

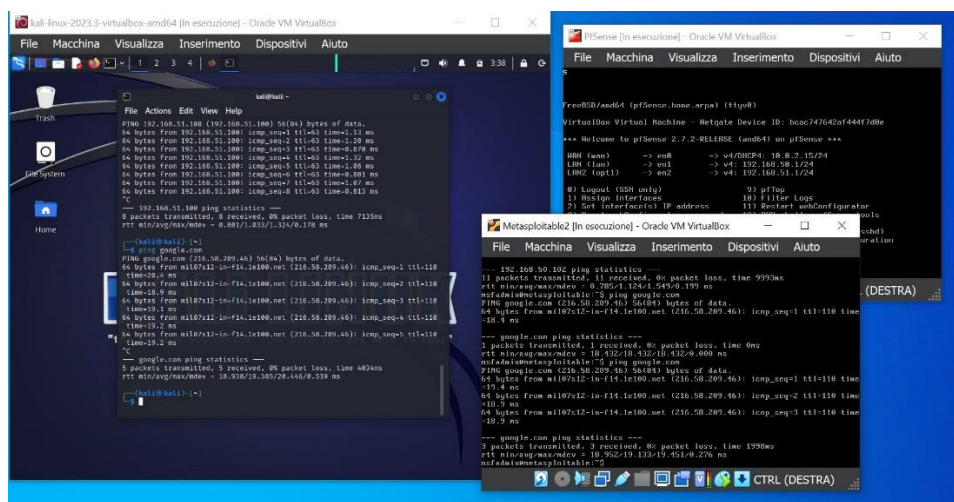
# Remediation Metasploitable

## Configurazione

Prima di elencare le remediation applicate alle vulnerabilità critiche trovate, si riassume brevemente la configurazione delle macchine:

- **Kali Linux:** è la macchina attaccante con indirizzo IP 192.168.50.102, dalla quale sono state eseguite le scansioni con Nessus;
- **Metasploitable:** è la macchina target con indirizzo IP 192.168.51.100;
- **PfSense:** configurato per fare da router tra le due macchine appartenenti a reti diverse. Dunque sono presenti una rete WAN, una rete LAN 192.168.50.1/24 e una rete LAN2 192.168.51.1/24;

Fatti i corretti collegamenti, si è verificata la reciproca connettività tra le macchine ed anche la connettività delle macchine verso internet.



La scansione ha evidenziato, come visibile dal report, la presenza di 11 vulnerabilità critiche; ne sono state analizzate 4 per le quali sono state svolte remediation che hanno portato alla risoluzione di tali vulnerabilità, come si evidenzia nel report della scansione finale. Inoltre è stata presa in considerazione anche una quinta vulnerabilità, per la quale sono state eseguite diverse azioni per risolverla, ma che evidentemente non sono state sufficienti. Ai fini del progetto, ho deciso di mostrare comunque il ragionamento svolto. Si riporta nella figura sottostante, l'elenco delle vulnerabilità analizzate (in giallo).

Sev	CVSS	VPR	Name	Family	Count	
CRITICAL	10.0 *	5.9	NFS Exported Shar...	RPC	1	
CRITICAL	10.0		Unix Operating Sy...	General	1	
CRITICAL	10.0 *		VNC Server 'passw...	Gain a shell remotely	1	
CRITICAL	9.8		SSL Version 2 and ...	Service detection	2	
CRITICAL	9.8		Bind Shell Backdo...	Backdoors	1	
MIXED	...	...	Apache Tomc...	Web Servers	4	
CRITICAL	...	...	SSL (Multiple ...	Gain a shell remotely	3	
HIGH	7.5		NFS Shares World ...	RPC	1	
HIGH	7.5 *	5.9	rlogin Service Dete...	Service detection	1	
HIGH	7.5 *	5.9	rsh Service Detecti...	Service detection	1	
HIGH	7.5	6.7	Samba Badlock Vu...	General	1	

## Remediation per la vulnerabilità NFS Exported Share Information Disclosure (ID 11356)

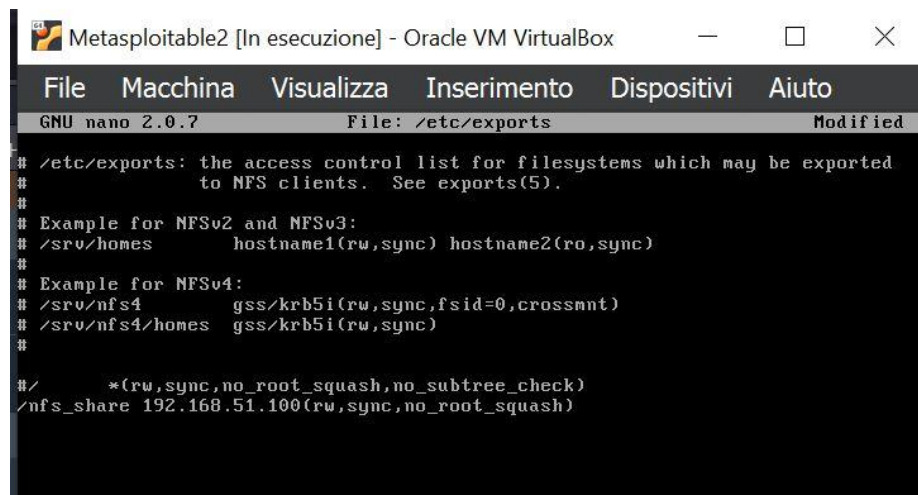
Tale vulnerabilità indica che è possibile accedere alle condivisioni NFS sull'host remoto e almeno una delle condivisioni NFS esportate dal server remoto può essere montata dall'host di scansione. Un attaccante potrebbe essere in grado di sfruttare questa possibilità per leggere (ed eventualmente scrivere) i file sull'host remoto. Bisogna dunque configurare NFS sull'host remoto in modo che solo gli host autorizzati possano montare le sue condivisioni remote.

Innanzitutto si verifica che effettivamente il servizio NFS è attivo sulla macchina:

```
msfadmin@metasploitable:~$ sudo /etc/init.d/nfs-kernel-server status
nfsd running
msfadmin@metasploitable:~$ _
```

Successivamente vanno modificati opportunamente dei file in modo da:

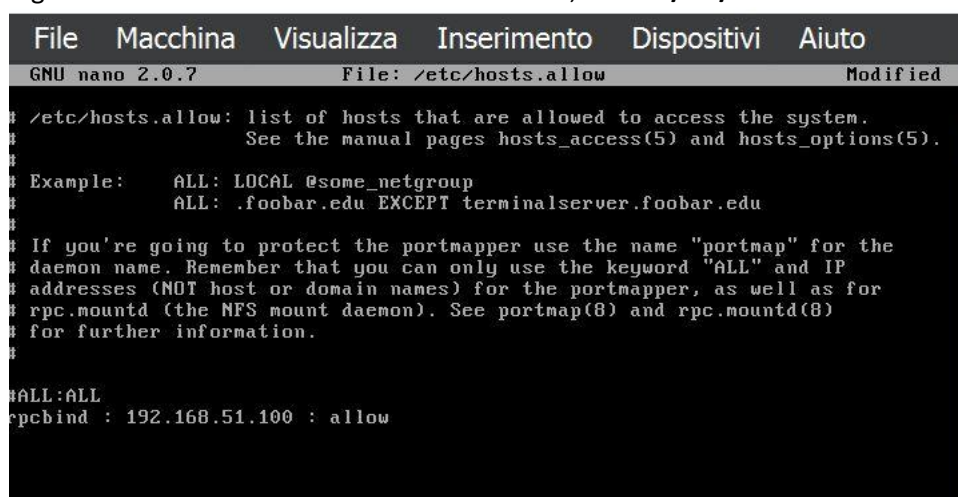
- Limitare le condivisioni NFS solo agli host autorizzati, modificando il file **/etc/exports**:



```
Metasploitable2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
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)
/nfs_share 192.168.51.100(rw,sync,no_root_squash)
```

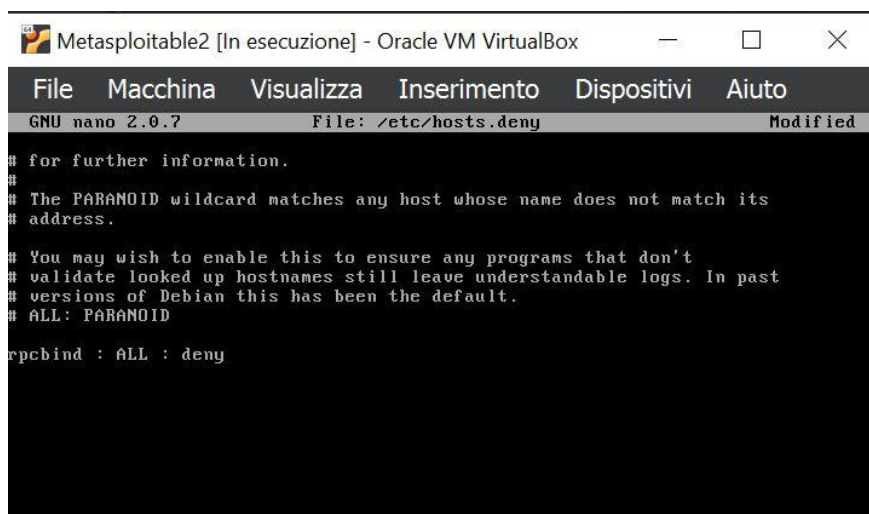
Per fare ciò è dapprima stata creata la directory tramite comando **mkdir /nfs\_share** per la quale è autorizzata solo la macchina Metasploitable;

- Specificare gli host autorizzati a connettersi al server NFS, nel file **/etc/hosts.allow**:



```
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
GNU nano 2.0.7      File: /etc/hosts.allow  Modified
# /etc/hosts.allow: list of hosts that are allowed to access the system.
#                  See the manual pages hosts_access(5) and hosts_options(5).
#
# Example:      ALL: LOCAL @some_netgroup
#              ALL: .foobar.edu EXCEPT terminalserver.foobar.edu
#
# If you're going to protect the portmapper use the name "portmap" for the
# daemon name. Remember that you can only use the keyword "ALL" and IP
# addresses (NOT host or domain names) for the portmapper, as well as for
# rpc.mountd (the NFS mount daemon). See portmap(8) and rpc.mountd(8)
# for further information.
#
#ALL:ALL
rpcbind : 192.168.51.100 : allow
```

- Negare l'accesso a tutti gli host, tranne quelli specificati in /etc/hosts.allow, modificando il file /etc/hosts.deny:



```
Metasploitable2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
GNU nano 2.0.7      File: /etc/hosts.deny      Modified

# for further information.
#
# The PARANOID wildcard matches any host whose name does not match its
# address.
#
# You may wish to enable this to ensure any programs that don't
# validate looked up hostnames still leave understandable logs. In past
# versions of Debian this has been the default.
# ALL: PARANOID

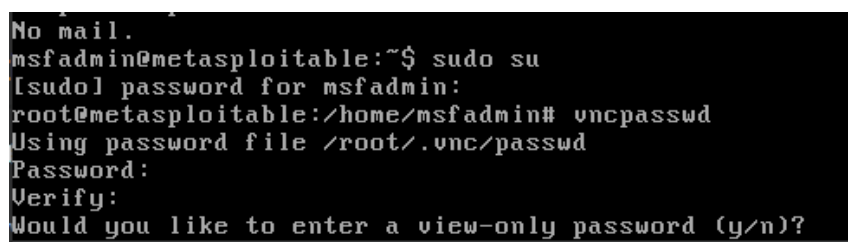
rpcbind : ALL : deny
```

Questi passaggi hanno portato alla risoluzione della vulnerabilità.

#### **Remediation per la vulnerabilità VNC Server 'password' Password (ID 61708)**

Il server VNC in esecuzione sull'host remoto è protetto da una password molto debole, per cui Nessus è riuscito ad accedere utilizzando l'autenticazione VNC e una password "password". Un aggressore remoto e non autenticato potrebbe sfruttare questo per prendere il controllo del sistema. Bisogna dunque proteggere il servizio VNC con una password più robusta.

Per cambiare la password VNC su Metasploitable si sono prima ottenuti i privilegi di amministratore con il comando **sudo su** e poi con il comando **vncpasswd** viene presentata una schermata in cui la macchina ci permette di inserire la nuova password due volte, negli input **Password** e **Verify**.



```
No mail.
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)?
```

Questi passaggi hanno portato alla risoluzione della vulnerabilità.

#### **Remediation per la vulnerabilità Bind Shell Backdoor Detection (ID 51988)**

Questa vulnerabilità ci dice che l'host potrebbe essere stato compromesso a causa di una shell in ascolto sulla porta 1524 senza richiesta di autenticazione. Quindi un attaccante potrebbe usarla per connettersi da remoto e inviare dalla porta comandi.

Ai fini della risoluzione, pur essendo una remediation di carattere temporaneo e non del tutto risolutiva, si è deciso di creare una regola firewall che blocca l'accesso alla porta in esame ovvero la 1524, che è una porta tcp. Per fare ciò, è stato usato il comando:

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

Lo switch -A specifica che verrà aggiunta una regola alla catena di INPUT che gestisce il traffico in ingresso alla porta tcp, specificato dallo switch -p, numero 1524 (switch -dport) e in particolare si specifica che il traffico in ingresso deve essere bloccato (azione di DROP con switch -j).

Con il comando **iptables -L**, si verifica la regola appena inserita. Questi passaggi hanno portato alla risoluzione (seppur temporanea) della vulnerabilità.

### **Remediation per la vulnerabilità Apache Tomcat AJP Connector Request Injection (Ghostcat) (ID 134862)**

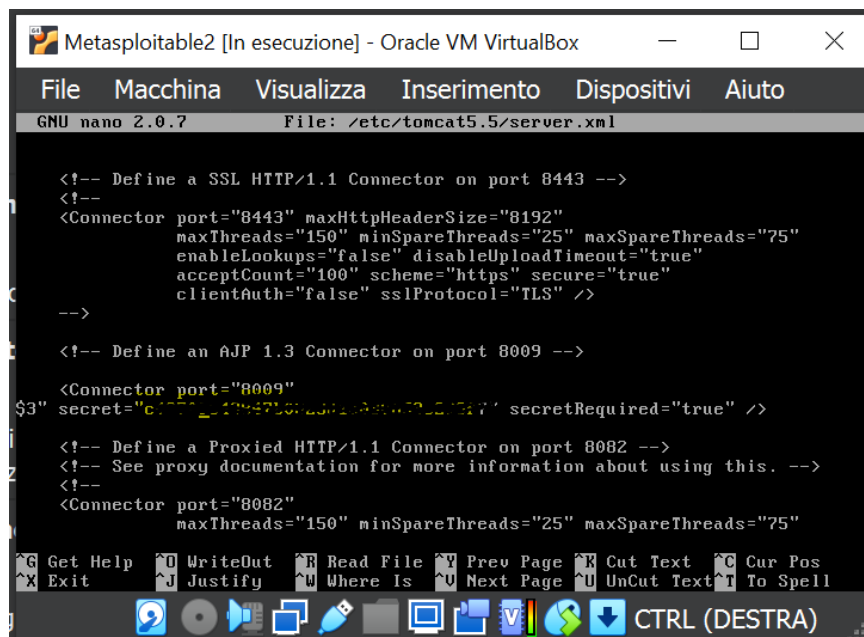
Si tratta di una vulnerabilità dovuta ad un connettore AJP vulnerabile sull'host remoto. È una vulnerabilità nella lettura/inclusione di file nel connettore AJP che un aggressore remoto non autenticato potrebbe sfruttare per leggere i file dell'applicazione web da un server vulnerabile. Nei casi in cui il server vulnerabile consenta l'upload di file, un utente malintenzionato potrebbe caricare codice JavaServer Pages (JSP) dannoso all'interno di una serie di tipi di file, ottenendo così un accesso remoto.

Si deve dunque lavorare sulla configurazione del connettore AJP, aggiungendo una autenticazione.

Per farlo, si genera una chiave segreta da inserire nel file **server.xml**, attraverso il comando:

```
cat /dev/urandom | tr -dc 'a-f0-9' | head -c 32
```

Successivamente, si modifica il file nel percorso **/etc/tomcat5.5/server.xml**, inserendo la chiave generata e inserendo **secretRequired="true"**.



```
Metasploitable2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
GNU nano 2.0.7      File: /etc/tomcat5.5/server.xml

<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<!--
<Connector port="8443" maxHttpHeaderSize="8192"
          maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
          enableLookups="false" disableUploadTimeout="true"
          acceptCount="100" scheme="https" secure="true"
          clientAuth="false" sslProtocol="TLS" />
-->

<!-- Define an AJP 1.3 Connector on port 8009 -->
$3" <Connector port="8009"
secret="c1777_477475012381080477227217" secretRequired="true" />

<!-- Define a Proxied HTTP/1.1 Connector on port 8082 -->
<!-- See proxy documentation for more information about using this. -->
<!--
<Connector port="8082"
          maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
-->

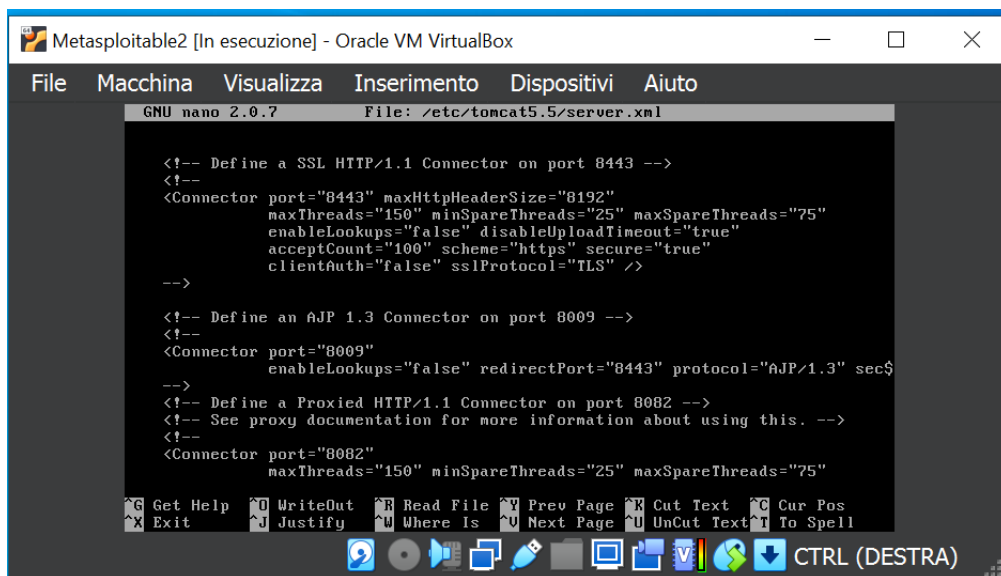
^G Get Help  ^O WriteOut  ^R Read File  ^V Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^U Next Page  ^U UnCut Text ^T To Spell

CTRL (DESTRA)
```

Questa azione di per sé dovrebbe risolvere la vulnerabilità, ma ripetendo la scansione la vulnerabilità era sempre presente. L'idea successiva è stata dunque di ragionare su una soluzione più drastica e sicuramente temporanea, ovvero disabilitare il connettore.

Di certo è una soluzione da intraprendere solo se si è sicuri che le applicazioni con cui si lavora non siano dipendenti dal connettore AJP per le comunicazioni con gli altri sistemi e va intrapresa come azione d'emergenza in caso di rischi, in attesa di una soluzione migliore e definitiva, come anche l'aggiornamento del server Tomcat a una versione più recente.

Con la disabilitazione del connettore, eseguita commentando la riga di codice del file **/etc/tomcat5.5/server.xml** correlata al connettore AJP, effettivamente la vulnerabilità non riappare nella scansione finale.



```
Metasploitable2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
GNU nano 2.0.7  File: /etc/tomcat5.5/server.xml

<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<!--
<Connector port="8443" maxHttpHeaderSize="8192"
maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
enableLookups="false" disableUploadTimeout="true"
acceptCount="100" scheme="https" secure="true"
clientAuth="false" sslProtocol="TLS" />
-->

<!-- Define an AJP 1.3 Connector on port 8009 -->
<!--
<Connector port="8009"
enableLookups="false" redirectPort="8443" protocol="AJP/1.3" sec$
-->
<!-- Define a Proxy HTTP/1.1 Connector on port 8082 -->
<!-- See proxy documentation for more information about using this. -->
<!--
<Connector port="8082"
maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
-->

^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  ^U UnCut Text ^T To Spell
CTRL (DESTRA)
```

### **Remediation per la vulnerabilità SSL Version 2 and 3 Protocol Detection (ID 20007)**

È stata presa in esame anche tale vulnerabilità, per la quale sono state eseguite diverse azioni che probabilmente hanno mitigato la sua importanza, ma che non sono state sufficienti per risolverla. Di seguito si espongono le azioni e i ragionamenti svolti.

Dal report si evince che il servizio remoto cripta il traffico utilizzando un protocollo con debolezze note. Infatti accetta connessioni crittografate con SSL 2.0 e/o SSL 3.0, versioni affette da diversi difetti crittografici, tra cui:

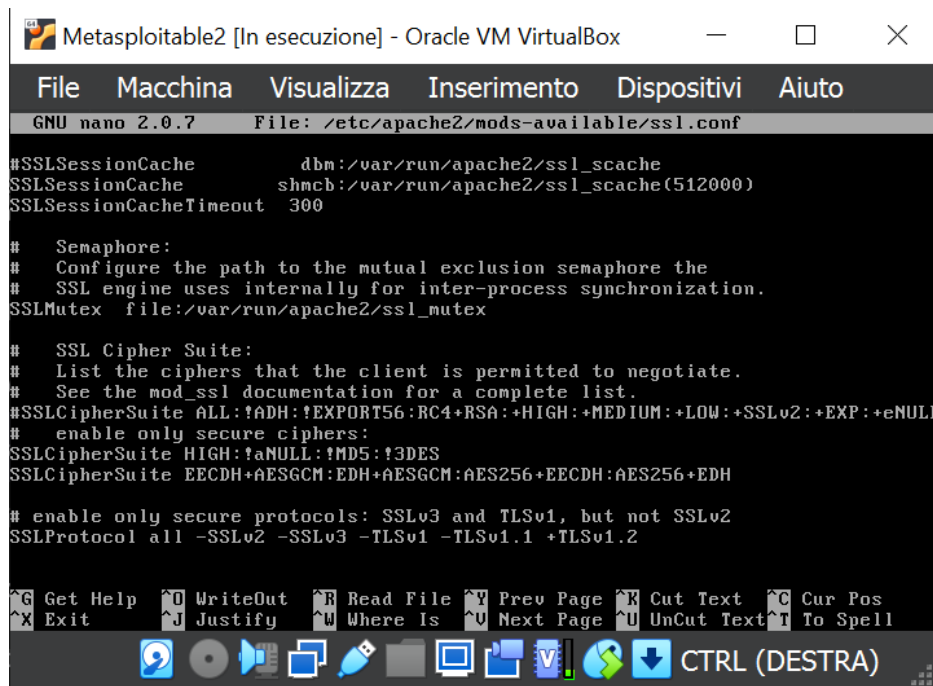
- Uno schema di imbottitura insicuro con i cifrari CBC;
- Schemi di rinegoziazione e ripresa della sessione non sicuri;

Un utente malintenzionato può sfruttare queste falle per condurre attacchi man-in-the-middle o per decriptare le comunicazioni tra il servizio interessato e i client. Sebbene SSL/TLS disponga di un metodo sicuro per scegliere la versione più alta del protocollo supportata (in modo che queste versioni vengano utilizzate solo se il client o il server non supportano nulla di meglio), molti browser web lo implementano in modo non sicuro, consentendo a un aggressore di declassare una connessione. Pertanto, è raccomandato da Nessus di disabilitare completamente questi protocolli. Il NIST ha stabilito che SSL 3.0 non è più accettabile per le comunicazioni sicure. A partire dalla data di applicazione prevista da PCI DSS v3.1, qualsiasi versione di SSL non soddisfa la definizione di "crittografia forte" data da PCI SSC.

Quindi ci suggerisce di consultare la documentazione per disabilitare SSL 2.0 e 3.0 e di usare TLS 1.2 (o versione maggiore) con un cifrario appropriato.

Per prima cosa è stato ricercato il file di configurazione che contenesse informazioni riguardo il protocollo SSL. Tale file si trova nel percorso `/etc/apache2/mods-available/ssl.conf`.

Sono state eseguite le seguenti modifiche:



```
Metasploitable2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
GNU nano 2.0.7  File: /etc/apache2/mods-available/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:!aNULL:!MD5:!3DES
SSLCipherSuite ECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH

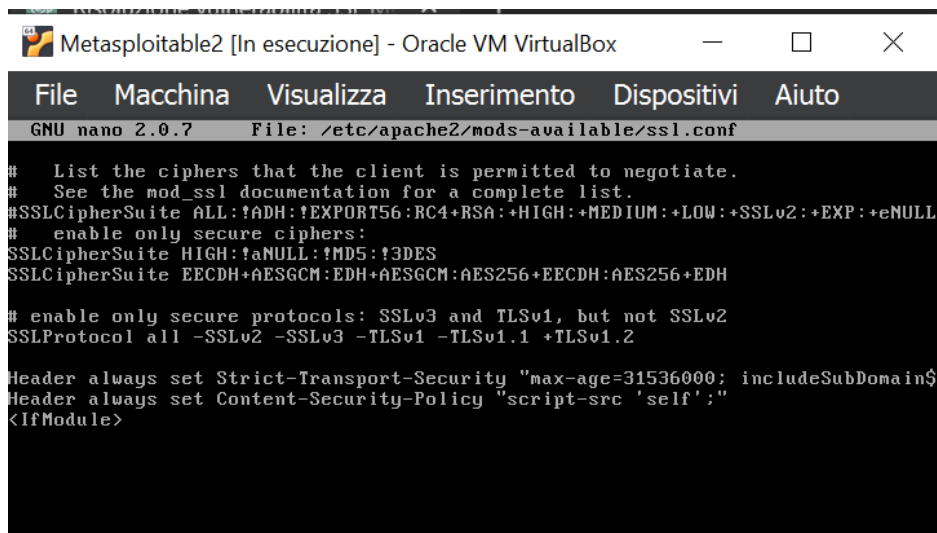
# enable only secure protocols: SSLv3 and TLSv1, but not SSLv2
SSLProtocol all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1 +TLSv1.2

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

[Icons] CTRL (DESTRA)
```

- In SSLProtocol è stato specificato di non considerare le versioni SSLv2, SSLv3, TLSv1 e TLSv1.1, a favore della versione TLSv1.2;
- In SSLCipherSuite si è cercato di disabilitare cifrari più deboli e aggiungendone di più sicuri;

Sono stati inoltre aggiunti due comandi per le direttive HSTS (HTTP Strict Transport Security) e CSP (Content Security Policy):



```
Metasploitable2 [In esecuzione] - Oracle VM VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
GNU nano 2.0.7  File: /etc/apache2/mods-available/ssl.conf

#   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:!aNULL:!MD5:!3DES
SSLCipherSuite ECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH

# enable only secure protocols: SSLv3 and TLSv1, but not SSLv2
SSLProtocol all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1 +TLSv1.2

Header always set Strict-Transport-Security "max-age=31536000; includeSubDomain$
Header always set Content-Security-Policy "script-src 'self';"
<IfModule>
```

In aggiunta è stato modificato il file **/etc/postfix/main.cf**, nell'ottica di tentare l'utilizzo di cifrari di più alta sicurezza:

```
# TLS parameters
smtpd_tls_security_level=may
smtpd_tls_protocols=TLSv1.2, TLSv1.3
smtpd_tls_ciphers=high
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
```

Nonostante queste azioni, la vulnerabilità risulta ancora presente nella scansione finale, a differenza delle altre precedenti che risultano risolte. Probabilmente sarebbe necessario aggiornare i cifrari, approfondendo con ricerche mirate per trovare quelli più robusti allo stato dell'arte.

### Confronto tra le scansioni

Come visibile dalle immagini, la seconda scansione ha riportato 6 vulnerabilità critiche, anziché 11. Questo perché sono state risolte le vulnerabilità:

- **Apache Tomcat AJP Connector Request Injection (Ghostcat)** e non ha rilevato **Apache Tomcat SEoL (<= 5.5.x)**;
- **Bind Shell Backdoor Detection**;
- **NFS Exported Share Information Disclosure**;
- **VNC Server 'password' Password**;

