aconet

www.aco.net

ISPA Academy
IPv6 Workshop

13. Jänner 2011 Moderator: Wilfried Wöber

...unter Verwendung von Material zusammen getragen von:

Kurt Bauer	kurt.bauer@univie.ac.at
Wolfgang Hennerbichler	<derzeit OÖ Landesregierung>
Ulrich Kiermayr	ulrich.kiermayr@univie.ac.at
Harald Michl	harald.michl@univie.ac.at
Wilfried Wöber	wilfried.woeber@univie.ac.at

Ein Service der universität wien

Agenda

…in Teil I:

- Motivation für IPv6 (warum ein neue IP Version)
- Protokollübersicht
- Änderungen aus Benutzersicht
- Transition / CoExistence / Routing
- DNS (separate Präsentation)

…in Teil II:

- Security inkl. Situation Firewalls et al.
- Erfahrungsbericht IPv6 im Campusnetz der Uni Wien

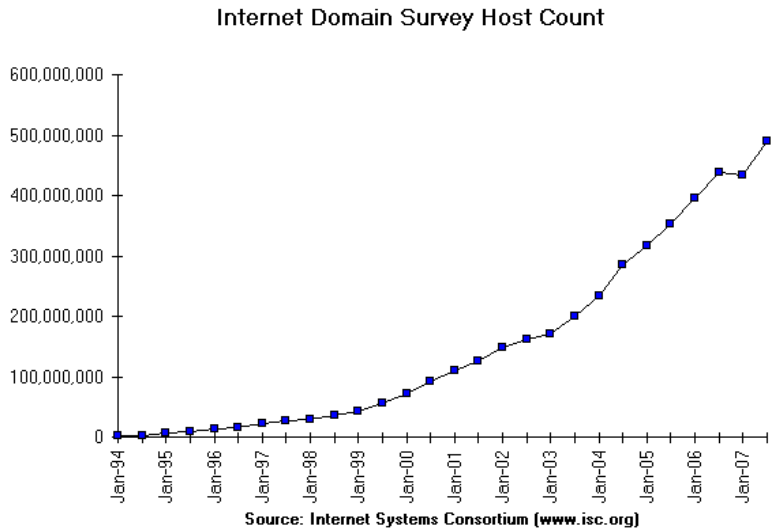
Agenda

- Motivation für IPv6 (warum ein neue IP Version)
- Protokollübersicht
- Änderungen aus Benutzersicht
- Transition / CoExistence / Routing

Motivation für ein neues Internet Protokoll

Fakten:

- IPv4 (RFC 791) wurde 1981 standardisiert
- theoretisch ca. 4,3 Milliarden Adressen
- praktisch viel weniger (127.0.0.0/8, RFC1918, Multicast ⇨ ca. 14%)
- ca. 68% der verfügbaren Adressen sind an RIRs ‘assigned’
- noch ca. 18% im ‘unallocated pool’ der IANA
- RFC 3194 (host-density ratio) ⇨ max. 240 Mio adressierbare Hosts



Motivation für ein neues Internet Protokoll

- diverse Technologien ermöglichen ‘funktionierendes Internet’, trotz zuweniger Adressen (hptsl. NAT)

ABER

- NAT bricht mit dem ‘end-to-end’ Prinzip
- NAT erhöht Komplexität und Fehleranfälligkeit
- NAT wird als ‘Security’ Mechanismus missverstanden,
- aber im Gegenteil - erschwert z.B. Incident Handling

- aktuelle Vorhersagen sehen 2011 als das Jahr, in dem die IANA den letzten IPv4 Prefix vergeben wird (“game over”) - was dann ??
  - RIRs haben noch Adressen, LIRs haben noch Adressen
  - mglw. Zurückforderung bzw. Dokumentation freier Adressräume
  - mglw. ‘wilder, freier’ Adresshandel (wieviel wird eine IPv4 Adresse kosten ?)

- diverse “policy proposals” (regional+global), Reservierungen, weird stuff?

Motivation für ein neues Internet Protokoll

Trotz diverser anderer Verbesserungen im IPv4-Stack ist daher die einzige 'wirkliche' Motivation für IPv6:

3.402.823.669.293.846.346.337.467.431.768.211.456

Adressen (ie. 2^{128} bzw. $3{,}4 \times 10^{38}$)

- IPv6 (RFC 1883) wurde 1995 standardisiert, dzt. aktuell RFC2460
- Status IPv6 (Stand 10/07):
  - 1433 LIR Blöcke zugewiesen (682 von RIPE)
  - 689 davon in der BGP-Routingtable sichtbar
  - diverse PI Zuordnungen und nicht vollständig aggregierte Prefixe ergeben dzt. (Stand 10/10) eine Routingtable von ca. 2.5K Prefixes für IPv6 und ca. 337K Prefixes bei IPv4

Randnotiz1: IPv1-IPv3 gibt's (eigentlich) nicht, IPv5 bezeichnet das Internet Stream Protocol (RFC1819)
Randnotiz2: diverse IPv6-Statistiken von Gert Döring, SpaceNet, München

Motivation für ein neues Internet Protokoll

Weiters… (Theorie und Wirklichkeit ☺)

- The Internet of Things
  - windscreen wipers, light bulbs, temperature sensors,
  - breathing monitors, intelligent cushions in hospitals
- Zero-Configuration Networks
  - yeah – autoconfiguration – but...
- Mobility
- Security
- Killer Applications

Motivation für ein neues Internet Protokoll

Links:

- ISC Internet Domain Survey - http://www.isc.org/ops/ds/
- IPv6 observations - http://www.space.net/~gert/RIPE/R59-v6-table/
- IPv4 address report - http://www.potaroo.net/tools/ipv4/index.html

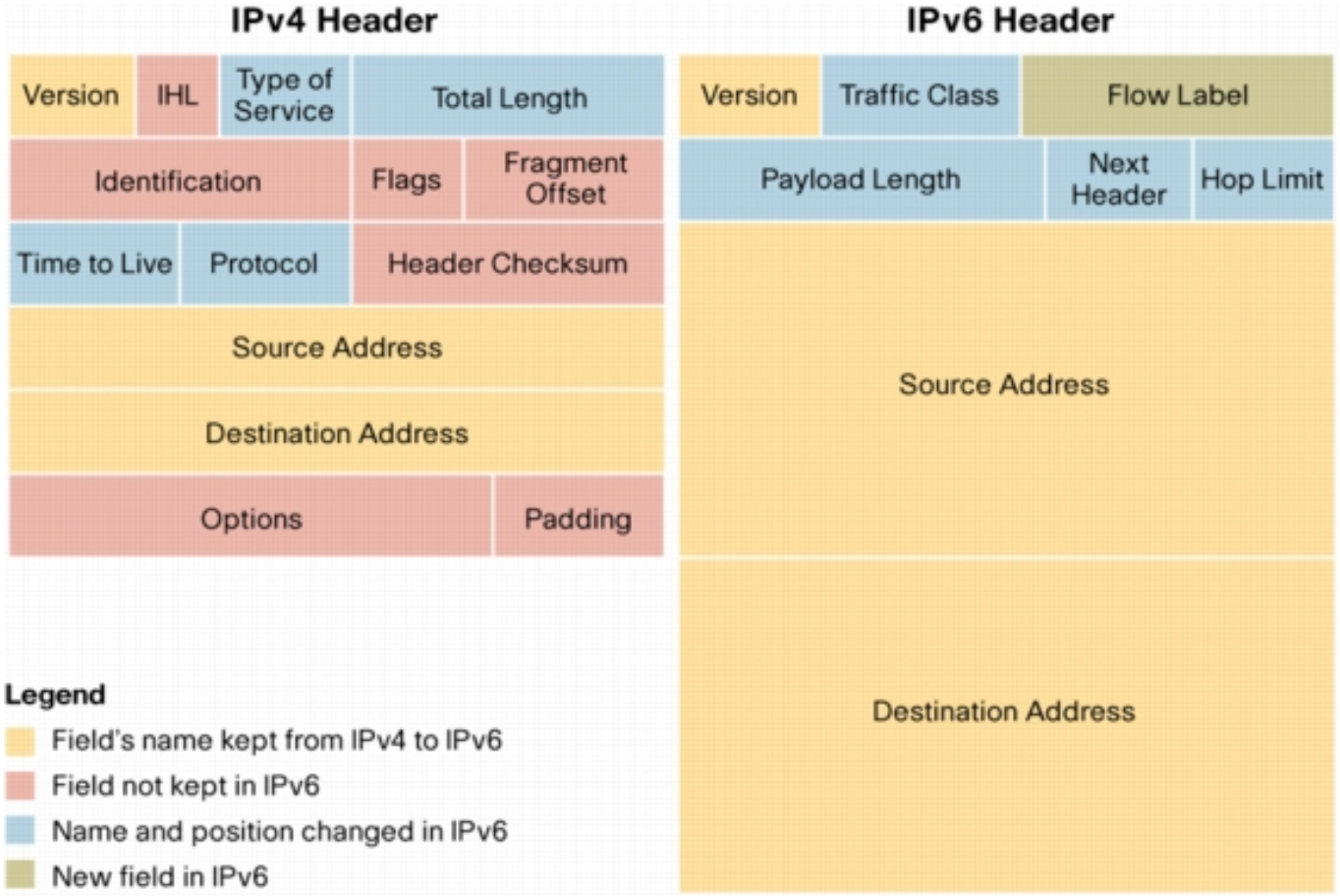
?? Fragen ??

Agenda

- Motivation für IPv6 (warum ein neue IP Version)
- Protokollübersicht
- Änderungen aus Benutzersicht
- Transition / CoExistence / Routing

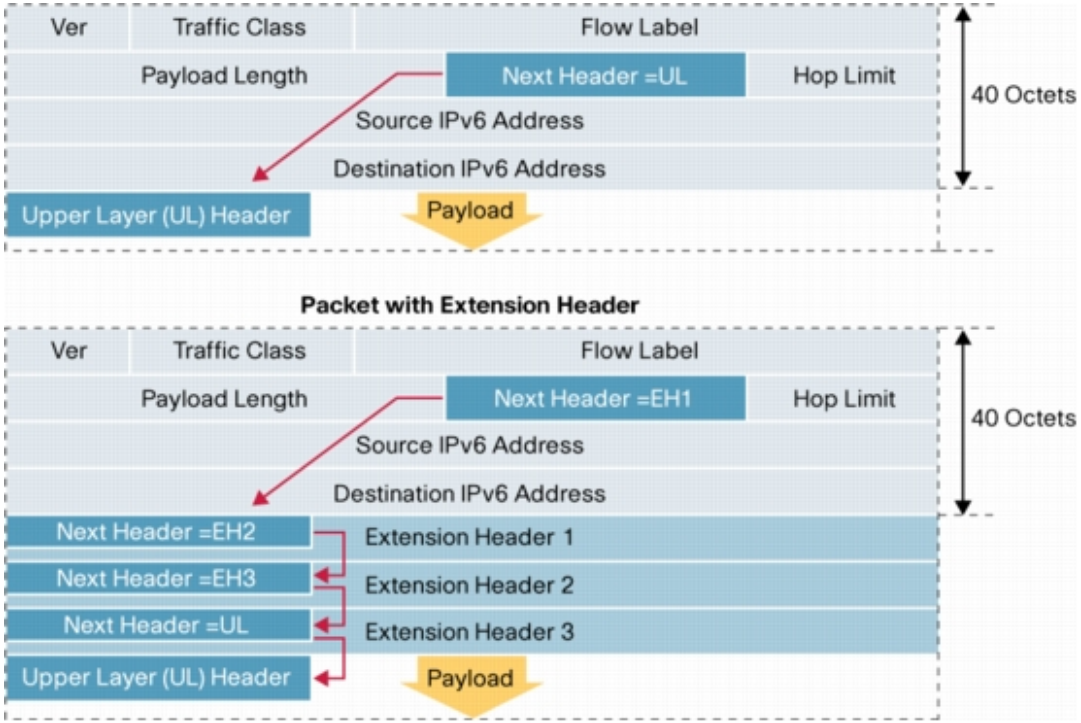
Protokollübersicht - Headerformat

Vereinfachter Header mit fixer Länge von 40byte:



Protokollübersicht - Headerformat

Optionen werden in Extension-Headern transportiert:



<u>mögliche Extension Header:</u>

1. Hop-by-Hop Options header (muss, wenn vorhanden, der erste Ext.-header sein)
2. Destination Options header
3. Routing header
4. Fragment header
5. Authentication header
6. Encapsulating Security Payload header

Protokollübersicht - Adressformat

128 bit Adresse, repräsentiert durch 8 x 16bit, angeschrieben in Hex, getrennt durch Doppelpunkte:

2001:0628:0000:0000:0000:0000:0000:0001

führende Nullen können weggelassen werden:

2001:628:0:0:0:0:0:1

Gruppen von Nullen können zusammengefasst und durch 2 Doppelpunkte ersetzt werden (aber nur 1x!):

2001:628::1

Die Länge des Prefix wird (wie gewohnt) in CIDR Schreibweise angegeben:

2001:628::1/64

Protokollübersicht - Adressarten

Unicast (One-to-One)

- global (2000::/3)
- link-local (FE80::/10)
- (site-local)
- Unique Local (FC00::/7)
- (IPv4-compatible)
- IPv6-mapped (::FFFF:a.b.c.d/128)
- spezielle Adressen
  - unspecified (::/128)
  - loopback (::1/128)

Multicast (One-to-Many)

FF00::/8

Anycast (One-to-Nearest)

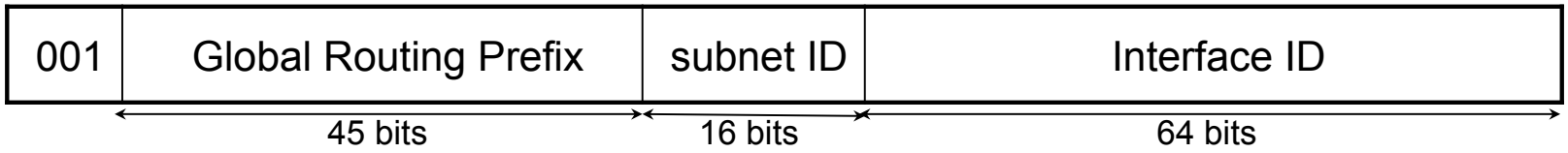
aus dem unicast Adressraum

Reserved

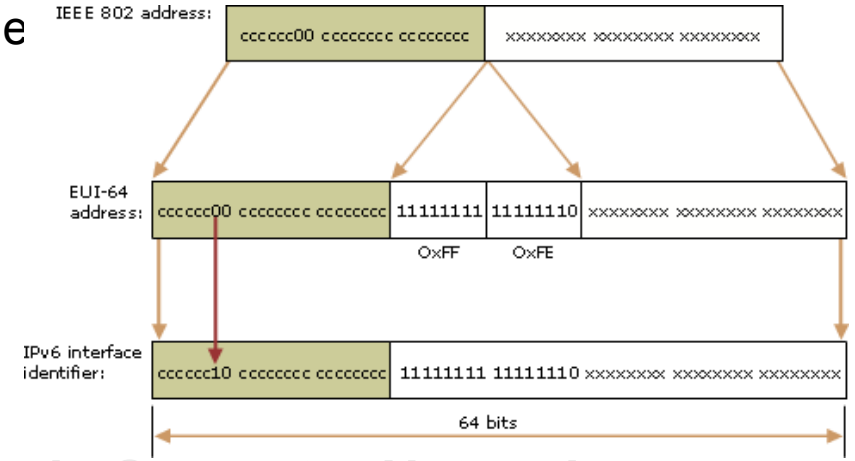
ca. 7/8 des Adressraums

Protokollübersicht - Adressarten

Global Unicast Adresses - RFC 3587



- Der ‘global routing prefix’ wird einer Zone (site) zugewiesen.
  - ermöglicht eine hierarchische Struktur aus globaler Sicht
- Die ‘subnet ID’ identifiziert die Subnetze innerhalb einer Zone (site).
  - ermöglicht eine hierarchische Struktur innerhalb der Site
- Die ‘Interface ID’ wird (meist) aus der MAC-Adresse mittels EUI-64 Format gebilde



Protokollübersicht - Adressarten

Global Unicast Adresses - RFC 3587

- per default erhalten LIRs /32 Prefixe
  - falls nötig auch grösser (z.B. ACOnet dzt. 2001:628::/30)

- LIR vergibt per default /48 Prefixe
  - falls nötig auch grösser (z.b. UniVie /31, TU Wien /32),
  - aber Zustimmung von RIPE NCC nötig

- Prefixe zw. /48 und /128 werden an Endbenutzer vergeben
  - nach RFC 3177 und aktuellen policies:
    - per default /48
    - /64 wenn ein und nur ein Netz gebraucht wird
      - auch für P-t-P links
    - /128 wenn ein und nur ein Endgerät angeschlossen wird

- neuere Policies erlauben den ISPs mehr Flexibilität

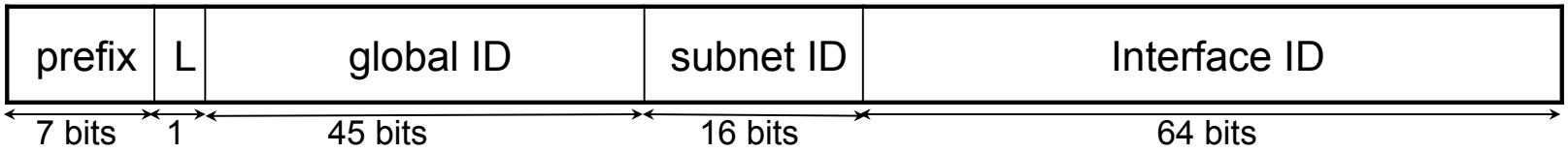
Protokollübersicht - Adressarten

Link-Local Adresse

- fe80::/10
- Jede Schnittstelle zum Netz weist sich selbst eine link-local Adresse zu. Im Normalfall besteht diese aus dem Prefix und dem EUI-64 Identifier.
- Mit diesen Adressen ist die Kommunikation auf einem Layer-2 Link gewährleistet, ohne irgendeine manuelle Konfiguration vorauszusetzen.
- Wichtig für ‘neighbor discovery’ und ‘autoconfiguration’

Protokollübersicht - Adressarten

Unique Local IPv6 Unicast Addresses - RFC4193



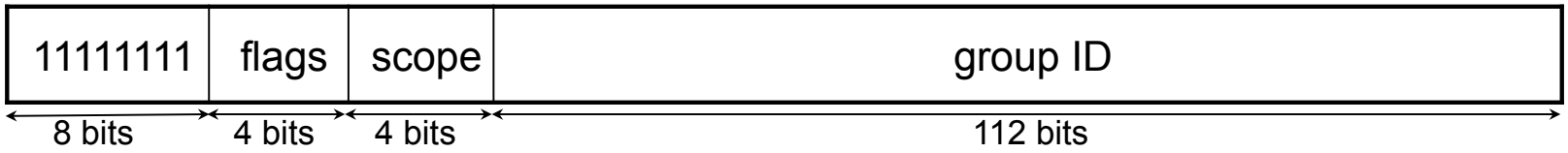
- Global eindeutiger Prefix - FC00::/7
- L=1 bedeutet 'locally assigned', L=0 dzt. nicht verwendet
- global ID wird nach einem bestimmten Algorithmus berechnet (siehe RFC)

Verwendung:

- für lokale Kommunikation, normalerweise innerhalb einer Zone (site)
- sollten global nicht geroutet werden, wohl aber innerhalb einer Zone (site)
- könnten auch zwischen einer bestimmten Anzahl von Zonen (sites) routbar gemacht werden.
- durch den gewählten Algorithmus prinzipiell global eindeutig
- auch mehrere pro Zone (site) möglich

Protokollübersicht - Adressarten

Multicast Adressen



- Prefix ff00::/8
- flags: niedrigstes Bit bestimmt ob Gruppe permanent oder transient, Rest reserviert
- scope:

1 - node local
2 - link-local
5 - site-local
8 - organization-local
B - community-local
E - global
Rest reserviert

- group ID: identifiziert die Multicast Gruppe

Protokollübersicht - Adressarten

Lebensdauer von Adressen

- tentative - während der Adress-Zuordnung und Duplikats-Prüfung
- valid...
- preferred - wird für den Aufbau neuer Verbindungen verwendet
- deprecated - noch gültig, aber sollte nicht mehr verwendet werden
- invalid...

- Adressen haben einen „best before“-Kleber ☺

- es ist bei IPv6 für ein Interface völlig “normal”, mehrere Adressen gleichzeitig konfiguriert zu haben!!

- Verteilung von Prefixes kann von den Routern übernommen werden!!

Protokollübersicht - Adressarten

?? Fragen ??

Agenda

- Motivation für IPv6 (warum ein neue IP Version)
- Protokollübersicht
- Änderungen aus Benutzersicht
- Transition / CoExistence / Routing

Welche Adressen hat ein Rechner ohne Konfiguration ?

IPv4	IPv6
	loopback address ::1/128
	link-local address (auf jedem IF)
	all-nodes multicast addr. (ff00::1)

Änderungen aus Benutzersicht

Weitere essentielle Unterschiede zu IPv4:

- Ein Interface hat per default mehrere Adressen,
  kann aber auch mehrere ‘global unicast’ Adressen haben

- Ein Host muss weder manuell noch per DHCP konfiguriert werden,
  um eine global gültige IPv6 Adresse zu erhalten
  stateless address autoconfiguration - RFC4862

- kein ARP (address resolution protocol)
  Neighbor Discovery - RFC4861

- keine Fragmentierung am Weg, nur mehr in den Endgeräten
  Path MTU Discovery - RFC1981

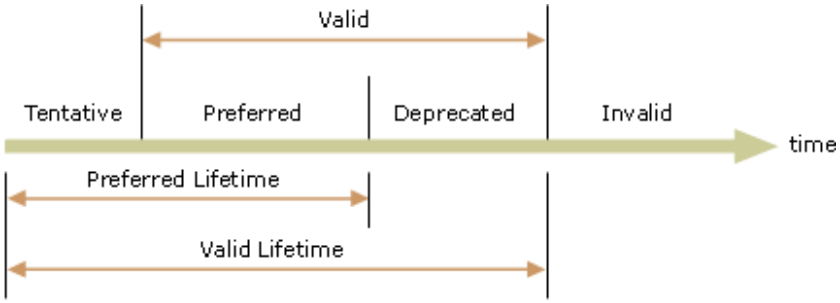
Änderungen aus Benutzersicht

address auto-configuration - Konfigurationstypen

- **stateless**
  Die Adresskonfiguration wird ausschliesslich durch den/die Router am jeweiligen Link und den Host selbst durchgeführt. Kein Status am Router über zugewiesene Prefixe
- **stateful**
  DHCPv6 - prinzipiell wie IPv4, Status über zugewiesene Adresse am DHCP Server
- **stateless & stateful part**
  Beide oben genannten Typen können auch gemeinsam auftreten, in nahezu jeder beliebigen Kombination
- **stateless + privacy-enhanced**
  Der “neueste Schrei” (ab XP Service Packs) und die default config: Das Interface wechselt in regelmäßigen Abständen die IF-ID um zu verhindern, dass Systeme über die MAC/EUI-64 getraced werden können. → Desaster für revDNS! →mühsam zu entfernen ☹

Änderungen aus Benutzersicht

address auto-configuration - address lifetime



IPv6 Adressen haben eine spezifiziert Lebenszeit, die durch 4 Zustände charakterisiert ist

Adresszustand	wann ?	wofür ?
tentative	während die Adresse auf Eindeutigkeit überprüft wird	duplicate address detection
preferred	Eindeutigkeit überprüft und gewährleistet	Verkehr kann empfangen und gesendet werden. Neue Kommunikation kann aufgebaut werden
deprecated	keine neuer Kommunikationsaufbau	Verwendung für bestehende Kommunikation erlaubt
invalid	keinerlei Kommunikation	Pakete werden verworfen

Änderungen aus Benutzersicht

stateless address auto-configuration - Ablauf

Host gibt sich selbst ‘link-local’ Adresse

- ★ link-local prefix (fe80) + interface identifier (meist EUI-64)

Überprüfung ob Adresse am link eindeutig (duplicate address detection)

- ★ ICMPv6 Paket an die eigene LL-Adresse wird verschickt
  - ➡ keine Antwort - Adresse verwenden
  - ➡ Antwort - automat. Prozess unterbrechen, manuelle Konf.

‘joinen’ der notwendigen Multicast-Gruppen

- ★ FF02::1 - all nodes
- ★ FF02::2 - all routers

absenden einer ‘router solicitation’ Message

- ★ destination address: ff02::2 (all routers)

Änderungen aus Benutzersicht

stateless address auto-configuration - Ablauf cont.

Antwort von Router

★ keine Antwort - kein Router am Netzwerk, keine globale Connectivity

★ 'router advertisment' erhalten - 2 Möglichkeiten:

➡ Paket enthält alle notwendigen Informationen (Prefix, Timer, Gateway)
'global unicast' Adresse bestehend aus erhaltenem Prefix und Interface Identifier (meist EUI-64) erstellen.

➡ Paket hat DHCP-Bit gesetzt
falls Prefix von Router erhalten, diesen verwenden, restliche Information von DHCP-Server erfragen, je nachdem Adresse aus Prefix + Interface Id. erstellen oder gesamte Adresse von DHCP verwenden

duplicate address detection

★ wie bei LL-Adresse, Paket mit eigener Adresse als Zieladresse verschicken

➡ keine Antwort - Adresse verwenden

➡ Antwort - automat. Prozess unterbr

Änderungen aus Benutzersicht

Neighbor discovery - RFC4861

Funktionalität der IPv6 Neighbor Discovery:

- Feststellen der 'link-layer' Adresse anderer Geräte am selben Link (analog ARP in IPv4, *neighbor solicitation / advertisment*)
- Finden benachbarter Router (*router solicitation / advertisment*)
- Überprüft laufend die Erreichbarkeit benachbarter Geräte und löscht unerreichbare aus dem (Neighbor-) Cache bzw. berichtigt deren Einträge
- Aktive Suche nach Alternativen, falls der bevorzugte Router nicht mehr erreichbar ist

ersetzt ARP, ICMP redirect und ICMP router discovery

Änderungen aus Benutzersicht

ICMPv6 - RFC4443, updated by RFC4884

Sowohl router-solicitation/advertisment Messages, als auch neighbor-solicitation/advertisment Messages sind ICMPv6 Pakete.

Darüberhinaus wird ICMPv6 unter anderem für folgende Aufgaben gebraucht:

- *Ping* - Echo request/reply
- *Traceroute* - Time exceeded
- *IPv6 Multicast* - Group Membership
- *Path MTU Discovery* - Packet too big
- *Destination unreachable*
- *Redirect*
- *Router renumbering*

Änderungen aus Benutzersicht

Path MTU Discovery - RFC1981

- Default MTU size: statt 576 bytes für IPv4 Pakete, 1280 bytes für IPv6
  verbessertes Verhältnis von Payload zu Header
- keine Fragmentierung ‘am Weg’ des Pakets von der Quelle zum Ziel
  daher schnelleres ‘packet-forward’ im Router
- Path MTU Discovery
  1. Quellhost versucht Paket mit MTU des lokalen Links zu schicken
  2. Falls ein Router ein Paket dieser Grösse nicht weiterschicken kann, verwirft er das Paket und sendet die ICMPv6 Meldung ‘packet too big’, in der auch die maximale MTU zu finden ist, zurück.
  3. zurück zu 1., bis Paket bei Zielhost angelangt

Vorteil: schnelleres packet-forward im Router

Nachteil: Quelle/Ziel/MTU-size ist statisch, bei Änderung des Pfads mglw. neue PMTU Discovery nötig

Empfehlung: 1500 bytes Minimum auf allen Links

Änderungen aus Benutzersicht

Multicast

- MLD (Multicast Listener Discovery) statt IGMP zur Router-Host Kommunikation
  - MLDv1 enspricht IGMPv2
  - MLDv2 enspricht IGMPv3
  - MLD benutzt ICMPv6
- PIM adaptiert
  - unterstützt PIM-SM (aber kein MSDP), PIM-SSM und Bidir-PIM
- BGP adaptiert
  - Multiprotocol BGP notwendig
- kein MSDP ⇨ embedded RP (RFC3956)
  - Unicast Adresse der RP ist in der Multicast-Adresse eingebettet

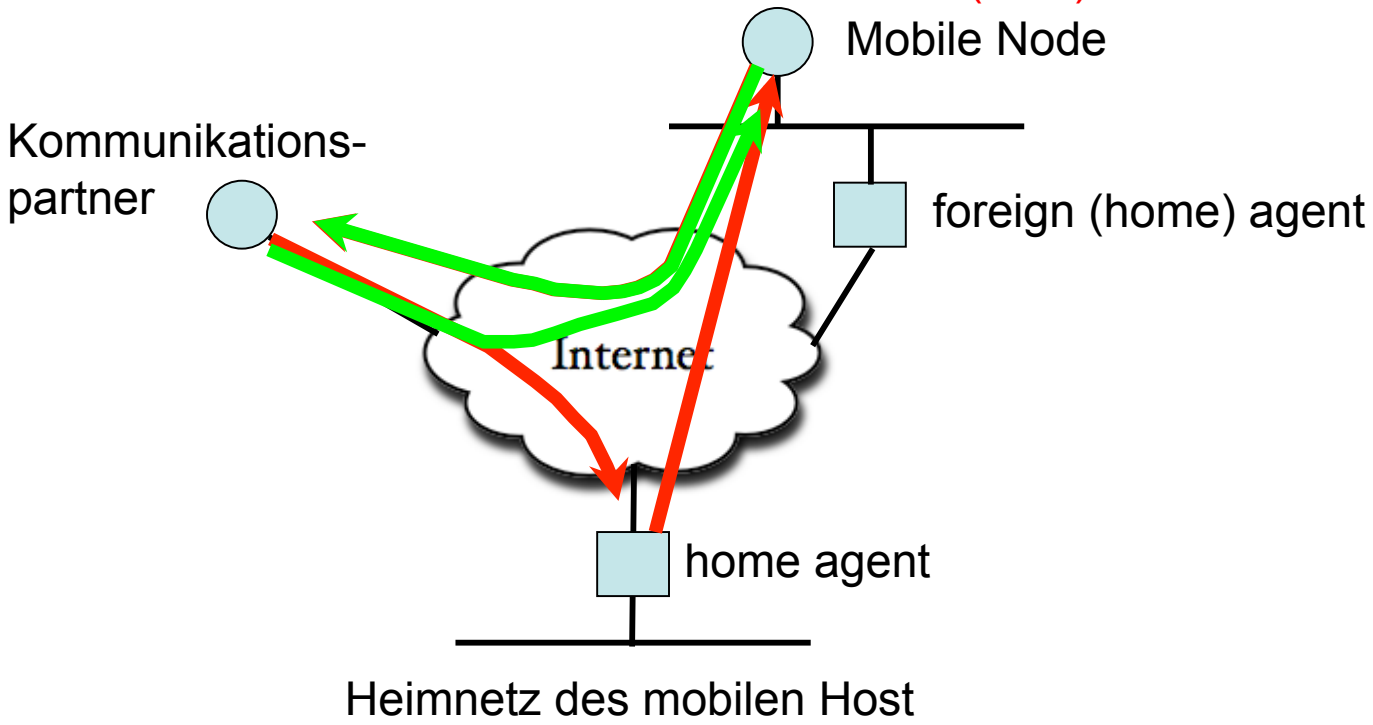
FF7x:y40:2001:628:1:1::<gID>/96

x - scope
y - RIID (RP Interface ID)
gID - 32bit für Multicast Gruppen
RP Adresse - 2001:628:1:1::y/64

Änderungen aus Benutzersicht

Mobility - RFC3775

- Mobile Node (MN) ist durch seine Home Address (HoA) global erreichbar
- Home Agent (HA) übernimmt die Erreichbarkeit der MN im Heimnetz
- MN ist im Gastnetz unter seiner Care of Address (CoA) erreichbar



Änderungen aus Benutzersicht

Multihoming

- Mehrere Ansätze, oft ähnlich oder gleich wie IPv4, ‘work in progress’:
  - Provider Independent Adressen (routing table ‘explodiert’): oft nur /48 von der RIR, aber Backbone Filter oft </32
    - Vorteil ⇨ mehrere Adressen pro Host
      - Erreichbarkeit: mehrere Adressen im DNS notwendig
      - Verbindungsaufbau: welche Adresse wird verwendet ?
- weitere nicht sehr vielversprechende Ansätze
- Überlegungen für next-generation Routing in der IETF
  - Locator - Identifier Split
    - ein global eindeutiger Identifier, aber Locator je nach aktuellem Standort (abhängig von Netzwerk, Provider, usw.)
    - dzt. noch keine Implementierung, einige gute Ansätze
    - IRTF - Host Identity Protocol Research Group

Änderungen aus Benutzersicht

IPsec

- mandatory im IPv6 Protokoll
- realisiert in Extension Headern
- ESP - Encapsulated Security Payload
- AH - Authentication Header

ABER

- viele OS-Hersteller haben die IPsec Funktionalität nicht implementiert
- die grundlegenden IPsec Probleme sind natürlich auch in IPv6 nicht gelöst
  - IKE Probleme
    - unterschiedliche Herstellerdefaults
    - IKEv2 nicht abwärtskompatibel zu IKEv1
- auch die PKI Probleme bestehen weiter
  - welche Zertifikate sind vertrauenswürdig
  - sollen root-Zertifikate automatisch ‘mitgeliefert’ werden ?
- neue Mobility-Standards verlassen sich auf IPsec, trotz der Probleme

Änderungen aus Benutzersicht

?? Fragen ??

Agenda

- Motivation für IPv6 (warum ein neue IP Version)
- Protokollübersicht
- Änderungen aus Benutzersicht
- Transition / CoExistence / Routing

Transition

im Sinne von IPv6 ersetzt IPv4 wird es wohl nie bzw. noch sehr lange nicht geben.

CoExistence

Dual Stack

- IPv4 und IPv6 parallel aktiv
- gewohnter, bekannter Ansatz
- IPv6 bei allen aktuellen Betriebssystemen inkludiert
- beide ‘Welten’ erreichbar, Host wählt (hoffentlich) das richtige Protokoll mittels DNS, zuerst ‘AAAA-Record’ (IPv6), wenn keine Antwort ‘A Record’ (IPv4)
- erlaubt langsamen, sanften Übergang
- löst allerdings nicht das Problem der Adressknappheit

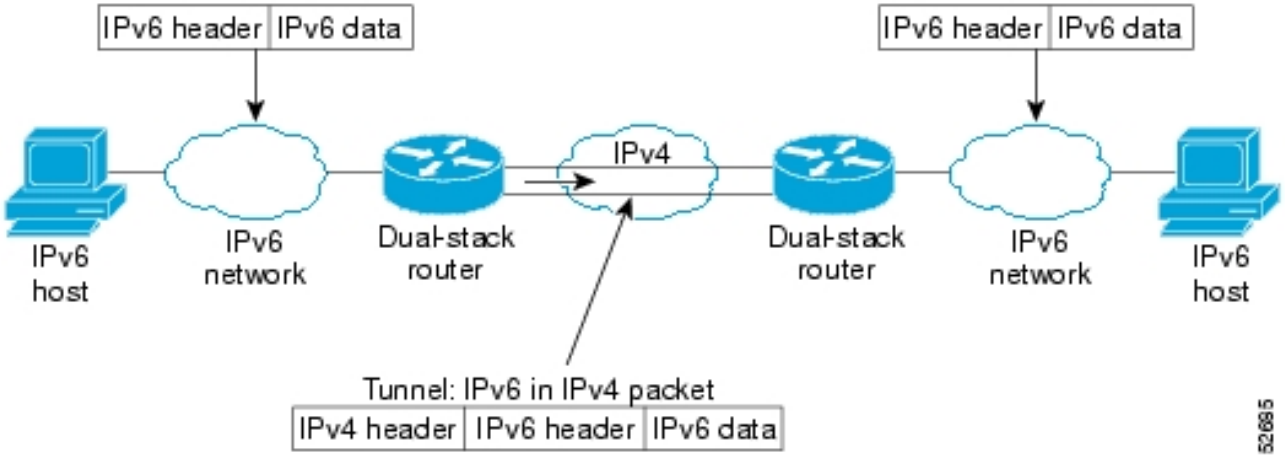
CoExistence

Tunnel

zur Verbindung von Inseln der einen IP-Version über Netze der anderen IP-Version, ie. IPv6-in-IPv4 Tunnel oder IPv4-in-IPv6 Tunnel

manuell konfiguriert
nur sehr begrenzt sinnvoll, Skalierbarkeit, Wartbarkeit

tunnel broker
Spezialfall, wo eine zentrale Entität Tunnel-Infrastruktur zur Verfügung stellt, meist gescriptet/automatisiert (z.B. http://www.sixxs.net)



CoExistence

Tunnel - 6to4 - RFC3056 - Automatischer Tunnelaufbau, zur Verbindung von IPv6 Inseln über das IPv4 Netz

- definierter Prefix **2002::/16**
- Eine global eindeutige IPv4-Adresse (am Border-Router) genügt, um eine ganze Site IPv6-fähig zu machen
- erstellt automatisch die notwendigen IPv6-in-IPv4 Tunnel, ie. verwendet das IPv4-Netz als Point-to-Point link
- erfordert 6to4-Relays um auch die Kommunikation zum 'eigentlichen' IPv6-Netz zu gewährleisten
- RFC3068 definiert einen anycast prefix für 6to4-relays, d.h. man muss sich keinen offenen relay suchen, sondern die anycast Adresse 192.88.99.1 (in 6to4 2002:c058:6301::) sorgt dafür, dass das nächste 6to4-relay verwendet wird.

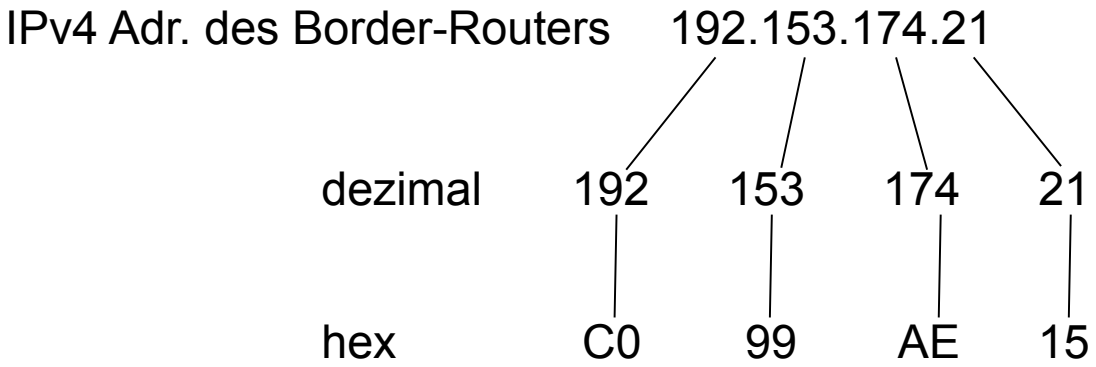
```
Wien21#sho bgp 192.88.99.1     BGP routing table entry for
192.88.99.0/24, version 80129474Paths: (3 available, best #2,
table Default-IP-Routing-Table) 1239    80.66.136.3 from
80.66.136.3 (80.66.136.3)      Origin IGP, localpref 100, valid,
external  20965 559    62.40.124.1 from 62.40.124.1
(62.40.114.1)      Origin IGP, localpref 130, valid, external, best
Community: 20965:155
```

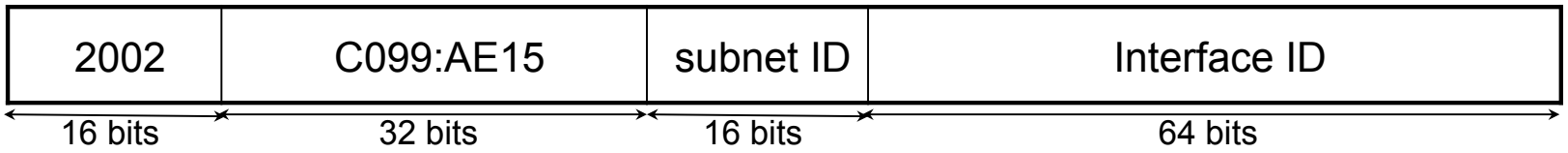
AS559 - Switch

CoExistence

Tunnel - 6to4

Wie kommt der (global eindeutige) 6to4-Prefix zustande:

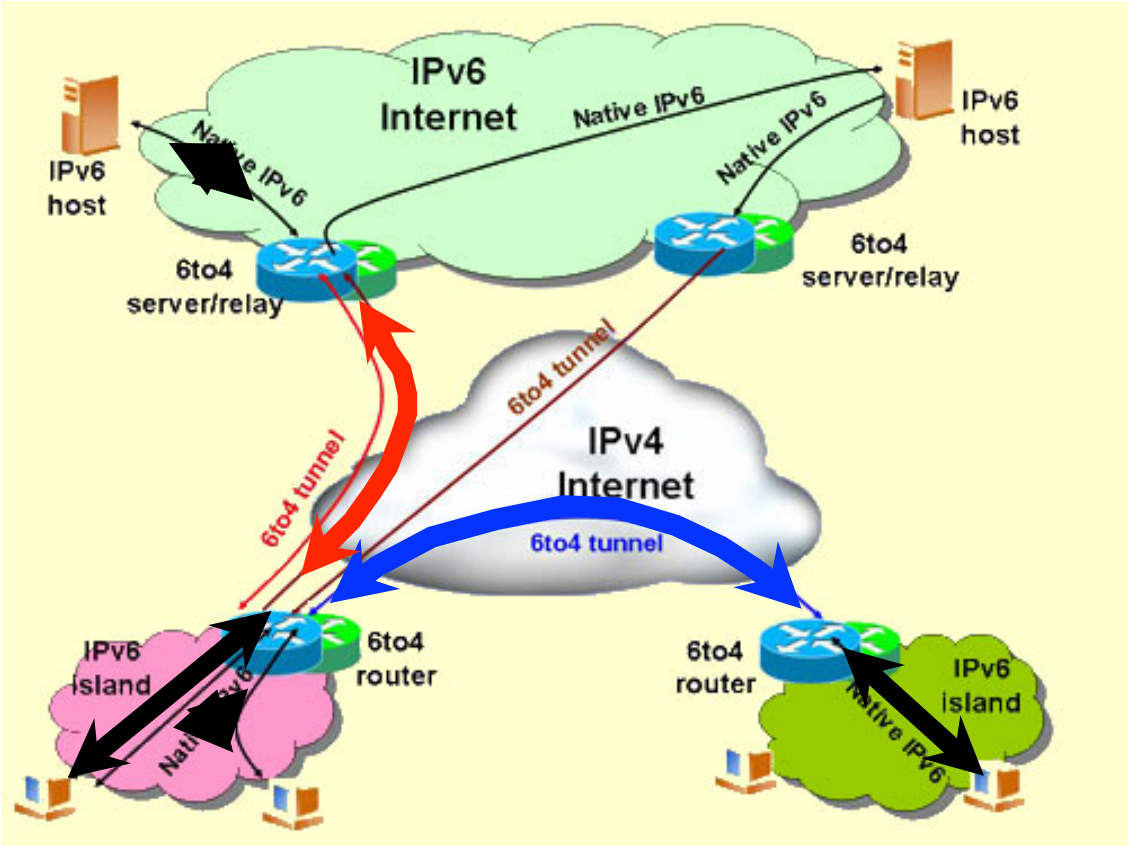




z.B.: 2002:C099:AE15:0001:216:cbff:fe96:a591/64

CoExistence

Tunnel - 6to4



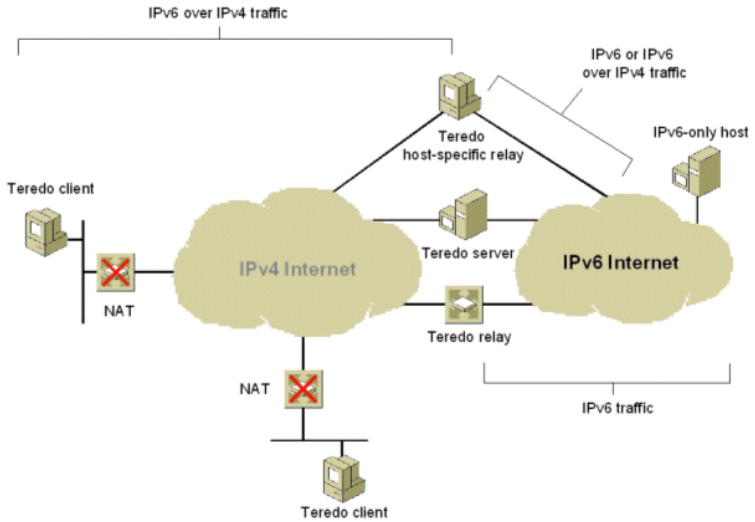
CoExistence

Tunnel - Hostbasiert

isatap (Intra-Site Automatic Tunnel Addressing Protocol)

Ermöglicht innerhalb einer Site einzelne IPv6-fähige isolierte Hosts, durch das IPv4-Netz getunnelt, an das IPv6-Netz anzuschließen.

teredo - RFC4380



Erlaubt es IPv4 Rechnern hinter einem NAT System mittels eines TEREDO Servers das IPv6 Internet zu erreichen. Dazu werden die IPv6 Pakete in IPv4 UDP Pakete verpackt, die das NAT System traversieren können.

CoExistence

weitere CoExistence-Mechanismen

NAT-PT (RFC2766 bzw. RFC4966)

- Probleme mit Diensten die IPv4 Adressen in der Payload mitführen
- fragmentation Probleme
- mobility Probleme
- kein Multicast
- Probleme mit Application Layer Gateways
- ...

➡ Empfehlung RFC2766 als ‘historic’ zu definieren

MPLS - 6PE (RFC4798)

- analog zu MPLS VPN Service
- Provider Edge Router müssen Dual-Stack und MP-BGP fähig sein
- Core ist IPv4-only
- BGP Next-Hop Field enthält die IPv4 Adresse des 6PE Routers, um den Label Switched Path zu erstellen (ohne explizite Tunnelkonfiguration)

Transition / CoExistence

?? Fragen ??

Routing

- **statisch**
  - wie gehabt
- **RIP**
  - RIPng unterstützt IPv6
- **OSPF**
  - OSPFv3 unterstützt nur IPv6, d.h. im Dual-Stack Szenario müssen OSPFv2 (für IPv4) und OSPFv3 (für IPv6) parallel betrieben werden
- **IS-IS**
  - Multiprotokoll-fähig, daher problemlose Unterstützung von IPv6
- **EIGRP**
  - ebenfalls Multiprotocol fähig, ab 12.4(6)T bzw. 12.2(33)SRB
- **BGP**
  - MultiProtocol-BGP (BGP4+) notwendig, unterstützt diverse Adressfamilien, IPv4 und IPv6, jeweils Unicast und Multicast, MPLS, ...

Routing

?? Fragen ??

Agenda

- Motivation für IPv6 (warum ein neue IP Version)
- Protokollübersicht
- Änderungen aus Benutzersicht
- Transition / CoExistence / Routing
- → DNS slides