ( https://nmap.org ) at 2025-03-05 00:46 GMT
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 00:46
```

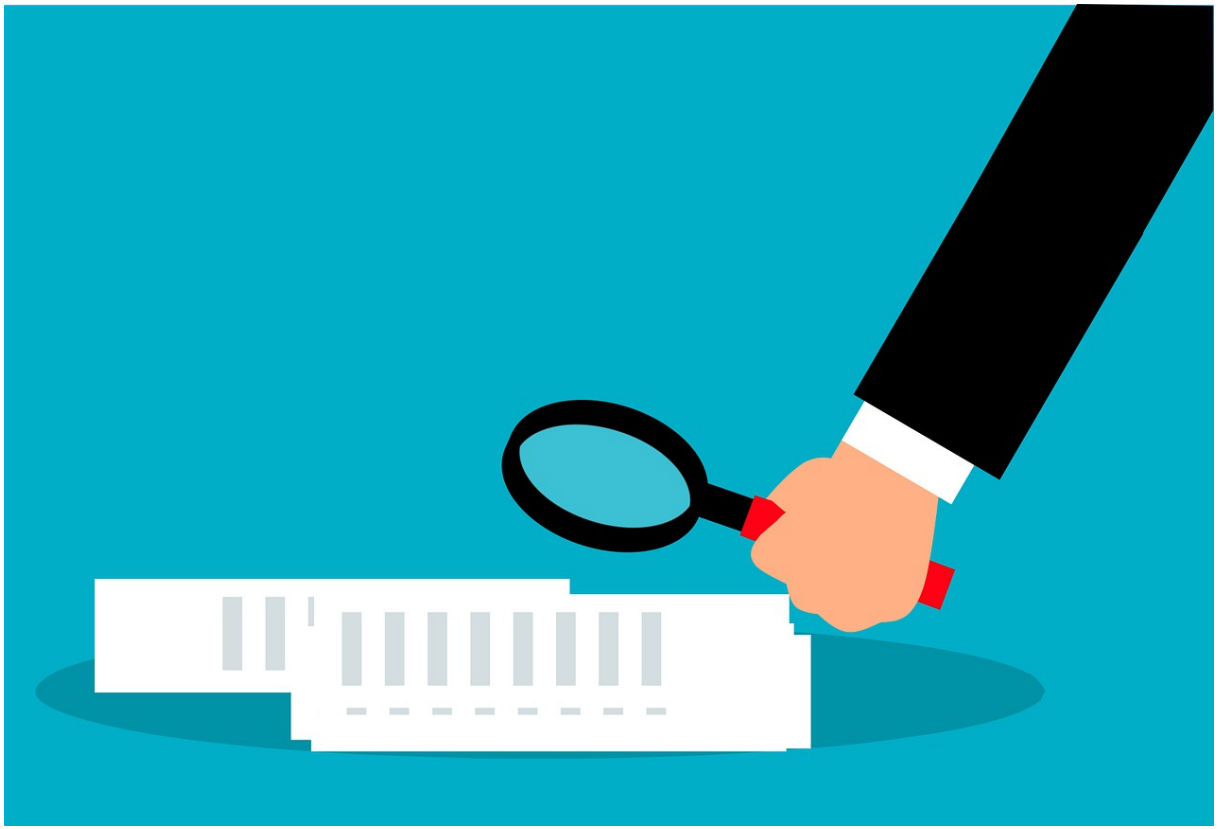
`-vv` <-- Very verbose output

`-sCV` <-- Common scripts and versioning output

`-p-` <-- Scan all networking ports

`-T4` <-- Run with 4 threads (very fast)

Summary



Let's review the network hacking concepts we learned in this workshop:

Ping Sweeping

```
└─$ nmap -sn 192.168.10.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-04 16:32 PST
Nmap scan report for 192.168.10.1
Host is up (0.00017s latency).
MAC Address: 0A:00:27:00:00:05 (Unknown)
```

Ping sweeping is the act of sending ping packets to each address on a network range for the purpose of determining which IP addresses are online

Ping Command

```
root@ip-10-10-88-142:~# ping -c 4 10.10.221.251
PING 10.10.221.251 (10.10.221.251) 56(84) bytes of data.
64 bytes from 10.10.221.251: icmp_seq=1 ttl=64 time=0.247 ms
64 bytes from 10.10.221.251: icmp_seq=2 ttl=64 time=0.235 ms
64 bytes from 10.10.221.251: icmp_seq=3 ttl=64 time=1.30 ms
64 bytes from 10.10.221.251: icmp_seq=4 ttl=64 time=0.302 ms
```

The Ping command is the most common way we determine connectivity between two network devices

Looking Up Your Networking Info
Ip and Ifconfig commands

```
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
```

We'll also need to get our own networking info, which we can do with the Ip command. Ip command syntax:

```
ip a
```

Enumerating Open Ports /w Nmap

```
root@ip-10-10-88-142:~# nmap -vv -sCV -p- -T4 10.10.221.251
Starting Nmap 7.80 ( https://nmap.org ) at 2025-03-05 00:46 GMT
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 00:46
```

Nmap is very common network security tool that can be used to determine which networking ports and services are open on remote servers

What's Next?

In the next HackerFrogs Afterschool Network Hacking workshop, we'll be learning about how to interact with a few common networking services, including SMB, FTP, and Telnet