VPN-Server mit Wireguard / Side-to-Side-VPN / alles tunneln

Es kann mehrere Gründe geben, einen VPN-Tunnel zu nutzen. Sei es zwecks der Anonymität, weil man geogeblockte Seiten öffnen möchte oder die normale Praxis - einfach nur, weil man auf der Gegenstelle einen Server hat, deren Dienste man sicher erreichen möchte.

In unserem Fall möchten wir als IP gerne diese externe IP haben, sodass wir geogeblockte Seiten öffnen können.

Als Server kommt ein V-Server in einem Rechenzentrum zum Einsatz (Debian), als Client ist es lokal ein LXC-Container in Proxmox (Ubuntu). Dies kann natürlich auch der lokale Rechner sein oder ein lokaler Laptop mit Linux drauf.

Wir gehen davon aus, dass wir auf beiden Systemen root sind, ansonsten ist - in diesem Fall - vor **jedem** Befehl ein sudo zu setzen.

Allgemein/Vorbereiten

Dieser Abschnitt ist auf beiden Systemen auszuführen.

Sollte das System recht frisch sein, ist unbedingt ein Aktualisieren der apt-Datenbank von nöten, in allen anderen Fällen kann es nicht schaden:

```
apt update -y && apt upgrade -y
```

Ggf. Wireguard selbst installieren - sollte es nicht vorhanden sein:

```
apt install wireguard
```

Als nächstes muss in der Datei */etc/sysctl.conf* unbedingt die Zeile *#net.ipv4.ip_forward=1* auskommentiert werden:

```
nano /etc/sysctl.conf
```

aus

```
#net.ipv4.ip_forward=1
```

wird

```
net.ipv4.ip_forward=1
```

Als nächstes generieren wir die Schlüssel für den gegenseitigen Austausch:

```
umask 077; wg genkey | tee /etc/wireguard/privatekey | wg pubkey > /etc/wireguard/publickey
```

Diesen lassen wir uns jeweils auf beiden Systemen anzeigen für die Konfigurationsdatei nachher:

```
cat /etc/wireguard/privatekey && cat /etc/wireguard/publickey
```

Serverkonfiguration

Als erstes fragen wir ab, welches Interface hardwareseitig benutzt wird. Dies ist meist eth0. Über dieses Interface soll nachher alles laufen. Dieses ist mit

```
ip a
```

abzufragen.

Nun beginnen wir mit der Erstellung der Konfigurationsdatei auf dem Server. Folgender Inhalt muss in die Wireguard-Konfiguration *wg0.conf*, wie wir sie hier nennen. Wenn ein anderer Name als wg0 verwendet wird, muss in allen Fällen, wo wir wg0 verwenden, natürlich der individuelle Name stehen.

```
nano /etc/wireguard/wg0.conf
```

```
[Interface]
PrivateKey = SERVER-PRIVATEKEY
Address = 172.31.0.1/32
SaveConfig = true
PostUp = iptables -A FORWARD -i wg0 -j ACCEPT; iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
PostDown = iptables -D FORWARD -i wg0 -j ACCEPT; iptables -t nat -D POSTROUTING -o eth0 -j MASQUERADE
ListenPort = 51820


[Peer]
```

```
PublicKey = CLIENT-PUBLICKEY
AllowedIPs = 172.31.0.2/32, X.X.X.X/XX
```

Hier gibt es einiges zu beachten. Der SERVER-PRIVATEKEY und CLIENT-PUBLICKEY muss mit den Keys vom Server und Clienten ersetzt werden. Das Interface ist zu beachten, wie oben beschrieben. Die IP-Adresse 172.31.0.1/32 suggeriert ein Netz, deren /32-Subnetz technischer Weise nur 2 IP-Adressen erlaubt. Nämlich Server (172.31.0.1) und Client (172.31.0.2). Diese beiden benutzen dieses Netz und kommunizieren über den Tunnel. Der Port 51820 ist der Wireguard-Standard-Port, kann aber geändert werden, muss aber dann natürlich auf beiden Seiten gleich sein. Die PostUp und PostDown-Zeilen bilden einfach die IPtables, wie wo was geroutet wird.

Im Abschnitt Peer bei den Allowed IPs muss unbedingt die IP des Clienten enthalten sein, in diesem Fall eben 172.31.0.2/32. Für unseren Fall des Gateways ist es unbedingt erforderlich, das eigene Netz, welches wir daheim haben, dort einzusetzen.

Haben wir die 192.168.1.6 als IP in einem /24er-Netz, ist, sofern das ganze Netz geroutet werden darf, die 192.168.1.0/24 stehen. Im Falle eines 172.20.2.86 als IP bei einem /16er-Netz, ist dort auch die 172.20.0.0/16 zu wählen. Einzelne IPs können auch gewählt werden, wenn nur ein Gerät durch den Tunnel sollte.

Clientkonfiguration

Wir legen auch auf dem Client die Konfigurationsdatei mit gleichem Namen wg0 an:

```
nano /etc/wireguard/wg0.conf
```

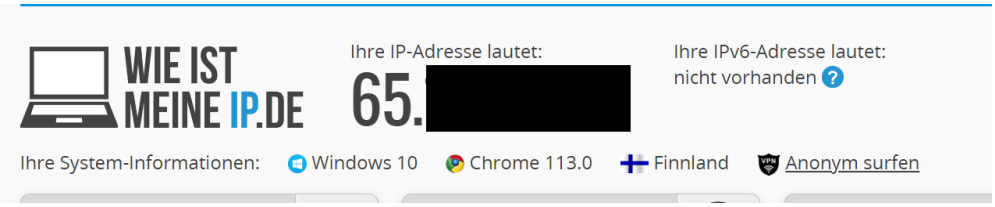
```
[Interface]
PrivateKey = CLIENT-PRIVATEKEY
Address = 172.31.0.2/32

[Peer]
PublicKey = SERVER-PUBLICKEY
Endpoint = Adresse:51820
AllowedIPs = 0.0.0.0/0
PersistentKeepalive = 25
```

Hier ist natürlich wieder zu beachten, dass die Keys entsprechend eingesetzt werden. Bei Adresse kann die IP des Servers oder auch eine Domain stehen. Die AllowedIPs mit dem Wert 0.0.0.0/0 sagt aus, dass alles über den VPN-Server geroutet wird, also auch Internet.

Und da wir hier mit Wireguard im UDP-Protokoll unterwegs sind, gibt es keine Bestätigung der Verbindung. Demzufolge muss permanent der PersistentKeepalive abgefragt werden, hier alle 25 Sekunden.

Abschluss

Sollte der Dienst nach einem Neustart eines der beiden Systemen automatisch wieder starten, ist folgender Befehl abzusetzen:



War man erfolgreich, sollte dies bei https://wieistmeineip.de erscheinen.

```
systemctl enable wg-quick@wg0
```

Revision #5
Created 24 November 2024 09:54:20 by Admin
Updated 20 April 2026 13:18:41 by Admin