Unterricht im Netzwerklabor mit Routerboards von Mikrotik

Eine preisgünstige, alternative Netzwerkhardware

Michael Dienert

Walther-Rathenau-Gewerbeschule Freiburg

21. Mai 2019

Inhalt

Wer ist die Firma MikroTik und was sind Routerboards

Routing

Paketfilterung auf den RouterBOARDs

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

MikroTik

- *MikroTik* wurde 1996 in Riga, Lettland gegründet.
- 140 Mitarbeiter in Riga
- 1997: Vorstellung der Softwarelösung *RouterOS* auf Standard-PC-Hardware
- *RouterBOARD*-Markenname seit 2002: Beginn der eigenen Hardware-Fertigung.
- RouterBOARDs waren ursprünglich das, was der Name sagt: gehäuselose Platinen mit der Routerhardware. Inzwischen werden diese selbstverständlich auch mit Gehäuse vertrieben.
- Viele RouterBOARDs haben WLAN integriert
- Weitere Produkte: professionelle, drahtlose Übertragungsstrecken, Switches, RouterOS als Softwarelösung

In der Fortbildung verwendete Geräte

In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

In der Fortbildung verwendete Geräte

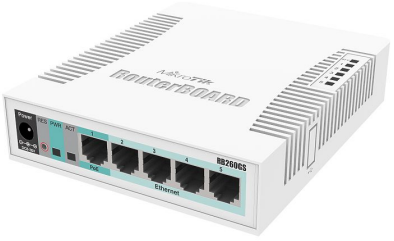
RB951G
WLAN-Router

RB750Gr3

In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

RB750Gr3

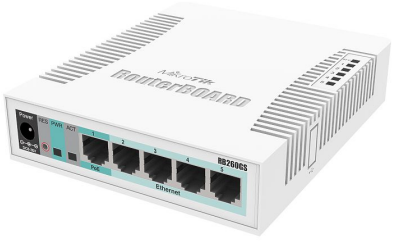


RouterBOARD
260GS

In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

RB750Gr3



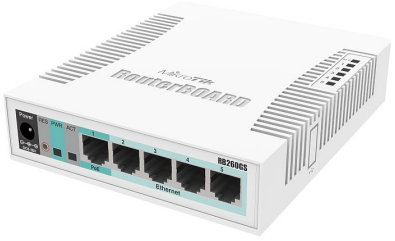
RouterBOARD
260GS

- 802.11b/g/n

In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

RB750Gr3



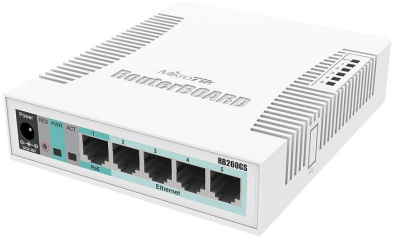
RouterBOARD
260GS

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports

In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

RB750Gr3



RouterBOARD
260GS

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

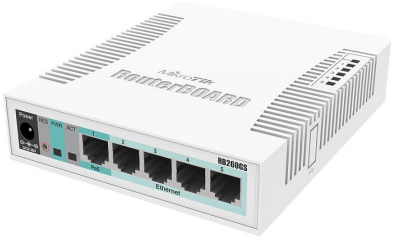
In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

RB750Gr3

- kein WiFi, interner 5-Port-Switch.



RouterBOARD
260GS

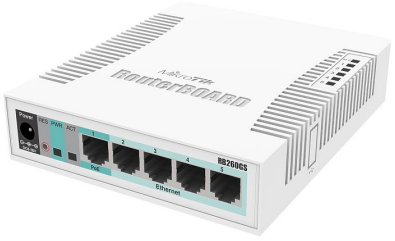
In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

RB750Gr3

- kein WiFi, interner 5-Port-Switch.
- IPsec Hardware-Verschlüsselung.



RouterBOARD
260GS

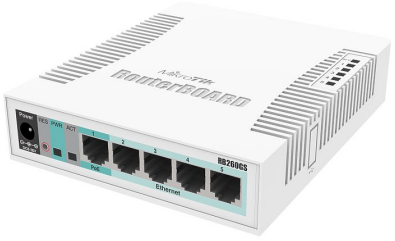
In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

RB750Gr3

- kein WiFi, interner 5-Port-Switch.
- IPsec Hardware-Verschlüsselung.
- ca. EUR 60.-



RouterBOARD
260GS

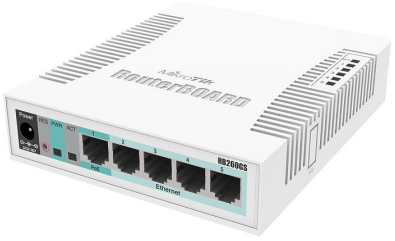
In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

RB750Gr3

- kein WiFi, interner 5-Port-Switch.
- IPsec Hardware-Verschlüsselung.
- ca. EUR 60.-



RouterBOARD
260GS

- 5-Port-Gigabit-Switch.

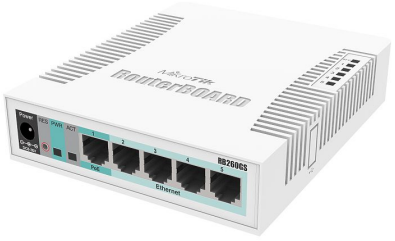
In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

RB750Gr3

- kein WiFi, interner 5-Port-Switch.
- IPsec Hardware-Verschlüsselung.
- ca. EUR 60.-



RouterBOARD
260GS

- 5-Port-Gigabit-Switch.
- SFP-Port!

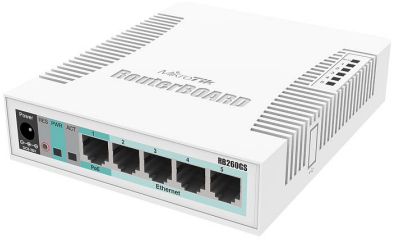
In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

RB750Gr3

- kein WiFi, interner 5-Port-Switch.
- IPsec Hardware-Verschlüsselung.
- ca. EUR 60.-



RouterBOARD
260GS

- 5-Port-Gigabit-Switch.
- SFP-Port!
- ca. EUR 40.-

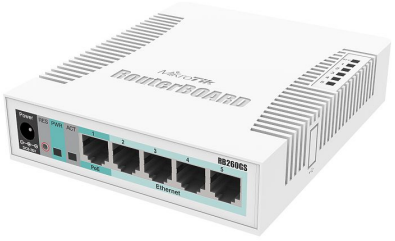
In der Fortbildung verwendete Geräte

RB951G
WLAN-Router

- 802.11b/g/n
- interner Switch mit 5 1000BaseT-Ports
- ca. EUR 80.-

RB750Gr3

- kein WiFi, interner 5-Port-Switch.
- IPsec Hardware-Verschlüsselung.
- ca. EUR 60.-



RouterBOARD
260GS

- 5-Port-Gigabit-Switch.
- SFP-Port!
- ca. EUR 40.-
- leider nur Web-Interface

Aufbau des RouterOS

- Das dem RouterOS zugrunde liegende System ist *Linux*.
- Die Benutzerschnittstelle ähnelt entfernt dem IOS von Cisco.
- Es gibt keine Shell.
- Die tab -Taste oder die ? -Taste helfen immer weiter.
- Die Administrations-Kommandos sind *hierarchisch* angeordnet.
- Es ergibt sich eine *Baumstruktur*.

Aufbau des RouterOS

- Das dem RouterOS zugrunde liegende System ist *Linux*.
- Die Benutzerschnittstelle ähnelt entfernt dem IOS von Cisco.
- Es gibt keine Shell.
- Die tab -Taste oder die ? -Taste helfen immer weiter.
- Die Administrations-Kommandos sind *hierarchisch* angeordnet.
- Es ergibt sich eine *Baumstruktur*.

Aufbau des RouterOS

- Das dem RouterOS zugrunde liegende System ist *Linux*.
- Die Benutzerschnittstelle ähnelt entfernt dem IOS von Cisco.
- Es gibt keine Shell.
- Die tab -Taste oder die ? -Taste helfen immer weiter.
- Die Administrations-Kommandos sind *hierarchisch* angeordnet.
- Es ergibt sich eine *Baumstruktur*.

Aufbau des RouterOS

- Das dem RouterOS zugrunde liegende System ist *Linux*.
- Die Benutzerschnittstelle ähnelt entfernt dem IOS von Cisco.
- Es gibt keine Shell.
- Die tab -Taste oder die ? -Taste helfen immer weiter.
- Die Administrations-Kommandos sind *hierarchisch* angeordnet.
- Es ergibt sich eine *Baumstruktur*.

Aufbau des RouterOS

- Das dem RouterOS zugrunde liegende System ist *Linux*.
- Die Benutzerschnittstelle ähnelt entfernt dem IOS von Cisco.
- Es gibt keine Shell.
- Die tab -Taste oder die ? -Taste helfen immer weiter.
- Die Administrations-Kommandos sind *hierarchisch* angeordnet.
- Es ergibt sich eine *Baumstruktur*.

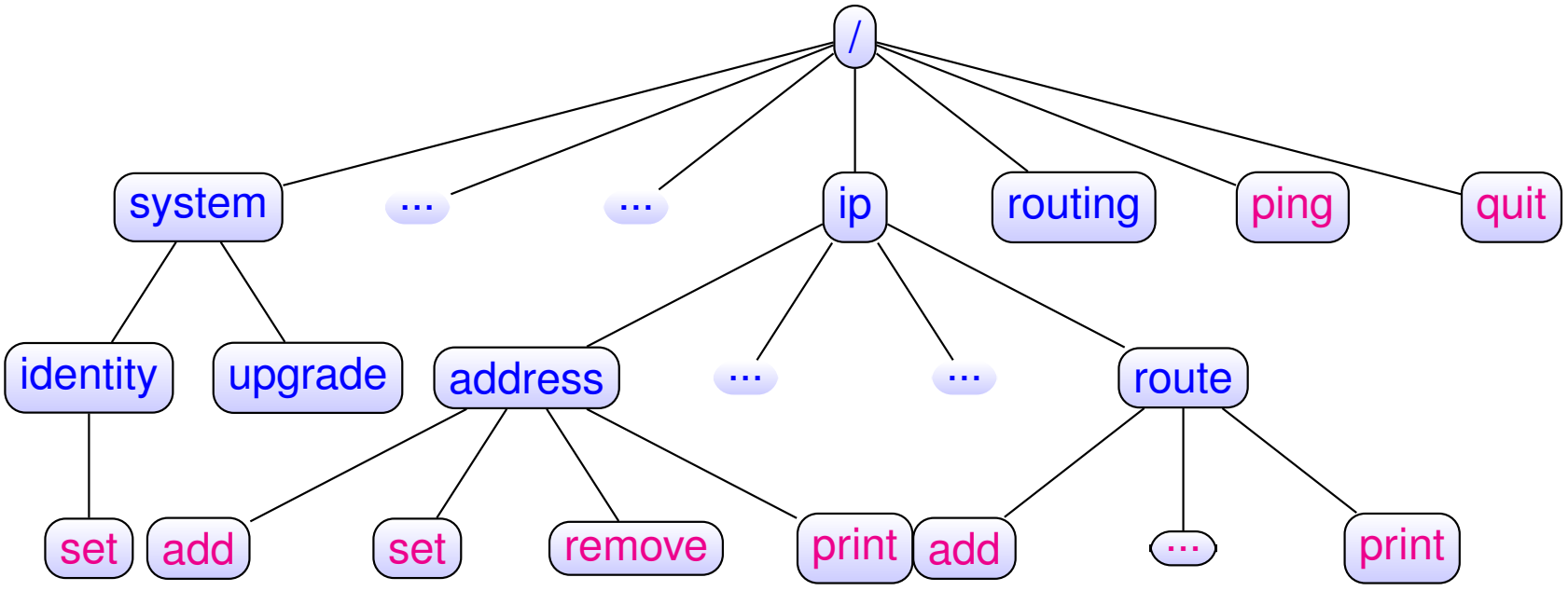
Aufbau des RouterOS

- Das dem RouterOS zugrunde liegende System ist *Linux*.
- Die Benutzerschnittstelle ähnelt entfernt dem IOS von Cisco.
- Es gibt keine Shell.
- Die tab -Taste oder die ? -Taste helfen immer weiter.
- Die Administrations-Kommandos sind *hierarchisch* angeordnet.
- Es ergibt sich eine *Baumstruktur*.

Aufbau des RouterOS

- Das dem RouterOS zugrunde liegende System ist *Linux*.
- Die Benutzerschnittstelle ähnelt entfernt dem IOS von Cisco.
- Es gibt keine Shell.
- Die tab -Taste oder die ? -Taste helfen immer weiter.
- Die Administrations-Kommandos sind *hierarchisch* angeordnet.
- Es ergibt sich eine *Baumstruktur*.

Anordnung der Kommandos in einer Hierarchie



Eingabe der Kommandos

- Die Knoten des Baums stellen den `Kontext` dar, in dem man sich gerade befindet. Abhängig vom Kontext stehen weitere Kommandos bzw. Kontexte zur Verfügung.
- Die Kommandos im Baum müssen über ihren Pfad aufgerufen werden. Der Pfad kann absolut oder relativ angegeben werden. Dabei wird der übergeordnete Kontext durch `..` dargestellt.
- Die Wurzel des Baums wird durch das Zeichen `/` repräsentiert.
- Die einzelnen Pfadelemente werden durch `Leerzeichen` getrennt!

Eingabe der Kommandos

- Die Knoten des Baums stellen den `Kontext` dar, in dem man sich gerade befindet. Abhängig vom Kontext stehen weitere Kommandos bzw. Kontexte zur Verfügung.
- Die Kommandos im Baum müssen über ihren Pfad aufgerufen werden. Der Pfad kann absolut oder relativ angegeben werden. Dabei wird der übergeordnete Kontext durch `..` dargestellt.
- Die Wurzel des Baums wird durch das Zeichen `/` repräsentiert.
- Die einzelnen Pfadelemente werden durch `Leerzeichen` getrennt!

Eingabe der Kommandos

- Die Knoten des Baums stellen den Kontext dar, in dem man sich gerade befindet. Abhängig vom Kontext stehen weitere Kommandos bzw. Kontexte zur Verfügung.
- Die Kommandos im Baum müssen über ihren Pfad aufgerufen werden. Der Pfad kann absolut oder relativ angegeben werden. Dabei wird der übergeordnete Kontext durch .. dargestellt.
- Die Wurzel des Baums wird durch das Zeichen / repräsentiert.
- Die einzelnen Pfadelemente werden durch Leerzeichen getrennt!

Eingabe der Kommandos

- Die Knoten des Baums stellen den Kontext dar, in dem man sich gerade befindet. Abhängig vom Kontext stehen weitere Kommandos bzw. Kontexte zur Verfügung.
- Die Kommandos im Baum müssen über ihren Pfad aufgerufen werden. Der Pfad kann absolut oder relativ angegeben werden. Dabei wird der übergeordnete Kontext durch .. dargestellt.
- Die Wurzel des Baums wird durch das Zeichen / repräsentiert.
- Die einzelnen Pfadelemente werden durch Leerzeichen getrennt!

Eingabe der Kommandos

- Die Knoten des Baums stellen den Kontext dar, in dem man sich gerade befindet. Abhängig vom Kontext stehen weitere Kommandos bzw. Kontexte zur Verfügung.
- Die Kommandos im Baum müssen über ihren Pfad aufgerufen werden. Der Pfad kann absolut oder relativ angegeben werden. Dabei wird der übergeordnete Kontext durch .. dargestellt.
- Die Wurzel des Baums wird durch das Zeichen / repräsentiert.
- Die einzelnen Pfadelemente werden durch Leerzeichen getrennt!

Eingabe der Kommandos

- Unabhängig vom gerade gewählten Kontext, können die Kommandos mit einem absoluten Pfad aufgerufen werden:

```
[admin@R1] /routing> / system identity set name=zzz
```

- Relative Angaben sind auch möglich. Vorwärts im Baum:

```
[admin@zzz] /ip> address print
```

- Rückwärts funktioniert selbstverständlich auch:

```
[admin@R1] > / routing ospf
[admin@R1] /routing ospf> .. .. system identity print
```

Eingabe der Kommandos

- Unabhängig vom gerade gewählten Kontext, können die Kommandos mit einem absoluten Pfad aufgerufen werden:

```
[admin@R1] /routing> / system identity set name=zzz
```

- Relative Angaben sind auch möglich. Vorwärts im Baum:

```
[admin@zzz] /ip> address print
```

- Rückwärts funktioniert selbstverständlich auch:

```
[admin@R1] > / routing ospf
[admin@R1] /routing ospf> .. .. system identity print
```

Eingabe der Kommandos

- Unabhängig vom gerade gewählten Kontext, können die Kommandos mit einem absoluten Pfad aufgerufen werden:

```
[admin@R1] /routing> / system identity set name=zzz
```

- Relative Angaben sind auch möglich. Vorwärts im Baum:

```
[admin@zzz] /ip> address print
```

- Rückwärts funktioniert selbstverständlich auch:

```
[admin@R1] > / routing ospf
[admin@R1] /routing ospf> .. .. system identity print
```

Eingabe der Kommandos

- Unabhängig vom gerade gewählten Kontext, können die Kommandos mit einem absoluten Pfad aufgerufen werden:

```
[admin@R1] /routing> / system identity set name=zzz
```

- Relative Angaben sind auch möglich. Vorwärts im Baum:

```
[admin@zzz] /ip> address print
```

- Rückwärts funktioniert selbstverständlich auch:

```
[admin@R1] > / routing ospf
[admin@R1] /routing ospf> .. .. system identity print
```

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Rücksetzen eines RouterBOARDs

- MikroTik beschreibt mehrere Prozeduren für das Zurücksetzen.
- Bewährte Prozedur:
  - RouterBOARD von Spannungsversorgung trennen.
  - RB hochkant stellen und mit Kugelschreiber Reset-Taste rechts neben der Versorgungsbuchse drücken und halten.
  - DC-Stecker einstecken, Reset gedrückt halten.
  - Sobald LED rechts neben der PWR-LED zu blinken beginnt, eine bis zwei Blink-Perioden abwarten, dann Reset-Taster loslassen.
  - Zeitpunkt des Reset-Taster-Lösens: wartet man zu lange, startet das RouterBOARD den CAPs- (Controlled Access Point system Manager) oder Netinstall-Modus

Aktualisieren eines RouterBOARDs

- Voraussetzung: Verbindung vom RouterBOARD ins Internet.
- Beispiel für Aktualisierung im Schulnetz der WaRa:
  - Router an eth1 oder eth2 des PC anschliessen
  - eth1 mit statischer Adresse 192.168.88.2/24 starten
  - PC mit folgendem Skript zum nat-router machen:

```
#Routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward
#Nat loeschen
iptables -t nat -F
#Nat neu konfigurieren
iptables -A FORWARD -o eth0 -i eth1 -s 192.168.88.0/24 -m conntrack \
         --ctstate NEW -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
#Nat Tabelle auflisten
iptables -t nat -n -L
```

Aktualisieren eines RouterBOARDs

- Voraussetzung: Verbindung vom RouterBOARD ins Internet.
- Beispiel für Aktualisierung im Schulnetz der WaRa:
  - Router an eth1 oder eth2 des PC anschliessen
  - eth1 mit statischer Adresse 192.168.88.2/24 starten
  - PC mit folgendem Skript zum nat-router machen:

```
#Routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward
#Nat loeschen
iptables -t nat -F
#Nat neu konfigurieren
iptables -A FORWARD -o eth0 -i eth1 -s 192.168.88.0/24 -m conntrack \
         --ctstate NEW -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
#Nat Tabelle auflisten
iptables -t nat -n -L
```

Aktualisieren eines RouterBOARDs

- Voraussetzung: Verbindung vom RouterBOARD ins Internet.
- Beispiel für Aktualisierung im Schulnetz der WaRa:
  - Router an eth1 oder eth2 des PC anschliessen
  - eth1 mit statischer Adresse 192.168.88.2/24 starten
  - PC mit folgendem Skript zum nat-router machen:

```
#Routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward
#Nat loeschen
iptables -t nat -F
#Nat neu konfigurieren
iptables -A FORWARD -o eth0 -i eth1 -s 192.168.88.0/24 -m conntrack \
         --ctstate NEW -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
#Nat Tabelle auflisten
iptables -t nat -n -L
```

Aktualisieren eines RouterBOARDs

- Voraussetzung: Verbindung vom RouterBOARD ins Internet.
- Beispiel für Aktualisierung im Schulnetz der WaRa:
  - Router an eth1 oder eth2 des PC anschliessen
  - eth1 mit statischer Adresse 192.168.88.2/24 starten
  - PC mit folgendem Skript zum nat-router machen:

```
#Routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward
#Nat loeschen
iptables -t nat -F
#Nat neu konfigurieren
iptables -A FORWARD -o eth0 -i eth1 -s 192.168.88.0/24 -m conntrack \
         --ctstate NEW -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
#Nat Tabelle auflisten
iptables -t nat -n -L
```

Aktualisieren eines RouterBOARDs

- Voraussetzung: Verbindung vom RouterBOARD ins Internet.
- Beispiel für Aktualisierung im Schulnetz der WaRa:
  - Router an eth1 oder eth2 des PC anschliessen
  - eth1 mit statischer Adresse 192.168.88.2/24 starten
  - PC mit folgendem Skript zum nat-router machen:

```
#Routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward
#Nat loeschen
iptables -t nat -F
#Nat neu konfigurieren
iptables -A FORWARD -o eth0 -i eth1 -s 192.168.88.0/24 -m conntrack \
         --ctstate NEW -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
#Nat Tabelle auflisten
iptables -t nat -n -L
```

Aktualisieren eines RouterBOARDs

- Voraussetzung: Verbindung vom RouterBOARD ins Internet.
- Beispiel für Aktualisierung im Schulnetz der WaRa:
  - Router an eth1 oder eth2 des PC anschliessen
  - eth1 mit statischer Adresse 192.168.88.2/24 starten
  - PC mit folgendem Skript zum nat-router machen:

```
#Routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward
#Nat loeschen
iptables -t nat -F
#Nat neu konfigurieren
iptables -A FORWARD -o eth0 -i eth1 -s 192.168.88.0/24 -m conntrack \
         --ctstate NEW -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
#Nat Tabelle auflisten
iptables -t nat -n -L
```

Aktualisieren eines RouterBOARDs

- Anschliessend auf dem RouterBOARD das Folgende ausführen:

```
#dns-server auf router setzen (adresse ggfs. anpassen):
/ip dns set servers=129.143.2.1
#default-gw auf router setzen:
/ip route add  dst-address=0.0.0.0/0 gateway=192.168.88.2
/system package update> check-for-updates
/system package update download
/system reboot
/system package print
```

Aktualisieren eines RouterBOARDs

- Anschliessend auf dem RouterBOARD das Folgende ausführen:

```
#dns-server auf router setzen (adresse ggfs. anpassen):
/ip dns set servers=129.143.2.1
#default-gw auf router setzen:
/ip route add  dst-address=0.0.0.0/0 gateway=192.168.88.2
/system package update> check-for-updates
/system package update download
/system reboot
/system package print
```

Kontaktaufnahme mit dem RouterBOARD

- Im Auslieferungszustand haben die RouterBOARDs die feste IP-Adresse 192.168.88.1
- Auf den RouterBOARDs läuft ein DHCP-Server, der Adressen aus dem Pool 192.168.88.0/24 vergibt.
- Die RouterBOARD-Schnittstelle **ether2** wird mit **eth1** oder **eth2** der Schulrechner verbunden.
- root-Rechte erlangen: Terminal-Programm starten, Kommando `su` eingeben, Passwort ist `toor`.

Kontaktaufnahme mit dem RouterBOARD

- Im Auslieferungszustand haben die RouterBOARDs die feste IP-Adresse 192.168.88.1
- Auf den RouterBOARDs läuft ein DHCP-Server, der Adressen aus dem Pool 192.168.88.0/24 vergibt.
- Die RouterBOARD-Schnittstelle **ether2** wird mit **eth1** oder **eth2** der Schulrechner verbunden.
- root-Rechte erlangen: Terminal-Programm starten, Kommando `su` eingeben, Passwort ist `toor`.

Kontaktaufnahme mit dem RouterBOARD

- Im Auslieferungszustand haben die RouterBOARDs die feste IP-Adresse 192.168.88.1
- Auf den RouterBOARDs läuft ein DHCP-Server, der Adressen aus dem Pool 192.168.88.0/24 vergibt.
- Die RouterBOARD-Schnittstelle **ether2** wird mit **eth1** oder **eth2** der Schulrechner verbunden.
- root-Rechte erlangen: Terminal-Programm starten, Kommando `su` eingeben, Passwort ist `toor`.

Kontaktaufnahme mit dem RouterBOARD

- Im Auslieferungszustand haben die RouterBOARDs die feste IP-Adresse 192.168.88.1
- Auf den RouterBOARDs läuft ein DHCP-Server, der Adressen aus dem Pool 192.168.88.0/24 vergibt.
- Die RouterBOARD-Schnittstelle **ether2** wird mit **eth1** oder **eth2** der Schulrechner verbunden.
- root-Rechte erlangen: Terminal-Programm starten, Kommando `su` eingeben, Passwort ist `toor`.

Kontaktaufnahme mit dem RouterBOARD

- Im Auslieferungszustand haben die RouterBOARDs die feste IP-Adresse 192.168.88.1
- Auf den RouterBOARDs läuft ein DHCP-Server, der Adressen aus dem Pool 192.168.88.0/24 vergibt.
- Die RouterBOARD-Schnittstelle **ether2** wird mit **eth1** oder **eth2** der Schulrechner verbunden.
- root-Rechte erlangen: Terminal-Programm starten, Kommando `su` eingeben, Passwort ist `toor`.

Kontaktaufnahme mit dem RouterBOARD

- Editieren Sie die Datei (mit root-Rechten) `/etc/network/interfaces`, so dass **eth1** oder **eth2** eine feste Adresse bekommen:

```
      ...
auto eth1
iface eth1 inet static
  address 192.168.88.2
  netmask 255.255.255.0
```

- PC-Interface neu starten (mit root-Rechten):

```
      ifdown eth1
      ifup eth1
```

Kontaktaufnahme mit dem RouterBOARD

- Editieren Sie die Datei (mit root-Rechten) `/etc/network/interfaces`, so dass **eth1** oder **eth2** eine feste Adresse bekommen:

```
      ...
auto eth1
iface eth1 inet static
  address 192.168.88.2
  netmask 255.255.255.0
```

- PC-Interface neu starten (mit root-Rechten):

```
      ifdown eth1
      ifup eth1
```

Kontaktaufnahme mit dem RouterBOARD

- Editieren Sie die Datei (mit root-Rechten) `/etc/network/interfaces`, so dass **eth1** oder **eth2** eine feste Adresse bekommen:

```
      ...
auto eth1
iface eth1 inet static
  address 192.168.88.2
  netmask 255.255.255.0
```

- PC-Interface neu starten (mit root-Rechten):

```
      ifdown eth1
      ifup eth1
```

Kontaktaufnahme mit dem RouterBOARD

- Je nach Version des RouterOS können Sie sich nun mit `telnet` oder bevozugt `ssh` auf dem Routerboard anmelden. An einem nicht Unix-basierten Rechner, können sie problemlos mit dem Programm putty arbeiten.
- ```
  ssh admin@192.168.88.1
  ```
- Auf dem RouterBOARD stoppen wir den DHCP-Server, damit die Schul-PCs nicht falsche Gateways und Nameserver eintragen:

```
# auflisten der dhcp-pools
[admin@zzz] > /ip dhcp-server print

# gewuenschten pool stoppen
[admin@zzz] > /ip dhcp-server disable 0
```
```

Kontaktaufnahme mit dem RouterBOARD

- Je nach Version des RouterOS können Sie sich nun mit `telnet` oder bevozugt `ssh` auf dem Routerboard anmelden. An einem nicht Unix-basierten Rechner, können sie problemlos mit dem Programm putty arbeiten.

```
ssh admin@192.168.88.1
```

- Auf dem RouterBOARD stoppen wir den DHCP-Server, damit die Schul-PCs nicht falsche Gateways und Nameserver eintragen:

```
# auflisten der dhcp-pools
[admin@zzz] > /ip dhcp-server print

# gewuenschten pool stoppen
[admin@zzz] > /ip dhcp-server disable 0
```

Kontaktaufnahme mit dem RouterBOARD

- Je nach Version des RouterOS können Sie sich nun mit `telnet` oder bevozugt `ssh` auf dem Routerboard anmelden. An einem nicht Unix-basierten Rechner, können sie problemlos mit dem Programm putty arbeiten.
- ```
  ssh admin@192.168.88.1
  ```
- Auf dem RouterBOARD stoppen wir den DHCP-Server, damit die Schul-PCs nicht falsche Gateways und Nameserver eintragen:

```
# auflisten der dhcp-pools
[admin@zzz] > /ip dhcp-server print

# gewuenschten pool stoppen
[admin@zzz] > /ip dhcp-server disable 0
```
```

Kontaktaufnahme mit dem RouterBOARD

- Je nach Version des RouterOS können Sie sich nun mit `telnet` oder bevozugt `ssh` auf dem Routerboard anmelden. An einem nicht Unix-basierten Rechner, können sie problemlos mit dem Programm putty arbeiten.
- ```
  ssh admin@192.168.88.1
  ```
- Auf dem RouterBOARD stoppen wir den DHCP-Server, damit die Schul-PCs nicht falsche Gateways und Nameserver eintragen:

```
# auflisten der dhcp-pools
[admin@zzz] > /ip dhcp-server print

# gewuenschten pool stoppen
[admin@zzz] > /ip dhcp-server disable 0
```
```

Alternative Adressvergabe am Konfigurationsrechner

- Selbstverständlich kann man die PC-Schnittstelle auch als DHCP-Client konfigurieren, so dass sie vom Routerboard eine Adresse bekommt.
  Nachteil: das Routerboard setzt dann auch Standard-Gateway und Nameserver, so dass man evtl. keinen Zugang mehr zum Schulnetz/Internet hat.
- Wenn auf dem PC der Network-Manager läuft, lässt sich die PC-IP-Adresse mit den zugehörigen grafischen Werkzeugen einstellen.
- Falls Sie ein RouterBoard mit WLAN haben (RB951G), kann man sich, z.B. von einem Notebook aus auch darüber anmelden.

Alternative Adressvergabe am Konfigurationsrechner

- Selbstverständlich kann man die PC-Schnittstelle auch als DHCP-Client konfigurieren, so dass sie vom Routerboard eine Adresse bekommt.
  Nachteil: das Routerboard setzt dann auch Standard-Gateway und Nameserver, so dass man evtl. keinen Zugang mehr zum Schulnetz/Internet hat.
- Wenn auf dem PC der Network-Manager läuft, lässt sich die PC-IP-Adresse mit den zugehörigen grafischen Werkzeugen einstellen.
- Falls Sie ein RouterBoard mit WLAN haben (RB951G), kann man sich, z.B. von einem Notebook aus auch darüber anmelden.

Alternative Adressvergabe am Konfigurationsrechner

- Selbstverständlich kann man die PC-Schnittstelle auch als DHCP-Client konfigurieren, so dass sie vom Routerboard eine Adresse bekommt.
  Nachteil: das Routerboard setzt dann auch Standard-Gateway und Nameserver, so dass man evtl. keinen Zugang mehr zum Schulnetz/Internet hat.
- Wenn auf dem PC der Network-Manager läuft, lässt sich die PC-IP-Adresse mit den zugehörigen grafischen Werkzeugen einstellen.
- Falls Sie ein RouterBoard mit WLAN haben (RB951G), kann man sich, z.B. von einem Notebook aus auch darüber anmelden.

Alternative Adressvergabe am Konfigurationsrechner

- Selbstverständlich kann man die PC-Schnittstelle auch als DHCP-Client konfigurieren, so dass sie vom Routerboard eine Adresse bekommt.
  Nachteil: das Routerboard setzt dann auch Standard-Gateway und Nameserver, so dass man evtl. keinen Zugang mehr zum Schulnetz/Internet hat.
- Wenn auf dem PC der Network-Manager läuft, lässt sich die PC-IP-Adresse mit den zugehörigen grafischen Werkzeugen einstellen.
- Falls Sie ein RouterBoard mit WLAN haben (RB951G), kann man sich, z.B. von einem Notebook aus auch darüber anmelden.

Alternative Adressvergabe am Konfigurationsrechner

- Selbstverständlich kann man die PC-Schnittstelle auch als DHCP-Client konfigurieren, so dass sie vom Routerboard eine Adresse bekommt.
  Nachteil: das Routerboard setzt dann auch Standard-Gateway und Nameserver, so dass man evtl. keinen Zugang mehr zum Schulnetz/Internet hat.
- Wenn auf dem PC der Network-Manager läuft, lässt sich die PC-IP-Adresse mit den zugehörigen grafischen Werkzeugen einstellen.
- Falls Sie ein RouterBoard mit WLAN haben (RB951G), kann man sich, z.B. von einem Notebook aus auch darüber anmelden.

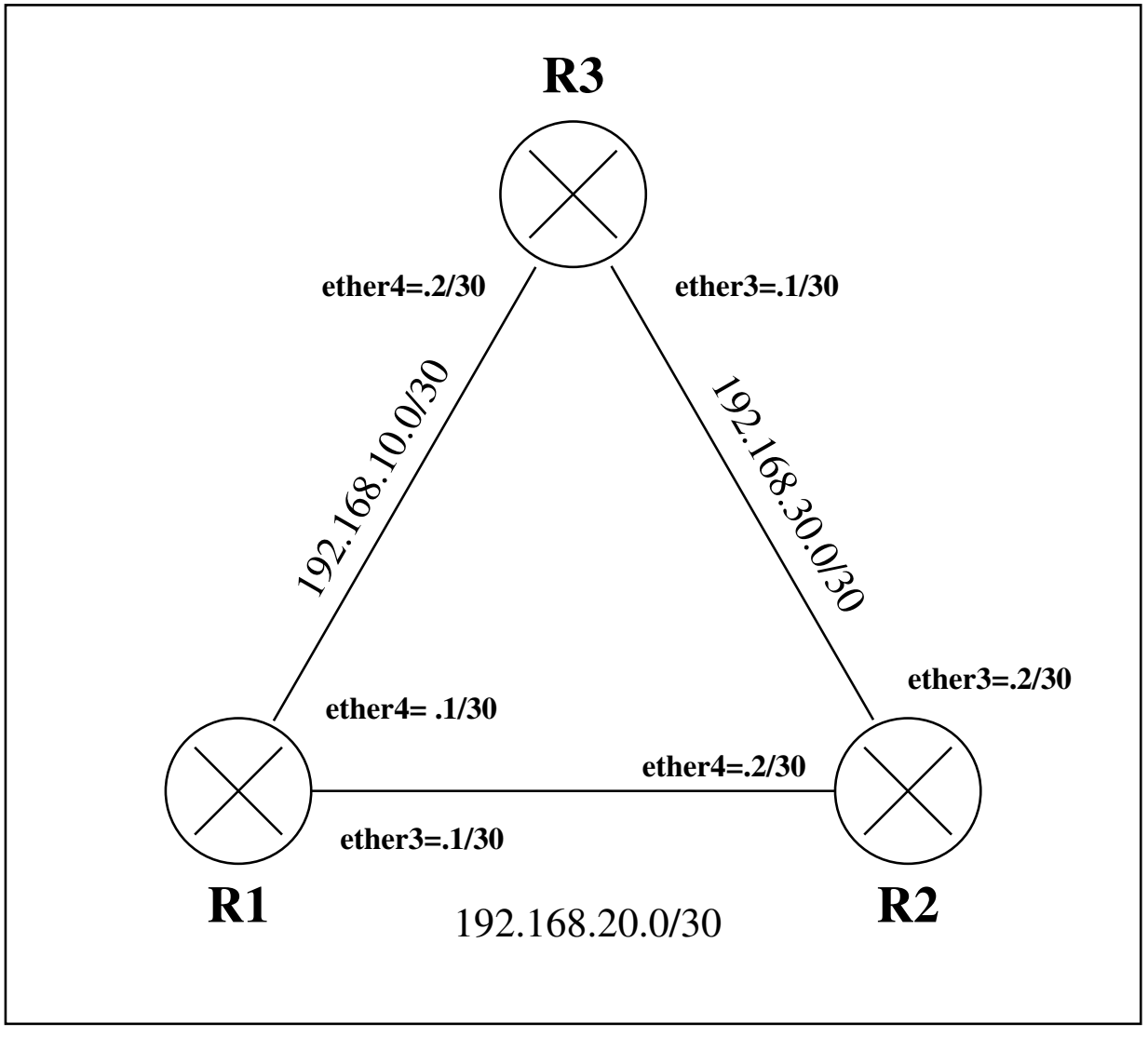
Inhalt

Wer ist die Firma MikroTik und was sind Routerboards

Routing

Paketfilterung auf den RouterBOARDs

Laboraufbau mit drei Routern



Statisches Routing mit drei Routern

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: `/system identity`
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: `/interface ethernet`
  - Adressen an ether3 und ether4 vergeben. Kontext: `/ip address`
  - Statische Routen setzen. Kontext: `/ip route`
  - Testen mit `/ping`

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: `/system identity`
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: `/interface ethernet`
  - Adressen an ether3 und ether4 vergeben. Kontext: `/ip address`
  - Statische Routen setzen. Kontext: `/ip route`
  - Testen mit `/ping`

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: `/system identity`
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: `/interface ethernet`
  - Adressen an ether3 und ether4 vergeben. Kontext: `/ip address`
  - Statische Routen setzen. Kontext: `/ip route`
  - Testen mit `/ping`

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: `/system identity`
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: `/interface ethernet`
  - Adressen an ether3 und ether4 vergeben. Kontext: `/ip address`
  - Statische Routen setzen. Kontext: `/ip route`
  - Testen mit `/ping`

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: `/system identity`
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: `/interface ethernet`
  - Adressen an ether3 und ether4 vergeben. Kontext: `/ip address`
  - Statische Routen setzen. Kontext: `/ip route`
  - Testen mit `/ping`

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: `/system identity`
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: `/interface ethernet`
  - Adressen an ether3 und ether4 vergeben. Kontext: `/ip address`
  - Statische Routen setzen. Kontext: `/ip route`
  - Testen mit `/ping`

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: `/system identity`
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: `/interface ethernet`
  - Adressen an ether3 und ether4 vergeben. Kontext: `/ip address`
  - Statische Routen setzen. Kontext: `/ip route`
  - Testen mit `/ping`

Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: /system identity
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: /interface ethernet
  - Adressen an ether3 und ether4 vergeben. Kontext: /ip address
  - Statische Routen setzen. Kontext: /ip route
  - Testen mit /ping

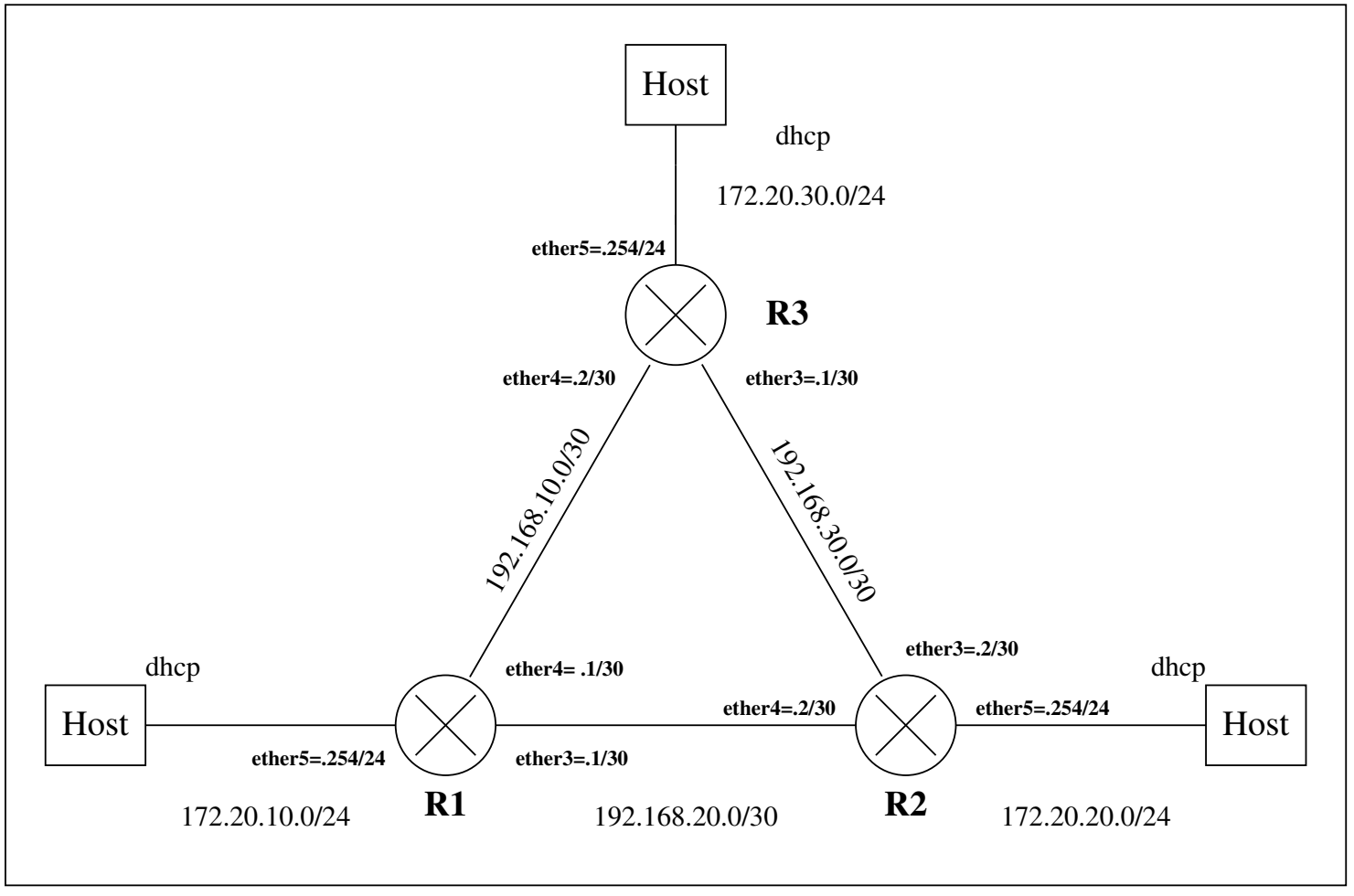
Statisches Routing

- Es soll ein Dreieck mit Routern aufgebaut und so konfiguriert werden, dass jeder Router die beiden Adressen im dritten, nicht direkt verbundenen Netzwerk erreichen kann.
- Folgende Schritte sind notwendig:
  - Router zurücksetzen, alles verkabeln.
  - Mit Router verbinden, Hostname vergeben. Kontext: /system identity
  - Auf jedem Router den Master-Port von ether3, ether4 und ether5 entfernen. Kontext: /interface ethernet
  - Adressen an ether3 und ether4 vergeben. Kontext: /ip address
  - Statische Routen setzen. Kontext: /ip route
  - Testen mit /ping

Statisches Routing: Lösung (Beispiel R2)

```
/system identity set name=R2
/interface ethernet set ether3,ether4,ether5 \
   master-port=none
/ip address
   add address=192.168.30.2/30 \
   interface=ether3 network=192.168.30.0
   add address=192.168.20.2/30 \
   interface=ether4 network=192.168.20.0
/ip route add dst-address=192.168.10.0/30 \
   gateway=192.168.20.1/30
/ip route add dst-address=192.168.10.0/30 \
   gateway=192.168.30.1/30
```

Laboraufbau mit drei Routern



Dynamisches Routing mit OSPF

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Das bereits aufgebaute Labornetz mit drei Routern soll auf dynamisches Routing mit OSPF umgestellt werden.
- Die Router sollen so konfiguriert werden, dass an ether5 Testrechner angeschlossen werden können, die ihre Adresse per DHCP beziehen.
  - IP-Adressen von ether5 fest vergeben.
  - IP-Adress-Pool erzeugen, Bereich .10 ... .50: /ip pool
  - DHCP-Server konfigurieren: /ip dhcp-server
  - DHCP-Netzwerk konfigurieren. Netzadresse, Gateway und DNS-Server vergeben: /ip dhcp-server network
  - Auf PC: `/etc/network/interfaces` anpassen: eth1 auf dhcp
  - Auf PC: `ifdown eth1` dann `ifup eth1` oder `dhclient eth1`

Dynamisches Routing mit OSPF

- Die Konfiguration von OSPF beschränkt sich auf die Angabe der mit dem Router verbundenen Netze.
- Eine OSPF-Area ist vorkonfiguriert (backbone) und kann als Area auf allen Routern verwendet werden.
  - Statische Routen löschen:

```
/ip route print
/ip route delete numbers=<zeilenNr>
```

  - Netzwerke für OSPF bekannt machen:
    /routing ospf network

Dynamisches Routing mit OSPF

- Die Konfiguration von OSPF beschränkt sich auf die Angabe der mit dem Router verbundenen Netze.
- Eine OSPF-Area ist vorkonfiguriert (backbone) und kann als Area auf allen Routern verwendet werden.
  - Statische Routen löschen:

```
/ip route print
/ip route delete numbers=<zeilenNr>
```

  - Netzwerke für OSPF bekannt machen:
    `/routing ospf network`

Dynamisches Routing mit OSPF

- Die Konfiguration von OSPF beschränkt sich auf die Angabe der mit dem Router verbundenen Netze.
- Eine OSPF-Area ist vorkonfiguriert (backbone) und kann als Area auf allen Routern verwendet werden.
  - Statische Routen löschen:

```
/ip route print
/ip route delete numbers=<zeilenNr>
```

  - Netzwerke für OSPF bekannt machen:
    /routing ospf network

Dynamisches Routing mit OSPF

- Die Konfiguration von OSPF beschränkt sich auf die Angabe der mit dem Router verbundenen Netze.
- Eine OSPF-Area ist vorkonfiguriert (backbone) und kann als Area auf allen Routern verwendet werden.
  - Statische Routen löschen:

```
/ip route print
/ip route delete numbers=<zeilenNr>
```

  - Netzwerke für OSPF bekannt machen:
    /routing ospf network

Dynamisches Routing mit OSPF

- Die Konfiguration von OSPF beschränkt sich auf die Angabe der mit dem Router verbundenen Netze.
- Eine OSPF-Area ist vorkonfiguriert (backbone) und kann als Area auf allen Routern verwendet werden.
  - Statische Routen löschen:

```
/ip route print
/ip route delete numbers=<zeilenNr>
```

  - Netzwerke für OSPF bekannt machen:
    /routing ospf network

Dynamisches Routing mit OSPF

- Zusatzaufgaben für **CCNA**s:
  - eigene Area *fobi* erzeugen `/routing ospf area` .
  - loopback-Interface mit Namen *loopback* erzeugen `/interface bridge` .
  - ip-Adresse auf loopback konfigurieren.
  - Router-ID setzen `/routing ospf instance` .

Dynamisches Routing mit OSPF

- Zusatzaufgaben für **CCNA**s:
  - eigene Area *fobi* erzeugen `/routing ospf area` .
  - loopback-Interface mit Namen *loopback* erzeugen `/interface bridge` .
  - ip-Adresse auf loopback konfigurieren.
  - Router-ID setzen `/routing ospf instance` .

Dynamisches Routing mit OSPF

- Zusatzaufgaben für **CCNA**s:
  - eigene Area *fobi* erzeugen `/routing ospf area` .
  - loopback-Interface mit Namen *loopback* erzeugen `/interface bridge` .
  - ip-Adresse auf loopback konfigurieren.
  - Router-ID setzen `/routing ospf instance` .

Dynamisches Routing mit OSPF

- Zusatzaufgaben für **CCNA**s:
  - eigene Area *fobi* erzeugen /routing ospf area .
  - loopback-Interface mit Namen *loopback* erzeugen /interface bridge .
  - ip-Adresse auf loopback konfigurieren.
  - Router-ID setzen /routing ospf instance .

Dynamisches Routing mit OSPF

- Zusatzaufgaben für **CCNA**s:
  - eigene Area *fobi* erzeugen /routing ospf area .
  - loopback-Interface mit Namen *loopback* erzeugen /interface bridge .
  - ip-Adresse auf loopback konfigurieren.
  - Router-ID setzen /routing ospf instance .

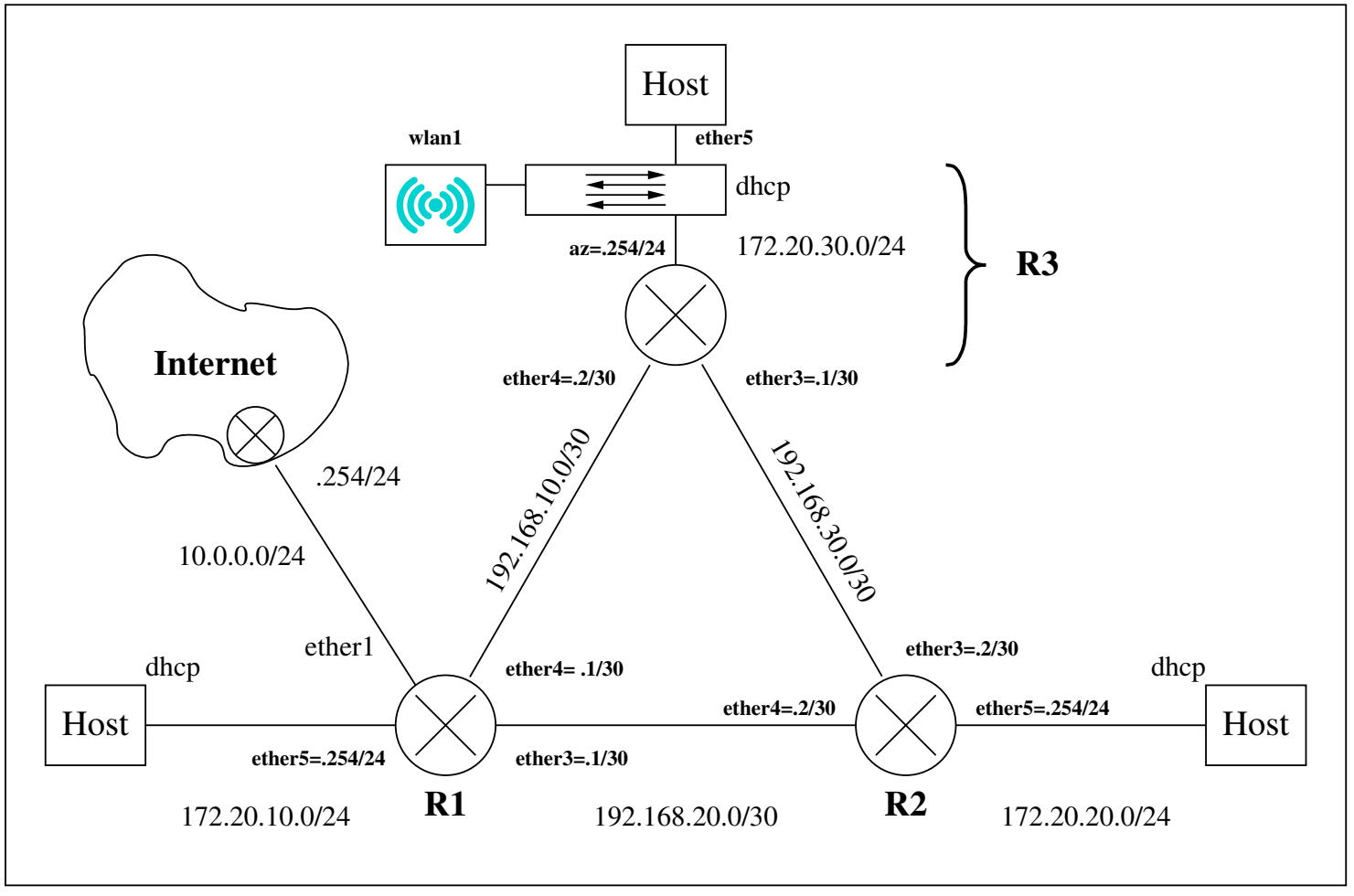
Dynamisches Routing mit OSPF

- Zusatzaufgaben für **CCNA**s:
  - eigene Area *fobi* erzeugen /routing ospf area .
  - loopback-Interface mit Namen *loopback* erzeugen /interface bridge .
  - ip-Adresse auf loopback konfigurieren.
  - Router-ID setzen /routing ospf instance .

Dynamisches Routing: Lösung (Beispiel R2)

```
/ip address add address=172.20.20.254/24 interface=ether5
/ip pool
   add name=fobi-dhcp ranges=172.20.20.10-172.20.20.50
/ip dhcp-server
   add address-pool=fobi-dhcp disabled=no interface=ether5 \
    lease-time=30m name=testnetzEther5
/ip dhcp-server network
    add address=172.20.20.0/24 comment="netzwerk an ether5" \
    dns-server=10.16.1.1 gateway=172.20.20.254
#loeschen der statischen routen
/routing ospf network
    add area=backbone network=192.168.30.0/30
    add area=backbone network=192.168.20.0/30
    add area=backbone network=172.20.20.0/24
```

OSPF-Routing mit Default-GW, WLAN und Bridge



Routernetzwerk mit WLAN und Internetzugang

Routernetzwerk mit WLAN und Internetzugang

Routernetzwerk mit WLAN und Internetzugang

- Das OSPF-Netzwerk soll um WLAN-Zugang an Router3 und Internetverbindung an Router R1 oder R2 erweitert werden.

Routernetzwerk mit WLAN und Internetzugang

- Das OSPF-Netzwerk soll um WLAN-Zugang an Router3 und Internetverbindung an Router R1 oder R2 erweitert werden.
- Schnelleinrichtung des WLANs unter Verwendung des vorgefertigten Sicherheitsprofils *default*:

Routernetzwerk mit WLAN und Internetzugang

- Das OSPF-Netzwerk soll um WLAN-Zugang an Router3 und Internetverbindung an Router R1 oder R2 erweitert werden.
- Schnelleinrichtung des WLANs unter Verwendung des vorgefertigten Sicherheitsprofils *default*:
  Sicherheitsprofil default anpassen (wpa2-psk, AES):

```
/interface wireless security-profiles
   set 0 authentication-types=wpa2-psk mode=dynamic-keys \
   wpa2-pre-shared-key=arbeitszimmer
```

Routernetzwerk mit WLAN und Internetzugang

- Das OSPF-Netzwerk soll um WLAN-Zugang an Router3 und Internetverbindung an Router R1 oder R2 erweitert werden.
- Schnelleinrichtung des WLANs unter Verwendung des vorgefertigten Sicherheitsprofils *default*:
  Sicherheitsprofil default anpassen (wpa2-psk, AES):

```
/interface wireless security-profiles
   set 0 authentication-types=wpa2-psk mode=dynamic-keys \
   wpa2-pre-shared-key=arbeitszimmer
```

  WLAN konfigurieren (SSID):

```
/interface wireless
   set 0 security-profile=default disabled=no \
   ssid=arbeitszimmer
```

Routernetzwerk mit WLAN und Internetzugang

Routernetzwerk mit WLAN und Internetzugang

- Die Schnittstellen *ether5* und *wlan1* sollen an einem logischen Switch (*bridge*, `name=az`) zusammengefasst werden. /interface bridge und /interface bridge port

Routernetzwerk mit WLAN und Internetzugang

- Die Schnittstellen *ether5* und *wlan1* sollen an einem logischen Switch (*bridge*, `name=az`) zusammengefasst werden. /interface bridge und /interface bridge port
- Der DHCP-Server soll so konfiguriert werden, dass er auf dhcp-Anfragen an der Bridge-Schnittstelle az antwortet.

Routernetzwerk mit WLAN und Internetzugang

- Die Schnittstellen *ether5* und *wlan1* sollen an einem logischen Switch (*bridge*, `name=az`) zusammengefasst werden. /interface bridge und /interface bridge port
- Der DHCP-Server soll so konfiguriert werden, dass er auf dhcp-Anfragen an der Bridge-Schnittstelle az antwortet.
- Damit OSPF Netze an der Bridge-Schnittstelle auch an andere Router verteilt:

```
/routing ospf instance
   set 0 redistribute-connected=as-type-1
```

Routernetzwerk mit WLAN und Internetzugang

Routernetzwerk mit WLAN und Internetzugang

- Source-NAT über ether1 Richtung Internet ist bereits vorkonfiguriert:

```
/ip firewall nat
   add action=masquerade chain=srcnat  \
   ipsec-policy=out,none out-interface-list=WAN
```

Routernetzwerk mit WLAN und Internetzugang

- Source-NAT über ether1 Richtung Internet ist bereits vorkonfiguriert:

```
/ip firewall nat
   add action=masquerade chain=srcnat  \
   ipsec-policy=out,none out-interface-list=WAN
```

- Das Default-Gateway auf R1(in Raum R023: 10.0.0.254) muss statisch gesetzt werden.

Routernetzwerk mit WLAN und Internetzugang

- Source-NAT über ether1 Richtung Internet ist bereits vorkonfiguriert:

```
/ip firewall nat
   add action=masquerade chain=srcnat  \
   ipsec-policy=out,none out-interface-list=WAN
```

- Das Default-Gateway auf R1(in Raum R023: 10.0.0.254) muss statisch gesetzt werden.
- Mit folgender Router-Konfig auf R1, wird das Default-GW mit OSPF an R2 und R3 verteilt:

```
/ip routing ospf instance
   set 0 distribute-default=always-as-type-1
```

Hinweise zum Standardgateway

Hinweise zum Standardgateway

- Falls an den RouterBoard-Schnittstellen ether5 Rechner aus dem Schulnetz der WaRa angeschlossen werden, haben diese bereits ein Standardgateway, das sie vom Schulsverver über eth0 beziehen.

Hinweise zum Standardgateway

- Falls an den RouterBoard-Schnittstellen ether5 Rechner aus dem Schulnetz der WaRa angeschlossen werden, haben diese bereits ein Standardgateway, das sie vom Schulsverver über eth0 beziehen.
- Mit folgenden Kommandos auf einer Root-Konsole auf dem Schulrechner, kann man sich vom Zwangs-Routing durch den Schulserver befreien. Das Beispiel setzt voraus, dass Sie mit Schnittstelle eth1 mit dem RB verbunden sind:

```
ifdown eth1
ifup eth1
ip a #Adr. verifizieren
ip route del 0.0.0.0/0 dev eth0
ifdown eth1
ifup eth1
ip route show  #GW verifizieren
cat /etc/resolv.conf #DNS verifizieren
```

Inhalt

Wer ist die Firma MikroTik und was sind Routerboards

Routing

Paketfilterung auf den RouterBOARDs

Kurzer Ausflug in die IP-Tables unter Linux

- Die Firewall besteht aus **Tabellen**.
- Eine Tabelle enthält mehrere Filter-**Ketten**.
- Eine Kette besteht aus **Regeln**, die Regeln sind also die Kettenglieder. Die Regeln einer Kette werden nacheinander durchlaufen. Trifft eine Regel zu, wird die Kette verlassen.
- Eine Regel endet mit der Angabe eines Sprung-**Ziels**. Das Ziel bestimmt, was mit dem Paket gemacht wird: DROP, ACCEPT, DNAT, ... oder ob man zu einer anderen Kette springt.

Kurzer Ausflug in die IP-Tables unter Linux

- Die Firewall besteht aus **Tabellen**.
- Eine Tabelle enthält mehrere Filter-**Ketten**.
- Eine Kette besteht aus **Regeln**, die Regeln sind also die Kettenglieder. Die Regeln einer Kette werden nacheinander durchlaufen. Trifft eine Regel zu, wird die Kette verlassen.
- Eine Regel endet mit der Angabe eines Sprung-**Ziels**. Das Ziel bestimmt, was mit dem Paket gemacht wird: DROP, ACCEPT, DNAT, ... oder ob man zu einer anderen Kette springt.

Kurzer Ausflug in die IP-Tables unter Linux

- Die Firewall besteht aus **Tabellen**.
- Eine Tabelle enthält mehrere Filter-**Ketten**.
- Eine Kette besteht aus **Regeln**, die Regeln sind also die Kettenglieder. Die Regeln einer Kette werden nacheinander durchlaufen. Trifft eine Regel zu, wird die Kette verlassen.
- Eine Regel endet mit der Angabe eines Sprung-**Ziels**. Das Ziel bestimmt, was mit dem Paket gemacht wird: DROP, ACCEPT, DNAT, ... oder ob man zu einer anderen Kette springt.

Kurzer Ausflug in die IP-Tables unter Linux

- Die Firewall besteht aus **Tabellen**.
- Eine Tabelle enthält mehrere Filter-**Ketten**.
- Eine Kette besteht aus **Regeln**, die Regeln sind also die Kettenglieder. Die Regeln einer Kette werden nacheinander durchlaufen. Trifft eine Regel zu, wird die Kette verlassen.
- Eine Regel endet mit der Angabe eines Sprung-**Ziels**. Das Ziel bestimmt, was mit dem Paket gemacht wird: DROP, ACCEPT, DNAT, ... oder ob man zu einer anderen Kette springt.

Kurzer Ausflug in die IP-Tables unter Linux

- Die Firewall besteht aus **Tabellen**.
- Eine Tabelle enthält mehrere Filter-**Ketten**.
- Eine Kette besteht aus **Regeln**, die Regeln sind also die Kettenglieder. Die Regeln einer Kette werden nacheinander durchlaufen. Trifft eine Regel zu, wird die Kette verlassen.
- Eine Regel endet mit der Angabe eines Sprung-**Ziels**. Das Ziel bestimmt, was mit dem Paket gemacht wird: DROP, ACCEPT, DNAT, ... oder ob man zu einer anderen Kette springt.

Die Tabellen FILTER, NAT und MANGLE

filter ist die Standardtabelle. Ist keine Tabelle angegeben (Option `-t`), wird *filter* verwendet.

nat Die Tabelle für NAT wird mit `-t nat` aufgerufen.

mangle Die Tabelle `mangle` wird hier nicht weiter besprochen.

Die Tabellen FILTER, NAT und MANGLE

filter ist die Standardtabelle. Ist keine Tabelle angegeben (Option `-t`), wird *filter* verwendet.

nat Die Tabelle für NAT wird mit `-t nat` aufgerufen.

mangle Die Tabelle `mangle` wird hier nicht weiter besprochen.

Die Tabellen FILTER, NAT und MANGLE

filter ist die Standardtabelle. Ist keine Tabelle angegeben (Option `-t`), wird *filter* verwendet.

nat Die Tabelle für NAT wird mit `-t nat` aufgerufen.

mangle Die Tabelle `mangle` wird hier nicht weiter besprochen.

Die Tabellen FILTER, NAT und MANGLE

filter ist die Standardtabelle. Ist keine Tabelle angegeben (Option `-t`), wird *filter* verwendet.

nat Die Tabelle für NAT wird mit `-t nat` aufgerufen.

mangle Die Tabelle `mangle` wird hier nicht weiter besprochen.

Sprungziele

Sprungziele (targets) bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit **-j** oder **–jump** aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

- DROP
- ACCEPT
- MASQUERADE: gibt es nur in der nat-Tabelle, entspricht Source-NAT
- DNAT (Destination-NAT): für Port-Forwarding; gibt es nur in der nat-Tabelle

Sprungziele

Sprungziele (targets) bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit **-j** oder **–jump** aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

- DROP
- ACCEPT
- MASQUERADE: gibt es nur in der nat-Tabelle, entspricht Source-NAT
- DNAT (Destination-NAT): für Port-Forwarding; gibt es nur in der nat-Tabelle

Sprungziele

Sprungziele (targets) bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit **-j** oder **–jump** aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

- DROP
- ACCEPT
- MASQUERADE: gibt es nur in der nat-Tabelle, entspricht Source-NAT
- DNAT (Destination-NAT): für Port-Forwarding; gibt es nur in der nat-Tabelle

Sprungziele

Sprungziele (targets) bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit **-j** oder **–jump** aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

- DROP
- ACCEPT
- MASQUERADE: gibt es nur in der nat-Tabelle, entspricht Source-NAT
- DNAT (Destination-NAT): für Port-Forwarding; gibt es nur in der nat-Tabelle

Sprungziele

Sprungziele (targets) bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit **-j** oder **–jump** aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

- DROP
- ACCEPT
- MASQUERADE: gibt es nur in der nat-Tabelle, entspricht Source-NAT
- DNAT (Destination-NAT): für Port-Forwarding; gibt es nur in der nat-Tabelle

Sprungziele

Sprungziele (targets) bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit **-j** oder **–jump** aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

- DROP
- ACCEPT
- MASQUERADE: gibt es nur in der nat-Tabelle, entspricht Source-NAT
- DNAT (Destination-NAT): für Port-Forwarding; gibt es nur in der nat-Tabelle

Vordefinierte Filterketten

Es gibt 5 vordefinierte *Ketten* (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding = Destination NAT*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *Source NAT*)

Vordefinierte Filterketten

Es gibt 5 vordefinierte *Ketten* (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding = Destination NAT*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *Source NAT*)

Vordefinierte Filterketten

Es gibt 5 vordefinierte *Ketten* (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding = Destination NAT*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *Source NAT*)

Vordefinierte Filterketten

Es gibt 5 vordefinierte *Ketten* (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding = Destination NAT*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *Source NAT*)

Vordefinierte Filterketten

Es gibt 5 vordefinierte *Ketten* (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding = Destination NAT*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *Source NAT*)

Vordefinierte Filterketten

Es gibt 5 vordefinierte *Ketten* (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding = Destination NAT*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *Source NAT*)

Vordefinierte Filterketten

Es gibt 5 vordefinierte *Ketten* (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding = Destination NAT*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *Source NAT*)

Zusammenfassung

filter
FORWARD
INPUT
OUTPUT

Tabelle: filter

nat
PREROUTING
OUTPUT
POSTROUTING

Tabelle: nat

mangle
PREROUTING
POSTROUTING
OUTPUT
INPUT
FORWARD

Tabelle: mangle

Zusammenfassung

filter
FORWARD
INPUT
OUTPUT

Tabelle: filter

nat
PREROUTING
OUTPUT
POSTROUTING

Tabelle: nat

mangle
PREROUTING
POSTROUTING
OUTPUT
INPUT
FORWARD

Tabelle: mangle

Zusammenfassung

filter
FORWARD
INPUT
OUTPUT

Tabelle: filter

nat
PREROUTING
OUTPUT
POSTROUTING

Tabelle: nat

mangle
PREROUTING
POSTROUTING
OUTPUT
INPUT
FORWARD

Tabelle: mangle

Zusammenfassung

filter
FORWARD
INPUT
OUTPUT

Tabelle: filter

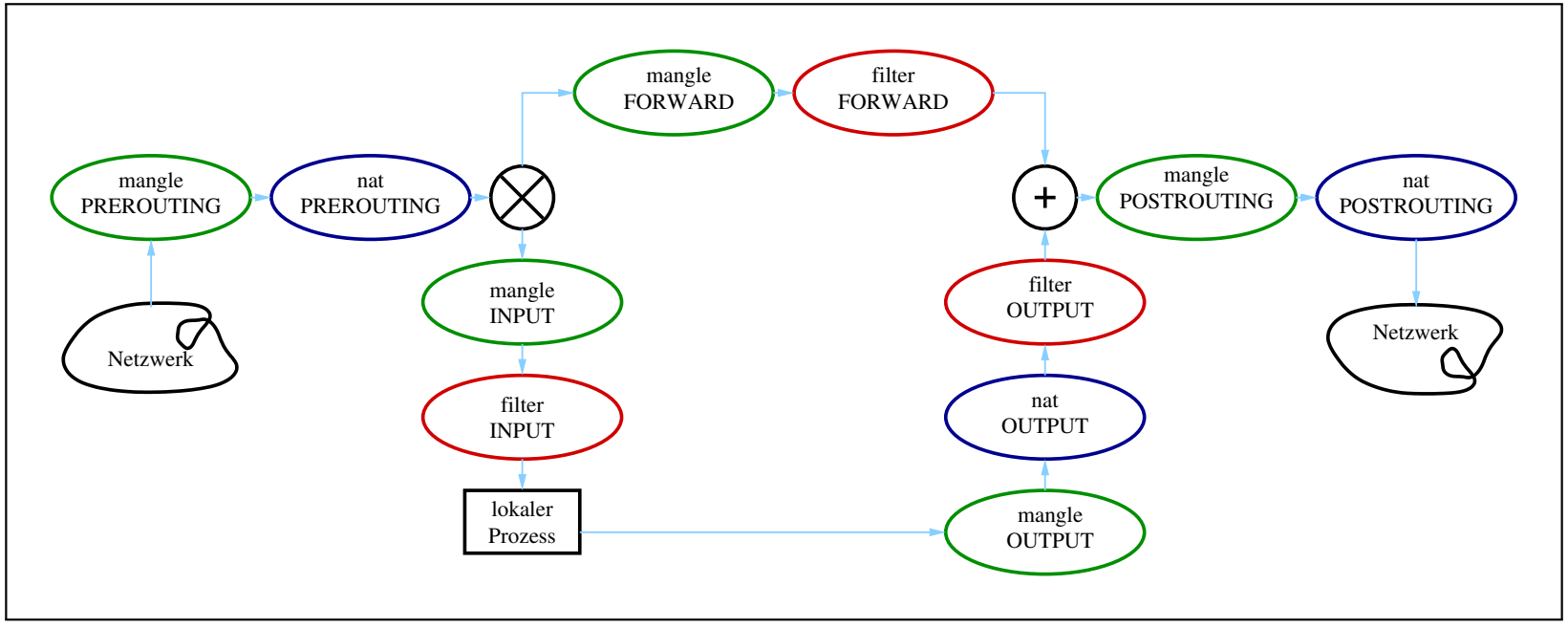
nat
PREROUTING
OUTPUT
POSTROUTING

Tabelle: nat

mangle
PREROUTING
POSTROUTING
OUTPUT
INPUT
FORWARD

Tabelle: mangle

Weg eines Pakets durch die Tables



Weg der Pakete durch iptables

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act $\Rightarrow$ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act ⇒ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act ⇒ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act ⇒ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act ⇒ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act ⇒ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act ⇒ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act ⇒ requires the routers in USA to have ability to intercept and log network traffic

MikroTik-Firewall = IP-Tables

- Das OS der MikroTik-Router ist Linux
- D.h. die MikroTik-Firewall ist nichts Anderes als das Linux-Paketefilter *IPTables*
- Syntax ist abweichend. Z.B `-j MASQUERADE` (Jump) wird ersetzt durch `action=masquerade`
- es gibt die Kontexte:
  - /ip firewall filter
  - /ip firewall nat
  - sowie Kontexte für die Tabellen `mangle` und `raw` und für ein L7-Filter
  - `/ip firewall calea`: Communications Assistance for Law Enforcement Act $\Rightarrow$ requires the routers in USA to have ability to intercept and log network traffic

Ein Firewall-Beispiel

```
/interface list
   name=WAN
/interface list member
   add interface=ether1 list=WAN
/ip firewall nat
   add action=masquerade chain=srcnat comment=''defconf: masquerade''
   ipsec-policy=out,none out-interface-list=WAN
```