

# Routing

Bevor ein Router ein Paket weiterleitet muss er entscheiden, welcher Weg der „beste“ Weg zum Ziel ist. Somit auch bei welchem Interface (Netz) er das Paket rausschickt. Diesen Prozess nennt man Routing. Dazu nutzt der Router seine Routing-Tabelle. Diese baut er dynamisch (durch Routingprotokolle) oder durch statische Einträge auf. In der Praxis ist es meist eine Mischung aus beiden.

Der Router wählt jene Route, bei der am meisten Bits mit dem Ziel übereinstimmen. Dabei vergleicht er die Route mit der Destination IP-Adresse.

## 1 Einträge in der Routing Tabelle

- Direkt verbundene Netze: Aktive und angeschlossene Netze am Router mit IP- Konfiguration, diese fügt der Router automatisch hinzu (Status Code: C, L)
- Remote Netze: Werden statisch oder dynamisch (vom Routingprotokoll) eingetragen. (Status Code: S, R, O, E, ...)
- Default Route (Gateway of last Resort): Next Hop falls der Router keine passende Route findet. Kann statisch oder dynamisch gelernt werden. Route: 0.0.0.0/0 ... 0-Bits müssen übereinstimmen.

## 2 Eintrag in der Cisco CLI

Beispiel:

R	30.0.4.0/24	[120/7]	via 10.0.3.2	00:13:29	Serial0/1/1
Status	Ziel	AD/Metrik	IP (ausgehendes Interface)	Zeitstempel	Interface

Die einzelnen Abschnitte werden in den folgenden nächsten Abschnitte genauer beschrieben.

### Status Code

Der Status Code gibt an woher der Router die Route kennt.

C ...	Connected	Direkt verbundenes Netz
L ...	Local	IP vom Interface, lokale Route
S ...	Static	statisch eingegebene Route
R ...	RIP	entsprechendes Routingprotokoll
O ...	OSPF	entsprechendes Routingprotokoll
E ...	EIGRP	entsprechendes Routingprotokoll

### Ziel

Dort steht die IP-Adresse des Zielnetzes mit einem Präfix (entspricht nicht unbedingt der Subnetzmaske).

Beim Ziel müssen mindestens die angegebene Anzahl an Bits (Präfix) mit der Destination IP-Adresse übereinstimmen, damit die Route in Frage kommt.

Gewählt wird jene Route bei der am meisten Bits (von links) übereinstimmen.

Problem: Im Packet wird bei der Destination IP-Adresse keine Subnetzmaske mitgeschickt. Diese ist normalerweise auch nicht bekannt.

Beispiel: Destination IP 172.16.0.10 (letztes Oktette: 00001010)

Routen:

1. 172.16.0.0 /16 Bis Bit 16 stimmt es mit dem Ziel überein
2. 172.16.0.0 /24 Bis Bit 24 stimmt es mit dem Ziel überein
3. 172.16.0.0 /26 (letztes Oktette 00000000 - Bis Bit 26 stimmt es mit dem Ziel überein)
4. 172.16.0.0 /30 (letztes Oktette 00000000 - Bit Nummer 29 stimmt nicht überein)
5. 172.17.0.0 /24 (am zweiten Oktette stimmt es nicht mit dem Ziel überein)

Hier würde sich der Router für die dritte Route entscheiden. Dort stimmt die angegebene Anzahl der Bits (Präfix) überein. Bei Route 4 und 5 wäre dies nicht der Fall. Bei Route 1 bis 3 passt die Anzahl und von daher nimmt er jene Zeile wo am meisten Bits übereinstimmen.

### AD: Administrative Distanz

Der Router kann eine Route auf mehrere Arten lernen (z.B. statisch, RIP, OSPF, ...). Die AD gibt an, wie „vertrauenswürdig“ eine Route ist. Der Router nimmt jene Route mit der niedrigsten AD. Die anderen Route sind Backups und werden vorerst nicht in der Routing-Tabelle angezeigt. Erst wenn die „beste“ Route ausfallen würde, würde er auf die nächst beste zurückgreifen.

Standard Werte bei Cisco Routern:

	AD
Direkte Routen	0
Statische Routen	1
Internat EIGRP	90
OSPF	110
RIP	120

### Metrik

Von einem Routingprotokoll kann der Router mehrere Routen zum gleichen Ziel lernen. Die Metrik gibt an, „wie weit“ das Ziel entfernt ist.

Der Router nimmt die Route mit der niedrigsten Metrik. Falls eine Route ausfällt würde auch hier der Router auf Backup-Routen zurückgreifen.

## 3 Statische Routen

Sind in kleineren Netzen, die sich kaum ändern, sinnvoll. Für spezielle Zielnetze kann es auch sinnvoll sein, einzelne statische Routen zu setzen. Für Router die nur einen Nachbar haben (Stub-Network) machen statische Default-Routen Sinn.

Problem: Diese werden nicht automatisch aktualisiert. Änderungen müssen somit händisch

vorgenommen werden.

Konfiguration: `ip route Ziel-IP Präfix Interface IP/Interface (oder beides) [AD]`

Beispiel: `ip route 192.168.10.0 255.255.255.0 s0/1/0 5`

## 4 Dynamische Routingprotokolle

Je nach Ablauf des Routingprotokolls unterscheidet man unterschiedliche Kategorien.

- **Pfadvektorprotokolle**

Diese Protokolle speichern den Pfad/Weg zum Ziel, sie sind besonders effizient gegen Routing-Schleifen. Dadurch eignen sie sich besonders zum Routen zwischen autonomen Systemen.

Beispiel: BGP (Border Gateway Protocol) - Metrik: Anzahl der autonomen Systeme bis zum Ziel (Zusatzinformation IGP-Metrik: wie lange dauert es durch ein autonomes System)

- **Distanzvektorprotokolle** Diese Routingprotokolle speichern nur die Distanz zum Ziel

Beispiele: RIP (Routing Information Protocol) - Metrik: Anzahl der Hops

EIGRP (Enhanced Interior Gateway Routing Protocol) - Metrik: Bandbreite (b), Auslastung (a), Delay (d), Zuverlässigkeit (z)

$$\text{Metrik} = \left( k_1 \cdot b + \left( \frac{k_2 \cdot b}{256} - a \right) + k_3 \cdot d \right) \cdot \left( \frac{k_5}{z + k_4} \right)$$

$k_1, k_2, k_3, k_4 \dots k_5$  sind Konstanten die man beliebig anpassen kann.

EIGRP kennt auch die ganze Topologie im System speichert diese aber nicht direkt ab.

- **Link-State-Protokolle** Diese Routingprotokolle kennen die ganze Topologie im System. Daraus berechnet sich jeder Router die besten Routen zu allen Zielen.

Beispiel: OSPF (Open Shortest Path First) - Metrik: Bandbreite

## 5 Algorithmen zur Bestimmung des kürzesten Weges

- Bellman Ford (RIP)
- Dijkstra (OSPF)
- DUAL (EIGRP)

### Dijkstra Algorithmus

Ablauf:

1. Startknoten mit 0 markieren, alle anderen mit  $\infty$ :  
„Distanz“ und „besucht“ merken
2. Solange es unbesuchte Knoten gibt:
  - Jenen Knoten mit der kürzesten Distanz wählen
  - Als besucht markieren
  - Für alle unbesuchten Knoten die Distanz berechnen
  - Falls der Wert kleiner ist, als der aktuelle, diesen speichern