

## EXPLOIT TELNET CON METASPLOIT

### Task:

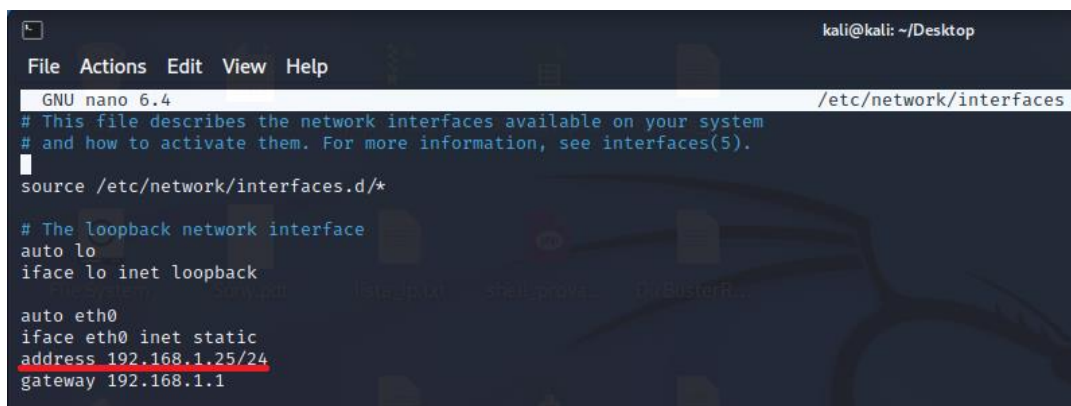
1. Configurazione degli indirizzi di rete di Kali e Metasploitable
2. Exploit del servizio Telnet su Metasploitable utilizzando il modulo *auxiliary telnet\_version*

### 1. Configurazione degli indirizzi di rete di Kali e Metasploitable

Per prima cosa, configuriamo Kali e Metasploitable con i seguenti indirizzi di rete:

Kali → **192.168.1.25**

Metasploitable → **192.168.1.40**

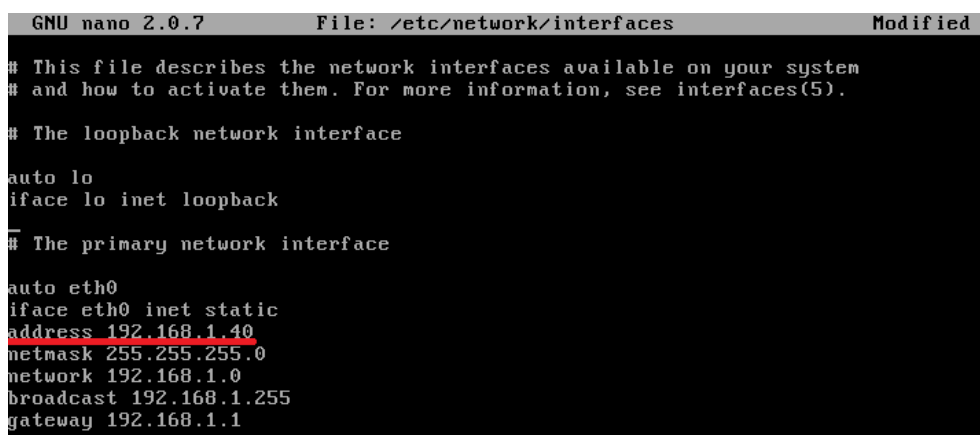


```

kali@kali: ~/Desktop
File Actions Edit View Help
GNU nano 6.4 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.1.25/24
gateway 192.168.1.1
  
```



```

GNU nano 2.0.7 File: /etc/network/interfaces Modified
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.1.40
netmask 255.255.255.0
network 192.168.1.0
broadcast 192.168.1.255
gateway 192.168.1.1
  
```

## 2. Exploit del servizio Telnet su Metasploitable utilizzando il modulo auxiliary *telnet\_version*

L'exploit che vogliamo compiere ha in oggetto il servizio Telnet di Metasploitable; per procedere, ci serviremo del modulo ausiliario (**auxiliary**) **telnet\_version** di Metasploit, che esegue un banner grab del servizio Telnet sulla macchina target. Avviamo dunque la ricerca con il comando **search telnet\_version**

```
msf6 > search telnet_version

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  auxiliary/scanner/telnet/lantronix_telnet_version  normal         No     Lantronix Telnet Service Banner Detection
1  auxiliary/scanner/telnet/telnet_version           normal         No     Telnet Service Banner Detection

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/telnet/telnet_version

msf6 > |
```

Scegliamo il modulo al path 1 con il comando **use auxiliary/scanner/telnet/telnet\_version** e richiediamo il set di parametri di configurazione con il comando **show options**

```
Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  auxiliary/scanner/telnet/lantronix_telnet_version  normal         No     Lantronix Telnet Service Banner Detection
1  auxiliary/scanner/telnet/telnet_version           normal         No     Telnet Service Banner Detection

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/telnet/telnet_version

msf6 > use auxiliary/scanner/telnet/telnet_version
msf6 auxiliary(scanner/telnet/telnet_version) > show options

Module options (auxiliary/scanner/telnet/telnet_version):

Name      Current Setting  Required  Description
-  -  -  -
PASSWORD  no              no        The password for the specified username
RHOSTS    yes             yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT     23              yes        The target port (TCP)
THREADS   1               yes        The number of concurrent threads (max one per host)
TIMEOUT   30              yes        Timeout for the Telnet probe
USERNAME  no              no        The username to authenticate as

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

msf6 auxiliary(scanner/telnet/telnet_version) > |
```

Come si vede, manca la sezione “payload options” perché stiamo operando su un modulo ausiliario, quindi non dovremo impostare alcun payload. L'unico parametro da configurare è RHOSTS, che andremo a popolare con l'indirizzo IP target di Metasploitable, quindi **192.168.1.40**. Fatto ciò, siamo pronti per eseguire il modulo con il comando **run**

