ZF KT2 - IPv6

Chronologie

- 90er Jahre: Internet Engin. Task Force (IETF) beginnt Nachfolger für IPv4 zu entw. Motivation: knapper Adressraum

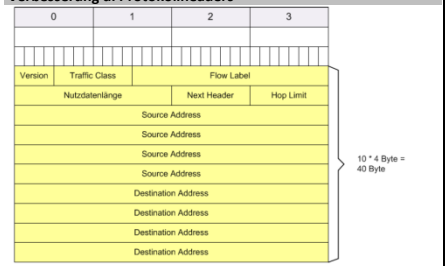
Angestrebte Verbesserungen

- Erw. Adr.raum / Verbess. d. Protokollheaders / Flow Label / Verbess. Routing (Routing Header) / Verb. d. Sicherheits-mechanismen / Ermittl. MTU durch Endgerät → keine Fragm. durch Router

Erweiterung des Adressraums

- 16Byte (128Bit) → 3.4*10exp(38) Adr.
- 340 Billionen Quadrillionen Adr.

Verbesserung d. Protokollheaders



- Version (0110)/TrafficClass (00000000)

Flow Label: Komm.-Paar (S&E) identifiz. Nutzdatenlänge: max. 65535 Byte (mit Jumbo Payload Option → grössere Datagramme)

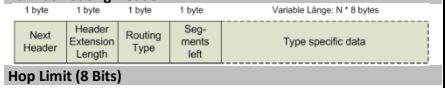
Flow Label

- S weist FL zu / zufäll gener.hex.Z. 00000..FFFFF
- Default (00000)
- Vorteile: Datenflüsse auf einfache Weise klassifizieren → spez. Abarb. d. bez. Datenpak. für Übermittlung. / Ziel: effiz. Datenfluss / Ohne Flows muss Router R-Tabellen konsult. für Bestimmung des Output-Port
- Flow = Seq. v. Pak., gesendet von Quelle zu Ziel: Unicast / Anycast / Multicast IPv6 Adr.

NextHeader

Protokoll	Wert	Protokoll	Wert	Protokoll	Wert
ICMP	1	UDP	17	EIGRP	88
IGMP	2	RSVP	46	OSPF	89
TCP	6	ICMPv6	58	.....	.....
Protokoll	**Wert**	**Protokoll**	**Wert**	**Protokoll**	**Wert**
IP in IP	4	Routing Header	43	Authentica-tion Header	51
EGP	8	Fragmenta-tion Header	44	Mobile IPv6 (Draft)	135
IGP (IGRP)	9	Encrypted Security Payload Header	50	.....	.....

Format Routing Header



Hop Limit (8 Bits)

- Entspricht TTL bei IPv4

Source Address (128 Bits)

- komplette IPv6 Quelladresse

Destination Address (128 Bits)

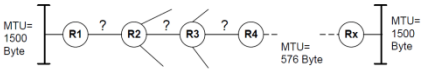
- komplette IPv6 Zieladresse

Verbesserung d. Sicherheitsmechanismen

- IPv4: keine Sicherheitsmechanismen
- Neuerungen: grösserer Adr.raum / Konzept "NextHeader" / Tunneling-Mechanismen / Autoconfiguration
→ erlauben neue Arten von Angriffen

Ermittlung MTU

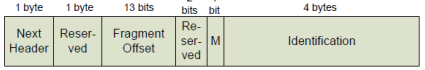
- Keine Frag. durch Router aber: Ermittl. mittels Path MTU Discovery (Pfad bis zum E bzgl. kleinsten vorkommenden MTU untersuchen)



- Kleinste MTU bei IPv6: 1280 Byte
- Nur S darf Fragm. vornehmen
- Ist zu versend. Paket grösser als kleinste vorkommende MTU -> S fragm. Paket
- Reassembly (zusammensetzen d. Fragm.) bei E

(evtl. Folien S. 27 einfügen)

Format des Fragment Headers



- NextHeader (1. Typ d. fragmentierbar. Teils d. urspr. IPv6 Pakets (NextHeader = TCP(6)))
- Reserved: ungenutzt (null)
- Fragment Offset: Versatz in Mengen v. 8 Bytes
- M-Flag (more fragments): 1, falls mehr Fragm.
- Identification (ID): Alle Fragm. eines Pakets haben dieselbe ID

Adressierung bei IPv6

Darstellung / Notation

2001:0620:0000:0004:0A00:20FF:FE9C:7E4A
= 2001:620:0:4:A00:20FF:FE9C:7E4A
1023:0000:0000:0000:1736:a673:88a0:a620
1023::1736:a673:88a0:a620
IPv4-Adressen, die unter IPv6 weiterhin benötigt sind, können in Dezimalcodierung notiert werden. Beispiel: 138.12.16.10

Binär: 10001010.00001100.00010000.00001010

Hexadezimale Notation, bei der jeweils zwei Bytes zu einer Hex-Ziffer zusammengefasst sind: 8a.0c.10.0a

Vollständige IPv6 Darstellung: 0:0:0:0:0:0:8a0c:100a =
0000:0000:0000:0000:0000:0000:8a0c:100a = ::8a0c:100a

Notation in Dezimaldarstellung für den IPv4-Teil:
0:0:0:0:0:0:138.12.16.10 = 0:0:0:0:0:0:8a0c:100a = ::8a0c:100a

Notation für Hosts mit hybrider IPv6/IPv4-Einbindung

0000:0000:0000:0000:0000:FFFF:.....
Es verbleiben 4 Bytes. Hier steht die unter IPv4 verwendete Adresse, entweder die Unicast-, Multicast- oder Anycast-Adresse.

Wiederum das Beispiel mit der IPv4-Adresse 138.12.16.10:
Schreibweise der IP-Adresse mit hybrider IPv6/IPv4-Einbindung:
0:0:0:0:0:ffff:138.12.16.10 = ::ffff:138.12.16.10 = ::ffff:8a0c:100a

Adressstruktur von IPv6

Subnetz-Präfix (64 Bits) + Hostteil (64 Bits)
- IPv6 Global Unicast Adress: 64-n Bits: Globaler Routing Präfix / N Bits: Subnetz-ID / 64 Bits: Schnittstellen-ID
- Real World-Struktur der IPv6 Unicast-Adress: 48 Bits: Globaler Routing Präfix/ 16 Bits: Subnetz-ID / 64 Bits: Schnittstellen-ID

IPv6 Link-lokale Adressen

Präfix: fe80::/10

Sinn und Zweck: Ausschliesslich für Subnetz-interne Kommunikation, bei welcher die zu verbindenden Einheiten eine Layer 2 Verbindung haben

Ein Router darf ein Paket einer link-lokalen Adresse nicht zu einem anderen Subnetz weiterleiten.

IPv6 Unique local Address

Präfix: FC00::/7 oder FD00::/7

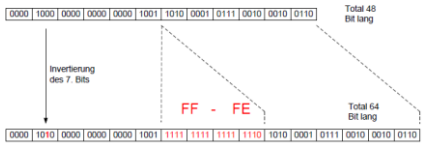
Sinn und Zweck: Nur für den internen Gebrauch innerhalb einer Unternehmung gedacht

Autokonfiguration

Stateless Adress Autoconfiguration

- Motivation: vereinfachte, teilw. autom. Zuteilung einer. IPv6-Adr
- Weshalb: Zukunft viele Devices mit IPv6 Adr.
- Wie: leicht modifiz. MAC Adr. wird beigezogen, um letzte 64Bits der 128 Bit langen IPv6 Adr. zu erzeugen
  - Modifiziert: Bit Nr. 7 ist im Fokus:
  "0" → link-lokale Adresse
  "1" → globale Adresse

08 - 00 - 09 - A1 - 72 - 26



0A - 00 - 09 - FF - FE - A1 - 72 - 26

- Weltweit eindeutig, falls MAC-Adr im gleichen Subnetz nicht nochmals vorkommt

Nachteil

- Verfolgbarkeit d. Rechners / d. Aktivitäten weil immer dieselbe Adr. erhalten wird
- Beheben durch: Pro Schnittst. mehrere ID verw. / Fixe EUI-64 nur für ankomm. Verb. nutzen / für abgehende Verb. IDs durch Zufall erzeugen

Dynamic Host Configuration Protocol v6 (DHCPv6)

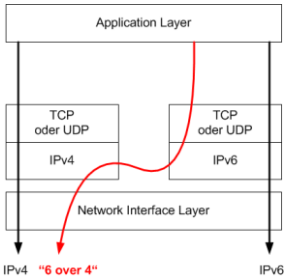
Neu mit „**Stateless** Address Autoconfiguration" → Host kann seine IPv6 auf Basis d. MAC-Adr. beziehen. (brauch deshalb kein DHCP)
Bei „**Stateful** Address Autoconfiguration" **DHCP nötig**.
Stateless und Stateful: Adressbezig erfolgt stateless und zusätzlich KonfigInformationen via DHCP
Pro Schnittstelle mehrere IP-Adressen.

Domain Name System (DNS) für IPv6

Record AAAA („Quad-A"): speichert die 128 bit lange IPv6-Adr. und enthält die Abbildung auf den Hostnamen
Bsp: larbor.zhaw.ch IN AAAA 2001:620:190:fff1::1 =
2001:620:190:fff1:0000:0000:0000:0001
Zugehöriger **Reverse Lookup Domain Name**:
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.f.f.f.0.9.1.0.0.2.6.0.1.0.0.2.IP6.ARPA
DNS Support für IPv6 ab Version 9 von BIND vorhanden.
DNS Resolver: stellt Anfrage nach AAAA-Request, 2 Anfragen v4 (A-Record) und v6 (AAAA-Record). Als Response kann beides oder eines zurückkommen.

Migration von IPv4 zu IPv6

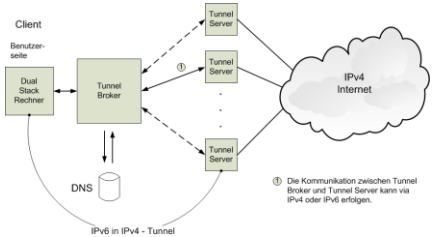
Dual Stack



Beide Protokolle im Betrieb | typisch ist Router | wichtig Zugang zu IPv6 DNS Server | Endgeräte meist nur IPv4 oder v6

Tunnel Broker

IPv6 Pakete werden in Pakete eines andern Protokolls gepackt | häufig IPv6 als Nutzlast von IPv4 Paketen | administrative Aufwand für Konfig sehr gross | deswegen Tunner Broker als Server, die Benutzeranfragen für Tunnels automatisch bearbeiten und Tunnels aufsetzen.



Tunnel Broker: Via IPv4 erreichbar, Benutzer wählt sich ein um Tunnel zu registrieren. Broker setzt und schliesst Tunnel.
Tunnel Server: Dienen der Skalierbarkeit: Gesamtlast der Tunnelendpunkte kann aufgeteilt werden. Typischerweise Router im Dual-Stack-Betrieb.
Client: Ein Dual-Stack Rechner, der ans IPv4-Netz geht. Teilt Broker seine Identität evt. mit Passwort mit. Borker prüft und erteilt Berechtigung
Ablauf: Server wird ausgewählt | Wahl der IPv6 Adr. für Client | Festlegung der Laufzeit für Tunnel | im DNS registrieren (muss globale IPv6-Adr. sein) | Konfig des Servers

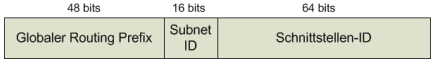
Tunnelverfahren 6in4

Pack IPv6 Paket direkt in IPv4. Im IPv4-Header: Protocol Type ist 41 für IPv6 | „proto-41 static" als anderer Name | meist manuell konfiguriert oder automatisierte Konfig. mit TIC Server

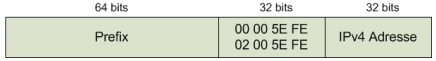
Tunnelverfahren ISATAP

ISATAP: Intra-Site Automatic Tunnel Addressing Protocol
Zählt zu den automatischen Tunnelverfahren | für Endgeräte via IPv6 aber Routernetz mit IPv4 | ISATAP Adresse: IPv4 Adresse ist in der Schnittellen-ID der IPv6 eingefügt.

IPv6 Unicast-Adresse:



ISATAP-Adressformat



Prefix: Kann globaler Prefix, Link-Lokaler Prefix oder 6to4-Prefix sein.
00 05 5E: private Adresse | **02** 00 5E: offizielle Adresse | **FE**: IPv6 Adresse mit eingebetteter IPv4 Adresse

Translation mit 6to4

Um IPv6 Standorte über IPv4 Routernetz zu verbinden ohne Tunnel | IPv4 dient als Punkt-zu-Punkt-Link und Router werden als 6to4-Gateways bezeichnet | offizielle IPv4 Unicastadr. benötigt | Translation aufwändiger als Tunnerling | gut bei Migration über längeren Zeitraum
Präfix von IANA: **2002::/16 = Präfix 2002 ist 16 Bits lang**
→ 2002:<IPv4-Adr.>/48, Präfix inkl. Interface ID und SLA ID (Site Level Aggregator), SLA ist 16 Bit lang → 655536 Subnetze

Translation mit Teredo (NAT-T)

Falls IPv6 Geräte über öffentl. IPv4 Netz kommunizieren wollen und NAT (Network Address Translation) verwenden sollen.
Teredo: Tunnelrealisation, ermöglicht den IPv6-Hosts die sich hinter IPv4-NAT befinden, mit anderen Geräten zu kommunizieren. **IPv6 Pakete werden in UDP getunnelt.**
- Format der Teredo-Adresse:

Teredo-Präfix: 2001:00000:/32
Server: Teredo Server
Flags: Addresstyp von NAT-Typ
Portnummer: mapped UDP-Port des Teredo-Dienstes auf Client, Mapped = jedes Bit invertiert
IPv4 Client: mapped IPv4 Adresse des Clients
IPv4 Teredo Client: Recher mit privaten IPv4 Adresse

Translation mit StatelessIP-ICMP Translation (SIIT)

Jeder „Translator" muss IP-Header und ICMP-Header übersetzen können. Deshalb Adresstyp **„IPv4-translatable address"**

0000:0000:0000:0000:FFFF:0000:a.b.c.d
IPv4-Adresse

Andere Schreibweise:
0::FFFF:0:a.b.c.d

Der Präfix lautet:
0::FFFF:0:0:0/96
32 Bits

Translator hat Adresspool und IPv6-Rechner holt dort eine IPv4 Adresse | Translator = meist Router | kennt IPv4 Adressen vom Pool | entfernt den IPv4 Header und setzt IPv6 Header.
Übersetzung der Header-Felder: IPv4 → IPv6

IPv6 Headerfeld	Aktion / Information
Version	6
Traffic Class	Die 8 Bits des TOS-Felds (IPv4) werden übernommen
Flow Label (20 Bits)	0, ausgeschrieben: 0000 0000 0000 0000 0000
Nutzdatenlänge	Übernahme des Inhalts von 'Total Length' (abzüglich Grösse IPv4 Header)
Next Header	Übernahme des Inhalts des 'Protocol' Felds
Hop Limit	TTL-Wert von IPv6 -1
Source Address	Beispiel: 0::ffff:0:192.168.30.10
Destination Address	Beispiel: 0::ffff:0:192.168.100.20
IPv4 Options	IPv4 Options sind ignoriert

Übersetzung von IPv6 in IPv4

IPv4 Headerfeld	Aktion / Information
Version	4
IHL (Internet Header Length)	5
TOS (Type of Service)	Alle 8 Bits des Traffic Class Feldes von IPv6 werden übernommen und ins TOS-Feld des IPv4-Headers kopiert.
Total Length	Nutzdatenlänge aus dem IPv6-Header, zusätzlich vergrössert um die Länge des IPv4-Headers.
Identification	Auf 0 setzen
Flags	More Fragments Flag: Auf 0 setzen. Don't Fragment Flag: Auf 1 setzen. Bit 0 ist reserviert bei IPv4; muss 0 sein.
Fragment Offset	Auf 0 setzen
TTL	Hop Limit aus dem IPv6-Header kopieren. Der Translator selbst ist ein Router, daher muss der Wert noch um 1 dekrementiert werden.
Protocol	Der Inhalt des 'Next Header' Felds von IPv6 wird ins 'Protocol' Feld kopiert.
IP Header Checksum	Vom IPv6-Header liegen keine Daten vor. → die Checksum des IPv4-Headers muss *nach* der Erstellung des IPv4-Datengramms erstellt werden.
Source Address	Die niederwertigen 32 Bits der 'IPv4 translated Address' werden als IPv4 source address eingefügt.
Destination Address	Die niederwertigen 32 Bits der 'IPv4 mapped destination address' werden als IPv4 source address eingefügt.
Optionen	Folgende IPv6 Parameter werden **nicht** übersetzt: - IPv6 Hop-by-Hop Options Header - Destination Options Header - Routing Header mit einem 'Segments Left'-Feld von 0

Internet Control Message Potocol v6 (ICMPv6) RFC 4443

- Aufgaben:-Router auf Pfad S → E muss S Mitteilung zrü. senden
- E muss S Mitteilung zurücksenden
- Falls NextHeader-Feld = 58 →Payload hat ICMPv6 Paket

- Type: ident. Typ d. Nachricht (bestim. Format nachfolg. Meldu.)
- Code: detaillierte Info zu einem Type
- Checksum: mit Hilfe "Pseudoheader". (Quelle / Ziel / Nutzdaten-länge / NextHeader. Prüfsumme bezieht Header inkl. Pseudohead.
- Message Body: eigentliche Meldung (Fehler- / Infomeld.)

Regeln für die Verarbeitung v. ICMPv6-Paketen

1. ICMP-Fehlermeld. mit unbek. Typ → an Layer 4-7 leiten
2. Infomeld. mit unbek. Typ → Meldung verwerfen
3. Fehlermeld.: möglichst viele Daten d. problembehaft. Paket aber: MTU nicht mehr als 1280 Bytes
4. Falls Meldung an Layer 4-7: Protokolltyp aus Originalpaket herauslesen (Datenfeld). Falls Protokolltyp nicht eruierbar → verw

Wann darf keine ICMPv6-Fehlermeldung erfolgen?

- Fehlermeldung darf keine Fehlermeldung erzeugen / Redirect Meldung (Typ 137) darf keine Fehlerm. erzeugen / Als Folge e. Pakets, das an IPv6 Multicast-Adr. gesendet darf keine erzeugen / Als Folge e. Pakets, das Layer2 Multicast enthält → keine erzeug. / Als Folge e. Pakets, dessen S nicht eindeutig einer einz. Schnittst zugeordnet werden kann → keine Fehlermeldung erstellen

ICMPv6 Typ	Fehlermeldung	ICMPv6 Typ	Informationsmeldung
1	Destination unreachable	128	Echo Request (für ping verwendet)
2	Packet too big	129	Echo Reply (für ping verwendet)
3	Time exceeded	144	Home Agent Address Discovery Request
4	Parameter problem	145	Home Agent Address Discovery Reply
		146	Mobile Prefix Solicitation (engl. Bittstellung)
		147	Mobile Prefix Advertisement

Typ	F-meld	Bedeutung
1	0	No route to destination
1	1	Comm. with dest. administr. prohibited
1	2	Beyond scope of source address
1	3	Adress unreachable
1	4	Port unreachable
1	5	Source adress failed in-/egress policy
1	6	Reject Route to destination
3	0	Hop Limit exceeded in transit
3	1	Fragment reassambly time exceeded
4	0	Erroneous header field
4	1	Unrecognized next Header type encountered
4	2	Unrecognized IPv6 option encountered

Neighbour Discovery

- kombiniert ARP, ICMP Router Discovery und Redirect
- wird genutzt: Zur Detekt. d. erreichb. Nach-barn / Autokonfig. v. IPv6 Adr. / Ermittl. v. Netzwerk-Präfixes /

ICMPv6 Typ	Informationsmeldung133
133	Router Solicitation
134	Router Advertisement
135	Neighbour Solicitation
136	Neighbour Advertisement
137	Redirect Message

Ermittlung v. Routen / Detektion v. IPv6-Adressduplikaten / Auf-lösung v. MAC-Adr v. Rechnern am gleichen Ethernet LAN / Bemerken v. ändernden MAC-Adressen