Vulnerabilty Assessment

Made by: Max Aldrovandi

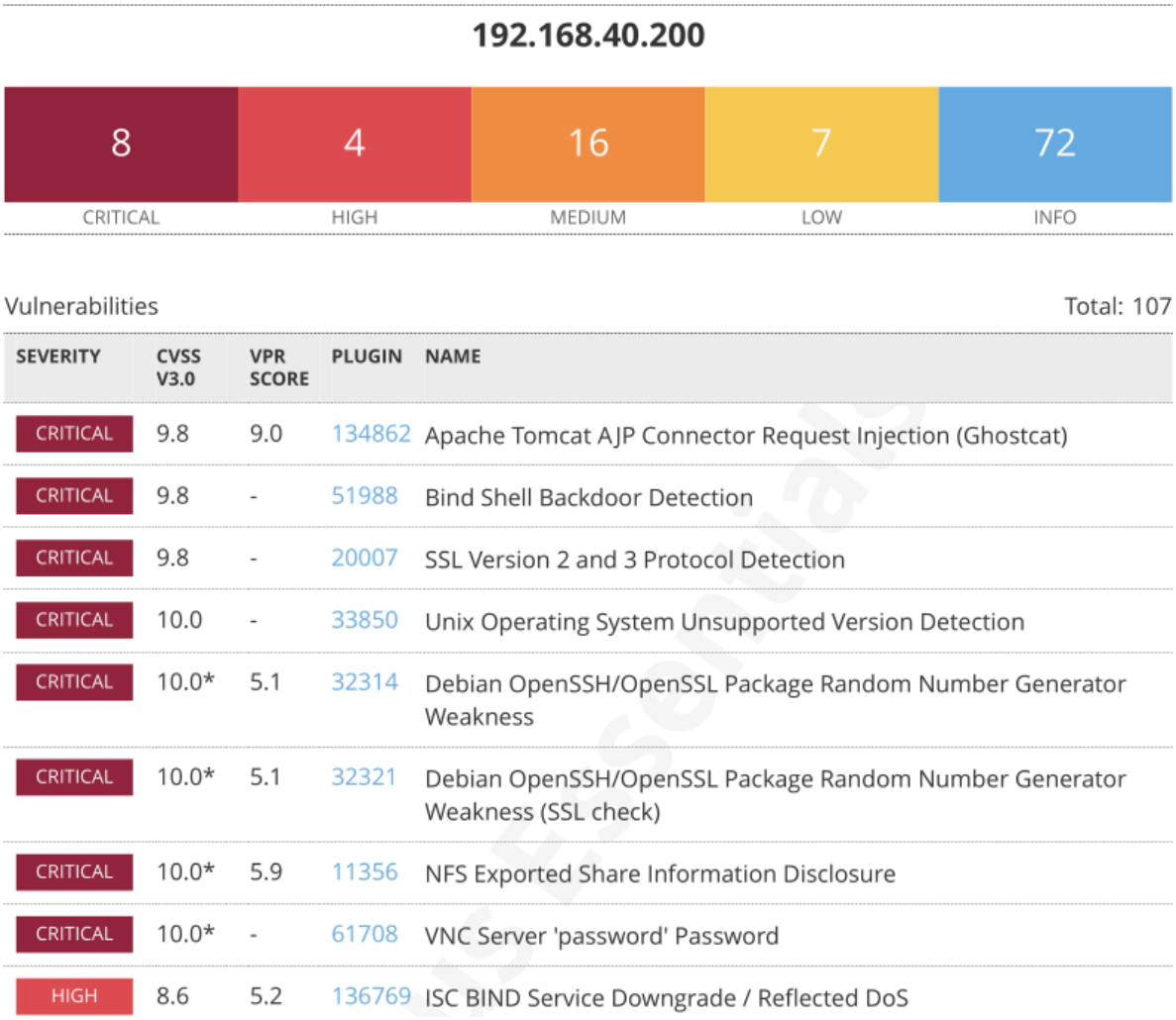
Traccia:

Effettuare una scansione completa sul target **Metasploitable**.
Scegliete da un minimo di 2 fino ad un massimo di 4 vulnerabilità **critiche / high** e provate ad implementare delle **azioni di rimedio**.

N.B. le azioni di rimedio, in questa fase, potrebbero anche essere delle regole firewall ben configurate in modo da limitare eventualmente le esposizioni dei servizi vulnerabili.
Vi consigliamo tuttavia di utilizzare magari questo approccio per non più di una vulnerabilità.

Per dimostrare l'efficacia delle azioni di rimedio, eseguite nuovamente la scansione sul target e confrontate i risultati con quelli precedentemente ottenuti.

Scansione delle vulnerabilità tramite l'utilizzo di Nessus



192.168.40.200

8	4	16	7	72
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 107

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	9.8	9.0	134862	Apache Tomcat AJP Connector Request Injection (Ghostcat)
CRITICAL	9.8	-	51988	Bind Shell Backdoor Detection
CRITICAL	9.8	-	20007	SSL Version 2 and 3 Protocol Detection
CRITICAL	10.0	-	33850	Unix Operating System Unsupported Version Detection
CRITICAL	10.0*	5.1	32314	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness
CRITICAL	10.0*	5.1	32321	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)
CRITICAL	10.0*	5.9	11356	NFS Exported Share Information Disclosure
CRITICAL	10.0*	-	61708	VNC Server 'password' Password
HIGH	8.6	5.2	136769	ISC BIND Service Downgrade / Reflected DoS

Per la scansione delle vulnerabilità abbiamo utilizzato **Nessus**, un software proprietario di tipo **client-server** che permette la scansione di numerosi tipi di **vulnerabilità**.

Le vulnerabilità in Nessus vengono classificate in ordine di pericolosità, di seguito le classi di rischio utilizzate da Nessus:

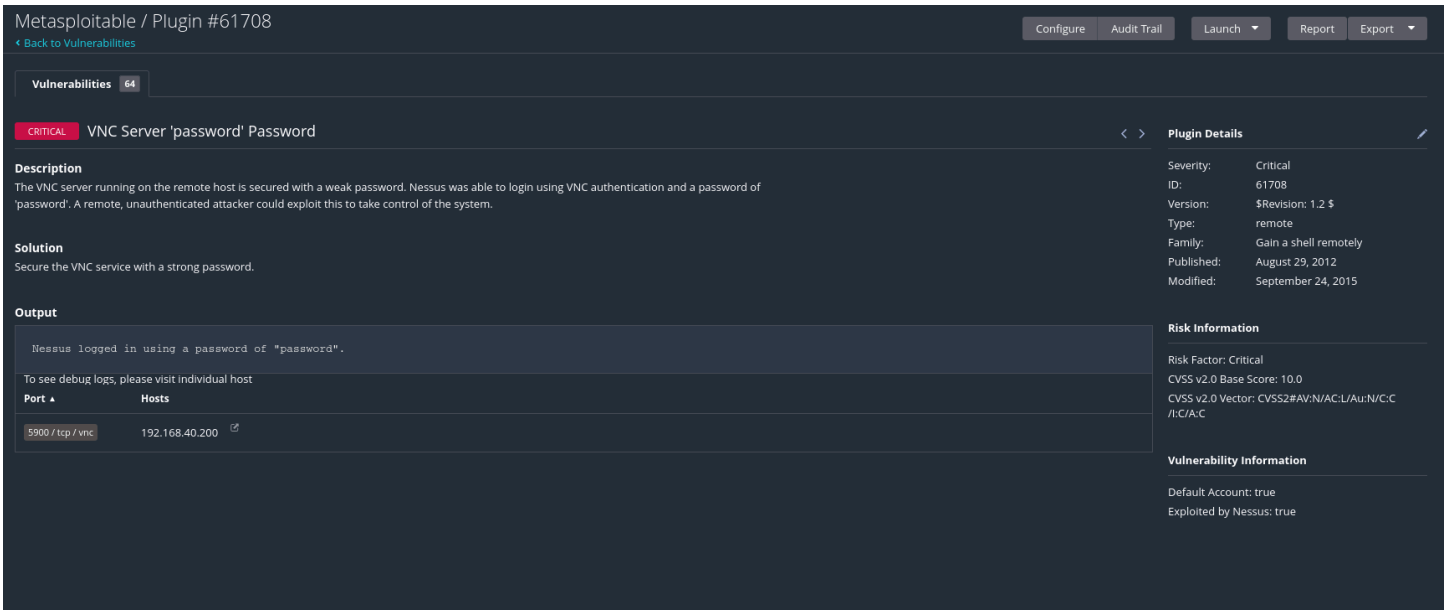
1. **Critical**: Queste sono le vulnerabilità più gravi e possono consentire a un attaccante di ottenere un accesso completo e non autorizzato al sistema o di compromettere seriamente la sicurezza del sistema.
2. **High**: Le vulnerabilità classificate come "High" rappresentano un rischio significativo e possono consentire a un attaccante di compromettere la sicurezza del sistema o di accedere a risorse sensibili.
3. **Medium**: Queste vulnerabilità rappresentano un rischio moderato e potrebbero consentire a un attaccante di ottenere un certo livello di accesso non autorizzato o di compromettere la sicurezza del sistema in misura limitata.
4. **Low**: Le vulnerabilità "Low" rappresentano un rischio relativamente minore e potrebbero non essere sfruttabili senza altre vulnerabilità o senza circostanze particolari.

5. **Info**: Questa categoria include le vulnerabilità che non consentono un accesso diretto o un compromesso del sistema, ma forniscono informazioni sensibili o utili agli attaccanti.

Ai fini del report abbiamo deciso di utilizzare la **lista sommaria** delle **vulnerabilità** fornitaci da Nessus in seguito alla nostra prima scansione della macchina virtuale '**Metasploitable**'.

Una volta identificate le vulnerabilià abbiamo proseguito col '**fixing**' delle medesime.

Fix p.'61708' "VNC Server 'password' Password"



Per 'fixare' questa vulnerabilità abbiamo proceduto con la chiusura della **porta 5900** sul terminale di **Metasploitable**, in quanto il protocollo VNC era associato a quella porta.

VNC (Virtual Network Computing) è un sistema di **accesso remoto** che consente a un utente di controllare un computer da un altro dispositivo attraverso una connessione di rete.

Nel nostro caso questa funzionalità non era neccessaria, quindi abbiamo proceduto con la chiusura diretta della porta. In futuro se dovesse essere necessario utilizzare il protocollo VNC si potrà riaprire la porta 5900 e proteggerla con una password forte ed eventualmente un firewall.

Per la chiusura della porta abbiamo utilizzato il comando:

sudo iptables -A INPUT -p tcp --dport 5900 -j DROP

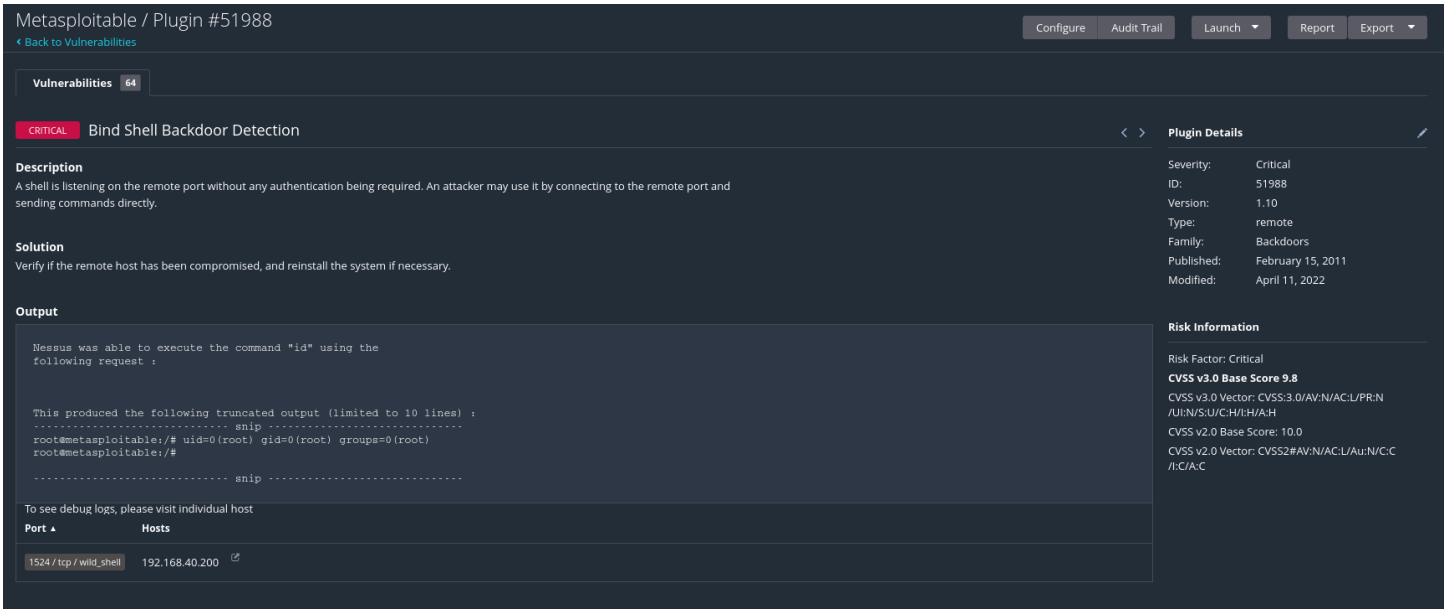
```
msfadmin@metasploitable:~$ sudo iptables -A INPUT -p tcp --dport 5900 -j DROP
```

Successivamente abbiamo salvato le impostazioni con il comando:

sudo iptables-save

```
msfadmin@metasploitable:~$ sudo iptables-save
# Generated by iptables-save v1.3.8 on Sat May 11 11:56:21 2024
*filter
:INPUT ACCEPT [18984:1978963]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [15883:2510409]
-A INPUT -p tcp -m tcp --dport 1524 -j DROP
-A INPUT -p tcp -m tcp --dport 5900 -j DROP
COMMIT
# Completed on Sat May 11 11:56:21 2024
msfadmin@metasploitable:~$
```

Fix p.'51988' “Bind Shell Backdoor Detection”



Per ‘fixare’ questa vulnerabilità abbiamo proceduto in modo analogo alla chiusura della porta 5900, in questo caso la porta interessata era la **1524**.

Pertanto per la chiusura della porta abbiamo utilizzato il comando:

sudo iptables -A INPUT -p tcp --dport 1524 -j DROP

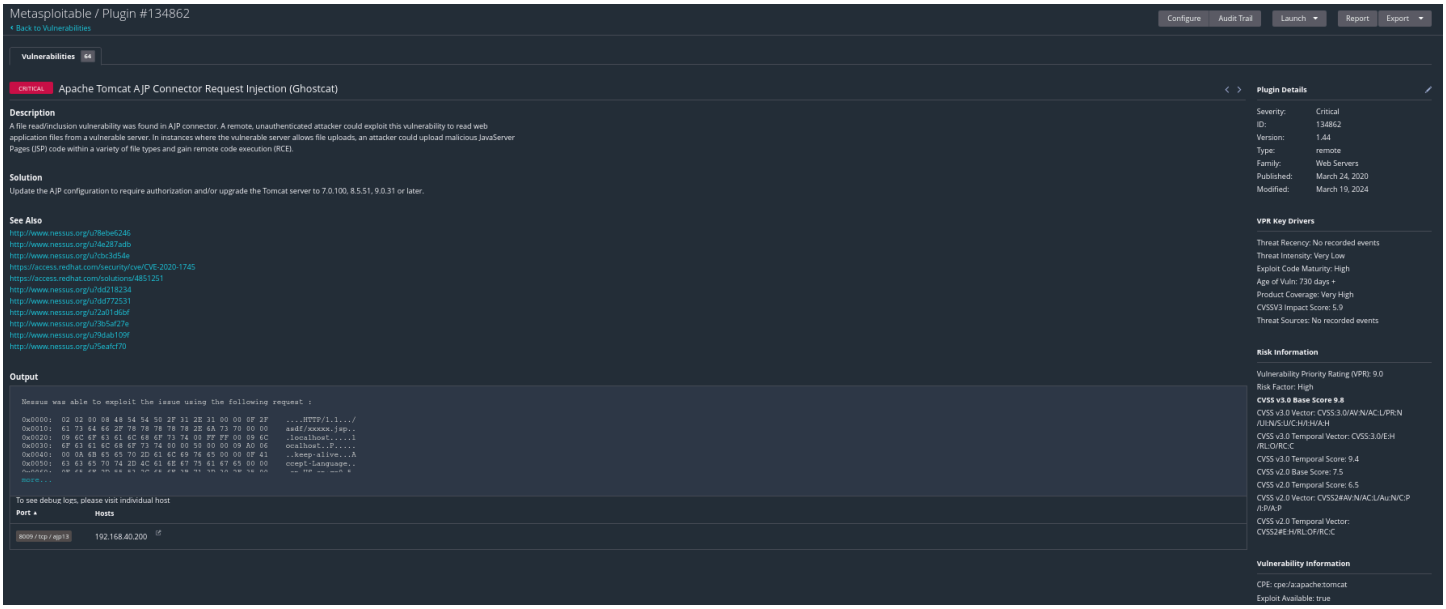
```
msfadmin@metasploitable:~$ sudo iptables -A INPUT -p tcp --dport 1524 -j DROP
```

Successivamente abbiamo salvato le impostazioni con il comando:

sudo iptables-save

```
msfadmin@metasploitable:~$ sudo iptables-save
# Generated by iptables-save v1.3.8 on Sat May 11 11:56:21 2024
*filter
:INPUT ACCEPT [18984:1978963]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [15883:2510409]
-A INPUT -p tcp -m tcp --dport 1524 -j DROP
-A INPUT -p tcp -m tcp --dport 5900 -j DROP
COMMIT
# Completed on Sat May 11 11:56:21 2024
msfadmin@metasploitable:~$
```

Fix p.'134862' “Apache Tomcat AJP Connector Request Injection (Ghostcat)”



Per ‘fixare’ questa vulnerabilità abbiamo proceduto nel seguente modo:

- Abbiamo localizzato il file ‘server.xml’ tramite il comando

  find / -name "server.xml"

- Abbiamo modificato il file ‘**server.xml**’ tramite un editor di testo, in questo caso abbiamo utilizzato il comando ‘nano’.

  Per modificare il file abbiamo inizialmente individuato nel file la parte relativa alla configurazione del **connetore AJP**, successivamente abbiamo aggiunto l'attributo ‘**requiredSecret**’ alla configurazione del connettore per **richiedere l'autorizzazione**.

```
    <!-- Define an AJP 1.3 Connector on port 8009 -->
    <Connector port="8009"
               enableLookups="false" redirectPort="8443" protocol="AJP/1.3"
               requiredSecret"msfadmin" />
```

- Infine abbiamo **riavviato** il servizio **Tomcat** tramite i comandi:

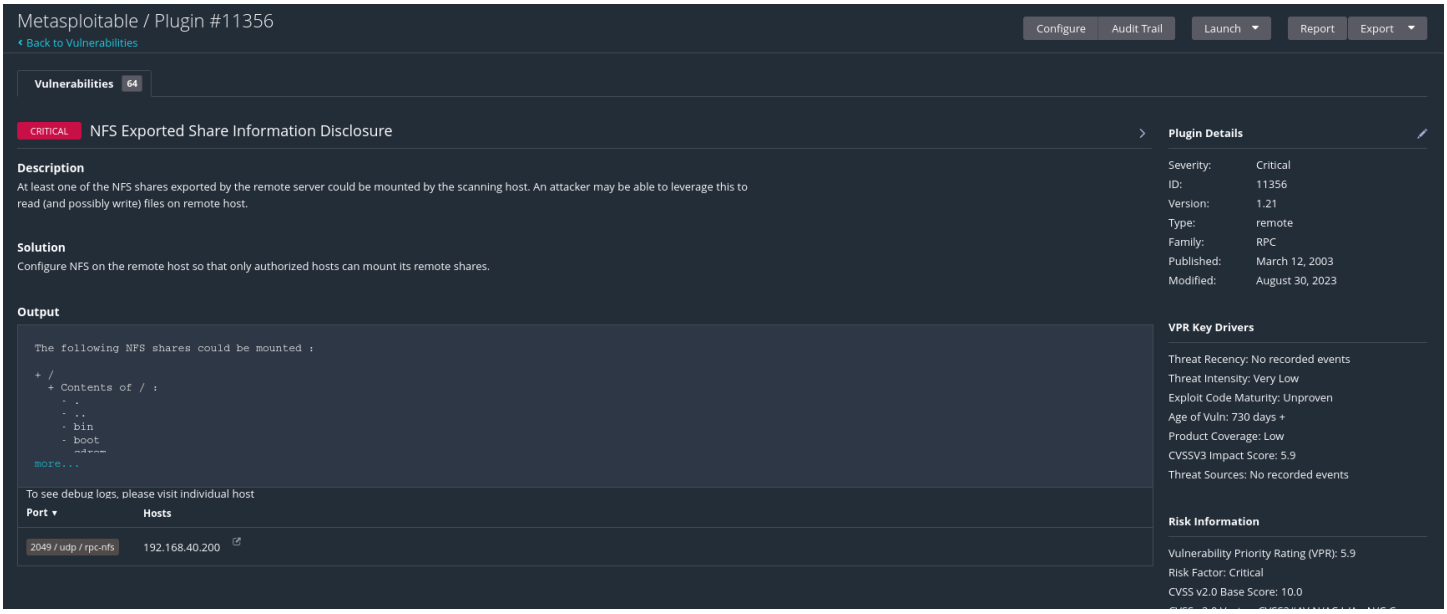
  sudo /etc/init.d/tomcat5.5 stop

  e

  sudo /etc/init.d/tomcat5.5 start

```
msfadmin@metasploitable:~$ sudo /etc/init.d/tomcat5.5 stop
 * Stopping Tomcat servlet engine tomcat5.5                      [ OK ]
msfadmin@metasploitable:~$ sudo /etc/init.d/tomcat5.5 start
 * Starting Tomcat servlet engine tomcat5.5                      [ OK ]
msfadmin@metasploitable:~$
```

Fix p.'11356' “NFS Exported Share Information Disclosure”



Per ‘fixare’ questa vulnerabillità abbiamo agito in due modi:

- Abbiamo limitato la lista dei volumi condivisibili e abbiamo creato una cartella fittizia tramite i comandi:

  sudo nano /etc/exports (lista dei volumi condivisibili)

  / *(rw,sync,no_root_squash,no_subtree_check) (volumi inizialmente condivisi)

/srv/nfs (creazione cartella fittizia)

/srv/nfs *(rw, sync) (volumi condivisi aggiornati)

```
  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,no_root_squash,no_subtree_check)









                            [ Read 12 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

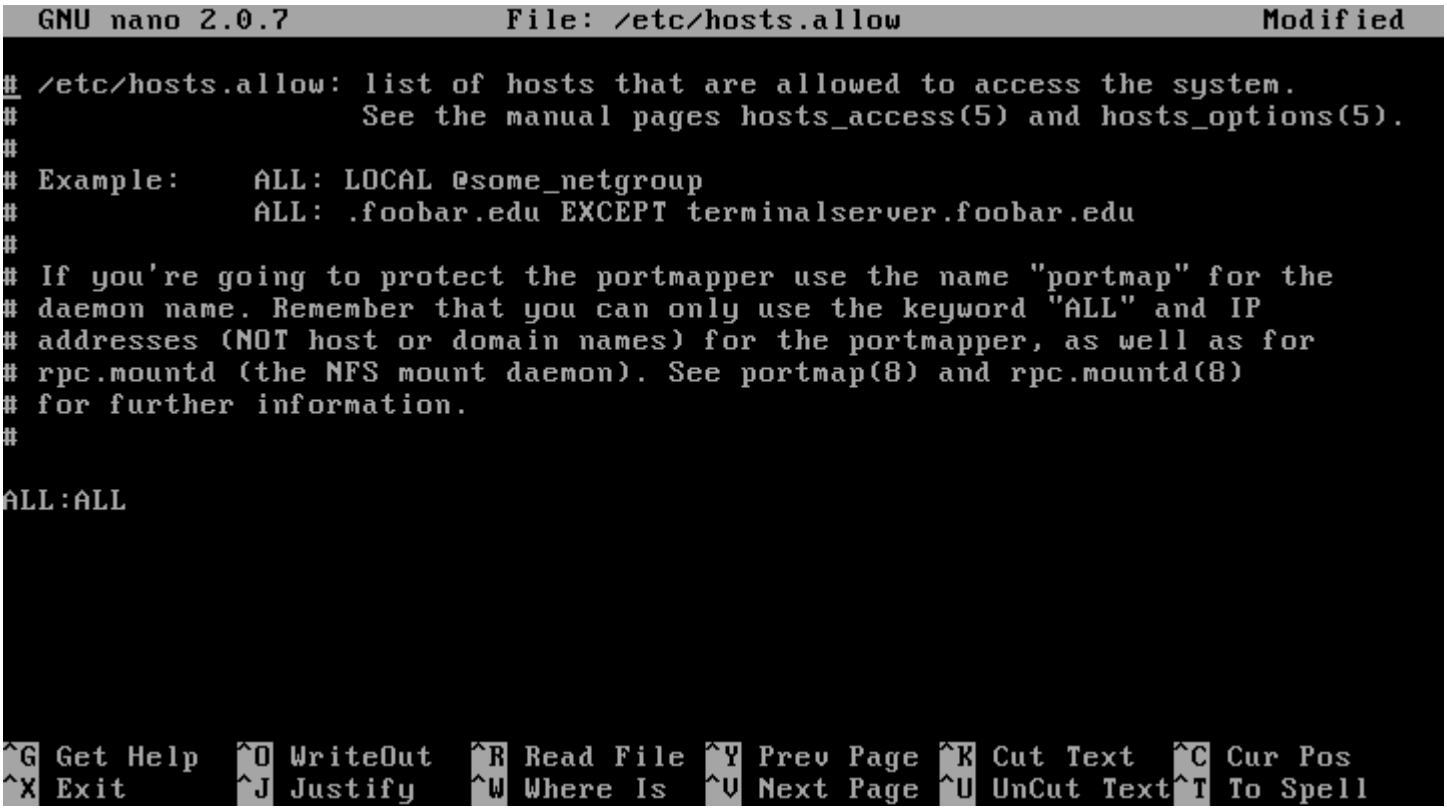
volumi condivisibili iniziali

```
  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)
#

/svr/nfs   *(rw,sync)_









                            [ Wrote 12 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

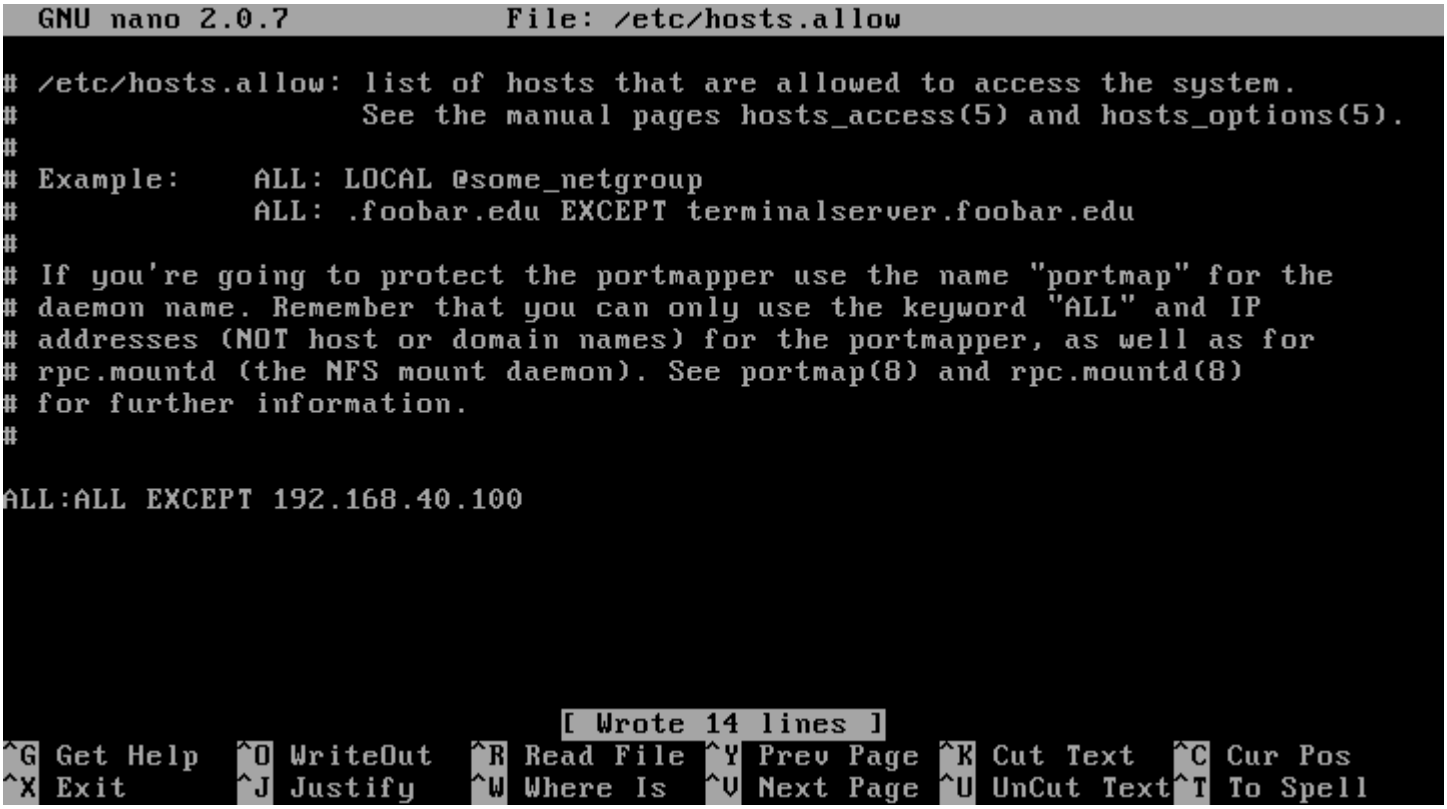
volumi condivisibili aggiornati

- Abbiamo limitato la lista degli host che potevano accedere alle share condivise tramite il comando

  sudo nano /etc/hosts.allow



host autorizzati iniziali



Nuova scansione per verificare la reale risoluzione delle vulnerabilità

Abbiamo infine proceduto con una **nuova scansione** con Nessus per verificare il reale '**fixaggio**' delle **vulnerabilità** sopra elencate.

192.168.40.200

4	4	16	7	68
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 99

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	9.8	-	20007	SSL Version 2 and 3 Protocol Detection
CRITICAL	10.0	-	33850	Unix Operating System Unsupported Version Detection
CRITICAL	10.0*	5.1	32314	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness
CRITICAL	10.0*	5.1	32321	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)
HIGH	8.6	5.2	136769	ISC BIND Service Downgrade / Reflected DoS
HIGH	7.5	-	42256	NFS Shares World Readable
HIGH	7.5	5.1	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)
HIGH	7.5	5.9	90509	Samba Badlock Vulnerability
MEDIUM	6.5	3.6	139915	ISC BIND 9.x < 9.11.22, 9.12.x < 9.16.6, 9.17.x < 9.17.4 DoS

E come previsto le vulnerabilità che abbiamo 'fixato' non sono più presenti.