

Progetto di fine modulo 3

Scegliete da un minimo di 2 fino ad un massimo di 4 vulnerabilità critiche e provate ad implementare delle azioni di rimedio.

Le vulnerabilità critiche che ho scelto sono le seguenti:

Vulnerabilità critiche	CVSS
51988 - Bind Shell Backdoor Detection	9.8
61708 - VNC Server 'password' Password	10
11356 - NFS Exported Share Information Disclosure	10

1. Bind Shell Backdoor Detection

Questa è una vulnerabilità molto pericolosa che ci indica che una shell è in ascolta su una porta remota. Un utente malintenzionato potrebbe collegarsi e eseguire comandi da super user.

Solution

Per prima cosa ho eseguito una scansione con nmap per individuare il servizio Bindshell e la sua porta.

```
(kali@kali) [~]
$ nmap -sV 192.168.32.100
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-26 21:19 CET
Nmap scan report for 192.168.32.100
Host is up (0.0044s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    filtered http
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login?
514/tcp   open  shell        Netkit rshd
1099/tcp  open  java-rmi     GNU Classpath grmiregistry
1524/tcp  open  bindshell    Metasploitable root shell
```

```
(kali@kali)-[~]
$ nmap -p 1524 192.168.32.100
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-26 21:23 CET
Nmap scan report for 192.168.32.100
Host is up (0.0012s latency).

PORT      STATE SERVICE
1524/tcp  open  ingreslock

Nmap done: 1 IP address (1 host up) scanned in 13.09 seconds
```

Con telnet ho potuto realmente verificare la vulnerabilità e sono entrato nella shell di metasploitable attraverso la porta in ascolto.

```
(kali@kali)-[~]
$ telnet 192.168.32.100 1524
Trying 192.168.32.100 ...
Connected to 192.168.32.100.
Escape character is '^]'.
root@metasploitable:/# whoami
root
```

Per la risoluzione di questa vulnerabilità sono andato ad impostare una regola firewall su pfSense che bloccasse l'accesso alla porta 1524.

Action	Block		
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.			
Disabled	<input type="checkbox"/> Disable this rule Set this option to disable this rule without removing it from the list.		
Interface	LAN		
Choose the interface from which packets must come to match this rule.			
Address Family	IPv4		
Select the Internet Protocol version this rule applies to.			
Protocol	TCP		
Choose which IP protocol this rule should match.			
Source			
Source	<input type="checkbox"/> Invert match	Any	Source Address
Display Advanced			
The Source Port Range for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any .			
Destination			
Destination	<input type="checkbox"/> Invert match	Address or Alias	192.168.32.100
Destination Port Range	(other)	1524	(other) 1524
From Custom To Custom			
Specify the destination port or port range for this rule. The "To" field may be left empty if only filtering a single port.			
Extra Options			
Log	<input checked="" type="checkbox"/> Log packets that are handled by this rule		
Hint: the firewall has limited local log space. Don't turn on logging for everything. If doing a lot of logging, consider using a remote syslog server (see the Status: System Logs: Settings page).			

Ed ho inserito anche con iptables direttamente da metasploitable una regola che bloccasse l'accesso alla porta 1524.

```
root@metasploitable:~#  
root@metasploitable:~# iptables -A INPUT -s 192.168.32.100 -p tcp --destination-  
port 1524 -j DROP  
root@metasploitable:~# /sbin/iptables /L  
Bad argument '/L'  
Try 'iptables -h' or 'iptables --help' for more information.  
root@metasploitable:~# /sbin/iptables -L  
Chain INPUT (policy ACCEPT)  
target      prot opt source                destination          tcp dpt:ingreslock  
DROP        tcp  --  192.168.32.100         anywhere             tcp dpt:ingreslock  
  
Chain FORWARD (policy ACCEPT)  
target      prot opt source                destination  
  
Chain OUTPUT (policy ACCEPT)  
target      prot opt source                destination  
root@metasploitable:~#
```

Successivamente sono andato a verificare se la regola fosse impostata correttamente riprovando ad accedere alla porta 1524 in questione.

```
(kali㉿kali)-[~]  
$ telnet 192.168.32.100 1524 google Hacking DB - OffSec  
Trying 192.168.32.100 ...  
telnet: Unable to connect to remote host: Connection timed out
```

L'accesso come possiamo notare è stato negato e quindi possiamo dire di aver risolto la seguente vulnerabilità.

2. VNC server “password” Password

Questa vulnerabilità ci indica che la password del VNC server è debole e quindi facilmente hackerabile. Andremo quindi a modificarla con una più complessa.

```
root@metasploitable:~/.vnc# ls -la
total 68
drwx-----  2 root root  4096 2024-01-26 14:51 .
drwxr-xr-x 13 root root  4096 2024-01-26 14:51 ..
-rw-r--r--  1 root root 14368 2024-01-26 15:19 metasploitable:0.log
-rw-r--r--  1 root root    5 2024-01-26 14:51 metasploitable:0.pid
-rw-r--r--  1 root root 15236 2012-05-20 14:48 metasploitable:1.log
-rw-r--r--  1 root root 13822 2012-05-20 14:47 metasploitable:2.log
-rw-----  1 root root   16 2024-01-26 18:54 passwd
-rwxr-xr-x  1 root root   151 2012-05-20 15:16 xstartup
root@metasploitable:~/.vnc# vncpasswd
Using password file /root/.vnc/passwd
Password:
Verify:
Would you like to enter a view-only password (y/n)? y
Password:
Verify:
root@metasploitable:~/.vnc#
```

3. 11356 - NFS Exported Share Information Disclosure

Questa vulnerabilità ci indica che almeno una delle condivisioni NFS esportate dal server remoto può essere montata dall'host di scansione. Un utente malintenzionato potrebbe essere in grado di sfruttare questa funzionalità per leggere (ed eventualmente scrivere) file su un host remoto.

Per risolverla andremo a configurare NFS sull'host remoto in modo che solo gli host autorizzati possano montare le condivisioni remote.

Ho creato una cartella su kali per poter montare il file da esportare (/mnt/nfs_mount)
Su metasploitable una cartella da esportare al nome /home/user/shared con il comando mkdir.
A questo punto sono andato a modificare il file del server Nfs con il comando sudo nano /etc/exports inserendo la riga in figura.

```
GNU nano 2.0.7      File: /etc/exports

# /etc/exports: the access control list for filesystems which may be exported
#                 to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes      hostname1(rw,sync) hostname2(ro,sync)
#
# Example for NFSv4:
# /srv/nfs4       gss/krb5i(rw,sync,fsid=0,crossmnt)
# /srv/nfs4/homes gss/krb5i(rw,sync)
#
/srv/nfs/homes    192.168.50.100(r)
```

```
(kali㉿kali)-[~]
$ sudo mount -t nfs 192.168.32.100:/srv/nfs/homes /mnt/nfs_mount
```