FritzBox Heimnetz-Zugriff hinter DSLite mit Raspberry PI über Wireguard

Dieser Beitrag behandelt die Problematik des Heimnetz-Zugriffs auf eine FritzBox bzw. auf das Netz hinter einer FritzBox mit einem DSLite Internetanschluss sowie eine mögliche Lösung.

Das Problem

Das wesentliche Problem, weshalb die Heimnetzverbindung nicht einfach per FritzBox interner VPN realisiert werden kann, liegt in der Funktionalität des DSLite Verfahrens.

Bei DualStack Lite (DSLite) handelt es sich um eine Anschlussform bei Internetanschlüssen wo neben einer nativen IPv6 Adresse, eine geteilte IPv4 Adresse vom Internetanbieter vergeben wird. Dadurch werden alle Anfragen von der FritzBox über die IPv6 Adresse versendet. Verbindungen die über IPv4 laufen müssen werden bei DSLite von der FritzBox in einen IPv6 Request verpackt und später von einer Gegenstelle des Internetanbieters wieder entpackt und über IPv4 weitergeleitet.

Da die FritzBox also über keine eigene IPv4 Adresse verfügt, ist die Heimnetzverbindung über die FritzBox VPN technisch leider aktuell nicht möglich. Diese Problematik ist hier auf der Seite von AVM auch erwähnt.

Mögliche Lösung

Um dennoch Zugriff auf das Netzwerk daheim zu erhalten, habe ich nach einer eigenen VPN Lösung gesucht, welche sowohl von IPv4 (Aus dem Mobilnetz) als auch von IPv6 Anschlüssen zu erreichen ist. Es gibt zwar im Internet jede Menge an Anbietern welche IPv4-zu-IPv6 Tunnel anbieten, doch weiß man hierbei nie genau wo seine Daten landen und über welche Server diese laufen. Zusätzlich wollte ich eine Lösung, welche mit den vollen Zugriff auf mein lokales Netz gibt und auch die Interaktion zu SmartHome Gerätenzulässt.

Einem wird hier schnell klar, dass eine eigene Lösung für den Zugriff ins Heimnetz notwendig wird.

Meine Lösung

Meine nun funktionierende Lösung besteht aus einem virtuellen Server in einem Rechenzentrum auf dem ein Wireguard VPN Server läuft. Dieser erlaubt durch eine zugewiesene IPv4 und IPv6 Adresse Zugriffe aus allen Netzen und somit auch von Handys aus dem Mobilfunk.

Damit nun das Heimnetz erreichbar ist, befindet sich in meinem lokalen LAN ein Raspberry Pi 3 der als VPN Client fungiert und einen permanenten Tunnel zu dem VPN Server offen hält. Durch die Konfiguration von statischen Routen ist mir nun der Kontakt zu meinem Heimnetz möglich.

Somit ist es vollbracht!

Die Konfiguration im Detail

Vorraussetzung

-

- virtueller Server mit IPv4 und IPv6 Adresse
- Raspberry Pi 3 oder neuer mit Raspbian Buster Lite
- IPv6 Subnet (optinal)
- FritzBox Router (Wird hier verwendet, ggf. anderer Router der statische Routen unterstützt)

Installation des VPN Servers

Grundsätzlich ist die Installation von Wireguard auf dem virtuell Server denkbar einfach. Als Betriebssystem für den Server kann nahezu jede Linux Distribution verwendet werden.

Die Installation von Wireguard ist für die einzelnen System hier aufgeführt.
Damit später die Pakete korrekt über den Server laufen, muss das Routing von IPv4 Paketen aktiviert werden:

$ echo "net.ipv4.ip_forward = 1" > /etc/sysctl.d/wg.conf $
sysctl --system

Für den Server (und auch später für den Client) benötigen wir ein Schlüsselpaar:

1. # Berechtigungen korrekt setzen
2. $ umask 077
3. # privaten Schlüssel erstellen
4. $ wg genkey > privatekey6.
7. # öffentlichen Schlüssel aus privatem Schlüssel erstellen
8. $ wg pubkey < privatekey > publickey

Nun erstellen wir die Server Konfig:

$ nano /etc/wireguard/wg0.conf

1. [Interface]
2. Table = off
3. Address = 172.16.100.1/24
4. PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE; ip route add 192.168.178.0/24 via 172.16.100.1 mtu 1420
5. PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -t nat -D POSTROUTING -o eth0 -j MASQUERADE 6. ListenPort = <Gewünschter Server Port>
7. PrivateKey = <Server Private Key einfügen> 8.
9. # Raspberry Pi Heimnetz
10. [Peer]
11. PublicKey = <Public Key Raspberry>
12. AllowedIPs = 0.0.0.0/0 13.
14. # ... weitere Clients einfügen

Die Konfiguration erhält in der Form oben den privaten Adressbereich 172.16.100.0 welcher dann das Netz des VPN wird. Auszufüllen sind nun noch die entsprechenden Felder mit den Private- und PublicKeys.

Entscheidend dafür, dass das lokale Netzwerk hinter FritzBox später über den VPN Tunnel erreichbar ist, ist das setzen einer Route im vServer, welche ins Netz der Fritzbox führt. Damit die Route immer vorhanden ist, wird diese über die VPN Config mit dem Befehl ‚PostUp' gesetzt.
Wichtig ist, dass in der Route das korrekte Netz der FritzBox gesetzt ist!

Nachdem wir nun die Konfiguration gesetzt haben, können wir den VPN Server starten:

```
$ wg-quick up wg0
```

Installation des Raspberry Pis

Für eine einfache Installation und später einen performanten Betrieb empfiehlt sich den Raspberry Pi per LAN direkt an die FritzBox anzuschließen. Damit der PI direkt per SSH erreichbar ist, sollte nach dem flashen des Betriebssystems auf der Hauptordner-Ebene eine leere Datei mit dem Namen

'ssh'

erstellt werden. Auf diese Weise benötigen wir später keinen Monitor am Pi um die erste Konfiguration durchzuführen.

Die Installation von Wireguard erfolgt auf dem Pi wie folgt:

1. $ apt-get update
2. $ apt-get upgrade
3. $ apt-get install raspberrypi-kernel-headers
4. $ echo "deb http://deb.debian.org/debian/ unstable main" | tee --append /etc/apt/sources.list.d/unstable.list
5. $ apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 04EE7237B7D453EC
6. $ printf 'Package: *\nPin: release a=unstable\nPin-Priority: 150\n' | tee --append /etc/apt/preferences.d/limit-unstable
7. $ apt-get update
8. $ apt-get install wireguard

Die Wireguard Konfiguration erstellen wir wie beim Server auch mit Nano und füllen sie mit folgendem Inhalt:

1. [Interface]
2. Table = off
3. PrivateKey = <Private Key vom Pi>
4. Address = 172.16.100.9/24 5.
6. [Peer]
7. PublicKey = <Public Key vom VPN Server>
8. AllowedIPs = 0.0.0.0/0
9. Endpoint = <IPv6 Adresse vom VPN Server>:<gewählter Port vom VPN Server> 10. PersistentKeepalive = 25

In der Konfiguration ist die Adresse 172.16.100.9 die von mir gewählte Adresse aus dem VPN Netz für den Pi. Zu Beachten ist bei der Adresse vom VPN Server bei dem Punkt ‚Endpoint', dass die IPv6 Adresse in eckige Klammern gesetzt werden muss, sofern keine Domain angegeben wird. Also: [2630:dedc:d98b::871f]:1234 bzw. myvpn.mydomain.de:1234

Nach einem $ wg-quick up wg0 ist das Interface auf dem Pi aktiv und verbindet sich automatisch zum VPN Server.

Route in der Fritzbox

Damit möglichem Traffic klar ist, wie er zu fließen hat, muss eine Rückroute von der FritzBox über den Pi zur VPN gesetzt werden.
Erstellt werden muss die statische Route in der FritzBox Oberfläche unter Heimnetz -> Netzwerk -> Reiter Netzwerkeinstellungen -> Statische Routingtabelle -> IPv4-Routen.

Sollte die Einstellung nicht auffindbar sein, stelle sich er, dass sich die FritzBox Oberfläche im Expertenmodus befindet. Die entsprechende Option findet man links unten in der Menüleiste mit einem Klick auf ‚Ansicht: Standrad'.

IPv4-Netzwerk: Netz des VPN
Subnetzmaske: Abhängig von dem gewählten Netz (meistens 255.255.255.0)
Gateway: lokale LAN IP-Adresse des Raspberry Pis (Ersichtlich in der FritzBox bei Heimnetz)

Abschluss

Durch die gerade getätigte Konfiguration hat man nun wieder die Möglichkeit auf das lokale Netzwerk Zuhause von überall auf der Welt zuzugreifen. Durch das geschickte verwenden von Routen kann man nun von allen Geräten welche zum VPN verbunden sind Geräte aus dem 192.168.178.0 Netz erreichen.

Erdenklich wäre nun das Steuern von Computern aus der Ferne über Remote Desktop, Abfragen und Bedienen von SmartHome Geräten oder gar das Drucken aus der Ferne. Grundsätzlich sind hier der Kreativität keine Grenzen gesetzt.

Das Gange ist nicht von mir, es wurde nur zusammen gefasst.

Prisrak