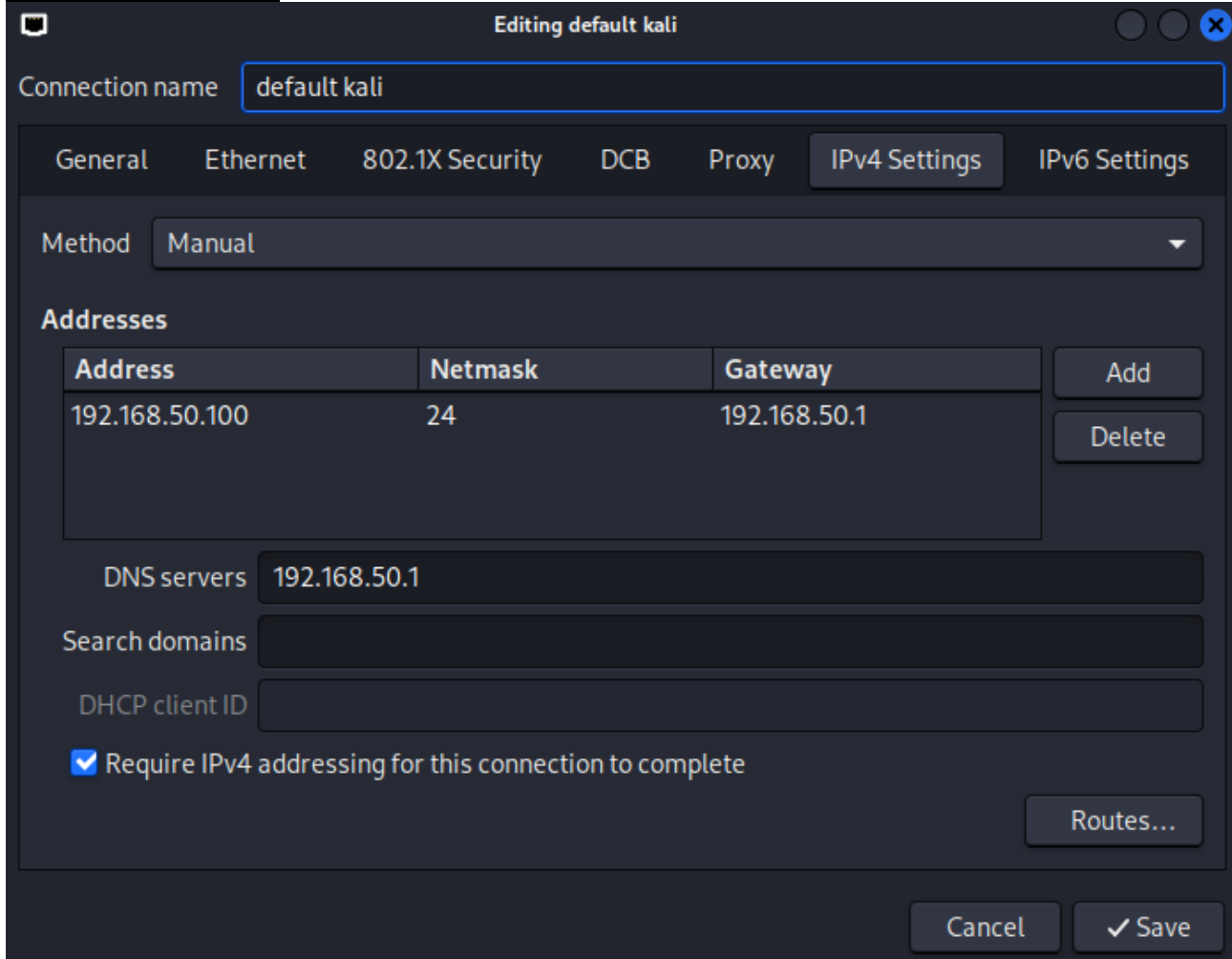


## Creazione e configurazione laboratorio virtuale

In questa esercitazione a seguito dell'installazione file .ova per la creazione delle VM ho configurato ogni scheda di rete per simulare una rete locale con le mie 3 VM.

Configurazione Kali:



Editing default kali

Connection name: default kali

General Ethernet 802.1X Security DCB Proxy **IPv4 Settings** IPv6 Settings

Method: Manual

Addresses

Address	Netmask	Gateway
192.168.50.100	24	192.168.50.1

DNS servers: 192.168.50.1

Search domains:

DHCP client ID:

☒ Require IPv4 addressing for this connection to complete

Routes...

Cancel Save

Per verificare la corretta applicazione ho digitato su terminale "ip a"

Configurazione Metasploitable:

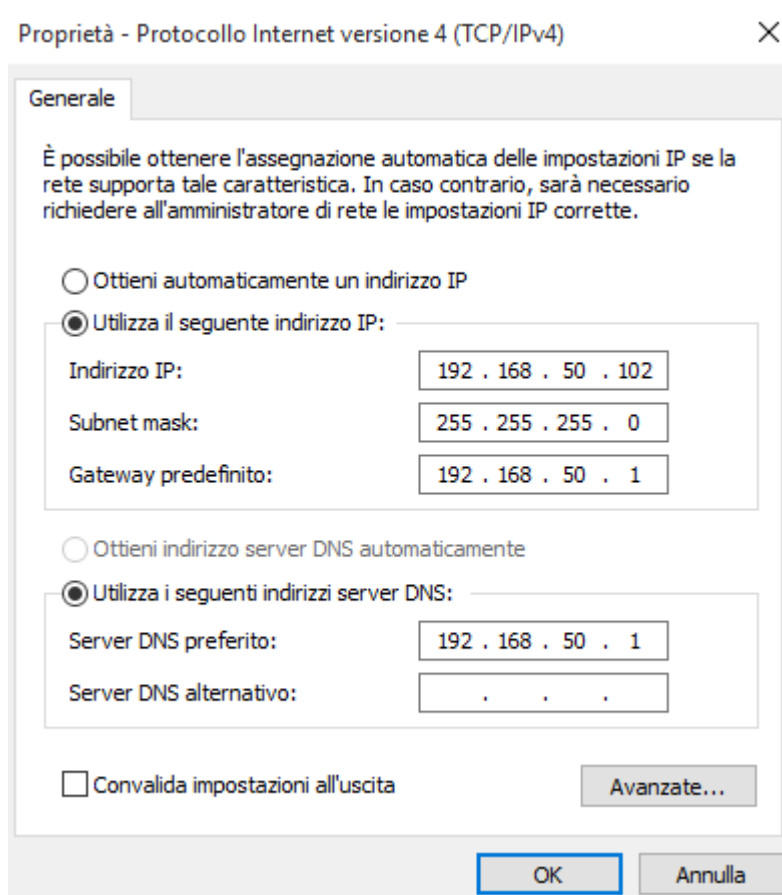
```
msfadmin@metasploitable:~$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.50.101
netmask 255.255.255.0
gateway 192.168.50.1
msfadmin@metasploitable:~$ _
```

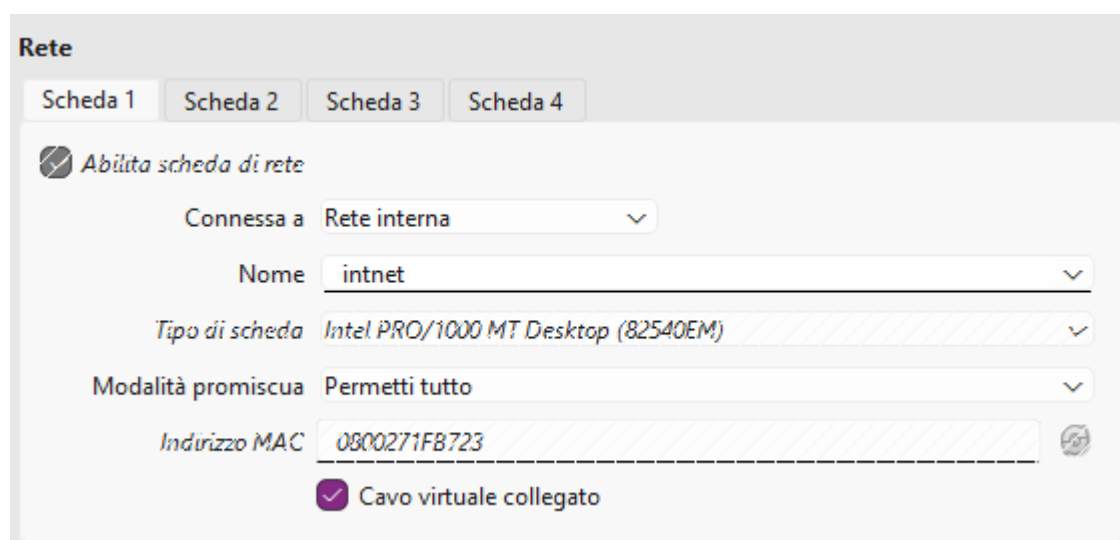
Per verificare la corretta applicazione ho digitato "ip a"

Configurazione Windows 10:



Per verificare la corretta applicazione ho digitato sul terminale "ipconfig"

Oltre alla configurazione delle schede di rete su ogni VM sarà necessario configurare le Virtual Network Interfaces sulle impostazioni di Virtualbox:



Ho impostato la connessione su "Rete Interna" in modo da avere una rete locale in cui le VM comunicassero isolate dal resto, compreso il sistema host. Per far comunicare le VM è importante assegnare alle loro schede di rete virtuali lo stesso nome di rete, altrimenti non comunicheranno anche se il packet loss risulterà allo 0%.

**Prove di ping:**

Kali → Meta

```
(kali㉿kali)-[~]  
$ ping 192.168.50.101  
PING 192.168.50.101 (192.168.50.101) 56(84) bytes of data.  
64 bytes from 192.168.50.101: icmp_seq=1 ttl=64 time=19.3 ms  
64 bytes from 192.168.50.101: icmp_seq=2 ttl=64 time=0.787 ms  
64 bytes from 192.168.50.101: icmp_seq=3 ttl=64 time=11.7 ms  
64 bytes from 192.168.50.101: icmp_seq=4 ttl=64 time=13.6 ms  
^C  
— 192.168.50.101 ping statistics —  
4 packets transmitted, 4 received, 0% packet loss, time 3028ms  
rtt min/avg/max/mdev = 0.787/11.352/19.279/6.704 ms
```

Meta → Kali

```
msfadmin@metasploitable:~$ ping 192.168.50.100  
PING 192.168.50.100 (192.168.50.100) 56(84) bytes of data.  
64 bytes from 192.168.50.100: icmp_seq=1 ttl=64 time=0.115 ms  
64 bytes from 192.168.50.100: icmp_seq=2 ttl=64 time=0.811 ms  
64 bytes from 192.168.50.100: icmp_seq=3 ttl=64 time=0.814 ms  
64 bytes from 192.168.50.100: icmp_seq=4 ttl=64 time=1.70 ms  
  
--- 192.168.50.100 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3078ms  
rtt min/avg/max/mdev = 0.115/0.860/1.702/0.563 ms
```

Kali → Windows

```
(kali㉿kali)-[~]  
$ ping 192.168.50.102  
PING 192.168.50.102 (192.168.50.102) 56(84) bytes of data.  
64 bytes from 192.168.50.102: icmp_seq=1 ttl=128 time=1.44 ms  
64 bytes from 192.168.50.102: icmp_seq=2 ttl=128 time=1.24 ms  
64 bytes from 192.168.50.102: icmp_seq=3 ttl=128 time=0.735 ms  
64 bytes from 192.168.50.102: icmp_seq=4 ttl=128 time=1.07 ms  
^C  
— 192.168.50.102 ping statistics —  
4 packets transmitted, 4 received, 0% packet loss, time 3208ms  
rtt min/avg/max/mdev = 0.735/1.121/1.444/0.260 ms
```

Meta → Windows

```
msfadmin@metasploitable:~$ ping 192.168.50.102  
PING 192.168.50.102 (192.168.50.102) 56(84) bytes of data.  
64 bytes from 192.168.50.102: icmp_seq=1 ttl=128 time=38.3 ms  
64 bytes from 192.168.50.102: icmp_seq=2 ttl=128 time=0.839 ms  
64 bytes from 192.168.50.102: icmp_seq=3 ttl=128 time=0.831 ms  
64 bytes from 192.168.50.102: icmp_seq=4 ttl=128 time=0.863 ms  
  
--- 192.168.50.102 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3031ms  
rtt min/avg/max/mdev = 0.831/10.221/38.351/16.240 ms
```

Windows → Kali

```
C:\Users\user>ping 192.168.50.100

Esecuzione di Ping 192.168.50.100 con 32 byte di dati:
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64

Statistiche Ping per 192.168.50.100:
    Pacchetti: Trasmessi = 4, Ricevuti = 4,
    Persi = 0 (0% persi),
Tempo approssimativo percorsi andata/ritorno in millisecondi:
    Minimo = 0ms, Massimo = 0ms, Medio = 0ms
```

Windows → Meta

```
C:\Users\user>ping 192.168.50.101

Esecuzione di Ping 192.168.50.101 con 32 byte di dati:
Risposta da 192.168.50.101: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.101: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.101: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.101: byte=32 durata<1ms TTL=64

Statistiche Ping per 192.168.50.101:
    Pacchetti: Trasmessi = 4, Ricevuti = 4,
    Persi = 0 (0% persi),
Tempo approssimativo percorsi andata/ritorno in millisecondi:
    Minimo = 0ms, Massimo = 0ms, Medio = 0ms
```

### Conclusione:

Tutta la configurazione è andata a buon fine, le macchine comunicano correttamente tra loro.

L'unico aspetto curioso che ho notato è stato in fase di ping dalle VM verso il sistema host, infatti dalla kali e meta il packet loss risultava allo 100% mentre dalla Windows al sistema host risultava allo 0%, anche se la risposta era di non raggiungibilità.

```
C:\Users\user>ping 192.168.56.1

Esecuzione di Ping 192.168.56.1 con 32 byte di dati:
Risposta da 192.168.50.102: Host di destinazione non raggiungibile.
Risposta da 192.168.50.102: Host di destinazione non raggiungibile.

Statistiche Ping per 192.168.56.1:
    Pacchetti: Trasmessi = 2, Ricevuti = 2,
    Persi = 0 (0% persi),
```

Quello che succede invece se pingo dal sistema host ad una qualsiasi delle VM è questo:

```
C:\Users\User>ping 192.168.50.100

Esecuzione di Ping 192.168.50.100 con 32 byte di dati:
Richiesta scaduta.

Statistiche Ping per 192.168.50.100:
    Pacchetti: Trasmessi = 1, Ricevuti = 0,
    Persi = 1 (100% persi),
```