IPv6 - die neue IP-Version

- Probleme mit dem klassischen IP ("IPv4"):
  - zu wenig Adressen (sollte nicht am besten jeder TV-Settop, jeder Stromzähler und jeder Toaster seine eigene IP-Adresse bekommen?)
  - Anzahl der am Internet angeschlossenen Netze verdoppelt sich z.Z. mindestens jährlich; weiteres Wachstum weniger im Computer-Bereich als vielmehr bei "networked entertainment", der Fernsteuerung von Geräten in Gebäuden und Haushalten etc.
  - schlecht für "Hochgeschwindigkeit" (zu viel Aufwand in den Routern, z.B. Prüfsumme berechnen, Längenfeld evaluieren...)
  - schlecht geeignet für neuere Anforderungen wie Realzeitfähigkeit, Multicasting, Sicherheit...

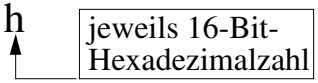
- Wesentliche Eigenschaften von IPv6:
  - 128-Bit-Adressen
  - Multicast-Möglichkeit
  - realzeitfähig
  - optionale Authentifizierung und Verschlüsselung
  - vereinfachter Header mit weniger Feldern (--> Beschleunigung)
  - optionale Erweiterungsmöglichkeiten (--> "offenes Design")
  - Unterstützung zur Autokonfiguration von Netzadressen neuer Geräte

- Koexistenz (über viele Jahre) mit IPv4
  - Versionsnummer (4 bzw. 6) steht am Anfang beider Header
  - nach und nach Router, die IPv4 und IPv6 können ("dual stack")
  - ggf. Tunneln von IPv6 über länger IPv4-Strecken und Teilnetze

IPv6: Adressformat

- 128 Bit (=16 Byte) lange Adressen (statt 32 Bit in IPv4)
  - Adressraumgrösse $2^{128} \approx 10^{38}$ --> ca. 10^{24} Adressen pro m^2 der Erde
  - 2^{96} Mal mehr als bei IPv4
  - letzten 48 Bits könnten u.U. die MAC-Adresse sein

- Notation: a:b:c:d:e:f:g:h



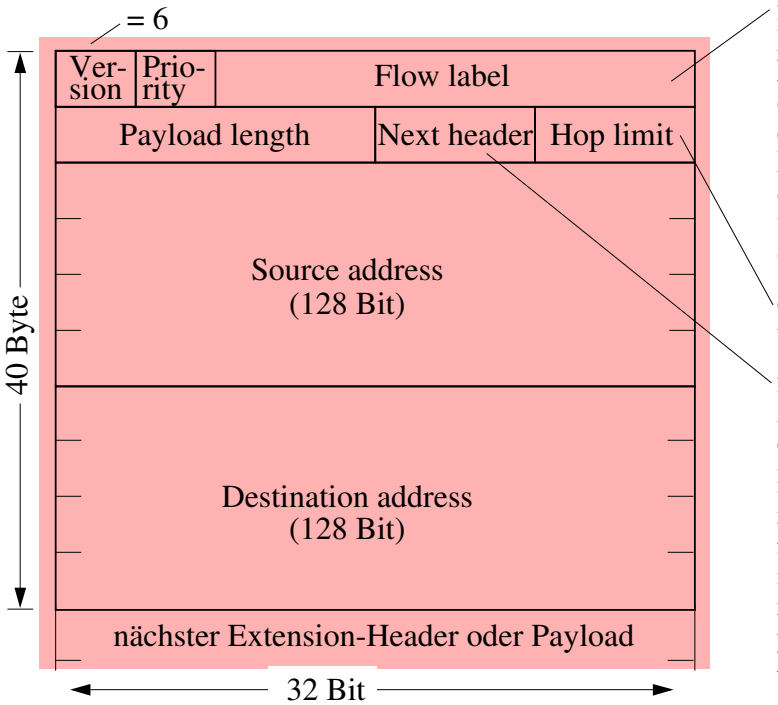
  - klassische IP-Adresse ("IPv4") jetzt so: ::128.83.5.72

- Jedes Kommunikationsinterface eines Knoten (Rechner, Router,...) kann *mehrere* IPv6-Adressen haben

- Lange Adressen mit hierarchischer Gliederung erlaubt in Routern die Aggregation von Adressbereichen
  - z.B.: "alles, was den Präfix xxx hat zu Knoten y"
  - dadurch kleinere Routingtabellen

- Neuer Vorschlag: *Anycast-Adressen*
  - Zieladresse spezifiziert das Ziel nur "vage"; die Router suchen sich dann einen besten / nächsten Knoten, der zu dieser Adresse passt (z.B. für Auskunftsdienste, mobile Stationen, lokale Filiale von McDonalds, ein Router eines Subnetzes...)

IPv6: Header



für Folge von Paketen zu einem Empfänger, für die der Sender eine spezielle Behandlung bei den Routern wünscht ("quality of service")

entspricht time to live bei IPv4

für zusätzliche Angaben in einem (oder mehreren) Headern am Ende dieses Headers; letzter Header nennt hier das Transportprotokoll (z.B. TCP oder UDP)

- Priority (4 Bits); zwei unabhängige Klassen:
  - 8-15 für Realzeit; z.B. Audio oder Video (Sender kann nicht gebremst werden, notfalls Datenpaket wegwerfen)
  - 1-7 falls Sender mittels Laststeuerung gebremst werden kann (z.B. 1 für news, 2 für Email, 4 für ftp / http, 6 für telnet, 7 für Routinginfo)
- Hop limit: reichen 255 hops für alle Zukunft aus?
  - was ist z.B. mit langen Nachsendeketten bei mobilen Stationen?
- Payload length: Genügt eine Maximallänge von 2^{16}?
  - ein Supercomputer möchte z.B. nicht alle 65535 Bits unterbrochen werden, wenn sehr grosse Datenmengen schnell zu versenden sind
  - hierfür ist bereits eine Lösung vorgesehen: "Jumbogramme"

IPv6: Flow Label

- Beispiel Multimedia-Konferenz (typischerweise mit Multicast-Adresse): getrennte flows für
  - Audio
  - Video
  - Graphik

  } unterschiedliche Anforderungen an Bandbreite, Delay, Varianz...
- Aus Sicht eines Routers haben alle Pakete eines flows identische Anforderungen z.B. hinsichtlich
  - maximale Verzögerung
  - Priorität
  - accounting
  - Sicherheit
  - Routing-Weg

  } Router kann Ressourcen für den flow einmalig reservieren (Puffer, Bandbreite, Service-Qualität von Subnetzen etc.) und braucht nicht immer alle Felder aller (Extension-) Header auszuwerten
- Quelle wählt eine flow-Nummer zufällig aus dem Bereich 1...2^{24}-1
  - natürlich verschieden von anderen flows der gleichen Quelle
  - keine "Kollisionen", da durch Quelladresse eindeutig

IPv6: Extension-Header

- Optionale weitere Header (angehängt an den Haupt-Header) für verschiedene Zwecke, z.B.:

- für Datenpakete > 64k ("Jumbogram" bis 2^{32} Byte)

- für Routing-Unterstützung (z.B. Source-Routing)

- Authentifizierung

  - Idee: Felder im Header, die sich unterwegs nicht ändern, mit einem Geheimschlüssel (nur Sender und Empfänger bekannt) konkatenieren und darauf kryptographische Prüfsumme (z.B. MD 5) anwenden

-Verschlüsselung

  - Felder für Schlüsselkennung etc. vorgesehen, Verschlüsselungsverfahren selbst ist aber nicht festgelegt (Default: DES mit cipher block chaining)

  - Verschlüsselung wurde bewusst getrennt von der Authentifizierung gehalten, da es bzgl. Verschlüsselung ggf. politische Vorbehalte gibt (Verbot der Anwendung bzw. des Exports), die bzgl. Authentifizierung i.a. nicht bestehen

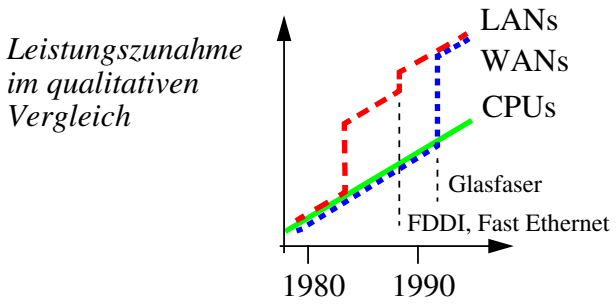
-Destinations options

  - Für Informationen, die nur den Empfänger betreffen (brauchen die Router nicht zu betrachten)

IPv6: Routing-Effizienz

- Routing soll effizient sein

  - Routen werden tendenziell eher länger (viele Provider...)
  - Bandbreite von Netzen wächst schneller als cpu-Leistung



- Header hat konstante Grösse (40 Byte)

  - kein Längenfeld, das ausgewertet werden muss

- Keine Berechnung von Prüfsummen (wie in IPv4)

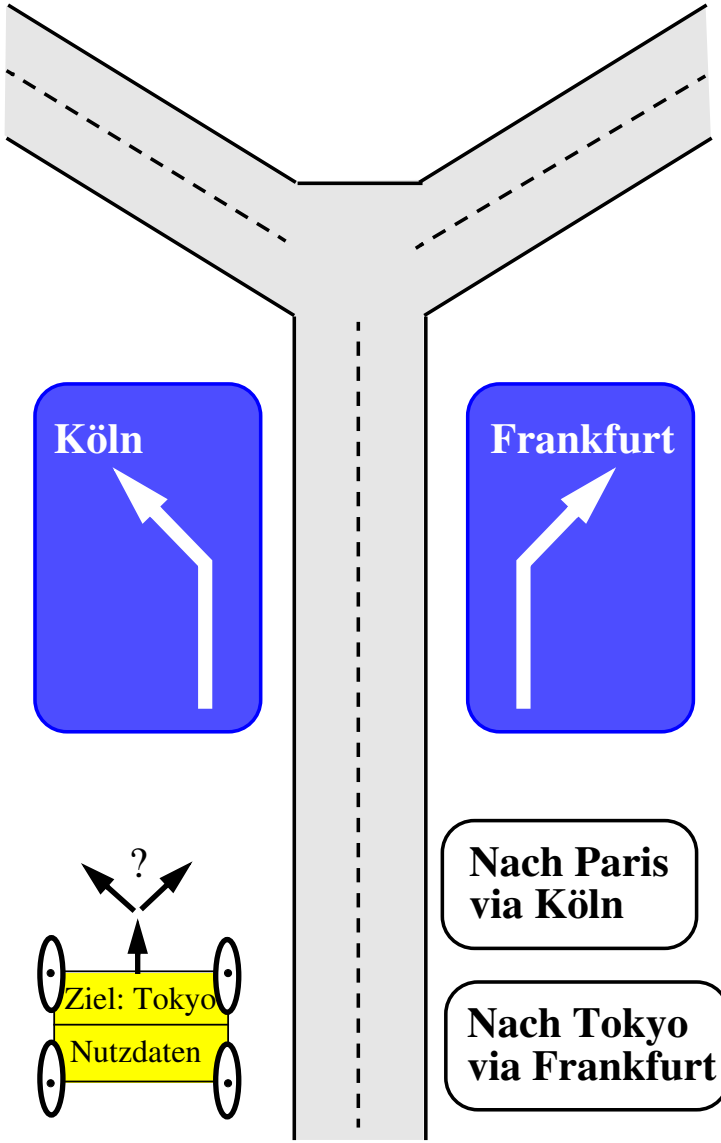
  - wesentlicher Zeitgewinn!
  - Kritik: Entfernen der Bremsen macht ein Auto auch leichter und schneller
  - Rechtfertigung: Prüfsummen werden sowohl auf der tieferen als auch der höheren Ebene ("end-to-end-Argument") verwendet

---

- Zu IPv6 lese man folgenden Artikel: *Robert M. Hinden: IP Next-Generation. Commun. of the ACM, Vol 39 No 6 (Jun. 1996), 61-71*

- Mehr zu IPv6 auch im Internet (auf den "offiziellen" Seiten der Arbeitsgruppen)

Routing (1)



Ein Knoten-punkt der Datenautobahn

Routingtabelle eines Netz-knotens

Problem: Wer stellt die Wegweiser auf?

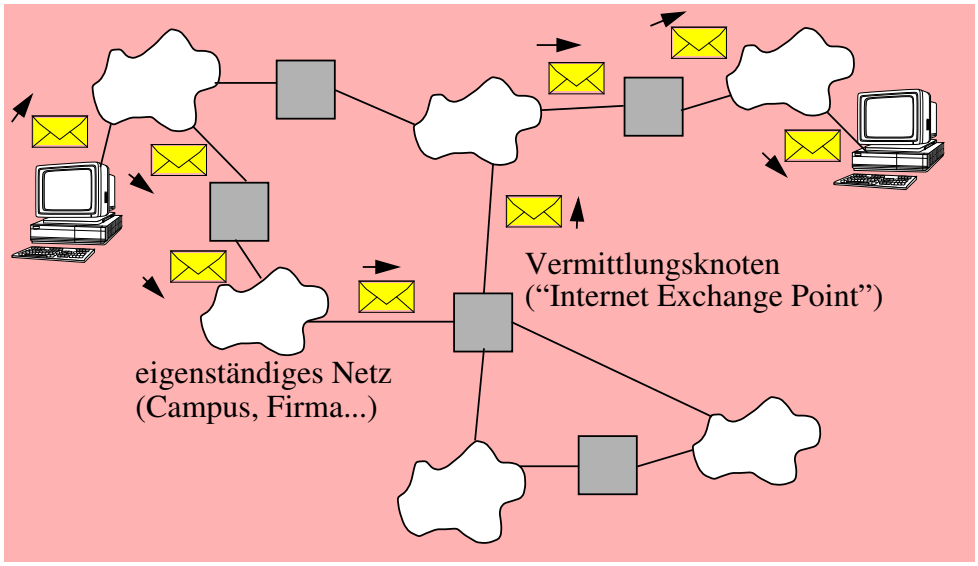
Routing (2)

- Jede Nachricht hat eine *Zieladresse* und muss ihren Weg durch das vermaschte Netz finden

Nachricht als Datenpaket

Daten (Bits)	Zieladresse

  - Problem: zyklenfreier, kurzer, schneller, billiger Weg



- Wegbestimmung meist mit *Optimierungszielen*, z.B.:
  - gleichmässige Auslastung der Leitungen
  - kurze Wartezeiten
  - hoher Durchsatz
  - Wege mit geringer Knotenzahl
  - Robustheit (ggf. alternative Wege bei Störungen)
  - geringe Kosten
  - geringe Fehlerrate

Routing (3)

- Unterscheide strenggenommen:
  - Ermittlung von Routing-Tabellen (eigentliches Routing)
  - Weiterleitung einer Nachricht mittels Routing-Tabellen (“forwarding”)

- Unterscheide:
  - Routing-Entscheidung wird beim Sender getroffen (“Source-Routing”)
  - Routing-Entscheidung wird bei den Knoten unterwegs getroffen

- Unterscheide:
  - “virtual circuit”-Verbindungen auf dem network layer
    --> Routing nur bei Aufbau der Verbindung
  - “connection-less”-Verbindungen
    --> dediziertes Routing (eigentlich “forwarding”) für jedes einzelne Paket (“Datagram”)
    --> Pakete können auf “parallelen” Pfaden laufen (Reihenfolge?!)

- Routing kann bei kleinen Fehlern zu Disastern führen!
  - Bsp.: Bedienungs- bzw. Konfigurationsfehler führt dazu, dass ein Knoten meint, jeden anderen Knoten (im Internet) mit Kosten 0 erreichen zu können
  - er erhält damit eine grosse Menge von Datenpaketen
  - kann mit diesen (und dieser Menge) aber nichts anfangen und ist überlastet
  - viele Datenpakete verschwinden in diesem “attraktiven” schwarzen Loch

Routing (4)

- Wir kennen bereits einige Routing-Grundverfahren
  - broadcast oder flooding (eigentlich *kein* Routing)
  - Aufbau eines Spannbaums (einige Verbindungen bleiben ungenutzt)
  - Vector-Distance-Verfahren zum Aufbau der Routing-Tabellen
    - da Austausch ganzer Tabellen: eher für kleine Netze geeignet
    - verwendet bei RIP (“Routing Information Protocol”, war Teil der BSD-UNIX-Distribution und wurde im Internet eingesetzt)
    - RIP: autonomes Versenden der Tabellen an alle Nachbarn alle 30s; getriggertes Versenden durch Änderung der eigenen Tabelle
    - RIP war relativ primitiv: nur Hopcount als Metrik, max. 15 Hops (neuere Version: RIP-2)

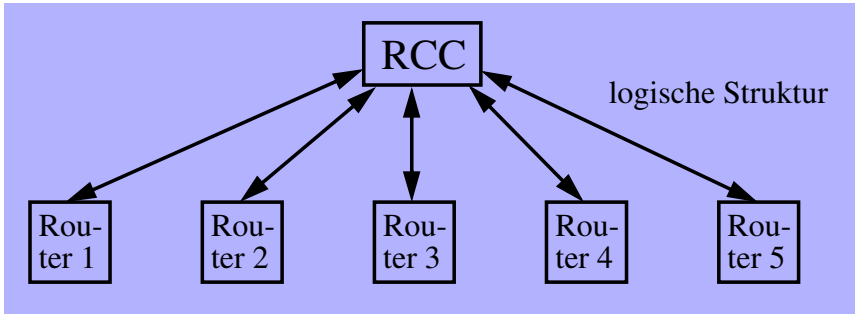
- Beachte: Routing im Internet auf IP-Ebene basiert nur auf netid-Teil der Adresse, nicht auf hostid-Teil
  - gesucht wird in der Routing-Tabelle der Eintrag mit der längsten Übereinstimmung bzgl. der Adresse
  - Bsp.: Wenn die Tabelle einen Eintrag zu 130.32.1.0 und zu 130.32.0.0 enthält, wird bei der Zieladresse 130.32.11.1 der erste Eintrag gewählt

Zentralisierte Leitwegbestimmung

- Knoten senden in periodischen Abständen und bei wichtigen Veränderungen die relevanten Informationen einem "Routing Control Center" (RCC)

- Das RCC berechnet mit globaler Kenntnis optimale Wege (z.B. mit Kürzeste-Wege-Algorithmus)

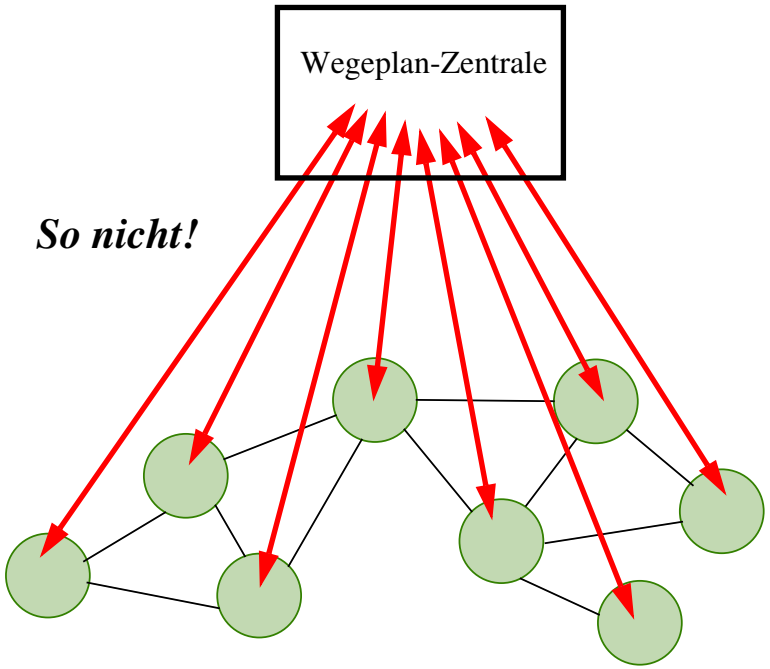
- Relevante Information wird dann wieder vom RCC an die einzelnen Knoten verteilt



- Vorteil: Konsistenz leichter durch globale Sicht

- Aber: Information oft überaltet; Verfahren nicht fehlertolerant; RCC bildet einen Engpass (schlecht skalierbar)

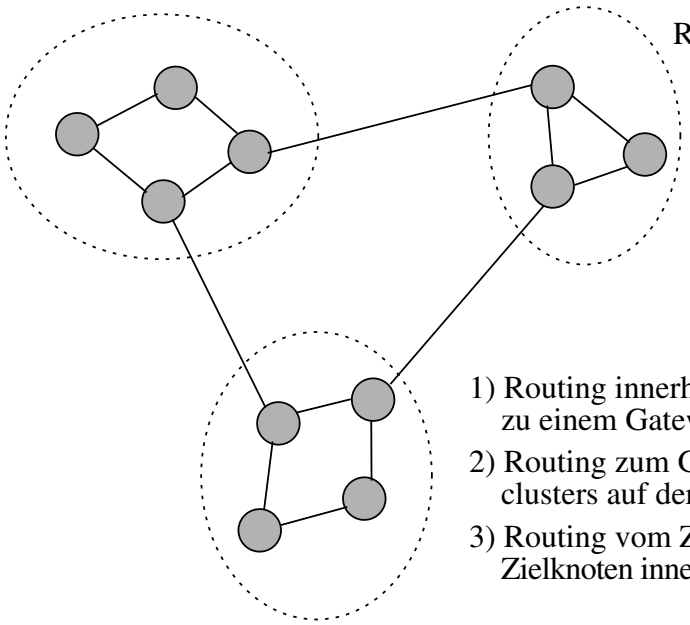
Wieso im Internet dezentral?



- Es gibt keinen Gesamtplan des Internet
- Zentrale Berechnung aller "Wegweiser" würde zu lange dauern
  - alle müssten bereit sein, Wegedaten an die Zentrale zu liefern und Routinginfomation von dort entgegenzunehmen
- Zentrale wäre störungsempfindlich
  - Sicherheitsrisiko, Probleme bei Ausfall oder Fehlverhalten
- Internet ist traditionell "anarchisch" organisiert

Hierarchisches Routing

- Hierarchie unterstützt Skalierbarkeit
  - Aufwand für Berechnung der Routing-Tabellen wächst (sonst) nichtlinear mit der Anzahl der Knoten
  - Router brauchen nur ihr eigenes Cluster genau zu kennen, nicht aber die interne Struktur anderer Cluster

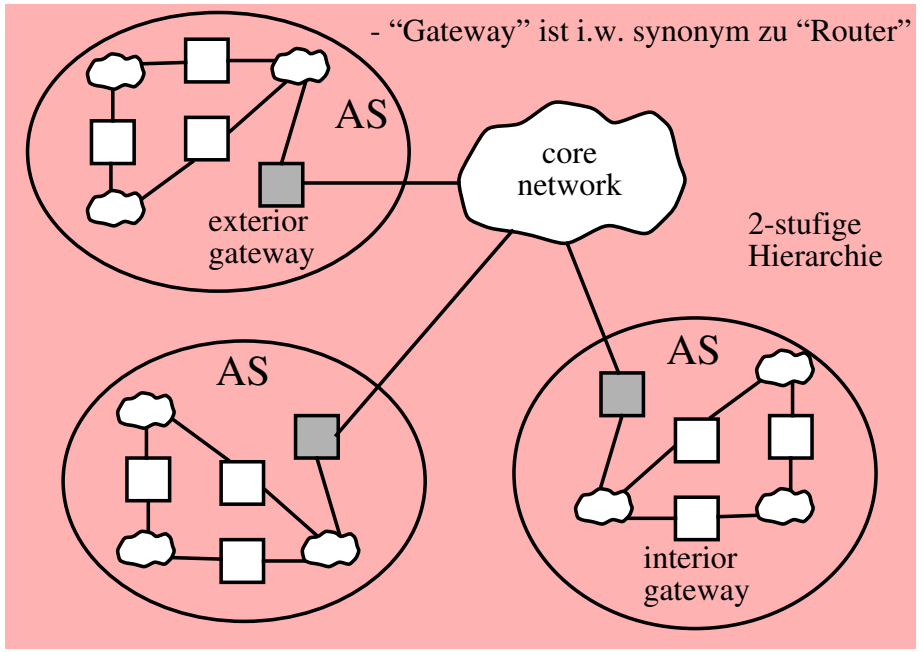


1) Routing innerhalb eines Clusters zu einem Gateway ("Repräsentant")
2) Routing zum Gateway des Zielclusters auf der höheren Ebene
3) Routing vom Zielgateway zum Zielknoten innerhalb des Zielclusters

- Vorteile:
  - Topologieänderungen und Änderungen der Kosten haben oft nur lokale Auswirkungen (Update von Tabellen etc.)
  - Routing-Tabellen sind kleiner

- Nachteile:
  - Setzt i.w. eine hierarchischeAdressstruktur voraus (vgl. Telefonnetz)
  - ggf. längere Routen als eigentlich nötig

"Autonome Systeme" (AS) im Internet



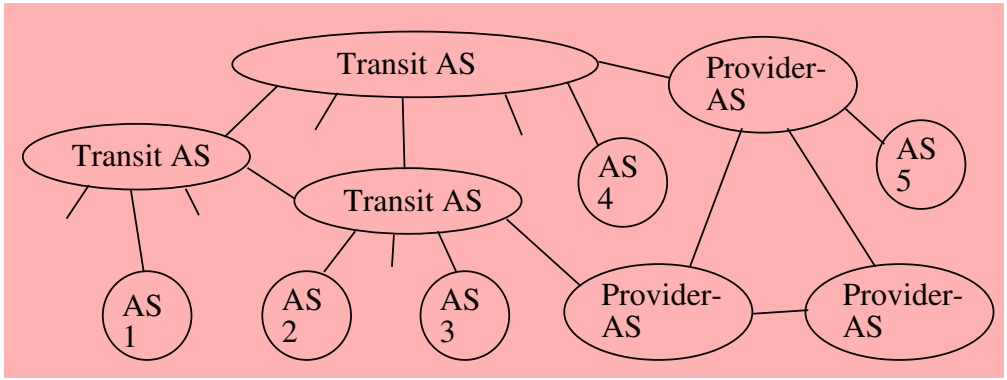
- "Autonomes System": Gruppe von IP-Netzen, welche untereinander ein bestimmtes Routing-Verfahren einsetzen (und i.a. gemeinsam verwaltet werden)
  - jedes AS (bzw. "routing domain") hat eine Nummer (16 Bit)
- AS haben sich i.a. historisch unterschiedlich entwickelt
  - daher oft verschiedene Routing-Strategien in verschiedenen AS
- Interior Gateway Protocol (IGP) für Routing innerhalb
  - z.B. RIP, OSPF ("Open Shortest Path First")
- Exterior Gateway Protocol (EGP) - Standard im Internet
  - derzeit BGP ("Border Gateway Protocol")

Autonome Systeme

- Auszug aus der Liste der Institutionen
  - es gibt derzeit einige tausend autonome Systeme
  - Nummern von der IANA (“Internet Assigned Numbers Authority”) verwaltet

11 HARVARD	1744 LOCKHEED
26 CORNELL	1748 FINNAIR
32 STANFORD	1754 DESY-HAMBURG
71 HP	1893 DKRZ-HAMBURG
108 XEROX	2022 SIEMENS
163 IBM-RESEARCH-	2278 FRANCE-TELECOM
294 FRANCE-IP-NET	2535 BP
513 CERN	3680 NOVELL
517 XLINK-UKA	3958 AIRCANADA
553 BELWUE	4179 CITY-OF-LA
680 WIN	4183 COMPUSERVE
715 APPLE	6142 SUN-JAVA
1248 NOKIA	6307 AMERICAN-EXPRESS
1275 DFN	6431 ATT-RESEARCH

- Typischerweise drei Klassen von AS:
  - Institutionen mit einer einzigen Verbindung zum restlichen Graph
  - regionale Internet-Provider; Institutionen mit Peering-Abkommen (Durchleiten von fremdem Datenverkehr i.a. nur beschränkt)
  - “Transit AS”: Internet-Backbone hoher Kapazität



Link-State-Routing-Verfahren (1)

- Jeder Knoten (= Router) kennt den Zustand der Verbindung (“link state”) zu jedem seiner Nachbarn
  - sowie die Kosten (delay etc.)

- Dieses Wissen wird allen(!) Knoten im Netz mitgeteilt
  - nicht nur den Nachbarn
  - Information wird dazu in ein Link-State-Paket (LSP) verpackt

- LSP enthält
  - ID des Erzeugerknotens
  - Kosten (bzw. ∞ bei “down”) zu allen direkten Nachbarn
  - Sequenznummer
  - Time-to-live (TTL)

- LSP wird mit “reliable flooding” verteilt
  - eigenes LSP periodisch versenden (Sequenznummer jeweils erhöhen)
  - ein Empfänger merkt sich das jeweils neueste eines Absenders
  - ein neueres empfangenes LSP wird allen (anderen) Nachbarn mitgeteilt
  - alle LSPs werden mit einer Ack-Nachricht bestätigt

- Alterungsprozess:
  - TTL jedes gespeicherten LSPs wird langsam heruntergezählt
  - TTL wird stets dekrementiert, bevor LSP weitergeleitet wird
  - falls TTL 0 wird: LSP löschen, aber an Nachbarn weiterleiten (diese löschen dann auch etc.)

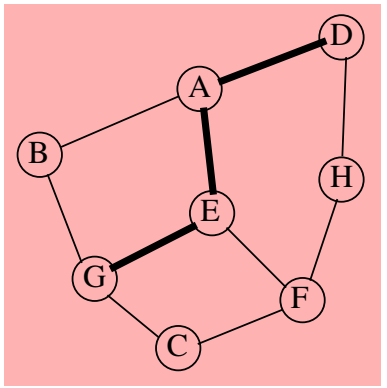
Link-State-Routing-Verfahren (2)

- Nachdem ein Knoten ein LSP aller empfangen hat, berechnet er die jeweils kürzeste Route zu den anderen
  - z.B. mit Kürzeste-Wege-Algorithmus nach Dijkstra
  - woher aber ist die Gesamtmenge aller Knoten (von denen ein LSP eintreffen muss) bekannt?
- Eigenschaften
  - reagiert relativ schnell auf Topologieänderungen
  - Nachrichtenoverhead in der Praxis erstaunlich gut verkraftbar
  - Informationsmenge pro Knoten ist allerdings beachtlich (ein LSP für jeden anderen Knoten)
  - Berechnung ist bei grossen Topologien relativ aufwendig (allerdings ist beim "shortest path first"-Verfahren die Nutzung grösserer Teile einer früheren Berechnung bei einer Neuberechnung möglich)
  - Skalierbarkeit?
- Link-State-Verfahren wird in der Praxis oft benutzt, z.B. beim OSPF ("Open Shortest Path First") im Internet

  "offener" Standard
- OSPF:
  - wird oft für das Routing *innerhalb* eines AS verwendet
  - weitere Hierarchiestufe innerhalb eines AS möglich ("area")
  - einfache Authentifizierungsmöglichkeit (Password; Prüfsumme)
  - Ermittlung von up/down durch periodisches Versenden von zu bestätigenden IP-Kontrollpaketen an alle Nachbarn
  - Kennenlernen der Nachbaridentitäten durch Hello-Pakete (z.B. nach einem reboot)

BGP (Border Gateway Protocol)

- Routing zwischen verschiedenen AS (Internet-Standard)
- Jeder BGP-Router eines AS kennt stets die beste Route als vollständigen Pfad zu jedem anderen BGP-Router



Beispiel: G kennt zu D die Route GEAD

Falls nun EA gestört ist, kann z.B. mit GBAD oder GCFHD ein alternativer Weg zu D gewählt werden, sofern die Routing-Policy dies gestattet

Dies spätestens dann, wenn G von seinen beiden Nachbarn deren Pfade BAE bzw. CFHD zu D erfährt

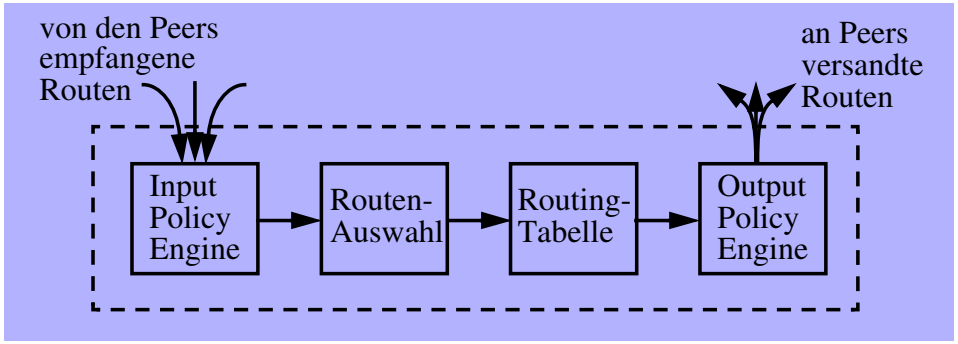
- Ein BGP-Router eines AS (der "speaker") informiert periodisch alle "Peers" (i.a. Nachbarn) über seine Routen
  - über TCP, port 179
  - alternative Routen zu einem Ziel werden unter Berücksichtigung der Routing-Politik bewertet (∞, falls eine Bedingung nicht erfüllt ist), die jeweils "billigste" Route wird dann verwendet
  - i.a. kein Austausch ganzer Tabellen, sondern Pfade neuer Routen inkrementell an die Peers weitermelden
  - eigene AS-Nummer beim Weiterleiten an das Pfadende anhängen, Zyklentest bei Empfang eines Pfades (taucht eigene AS auf?)
  - Widerrufen von Routing-Pfaden ist möglich
  - KEEPALIVE-Nachricht zu Nachbarn typw. jede Sekunde

BGP: Policy-Routing

- BGP kann eine "Routing-Policy" berücksichtigen
  - z.B. "Pakete von X nach Y dürfen nicht über Z laufen" (wobei Z eine fremde Firma, das Ausland, ein teurer Provider... sein kann)
  - z.B. "A leitet nur Pakete von P oder Q weiter"
  - Routing-Policy muss manuell spezifiziert werden
  - Routing-Policy meist durch ökonomische Bedingungen oder Sicherheitsüberlegungen (z.B. "route stets über Firewall") bestimmt

- Man spricht allgemein von "Policy-Routing", wenn neben der Zieladresse auch andere Aspekte in die Routingentscheidung eingehen (z.B. Absenderadresse)

- Modell eines BGP-Routers:



- Policy Engines können Routen filtern und Attribute manipulieren
  - Routen sind typischerweise durch reguläre Ausdrücke über AS-Nummern oder IP-Nummern spezifiziert
  - damit Änderung der Priorität bzgl. einiger Routen oder Nachbarn; Zurückhalten gewisser Informationen...
  - i.a. unterstützt durch Filter im Firewall

Austausch von Routing-Information

- Wer an wen Routing-Information (update-Nachrichten) sendet und empfängt, muss (manuell) spezifiziert werden
  - hierzu existieren einfache formale Sprachen (z.B. RIPE-181 oder RPSL)

- Die Update-Policy für ein AS sieht dann z.B. so aus:

```
aut-num: AS513
descr: CERN
as-in:
from AS559 100 accept AS559 AS3303
from AS789 100 accept AS789
from AS1199 100 accept AS-NORDUNET
from AS1836 105 accept AS-EUNET AND NOT AS3333
from AS283 120 accept NONE
from AS286  120 accept ANY
as-out:
to AS559 announce AS513 AS789 AS777 AS1754 AS2879
to AS137 announce AS513
to AS283 announce ALL
default: AS3561 30
```

relative Kosten

akzeptiere von AS559 nur updates, die AS559 oder AS3303 betreffen

akzeptiere alle updates, die von AS286 gemeldet werden

gibt an AS137 nur die eigenen Routen bekannt

Einige Routing-Probleme

1) Zyklen durch fehlerhafte Konfiguration etc.:

```
 1  seg14router (130.83.14.254)  3 ms  2 ms  2 ms
 2  cis12.hrz.tu-darmstadt.de (130.83.127.3)  2 ms  2 ms  1 ms
 3  TU-Darmstadt2.WiN-IP.DFN.DE (188.1.11.17)  2 ms  2 ms  2 ms
 4  ZR-Frankfurt1.WiN-IP.DFN.DE (188.1.11.113)  4 ms  4 ms  4 ms
 5  * ZR-Koeln1.WiN-IP.DFN.DE (188.1.144.33)  9 ms  9 ms
 6  ZR-Hannover1.WiN-IP.DFN.DE (188.1.144.25)  20 ms  24 ms *
 7  ZR-Hamburg1.WiN-IP.DFN.DE (188.1.144.21)  27 ms  20 ms  26 ms
 8  ZR-Berlin1.WiN-IP.DFN.DE (188.1.144.17)  46 ms  38 ms  49 ms
 9  ZR-Hamburg1.WiN-IP.DFN.DE (188.1.144.18)  34 ms  27 ms  29 ms
10  ZR-Berlin1.WiN-IP.DFN.DE (188.1.144.17)  56 ms  59 ms  54 ms
11  ZR-Hamburg1.WiN-IP.DFN.DE (188.1.144.18)  34 ms *  32 ms
12  ZR-Berlin1.WiN-IP.DFN.DE (188.1.144.17)  63 ms *  56 ms
13  ZR-Hamburg1.WiN-IP.DFN.DE (188.1.144.18)  39 ms  41 ms  40 ms
14  ZR-Berlin1.WiN-IP.DFN.DE (188.1.144.17)  50 ms  57 ms  51 ms
15  ZR-Hamburg1.WiN-IP.DFN.DE (188.1.144.18)  43 ms  43 ms  45 ms
16  ZR-Berlin1.WiN-IP.DFN.DE (188.1.144.17)  67 ms  68 ms  67 ms
17  ZR-Hamburg1.WiN-IP.DFN.DE (188.1.144.18)  55 ms  54 ms  47 ms
18  ZR-Berlin1.WiN-IP.DFN.DE (188.1.144.17)  75 ms  69 ms  75 ms
...
```

2) Stark wachsender Datenverkehr an zentralen Routern

3) Wachsende Routing-Tabellen

- Verdoppelung alle 10 Monate zwischen 1988 und 1991
- Gegenmassnahmen wie Aggregation von Routen bremsen das Wachstum
- Anzahl der Routen im Internet für einen typischen Router:
  - 1988: 350
  - 1992: 8500
  - 1994: 20000
  - 1995: 34000
  - 1996: 42000
  - 1999: 80000

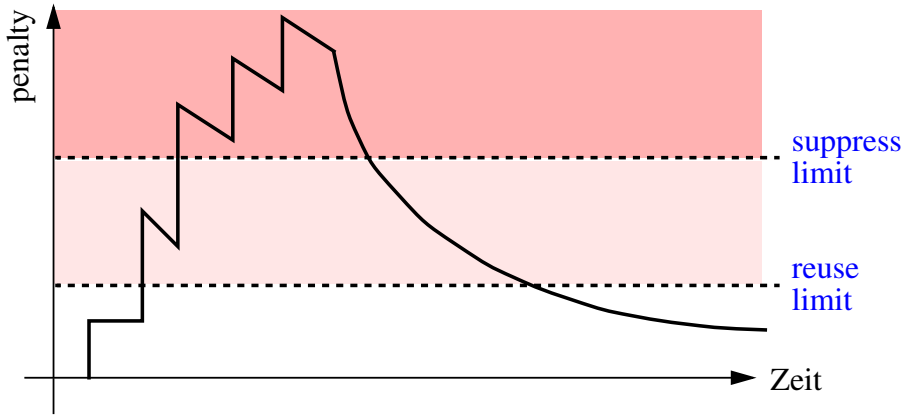
Routenfluktuation ("flapping")

- Routen können ggf. immer wieder verschwinden und kurz danach wieder auftauchen

- Ursachen vielfältig
  - z.B. Probleme der Hardware, in der Software, bei der Konfiguration

- Schafft Probleme
  - Probleme eines AS pflanzen sich ggf. im Internet fort
  - Overhead; Belastung der Router und des gesamten Internet
  - dadurch ggf. Nicht-Erreichbarkeit von Teilen des Internet

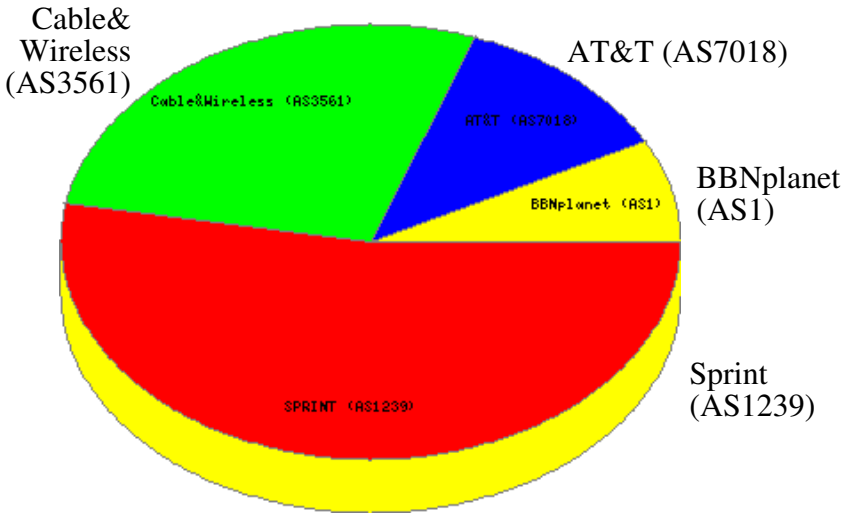
- Daher "Dämpfung" notwendig
  - Route oder Änderung einer Route unterdrücken, bis Vertrauen in die Stabilität der Route besteht



- Strafpunkte für eine Route
  - Route unterdrücken, wenn "penalty" grösser als suppress limit
  - Route erst bei Unterschreitung des reuse limits freigeben (Hysterese)
  - Strafpunkte verfallen exponentiell (Halbwertszeit konfigurierbar)

Routenfluktuation (2)

01/10/00 Internet Routing Instability at the Mae-East NAP
Instability Contributed by each Peer



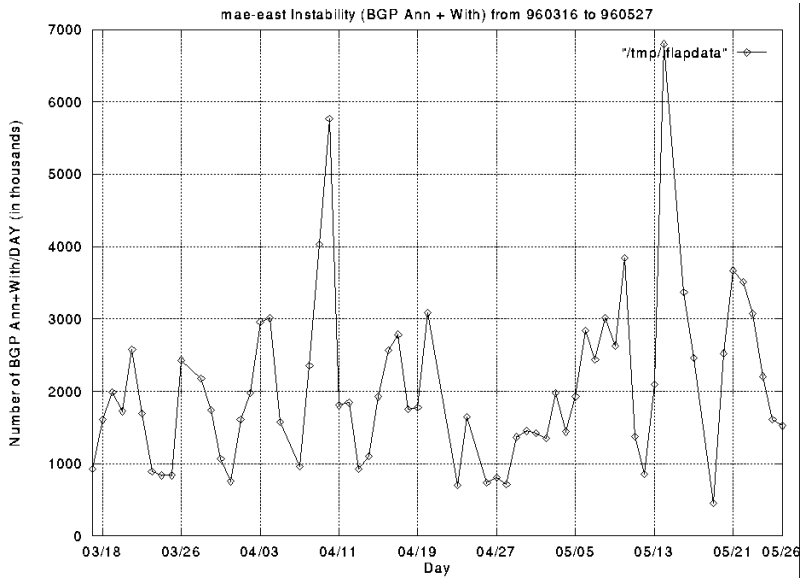
April 25, 1997. A misconfigured router maintained by a small Virginia service provider injected an incorrect routing map into the global Internet. This map indicated that the Virginia company's network provided optimal connectivity to all Internet destinations. Internet providers that accepted this map automatically diverted all of their traffic to the Virginia provider. The resulting network congestion, instability, and overload of Internet router table memory effectively shut down most of the major Internet backbones for up to two hours. Incorrect published contact information for operations staff, and lack of procedures for inter-provider coordination exacerbated the problem.

Quelle: http://www.eecs.umich.edu/techreports/cse/1998/CSE-TR-382-98.pdf

Routenfluktuation (3)

1996

The Internet core currently exchanges an average of **five million routing updates a day**. This graph shows the aggregate number of BGP route announcement and withdraws over a three month period. Note that April 10 shows a spike of ten million BGP updates and May 14 shows a peak of 7 million updates. Both of these large peaks were caused by single, small Internet service providers. These peaks show that individiual ISPs can have an inordinate affect on the level of routing instability in the core Internet.



Routers "flap," or lose their ability to route efficiently, for a number of reasons, including configuration errors, network links going up and down, software bugs, and other network problems. All these problems cause a very high number of routing updates to be passed to the core Internet exchange points. Route flaps spread from router to router and propagate throughout the network; at the extreme, route flaps have led to the collapse of portions of the Internet.

Routing instability, or the rapid fluction of Internet routing information appears to be growing at an exponetial rate in the Internet "core." This is a significant problem because severe levels of routing instability can lead to poor network performance (packet loss, latency, and interruptions of service).

Quelle: http://compute.merit.edu/analysis/routing.html