

# OpenVPN vs Wireguard

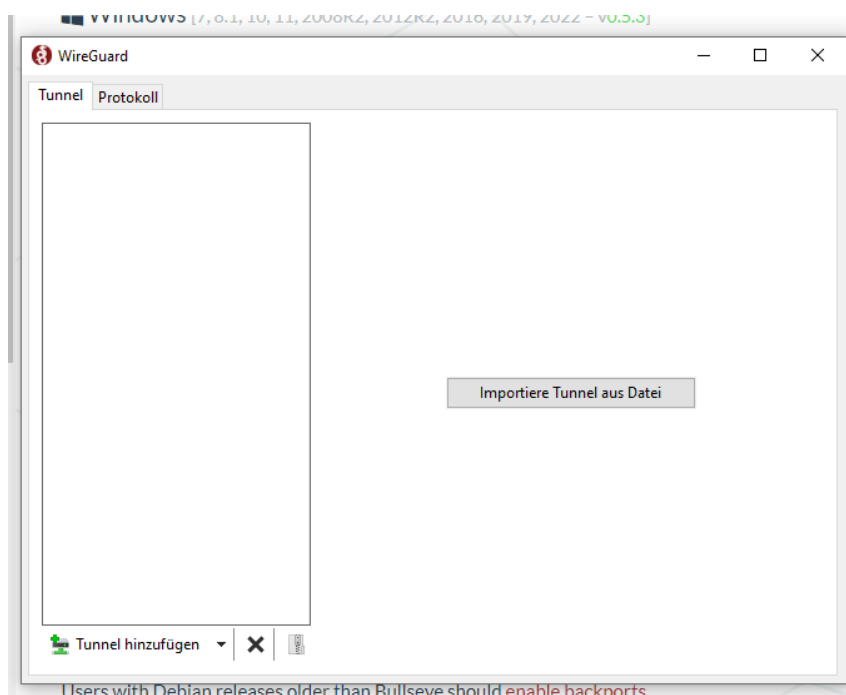
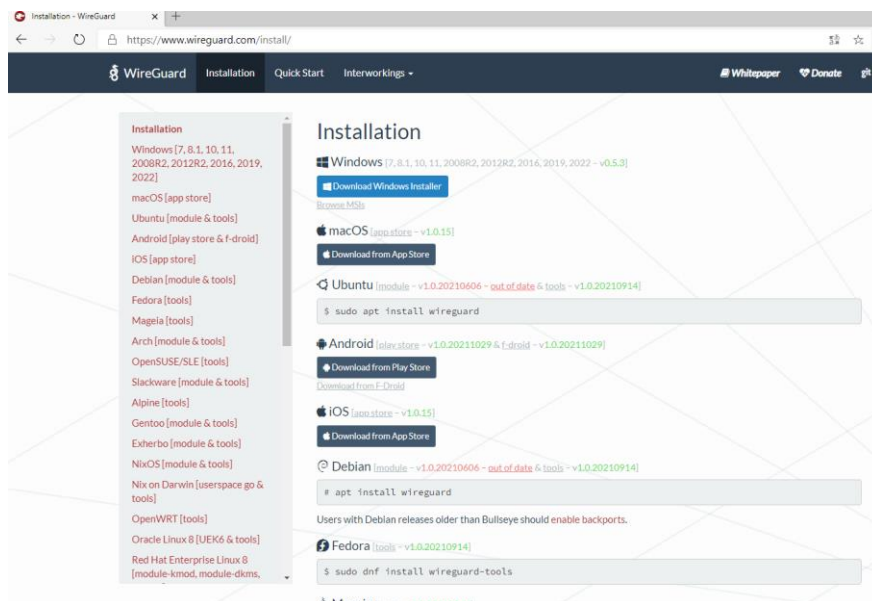
# VPN



## 1. Wireguard

WireGuard arbeitet als Peer-to-Peer-Protokoll ausschließlich über UDP und damit verbindungslos. Das Umgehen von Firewalls per TCP, wie es OpenVPN auf den HTTP/HTTPS-Ports anbietet, ist nicht möglich. Die Verbindung zwischen zwei Peers realisiert WireGuard über einen einzelnen, frei wählbaren UDP-Port.

### 1.1 Setup



## 1.2 Informieren

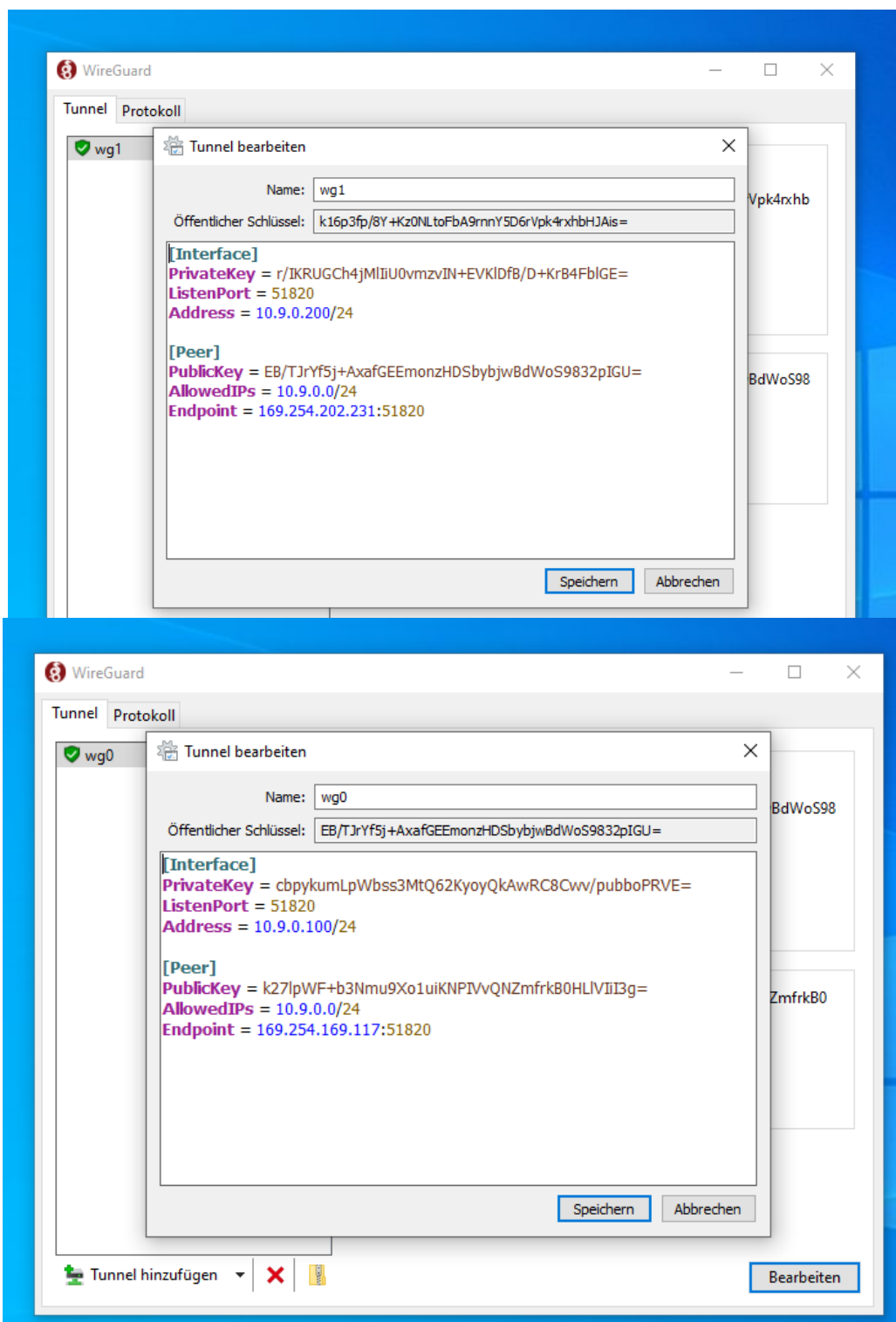
WireGuard ist eine einfache VPN welche von Client zu Client (P2P) wie auch von Client zu Server funktioniert. Die VPN ist auf sehr vielen Betriebssystemen verfügbar, auch auf Windows wie ich es gemacht habe.

## 1.3 Planen

Um einen aktiven VPN-Tunnel herzustellen muss auf beiden Clients WireGuard für Clients heruntergeladen werden und das Config File bearbeitet werden. Ich habe dafür 2 Clients vorbereitet, welchem im gleichen LAN Segment sind.

## 1.4 Entscheiden

## 1.5 Realisieren



1.6 Kontrollieren

1.7 Auswerten

## 1. OpenVPN

Das Open Source OpenVPN verwendet VPN-Technologien, um über das Internet gesendete Daten zu sichern und zu verschlüsseln. Sein benutzerdefiniertes VPN-Protokoll verwendet SSL/TLS für den Schlüsselaustausch. Seit seiner Gründung im Jahr 2001 hat es sich mit über 60 Millionen Downloads zum VPN-Standard im Open-Source-Netzwerkbereich entwickelt

### 1.1 Details / Infos über die Systeme

#### Ubuntu Server

Benutzername	roblwe
Passwort	ZEETMaster360
Computername	ubuntusrv

Hardware Options	
Device	Summary
Memory	4 GB
Processors	2
Hard Disk (SCSI)	30 GB
CD/DVD 2 (SATA)	Using file D:\ISO\ubuntu-20....
CD/DVD (SATA)	Using file autoinst.iso
Floppy	Using file autoinst.flp
Network Adapter	NAT
Network Adapter 2	LAN Segment
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

NAT → ens33

LAN → ens37 (192.168.0.2)

#### Windows 10 Client

Device	Summary
Memory	4 GB
Processors	2
Hard Disk (NVMe)	60 GB
CD/DVD (SATA)	Auto detect
Floppy	Auto detect
Network Adapter	NAT
Network Adapter 2	LAN Segment
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

NAT → Ethernet0

LAN → Ethernet1 (192.168.0.3)

## 1.2 Setup mit Gateway

Ubuntu VM:

```
$ wget https://git.io/vpn -O openvpn-install.sh
```

- Somit wird das Skript für die Installation von OpenVPN installiert.

```
$ sudo chmod +x openvpn-install.sh
```

```
$ sudo bash openvpn-install.sh
```

```
Welcome to this OpenVPN road warrior installer!

Which IPv4 address should be used?
  1) 192.168.40.129
  2) 192.168.0.2
IPv4 address [1]: 2_
```

Es soll die LAN-IP-Adresse benutzt werden, damit der Client im gleichen LAN das VPN benutzen kann.

```
Welcome to this OpenVPN road warrior installer!

Which IPv4 address should be used?
  1) 192.168.40.129
  2) 192.168.0.2
IPv4 address [1]: 2

This server is behind NAT. What is the public IPv4 address or hostname?
Public IPv4 address / hostname [77.109.144.90]: 192.168.0.2
```

Es soll die LAN-Adresse eingegeben werden, da dies auch die externe Adresse des VPN-Servers ist.

```
Which protocol should OpenVPN use?
  1) UDP (recommended)
  2) TCP
Protocol [1]: 1
```

1, Weil es das Default Protokoll ist.

```
What port should OpenVPN listen to?
Port [1194]: 1194
```

Port: 1194 für UDP / Port: 443 für TCP (Ich werde 1194 auswählen, da ich vorher das UDP ausgewählt habe)

```
Select a DNS server for the clients:
  1) Current system resolvers
  2) Google
  3) 1.1.1.1
  4) OpenDNS
  5) Quad9
  6) AdGuard
DNS server [1]: 2
```

2, weil ich diesen am meisten benutze.

```
Enter a name for the first client:
Name [client]: WindowsVPNClient
```

Ich werde meinen Windows Client, als ersten VPN-Client definieren.

```
OpenVPN installation is ready to begin.
Press any key to continue...
```

OpenVPN ist nun konfiguriert und gestartet.

```
[sudo systemctl «command» openvpn-server@server.service]
```

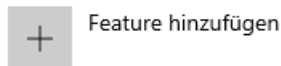
Dieser Command ist für das Neustarten, stoppen, starten usw. des VPN-Services.

Die Konfigurationsdatei namens: «WindowsVPNClient.ovpn» befindet sich im Ordner /root/

Die Datei muss nun auf den Windows Client Kopiert werden.

```
root@ubuntusrv:~# scp ~/WindowsVPNClient.ovpn roblwe@192.168.0.3:/C:/scp/WindowsVPNClient.ovpn
roblwe@192.168.0.3's password:
WindowsVPNClient.ovpn                                100% 5003      1.3MB/s   00:00
root@ubuntusrv:~# _
```

## Optionale Features



### Neueste Aktionen



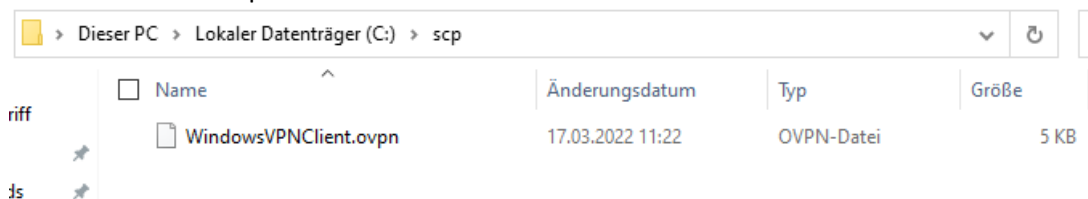
\*

\*

Scp funktioniert nur, wenn ein ssh Server auf dem Windows Client läuft.  
Dieses Feature muss noch hinzugefügt und dann gestartet werden.

Staren in PS: «Start-Service sshd»

Danach sollte der scp Command funktionieren.



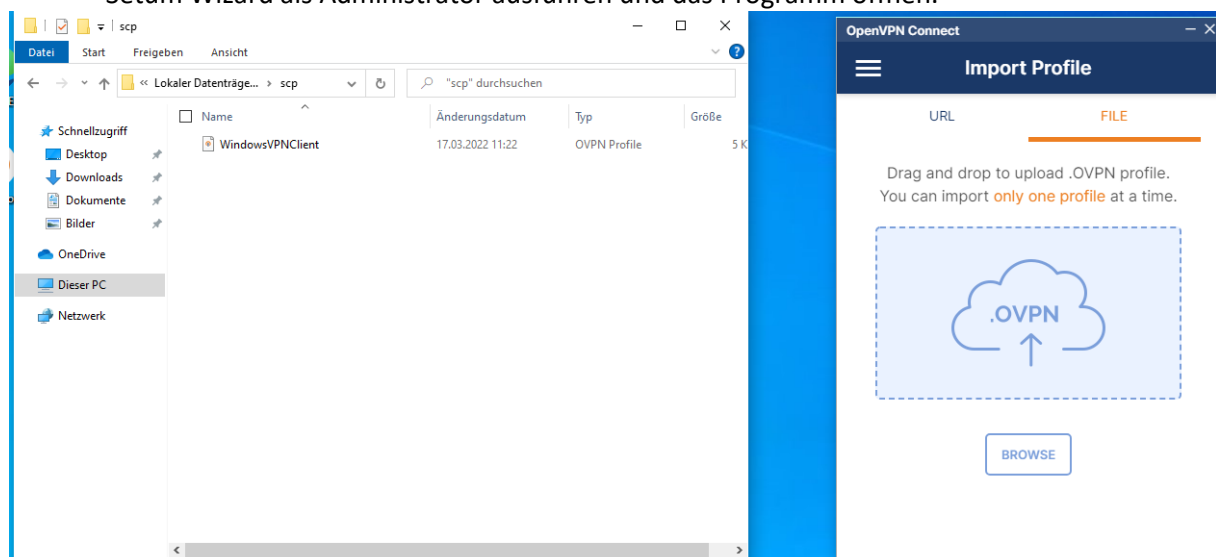
Wie zu sehen ist, befindet sich die Datei nun im Ordner «C:\scp».

Windows VM:

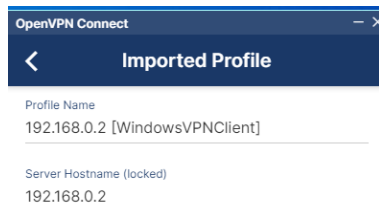
Unter folgendem Link soll der OpenVPN Client installiert werden.

<https://openvpn.net/downloads/openvpn-connect-v3-windows.msi>

- Setum Wizard als Administrator ausführen und das Programm öffnen.



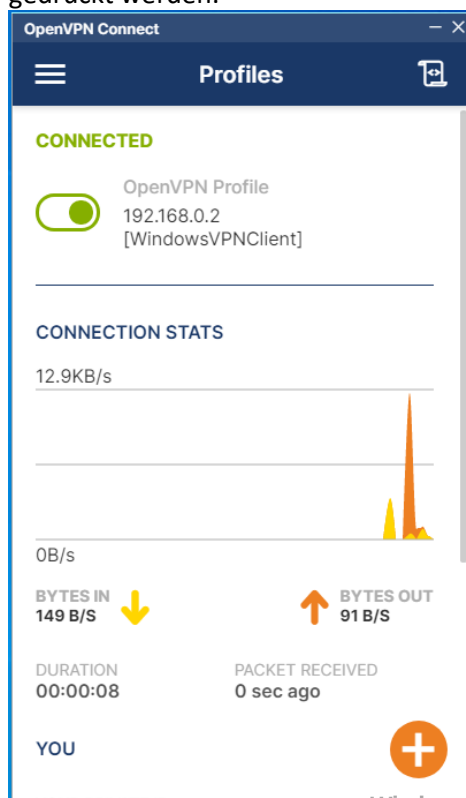
Nun muss das vorher kopierte File in das OpenVPN App importiert werden. (Drag and drop)



PROFILES

CONNECT

Die Konfiguration wird übernommen und automatisch eingetragen. Nun muss nur noch Connect gedrückt werden.



Windows ist nun über Windows mit dem Ubuntu-VPN-Server verbunden.

Dies kann nun mit so vielen Rechnern, wie gewünscht gemacht werden.

Ausserdem kommt man mit dem Rechner jetzt über des Server ins Internet, auch wenn der Windows Client kein NAT-Interface hat.

So ist nun alles aufgebaut:

VPN-Client (NAT)----->VPN>-----Ubuntu-srv (NAT + LAN)-----Client.

Ping von VPN-Client über des VPN-Tunnel zum Client. ( :



## Setup Client → Client

Device	Summary
Memory	4 GB
Processors	1
Hard Disk (SCSI)	40 GB
CD/DVD 2 (SATA)	Using file D:\ISO\ubuntu-20....
CD/DVD (SATA)	Using file autoinst.iso
Floppy	Using file autoinst.flp
Network Adapter	NAT
Network Adapter 2	LAN Segment
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Beide Ubuntu Client Geräte sind gleich.

Passwort: Brot16

LAN → ens37 (192.168.0.1/2)

```
$ sudo apt install openvpn
```

```
$ openvpn --genkey --secret ~/temp-p2p-network.key --keysize 20
```

```
$ touch /root/p2p.ovpn
```

Folgender Inhalt:

```
dev tun
port 51999
ifconfig 172.31.200.1 172.31.200.2
secret temp-p2p-network.key
# Compress traffic
comp-lzo
# These settings ensure that OpenVPN reconnects when the partner changes
his IP Address
keepalive 10 60
cipher AES-256-CBC
ping-timer-rem
auth-nocache
persist-tun
persist-key
```

OpenVPN als Root starten: `$ openvpn --config /root/p2p.ovpn`

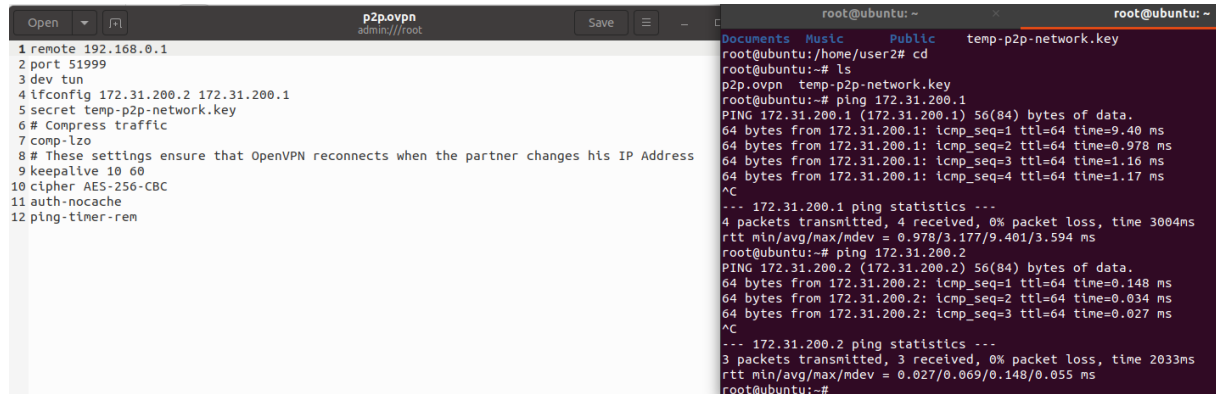
```
root@ubuntu:~# openvpn --config /root/p2p.ovpn
Thu Mar 17 06:52:13 2022 disabling NCP mode (--ncp-disable) because not in P2MP
client or server mode
Thu Mar 17 06:52:13 2022 OpenVPN 2.4.7 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO]
[LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Jul 19 2021
Thu Mar 17 06:52:13 2022 library versions: OpenSSL 1.1.1f 31 Mar 2020, LZO 2.10
Thu Mar 17 06:52:13 2022 TUN/TAP device tun0 opened
Thu Mar 17 06:52:13 2022 /sbin/ip link set dev tun0 up mtu 1500
Thu Mar 17 06:52:13 2022 /sbin/ip addr add dev tun0 local 172.31.200.1 peer 172.
31.200.2
Thu Mar 17 06:52:13 2022 Could not determine IPv4/IPv6 protocol. Using AF_INET
Thu Mar 17 06:52:13 2022 UDPv4 link local (bound): [AF_INET][undef]:51999
Thu Mar 17 06:52:13 2022 UDPv4 link remote: [AF_UNSPEC]
```

In beiden Computern bis hierher, genau das gleiche machen.

- Der Key muss nun auf beide /root/ kopiert werden.

Anschliessend muss OpenVPN auch noch beim anderen Gerät gestartet werden.

Nun kann man die IP-Adresse vom VPN pingen. Also 172-Adresse



The image shows two terminal windows side-by-side. The left window, titled 'p2p.ovpn', displays the configuration for an OpenVPN client. The right window, titled 'root@ubuntu: ~', shows the execution of commands to copy a key file and perform ping tests to the VPN server.

```
Open  p2p.ovpn  Save  root@ubuntu: ~  root@ubuntu: ~
1 remote 192.168.0.1
2 port 51999
3 dev tun
4 ifconfig 172.31.200.2 172.31.200.1
5 secret temp-p2p-network.key
6 # Compress traffic
7 comp-lzo
8 # These settings ensure that OpenVPN reconnects when the partner changes his IP Address
9 keepalive 10 60
10 cipher AES-256-CBC
11 auth-nocache
12 ping-timer-rem

Documents Music Public temp-p2p-network.key
root@ubuntu:/home/user2# cd
root@ubuntu:~# ls
p2p.ovpn temp-p2p-network.key
root@ubuntu:~# ping 172.31.200.1
PING 172.31.200.1 (172.31.200.1) 56(84) bytes of data.
64 bytes from 172.31.200.1: icmp_seq=1 ttl=64 time=9.40 ms
64 bytes from 172.31.200.1: icmp_seq=2 ttl=64 time=0.978 ms
64 bytes from 172.31.200.1: icmp_seq=3 ttl=64 time=1.16 ms
64 bytes from 172.31.200.1: icmp_seq=4 ttl=64 time=1.17 ms
^C
--- 172.31.200.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.978/3.177/9.401/3.594 ms
root@ubuntu:~# ping 172.31.200.2
PING 172.31.200.2 (172.31.200.2) 56(84) bytes of data.
64 bytes from 172.31.200.2: icmp_seq=1 ttl=64 time=0.148 ms
64 bytes from 172.31.200.2: icmp_seq=2 ttl=64 time=0.034 ms
64 bytes from 172.31.200.2: icmp_seq=3 ttl=64 time=0.027 ms
^C
--- 172.31.200.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2033ms
rtt min/avg/max/mdev = 0.027/0.069/0.148/0.055 ms
root@ubuntu:~#
```