Routingprotokolle

In Netzwerktoplogien, die nicht "schleifenfrei" sind, muss es Möglichkeiten geben, mit Hilfe derer Router bestimmen können, welches der beste Weg zu einem Zielrechner ist, dabei müssen selbstverständlich auch Wege über mehrere andere Router berücksichtigt werden.

Es existieren mehrere Protokolle, mit denen Router Informationen austauschen können, um ihrerseits ihre Routingkonfiguration zu optimieren - diese Falle grob in zwei Kategorieren: **Distanzvektor** Routing-Protokolle und **Link-State** Routing-Protokolle.

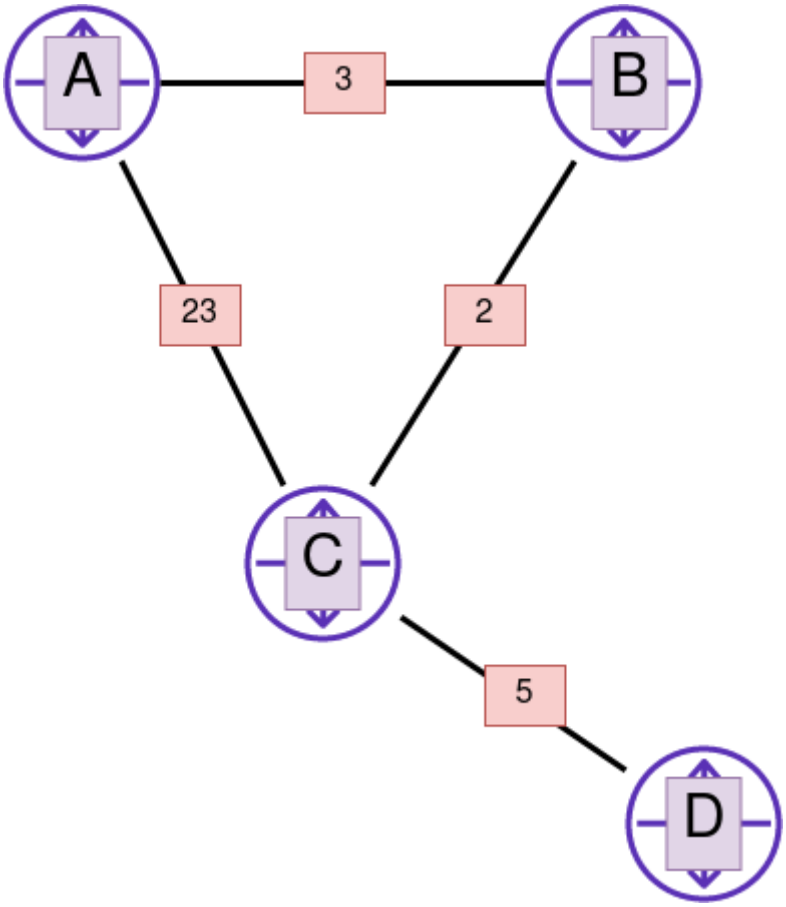
Link-State Protokolle verwenden den Dijkstra-Algorithmus. damit kann sich ein Router einen vollständigen Überblick über das ihn umgebende Netz verschaffen. Beispiele für Link-State Protokolle sind *OSPF (Open Shortest Path First)* oder *IS-IS (Intermediate System to Intermediate System Protocol)*. Da uns der Dijkstra-Algorithmus später ncohmal genauer beschäftigen wird, beschränken wir uns hier auf die Distanzvektor Protokolle.

Distanzvektor Routing-Protokolle basieren auf dem Bellman-Ford Algorithmus. Damit wird der "kürzeste" Weg ausgehend von einem Startknoten bestimmt, wobei die Verbindungen zwischen den Knoten verschiedene "Kosten" haben können. Der "kürzeste" oder "beste" Weg ist der, mit den geringsten Kosten.

Distanzvektor Routing-Protokolle geben in regelmäßigen Abständen (ca. alle 30 Sekunden und bei einer Änderung der Topologie) eine Kopie der eigenen Routing-Tabelle an ihren Nachbarn weiter. Auf diese Weise "wandern" Topologie-Änderungen durch das Netz.

Den Austausch der Routing-Tabellen kann man sich so vorstellen: Router A gibt seine Informationen zu Router B weiter. Router B gleicht die neuen Informationen von Router A mit den ihm bekannten Informationen ab und fügt schließlich seine Distanzvektorkosten (z.B. Anzahl der Hops) hinzu. Nun gibt Router B die „aktualisierte“ Routing-Tabelle an Router C weiter. Dieser Vorgang wiederholt sich bei jedem benachbarten Router.

Schritt für Schritt



T=0: Knoten A erzeugt seine initiale Kostenmatrix. Sie enthält nur unsere direkten Nachbarn B und C mit den uns bekannten Kosten. Wir schicken daraufhin unsere neuen besten Pfade (B mit Kosten 3, C mit Kosten 23) an unsere direkten Nachbarn. Die anderen drei Knoten machen dasselbe aus ihrer Sicht.

A_0	via A	via B	via C	via D	DV
nach A					
nach B		**3**			3
nach C			**23**		23
nach D					

B_0	via A	via B	via C	via D	DV
nach A	**3**				3
nach B					
nach C			**2**		2
nach D					

C_0	via A	via B	via C	via D	DV
nach A	**23**				23
nach B		**2**			2
nach C					
nach D				**5**	5

D_0	via A	via B	via C	via D
nach A				
nach B				
nach C			**5**	
nach D				

Beispiel: https://de.wikipedia.org/wiki/Distanzvektoralgorithmus

From:
https://www.info-bw.de/ -

Permanent link:
https://www.info-bw.de/faecher:informatik:oberstufe:netzwerke:wegedurchsnetz_i:start?rev=1602524992

Last update: **12.10.2020 17:49**