

## REMEDIATION ACTION METASPLOITABLE 2

|                          |          |        |     |  |
|--------------------------|----------|--------|-----|--|
| <input type="checkbox"/> | CRITICAL | 10.0 * | 5.9 | NFS Exported Share Information Disclosure                |
| <input type="checkbox"/> | CRITICAL | 10.0   |     | Unix Operating System Unsupported Version Detection      |
| <input type="checkbox"/> | CRITICAL | 10.0 * |     | VNC Server 'password' Password                           |
| <input type="checkbox"/> | CRITICAL | 9.8    | 9.0 | Apache Tomcat AJP Connector Request Injection (Ghostcat) |
| <input type="checkbox"/> | CRITICAL | 9.8    |     | Bind Shell Backdoor Detection                            |
| <input type="checkbox"/> | CRITICAL | ...    | ... | 2 SSL (Multiple Issues)                                  |

  

| <input type="checkbox"/> Sev ▼ | CVSS ▼   | VPR ▼  | Name ▲  |
|--------------------------------|----------|--------|---|
| <input type="checkbox"/>       | CRITICAL | 10.0 * | 7.4 Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check) |
| <input type="checkbox"/>       | CRITICAL | 10.0 * | 7.4 Debian OpenSSH/OpenSSL Package Random Number Generator Weakness             |

Si consideri che le operazioni verranno eseguite in modalità **root** (comando **sudo su**) e che per ogni modifica effettuata la macchina Metasploitable verrà riavviata dopo, ove necessario, aver salvato tali modifiche.

### NFS EXPORTED SHARE INFORMATION DISCLOSURE

Un NFS (Network File System) è un protocollo di rete che consente a dispositivi remoti o collegati alla stessa rete di condividere directory o file, quindi consentendo di interagire come se fossero disponibili localmente. La porta di riferimento è la **2049, nfs**.

Per risolvere questa criticità andremo a modificare il file di configurazione **/etc/exports** usando il **cancellietto #** e rendendo un commento il contenuto della riga in questione, qui **\*** è un *carattere jolly* per tenere conto di un raggruppamento di nomi di dominio completi che corrispondo ad una particolare stringa di lettere, **rw** indica l'azione che consente al server NFS di usare le richieste di lettura (r) e scrittura (w), **sync** consente al server NFS di rispondere alle richieste solo dopo che le modifiche sono state confermate nell'archiviazione stabile, **no\_root\_squash** disabilita il root\_squash ciò che impediva a gli utenti root connessi da remoto di disporre dei privilegi di root, **no\_subtree\_check** disabilita il controllo della sottostruttura che ha alcuni problemi di sicurezza impliciti.

```

GNU nano 2.0.7      File: /etc/exports      Modified
# /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)
#
#/(rw,sync,no_root_squash,no_subtree_check)

```

## VNC SERVER 'PASSWOR' PASSWORD

Tramite il VNC (Virtual Network Computing) è possibile controllare un'altra macchina se collegate tra loro in una rete Lan o tramite una rete pubblica. La porta di riferimento è la **5900, vnc**.

Per evitare che ciò accada va impostata un password complessa, solitamente si consiglia una lunghezza di 12 caratteri, sia maiuscoli che minuscoli, con anche numeri e caratteri speciali. Nella macchina scansionata è stata rilevata una password "debole", quindi per risolvere questa criticità si va a modificare la password con una complessa come **ZJT!S36R5o&t**, anche se in questo caso il sistema ci avvisa che la password verrà troncata e sarà lunga solo 8 caratteri.

La password del VNC si modifica con il comando **vncpasswd**.

```
root@metasploitable:/home/msfadmin# vncpasswd
Using password file /root/.vnc/passwd
Password:
Warning: password truncated to the length of 8.
Verify:
Would you like to enter a view-only password (y/n)? n
root@metasploitable:/home/msfadmin# _
```

## BIND SHELL BACKDOOR DETECTION

Una backdoor è, come dice il nome, una sorta di porta sul retro, questa può essere usata in senso "buono" dagli amministratori per recuperare dati in caso di emergenza. Nel caso in esame però dobbiamo considerare che potrebbe essere usata in senso "cattivo", da un utente mal intenzionato che può utilizzarla per collegarsi e inviare dei comandi al sistema. La porta di riferimento è la **1524, ingreslock**.

Per risolvere questa criticità si va a modificare il file **/etc/inetd.conf** andando ad aggiungere **#** all'inizio della riga di relativa all'ingreslock, andando così a trasformarlo in un commento.

```
#<off># netbios-ssn      stream  tcp      nowait  root    /usr/sbin/tcpd
telnet                stream  tcp      nowait  telnetd /usr/sbin/tcpd /usr/sb
#<off># ftp            stream  tcp      nowait  root    /usr/sbin/tcpd
tftp                  dgram   udp      wait     nobody  /usr/sbin/tcpd /usr/sb
shell                 stream  tcp      nowait  root    /usr/sbin/tcpd /usr/sb
login                 stream  tcp      nowait  root    /usr/sbin/tcpd /usr/sb
exes                   stream  tcp      nowait  root    /usr/sbin/tcpd /usr/sb
#ingreslock            stream  tcp      nowait  root    /bin/bash bash -i
```

## REXEC SERVICE DETECTION

La traccia dell'esercizio prevedeva la risoluzione della criticità **rexecd service detection** ma secondo le mie ricerche il service rexecd è stato sostituito da telnet e ssh, vado quindi a risolvere la criticità che secondo me è pertinente utilizzando una regola firewall dato che l'esercizio ce lo consente.

| SSL (MULTI ISSUE)  |
|--|
| DEBIAN OPENSSSH/OPENSSL PACKAGE RANDOM NUMBER GENERATOR WEAKNESS             |
| DEBIAN OPENSSSH/OPENSSL PACKAGE RANDOM NUMBER GENERATOR WEAKNESS (SSL CHECK) |

È presente una debolezza nel generatore di numeri casuali utilizzato da OpenSSL su sistemi Debian e Ubuntu, alcune chiavi di crittografia sono molto più comuni di quanto dovrebbero essere, un utente malintenzionato potrebbe indovinare la chiave attraverso un attacco di forza bruta con una conoscenza minima del sistema. La porta di riferimento è la **22, ssh**.

Generalmente potremmo scaricare un aggiornamento per correggere questa criticità, ma esistono exploit per trarre vantaggio da questo difetto. Se non volessimo scaricare nulla e possiamo creare una regola firewall che blocchi il traffico sulla porta 22 che risulta aperta.

Possiamo usare **Iptables**, il firewall di Linux, che tramite lo switch **-I** ci consente di inserire una nuova regola, specificando poi il protocollo TCP, la porta 22 e l'action , **iptables -I INPUT -p tcp -s --dport 22 -j DROP**

Andremo poi ad inserire una regola uguale per la porta **25, SMTP**, usata generalmente per ricevere messaggi di posta elettronica di spam, salvando poi le modifiche con il comando **iptables-save**. Controlleremo con **nmap** se le porte sono state chiuse correttamente.

```

nmap -sS 192.168.50.102
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-04 13:22 EDT
Nmap scan report for 192.168.50.102
Host is up (0.00071s latency).
Not shown: 979 closed tcp ports (reset)
PORT      STATE      SERVICE
21/tcp    open      ftp
22/tcp    filtered  ssh
23/tcp    open      telnet
25/tcp    filtered  smtp
53/tcp    open      domain
80/tcp    open      http
111/tcp   open      rpcbind
139/tcp   open      netbios-ssn
445/tcp   open      microsoft-ds
512/tcp   open      exec
513/tcp   open      login
514/tcp   open      shell
1099/tcp  open      rmiregistry
2049/tcp  open      nfs
2121/tcp  open      ccproxy-ftp
3306/tcp  open      mysql
5900/tcp  open      vnc
6000/tcp  open      X11
6667/tcp  open      irc
8009/tcp  open      ajp13
8180/tcp  open      unknown
MAC Address: 08:00:27:02:B2:FA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 14.57 seconds

```

Infine ci occuperemo della configurazione PostgreSQL, in questo momento consente le connessioni da remoto, ma andando a modificare la configurazione presente nel percorso **/etc/postgresql/8.3/main** e aprendo il file di testo **postgresql.conf**, cambieremo il parametro **listen\_address = '\*'** con **listen\_address='localhost'**, negando a tutti (\*) gli indirizzi IP di essere connessi al server del database.

Nella stessa directory potremo modificare un altro file **pg\_hba.conf**, andando ad indicare gli utenti che vogliamo essere collegati al database, in questo caso iniserisco l'IP di Metasploitable.

```
GNU nano 2.0.7  File: /etc/postgresql/8.3/main/postgresql.conf

# CONNECTIONS AND AUTHENTICATION
#-----
# - Connection Settings -

listen_addresses = 'localhost'          # what IP address(es) to listen on;
                                         # comma-separated list of addresses;
                                         # defaults to 'localhost', '*' = all
                                         # (change requires restart)
port = 5432                             # (change requires restart)
max_connections = 100                   # (change requires restart)
# Note: Increasing max_connections costs ~400 bytes of shared memory per
# connection slot, plus lock space (see max_locks_per_transaction). You might
# also need to raise shared_buffers to support more connections.
#superuser_reserved_connections = 3     # (change requires restart)
unix_socket_directory = '/var/run/postgresql' # (change requires restart)
unix_socket_group = ''                 # (change requires restart)
unix_socket_permissions = 0777         # begin with 0 to use octal notation
                                         # (change requires restart)
#bonjour_name = ''                     # defaults to the computer name
```

```
# TYPE  DATABASE  USER  CIDR-ADDRESS  METHOD
# "local" is for Unix domain socket connections only
local   all             all                                     ident sameuser
# IPv4 local connections:
host    all             all          192.168.50.102/24  md5
# IPv6 local connections:
host    all             all          ::0/0             md5
```