

Tutorial

Einführung in IP Version 6

GUUG Frühjahrs-Fachgespräch '97
26. Februar 1997 Köln

Guido Wessendorf

Universitätsrechenzentrum
Westfälische Wilhelms-Universität Münster

E-Mail: wessend@uni-muenster.de

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Themen

- Warum IPng ?
- Header-Formate
- Security
- Adressierung
- Routing
- Quality of Service
- Autokonfiguration
- Mobility
- Transition
- Aktueller Stand IPv6
- 6bone
- Implementations-Beispiele
- DFN-Projekt: IPv6 im WiN
- Weitere Info-Quellen

Warum IPng ?

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Internet-Wachstum

IPv4 wurde Anfang der 80'er Jahre entwickelt.

- Damals heutige Internet-Größe und heutiges Internet-Wachstum nicht vorhersehbar:
 - Im Juli 1996: ≈ 13 Millionen Hosts.
 - Verdopplung etwa jährlich!

IPv4 Adreß-Dilemma

- Obwohl mit 32-Bit IPv4-Adressen theoretisch etwa 4 Milliarden Systeme adressiert werden können, ist der IPv4-Adreßraum bald erschöpft:
 - Class-C-Adreßraum oft zu klein, also Vergabe von Class-B-Adressen (1990 Aufbrauch aller Class-B-Adressen für März 94 prognostiziert).
 - Zuweisung vieler Class-C-Adressen verschlimmert das Problem der Routing-Tabellen-Größe.
 - CIDR schafft zunächst Abhilfe, aber Interdomain-Routing-Tabellen-Wachstum konnte nicht reduziert werden.
- ☞ Gesamt-IPv4-Adreßraum zwischen 2005-2011 erschöpft (ALE-WG, IETF-Meeting Juli 94 Toronto)

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Explosion der Routing-Tabellen-Größe

- Inter-Domain-Routing-Tabellen wachsen exponentiell und schneller als Speichertechnologie Fortschritte macht.
- Router haben z. Z. bis zu 43.000 Routing-Einträge.
 - ➔ großes administratives und Performance-Problem.

Also:

- Scaling-Probleme von IPv4 lösen.
- Neue Technologien/Features besser unterstützen.

Stichworte: Security, Mobility, Plug&Play,
Multimedia-Anwendungen,
Quality-of-Service

☞ Neues Internet-Protokoll notwendig.

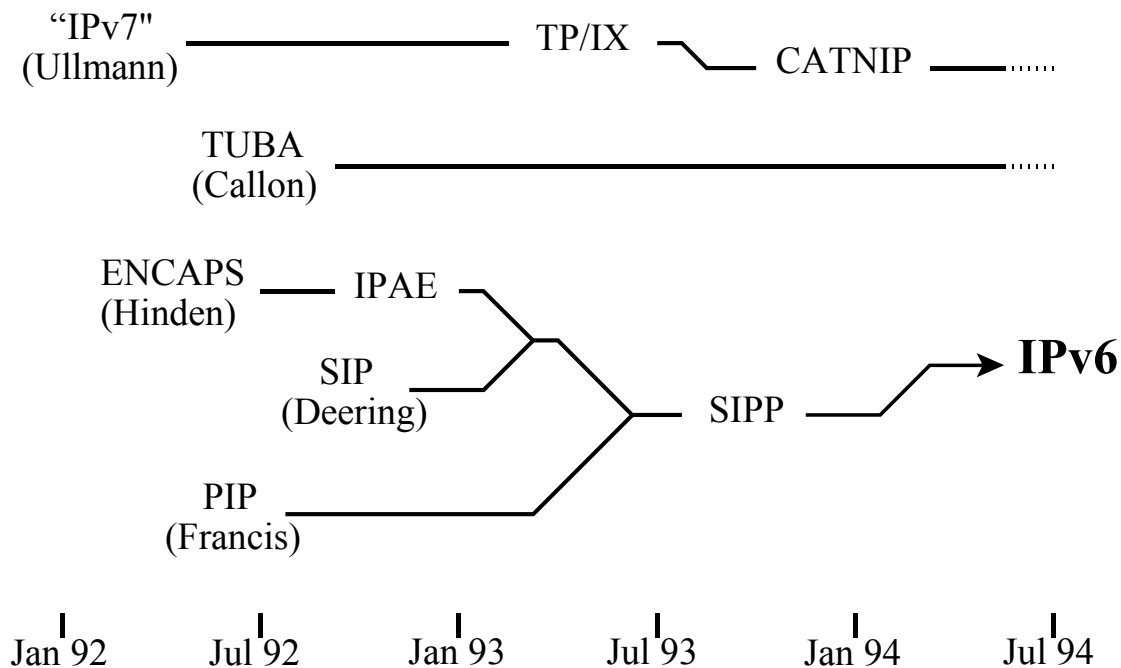
Der Anfang...

- Ende 1991 startete die IETF ihre Bemühungen um einen IPv4-Nachfolger

IP Next Generation (IPng)

- Im Juli 1992 wurden mehrere IPng-Proposals in Auftrag gegeben und Working-Groups eingerichtet.

IPng-Kandidaten



GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

IP Version 6 – Die wichtigsten Features

- erweiterte Adressierungs- und Routing-Möglichkeiten
- vereinfachtes Header-Format
- verbesserte Unterstützung von Optionen und Erweiterungen
- Möglichkeit des Flow-Labeling
- Unterstützung von Security, Autoconfiguration ("Plug & Play") und Mobility
- schrittweise ins Internet einführbar!

Header-Formate

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

IPv6-Header

```
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|Version| Prio.|                               Flow Label              |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Payload Length   | Next Header | Hop Limit    |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                                                 |
|                                                                 |
|                                Source Address                    |
|                                                                 |
|                                                                 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                                                 |
|                                                                 |
|                                Destination Address               |
|                                                                 |
|                                                                 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---
```

zum Vergleich

IPv4-Header

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|Version|  IHL  |Type of Service|          Total Length          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          Identification          |Flags|      Fragment Offset      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  Time to Live  |   Protocol   |          Header Checksum          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          Source Address          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          Destination Address          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          Options          |          Padding          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Die Header-Felder:

Version:	Protokoll-Versionsnummer, 6 = IPv6.
Prio.:	Prioritäts-Feld (später erklärt).
Flow Label:	24-Bit Flow-Label (später erklärt).
Payload Length:	Länge des IPv6-Paketes in Oktetts exklusive dem 40-Oktett langen IPv6-Basisheader. Eine 0 indiziert das Vorhandensein einer “Jumbo-Payload”-Option.
Next Header:	Identifiziert den Typ des unmittelbar folgenden Headers (benutzt gleiche Werte wie im IPv4-Protokoll-Feld).
Hop Limit:	Wird bei jedem Node welches das Paket weiterleitet um 1 erniedrigt. Falls auf 0 erniedrigt, wird das Paket verworfen.
Source Address:	128-Bit-Adresse vom Absender.
Destination Address:	128-Bit-Adresse des beabsichtigten Empfängers (nicht notwendigerweise der endgültige Empfänger, falls Routing-Header vorhanden).

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Header-Änderungen gegenüber IPv4

- Version 4 → 6
- Flow-Label neu
- Internet-Header-Length (IHL) -Feld entfernt
- Type-of-Service-Bits entfernt
- Payload-Length jetzt exklusive 40-Oktett IPv6-Header
- Fragmentation-Felder aus dem Basis-Header genommen
- Time-to-Live (TTL) → Hop-Limit
- Protocol → Next-Header
- Header-Checksum entfernt
- Adressen von 32- auf 128-Bit verlängert
- Alignment von 32- auf 64-Bit geändert
- keine Optionen- und Padding-Felder mehr

Extension-Header – Ein neues Design-Konzept

- Basis-IPv6-Header mit 40-Oktett minimal und von konstanter Länge.
- Optionen/Zusätze wirklich nur bei Bedarf.
- Zukunftssicher, da leicht weitere Extension-Header definierbar.

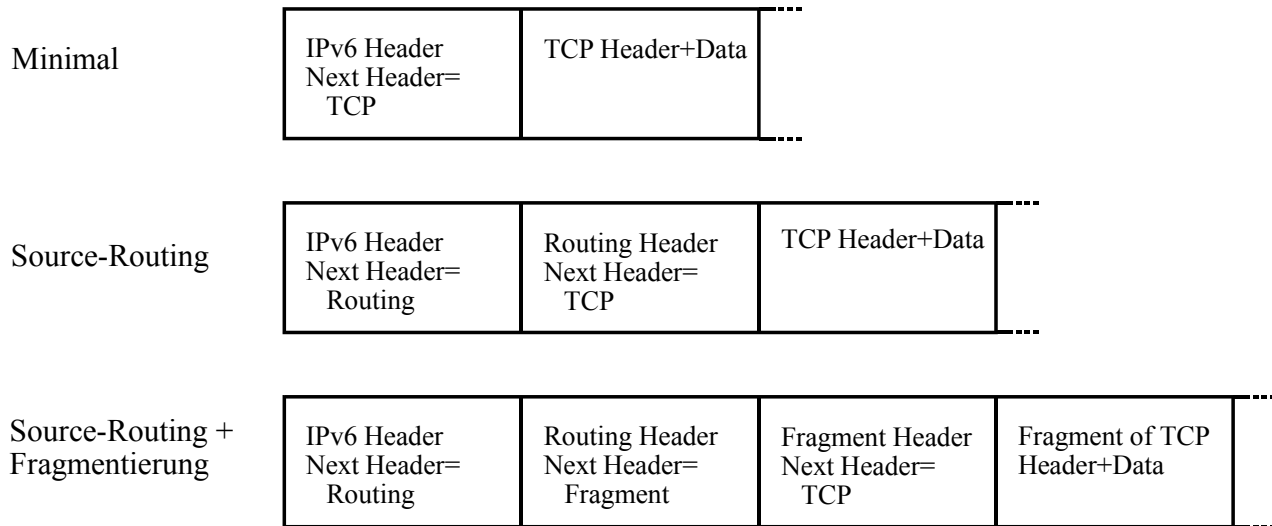
GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Derzeit definierte Extension-Header

- Hop-by-Hop Options
- Routing (Typ-0)
- Fragment
- Destination Options
- Authentication
- Encapsulating Security Payload (ESP)

Eine vollständige IPv6-Implementierung liegt nur dann vor, wenn mindestens *alle* obigen Extension-Header implementiert sind!

Extension-Header – Eine TCP-Verbindung als Beispiel

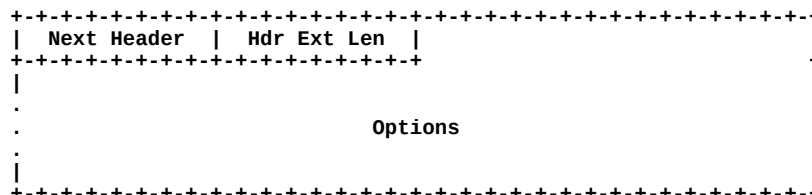


GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

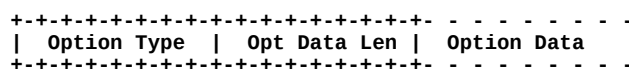
Hop-by-Hop-Options-Header

(*Next-Header = 0)

Transportiert optionale Informationen, die von jedem Node entlang des Transportweges bearbeitet werden müssen.



- Options: eine oder mehrere Type-Length-Value (TLV) kodierte Optionen:



Option-Type-Kodierung

```
+--+--+--+--+--+--+--+
|AIU|C|Option ID|
+--+--+--+--+--+--+--+
```

- AIU: “Action if Unrecognized”:
 - 00: übergehe diese Option
 - 01: verwirfe Paket
 - 10: verwirfe Paket und sende ICMP “Parameter Problem, Code 2” zum Absender.
 - 11: –”, nur wenn Zieladresse keine Multicast-Adresse ist.
- C: “Change en-route”. Wird gesetzt, wenn die Daten der Option die Route ändern können.
- Option ID: Identifiziert die Option

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Optionen-Padding und -Alignment

Es gibt zwei Padding-Optionen:

- Pad1

```
+--+--+--+--+--+--+--+
|          0          |
+--+--+--+--+--+--+--+
```
- PadN

```
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          1          | Opt Data Len | Option Data
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
- - - - -
```

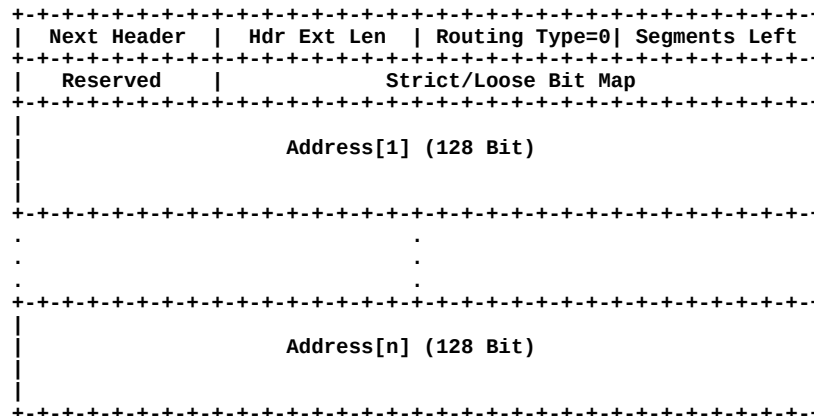
Sie werden benutzt um

- Multi-Oktett-Felder in jedem Extension-Header an ihre “natürlichen Ränder” auszurichten, und um
- die Gesamt-Länge des Headers auf ein Vielfaches von 64-Bit auszurichten.

Routing-Header (Typ 0)

(*Next-Header = 43)

Auflistung von einem oder mehreren Nodes, die auf dem Weg zum Ziel “besucht” werden sollen.



GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Paket-Größen in IPv6

Definitionen:

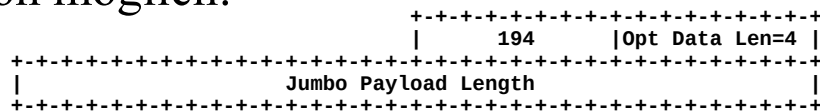
Link-MTU: Maximum Transmission Unit, d.h. die maximale IPv6 Paket-Größe, die in einem Stück über den Link transportiert werden kann.

Path-MTU: die kleinste Link-MTU auf allen Links auf dem Weg vom Absender zum Ziel.

- kleinste Link-MTU für IPv6 im Internet ist 576-Oktetts (gegenüber 68-Oktetts in IPv4).
- Links mit Link-MTU<576-Oktetts müssen Fragmentierung und Zusammensetzung auf Link-Layer-Ebene durchführen.

Maximale Paket-Größe

- Payload-Length-Feld im IPv6-Basis-Header unterstützt Paketlängen bis 65535 Oktetts (ohne 40-Oktett Header).
- Größere Payloads werden mit Payload-Length-Feld=0 zusammen mit der “Jumbo-Payload” Hop-by-Hop-Option möglich:



- Jumbo-Payloads dürfen nicht in Paketen mit Fragment-Header benutzt werden.

Path-MTU-Discovery

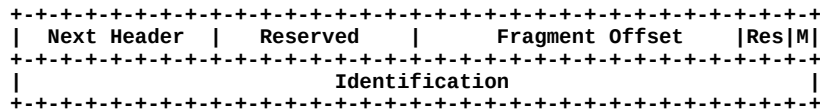
Path-MTU-Discovery ist jeder IPv6-Implementierung dringend empfohlen, um den Vorteil größerer Path-MTUs

1. Absender initiiert Verbindung mit Path-MTU=Link-MTU des ersten verwendeten Links.
2. Falls Link erreicht wird, der eine kleinere MTU hat, wird ICMP-Paket “Packet too big” zusammen mit der dort möglichen MTU zurückgeschickt.
3. Die so iterativ gewonnene Path-MTU wird beim Absender für das jeweilige Ziel gecached. Gelegentlich werden die Cache-Einträge gelöscht,

Fragment-Header

(*Next-Header = 44)

Wird benutzt, um IPv6-Pakete $>$ Path-MTU zum Ziel zu transportieren.

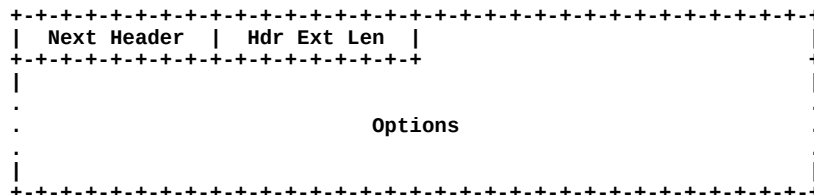


- Fragmentierung soll weitestgehend vermieden werden
→ Path-MTU-Discovery!
- End-zu-End-Funktion: Fragmentierung wird nur vom Absender durchgeführt, und Zusammensetzung wird nur am Ziel durchgeführt. Router führen “unterwegs” keine Fragmentierung durch. Falls Paket zu groß, wird ICMP-Message “Paket too big” zurückgeschickt...

Destination-Options-Header

(*Next-Header = 60)

Transportiert optionale Informationen, die für den (die) Ziel-Node(s) bestimmt sind.



Security

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Security – Grundlegende Begriffe

- **Authentication**
Die Verifikation der Identität des Absenders der Daten.
- **Integrity**
Echtheit der Daten, d.h. Schutz der Daten vor nicht-authorisierter Veränderung.
- **Confidentiality**
Schutz der Daten vor nicht-authorisiertem Zugriff. Üblicherweise wird die Vertraulichkeit der Daten durch Encryption (Verschlüsselung) erreicht.
- **Non-Repudiation**
Das Vermögen des Empfängers der Daten zu überprüfen, ob der angegebene Sender in der Tat die Daten gesendet hat, auch wenn er dies später leugnen sollte.

IPv6-Security – Design-Grundsätze

- IPv6 stellt *solide* Security-Mechanismen zur Verfügung.
- Die Anwendung von IPv6-Security ist rein *optional*, d.h. Anwender ohne Sicherheits-Ansprüche werden nicht negativ betroffen.
- Die angebotenen Mechanismen sind *Algorithmus-unabhängig*, d.h. z.B. die verwendeten Kryptographischen Verfahren können bei Bedarf und ohne Auswirkungen auf andere Teile der Implementation gewechselt werden.
- IP-Security sollte einer möglichst großen Vielfalt von Security-Anforderungen und -Policies genügen.

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

IPv6-Security – Mechanismen

- Authentication Extension Header
Bietet Integrität und Authentizität ohne Vertraulichkeit.
Non-Repudiation nur bei bestimmten Algorithmen.
- Encapsulating Security Payload (ESP) Extension Header
Bietet Vertraulichkeit und Integrität der Daten. Authentizität nur bei bestimmten Algorithmen, Non-Repudiation gar nicht.

Die Verfahren können bei Bedarf kombiniert werden.

- ☞ Jede IPv6-Implementation *muß* Authentication- und ESP-Extension-Header unterstützen.

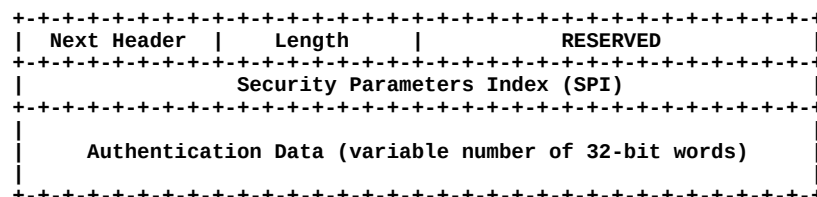
Key-Management

- Key-Management-Protokolle in der Entwicklung
- Key-Management *kein* Bestandteil der IPv6-Security-Implementation
 - klare Trennung von Verteilung der Keys und den eigentlichen Sicherheits-Mechanismen
 - Key-Management-Protokolle leicht wechselbar ohne die IPv6-Security-Implementation zu betreffen
- Jede IPv6-Implementation *muß* manuelle Key-Konfiguration unterstützen

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

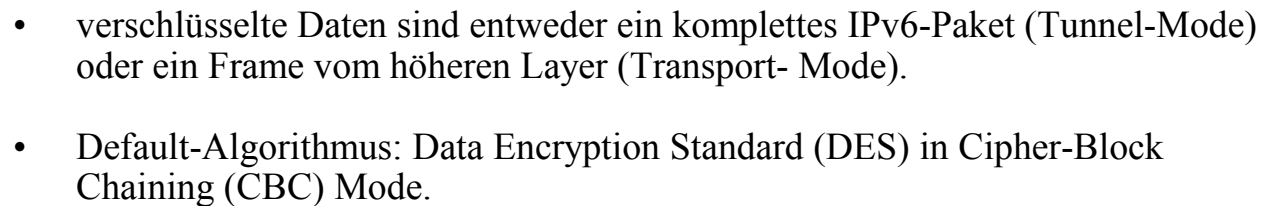
Authentication-Security-Header

(*Next-Header = 51)

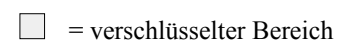


- Destination-Address + SPI identifizieren Security-Verbindung.
- bietet Authentizität und Integrität für alle Felder des IPv6-Pakets, die sich unterwegs *nicht* verändern (wie z.B. Hop-Limit oder Routing-Informationen).
- Default-Algorithmus ist Keyed-MD5.

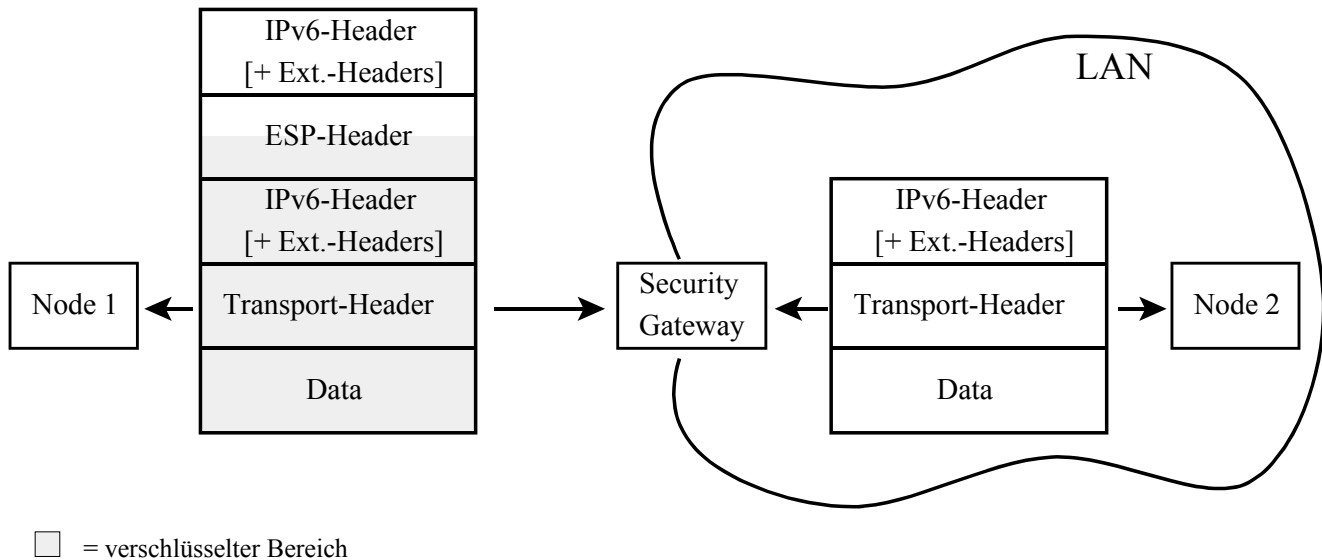
(*Next-Header = 50)



(End-to-End)

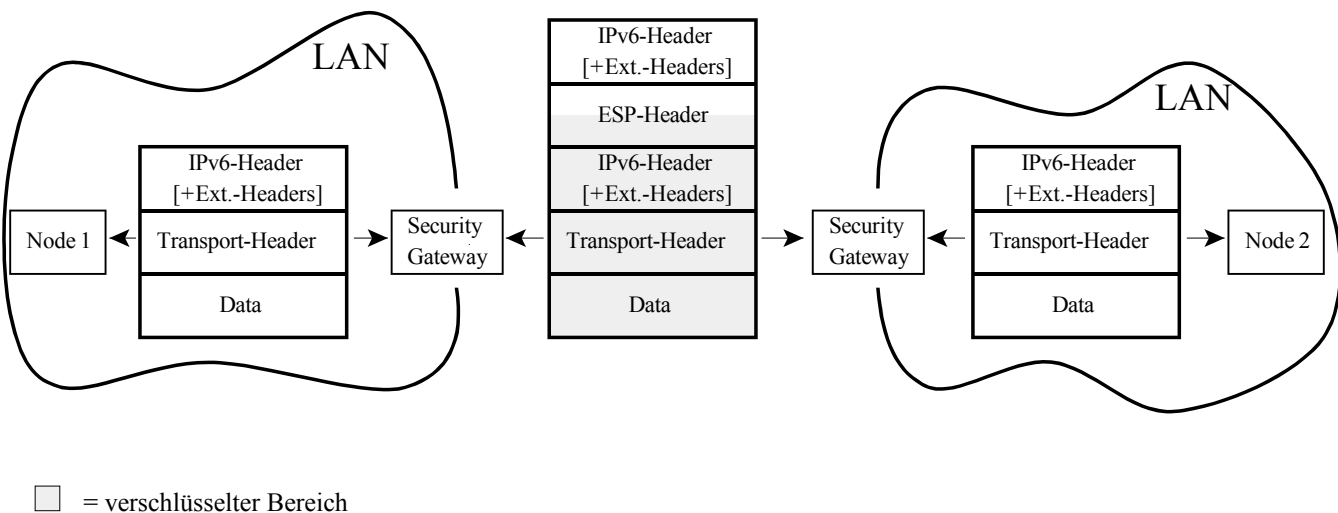


ESP in Tunnel-Mode (End-to-Security-Gateway)



GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

ESP in Tunnel-Mode (Security-Gateway-to-Security-Gateway)



Adressierung

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Repräsentation der 128-Bit IPv6-Adressen

- “normal”: 5F04:FB00:80B0:0000:BF42:00C0:3302:0014
⇔ 5F04:FB00:80B0:0:BF42:C0:3302:14
- komprimiert: FF02:0:0:0:0:0:0:2
⇔ FF02::2
- 0:0:0:0:0:0:0:1
⇔ ::1
- IPv4-kompatibel: 0:0:0:0:0:0:128.176.191.66
⇔ ::128.176.191.66

IPv6-Adreß-Typen

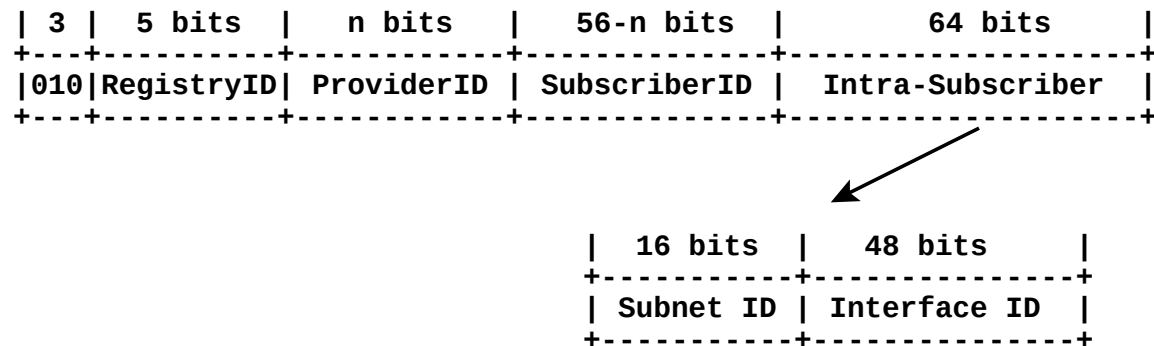
- Unicast (one-to-one)
 - global
 - site-local
 - link-local
 - compatible (IPv4, NSAP, IPX)
- Anycast (one-to-nearest)
- Multicast (one-to-many)
- Reserved

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Adreß-Typ-Präfixe

<u>Adreß-Typ</u>	<u>Präfix</u>
• IPv4-kompatibel	(96 Null-Bits)
• NSAP-kompatibel	0000 001
• IPX-kompatibel	0000 010
• Provider-based Unicast	010
Multit-Regional (IANA)	010 10000
Europe (RIPE-NCC)	010 01000
North America (Internic)	010 11000
Asia-Pacific (APNIC)	010 00100
IPv6-Testing Address Block	010 11111
• Geographic-based Unicast	100
• Link-local Unicast	1111 1110 10
• Site-local Unicast	1111 1110 11
• Multicast	1111 1111
• alle anderen Präfixe derzeit reserviert bzw. noch nicht zugewiesen	
• Anycast-Adressen werden aus Unicast-Adreß-Raum zugewiesen	

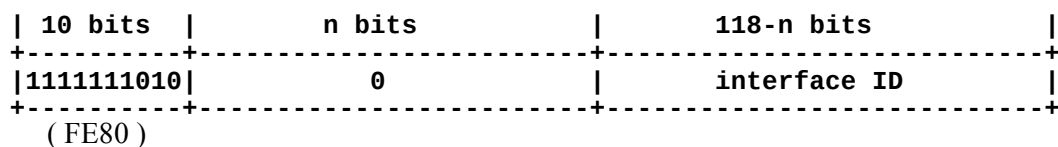
Provider-Based Unicast Adreßformat



- Registry, Hierachiestufen weiter unterteilt werden.
- IPv6-Adreß-Autokonfiguration nutzt linkspezifische Adressen wie z.B. 48-Bit IEEE-802 MAC-Adressen als Interface-ID.

Lokale IPv6-Adressen

- Link-Local



- Adressen nur auf einzelнем Link gültig.
- Verwendung z.B. bei Adreß-Autokonfiguration, Neighbor-Discovery, oder wenn kein Router vorhanden ist.
- Router dürfen IPv6-Pakete mit Link-Local-Adressen nicht weiterleiten.

Lokale IPv6-Adressen (2)

- Site-Local

10 bits	n bits	m bits	118-n-m bits
1111111011	0	subnet ID	interface ID

(FE90)

- Adressen nur in “Site” gültig.
- Verwendung für Sites oder Organisationen, die (noch) nicht am globalen Internet angeschlossen sind.
- Bei späterem Internet-Anschluß kann Site-Local-Präfix durch Subscriber-Präfix vom Provider ersetzt werden.
- Router dürfen IPv6-Pakete mit Site-Local-Adressen nicht aus der “Site” herausleiten.

Anycast-Adressen

- sind Unicast-Adressen und mehr als einem Interface zugeordnet (typischerweise in verschiedenen Nodes).
- werden zum gemäß dem Routing-Protokoll “nächstgelegenen” Interface gerouted.
- Typische Anwendungen:
 - Identifikation eines Sets von Routern eines bestimmten Internet-Providers, z.B. Verwendung im IPv6-Routing-Extension-Header um den Weg über diverse Provider vorzugeben.
 - Adressierung eines Sets von Routern in einem bestimmten Subnetz.
- Restriktionen:
 - dürfen nicht als Source-Adresse verwendet werden
 - dürfen nur Routern zugewiesen werden

Multicast-Adressen



- Flags:

+--+--+--+	T=1	Wohlbekannte (von IANA zugewiesene)
0 0 0 1		Multicast-Adresse
+--+--+--+	T=0	nur temporär gültige Multicast-Adresse
Reserviert		
- Scope:

1	Node-Local
2	Link-Local
5	Site-Local
8	Organisation-Local
E	Global
- Group-ID: Identifiziert Multicast-Group

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Derzeit definierte Multicast-Adressen

- All-Nodes FFxx:0:0:0:0:0:0:1
- All-Routers FFxx:0:0:0:0:0:0:2
- All-RIP-Routers FFxx:0:0:0:0:0:0:9
- Linkname FFxx:0:0:0:0:0:1:1
- All-DHCP-Agents FFxx:0:0:0:0:0:1:2
 - Servers FFxx:0:0:0:0:0:1:3
 - Relays FFxx:0:0:0:0:0:1:4
- Solicited-Nodes FFxx:0:0:0:0:1:x:x

unteren 32-Bit der Unicast-Adresse

Routing

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Routing in IPv6

- Präfix-Routing
 - Ähnlich IPv4+CIDR: der längste passende Präfix wird genommen.
 - Umfangreiche Adreßhierarchien erlauben effizientes Präfix-Routing.
 - Kleinere Routing-Tabellen.
- (derzeit) keine neuen Routing-Protokolle für IPv6 geplant
 - Erweiterung der existierenden (IPv4-) Routing-Protokolle zur Handhabung der längeren Adressen, z.B.
RIP, OSPF, BGP, IDRP, DVMRP, MOSPF, PIM.
- Routing-Extension-Header
 - Kann in Verbindung mit Anycast-Adressen zum Routing durch bestimmte Regionen genutzt werden.

Quality of Service

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Priority

4-Bit-Priority-Feld im IPv6-Header gibt gewünschte Priorität von Paketen, relativ zu den Paketen vom *gleichen* Absender:

- 0-7 Priorität von Verkehr mit “Stau-Kontrolle”, d.h. Verkehrslast wird in Stau-Situationen reduziert (z.B. bei TCP-Verkehr):
 - 0 nicht charakterisierter Verkehr
 - 1 “Füll”-Verkehr (z.B. netnews)
 - 2 unbegleiteter Daten-Verkehr (z.B. email)
 - 3 (reserviert)
 - 4 begleiteter großer Daten-Verkehr (z.B. FTP, NFS)
 - 5 (reserviert)
 - 6 interaktiver Verkehr (z.B. Telnet, X)
 - 7 Internet-Kontroll-Verkehr (z.B. Routing-Protokolle, SNMP)
 - 8-15 Verkehr ohne “Stau-Kontrolle”, z.B. Realtime-Verkehr mit konstanten Bitraten (Audio/Video)
-

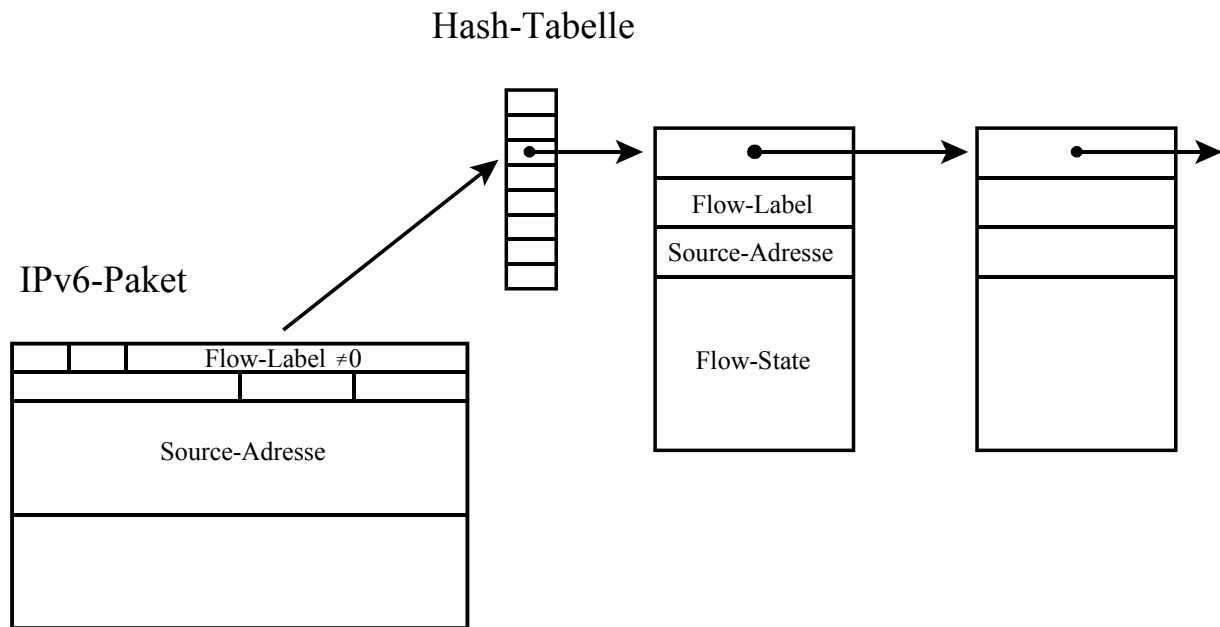
Flows in IPv6

Ein Flow ist eine Folge von Paketen von einem bestimmten Absender zu einem bestimmten (Unicast- oder Multicast-) Ziel, für die der Absender eine spezielle Behandlung bei der Übertragung durch die Router wünscht, wie z.B. Nicht-Default- oder Real-Time-Services für Audio-/Video-Übertragungen.

24 Bit Flow-Label

- Flow-Label-Feld ($\neq 0$) + Source-Adresse identifizieren einen IPv6-Flow eindeutig.
- Flow-Label sind (Pseudo-) Zufallszahlen, deren Bits in den Routern als Hash-Keys für Hashtabellen dienen können (→ effizienter Zugriff auf Flow-States).
- Für alle Pakete eines IPv6-Flows gilt:
Flow-Label $\neq 0$, Source-Adresse, Destination-Adresse, Priority und (falls vorhanden) Hop-by-Hop-Options- oder/und Routing-Header müssen gleich sein.

Flow-State-Lookup



GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Autokonfiguration

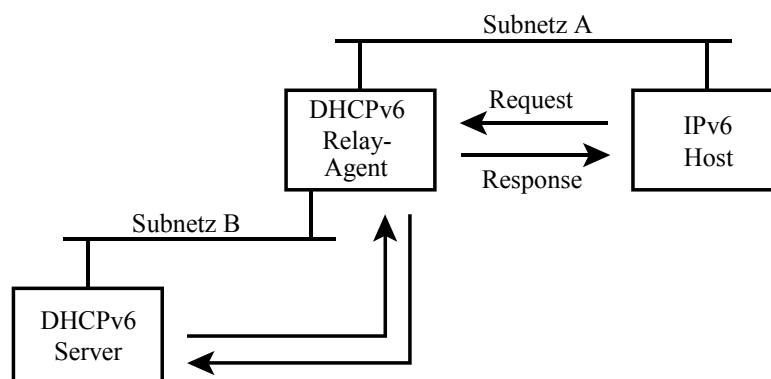
Stateless Adreß-Autokonfiguration

- Für jedes Netzwerk-Interface:
 - Bilden einer Link-Local-Adresse.
 - Sende Router-Solicitation zur All-Router-Multicast-Adresse.
 - Router sendet Router-Advertisement mit Präfix-Information(en).
 - Bilden weiterer IPv6-Adresse(n) aus Router-Präfix(e)+Interface-ID.
 - Jede so gewonnene Interface-Adresse wird auf Eindeutigkeit auf dem Link überprüft (Duplicate Address Detection, DAD).
- Geeignet für kleinere Topologien: keine zusätzlichen Server und wenig oder keine administrative Eingriffe notwendig.
- Hinzufügen neuer Subnetz-Präfixe im Router und Löschen alter bewirkt automatische Umnummerierung im Subnetz.

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Stateful Autokonfiguration

- Client-Server-Modell für mehr administrative Kontrolle.
- Interface-Adressen und weitere mögliche Konfigurations-Informationen über Dynamic-Host-Configuration-Protokoll für IPv6 (DHCPv6).
- Besser geeignet für komplexere Netzwerk-Topologien.
- Kombinierbar mit Stateless-Adress-Autokonfiguration.

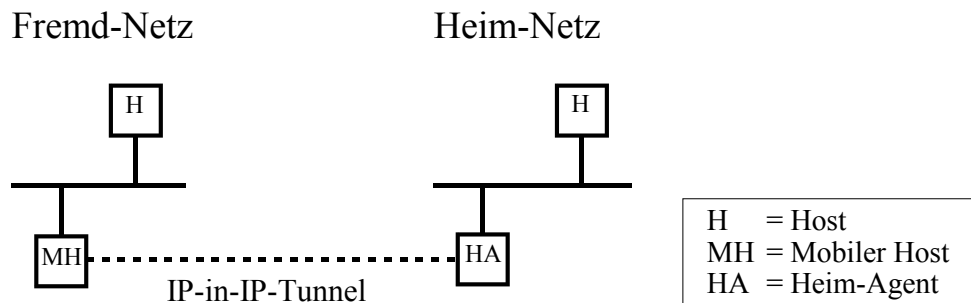


Mobility

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Mobility-Support in IPv6

Mobile Hosts können weltweit im Internet mit ihrer Heimat-IP-Adresse erreicht werden.



Ablauf:

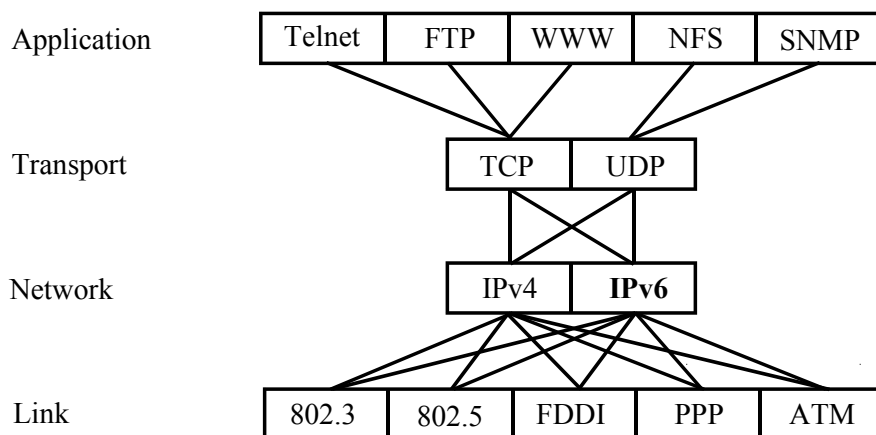
1. MH konfiguriert Care-of-Adresse (Präfix ist der vom Fremd-Subnetz).
2. MH registriert Bindung Care-of-Adresse/Heimat-IP-Adresse beim HA.
3. HA fängt alle Pakete für MH über Proxy-Neighbor-Discovery im Heimatsubnetz ab und tunnelt sie zur Care-of-Adresse des MH.
4. Optimierung: Nodes können Bindung cachen: Falls ein Node Pakete zum MH schicken möchte und die Bindung bereits kennt, wird der Routing-Extension-Header mit der Care-of-Adresse verwendet (kein Tunneln mehr!).

Transition

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Transition IPv4 ➡ IPv6

- Grundkonzept: Dual-IP-Stacks
(IPv4-Hosts und -Router werden nach und nach um IPv6-Stack ergänzt)

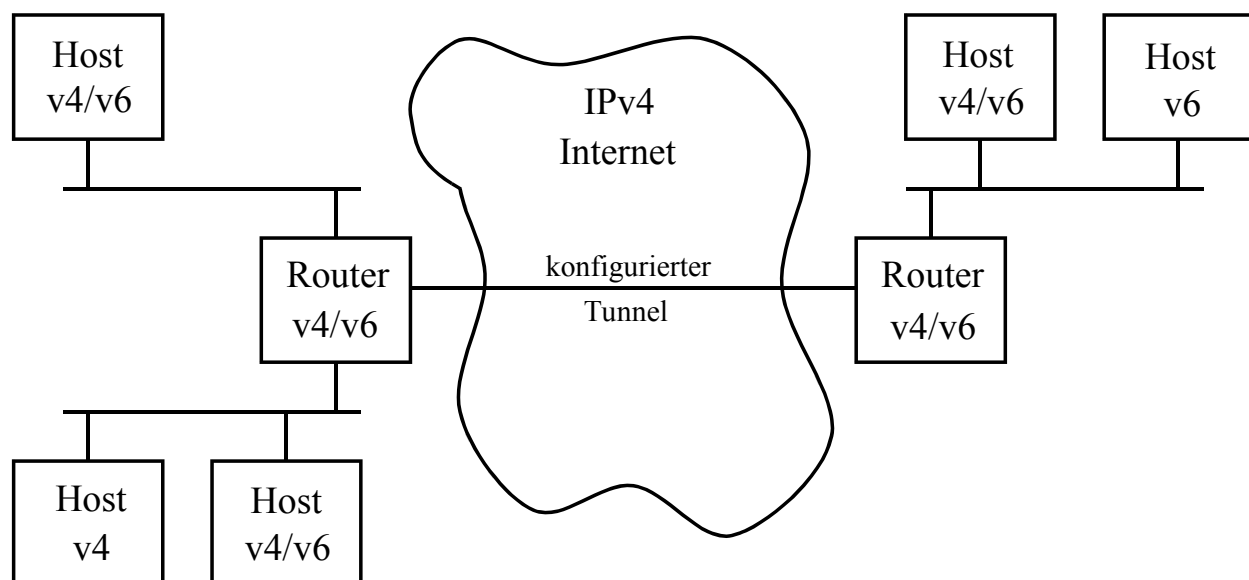


Transition IPv4 ➡ IPv6 (2)

- Überbrücken von (noch) nicht IPv6-fähigen Topologien durch Tunnel-Technik:
 - Tunnel durch IPv6-in-IPv4-Encapsulation.
 - Zwei verschiedene Tunnel-Methoden vorgesehen:
 - Konfigurierte Tunnel:
Die IPv4-Tunnelendpunkte werden manuell konfiguriert; Die Routingtabelle gibt vor, welche IPv6-Pakete über welchen Tunnel gerouted werden.
 - Automatische Tunnel:
Falls IPv4-kompatible-IPv6-Adressen als Ziel-Adresse benutzt werden, wird automatisch ein Tunnel zu der enthaltenen IPv4-Adresse errichtet.

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Transition IPv4 ➡ IPv6 (3)



Transition IPv4 ➡ IPv6 (4)

- Sobald Router um IPv6-Stack ergänzt sind, können IPv6-in-IPv4-Tunnel wieder entfernt werden.
- IPv6 wird sich neben IPv4 im Internet etablieren.
- IPv4 wird noch viele Jahre im Internet geroutet werden.
- IPv4-in-IPv6-Tunnel um später IPv4-Inseln über IPv6-Topologien zu verbinden.

Aktueller Stand

IPv6

Geplante Meilensteine für die Einführung von IPv6 (IETF, Juli 1994, Toronto)

- 07/94 IPng Entscheidung
- 08/94 erste Test-Implementationen
 - 4/95: INRIA/NetBSD
 - 8/95: TCP/IP EXPO, erste Interop.-Tests
- 09/94 Dokumente als Internet-Drafts
- 12/94 Dokumente als Proposed-Standards
 - 12/95 erste wichtige Basis-Dokumente als PS
- 07/95 erste β -Versionen von Released-Software
 - ab Ende 95
- 12/95 Abschluß erster Scaling-Tests
- 12/95 erste Produktions-Software-Releases
 - breite Verfügbarkeit bis Mitte 1998 (Steve Deering, JENC6, Tel Aviv)

☞ Zeitplan hinkt etwa 1-2 Jahre hinterher ...

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Wichtige Dokumente bereits Proposed-Standard

- IPv6-Spezifikation (RFC-1883)
- Adreß-Architektur (RFC-1884)
- Provider-Based Unicast Adress-Format (RFC-2073)
- Neighbor-Discovery (RFC-1970)
- ICMPv6 (RFC-1885)
- Path-MTU-Discovery (RFC-1981)
- DNS-Extensions für IPv6 (RFC-1886)
- Transitions-Mechanismen (RFC-1933)
- RIPng (RFC-2080)
- Stateless Adreß-Autokonfiguration (RFC-1971)
- Security-Architektur für IP (RFC-1825)
- Authentication-Header (RFC-1826)
- Encapsulating-Security-Payload (RFC-1827)
- IPv6 über Ethernet (RFC -1972)
- IPv6 über FDDI (RFC-2019)
- IPv6 über PPP (RFC-2023)

(Stand: Januar 1997)

Host-Implementationen

- 4.4-Lite BSD US Naval Reserach Laboratory (NRL)
- AIX IBM
- BSD/OS WIDE-Projekt (Japan)
- Digital UNIX DEC
- Windows 95/NT FTP Software
- HP-UX SICS (Swedish Inst. of Computer Science)
- Linux Linux-Gemeinde ...
- Linux DRET-Projekt (Frankreich)
- MacOS Apple und Mentat
- NetBSD/FreeBSD INRIA Rocquencourt (Frankreich)
- AIX BULL
- Novell Novell
- Embedded Systems Pacific Softworks
- SCO Unix SCO
- BS2000 Siemens Nixdorf
- Solaris 2.5 Sun
- Streams Mentat
- OpenVMS Process Software

Quelle: <http://playground.sun.com/pub/ipng/ipng-implementations.html>

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Router-Implementationen

- 3Com
- Bay Networks
- Cisco Systems
- Digital Equipment Corporation
- Ipsilon Networks, Inc.
- Penril Datability Networks
- Sumitomo Electric
- Telebit Communications A/S

Quelle: <http://playground.sun.com/pub/ipng/ipng-implementations.html>

6bone

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Der 6bone – Ein weltweites IPv6-Testnetzwerk

Motivation:

- Erprobung und Einsatz von IPv6 und den zugehörigen Routing-Protokollen im globalen Internet mit Hilfe des 6bone-Testbed.
 - Austausch von Erfahrungen im praktischen Einsatz von IPv6-Techniken, z.B.
 - Feedback zu diversen IPv6-Aktivitäten und Working-Groups der IETF.
 - Entwicklung und Erprobung von Mechanismen und Vorgehensweisen sowohl für den Übergang zum späteren IPv6-Internet als auch für die Koordination und den Informationsaustausch beim Betrieb von globalem IPv6-Routing.
-

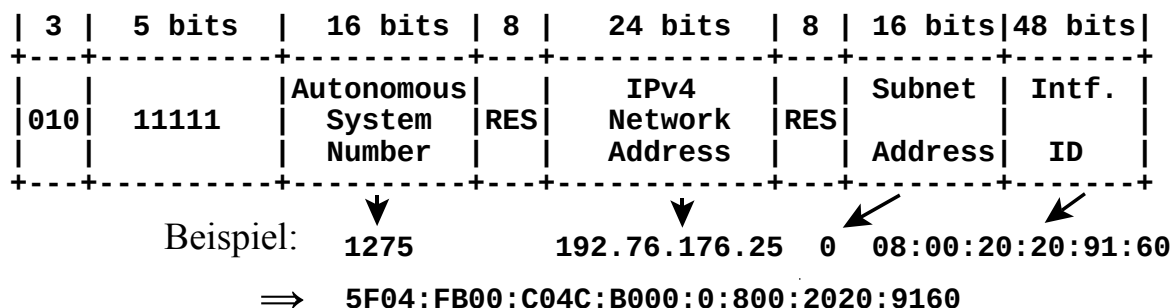
Was ist der 6bone?

- Solange im (Produktions-) Internet noch keine “echte” IPv6-Connectivity besteht, ist der 6bone ein virtuelles IPv6-Netzwerk basierend auf der vorhandenen IPv4-Infrastruktur.
- Inseln von IPv6-fähigen Netzwerken und Systemen sind über virtuelle Punkt-zu-Punkt-Verbindungen (IPv6-in-IPv4-Tunnel) verbunden.
- Tunnelendpunkte sind typischerweise Workstations oder Router mit IPv6-fähigem Betriebssystemen.
- Der 6bone wird als weltweites IPv6-Testnetz wieder verschwinden und in das IPv6-Internet übergehen.
- Der 6bone realisiert somit die Transitions-Strategien der IETF und leitet die “sanfte” Migration von IPv4 nach IPv6 im Internet ein.

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

IPv6-Adressen im 6bone

- Solange Internet-Registaturen (noch) keine IPv6-Adreßräume verteilen, werden im 6bone (derzeit) IPv6-Adressen gemäß RFC-1897 “IPv6 Testing Address Allocation” verwendet.
- RFC-1897 beschreibt ein einfaches Schema, global eindeutige 128-Bit lange IPv6-Adressen zu bilden:



Die RIPE-NCC Datenbank

- 6bone-Teilnehmer registrieren ihre IPv6-Adreß-Präfixe und weitere Informationen, wie z.B. die Adressen der Tunnelendpunkte, in einer RIPE-Datenbank (<ftp://ftp.ripe.net/ipv6/ip6rr/>).
- Beispiel:

```
site:      JOIN project
location:  University of Muenster, Germany
prefix:    5f04:fb00::/32
ping:      6bone.join.uni-muenster.de (5f04:fb00:80b0:0:bf42:c0:3302:14)
tunnel:    128.176.191.66  130.225.231.5  UNI-C      IDRPv6
tunnel:    128.176.191.66  194.182.135.253 Telebit    IDRPv6
tunnel:    128.176.191.66  198.128.4.72   ESnet      ::/0
tunnel:    128.176.191.66  192.32.29.62   BAY        5f02:3000:c020:ae00::/64
tunnel:    128.176.191.66  130.149.62.18  TU Berlin  5f04:fb00:8295:3e00::/64
tunnel:    128.176.191.66  131.188.34.135 FAUERN     5f04:fb00:83bc::/64
tunnel:    128.176.191.66  134.169.34.114 TU-BS      5f04:fb00:86a9:2200::/64
tunnel:    128.176.191.66  141.39.66.24   DETEBERKOM 5f04:fb00:8d27:4200::/64
tunnel:    128.176.191.66  192.76.176.25  DFN        5f04:fb00:c04c:b000::/64
tunnel:    128.176.191.66  129.88.26.1    G6         5f06:b500::/32
contact:    join@uni-muenster.de
status:     operational
remark:     http://www.join.uni-muenster.de
changed:    join@uni-muenster.de 961113
source:     RIPE
```

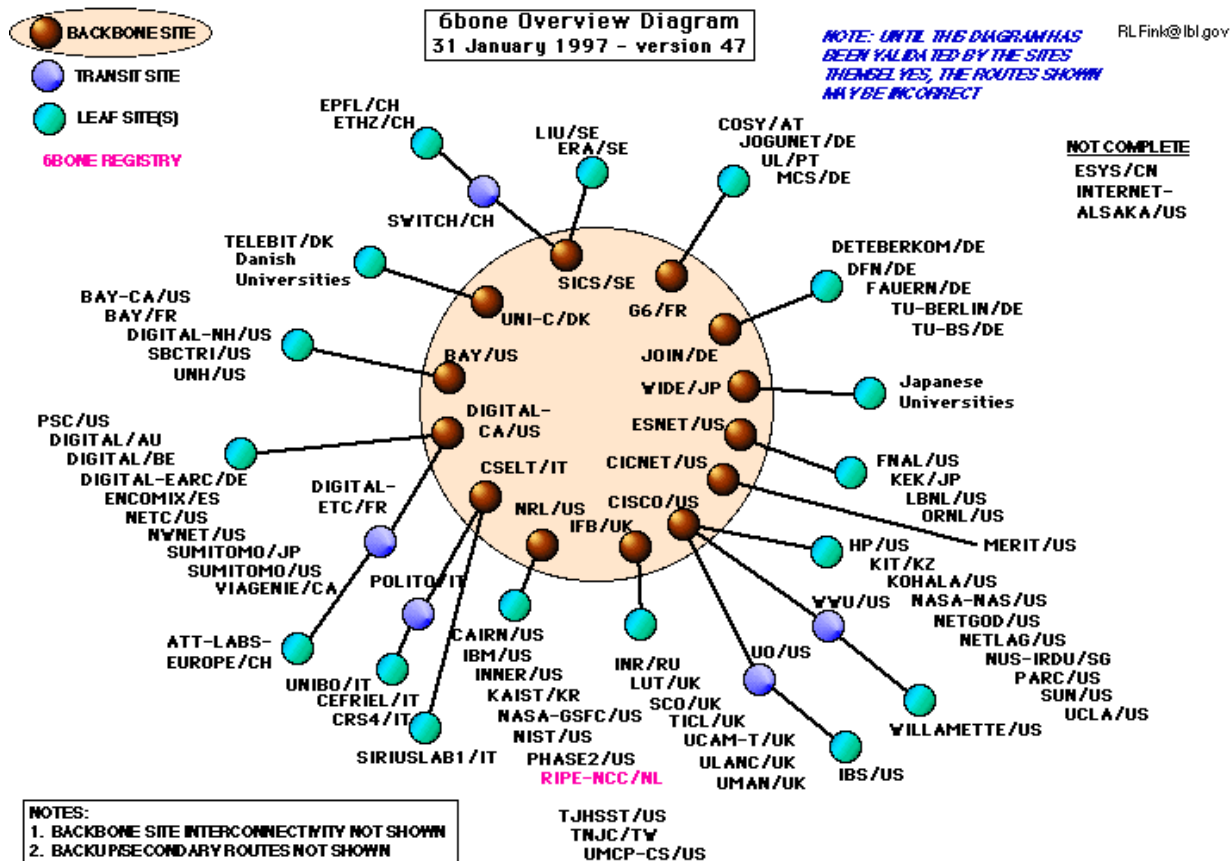
GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Routing im 6bone

- Anfangs nur statisches Routing, d.h. die notwendigen Routing-Informationen werden manuell konfiguriert.
- Auf breiter Basis wird zur Zeit RIPng ausprobiert.
- IDRP oder BGP aber wesentlich prädestinierter für globales Routing.
(IDRPv6 wird z.Z. nur vom dänischen Router-Hersteller Telebit unterstützt)

Der 6bone – Ein rasant wachsendes Gebilde

- Am 15. Juli. 1996 begann der 6bone mit zunächst drei Teilnehmern (G6 in Frankreich, UNI-C in Dänemark und WIDE in Japan).
- Im Januar 1997 sind bereits 92 Teilnehmer registriert.



Implementations-Beispiele

- Solaris 2.5
- Linux

IPv6 für Solaris 2.5/2.5.1, Release 5.0:

IP-Layer:

- automatic/configured tunneling
- IPv6 Unicast/Multicast
- Fragmentation/Reassembly
- Packet-Forwarding
- Path-MTU-Discovery
- ICMPv6
- IPv6 Address-Resolution
- Neighbor-Discovery (ND)
- Stateless Address-Autokonfiguration
- TCP/UDP über IPv6
- RIPng

IPv6-Applikationen:

- telnet / in.telnetd
- ftp / in.ftpd
- tftp (Server/Client)
- finger / in.fingerd
- rlogin / in.rlogind
- rsh / in.rshd
- rcp
- rdist
- rdate
- getent
- mconnect
- sendmail
- NCSA HTTPD-Server
- NCSA Mosaic-Browser
- DNS (bind): named
- NIS+

IPv6-Administration:

- netstat
 - snoop
 - ping
 - route
 - ifconfig
 - nslookup
-

Beispiel einer einfachen Konfiguration

Konfiguration des Auto-Tunnel-Treibers:

```
modload /kernel/drv/atun
/usr/ipv6/sbin/ifconfig ip0.atun plumb6
/usr/ipv6/sbin/ifconfig ip0 ipv6addr ::128.176.191.83 ipv6mask 96 mtu 1480 up
```

Konfiguriere Ethernet-Interface le0:

```
/usr/ipv6/sbin/ifconfig le0\#v6 plumb6
/usr/ipv6/sbin/ifconfig le0\#v6 up
```

Konfiguriere IPv4-kompatible-IPv6-Adresse (Netzmaske = 96+IPv4-Netzmaske):

```
/usr/ipv6/sbin/ifconfig le0\#v6:150 ipv6addr ::128.176.191.83 ipv6mask 125 up
```

Füge Route für Multicast-Adressen hinzu:

```
/usr/ipv6/sbin/route add ipv6 ff00:: fe80::1:0:800:201b:d640 8
```

Starte Router-Discovery-Daemon (ermöglicht Stateless-Adreß-Autokonfiguration):

```
/usr/ipv6/sbin/in.rdisc6 &
```

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

Die Interface-Konfiguration

Von einem IPv6-Router angebotene IPv6-Präfixe: ::128.176.191.80/125
 5f04:fb00:80b0:0:bf53::/80

```
# /usr/ipv6/sbin/ifconfig -a
lo0#v6: flags=849<UP,LOOPBACK,RUNNING,MULTICAST> mtu 8232
        inet6 ::1 netmask 128
lo0: flags=849<UP,LOOPBACK,RUNNING,MULTICAST> mtu 8232
        inet 127.0.0.1 netmask ff000000
le0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST> mtu 1500
        inet 128.176.191.83 netmask ffffffff8 broadcast 128.176.255.255
        ether 8:0:20:1b:d6:40
ip0: flags=c1<UP,RUNNING,NOARP> mtu 4196
        inet6 COMPAT::128.176.191.83 netmask 96
le0#v6: flags=843<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 fe80::1:0:800:201b:d640 netmask 10
le0#v6:150: flags=843<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 COMPAT::128.176.191.83 netmask 125
le0#v6:200: flags=843<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 COMPAT::128.176.191.80 netmask 125 not-addressed
le0#v6:201: flags=843<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 5f04:fb00:80b0:0:bf53:800:201b:d640 netmask 80
```


Die Routing-Tabelle

```
# /usr/ipv6/bin/netstat -f inet6 -r -n
```

Routing Table: IPV6

Destination/Mask	Gateway	Flags	Ref	Use	Interface
-----	-----	-----	-----	-----	-----
::1/128	::1	U	0	0	lo0#v6
COMPAT::128.176.191.80/125	COMPAT::128.176.191.83	U	7	695	le0#v6:150
COMPAT::128.176.191.80/125	fe80::1:0:800:201b:d640	U	7	0	le0#v6:200
::0/96	COMPAT::128.176.191.83	U	2	3	ip0
5f04:fb00:80b0:0:bf53::/80	5f04:fb00:80b0:0:bf53:800:201b:d640	U	7	0	le0#v6:201
fe80::/10	fe80::1:0:800:201b:d640	U	7	5	le0#v6
ff00::/8	fe80::1:0:800:201b:d640	U	7	0	le0#v6
default	fe80::1:0:800:201b:d640	U	7	10889	le0#v6
default	fe80::c0:3300:1b	U	0	20382	le0#v6

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

IPv6 und Linux

- ab Kernel 2.1.8 ist β -IPv6-Code enthalten
- Produktions-Kernel 2.2 wird IPv6-fähig sein

IP-Layer:

- automatic/configured tunneling
- IPv6 Unicast/Multicast
- Fragmentation/Reassembly
- Packet-Forwarding
- Path-MTU-Discovery
- ICMPv6
- IPv6 Address-Resolution
- Neighbor-Discovery (ND)
- Stateless Address-Autokonfiguration
- TCP/UDP über IPv6

IPv6-Applikationen:

- telnet / telnetd
- ftp / ftpd
- tftp / tftpd
- wrapper
- finger / fingerd
- sendmail
- popd

IPv6-Administration:

- netstat
- tcpdump
- ping
- traceroute
- route
- ifconfig

Beispiel einer einfachen Konfiguration

Es wird nur das Autotunnel-Interface (sit0) konfiguriert:

```
/sbin/ifconfig sit0 up
```

```
/sbin/route -A inet6 add ::/96 sit0
```

Die restliche Adreß-Konfiguration erfolgt über Stateless-Adreß-Autokonfiguration ...

Die Interface-Konfiguration

Von einem IPv6-Router angebotener IPv6-Präfix: 5f04:fb00:80b0:0:bf42::/80

```
# /sbin/ifconfig -a
lo          Link encap:Local Loopback
            inet addr:127.0.0.1  Bcast:127.255.255.255  Mask:255.0.0.0
            inet6 addr: ::1/0  Scope:Host
            UP BROADCAST LOOPBACK RUNNING  MTU:3584  Metric:1
            RX packets:18 errors:0 dropped:0 overruns:0
            TX packets:18 errors:0 dropped:0 overruns:0

eth0        Link encap:10Mbps Ethernet  HWaddr 00:00:C0:D0:A4:C3
            inet addr:128.176.0.50  Bcast:128.176.255.255  Mask:255.255.0.0
            inet6 addr: fe80::c0d0:a4c3/10  Scope:Link
            inet6 addr: 5f04:fb00:80b0:0:bf42:0:c0d0:a4c3/80  Scope:Global
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:13563 errors:0 dropped:0 overruns:0
            TX packets:297 errors:0 dropped:0 overruns:0
            Interrupt:10 Base address:0x310 Memory:d0000-d4000

sit0        Link encap:IPv6-in-IPv4
            unspec addr:[NONE SET]  Bcast:[NONE SET]  Mask:[NONE SET]
            inet6 addr: ::128.176.0.50/0  Scope:Compat
            inet6 addr: ::127.0.0.1/0  Scope:Unknown
            UP RUNNING NOARP  MTU:1480  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0
            TX packets:0 errors:0 dropped:0 overruns:0
```

Die Routing-Tabelle

```
# /bin/netstat -A inet6 -r -n
```

```
Kernel IPv6 routing table
```

Destination	Next Hop	Flags	Metric	Ref	Use	Iface
::1/128	::0	UH	1	2	0	lo
::0/96	::0	U	1	1	0	sit0
5f04:fb00:80b0:0:bf42::0/80	::0	U	1	1	0	eth0
fe80::c0d0:a4c3/10	::0	U	1	1	0	eth0
fe80::c0:3302:14/0	fe80::c0:3302:14	GD	1	1	0	eth0

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

DFN-Projekt: *IP Version 6 im WiN*

Grundsätzliches Ziel ist es, das WiN auf die kommende weltweite Einführung von IPv6 ins Internet vorzubereiten und seinen Teilnehmern dabei Hilfestellung zu leisten.

Hierzu: Aufbau eines Referenzzentrums

- Koordination und Kooperation auf internationaler Ebene.
- Frühzeitiges Einbringen WiN-spezifischer Interessen.
- Vorbereitung der Migration.

Programm des Referenzzentrums

- Aufbau eines Testzentrums.
- Allgemeine Untersuchungen.
- Empfehlungen für den Betrieb von IPv6-Netzwerken.
- Netzbetreiber- und Anwenderunterstützung.
- Informationsvermittlung.

GUUG-Frühjahrs-Fachgespräch '97 – Tutorial: Einführung in IP Version 6 – Guido Wessendorf

JOIN-Projekt

- Gemeinsame E-Mail-Adresse:
join@uni-muenster.de
- WWW-Seiten:
<http://www.join.uni-muenster.de>

(hier finden Sie viele weitere Informationen zu IPv6 und Umfeld
und Links zu weiteren Informations-Quellen ...)

Guido Wessendorf
Universitätsrechenzentrum
Einsteinstraße 60
D-48161 Münster
Tel.: 0251/83-31639, Fax: -31653