

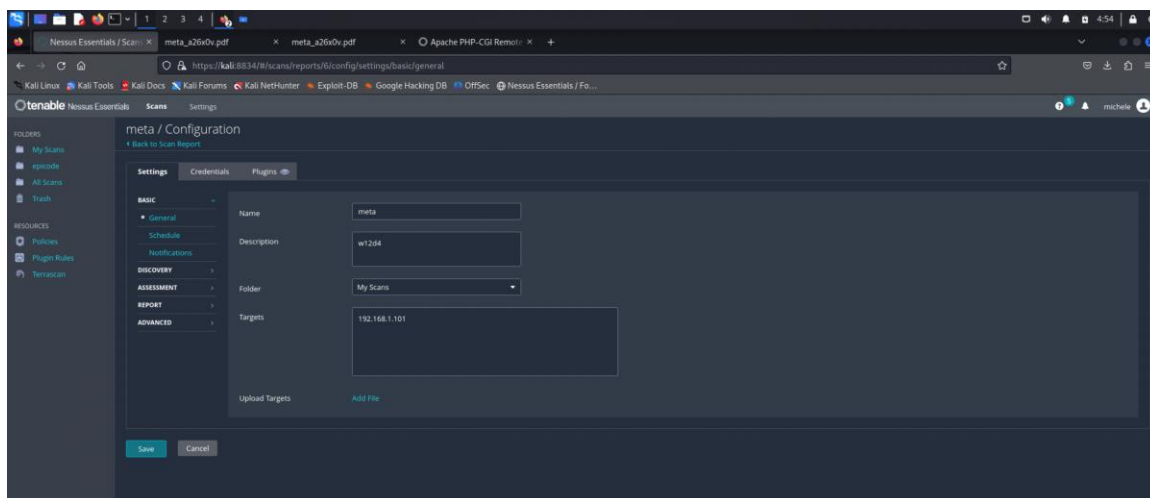
## ESERCIZIO W12D4 Michele Ungolo

**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. Ai fini della soluzione, abbiamo scelto le vulnerabilità in giallo nella figura in slide 3.

### Soluzione:

1. Per prima cosa andiamo ad aprire le macchine Metasploitable e Kali, su VirtualBox, accediamo alla schermata di Kali e procediamo all'apertura di Nessus con il comando `Nessus.service` da terminale.

Effettuato il login di Nessus attraverso la rete internet, avviamo il tool e modifichiamo le impostazioni per effettuare la scansione delle vulnerabilità:



## meta / Configuration

[← Back to Scan Report](#)

### Settings

Credentials

Plugins 

BASIC >

DISCOVERY >

ASSESSMENT >

REPORT >

ADVANCED ✓

Scan Type

Default

#### Performance options:

30 simultaneous hosts (max)

4 simultaneous checks per host (max)

5 second network read timeout

Save

Cancel

## meta / Configuration

[← Back to Scan Report](#)

### Settings

Credentials

Plugins 

BASIC >

DISCOVERY ✓

ASSESSMENT >

REPORT >

ADVANCED >

Scan Type

Port scan (common ports)

#### General Settings:

Always test the local Nessus host

Use fast network discovery

#### Port Scanner Settings:

Scan common ports

Use netstat if credentials are provided

Use SYN scanner if necessary

#### Ping hosts using:

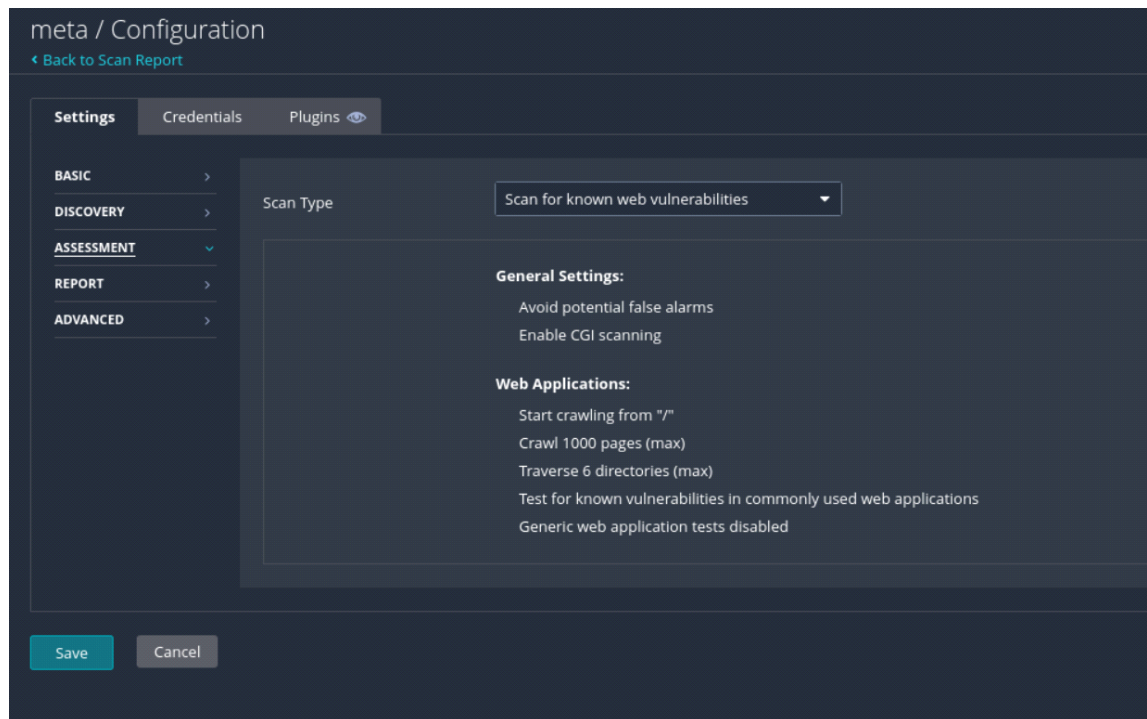
TCP

ARP

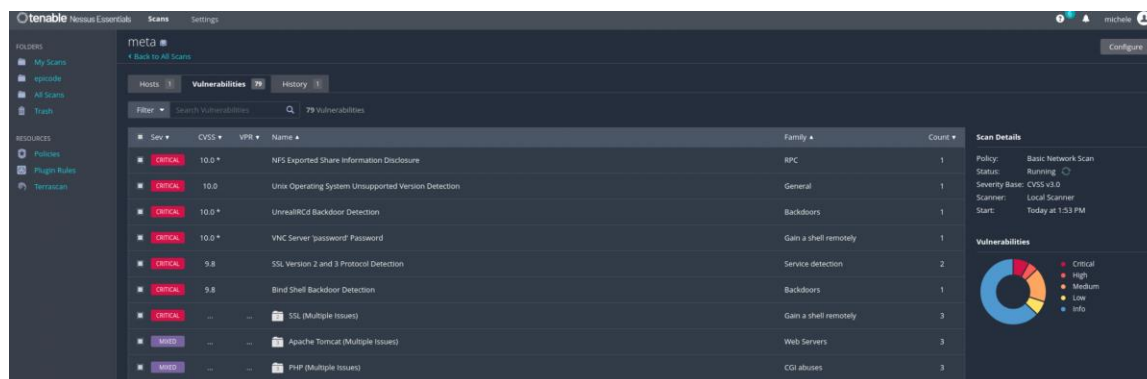
ICMP (2 retries)

Save

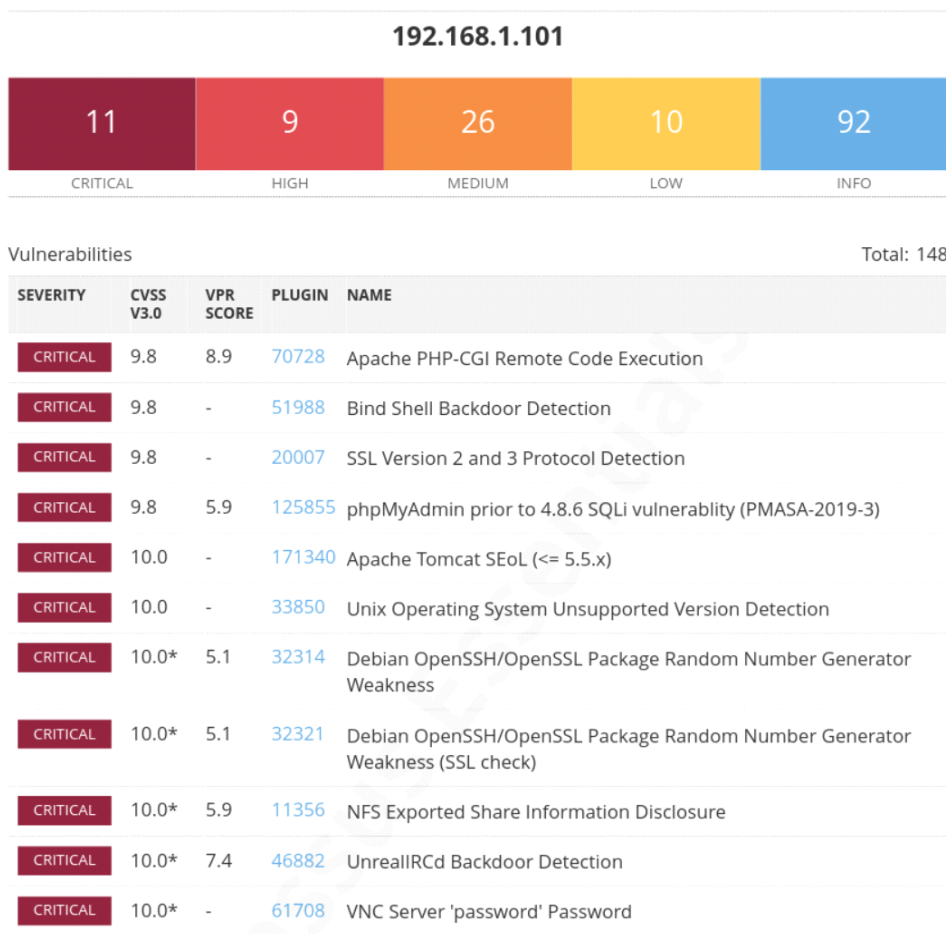
Cancel



Terminata la configurazione, lanciamo la scansione su Nessus attraverso il comando "play" e attendiamo il completamento. Alla fine della scansione avremo una schermata simile alla seguente, la quale indicherà le vulnerabilità riscontrate e il loro livello di criticità:



Eseguendo il comando "Report" avremo nel dettaglio le scansioni eseguite:



2. Da qui possiamo procedere con il risolvere le vulnerabilità riscontrate:

**Prima vulnerabilità: NFS Exported Share Information Disclosure**

# NFS Exported Share Information Disclosure

Language: English

CRITICAL

Nessus Plugin ID 11356

Information

Dependencies

Dependents

Changelog

## Synopsis

It is possible to access NFS shares on the remote host.

## Description

At least one of the NFS shares exported by the remote server could be mounted by the scanning host. An attacker may be able to leverage this to read (and possibly write) files on remote host.

## Solution

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

## Plugin Details

Severity: Critical

ID: 11356

File Name: nfs\_mount.nasl

Version: 1.21

Type: remote

Family: [RPC](#)

Published: 3/12/2003

Updated: 8/30/2023

Supported Sensors: Nessus

## Risk Information

[VPR](#)

Risk Factor: Medium

Score: 5.9

[CVSS v2](#)

Risk Factor: Critical

Base Score: 10

Vector: CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C

CVSS Score Source: [CVE-1999-0554](#)

Secondo lo scanner, il servizio NFS espone gran parte delle directory, rendendole montabili su sistemi client esterni. Non dovrebbero essere accessibili da tutti, ma solo dall'utenza autorizzata. Per correggere questa vulnerabilità andiamo a cambiare la configurazione direttamente da Metasploitable , creando una cartella privata con mkdir visibile solo da localhost.

```
msfadmin@metasploitable:/home/user$ cd ..
msfadmin@metasploitable:/home$ cd ..
msfadmin@metasploitable:/ $ ls
bin      dev      initrd   lost+found  nohup.out  root    sys    var
boot     etc      initrd.img  media      opt        sbin    tmp    vmlinuz
cdrom    home    lib      mnt        proc       srv     usr
msfadmin@metasploitable:/ $ sudo mkdir privateshare
[sudo] password for msfadmin:
msfadmin@metasploitable:/ $ ls
bin      dev      initrd   lost+found  nohup.out  proc    srv     usr
boot     etc      initrd.img  media      opt        root    sys     var
cdrom    home    lib      mnt        privateshare  sbin    tmp     vmlinuz
msfadmin@metasploitable:/ $ sudo chmod 777 privateshare
msfadmin@metasploitable:/ $ msfadmin
```

Accediamo poi al file di testo localizzato in /etc/exports. Usiamo sudo

nano per aprire il foglio e specifichiamo la cartella privateshare come usufruibile solo da Metaploitable.

```
# /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)
#
/private share localhost(rw,sync,no_root_squash,no_subtree_check)
```

Configurazione Exports

Salvate le nuove impostazioni effettuiamo un reboot di Metaploitable.

Su Kali creiamo una cartella mount per la condivisione, dentro la directory temp.

```
File Actions Edit View Help
(kali@kali)-[~]
$ sudo mkdir /tmp/mount
[sudo] password for kali:
(kali@kali)-[~]
$ cd /tmp/
(kali@kali)-[/tmp]
$ ls
hsperfdata_root
mount
ssh-b4u3yG785fp8
systemd-private-67b84ce6ea324dae8c0e6ed94cad108-color.service-nCFBGy
systemd-private-67b84ce6ea324dae8c0e6ed94cad108-haveged.service-4XF6zg
systemd-private-67b84ce6ea324dae8c0e6ed94cad108-ModemManager.service-MIIiDJ
systemd-private-67b84ce6ea324dae8c0e6ed94cad108-polkit.service-QRU2IJ
systemd-private-67b84ce6ea324dae8c0e6ed94cad108-systemd-logind.service-LWNhcJ
systemd-private-67b84ce6ea324dae8c0e6ed94cad108-upower.service-76WEDc
Temp-8861503f-6e8e-4da0-a413-627ecabf78ac
(kali@kali)-[/tmp]
$ sudo mount -t
mount: option requires an argument --'t'
Try 'mount --help' for more information.
(kali@kali)-[/tmp]
$ sudo mount -t nfs 192.168.1.101:/privateshare /tmp/mount -nolock
Created symlink /run/systemd/system/remote-fs.target.wants/rpc-statd.service → /usr/lib/systemd/system/rpc-statd.service.
(kali@kali)-[/tmp]
$
```


Usiamo il comando: **sudo mount -t nfs 192.168.1.100:/privateshare /tmp/mount -nolock** che ci consentirà di tentare un montaggio di rete:

- il comando **mount** viene utilizzato per montare un file system
- **-t nfs** indica il tipo di file system da montare
- **192.168.1.100:/privateshare** specifica l'indirizzo IP del server NFS e la directory condivisa da montare.
- **/tmp/mount** conferma il punto di mount locale in cui verrà montata la cartella condivisa.
- **-nolock** indica che il mount deve essere effettuato senza il lock manager NFS.

La figura riportata sopra conferma che non è stato possibile completare il montaggio di rete, questo perché il server ha vietato l'accesso alla macchina attaccante.

## Seconda vulnerabilità: VNC Server 'password' Password



 | Plugins Settings ▾

DETECTIONS

Plugins ▾

Overview

Plugins Pipeline

Release Notes

Newest

Updated

Search

Nessus Families

WAS Families

NNM Families

LCE Families

Tenable OT Security Families

About Plugin Families

Audits >

Policies >

Indicators >

ANALYTICS

CVEs >

Attack Path Techniques >

Plugins / Nessus / 61708

VNC Server 'password' Password

Language: English ▾

CRITICAL

Nessus Plugin ID 61708

Information

Dependencies

Dependents

Changelog

Synopsis

A VNC server running on the remote host is secured with a weak password.

Description

The VNC server running on the remote host is secured with a weak password. Nessus was able to login using VNC authentication and a password of 'password'. A remote, unauthenticated attacker could exploit this to take control of the system.

Solution

Secure the VNC service with a strong password.

Plugin Details

Severity: Critical

ID: 61708

File Name: vnc\_password\_password.nasl

Version: Revision: 1.2

Type: remote

Family: [Gain a shell remotely](#)

Published: 8/29/2012

Updated: 9/24/2015

Supported Sensors: Nessus

Risk Information

CVSS v2

Risk Factor: Critical

Questa vulnerabilità indica che ci troviamo con una password non adeguata per il server VNC, ossia, il servizio che consente l'accesso e il controllo remoto di un computer. Per risolverla ci basterà configurare una password più efficace. Utilizziamo il comando `vncpasswd` entrando in root.



```

root@metasploitable:/home/msfadmin# vncpasswd
Using password file /root/.vnc/passwd
Password:
Verify:
Passwords do not match. Please try again.

Password:
Verify:
Passwords do not match. Please try again.

Password:
Verify:
Would you like to enter a view-only password (y/n)? y
Password:
Verify:
Passwords do not match. Please try again.

Password:
Verify:
root@metasploitable:/home/msfadmin# exit
exit
msfadmin@metasploitable:/ $ _

```

## Terza vulnerabilità: Bind Shell Backdoor Detection

The screenshot shows the Nessus interface for the 'Bind Shell Backdoor Detection' vulnerability. The interface is divided into several sections:

- Description:** 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.
- Solution:** Verify if the remote host has been compromised, and reinstall the system if necessary.
- Output:**

```

Nessus was able to execute the command "id" using the following request:

This produced the following truncated output (limited to 10 lines):
.....
root@metasploitable:/# whoami
root
root@metasploitable:/# id
root@metasploitable:/#
.....

```
- Risk Information:**
  - Risk Factor: Critical
  - CVSS v3.0 Base Score: 9.8
  - CVSS v3.0 Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A/H
  - CVSS v2.0 Base Score: 10.0
  - CVSS v2.0 Vector: CVSS2#AV:N/AC:L/Au:N/C:C/I:N/A:C

Nessus ci segnala che sulla porta 1524 c'è una backdoor, più precisamente sembra sia una remote shell. Proviamo a connetterci sulla porta 1524 con nc, per averne la certezza.

Il comando fuser ci permetterà di rintracciare il file.

```
msfadmin@metasploitable:/$ sudo fuser 1524/tcp
[sudo] password for msfadmin:
1524/tcp:          4505
msfadmin@metasploitable:/$
```

Attraverso questo comando, veniamo a conoscenza che il PID (process ID) assegnato alla porta corrisponde a 4505. A questo punto eseguiamo il comando **sudo readlink -f <percorso .exe>**:

-f: segue il collegamento simbolico. Tutti i componenti, tranne l'ultimo, devono esistere.

```
msfadmin@metasploitable:/$ sudo fuser 1524/tcp
[sudo] password for msfadmin:
1524/tcp:          4505
msfadmin@metasploitable:/$ sudo readlink -f /proc/4505/exe
/usr/sbin/xinetd
msfadmin@metasploitable:/$
```

Abbiamo dunque il percorso che ci conduce al programma stesso. Non ci resta che eliminarlo.

```
msfadmin@metasploitable:/usr/sbin$ file xinetd
xinetd: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for GNU/Linux 2.6.8, dynamically linked (uses shared libs), stripped
```

```
msfadmin@metasploitable:/usr/sbin$ sudo rm xinetd
```

Riavviamo Metasploitable e controlliamo se la porta tcp 1524 è ancora in funzione. Dalla scansione nmap stealth non visualizziamo più la backdoor.

```
(kali@kali)-[~]
$ sudo nmap -sS 192.168.1.101
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-06 06:37 EST
Nmap scan report for 192.168.1.101 (192.168.1.101)
Host is up (0.00024s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:06:BC:AD (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 6.65 seconds

(kali@kali)-[~]
$
```

Completato il lavoro di ricerca e sistemazione delle vulnerabilità, andiamo a fare una nuova scansione su Nessus, così da verificare se il lavoro effettuato in precedenza, abbia portato ai risultati attesi.

La nuova scansione, come riportato dalla figura in basso, ci permette di comprendere che le vulnerabilità critiche sono passate da 11 a 9 e quelle medie da 26 a 25, tuttavia, possiamo asserire che le minacce non sono state del tutto annullate, ma siamo sulla strada giusta per risolvere i problemi che presenta ancora la macchina.

192.168.1.101



Vulnerabilities

Total: 141

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	9.8	8.9	70728	Apache PHP-CGI Remote Code Execution
CRITICAL	9.8	-	51988	Bind Shell Backdoor Detection
CRITICAL	9.8	-	20007	SSL Version 2 and 3 Protocol Detection
CRITICAL	9.8	5.9	125855	phpMyAdmin prior to 4.8.6 SQLi vulnerability (PMASA-2019-3)
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*	7.4	46882	UnrealIRCd Backdoor Detection
HIGH	8.8	7.4	19704	TWiki 'rev' Parameter Arbitrary Command Execution
HIGH	8.6	5.2	136769	ISC BIND Service Downgrade / Reflected DoS
HIGH	7.5	-	42256	NFS Shares World Readable