

VPN-Laborübung Vertiefung

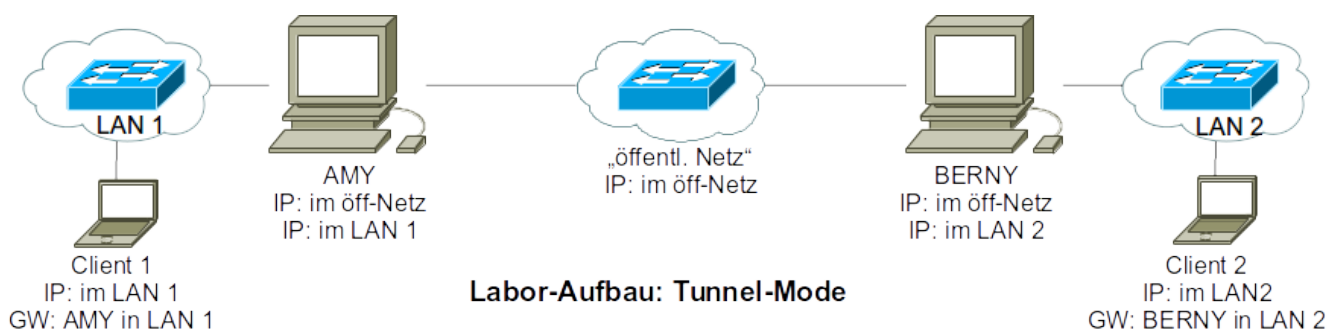
Einleitung und Übersicht

Als Anwendung soll ein Site-to-Site VPN aufgebaut werden. Hierzu werden folgende Elemente benötigt:

- zwei VMs mit dem **MikroTik-Router als VPN-Gateways** auf jeweils einem Labor-PC installiert werden (public/private networks)
- zwei **Labor-PCs als Clients** (private networks), die als LAN-Clients konfiguriert werden.
- ein-zwei Switch ggf. ein Hub zum Mitschneiden der VPN-Verbindung ¹⁾

Insgesamt werden also 4 Labor-PCs und 1-2 Switche benötigt.

Das folgende Bild zeigt den prinzipiellen Aufbau.



Die Verbindung soll als Tunnel mit automatischer Schlüsselaushandlung (IKE/ISAKMP) aufgebaut werden. Als VPN-Gateway wird der MikroTik-Server eingesetzt, indem dieser auf zwei PCs jeweils als virtuelle Maschine angelegt wird. Als Clients werden zwei normale PCs verwendet.

Vorbereitung für die Inbetriebnahme der MikroTik-Router

HINWEIS: Wenn mit virtuellen Maschinen gearbeitet wird, dann sind die **notwendigen Anpassungen** auf jeden Fall zu berücksichtigen. Ansonsten wird die Anwendung u.U. nicht funktionieren.

Das **x86/CD-Image** für die Installation lässt sich **hier die aktuellste Version** oder ältere Version **hier** runterladen. Es sollte das **Current-CD-Image** für **x86** verwendet werden. Die Dateinamen folgen dem Schema **mikrotik-M.SV.B.iso** wobei M.SV.B²⁾ die Version angibt z.B. **mikrotik-6.40.5.iso**.

MikroTik-VM Installation

Der MikroTik-Router wird als ISO-Image in eine neue VM³⁾ eingebunden. Die Maschine sollte mit den folgenden Parametern problemlos laufen:

- Linux → Other Linux
- 1024 MB RAM
- 2 GB Festplatte (dynamisch)
- 2 Netzwerkschnittstellen; eine als öffentliche (public) und eine als lokale (private)

Bei der Installation wird man nach Paketen gefragt. Folgende Pakete sollten ausgewählt werden:

- system (default aktiviert)
- security (IPsec Funktionen)

Die Installation wird durch `i`⁴⁾ gestartet. Anschließend muss man nochmals bestätigen, dass der Vorgang alles löscht. Die Software geht von einer englischen Tastatur aus, das bedeutet es muss z eingegeben werden, um die Installation zu starten.

Nach der erfolgreichen Installation wird das System neugestartet. **ACHTUNG:** Das CDROM-Image muss entfernt werden, da ansonsten immer von diesem gestartet wird. Dies kann über das CD-ROM-Symbol am unteren Fensterrand der VM durchgeführt werden. Das Auswerfen des Image kann erzwungen werden, sobald der Router nach einem Neustart fragt.

Standard-Zugangsdaten:

Benutzer:	admin
Passwort:	KEIN PASSWORT ⁵⁾

MikroTik-VM Grundeinrichtung

Zuerst noch ein paar Tipps, die das Leben mit dem MikroTik-Router erleichtern können.

HINWEIS: Das vorangestellte Zeichen `/`⁶⁾ bedeutet, dass der nachfolgende Befehl auf der höchsten Konfigurationsebene ausgeführt wird. Man kann sich die Befehlsstruktur wie ein Dateisystem vorstellen wobei `/` wie unter Linux üblich die Wurzel (root) darstellt. Mit `..`⁷⁾ gelangt man eine Befehlsebene höher.

TIPP TAB-Taste „Dein Freund“: Der MikroTik-Router unterstützt auf der Command-Line (CLI) wie viele Netzelemente die Autovervollständigung per TAB-Taste. D.h. wenn man den Befehl nicht genau kennt, kann man durch TAB die nächsten Optionen anzeigen lassen. Hat man noch keinen Befehl eingegeben, so werden alle auf aktuellen Ebene möglichen Befehle angezeigt.

TIPP Befehle abkürzen: Für die ganz Faulen → Wird ein Befehl durch die ersten Buchstaben eindeutig erkannt, dann verändert sich die Farbe der Schrift zu türkis (Befehl) oder lila (Parameter). Man kann sich dann das vervollständigen ersparen.

TIPP englisches Tastatur-Layout: Leider unterstützt der MikroTik nur ein englisches Tastatur-Layout. Hier die häufigsten Zeichen und wo sie auf einer deutschen Tastatur im englischen Layout zu finden sind:

gewünschtes Zeichen	deutsche Tastatur
/	-
-	ß
?	_ also SHIFT+ -
=	' (links neben BACKSPACE)

gewünschtes Zeichen	deutsche Tastatur
y	z
z	y
*	SHIFT+8
:	ö

Damit die IP-Adressen auf den richtigen Schnittstellen eingerichtet werden, muss zunächst die Zuordnung der physikalischen Schnittstellen (eth1, eth2) des PCs mit den internen Schnittstellen des MikroTik-Router (ether1, ether2) notiert werden. Gegebenenfalls muss hier die Zuordnung über VirtualBox angepasst werden.

Die MAC-Adressen lassen sich mit dem folgenden Befehl anzeigen:

```
/interface ethernet print
```

Einrichten der IP-Adresse des MikroTik über die Konsole (die **IP-Adresse/Schnittstelle sind anzupassen**)

```
/ip address add address=10.0.0.1/8 interface=ether1
```

```
/ip address add address=80.0.0.1/8 interface=ether2
```

Mit dem folgenden Befehl lässt sich die IP-Adresse überprüfen:

```
/ip address print
```

Sollten noch weitere IP-Adressen (insbesondere die Default Adresse) aufgelistet werden, so sollten diese durch den folgenden Befehl gelöscht werden:

```
/ip address remove numbers=0
```

VPN-Konfiguration der VPN-Gateways

Nach dem die MikroTik-Router per IP-Adressen erreichbar sind, können diese entweder über das Webfrontend konfiguriert werden oder weiterhin über die CLI. Im Firefox muss der Proxy unter Bearbeiten → Einstellungen → Erweitert → Netzwerk → Verbindungen → Einstellungen deaktiviert werden, damit das Webfrontend über den Browser erreichbar wird. Das Webfrontend kann dann über den Browser unter der oben konfigurierten IP erreicht werden. Hier ist das Handbuch zum Webfrontend zu finden: [MikroTik-Webfig-Handbuch](#)

Im Folgenden wird nur die CLI-Konfiguration beschrieben. Im Webfrontend finden sich die Parameter unter IP, IP→IPsec, Firewall bzw. Routes.

Folgende Punkte sind für die Site-to-Site Verbindung auf dem MikroTik-Router zu konfigurieren:

- **IP-Adressen** (private/public)
- **Route** ins remote-private-Netz mit dem zweiten MikroTik als Gateway
- **IPsec-Proposal** (Authentication-Algorithmus, Encryption-Algorithmus, Name)
- **IPsec-SAs** wird im MikroTik als peer bezeichnet (DH-Group, Encryption-Algorithmus, Secret)

- **IPsec-Policy** (private Netze und VPN-Gateways, Tunnel-Mode)
- **NAT aktivieren der Firewall** für die Verbindung local-private zu remote-private

In der Dokumentation des MikroTik-Servers befindet sich eine **Beispiel-Konfiguration** für ein Site-to-Site System. Hier wird allerdings davon ausgegangen, dass der private Adressbereich 192.168.80.0 und 192.168.90.0 für das öffentliche Netz eingesetzt wird und so in zwei Netzen dargestellt wird. Um den Laboraufbau möglichst schlank zu halten soll mit nur EINEM öffentlichen Netz gearbeitet werden z.B. 80.0.0.0/8. Die beiden lokalen Netze LAN1 und LAN2 könnten 10.0.0.0/8 (Amy) und 20.0.0.0/8 (Berny) lauten.

Die folgende Tabelle zeigt ein mögliches Adressschema für den Laboraufbau:

Netzelement/Bereich	Parameter	Wert	Bedeutung
locale-privat	IP-Netz	10.0.0.0/8	privates LAN auf Amys-Seite
remote-privat	IP-Netz	20.0.0.0/8	privates LAN auf Bernys-Seite
public	IP-Netz	80.0.0.0/8	öffentliches Netz für die Verbindung der VPN-Gateways
public-Amy	IP-Adresse	80.0.0.1/8	öffentliche IP-Adresse von Amy
locale-privat-Amy	IP-Adresse	10.0.0.1/8	private IP-Adresse von Amy (dient als Gateway für LAN)
public-Berny	IP-Adresse	80.0.0.2/8	öffentliche IP-Adresse von Berny
remote-privat-Berny	IP-Adresse	20.0.0.1/8	private IP-Adresse von Amy (dient als Gateway für LAN)

HINWEIS: remote und privat ist hier bezogen auf Amy. Für Berny sind diese Beziehung jeweils entgegengesetzt. Die nächsten Abschnitte erklären die notwendigen Konfigurationen bezogen auf das vorangegangene Schema.

Route in lokale Netze

Damit die VPN-Gateways die jeweils gegenüberliegenden LAN-Netze (remote-private) kennen, müssen diese per Routing-Eintrag bekannt gegeben werden.

Der Befehl für das Einrichten der Route in das remote-private-Zielnetz ist wie folgt aufgebaut:

```
/ip route add distance=1 dst-address=20.0.0.0/8 gateway=80.0.0.2
```

Parameter	Bedeutung	Wert	Bemerkung
distance	Metrik	1	Es befindet sich nur ein Router auf dem Weg zum Zielnetz.
dst-address	remote-private-Zielnetz	20.0.0.0/8	Zielnetz hinter remote-VPN-Gateway
gateway	VPN-Gateway Berny	80.0.0.2	Partner-VPN-Gateway; Hier Berny aus Sicht von Amy

Zur Überprüfung der Konfiguration kann folgender Befehl verwendet werden:

```
/ip route print
```

Firewall-Regel für NAT zwischen privaten Netzen

Damit die lokalen Adressen auf die öffentlichen Adressen der Router umgesetzt werden, wird der NAT⁸⁾-Mechanismus benötigt. Dies geschieht bei den MikroTik-Router per Firewall-Regel.

Der Befehl für das Einrichten der notwendigen Firewall-Regel ist wie folgt aufgebaut:

```
/ip firewall nat add action=accept chain=srcnat dst-address=20.0.0.0/8 src-address=10.0.0.0/8
```

Parameter	Bedeutung	Wert	Bemerkung
action	Aktion der Firewall	accept	Pakete sollen passieren können.
chain	Regelkette	srcnat	Aktion soll in Quell-NAT-Regelkette eingefügt werden.
dst-address	Zielnetz	20.0.0.0/8	remote-privat-Netz (Berny-Seite)
src-address	Quellnetz	10.0.0.0/8	locale-privat-Netz (Amy-Seite)

Zur Überprüfung der Konfiguration kann folgender Befehl verwendet werden:

```
/ip firewall nat print
```

Proposal für initialen Verbindungsaufbau

Das Proposal wird für die erste Kontaktaufnahme benötigt. Hier werden die Verschlüsselungsmechanismen festgelegt, um im Anschluss die weiteren Daten zur Authentifizierung verschlüsselt zu übertragen.

Der Befehl für das Einrichten des Proposal ist wie folgt aufgebaut:

```
/ip ipsec proposal add auth-algorithms=sha256 enc-algorithms=aes-256-cbc name=labor1
```

Parameter	Bedeutung	Wert	Bemerkung
auth-algorithms	Authentifizierungsalgorithmus	sha256	Authentifizierung für den ersten Kontakt
enc-algorithms	Verschlüsselungsalgorithmus	aes-256-cbc	Authentifizierung für den ersten Kontakt
name	Verwaltungsname	labor1	Beliebiger Name, der die Verbindung charakterisiert.

Zur Überprüfung der Konfiguration kann folgender Befehl verwendet werden:

```
/ip ipsec proposal print
```

Security-Association (SA) für die VPN-Gegenstellen

Die SA beschreibt die Parameter der eigentlichen Nutzverbindung. Hier werden die Parameter für die

Schlüsselgenerierung (DH) sowie der Verschlüsselungsalgorithmus festgelegt. Darüberhinaus müssen noch die Gegenstelle sowie ein Passwort zur Authentifizierung festgelegt werden. Dies wird in diesem Beispiel per Preshared-Key realisiert.

Der Befehl für das Einrichten der SA einer Seite ist wie folgt aufgebaut:

```
/ip ipsec peer add address=80.0.0.2/32 dh-group=modp1024 enc-  
algorithm=aes-128 secret=passwort
```

Parameter	Bedeutung	Wert	Bemerkung
address	remote-SA	80.0.0.2/32	Ist bei der remote-Seite entsprechend anzupassen.
dh-group	Diffie-Hellmann Group	modp1024	Parameter für die Schlüsselaushandlung
enc-algorithm	Verschlüsselungsalgorithmus	aes-128	
secret	Preshared-Key zur Authentifizierung	<GEHEIMNIS>	Hier sollte ein eigenes Secret verwendet werden. Auf beiden Gateways identisch.

ACHTUNG: Die IP-Adresse ist als Host-Adresse also mit /32 anzugeben. Hier darf **KEIN** Netz angegeben werden. Hintergrund: Aus Sicherheitsgründen darf nur pro SA nur mit genau **EINER** Gegenstelle eine Verbindung aufgebaut werden.

Zur Überprüfung der Konfiguration kann folgender Befehl verwendet werden:

```
/ip ipsec peer print
```

Security-Policy (SP) für den Nutzdatentransport

Die SPs stellen die Regeln für die Behandlung der eigentlichen Nutzdaten dar. Diese enthalten jeweils Quell-/Ziel-LAN und die weiterleitenden VPN-Gateways.

Der Befehl für das Einrichten der SP ist wie folgt aufgebaut:

```
/ip ipsec policy add dst-address=20.0.0.0/8 sa-dst-address=80.0.0.2 sa-src-  
address=80.0.0.1\  
src-address=10.0.0.0/8 tunnel=yes
```

Parameter	Bedeutung	Wert	Bemerkung
dst-address	remote-privat Netz	20.0.0.0/8	Ist bei der remote-Seite entsprechend anzupassen.
sa-dst-address	remote-SA	80.0.0.2	Ist bei der remote-Seite entsprechend anzupassen.
sa-src-address	locale-SA	80.0.0.1	Ist bei der remote-Seite entsprechend anzupassen.
src-address	locale-privat Netz	10.0.0.0/8	Ist bei der remote-Seite entsprechend anzupassen.
tunnel	IPsec-Modus	yes	Der Tunnel-Modus für Nutzdaten verwenden.

ACHTUNG: Für die Gegenstelle (Beryn als remote-Seite) müssen die Parameter entsprechend angepasst werden.

Zur Überprüfung der Konfiguration kann folgender Befehl verwendet werden:

```
/ip ipsec policy print
```

Zusammenfassung der VPN-Gateway-Konfiguration

Die folgende Datei bildet den Laboraufbau ab und kann als Beispiel für EINE Seite dienen. Es sind allerdings noch Anpassungen an den eigenen Laboraufbau vorzunehmen. Eine entsprechend gespiegelte Konfiguration ist für die Gegenstelle vorzunehmen.

HINWEIS zu den Befehlen: Das Zeichen \⁹⁾ am Ende der Zeilen ist nur in Konfigurationsdateien notwendig und zeigt an, dass der Befehl in der nächsten Zeile fortgesetzt wird. Das Zeichen # am Anfang einer Zeile bedeutet, dass diese Zeile ein Kommentar ist und nicht wirksam ist.

[vpn_gw_amy.rsc](#)

```
# Basis-Konfiguration für Amy
# Prüfen der IP-Adressen
/ip address print

# ACHTUNG: Die nächsten Befehle nur ausführen, falls die IP-Adressen
# der beiden Schnittstellen noch nicht konfiguriert wurden; sollte
# bereits mit der Grundeinrichtung erledigt sein
#/ip address
#add address=80.0.0.1/8 interface=ether1
#add address=10.0.0.1/8 interface=ether2

# Routen ins jeweilige remote-private-Netz bekannt machen; Gateway
# jeweils das VPN-Partner-Gateway
/ip route
add distance=1 dst-address=20.0.0.0/8 gateway=80.0.0.2

# Firewall-Regel erstellen, um NAT für local zu remote Verbindung zu
# aktivieren
/ip firewall nat
add action=accept chain=srcnat dst-address=20.0.0.0/8 src-
address=10.0.0.0/8

# --- IPsec Konfiguration ---
# IPsec-Proposal für Verbindungsaufbau festlegen
/ip ipsec proposal
add auth-algorithms=sha256 enc-algorithms=aes-256-cbc name=labor1

# IPsec-Security SA festlegen
# ACHTUNG: "password" durch eigenes Passwort ersetzen!
/ip ipsec peer
add address=80.0.0.2/32 dh-group=modp1024 enc-algorithm=aes-128
```

```
secret=\
  password

# IPsec-Policy für Nutzdaten festlegen
/ip ipsec policy
add dst-address=20.0.0.0/8 sa-dst-address=80.0.0.2 sa-src-
address=80.0.0.1\
  src-address=10.0.0.0/8 tunnel=yes
```

HINWEIS: Der MikroTik-Router bietet die Möglichkeit über das Webfrontend Dateien auf den Router hochzuladen. Man könnte die Datei `vpn_gw_amy.rsc`-Datei so hochladen und die enthaltene Konfiguration anschließend durch folgenden Befehl aktivieren:

```
import vpn_gw_amy.rsc
```

Leider führen kleinste Syntax-Fehler dazu, dass dies fehlschlägt.

Client Konfiguration

Die Client-PCs benötigen keine aufwendige Konfiguration. Hier müssen lediglich die entsprechenden IP-Adressen in die privaten Netze gesetzt werden und als Gateways jeweils die private IP-Adresse des lokalen VPN-Gateways.

Zur Erinnerung hier die Befehle:

```
# IP-Adresse setzen
sudo ip addr add <IP-ADRESSE>/<PREFIX> dev eth1
# Default-Route setzen
sudo ip route add default via <GW-ADRESSE>
```

<IP-ADRESSE>, <PREFIX> und <GW-ADRESSE> sind selbstverständlich an die eigenen Bedürfnisse anzupassen.

1) Es können theoretisch alle Verbindungen über einen Switch geführt werden, da bis auf die VPN-Verbindung keine logische Kommunikation möglich ist

2) M=main, SV=subversion, B=bugfix-level

3) VirtualBox

4) i: install

5) einfach Return

6) /: Slash

7) .. unter Linux cd .. change directory

8)

NAT: **N**etwork **A**ddress **T**ranslation; Umsetzung von privaten auf öffentliche Adressen

⁹⁾

\: Backslash

From:

<https://www.kopfload.de/> - **kopfload - Lad Dein Hirn auf!**

Permanent link:

https://www.kopfload.de/doku.php?id=lager:oeff_netze:vpn_vertiefung&rev=1535300613

Last update: **2025/11/19 16:13**

