

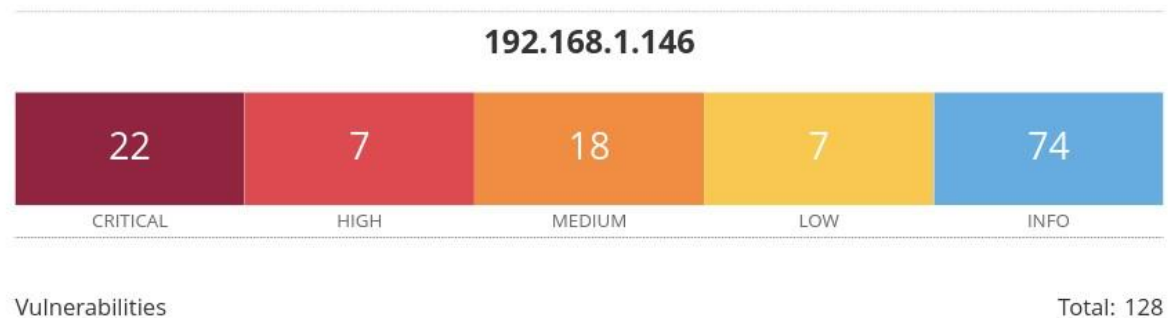
Progetto S5-L5

Studente: Simone Mininni

Obiettivo: Fase di scansione della rete o di un sistema con l'ausilio di un vulnerability scanner alla ricerca di vulnerabilità conosciute, e successivamente applicare le remediation action sulle vulnerabilità critiche e high.

Software utilizzato per la scansione è Nessus, mentre il target della scansione è la macchina Metasploitable.

Analizziamo i risultati prodotti dalla scansione iniziale:



Possiamo notare che sono state trovate diverse vulnerabilità di cui 22 critiche, 7 high, 18 medium, 7 low...

Sicuramente i risultati critici meritano la priorità assoluta e sono quelli su cui cercheremo di trovare e applicare una soluzione.

Link report Nessus iniziale:

<https://github.com/simonemin/S5-Progetto/blob/main/scansioneInizio.pdf>

Ai fini del progetto andrò a rimediare a quattro situazioni critiche/high.

1) Vulnerabilità critica: Vnc server password

CRITICAL	10.0*	-	61708	VNC Server 'password' Password
----------	-------	---	-------	--------------------------------

Vnc server , applicazioni software che permettono il controllo da remoto di una macchina. Si nota l' utilizzo di una password debole, facilmente trovabile con un attacco brute force.

Soluzione: sostituire con una password mediamente complessa.

```
root@metasploitable:/home/msfadmin# 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:/home/msfadmin# _
```

```
sudo su
vncpasswd
```

Passaggi per cambiare la password sulla metasploitable con i privilegi root.

2) Vulnerabilità critica: NFS

CRITICAL	10.0*	5.9	11356	NFS Exported Share Information Disclosure
HIGH	7.5	-	42256	NFS Shares World Readable

NFS (Network File Sharing) è un protocollo che permette di condividere file system(directories e files) sulla rete.

In questo caso vi è una criticità importante poichè un malintenzionato(client linux) potrebbe montare il file system e ispezionare tranquillamente directories file e magari anche manipolarli.

Bisogna quindi impostare NFS in modo che possa essere accessibile solo da client autorizzati.

Nel nostro caso andiamo a manipolare il file di configurazione (/etc/exports) ed a commentare la riga di condivisione della directory, in modo che nfs non condivida nessun filesystem sulla rete.

```
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)
#
#/*(rw,sync,root_squash,no_subtree_check)
```

In questo modo andiamo a risolvere le due vulnerabilità.

3)Vulnerabilità high: rlogin service detection

HIGH	7.5*	6.7	10205	rlogin Service Detection
------	------	-----	-------	--------------------------

Description

The rlogin service is running on the remote host. This service is vulnerable since data is passed between the rlogin client and server in cleartext. A man-in-the-middle attacker can exploit this to sniff logins and passwords. Also, it may allow poorly authenticated logins without passwords. If the host is vulnerable to TCP sequence number guessing (from any network) or IP spoofing (including ARP hijacking on a local network) then it may be possible to bypass authentication.

Finally, rlogin is an easy way to turn file-write access into full logins through the .rhosts or rhosts.equiv files.

Solution

Comment out the 'login' line in /etc/inetd.conf and restart the inetd process. Alternatively, disable this service and use SSH instead.

Dal report di Nessus si evince che è possibile risolvere facilmente il problema commentando la riga di login nel file /etc/inetd.conf.

```
File Macchina Visualizza Inserimento Dispositivi Aiuto
GNU nano 2.0.7 File: inetd.conf
#<off># netbios-ssn stream tcp nowait root /usr/sbin/tcpd /usr/sbin/
telnet stream tcp nowait telnetd /usr/sbin/tcpd /usr/sbin/in.telnetd
#<off># ftp stream tcp nowait root /usr/sbin/tcpd /usr/sbin/ftpd
tftp dgram udp wait nobody /usr/sbin/tcpd /usr/sbin/in.tftpd
shell stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.rshd
#login stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.rlogind
exec stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.rexecd
ingreslock stream tcp nowait root /bin/bash bash -i
```

4) Vulnerabilità critica: Bind Shell Backdoor Detection

CRITICAL	9.8	51988	Bind Shell Backdoor Detection
----------	-----	-------	-------------------------------

Bind shell backdoor permette all' attaccante di connettersi alla macchina vittima bypassando i sistemi di autenticazione e quindi prendere il controllo della stessa.

Soluzione: ho impostato una rule sul firewall iptables in modo tale da rigettare tutto il traffico in entrata sulla porta '1524' dove è in ascolto il servizio della backdoor.

comando da shell:

'sudo /sbin/iptables -A INPUT -p tcp --dport 1524 -j DROP'

```
msfadmin@metasploitable:/etc$ sudo /sbin/iptables -A INPUT -p tcp --dport 1524 -j DROP
```

```
msfadmin@metasploitable:/etc$ sudo /sbin/iptables -L -n -v
Chain INPUT (policy ACCEPT 13804 packets, 1368K bytes)
pkts bytes target      prot opt in      out     source      destination
    2    96 DROP        tcp  --  *      *       0.0.0.0/0    0.0.0.0/0
      tcp dpt:1524

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in      out     source      destination

Chain OUTPUT (policy ACCEPT 10965 packets, 1651K bytes)
pkts bytes target      prot opt in      out     source      destination
```

Risultati scansione post remediation actions



Vulnerabilities

Total: 124

Rispetto al report iniziale si nota quattro criticità in meno.

Link report finale:

<https://github.com/simonemin/S5-Progetto/blob/main/scansioneFine.pdf>