

Routing

statisch: Tabelle

Dieses Verfahren ist *nicht adaptiv*, sehr einfach und daher viel benutzt. Jeder Knoten unterhält eine Tabelle mit einer Zeile für jeden möglichen Zielknoten. Eine Zeile enthält Einträge, welche die beste, zweitbeste usw. Übertragungsleitung für dieses Ziel ist, zusammen mit einer Gewichtung. Vor Weiterleitung eines Paketes wird der entsprechende Eintrag aus der Tabelle gewählt und auf eine der möglichen Leitungen gegeben. Die Gewichtung spiegelt hier die Wahrscheinlichkeit wider, dass diese Leitung gewählt wird.

Dynamisch:

Verteiltes adaptives Routing Bei diesem Verfahren tauscht jeder Knoten periodisch Routing-Informationen mit jedem seiner Nachbarn aus. Auch hier unterhält jeder Knoten eine Routing-Tabelle, die für jeden anderen Knoten im Netz einen Eintrag enthält. In dieser Tabelle sind die bevorzugte Übertragungsleitung für diesen Knoten sowie eine Schätzung zu Zeit oder Entfernung zu diesem Knoten enthalten:

- Anzahl Hops
- Geschätzte Verzögerung in Millisekunden
- Geschätzte Gesamt-Anzahl von Paketen, die entlang des Weges warten

Diese Schätzungen werden gewonnen aus der Zeit/Entfernung zu den Nachbarn (z. B.: mittels speziellen Echo-Paketen mit Zeitstempel) und/oder Schätzungen der Nachbarn. Ein Austausch der Routing-Informationen kann entweder synchron in bestimmten Aktualisierungsintervallen oder asynchron bei signifikanten Änderungen erfolgen. Zu diesem Verfahren gehören unter anderem das

- Distance Vector Routing
- Link State Routing

Distance Vector Routing Ein verteiltes, adaptives Routing, welches als RIP früher im Internet benutzt wurde. Hierbei speichert jeder Router eine Tabelle mit der besten Entfernung (z. B.: Anzahl hops, Verzögerung in ms) zu jedem Ziel und dem dazugehörigen Ausgang. In der Praxis hat dieses Verfahren eine zu langsame Konvergenz zu einem konsistenten Zustand für viele Router, aufgrund der „Count-To-Infinity“-Problematik.

Link State Routing Ein verteiltes, adaptives Routing, welches als OSPF und IS-IS im Internet eingesetzt wird. Dabei findet folgender Algorithmus Anwendung:

- Entdecken neuer Nachbarn mittels HELLO-Paket
- Messung der Verzögerung bzw. der Kosten zu jedem Nachbarn, mittels ECHO-Paket
- Erstellung eines LINK-STATE-Pakets mit allen gelernten Daten (Sender, Liste der Nachbarn mit Verzögerung, Alter, ...), welches periodisch oder ereignisgesteuert (z. B.: neuer Nachbar, Ausfall, ...) erzeugt wird
- Versenden dieses Pakets an alle Nachbarn (prinzipiell mittels Fluten, aber mit Verfeinerung: Vernichten von Duplikaten, Zerstören der Information nach gewissem Alter etc.)
- Berechnung des kürzesten Pfades zu allen anderen Routern (z. B.: Dijkstra)

Dieses Verfahren ist sehr rechenaufwendig, aber Optimierungen dieses Verfahrens existieren und gehören dann zu der jeweiligen Topologie usw. des Netzes.

OSPF: costs (Kosten) billigster (schnellster) Weg wird genommen

Netzwerk - 07.11.08 Routing

SYN Verbindung aufbauen

FIN Verbindung abbauen

ACK Schickt die Nummer des nächsten erwarteten Pakets zurück