

## **Report vulnerability scanner su target Metasploitable 2 (IP: 192.168.64.10)**

Per sviluppare questo progetto ho utilizzato un software chiamato Nessus.

Nessus è un vulnerability scanner, attraverso un processo in quattro fasi evidenzia le vulnerabilities sul nostro target.

La prima fase è di port scanning, ovvero nessus va ad analizzare il/i target in modo tale da capire se sono attivi e quali porte sono accessibili o meno.

La seconda fase sussegue immediatamente il port scanning andando ad analizzare i processi attivi su ogni porta che viene trovata aperta e prende il nome di service-detection.

La terza parte è la ricerca di vulnerabilità note nel database di Nessus in base alla versione della tecnologia del target.

Durante la quarta fase Nessus testa le vulnerabilità evidenziate in base alle impostazioni inserite in fase di avvio della ricerca, generando un report finale.

Una volta terminata la fase di scansione ed enumerazione delle vulnerabilità occorre procedere con l'analisi del rischio ed eventualmente creare una roadmap in modo tale da mitigare il più possibile le vulnerabilità. Quello che segue è un esempio di roadmap da adottare in questi casi.

- Prima scansione ed enumerazione
- Analisi del rischio
- Implementazione remediation delle criticità Critic
- Implementazione remediation delle criticità High
- Implementazione remediation delle criticità Medium
- Implementazione remediation delle criticità Low
- Scansione per la verifica del raggiungimento degli obiettivi di sicurezza

Nel caso preso in esame ho valutato la situazione iniziale con svariate vulnerabilità.

Sono intervenuto su 3 vulnerabilità critiche e una high.

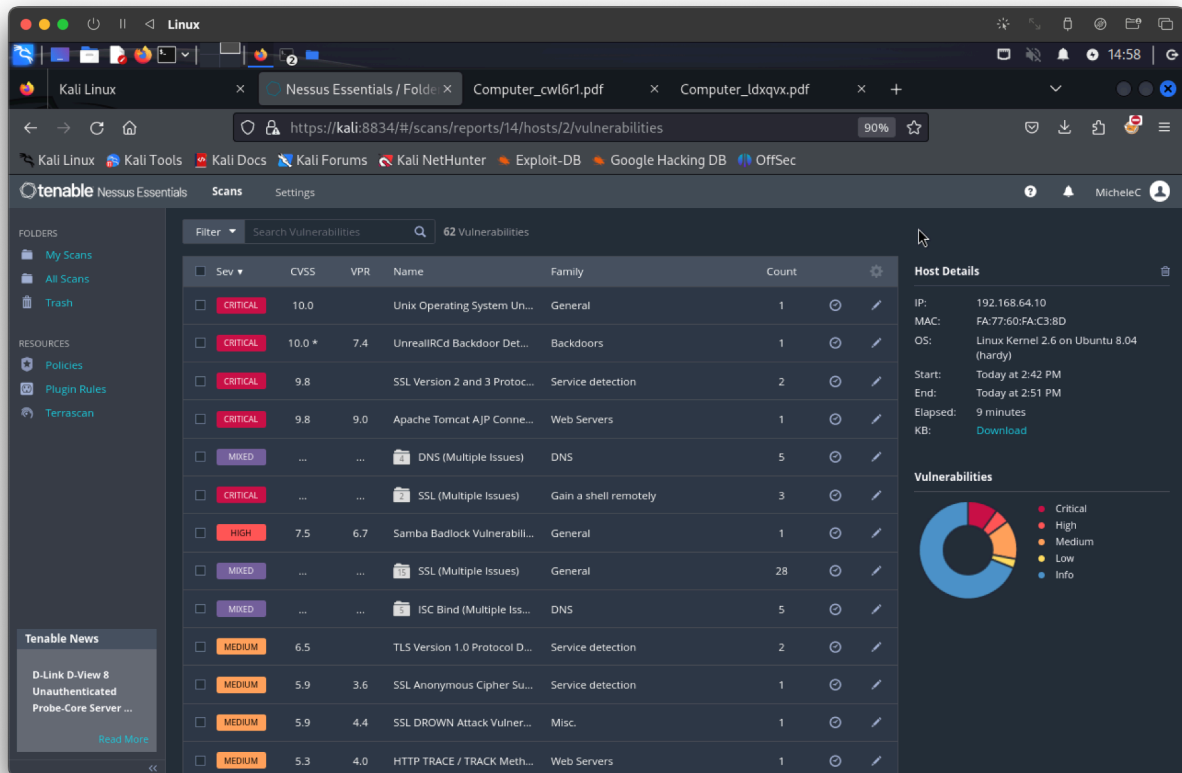
Ovviamente è la prima fase di un completo vulnerability assessment e dovranno essere sistemate anche le altre vulnerabilità.

Confrontando le successive immagini è possibile notare che sono state mitigate alcune vulnerabilità.

The screenshot shows the Tenable Nessus Essentials interface. The main panel displays a list of 72 vulnerabilities for a specific host. The table includes columns for severity, CVSS score, VPR, name, family, and count. The vulnerabilities are sorted by severity, with critical issues at the top.

Sev	CVSS	VPR	Name	Family	Count
CRITICAL	10.0 *	5.9	NFS Exported Share Infor...	RPC	1
CRITICAL	10.0		Unix Operating System Un...	General	1
CRITICAL	10.0 *	7.4	UnrealIRCd Backdoor Det...	Backdoors	1
CRITICAL	10.0 *		VNC Server 'password' Pas...	Gain a shell remotely	1
CRITICAL	9.8		SSL Version 2 and 3 P... Plugin ID: 51988 e detection		2
CRITICAL	9.8		Bind Shell Backdoor Detec...	Backdoors	1
MIXED	...	...	DNS (Multiple Issues)	DNS	5
MIXED	...	...	Apache Tomcat (Multi...	Web Servers	4
CRITICAL	...	...	SSL (Multiple Issues)	Gain a shell remotely	3
HIGH	7.5		NFS Shares World Readable	RPC	1
HIGH	7.5 *	6.7	rlogin Service Detection	Service detection	1
HIGH	7.5 *	6.7	rsh Service Detection	Service detection	1
HIGH	7.5		Samba Badlock Vulnerabili...	General	1

The right sidebar shows host details for IP 192.168.64.10, including MAC, OS (Linux Kernel 2.6 on Ubuntu 8.04 (hardy)), and scan statistics. A pie chart titled 'Vulnerabilities' shows the distribution of severity levels: Critical (red), High (orange), Medium (yellow), Low (green), and Info (blue).



La prima vulnerabilità che ho considerato è questa:

**CRITICAL** NFS Exported Share Information Disclosure

### 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.

### Output

```
The following NFS shares could be mounted :  
  
+ /  
+ Contents of / :  
- .  
- ..  
- bin  
- boot  
- ----  
more...
```

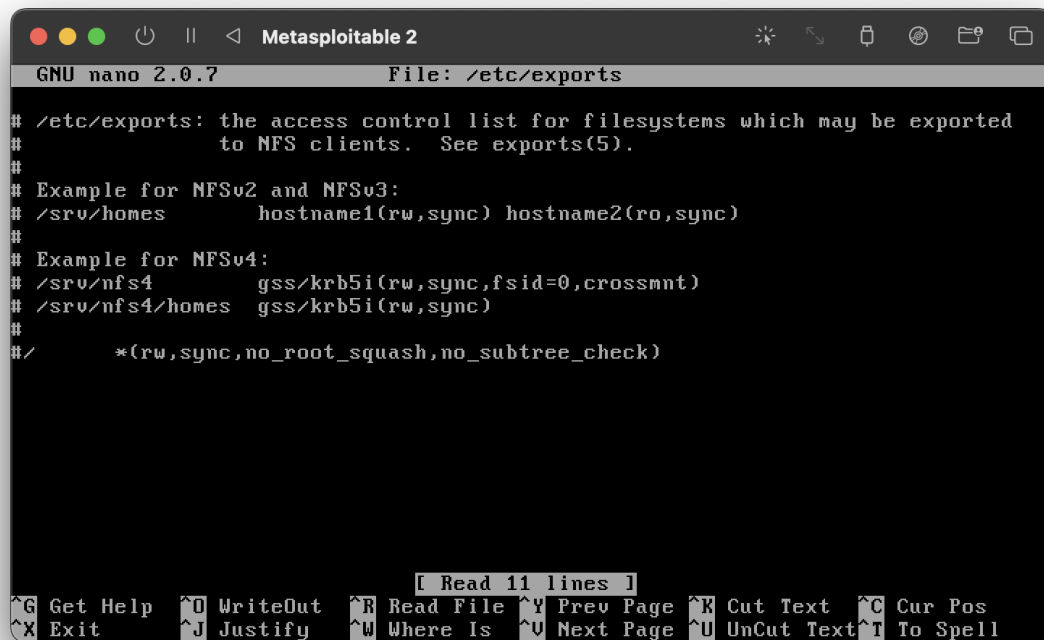
To see debug logs, please visit individual host

Port ▲	Hosts
2049 / udp / rpc-nfs	192.168.64.10

NFS è uno dei protocolli più utilizzati su base UNIX per il file sharing, questo favorisce sicuramente l'accessibilità ma spesso va ad aumentare la vulnerabilità del sistema.

La soluzione è quella di limitare l'accesso solo a determinati host oppure limitarne i permessi.

Andando a disattivare l'ultima linea del file all'interno della directory /etc/exports risolviamo questo tipo di vulnerabilità.

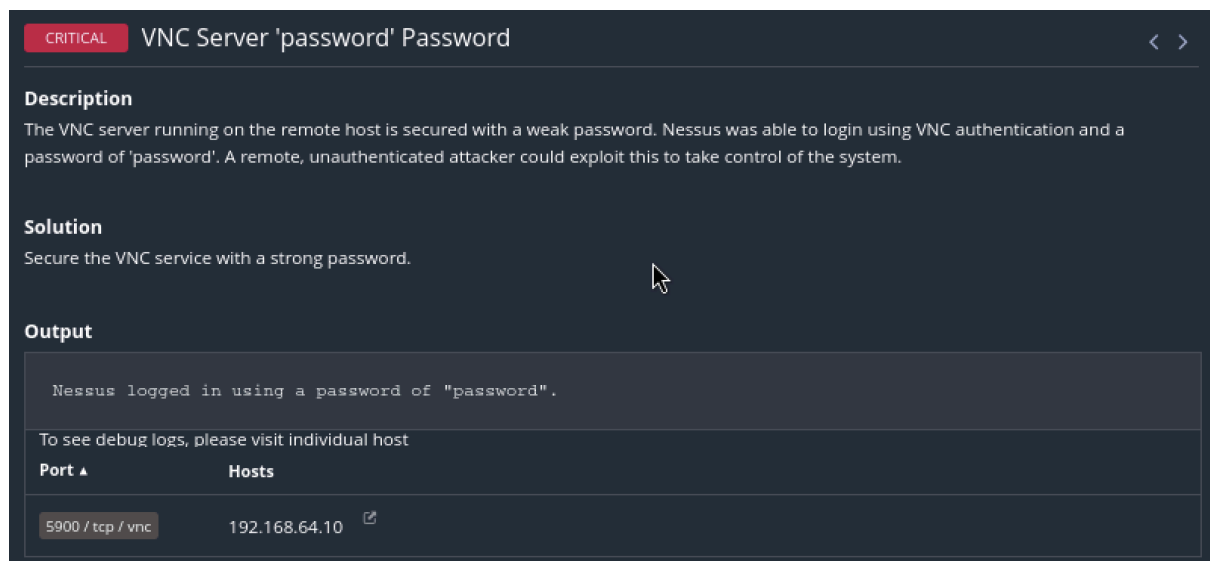


The screenshot shows a terminal window titled 'Metasploitable 2' running the GNU nano 2.0.7 editor. The file being edited is /etc/exports. The content of the file is as follows:

```
# /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)
#
#/*                  *(rw,sync,no_root_squash,no_subtree_check)
```

The bottom of the window shows a status bar with various keyboard shortcuts like ^G Get Help, ^O WriteOut, ^R Read File, etc.

La seconda criticità è:



The screenshot shows a Nessus vulnerability report for a 'CRITICAL' issue titled 'VNC Server 'password' Password'. The report includes a description, a solution, and an output section.

**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.

**Output**  
Nessus logged in using a password of "password".  
To see debug logs, please visit individual host

Port ▲	Hosts
5900 / tcp / vnc	192.168.64.10

Questa vulnerabilità è dovuta al fatto che il server VNC che solitamente è utilizzato per l'accesso remoto al client, non è protetto da password. Per ovviare a questo problema è possibile impostare una password a VNC in modo tale da renderlo crittografato.

Un'altra opzione è quella di non utilizzare il server VNC (porta 5900) e utilizzare direttamente una porta crittografata come la SSH (porta 22).

```
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
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)? n
root@metasploitable:/home/msfadmin#
root@metasploitable:/home/msfadmin# vncpasswd
Using password file /root/.vnc/passwd
Password:
Warning: password truncated to the length of 8.
Verify:
Passwords do not match. Please try again.

Password:
Verify:
Would you like to enter a view-only password (y/n)? n
root@metasploitable:/home/msfadmin#
```

La terza criticità è questa:

**CRITICAL** Bind Shell Backdoor Detection

**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) :
----- snip -----
root@metasploitable:/# uid=0(root) gid=0(root) groups=0(root)
root@metasploitable:/#
----- snip -----
```

To see debug logs, please visit individual host

Port ▲	Hosts
1524 / tcp / wild_shell	192.168.64.10

Vi è una backdoor aperta sulla porta 1524 e come da test è possibile installare un shell sulla porta, prima di tutto ho individuato quale servizio è attivo sulla porta, poi per ovviare a questo problema è stato chiuso il servizio della porta, successivamente verificando i processi attivi si può notare che non c'è più nessun tipo di servizio attivo sulla porta.

```
msfadmin@metasploitable:~$ lsof -i:1524
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$ sudo su
root@metasploitable:/home/msfadmin# lsof -i: 1524
lsof: unacceptable port specification in: -i :
lsof 4.78
latest revision: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/
latest FAQ: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/FAQ
latest man page: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/lsof_man
usage: [-?abhlnNoOPRstUvUX] [+!-c c] [+!-d s] [+D D] [+!-f]
[-F [f]] [-g [s]] [-i [i]] [+!-L [l]] [+m [m]] [+!-M] [-o [o]]
[-p s] [+!-r [t]] [-S [t]] [-T [t]] [-u s] [+!-w] [-x [f]] [--] [names]
Use the '-h' option to get more help information.
root@metasploitable:/home/msfadmin# lsof -i :1524
COMMAND PID USER  FD   TYPE DEVICE SIZE NODE NAME
xinetd  4643 root   11u  IPv4 13040      TCP *:ingreslock (LISTEN)
root@metasploitable:/home/msfadmin# _
```

```
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$ sudo su
root@metasploitable:/home/msfadmin# lsof -i: 1524
lsof: unacceptable port specification in: -i :
lsof 4.78
latest revision: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/
latest FAQ: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/FAQ
latest man page: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/lsof_man
usage: [-?abhlnNoOPRstUvUX] [+!-c c] [+!-d s] [+D D] [+!-f]
[-F [f]] [-g [s]] [-i [i]] [+!-L [l]] [+m [m]] [+!-M] [-o [o]]
[-p s] [+!-r [t]] [-S [t]] [-T [t]] [-u s] [+!-w] [-x [f]] [--] [names]
Use the '-h' option to get more help information.
root@metasploitable:/home/msfadmin# lsof -i :1524
COMMAND PID USER  FD   TYPE DEVICE SIZE NODE NAME
xinetd  4643 root   11u  IPv4 13040      TCP *:ingreslock (LISTEN)
root@metasploitable:/home/msfadmin# kill 4643
root@metasploitable:/home/msfadmin#
```

Metasploitable 2											
root	4537	0.0	0.0	0	0	?	S	08:16	0:00	[lockd]	
root	4538	0.0	0.0	0	0	?	S<	08:16	0:00	[nfsd4]	
root	4539	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4540	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4541	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4542	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4543	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4544	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4545	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4546	0.0	0.0	0	0	?	S	08:16	0:00	[nfsd]	
root	4550	0.0	0.0	2424	328	?	Ss	08:16	0:00	/usr/sbin/rpc.m	
daemon	4612	0.0	0.0	2316	216	?	SN	08:16	0:00	distccd --daemo	
root	4617	0.0	0.1	5412	1732	?	Ss	08:16	0:00	/usr/lib/postfi	
postfix	4620	0.0	0.1	5420	1644	?	S	08:16	0:00	pickup -l -t fi	
postfix	4622	0.0	0.1	5460	1684	?	S	08:16	0:00	qmgr -l -t fifo	
root	4624	0.0	0.1	5388	1216	?	Ss	08:16	0:00	/usr/sbin/nmbd	
root	4626	0.0	0.1	7724	1404	?	Ss	08:16	0:00	/usr/sbin/smbd	
root	4631	0.0	0.0	7724	812	?	S	08:16	0:00	/usr/sbin/smbd	
daemon	4636	0.0	0.0	2316	216	?	SN	08:16	0:00	distccd --daemo	
proftpd	4682	0.0	0.1	9948	1608	?	Ss	08:16	0:00	proftpd: (accep	
daemon	4696	0.0	0.0	1984	420	?	Ss	08:16	0:00	/usr/sbin/atd	
root	4707	0.0	0.0	2104	892	?	Ss	08:16	0:00	/usr/sbin/cron	
root	4735	0.0	0.0	2052	352	?	Ss	08:16	0:00	/usr/bin/jsvc -	
root	4736	0.0	0.0	2052	480	?	S	08:16	0:00	/usr/bin/jsvc -	
tomcat55	4738	6.1	8.8	364328	91996	?	S1	08:16	1:03	/usr/bin/jsvc -	

La quarta vulnerabilità è questa:

HIGH
rlogin Service Detection

### Description

The rlogin service is running on the remote host. This service is vulnerable since data is passed between the rlogin client and server in cleartext. A man-in-the-middle attacker can exploit this to sniff logins and passwords. Also, it may allow poorly authenticated logins without passwords. If the host is vulnerable to TCP sequence number guessing (from any network) or IP spoofing (including ARP hijacking on a local network) then it may be possible to bypass authentication. Finally, rlogin is an easy way to turn file-write access into full logins through the .rhosts or rhosts.equiv files.

### Solution

Comment out the 'login' line in /etc/inetd.conf and restart the inetd process. Alternatively, disable this service and use SSH instead.


### Output

No output recorded.

To see debug logs, please visit individual host

Port ▲	Hosts
513 / tcp / rlogin	192.168.64.10

Rlogin è un protocollo che viene utilizzato per il controllo remoto. Per ovviare a questo problema occorre disabilitare questo tipo di servizio. Per farlo occorre eliminare una riga nel file di impostazione del servizio.



The image shows a terminal window titled "Metasploit 2" running the GNU nano 2.0.7 text editor. The editor is editing the file /etc/inetd.conf. The content of the file is as follows:

```
#<off># netbios-ssn      stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.tftpd
telnet      stream  tcp      nowait  telnetd /usr/sbin/tcpd  /usr/sbin/in.telnetd
#<off># ftp          stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.ftp
tftp        dgram   udp      wait    nobody   /usr/sbin/tcpd  /usr/sbin/in.tftpd
shell       stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.rsh
#login     stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.rlogin
exec        stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.rexecd
ingreslock stream  tcp      nowait  root    /bin/bash bash -i
```

The bottom of the window shows the nano editor's command palette with various shortcuts like ^G Get Help, ^O WriteOut, ^R Read File, etc.

Dopo la risoluzione di queste criticità possiamo affermare che il rischio generale è diminuito, ma il sistema ancora non è del tutto sicuro in quanto le vulnerabilità più critiche non sono ancora state mitigate tutte. L'ideale sarebbe seguire la roadmap proposta inizialmente partendo con la risoluzione delle criticità alte e successivamente le altre.

Nei report allegati è possibile approfondire anche tutte le vulnerabilità non trattate.