

RIP konfigurieren

RIP, das Routing Information Protocol, ist eines der ältesten Routing-Protokolle. Die wichtigsten Fakten:

- RIP ist ein Distance Vector-Protokoll.
- RIPv1 arbeitet ausschließlich **classful** und unterstützt keine Authentifizierung
- RIPv1 überträgt die Routing-Updates über Broadcasts.
- RIPv2 kann(!) **classless** arbeiten und unterstützt sowohl Klartext- als auch MD5-Authentifizierung.
- RIPv2 überträgt die Routing-Updates über die Multicast-Adresse 224.0.0.9.
- Per Default wird alle 30 Sekunden die komplette Routing-Tabelle übermittelt.
- Die **Metrik** von RIP sind die Hop-Counts, maximal 15 Hops werden unterstützt.
- **Split Horizon** verhindert, dass Routen, die über ein Interface gelernt werden, über das gleiche Interface propagiert werden.
- **Route Poisoning** setzt die Metrik einer Route auf 16, um sie als unerreichbar zu markieren.
- Über Triggered Updates können Änderungen im Netzwerk unmittelbar kommuniziert werden.
- **Poison Reverse** ist die Bestätigung eines Nachbarrouters, dass die als unerreichbar gemeldete Route registriert wurde.
- Der **Holddown-Timer** (Default: 180 Sek.) verhindert Routing-Schleifen und –Inkonsistenzen, in dem keine anderen Routing-Informationen über als unerreichbar gemeldete Routen von anderen Routern als dem originären angenommen werden, der den Ausfall gemeldet hat.

Konfiguration von RIP

RIP konfigurieren:

```
Router(config)#router rip
```

Alle Interfaces einschließen, die im 10.0.0.0/8-Netzwerk eine IP-Adresse haben:

```
Router(config-router)#network 10.0.0.0
```

RIP unterstützt nicht die Angabe von Wildcards bzw. Subnetzmasken, es können immer nur die kompletten Klassen-Netze festgelegt werden!

RIP Version 2 aktivieren:

```
Router(config-router)#version 2
```

Classless-Routing aktivieren:

```
Router(config-router)#no auto-summary
```

Die maximale Anzahl gleichwertiger Routen für das Load-Balancing auf 2 festlegen (default: 4):

```
Router(config-router)#maximum-paths 2
```

RIP-Authentifizierung

Erstellen eines Schlüsselbundes namens STANDARD mit einem ersten Schlüssel, der "cisco" lautet:

```
Router(config)#key chain STANDARD  
Router(config-keychain)#key 1  
Router(config-keychain-key)#key-string cisco
```

Ein Interface für die MD5-Authentifizierung mit der Key-Chain STANDARD konfigurieren:

```
Router(config-if)#ip rip authentication mode md5  
Router(config-if)#ip rip authentication key-chain STANDARD
```

Der Key-String muss auf beiden Seiten identisch sein, der Name der Key-Chain ist frei wählbar.

RIP überprüfen

Die konfigurierten Routing-Protokolle anzeigen:

```
Router#show ip protocols
```

Die Routing-Tabelle anzeigen:

```
Router#show ip route
```

Alle über RIP gelernten Routen anzeigen:

```
Router#show ip route rip
```

RIP debuggen:

```
Router#debug ip rip
```

Erweiterte RIP-Konfiguration (CCNP+CCIE)

Redistribution für einen lokalen Prozess EIGRP 100 mit der Metrik 2 konfigurieren (Router hat zwei Routingprotokolle):

```
Router(config-router)#redistribute eigrp 100 metric 2
```

Distribute-List ausgehend auf S0 für das durch RIP gelernte Netzwerk 10.1.1.0/24 konfigurieren:

```
Router(config)#access-list 10 deny 10.1.1.0 0.0.0.255  
Router(config)#access-list 10 permit any  
Router(config-router)#distribute-list 10 out s0
```

Einen spezifischen RIP-Nachbarn 192.168.5.2 für Unicast-Updates festlegen:

```
Router(config-router)#neighbor 192.168.5.2
```

Einen Offset von 12 (hinzuzählen von 12 zur Metrik) für das Netzwerk 10.1.1.0/24 ausgehend festlegen:

```
Router(config)#access-list 10 permit 10.1.1.0 0.0.0.255  
Router(config-router)#offset-list 10 out 12
```

Timers für RIP auf 60 Sek. Update-Intervall, 200 Sek. Invalid, 240 Sek. Holddown, 360 Sek. Flush und festlegen:

```
Router(config-router)#timers basic 60 200 240 360
```