vpn1.erfurt.freifunk.net

Dies ist ein VPN-Server.

HINWEISE

- DNS-Name noch nicht angepasst: Für IPv6 funktioniert SSH-Login also nicht über DNS-Namen!
- Port 1234 für fastd und 10000 für fastd-Backbone zeigen bereits auf die VM
- Port 10001 zeigt weiterhin auf die VM gluon-ffef, diese hat noch die Karte und darf im Backbone nicht aus dem VPN entfernt werden

Serverinformationen

Administratoren

- mape2k (Inhaber/Bereitsteller)
- bt909
- hipposen

IP/DNS

- vpn1.erfurt.freifunk.net
  - 144.76.76.98
  - 2a01:4f8:191:9461:13::1

Dienste

- SSH (Port 1035)
- fastd (Port 1234)

Software

- Debian 8 (Jessie)
  - Installation-Optionen: SSH-Server, Standard-Systemutilities

Installation

Installierte Pakete (System)

- mc
- screen
- vim
- sudo

Netzwerk

Pakete

- bridge-utils

Konfiguration Routing

- IPv6-Forwarding generell aktivieren
  - kann nicht Interface-bezogen aktiviert werden
- IPv4-Forwarding wird von fastd Interface-bezogen aktiviert

```
net.ipv6.conf.all.forwarding = 1
```

Konfiguration Routingtabellen

- gesonderte Routingtabelle für Freifunk-internen Datenverkehr

```
23 ffef
```

Konfiguration Bridge (Freifunk-Netz)

```
# Bridge (Freifunk)
iface brffef inet static
  bridge_ports none
  address 10.99.1.1
  broadcast 10.99.1.255
  netmask 255.255.128.0
  post-up /sbin/ip route add 10.99.0.0/17 dev $IFACE table ffef
  post-up /sbin/ip rule add iif $IFACE table ffef priority 200
  post-up /sbin/ip rule add oif $IFACE table ffef priority 201
  post-up echo 1 > /proc/sys/net/ipv4/conf/$IFACE/forwarding
  pre-down echo 0 > /proc/sys/net/ipv4/conf/$IFACE/forwarding
  pre-down /sbin/ip route del 10.99.0.0/17 dev $IFACE table ffef
  pre-down /sbin/ip rule del oif $IFACE table ffef priority 201
  pre-down /sbin/ip rule del iif $IFACE table ffef priority 200
iface brffef inet6 static
  address fd0a:d928:b30d:94f7:1::1
  netmask 64
```

fastd

Repository

- Jessie-Backports verwenden

```
deb http://ftp.debian.org/debian jessie-backports main
```

Pakete

- fastd
  - apt-get -t jessie-backports install fastd

Workaround für fehlerhafte Startskripte

- cp /lib/systemd/system/fastd.service /etc/systemd/system/fastd@.service
- systemctl daemon-reload

Quelle: https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=823801

Backbone-Verbindung

- mkdir -p /etc/fastd/backbone/peers
- fastd --generate-key

--generate-key

```
2016-05-23 18:40:15 +0000 --- Info: Reading 32 bytes from /dev/random...
Secret: XXX
Public: YYY
```

- /etc/fastd/backbone/secret.conf mit Secret-Key befüllen

```
secret "XXX";
```

- Public-Key auf anderen Backbone-VPN-Servern einrichten

```
# VPN-Server vpn1.erfurt.freifunk.net
```

```
key "YYY";
remote "vpn1.erfurt.freifunk.net" port 10000;
```

- Fastd-Konfiguration
  - IP-Adresse des VPN-Servers im Backbone setzen
  - Policy-Routing für ffef-Routingtabelle setzen
  - IPv4-Forwarding für fastd-Interface aktivieren
  - Keepalived starten/beenden (Floating IP für statische)

```
log level info;
interface "mesh-vpn-bb";
mode tap;
method "null+salsa2012+umac";
method "null";
include "secret.conf";
bind any:10000;
mtu 1426;
include peers from "peers";

on up "
   ip link set up dev $INTERFACE
   ip address add 10.99.254.7/24 broadcast 10.99.254.255 dev $INTERFACE
   ip route add 10.99.254.0/24 dev $INTERFACE table ffef
   ip rule add iif mesh-vpn-bb table ffef priority 300
   ip rule add from 10.99.254.7 table ffef priority 301
   ip route add default via 10.99.254.1 table ffef
   echo 1 > /proc/sys/net/ipv4/conf/$INTERFACE/forwarding
   systemctl start keepalived
";

on down "
   systemctl stop keepalived
   echo 0 > /proc/sys/net/ipv4/conf/$INTERFACE/forwarding
   ip route del default via 10.99.254.1 table ffef
   ip rule del iif mesh-vpn-bb table ffef priority 300
   ip rule del from 10.99.254.7 table ffef priority 301
   ip route del 10.99.254.0/24 dev $INTERFACE table ffef
   ip address del 10.99.254.7/24 broadcast 10.99.254.255 dev $INTERFACE
   ip link set down dev $INTERFACE
";
```

- Dateien aus /etc/fastd/backbone/peers/ von anderen VPN-Servern übernehmen
  - Fix Me!: Synchronisierbar gestalten oder aus zentralem Repository beziehen

Node-Verbindung

- mkdir -p /etc/fastd/nodes/peers
- fastd --generate-key

--generate-key

```
2016-05-23 23:07:46 +0000 --- Info: Reading 32 bytes from /dev/random...
Secret: XXX
Public: YYY
```

- /etc/fastd/nodes/secret.conf mit Secret-Key befüllen

```
secret "XXX";
```

- Public-Key ins Wiki und die Firmware übernehmen

- Fastd-Konfiguration
  - IP-/MAC-Adressen der Nodes nicht loggen
  - IPv4-Forwarding für fastd-Interface aktivieren

```
log level info;
interface "mesh-vpn";
mode tap;
method "null+salsa2012+umac";
method "salsa2012+gmac";
hide ip addresses yes;
hide mac addresses yes;
include "secret.conf";

bind any:1234;
mtu 1426;
include peers from "peers";

on up "
   ip link set address de:ff:ef:ff:ef:01 up dev $INTERFACE
   ip link set up dev $INTERFACE
   echo 1 > /proc/sys/net/ipv4/conf/$INTERFACE/forwarding
";

on down "
   echo 0 > /proc/sys/net/ipv4/conf/$INTERFACE/forwarding
   ip link set down dev $INTERFACE
";
```

- Netzwerkeinstellungen für Batman über Distribution vornehmen

```
# Fastd-Interface (Nodes)
allow-hotplug mesh-vpn
iface mesh-vpn inet6 manual
  post-up         /usr/local/sbin/batctl -m bat0 if add $IFACE
  post-up         /sbin/ip link set dev bat0 up
```

- Dateien für Nodes nach /etc/fastd/nodes/peers/ kopieren
  - Fix Me!: Synchronisierbar gestalten oder aus zentralem Repository beziehen

Cronjob zum Syncen der Node-VPN-Keys

```
# Get vpn keys for nodes
* * * * *    root    [[ $(rsync -ai --delete 10.99.254.10::nodes/ /etc/fastd/nodes/peers/) ]]
&& killall -SIGHUP fastd
```

Starten und zum Runlevel hinzufügen

- systemctl start fastd@backbone
- systemctl enable fastd@backbone
- systemctl start fastd@nodes
- systemctl enable fastd@nodes

Batman

Wir verwenden noch Batman adv 2013.4.0 (compat level 14). Deshalb müssen wir die Kernel-Pakete und batctl selbst bauen

Pakete

- install
- build-essential
- linux-headers-amd64

- git
- gnupg-curl

Kernelmodul bauen

- mkdir ~/build
- cd ~/build
- git clone https://github.com/freifunk-gluon/batman-adv-legacy
- cd batman-adv-legacy
- make
- make install

- modprobe batman-adv
- dmesg

```
[42600.480585] batman_adv: B.A.T.M.A.N. advanced 2013.4.0-23-g91eab38-dirty (compatibility
version 14) loaded
```

```
batman-adv
```

batctl

- mkdir ~/build
- cd ~/build
- wget http://downloads.open-mesh.org/batman/releases/batman-adv-2013.4.0/batctl-2013.4.0.tar.gz
- tar xzf batctl-2013.4.0.tar.gz
- cd batctl-2013.4.0
- make
- make install

Netzwerkkonfiguration

```
# Batman-Interface
allow-hotplug bat0
iface bat0 inet6 manual
  post-up         /sbin/brctl addif brffef $IFACE
  post-up         /usr/local/sbin/batctl -m $IFACE it 10000
  post-up         /usr/local/sbin/batctl -m $IFACE gw server 96mbit/96mbit
  pre-down        /sbin/brctl delif bat0 $IFACE || true
```

Quagga

* Fix Me!: Generell überprüfen, ICVPN1 Konfiganpassung

Pakete

- quagga
- telnet

```
zebra=yes
bgpd=yes
```

```
! -*- zebra -*-
!
! zebra sample configuration file
!
! $Id: zebra.conf.sample,v 1.1 2002/12/13 20:15:30 paul Exp $
!
hostname vpn1.erfurt.freifunk.net
password xxxx
enable password xxxx
!
```

```
! Interface's description.
!
!interface lo
! description test of desc.
!
!interface sit0
! multicast

!
! Static default route sample.
!
!ip route 0.0.0.0/0 203.181.89.241
!

log file /var/log/quagga/zebra.log

! use src ip for local connection
route-map RM_SET_SOURCE permit 10
set src 10.99.254.7
ip protocol bgp route-map RM_SET_SOURCE

table 23
```

```
hostname vpn1
password [PASSWORD]
!
! enable debug log
!
debug bgp updates
!
!
router bgp 65099002
 bgp router-id 10.99.254.7
 bgp confederation identifier 65099
 bgp confederation peers 65099001
 network 10.99.8.0/22

 neighbor ffef-backbone peer-group
 neighbor ffef-backbone soft-reconfiguration inbound
 neighbor ffef-backbone prefix-list ffef-backbone-in in
 neighbor ffef-backbone prefix-list ffef-backbone-out out

! neighbor 10.99.254.1 remote-as 65099001
! neighbor 10.99.254.1 description icvpn2_suicider
! neighbor 10.99.254.1 prefix-list ffef-backbone-in in
! neighbor 10.99.254.1 prefix-list ffef-backbone-out out

 neighbor 10.99.254.10 remote-as 65099001
 neighbor 10.99.254.10 description icvpn2_hipposen
 neighbor 10.99.254.10 prefix-list ffef-backbone-in in
 neighbor 10.99.254.10 prefix-list ffef-backbone-out out

! neighbor 10.99.254.8 remote-as 65099002
! neighbor 10.99.254.8 description vpn3_ichirou
! neighbor 10.99.254.8 peer-group ffef-backbone

 neighbor 10.99.254.9 remote-as 65099002
 neighbor 10.99.254.9 description vpn2_bt909
 neighbor 10.99.254.9 peer-group ffef-backbone

ip prefix-list ffef-backbone-in description *** Backbone IP-Filter eingehend ***
```

```
ip prefix-list ffef-backbone-in seq 10 permit 0.0.0.0/0
ip prefix-list ffef-backbone-in seq 19 deny 10.99.16.0/22
ip prefix-list ffef-backbone-in seq 20 permit 10.99.0.0/16 le 32
ip prefix-list ffef-backbone-in seq 21 permit 10.0.0.0/8 le 32
ip prefix-list ffef-backbone-in seq 30 permit 172.16.0.0/12 le 32
ip prefix-list ffef-backbone-in seq 99 deny 0.0.0.0/0 le 32

ip prefix-list ffef-backbone-out description *** Backbone IP-Filter ausgehend ***
ip prefix-list ffef-backbone-out seq 10 deny 0.0.0.0/0
ip prefix-list ffef-backbone-out seq 20 permit 10.99.0.0/16 le 32
ip prefix-list ffef-backbone-out seq 99 deny 0.0.0.0/0 le 32
!
!
log file /var/log/quagga/bgpd.log
!
!log stdout
```

Dauerhafter Link zu diesem Dokument:
https://wiki.technikkultur-erfurt.de/freifunk:infrastruktur:server:vpn1

Dokument zuletzt bearbeitet am: **28.06.2020 19:05**

Verein zur Förderung von Technikkultur in Erfurt e.V
https://wiki.technikkultur-erfurt.de/