

**Titel:** RIP NG

**Klasse:** 4BHIF

**Name:** Haiden

**Gruppe:** 01

**Aufgabe:** 25.05.2021 **Abgabe:** 08.06.2021

# Inhaltsverzeichnis

1	Theorie-Teil.....	1
1.1	Overview.....	1
1.2	Protocol .....	1
1.3	Versions .....	1
1.3.1	RIP version 1 .....	1
1.3.2	RIP version 2 .....	2
1.3.3	RIPng.....	2
1.4	Packet Structure .....	3
1.5	RIPNg .....	3
1.5.1	Overview.....	4
1.5.2	Packets.....	4
2	Praxisteil .....	5
2.1	Aufbau .....	5
2.2	IP Adressen Konfiguration (v6).....	6
2.2.1	Berechnung der Subnets .....	6
2.2.1.1	Subnet X.....	6
2.2.2	Subnet y:.....	6
2.2.3	Subnet z: .....	7
2.2.4	Router Konfiguration .....	7
2.2.5	Ping test zu anderem Router im Netzwerk .....	8
2.2.6	Router zu PC: .....	8
2.2.7	PC (VM) Konfiguration:.....	9
2.2.8	From Router: .....	9
2.3	RIP Konfiguration.....	9
2.3.1	Loopback Konfiguration .....	10
2.3.2	Anzeigen der RIP Neighbors .....	10
2.3.3	Anzeige der IPV6 Konfiguration.....	12
2.4	Verbinden Sie die einzelnen Netzwerke miteinander. Können Sie die Routen untereinander austauschen?.....	12
2.4.1	Netzwerksskizze .....	12
2.4.2	Aufbau .....	12
2.4.3	Anzeige der Routen nach Konfiguration .....	13

# 1 Theorie-Teil

<https://www.juniper.net/documentation/us/en/software/junos/rip/topics/topic-map/rip-and-ripng-overview.html>

[https://en.wikipedia.org/wiki/Routing\\_Information\\_Protocol](https://en.wikipedia.org/wiki/Routing_Information_Protocol)

## 1.1 Overview

RIP is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using the hop count as the metric.

In a RIP network, each router's forwarding table is distributed among the nodes through the flooding of routing table information. Because topology changes are flooded throughout the network, every node maintains the same list of destinations. Packets are then routed to these destinations based on path-cost calculations done at each node in the network.

## 1.2 Protocol

The RIP IGP uses the Bellman-Ford, or distance-vector, algorithm to determine the best route to a destination. RIP uses the hop count as the metric. RIP enables hosts and routers to exchange information for computing routes through an IP-based network. RIP is intended to be used as an IGP in reasonably homogeneous networks of moderate size.

RIP version 1 packets contain the minimal information necessary to route packets through a network. However, this version of RIP does not support authentication or subnetting.

RIP uses User Datagram Protocol (UDP) port 520.

RIP has the following architectural limitations:

- The longest network path cannot exceed 15 hops (assuming that each network, or hop, has a cost of 1).
- RIP depends on counting to infinity to resolve certain unusual situations—When the network consists of several hundred routers, and when a routing loop has formed, the amount of time and network bandwidth required to resolve a next hop might be great.
- RIP uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

## 1.3 Versions

There are three standardized versions of the Routing Information Protocol: RIPv1 and RIPv2 for IPv4, and RIPv6 for IPv6.

### 1.3.1 RIP version 1

The original specification of RIP, defined in RFC 1058, was published in 1988.[5] When starting up, and every 30 seconds thereafter, a router with RIPv1 implementation broadcasts to 255.255.255.255 a request message through every RIPv1 enabled interface. Neighbouring routers receiving the request message respond with a RIPv1 segment, containing their routing table. The requesting router updates its own routing table, with the reachable IP network address, hop count and next hop, that is the router interface IP address from which the RIPv1 response was sent. As the requesting router receives updates from different neighbouring routers it will only update the reachable networks in its routing table, if it receives information about a reachable network it has not yet in its routing table or information that a network it has in its routing table is reachable with a lower hop count. Therefore, a RIPv1 router will in most cases only have one entry for a reachable network, the one with the

lowest hop count. If a router receives information from two different neighbouring router that the same network is reachable with the same hop count but via two different routes, the network will be entered into the routing table two times with different next hop routers. The RIPv1 enabled router will then perform what is known as equal-cost load balancing for IP packets.[4]

RIPv1 enabled routers not only request the routing tables of other routers every 30 seconds, they also listen to incoming requests from neighbouring routers and send their own routing table in turn. RIPv1 routing tables are therefore updated every 25 to 35 seconds.[4] The RIPv1 protocol adds a small random time variable to the update time, to avoid routing tables synchronizing across a LAN.[6] It was thought, as a result of random initialization, the routing updates would spread out in time, but this was not true in practice. Sally Floyd and Van Jacobson showed in 1994 that, without slight randomization of the update timer, the timers synchronized over time.[7]

RIPv1 can be configured into silent mode, so that a router requests and processes neighbouring routing tables, and keeps its routing table and hop count for reachable networks up to date, but does not needlessly send its own routing table into the network. Silent mode is commonly implemented to hosts.[8]

RIPv1 uses classful routing. The periodic routing updates do not carry subnet information, lacking support for variable length subnet masks (VLSM). This limitation makes it impossible to have different-sized subnets inside of the same network class. In other words, all subnets in a network class must have the same size. There is also no support for router authentication, making RIPv1 vulnerable to various attacks.

### 1.3.2 RIPv2

Due to the deficiencies of the original RIP specification, RIPv2 was developed in 1993[4], published as RFC 1723 in 1994, and declared Internet Standard 56 in 1998.[9] It included the ability to carry subnet information, thus supporting Classless Inter-Domain Routing (CIDR). To maintain backward compatibility, the hop count limit of 15 remained. RIPv2 has facilities to fully interoperate with the earlier specification if all Must Be Zero protocol fields in the RIPv1 messages are properly specified. In addition, a compatibility switch feature[9] allows fine-grained interoperability adjustments.

In an effort to avoid unnecessary load on hosts that do not participate in routing, RIPv2 multicasts the entire routing table to all adjacent routers at the address 224.0.0.9, as opposed to RIPv1 which uses broadcast. Unicast addressing is still allowed for special applications.

(MD5) authentication for RIPv2 was introduced in 1997.[10][11]

Route tags were also added in RIPv2. This functionality allows a distinction between routes learned from the RIPv2 protocol and routes learned from other protocols.

### 1.3.3 RIPv6

RIPv6 (RIPv6 next generation), defined in RFC 2080,[12] is an extension of RIPv2 for support of IPv6, the next generation Internet Protocol. The main differences between RIPv2 and RIPv6 are:

Support of IPv6 networking.

While RIPv2 supports RIPv1 updates authentication, RIPv6 does not. IPv6 routers were, at the time, supposed to use IPsec for authentication.

RIPv2 encodes the next-hop into each route entry, RIPv2 requires specific encoding of the next hop for a set of route entries.

RIPv2 sends updates on UDP port 521 using the multicast group ff02::9.

## 1.4 Packet Structure

RIP packets contain the following fields:

- **Command**—Indicates whether the packet is a request or response message. Request messages seek information for the router's routing table. Response messages are sent periodically and also when a request message is received. Periodic response messages are called update messages. Update messages contain the command and version fields and 25 destinations (by default), each of which includes the destination IP address and the metric to reach that destination.
- **Version number**—Version of RIP that the originating router is running.
- **Address family identifier**—Address family used by the originating router. The family is always IP.
- **Address**—IP address included in the packet.
- **Metric**—Value of the metric advertised for the address.
- **Mask**—Mask associated with the IP address (RIP version 2 only).
- **Next hop**—IP address of the next-hop router (RIP version 2 only).

Routing information is exchanged in a RIP network by RIP request and RIP response packets. A router that has just booted can broadcast a RIP request on all RIP-enabled interfaces. Any routers running RIP on those links receive the request and respond by sending a RIP response packet immediately to the router. The response packet contains the routing table information required to build the local copy of the network topology map.

In the absence of RIP request packets, all RIP routers broadcast a RIP response packet every 30 seconds on all RIP-enabled interfaces. The RIP broadcast is the primary way in which topology information is flooded throughout the network.

Once a router learns about a particular destination through RIP, it starts a timer. Every time it receives a new response packet with information about the destination, the router resets the timer to zero. However, if the router receives no updates about a particular destination for 180 seconds, it removes the destination from its RIP routing table.

In addition to the regular transmission of RIP packets every 30 seconds, if a router detects a new neighbor or detects that an interface is unavailable, it generates a triggered update. The new routing information is immediately broadcast out all RIP-enabled interfaces, and the change is reflected in all subsequent RIP response packets.

## 1.5 RIPv2

The Routing Information Protocol next generation (RIPv2) is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using hop count as the

metric. RIPng exchanges routing information used to compute routes and is intended for IP version 6 (IPv6)-based networks.

### 1.5.1 Overview

The RIPng IGP uses the Bellman-Ford distance-vector algorithm to determine the best route to a destination, using hop count as the metric. RIPng allows hosts and routers to exchange information for computing routes through an IP-based network. RIPng is intended to act as an IGP for moderately-sized autonomous systems.

RIPng is a distinct routing protocol from RIPv2. Following differences:

- RIPng does not need to implement authentication on packets.
- Junos OS does not support multiple instances of RIPng.
- Junos OS does not support RIPng routing table groups.
- RIPng is a UDP-based protocol and uses UDP port 521.
- RIPng has the following architectural limitations:
- The longest network path cannot exceed 15 hops (assuming that each network, or hop, has a cost of 1).
- RIPng is prone to routing loops when the routing tables are reconstructed. Especially when RIPng is implemented in large networks that consist of several hundred routers, RIPng might take an extremely long time to resolve routing loops.
- RIPng uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

### 1.5.2 Packets

A RIPng packet header contains the following fields:

- Command—Indicates whether the packet is a request or response message. Request messages seek information for the router's routing table. Response messages are sent periodically or when a request message is received. Periodic response messages are called update messages. Update messages contain the command and version fields and a set of destinations and metrics.
- Version number—Specifies the version of RIPng that the originating router is running. This is currently set to Version 1.

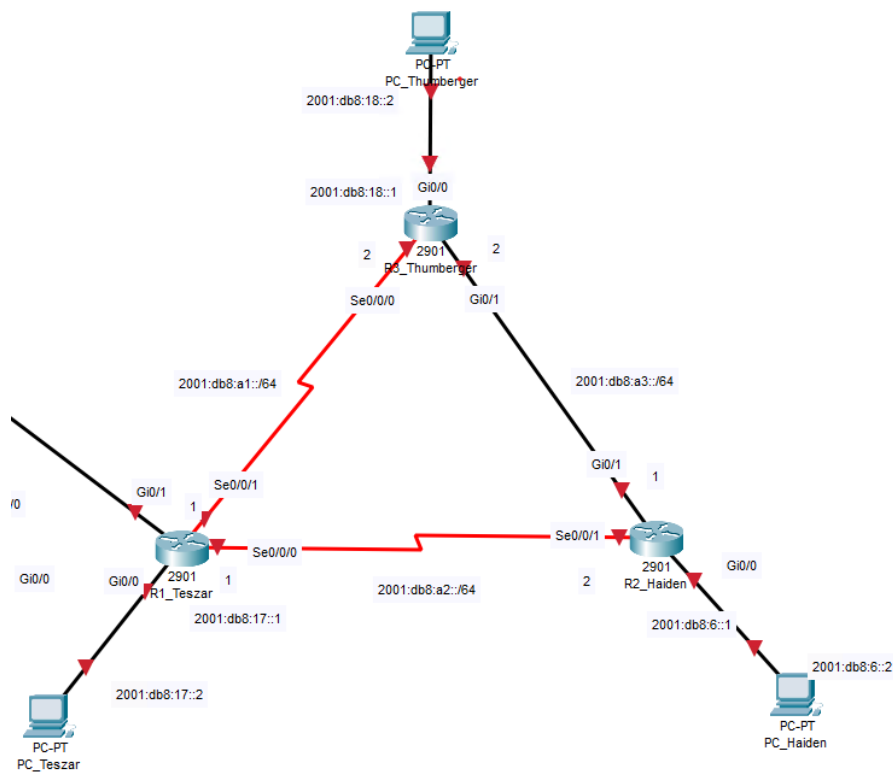
The rest of the RIPng packet contains a list of routing table entries consisting of the following fields:

- Destination prefix—128-bit IPv6 address prefix for the destination.
- Prefix length—Number of significant bits in the prefix.
- Metric—Value of the metric advertised for the address.
- Route tag—A route attribute that must be advertised and redistributed with the route. Primarily, the route tag distinguishes external RIPng routes from internal RIPng routes when routes must be redistributed across an exterior gateway protocol (EGP).

## 2 Praxisteil

### 2.1 Aufbau

Aufbau im Labor:



## 2.2 IP Adressen Konfiguration (v6)

### 2.2.1 Berechnung der Subnets

#### 2.2.1.1 Subnet X

CIDR Notation ⓘ	Address
<input type="text" value="2001:0db8:00a1:0000:0000"/>	<input type="text" value="2001:0db8:00a1:0000:0000"/>
Address Range Start	Address Range End
<input type="text" value="2001:0db8:00a1:0000:0000"/>	<input type="text" value="2001:0db8:00a1:0000:ffff:ff"/>

#### 2.2.2 Subnet y:

CIDR Notation ⓘ	Address
<input type="text" value="2001:0db8:00a2:0000:0000"/>	<input type="text" value="2001:0db8:00a2:0000:0000"/>
Address Range Start	Address Range End
<input type="text" value="2001:0db8:00a2:0000:0000"/>	<input type="text" value="2001:0db8:00a2:0000:ffff:ff"/>



### 2.2.3 Subnet z:

CIDR Notation ⓘ	Address
<input type="text" value="2001:0db8:00a3:0000:0000"/>	<input type="text" value="2001:0db8:00a3:0000:0000"/>
Address Range Start	Address Range End
<input type="text" value="2001:0db8:00a3:0000:0000"/>	<input type="text" value="2001:0db8:00a3:0000:ffff:ff"/>

### 2.2.4 Router Konfiguration

In diesem Abschnitt wird der Router für die Verwendung konfiguriert.

```
ROUTER2 (config) #ipv6 unicast-routing
ROUTER2 (config) #int s0/0/1
ROUTER2 (config-if) #ipv6 address 2001:db8:a1::1/64
ROUTER2 (config-if) #no shut
ROUTER2 (config-if) #exit
ROUTER2 (config) #int s0/0/1
ROUTER2 (config-if) #clock rate 64000
ROUTER2 (config-if) #no shut
ROUTER2 (config-if) #exit
ROUTER2 (config) #int g0/1
ROUTER2 (config-if) #exit
ROUTER2 (config) #int s0/0/1
ROUTER2 (config-if) #ipv6 enable
ROUTER2 (config-if) #ipv6 address 2001:db8:a1::1/64
ROUTER2 (config-if) #no shut
ROUTER2 (config-if) #exit
ROUTER2 (config) #int g0/1
ROUTER2 (config-if) #ipv6 enable
ROUTER2 (config-if) #ipv6 address 2001:db8:A3::1/64
```

```
ROUTER2(config-if)#no shut
ROUTER2(config-if)#exit
ROUTER2(config)#int s0/0/1
ROUTER2(config-if)#ipv6 address 2001:db8:A2::2/64
ROUTER2(config-if)#no shut
ROUTER2(config-if)#clock rate 64000
ROUTER2(config-if)#exit
ROUTER2(config)#
```

### 2.2.5 Ping test zu anderem Router im Netzwerk

Zuerst wird Teszar im A2 Netzwerk angepingt und danach Thumberger, um sicher zu stellen, dass ich und sie beide ihre Konfiguration richtig eingestellt haben.

```
ROUTER2#ping 2001:db8:A2::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A2::1, timeout is 2
seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/28
ms
ROUTER2#ping 2001:db8:A3::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A3::2, timeout is 2
seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

### 2.2.6 Router zu PC:

Einrichten des Gigabit Interfaces welches mit dem PC verbunden ist und an die VM durchgereicht wird:

```
ROUTER2(config)#int g0/0
ROUTER2(config-if)#ipv6 enable
ROUTER2(config-if)#ipv6 address 2001:db8:6::1/64
ROUTER2(config-if)#no shut
ROUTER2(config-if)#
```

## 2.2.7 PC (VM) Konfiguration:

Adresse	Präfix	Gateway
2001:db8:6::2	64	2001:db8:6::1

DNS-Server

Versuch eines Pings vom PC zum Router:

```
schueler@Debian10nvs:~$ ping 2001:db8:6::1
PING 2001:db8:6::1 (2001:db8:6::1) 56 data bytes
64 bytes from 2001:db8:6::1: icmp_seq=1 ttl=64 time=0.695 ms
64 bytes from 2001:db8:6::1: icmp_seq=2 ttl=64 time=0.798 ms
64 bytes from 2001:db8:6::1: icmp_seq=3 ttl=64 time=1.20 ms
```

## 2.2.8 From Router:

Versuch des Pings vom Router aus zum PC:

```
ROUTER2#ping 2001:db8:6::2
*May 18 06:16:27.719: %SYS-5-CONFIG_I: Configured from console by
console
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:6::2, timeout is 2
seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

## 2.3 RIP Konfiguration

Zuerst definiere ich einen RIP Namen (nvsripniklas) und dann schalte ich an jedem Interface, welches für RIP Routing verwendet soll, RIP mit v6 an. Zwischen mir und Thumberger war ein Netzwirkkabel gesteckt und nicht wie bei Teszar & mir ein seriellles Kabel, daher werden hier beide Gigabit Interfaces einmal für die Verbindung zwischen mir und Thumberger (g0/1) und einmal für den PC (g0/0).

```
ROUTER2(config)#ipv6 router rip nvsripniklas
ROUTER2(config-rtr)#exit
ROUTER2(config)#int g0/1
ROUTER2(config-if)#ipv6 rip nvsripniklas enable
ROUTER2(config-if)#exit
ROUTER2(config)#int s0/0/0
```

```
ROUTER2(config-if)#ipv6 rip nvsripniklas enable
% RIPng: IPv6 is not enabled on this interface.
ROUTER2(config-if)#exit
ROUTER2(config)#int s0/0/1
ROUTER2(config-if)#ipv6 rip nvsripniklas enable
ROUTER2(config-if)#exit
ROUTER2(config)#int g0/0
ROUTER2(config-if)#ipv6 rip nvsripniklas enable
ROUTER2(config-if)#exit
ROUTER2(config)#
```

### 2.3.1 Loopback Konfiguration

```
ROUTER2(config)#interface loopback 0
ROUTER2(config-if)#ipb
*May 18 07:02:43.183: %LINEPROTO-5-UPDOWN: Line protocol on
Interface Loopback0, changed state to
ROUTER2(config-if)#ipv6 rip nvsripniklas enable
% RIPng: IPv6 is not enabled on this interface.
ROUTER2(config-if)#ipv6 enable
ROUTER2(config-if)#ipv6 rip nvsripniklas enable
ROUTER2(config-if)#
```

### 2.3.2 Anzeigen der RIP Neighbors

Wie man an der Ausgabe sieht, bekomme ich meine eigenen netzwerke über directed connected angezeigt und die Netzwerke, die von beiden anderen advertised werden, angezeigt. Erkennen lässt sich dies durch ein R.

```
ROUTER2#show ipv6 route
IPv6 Routing Table - default - 10 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static
route
        B - BGP, R - RIP, I1 - ISIS L1, I2 - ISIS L2
        IA - ISIS interarea, IS - ISIS summary, D - EIGRP, EX - EIGRP
external
        ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr -
Redirect
```

```
O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF
ext 2
ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2, a - Application
C 2001:DB8:6::/64 [0/0]
    via GigabitEthernet0/0, directly connected
L 2001:DB8:6::1/128 [0/0]
    via GigabitEthernet0/0, receive
R 2001:DB8:17::/64 [120/2]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
R 2001:DB8:18::/64 [120/2]
    via FE80::FE5B:39FF:FEDA:20E1, GigabitEthernet0/1
R 2001:DB8:A1::/64 [120/2]
    via FE80::FE5B:39FF:FEDA:20E1, GigabitEthernet0/1
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
C 2001:DB8:A2::/64 [0/0]
    via Serial0/0/1, directly connected
L 2001:DB8:A2::2/128 [0/0]
    via Serial0/0/1, receive
C 2001:DB8:A3::/64 [0/0]
    via GigabitEthernet0/1, directly connected
L 2001:DB8:A3::1/128 [0/0]
    via GigabitEthernet0/1, receive
L FF00::/8 [0/0]
    via Null0, receive
```

### 2.3.3 Anzeige der IPV6 Konfiguration

Hier sieht man, für welche Interfaces v6 eingeschaltet ist, welches Routing Protocol verwendet wird.

```
ROUTER2#show ipv6 protocol

IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "application"
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "rip nvsripniklas"

Interfaces:

  Loopback0

  GigabitEthernet0/0

  Serial0/0/1

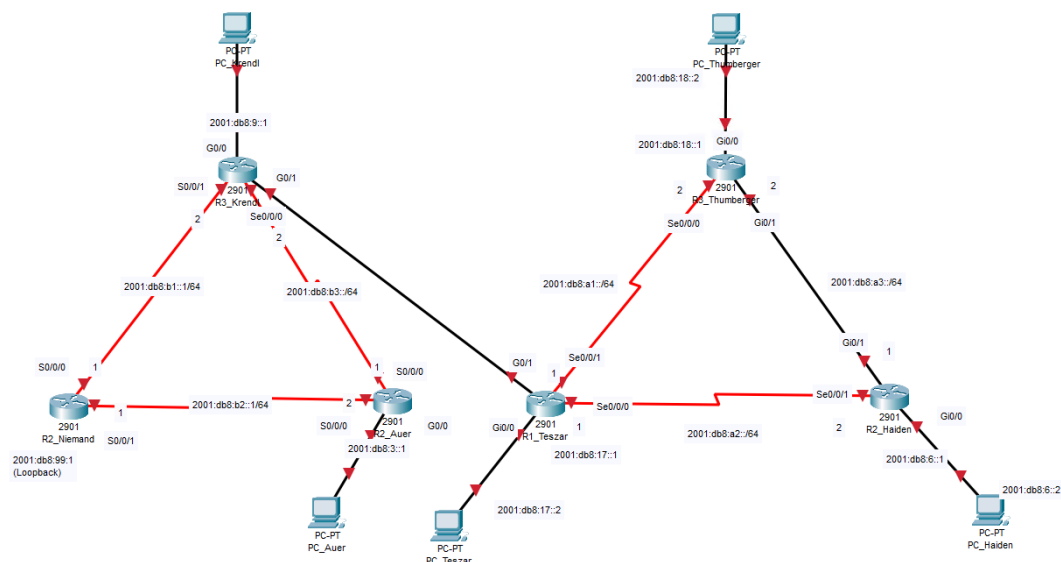
  GigabitEthernet0/1

Redistribution:

  None
```

## 2.4 Verbinden Sie die einzelnen Netzwerke miteinander. Können Sie die Routen untereinander austauschen?

### 2.4.1 Netzwerkksskizze



### 2.4.2 Aufbau

Alex & Kilian, hier links in der Skizze, hatten ein OSPF Netzwerk zu konfigurieren. Nun wollten wir beide Netzwerke mit unterschiedlichen Routing Protokollen miteinander vereinen. Angesteckt wurde ein Gigabit Ethernet Kabel an Thumbergers Router und einer an Krendls Router beide, über G0/1 angebunden. Danach hat Thumberger seinen Router so konfiguriert, dass RIP in Alex seinem OSPF Netzwerk verteilt wird, bzw. redistributed wird, danach hat Alex sein OSPF Netzwerk so konfiguriert, dass seine OSPF Advertisements in für RIP verständliche Pakete umgewandelt werden und so in

unserem Netzwerk verteilt werden. Durch die Konfiguration wandeln beidseitige die Routing Information jeweils so um, dass sie für deren Netzwerk, in denen es serviert werden soll, verständlich ist.

### 2.4.3 Anzeige der Routen nach Konfiguration

```
ROUTER2#show ipv6 route
R   2001:DB8:3::/64 [120/5]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
C   2001:DB8:6::/64 [0/0]
    via GigabitEthernet0/0, directly connected
L   2001:DB8:6::1/128 [0/0]
    via GigabitEthernet0/0, receive
R   2001:DB8:9::/64 [120/3]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
R   2001:DB8:17::/64 [120/2]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
R   2001:DB8:18::/64 [120/2]
    via FE80::FE5B:39FF:FEDA:20E1, GigabitEthernet0/1
R   2001:DB8:99::1/128 [120/5]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
R   2001:DB8:A1::/64 [120/2]
    via FE80::FE5B:39FF:FEDA:20E1, GigabitEthernet0/1
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
C   2001:DB8:A2::/64 [0/0]
    via Serial0/0/1, directly connected
L   2001:DB8:A2::2/128 [0/0]
    via Serial0/0/1, receive
C   2001:DB8:A3::/64 [0/0]
    via GigabitEthernet0/1, directly connected
L   2001:DB8:A3::1/128 [0/0]
    via GigabitEthernet0/1, receive
R   2001:DB8:B1::/64 [120/3]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
R   2001:DB8:B2::/64 [120/5]
```

```
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
R    2001:DB8:B3::/64 [120/3]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
R    2001:DB8:100::/64 [120/2]
    via FE80::FE5B:39FF:FEDA:2008, Serial0/0/1
L    FF00::/8 [0/0]
    via Null0, receive
```