EPICODE-CS0124
S5/L5 - Pratica

Flaviano Sedici

Pratica

Effettuare una scansione completa sul target Metasploitable.

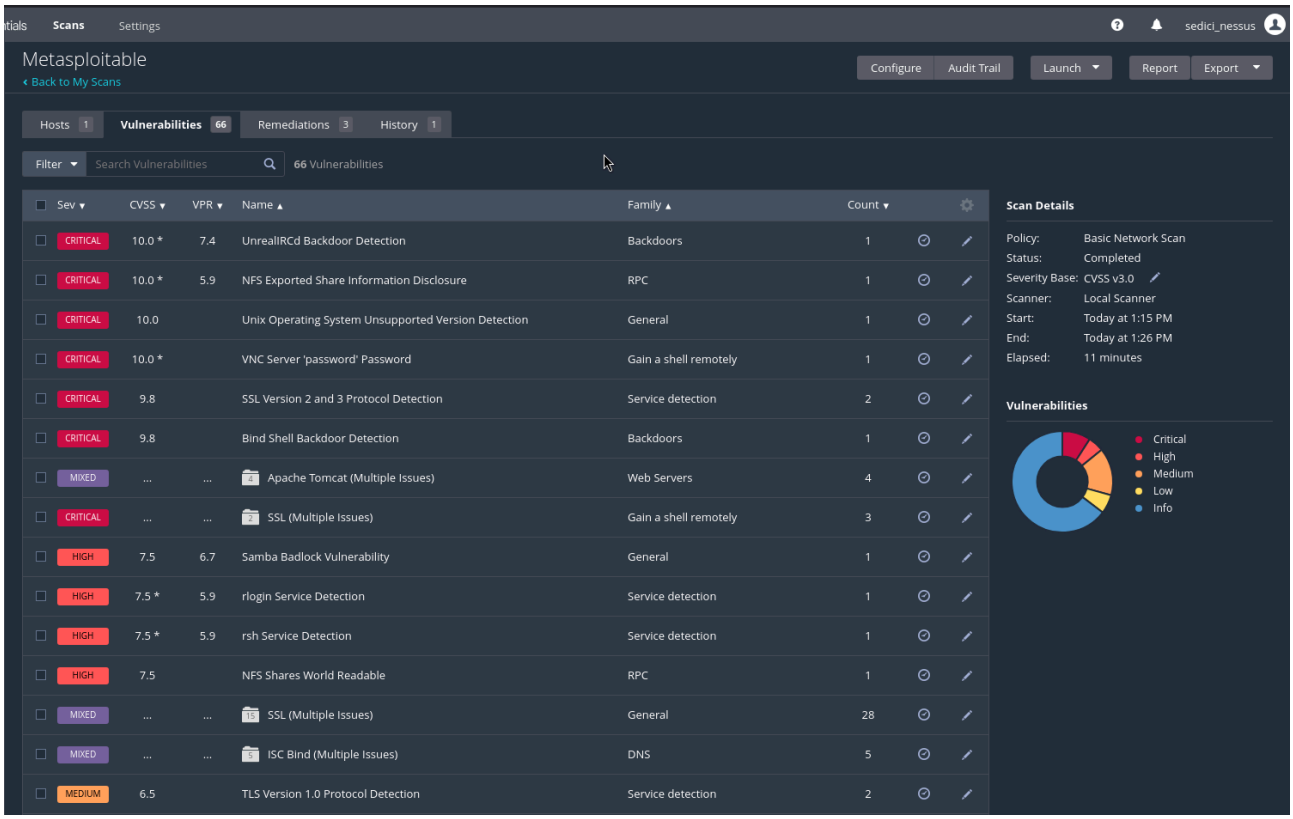
Scegliete da un minimo di 2 fino ad un massimo di 4 vulnerabilità critiche / high e provate ad implementare delle azioni di rimedio.

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.

1. Scansione Iniziale con Nessus

Per configurare la scansione delle vulnerabilità di Metasploitable, con IP 192.168.60.101, sono state utilizzate le impostazioni di default con la funzionalità "basic network scan".



Dettaglio dei risultati di scansione

2. Vulnerabilità selezionate per l’analisi

- NFS Exported Share Information Disclosure
- VNC Server 'password' Password
- Bind Shell Backdoor Detection
- SSL Version 2 and 3 Protocol Detection

2.1. NFS Exported Share Information Disclosure

2.1.1. Analisi

L'NFS è un file system che consente a computer client di utilizzare la rete per accedere a directory condivise da server remoti come fossero disponibili in locale.

In questo caso è stata rilevata una vulnerabilità derivante dalla non corretta e troppo permissiva configurazione del sistema NFS.

Come si evince dall’immagine, l’impostazione (ora commentata) dava i permessi di scrittura su tutte le directory della root, con accessi di root, a tutte le connessioni.

2.1.2. Remediation

Al fine di mantenere attivo il sistema NFS riducendo comunque la vulnerabilità, abbiamo deciso di implementare una soluzione che consenta alla sola macchina con IP 192.168.60.102 (nel nostro caso Windows 7) di accedere in scrittura alla directory di root, senza però dare permessi di root o concedere il discovery dell’alberatura delle sotto cartelle.

```
# /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)
/       192.168.60.102(rw,root_squash,subtree_check)









                                [ Wrote 13 lines ]

msfadmin@metasploitable:~$ sudo exportfs -ra
msfadmin@metasploitable:~$
```

File /etc/exports modificato

sudo nano /etc/exports per effettuare l’editing della configurazione

sudo exportfs -ra per restartare il NFS

2.1.3. Fonti

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/5/html/deployment_guide/s1-nfs-server-config-exports

2.2. VNC Server 'password' Password

2.2.1. Analisi

Virtual Network Computing (o VNC) sono applicazioni software di accesso/controllo remoto utilizzate per l'amministrazione del proprio computer a distanza, potendo essere anche usate per controllare in remoto server che naturalmente non posseggono né monitor né tastiera. Tipicamente si tratta di tools ad interfaccia testuale; quando sono ad interfaccia grafica si parla di applicazione di desktop remoto.

Nel caso in questione Nessus ha rilevato che la password per accedere al servizio è stata incautamente impostata su “password”.

2.2.2. Remediation

L’analisi condotta ha portato all’identificazione del metodo per il cambio password del servizio tramite il comando ***sudo vncpasswd*** tramite il quale abbiamo modificato la password sia per il controllo remoto sia per la modalità view-only con due password più robuste della precedente.

2.2.3. Fonti

https://linuxconfig.org/how-to-change-vnc-password-on-linux

2.3. Bind Shell Backdoor Detection

2.3.1. Analisi

L’analisi ha rilevato una backdoor con la possibilità di accedere a comandi shell in ascolto sulla porta 1524 alla quale chiunque poteva accedere.

2.3.2. Remediation

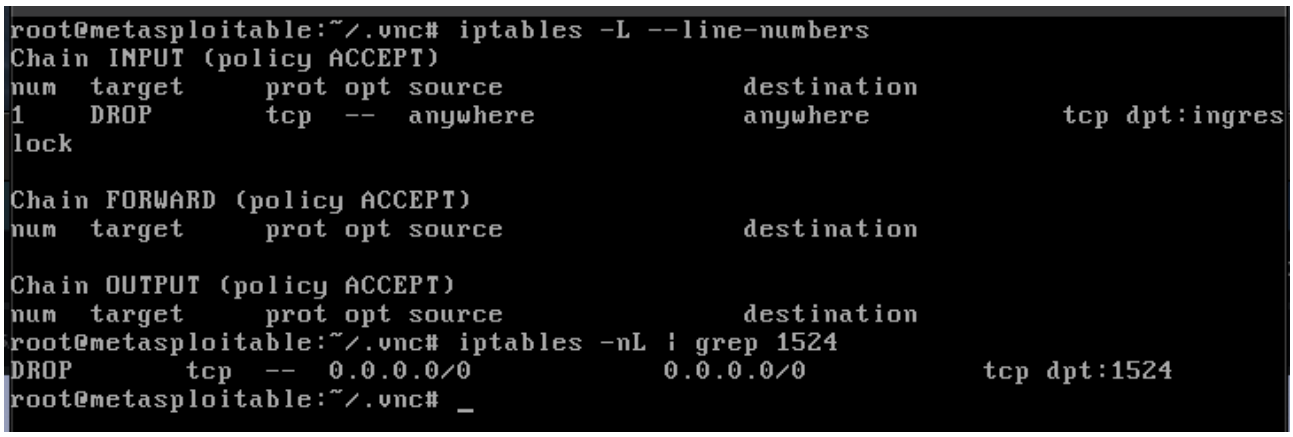
La possibile remediation sarebbe stata quella di identificare la modalità con cui la backdoor era stata aperta ed eliminare il codice colpevole della vulnerabilità. Non essendoci però alcun servizio utile per il sistema su quella porta, si è scelta la via del blocco tramite firewall.

Utilizzando iptables è stata impostata una regola sul firewall della macchina target in modo che blocchi la porta 1524 con un DROP (per evitare di restituire un risultato certo agli attaccanti), con i seguenti comandi:

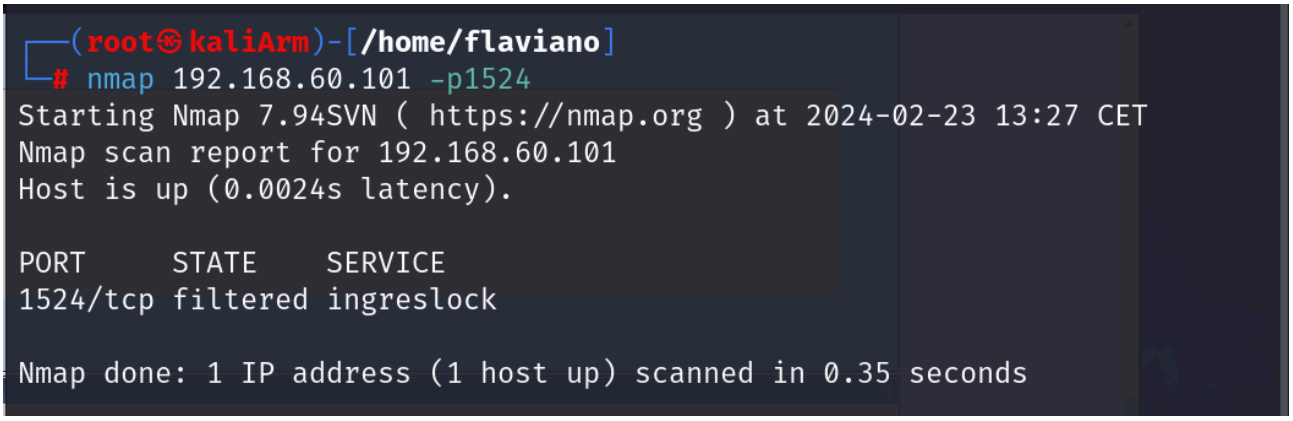
sudo iptables -L --line-numbers per fare il listato delle regole con il numero

sudo iptables -A INPUT -p tcp --dport 1524 -j DROP per impostare la regola

sudo iptables -nL \| grep 1524 per vedere la sola regola impostata e verificarne la correttezza



iptables a valle della modifica



Test effettuato sulla porta con nmap

2.3.3. Fonti

Manuale di utilizzo ***iptables -h | more***

2.4. SSL Version 2 and 3 Protocol Detection

2.4.1. Analisi

Nessus ha riportato l'utilizzo di politiche di crittazione di alcuni servizi deprecate.

In particolare per i protocolli SSL 2 e 3 il sistema target usa degli schemi di cifratura obsoleti.

Tale vulnerabilità interessa soprattutto le porte 25 e 5432.

Approfondendo l'analisi tramite nmap, abbiamo rilevato che le porte appartengono rispettivamente ai servizi postfix (25) e postgresql (5432).

Risultati analisi Nessus

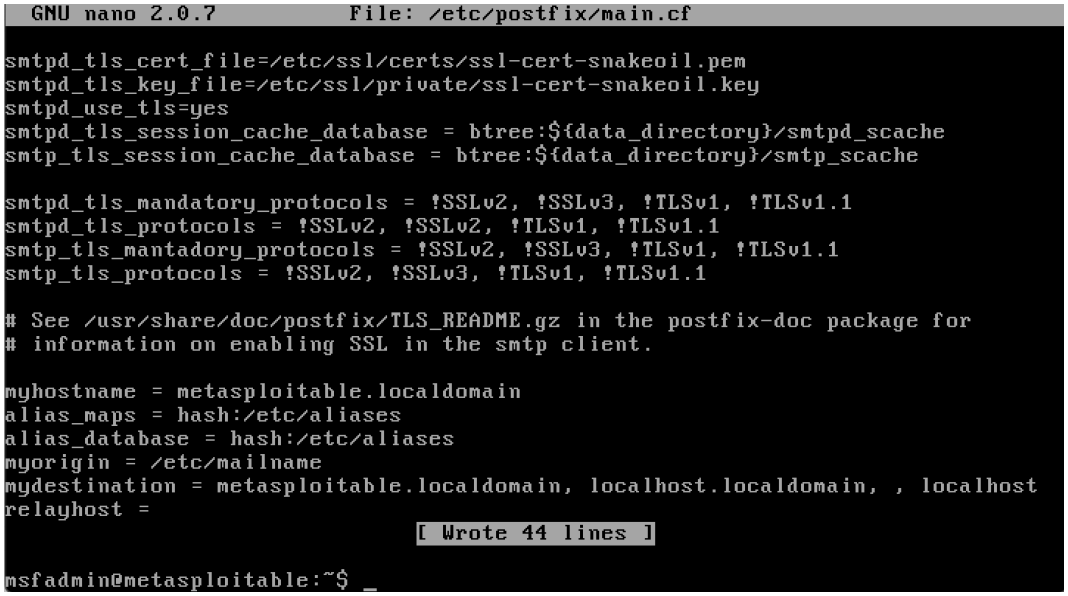
2.4.2. Remediation

La soluzione più opportuna sembra essere quella di inibire l'utilizzo dei protocolli SSL v2 e SSL v3 a favore di un più aggiornato e sicuro TLS v1.2 per entrambi i servizi.

Abbiamo quindi provato ad aggiornare prima il file di configurazione di postfix imponendo il non utilizzo dei protocolli deprecati a favore del TLS v1.2.

sudo nano /etc/postfix/main.cf

sudo /etc/init.d/postfix restart

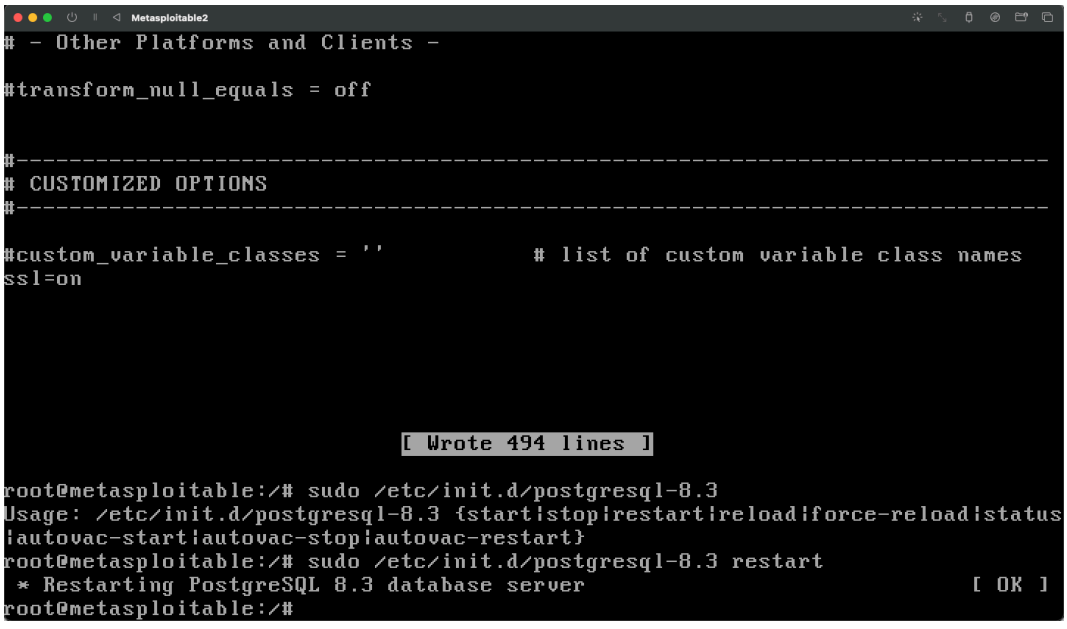


Didascalia

Abbiamo poi effettuato la stessa operazione sul file di configurazione del db postgresql.

sudo nano /etc/postgresql/8.3/main/postgresql.conf

sudo /etc/init.d/postgresql-8.3 restart



Modifica del file postgresql.conf

Sfortunatamente tale modifica non è stata sufficiente in entrambi i casi in quando il problema risiede più nel profondo, ovvero nelle chiavi di cifratura .pem.

```
┌──(root㉿kaliArm)-[/tmp/r00t/home]
└─# nmap -sV 192.168.60.101 -p25
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-23 15:15 CET
Nmap scan report for 192.168.60.101
Host is up (0.0016s latency).

PORT   STATE SERVICE VERSION
25/tcp open  smtp    Postfix smtpd
Service Info: Host:  metasploitable.localdomain

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.72 seconds

┌──(root㉿kaliArm)-[/tmp/r00t/home]
└─# nmap -sV 192.168.60.101 -p5432
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-23 15:15 CET
Nmap scan report for 192.168.60.101
Host is up (0.0016s latency).

PORT     STATE SERVICE    VERSION
5432/tcp open  postgresql PostgreSQL DB 8.3.0 - 8.3.7

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.52 seconds
```

Analisi condotta con nmap

2.4.3. Fonti

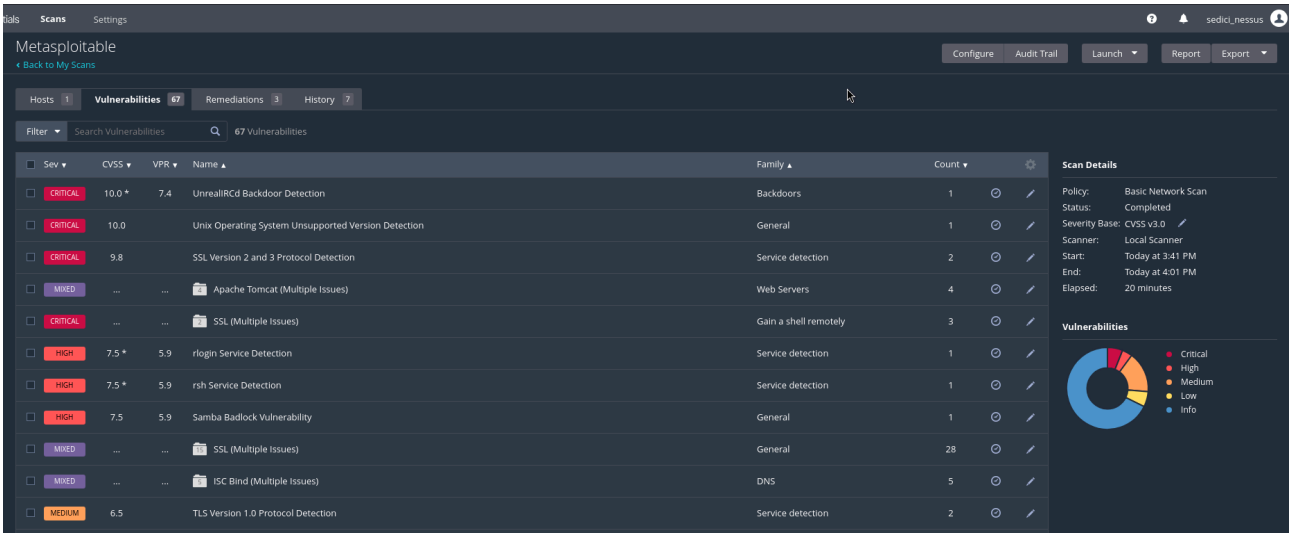
https://access.redhat.com/articles/1468593

https://access.redhat.com/documentation/fr-fr/red_hat_enterprise_linux/9/html/configuring_and_using_database_servers/proc_configuring-tls-encryption-on-a-postgresql-server_using-postgresql

3. Conclusioni

In conclusione siamo riusciti a sanare 3 vulnerabilità critiche su 4, identificandone però le cause in modo da poter, con maggior tempo a disposizione, intervenire efficacemente con l'obiettivo di sanare anche quest'ultima.

Di seguito i risultati della scansione finale dalla quale si evince che le 3 vulnerabilità sono state mitigate con successo.



Risultati scansione finale