CAN304 Lab 7

Network Scanner - Nmap

In this lab, students will learn the network scanner, i.e., nmap, which is used to scan the target (a single node or an organization's network) via sending probing packets.

Prerequisite

1. You have followed the previous lab for creating Ubuntu VM, and this lab needs 2 VMs
2. Assuming you have installed wireshark on the Ubuntu VM, if not, please refer to [1]
3. Install nmap on the Ubuntu VM by using this command-line: **apt-get install nmap**

1. Installing dependencies

1.1. Install wireshark (configuration details referring to [1])
   1) sudo apt-get update
   2) sudo apt-get install wireshark

1.2. You can launch Wireshark from the terminal by running the command:
   1) sudo wireshark

1.3. Install nmap on Ubuntu
   1) sudo apt-get install nmap

2. Network Scanner

2.1. A network scanner can disclose the target's service as well as the corresponding open ports via sending probing traffic against the target (which is a node on the internet).

2.2. Nmap is a free and open source utility for network discovery (scanner). It uses raw IP packets in novel ways to determine:
   1) what hosts are available on the network
   2) what services (application name and version) those hosts are offering
   3) what operating systems (and OS versions) they are running
   4) what type of packet filters/firewalls are in use
   5) dozens of other characteristics

2.3. How to use nmap?
   - **nmap [scan types] <options> {target specification}**
     - ✓ target specification could be: hostnames, IP addresses, networks, etc.
     - ✓ scan types + options could be the following:
     - ✓ scan techniques:
       - -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
       - -sU: UDP Scan

- ✓ service/version detection:
  - -sV: Probe open ports to determine service/version info
- ✓ OS DETECTION:
  - -O: Enable OS detection

2.4. How to retrieve specific info about the nmap scan options?

- use command-line “man nmap”
- access to https://explainshell.com/

3. Conduct the experiment

3.1. Step 1

Start two VMs, i.e., A and B, and they locate on the same network (e.g., both A and B use “**NAT network**” of Virtualbox, not NAT). In my case, VM A uses IP address 10.0.2.9, and VM B uses IP address 10.0.2.4

3.2. Step 2

On VM A, open a terminal and create a simple http server by typing the command “**python3 -m http.server --bind 10.0.2.9 80**”

```
root@bitcoinattacker:/home/wfan# python3 -m http.server --bind 10.0.2.9 80
Serving HTTP on 10.0.2.9 port 80 (http://10.0.2.9:80/) ...
```

3.3. Step 3

On VM B, open a terminal and test if the http server on VM A is working correctly by using the following command “**wget -o - 10.0.2.9**”

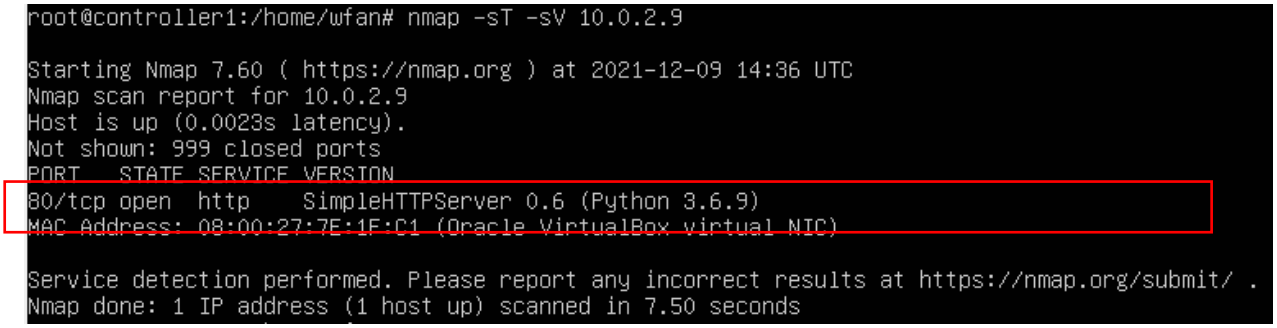
```
root@controller1:/home/wfan# wget -o - 10.0.2.9
--2021-12-09 14:32:24--  http://10.0.2.9/
Connecting to 10.0.2.9:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1937 (1.9K) [text/html]
Saving to: ‘index.html.2’

     OK .                                                     100%  265M=0s

2021-12-09 14:32:24 (265 MB/s) - ‘index.html.2’ saved [1937/1937]
```

3.4. Step 4

On VM B, scan the VM A by using “**nmap -sT -sV 10.0.2.9**”



Caution: refrain from using nmap to scan public network, which causes ethical issue.

Homework:

Follow the aforementioned lab steps to test nmap (with different scan types and options) on VM B to scan the http server on VM A, meanwhile use wireshark on VM A to observe the probing traffic sent from B to A.

Reference

[1] https://cloudcone.com/docs/article/how-to-install-wireshark-on-ubuntu-18-04-lts/