

IP Version 6

Felix von Leitner

Convergence

`felix@convergence.de`

Dezember 1999

Zusammenfassung

IPv6 ist die nächste Version des Internet Protokolls. Wie weit ist die Standardisierung? Welche Vor- und Nachteile gibt es? Warum haben wir es noch nicht alle?

Agenda

1. Was ist IPv6? Was ist neu?
2. Wozu brauche ich es?
3. Wieso haben wir es noch nicht?
4. Wie ist der Adressraum aufgeteilt?
5. Quality of Service

Was ist IPv6?

IPv6 ist der „Nachfolger“ des Internet Protokolls.

Das Ausgangsmotiv war die Vergrößerung der Adressen, aber IPv6 adressiert auch die anderen Probleme, die sich Anfang der 90er Jahre abzeichneten.

IPv6 ist ursprünglich 1995 in den RFCs 1883ff spezifiziert worden.

Was ist denn da anders?

- Adress-Autokonfiguration
- Bessere Routing-Effizienz durch „Flowlabel“
- Adressen haben eine Lebenszeit, Renumbering ist vorgesehen!
- Multicast, Quality of Service
- Verschlüsselung und Authentisierung, Mobile IP
- Deutlich vereinfachter Header
- Und natürlich die größeren Adressen: 128 statt 32 Bits

Treibende Kräfte hinter IPv6

1. KabelTV-Netzbetreiber

- Hat dicke Leitungen innerhalb einer Stadt
- Möchte Video, Radio und Internet anbieten
- Protokoll abhängig vom Frequenzbereich
- Wie macht er Pay-TV?
- Wie wird der Traffic geroutet?
- Massive Mehrkosten für Internet-Infrastruktur

2. Mobilfunknetzbetreiber

- 10000e Zellen, mit Glasfaser vernetzt
- Möchte Telefon und Internet anbieten
- Protokoll abhängig von Streckenteil
- Internet über digitale „Leitung“?
- Kann Zellen nicht halbjährlich upgraden
- Massive Mehrkosten für Internet-Infrastruktur

3. IP-Backbone-Betreiber

- Hat dicke Leitungen zwischen Städten
- Möchte mit KabelTV-Netzbetreiber Telco arbeitslos machen
- Kann alle Content-Typen parallel fahren!
- Telefonie, Fernsehen verursachen keine Mehrkosten

4. Telcos

- Hat überall dicke Leitungen
- Sprachvolumen abnehmend
- Datenvolumen stark steigend
- Existenzangst! Muß IP anbieten!
- Pro Dienst andere Infrastruktur?!

Wieso haben wir denn dann kein IPv6?

1. KabelTV-Netzbetreiber

- bauen Internet-Knowhow gerade auf
- haben weniger Geld als Telcos

2. Mobilfunknetzbetreiber

- kalkulieren 1 Mrd. Handies in 2005
- haben alte Zellen noch nicht abgeschrieben
- Nächste Hardware muß länger halten

3. IP-Backbone-Betreiber

- Warten auf Kundennachfrage
- Upgraden periodisch alle Router
- Warten auf Hardware-IPv6-Router
- Solange kann man tunneln

4. Telcos

- Haben Angst vor Komplet-Umbau
- Setzen auf alle Pferde gleichzeitig

Frühere Rationale

1995: IPng requirements. RFC 1752.

- Große Routingtabellen.
- Konkret: keine Class B Netze mehr
- Prinzipiell gleich lassen, aber Erfahrungen nutzen
- Goodies integrieren:
 1. Autoconfiguration
 2. Security
 3. Real-Time Flows

Was ist geschehen?

- CIDR (Classless Inter-Domain Routing)
 1. Neue IPs topologisch vergeben
 2. Subnets variabler Prefixgröße (Class B zu groß, Class C zu klein)
- Network Address Translation, Firewalls
- IETF: genügend IP-Adressen bis nach 2010
- Routingtabellen von 90k auf 60k, jetzt wieder 70k
- Multicast, Traffic Flows, IPsec, DHCP gibt es jetzt auch fuer IPv4.

Heutige Rationale

- IP überall: Data, Voice, Audio, Video
- Nächster GSM-Standard (UMTS) unterstützt IP
- 1 Mrd Autos in 2010 mit GPS und Gelben Seiten
- PDAs!
- IP-„Entwicklungsländer“: China, Indien, Rußland
- Internet in jeder Schule
- Kabelmodem, xDSL, Wireless: 1 LAN / Familie!

Nutzen für Router

1. Viel einfacherer Header!

- Feste Headerlänge (optionale Header als Piggyback)
- Keine Checksummen (macht das Link Layer)
- Keine Fragmentierung im Router (Path MTU Discovery Pflicht)
- Flow Label, Traffic Class (QoS)
- TTL heißt jetzt „Hop Limit“
- Keine Flags

2. Adressen, Größe und optionale Header sind 64-bit aligned

3. Internet verdoppelt sich schneller als Hardware-Leistung

Nutzen für den Einzeluser

- Autokonfiguration (Notebooks!)
- IPsec
- QoS?
- MobileIP?
- (noch) Hack Value

Nachteile für den Einzeluser

- Patches sind bestenfalls β -Qualität
- Lieferumfang: AIX, *BSD, Linux.
- Angekündigt: Tru64, Solaris, Windows
- Kein IPv6-Netscape!
- NTT-Backbone Traffic-Statistik: #1: ssh, #2: ping
- Keine DNS-Infrastruktur
- Meistens IPv6 nur über (manuelle) Tunnel

Nutzen für Administratoren

- Autokonfiguration, auch Router finden sich gegenseitig
- Renumbering!!! (schmerzlos den ISP wechseln)
- Site Local IP Nummern
- Quality of Service
- Scope-Feld bei Multicast
- *Anycast* für Ausfallsicherheit

Nachteile für Administratoren

- Kein Know-How da
- Es gibt keine Schulungen
- Keine Shrink-Wrap Tools, man muß selber patchen und kompilieren
- Keine Zeit für „Spielchen“

Nutzen für den ISP

- Router sollten mit der Zeit billiger werden
- Keine Helpdesk-Anfragen wegen falschen Einstellungen
- Weniger Streß mit IP-Raum

Das sind alles (nur) *langfristige* Vorteile!

Nachteile für den ISP

Kurzfristig hat der ISP nur Nachteile:

- Neue Router kosten Geld
- Windows kommt nicht mit IPv6-Support
- Parallelbetrieb zu IPv4 ist unnötig teuer
- Kein Know-How am Markt
- Router-Firmwares nur unter NDA/Aufpreis/...

Nutzen für den Backbone-Provider

- Potential für viel effizienteres Routing
- Kleinere Routing-Tabellen durch Renumbering

Nachteile für den Backbone-Provider

- Keine *proven technology*
- Router (bislang) nicht in Hardware, d.h. sogar deutlich *langsamer* als IPv4

Details

- Keine Broadcast-Adressen mehr (Funktionalität über Multicast)
- Prefixe reserviert für:
 1. IPX
 2. NSAP (OSI)
 3. Site-Local
 4. Link-Local
- Initial: 85% des Adressraums sind reserviert

Provider Based Unicast Addresses

3	n bits	m bits	o bits	p bits	rest
010	reg	prov	subsc	subnet	intf

- reg Registry ID (z.B. RIPE)
- prov Provider ID (z.B. Snafu)
- subsc Subscriber ID (z.B. CCC Berlin)
- subnet Subnet ID (1-n Subnetze pro Link)
- intf Interface ID (z.B. FUCKUP)

Ähnlich der heutigen CIDR Adressvergabe in IPv4.

Local-Use Addresses

10 bits	n bits	rest
1111111010	0	Interface ID

- Lokale Unicast-Adressen
- Für lokale „plug-and-play“ Kommunikation
- Für Bootstrapping

Site-Local Addresses

10 bits	n bits	m bits	rest
1111111011	0	Subnetz	Interface

- Interface z.B. MAC-Adresse
- Subnetz+Interface mit anderem Prefix=globale IP
- Kein NAT oder Renumbering, wenn man ins Internet geht!

Embedded IPv4 Addresses

80 bits	16 bits	32 bits
0	0	IPv4

Für IPv6 Hosts in IPv4 Routing-Infrastruktur

80 bits	16 bits	32 bits
0	<i>FFFF</i>	IPv4

Zur Darstellung von IPv4-only Nodes

Anycast

- Sehen aus wie Unicast-Adressen
- Werden zur nächsten Instanz geroutet
- Prima für Dienst-Autoconfiguration

Multicast Addresses

8	4	4	112
11111111	Flags	Scope	Group ID

Die ersten drei Bits der Flags sind reserviert, das 4. ist bei transienten (d.h. nicht von IANA/ICANN vergebenen) Group-IDs gesetzt.

Scope geht von Node-Local bis Global.

Routing Header

- Optionale Routing-Header listen Zwischenhops
- Vergleichbar mit IPv4 „loose source routing“
- Rückroute geht rückwärts über die gelisteten Hops
- Provider-Selektion
- Mobility

Quality of Service

„Fähigkeit des Netzwerkes, einigen Anwendungen besseren oder speziellen Service anzubieten“

Das heißt: konsistente und vorhersagbare Performance:

- Bandbreite
- Verzögerung
- Jitter
- Packet Loss

Folgen von QoS im Internet?

Microsoft kauft die ganze Bandbreite?

Nicht die Anbindung entscheidet über die Performance, sondern der Geldbeutel für den Backbone? Strato, Puretec und AOL abgeklemmt?

Mehr Backbone-Priorität ist billiger als SAP zu optimieren?

Kein IRC, während der CEO seine Videokonferenz hält?

Amazon.com kann sich bunte Bilder leisten, lob.de nicht?

Alles Quatsch!

Tatsächliche Folgen von QoS im Internet!

- Router haben schon Schwierigkeiten, nur die Routing-Tabelle zu speichern! Kein Platz für Priorität pro ISP.
- Konzepte: Int-Serv vs. Diff-Serv
- Integrated Services (RFC 1633)
 - Garantierter, deterministischer Service für **einzelne Verbindungen**
 - Router muß State führen!
 - Muß so teuer sein, daß sich IRCer das nicht leisten können.

- Differentiated Services (RFC 2475)
 - Resource-Reservierung für **Traffic-Klassen**
 - Router muß keinen State führen.
 - „Traffic Shaping“

QoS: Int-Serv

- Ende-zu-Ende QoS
- Garantierte Bandbreite
- Beschränkte Verzögerung
- Signalling mit RSVP (RFC 2205)
 - Findet Route
 - sammelt Netzwerkdaten
 - reserviert Ressourcen
- Geht nur, wenn alle Router auf dem Weg zustimmen!

QoS: Warum IPv6?

- IPv4. Router erkennt Flow anhand von
 1. Absender IP
 2. Empfänger IP
 3. IP Protokoll
 4. Layer 4 Port
- IPv6. Router erkennt Flow anhand von
 1. Flow Label

Heutige IPv4-Router machen das in Hardware.

Cooler Goodies

- IPv6-Ethernetemulation über IPv4 (routebar!). Einzige Voraussetzung an die Infrastruktur: IPv4 Multicast ist aktiviert.
- „Standleitungen“. Route, Queue-Länge fest, Jitter und packet loss garantiert niedrig. (**ISDN über ein IP!**)
- Mit IPsec-Tunnel: VPN eingebaut!
- Man kann mit IPv6 eine ATM-Emulation betreiben

Der Pferdefuß

- Noch sehr beta (andererseits: noch hack value)
- Man kriegt beim JOIN einen Tunnel
- Kein ISP in Deutschland hat IPv6-PPP
- Einziger Backbone mit nativem IPv6: NTT

Ich habe Linux installiert und...

- `insmod ipv6`
- Debian hat es eingebaut (mit Userland-Basics)
- Router-Adressvergabe: `radvd`
- BIND selber patchen
- Kein IPsec/QoS/MobileIP
- Sehr eingeschränktes Routing