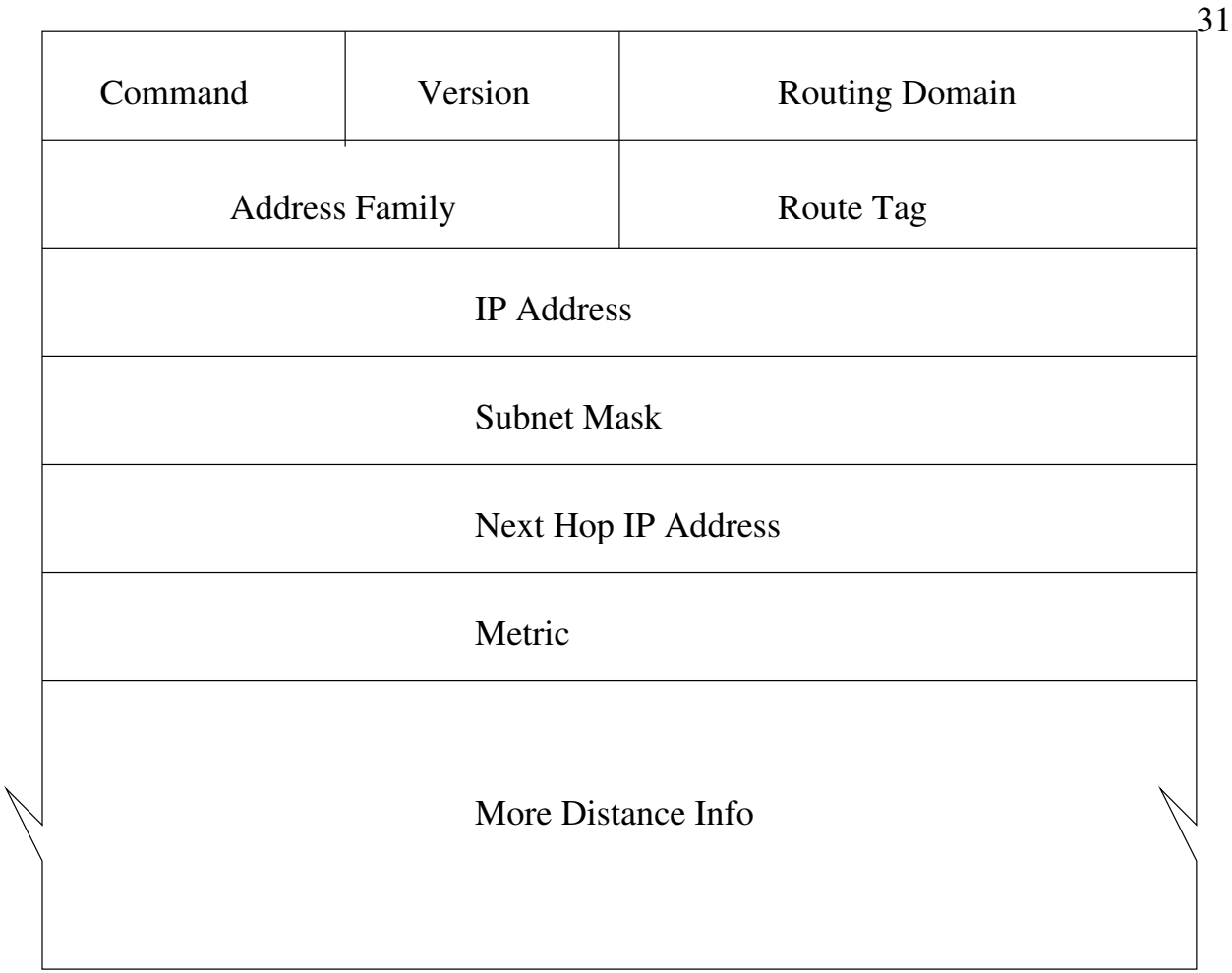
Routing
(Fortsetzung)

RIP Version 2 (vgl. RFC2453)



RIP Version 2

RIP Version 2 ist ein Distance Vector basiertes Verfahren.

- **Command:** Entweder Request(1) oder Reply(2)
- **Version:** RIP Version 2 (RIP Version 1 unterstützt kein CIDR, das Rahmenformat ist gleich)
- **Routing Domain:** Indentifier für RIP process (in Version 2)
- **Address Family:** Für IP immer 2
- **Route Tag:** Tag zur Identifikation des AS (ASN, RFC1930)
- **IP Address:** Addresse des Subnetzes, zu dem die Information gehört
- **Subnet Mask:** Netzmaske des Subnetzes
- **Next Hop IP Address:** IP Addresse des Routers, über den das Subnetz erreicht werden kann
- **Metric:** Kosten, bei RIP Anzahl Hops zum Ziel, max. 15
- Pro Rahmen bis zu 25 Distanzinformationen je 20 Byte.

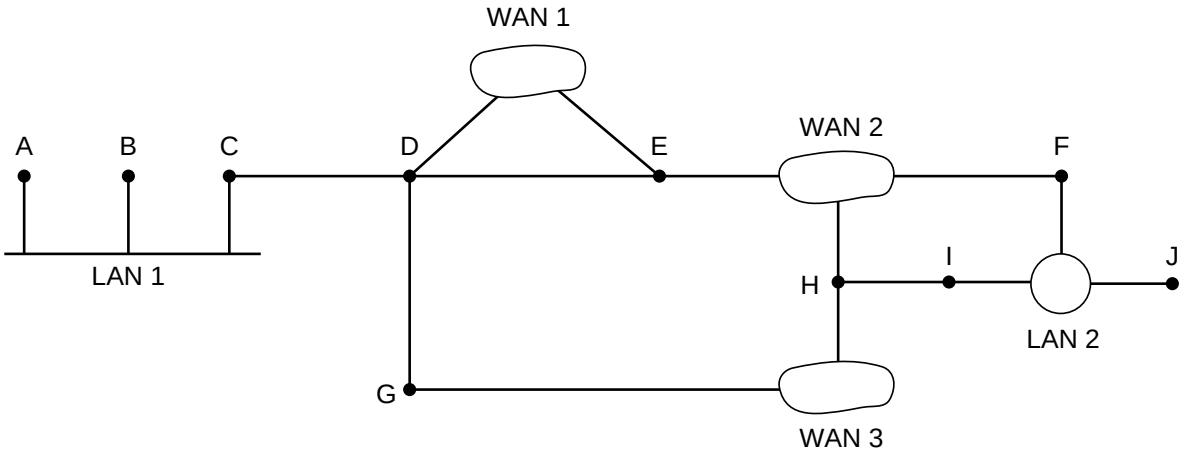
RIP Version 2

- Bei Start schickt der Router einen RIP2 Request mit Address Family 0 (statt 2) auf allen Interfaces. Dieser fordert von allen Routern den kompletten Satz Distanzvektoren an.
- In einem Request mit Address Family 2 wird jeder Eintrag im Rahmen bearbeitet. Falls eine Route vorhanden ist, setze Metric, andernfalls setze Metric auf 16 (unendlich).
- Wird ein Reply empfangen, verwende den Distance Vector Algorithm zum Neuaufbau der Routingtabelle.
- Alle 30 Sekunden wird die komplette Routingtabelle an alle Nachbarn verschickt.
- Bei jeder Veränderung der eigenen Routingtabelle werden die Änderungen der Metric übertragen.
- Jeder Routingeintrag verfällt nach max. 2 Minuten, wenn er nicht erneuert wird.

Open Shortest Path First (vgl. RFC2328)

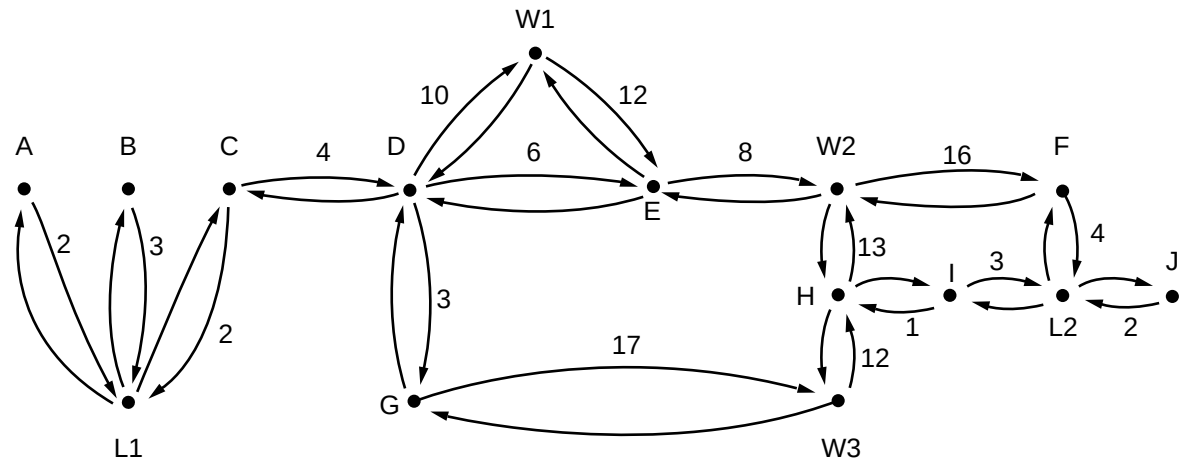
- OSPF Daten werden im IP Rahmen übertragen (Protocol 89)
- OSPF benutzt den Link State Algorithmus zur Berechnung der Routingtabelle.
- Anwendung im Intra-AS Routing
- Bei OSPF werden die Linkstati eines Routers zu allen Routern des AS übertragen.
- Jeder Router berechnet mit dem Dijkstra Algorithmus die Route geringster Kosten zu allen anderen Routern.
- Links werden mit OSPF Paketen auf Funktionalität geprüft.

Beispiel OSPF Graph



(a)

Beispiel AS



(b)

Beispiel Graph

(c) Tanenbaum, Computer Networks

Eigenschaften von OSPF

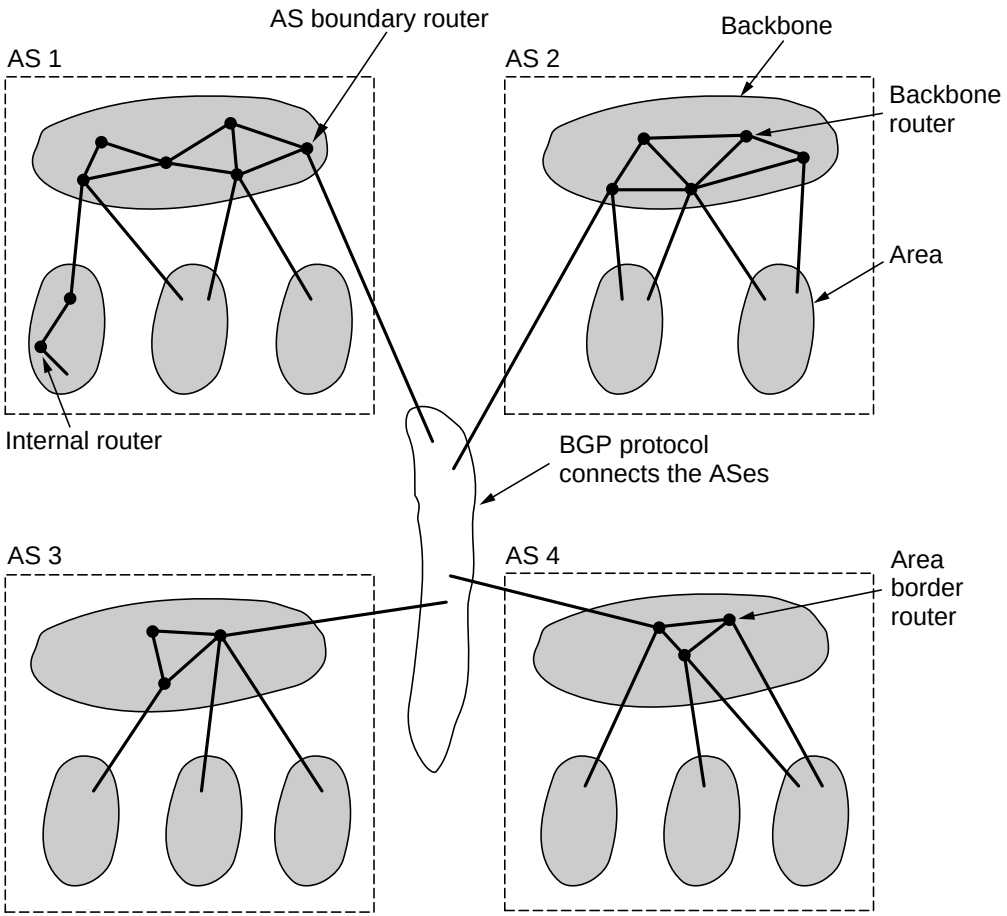
- OSPF unterstützt echte Authentifizierung.
- Mehrere Pfade mit gleichen Kosten können parallel benutzt werden (Load Balancing).
- OSPF unterstützt verschiedene Routen abhängig vom TOS Feld.
- Kosten eines Links sind dimensionslos, unterschiedliche Kosten können für unterschiedliche Werte des TOS Feldes benutzt werden.
- Routen müssen nicht durch IP Adressen identifiziert werden (vgl. PPP)
- OSPF erlaubt eine hierarchische Aufteilung des AS in kleinere AS (“Areas”), die durch **Area Border Router** mit dem **Backbone AS** verbunden sind.

Routertypen von OSPF

Vier Typen von Routern existieren in OSPF

- Internal Router: Router in einer Area, die nicht mit dem Backbone verbunden sind
- Area Border Router: Verbinden Area und Backbone
- Backbone Router: Interne Router im Backbone
- Boundary Router: Verbindung zu anderen AS

Beispiel OSPF Hierarchie



(c) Tanenbaum, Computer Networks

Border Gateway Protocol Version 4 (vgl. RFC4271)

- Beispiel eines Path-Vector Protokolls
- Dient als Inter-AS Routing Protokoll im Internet
- BGP4 muß von jedem ISP eingesetzt werden, um Routing zu anderen AS zu unterstützen.
- In Benutzung seit 1994 (Version 4 unterstützt im Gegensatz zu Version 3 CIDR)
- Routingentscheidungen zwischen AS basieren nicht auf Kosten sondern auf Regeln.

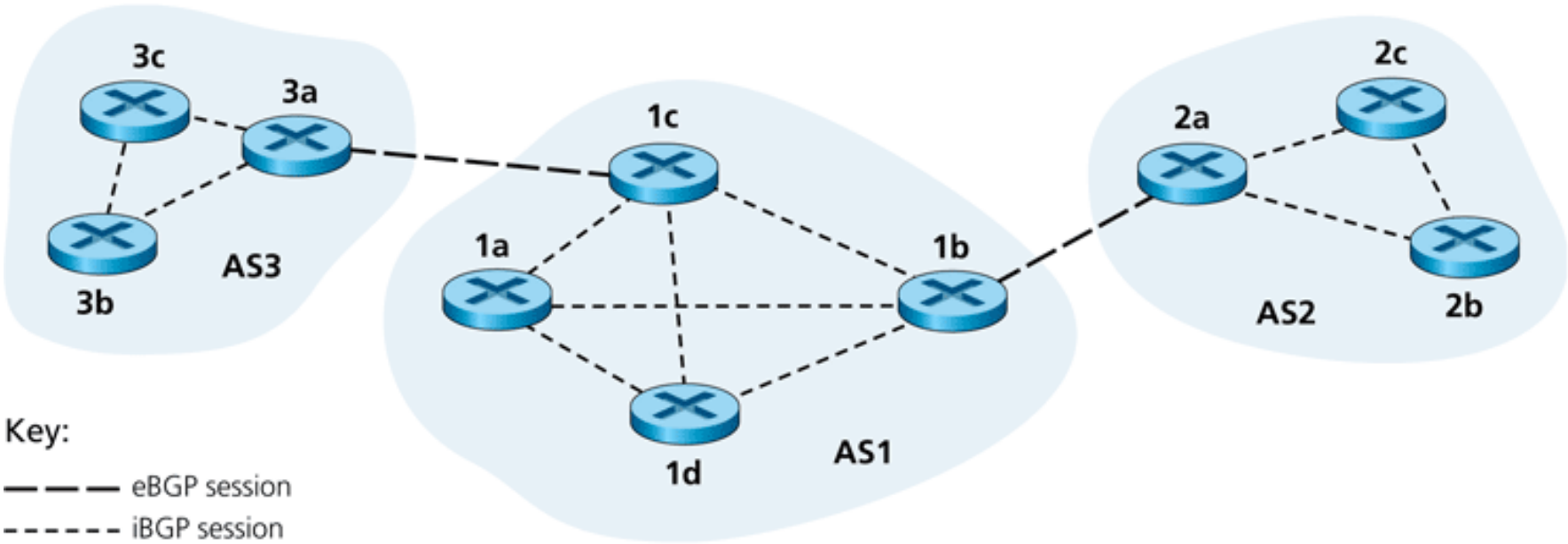
Bezeichnungen

- **BGP Speaker:** Knoten, der das BGP Protokoll implementiert.
- **eBGP Kommunikation:** BGP Datenaustausch zwischen AS
- **iBGP Kommunikation:** Datenaustausch zwischen BGP Speaker und seinem AS
- **Präfix:** Netzwerkidentifikation bestehend aus IP und Maske, z.B. 134.130.35.128/25
- **BGP Attribut:** Zusätzliche Information zu einem Präfix, besonders wichtig sind die Attribute AS-Path und Next-Hop
- **Route:** Kombination von Präfix und zugehörigen Attributen

Generelle Arbeitsweise

- Jede Router hat eine Liste seiner Nachbarn, zu denen er eine Verbindung aufbaut.
- Verbindungen werden durch einen Keepalive Mechanismus permanent überwacht.
- Gateway Router tauschen via eBGP Kommunikation die verwendeten Routen mit ihren Nachbarn aus.
- Router innerhalb des AS informieren sich gegenseitig über die verwendeten Routen via iBGP Kommunikation
- Aus den möglichen Routen wählt jeder Router anhand seiner Konfiguration die für ihn beste aus.

Beispiel BGP Sessions



(c) Kurose and Ross, Computer Networking

Routenerzeugung

- Wird ein Präfix von einem Router via eBGP an einen anderen Router gemeldet, fügt er seine AS Nummer ASN (vgl. RFC1930) an das AS-Path Attribut an und setzt das Next-Hop Attribut auf die IP Adresse des externen Interfaces.
- Jeder Router innerhalb eines AS wird eine Route zum Präfix über den Next-Hop berechnen. Dazu wird ein Intra-AS Routingverfahren (z.B. OSPF) verwendet.

Routenselektion

Ein Router kann verschiedene Routen zum selben Ziel über verschiedene BGP Speaker empfangen.

- Kann der Next-Hop nicht erreicht werden, verwerfe den Pfad
- Wähle die Route mit der höchsten Präferenz (wird über Regeln vom Administrator festgelegt).
- Bei gleicher Präferenz, wähle die kürzeste Route, d.h. mit dem kürzesten AS-Path Attribut.
- Unter den verbliebenen Routen, wähle die mit den günstigsten Kosten zum Next-Hop.

Internet Protokoll Version 6 (IPv6)

Zustand und Probleme von IPv4

- Router im Internet haben > 200000 Einträge in der Routingtabelle
- IP Adressen sind eine extrem knappe Resource (mit 32 Bit sind $2^{32} = 4294967296 \approx 4 \cdot 10^9$ Adressen möglich, aber durch den hierarchischen Aufbau, reserviert Adressbereiche,... ist der Nutzungsgrad nicht hoch)
- Temporäre Adressvergabe mittels DHCP, private Adressbereiche und NAT helfen, die vorhandenen Adressen effizient zu nutzen.
- Viele Dienste sind nur mit Hilfe neuer und komplizierter Protokolle möglich, z.B.:
  - Multimedia Messaging Service (MMS) (vgl. www.openmobilealliance.org)
  - Mobile E-Mail
  - ...

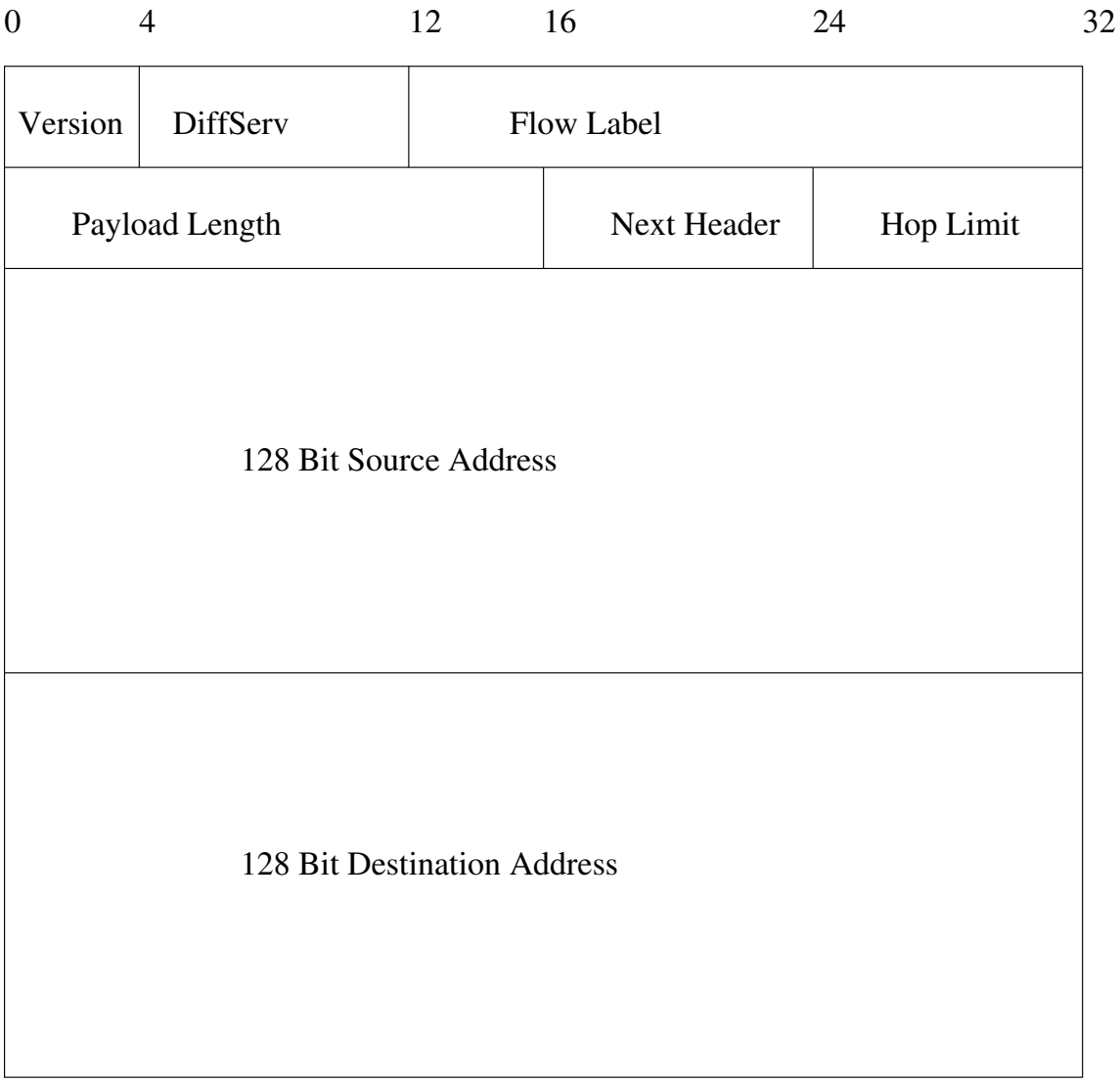
Ziele und Neuerungen von IPv6

- Massive Vergrößerung des Adressraums, sodass selbst bei ineffizienter Nutzung immer genug Adressen vorhanden sind.
  - → 128Bit für Adressen, nicht 32Bit
  - → 128Bit ermöglicht $2^{128} = 3 \cdot 10^{38}$ Adressen
  - → $\approx 7 \cdot 10^{23}$ Adressen pro Quadratmeter Erdoberfläche
  - → Bei “ineffizientester” Nutzung $\approx 1000/m^2$ (vgl. RFC3194)
- Verkleinerung der Routingtabellen
  - → Hierarchische Netzstruktur durch gezielte Adresszuweisung
- Vereinfachung des Protokolls um Verarbeitung in Routern zu beschleunigen
  - → Minimaler Basisheader
  - → Keine Checksum mehr im Header
- Koexistenz mit IPv4 und zukünftige Erweiterbarkeit
  - → Flexible Erweiterungsmöglichkeiten des Basisheaders

Ziele und Neuerungen von IPv6 (2)

- Besser Anpassungen an verschiedene Serviceklassen (Real-Time, ...)
  - → Differentiated Services sind Standard
- Automatische Konfiguration ist ohne Zusatzprotokolle möglich.
- Host sollen ihren Standort verändern können ohne Adressänderung (Roaming)
  - → Dienste für mobile Terminals werden unterstützt
- Verbesserung bei der Sicherheit (Authentifizierung und Vertraulichkeit)
  - → Sicherheitsdienste werden von der Vermittlungsschicht transparent angeboten.
- Verbesserte Integration von Multicast Diensten

IPv6 Header


0
4
12
16
24
32
Version
DiffServ
Flow Label
Payload Length
Next Header
Hop Limit
128 Bit Source Address
128 Bit Destination Address

IPv6 Header

- **Version:** 4 Bit Version, 6 für IPv6
- **DiffServ:** Differentiated Services Kennung, entspricht IPv4
- **Flow Label:** Kennung zusammengehöriger Pakete, vgl. RFC3697
- **Payload Length:** 16 Bit Länge der Daten in Byte (d.h. exklusive des 40 Byte Basisheaders)
- **Next Header:** Typ des nächsten Headers, entweder Protocol (RFC1700) wie in IPv4, oder einer der standardisierten Erweiterungsheader
- **Hop Limit:** Max. Anzahl Hops, vgl. IPv4 TTL
- **Source Address:** 128 Bit Quelladresse
- **Destination Address:** 128 Bit Zieladresse

IPv6 Adressen (vgl. RFC3513)

Adressen identifizieren Schnittstellen (Interfaces, vgl. RFC2460, Section 2), mit denen Knoten mit dem Netzwerk verbunden sind.

In IPv6 werden drei Typen von Adressen unterschieden:

- **Unicast:** Adresse einer Schnittstelle, Pakete werden zu genau dieser Schnittstelle weitergeleitet.
- **Anycast:** Adresse für eine Gruppe von Schnittstellen, ein Paket wird zu einer dieser Schnittstellen weitergeleitet.
- **Multicast:** Adresse für eine Gruppe von Schnittstellen, Pakete werden zu allen Schnittstellen der Gruppe ausgeliefert.

Schreibweise für IPv6 Adressen

- Die 16 Byte einer Adresse in Network Byte Order werden geschrieben als 8 Segmente von je 2 Byte in hexadezimaler Darstellung, durch Doppelpunkte getrennt.
- Führende Nullen in Segmenten können weggelassen werden.
  Beispiel: 00 01 02 03 04 05 06 07 18 19 1A 1B 1C 1D 1E 1F
  geschrieben: 1:203:405:607:1819:1A1B:1C1D:1E1F
- Eine Gruppe von aufeinander folgenden, leeren Segmenten kann durch zwei Doppelpunkte abgekürzt werden:
  Beispiel: 00 01 00 00 00 00 00 00 00 00 1A 1B 1C 1D 1E 1F
  geschrieben: 1::1A1B:1C1D:1E1F
- Alternative Schreibweise: Die letzten 4 Byte können als “Dotted Notation” geschrieben werden:
  Beispiel: 00 01 00 00 00 00 00 00 00 00 1A 1B 01 02 03 04

Schreibweise für Netzwerke

- Netzwerkpräfixe werden in CIDR Notation (Adresse/Länge) geschrieben.
- Segmente, die außerhalb der Maske liegen, brauchen nicht aufgeführt zu werden.
  Beispiel: das 60 Bit Präfix `12AB 0000 0000 CD3` kann geschrieben werden als:
  - 12AB:0:0:CD30/60
  - 12AB:0:0:CD30:0:0:0:0/60
  - 12AB:0:0:CD30::/60
- Alle möglichen Mehrdeutigkeiten sind **nicht** erlaubt, z.B.:
  - 12AB:0:0:CD3/60, Segment vier kann auch 0CD3 sein.
  - 12AB::CD30/60, Würde interpretiert als 12AB 0000 0000 000

Adresstypen

Die unterschiedlichen Adresstypen und Adressbereiche sind Subnetze des IPv6 Adressraumes:

Type	Prefix
Unspecified	::/128
Loopback	::1/128
Multicast/Anycast	FF00::/8
Link-local unicast	FE80::/10
Site-local unicast	FEC0::/10
Global unicast	everything else

Aufbau von Global Unicast Adressen

Adressen, die nicht mit ::/12 beginnen, enden in einer 64 Bit Schnittstellenadresse gemäß IEEE EUI-64.

Beispiel: Die 48 Bit Ethernetadresse einer Schnittstelle wird zu einer 64 Bit Schnittstellenadresse expandiert, indem Bit 1 des ersten Bytes auf 1 gesetzt wird (ist in der Ethernetadresse immer 0, da OUI), Byte 2 und 3 unverändert übernommen werden, dann wird 0xFFFE eingefügt, dann folgen die letzten 3 Byte der Ethernetadresse:

Beispiel: Aus der Ethernetadresse 08:00:46:9E:92:0E wird . . .:0A00:46FF:FE9E:920E

Für andere Interfacetypen finden sich entsprechende Abbildungsregeln in den RFCs, die die Übertragung von IPv6 über den Link spezifizieren.

Mapping von IPv4 Adressen

Schnittstellen, die sowohl IPv4 als auf IPv6 bedienen können, können spezielle IPv6 Adressen bekommen (**IPv4 compatible address**), die die IPv4 Adresse als letzte 4 Bytes enthalten. Diese Adressen werden heute kaum noch verwendet:

Beispiel: IPv6 fähige Schnittstelle mit IPv4 Adresse 1.2.3.4 hat IPv6 Adresse ::1.2.3.4

Soll eine Anwendung sowohl mit IPv4 als auch IPv6 funktionieren, wird oft IPv6 als Obermenge verwendet. Adressen einer Schnittstelle werden dann als sogenannte **IPv4 mapped address** vom Stack geliefert, wenn es sich um eine IPv4 Adresse handelt.

Beispiel: Schnittstelle mit IPv4 Adresse 1.2.3.4 hat IPv4 mapped address ::FFFF:1.2.3.4

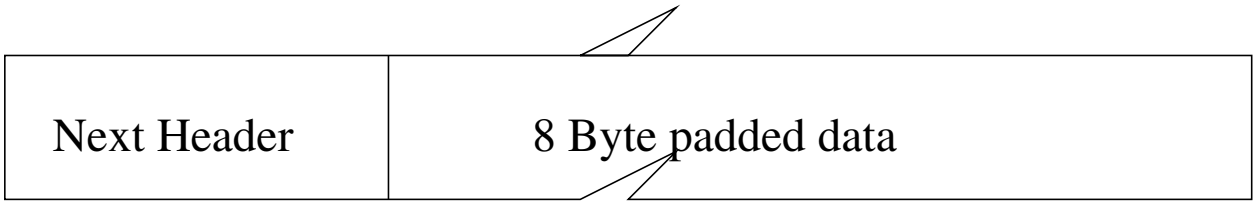
IPv6 Erweiterungsheader

Erweiterungsheader beginnen auf 8 Byte Grenzen, die folgende Reihenfolge muß eingehalten werden:

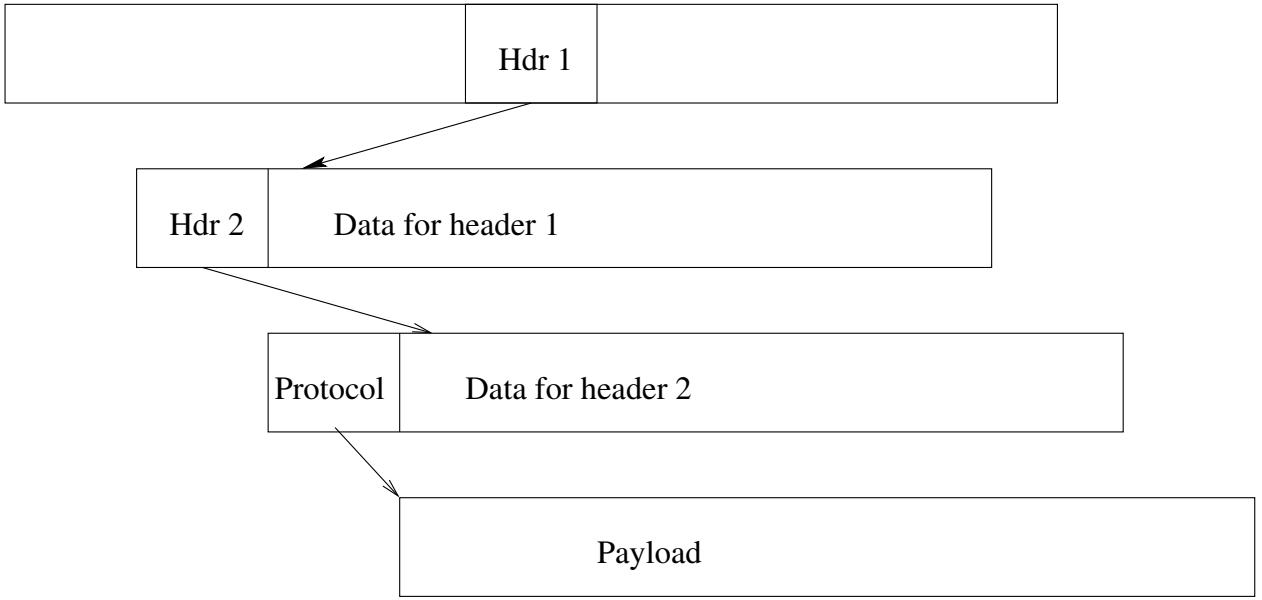
0	Hop-By-Hop Option (RFC1883)
60	Destination Option
43	Routing Header
44	Fragment Header
51	Authentication Header (RFC2402)
50	Encapsulating Security Payload (RFC2406)
59	No Next Header
60	Destination Option
135	Mobility Header

Erweiterungsheader: Format und Verkettung

Format eines Erweiterungsheaders:



Headerfolge, ähnlich IPv4 Optionen:



Hop-By-Hop Option

- Alle Optionen, die von jedem Router ausgewertet werden müssen, werden in Hop-By-Hop Headern übertragen.
- Der Header hat immer die Felder
  1. Next Header
  2. Length, gezählt in Bytes ab Byte 8
  3. Parameter
- Parameter sind Typ/Länge/Wert kodiert, bis auf Typ 0, der keine Parameter besitzt.
- Type 0 und 1 dienen als Füller
- Typ 194 (Jumbo Payload) signalisiert Pakete bis 4GByte (sonst bis 64kByte)
- Weitere Typen sind in der Standardisierung

Destination Option

- Die Destination Option ist von jedem Router auszuwerten, sofern sie vor dem Routing Header auftritt, sonst nur vom Zielhost.
- Der Aufbau entspricht den Hop-By-Hop Headern.
- Einige Optionen sind standardisiert, z.B.:
  - ein Header, der die Zahl geschachtelter IPv6 Tunnel limitiert.
  - Home Address bei Mobile IP
- Optionen können aus Anwendungen gesetzt werden.

Routing Header

- Der Routing Header hat im wesentlichen dieselbe Funktion, wie Loose Source Routing bei IPv4, allerdings ohne deren Limitierungen.
- Ein Äquivalent zu Strict Source Routing wird in IPv6 bisher nicht angeboten.
- Der Header besteht wie bei IPv4 aus einer Länge (8 Bit gezählt in 8 Byte Einheiten), einer Liste von Adressen und einem Zeiger in diese Liste, damit die Router das nächste Ziel erkennen können.
- Ein 8 Bit Typ Feld sowie 4 Byte Padding sichern Erweiterbarkeit.

Fragment Header

- Bei IPv6 fragmentiert nur der sendende Endpunkt einer Kommunikationsbeziehung, keine Router im Pfad.
- Die MTU wird mit dem Path MTU Discovery Protocol (RFC1981) bestimmt.
- Der Fragment Header entspricht weitgehend den IPv4 Headerfeldern, die zur Fragmentierung genutzt werden, d.h. er enthält:
  - 13 Bit Offset in 8 Byte Einheiten
  - 32 Bit Identification
  - 1 Bit More Fragments
  - Mit 8 Bit Next Header und 10 Bit Padding erhält man 64 Bit
- Daten des IP Headers bis zum Fragment Header können nicht fragmentiert werden.

Authentication Header (AH)

- AH ist eines der Protokolle, die unter dem Namen IPSec transparente Sicherheitsdienste auf Ebene der Vermittlungsschicht anbieten.
- IPSec ist integraler Bestandteil von IPv6.
- Es gibt inzwischen viele Implementationen von IPSec auf Basis von IPv4.
- Authentizität wird von AH durch gegenseitige Authentifizierung gesichert.
- Integrität wird durch Signatur der Header und Daten sichergestellt.
- Sicherung gegen Mehrfacheinspielung (Replay Attack) von Daten ist optional.
- Vertraulichkeit ist **nicht** Bestandteil von AH

Encapsulating Security Payload (ESP)

- ESP bietet gegenüber AH erweiterte Sicherheitsdienste
- ESP kann in Verbindung mit AH verwendet werden.
- Es werden Tunnel Modus und Transport Modus unterschieden.
- Vertraulichkeit wird durch Verschlüsselung der Daten und im Tunnel Modus durch Aggregierung von Datenströmen erreicht.
- Gegenseitige Authentifizierung der Endpunkte (Host oder Router/Security Gateway) ist möglich.
- Verhinderung von Mehrfacheinspielung ist möglich.

Mobility Header

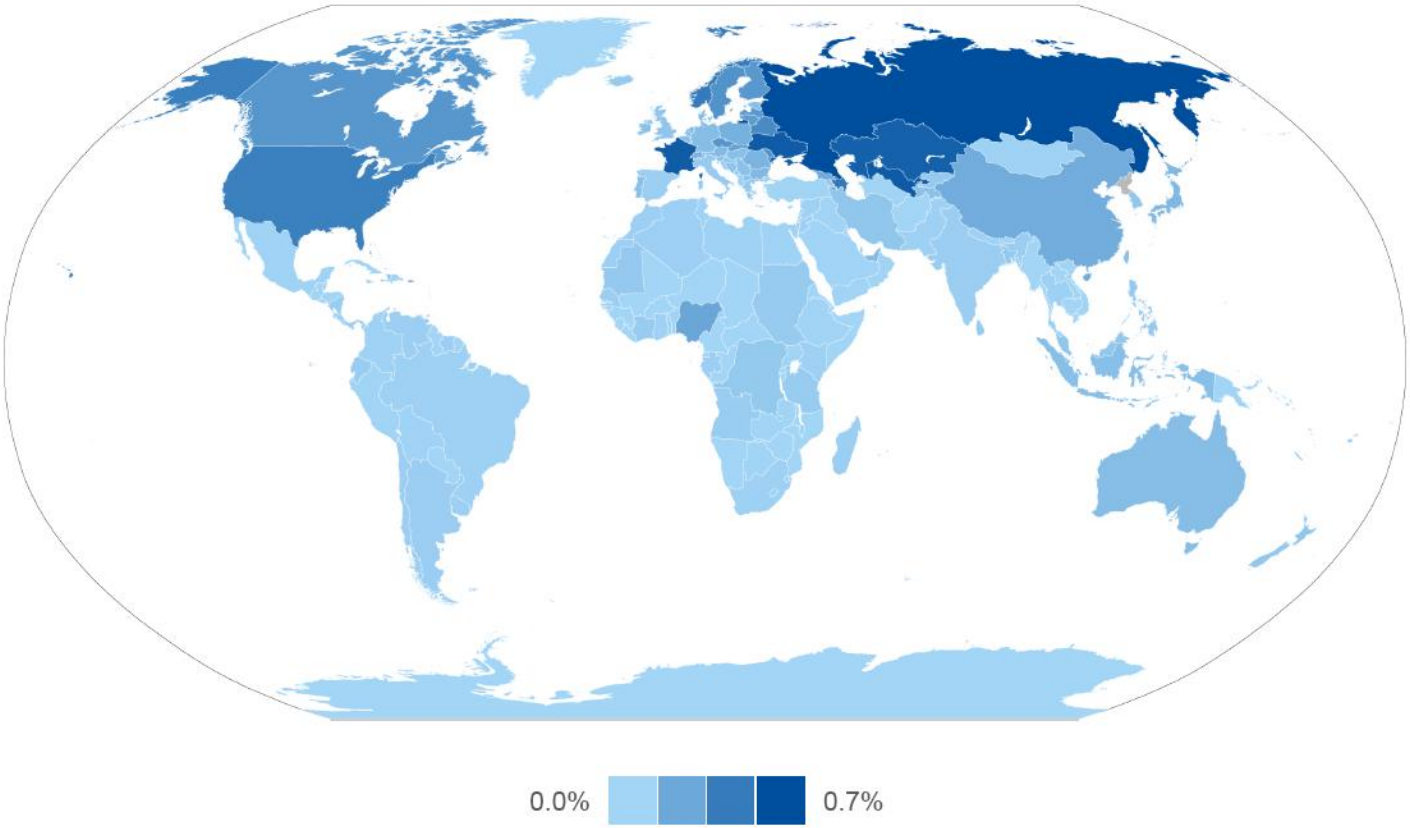
- Der Mobility Header wird benutzt, um Assoziationen zwischen mobilem Endgerät und Home-Agent herzustellen, aufzulösen oder zu testen.
- Folgende Nachrichtentypen sind spezifiziert:
  1. Binding Refresh Request Message, erzeugt Assoziation im Home-Agent
  2. Home Test Init Message, Care-of Test Init Message: Teil der “Return Routability Procedure”, die die Erreichbarkeit des Endgerätes sicherstellt
  3. Home Test Message: Antwort zur Home Test Init Message
  4. Care-of Test Message: Antwort zur Care-of Test Init Message
  5. Binding Update Message: Setzen einer neuen Care-Of-Address
  6. Binding Acknowledgement Message: Antwort auf den Binding Update Message

Migration zu IPv6

- Aktuelle Betriebssysteme (Windows, Unix Derivate, Symbian, IOS, ...) unterstützen IPv6
- Server und Infrastruktur im Internet sind inzwischen auf IPv6 eingerichtet, Protokolle angepasst, z.B.:
  - RIPng
  - BGP4+
  - DNS
- Viele Programme sind nicht fähig, IPv6 zu nutzen und werden es niemals sein.
- Know How für IPv6 ist kaum verfügbar, Erfahrungen gibt es kaum.
- Die “IPng Transition (ngtrans) working group” erklärt am 14.8.2002: **v6 considered operational**
- vgl. D.J. Bernstein: The IPv6 Mess, `http://cr.yp.to/djbdns/ipv6mess.html`

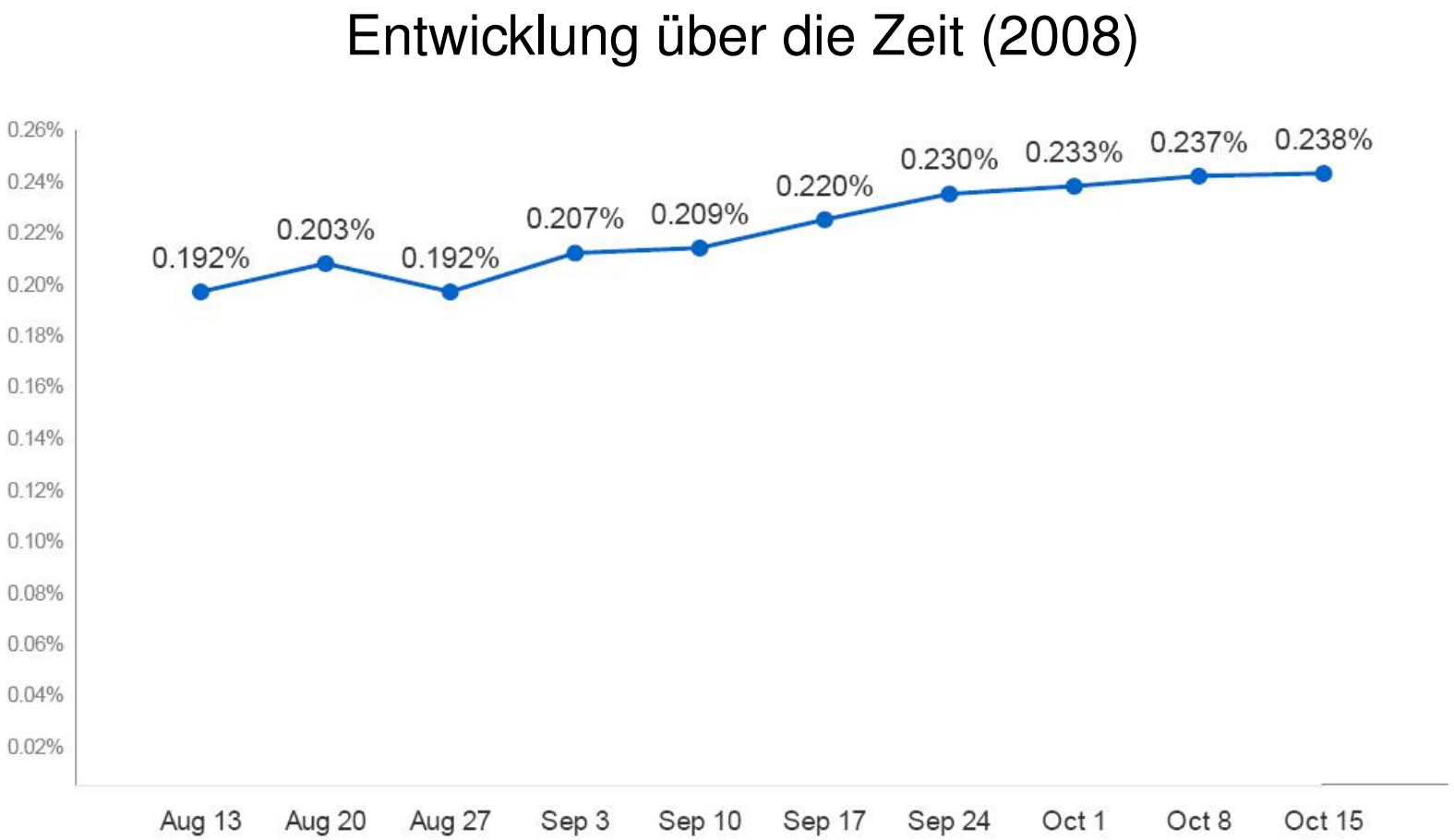
IPv6 Verbreitung - einige Statistiken von 2008

Verbreitung nach Ländern



Quelle: Global IPv6 statistics, S. H. Gunderson, Google, RIPE57, Dubai 2008

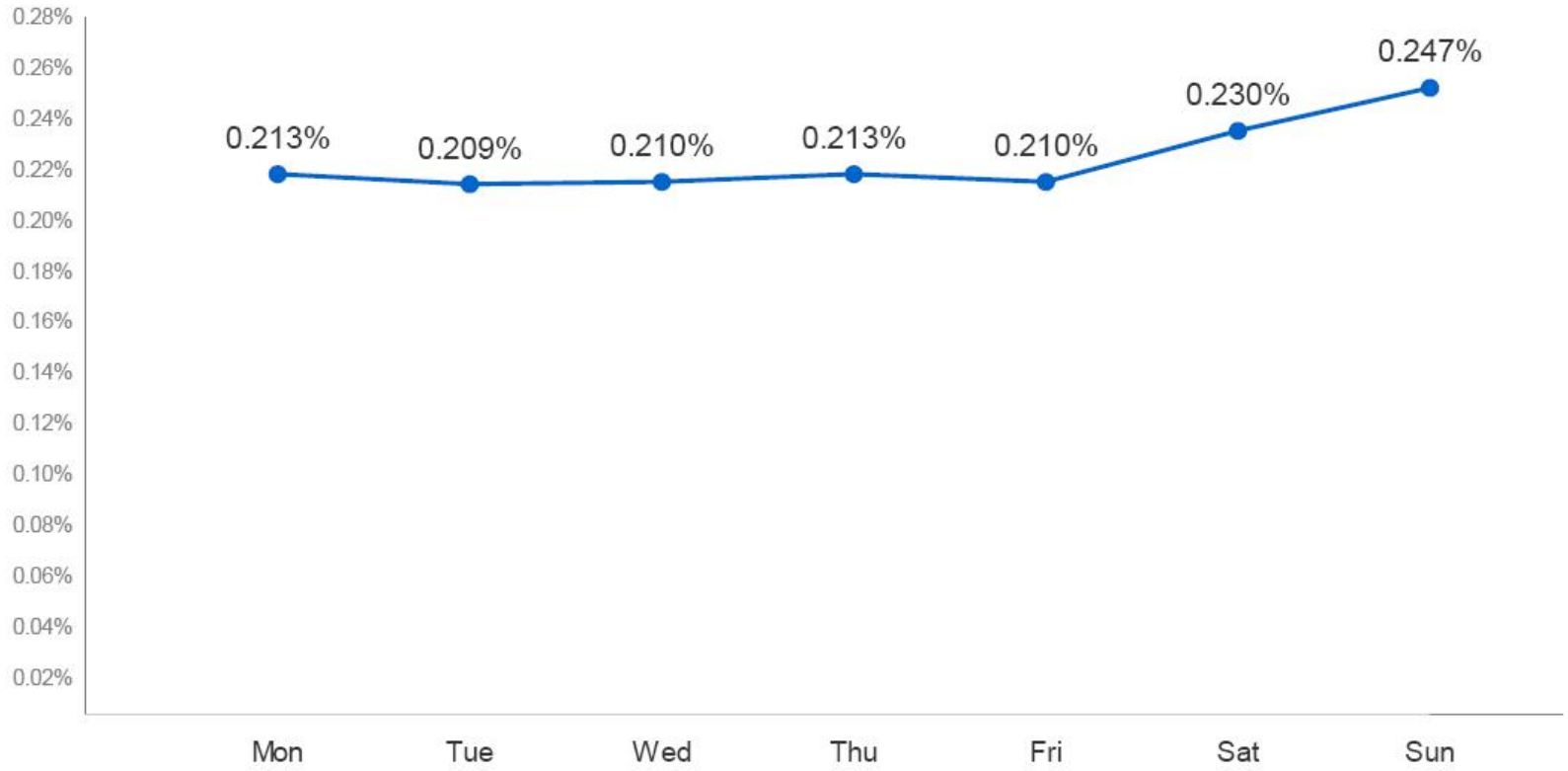
IPv6 Verbreitung - einige Statistiken von 2008



Quelle: Global IPv6 statistics, S. H. Gunderson, Google, RIPE57, Dubai 2008

IPv6 Verbreitung - einige Statistiken von 2008

Aufteilung auf Wochentage



Quelle: Global IPv6 statistics, S. H. Gunderson, Google, RIPE57, Dubai 2008

IPv6 Verbreitung - einige Statistiken von 2008

Aufteilung auf Betriebssysteme

IPv6 penetration and connectivity type by operating system
Ranked by overall IPv6 penetration

Operating system	IPv6 penetration	Native/other proportion	6to4 proportion	Teredo/ISATAP proportion
Mac OS	**2.44%**	**9%**	**91%**	**0%**
Linux	0.93%	86%	13%	1%
Windows Vista	0.32%	55%	43%	2%
Windows Server 2003	0.07%	–	–	–
Windows XP	0.03%	50%	30%	20%
Windows 2000	<0.01%	–	–	–

52% of all IPv6 hits are from Macs with 6to4

97% of all Teredo users are on Windows (even undercounting Vista)

Quelle: Global IPv6 statistics, S. H. Gunderson, Google, RIPE57, Dubai 2008

IPv6 Verbreitung - einige Statistiken von 2008

Fazit

- Große Unterschiede zwischen Ländern
- Starke regionale Abhängigkeit von einzelnen Faktoren, z.B. ISP free.fr in Frankreich
- Starke Abhängigkeit vom Betriebssystem
- Großteil des IPv6-Verkehrs wird über das 6to4 Protkoll über IPv4 getunnelt