Digitale Kommunikation und Internetdienste 1

Wintersemester 2004/2005 – Teil 8

Belegnummer Vorlesung: 39 30 02

Übungen: 39 30 05

Jan E. Hennig

AG Rechnernetze und Verteilte Systeme (RVS)
Technische Fakultät
Universität Bielefeld

jhennig@rvs.uni-bielefeld.de

basierend auf den Arbeiten von Michael Blume, Heiko Holtkamp, Marcel Holtmann und I Made Wiryana

Übersicht

- Internetschicht: Internet Protocol Version 6 (IPv6)

Erinnerung

- Internet Protokoll Version 6 – IPv6
- auch: IP Next Generation (IPnG)
- vorrangiger Grund: begrenzter Adreßraum
- klassenlose Netzwerkeinteilung und CIDR schaffen zwar etwas Luft
- dennoch ist klar absehbar, daß auch diese Maßnahmen nicht ausreichen
- z.B. Video on demand, Internettelefonie, Vernetzung eines jeden Haushaltgerätes

IPv6-Ziele

- Hauptziele bei der Entwicklung von IPv6:
  - Unterstützung von Milliarden von Hosts, auch bei ineffizienter Nutzung des Adreßraums
  - Reduzierung des Umfangs der Routing-Tabellen
  - Vereinfachung des Protokolls, damit Router Pakete schneller abwickeln können
  - Höhere Sicherheit (Authentifikation und Datenschutz) als das heutige IP
  - Mehr Gewicht auf Dienstarten, insbesondere für Echtzeitanwendungen
  - Unterstützung von Multicasting durch die Möglichkeit, den Umfang zu definieren
  - Möglichkeit für Hosts, ohne Adreßänderung auf Reise zu gehen
  - Möglichkeit für das Protokoll, sich zukünftig weiterzuentwickeln
  - Unterstützung der alten und neuen Protokolle in Koexistenz für Jahre

IPv6 RFCs

- ursprüngliches RFC:
- RFC 1883, Internet Protocol, Version 6 (IPv6) Specification, Dec. 1995
- im Folgenden:
- RFC 2460, Internet Protocol, Version 6 (IPv6) Specification, Dec. 1998
- enthält einige wesentliche Änderungen der Spezifikation

IPv6-Merkmale (1)

- viele der als erfolgreich betrachteten Merkmale von IPv4 bleiben in IPv6 voll erhalten
- dennoch ist es im allgemeinen nicht mit IPv4 kompatibel
- wohl aber zu den Protokollen von Transportschicht und Netzwerkschicht
- nach geringfügigen Modifikationen:
- Transportschichtprotokolle müssen IPv6-Adressen weiterreichen können

IPv6-Merkmale (2)

gegenüber IPv4 wichtigste Änderungen:

- Adreßgröße
- Header-Format
- Erweiterte Unterstützung von Optionen und Erweiterungen
- Dienstarten
- Sicherheit
- Erweiterbarkeit

IPv6-Merkmale (3)

- Adreßgröße:
- wichtigstes Merkmal von IPv6 gegenüber IPv4: größere Adressen
- statt bisher 32 Bit
- nun 128 Bit
- theoretisch lassen sich damit $2^{128} = 3.4 \times 10^{38}$ Adressen vergeben

IPv6-Merkmale (4)

- Header-Format:
- der (Basis-)Header wurde vollständig geändert
- er enthält nur 7 statt bisher 13 Felder
- diese Änderung ermöglicht Routern, Pakete schneller zu verarbeiten
- im Gegensatz zu IPv4 gibt es bei IPv6 nicht mehr nur einen Header, sondern mehrere Header
- ein Datengramm besteht aus einem Basis-Header, sowie einem oder mehreren Zusatz-Headern, gefolgt von den Nutzdaten.

IPv6-Merkmale (5)

- Erweiterte Unterstützung von Optionen und Erweiterungen:
- einige, bei IPv4 notwendige Felder sind nun optional
- daher ist die Erweiterung der Optionen notwendig geworden
- darüber hinaus unterscheidet sich auch die Art, wie die Optionen dargestellt werden
- für Router wird es damit einfacher, Optionen, die nicht für sie bestimmt sind, zu überspringen
- dies ermöglicht ebenfalls eine schnellere Verarbeitung von Paketen

IPv6-Merkmale (6)

- Dienstarten:
- IPv6 legt mehr Gewicht auf die Unterstützung von Dienstarten
- damit kommt IPv6 den Forderungen nach einer verbesserten Unterstützung der Übertragung von Video- und Audiodaten entgegen
- IPv6 bietet hierzu eine Option zur Echtzeitübertragung

IPv6-Merkmale (7)

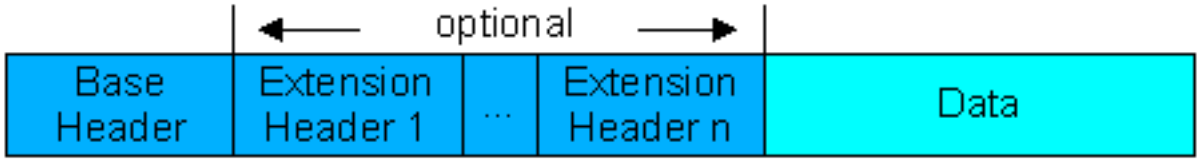
- Sicherheit:
- IPv6 beinhaltet nun im Protokoll selbst Mechanismen zur sicheren Datenübertragung
- wichtige neue Merkmale von IPv6 sind hier:
- *Authentifikation (authentication)*
- *Datenintegrität (data integrity)*
- *Datenverlässlichkeit (data confidentiality)*

IPv6 Merkmale (8)

- Erweiterbarkeit:
- IPv6 ist ein erweiterbares Protokoll
- bei der Spezifikation des Protokolls wurde nicht versucht alle potentiell möglichen Einsatzfelder für das Protokoll in die Spezifikation zu integrieren
- vielmehr bietet IPv6 die Möglichkeit das Protokoll zu erweitern
- dies kann über Erweiterungs-Header geschehen
- damit ist das Protokoll offen für zukünftige Verbesserungen

IPv6 Datengramm

- ein IPv6-Datengramm besteht aus
  - dem Basis-Header
  - optionalen Zusatz-Headern
  - den Nutzdaten



IPv6-Basisheader (1)

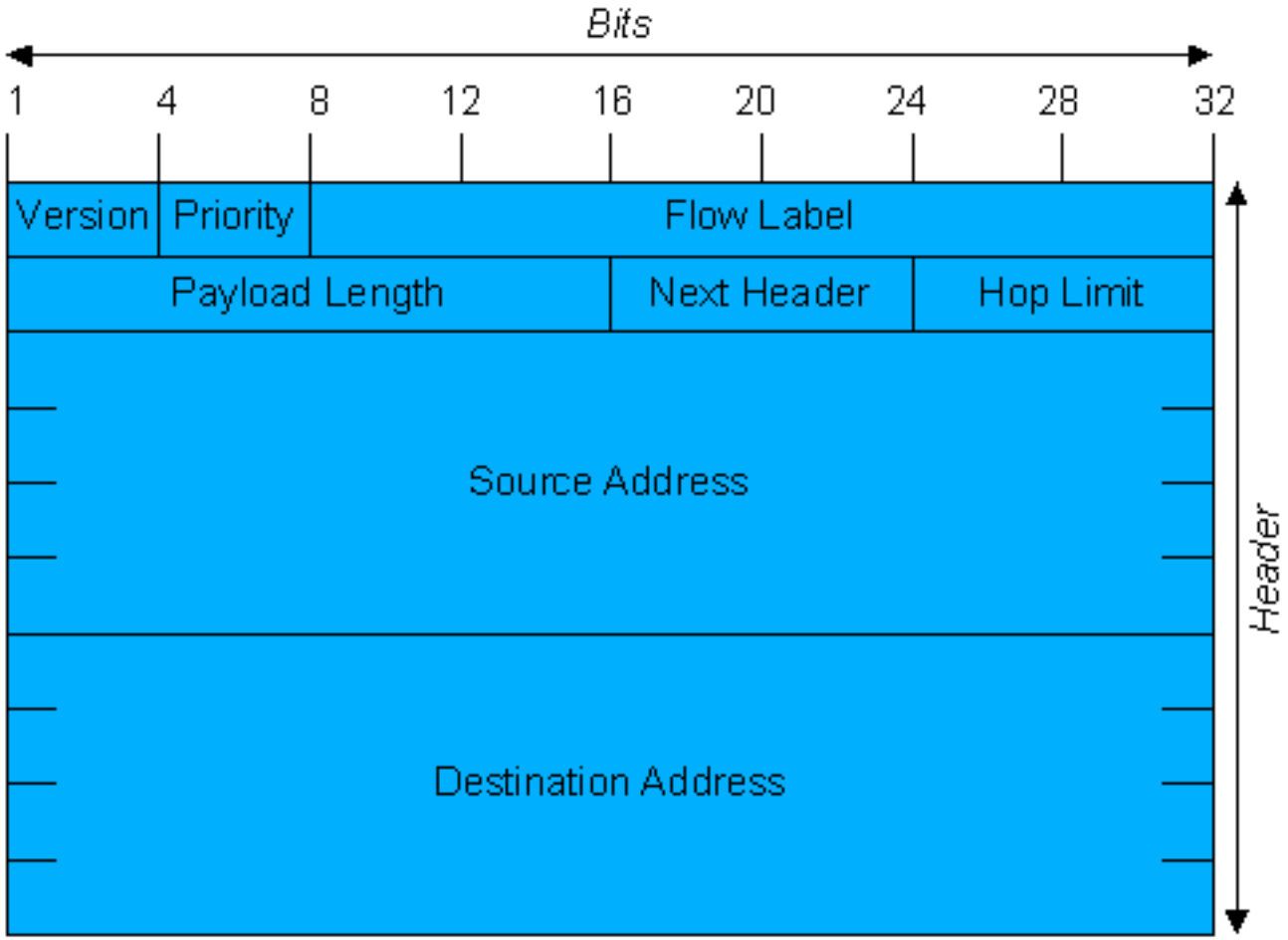
- der IPv6-Basis-Header ist doppelt so groß wie der IPv4-Header
- der IPv6-Basis-Header enthält aber weniger Felder als der IPv4-Header,
- dafür ist die Adreßgröße für die Quell- und Zieladresse von bisher 32 Bit auf nunmehr 128 Bit erweitert worden
- diese verbrauchen entsprechend viel Platz

IPv6-Basisheader (2)

Felder im Basisheader:

- Version
- Priority (auch: Traffic Class)
- Flow Label
- Payload Length
- Next Header
- Hop Limit
- Source Address
- Destination Address

IPv6-Basisheader (3)



IPv6-Basisheader (4)

- Version:
- damit können Router überprüfen, um welche Version des Protokolls es sich handelt
- für ein IPv6-Datengramm ist dieses Feld immer 6
- und für ein IPv4-Datengramm dementsprechend immer 4
- mit diesem Feld ist es möglich für eine lange Zeit die unterschiedlichen Protokollversionen IPv4 und IPv6 nebeneinander zu verwenden
- wenn der Router beide versteht
- über die Prüfung des Feldes Version können die Daten an das jeweils richtige „Verarbeitungsprogramm“ weitergeleitet werden

IPv6-Basisheader (5)

- Priority (auch: Traffic Class)
- bezeichnet die Priorität mit der ein Paket behandelt werden soll
- Prioritäten richten sich nach der Klassifizierung der Daten:

Wert	Bedeutung	Beispiel
0	nicht genau spezifizierte Inhalte	
1	Fülldaten	Netnews
2	nicht zeitkritische Daten	E-Mail
3	reserviert	
4	schnelle Übertragung großer Datenmengen	FTP
5	reserviert	
6	interaktive Anwendungen	Telnet, ssh
7	netzrelevante Steuerungsdaten	Netzmanagment, Routing Protokolle
8 - 15	zeitkritische Anwendungen	Audio, Video, Multimedia

IPv6-Basisheader (6)

- Flow Label:
- zufälliger Identifikator (ID)
- bezeichnet eine virtuellen Ende-zu-Ende-Verbindung
- zum Zuordnen des Datenflusses an der Gegenstelle

IPv6-Basisheader (7)

- Payload Length:
- bezeichnet die Nutzdatenlänge
- gibt an, wie viele Bytes dem IPv6-Basis-Header folgen
- der IPv6-Basis-Header ist ausgeschlossen
- die Erweiterungs-Header werden bei der Berechnung der Nutzdatenlänge mit einbezogen
- das entsprechende Feld wird in der Protokollversion 4 mit Total Length bezeichnet
- allerdings bezieht IPv4 den 20 Byte großen Header auch mit in die Berechnung ein

IPv6-Basisheader (8)

- Next Header:
- gibt an, welcher Erweiterungs-Header dem IPv6-Basis-Header folgt
- jeder folgende Erweiterungs-Header beinhaltet ebenfalls ein Feld Next Header
- das dort auf den nachfolgenden Header verweist
- ist dies der letzte zu IPv6 zugehörige Header, so gibt das Feld an, welches Transportprotokoll (z.B. TCP oder UDP) folgt
- dazu wurden die Lücken der standardisierten IPv4-Protokollnummern genutzt

Ausschnitt aus /etc/protocols


```
ip          0     IP              # internet protocol, pseudo protocol number
icmp        1     ICMP            # internet control message protocol
igmp        2     IGMP            # Internet Group Management
ggp         3     GGP             # gateway-gateway protocol
ipencap     4     IP-ENCAP        # IP encapsulated in IP (officially ‘‘IP’’)
st          5     ST              # ST datagram mode
tcp         6     TCP             # transmission control protocol
egp         8     EGP             # exterior gateway protocol
pup         12    PUP             # PARC universal packet protocol
udp         17    UDP             # user datagram protocol
hmp         20    HMP             # host monitoring protocol
xns-idp     22    XNS-IDP         # Xerox NS IDP
rdp         27    RDP             # "reliable datagram" protocol
iso-tp4     29    ISO-TP4         # ISO Transport Protocol class 4
xtp         36    XTP             # Xpress Tranfer Protocol
ddp         37    DDP             # Datagram Delivery Protocol
idpr-cmtp   38    IDPR-CMTP       # IDPR Control Message Transport
ipv6        41    IPv6            # IPv6
ipv6-route  43    IPv6-Route      # Routing Header for IPv6
ipv6-frag   44    IPv6-Frag       # Fragment Header for IPv6
idrp        45    IDRP            # Inter-Domain Routing Protocol
gre         47    GRE             # General Routing Encapsulation
esp         50    ESP             # Encap Security Payload for IPv6
ah          51    AH              # Authentication Header for IPv6
ipv6-icmp   58    IPv6-ICMP       # ICMP for IPv6
ipv6-nonxt  59    IPv6-NoNxt      # No Next Header for IPv6
ipv6-opts   60    IPv6-Opts       # Destination Options for IPv6
rspf        73    RSPF            # Radio Shortest Path First.
ospf        89    OSPFIGP         # Open Shortest Path First IGP
ipip        94    IPIP            # IP-within-IP Encapsulation Protocol
pim         103   PIM             # Protocol Independent Multicast
```

IPv6-Basisheader (9)

- Hop Limit:
- damit wird festgelegt, wie lange ein Paket überleben darf
- der Wert des Feldes wird nach jeder Teilstrecke gesenkt
- ein Datengramm wird dann verworfen, wenn das Feld Hop Limit auf Null herunter gezählt ist
- bevor das Datengramm sein Ziel erreicht hat
- IPv4 verwendet hierzu das Feld Time to Live, welches die Zeit in Sekunden angibt, die ein Paket überleben darf
- allerdings wird dieses Feld von den meisten Routern nicht so interpretiert
- in IPv6 wurde das Feld deshalb umbenannt, um die tatsächliche Nutzung wiederzugeben

IPv6-Basisheader (10)

- Source Address, Destination Address:
- Felder für Quell- und Zieladresse
- diese dienen zur Identifizierung des Senders und Empfängers eines IP-Datengramms
- IPv6 verwendet zur Adressierung 4 mal so große Adressen wie IPv4
- 128 Bit statt 32 Bit (16 Byte statt 4 Byte)

Adreßaufbau (1)

- es gibt wie für IPv4 auch eine IPv6-Notation
- x:x:x:x:x:x:x:x
- jedes x repräsentiert hexadezimal einen 16-Bit-Abschnitt der Adresse
- z.B. 47CD:1234:AC02:0022:1234:A456:0124
- Abkürzung möglich bei aufeinanderfolgenden Nullen
- z.B. 47CD:0000:0000:0000:0000:0000:A456:0124
- kann geschrieben werden als 47CD::A456:0124

Adreßaufbau (2)

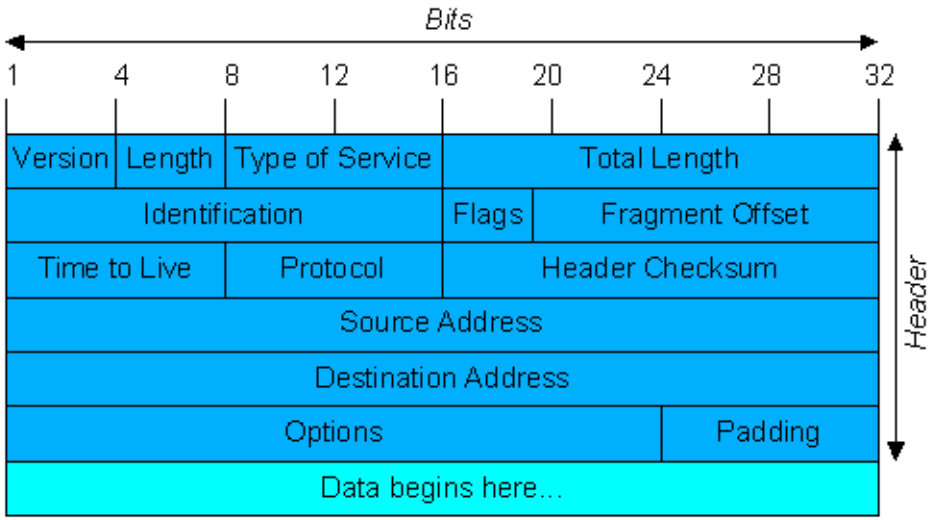
- IPv4-Adressen können „integriert” werden
- dazu werden die letzten 4 Byte in herkömmlicher IPv4-Dezimalnotation geschrieben
- z.B. aus 128.96.33.81
- wird ::00FF:128.96.33.81
- Vergabe eingeteilt in mehrere Stufen, *Aggregatable Global Unicast Addresses*
- RegistryID, ProviderID, SubscriberID, SubnetID, InterfaceID

Adreßaufbau (3)

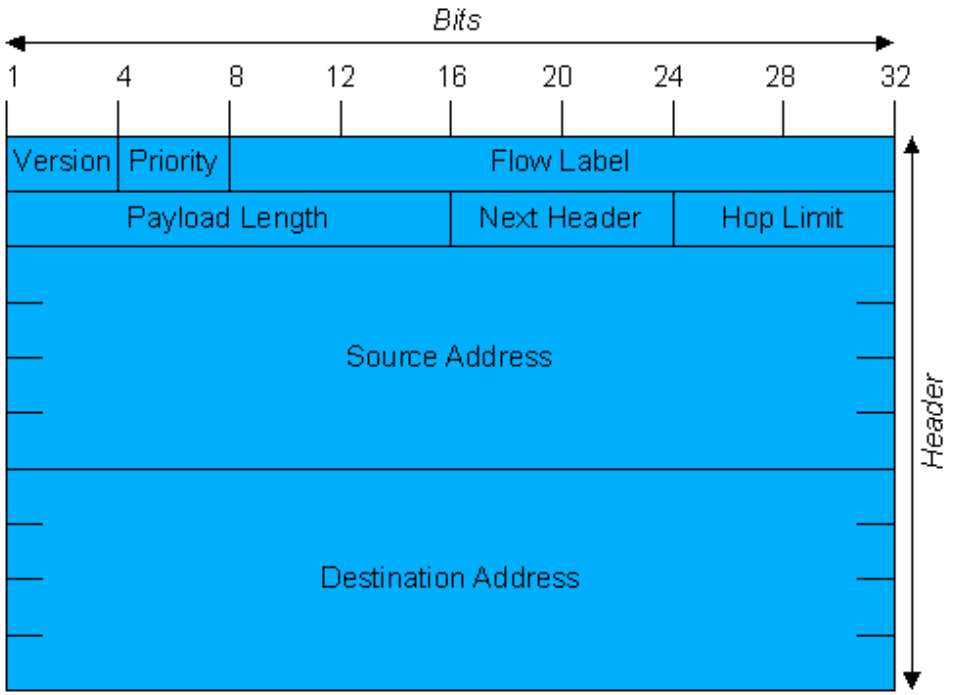
- einige Adreßbereiche sind für spezielle Anwendungen reserviert
- Präfix 0000 001 reserviert für OSI NSAP (network service access point)
- Präfix 0000 010 reserviert für IPX
- Präfix 001 reserviert für Aggregatable Global Unicast Addressen
- Präfix 1111 1110 10 reserviert für rein lokale Benutzung (link local)
- Präfix 1111 1110 11 reserviert für rein lokale Benutzung (site local)
- Präfix 1111 1111 reserviert für Multicast

Vergleich der Header

IPv4



IPv6



Weggelassene Felder (1)

- bei IPv6 wurden Felder aus IPv4 weggelassen:
- Length (Internet Header Length – IHL)
- Protocol (Funktion integriert durch *Next Header*)
- Checksum
- Fragmentierung-Felder Identification, Flags, Fragment Offset

Weggelassene Felder (2)

- Das Feld *Length (Internet Header Length - IHL)* ist nicht mehr vorhanden
- da der IPv6-Basis-Header eine feste Länge von 40 Byte hat
- bei IPv4 ist dieses Feld notwendig, da der Header aufgrund der Optionen eine variable Länge hat
- Das Feld *Protocol* wird nicht mehr benötigt
- da das Feld *Next Header* angibt, was nach dem letzten IP-Header folgt (z.B. TCP oder UDP)

Weggelassene Felder (3)

- Das Feld *Checksum* ist nicht mehr vorhanden
- die Berechnung der Prüfsumme hat sich nachteilig auf die Leistung der Datenübertragung ausgewirkt
- Das Entfernen der Prüfsumme aus dem Internet Protokoll hat zu heftigen Diskussionen geführt
- die eine Seite kritisierte heftig das Entfernen der Prüfsumme
- die andere Seite argumentierte, daß Prüfsummen etwas sind, das auch von Anwendungen übernommen werden kann
- sofern sich die Anwendung tatsächlich um Datenintegrität kümmert
- weiteres Gegenargument: Prüfsumme bereits auf Transportschicht vorhanden
- letztendlich fiel die Entscheidung, daß IPv6 keine Prüfsumme enthält

IPv6-Fragmentierung (1)

- alle Felder, die zur Fragmentierung eines IPv4-Datengramms benötigt wurden, sind im IPv6-Basis-Header nicht mehr vorhanden
- Identification, Flags, Fragment Offset
- die Fragmentierung wird in IPv6 gegenüber IPv4 anders gehandhabt
- alle IPv6 kompatiblen Hosts und Router müssen Pakete mit einer Größe von 1280 Byte unterstützen
- (RFC 1883 legte diese Größe noch auf 576 Byte fest)
- dadurch wird eine Fragmentierung im Prinzip selten notwendig

IPv6-Fragmentierung (2)

- empfängt ein Router ein zu großes Paket, so führt er keine Fragmentierung mehr durch
- sondern sendet eine Nachricht an den Absender des Pakets zurück (via ICMP)
- darin wird der sendende Host angewiesen, alle weiteren Pakete zu diesem Ziel aufzuteilen
- das bedeutet, daß von den Hosts erwartet wird, daß sie von vornherein eine Datengrammgröße wählen, die keine Fragmentierung voraussetzt
- dadurch wird eine größere Effizienz bei der Übertragung erreicht
- als wenn Pakete von Routern auf dem Weg fragmentiert werden müssen
- die Steuerung der Fragmentierung erfolgt bei IPv6 über den *Fragment Header*

Erweiterungs-Header (1)

- IPv6 nutzt das Konzept der Erweiterungs-Header, um
- a) eine effiziente Datenübertragung und
- b) eine Erweiterung des Protokolls zu ermöglichen
- der Basis-Header enthält nur Felder, die unbedingt für die Übermittlung eines Datengramms notwendig sind
- erfordert die Übertragung weitere Optionen, so können diese über einen Erweiterungs-Header angegeben werden

Erweiterungs-Header (2)

- IPv6 sieht vor, das einige Merkmale des Protokolls nur gezielt benutzt werden
- z.B. die Fragmentierung von Datengrammen:
- obwohl viele IPv4-Datengramme nicht fragmentiert werden müssen, enthält der IPv4-Header Felder für die Fragmentierung
- IPv6 gliedert die Felder für die Fragmentierung in einen separaten Header aus
- der muß wirklich nur dann verwendet werden, wenn das Datengramm tatsächlich fragmentiert werden muß
- alle Erweiterungs-Header sind optional

Erweiterungs-Header (3)

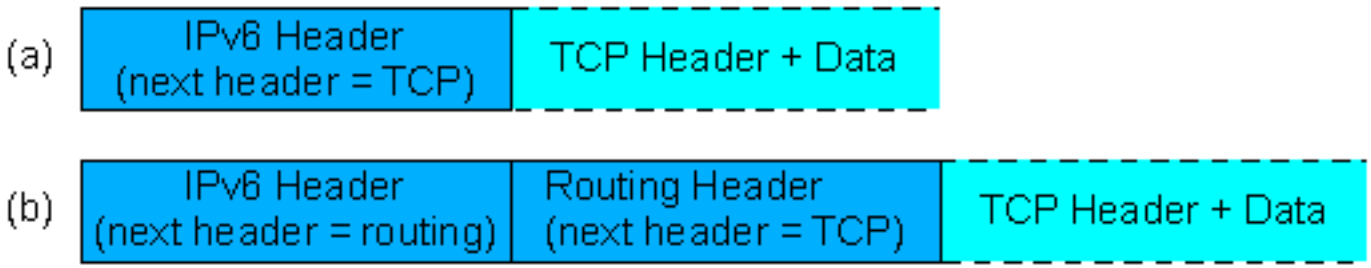
- weiterer wesentlicher Vorteil:
- das Protokoll kann um neue Funktionen erweitert werden
- es genügt, für das Feld Next Header einen neuen Typ und ein neues Header-Format zu definieren
- IPv4 erfordert hierzu eine vollständige Änderung des Headers
- derzeit sind 6 IPv6-Erweiterungs-Header definiert
- werden mehrere Erweiterungs-Header verwendet, so ist es erforderlich, sie in einer festen Reihenfolge anzugeben

Erweiterungs-Header (4)

IPv6-Basis-Header	Zwingend erforderlicher IPv6-Basis-Header
Optionen für Teilstrecken (Hop-by-Hop Options Header)	Verschiedene Informationen für Router (wird als erster Options-Header eingebaut)
Optionen für Ziele (Destination Options Header)	Zusätzliche Informationen für das Ziel (für Optionen, die nur vom endgültigen Ziel des Paketes verarbeitet werden müssen)
Routing (Routing Header)	Definition einer vollständigen oder teilweisen Route
Fragmentierung (Fragment Header)	Verwaltung von Datengrammfragmenten
Authentifikation (Authentication Header)	Echtheitsüberprüfung des Senders
Verschlüsselte Sicherheitsdaten (Encapsulating Security Payload Header)	Informationen über den verschlüsselten Inhalt
Header der höheren Schichten (Upper Layer Header)	Header der höheren Protokollschichten (TCP, UDP, . . .)

Erweiterungs-Header (5)

- IPv6 Datengramme:
- (a) IPv6-Basis-Header und Nutzdaten
- (b) IPv6-Basis-Header mit einem Zusatz-Header für Routing-Informationen, gefolgt von Nutzdaten



Options-Header (1)

- Aufbau:
- 8 Bit Next Header (Verweis auf nächsten Headertypus)
- 8 Bit Hdr Ext Len (Länge dieses Options-Headers)
- variabel: mehrere Optionen, ihrerseits aufgebaut aus
  - 8 Bit Option Type (definiert Aktion, welche bei Nichtunterstützung einer Option vom betreffenden System ausgeführt werden soll)
  - 8 Bit Option Data Len (Länge der folgenden Daten)
  - variabel: Daten dieser Option

Options-Header (2)

- durch zwei Bits des *Option Type* wird festgelegt:

00	Überspringen
01	Paket verwerfen, keine Fehlermeldung an den Absender
10	Pakte verwerfen, Fehlermeldung an Absender falls Zieladresse eine Multicast-Adresse ist
11	Pakte verwerfen, Fehlermeldung an Absender falls Zieladresse keine Multicast-Adresse ist

- ein weiteres Bit des *Option Type* left fest, ob die Optionsdaten auf dem Weg verändert werden dürfen (1) oder nicht (0)

Beispiel: Jumbo-Paket (1)

- Sonderfall: Payload Length im Basis-Header ist 0
- kennzeichnet Jumbo-Paket
- das ist ein Paket mit mehr als 65535 Bytes an Daten
- wird durch Erweiterungsheader „Hop-by-Hop Options” ermöglicht

Beispiel: Jumbo-Paket (2)

- im Basis-Header:
- Payload-Length = 0
- Next-Header = 0
- Hop-by-Hop-Header:

1 Byte	Next Header	= 6	nächstes Protokoll ist TCP
1 Byte	Hdr Ext Len	= 6	in Byte – Gesamtlänge der Options-Liste (Type + Opt Data Len + Länge)
1 Byte	Type	= 194	Falls eine Option nicht unterstützt wird, wird Paket verworfen
1 Byte	Opt Data Len		in Byte = 32 Bit
var.	Jumbo-Payload	bis 2^{32}	durch „Opt Data Len” begrenzt

Fragmentation Header

- der Fragment-Header dient, wie auch bei IPv4, zur Zerlegung von IP-Paketen, deren Größe den vereinbarten MTU-Wert überschreitet

- im Gegensatz zu IPv4 kann die Segmentierung bei IPv6 nur von der Quelle vorgenommen werden

- Felder:
  - 8 Bit *Next Header*
  - 13 Bit *Fragment Offset*
  - 32 Bit *Identification*
  - 1 Bit *M-Flag* (M=0 kennzeichnet das letzte Fragment, sonst ist M=1)

Routing Header

- dient dem *Source-Routing*: Mit diesem Header ist es möglich, den Weg des Paketes durch ein Netz festzulegen
- die Zwischenstationen werden im Routing Header eingetragen
- Felder:
  - *Next Header, Hdr Ext Len*
  - *Routing Type* = 0 (momentan nur Source Routing definiert)
  - *Segment Left* (Anzahl der restlichen „Routing-Segmente" bis zum Ziel)
  - *Strict / Loose Bit Map* (Bitfolge, welche Auskunft gibt, ob der nächste Router i direkter Nachbar des aktuellen Routers $i - 1$ ist (Bit $i = 1$) oder ob er selbst eine Route zum nächsten Router ermitteln soll (Bit $i = 0$))
  - *Address 1, . . . , Address* n

Authentication Header

- ergänzt das IP-Datagramm durch kryptographische Informationen
- dem Empfänger werden so Manipulationen am Datagramm sichtbar
- zur Berechnung der kryptographischen Informationen wird „Keyed MD5” verwendet
- zusätzlich zu diesem symmetrischen kryptographischen können optional auch andere Verfahren implementiert werden
- „Keyed MD5” ist aber zwingend vorgeschrieben

Encapsulated Security Payload (1)

- ermöglicht die Verschlüsselung der zu übertragenden Daten
- dadurch wird Vertraulichkeit gesichert
- als Standardverfahren wird DES eingesetzt
- in einigen Länder ist die Verschlüsselung verboten
- somit kann es nicht von jeder Station unterstützt werden
- zwei Grundfunktionen: Transport- und Tunnelmodus

Encapsulated Security Payload (2)

- *Transport Mode*: Nur die Daten werden verschlüsselt
- z.B. wird bei einer TCP/IP-Verbindung so das TCP-Protokoll verschlüsselt
- dient z.B. der Kennwortübertragung
- Reihenfolge: IPv6-Header – ESP-Header – Nutzdaten
- *Tunnel Mode*: ganzes IP-Paket wird verschlüsselt
- Reihenfolge: IPv6-Header – ESP-Header – IPv6-Header – Nutzdaten

AH & ESP

- AH und ESP realisieren einen gesicherten Kommunikationskanal
- jedoch müssen beide Kommunikationspartner authentisiert werden
- es ist erforderlich, alle Kommunikationspartner mit ihren public-keys zu registrieren
- dieses Problem ist derzeit noch nicht vollständig gelöst
- es gibt erste Ansätze (z.B. Internet Key Exchange – IKE)

IPv6 heute

Aus c't 25/2004, S. 44: **„IPv6: theoretisch ja, praktisch nein"**

Der Umbau von IPv4 auf die neuen, längeren IPv6-Adressen ist für die IETF (Internet Engineering Task Force) ein Dauerbrenner. Mancher ist zwar der Meinung, die technischen Grundlagen seien längst fertig gestellt, aber Tony Hain, IPv6-Experte von Cisco, betont: „Der Umstieg bedeutet einen kompletten Architekturwechsel." Und immer noch gibt es eine lange Liste von neuen Vorschlägen, die diskutiert werden müssen.

In der Praxis droht die Einführung von IPv6 noch lange Zeit an der mangelhaften Unterstützung durch Anwendungen zu scheitern. Praktisch täglich, so warnen die Experten, werden Anwendungen auf Basis von IPv4 entwickelt. Das kann sich verheerend auswirken, da nur selten auf eine saubere Trennung von Applikations- und Transportschicht geachtet wird, sodass eine spätere Umstellung der Anwendungen nur schwer möglich ist.

(Monika Ermert/Axel Kossel)

Ausblick

- viele Probleme scheinen sich mit Wechsel von IPv4 zu IPv6 lösen zu lassen
- diese „Renovierung” umfaßt aber nur die Internetschicht
- diese ist zwar zentral, aber kann nicht alle Probleme lösen
- bei sehr schnellen Datenverbindungen gibt es z.B. auch mit TCP Probleme
- IPv6 bietet aber einen vielversprechender Ansatz
- spiegelt Kompromiß aus dem Wissen und Anforderungen der Internetgemeinschaft wider

Aussicht

Themenübersicht für die kommende Vorlesung:

- Transportschicht
- TCP
- UDP

Ende Teil 8. Danke für die Aufmerksamkeit.