

# Progetto Modulo 3

AZIONI DI RIMEDIO PER LE VULNERABILITA' RILEVATE SU METASPLOITABLE 2

# La Traccia

Effettuare una scansione completa sul target Metasploitable.

Scegliete da un minimo di 2 fino ad un massimo di 4 vulnerabilità **critiche** 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.

# La Consegna

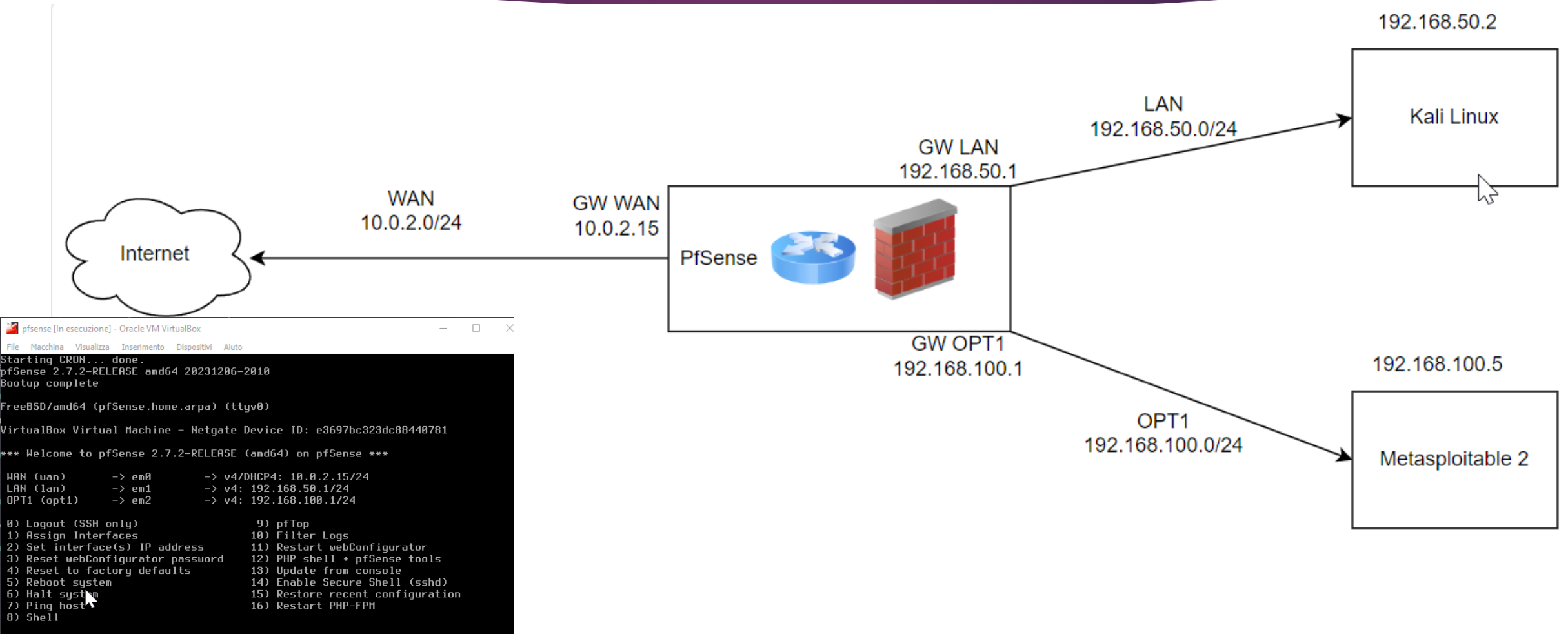
1. Scansione iniziale dove si vede il grafico con tutte le vulnerabilità e le vulnerabilità da risolvere (tecnico, già riassunto) - **ScansioneInizio.pdf**
2. **Screenshot e spiegazione dei passaggi della remediation - RemediationMeta.pdf**
3. Scansione dopo le modifiche che evidenzia la risoluzione dei problemi/vulnerabilità (il grafico che mostra tutte le vulnerabilità) - **ScansioneFine.pdf**

**Oppure un report unico, a vostra scelta. Penso sia più comodo farne tre comunque.**

**Nota: i report possono essere lasciati in inglese, senza problemi.**

Se risolvete le 4 vulnerabilità, potete risolverne una quinta (a scelta), ad esempio con una regola di firewall

# Il Laboratorio



# Remediation: Introduzione e premesse.

Dall'analisi della macchina Metasploitable 2 risultano diverse vulnerabilità come evidenziato nel file **Scansionelinizio.pdf** .

Di seguito analizzeremo le azioni di Remediation di alcune delle vulnerabilità **CRITICHE**.

Ogni vulnerabilità verrà introdotta in base alla denominazione presente nel report **Scansionelinizio.pdf** e verranno evidenziate le azioni attuate al fine di realizzare la Remediation.

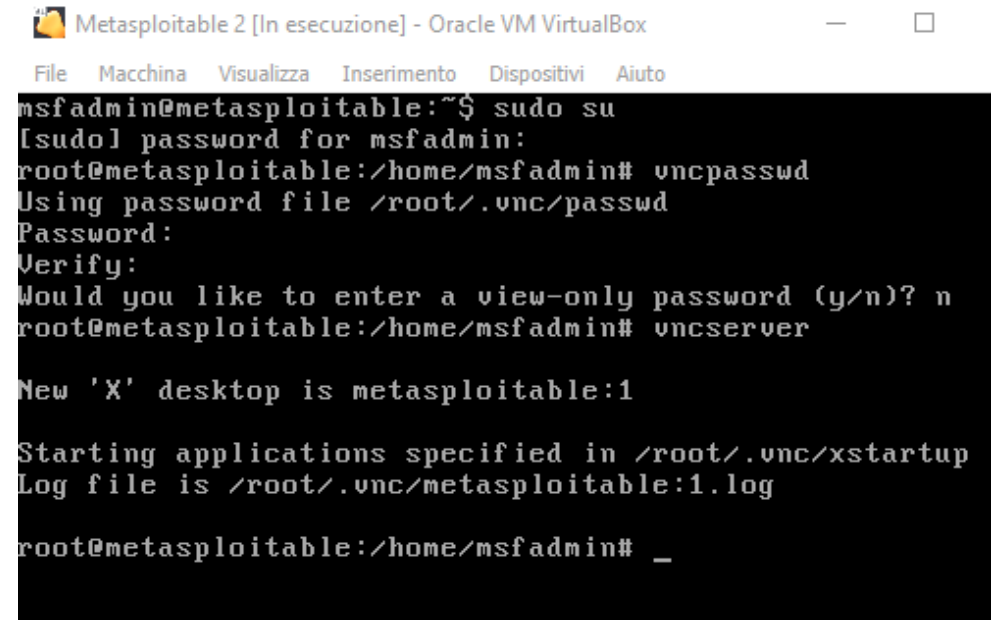
Per «Kali» si intenderà la macchina Kali Linux utilizzata per testare le azioni di Remediation, per «Meta» si intenderà la macchina Metasploitable 2 ovvero la macchina oggetto dell'analisi.

Per la creazione delle password verranno seguite le guidelines dello standard **NIST SP 800-63-3**.

# 1) VNC Server 'password' Password

La prima vulnerabilità in analisi è legata al servizio Vnc Server di Meta, tale servizio è accessibile tramite una password debole, l'azione di Remediation in tal caso sarà quella di modificare la password di tale servizio scegliendone una più robusta.

Per prima cosa accediamo come utente 'root', poi utilizziamo il comando 'vncpasswd' per modificare la password, la nuova sarà 'Ep1c0d3.'. Fatto ciò avvieremo il servizio e verificheremo da Kali se riusciamo ad accedere al servizio con la nuova password.



```
Metasploitable 2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
msfadmin@metasploitable:~$ sudo su
[sudo] password for msfadmin:
root@metasploitable:/home/msfadmin# vncpasswd
Using password file /root/.vnc/passwd
Password:
Verify:
Would you like to enter a view-only password (y/n)? n
root@metasploitable:/home/msfadmin# vncserver

New 'X' desktop is metasploitable:1

Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/metasploitable:1.log

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

# 1) VNC Server 'password' Password

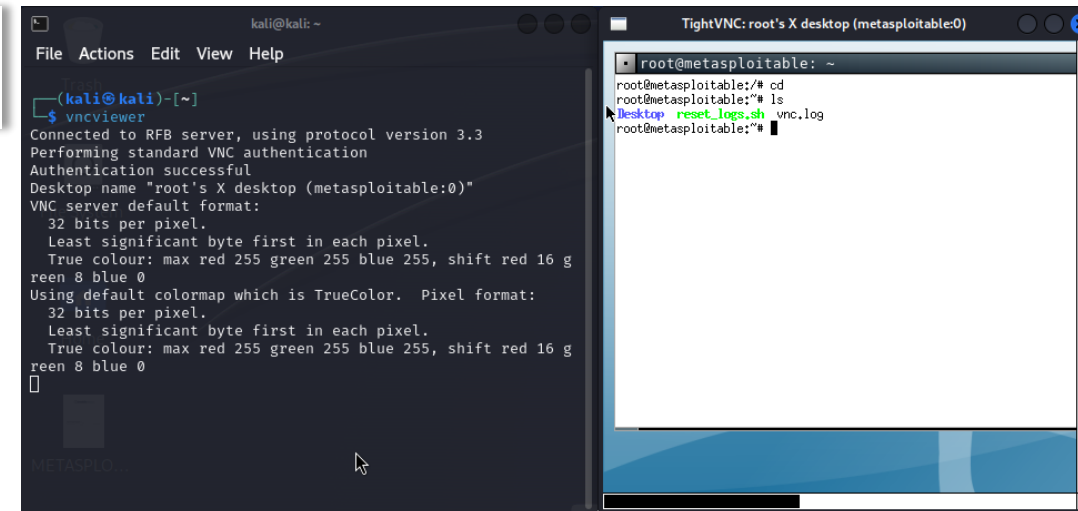
Per accedere al servizio vnc da Kali usiamo il comando 'vncviewer' proviamo prima ad inserire la vecchia password cioè 'password', ci da accesso negato. Inseriamo la nuova 'Ep1c0d3.', date le modifiche effettuate si collegherà solo nel secondo caso.



1)

```
(kali@kali)-[~]  
$ vncviewer  
Connected to RFB server, using protocol version 3.3  
Performing standard VNC authentication  
Authentication failure
```

2)



## 2) Bind Shell Backdoor Detection

Nella descrizione di tale vulnerabilità abbiamo il seguente testo:

“A shell is listening on the remote port **without any authentication** being required. An attacker may use it by connecting to the remote port and sending commands directly.»

Abbiamo una ‘wild shell’ sulla porta 1524 senza autenticazione, ad esempio da kali con un semplice comando di netcat posso collegarmi a tale shell senza usare password ed accedere al sistema.

Nel caso fosse una shell poco utilizzata potrebbe aver senso chiudere la porta ma visto che con un meccanismo di autenticazione possiamo sfruttare ancora il servizio ma con una protezione opteremo per la seconda.

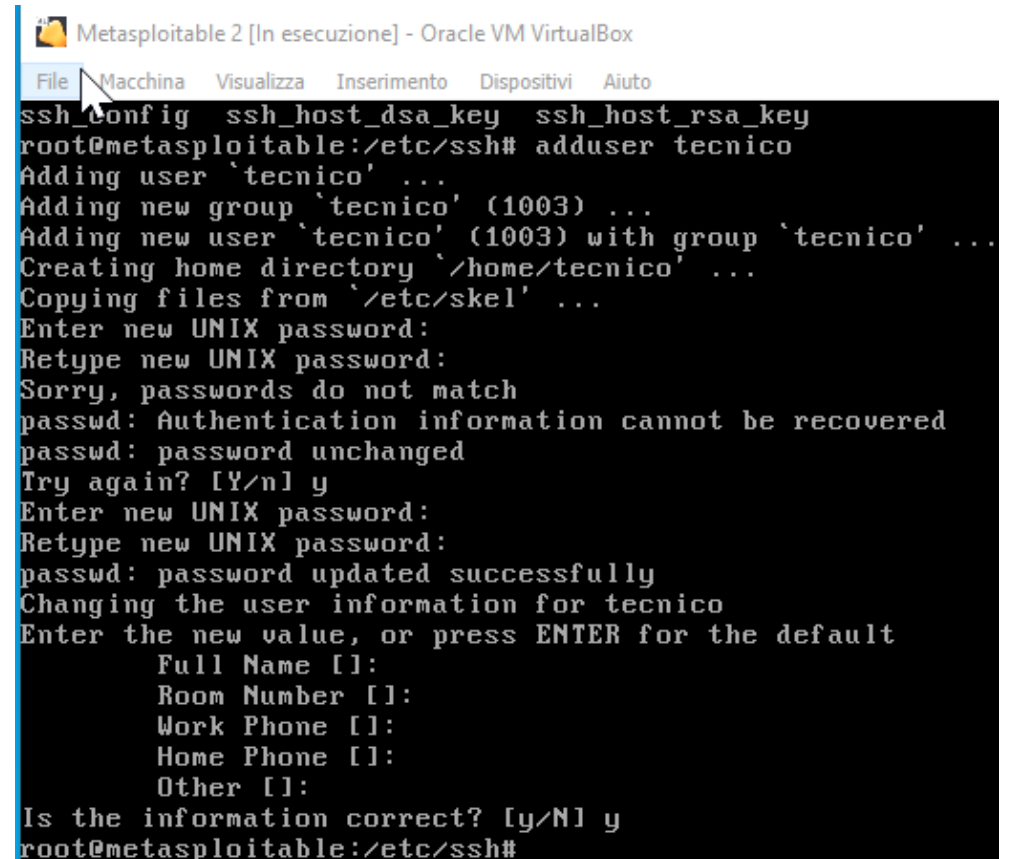


## 2) Bind Shell Backdoor Detection

Pensiamo al caso in cui vogliamo lasciare tale servizio in essere per permettere a dei tecnici da remoto di operarci su.

Avrebbe senso creare innanzitutto una utenza esclusiva per tali tecnici (utente 'tecnico') con relativa password e permettere solo a questi di accedere al sistema in questo modo.

Creiamo l'utente col comando 'adduser' di nome 'tecnico' con password 'T3c.n1c0'.



```
Metasploitable 2 [In esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
ssh_config ssh_host_dsa_key ssh_host_rsa_key
root@metasploitable:/etc/ssh# adduser tecnico
Adding user `tecnico' ...
Adding new group `tecnico' (1003) ...
Adding new user `tecnico' (1003) with group `tecnico' ...
Creating home directory `/home/tecnico' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
Sorry, passwords do not match
passwd: Authentication information cannot be recovered
passwd: password unchanged
Try again? [Y/n] y
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for tecnico
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [y/N] y
root@metasploitable:/etc/ssh#
```

## 2) Bind Shell Backdoor Detection

Al momento il servizio è ancora accessibile senza autenticazione.(1)

1)

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ nc 192.168.100.5 1524  
root@metasploitable:/#
```

Per cui andiamo a modificare il file 'sshd\_config' (2)...

2)

```
root@metasploitable:/etc/ssh# nano sshd_config
```

## 2) Bind Shell Backdoor Detection

...in modo da attivare il servizio ssh sulla porta 1524(3) con autenticazione(4) esclusiva per l'utente 'tecnico'(5).

3)

```
GNU nano 2.0.7      File: sshd_config
# Package generated configuration file
# See the sshd(8) manpage for details

# What ports, IPs and protocols we listen for
# Port 1524
# Use these options to restrict which interface
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
```

4)

```
GNU nano 2.0.7      File: sshd_config
# To enable empty passwords, change to yes (NOT RECOMMENDED)
PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware of
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Change to no to disable tunnelled clear text passwords
PasswordAuthentication yes
```

5)

```
GNU nano 2.0.7      File: sshd_config
X11Forwarding yes
X11DisplayOffset 10
PrintMotd no
PrintLastLog yes
TCPKeepAlive yes
#UseLogin no

#MaxStartups 10:30:60
#Banner /etc/issue.net

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

Subsystem sftp /usr/lib/openssh/sftp-server

UsePAM yes

AllowUsers tecnico_
```

Riavvio Meta per aggiornare i servizi, riprovo su Kali ad accedere alla shell senza autenticazione, come vediamo l'accesso mi viene negato con netcat in quanto non si aspetta quel protocollo su quella porta. (6)

6)

```
(kali@kali)-[~]
$ nc 192.168.100.5 1524
SSH-2.0-OpenSSH_4.7p1 Debian-8ubuntu1
s
Protocol mismatch.
```

## 2) Bind Shell Backdoor Detection

Provo poi col servizio ssh a connettermi e verifico che funzioni. In un caso con l'utente principale 'msfadmin' per verificare che invece funzioni solo con l'accesso esclusivo dell'utente 'tecnico'.

7)

```
(kali㉿kali)-[~]  
$ ssh -oHostKeyAlgorithms=ssh-rsa -p 1524 msfadmin@192.168.100.5  
The authenticity of host '[192.168.100.5]:1524 ([192.168.100.5]:1524)' can't be established.  
RSA key fingerprint is SHA256:BQHm5EoHX9GCiOLuVscegPXLQ0suPs+E9d/rrJB84rk.  
This host key is known by the following other names/addresses:  
  ~/.ssh/known_hosts:1: [hashed name]  
Are you sure you want to continue connecting (yes/no/[fingerprint])? y  
Please type 'yes', 'no' or the fingerprint: yes  
Warning: Permanently added '[192.168.100.5]:1524' (RSA) to the list of known hosts.  
msfadmin@192.168.100.5's password:  
Permission denied, please try again.  
msfadmin@192.168.100.5's password:
```

8)

```
(kali㉿kali)-[~]  
$ ssh -oHostKeyAlgorithms=ssh-rsa -p 1524 tecnico@192.168.100.5  
tecnico@192.168.100.5's password:  
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
tecnico@metasploitable:~$
```

### 3) NFS Exported Share Information Disclosure

Il servizio NFS permette al server in sostanza di mettere un drive di Meta a disposizione della rete facendo sì che utenti da remoto possano montare l'immagine del drive sul proprio dispositivo e accedere ai contenuti di tale drive condiviso.

Per come è configurato al momento il servizio presenta una vulnerabilità:

*"An attacker may be able to leverage this to read (and possibly write) files on remote host.»*

Al momento il drive condiviso è la root, la prima azione è quella di mettere in condivisione un percorso in particolare di Meta in modo che in ogni caso limitiamo la condivisione a una parte del sistema.

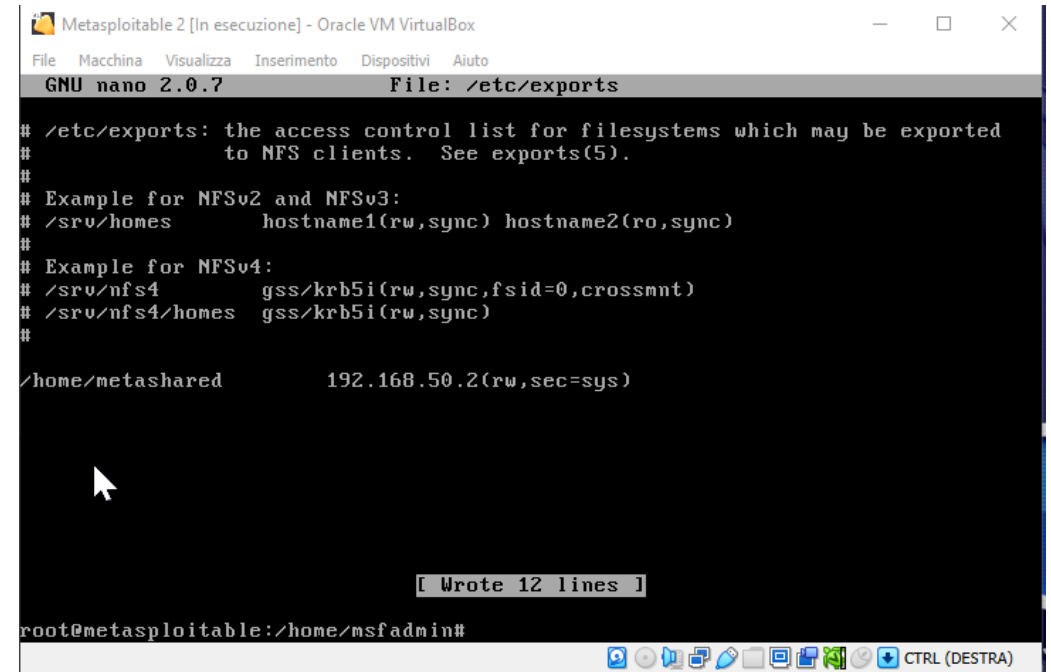
Imposteremo un ip del client che potrà connettersi al servizio, in questo caso a titolo esemplificativo meta.

Infatti la solution di Nessus suggeriva:

*"Configure NFS on the remote host so that only authorized hosts can mount its remote shares.»*

### 3) NFS Exported Share Information Disclosure

Modifichiamo il file 'exports' in modo che indichiamo il nuovo percorso del drive da condividere e accanto indichiamo l'unico utente che potrà montare il drive condiviso sul proprio dispositivo.



The screenshot shows a terminal window titled 'Metasploitable 2 [In esecuzione] - Oracle VM VirtualBox'. The terminal is running the GNU nano 2.0.7 text editor, editing the file /etc/exports. The file content is as follows:

```
# /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)
#
/home/metashared 192.168.50.2(rw,sec=sys)
```

At the bottom of the terminal, a status bar indicates '[ Wrote 12 lines ]' and the prompt shows the user is root@metasploitable:/home/msfadmin#.

# 3) NFS Exported Share Information Disclosure

Su Kali proviamo a montare l'immagine, entrare nel drive condiviso e vedere se riusciamo ad accedere ai file del drive condiviso.

In 'metashared', drive condiviso da meta, abbiamo solo il file 'prova'.(1)

Digitiamo il comando per montare il drive condiviso sulla cartella locale di kali 'drivecondiviso'.(2)

Vediamo che in 'drivecondiviso' c'è il file 'prova'.(3)

1)

```
Metasploitable 2 [In esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
root@metasploitable:/home/msfadmin# ls
vulnerable
root@metasploitable:/home/msfadmin# cd ..
root@metasploitable:/home# ls
ftp metashared msfadmin service share tecnico user
root@metasploitable:/home# cd metashared/
root@metasploitable:/home/metashared# ls
prova
root@metasploitable:/home/metashared# _
```

2)

```
kali@kali: ~/drivecondiviso
File Actions Edit View Help
(kali@kali)~[~/drivecondiviso]
$ sudo mount -t nfs -o sec=sys 192.168.100.5:/home/metashared /home/kali/driv
econdiviso
(kali@kali)~[~/drivecondiviso]
$ cd drivecondiviso
cd: no such file or directory: drivecondiviso
(kali@kali)~[~/drivecondiviso]
$ ls
prova
(kali@kali)~[~/drivecondiviso]
$
```

3)

## 4) SSL Version 2 and 3 Protocol Detection

Per risolvere tale vulnerabilità ho dovuto disabilitare le connessioni di tipo SSL all'interno dei file di configurazione dei servizi che utilizzano tale protocollo ormai obsoleto e non sicuro.

I servizi in questione erano 'Postgres', 'Postfix' e 'apache2'

```
Metasploitable 2 [In esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
GNU nano 2.0.7 File: postgresql.conf

# (change requires restart)

# - Security and Authentication -
#authentication_timeout = 1min      # 1s-600s
ssl = false                          # (change requires restart)
#ssl_ciphers = 'ALL:!ADH:!LOW:!EXP:!MD5:@STRENGTH' # allowed SSL ciphers
# (change requires restart)

#password_encryption = on
#db_user_namespace = off

# Kerberos and GSSAPI
#krb_server_keyfile = ''             # (change requires restart)
#krb_srvname = 'postgres'            # (change requires restart, Kerberos on$
#krb_server_hostname = ''            # empty string matches any keytab entry
# (change requires restart, Kerberos on$
#krb_caseins_users = off             # (change requires restart)
#krb_realm = ''                      # (change requires restart)

# - TCP Keepalives -

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

```
Metasploitable 2 [In esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
GNU nano 2.0.7 File: ssl.conf

#SSLSessionCache      dbm:/var/run/apache2/ssl_scache
#SSLSessionCache      shmcb:/var/run/apache2/ssl_scache(512000)
#SSLSessionCacheTimeout 300

# Semaphore:
# Configure the path to the mutual exclusion semaphore the
# SSL engine uses internally for inter-process synchronization.
#SSLMutex file:/var/run/apache2/ssl_mutex

# SSL Cipher Suite:
# List the ciphers that the client is permitted to negotiate.
# See the mod_ssl documentation for a complete list.
#SSLCipherSuite ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
# enable only secure ciphers:
#SSLCipherSuite HIGH:MEDIUM:!ADH

# enable only secure protocols: SSLv3 and TLSv1, but not SSLv2
SSLProtocol all -SSLv2 -SSLv3 +TLSv1 +TLSv1.2

</IfModule>
root@metasploitable:/etc/apache2/mods-available#
```

```
Metasploitable 2 [In esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
GNU nano 2.0.7 File: main.cf Modified

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# TLS parameters
#smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
#smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package for
# information on enabling SSL in the smtp client.

myhostname = metasploitable.localdomain

^G Get Help ^O WriteOut ^R Read File ^V Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^U Where Is ^N Next Page ^O UnCut Text ^T To Spell
CTRL (DESTRA)
```



## 5) Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)

Avendo risolto la vulnerabilità 'Bind Shell Backdoor Detection' 2) ed avendo attribuito la ssh alla porta 1524 tale vulnerabilità non è più in essere.

```
GNU nano 2.0.7      File: sshd_config
# Package generated configuration file
# See the sshd(8) manpage for details

# What ports, IPs and protocols we listen for
Port 1524
# Use these options to restrict which interfaces
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
```

# Conclusioni

Rimangono ancora alcune vulnerabilità critiche che possiamo evincere dal file **ScansioneFine.pdf**, altre invece sono state risolte.