



Motivierende Fragen

- Welche IP-Dienste gibt es?
- Wie lassen sich IP-Routing-Protokolle klassifizieren?
- Auf welche Algorithmen basieren die Internet-Routing-Protokolle?
- Was steckt hinter IPv6?
- Wie lassen sich Internet und Mobilität vereinen?



Übersicht

- | | |
|---|--|
| <ol style="list-style-type: none"> 1. Einführung und Motivation <ul style="list-style-type: none"> ▪ Bedeutung, Beispiele 2. Begriffswelt und Standards <ul style="list-style-type: none"> ▪ Dienst, Protokoll, Standardisierung 3. Direktverbindungsnetze <ul style="list-style-type: none"> ▪ Fehlererkennung, Protokolle ▪ Ethernet 4. Vermittlung <ul style="list-style-type: none"> ▪ Vermittlungsprinzipien ▪ Wegwahlverfahren 5. Internet-Protokolle <ul style="list-style-type: none"> ▪ IP, ARP, DHCP, ICMP ▪ Routing-Protokolle 6. Transportprotokolle <ul style="list-style-type: none"> ▪ UDP, TCP 7. Verkehrssteuerung <ul style="list-style-type: none"> ▪ Kriterien, Mechanismen ▪ Verkehrssteuerung im Internet | <ol style="list-style-type: none"> 8. Anwendungsorientierte Protokolle und Mechanismen <ul style="list-style-type: none"> ▪ Netzmanagement ▪ DNS, SMTP, HTTP 9. Verteilte Systeme <ul style="list-style-type: none"> ▪ Middleware ▪ RPC, RMI ▪ Web Services 10. Netzsicherheit <ul style="list-style-type: none"> ▪ Kryptographische Mechanismen und Dienste ▪ Protokolle mit sicheren Diensten: IPSec etc. ▪ Firewalls, Intrusion Detection 11. Nachrichtentechnik <ul style="list-style-type: none"> ▪ Daten, Signal, Medien, Physik 12. Bitübertragungsschicht <ul style="list-style-type: none"> ▪ Codierung ▪ Modems |
|---|--|



Grundlagen: Rechnernetze und Verteilte Systeme

Kapitel 5:

Internet-Protokolle

Internet-Protokolle der Netzwerkschicht

Prof. Dr.-Ing. Georg Carle
Lehrstuhl für Netzarchitekturen und Netzdienste
Technische Universität München
carle@net.in.tum.de
http://www.net.in.tum.de



Ziele

- In diesem Kapitel wollen wir vermitteln
 - TCP/IP-Protokollfamilie
 - Funktionalität von IP-Adressen
 - Zusammenspiel von Protokollen
 - Hierarchie von Routing
 - Funktionalität von IPv6
 - Mobilität im Internet

Kapitelgliederung

Internet-Protokolle

- 5.1 Internet-Architektur
- 5.2 Internet-Protokollfamilie
- 5.3 IP-Adressen
- 5.4 IP-Datagramm
- 5.5 IP-Mechanismen
- 5.6 NAT
- 5.7 DHCP
- 5.8 Routing-Hierarchie (u.a. OSPF, RIP, BGP, CIDR, IGMP)
- 5.9 Steuerung von IP: ICMP
- 5.10 IP-Multicast
- 5.11 IPv6
- 5.12 Mobilität im Internet

Wiederholung: Die Internet-Protokollhierarchie

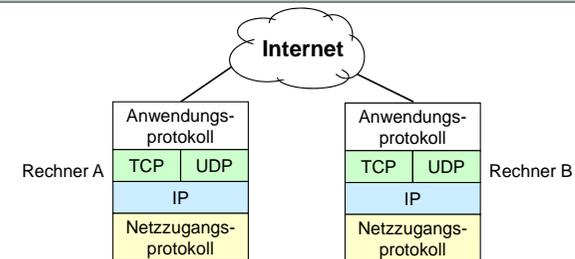
| | |
|-------------------|--|
| Application Layer | Anwendungsspezifische Funktionen zusammengefasst in Anwendungsprotokollen |
| Transport Layer | Ende-zu-Ende-Datenübertragung zwischen zwei Rechnern |
| Network Layer | Wegwahl im Netz mit Kopplung von Subnetzen |
| Data Link Layer | Medienzugriffssteuerung sowie schnittstelle zum physikalischen Medium |
| Physical Layer | Software: "Netzwerkkartentreiber" |

- Gegenüber ISO/OSI wurden die drei anwendungsorientierten Schichten zu einer einzigen Schicht zusammengefasst.

5.1 Internet-Architektur: Merkmale

- Grundlegende Entwurfsprinzipien:
 - Keine Zustandsinformation in den Zwischensystemen halten
⇒ bei Ausfall keine Resynchronisation notwendig
 - Datenstrom-spezifische (Flow-spezifische) Information wird in den Endsystemen gespeichert
⇒ Bestandteil des Ende-zu-Ende-Prinzips
 - Trennung der Weiterleitung der Pakete („Forwarding“) vom „Routing“ = Erstellung der Weiterleitungstabellen
- IP-Basiskommunikationsdienst:
 - verbindungslos, unzuverlässig
 - abschnittsweise paketorientierte Weiterleitung, speichervermittelt
 - „Best Effort“-Dienstleistung: so gut wie möglich mit den momentan vorhandenen Ressourcen

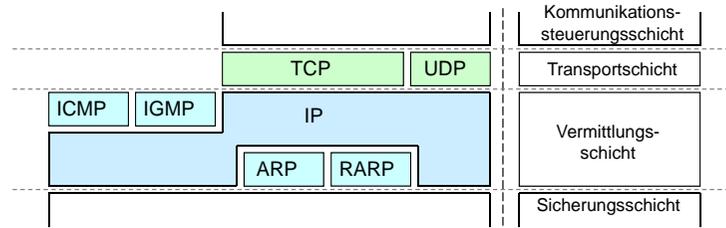
5.2 Die Internet-Protokollfamilie



- **TCP** (Transmission Control Protocol):
 - Zuverlässiges, verbindungsorientiertes Transportprotokoll über unzuverlässigem IP (Internet Protocol).
- **UDP** (User Datagram Protocol):
 - Verbindungsloses Transportprotokoll, bietet Anwendungsschnittstelle zu IP und Multiplexdienst.
- Beispiele für **Anwendungsprotokolle**:
 - HTTP: HyperText Transfer Protocol (im WWW benutzt)
 - FTP: File Transfer Protocol
 - Telnet: Protokoll für virtuelle Terminals

Die TCP/IP-Protokollfamilie – Überblick

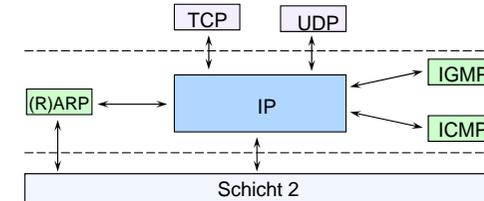
- Die Bezeichnung TCP/IP wird häufig als Synonym für die gesamte Protokollfamilie verwendet
- Einordnung der Internetprotokolle in das ISO/OSI-Referenzmodell:



- Die IP-Steuerungsprotokolle ICMP und IGMP nutzen den IP-Dienst, werden aber dennoch der Vermittlungsschicht zugeordnet
- In den anwendungsbezogenen Schichten 5-7 werden im Internet Protokolle wie z.B. FTP, TELNET oder SMTP eingesetzt (Schichten 5-7 im Internet zusammengefasst zur Anwendungsschicht)

Zusammenspiel: IP-Instanz und angrenzende Instanzen

- Sender
 - Schicht-4-Instanz (TCP- bzw. UDP) übergibt die Daten zusammen mit der IP-Adresse des Empfängers zur Übertragung an die IP-Instanz
 - IP-Instanz beauftragt ARP-Instanz mit Ermittlung der Schicht-2-Adresse
 - IP-Instanz übergibt PDUs zusammen mit der ermittelten Schicht-2-Adresse an die Instanz der Sicherungsschicht
- Empfänger: IP-Instanz reicht empfangene Daten an TCP- bzw. UDP-Instanzen weiter
- Steuerungsprotokolle
 - Probleme während der Übermittlung können den Partnerinstanzen mittels ICMP mitgeteilt werden
 - Informationen über Gruppenzugehörigkeiten werden mittels IGMP (Internet Group Management Protocol) im Netz verbreitet

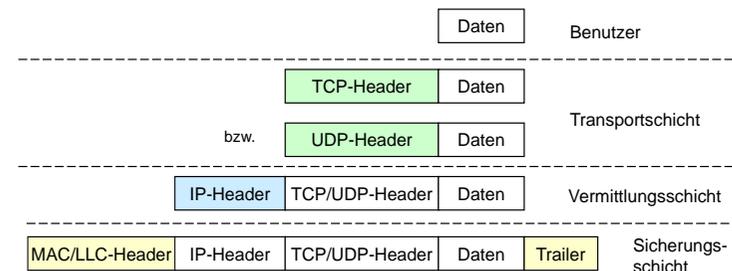


Die TCP/IP-Protokollfamilie: Protokollaufgaben

- TCP** (Transmission Control Protocol): Stellt verbindungsorientierten, gesicherten Transportdienst bereit
- UDP** (User Datagram Protocol): Stellt verbindungslosen, ungesicherten Transportdienst bereit
- IP** (Internet Protocol): Sorgt für Wegewahl und ungesicherte Übertragung von Datagrammen
- ICMP** (Internet Control Message Protocol): Unterstützt den Austausch von Steuerungsinformationen innerhalb der Vermittlungsschicht
- IGMP** (Internet Group Management Protocol): Unterstützt die Verwaltung von Kommunikationsgruppen
- ARP** (Address Resolution Protocol): Unterstützt die Zuordnung von IP-Adressen zu den entsprechenden Adressen der Sicherungsschicht
- RARP** (Reverse Address Resolution Protocol): Stellt die Umkehrfunktion von ARP zur Verfügung (Zuweisung einer Netzwerkadresse aufgrund einer gegebenen Adresse der Sicherungsschicht)
- DHCP** (Dynamic Host Configuration Protocol): Protokoll zur Zuweisung einer Netzwerkadresse an ein Endsystem, sowie Zuweisung weiterer Informationen zur Nutzung von Netzwerkdiensten, wie z.B. IP-Adresse eines DNS-Servers (Domain-Name-Servers)

Zusammenspiel der Protokollinstanzen: die PDUs

- IP leitet Datenpakete durch das Netzwerk zum Empfänger
- TCP/UDP fügen Prozessadressierung (Ports) zu IP hinzu
- TCP sichert darüberhinaus die Datenübertragung
- Protokolldateneinheiten (PDUs) werden gekapselt





5.3 IP-Adressen / Adressklassen (historisch)

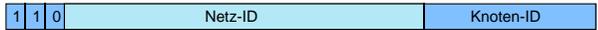
- Class A für Netze mit bis zu 16 Mio. Knoten (0.0.0.0 - 127.255.255.255):



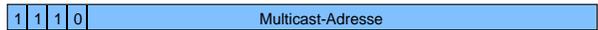
- Class B für Netze mit bis zu 65.536 Knoten (128.0.0.0 - 191.255.255.255):



- Class C für Netze mit bis zu 256 Knoten (192.0.0.0 - 223.255.255.255):



- Class D für Gruppenkommunikation (Multicast) (224.0.0.0 - 239.255.255.255):



- Class E, noch reserviert für zukünftige Anwendungen (240.0.0.0 - 247.255.255.255):



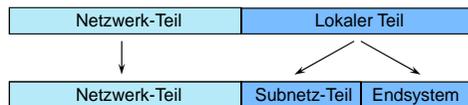
CIDR: Classless Inter-Domain Routing

- Bisher: 3 Adressklassen für Unicast (A, B und C)
 - schlechte Ausnutzung durch ungenutzte Adressen („Verschnitt“)
 - ⇒ Granularität der Netze häufig nicht passend; Anzahl der Netze zu klein
- Beispiel:
 - Kleinbetrieb, der 100 IP-Adressen braucht, beantragt Class-C-Netz
 - 254 Adressen könnten vergeben werden, damit **154 ungenutzte Adressen** ⇒ besser passend: Netzwerk mit $2^7=128$ Adressen
- CIDR: Ersetzen der festen Klassen durch **Netzwerk-Präfixe** variabler Länge
 - Bsp.: 129.24.12.0/25 → Die ersten 25 Bit der IP-Adresse werden für die Netzwerk-Identifikation verwendet (verbleibende 7 Bit für Hosts)
- Einsatz in Verbindung mit hierarchischem Routing:
 - Backbone-Router, z.B. an Transatlantik-Link, betrachtet nur z.B. die ersten 13 Bit; dadurch kleine Routing-Tabellen, wenig Rechenaufwand
 - Router eines angeschlossenen Providers z.B. die ersten 15 Bit
 - Router in einem Firmennetz mit 128 Hosts betrachtet die ersten 25 Bit



IP-Subnetz-Adressen

- IP-Adresse (hier Class B):



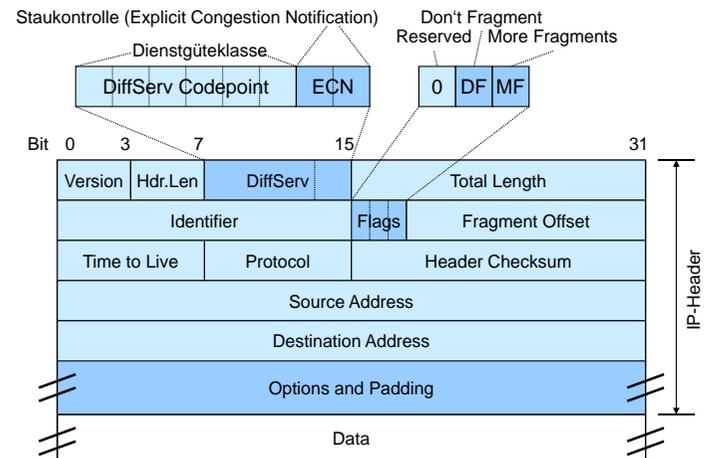
- Subnetzmasken kennzeichnen den Bereich der IP-Adresse, der das Netzwerk und das Subnetzwerk beschreibt. Dieser Bereich wird dabei durch Einsen („1“) in der binären Form der Subnetzmaske festgestellt. Subnetzmasken haben keine Internet-weite Gültigkeit.

- Beispiel:

| | | | | |
|---------------|-----------|-----------|-----------|-----------|
| IP-Adresse: | 129. | 13. | 3. | 64 |
| | 1000 0001 | 0000 1101 | 0000 0011 | 0100 0000 |
| Subnetzmaske: | 255. | 255. | 255. | 0 |
| | 1111 1111 | 1111 1111 | 1111 1111 | 0000 0000 |
| Netzwerk: | 129. | 13. | | |
| Subnetz: | | | 3. | |
| Endsystem: | | | | 64 |

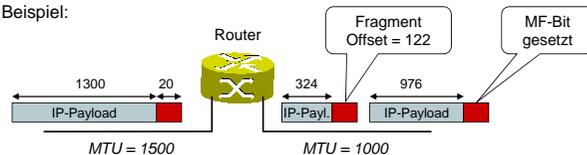


5.4 IP Datagramm: Aufbau



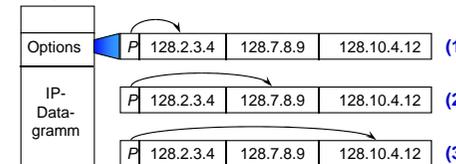
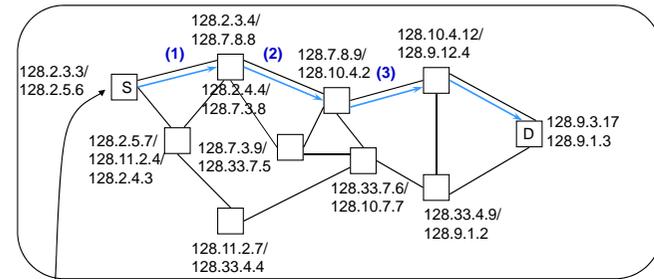
IP-Fragmentierung

- Größe eines IP-Paketes durch maximale Rahmengröße auf Schicht 2 begrenzt
 - MTU (Maximum Transport Unit): maximale Nutzdatenlänge in Schicht-2-Rahmen
 - Beispiel Ethernet: MTU = 1500 Byte, bzw. 1492 Byte bei LLC+SNAP Header
- IP-Endsysteme kennen MTU der angeschlossenen Netzwerkadapter
 - Sender passt i.d.R. Paketgröße an lokale MTU an
- IP-Fragmentierung wird notwendig, wenn Paket über einen Link mit kleinerer MTU geroutet wird
 - Fragmentierung der IP-Payload an 8-Byte Grenzen
 - Beispiel:



- Fragmentierung durch Path-MTU-Discovery vermieden (→ ICMP)

Optionale IP-Mechanismen: Source Routing



Hinweis:
Jeder Schnittstelle (Interface) ist eine eigene IP-Adresse zugeordnet. Im Diagramm sind aber nicht alle IP-Adressen aufgeführt.

5.5 IP-Mechanismen: Überprüfung des Paketkopfes

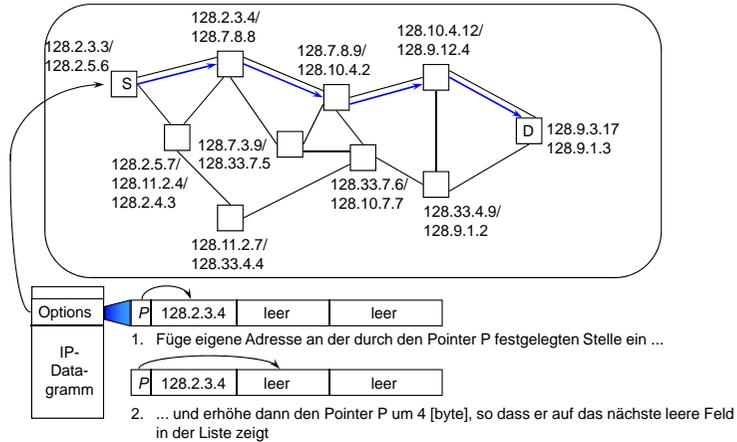
- Überprüfungen, die nach dem Empfang eines IP-Datagrammes am Header durchgeführt werden
 - Überprüfung der korrekten Länge des Headers
 - Test der IP-Versionsnummer
 - Überprüfung der korrekten Datagrammlänge
 - Prüfsummenbildung über den IP-Header
 - Überprüfung der Paketlebenszeit
 - Überprüfung der Protokoll-ID
- Bei negativem Resultat eines der oben aufgeführten Tests wird das Paket einfach verworfen und eine Fehlermeldung über ICMP an den Sender des Pakets gesendet

Optionale IP-Mechanismen: Zeitstempel

- Jeder Router fügt im Optionsfeld einen Zeitstempel ein, der den Zeitpunkt charakterisiert, zu dem das Paket vom Router bearbeitet wurde.
 - Aussagen über die Belastung der Netzwerke sind möglich
 - Die Effizienz der benutzten Routing-Algorithmen kann abgeschätzt werden
- Dabei existieren folgende Möglichkeiten, die durch ein 4 Bit langes Flag-Feld definiert werden:
 - Flag-Wert = 0: Nur Zeitstempel aufzeichnen, keine Adressen.
 - Flag-Wert = 1: Sowohl Zeitstempel als auch Adressen (Route Recording) aufzeichnen
 - Flag-Wert = 3: Die Adressen sind vom Sender vorgegeben (Source Routing), die adressierten Router tragen nur ihren Zeitstempel ein

Optionale IP-Mechanismen: Route Recording

- Im Record-Route-Options-Feld wird der durchlaufene Weg festgehalten



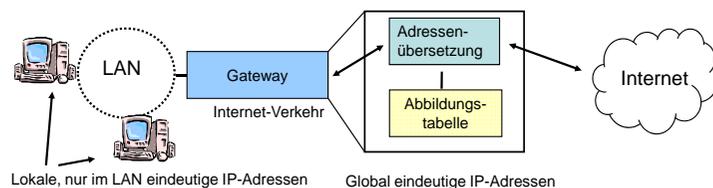
NAT - weitere Eigenschaften

- Abbildungsarten:
 - Statisch: lokale Adresse \leftrightarrow globale Adresse
 - z.B. 192.168.39.100 \leftrightarrow 129.13.41.100
 - Dynamisch: lokale Adresse \leftrightarrow globale Adresse aus Adresspool
 - Erzeugen eines einfachen Eintrags („Simple Entry“) in Abbildungstabelle (IP_Ad_{lokal} \leftrightarrow IP_Ad_{global})
 - Overloading:
 - Abbildung aller lokalen Adressen auf eine einzige globale Adresse
 - zusätzliches Unterscheidungskriterium: Portnummern
 - Erzeugen eines erweiterten Eintrags („Extended Entry“) in Abbildungstabelle
 - Protokoll, (Port_{lokal}, IP_Ad_{lokal}) \leftrightarrow (IP_Ad_{global}, Port_{global})

| Protokoll | lokaler Port | lokale IP-Addr. | globaler Port | globale IP-Addr. | Ziel-IP-Addr. | Ziel-Port |
|-----------|--------------|-----------------|---------------|------------------|---------------|-----------|
| TCP | 1024 | 192.168.1.1 | 1024 | 129.133.3.1 | 207.171.4.4 | 1234 |
| TCP | 1500 | 192.168.1.2 | 1500 | 129.133.3.1 | 134.100.4.4 | 80 |

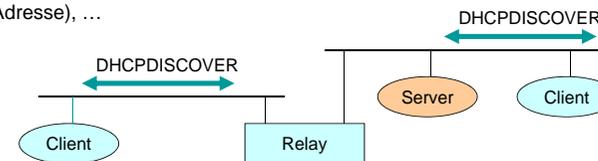
5.6 Network Address Translation/Translator (NAT)

- Nur Rechner, die gerade mit der Außenwelt kommunizieren, benötigen eine global eindeutige Adresse
 - Diese global eindeutige Adresse kann temporär vergeben werden
- Gateway/Router nimmt transparente Umsetzung zwischen Adressen/Adressbereichen vor
 - Speicherung in Abbildungstabelle
 - keine Änderungen an Endgeräten erforderlich
 - \Rightarrow Identitäten der Hosts werden verborgen

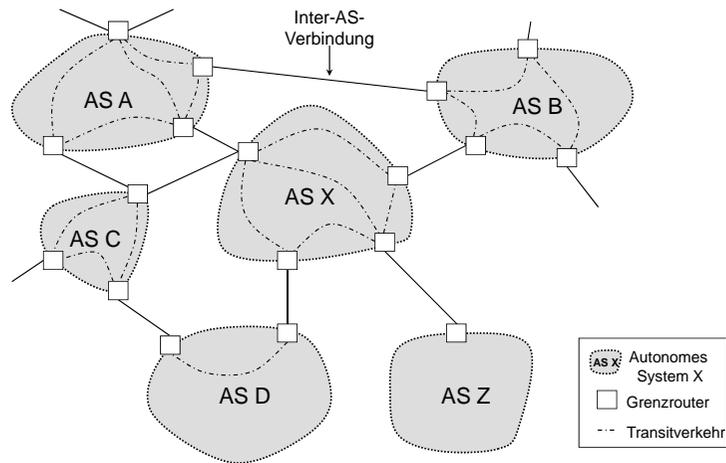


5.7 DHCP: Dynamic Host Configuration Protocol

- Anwendung
 - Vereinfachung der Installation und Verwaltung von vernetzten Rechnern
 - liefert Rechnern notwendige Informationen über IP-Adresse, DNS-Server-Adresse, Domain-Namen, Subnetz-Masken, Router etc.
 - damit weitgehend automatische Integration eines Rechners in das Internet bzw. Intranet
- Client/Server-Modell
 - ein Client sendet via IP-Broadcast eine Anfrage an einen DHCP-Server an UDP Port 67 (unter Umständen über ein DHCP-Relay)
 - Clientanfragen: DHCPDISCOVER, DHCPREQUEST, DHCPRELEASE, ...
 - der Server antwortet (initial via IP-Broadcast) und liefert die angeforderte Konfiguration. Serverantworten: DHCPOFFER, DHCPACK (mit IP-Adresse), ...



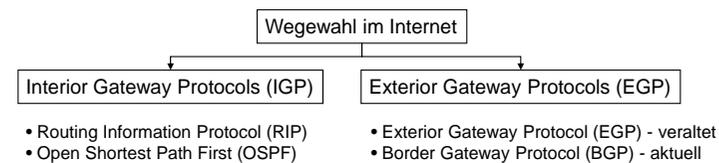
5.8 Die Internet-Routing-Hierarchie



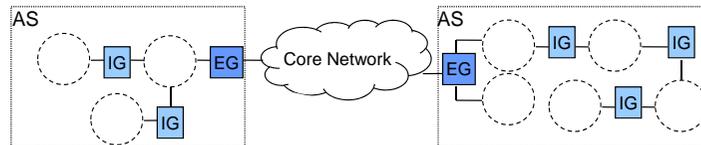
Übersicht: IP-Routing Protokolle

- **IGP** (Interior Gateway Protocol): zur Wegewahl *innerhalb* einer Verwaltungseinheit (Administrative Domain oder Autonomous System)
 - **RIP** (Routing Information Protocol) basierend auf Distance-Vector-Algorithmus (überall verfügbar, aber nur in einfachen Topologien sinnvoll)
 - **OSPF** (Open Shortest Path First) basierend auf Link-State-Algorithmus (neuer Standard)
 - **IS-IS** (Intermediate System-Intermediate System), ebenfalls Link-State-Algorithmus, aus der OSI-Welt, teilweise auch im Internet eingesetzt
- **EGP** (Exterior Gateway Protocol): Wegewahl *zwischen* Verwaltungseinheiten
 - EGP (Protokoll gleichen Namens!, veraltet)
 - **BGP** (Border Gateway Protocol, derzeit Version BGP4, u.a. RFC 1654)
 - Durch entsprechende Konfiguration kann gewährleistet werden:
 - Verhindern unerwünschtes Durchleiten „fremder“ Pakete durch eigenes Netz, auch wenn der Weg kürzer ist
 - politische Restriktionen
 - Firmenpolitik (Firma X will nicht für den Transport der Pakete von Firma Y bezahlen)

Dynamische Wegewahl im Internet



Autonomes System (AS): Netz(e) unter einheitlicher Verwaltung
Interior Gateway (IG): Interner Router eines AS
Exterior Gateway (EG): Router am Rande eines AS



IP-Routing-Protokolle

- **Intra-Domain-Routing:**
 - **RIP** (Routing Information Protocol)
 - nur in kleinen Netzen bzw. einfachen Topologien sinnvoll
 - wenig robust in komplexeren Netzwerken (Schleifenbildung)
 - langsamer bei Änderungen
 - Distanzvektorverfahren (Distance Vector Routing)
 - **OSPF** (Open Shortest Path First)
 - vom IAB empfohlenes Protokoll
 - „Link State“-Verfahren
 - **IS-IS** (Intermediate System-Intermediate System)
 - ebenfalls Link-State-Algorithmus, aus der OSI-Welt, auch im Internet eingesetzt
- **Inter-Domain-Routing:**
 - **BGP** (Border Gateway Protocol)
 - Pfad-Vektor-Verfahren
 - BGP Version 4 (BGP4) unterstützt Classless Inter-Domain Routing (CIDR)
 - Durch entsprechende Konfiguration kann gewährleistet werden:
 - Verhindern unerwünschtes Durchleiten „fremder“ Pakete durch eigenes Netz, auch wenn der Weg kürzer ist
 - politische Restriktionen
 - Firmenpolitik (Firma X will nicht für den Transport der Pakete von Firma Y bezahlen)

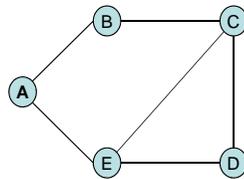


Distance Vector Routing

- **Verteiltes, adaptives Routing**
- Von RIP (Routing Information Protocol) realisiert
- Jeder Router speichert eine Tabelle mit der besten Entfernung (z.B. Anzahl Hops, Verzögerung in ms) zu jedem Ziel und dem dazugehörigen Ausgang bzw. nächstem Hop
- Benachbarte Router teilen sich in regelmäßigen Abständen den Inhalt ihrer Routing-Tabelle mit und aktualisieren damit ihre eigene Tabelle (Bellman-Ford-Algorithmus)

Routing-Tabelle von A:

| Ziel | Nächster Hop | Entfernung |
|------|-------------------|------------|
| B | - | 1 |
| C | B/E ¹⁾ | 2 |
| D | E | 2 |
| E | - | 1 |



¹⁾ je nach dem, welche Route zuerst bekannt ist



Distance Vector Routing

Lösungsansätze für schnellere Konvergenz

- **Split Horizon**
 - Eine Pfadinformation darf nicht über das selbe Interface veröffentlicht werden, über das sie empfangen wurde (simple split horizon).
- **Poisoned Reverse**
 - Die Pfadinformation wird zwar an das Interface zurückgeschickt, über die sie empfangen wurde, aber die Entfernung wird auf unendlich gesetzt.
 - Beispiel zu voriger Folie: C darf Route nach A nicht an B weitergeben (bzw. nur mit Entfernung unendlich), weil diese Route von B annociert wurde
 - funktioniert nicht bei größeren Schleifen
- **Triggered Updates**
 - Ändert sich eine Metrik, wird die Pfadinformation sofort propagiert und nicht erst nach einem Timeout.
- **Path Vector**
 - Es wird nicht nur der nächste Hop, sondern der ganze Pfad bis zu Ziel weitergegeben bzw. gespeichert
 - Kommt bei BGP zum Einsatz



Distance Vector Routing (Fortsetzung)

- Problem bei Distance Vector Routing:
 - Für Netze mit vielen Routern: langsame Konvergenz zu einem konsistenten Zustand wegen „count-to-infinity“ –Problematik
- Beispielszenario: Router A bis E verbunden, plötzlich fällt A aus

| | A | B | C | D | E |
|---|-----|-----|-----|-----|-----|
| A | ∞ | 1 | 2 | 3 | 4 |
| B | 3 | ∞ | 2 | 3 | 4 |
| C | 3 | 4 | ∞ | 3 | 4 |
| D | 5 | 4 | 5 | ∞ | 4 |
| E | 5 | 6 | 5 | 6 | ∞ |
| | 7 | 6 | 7 | 6 | 6 |
| | 7 | 8 | 7 | 8 | 8 |
| | ... | ... | ... | ... | ... |
| | ∞ | ∞ | ∞ | ∞ | ∞ |

Achtung: stets das Minimum wird gewählt!

Startabstand in hops von A

1. Austausch
 2. Austausch
 3. Austausch
 4. Austausch
 5. Austausch
 6. Austausch
 - ...
- je nach Festlegung der Obergrenze (typ. 16)

- Diverse Lösungsansätze (c.f. nächste Folie):
 - Split Horizon, Poisoned Reverse, Triggered Updates, Path Vector



Link State Routing

- **Verteiltes, adaptives Routing**
- Als OSPF (Open Shortest Path First) und IS-IS (Intermediate System - Intermediate System) im Internet eingesetzt
- Algorithmus:
 - Entdecken neuer Nachbarn über HELLO-Pakete
 - Bestimmung der Link-Kosten:
 - entweder durch Konfiguration vorgegeben, oder ermittelt über Messung der Verzögerung zu jedem Nachbarn (ECHO-Paket misst Umlaufzeit)
 - Erstellen eines „Link-State“-Paketes mit allen gelernten Daten
 - beinhaltet ID des Senders (Routers), Liste der Nachbarn mit Verzögerung, Alter
 - periodische oder ereignisgesteuerte (z.B. neuer Nachbar, Ausfall) Erzeugung
 - Aussenden dieses Paketes an alle Nachbarn
 - Nachbarn geben Link-States wiederum an ihre Nachbarn weiter usw. → prinzipiell Fluten an alle Router, aber mit Verfeinerungen: Vernichten von Duplikaten, Zerstören der Information nach gewissem Alter etc.
 - Jeder Router kennt am Ende die Topologie des Netzwerkes
 - Berechnung des kürzesten Pfades zu allen anderen Routern (z.B. Dijkstra)
 - rechenaufwendig, Optimierungen existieren

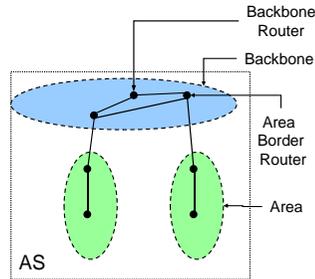
OSPF (Überblick)

Grundlage:

- Verbessertes Link State Routing
- Basiert auf der Berechnung eines kürzesten Pfades von Quelle zum Ziel
- Hierarchische Aufteilung des AS
- Intra-Area-, Inter-Area-, Inter-AS-Verkehr
- Je Area Auswahl eines "Designated Router" (DR) und eines Backup-DR

Eigenschaften:

- Unterstützung unterschiedlicher Metriken, z.B. Anzahl Hops, Verzögerung
- Adaptiv (d.h. reagiert auf Topologieänderungen)
- Lastausgleich (Berücksichtigung verschiedener Wege) zum Zielknoten
- Unterstützung hierarchischer (Sub-)Netze
- Unterstützung verschiedener Wege



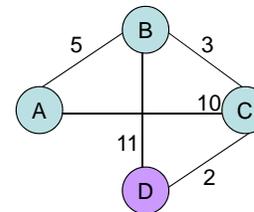
Dijkstra's Shortest Path Algorithmus

- Realisierung durch Forward Search Algorithmus
- Jeder Router hält 2 Listen: Confirmed und Tentative mit Einträgen der Form {Ziel, Kosten, nächster Knoten}
- Algorithmus
 1. Initialisierung von Confirmed mit eigenem Eintrag (Kosten 0)
 2. Selektiere den zuletzt in Confirmed aufgenommenen Knoten (Next)
 3. Berechne die Kosten zu allen Nachbarn von Next:
 $\text{Kosten} = \text{Kosten zu Next} + \text{Kosten von Next zu Nachbar}$
 - a. Falls Nachbar nicht in Confirmed: Aufnahme des Nachbarn in Tentative
 - b. Falls Nachbar in Tentative: Ersetzen des Eintrags, falls die Kosten niedriger als die des Eintrags sind.
 4. Stop, falls Tentative leer ist, sonst: übernehme in Confirmed den Eintrag mit den geringsten Kosten aus Tentative, gehe zu 2

Intra-Domain-Routing mit OSPF

- Namensgebung:
 - Open: offener Entwicklungsprozess der IETF
 - SPF: durch Dijkstras Algorithmus kürzeste Pfade in Graph zu finden
Komplexität: $O(n \log n)$, $n = \# \text{Links}$
- Je OSPF-Area nicht mehr als 200 Router empfohlen
- Jeder Knoten
 - besitzt komplettes Abbild des Netzwerks (Routing-Datenbank) und
 - berechnet selbständig alle Pfade mit SPF-Algorithmus
- Austausch von geänderten Einträgen durch Fluten (bestätigt)
- Periodischer Austausch von Einträgen (Link-ID, Version)
→ interessante Einträge werden explizit angefordert
- Unterstützung mehrerer Metriken und mehrfacher Pfade
- Externe Routen werden gesondert eingetragen und vermerkt

Forward Search Algorithmus



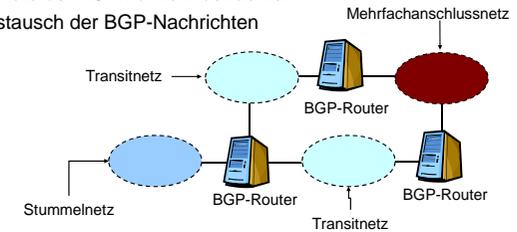
| Schritt | Confirmed | Tentative |
|---------|-------------------------------------|-------------------|
| 1 | (D,0,-) | |
| 2 | (D,0,-) | (C,2,C), (B,11,B) |
| 3 | (D,0,-), (C,2,C) | (B,11,B) |
| 4 | (D,0,-), (C,2,C) | (B,5,C), (A,12,C) |
| 5 | (D,0,-), (C,2,C), (B,5,C) | (A,12,C) |
| 6 | (D,0,-), (C,2,C), (B,5,C) | (A,10,C) |
| 7 | (D,0,-), (C,2,C), (B,5,C), (A,10,C) | |

Autonome Systeme im Internet

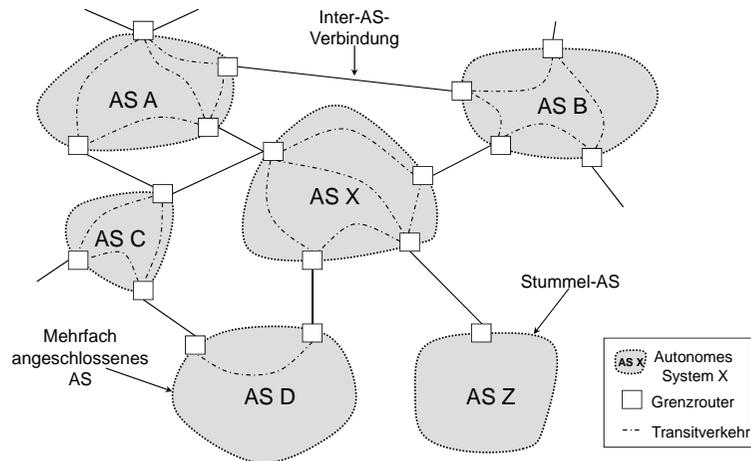
- **Autonomes System (AS):** Zusammenhängende Menge von Routern und Netzwerken unter derselben Administration
- Änderungen innerhalb des Systems sollen verborgen bleiben
- Betreiber möchte i.A. interne Struktur nicht bekannt geben
- Innerhalb eines AS sind auch unterschiedliche Intra-Domain-Routing-Protokolle möglich (RIP, OSPF)
- Jedes AS erhält eine eindeutige AS-Nummer (16 Bit)
- Derzeit > 14000 Autonome Systeme, davon
 - ~ 80% Stub/Origin only-AS
 - ~ 19% Mixed-AS
 - ~ 1% Transit-AS
 - Details siehe u.a.: <http://bgp.potaroo.net/> (bgp table growth)

BGP (Überblick)

- **Aufgabe:**
 - Wegewahl zwischen autonomen Systemen (AS) unter Berücksichtigung besonderer politischer, wirtschaftlicher oder sicherheitsbezogener Regeln (Policies).
- **Grundlage:**
 - Modifikation des Distance Vector Routing
 - Exakter Pfad zum Zielknoten wird gespeichert
 - BGP-Router tauschen komplette Pfade aus (→ Path Vector Routing)
 - Routing innerhalb der AS wird nicht betrachtet
 - TCP zum Austausch der BGP-Nachrichten



Internet-Architektur: Autonome Systeme



Inter-Domain-Routing mit BGP

- Welche Netzwerke sind über die Nachbardomänen erreichbar?
- Protokoll zum Austausch dieser Informationen: BGP-4 [RFC 1771]
- Unterstützung von Classless Interdomain Routing (CIDR)
- Berücksichtigung politischer Entscheidungen (Routing policies) möglich, daher Angabe kompletter Pfade (Schleifenfreiheit):

| Dest. Net. | Next-Hop | Pfad |
|--------------|-----------------|---------------|
| 141.3.0.0/16 | 195.221.222.254 | 5409 1275 553 |
- Austausch der Routing-Tabellen bzw. der Änderungen erfolgt über TCP-Verbindungen
- Größe der Routingtabelle: derzeit bis zu 140000 Einträge
- Import von Routen des Intra-Domain-Routing-Protokolls



Zusammenspiel: OSPF ↔ BGP

- Innerhalb eines AS muss jeder Router zu jedem Netzwerk-Präfix eine Route kennen
- Die Border-Router im Backbone-Bereich importieren und exportieren Routen zwischen BGP und OSPF
- Jeder Border-Router stellt zu jedem anderen Border-Router des AS eine interne BGP-Verbindung her
- Einfacher Fall: nur ein Router stellt die Verbindung zu externen Systemen her (Stub-Area) → Default-Route eintragen
- Weiterer Fall: Default-Route mit zusätzlichen expliziten Angaben
- Mehrere Router mit externer Verbindung: komplette Routing-Tabelle mit sämtlichen Zielnetzwerk-Präfixen notwendig (in OSPF importierte externe Routen)



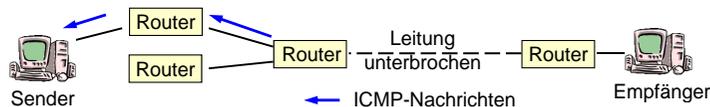
ICMP: Fehlermeldungen

- **Zieladresse nicht erreichbar** (destination unreachable): Ein Datenpaket konnte (z.B. wegen einer unterbrochenen Leitung oder eines ausgefallenen Routers) nicht zugestellt werden.
→ Die Fehlermeldungen enthalten jeweils ein Feld zur genauen Angabe der Fehlerursache (z.B. „Netzwerk nicht erreichbar“ oder „Endsystem nicht erreichbar“ für die Meldung „Zieladresse nicht erreichbar“)
- **Zeit abgelaufen** (time exceeded): Datenpaket wurde wegen Ablauf seiner Lebenszeit (TTL=0) von einem Router verworfen.
- **Falscher Parameter** (parameter problem): Datenpaket wurde wegen eines unzulässigen Wertes im IP-Paketkopf verworfen.
- **Quellendämpfung** (source quench): Ein überlastetes Kommunikationssystem fordert den Sender auf, die Übertragungsrate zu senken.
- **Umleiten** (redirect): Ein Datenpaket sollte besser über einen anderen Router gesendet werden.



5.9 Steuerung von IP: ICMP

- IP ist nur für den (unzuverlässigen) Datenaustausch zuständig.
- Für Fehlerfälle oder Testzwecke wird ICMP (Internet Control Message Protocol) verwendet.



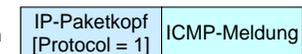
Nachrichtentypen, Beispiele:

- *Destination Unreachable*: Ziel nicht erreichbar.
- *Time Exceeded*: Time-to-Live-Feld eines Pakets ist abgelaufen.
- *Echo Request / Reply*: Echo Reply wird angefordert ("ping").
- *Timestamp Request / Reply*: Ähnlich Echo Request. Zusätzlich Zeitstempel mit Ankunftszeit der Anfrage/Sendezeit der Antwort.



ICMP: Paketformat

- Übertragung der ICMP-Meldungen
 - ICMP-Meldungen werden im Datenteil von IP-Paketen übertragen und durch den Wert „1“ im Protocol-Feld des IP-Paketkopfes kenntlich gemacht.

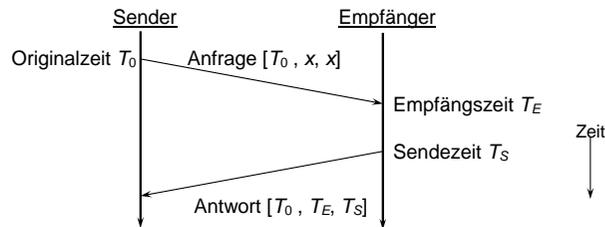


| | | | |
|------|------|----------|------|
| Type | Code | Checksum | Info |
|------|------|----------|------|

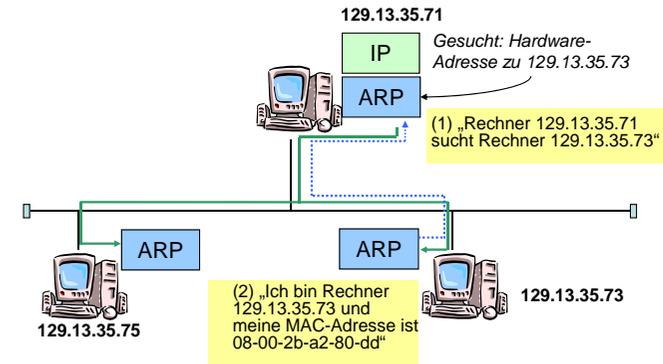
- Format der ICMP-Meldungen
 - Type: Typ der Meldung (z.B. Type = 3 entspricht „Zieladresse nicht erreichbar“)
 - Code: Genaue Beschreibung der Meldung (z.B. „Netzwerk nicht erreichbar“)
 - Checksum: Prüfsumme über die gesamte ICMP-Meldung
 - Der Inhalt des Info-Teils ist abhängig vom Typ der ICMP-Meldung (z.B. Felder für Zeitstempel bei Meldung „Zeitstempel und Zeitstempelantwort“)
 - ICMP „Packet too big“ Nachricht: enthält Typ = 2, Code = 0, Checksum, MTU-size, gefolgt von Original-Paket (max. 576 byte)

ICMP: Statusanfragen

- **Echo und Echoantwort** (echo and echo reply): Dient der Überprüfung der Aktivität von Kommunikationssystemen. Der Empfänger einer Echo-Anfrage sendet in der Echo-Antwort die erhaltenen Daten an den Kommunikationspartner zurück.
- **Zeitstempel und Zeitstempelantwort** (timestamp and timestamp reply): Dient der Bestimmung von Paketumlaufzeiten. Die Meldungen umfassen mehrere Felder zur Aufnahme von Zeitstempeln, anhand derer die Paketbearbeitungszeiten beim Empfänger und die Verzögerung im Netzwerk bestimmt werden können.



ARP – Beispiel



Sicherheitslücke?

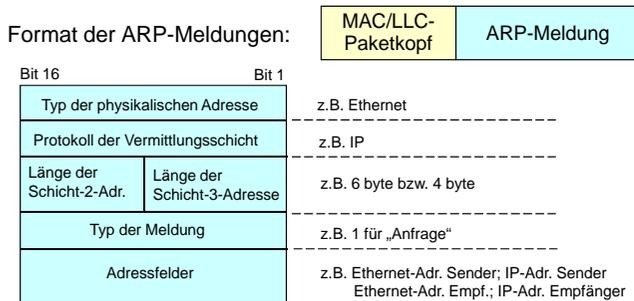
ARP: Address Resolution Protocol

- **Problem:**
 - Abbildung der Internet-Adresse eines Rechners auf die physikalische Adresse der Station (MAC-Adresse = Adresse der Adapterkarte)
- **Lösungsalternativen:**
 - Unterhalten einer Abbildungstabelle in jedem Rechner.
 - Unterhalten einer Abbildungstabelle in einem Server, der die Anfragen der Kommunikationssysteme beantwortet.
 - Dynamische Abbildung der Adressen durch Senden einer entsprechenden Anfrage an alle Rechner im LAN (Broadcast-Anfrage)
- Bei den beiden erstgenannten Methoden müssen die Abbildungstabellen bei jeder Änderung manuell abgeglichen werden.
- Ein Verfahren zur dynamischen Abbildung der Adressen ist durch das Address Resolution Protocol (ARP) festgelegt.

ARP: Paketformat

- Übertragung der ARP-Meldungen
- Eine ARP-Meldung wird im Datenteil eines Paketes der Sicherungsschicht übertragen.

- Format der ARP-Meldungen:



Länge und Aufbau der Adressfelder sind vom Typ der Adressen abhängig

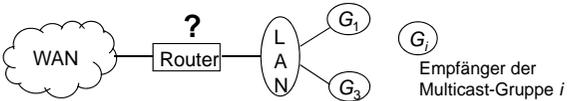


Reverse Address Resolution Protocol (RARP)

- **Aufgabe:**
 - Umsetzen MAC-Adresse \Rightarrow IP-Adresse.
 - Wichtig z.B. für plattenlose Workstations, die von einem Dateiserver booten. Dazu müssen sie ihre IP-Adresse wissen, welche die Station allerdings beim Einschalten noch nicht kennt.
- **Vorgehensweise:**
 - Station schickt einen Rundruf ins lokale Netz unter Angabe der eigenen MAC-Adresse, die durch die Hardware vorgegeben ist.
 - RARP-Server sieht die Anfrage und bestimmt anhand einer Konfigurationsdatei die zugehörige IP-Adresse.
 - RARP-Server schickt die IP-Adresse in einer RARP-Antwort an die anfragende Station zurück.



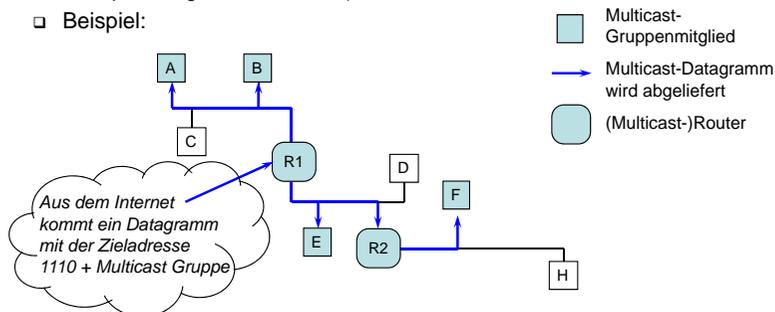
IGMP: Internet Group Management Protocol

- **Problem:**
 - Wie erkennt ein Router, dass Multicast-Nachrichten bestimmter Gruppen von ihm weitergeleitet werden müssen?
- **Beispiel:**

 - Nachrichten der Gruppen 1 und 3 müssen vom Router in das angeschlossene LAN weitergeleitet werden, wohingegen Nachrichten anderer Gruppen das LAN nicht erreichen sollten.
- **Lösungen:**
 - Manuelle Eingabe von Gruppenzugehörigkeiten in der Router-konfiguration \Rightarrow hoher Verwaltungsaufwand bei dynamischen Gruppen.
 - Selbstständiges Erlernen der Gruppenzugehörigkeiten durch den Austausch entsprechender Information \Rightarrow ein solches Verfahren wird durch das **Internet Group Management Protocol (IGMP)** beschrieben.

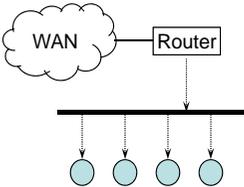
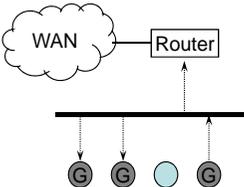


5.10 IP-Multicast

- Ein einzelnes IP-Datagramm wird an mehr als eine Station adressiert
- Es werden Class-D-Adressen verwendet
 - Die ersten vier Bits des Adressfelds im Kopfes entsprechen 1110
 - Danach folgt die 28 bit lange ID der Gruppe
- Lokale Multicast-Gruppenverwaltung erfolgt über das IGMP (Internet Group Management Protocol)
- Beispiel:



IGMP: Protokollablauf I

1. Der Router sendet periodisch Gruppenzugehörigkeitsanfragen an alle Rechner des LAN (via Broadcast). Setzen der „Time To Live“ (TTL) auf 1.
 
2. Nach Erhalt einer Anfrage startet jeder Rechner für jede Gruppe, welcher er angehört, einen Zeitgeber. Dieser wird mit einem Zufallswert initialisiert. Nach Ablauf des Zeitgebers sendet der Host eine Antwort bzgl. der Gruppenzugehörigkeit an alle Gruppenmitglieder im LAN (Gruppenadresse, TTL=1). Multicast-Router erhalten alle IP-Multicast-Nachrichten.
 

IGMP: Protokollablauf II

3. Weitere Gruppenmitglieder erhalten die Antwort und stoppen den entsprechenden Zeitgeber. Dadurch werden redundante Antworten vermieden.
4. Der Router erhält alle Antworten und aktualisiert seine Routing-Tabelle entsprechend. Erhält ein Router nach mehrmaliger Anfrage keine Antwort bzgl. einer bestimmten Gruppe, so wird ihr Eintrag aus der Routing-Tabelle gelöscht.
5. Tritt ein Rechner einer Gruppe bei, so sendet er sofort eine entsprechende Mitteilung an alle Router im LAN. Aus Gründen der Fehlertoleranz wird die Mitteilung wiederholt gesendet.

5.11 IPv6 – Motivation (ursprünglich)

Das Internet funktioniert seit Jahrzehnten! Warum ein neues IP-Protokoll?



Anwachsen des Internets: Der überwältigende Erfolg des Internets führte zu stark anwachsenden Benutzerzahlen
→ Bedarf an Adressen, sowie Änderungen von Adressen mit IPv4 nicht zufriedenstellend lösbar

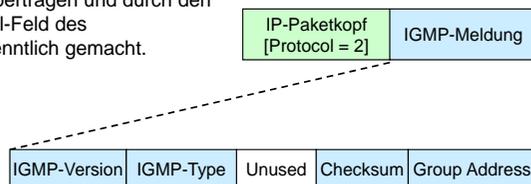
Wachsende Routingtabellen: Router müssen sich Wege zu immer mehr Zielnetzen merken.

Neue Anwendungen: Neuartige Anwendungen erfordern neuartige Dienste und zusätzliche Funktionalität (z.B. garantierte Netzdienste, Synchronisation von Audio- und Videodaten, automatisierte Konfiguration)
→ Konzepte zur funktionalen Erweiterung erforderlich

Hohe Datenraten: Hochleistungsfähige Zwischensysteme benötigen geeignete Paketformate zur effizienten Bearbeitung

IGMP: Paketformat

- Übertragung der IGMP-Meldungen
- IGMP-Meldungen werden im Datenteil von IP-Paketen übertragen und durch den Wert 2 im Protocol-Feld des IP-Paketkopfes kenntlich gemacht.



- Format der IGMP-Meldungen
 - IGMP-Version: Versionsnummer des eingesetzten IGMP-Protokolls.
 - IGMP-Type: Typ der Meldung (z.B. 1 = Anfrage, 0 = Antwort).
 - Unused: Wird nicht genutzt (immer zu 0 gesetzt).
 - Checksum: Prüfsumme über die gesamte IGMP-Meldung.
 - Group Address: Wird bei einer Anfrage auf 0 gesetzt, bei einer Antwort enthält das Feld die Adresse derjenigen Gruppe, auf welche sich die Meldung bezieht.

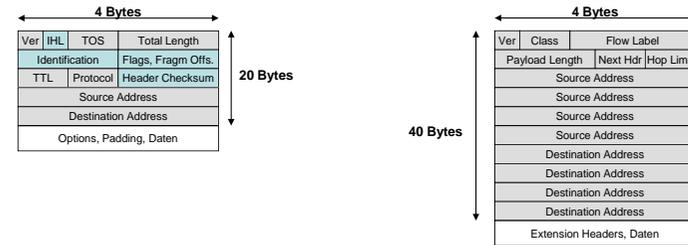
Überblick: Neuerungen in IPv6 (1)

- **Flexibles Paketformat**
 - Vereinfachung des (Standard-) Paketkopfes: 8 statt 13 Felder (IPv4)
 - Paketkopf fester Länge
 - Verschieben zahlreicher Optionen in optionale Paketkopferweiterungen
- **Erweiterte Adressierung**
 - Erhöhung der Adresslänge von 32 Bit auf 128 Bit
 - 2^{32} bit: ~ 4 Mrd. IPv4-Adressen
 - 2^{128} bit: ~ $3,4 \cdot 10^{38}$ IPv6-Adr., ~ 6×10^{23} Adressen pro m^2 Erdoberfläche
 - Definition mehrerer Hierarchieebenen (effizienteres Routing)
 - Einführung von *Anycast-Adressen* (Kommunikation zu einem Mitglied einer Gruppe)
- **Unterstützung von Ressourcenreservierung**
 - *FlowLabel* (Flussmarke) und Traffic Class pro IPv6-Paket
 - Ermöglichen die Nutzung von Protokollen zur Ressourcenreservierung

Überblick: Neuerungen in IPv6 (2)

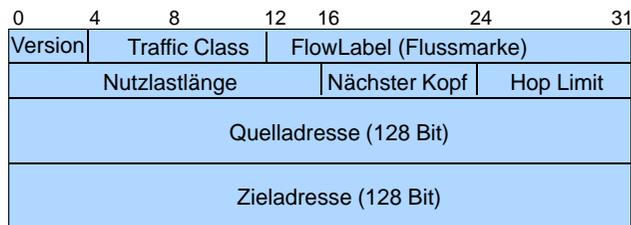
- **Neighbor Discovery**
 - Adressauflösung: IPv6-Adr.->MAC-Adr. (in ICMP integriert, ersetzt ARP)
 - Erkennen des nächsten Routers
- **Automatische Systemkonfiguration**
 - „Plug'n Play“: Inbetriebnahme eines Internet-Rechners ohne manuelle Konfiguration.
 - Realisiert durch Autokonfigurationsmechanismen und mit Unterstützung durch DHCP (Dynamic Host Configuration Protocol)
- **Unterstützung mobiler Systeme**
 - Adresszuweisung durch automatische Systemkonfiguration
 - Die Option „Binding Update“ im Destination-Options-Header ermöglicht die direkte Umleitung der IP-Pakete an den aktuellen Standort
- **Berücksichtigung von Sicherheitsaspekten**
 - Unterstützung von Authentifizierung und Datenintegrität
 - Sicherheitsmechanismen werden bei IPv6-Implementierungen gefordert (sind bei IPv4 in Form von IPSec lediglich optional vorhanden)

Vergleich: IPv4 und IPv6 Header



| Version = 4 | Version = 6 |
|--|---|
| IHL (Internet Header length) | Entfällt, immer 40 Bytes |
| TOS (Type of Service) | Traffic Class + Flow Label zur Identifikation von Flows |
| Total Length (Header + Paket) | Payload Length (ohne Header) |
| Identification, Flags, Fragment Offset | Fragmentierung wenn nötig per Extension Header |
| TTL (Time To Live) | Hop Limit |
| Protocol (z.B. TCP) | Next Header (z.B. TCP oder ein Extension Header) |
| Header Checksum | Entfällt |
| Source / Destination je 32 Bit | Source / Destination je 128 Bit |

IPv6-Paketformat

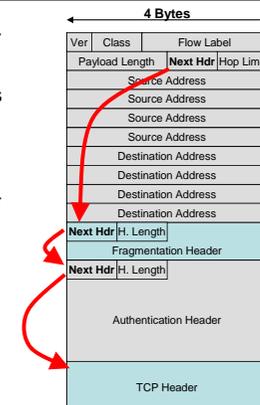


RFC 2460 - Internet Protocol, Version 6 (IPv6) Specification

- Erste Version von IPv6: RFC 1883 von 1995

Kopferweiterungen (1)

- Die Länge des IPv6-Headers ist immer 40 Bytes
 - Optionen werden in Extension Headers gesendet.
 - Der IPv6-Header und jeder Extension Header hat ein „Next Header“-Feld.
 - Im letzten „Next Header“-Feld steht der Bezeichner des Layer 4-Protokolls (6 = TCP, 17 = UDP)





Kopferweiterungen (2)

- Kopferweiterungen (Header Extensions) erlauben eine effizientere Implementierung und bewahren die Flexibilität für spätere Erweiterungen
- Definition unterschiedlicher Kopferweiterungen: Knoten-zu-Knoten-Optionen, Ziel-Optionen, Routing, Fragmentierung, Authentifizierung, Verschlüsselung
- Jeder Typ einer (optionalen) Kopferweiterung darf höchstens einmal in einer Dateneinheit enthalten sein
- Der Typ der jeweils nachfolgende Kopferweiterung wird im Feld „Nächster Kopf“ eingetragen

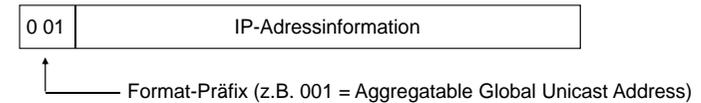
- Beispiel:

| | | |
|--|---------------------------------------|-----------------------------|
| Standard-Kopf (Nächster Kopf = Routing) | Routing-Kopf (Nächster Kopf = TCP) | TCP-Kopf + Benutzerdaten |
|--|---------------------------------------|-----------------------------|



IPv6: Adressierung

- IPv6-Adressen (Länge 128 Bit)
 - Angabe von Netzpräfixen in CIDR-Notation:
FEDC:BA98:7600::/40 → 40 Bit langes Präfix für das Routing
 - Unterscheidung von **Adressklassen**:
 - Unicast-Adressen
 - Anycast-Adressen
 - Multicast-Adressen – Präfix ff00::/8 (ff...)
 - Unterscheidung von **Adresstypen**:
 - Link-Local Address
 - Site-Local Address
 - Aggregatable Global Unicast Address
 - Typ einer Adresse wird durch *Format-Präfix* (führende Bits) festgelegt:

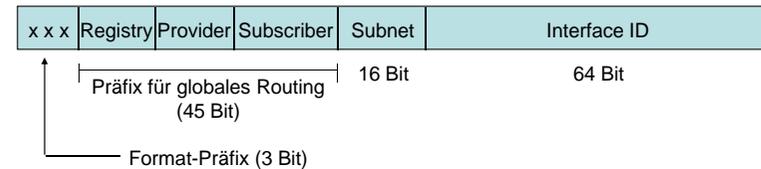


IPv6 Adressen (RFC 3513)

- Wie viele Adressen?
 - Adressfeld von 32 auf 128 Bit verlängert
 - 340 282 366 920 938 463 463 374 607 431 768 211 456 Adressen
 - Theoretisch 2^{96} Adressen pro Mensch (2006: 6,5 Mrd.)
 - Man kann es sich (hoffentlich) leisten, bei der Adressvergabe großzügig zu sein
 - Wichtiger noch ist die bessere Strukturierung (Hierarchie, Routenaggregation) des Adressraums
- Darstellung
 - Colon Hexadecimal Notation
 - 4 Bit pro Zeichen (0-F), nach jeweils 4 Zeichen ein Doppelpunkt zur Trennung
 - Führende Nullen in einem Abschnitt werden unterdrückt
2001:db8:85a3:8d3:1319:8a2e:370:7344
 - An einer Stelle einer Adresse darf eine längere Sequenz von Nullen weggelassen werden
2001:db8:0000:0000:0000:0000:1428:57ab wird zu
2001:db8::1428:57ab
 - Portangabe in URLs
 - `http://[2001:0db8:85a3:08d3:1319:8a2e:0370:7344]:80/`



Unicast-Adressen in IPv6 (1)



- Aggregatable Global Unicast Address (Provider-Based Unicast Address)
 - Hierarchischer Aufbau zur Vereinfachung der Vermittlung von IPv6-Daten.
 - Globale Gültigkeit im Internet.
 - Format-Präfix 001
- Link-Local Address
 - Besteht aus Präfix und Kennung der Netz Karte (Interface-Identifikation).
 - Format-Präfix fe80::/10 (also 1111 1110 10)
 - Registry = Provider = Subscriber = Subnet = '0'
 - Ausschließlich innerhalb eines Subnetzes gültig.
- Site-Local Address
 - Ausschließlich innerhalb eines Netzwerkes gültig.
 - veraltet (deprecated), stattdessen Unique Local Unicast RFC 4193



Unicast-Adressen in IPv6 (2)

- Interface Identifier
 - Modified EUI-64 Identifier
 - Können einen global eindeutiges Token (üblicherweise die MAC-Adresse) beinhalten
 - Privacy-Problem!
- Loopback-Adresse ::1



Unicast-Adressen in IPv6 (3)

- Adressformate für den Übergang von IPv4 nach IPv6
 - IPv4 Mapped Address (repräsentiert der IPv6-Anwendung IPv4-Adresse als IPv6 Adresse)

| | | |
|--------------|--------------|-----------------------|
| '0' (80 bit) | '1' (16 bit) | IPv4 Address (32 bit) |
|--------------|--------------|-----------------------|
 - IPv4 Compatible Address (zum Tunneln von IPv6-Paketen über IPv4-Router)

| | |
|--------------|-----------------------|
| '0' (96 bit) | IPv4 Address (32 bit) |
|--------------|-----------------------|
- Einfache Abbildung zwischen IPv4- und IPv6-Adressen.
 - Ergeben die gleiche Prüfsumme wie die entsprechenden IPv4-Adressen.
 - keine Neuberechnung des TCP-Pseudo-Headers nötig.



Multicast-Adressen in IPv6

- Multicast-Adressen identifizieren eine Menge von Netzinterfaces
- Multicast-Adressen sind durch ein spezielles Präfix gekennzeichnet:



- Zwei Typen von Multicast-Gruppen (Unterscheidung durch das Feld "Flag")
 - Permanente Gruppen existieren dauerhaft
 - Transiente Gruppen existieren nur zeitweise
- Gültigkeitsbereich der Adresse wird durch das Feld "Scope" festgelegt
- Einige Multicast-Adressen sind bereits für bestimmte Zwecke reserviert
 - (z.B. "All Systems on this Subnet", "DVMRP Routers")
 - FF02:0:0:0:0:0:0:1 → All Nodes Link-Local
 - FF05:0:0:0:0:0:0:101 → All NTP-Servers Site-Local

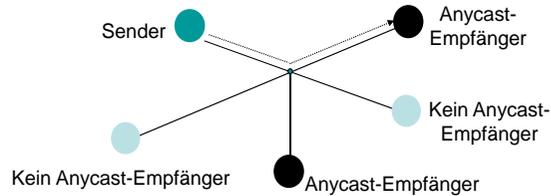


Multicast II

- IPv6 Multicast-Adressen werden auch auf MAC Multicast-Adressen gemappt. Die letzten 32 Bit der Layer 3-Adresse werden in die Ethernet MAC-Adresse kopiert.
 - IPv6: FF3E:40:2001:660:3007:123:1234:5678
 - MAC: 33-33-12-34-56-78
- Das 8. Bit einer Ethernet MAC-Adresse zeigt an, ob es sich um eine Unicast- oder Broadcast/Multicast-Adresse handelt.
 - Hier auf 1.
- Ethernet-Switches können heute meistens kein Multicast.
 - Nur durch Parsen von Layer3-Informationen könnten sie Multicast-Gruppenzugehörigkeiten lernen.
 - Pakete werden üblicherweise als Broadcast behandelt → belastigen doch alle Rechner.
 - Bei Rechnern, die nicht in der Gruppe sind, werden die Pakete auf Layer 3 verworfen.

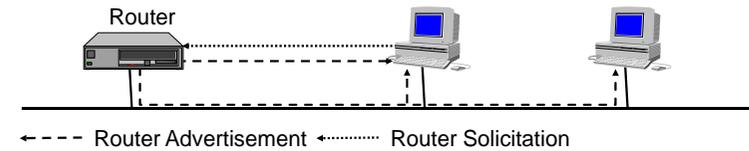
Anycast-Adressen in IPv6

- Die an eine Anycast-Adresse gesendete Dateneinheit wird an ein Mitglied der Anycast-Gruppe ausgeliefert.



- Anycast-Adressen entsprechen syntaktisch den Unicast-Adressen
- Es ist nicht gewährleistet, dass aufeinanderfolgende Anycast-Dateneinheiten an den gleichen Empfänger übermittelt werden
- Anycast kann z.B. zur Lokalisierung von Netz-Ressourcen genutzt werden

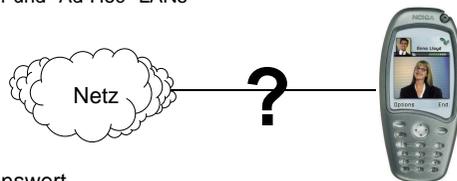
Router-Erkennung



- Problem:**
 - Ermittlung eines Routers zum Senden von Dateneinheiten an Rechner außerhalb des eigenen Netzsegmentes.
- Lösung:**
 - Router senden periodisch *Router Advertisement* Nachrichten an die "All Hosts" Adresse `FF02::1`. Diese enthalten unter anderem:
 - Router Lifetime*: Zeitspanne bis zur Ungültigkeit der enthaltenen Information
 - Rechner können mittels *Router Solicitation* ein *Router Advertisement* anfordern, welches dann jedoch per Unicast gesendet wird.

Automatische Systemkonfiguration

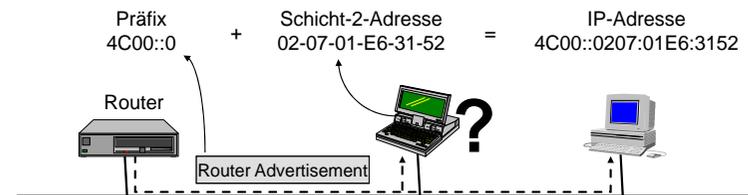
- Problem**
 - Aufwand durch manuelle Konfiguration von Internet-Rechnern (z.B. eigene IP-Adresse, Adresspräfix, Nameserver), keine Unterstützung für mobile Rechner und "Ad-Hoc"-LANs



- Wünschenswert**
 - "Plug'n Play" Lösung, d.h. Inbetriebnahme eines Internet-Rechners ohne manuelle Konfiguration allein durch physikalische Anbindung an das Netz
- Generell hat jeder Client sofort eine Link-Local IPv6-Adresse**
 - `FE80::<Interface ID>`

Automatische Systemkonfiguration durch Neighbor Discovery (RFC 2461)

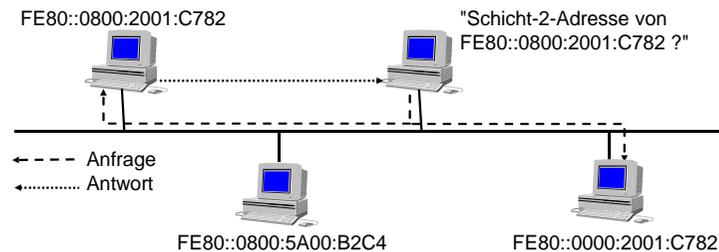
- Präfix des Subnetzes ist global eindeutig
- Schicht-2-Adresse ist zumindest innerhalb des Subnetzes eindeutig. Daraus lässt sich (nach verschiedenen Methoden) ein EUI64 Interface Identifier berechnen.
- IP-Adresse (im ganzen Internet routbar) kann aus dem Präfix des Subnetzes und dem EUI64-Identifizier zusammengesetzt werden.
- Neighbor Discovery verwendet ICMPv6-Nachrichten *Router Advertisement*, *Router-Solicitation*, *Neighbor-Solicitation*, *Neighbor-Advertisement*, ...
- Vereinfachte Darstellung:**



DHCPv6

- Nach automatischer Vergabe der IPv6-Adresse werden weitere Informationen wie z.B. DNS-Server per DHCPv6 erfragt
 - Stateless: Weder Client noch Server müssen sich Gültigkeiten merken
- Alternative: Normales DHCPv6 (RFC 3315)
 - Wer mehr Kontrolle über die Rechner im eigenen Netz haben möchte, kann die Adressen auch wie bisher per DHCP vergeben.
 - UDP als Transportprotokoll
 - Client verwendet als Quelladresse seine Link-Local IPv6-Adresse
 - Zieladresse All_DHCP_Relay_Agents_and_Servers / FF02::1:2
 - IPv6-Philosophie: Multicast statt Broadcast

Neighbor Discovery



- **Problem:**
 - Abbildung einer IPv6-Adresse auf die entsprechende Schicht-2-Adresse
- **Lösung:** Anfrage mittels *Neighbor Solicitation*
 - Rechner hören auf *Solicited Nodes Address* (Präfix FF02::1:FF gefolgt von den letzten 24 Bit (z.B. 01:C782) der Unicast-Adresse, z.B. FF02::1:FF01:C782)
 - Anfrage an die *Solicited Nodes Address* des Zielrechners (per L2-Multicast)
 - Zielrechner übermittelt seine Schicht-2-Adresse an den anfragenden Rechner

Präfix-Erkennung durch Neighbor Discovery

- **Problem:**
 - Absender einer IP-Dateneinheit muss feststellen, ob sich der Zielrechner im eigenen Subnetz befindet (direktes Senden oder Senden über Router)
- **Lösung:**
 - Entscheidung basierend auf dem Präfix des eigenen Subnetzes.
 - Router Advertisement Nachrichten enthalten Präfix-Listen des Subnetzes.
 - Vergleich der Zieladresse mit den Präfixen durch logische UND-Verknüpfung.
 - Entspricht das Präfix der Zieladresse einem Präfix des Subnetzes, so wird das Paket direkt gesendet; ansonsten wird es an einen Router übermittelt.
- **Beispiel:**
 - Ein Rechner im Subnetz mit dem Präfix 4C00:0:1::/48 möchte an den Rechner mit Adresse 4C00:0:2::0800:5A01:3982 senden:

```
4C00:0000:0002::0800:5A01:3982
AND  FFFF:FFFF:FFFF::0000:0000:0000
=====
4C00:0000:0002::
```

(ungleich dem Präfix des Subnetzes → an einen Router übermitteln)

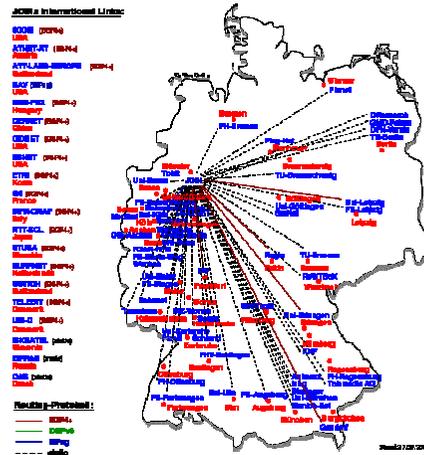
Übergang von IPv4 zu IPv6

- Langsamer Übergang geplant, keine abrupte Umstellung
 - wird sich über mehrere Jahre hinweg ziehen
- Zwei Strategien können verfolgt werden
 - Doppelter Stack
 - Während der Übergangsphase implementiert jeder IPv6-Knoten zusätzlich noch den IPv4-Protokollstack
 - IPv6 Tunneling
 - Tunnels werden benutzt, um IPv4-Pfade zu überbrücken. Diese Technik wurde im 6Bone angewendet
- Erforderliche Veränderungen in anderen Protokollen:
 - Modifiziertes Socket-API ist erforderlich
 - Modifizierte Routingprotokolle müssen eingeführt werden
 - DNS muss IPv6-Adressen unterstützen



Migration von IPv4 nach IPv6: Das 6Bone

- Weltweites „Overlay“-Netz zur Entwicklung und zum Testen von IPv6
- Das 6Bone verband IPv6-Subnetze über IPv4-Netze durch Konfiguration von Tunneln
- Abgeschaltet im Jahre 2006.



5.12 Internet und Mobilität

- Wegwahl
 - basiert auf IP-Zieladresse, Netzwerk-Präfix (z.B. 129.13.42) legt physikalisches Subnetz fest
 - wird das Subnetz gewechselt, **muss** auch die IP-Adresse passend gewechselt werden (bei IP ohne Mobilitätsunterstützung) oder ein spezieller Routing-Eintrag vorgenommen werden
- Spezifische Routen zum Endgerät?
 - anpassen aller Routing-Einträge, damit Pakete umgeleitet werden
 - skaliert nicht mit Anzahl der mobilen Geräte und u.U. häufig wechselnden Aufenthaltsorten, Sicherheitsprobleme
- Wechseln der IP-Adresse?
 - je nach Lokation wird entsprechende IP-Adresse gewählt
 - wie sollen Rechner nun gefunden werden – DNS-Aktualisierung dauert lange
 - TCP-Verbindungen brechen ab, Sicherheitsprobleme!

⇒ **Mobile IP**



Zukunft von IPv6

- Mittlerweile ein Regeldienst – neben IPv4
- Sollte IPv4 ersetzen, ob und wann aber derzeit nicht absehbar
- Ursprüngliche Motivation für IPv6:
 - Anwachsende Benutzerzahlen, Adressknappheit
 - Gegenmaßnahmen: **CIDR, NAT**; dadurch vorübergehend kein dringender Handlungsbedarf mehr im Festnetz – dafür aber im 3G-Mobilnetz!
 - Aktuell: IPv4-Adressen werden noch etwas mehr als 1 Jahr reichen.
 - Neuartige Dienste erfordern Unterstützung
 - Einsatz von QoS/RSVP umstritten, daher kein dringender Bedarf für flow-label-Feld in IPv6
 - Nutzung von Type-of-Service-Feld in IPv4 möglich, um Dienstgüteunterstützung zu realisieren („Differentiated Services“-Architektur)
 - Hohe Datenraten erfordern effizientes Paketformat
 - könnte bei breitem Einsatz von Verschlüsselungsverfahren auf Netzwerkschicht an Bedeutung gewinnen

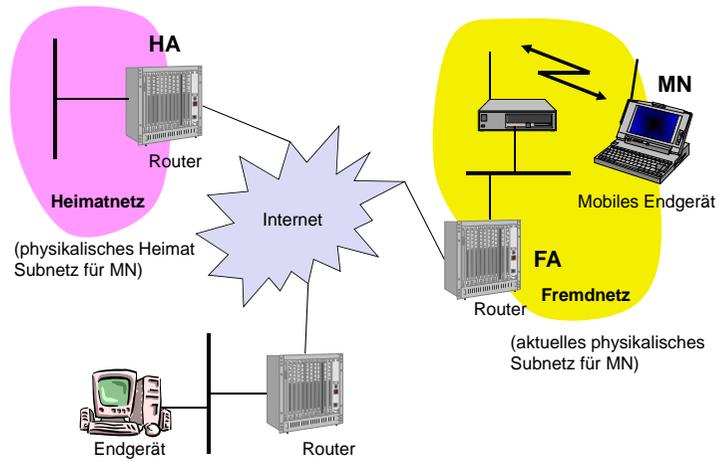


Terminologie

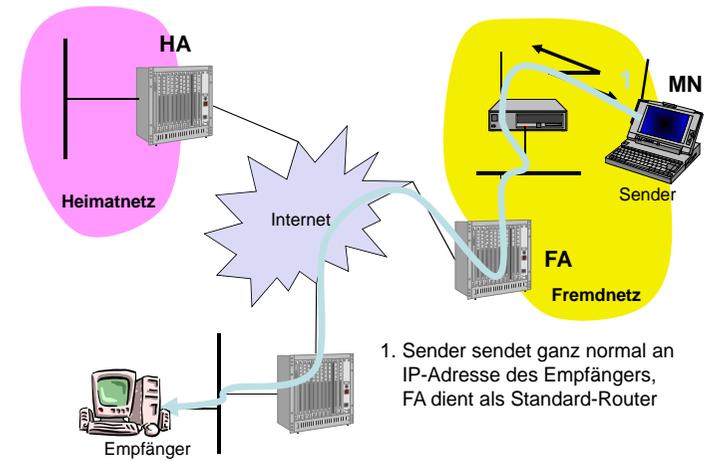
- Mobile Node (MN)
 - Knoten, der den Ort des Netzanschlusses wechseln kann, ohne seine IP-Adresse ändern zu müssen
 - typischerweise mobiles Endgerät
- Home Agent (HA)
 - Einheit im „Heimatnetz“ des MN, typischerweise Router
 - verwaltet Aufenthaltsort des MN, tunnelt IP-Datagramme zur COA
- Foreign Agent (FA)
 - Einheit im momentanen „Fremdnetz“ des MN, typ. Router
 - weiterleiten der getunnelten Datagramme zum MN, stellt meist auch default-Router für den MN dar, stellt COA zur Verfügung
- Care-of Address (COA)
 - Adresse des für den MN aktuell gültigen Tunnelendpunkt
 - stellt aus Sicht von IP aktuelle Lokation des MN dar
 - kann z.B. via DHCP gewählt werden



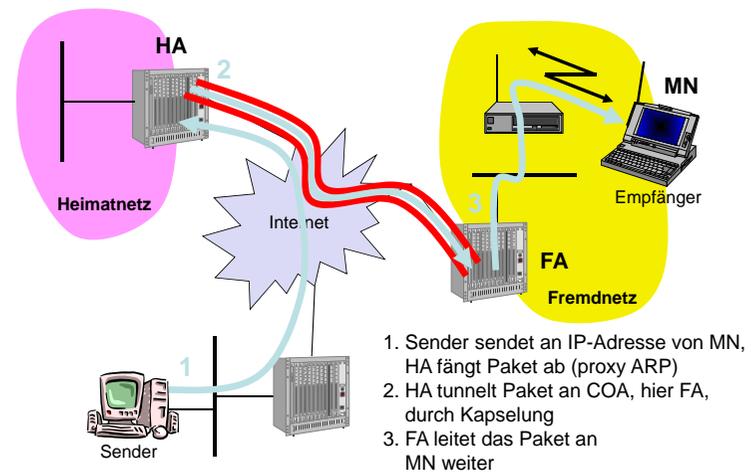
Beispielnetz



Datentransfer vom Mobilrechner



Datentransfer zum Mobilrechner



Einige Probleme mit Mobile IP

- **Sicherheit**
 - Authentifizierung mit FA problematisch, da u.U. nicht unter eigener Kontrolle (fremde Organisation)
 - Sicherheitsprotokolle, Schlüsselverwaltung und -verteilung
- **Firewalls**
 - verhindern typischerweise den Einsatz von Mobile IP, spezielle Konfigurationen sind nötig (z.B. reverse tunneling)
- **QoS**
 - häufige erneute Reservierungen im Fall von RSVP
 - Tunneln verhindert das Erkennen eines gesondert zu behandelten Datenstroms
- Sicherheit, Firewalls, QoS etc. sind aktueller Gegenstand vieler Arbeiten und Diskussionen!