

Wireguard Public IP weiterleiten

- Einrichtung Loadbalancer
- Erstellung Wireguard Server und Client Configs
- Einrichtung Client unter OPNSense

Einrichtung Loadbalancer

Einrichtung Loadbalancer

Auf Loadbalancer 1 einloggen und folgende Pakete installieren

```
apt install lvs
```

Nun eine neue Datei mit folgender Config anlegen

```
nano /etc/keepalived/keepalived.conf
```

Inhalt einfügen

```
global_defs {
    notification_email {
        info@hacker-net.de
    }
    notification_email_from lb001@strange-hosting.com
    smtp_server hacker-net.de
    smtp_connect_timeout 60
}

vrrp_instance VI_1 {
    state MASTER
    interface eth0
    virtual_router_id 101
    unicast_src_ip 159.69.13.9
    unicast_peer { 49.12.74.225 }
    priority 101
    notify /root/failover.sh
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 775567567ghBysA!asHjk99gf!
    }
}
```

```
virtual_ipaddress {
    116.202.189.172
}
virtual_ipaddress_excluded {
    2a01:4f8:1c0c:8218::1
}
}
```

Nun die failover.sh im root Verzeichnis anlegen

```
nano /root/failover.sh
```

Nun Inhalt einfügen

```
#!/bin/bash

#lb0001 ID : 5605626
#lb0002 ID : 5605974
#floating ip LBS 116.202.189.172 =: 245713
#floating id KC 78.46.239.42 =: 617007
#floatingip6 id 2a01:4f8:1c17:8106::/64 =: 246612
#floating ip GBL 16126703 =: 78.47.118.115

TYPE=$1
NAME=$2
STATE=$3

case $STATE in
    "MASTER") curl -X POST -H "Content-Type: application/json" -H "Authorization: Bearer
5BEoQe1HDjITxzhCPmnp8cJq1Ybv6ehNIgdvrfkiECG0fyASSOQbIIFMj9GF0IsV" -d '{"server": 5605626}'
'https://api.hetzner.cloud/v1/floating_ips/245713/actions/assign'
        curl -X POST -H "Content-Type: application/json" -H "Authorization: Bearer
5BEoQe1HDjITxzhCPmnp8cJq1Ybv6ehNIgdvrfkiECG0fyASSOQbIIFMj9GF0IsV" -d '{"server": 5605626}'
'https://api.hetzner.cloud/v1/floating_ips/246612/actions/assign'
        curl -X POST -H "Content-Type: application/json" -H "Authorization: Bearer
5BEoQe1HDjITxzhCPmnp8cJq1Ybv6ehNIgdvrfkiECG0fyASSOQbIIFMj9GF0IsV" -d '{"server": 5605626}'
'https://api.hetzner.cloud/v1/floating_ips/16126703/actions/assign'
        curl -X POST -H "Content-Type: application/json" -H "Authorization: Bearer
5BEoQe1HDjITxzhCPmnp8cJq1Ybv6ehNIgdvrfkiECG0fyASSOQbIIFMj9GF0IsV" -d '{"server": 5605626}'
```

```
'https://api.hetzner.cloud/v1/floating_ips/617007/actions/assign'
```

```
    echo "Master"
    exit 0
;;
"BACKUP") echo "backup"
    exit 0
;;
"FAULT") echo "fault"
    exit 0
;;
*)    echo "unknown state"
    exit 1
;;
esac
```

Nun auf Loadblancer zwei einloggen

Folgene pakete installieren

```
apt install lvs
```

Nun eine neue Datei mit folgender Config anlegen

```
nano /etc/keepalived/keepalived.conf
```

Inhalt einfügen

```
nano /root/failover.sh
```

Erstellung Wireguard Server und Client Configs

Auf dem Server:

Erstellung der Server Config, Muss nur einmal gemacht werden.

Schlüsselpaar erstellen.

Sollen mehre IPs erstellt werden empfehle ich für jeden Server + client ein Verzeichnis zu erstellen

Beispiel:

Dazu ein Verzeichnis keys anlegen mit folgenden Unterverzeichnissen je IP (Diese IP Adressen abhe ich nicht, Beispiel

```
.
├── ip_72.44.34.21
│   ├── client
│   └── server
├── ip_72.44.34.22
│   ├── client
│   └── server
```

Nun in das Verzeichnis Server der ersten IP gehen.

```
wg genkey | tee privatekey | wg pubkey > publickey
```

Nun haben wir zwei Schlüsseldateien.

Einmal private Key und publickey

Jetzt sieht das ganze so aus

```
.
├── ip_72.44.34.21
│   ├── client
│   └── server
│       ├── privatekey
│       └── publickey
├── ip_72.44.34.22
│   ├── client
│   └── server
```

Nun die Config Datei erstellen mit den Schlüsseldateien.

Eine Wireguard Config ist in zwei Teile geteilt.

Einmal Interface das der Lokale Teil für den Computer

(Auch ein Client hat einen lokalen Teil, der dient dazu das Interface bereitzustellen.

Der zweite Teil sind die Peers, es können auch mehrere Peers eingetragen sein. Aber wir erstellen hier für jede IP eine eigene config. Den Publickey für den Peer haben wir noch nicht.

```
nano ~/keys/ip_27.44.34.21/server/ip27443421server.conf
```

```
[Interface]
```

```
Address = 172.16.0.1 # oder irgendein anderes privates Netz from RFC1918 (privates Netz) 10.10.0.x oder 192.168.0.100 oder oder egal. Wichtig, auf einem Computer dürfen die entze nicht gleich sein
```

```
PrivateKey = <private_key_vom_server>
```

```
ListenPort = 51820 # or jeder andere port der beliebt
```

```
[Peer]
```

```
PublicKey = <public_key_vom_client_also_der_derdie_ip_bekommen_soll>
```

```
AllowedIPs = publicip_v4/32,ipv6 mit subnet # Here tragen wir unsere PUBLIC IPS ein, die auf der PEER Seite zur Verfügung stehen sollen
```

Ein Beispiel, diese Keys wurden NIE verwendet, also viel Spaß damit

Inhalt Privatekey Server

```
mJn1IWnRFTze3wojk3a+d5TsPSOolRqt3dfN7ekgukU=
```

Inhalt Publickey Server

```
/amPOUXRsKhzW1QsLvQ/7UjchU1oFfkWYZbZj/mWtxA=
```

```
[Interface]
```

```
Address = 192.168.100.1
```

```
PrivateKey = mJn1IWnRFTze3wojk3a+d5TsPSOolRqt3dfN7ekgukU=
```

```
ListenPort = 51820
```

```
[Peer]
```

```
PublicKey = <haben_wir_noch_nicht_leer_lassen>
AllowedIPs = 27.44.34.21/32, 3a01:4d8:xxxx:8106::/64
```

Nun die Client Schlüssel erstellen konfig erstellen
Dazu ins client Verzeichnis gehen und wieder

```
wg genkey | tee privatekey | wg pubkey > publickey
```

Nun haben wir auch dort im Verzeichnis Client auch zwei Schlüssel

```
├── ip_72.44.34.21
│   ├── client
│   │   ├── privatekey
│   │   └── publickey
│   └── server
│       ├── privatekey
│       └── publickey
├── ip_72.44.34.22
│   ├── client
│   └── server
```

Unserer Privatekey

```
0ls0rjpsSpzBPD0T+Quv/3GWGe1nPKrNfXJa80wUY0E=
```

Unserer Public Key

```
2gWEqeQxoGa9hI94UvV4trVVdiG1G7sq0iLr6W5ymUE=
```

Nun in der Serverconfig den Public key nachtragen unter Peers

```
nano ~/keys/ip_27.44.34.21/server/ip27443421server.conf
```

Nun im peer bereich public key "haben wir noch nicht erstezen" Platzhalter mit dem öffentlichem Schlüssel vom Client ersetzen

```
....
[Peer]
PublicKey = 2gWEqeQxoGa9hI94UvV4trVVdiG1G7sq0iLr6W5ymUE=
AllowedIPs = 27.44.34.21/32, 3a01:4d8:xxxx:8106::/64
```

Nun die Client Config erstellen

```
nano ~/keys/ip_27.44.34.21/client/ip27443421client.conf
```

[Interface]

Address = publicip_v4/32,ipv6 OHNE subnet

PrivateKey = <private_key_vom_client>

ListenPort = 51820 # irgendeiner darf nur auf dem client nicht doppelt sein. Denn dieser Port wird ja nie benutzt. Denn wir verbinden uns ja zum Server und nicht umgekehrt

[Peer]

PublicKey = <public_key_vom_server_also_der_derdie_ip_bekommen_soll>

Endpoint: ip_adresse_vom_server(Die die der Server fest hat) und den Listen Port aus der Server conf

AllowedIPs = 0.0.0.0/0,::/0 Der erste Wert heißt als standard Gateway für ipv4, der zweite standard gateway für ipv6. Spricht route alles

Beispiel:

[Interface]

Address = 27.44.34.21

PrivateKey = 0ls0rjpsSpzBPDoT+Quv/3GWGe1nPKrNfXJa80wUY0E=

ListenPort = 51820

[Peer]

PublicKey = /amP0UXRsKhzW1QsLvQ/7UjchU1oFfkWYZbZj/mWtxA=

Endpoint: 27.44.34.20:51820

AllowedIPs = 0.0.0.0/0,::/0

Nun sind die Configs fertig und können ausgerollt werden.

Dazu die Server config nach /etc/wireguard kopieren

```
cp ~/keys/ip_27.44.34.21/server/ip27443421server.conf /etc/wireguard
```

Nun den Tunnel beim start aktiv schalten

```
systemctl enable wg-quick@ip27443421server
```

```
systemctl start wg-quick@ip27443421server
```

Wenn ein Loadbalancer im einsatz ist, die Schlüssel und configs auch auf dem Slaveloadbalancer kopieren.

Dort im /etc/wireguard die config auch reinkopieren und auch dort mit systemctl enable den Dienst in die Autostart packen und danach starten

Nun die Client config auf den Client kopieren, entweder den text per Copy und paste übermitteln in eine neue Datei auf den client oder per scp.

Diese muss dann auch in

```
/etc/wireguard/ip27443421client.conf
```

Nun diese auch beim starten ausführen

```
systemctl enable wg-quick@ip27443421client  
systemctl start wg-quick@ip27443421client
```

Fertig.

Einrichtung Client unter OPNSense

In OPNSense einloggen dann unter -> System -> Firmware -> Packages ->

The screenshot shows the OPNsense web interface. On the left is a navigation sidebar with the following items: Lobby, Reporting, System (highlighted), Access, Configuration, Firmware, Status, Settings, Changelog, Updates, Plugins, Packages, Reporter, Log File, Gateways, High Availability, Routes, Settings, Trust, Wizard, Log Files, Diagnostics, and Interfaces. The main content area displays a list of operating system packages, with 'os-wireguard' highlighted. The list includes: os-theme-tukan, os-theme-vicuna, os-tinc, os-tor, os-udpbroadcastrelay, os-upnp, os-virtualbox, os-vmware, os-vnstat, os-web-proxy-sso, os-wireguard, os-wol, os-xen, os-zabbix-agent, os-zabbix4-proxy, os-zabbix5-proxy, os-zabbix6-agent, os-zabbix6-proxy, os-zabbix62-agent, os-zabbix62-proxy, and os-zerotier.

Wireguard auswählen und installieren

The screenshot shows the OPNsense web interface. The left sidebar contains a navigation menu with categories: Lobby, Reporting, System, Access, Configuration, Firmware, Reporter, Log File, Gateways, High Availability, Routes, Settings, and Trust. The 'System' category is expanded, and 'Firmware' is selected. The main content area is titled 'System: Firmware' and has tabs for Status, Settings, Changelog, Updates, Plugins, and Packages. The 'Status' tab is active, displaying the following log output:

```
***GOT REQUEST TO INSTALL***  
Currently running OPNsense 22.7 (amd64/OpenSSL) at Tue Nov 15 07:29:41 UTC 2022  
Installation out of date. The update to opnsense-22.7.7_1 is required.  
***DONE***
```

At the bottom of the log output, there is a note: "Output shown here for diagnostic purposes. There is no general need for manual system intervention. [Click here to copy to clipboard.](#)"

Dann F5 drücken damit die Seite aktualisiert.

Nun unter VPN Wireguard gehen.

Dort den Haken bei Wireguard rein.

The screenshot shows the OPNsense web interface. The left sidebar contains a navigation menu with categories: Lobby, Reporting, System, Interfaces, Firewall, VPN, Services, Power, and Help. The 'VPN' category is expanded, and 'WireGuard' is selected. The main content area is titled 'VPN: WireGuard' and has tabs for General, Local, Endpoints, Status, and Handshakes. The 'General' tab is active, displaying the following configuration options:

- Enable WireGuard** (with a green checkmark icon)
- Apply** button

Nun auf den Registerreiter Local und dort auf das kleine + drücken

VPN: WireGuard

General Local Endpoints Status Handshakes

Search [7] [List Icon]

<input type="checkbox"/> Enabled	Name	Interface	Tunnel Address	Port	Endpoints	Commands
No results found!						
						<input style="border: 1px solid red;" type="button" value="+"/>

Showing 0 to 0 of 0 entries

« < 1 > »

Nun folgendes Ausfüllen:

Die Daten können per Copy and paste aus der vorher herstellten conf Datei gezogen werden.
Leider unterstützt OPNSense kein import von Conf Dateien

Name : ipadresse ohne punkte dient zur besseren Übersicht

Private Key : den Privaten Key vom Client

Listen Port : irgendeiner hauptsache auf dem Client nicht schon vergeben : 5555

Tunnel Address : die ipdresse mit 32 Netz

alles andere so lassen

Dann Save

Edit Local Configuration



advanced mode full help

Enabled

Name

Instance 1

Public Key
Public key of this instance. You can specify your own one, or a key will be generated after saving.

Private Key
Private key of this instance. You can specify your own one, or a key will be generated after saving. Please keep this key safe.

Listen Port

Tunnel Address
✖ Clear All 📄 Copy

Peers
✖ Clear All

Disable Routes

Cancel

Save

Nun auf den Registerreiter Endpoints und dort auf das kleine Plus klicken

VPN: WireGuard

General Local **Endpoints** Status Handshakes

Enabled

Name	Endpoint Address	Endpoint Port	Allowed IPs	Commands
No results found!				

⏪ ⏩ 1 ⏪ ⏩

+

Showing 0 to 0 of 0 entries

Apply

Nun folgendes ausfüllen

Name PUBIP : PUIP und dahinter die IP ohne punkte

Den Public Key vom Server nicht Client


Allowed IP alle

Endpoint Adresse : Die Adresse wo der VPN Server Erreichbar ist.

dann auf Save.

Edit Endpoint

×

full help 



Enabled	<input checked="" type="checkbox"/>
Name	PUBIP784
Public Key	lBg6T78E [redacted] wzOgV02...
Shared Secret	
Allowed IPs	0.0.0.0/0 × Clear All Copy
Endpoint Address	116.20 [redacted]
Endpoint Port	51821
Keepalive Interval	25

CancelSave

Nun wieder zurück auf den Registerreiter Local da den Eintrag editieren.
Nun im Dropdownmenü peers, den Endpunkt auswählen und Speichern.

Edit Local Configuration

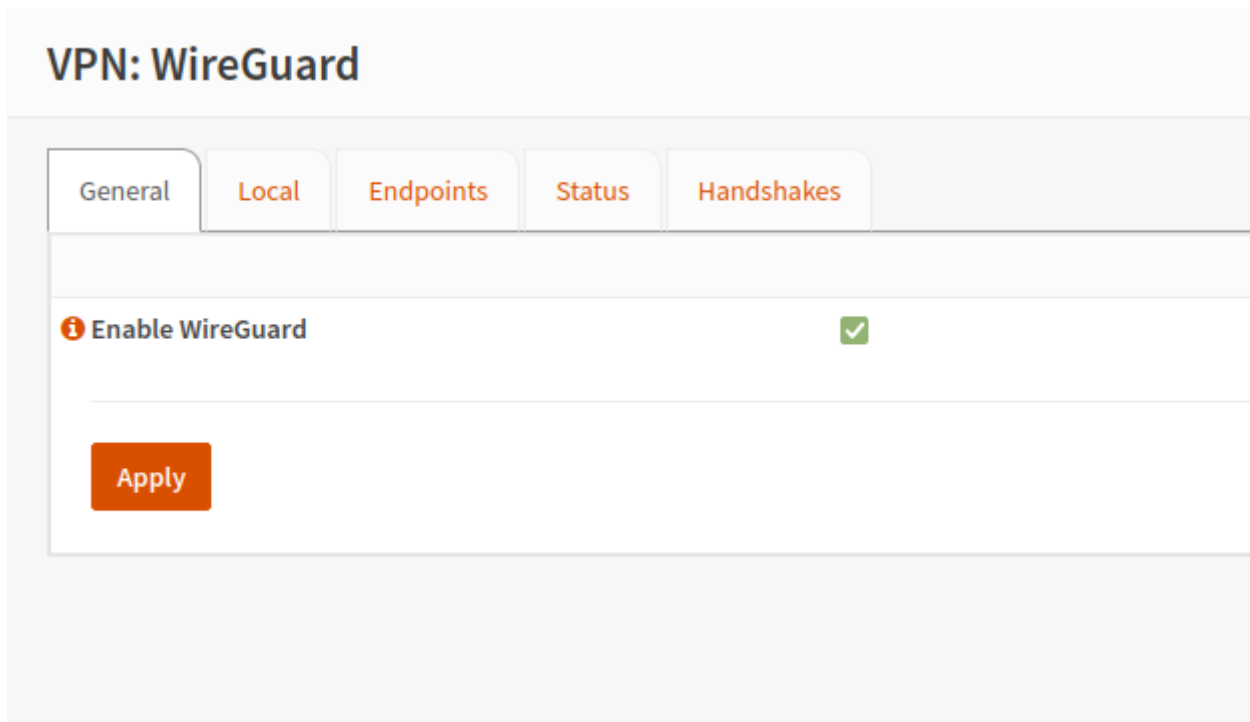
×

 advanced mode full help 

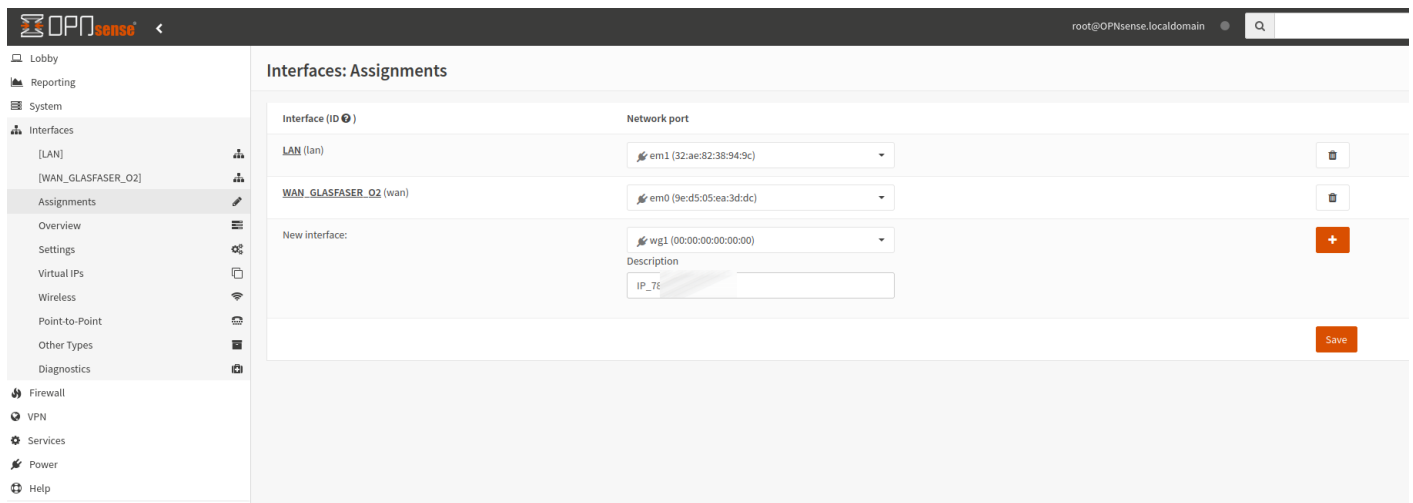
Enabled	<input checked="" type="checkbox"/>
Name	784 [redacted]
Instance	1
Public Key	
Private Key	kArZl [redacted] ;+ ...
Listen Port	
Tunnel Address	78.4 [redacted] /32 × Clear All Copy
Peers	Nothing selected ▼
Disable Routes	<input type="text" value="PUBIP784 [redacted]"/>

CancelSave

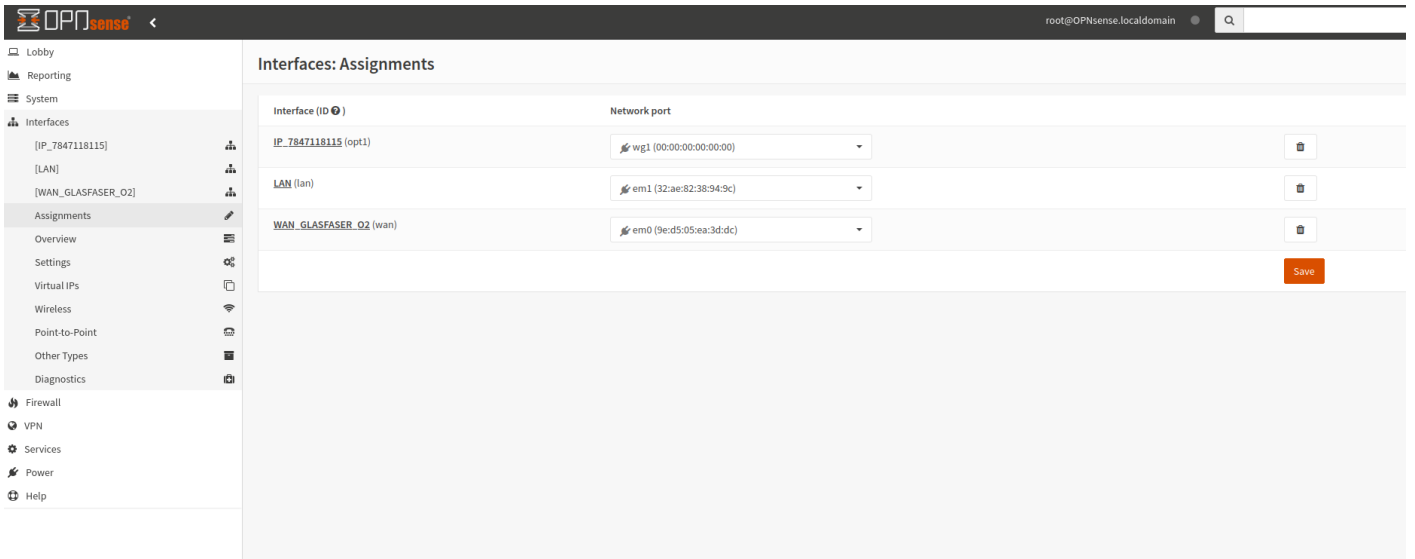
Danach im Registerreiter General,
den Haken wieder raus. Apply klicken,
Dann haken wieder rein. Apply klicken.



Nun Unterinterfaces -> Assignments -> In der Beschreibung WG1 IP_eintragen. Und dann auf +
drücken



Danach auf Save klicken



Nun in Interfaces , dann auf neu erstelle Interface



Nun folgende Einstellungen vornehmen