2015

R. Warrior und Partner Consulting

Technische Dokumentation

In diesem Dokument sind alle technischen Informationen festgehalten.

Virtual Private Network

Projektbeginn: 01.12.2014
Project End: 22.05.2015

Auftraggeber:
Anna Nass, Rainer Stoff
R. WARRIOR UND PARTNER CONSULTING
Fucking 42, 5121 Österreich

Auftragnehmer:
Louis Ritter, Maximilian Netter, Ramon Fischer
WIR-MACHEN-DAS GmbH
Hildastraße 24, 69226 Nußloch

Version 347
Technische Dokumentation.odt
Druckdatum: 19. July 2016

Inhaltsverzeichnis

1 Installation und Konfiguration der Server

Konfiguration gültig für VPN-Server1-Debian und VPN-Server2-Debian.

VPN-Server1-Debian / VPN-Server2-Debian
Einrichtung der Sprache und des Tastaturlayouts:

```
Language settings
    Choose Language → German
    Country, territory or area → other → Europe → Germany
    Choose Location → Germany
    Configure the Keyboard → German
```

CD-ROM-Laufwerke in das System einbinden:

```
Detect and mount CD-ROM
```

Komponenten und Module von der CD in das System implementieren:

```
Load installer components from CD
    cfdesk-udeb,
    cpuburn-udeb,
    ntfs-modules,
    udf-modules,
    parted-udeb
```

Konfiguration der Netzwerkkarte:

```
Configure the network
    Waiting time for detection → 3
    Primary network interface → eth1
    Auto configure network → yes
    Waiting time for detection → 3
    Hostname → VPN-Server1-Debian
    Domain Name → “Leer lassen”
```

Einrichtung eines Administratorkontos bzw. eines Passworts:

```
Set up users and passwords
    Enable shadow passwords → yes
    Allow login as root → yes
    Root password: lkwpeter
    Create a normal user account → no
```

Konfiguration der Systemuhr:

```
Configure the clock
    Using NTP → yes
    NTP Server → continue
    Time Zone → Europe/Berlin
```

Durchführung eines Leistungstest?:

```
Perform CPU stresstest (burn in)?
    Burn test → no
```

Eingebundene Festplatten ermitteln:

```
Detect disks
```

Festplatten partitionieren:

```
Partition disks
    Partition method → manually
    SCSI2 → Enter-Taste → Partition table type → gpt
    Free Space → Enter-Taste → Create a new partition → 8GB →
    Use as: Swap area → Done setting up the partition
    Free Space → Enter-Taste → Create a new partition → 72GB →
    Use as: ext4 → mount point: / → Done setting up the
    partition
    Finish partition and write changes to disk → yes
```

Installation des Hauptsystems (Kernel):

```
Install the base system
    Kernel to install → linux-image-amd64
    Drivers to include in the initrd → generic: include all
    available drivers
```

Konfiguration des Paketmanagers:

```
Configure the package manager
    Use a network mirror → yes
    Protocol for filedownloads → http
    Debian archive server → Germany
    Debian archive mirror → ftp.de.debian.org
    HTTP proxy information → “Leer lassen”
    Use non-free software → yes
    Use contrib software → yes
    Services to use → Security updates (from
    security.debian.org), release updates
```

Auswahl und Installation der Programme:

```
Select and install Software
    An der Paketverwendungserfassung teilnehmen? → nein
    Möchten Sie man und mandb “setuid man” installieren? → ja
    Welche Software soll installiert werden? →
    Standardsystemwerkzeuge, SSH-Server
```

GRUB-Bootloader in den “Master Boot Record” installieren:

```
Install the GRUB bootloader on a hard disk
    Install the GRUB bootloader to the master boot record? →
    yes
    Gerät von Hand eingeben → /dev/sda
    Efi → Nein
```

Finish the installation

```
Is the system clock set to UTC? → yes
Installation complete → continue
```

Der Rechner wird neu gestartet und nun folgt die Konfiguration des Servers, wobei man sich als Benutzer “root” anmeldet:

Anmelden als root:

```
login as: root
password: lkwpeter
```

Installation des textbasierten Textverarbeitungsprogramms “vi” (= visual improved):

```
apt-get install vim
```

1.1 Patchplan

Die Server sind folgendermaßen im Schulnetzwerk gepatched:

VPN-Server1-Debian (Mesh):

E.V8.7.1/2 (L) → Panel 1 Port 13 → Cisco-Switch (SG3000-20) Port 3

VPN-Server2-Debian (ISIS/Vodafone):

E.V8.8.1/2 (L) → Panel 1 Port 15

1.2 Netzwerkkonfiguration

VPN-Server1-Debian / VPN-Server2-Debian
DNS-Server eintragen

```
vi /etc/resolv.conf
nameserver 213.73.91.35
```

Installation des DHCP-Programms:

```
apt-get install isc-dhcp-server
```

Konfiguration der Netzwerkschnittstelle, die vom DHCP benutzt wird:

```
vi /etc/default/isc-dhcp-server
    INTERFACES=”eth1”
```

Installation der Netzwerktreiber für Schnittstelle eth1:

```
apt-get install firmware-linux-nonfree firmware-realtek
```

1.3 Einrichtung des DHCPs

VPN-Server1-Debian

```
vi /etc/dhcp/dhcpd.conf
     option domain-name “w-u-p-c.lokal”;
     # Chaos-Computer-Club-DNS-Server
     option domain-name-servers 213.73.91.35;
     subnet 192.168.11.0 netmask 255.255.255.0 {
          range 192.168.11.2 192.168.11.254;
          option routers 192.168.11.1;
     }
```

VPN-Server2-Debian

```
vi /etc/dhcp/dhcpd.conf
     option domain-name “w-u-p-c.lokal”;
     # Chaos-Computer-Club-DNS-Server
     option domain-name-servers 213.73.91.35;
     subnet 192.168.12.0 netmask 255.255.255.0 {
          range 192.168.12.2 192.168.12.254;
          option routers 192.168.12.1;
     }
```

Konfiguration der Netzwerkschnittstellen:

VPN-Server1-Debian

```
vi /etc/network/interfaces
     # Mesh
     auto eth0
     iface eth0 inet static
          address 212.72.180.242
          netmask 255.255.255.224
          gateway 212.72.180.225

     # DHCP
     auto eth1
     iface eth1 inet static
          address 192.168.11.1
          netmask 255.255.255.0
```

VPN-Server2-Debian

```
vi /etc/network/interfaces
     # ISIS/Vodafone
     auto eth0
     iface eth0 inet static
          address 195.158.140.91
          netmask 255.255.255.248
          gateway 195.158.140.90

     # DHCP
     auto eth1
     iface eth1 inet static
          address 192.168.12.1
          netmask 255.255.255.0
```

VPN-Server1-Debian / VPN-Server2-Debian

Aktivierung der Schnittstelle:

```
ifup eth0
ifup eth1
```

1.4 Sicherheitsmaßnahmen

VPN-Server1-Debian / VPN-Server2-Debian

Direktes Anmelden über SSH für Benutzer „root“ verbieten:

```
vi /etc/ssh/sshd_config
     PermitRootLogin no
```

fail2ban installieren, um Angriffe von Außerhalb abzuwehren:

```
apt-get install fail2ban
```

Benutzer hinzufügen:

```
adduser vpn
Password: pkwpeter
```

1.5 Sonstiges

VPN-Server1-Debian / VPN-Server2-Debian

Z shell installieren und als Standard-Shell definieren:

```
apt-get install zsh
chsh -s /bin/zsh root
chsh -s /bin/zsh vpn
```

2 VPN-Verbindung einrichten

VPN-Server1-Debian / VPN-Server2-Debian

iptables-persistent installieren:

```
apt-get install iptables-persistent
```

2.2 NAT einrichten

VPN-Server1-Debian / VPN-Server2-Debian

Um NAT temporär zu aktivieren, muss folgender Befehl ausgeführt werden.

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Um NAT permanent zu aktivieren, wird die Datei */etc/sysctl.conf* geöffnet und folgende Zeile einkommentiert:

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

iptables-Regel definieren und (permanent) speichern:

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
iptables-save > /etc/iptables/rules.v4
```

2.3 openVPN installieren und Zertifikate generieren

VPN-Server1-Debian / VPN-Server2-Debian

openVPN installieren:

```
apt-get install openvpn
```

Standardkonfiguration nach */etc/openvpn* kopieren und extrahieren:

```
cp /usr/share/doc/openvpn/examples/example-config-
files/server.conf.gz /etc/openvpn
gunzip server.conf.gz
```

easy-rsa nach ~ kopieren und *keys*-Ordner im *easy-rsa-Ordner* erstellen:

```
cp -R /usr/share/easy-rsa/ ~
cd ~/easy-rsa
mkdir keys
```

Vorhandene openssl-Datei umbenennen:

```
cp openssl-1.0.0.cnf openssl.cnf
```

Zertifikat-Variablen bearbeiten und dem System bekannt machen:

```
vi vars
     export KEY_COUNTRY=“DE“
     export KEY_PROVINCE=“NRW“
     export KEY_CITY=“Duesseldorf“
     export KEY_ORG=“WIR-MACHEN-DAS“
     export KEY_EMAIL=“noc@wir-machen-d.as“

source vars
```

***keys*-Ordner bereinigen:**

```
./clean-all
```

CA-Zertifikat generieren:

```
./build-ca
```

```
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [DE]: DE
State or Province Name (full name) [NRW]: NRW
Locality Name (eg, city) [Duesseldorf]: Duesseldorf
Organization Name (eg, company) [WIR-MACHEN-DAS]: WIR-MACHEN-DAS
Organizational Unit Name (eg, section) [changeme]:.
Common Name (eg, your name or your server's hostname) [WIR-MACHEN-DAS CA]:VPN-Server2-
Debian
Name [changeme]: VPN-Server2
Email Address [noc@wir-machen-d.as]:noc@wir-machen-d.as
```

Server-Zertifikat generieren:

```
./build-key-server
```

```
Country Name (2 letter code) [DE]:DE
State or Province Name (full name) [NRW]:NRW
Locality Name (eg, city) [Duesseldorf]:Duesseldorf
Organization Name (eg, company) [WIR-MACHEN-DAS]: WIR-MACHEN-DAS
Organizational Unit Name (eg, section) [changeme]:.
Common Name (eg, your name or your server's hostname) [VPN-Server2-Debian]:VPN-
Server2-Debian
Name [changeme]:VPN-Server2
Email Address [noc@wir-machen-d.as]:noc@wir-machen-d.as
Please enter the following 'extra' attributes
to be sent with your certificate request
A Challenge password []:pkwmaui
An optional company name []:WIR-MACHEN-DAS
Certificate is to be certified until Aug 14 12:52:31 2023 GMT (3650 days)
Sign the certificate? [y/n]:y
1 out of 1 certificate requests certified, commit? [y/n]y
```

Diffie-Hellmann-Parameter erstellen:

```
./build-dh
```

```
Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
........................+..........+............................
+...................................+..........++*++*++*
```

Client-Zertifikat generieren

```
./build-key client1
```

```
Country Name (2 letter code) [DE]:DE
State or Province Name (full name) [NRW]:NRW
Locality Name (eg, city) [Duesseldorf]:Duesseldorf
Organization Name (eg, company) [WIR-MACHEN-DAS]: WIR-MACHEN-DAS
Organizational Unit Name (eg, section) [changeme]:.
Common Name (eg, your name or your server's hostname) [client1]:client1
Name [client1]:client1
Email Address [noc@wir-machen-d.as]:noc@wir-machen-d.as
Please enter the following 'extra' attributes
to be sent with your certificate request
A Challenge password []:lkwkurva
An optional company name []:WIR-MACHEN-DAS
Certificate is to be certified until Aug 14 12:52:31 2023 GMT (3650 days)
Sign the certificate? [y/n]:y
1 out of 1 certificate requests certified, commit? [y/n]y
```

openVPN-Konfigurationsdatei anpassen:

```
vi /etc/openvpn/server.conf
```

```
# IP Adresse der Schnittstelle, an das der Server "binden" soll.
# Auskommentieren, falls an alle gebunden werden sollen.
# VPN-Server1-Debian: 212.72.180.242
local 195.158.140.91

# Port auf dem der Server "hören" soll.
port 1194

# Zu verwendendes Protokoll
proto tcp

# Erzeuge das Tunnel-Device (IP Tunnel)
dev tun
```

```
# Pfade zu den Zertifikaten
# VPN-Server1-Debian: ca.crt, VPN-Server1-Debian.crt, VPN-Server1-Debian.crt
ca /etc/ssl/keys/ca.crt
cert /etc/ssl/keys/VPN-Server2-Debian.crt
key /etc/ssl/keys/VPN-Server2-Debian.key  # Diese Datei muss geheim bleiben

# Diffie-Hellman-Schlüssel
dh /etc/ssl/keys/dh2048.pem

# Subnetz, welches der Server erzeugen soll
# Darf keines der vorhandenen privaten Netzwerke sein!
# VPN-Server1-Debian: 10.11.0.0
server 10.12.0.0 255.255.255.0

# In dieser Datei speichert der Server die Client-IPs
ifconfig-pool-persist ipp.txt

# Gebe dem Client Routen-Informationen mit, damit dieser das private Subnetz findet.
# VPN-Server1: 10.11.0.0
push "route 10.12.0.0 255.255.255.0"

# Damit wird erzwungen, dass am Client alle Anfragen (DNS, Browser) über den Tunnel laufen.
push "redirect-gateway def1 bypass-dhcp“

# Hier werden die Adressen von openDNS mitgeschickt, damit
# die Namensauflösung ebenfalls durch den Tunnel geschieht.
push "dhcp-option DNS 213.73.89.122"

# Pinge alle 10 Sekunden um Verbindung aufrecht zu erhalten.
keepalive 10 120

# Schnelle und sichere Blowfish-Verschlüsselung.
cipher BF-CBC

# Kompression. Wenn angegeben, müssen auch alle Clients dies unterstützen!
comp-lzo

# Bindet laufenden openVPN Prozess an Konto ohne spezielle Rechte. Benutzer und Gruppe müssen existieren!
user nouser
group nogroup
```

```
# Verbindung immer statisch aufbauen.
persist-key
persist-tun

# Optionen fürs Logging
status openvpn-status.log

# "Verbose"-Level: Logge nur Fehler
verb 3
```

3.3 End-to-Site-Verbindung einrichten

Die IP-Adresse des Servers und die erstellten Zertifikate: *ca.crt, client1.crt und client1.key* werden im Netzwerkmanager eingetragen.
Anschließend wird die Verbindung zum Internet aufgebaut und der VPN-Adapter aktiviert. Die Verbindung zum VPN-Server geschiet automatisch.

Screenshots siehe Anhang.

3.4 Site-to-Site-Verbindung einrichten

VPN-Server2-Debian

VPN-Schlüssel für beide Server generieren:

```
openvpn --genkey --secret /etc/ssl/keys/vpn.key
```

VPN-Schlüssel auf VPN-Server1-Debian kopieren:

```
scp /etc/ssl/keys/vpn.key vpn@212.72.180.242:/home/vpn
```

Site-to-Site-Konfigurationsdatei erstellen:

```
vi /etc/openvpn/site2site.conf
```

```
# IP-Adresse des Servers "VPN-Server1-Debian"
remote 212.72.180.242

# Aktzeptiere ankommende IP-Adressen
float

# Port auf dem der Server "hören" soll.
port 8000

# Erzeuge das Tunnel-Device (IP Tunnel)
dev tun

# Peer-to-Peer-Verbindung
# VPN-Server2-Debian: 10.0.0.2 → ← VPN-Server1-Debian: 10.0.0.1
ifconfig 10.0.0.2 10.0.0.1

# Verbindung immer dauerhaft aufbauen.
persist-tun
persist-local-ip

# Kompression. Wenn angegeben, müssen auch alle Clients dies unterstützen!
comp-lzo

# Pinge alle 15 Sekunden um Verbindung aufrecht zu erhalten.
ping 15

# Pfad zu dem VPN-Schlüssel
secret /etc/ssl/keys/vpn.key

# Route zu der internen IP-Adresse von "VPN-Server1-Debian"
route 192.168.11.0 255.255.255.0

# Bindet laufenden openVPN Prozess an Konto ohne spezielle Rechte. Benutzer und Gruppe müssen existieren!
user nobody
group nogroup
```

```
# "Verbose"-Level: Logge nur Fehler
verb 3
```

openVPN mitteilen, dass alle Konfigurationsdateien in */etc/openvpn* geladen werden:

```
vi /etc/default/openvpn
     AUTOSTART=”all”
```

Daemon neu starten:

```
/etc/init.d/openvpn restart
```

VPN-Server1-Debian

VPN-Schlüssel nach */etc/ssl/keys* verschieben:

```
mv /home/vpn/vpn.key /etc/ssl/keys/
```

Site-to-Site-Konfigurationsdatei erstellen:

```
vi /etc/openvpn/site2site.conf
```

```
# IP-Adresse des Servers “VPN-Server2-Debian”
remote 195.158.140.91

# Aktzeptiere ankommende IP-Adressen
float

# Port auf dem der Server “hören” soll.
port 8000

# Erzeuge das Tunnel-Device (IP Tunnel)
dev tun

# Peer-to-Peer-Verbindung
# VPN-Server1-Debian: 10.0.0.1 → ← VPN-Server2-Debian: 10.0.0.2
ifconfig 10.0.0.1 10.0.0.2

# Verbindung immer dauerhaft aufbauen.
persist-tun
persist-local-ip
```

 # Kompression. Wenn angegeben, müssen auch alle Clients dies unterstützen!
comp-lzo

 # Pinge alle 15 Sekunden um Verbindung aufrecht zu erhalten.
ping 15

 # Pfad zu dem VPN-Schlüssel
secret /etc/ssl/keys/vpn.key

 # Route zu der internen IP-Adresse
route 192.168.12.0 255.255.255.0

 # Bindet laufenden openVPN Prozess an Konto ohne spezielle Rechte. Benutzer und Gruppe müssen existieren!
user nobody
group nogroup

 # "Verbose"-Level: Logge nur Fehler
verb 3

openVPN mitteilen, dass alle Konfigurationsdateien in *./etc/openvpn* geladen werden:

```
vi /etc/default/openvpn
    AUTOSTART=”all”
```

Daemon neu starten:

```
/etc/init.d/openvpn restart
```

4 Performance-Betrachtung

4.1 Site-to-Site

VPN-Server1-Debian

nload für Netzwerküberwachung installieren:

```
apt-get install nload
```

nload starten und Einheiten als KByte/s anzeigen lassen:

```
nload -u K
```

sshfs installieren, um Festplatten über das Netzwerk einzubinden:

```
apt-get install sshfs
```

***tmp*-Ordner von VPN-Server2-Debian in */mnt* einbinden:**

```
sshfs vpn@192.168.12.1:/tmp /mnt/
```

VPN-Server2-Debian

iotop installieren, um Lese- und Schreibgeschwindigkeit zu messen:

```
apt-get install iotop
```

iotop starten:

```
iotop
```

htop installieren, um allgemeine Systeminformationen auszulesen:

```
apt-get install htop
```

htop starten:

```
htop
```

VPN-Server1-Debian

Image-Datei mit “dd” auf der eingebundenen Festplatte erstellen:

```
dd if=/dev/zero of=/mnt/leistung.img
```

Screenshot siehe Anhang.

4.2 End-to-Side

VPN-Server2-Debian

iotop und htop starten:

```
iotop
htop
```

Client

nload starten:

```
nload -u K
```

***tmp*-Ordner von VPN-Server2-Debian in */mnt* einbinden:**

```
sshfs vpn@192.168.12.1:/tmp /mnt/
```

Image-Datei mit “dd” auf der eingebundenen Festplatte erstellen:

```
dd if=/dev/zero of=/mnt/leistung.img
```

Screenshot siehe Anhang.

5 VPN-Verbindung für Android-Telefon

“openVPN Connect” App herunterladen.

VPN-Server1-Debian

Verzeichnis zu *~/easy-rsa* wechseln:

```
cd ~/easy-rsa
```

Zertifikat-Variablen dem System bekannt machen:

```
source vars
```

Client-Zertifikat generieren:

```
./build-key clientAndroid
```

```
Country Name (2 letter code) [DE]:DE
State or Province Name (full name) [NRW]:NRW
Locality Name (eg, city) [Duesseldorf]:Duesseldorf
Organization Name (eg, company) [WIR-MACHEN-DAS]: WIR-MACHEN-DAS
Organizational Unit Name (eg, section) [changeme]:.
Common Name (eg, your name or your server's hostname) [client1]:client1
Name [client1]:client1
Email Address [noc@wir-machen-d.as]:noc@wir-machen-d.as
Please enter the following 'extra' attributes
to be sent with your certificate request
A Challenge password []:lkwkurva
An optional company name []:WIR-MACHEN-DAS
Certificate is to be certified until Aug 14 12:52:31 2023 GMT (3650 days)
Sign the certificate? [y/n]:y
1 out of 1 certificate requests certified, commit? [y/n]y
```

Verzeichnis zu *keys* wechseln:

```
cd keys
```

Generierte Schlüssel und Zertifikate in PKCS#12-Format konvertieren:

```
openssl pkcs12 -export -in clientAndroid.crt -inkey
clientAndroid.key -certfile ca.crt -name clientAndroid -out
clientAndroid.p12
```

openvpn-install-exe-Datei herunterladen und *client.ovpn* extrahieren:

http://openvpn.net/index.php/open-source/downloads.html

```
7z e openvpn-install*.exe
```

client.ovpn anpassen:

```
vi client.ovpn
```

```
# IP-Adresse und Port des Servers "VPN-Server1-Debian"
212.72.180.242 1194

# Pfade zu den Zertifikaten
ca ca.key
cert clientAndroid.crt
key clientAndroid.key
```

client.ovpn und *clientAndroid.p12* auf das Smartphone über USB kopieren.

Android: openVPN Connect
client.ovpn importieren:

```
Import → Import Profile from SD card
client.ovpn auswählen
```

clientAndroid.p12 importieren:

```
Import → Import PKCS#12 from SD card
clientAndroid.p12 auswählen
```

6 Quellenverzeichnis

- https://www.df.eu/de/service/df-faq/cloudserver/anleitungen/openvpn-server-installieren-debian-ubuntu/
- http://wiki.nefarius.at/linux/mit_openvpn_ins_internet#weblinks
- https://www.youtube.com/watch?v=ViCeYVcKwms
- https://forums.openvpn.net/topic14629.html
- https://blog.rotzoll.net/2014/03/openvpn-auf-android-4-4-x-kitkat-einrichten/
- http://openvpn.net/index.php/open-source/downloads.html

7 Anhang

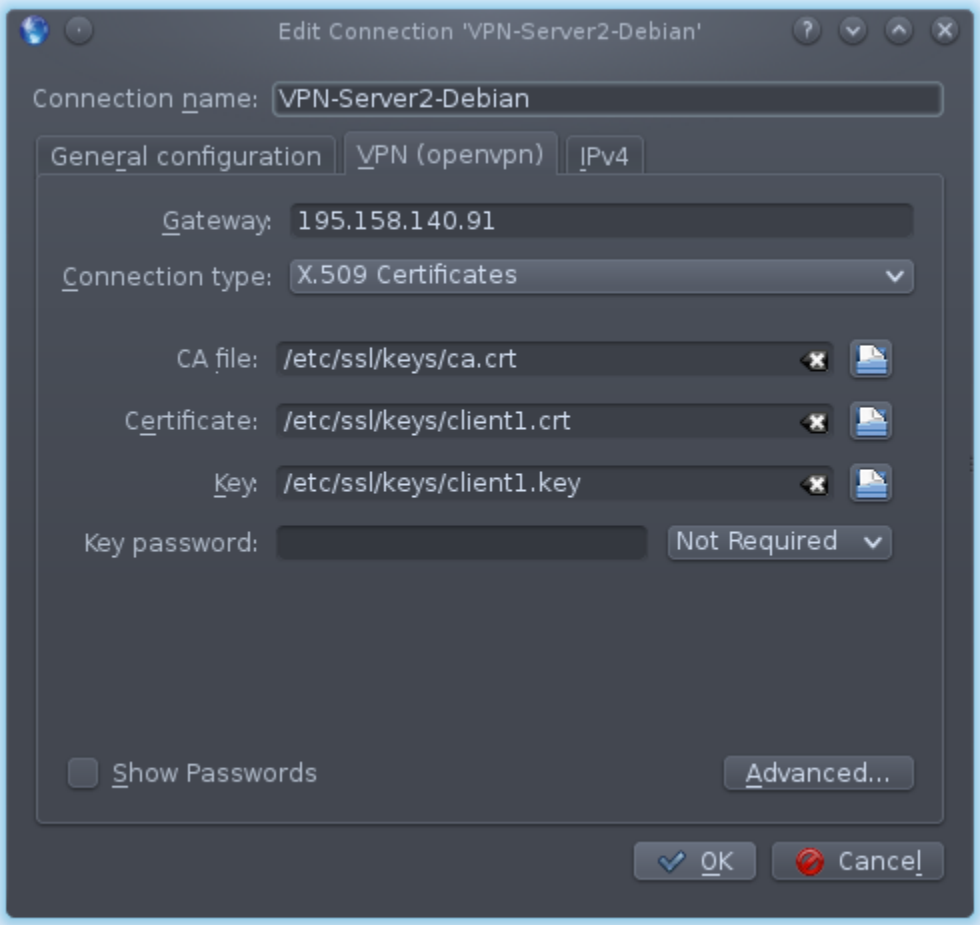

Illustration 1: Punkt 3: VPN-Verbindung mit Zertifikaten

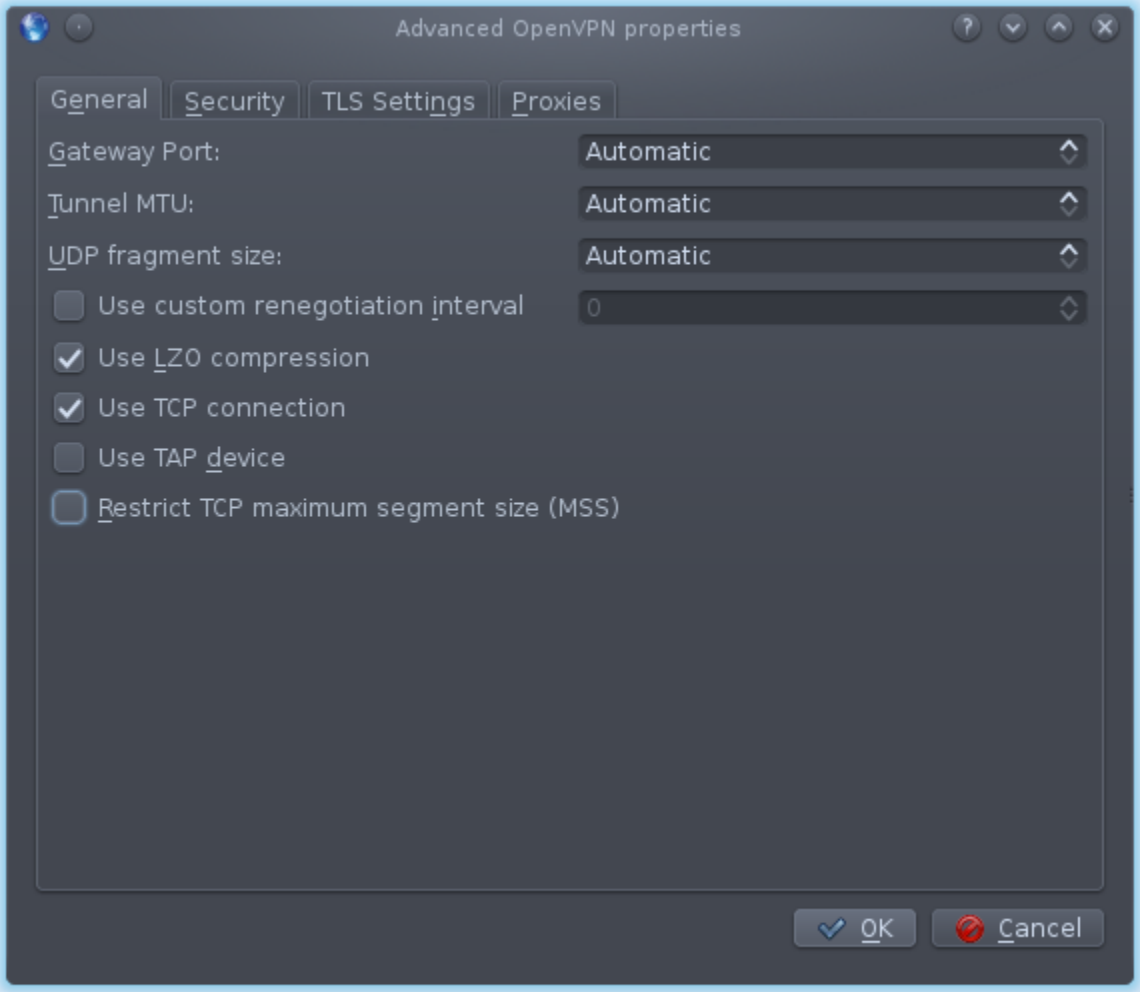

Illustration 2: Punkt 3: Advanced: LZO-Komprimierung und TCP-Verbindung aktivieren

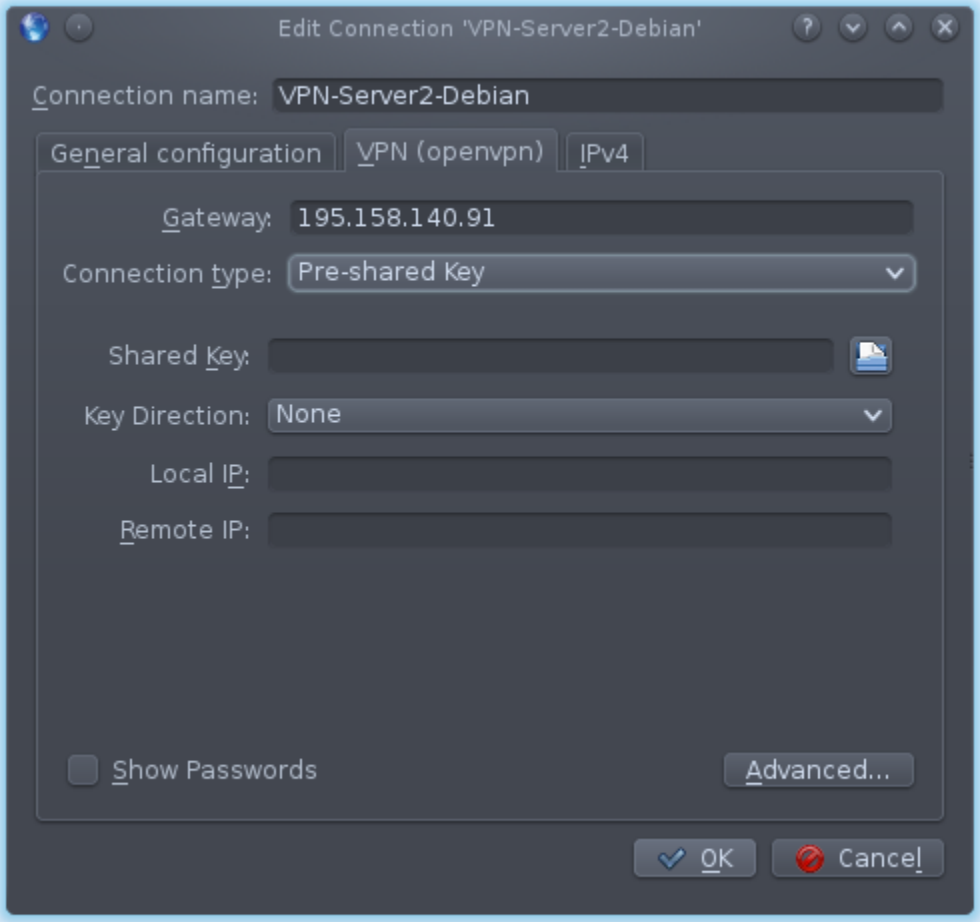

Illustration 3: Punkt 3: VPN-Verbindung mit Pre-shared Key

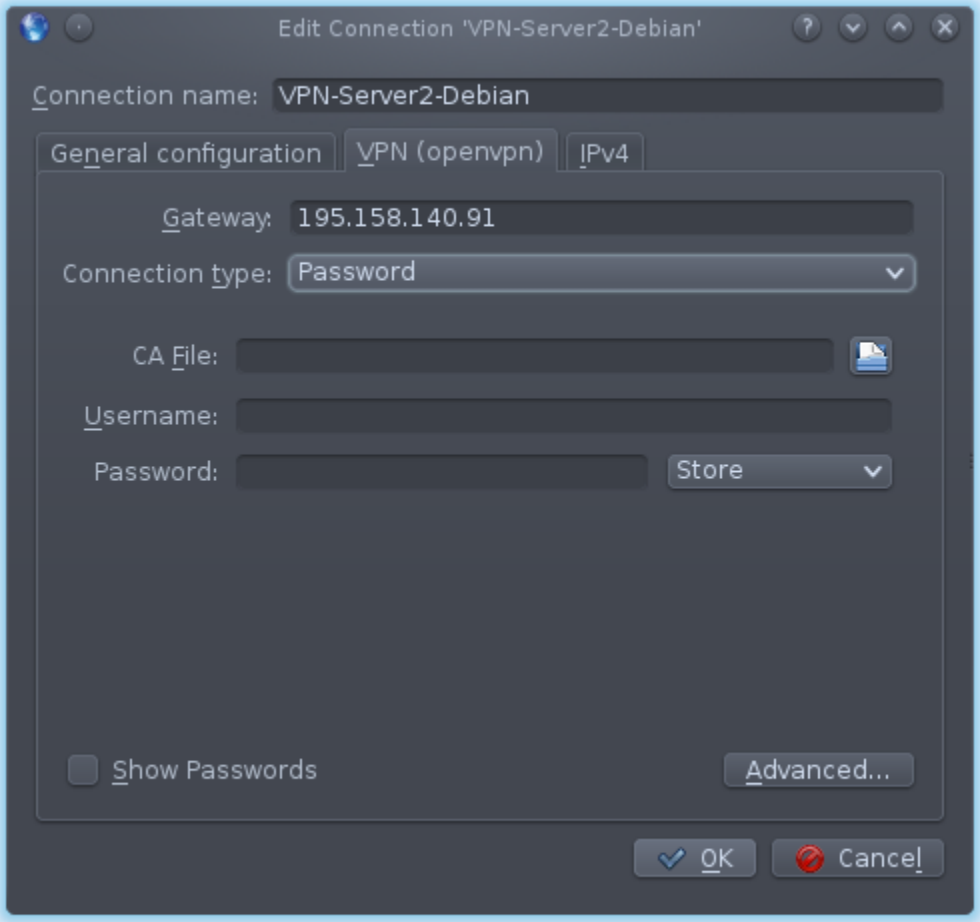

Illustration 4: Punkt 3: VPN-Verbindung mit Benutzername und Passwort

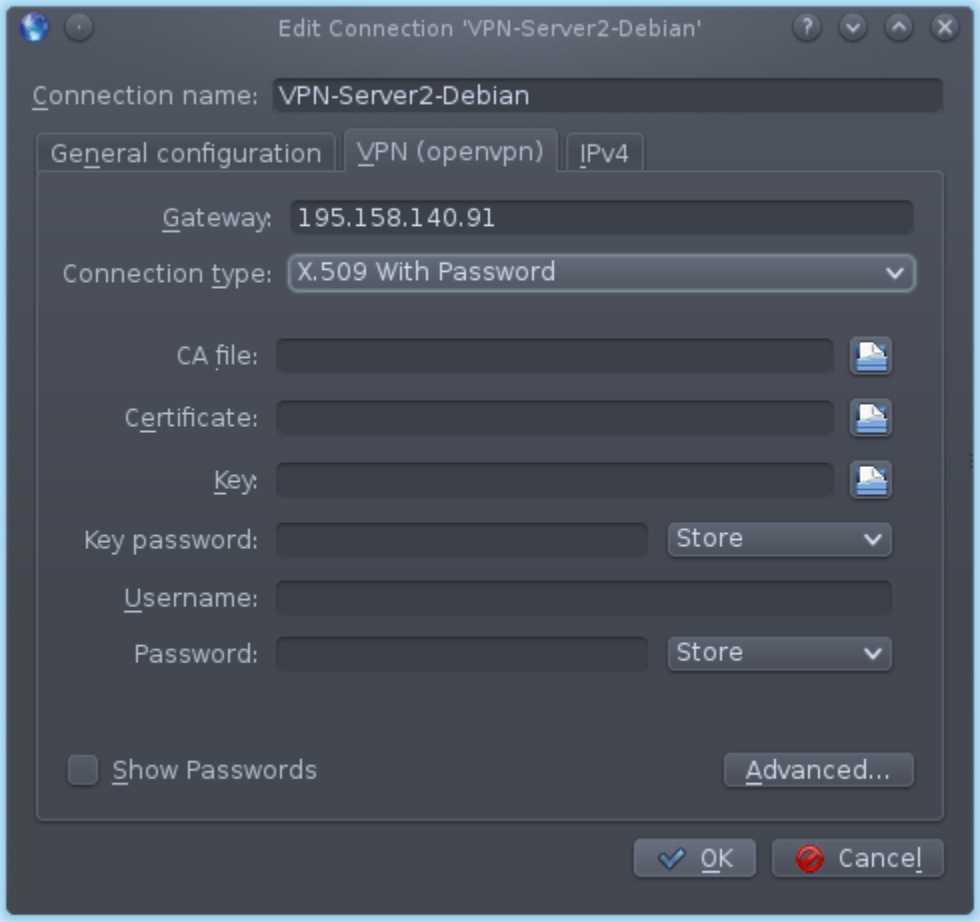

Illustration 5: Punkt 3: VPN-Verbindung mit Zertifikaten, Benutzername und Passwort

Illustration 6: Punkt 5.1: Site-to-Site-Leistungstest mit "nload", "iotop" und "htop"

Illustration 7: Punkt 5.2: End-to-Site-Leistungstest mit "nload", "iotop" und "htop"

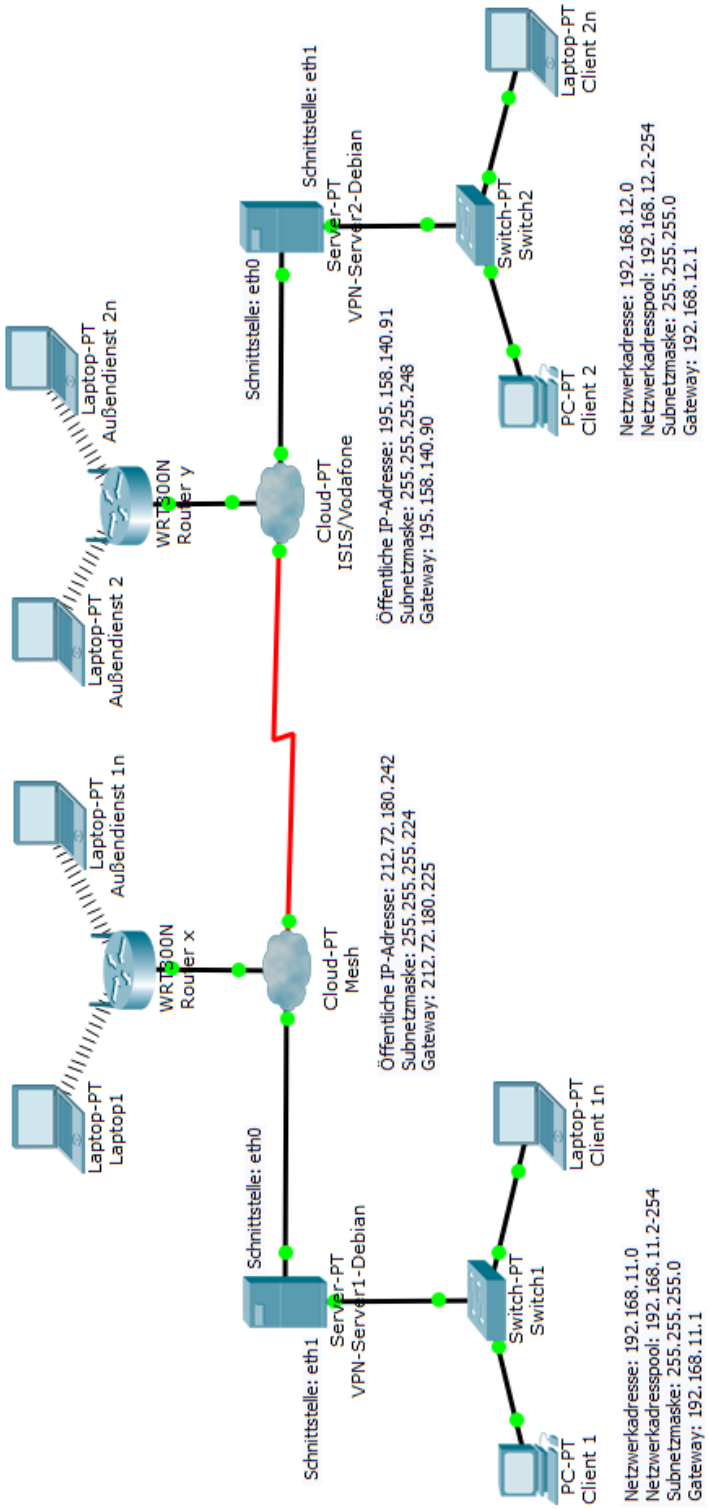

Illustration 8: Logischer Netzwerkstrukturplan