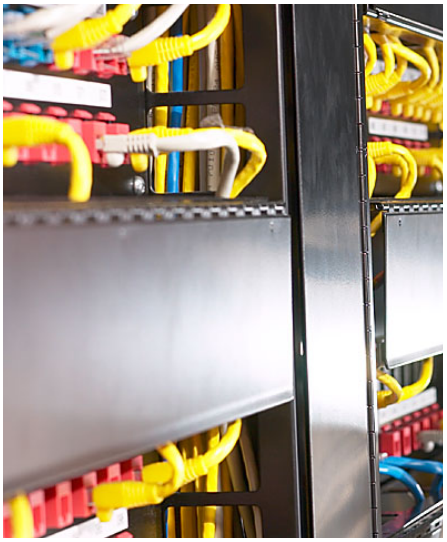
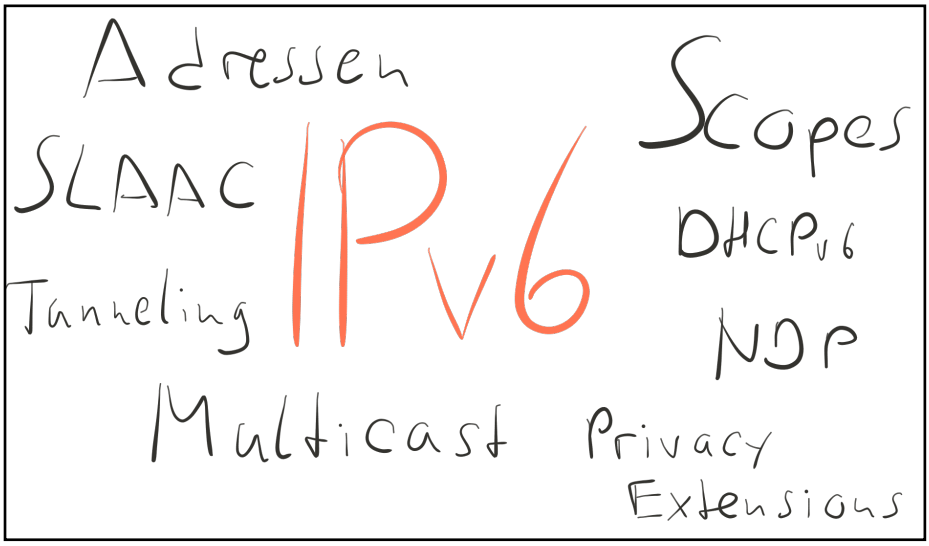
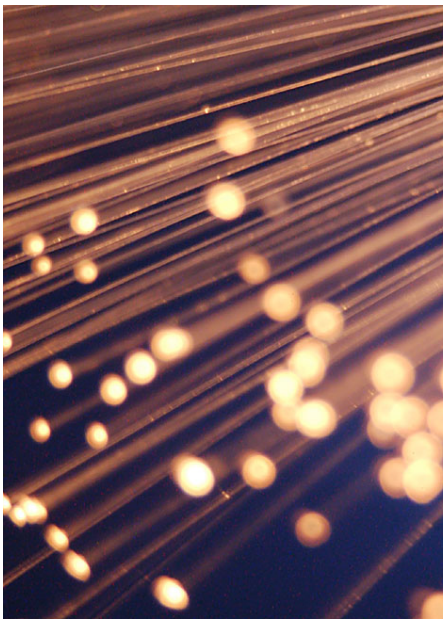




IPv6

Internet Protocol Version 6



IPv6 - Internet Protocol Version 6

Was Sie über IPv6 wissen müssen.

Patrick Schnabel

Systematische Einführung in die Grundlagen von IPv6 mit detailreichen Beschreibungen und zahlreichen Tabellen und Abbildungen. Mehr über die neuen Möglichkeiten erfahren und die Abläufe und Zusammenhänge im IPv6-Netzwerk verstehen.

www.elektronik-kompendium.de

IPv6 Internet Protocol Version 6

Was Sie über IPv6 wissen müssen.

Autor und Herausgeber

Patrick Schnabel
Droste-Hülshoff-Str. 22/4
D-71642 Ludwigsburg

Auflage

August 2016

http://www.elektronik-kompendium.de/

Vorwort

Ich wünsche Ihnen viel Freude und neue Erkenntnisse beim Lesen.

Patrick Schnabel

Einführung

IPv6 ist als Internet Protocol (Version 6) für die Vermittlung von Datenpaketen durch ein paketvermittelndes Netz, die Adressierung von Netzknoten und -stationen, sowie die Weiterleitung von Datenpaketen zwischen Teilnetzen zuständig. Mit diesen Aufgaben ist IPv6 der Schicht 3 des OSI-Schichtenmodells zugeordnet.

Die Aufgabe des Internet-Protokolls besteht im Wesentlichen darin, Datenpakete von einem System über verschiedene Netzwerke hinweg zu einem anderen System zu vermitteln (Routing).

IPv6 ist der direkte Nachfolger von IPv4 und Teil der Protokollfamilie TCP/IP. Seit Dezember 1998 steht IPv6 bereit und wurde hauptsächlich wegen der Adressknappheit und verschiedener Unzulänglichkeiten von IPv4 entwickelt spezifiziert. Da weltweit immer mehr Menschen, Maschinen und Geräte an das Internet mit einer eindeutigen Adresse angeschlossen werden sollen, reichen die 4 Milliarden IPv4-Adressen nicht mehr aus.

Warum IPv6?

IPv6 gilt als Wunderwaffe gegen so manche Probleme mit Netzwerkprotokollen und gleichzeitig wird es als Teufelszeug verdammt, das wieder neue unbekannte Probleme hervorruft. Eine Tatsache ist, dass Administratoren, Programmierer und Hersteller IPv6 neu lernen müssen. Viele Rezepte aus der IPv4-Welt taugen unter IPv6 nicht mehr. Erschwerend kommt hinzu, dass es bei IPv6 allen Beteiligten an Erfahrung fehlt. IPv6-Gurus, die man bei einem großen Problem befragen kann, gibt es nicht so viele.

Bei IPv6 ist das Ende-zu-Ende-Prinzip konsequent weiter gedacht. Ein Interface kann mehrere IPv6-Adressen haben und es gibt spezielle IPv6-Adressen, denen mehrere Interfaces zugeordnet sind.

IPv6 löst also nicht nur die Adressknappheit, sondern bietet auch Erleichterungen bei der Konfiguration und im Betrieb. Die zustandslose IPv6-Konfiguration und verbindungslokalen Adressen, die bereits nach dem Computerstart verfügbar sind, vereinfachen die Einrichtung und den Betrieb eines lokalen Netzwerks.

Damit das gelingt sind Planer und Errichter von IP-Netzen gefordert sich eine neue Denkweise anzueignen.

Internet Protocol Version 5 (IPv5)?

IPv5 hieß offiziell ST-2 (Internet Stream Protocol Version 2) und war ein experimentelles Protokoll für Echtzeit-Datenströme. ST-2 sollte ursprünglich Audio und Video per Multicast übertragen. Dadurch sollten die Bandbreitenreservierungsvorteile von ATM in die IP-Netze gelangen. Zur Serienreife hat es nicht gereicht. Deshalb gab es auch kein IPv5

im praktischen Einsatz. Und ST-2 wurde von RSVP (Resource Reservation Protocol) zur Bandbreitenanforderung bei Routern abgelöst.

Parallelbetrieb von IPv4 und IPv6 (Dual-Stack)

IPv4 hat keine Zukunft mehr und ein zügiger Wechsel zu IPv6 erscheint notwendig. Gleichzeitig muss nicht nur IPv6 eingeführt, sondern auch IPv4 parallel betrieben werden. Man bezeichnet diesen Betriebszustand als "Dual Stack".

Der Betrieb beider Protokolle muss erfolgen, bis alle Rechner auf der Welt IPv6 beherrschen. Und das kann dauern. Es gibt viele Netzwerk-Komponenten, die kein IPv6 unterstützen und erst gegen IPv6-fähige Komponenten ausgetauscht werden müssen. Auf der anderen Seite ist der Markt für IPv6 noch nicht groß genug, dass sich die Entwicklung von IPv4-vergleichbaren Produkten mit IPv6 lohnt.

Aber an IPv6 führt letztlich kein Weg vorbei. Sonst läuft man Gefahr den Anschluss an die technische Entwicklung zu verpassen.

Vorteile von IPv6

Für viele ist IPv6 einfach nur ein IPv4 mit längeren Adressen. Doch diese Ansicht ist völlig falsch. IPv6 ist ein Protokoll mit vielen neuen Funktionen. Die Erfahrungen, die jemand aus der IPv4-Welt mitbringt, lassen sich nur bedingt auf IPv6 übertragen.

- längere Adressen und dadurch ein größerer Adressraum
- mehrere IPv6-Adressen pro Host mit unterschiedlichen Gültigkeitsbereichen
- Autokonfiguration der IPv6-Adressen möglich
- Multicast durch spezielle Adressen
- schnelleres Routing
- Punkt-zu-Punkt-Verschlüsselung mit IPsec
- Quality of Service
- Datenpakete bis 4 GByte (Jumbograms)

IPv6-Adressen und Adressraum

Eine IPv6-Adresse besteht aus 128 Bit. Diese Adresslänge erlaubt eine unvorstellbare Menge von 2 hoch 128 oder 3,4 x 10 hoch 38 Adressen. Damit haben IPv6-Adressen genügend Raum, um möglichst viele Netzwerk-Topologien abbilden zu können. Gleichzeitig geht es auch darum, das Routing zu vereinfachen.

Wegen der unhandlichen Länge werden die 128 Bit der IPv6-Adresse in 8 Blöcke zu jeweils 16 Bit unterteilt. Je 4 Bit werden als eine hexadezimale Zahl dargestellt. Jeweils 4 Hexzahlen werden gruppiert und durch einen Doppelpunkt (":") getrennt.

Um die Schreibweise zu vereinfachen, können führende Nullen in den Blöcken wegfallen. Eine Folge von 8 oder mehr Nullen kann man einmalig durch zwei Doppelpunkte ("::") ersetzen.

Eine IPv6-Adresse besteht aus zwei Teilen. Dem Network Prefix (Präfix oder Netz-ID) und dem Interface Identifier (Suffix, IID oder EUI).

Der Network Prefix kennzeichnet das Netz, Subnetz bzw. den Adressbereich. Der Interface Identifier kennzeichnet einen Host in diesem Netz. Er wird aus der 48-Bit-MAC-Adresse des Interfaces gebildet und dabei in eine 64-Bit-Adresse umgewandelt. Es handelt sich dabei um das Modified-EUI-64-Format.

Auf diese Weise ist das Interface unabhängig vom Network Prefix eindeutig identifizierbar.

Die von IPv4 bekannte Netzwerkmaske bzw. Subnetzmaske fällt ersatzlos weg. Um den Adressbereich bzw. das Subnetz zu kennzeichnen wird ein Präfix verwendet, den man bedarfsweise durch ein "/" an die Adresse hängt.

In der Regel stehen die ersten 64 Bit einer IPv6-Adresse für das Netz und die restlichen 64 Bit für den Host. Somit beträgt die Präfix-Länge in der Regel immer "/64". Diese Regel gilt allerdings nicht in jedem Netzwerk.

Ein IPv6-Host hat nicht nur eine IPv6-Adresse, sondern in der Regel drei IPv6-Adressen. Eine link-lokale Adresse, eine globale Adresse und eine temporäre Adresse. Jede dieser Adressen hat einen anderen Gültigkeitsbereich. Die link-lokale Adresse ist nur im lokalen Netzwerk gültig. Sie entsteht ohne manuelle Konfiguration. Somit sind immer Verbindungen im lokalen Netzwerk möglich. Die link-lokale Adresse wird auch benötigt, um eine globale IPv6-Adresse zu bekommen, damit Verbindungen ins öffentliche Netz möglich sind.

IPv6-Autokonfiguration (SLAAC / DHCPv6)

IPv6 ermöglicht eine vollständige Autokonfiguration durch einen Host mit IPv6-Adresse, Standard-Gateway und DNS-Server. Hierbei muss man anmerken, dass ein IPv6-Host in der Regel mehrere IPv6-Adressen hat und diese und alle anderen Parameter für eine vollständige Autokonfiguration auf unterschiedlichen Wegen bekommen kann. Selbstverständlich ist auch eine manuelle, das heißt, statische IPv6-Konfiguration möglich.

Man unterscheidet zwischen einer "stateless" und einer "stateful" Autokonfiguration. Bei "stateless" erzeugt sich der IPv6-Host seine IP-Adresse selber. Bei "stateful" bekommt er sie zentral zugewiesen. Anders als bei IPv4 muss die IPv6-Konfiguration im lokalen Netzwerk nicht zentral vergeben werden. Die Einrichtung eines IP-Netzwerks ist unter IPv6 somit viel einfacher.

Dank der IPv6-Autokonfiguration sollte in jedem Fall eine Kommunikation im lokalen Netz möglich sein.

Privacy Extensions

Für den Interface Identifier der IPv6-Adresse wird typischerweise die MAC-Adresse des Netzwerk-Interfaces herangezogen. Da MAC-Adressen weltweit eindeutig sind, entsteht somit eine weltweit eindeutiger Interface Identifier.

Um die Bedenken bezüglich Datenschutz und Privatsphäre zu zerstreuen hat man Privacy Extensions eingeführt. Privacy Extensions erzeugen regelmäßig einen zufälligen Interface Identifier, der keinen Rückschluss auf die MAC-Adresse und damit den Host zulässt.

Address Selection

Address Selection ist ein Verfahren, welches darüber entscheidet, welche IP-Adresse verwendet wird. Wenn ein Host sowohl eine IPv4- als auch eine IPv6-Adresse hat (Dual Stack), dann stellt sich die Frage, welche er verwendet? Und wenn ein Host eine IPv6-Adresse verwendet, welche davon? Die link-lokale, die globale oder eine temporäre IPv6-Adresse?

Multicast

IPv6 fasst Netzwerkknoten, Router, Zeit-Server und andere Dienste bzw. Dienste-Anbieter in Multicast-Gruppen zusammen. Jede Gruppe ist über eine eigene Adresse erreichbar. Das bedeutet, man kann in einem lokalen Netzwerk einen zentralen Dienst ansprechen, ohne die IPv6-Adresse des Hosts zu wissen. Ein beliebiger Host kann sich einer Multicast-Gruppe zugewiesen fühlen und Pakete an eine Multicast-Adresse verarbeiten.

NDP und ICMPv6

Neighbor Discovery Protocol, kurz NDP, ist das IPv6-Protokoll zum Austausch link-lokal relevanter Nachrichten wie Router Discovery und Neighbor Discovery. Die Übertragung der NDP-Nachrichten erfolgt mit ICMPv6.

IPv6-Header und Extension Header

Jedes IPv6-Datenpaket besteht aus einem Header (Kopf) und dem Payload (Nutzdaten). Der Header ist den Nutzdaten vorangestellt. Der IPv6-Header enthält unter anderem die IP-Adresse von Sender und Empfänger und weitere Angaben, die für das IP-Routing wichtig sind und von den Routern auf dem Weg von Quelle zum Ziel ausgewertet werden.

Der IPv6-Header weist eine feste Länge von 40 Byte auf. Optionale Informationen sind in den Extension-Header ausgelagert.

IPsec

IPsec ist eine Erweiterung des Internet-Protokolls (IP) um Verschlüsselungs- und Authentifizierungsmechanismen. Damit erhält das Internet-Protokoll die Fähigkeit IP-Pakete kryptografisch gesichert über öffentliche und unsichere Netze zu transportieren.

IPsec wurde von der Internet Engineering Task Force (IETF) als Bestandteil von IPv6 entwickelt und später auch für IPv4 spezifiziert.

Multihoming

Ein an einem lokalen Netzwerk angeschlossenes Interface gilt dann als "multihomed", wenn es mehrere globale IPv6-Adressen hat, die unterschiedliche Präfixe aufweisen. Das heißt, dass das lokale Netzwerk über mehrere ISPs an das Internet angebunden ist. Dabei haben die Interfaces Adressen von jedem beteiligten ISP. Dies dient nicht nur der Redundanz. Es kann auch durch die Wahl der Absenderadresse das zum Übertragen verwendete Netz bestimmt werden.

Renumbering

Die Mechanismen zur "stateless" Autokonfiguration erlauben das Hinzufügen und Entfernen von globalen Präfixen und somit die Rekonfiguration eines Netzwerks im laufenden Betrieb.

Dank Renumbering lässt sich ein Interface relativ einfach mit neuen Adressen bestücken. Sei es um ein neues Adressschema einzuführen oder den Provider zu wechseln. Ein Interface wird in einen multihomed-ähnlichen Zustand gebracht. Gleichzeitig lässt man die Gültigkeit der alten Adressen langsam auslaufen.

Dazu kann man mehrere Netzzugangsrouter unterschiedlich Konfigurieren. Über Router Advertisements kann ein den Hosts sagen "priorisiere mich" und ein anderer Router "benutze mich nicht". Auf diese Weise kann man einen neuen Router in Betrieb und einen anderen außer Betrieb nehmen. Kleine SoHo-Router können das natürlich nicht.

Flow Labels

Flow Labels sind Kennzeichnungen für IP-Pakete, anhand derer Router oder Paketfilter, Pakete unterschiedlich behandeln können. Nach welchen Kriterien ein Flow Label vergeben wird, ist im Einzelfall festzulegen. Idealerweise sollten alle Pakete mit gleichem Flow Label auch gleich behandelt werden.

Flow Labels ersetzen kein MPLS und für QoS gibt es ein eigenes Feld. Ob es in der Zukunft eine Rolle spielt, das muss sich erst noch zeigen.

Mobile IPv6

Mobile IPv6 erlaubt es zwischen verschiedenen Netzen umher zuwandern, ohne dabei die Verbindung auf IP-Ebene zu verlieren. Es geht darum, unterbrechungsfrei zu

kommunizieren, auch wenn man das Netz und damit der Präfix wechselt. Es ist eine Art Handover auf der IP-Ebene.

Jumbograms

Die Nutzdatenlänge wird im Header vermerkt, das entsprechende Feld kann aber keine Werte jenseits von 65.535 annehmen. Mit der Jumbo Payload Option im zugehörigen Extension Header sind auch Nutzdaten bis knapp unter 4 Gigabyte möglich. Das setzt allerdings voraus, dass man Path-MTU-Pakete dieser Größe zulässt. Es bringt nichts große Paketlängen zu definieren, wenn die zu übertragenden Router deswegen einen Fehler zurückmelden. Jumbograms sind also nur für spezielle Anwendungen und unter Umständen nur in lokalen Netzwerken sinnvoll.

Übergangsverfahren von IPv4 auf IPv6

Mit der praktischen Umsetzung von IPv4 auf IPv6 hapert es, weil es unmöglich ist, alle Netzwerk-Geräte auf einmal IPv6-fähig zu machen. Damit der Wechsel leichter geht und Investitionen in alte IPv4-Technik nicht obsolet werden, gibt es verschiedene Verfahren, die den Übergang von IPv4 nach IPv6 erleichtern sollen.

Umstellung auf IPv6: Ja oder nein?

Sagen wir es mal ehrlich. IPv6 ist kein Hype, wie andere kommerziell getriebenen IT-Themen. Die Einführung von IPv6 ist eine Infrastruktur-Maßnahme, die im ersten Moment wenig Vorteile bietet, Mehraufwand bedeutet und in der Regel mit zusätzlichen Kosten verbunden ist.

Für normale Anwender ändert sich im ersten Moment nicht viel, weshalb IPv6 auch nicht Anwender-getrieben ist. Spannend wird IPv6 für den normalen Anwender erst, wenn es sinnvolle Anwendungen gibt, um Geräte zu Hause von unterwegs zu erreichen. Das ist jetzt noch nicht richtig möglich.

Für große Netze ist IPv6 eine Erlösung. Vor allem dort, wo selbst das große private 10/8er-Netz mehrfach verwendet wird, weil die Firmen-internen Strukturen sich nicht mit einem einzigen 10/8er-Netz abbilden lassen.

Es gibt keinen Grund IPv6 nicht zu nutzen. Ein Hemmnis sind häufig paranoide Administratoren, die jedes Bit unter Kontrolle behalten wollen und es als ihr Aufgabe ansehen Excel-Sheets mit IP-Adressen zu pflegen. Doch das klappt alles nicht. Es gibt keine praktikable Möglichkeit IPv6-Adressen für alle möglichen Devices sinnvoll zu verwalten. Das ist auch gar nicht notwendig.

Wer kein IPv6 anbietet, der muss damit rechnen, dass er irgendwann plötzlich von seinen Kunden abgehängt ist. Und dann ist die Frage, bis wann man IPv6 stabil umgesetzt bekommt.

Deshalb sollte man IPv6 umsetzen, solange man sich noch (zu korrigierende) Probleme bei der Umsetzung leisten kann, weil IPv4 als funktionierendes Fallback noch zur Verfügung steht. Wenn IPv6 erst einmal zwingend benötigt wird, kann man sich bei einer Umstellung keine Fehler mehr erlauben. Und welche Umstellung in der IT verläuft schon fehlerfrei?

IPv6-Adressen

Eine IPv6-Adresse ist eine Netzwerk-Adresse, die einen Host eindeutig innerhalb eines IPv6-Netzwerks logisch adressiert. Die Adresse wird auf IP- bzw. Vermittlungsebene (des OSI-Schichtenmodells) benötigt, um Datenpakete verschicken und zustellen zu können. Im Gegensatz zu anderen Adressen hat ein IPv6-Host mehrere IPv6-Adressen, die unterschiedliche Gültigkeitsbereiche haben.

Konkret bedeutet das, dass wenn von IPv6-Adressen die Rede ist, dass nicht immer klar ist, welchen Gültigkeitsbereich diese IPv6-Adressen aufweisen. Grob unterscheidet man zwischen verbindungslokalen und globalen IPv6-Adressen. Die verbindungslokale IPv6-Adresse ist nur im lokalen Netzwerk gültig und wird nicht geroutet. Die globale IPv6-Adresse ist über das lokale Netzwerk hinaus im Internet gültig.

Eine IPv6-Adresse hat eine Länge von 128 Bit. Diese Adresslänge erlaubt eine unvorstellbare Menge von 2128 oder 3,4 x 1038 IPv6-Adressen. Das sind 340.282.366.900.000.000.000.000.000.000.000.000.000 IPv6-Adressen, also rund 340 Sextillionen Adressen. Bei IPv4 spricht man von rund 4,3 Milliarden Adressen.

Der Adressraum von IPv6 reicht aus, um umgerechnet jeden Quadratmillimeter der Erdoberfläche inklusive der Ozeane mit rund 600 Billiarden Adressen zu pflastern. Weil man mit dieser großen Menge an Adressen verschwenderisch umgehen darf, spart man sich eine aufwendige Verwaltung, wie es bei IPv4-Adressen notwendig ist.

Segmentierung

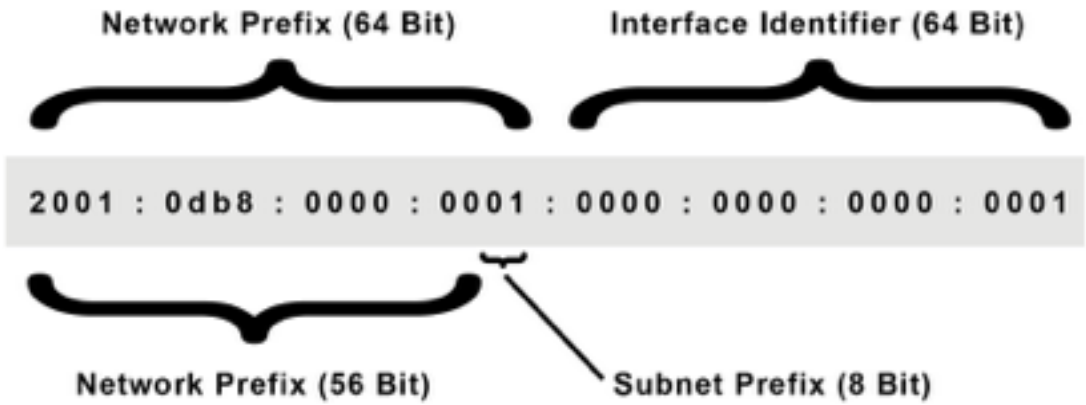
Einer der Gründe für den Wechsel von IPv4 auf IPv6 ist der größere Adressbereich von IPv6. Doch warum gleich 128 Bit Adressbreite? Der Grund ist der, dass die IP-Adressen lang genug sein sollten, um den gesamten Adressraum großzügig segmentieren bzw. aufteilen zu können. Es sollen möglichst alle Netzwerk-Topologien berücksichtigt werden können. Gleichzeitig soll das Routing vereinfacht werden.

Damit Router effizient arbeiten können, müssen Adressen hierarchisch strukturiert vergeben werden. Damit alle Ebenen der Hierarchie abgebildet werden können, muss die IP-Adresse lang genug sein. Wünschenswert wäre es, wenn dann auch noch genug Raum für zukünftige Entwicklungen übrig bleibt. Deshalb akzeptiert man bei der Segmentierung von IPv6-Adressen auch einen relativ großen Verschnitt.

IPv6-Adresse im Detail

Eine IPv6-Adresse besteht aus 128 Bit. Wegen der unhandlichen Länge werden die 128 Bit in 8 mal 16 Bit unterteilt. Je 4 Bit werden als eine hexadezimale Zahl dargestellt. Je 4 Hexzahlen werden gruppiert und durch einen Doppelpunkt (":") getrennt. Um die Schreibweise zu vereinfachen lässt man führende Nullen in den Blöcken weg. Eine Folge von 8 Nullen kann man durch zwei Doppelpunkte ("::") ersetzen.

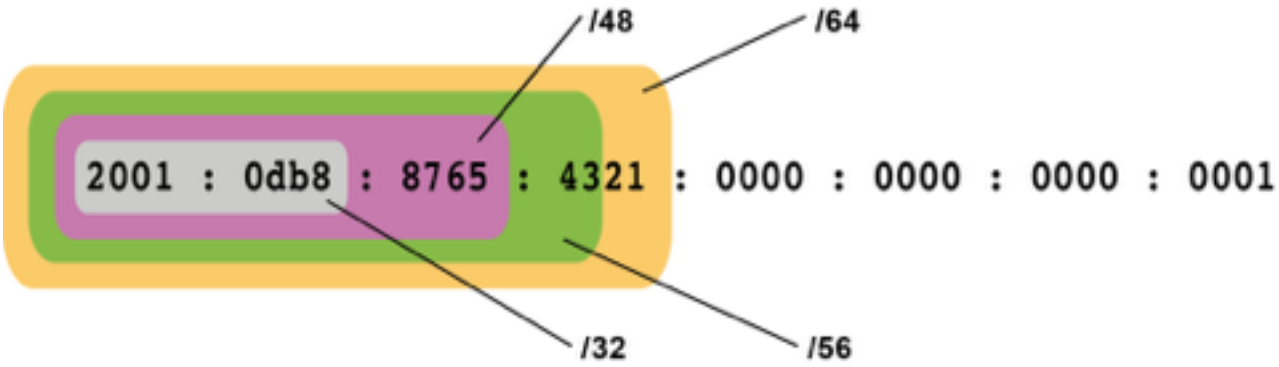
Eine IPv6-Adresse besteht aus zwei Teilen. Dem Network Prefix (Präfix oder Netz-ID) und dem Interface Identifier (Suffix, IID oder EUI).



Der Network Prefix kennzeichnet das Netz, Subnetz bzw. Adressbereich. Der Interface Identifier kennzeichnet einen Host in diesem Netz. Er wird aus der 48-Bit-MAC-Adresse des Interfaces gebildet und dabei in eine 64-Bit-Adresse umgewandelt. Es handelt sich dabei um das Modified-EUI-64-Format.

Auf diese Weise ist das Interface unabhängig vom Network Prefix eindeutig identifizierbar.

Segmentierung: Präfix und Präfixlänge



Die von IPv4 bekannte Netzmaske bzw. Subnetzmaske fällt bei IPv6 ersatzlos weg. Um trotzdem eine Segmentierung und Aufteilung von Adressbereichen bzw. Subnetzen vornehmen zu können, wird die Präfixlänge definiert und mit einem "/" (Slash) an die eigentliche IPv6-Adresse angehängt. Der hierarchische Aufbau des Präfix soll das Routing mit IPv6 vereinfachen.

Standardmäßig ist "/64" die Präfixlänge. Es gibt jedoch weitere typische Präfixe, die 32, 48 und 56 Bit lang sind. Das hat etwas mit der Zuteilung von Präfixen zu tun. Wer eigene Netze betreiben möchte, der bekommt von seinem Provider einen kürzeren Präfix als /64 und erhält damit mehr Adressraum.

Das bedeutet, dass jedes noch so kleine Netzwerk mindestens ein Subnetz zugewiesen bekommt. In diesem Subnetz können jeweils gigantische 264, also über 18 Trillionen

Einzeladressen vergeben werden. Das bedeutet, dass die Anwender sich den Einsatz von privaten IP-Adressen und Verfahren wie NAT sparen können. Der Adressüberfluss von IPv6 macht es möglich.

Mit IPv6 lassen sich Altlasten in der Netzaufteilung beseitigen und Dank des großen Adressraums den IPv6-Adressplan großzügig neu gestalten. Da jeder Host mehrere IPv6-Adressen haben kann, wäre es denkbar, dass jeder Dienst oder jede Anwendung auf einem Server eine eigene IPv6-Adresse bekommt. Innerhalb desselben Subnetzes kann ein Dienst dann beliebig auf eine andere Hardware wechseln, ohne dass sich die IPv6-Adresse des Dienstes ändern muss.

Hinweis: Die IPv6-Autokonfiguration funktionieren nicht mit weniger als 64 Bit im Interface Identifier. Das heißt natürlich nicht, dass es nicht doch jemand versucht. Aber dann gibt es zum Beispiel Probleme beim Generieren der globalen IPv6-Adresse, weil dieser Mechanismus davon ausgeht, dass er 64 Bit selber zuteilen darf. Wenn die Mechanismen der Autokonfiguration nicht mehr funktionieren, muss man IPv6-Adressen von Hand konfigurieren oder per DHCPv6 zuteilen. Erfahrungsgemäß ist es keine gute Idee damit zu experimentieren.

Adressvergabe durch IPv6-Provider (Zuteilung des Präfixes)

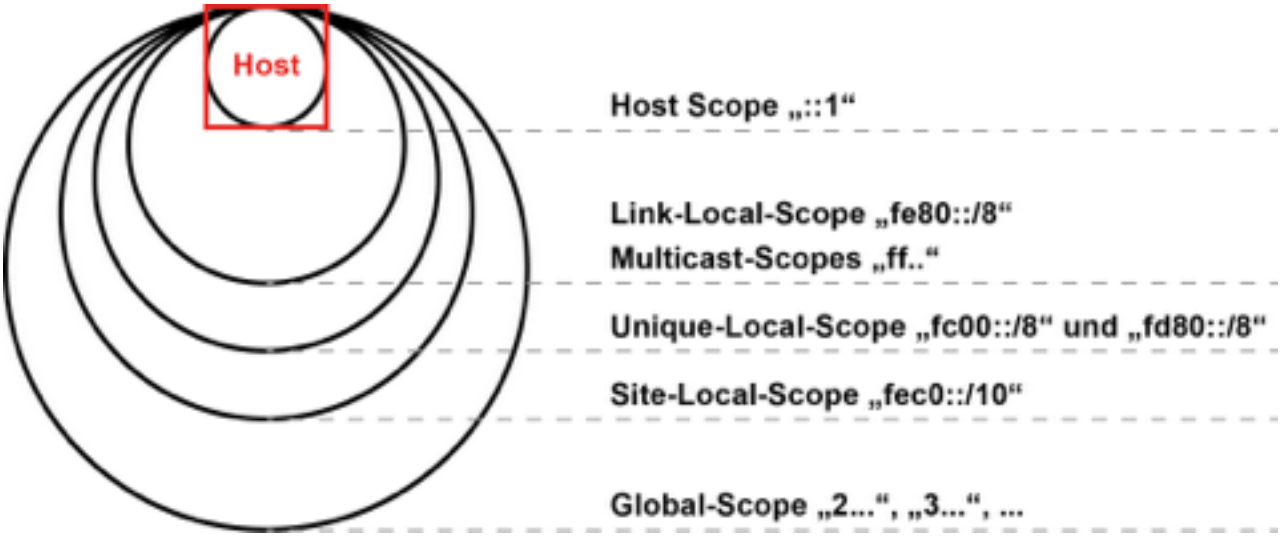
Der ursprüngliche Plan zur Aufteilung des Adressraums war, dass jeder Kunde ein /48er-Netz bekommen sollte. Dass das zu großzügig ist, hat man schnell erkannt und ist deshalb zu längeren Präfixen übergegangen. Entweder /56 oder /64. /56 sollte normal sein, weil man davon ausgehen muss, dass ein Kunden mehrere Netze betreibt. Unter Umständen auch im Heimbereich. Insbesondere kleine Unternehmen haben dann mehr Spielraum, ohne Einschränkungen hinnehmen zu müssen.

Enterprise-Kunden, die eigene Netze betreiben, bekommen von ihrem Provider in der Regel /48-Netze. Große Netzbetreiber und Provider bekommen generell /32er-Netze zugeteilt. Größere Netzbetreiber bekommen auch noch größere Netze.

Gültigkeitsbereiche

Address Scopes

IPv6 unterscheidet sich von IPv4 nicht nur durch längere Adressen, sondern durch Gültigkeitsbereiche (Scope). Jede IPv6-Adresse hat einen sogenannten Scope oder Gültigkeitsbereich. Der Scope ist der Teil eines Netzwerks in dem die zugehörige Adresse als gültig anerkannt und geroutet wird.



Während man bei IPv4 nur zwischen privaten und öffentlichen Adressen unterscheidet, können IPv6-Adressen vielschichtiger sein. Dazu muss man auch noch wissen, dass große Teile des IPv6-Adressraums für die normale Nutzung ausgenommen sind und beispielsweise ganze Gruppen von Empfängern (Multicast) adressieren.

- Host-Scope
- Link-Local-Scope
- Unique-Local-Scope
- Site-Local-Scope (veraltet)
- Global-Scope
- Multicast

Die beiden wichtigsten Scopes sind der Link-Loal-Scope und der Global-Scope. IPv6-Pakete mit einer globalen Zieladresse werden außerhalb des lokalen Netzwerks geroutet. Link-lokale Adressen sind nur innerhalb des lokalen Netzwerks gültig und werden auch nur dann verwendet, wenn das Ziel link-local ist.

Hinter dem Link-Local-Scope stecken Mechanismen wie Neighbor Discovery, dass das Address Resolution Protocol (ARP) ablöst oder die Stateless Address Autoconfiguration

(SLAAC) als Alternative zu DHCP. Neighbor Discovery zeichnet sich vor allem durch seine Unabhängigkeit von der darunterliegenden Übertragungstechnik (OSI-Schicht 1 und 2) aus.

Loopback Address (Host Scope)

Wie bei IPv4 gibt es bei IPv6 einen "Localhost". Es handelt sich dabei um eine Loopback-Adresse mit "::1". Unter IPv4 stand mit 127.0.0.1/8 noch ein beachtlich großes Netz zur Verfügung.

LLA - Link-Local Address (Link-Local-Scope)

Für die IPv6-Connectivity ist es wichtig, dass ein IPv6-Host beim Start eine eigene IPv6-Adresse hat. Dazu reicht es aus, wenn diese nur im lokalen Netzwerk gültig ist. Diese Link-Local-Address (LLA) bzw. verbindungslokale Adresse hat immer den Präfix "fe80::/64" und gilt nur für das jeweilige Netzwerksegment. In der Regel reicht der Link-Local-Scope bis zum nächsten Router. Link-lokal bedeutet, dass es sich um eine direkte Verbindung handelt, also direkt auf Schicht 1 und 2. In einem typischen lokalen Netzwerk per Ethernet oder WLAN.

Jedes Interface muss eine link-lokale bzw. verbindungslokale IPv6-Adresse haben, weil es bei IPv6 kein Broadcasting mehr gibt. Die link-lokale Adresse wird für die lokale Kommunikation innerhalb des Netzwerksegments gebraucht. Zum Beispiel für die Kommunikation mit dem Standard-Gateway, um sich selber eine global gültige IPv6-Adresse zu generieren oder per DHCPv6 besorgen zu können.

Auch bei IPv4 gibt es link-lokale Adressen. Aus dem Adressbereich 169.254.0.0/16. Diese Adressen werden aber nur dann per APIPA oder Zeroconf konfiguriert, wenn die IP-Konfiguration per DHCP nicht erfolgreich war.

ULA - Unique-Local Address (RFC 4193)

Für private lokale Netze gibt es in IPv6 reservierte Adressbereiche (Unique Local Adressses, ULA). Sie haben eine ähnliche Funktion, wie die privaten Adressen von IPv4. Die ULA-IPv6-Adressen befinden sich im Adressbereich "fc00::/7" (fc00... bis fdff...) und werden nicht im Internet geroutet.

Unterscheiden muss man zwischen dem Präfix "fc" und "fd". ULAs mit dem Präfix "fd" sind für lokal generierten Adressen vorgesehen und sind zufällig zu generieren und somit nur sehr wahrscheinlich eindeutig. Wenn man private Netze mit diesen Adressen zusammenführen will, dann können Adresskonflikte auftreten. Nur ULAs mit dem Präfix "fc" sind weltweit eindeutig und deshalb für global zugewiesene, eindeutige ULAs reserviert. Hier treten bei Netzzusammenführungen keine Adresskonflikte auf.

- unique local (zentral verwaltet): Diese Adressen werden vom Provider vergeben (fc00::/8).

- unique local (lokal verwaltet): Diese Adressen können im eigenen lokalen Netzwerk verwendet werden (fd80::/8).

Site-Local Address (Site-Local-Scope)

Site-lokale Adressen aus dem Adressbereich "fec0::/10" sind ähnlich wie die lokalen Unicast-Adressen nur in der aktuellen Site gültig und werden von Routern nicht nach außen geroutet oder von außen akzeptiert. Mit diesem Scope-Typ kann man private Netze unter IPv6 nachbilden.

Der Site-Local-Scope ist aber nicht genau definiert. Seit September 2004 gilt dieses Konzept als veraltet.

Global Address (Global-Scope)

Für eine Verbindung ins Internet benötigt ein Host eine Global Unique Address bzw. global routbare Adresse. Die globale IPv6-Adresse, von der ein Host mehrere haben kann, bezieht ein Host per Autokonfiguration. Hier gibt es die beiden Verfahren Stateless Address Autoconfiguration (SLAAC) und Stateful Address Autoconfiguration (DHCPv6).

Multicast Address (Multicast Scopes)

IPv6 fasst Netzwerkknoten, Router, Zeit-Server und andere Dienste bzw. Dienst-Anbieter in Multicast-Gruppen zusammen. Jede Gruppe ist über eine eigene Adresse erreichbar. Die zugewiesenen Hosts fühlen sich alle angesprochen, wenn die Gruppe in einem Paket per Multicast adressiert wird.

Lokales Netzwerk vs. Link-Local-Scope

Die Definition von Gültigkeitsbereichen (Scopes) ist einer der größten Unterschiede zwischen IPv4 und IPv6. Bezüglich der IPv6-Scopes gibt es in IPv4 nichts vergleichbares. Trotzdem gibt es manchmal den Vergleich zwischen den privaten IPv4-Adressen und den link-lokalen IPv6-Adressen. Doch das ist nicht das gleiche.

Wenn man vom Link-Local-Scope im Zusammenhang mit Neighbor Discovery oder Router Discovery spricht, darf man nicht vom lokalen Netzwerk sprechen. Denn in einem lokalen Netzwerk, im Sinne eines LAN, kann es auch mehrere Link-Local-Scopes geben. Beispielsweise wenn ein LAN durch IPv6-Router in mehrere Link-Local-Scopes geteilt ist.

In einem kleinen LAN, zum Beispiel in Privathaushalten, wo es nur einen Router gibt, da entspricht das lokale Netzwerk (LAN) dem Link-Local-Scope. In einem Unternehmensnetzwerk, bei dem das "LAN" aus mehreren lokalen Netzwerken besteht, kann es mehrere Link-Local-Scopes geben, die nur bis zur entsprechenden "link-lokalen" Netzgrenze reichen. Hier befindet sich ein Router, der Datenpakete mit link-lokalem Bezug nicht über die Netzgrenze hinweg weiterleitet. Gemeint ist, dass ein lokales Netzwerk größer sein kann als der Link-Local-Scope reicht.

Notation von IPv6-Adressen

Schreibweise und Kürzungsregeln

IPv6-Adressen bestehen aus insgesamt 128 Bit woraus sich eine Menge von 2128 möglichen Adressen ergibt. Die vorderen 64 Bit sind der Präfix bzw. Network-ID. Vereinfacht ausgedrückt, ist das die IPv6-Adresse des Subnetzes, in dem sich ein Host befindet.

Die hinteren 64 Bit werden als Interface Identifier (IID) bezeichnet. Das ist der Host-Adressanteil einer IPv6-Adresse.

Wegen der unhandlichen Länge werden die 128 Bit in 8 mal 16 Bit (2 Byte) unterteilt. Je 4 Bit werden als eine hexadezimale Zahl dargestellt. Je 4 Hexzahlen werden gruppiert und durch einen Doppelpunkt (":") voneinander getrennt.

Uneinheitlichkeit der Notation

Weil IPv6-Adressen sehr lang sein können, werden sie in der Regel gekürzt. Leider ergeben sich dabei viele unterschiedliche Schreibweisen. Die folgenden Schreibweisen sind Repräsentationen der gleichen IPv6-Adresse.

2001 : 0db8 : 0000 : 0000 : 0001 : 0000 : 0000 : 0001

2001 : db8 : 0 : 0 : 1 : 0 : 0 : 1

2001 : 0db8 : 0000 : 000 : 1 : 00 : 0 : 1

2001 : db8 : : 0 : 1 : 0 : 0 : 1

2001 : db8 : 0 : 0 : 1 : : 1

2001 : DB8 : 0 : 0 : 1 : : 1

2001 : db8 : : 1 : 0 : 0 : 1

Die einzige korrekte Schreibweise nach den verbindlichen Notationsregeln in RFC 5952 ist die letzte.

Verbindliche Notationsregeln

Um zu viele unterschiedliche Schreibweise zu vermeiden wurden folgende verbindliche Notationsregeln in RFC 5952 definiert:

1. Alle alphabetischen Zeichen werden grundsätzlich klein geschrieben.
2. Alle führenden Nullen eines Blocks werden grundsätzlich weggelassen.
3. Ein einzelner 4er Nullerblock wird zu einer "0" zusammengefasst.
4. Aufeinanderfolgende 4er Nullerblöcke werden durch zwei Doppelpunkte ("::") gekürzt.

5. Sind mehrere gleichwertige Kürzungen möglich, ist die erste von Links beginnend zu wählen.

Es wird empfohlen diese Notationsregeln einzuhalten, um Fehler und Fehlinterpretationen aufgrund unterschiedlicher Schreibweise zu vermeiden.

> Beispiel: 2001 : 0000 : 0000 : 0000 : 0001 : 0000 : 0000 : 0001
>
> Falsche Kürzung: 2001 : : 1 : : 1
>
> Richtige Kürzung: 2001 : : 1 : 0 : 0 : 1

Schreibweise der IPv6-Adresse in der URL

In URLs sind IPv6-Adressen problematisch, weil in URLs der Doppelpunkt (":") als Trennzeichen zwischen Host-Adresse (Domain-Name oder IP-Adresse) und Portnummer (optional) verwendet wird. Deshalb müssen IPv6-Adressen in eckige Klammern gesetzt werden ("[]"), wenn sie in URLs verwendet werden. Die Portnummer muss hinter der schließenden Klammer, mit einem Doppelpunkt abgetrennt, stehen.

> Beispiel: http://[2001:db8::1%25eth0]/

Weiterhin dient in URLs das Prozentzeichen (%) für die Kennzeichnung der hexadezimalen Zeichencodierung. Innerhalb der URL muss das Prozentzeichen durch seinen eigenen Hex-Code "%25" ersetzt werden (RFC 6874). Das ist dann notwendig, wenn man die Verbindung über eine bestimmte Schnittstelle erzwingen will.

Schreibweise der IPv6-Adresse im UNC-Pfad

UNC-Pfade dürfen nach der Definition von Microsoft keine Doppelpunkte enthalten. Für die Schreibweise von IPv6-Adressen ergeben sich dadurch Schwierigkeiten. Hier sind Doppelpunkte (":") als Trennzeichen vorgesehen. Um das Problem zu umgehen hat Microsoft einen Workaround definiert.

> Normale UNC-Schreibweise: \\2001:db8::1\share
>
> Angepasste UNC-Schreibweise: \\2001-db8--1.ipv6-literal.net\share

Vergabe von IPv6-Adressen

Verteilung des globalen Präfix

IP-Adressen werden von der Internet Assigned Numbers Authority (IANA) verwaltet und an regionale Registries, die Regional Internet Registries (RIR), vergeben. Davon gibt es fünf weltweit. Das RIPE NCC ist beispielsweise das RIR für Europa, Zentralasien und den Nahen Osten. Die RIRs verteilen IP-Adressen an Local Internet Registries (LIR). Von denen bekommen Endkunden ihre IP-Adressen.

Adressvergabe durch den Adressverwalter (RIR)

Die Organisation IANA ist der Verwalter der IPv6-Adressblöcke. Sie übergibt die Adressblöcke an regionale Adressverwalter (RIR). In Europa ist dass das RIPE.

Einen /32-Präfix bekommt heute jedes RIPE-Mitglied ohne Weiteres. Damit lassen sich 4.294.967.296 IPv6-Subnetze bilden (32 Bit Subnet Prefix), die jeweils 18.446.744.073.709.551.616 IPv6-Adressen enthalten (64 Bit Interface Identifier). Einen größeren Bedarf muss man gegenüber dem RIPE ausreichend begründen. Dann kann man auch einen /29-Präfix bekommen, mit dem sich 34.359.738.368 IPv6-Subnetze bilden lassen.

Aufgrund des großen Adressraums von IPv6 ist man in der Lage Präfixe möglichst großzügig zu vergeben. Das verbraucht zwar den Adressraum sehr schnell. Allerdings hält eine großzügige Präfix-Vergabe auch die IPv6-Routing-Tabellen klein.

Adressvergabe durch IPv6-Provider

Der ursprüngliche Plan zur Aufteilung des Adressraums war, dass jeder Kunde ein /48er-Netz bekommen sollte. Dass das zu großzügig ist, hat man schnell erkannt und ist deshalb zu längeren Präfixen übergegangen. Entweder /56 oder /64. /56 sollte normal sein, weil man davon ausgehen muss, dass ein Kunde mehrere Netze betreibt. Unter Umständen auch im Heimbereich. Insbesondere kleine Unternehmen haben dann mehr Spielraum, ohne Einschränkungen hinnehmen zu müssen.

Business-Kunden, die eigene Netze betreiben, bekommen von ihrem Provider in der Regel /48-Netze. Unter Umständen auch mehrere für mehrere Standorte oder Filialen.

Warum werden Präfixe so großzügig verteilt?

Dabei stellt man sich die Frage, warum gibt man so große Adressbereiche so großzügig heraus? Es ist doch ausgeschlossen, dass jemand soviel wirklich benötigt.

Die Idee dahinter ist, dass es einfacher ist, wenn alle IPv6-Umgebungen ähnlich oder fast gleich aussehen. So kann man zum Beispiel viel einfacher Adressen generieren, die eindeutig und global gültig sind.

Desweiteren hat man sich überlegt, dass es aus Sicht des Netzes egal sein sollte, wie die Hierarchie beim Betreiber des Endkunden-Netzes aussieht. Im Prinzip schert man alle Kunden über einen Kamm und teilt ihnen immer einen fast gleich großen Adressraum zu. Unabhängig davon, ob die das brauchen oder nicht. Eine Firma mit mehreren tausend Mitarbeitern bekommt einen fast gleich großen Adressraum, wie die Oma, die vielleicht nur ein Tablet in ihrem Netz betreibt.

Ein weitere Grund ist, dass man für jeden von der IANA bzw. den RIRs zugeteilten Präfixe einen Eintrag in der globalen Routing-Tabelle braucht. Größere Blöcke (bzw. kürzere Präfixe) bedeuten weniger Einträge und damit kleinere Routing-Tabellen. Deshalb findet auch eine relativ großzügige Zuteilung von 32er, 48er und 56er Präfixen statt.

Statischer oder dynamischer Präfix

Typische Endkunden-Internet-Anschlüsse, die von Internet-Service-Providern angeboten werden, bekommen keinen festen IPv6-Präfix, sondern einen dynamischen Präfix, der sich regelmäßig ändert. Das ist wie bei der dynamischen IPv4-Adresse.

Wenn das eigene Netz von außen erreichbar sein soll, dann ist eine statische IP-Adresse bzw. ein statischer Präfix notwendig. Doch in der Regel ist es so, dass man nur dann einen statischen Präfix bekommt, wenn man dafür bezahlt.

Die Fragestellung, wie man sicherstellen kann, dass der eigene Rechner immer unter der selben IPv6-Adresse erreichbar ist, bleibt also erhalten. Denn wenn der Präfix dynamisch vergeben wird, dann hat man immer auch eine andere IPv6-Adresse.

IPv6-Autokonfiguration

Ein IPv6-Host kann mehrere IPv6-Adressen haben. Wenn IPv6 im Host aktiviert ist, dann hat er zumindest eine link-lokale bzw. verbindungslokale Adresse. Wenn zusätzlich der Netzzugang und der Netzzugangsrouter IPv6-fähig sind, dann hat ein Host noch eine zweite IPv6-Adresse. Das ist die globale Adresse. Wenn Privacy Extensions im Host aktiviert ist, dann hat er noch zusätzlich eine temporäre globale Adresse, die für externe Verbindungen genutzt wird. Da temporäre Adressen irgendwann ihre Gültigkeit verlieren, kann ein Host auch mehrere temporäre Adressen haben.

- link-lokale IPv6-Adresse
- globale IPv6-Adresse
- (mehrere) temporäre globale IPv6-Adressen

Zu einer vollständigen IPv6-Konfiguration gehören aber nicht nur die IPv6-Adressen des Hosts, sondern mindestens noch die IPv6-Adressen des Standard-Gateways und eines DNS-Servers. Weitere Bestandteile der IPv6-Konfiguration sind netzabhängig und werden hier nicht berücksichtigt.

- Standard-Gateway
- DNS-Server

Die Fragestellung ist, wie kommt ein IPv6-Host an seine IPv6-Adressen und die anderen Teile der IPv6-Konfiguration, wie zum Beispiel das Standard-Gateway und der DNS-Server?

Woher bekommt ein Host seine link-lokale IPv6-Adresse?

Standardmäßig ist es so, dass wenn ein IPv6-Client gestartet wird, dann weist er sich selber eine link-lokale IPv6-Adresse zu. Verbindungen in andere Netze, zum Beispiel ins Internet, sind mit dieser link-lokalen IPv6-Adresse nicht möglich. Sie ist nur im lokalen Netz gültig. Vergleichbar, aber nicht ganz identisch, mit einer privaten IPv4-Adresse.

Die ersten 64 Bit einer link-lokalen IPv6-Adresse sind fest vorgegeben. Davon sind die ersten 16 Bit "fe80". Weitere 48 Bit werden mit Nullen aufgefüllt. Die restlichen 64 Bit der IPv6-Adresse entsprechen dem Interface Identifier für den die MAC-Adresse des Netzwerkadapters herangezogen wird. Dabei wird die 48-Bit-MAC-Adresse in der Mitte mit einem "ff:fe" auffüllt, damit eine Länge von 64 Bit entsteht. Zusätzlich wird das zweite Bit im ersten Byte invertiert.

Dieses Verfahren gehört zur Stateless Address Autoconfiguration (SLAAC).

Woher bekommt ein Host seine globale IPv6-Adresse?

Die globale IPv6-Adresse ist mit einer öffentlichen IPv4-Adresse vergleichbar, weil ein Host nur damit über das lokale Netz hinaus eine Verbindungen ins Internet aufbauen kann. Um eine globale IPv6-Adresse zu bekommen, benötigt der IPv6-Host die link-lokale IPv6-Adresse. Der hintere Teil der Adresse besteht aus dem Interface-Identifier und ist somit bei beiden Adressen gleich. Nur der vordere Teil, der Präfix, der muss für die globale Adresse noch ermittelt werden.

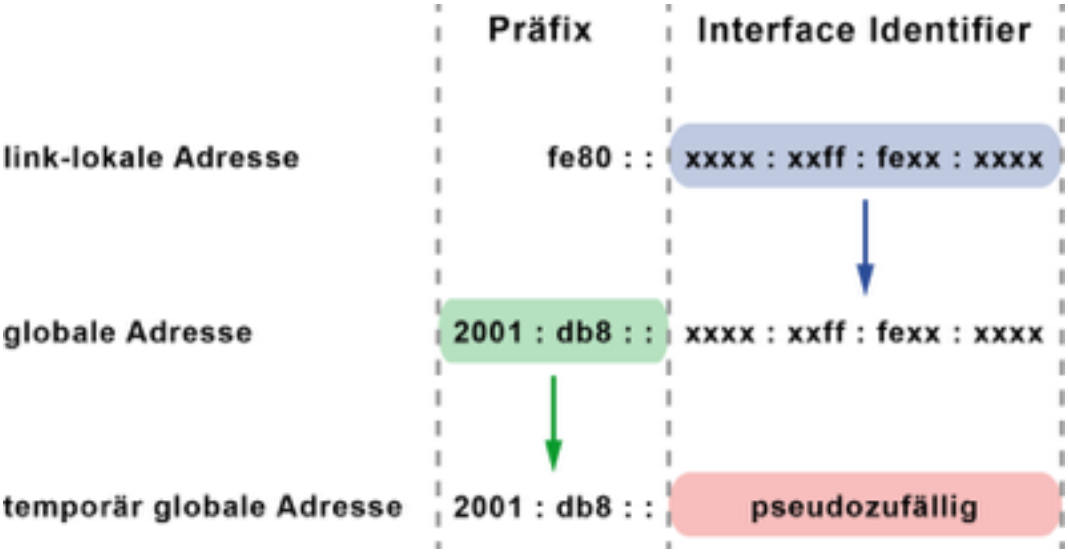
IPv6 kennt drei Wege, wie ein Host an eine globale IPv6-Adresse kommen kann. Entweder wird sie manuell konfiguriert, per Autokonfiguration (SLAAC) oder wie bei IPv4 per DHCP (DHCPv6) vergeben.

Mit SLAAC bezieht ein IPv6-Host nur den globalen Präfix per Router Advertisement und bildet sich zusammen mit dem bekannten Interface Identifier die globale IPv6-Adresse selber.
Sofern ein DHCPv6-Server eingerichtet ist, kann ein Host die globale IPv6-Adresse auch von dort beziehen, wie man es von IPv4 kennt.
DHCPv6 läuft unter der Bezeichnung Stateful Address Autoconfiguration. Hierbei wird an zentraler Stelle festgelegt, welcher Host, welche IPv6-Adresse bekommt/hat.

Woher bekommt ein Host seine temporären globalen IPv6-Adressen?

Die temporäre globale IPv6-Adresse basiert auf den Privacy Extensions. Hierbei wird der globale Präfix verwendet und der ursprüngliche Interface Identifier, der aus der MAC-Adresse gebildet wird, wird durch einen pseudozufälligen Interface Identifier ersetzt. Der wird regelmäßig geändert, um den Datenschutz zu gewährleisten.

Zusammenfassung: IPv6-Adressen



Woher bekommt ein Host die IPv6-Adresse des Standard-Gateways?

Im Rahmen der Router Advertisements wird nicht nur der globale Präfix, sondern auch die IPv6-Adresse des Standard-Gateways kommuniziert. Das ist ein Bestandteil von SLAAC.

Woher bekommt ein Host die IPv6-Adresse des DNS-Servers?

Es gibt zwei Möglichkeiten. Im Rahmen der Stateless Address Autoconfiguration (SLAAC) enthalten Router Advertisements die RDNSS-Option für Nameserver-Adressen. Alternativ kann die Bekanntgabe der Nameserver-Adresse über DHCPv6 erfolgen.

Ursprünglich war die Verteilung von DNS-Server-Adressen über einen zusätzlichen DHCPv6-Server vorgesehen. Erst mit dem RFC 6106 wurde im Rahmen von SLAAC mit Router-Advertisements die RDNSS-Option (Recursive DNS-Server) definiert. Mit der RDNSS-Option erfolgt die IPv6-Autokonfiguration zusammen mit den IPv6-Adressen eines oder mehrerer DNS-Server. Weitere Parameter einer Netzkonfiguration erfordert dann DHCPv6.

Beide Verfahren, SLAAC und DHCPv6, haben den Nachteil, dass sie für sich alleine nicht gut funktionieren. Das liegt daran, weil in älteren Betriebssystemen IPv6 nur unzureichend implementiert ist.

Leider fehlt in manchen Betriebssystemen die RDNSS-Option. Dazu gehören zum Beispiel Windows 7 und 8, sowie einige ältere Linux-Distributionen. Bei anderen Betriebssystemen und Geräten ist es womöglich ebenso.

Ein DHCPv6-Server ist deshalb für die vollständige IPv6-Konfiguration dringend notwendig, weil die Unterstützung der RDNSS-Option oftmals fehlt. Wenn die Betriebssysteme aber keinen DHCPv6-Client haben, wie zum Beispiel das veraltete Windows XP, dann ist es nur eingeschränkt IPv6-fähig. Es kann auf IPv6-Ebene keine Domain-Namen auflösen. Es sei denn man konfiguriert die IPv6-Adresse des DNS-Servers manuell.

Hinweis: Dass die RDNSS-Option bei manchen Clients fehlt, spielt in einer Dual-Stack-Umgebung (IPv4 und IPv6 im Parallelbetrieb) keine Rolle. Es ist ausreichend, wenn die Clients die IPv4-Adresse des DNS-Servers über DHCPv4 bekommen. Zur Namensauflösung verwendet der Client dann IPv4 und bekommt darüber dann die IPv6-Adresse zu einem Domain- oder Computer-Namen zurück. Sofern der Client eine globale IPv6-Adresse hat wird er die Verbindung wahlweise über IPv4 oder IPv6 aufbauen.

Stateless oder Stateful Address Autoconfiguration

Die "stateless" Autokonfiguration bietet den gleichen Komfort wie beim Betrieb eines sehr einfach gehaltenen DHCP-Servers (stateful). Ohne einen dedizierten DHCP-Server für derartige Informationen bereitstellen zu müssen. In kleinen Netzwerk ist das ein Segen. In

großen Netzwerken mag man sich darauf weniger gerne einlassen. Wenn man DHCP von IPv4 her kennt und dann mit SLAAC in den lokalen Netzen arbeiten muss, dann verliert man dabei an einigen Stellen auch Protokollierungs- und Kontrollmöglichkeiten. Davor graut es dem einen oder anderen Netzwerk-Administrator.

Prinzipiell muss man immer mit Router Advertisements arbeiten. Hierbei kann man sich überlegen, welche Informationen darüber verbreitet werden sollen und die fehlenden Teile der IPv6-Konfiguration per DHCPv6 angefordert werden müssen.

Momentan (Stand Anfang 2014) gibt es drei Szenarien die für eine IPv6-Autokonfiguration sinnvoll erscheinen:

1. IPv6-Autokonfiguration nur über Router Advertisement (stateless), ohne globale IPv6-Adresse.
2. Link-lokale und globale IPv6-Adresse und Default-Route über Router Advertisement, DNS-Adresse und weitere Parameter über DHCPv6 (stateless).
3. Globale IPv6-Adresse, DNS-Adresse und weitere Parameter über DHCPv6, die Default-Route über Router Advertisements (stateful).

Alternativ besteht sogar die Möglichkeit, SLAAC und DHCPv6 simultan zu betreiben. Das heißt, dass die IPv6-Konfiguration über beide Verfahren verteilt werden. Sowohl per Router Advertisement als auch per DHCPv6. Die Clients erhalten dann zwei globale Adressen. Der RFC 4862 sieht diese Variante vor. Eine doppelte IPv6-Adressvergabe dürfte aber nur in wenigen Fällen in Betracht kommen.

SLAAC

Stateless Address Autoconfiguration

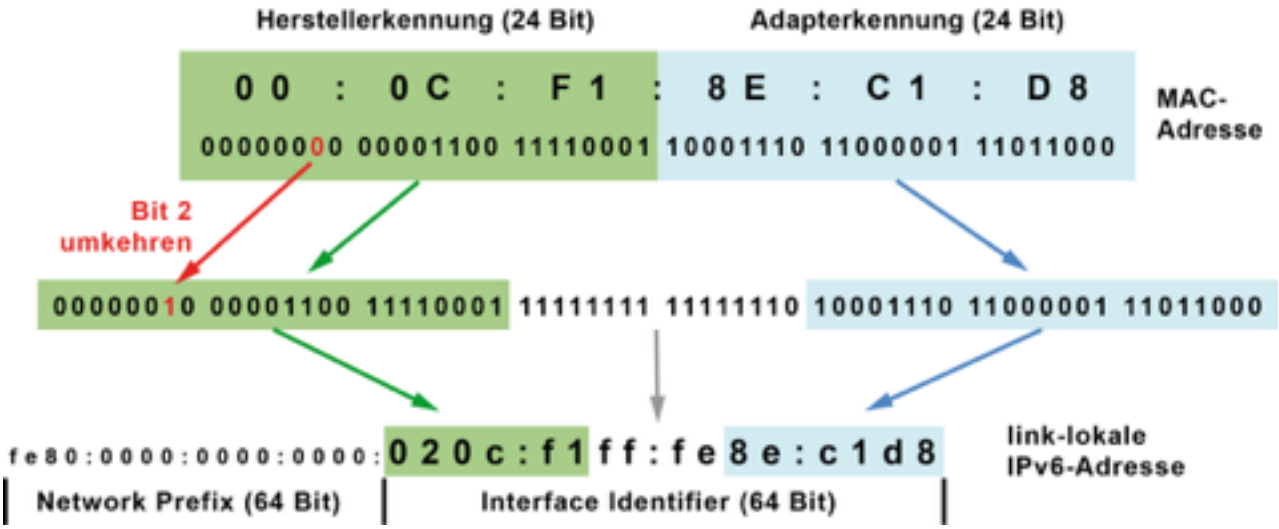
Stateless Address Autoconfiguration (SLAAC) ist ein Verfahren zur zustandslosen und automatischen Konfiguration von IPv6-Adressen an einem Netzwerk-Interface. Mit "stateless" bzw. "zustandslos" ist gemeint, dass die jeweilige IPv6-Adresse nicht zentral vergeben und gespeichert wird. Demnach erzeugt sich der Host seine IPv6-Adresse unter Zuhilfenahme zusätzlicher Informationen selbst. SLAAC ist die Weiterentwicklung von Verfahren für die klassische IP-Autokonfiguration unter IP4. Anders als bei IPv4 übernehmen IPv6-Router dabei eine aktive Rolle.

Man unterscheidet grob gesehen zwischen globalen IPv6-Adressen (Global Scope) und link-lokalen IPv6-Adressen (Local Scope). Mit der Stateless Address Autoconfiguration kann sich ein IPv6-Host sowohl eine link-lokale, als auch eine globale IPv6-Adresse erzeugen.

Damit bietet IPv6 den gleichen Komfort wie beim Betrieb eines sehr einfach gehaltenen DHCP-Servers.

Das Ziel von SLAAC ist, dass ein Host zumindest eine link-lokale IPv6-Adresse bekommt, mit der in jedem Fall eine Verbindung innerhalb des lokalen Netzwerks möglich ist. In einem weiteren Schritt würde sich ein Host per SLAAC ein globale IPv6-Adresse erzeugen, mit der er auch Verbindungen ins Internet aufbauen kann.

SLAAC für eine link-lokale IPv6-Adresse



Eine IPv6-Adresse besteht aus insgesamt 128 Bit. Eine link-lokale IPv6-Adresse wird aus einem Präfix (64 Bit) und einem Suffix (64 Bit) gebildet. Der Präfix für alle link-lokalen IPv6-Adressen ist immer "fe80:0000:0000:0000". Das Suffix (Interface Identifier) ist der EUI-64-Identifier oder IEEE-Identifier, der aus der MAC-Adresse (Hardware-Adresse des

Netzwerkadapters) gebildet wird. In der Mitte der 48-Bit-MAC-Adresse (zwischen dem dritten und dem vierten Byte) werden mit "ff:fe" zwei feste Bytes eingefügt, damit es 64 Bit werden. Zusätzlich wird noch das zweite Bit im ersten Byte der MAC-Adresse invertiert. Das heißt, aus "1" wird "0" und aus "0" wird "1".

Warum? Als man den Adressraum für MAC-Adressen festgelegt hat, hat man vorausschauend einen Adressbereich festgelegt, den man sich selber ausdenken kann. Die also nicht zugewiesen werden. Im zweiten Bit vom ersten Byte steckt ein Indikator drin, der diese Information enthält, ob die MAC-Adresse von der IEEE zugewiesen wurde oder ob sie selber ausgedacht ist. Wenn das Bit auf "0" ist, dann handelt es sich um eine MAC-Adresse, die von der IEEE zugewiesen wurde. Ist das Bit auf "1", dann ist es eine Phantasie-Adresse.

Wenn man sich jetzt die IPv6-Adresse bildet, dann können die Bits beliebig sein. Wenn man sich die MAC-Adresse ausgedacht hat, dann wird bei einer EUI-64 das Bit in der Regel zur "0" gedreht.

Auf diese Weise wird zum Beispiel die MAC-Adresse "00:0C:F1:8E:C1:D8" zum Interface Identifier "020c:f1ff:fe8e:c1d8". Und der Host bildet sich so die link-lokale Adresse "fe80:0000:0000:0000:020c:f1ff:fe8e:c1d8".

Bevor der Host diese link-lokale Adresse nutzen kann muss er eine Duplicate Address Detection (DAD) durchführen, um festzustellen ob die Adresse im lokalen Netz womöglich schon existiert.

DAD - Duplicate Address Detection

Um Adresskollisionen zu vermeiden sollte der Host bei einer neu generierten IPv6-Adresse eine Duplicate Address Detection (DAD) durchführen.

1. Neighbor Solicitation: Dazu schickt der Host eine Anfrage an die generierte Adresse ins lokale Netz. Als Antwort-Adresse dient eine Multicast-Adresse.
2. Neighbor Advertisement: Falls eine andere Station die IPv6-Adresse bereits nutzt, kommt eine Antwort zurück.

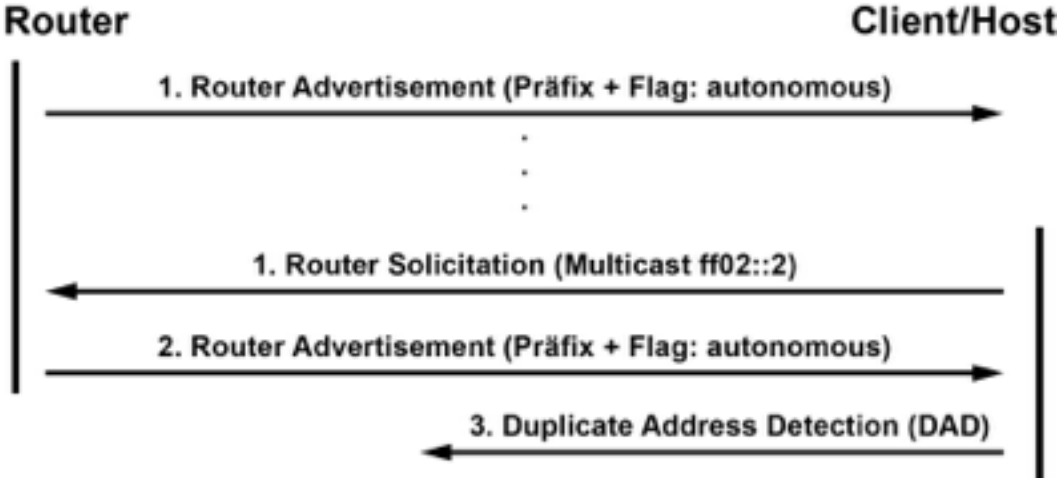
Erst wenn keine Antwort von dieser Adresse zurückkommt bindet sich das Interface an diese Adresse und kann sie für die Kommunikation nutzen.

Weil es keine Pflicht gibt eine DAD durchzuführen, sind Adresskollisionen durchaus möglich. Aufgrund des sehr großzügigen Adressraums und der weltweit eindeutigen MAC-Adressen aber eher unwahrscheinlich.

Sollte es doch einmal zu einer Kollision kommen und die IPv6-Adresse tatsächlich schon existieren, dann muss die IPv6-Adresse vom Anwender manuell geändert werden.

Dann sollte man gleich das ganze Netzwerk überprüfen. Es könnte dann sein, dass jemand eine MAC-Adresse gekapert hat und per MAC-Spoofing ins Netzwerk eingedrungen ist.

SLAAC für eine globale IPv6-Adresse (ohne Privacy Extensions)



Mit seiner link-lokalen IPv6-Adresse kann der Host nur im lokalen Netzwerk kommunizieren. Für das Internet braucht er eine zusätzliche IPv6-Adresse, die er sich ebenfalls selber generiert. Dazu muss der Host beim Standard-Gateway (nächster Router) nachfragen, was der Präfix des globalen Adressblocks ist. Dabei handelt es sich um den Adressraum, den man vom Netzzugangsprovider (ISP) zugeteilt bekommen hat. Der Präfix ist in der Regel 64 Bit lang.

Diesen Präfix gibt der Router in regelmäßigen Abständen per Router Advertisement bekannt. Sofern der Client oder Host diesen noch nicht erhalten hat, kann er den Präfix auch per Solicitation Message (Router Solicitation) anfordern. Als Antwort kommt ein oder auch mehrere Router Advertisements mit dem globalen Präfix zurück.

1. Router Solicitation (Solicitation Message): Mit seiner link-lokalen IPv6-Adresse bittet der Host auf der Multicast-Adresse "ff02::2" um den globalen Präfix (optional).
2. Router Advertisement (Advertisement Message): Der Router schickt daraufhin eine Nachricht mit dem globalen Präfix für dieses Netzwerk, der MTU (Größe der IP-Pakete) und dem Flag „autonomous".
3. Aus dem per Router Advertisement erhaltenen Präfix und dem Interface Identifier der link-lokalen Adresse wird dann die globale IPv6-Adresse gebildet. Danach prüft der Host, ob diese Adresse im lokalen Netzwerk schon vergeben ist (Duplicate Address Detection, DAD). Wenn sie frei ist, weist er die globale Adresse seiner Netzwerkschnittstelle zu.

SLAAC für eine globale IPv6-Adresse mit Privacy Extensions (Lösung des Datenschutz-Problems)

Der Hostanteil bzw. Interface Identifier einer per SLAAC erzeugten globalen IPv6-Adresse ist weltweit eindeutig, sofern die dafür verwendete MAC-Adresse weltweit eindeutig ist. Das bedeutet, am Interface Identifier kann man einen Host identifizieren. Unabhängig in welchem Netz (Präfix) er sich befindet. Da alle Computer und Computer-ähnlichen Geräte

über eine oder mehrere Hardware-Adressen verfügen ist jeder Host über seinen Interface Identifier identifizierbar.

Da viel Hosts, zum Beispiel Smartphones und Tablets, nur von einer Person genutzt werden, sind IPv6-Adressen mit einem auf eine MAC-Adresse bezogenen Interface Identifier personenbezogene Daten. Somit wäre jeder Nutzer jederzeit identifizierbar. Mit der Einführung von IPv6 ist damit die Angst um den Verlust der Privatsphäre gestiegen.

Deshalb gibt es aus Gründen des Datenschutzes die Erweiterung "Privacy Extensions", die standardmäßig in allen IPv6-Clients aktiviert sein sollte. Statt die eindeutige MAC-Adresse für den Interface Identifier zu verwenden, generiert der Host für den Interface Identifier einen pseudozufälligen Hash-Wert.

Ein anderes Verfahren erzeugt Cryptographically Generated Addresses (CGAs). Auch CGN verhindert die Identifizierung eines Hosts anhand seiner IPv6-Adresse.

Unvollständige Autokonfiguration

Leider fehlt in manchen Betriebssystemen eine vollständige Unterstützung von IPv6. Das betrifft Windows 7 und 8, sowie das veraltete Windows XP. Auch ältere Linux-Distributionen und mobile Betriebssysteme sind nur eingeschränkt IPv6-tauglich.

SLAAC bei Servern und Routern

Jeder IPv6-Client richtet automatisch eine eigene link-lokale und globale IPv6-Adressen ein. Für die meisten Endgeräte ist das eine praktische Lösung. Doch bei einem Server oder Router sollten sich die IP-Adressen nicht ändern. Insbesondere dann nicht, wenn dem Server per DNS ein Host- oder Domain-Name zugeordnet ist. Und fürs IP-Routing muss ein Router zwangsläufig eine feste IP-Adresse haben. Hier sollte man die IPv6-Adresse manuell zuweisen oder zentral per DHCPv6 eine statische IP-Adresse vergeben.

Bei Servern empfiehlt es sich daher, die Autokonfiguration (SLAAC) zu deaktivieren bzw. serverseitig zu ignorieren und eine statische IPv6-Adressen nach dem Zufallsprinzip zu erzeugen und nicht durchnummeriert vergeben.

Privacy Extensions

Privacy Extensions ist eine Erweiterung (RFC 4941) für die Stateless Address Autoconfiguration (SLAAC) von IPv6, um IPv6-Adressen zu bilden, die keinen Rückschluss auf den Nutzer zulassen. Privacy Extensions ist ein Verfahren, um den Hostanteil der IPv6-Adressen zu anonymisieren.

Hinweis: Privacy Extensions war in vielen Betriebssystemen lange nicht aktiviert. Inzwischen haben sich die Privacy Extensions als Standardeinstellung in fast allen Betriebssystemen etabliert.

Warum Privacy Extensions?

In der ursprünglichen Fassung von SLAAC erzeugt der IPv6-Host seine IPv6-Adresse aus einem Präfix und seinem eigenen Interface Identifier, der seine MAC-Adresse (Hardware-Adresse des Netzwerkadapters) enthält. In der Mitte der Hardware-Adresse werden zwei feste Bytes (ff:fe) eingefügt. Das siebte Bit im ersten Byte der MAC-Adresse wird umgekehrt.

> 00:0C:F1:8E:C1:D8 (MAC-Adresse, 48 Bit)
>
> 020c:f1ff:fe8e:c1d8 (Interface Identifier der IPv6-Adresse, 64 Bit)

Eine gewisse Ähnlichkeit ist nicht zu leugnen. Das bedeutet, am Interface Identifier kann man einen bestimmten Host erkennen, weil der sich wegen der festen MAC-Adresse nie ändert. Auch wenn man das Netz wechselt oder sich der Präfix aus einem anderen Grund ändert. Der Interface Identifier bleibt immer gleich. Und bei IPv6 gibt es kein NAT hinter dem man sich wie bei IPv4 verstecken kann.

Ein Traum für die Werbebranche und Ermittlungsbehörden. Jeder Internet-Nutzer wäre anhand seiner IPv6-Adresse identifizierbar, solange er das Gerät nicht mit anderen teilt. Außerdem kann man aus dem Interface Identifier auch den Hersteller oder das Gerät ablesen. Daraus können Werbefachleute durchaus ablesen, um welche Klientel es sich handelt. Deswegen gab und gibt es Bedenken bezüglich des Datenschutzes und der Privatsphäre bei der Nutzung von IPv6-Adressen.

Abhilfe schafft Privacy Extensions. Dahinter verbirgt sich ein Mechanismus, den Interface Identifier regelmäßig zu wechseln.

Wie funktioniert Privacy Extensions?

Privacy Extensions heben die Kopplung von Interface Identifier und MAC-Adresse auf und erzeugen mehr oder weniger zufällige Interface Identifier. Privacy Extensions geht sogar noch einen Schritt weiter und erzeugt periodisch neue Adressen mit verändertem Interface Identifier. Anschließend werden mit diesen wechselnden Adressen ausgehende Verbindungen hergestellt. Auf diese Weise wird auf IP-Ebene die Erstellung von

Bewegungsprofilen verhindert. Damit wird die Anonymität, wie bei IPv4 durch NAT, teilweise wieder hergestellt. Sofern Privacy Extensions aktiviert ist passiert das automatisch und regelmäßig. Darum kümmert sich das Betriebssystem.

Oft findet man den Hinweis, dass der Interface Identifier bei Privacy Extensions zufällig erzeugt wird. Das ist allerdings nicht ganz richtig. Die Vorgehensweise ist wie folgt. Zum aktuellen NTP-Zeitstempel mit 64 Bit kommt die MAC-Adresse hinzu und dann macht man daraus einen SHA1-Hash mit einer Länge von 64 Bit. Fertig ist der "zufällige" Interface Identifier. Zusammen mit dem globalen Präfix wird daraus eine temporär globale IPv6-Adresse.

Eine temporäre IPv6-Adresse bleibt solange bestehen, wie sie benutzt wird. Wenn eine neue erzeugt wird, wird die alte als "deprecated" gekennzeichnet. Die neue Adresse wird dann für alle neuen ausgehenden Verbindungen benutzt. Und die alte nur noch so lange, wie sie von eingehenden Verbindungen benutzt wird (temporary deprecated). In der Regel verfallen alle temporären Adressen automatisch beim Ausschalten eines Clients. Wird ein Client nie ausgeschaltet, beispielsweise ein Smartphone oder Tablet, dann wird eine "deprecated" IPv6-Adresse unter Umständen erst nach mehreren Tagen endgültig gelöscht.

Eine Identifikation eines IPv6-Nutzers kann nur noch über den Präfix erfolgen, aber der kann sich natürlich auch ändern, wenn zum Beispiel der Provider diesen regelmäßig wechselt. Man kann höchstens das Netz identifizieren, aber nicht mehr eine einzelne Person oder ein einzelnes Gerät.

Privacy Extensions in Servern und Routern

Privacy Extensions haben sich als Standardeinstellung bei allen Betriebssystemen etabliert. Aber, es gibt Situationen, in denen sie besser abgeschaltet werden. Denn die Adressverschleierung erschwert das System- und Netzmanagement. Beispielsweise bei zentralen Servern und Routern. Hier sind wechselnde IPv6-Adressen nicht gewünscht.

Privacy Extensions unter Windows

Nicht jedes Betriebssystem verhält sich gleich. Ab Windows Vista erzeugt das Betriebssystem bei der Installation einen konstanten Interface Identifier. So hat jede aktive Netzwerk-Schnittstelle unter Windows mindestens zwei global gültige IPv6-Adressen hat. Für die Dienstfreigaben ist ein konstanter Interface Identifier nötig.

Datenschutz und Privatsphäre mit Privacy Extensions ist nur die halbe Wahrheit

Dass Privacy Extensions die Privatsphäre schützen ist leider nur die halbe Wahrheit. Zwar wechselt Privacy Extensions ständig den Hostanteil einer IPv6-Adresse, doch der Präfix bleibt gleich. In kleinen privaten und lokalen Netzen, wo es viel Datenverkehr von

wenigen Clients gibt, wäre ein bestimmter Client immer noch über den Präfix (vordere 64 Bit) einer IPv6-Adresse identifizierbar.

Bei privaten Internet-Zugängen ist der Internet-Provider für die Zuteilung des IPv6-Präfixes zuständig. Manche der ISPs gehen auch über die normierten Verhaltensweisen hinaus und verbessern damit den Datenschutz erheblich, in dem sie den Internet-Zugangs-Routern ihrer Kunden regelmäßig ein neues Präfix zuteilen.

Warum gibt es die globale IPv6-Adresse mit konstantem Interface Identifier, wenn es doch Privacy Extensions gibt?

Die eigentliche Frage ist, was bringt es überhaupt die MAC-Adresse in den Hostanteil zu integrieren? Mit Privacy Extensions wäre dieser Schritt doch überflüssig.

Prinzipiell macht es durchaus Sinn einen festen Interface Identifier zu haben. Damit wäre ein Host von außen mit einer festen IPv6-Adresse erreichbar, wenn diese explizit für einen Dienst benötigt wird. Beispielsweise, weil man Dienste nach außen anbieten will oder weil man Ende-zu-Ende-Dienste nutzen will, für die eine globale, aber nicht temporäre Adresse vorteilhaft wäre, wie man es sich von IPv6 verspricht. Typischerweise für VoIP, SIP und Messaging. Bei IPv4 besteht das Problem, wegen dem Adressmangel, dass durch private Adressen und der damit verbundenen NAT-Umgebung ein Host keine eigene öffentliche Adresse hat. Weil dadurch keine echten Ende-zu-Ende-Dienste genutzt werden können, müssen für viele Kommunikationsprotokolle Umgehungsmechanismen für NAT geschaffen werden. Bei IPv6 kann man darauf verzichten, aber nur deshalb, weil es globale IPv6-Adressen mit konstantem Interface Identifier gibt.

Es sollte so sein, dass nur bei abgehenden Verbindungen, zum Beispiel beim Zugriff aufs Web, die temporäre IPv6-Adresse verwendet wird. Hier will man die Identifizierung anhand der IPv6-Adresse vermeiden.

Woran liegt es wenn Privacy Extensions nicht funktioniert?

Prinzipiell ist es so, dass Privacy Extensions in der Standardeinstellung bei allen Betriebssystemen aktiviert ist. Trotzdem wird man im einen oder anderen System trotzdem eine link-lokale IPv6-Adresse mit einem EUI-64-Suffix finden. Also ohne Privacy Extensions. Die Frage ist warum?

Privacy Extensions hat nichts mit dem Local Scope bzw. link-lokalen Adressen zu tun. Privacy Extensions ist nur dann aktiv, wenn ein Host einen globalen Präfix zugewiesen bekommen hat (Global Scope). Und erst dann wird ein zufälliger Interface Identifier gebildet. Und erst danach wird je nach Einstellung eine zusätzliche temporäre link-lokale IPv6-Adresse gebildet oder die EUI-64-Adresse ersetzt.

Für den Local Scope hat Privacy Extensions keinerlei Bedeutung. Für link-lokale Adressen ist das auch sinnlos, weil diese Adressen nur innerhalb des selben Netzsegments gültig

sind und dort die Maschine sowieso anhand ihrer MAC-Adresse identifizierbar ist. Also, ohne globalen Präfix macht Privacy Extensions nicht viel Sinn.

DHCPv6

Stateful Address Configuration

DHCP ist ein Protokoll, um die IP-Konfiguration in einem TCP/IP-Netzwerk zu verwalten und an die Hosts zu verteilen. Mit DHCP ist jede Netzwerk-Station in der Lage sich selber halbautomatisch zu konfigurieren.

Bei IPv6 benötigt die IP-Konfiguration eigentlich keinen DHCP-Dienst. Dafür gibt es die Stateless Address Autoconfiguration (SLAAC). Doch nicht alle IPv6-Clients können alle Optionen der IP-Konfiguration auf diese Weise entgegennehmen. Zum Beispiel die RDNSS-Option mit dem DNS-Server. DHCPv6 ist im Prinzip das einzige Verfahren, welches diese und weitere Angaben innerhalb der Autokonfiguration ergänzen kann. Um wie bei IPv4 mit DHCPv4 die gleichen Funktionalitäten für IPv6 zu ermöglichen, wurde DHCPv6 definiert.

Weil bei DHCPv6 die IP-Konfiguration zentral vergeben und gespeichert wird, spricht man von "Stateful" Address Configuration.

In der Praxis sieht die IP-Autokonfiguration häufig so aus: Per Router-Advertisement werden die IP-Grundparameter verteilt und mit DHCPv6 alles weitere. Die Autokonfiguration bleibt dabei "stateless". Anders sieht es aus, wenn auch DHCPv6 der Präfix verteilt wird. Dann ist die Autokonfiguration "stateful".

Hinweis: Auch bei einer "stateful" IPv6-Konfiguration muss das Router-Advertisement aktiviert sein. Nur so kann jeder Host seine link-lokale IPv6-Adresse erzeugen. Dann allerdings ohne den globalen Netzpräfix für die globale IPv6-Adresse. Der globale Präfix ist in diesem Fall im DHCPv6-Server hinterlegt (stateful). Der generiert daraus die 128 Bit lange IPv6-Adresse für die anfragenden Clients. Welchen Hostanteil (Interface Identifier) der Server aus dem möglichen Adressraum verwendet, hängt dabei von der Implementierung ab.

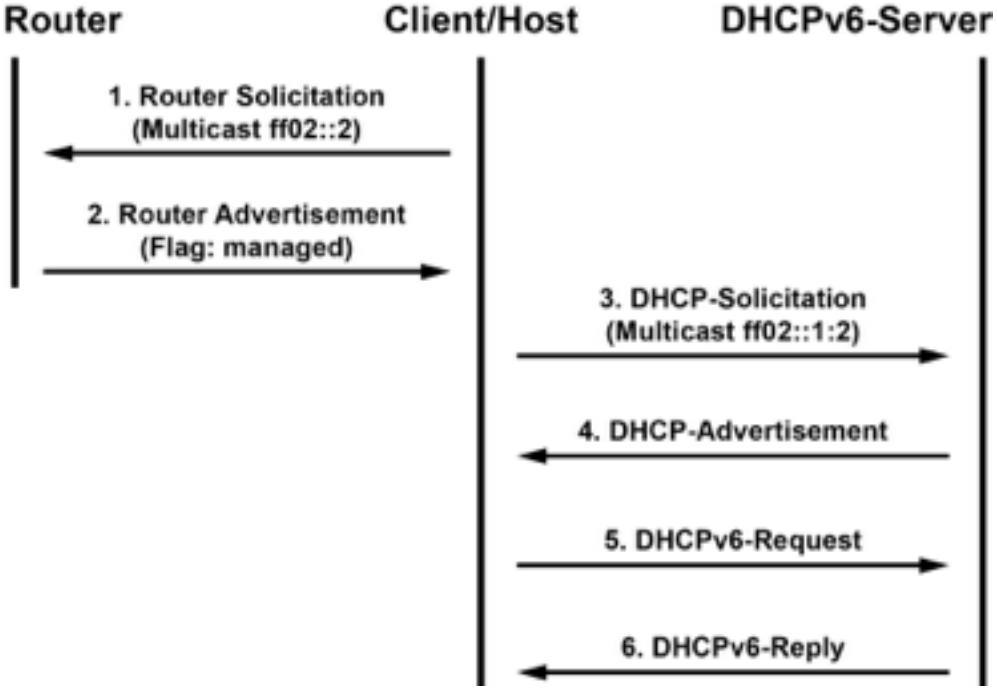
Ablauf von DHCPv6

Der eigentliche Ablauf von DHCPv6 erfolgt in 4 Schritten (Schritt 3 bis 6). In der folgenden Beschreibung ist die vorhergehende Autokonfiguration der globalen IPv6-Adresse über SLAAC berücksichtigt (Schritt 1 und 2). Wobei das Router Advertisement (Schritt 2) der Router Solicitation (Schritt 1) zuvorkommen kann.

1. Der Client bzw. Host sendet eine Router Solicitation (RS) an die Multicast-Adresse "ff02::2". Damit bittet der Client um einen Präfix für die globale IPv6-Adresse.
2. Der zuständige Router antwortet mit einem Router Advertisement (RA). Die Nachricht enthält die MTU (Größe der IP-Pakete) und den globalen Präfix für das Netzwerk (stateless) oder das Flag "managed" (stateful).

3. Dann sendet der Client eine DHCP-Solicitation-Nachricht an die DHCPv6-Multicast-Adresse "ff02::1:2" (alle DHCPv6-Server).
4. Die erreichbaren DHCPv6-Server antworten mit einer DHCP-Advertisement-Nachricht, die die Parameter (DNS-Server, NTP-Server etc.) zur Vervollständigung der IP-Konfiguration enthält (bei stateful auch den globalen Präfix).
5. Der Client wählt eine IP-Konfiguration aus und fordert sie beim jeweiligen DHCPv6-Server mit einem DHCPv6-Request explizit an.
6. Der DHCPv6-Server speichert die IP-Konfiguration mit der Client-ID (Stateful Address Configuration) und bestätigt dem Client die IP-Konfiguration per DHCPv6-Reply. Alle anderen DHCPv6-Server, die keine Anforderung des Clients erhalten haben, geben ihre angebotene IPv6-Adresse wieder frei.

Der Client konfiguriert sich nach Erhalt der Bestätigung und kann anschließend im Netz mit einer globalen IPv6-Adresse kommunizieren.



DHCPv6 vs. Router Advertisment

Ob man DHCPv6 ODER Router Advertisement für die IPv6-Autokonfiguration verwendet, darüber kann man gar nicht entscheiden. Grundsätzlich kommt man ohne Router Advertisements nicht aus. Auch wenn man keine "stateless" IPv6-Adressen haben möchte, sind Router Advertisements nötig. Der zuständige Router verteilt seine Router Advertisements mit dem Flag "managed", woran ein Host erkennt, dass er sich die globale IPv6-Adresse per DHCPv6 holen muss. Das heißt, ohne RA ist kein DHCPv6 möglich.

DHCPv6 hat noch einen weiteren Makel, weshalb RAs unverzichtbar sind. Die ursprüngliche Spezifikation von DHCPv6 sah die Konfiguration eines Standard-Gateways

nicht vor. Hierzu gibt es nur den Vorschlag "DHCPv6 Route Options", womit sich die Adresse des Default-Gateways über DHCPv6 konfigurieren lässt, wie es bei DHCPv4 üblich ist. "DHCPv6 Route Options" würde es erlauben, DHCPv6 ohne Router-Advertisements zu betreiben. Allerdings ist es fraglich, ob das jemals so zum Einsatz kommen wird.

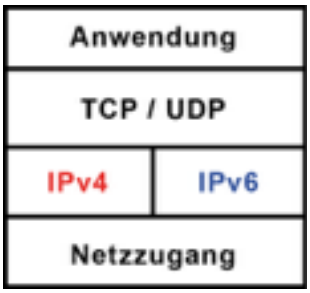
Deshalb gilt, dass jede IPv6-Konfiguration mit DHCPv6 auch immer parallel mit Router Advertisements erfolgen muss.

Address Selection

Address Selection ist ein Verfahren, welches darüber entscheidet, welche IP-Adresse verwendet wird, wenn ein Host sowohl eine IPv4- als auch eine IPv6-Adresse hat (Dual Stack).

Die ursprüngliche Strategie, ist IPv6 da, dann nehme ich auch IPv6, ist nicht mehr gültig. Inzwischen macht man es davon abhängig, welche Verbindung schneller ist. Das macht man deshalb, damit der Nutzer und auch die Anwendung immer die beste Verbindung bekommen sollen.

Das bedeutet, das Betriebssystem oder eine Anwendung probieren aus, was am besten funktioniert. Wie genau der Ablauf ist, dass ist nicht bekannt. Es könnte aber so sein, dass IPv4 und IPv6 wechselweise ausprobiert werden. Und das was am besten funktioniert, dass wird dann verwendet.



Das heißt, dass IPv6 nicht zwangsläufig verwendet wird. In manchen Netzwerken gibt es kein natives IPv6, sondern wird über IPv4 getunnelt und ist deshalb langsamer als IPv4. Wenn ein Betriebssystem zwischen einer IPv6-Verbindung als Tunnel und einer IPv4-Verbindung entscheiden kann, dann wird es sich eher für IPv4 entscheiden. Das hängt aber vom Betriebssystem ab.

An diesem Punkt ist die Address Selection in den IPv6-Standards nicht genau beschrieben. Das macht jede Implementierung etwas anders. Man muss damit rechnen, dass sich das über die Zeit ändert.

Jetzt ist es aber so, dass ein IPv6-Host an seinen Netzwerk-Interface mehrere IPv6-Adressen haben kann. Ein typischer IPv6-Client hat drei oder mehr IPv6-Adressen. Eine, die link-lokal gültig ist, eine die global gültig ist und eine die temporär global gültig ist. "Gültig" bedeutet, dass die IPv6-Adresse in einem bestimmten "Gültigkeitsbereich"

verwendet wird. Wird ein "link-lokales" Ziel adressiert, dann wird die link-lokale Adresse verwendet. Wird ein "globales" Ziel adressiert, zum Beispiel im Internet, dann wird eine globale Adresse verwendet. Wenn vorhanden, eine temporäre. Die Nutzung der globalen IPv6-Adresse, mit festem Interface Identifier, wird nur dann verwendet, wenn ein Kommunikationsprotokoll zur Anwendung kommt, dass eine IPv6-Adresse erwartet, die sich nicht ändert, also nicht temporär ist. Beispielsweise bei Messaging-Diensten.

Das ist aber noch nicht alles. Denkbar ist, dass in einem lokalen Netzwerk mehrere Präfixe verteilen werden, wovon einer bevorrechtigt sein sollte. Beispielsweise, weil ein Router außer Betrieb genommen werden soll, wird er als "deprecated" konfiguriert und ein anderer als Router priorisiert (Renumbering). Kleine SoHo-Router können das natürlich nicht.

Denkbar ist, dass ein Router seinen Präfix zurückzieht. Beispielsweise, wenn er an seinem Uplink die Connectivity verliert. Dann schickt er Router Advertisments, in dem er mitteilt, dass sein Präfix nicht mehr benutzt werden soll.

Mit diversen Routing-Protokollen ging das früher schon. Aber bei IPv6 ist das neu. Jeder IPv6-Client kann mit mehreren IPv6-Präfixen und mehreren globalen Adressen umgehen.

Wie weiß ein Host in so einem Fall, welchen Präfix er nehmen soll? Es gibt hierzu drei Möglichkeiten. Die Verbindung wurde von außen initiiert, dann antwortet der Host auf dem Präfix, auf dem der Request hereinkam. Alternativ gibt es die Anwendung vor. Wenn es der Anwendung egal ist, dann muss das Betriebssystem eine Adresse wählen.

Dabei tritt einer interessanter Effekt ein. Nur weil das Betriebssystem eine globale IPv6-Adresse auswählt, heißt das nicht, dass die Verbindung über die Route geführt wird, von der der Präfix kommt. Das bedeutet, dass ein Internet-Service-Provider unter Umständen Datenpakete mit einer Quell-Adresse routet, die nicht aus seinem Netz stammt. Die Frage ist, wie verhält sich der Provider? Routet er die Adresse, so dass die Verbindung zustande kommt, oder verwirft er die Adresse, wodurch die Verbindung nicht zustande kommt.

Multihoming / Renumbering

Das Multihoming und Renumbering bei IPv6-Netzwerken basiert darauf, dass ein IPv6-Host mehrere IPv6-Adressen haben kann. Unter anderem auch mehrere globale Adressen und davon auch mit unterschiedlichen Präfixen, die ein Host von mehreren Routern per SLAAC bzw. Router Advertisement bekommt.

Multihoming bedeutet in dem Fall, dass ein Host in mehreren IPv6-Netzwerken zu Hause ist, obwohl es physikalisch das selbe Netz ist.

Bei Renumbering geht es darum, einen, mehrere oder alle IPv6-Hosts im laufenden Betrieb in ein anderes IPv6-Netzwerk (Änderung des Präfixes) umzuziehen.

Wenn ein IPv6-Host mehrere globale Adressen mit unterschiedlichen Präfixen hat, dann stellt sich natürlich die Frage, welche davon verwendet er für ausgehende Verbindungen. Und dann stellt sich noch die Frage, welches Standard-Gateway verwendet der Host für ausgehende Verbindungen. Hierbei stellt sich wiederum die Frage, wie ein Provider-Netz darauf reagiert, wenn es IPv6-Pakete mit ihm unbekannten Absender-IPv6-Adressen von seinen Kunden-Anschlüssen routen soll.

Multihoming

Ein an einem lokalen Netzwerk angeschlossenes Interface gilt dann als "multihomed", wenn es mehrere globale IPv6-Adressen hat, die unterschiedliche Präfixe aufweisen. Das heißt, dass das lokale Netzwerk über mehrere ISPs an das Internet angebunden ist. Dabei haben die Interfaces Adressen von jedem beteiligten ISP. Dies dient nicht nur der Redundanz. Es kann auch durch die Wahl der Absenderadresse der verwendeten Provider bestimmt werden.

Bei der automatischen Ermittlung des Standard-Gateways kann ein Host auch mehrere Präfixe erhalten und damit auch mehrere IPv6-Adressen beziehen.

So ist es möglich, dass mit IPv6 ein lokales Netzwerk über mehrere Provider am Internet angebunden ist. Das vereinfacht das Umziehen von Netzen und kann genutzt werden, um eine höhere Ausfallsicherheit zu erreichen. Wenn eine Netzanbindung ausfällt, dann wählen die Hosts einfach die alternative Route.

Renumbering

Die Mechanismen zur "stateless" Autokonfiguration erlauben das Hinzufügen und Entfernen von Netzpräfixen und somit die Rekonfiguration eines Netzwerks im laufenden Betrieb.

Dank Renumbering lässt sich ein Interface relativ einfach mit neuen Adressen bestücken. Sei es um ein neues Adressschema einzuführen oder den Provider zu wechseln. Ein

Interface wird in einen multihomed-ähnlichen Zustand gebracht. Gleichzeitig lässt man die Gültigkeit der alten Adressen langsam auslaufen.

Dazu kann man mehrere Netzzugangsrouter unterschiedlich Konfigurieren. Über Router Advertisements kann ein Router sagen "priorisiere mich" und ein anderer Router "benutze mich nicht". Auf diese Weise kann man einen neuen Router in Betrieb und einen anderen außer Betrieb nehmen. Kleine SoHo-Router können das natürlich nicht.

IPv6-Multicast

IPv6-Multicast basiert auf der Frage, wie verschicken wir Pakete an Ziele, deren link-lokale Adresse uns noch nicht bekannt ist? Die Lösung heißt Multicast und die zugehörigen Adressen sind die Multicast-Adressen.

In IPv6 bilden Multicast-Adressen eigene Multicast Scopes (Gültigkeitsbereiche), die nicht mit den Address Scopes verwechselt werden dürfen.

Die Multicast-Adressen von IPv6 umfassen Netzwerk-Knoten und Netzwerk-Dienste in Multicast-Gruppen zusammen. Jede Gruppe ist über eine eigene Adresse erreichbar. Diesen Gruppen kann sich ein Host selber zuweisen und er kann auf Pakete antworten, die mit der entsprechenden Multicast-Adresse adressiert sind.

Multicast-Adressen und ihre Gültigkeitsbereiche

Multicast-Adressen sind durch das Präfix "ff::/8" gekennzeichnet. Sie beginnen immer mit "ff". Danach folgen 4 Bit für Flags und weitere 4 Bit für die Angabe des Multicast Scopes (Gültigkeitsbereich).

- Der Scope "ff01" gilt nur am lokalen Interface. Die so adressierten Pakete verlassen also nie die Schnittstelle (Node-Local Scope Multicast Addresses).
- Der Scope "ff02" gilt für link-lokale Multicast-Adressen (Link-Local Scope Multicast Addresses).

Es gibt noch weitere Gültigkeitsbereiche, die allerdings in der Praxis keine Rolle spielen.

Nach dem Gültigkeitsbereich folgt eine Reihe von Gruppen von Nullen, die mit "::" abgekürzt werden. Die Multicast-Adresse endet mit einer Nummer, die für eine Multicast-Gruppe steht. Eine solche Gruppe ist zum Beispiel "alle Nodes". Bei IPv4 kannte man den gute alten Broadcast. Diese exklusive Broadcast-Adresse kennt IPv6 nicht mehr. Dafür gibt es die Multicast-Adresse "ff02::1", die für alle IPv6-Hosts steht.

Ein Auszug der Multicast-Gruppen:

- ff0X::1 : alle IPv6-Stationen
- ff0X::2 : alle Router
- ff0X::f : UPnP
- ff0X::101 : alle Zeitserver (NTP)
- ff0X::1:2 : DHCPv6-Server

Hinweis: Nicht alle Multicast-Gruppen machen in jedem Gültigkeitsbereich Sinn.

Multicast mit Ping

Jede Multicast-Gruppe kann man auf der Kommandozeile mit Ping ansprechen. Beispielsweise alle link-lokale IPv6-Stationen.

Windows: ping -6 ff02::1

Linux: ping6 -c 5 ff02::1

Sofern die Geräte mit dem LAN verbunden sind und keine Firewall oder eine andere Sicherheitsmaßnahme es verhindert, antworten alle IPv6-Hosts.

Problem: Multicast im geswitchten Netzwerk

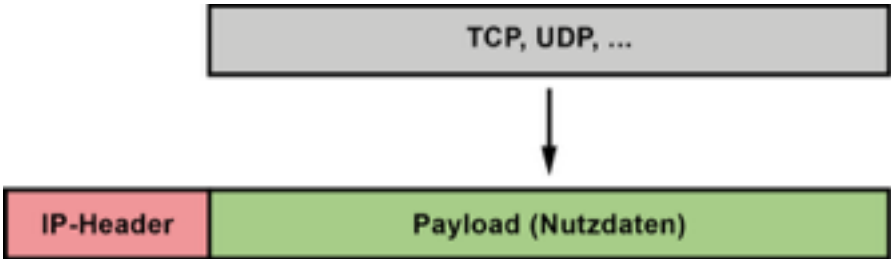
Grundlegende Mechanismen von IPv6 beruhen auf dem Neighbor Discovery Protocol (NDP) und Multicast. Das heißt, dass in einem Netzwerk eine Gruppe von Empfängern adressiert werden kann. In einem Ethernet-Netzwerk mit Switches ist das ein Problem, weil ein Switch versucht, eine Zuordnung zwischen Switch-Port und Hardware-Adresse herzustellen. Das heißt, ein Switch merkt sich, welcher Host an welchem Port hängt und schickt die Datenpakete nur an die entsprechenden Ports. Was aber, wenn der Switch ein Multicast-Paket und die Antwort-Pakete bekommt? Welche Adressen speichert er in seiner Address-Table?

Moderne Switche merken sich, wer zu welcher Multicast-Gruppe gehört. Das bedeutet, ein IPv6-Host muss mitteilen, an welchen Multicasts er teilnehmen will.

IPv6-Header

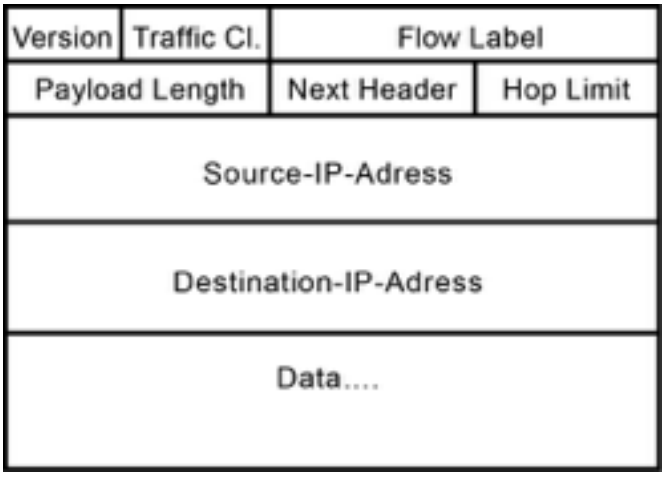
und Extension Headers

Jedes IPv6-Datenpaket besteht aus einem Header (Kopf) und dem Payload, in dem sich die Nutzdaten befinden. Der Header ist den Nutzdaten vorangestellt.



Bei IPv6 ist der Header auf einfache Verarbeitung optimiert. Konkret bedeutet das, dass er aus einem Bereich mit einer festen Länge von 40 Byte besteht, der nur Informationen enthält, die für das IP-Routing wichtig sind. Optionale Informationen sind in einem oder mehreren Extension Headers ausgelagert, die sich zwischen Header und Payload befinden. Allerdings dürften die meisten IPv6-Pakete ohne Extension Headers auskommen.

Aufbau des IPv6-Headers



Um die Verarbeitung der IPv6-Pakete zu vereinfachen wurde die Länge des IPv6-Headers auf 40 Byte fest definiert. Optionale Informationen werden in die Extension-Headers verlagert. Der feste Teil des IPv6-Headers enthält unter anderem die IPv6-Adresse von Sender und Empfänger. Das IPv4-Feld Time-to-live (TTL) ist das Hop Limit. Auf die Prüfsummenberechnung und Fragmentierung wird verzichtet, was den IPv6-Routern das Leben erleichtert.

Bedeutung der Felder im IPv6-Header

- Version (4 Bit): Hier ist die Version des IP-Protokolls abgelegt, nach der das IP-Paket erstellt wurde.
- Traffic Class (8 Bit): Der Wert des Feldes definiert die Priorität des Paketes.
- Flow Label (20 Bit): Das Flow Label kennzeichnet Pakete für ein viel schnelleres Routing. Das MPLS macht dieses Verfahren allerdings überflüssig.
- Payload Length (16 Bit): Hier steht die im IP-Paket transportierten Daten in Byte. Bisher musste der Wert aus dem Feld Paketlänge abzüglich dem Feld IHL ermittelt werden.
- Next Header (8 Bit): Hier ist das übergeordnete Transportprotokoll angegeben. Bei IPv4 hieß das Feld einfach Protokoll.
- Hop Limit (8 Bit): Dieses Feld enthält die Anzahl der verbleibenden weiterleitenden Stationen, bevor das IP-Paket verfällt. Es entspricht dem TTL-Feld von IPv4. Jede Station, die ein IP-Paket weiterleitet, muss von diesem Wert 1 abziehen.
- Source-Address (128 Bit): An dieser Stelle steht die IP-Adresse der Station, die das Paket abgeschickt hat (Quell-IP-Adresse).
- Destination-Address (128 Bit): An dieser Stelle steht die IP-Adresse der Station, für die das Paket bestimmt ist (Ziel-IP-Adresse).
- IPv6-Header-Erweiterungen (jeweils 64 Bit bzw. 8 Byte): Im IPv6-Header können optional Informationen im separaten Header dem IP-Kopf angehängt werden. Bis auf wenige Ausnahmen werden diese Header-Erweiterungen von IP-Routern nicht beachtet.

Extension Headers bzw. Header Extensions

Optionale Header-Informationen sind in den sogenannten Extension Headers zwischen Header und Payload untergebracht. Das hat den Vorteil, dass es keine Größenbeschränkung für Optionen gibt und neue Optionen eingeführt werden können ohne den Header zu verändern. Die einzige Grenze liegt darin, dass ein Extension Header im Datenpaket Platz finden muss, was unter Umständen den Platz für den Payload reduziert, weil das IPv6-Paket in seiner Gesamtheit nur eine bestimmte Größe haben darf. Das heißt, Extension Headers verkürzen den Platz für die Nutzdaten.

Ein Extension Header wird vom Sender zwischen IPv6-Header und den Nutzdaten (TCP- oder UDP-Datagramm) eingefügt. Die Header können verkettet werden. Es muss aber eine bestimmte Reihenfolge eingehalten werden und jeder Header darf nur einmal vorkommen. Mit einer Ausnahme. Befindet sich ein Routing Header im Paket, darf davor ein weiterer Destination Options Header stehen.

Die Größe einer Header-Erweiterung muss ein Vielfaches von 64 Bit (8 Byte) betragen, um die Verarbeitung zu vereinfachen.

Extension Headers werden, von einer Ausnahme abgesehen, vom Ziel verarbeitet, was wesentlich effizienter ist. Der Hop-by-Hop Options Header muss von jedem Router auf dem Pfad zum Ziel ausgewertet werden. Wenn er vorhanden ist, muss er direkt als erstes auf den IPv6-Header folgen.

IPv6-Header-Extensions

Header-Erweiterungen tragen eine definierte Nummer, die sich in das bestehende System der Protokollnummern einfügt.

- Hop-by-Hop Options: 0
- Routing: 43
- Fragment: 44
- Authentication Header (AH): 51
- Encapsulation Security Payload (ESP): 50
- Destination Options: 60
- Mobility: 135
- Host Identity Protocol: 139
- Shim6 Protocol: 140
- No Next Header (Ende eines Header-Stapels): 59

Die Reihenfolge bei einer Verkettung entspricht die der Tabelle mit Ausnahme des Destination Options Header, der ein zweites mal vorkommen darf.

Alle Extension Header enthalten ein Next-Header-Feld, in dem der nächste Extension Header oder das darüberliegende Protokoll gekennzeichnet ist.

IPv6-Paket-Fragmentierung und MTU-Discovery

Die Fragmentierung (Aufteilung) von IPv6-Paketen ist immer dann notwendig, wenn dessen Größe für ein Übertragungssystem auf dem Weg vom Sender zum Empfänger zu groß ist. Dann muss es aufgeteilt (fragmentiert) werden. Dazu muss der Sender die

maximale Paketgröße, die MTU (Maximum Transmission Unit) genannt wird, ermitteln. Man bezeichnet das als MTU Discovery und ist Teil von ICMPv6.

In der Vergangenheit hatten Netzwerk-Administratoren die Angewohnheit in ihren Firewalls ICMP-Pakete für IPv4 zu sperren. In der Regel wegen eines seltsamen Sicherheitsverständnisses oder einfach aus Gewohnheit.

Ja, es ist richtig, Hacker können mittels Tools, die das Protokoll ICMP verwenden, fremde Netze ausspionieren und Sicherheitslücken entdecken. Das Sperren von ICMP macht auf den ersten Blick also Sinn. Doch damit hält man höchstens ein paar Skript-Kiddies auf. Ein guter Hacker findet auch die Sicherheitslücken, die durch das Sperren von ICMP verdeckt werden. Ein Administrator, der sein Netz versucht durch Sperren von ICMP zu schützen, hat unter Umständen richtige Sicherheitsprobleme und er versucht sie auf diese Art und Weise zu verschleinern.

Das Sperren von ICMPv6 hat bei IPv6 zur Folge, dass dann MTU Discovery nicht mehr funktioniert. Doch bei IPv6 ist MTU Discovery Pflicht, weil die Fragmentierung bei IPv6 nicht mehr durch die Router, sondern durch den Client erfolgt. Und der Client kann nur dann von der Notwendigkeit der IP-Paket-Fragmentierung erfahren, wenn er Rückmeldung per ICMPv6 bekommt.

Wenn ein IPv6-Router Datenpakete zerschneiden muss, dann verwirft er sie und schickt eine Mitteilung per ICMP an den Absender zurück. Mit der Fehlermeldung, dass das Paket zu groß ist und wie groß das Paket maximal sein darf. Der Client schickt dann wieder ein Paket in der entsprechenden Größe, in der Hoffnung, dass es jetzt durchgeht. Wenn auf dem Weg zum Empfänger ein IPv6-Router ein noch kleineres Paket möchte, dann schickt auch der eine ICMP-Meldung zurück und der Client muss das Paket erneut verkleinern. So lange, bis das Paket erfolgreich beim Empfänger ankommt.

MTU Path Discovery muss bei IPv6 eingeschaltet sein. Schaltet man MTU Path Discovery ab, dann funktioniert IPv6 in der Regel in öffentlichen Netzen nicht mehr. Wenn die IPv6-Paket zu groß sind, dann kommen sie einfach nicht an.

NDP

Neighbour Discovery Protocol

Das Neighbor Discovery Protocol, kurz NDP, ist ein IPv6-Protokoll zum Austausch link-lokaler Nachrichten wie Router Discovery und Neighbor Discovery. NDP-Nachrichten sind Bestandteil von ICMPv6 und dürfen nicht in andere Netze gelangen. NDP vereint die Funktionen von ARP, RARP und IGMP bei IPv4, und erfüllt noch weitere Aufgaben.

Aufgaben von NDP

- Router- und Präfix-Ermittlung (Router Discovery und Prefix Discovery)
- Parameterermittlung (Parameter Discovery, z.B. MTU und Hop Limit)
- Adress-Autokonfiguration (Stateless Address Autoconfiguration, SLAAC)
- Adressauflösung (Address Resolution mit Neighbor Discovery)
- Bestimmung des nächsten Hops
- Erkennung der Nichterreichbarkeit des Nachbarn (Neighbor Unreachability Detection, NUD)
- Erkennung doppelter Adressen (Duplicate Address Detection, DAD)
- Umleitung (Redirect)

RD - Router Discovery

Bei Router Discovery handelt es sich um Verfahren, um die Hosts im Link-Local-Scope über die Anwesenheit eines Routers zu informieren.

Im Rahmen der Router Discovery mit Router Advertisement (RA) und Router Solicitation (RS) findet eine Prefix Discovery statt, der die Präfixe für die IPv6-Autokonfiguration (Stateless Address Autoconfiguration, SLAAC) der globalen IPv6-Adresse verteilt. Im Anschluss findet eine Duplicate Address Detection (DAD) statt.

Ein Router kann die Hosts über diesen Router Advertisement auch anweisen, IPv6-Adressen und weitere Teile der IP-Konfiguration über DHCPv6 zu beziehen.

Neighbor Discovery und Inverse Neighbor Discovery

Im Rahmen der Neighbor Discovery gibt es die Verfahren Duplicate Address Detection (DAD), Neighbor Unreachability Detection (NUD) und die Adressauflösung (Address Resolution).

Die Adressauflösung sorgt für die Zuordnung einer MAC- oder Hardware-Adresse zu einer IPv6-Adresse. Bei IPv4 findet die Zuordnung zwischen IP- und MAC-Adresse mittels ARP

bzw. RARP statt. Bei IPv6 ist das Neighbour Discovery Protocol (NDP) für die Adressauflösung zuständig.

Jeder IPv6-Node betreibt dazu einen Neighbor Cache in dem er die Ergebnisse der Link-Layer-Adressauflösungen zwischenspeichert. Diese Einträge sind je nach Betriebssystem und Zustand unterschiedlich lange gültig. Im Normalfall sind sie sehr kurzlebig.

Im Rahmen der Neighbor Discovery dienen Neighbor Solicitation und Neighbor Advertisement dazu, um Netzwerk-Nachbarn und Router zu bestimmen und link-lokale Adressen aufzulösen.

DAD - Duplicate Address Detection

Bei der Duplicate Address Detection, kurz DAD, fragt der Host per Neighbour Solicitation im LAN, ob andere Geräte bereits die gewählte Adresse nutzen. Bereits verwendete Adressen melden IPv6-Geräte per Neighbour Advertisement als belegt.

NUD - Neighbor Unreachability Detection

Bei der Neighbor Unreachability Detection, kurz NUD, geht es um die Erkennung der Nichterreichbarkeit der Nachbarn im Link-Local-Scope. Dabei werden einzelne IPv6-Adressen überprüft, ob sie on-link oder off-link sind.

ICMPv6

Internet Control Message Protocol Version 6

Das Internet Control Message Protocol Version 6 (ICMPv6) ist Bestandteil des Internet Protocols Version 6 (IPv6). Es wird aber als eigenständiges Protokoll behandelt, das zur Übermittlung von Meldungen über IP dient. Hauptaufgabe von ICMP ist die Übertragung von Statusinformationen und Fehlermeldungen der Protokolle IP, TCP und UDP.

ICMPv6 wird verwendet, um NDP-Nachrichten, wie Router Discovery und Neighbor Discovery zu verschicken. NDP umfasst die Funktionen von ARP, RARP und IGMP als Teil von IPv4. Für IPv6 sind das in NDP Neighbor Discovery, Inverse Neighbor Discovery und Multicast Listener Discovery (MLD).

ICMPv6 spielt eine wichtige Rolle für die Funktionsweise von IPv6-Verbindungen. Im Vergleich zu IPv4 dürfen ICMPv6-Datenpakete nicht einfach so blockiert werden.

Aufbau des ICMPv6-Headers

Der ICMPv6-Header besteht aus mindestens drei Feldern:

- 8 Bit für den Typ der NDP-Nachricht
- 8 Bit für den Code der NDP-Nachricht
- 16 Bit für die Prüfsumme des ICMPv6-Datenpakets

Die Prüfsumme wird über die gesamte ICMPv6-Nachricht und einem Pseudoheader gebildet. Der Pseudoheader besteht aus Quell- und Zieladresse, sowie der Länge des ICMPv6-Datagramms und dem Next-Header-Eintrag.

ICMPv6-Nachrichten-Typen

- Router Advertisement (ICMPv6-Typ 134)
- Router Solicitation (ICMPv6-Typ 133)
- Neighbor Advertisement (ICMPv6-Typ 136)
- Neighbor Solicitation (ICMPv6-Typ 135)
- Redirect (ICMPv6-Typ 137)

RA - Router Advertisement

Router Advertisements (RA) sind ICMPv6-Nachrichten vom Typ 134, mit denen sich Router im Netz bekanntmachen, Routing-Informationen verbreiten und Informationen für die IP-Autokonfiguration (Stateless Address Autoconfiguration, SLAAC) verteilen.

Dazu sendet ein Router in regelmäßigen Abständen ein sogenanntes Router Advertisement an die Multicast-Adresse "ff02::1". Davon fühlen sich alle Hosts im Link-Local-Scope angesprochen, die gegebenenfalls ihre IPv6-Konfiguration aktualisieren. Auf diese Weise erfahren alle Hosts die Adresse des Default-Routers und den globalen Präfix.

Router Advertisements lassen sich auch per Router Solicitation durch einzelne Hosts erzwingen.

RS - Router Solicitation

Router Solicitations (RS) sind ICMPv6-Nachrichten vom Typ 135, mit denen ein Host um einen Router Advertisement bittet.

Zwar sendet der Router periodisch Router-Advertisement-Nachrichten. Doch muss ein Host nicht darauf warten, sondern kann ein Router Advertisement erzwingen in dem er ein Router Solicitation senden. Der Router kann darauf mit einer Multicast-Nachricht an alle oder mit einer Unicast-Nachricht an den anfragenden Host antworten.

Neighbor Advertisement und Neighbor Solicitation (Neighbor Discovery)

Neighbor Advertisements sind ICMPv6-Nachrichten vom Typ 136, bei denen es sich um Antworten auf Neighbor Solicitations handelt. Neighbor Solicitation sind ICMPv6-Nachrichten vom Typ 137, bei denen es sich um Nachrichten mit der Bitte um Antwort handelt. Neighbor Advertisement und Neighbor Solicitation werden im Rahmen der Duplicate Address Detection (DAD), Neighbor Unreachability Detection (NUD) und Adressauflösung ausgetauscht.

Neighbor-Cache

Im Rahmen der Neighbor Discovery mit Neighbor Solicitation und Neighbor Advertisement entsteht der Neighbor-Cache. Das ist eine Liste von Netzwerkbekanntschaften aller Netzwerkschnittstellen, die ein Betriebssystem anlegt. Hier sind alle Rechner verzeichnet zu denen in letzter Zeit eine Verbindung bestand.

Der Neighbor-Cache entspricht dem ARP-Cache unter IPv4.

Mit den folgenden Befehlen kann man sich den Neighbor-Cache des eigenen Rechners anschauen:

Windows: netsh interface ipv6 show neighbors

Linux: ip -6 n s

Auf einem neu gestarteten Rechner ist diese Liste sehr kurz und enthält in der Regel nur die Adresse des nächsten IPv6-Routers im Link-Local-Scope.

Dank der Multicast-Adresse "ff02::1" kann man mit einem "ping" alle im Link-Local-Scope laufenden IPv6-Geräte herausfinden (Neighbor Solicitation).

Windows: ping -6 ff02::1

Linux: ping6 -c 5 ff02::1

Alle Systeme am Link-Local-Scope versenden dann Antwortpakete (Neighbor Advertisement), sofern die Geräte die Neighbor Solicitation erhalten haben. Dazu müssen sie mit dem LAN verbunden sein. Und eine Firewall oder eine andere Sicherheitsmaßnahme darf ICMPv6-Pakete nicht blockieren.

Anschließend finden sich alle Systeme, die geantwortet haben, im Neighbor Cache wieder. Diese Liste kann für weitere Untersuchungen im Link-Local-Scope nützlich sein.

Übergangsverfahren von IPv4 auf IPv6

Transition Strategy

IPv6 ist in aller Munde. Denn der weltweite IPv4-Adresspool ist seit Anfang 2012 erschöpft. Wer öffentliche IP-Adressen braucht, der muss sich einen IPv6-Präfix besorgen. Leider hat man damit das Problem nicht gelöst. Mit der praktischen Umsetzung hapert es, weil es unmöglich ist alle Netzwerk-Geräte auf einmal IPv6-fähig zu machen. Damit der Wechsel leichter geht und Investitionen in alte IPv4-Technik nicht gleich für die Tonne sind, gibt es verschiedene Übergangsverfahren, die ein Teil einer "Transition Strategy" sein können.

- Tunneling (z. B. Teredo, 6in4, 6to4, 6over4, DS Lite)
- Parallelbetrieb (z. B. Dual-Stack)
- Protokollübersetzung (z. B. NAT64)

Transition Strategy

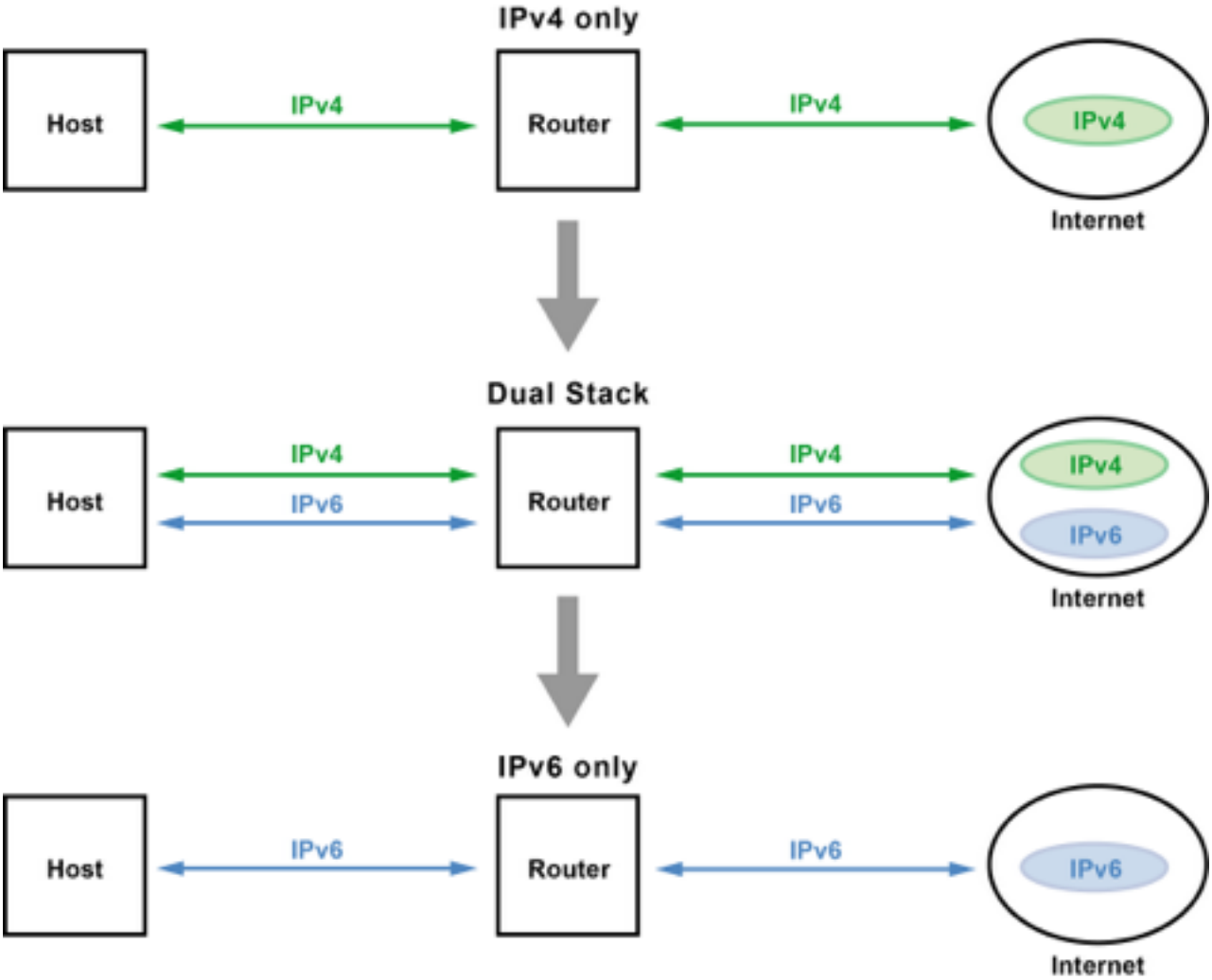
Das Hauptproblem bei der "Transition" ist, dass der Wechsel von IPv4 auf IPv6 nicht auf einen Schlag, sondern über Jahrzehnte erfolgt. Das bedeutet auch, dass auf der ganzen Welt von diesem Wechsel die Internet-Nutzer unterschiedlich stark betroffen sind. Das heißt, während in Europa nicht zwingend IPv6 eingeführt werden muss, und deshalb die Einführung etwas schleppend erfolgt, sieht das in Afrika und Asien ganz anders aus. Dort bleibt Netzbetreibern, Providern und Diensteanbietern nichts anderes übrig als ihre Server mit IPv6-only zu betreiben, weil es keine IPv4-Adressen mehr gibt.

Aber was macht man, wenn man in Europa als IPv4-only-Internet-Nutzer auf eine solche Webseite zugreifen möchte. Oder umgekehrt, ein asiatischer IPv6-only-Internet-Nutzer will auf eine IPv4-only-Webseite zugreifen? In beiden Fällen ist keine Verbindung möglich. Was macht man dann? Eine "Transition Strategie" sollte darauf eine Antwort geben können.

Wie sieht eine technische Lösung aus, wenn man nur IPv6 hat und auf eine Webseite zugreifen möchte, die nur IPv4-only erreichbar ist? Oder umgekehrt. Wie sieht eine technische Lösung aus, wenn da nur IPv4 ist, aber eine Webseite nur per IPv6 erreichbar ist?

Solche Szenarien scheinen heute vielleicht eher unwahrscheinlich. Aber niemand kann genau sagen, wann sich das ändert und wann diese Szenarien zum Problem werden. Wer

dann kein IPv6 hat, der ist plötzlich von seinen Kunden abgehängt. Und dann ist die Frage, bis wann man IPv6 stabil eingeführt bekommt?



Für jedes Szenario gibt es unterschiedliche Mechanismen, die man teilweise nur aktivieren und teilweise von Hand konfigurieren muss. Übergangsverfahren spielen dabei eine wichtige Rolle. Da in Zukunft immer mehr Internet-Service-Provider und Unternehmen auf echtes IPv6 umstellen, dürften die Übergangsverfahren nicht ganz verschwinden, aber zumindest abnehmen. Generell sollten alle Übergangsverfahren als vorübergehende Lösung auf dem Weg zu IPv6-only gesehen werden. Bis zur vollständigen Nutzung von IPv6 werden noch einige Jahre ins Land gehen.

Tunneling / Tunnelung

Wenn ein Client kein IPv6 hat, der Server aber nur darüber erreichbar ist, dann kommen Tunneling-Verfahren, zum Beispiel Teredo (Microsoft) oder 6to4/6over4, zum Einsatz.

Wenn bereits das eigene Netzwerk IPv6 unterstützt, aber der Service-Provider am Internet-Anschluss noch nicht, können die IPv6-Clients über einen Tunnel durch das IPv4-Netz mit IPv6-Servern kommunizieren. Dafür benötigt man einen Tunnel-Provider,

der die lokalen IPv6-Pakete über das IPv4-Internet ins IPv6-fähige Internet routet. Dafür gibt es verschiedene Tunneling-Verfahren.

- 4in6: Tunneling von IPv4 in IPv6
- 6in4: Tunneling von IPv6 in IPv4
- 6to4: Transport von IPv6-Datenpaketen über ein IPv4-Netzwerk
- 6over4: Transport von IPv6-Datenpaketen zwischen Dual-Stack Knoten über ein IPv4-Netzwerk
- Dual-Stack Lite: Dual-Stack mit globaler IPv6 und Carrier-NAT-IPv4
- Teredo: Kapselung von IPv6-Datenpaketen in IPv4-UDP-Datenpaketen

Protokollübersetzung (DNS64 und NAT64)

Bei DNS64 und NAT64 geht es darum mit einem IPv6-Client auf einen IPv4-Server zuzugreifen. Es findet praktisch eine Übersetzung zwischen internen IPv6-Adressen und externen IPv4-Adressen statt. Dabei fragt der IPv6-Client einen DNS64-Server nach der IPv6-Adresse des IPv4-Servers. Weil der Server noch keine IPv6-Adresse hat konvertiert der DNS64-Server die IPv4-Adresse des Servers in eine IPv6-Adresse, in etwa wie bei 6over4 und 6to4. Anschließend teilt der DNS64-Server dem Client die IPv6-Adressse mit, der die IPv6-Pakete zum NAT64-Gateway schickt. Das NAT64-Gateway, im Dual-Stack-Betrieb, erkennt in der IPv6-Adresse die IPv4-Adresse, generiert ein neues IPv4-Paket und leitet es an den IPv4-Server weiter. Die Antwort-Pakete nehmen den umgekehrten Weg zurück.

Der Vorteil der Protokollübersetzung mit DNS64 und NAT64 ist, dass der IPv6-Client nicht wissen muss, dass er eine Verbindung zu einem IPv4-Server unterhält.

Der Nachteil ist, dass man zwingend öffentliche IPv4-Adressen braucht, die unter Umständen nicht zur Verfügung stehen. Wie für NAT typisch werden mittels Portnummern die einzelnen internen IPv6-Clients unterschieden. Auf diese Weise kann man dann wiederum IPv4-Adressen einsparen.

Fazit

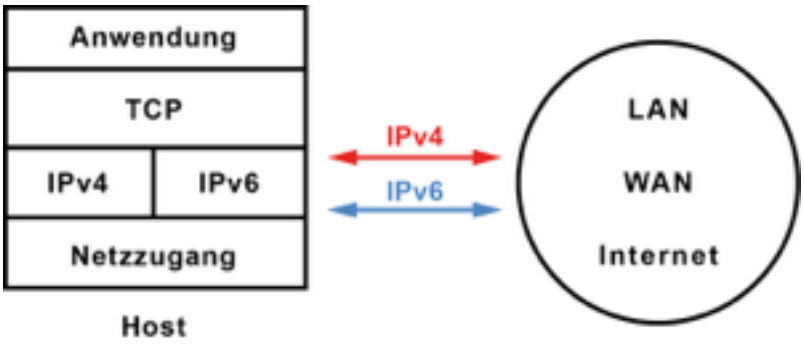
Momentan dominieren die Verfahren, die es ermöglichen IPv6-Pakete in IPv4-Netzen zu übertragen. Verfahren, die IPv4-Pakete über IPv6-Netze übertragen spielen eine immer größere Rolle.

Alle Übergangsverfahren von IPv4 auf IPv6 sind als Übergangslösung anzusehen. Beim Wechsel von IPv4 auf IPv6 ist eines entscheidend: Entweder man hat/bekommt vom Internetanbieter eine native IPv6-Verbindung (Dual-Stack) oder man realisiert es über einen IPv6-Tunnel.

Dual Stack

Parallelbetrieb von IPv4 und IPv6

Mit Dual Stack bezeichnet man den Parallelbetrieb von IPv4 und IPv6. Da keine direkte Umstellung von IPv4 auf IPv6 möglich und auch nicht sinnvoll ist, sieht eine "Transition Strategy" vor, dass alle Netzknoten sowohl IPv4 als auch IPv6 beherrschen. Längerfristig würde man dann auf IPv4 verzichten können. Ausgenommen da, wo bereits heute kein IPv4 mehr möglich ist, weil keine öffentliche IPv4-Adressen mehr verfügbar sind. Hier fährt man bereits IPv6-only.



Die Migration zu Dual Stack ist vergleichsweise einfach. Viele Betriebssysteme können mit Dual Stack, also IPv4 und IPv6 gleichzeitig, umgehen. Alle bestehenden Dienste sind weiterhin unter ihrer gewohnten IPv4-Adresse erreichbar. Nach und nach kann man bestehende Dienste per IPv6 erreichbar machen.

Der Schritt zu IPv6 und damit Dual Stack, wird nur häufig deshalb nicht vollzogen, weil während des Parallelbetriebs der doppelte Administrationsaufwand anfällt. Beispielsweise müssen statische IP-Konfigurationen und das Routing, Filterregeln und Access Control Lists doppelt geführt werden. Und das bedeutet auch, es gibt die doppelte Anzahl an Fehlerquellen.

Erschwerend kommt hinzu, dass man es in der Regel mit Mitarbeitern, Dienstleistern und Experten zu tun hat, die vor Jahren oder Jahrzehnten IPv4 gelernt und verinnerlicht haben. Die müssen IPv6 völlig neu lernen. Denn IPv4 und IPv6 sind in vielen Dingen vergleichbar, aber nicht identisch.

Das Problem der Adressknappheit bei IPv4 löst man aber nicht dadurch, dass man IPv6 ignoriert. Irgendwann kommt man an den Punkt, an dem neue Dienste per IPv6 erreichbar sein müssen, weil die Kunden nur noch eine eingeschränkte oder gar keine IPv4-Connectivity mehr haben.

Alle Lösungswege im Hinblick auf die aktuellen Probleme mit IPv4/IPv6 sollten in Richtung IPv6-only führen. Egal welche Erfahrungen der eine oder andere gemacht hat oder noch machen wird. An IPv6 führt kein Weg vorbei. Auf dem Weg Richtung IPv6-only

wird Dual Stack, also der Parallelbetrieb von IPv4 und IPv6, der Normalfall sein. Auf Jahrzehnte gesehen. Wie genau das Dual Stack aussieht, hängt aber vom Provider ab.

Address Selection

Im klassischen Dual-Stack-Betrieb hat ein Host oder Netzknoten sowohl eine globale IPv6-Adresse als auch eine öffentliche IPv4-Adresse. Ob ein Host für ausgehende Verbindungen IPv4 oder IPv6 verwendet, hängt dann vom Betriebssystem und der Konfiguration ab. Sobald ein Client eine globale IPv6-Adresse hat, sollte er gemäß RFC 6723 (IPv6 Support required) eine Verbindung per IPv6 bevorzugen. Das ist aber nicht grundsätzlich so. Inzwischen macht man es davon abhängig, welche Verbindung schneller ist. Das ist deshalb so, damit der Nutzer und auch die Anwendung immer die beste Verbindung bekommen.

Voraussetzung für IPv6 ist, dass der DNS-Server eine IPv6-Adresse bei der Namensauflösung zurückliefert (AAAA-Record). Ist der Server über IPv6 nicht erreichbar, wird der Client es mit der IPv4-Adresse versuchen. Wenn nur eine IPv4-Adresse per DNS geliefert wird, scheidet IPv6 aus und IPv4 kommt zum Zug.

Dual-Stack-PPPoE mit TR-187 (RFC 4241)

Die Aushandlung der IP-Konfigration per Dual-Stack-PPPoE mit der Spezifikation TR-187 (RFC 4241) erfolgt sowohl für IPv4 als auch für IPv6. Wenn ein Internet-Zugangsrouter beim Teilnehmer kein IPv6 beherrscht, dann fällt dieser Teil einfach weg.

Im Dual-Stack-Modus weist der Access-Server (des Netzbetreibers) dem Internet-Zugangsrouter (beim Kunden) für die WAN-Seite eine IPv4-Adresse und ein IPv6-Subnetz (typischerweise /64) bzw. einen globalen Präfix zu. Beides wird per Router Advertisement (RA) im Rahmen einer Dual-Stack-PPPoE-Sitzung vergeben.

Anschließend lässt sich der Zugangsrouter per Präfix Delegation ein weiteres IPv6-Subnetz zuteilen. Es handelt sich dabei um ein /56-Präfix, mit dem sich 256 IPv6-Subnetze adressieren lassen. Diesen Präfix (/56) reicht der Zugangsrouter per Router Advertisement oder DHCPv6 an die lokalen Hosts weiter, damit die sich per Stateless Address Autoconfiguration (SLAAC) eine globale IPv6-Adresse generieren können. Weitere IPv6-Parameter holt sich der Zugangsrouter per DHCPv6.

Für die Zukunft ist es wichtig, dass ein Internet-Zugangsrouter in einer Dual-Stack-PPPoE-Sitzung die Konfiguration für IPv4 und IPv6 vornehmen kann.

Dual-Stack-Problem

Generell ist es so, dass wenn ein Client sowohl IPv4 als auch IPv6 zur Verfügung hat (Dual-Stack), dann wird der Client parallel mit IPv4 und IPv6 arbeiten. Dabei wird er versuchen herauszufinden über welche IP-Version er eine bessere Verbindung bekommt. Es kann aber auch sein, dass er Verbindungen über IPv6 generell bevorzugt. Auch dann, wenn "keine zuverlässige IPv6-Verbindung zur Gegenstelle" besteht! Das hat dann zur

Folge, dass ein Client zu einem Server keine Verbindung aufbauen kann, obwohl beide Seiten eine funktionierende Verbindung haben. Man bezeichnet das dann als Dual-Stack-Problem.

Die Ursachen für das Verbindungsproblem können vielfältig sein. Beispielsweise kann der IP-Stack des Betriebssystems einen Fehler aufweisen, der Internet-Zugangsrouter fehlerhaft konfiguriert sein oder der lokale Router weist den Clients eine IPv6-Adresse zu, obwohl er selber IPv6 gar nicht routen kann (unvollständige IPv6-Implementierung). Auch eine Fehlkonfiguration oder Störung beim Internet-Service-Provider kann die Ursache sein. In so einem Fall versucht der Client eine IPv6-Verbindung aufzubauen und läuft in einen Timeout.

In dieser Situation kann der betroffene Anwender nicht erkennen, dass das Problem im eigenen Netz oder bei seinem Provider liegt, sondern unterstellt der betreffenden Webseite Nichterreichbarkeit. Betreiber von über IPv6 erreichbare Webseiten bekommen von diesen erfolglosen Zugriffsversuchen der Besucher nichts mit. Die können auch gar nichts dafür.

Wären beispielsweise Google oder Facebook wegen der beispielhaft beschriebenen Ursachen scheinbar offline, würden betroffene Nutzer früher oder später auf andere Plattformen wechseln.

Hinweis: Vom Dual-Stack-Problem sind nur ganz wenige Internet-Nutzer betroffen. Doch wenn, dann ohne es zu wissen. Und auch die Server-Betreiber bekommen davon nichts mit.

Was ist Dual-Stack Lite (DS-Lite / DSlite)?

Dual Stack Lite ist eine Betriebsart für einen Breitband-Anschluss mit globalem IPv6-Präfix und Carrier-Grade-NAT-IPv4-Adresse, wobei der IPv4-Datenverkehr in IPv6-Paketen getunnelt wird.

IPv6-Tunneling

mit 6in4 / 6to4 / 6over4 / 4in6

Leider bieten einige Internet-Provider ihren Kunden kein IPv6 an. In so einem Fall muss man auf IPv6 verzichten oder sich um eine Tunneling-Lösung bemühen. Ein Tunnel (IPv6 in IPv4) ist allerdings keine dauerhafte, sondern nur eine Notlösung. Abhängig von der Tunnel-Technik gibt es verschiedene Probleme, weshalb Tunneling nur kurz- oder höchstens mittelfristig in Frage kommt.

Wenn innerhalb des eigenen lokalen Netzwerks IPv6 funktioniert, aber am Internet-Anschluss noch nicht, dann können die IPv6-Clients über einen Tunnel durch das IPv4-Netz mit IPv6-Servern kommunizieren. Dafür benötigt man in der Regel einen Tunnel-Provider, der die lokalen IPv6-Pakete über das IPv4-Internet ins IPv6-fähige Internet routet. Dafür gibt es verschiedene Tunneling-Verfahren.

Tunneling-Verfahren, wie 6in4 oder 6to4, machen natürlich nur dann Sinn, wenn der eigene Netzzugangsprovider noch keinen Dual-Stack-Zugang anbietet, sondern nur IPv4. Da IPv6 sich früher oder später sowieso durchsetzen wird, sind Tunneling-Verfahren nur eine vorübergehende Lösung. Längerfristig wird man immer natives IPv6 anstreben, mittelfristig auf den Parallelbetrieb (Dual-Stack) setzen und vorübergehend Tunneling verwenden.

Funktionsweise von Tunneling

Die Grundidee von IPv6 ist, dass es über alle Übertragungssysteme funktioniert. Beispielsweise auch über IPv4. In der Regel sind zwei Router die Endstellen für den Tunnel. Der eigene Netzzugangsrouter und das IPv6-Gateway oder -Relay des Tunnel-Providers. Doch auch Clients können eigene Tunnelverbindungen aufbauen.

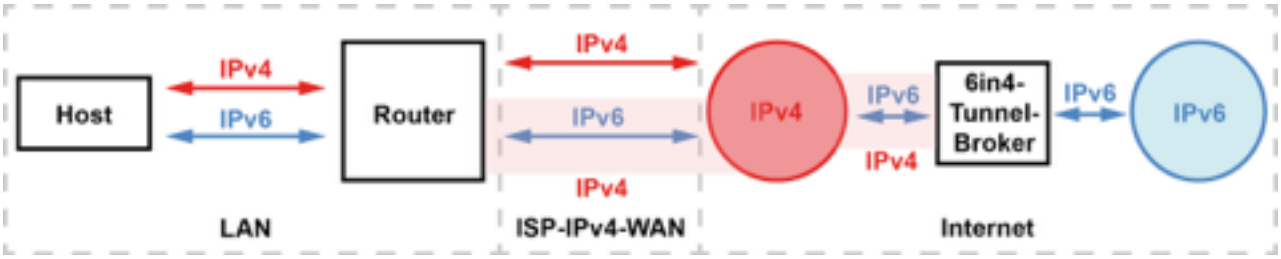
Bei diesem Szenario muss man berücksichtigen, dass mindestens ein NAT-Router und die gesamte IPv4-Infrastruktur unter dem IPv6-Tunnel liegt. Die Latenz dieser Infrastruktur muss zur Roundtrip-Zeit der getunnelten IPv6-Pakete hinzugerechnet werden. Das kann für die eine oder andere Anwendung ein Problem sein.

Dann muss man berücksichtigen, dass die Komplexität der Infrastruktur steigt und unter Umständen die Fehlersuche erschwert. Außerdem kann ein IPv6-Tunnel nur dann funktionieren, wenn auch IPv4 geht.

Deshalb eignet sich Tunneling nur als vorübergehende Übergangslösung hinzu Dual-Stack mit nativem IPv6. Für einen langfristigen und produktiven Betrieb ist ein IPv6-Tunnel nicht geeignet.

Einen großen Vorteil hat das Tunneling von IPv6 dann aber doch. In der Regel ist keine Änderung an der bestehenden Infrastruktur notwendig.

6in4-Tunneling



Beim Tunneling mit "6in4" müssen die Clients im lokalen Netzwerk über IPv6-Adressen verfügen und auch die Server im Internet unter einer IPv6-Adresse erreichbar sein. Zwischen dem Client und dem Server müssen sich zudem zwei Router oder Gateways im Dual-Stack-Betrieb befinden. Also sowohl IPv4 als auch IPv6 beherrschen. Der erste Dual-Stack-Router generiert IPv4-Pakete, in denen sich die IPv6-Pakete befinden. Über das IPv4-Netz gelangt das IPv4-Paket zum letzten Dual-Stack-Router, der das IPv6-Paket aus dem IPv4-Paket herausholt und an den IPv6-fähigen Server weiterleitet.

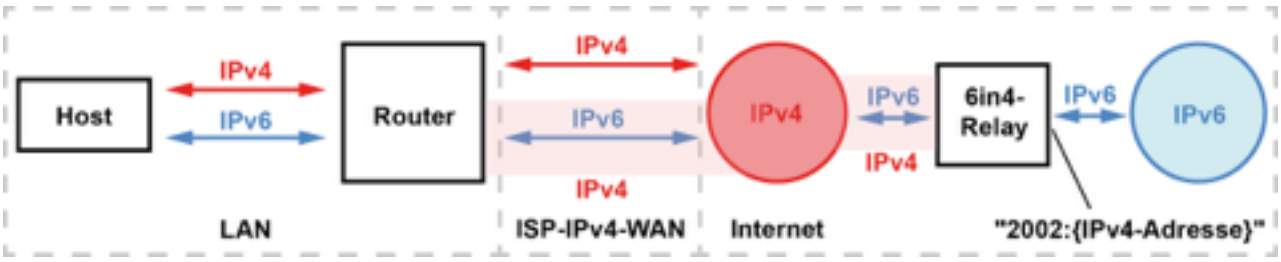
Konkret heißt das, man sucht sich jemanden, der IPv6-Connectivity hat. Zum Beispiel einen Tunnel-Provider oder Tunnel-Broker. Mit dem einigt man sich, wie der Tunnel zustande kommen soll (Protokoll) und lässt sich anschließend einen Präfix zuteilen (Adressvergabe).

4in6-Tunneling

Bei "4in6" wird das Prinzip von "6in4" umgekehrt. Es ist dann notwendig, wenn man IPv4-Pakete in einem IPv6-Netz transportieren will. Dazu benötigt man jeweils einen IPv4-Client und IPv4-Server und zwei im Dual-Stack-Betrieb befindliche Router. Beim Dual-Stack-Router auf der Client-Seite wird das IPv4-Paket in ein IPv6-Paket gepackt, über das IPv6-Netz übertragen und beim letzten Dual-Stack-Router wieder ausgepackt. Von dort wird das IPv4-Paket an den IPv4-Server weitergeleitet.

6to4-Tunneling

"6to4" ist ein Tunneling-Verfahren, dass nur dafür interessant ist, um IPv6-only Dienste zu erreichen. Und 6to4 macht nur dann Sinn, wenn man eine öffentliche IPv4-Adresse hat. In der Regel braucht man "6to4" nur zu aktivieren, wenn man Dienste ansprechen möchte, die nur IPv6 können.

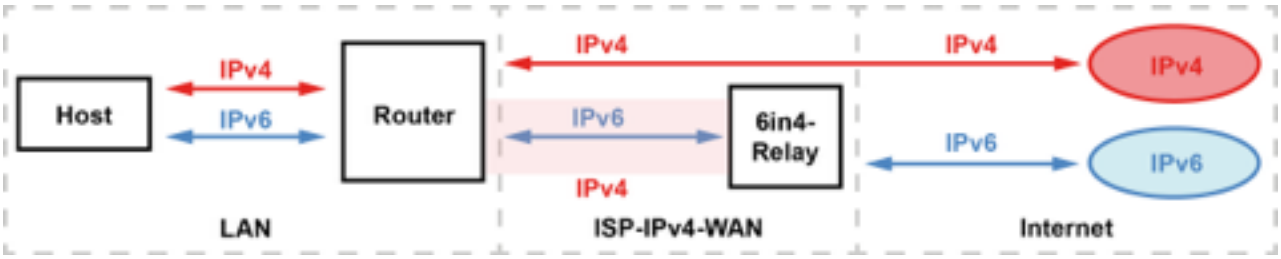


Für "6to4" ist der gesamte IPv6-Adressraum "2002::/16" reserviert. Dadurch ist es möglich alle öffentlichen IPv4-Adresse bei Bedarf in eine global gültige IPv6-Adresse umzuwandeln. Die IPv6-Adresse wird dabei in die Form "2002:{IPv4-Adresse}" gebracht. Die IPv4-Adresse wird in die hexadezimale Schreibweise umgewandelt. Anschließend kann der Paketinhalt in einem IPv6-Paket übertragen werden.

Das bedeutet, dass jeder, der eine öffentliche IPv4-Adresse hat automatisch auch eine IPv6-Adresse hat. Nachteil ist, ändert sich die IPv4-Adresse, dann ändert sich automatisch auch die IPv6-Adresse. Wenn man keine Dienste betreibt, ist das nicht weiter schlimm.

Die Frage ist wohin werden die IPv6-Pakete getunnelt? Gemeint ist, was ist die IPv4-Ziel-Adresse? Der Router tunnelt zu der IPv4-Anycast-Adresse "192.88.99.1". Das bedeutet, irgendwo gibt es ein Gateway, dass das IPv6-Paket aus dem IPv4-Paket auspackt und ins IPv6-Netz weiterleitet. Der Rückweg funktioniert genauso, nur umgekehrt. Das IPv6-Paket geht zu irgendeinem Gateway, dass das IPv6-Paket in ein IPv4-Paket packt und an die angegebene IPv4-Adresse, die in der IPv6-Adresse drinsteht weiterleitet.

Das bedeutet, dass die Pakete nie den selben Hin- und Rückweg nehmen, was ein bisschen problematisch ist.



Eine Variante von "6to4" ist "6rd" (IPv6 Rapid Deployment). Es eignet sich allerdings nur für typische Endkunden-Internet-Anschlüsse, da die IPv6-Adressen zentral vergeben werden.

"6to4" war eigentlich für den nahtlosen Umstieg auf IPv6 gedacht. Leider funktioniert es hinter Carrier-Grade-NAT (CG-NAT) nicht.

6over4-Tunneling

"6over4" funktioniert ähnlich wie "6to4". Bei "6over4" wird eine IPv6-Adresse in der Form "fe80::{IPv4-Adresse}" gebildet. Dabei wird die IPv4-Adresse in hexadezimaler

Schreibweise verwendet. Das IPv6-Paket mit der link-lokalen Adresse wird anschließend in IPv4 eingebettet und über ein IPv4-Multicast-Netzwerk übertragen.

Dieses Tunneling-Verfahren kommt allerdings wegen Durchsatz- und Sicherheitsproblemen nicht so häufig zum Einsatz. Die einfacheren und praktikableren Tunnelingverfahren "6in4" und "6to4" sind viel mehr verbreitet als "6over4".

AYIYA - Anything In Anything

Tunneling, wie 6in4 oder 6to4, führt an typischen Internet-Anschlüssen zu Problemen, wenn der Zugang über einen NAT-Router hergestellt wird. Die Address-Übersetzung von NAT und auch restriktive Firewalls, verhindern das Tunneling, wenn der Netzzugangsrouter das entsprechende Tunneling-Verfahren nicht beherrscht. Deshalb findet Tunneling per "6*4" nicht so häufig statt.

Was auch ein Problem ist, wenn ein Internet-Anschluss seine IPv4-Adresse dynamisch zugeteilt bekommt. Dann muss die neue IPv4-Adresse bei jedem Wechsel an den Tunnel-Endpunkt übermittelt werden. Dafür gibt es dann Tunnel-Management-Protokolle, wie zum Beispiel AYIYA (IPv4 in IPv6-UDP and IPv6 in IPv4-UDP).

Das Protokoll AYIYA ist ein Entwurf von SixXS und wird auch nur von dieser Organisation verwendet.

Um einen AYIYA-Tunnel aufzubauen braucht man eine Client-Software für den eigenen Rechner. Oder man hat einen Netzzugangsrouter, der einen solchen AYIYA-Tunnel aufbauen kann. Dort gibt man die Account-Daten des Tunnel-Providers ein und der loggt sich dann beim Tunnel-Provider ein, holt sich die Konfiguration und sagt dem Tunnel-Endpunkt auf der anderen Seite, wenn sich die IPv4-Adresse geändert hat.

Selbst wenn sich die IPv4-Adresse ändert, gibt es nur eine kurze Unterbrechung, sofern die Anwendung nicht in einen Timeout läuft.

Obwohl sich dann die IPv4-Adresse immer ändert, behält mein seinen eigenen IPv6-Präfix vom Tunnel-Provider.

Der Transport von IPv6-Paketen erfolgt nicht direkt in IPv4, sondern in UDP. Nur so kann man einen Tunnel in einer NAT-Umgebung betreiben. Gleichzeitig kann ein Tunnelendpunkt seine neue Adresse per AYIYA dem anderen Tunnelendpunkt mitteilen, ohne das die Stabilität des Tunnels beeinträchtigt wird.

Dual Stack Lite

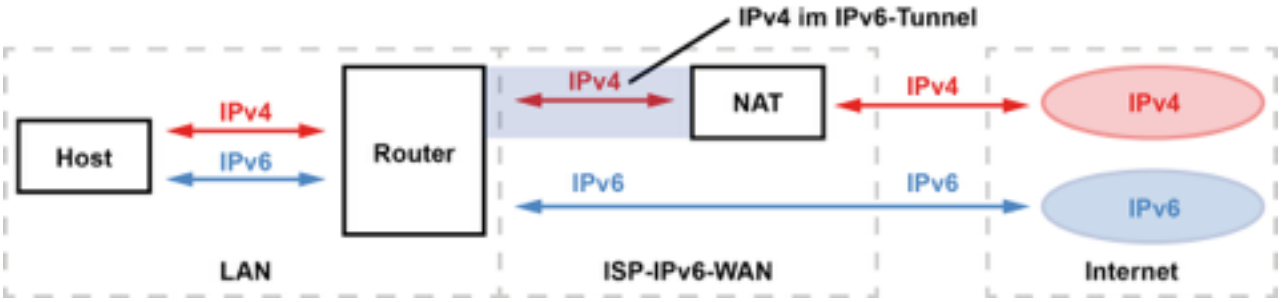
DS-Lite / DSlite

Dual Stack Lite (DS-Lite / DSlite) wird häufig als ein "IPv6 und mit privater IPv4-Adresse" definiert. Das ist so nicht richtig, weil DS-Lite ein definierter Standard für eine Tunnel-Technik (IPv4 in IPv6) ist. Die Bezeichnung DS-Lite ist für einen Internet-Anschluss nur dann korrekt, wenn eine globale IPv6-Adresse vorhanden ist und IPv4-Pakete in IPv6 getunnelt werden.

Irrtümlicherweise werden Internet-Zugänge mit globaler IPv6-Adresse und privater IPv4-Adresse als Dual Stack Lite bezeichnet. Dabei handelt es sich auch um ganz gewöhnliches Dual Stack.

Internet-Anschlüssen mit Dual Stack Lite betreiben typischerweise nur TV-Kabelprovider (in Deutschland).

Funktionsweise von Dual Stack Lite



Das kennzeichnende Merkmal von Dual Stack Lite ist, dass der Provider über kein IPv4-, sondern nur über ein IPv6-Netz verfügt. Darin bekommt jeder Teilnehmer einen globalen IPv6-Präfix zugeteilt. Um auch IPv4-Datenverkehr übertragen zu können, wird der ausgehende IPv4-Pakete im Endkunden-Router mit einer privaten IPv4-Adresse versehen und über einen 4in6-Tunnel ins öffentliche IPv4-Netz getunnelt. Dabei erhält jedes IPv4-Paket einen neuen IPv6-Header. Zwischen dem privaten Provider-Netz und dem öffentlichen IPv4-Netz vermittelt Carrier-Grade-NAT (CG-NAT). Der dafür zuständige NAT-Server steht beim Provider und kümmert sich um die Adressübersetzung zwischen den privaten und öffentlichen IPv4-Adressen und reicht die Pakete anschließend ins öffentliche IPv4-Netz weiter.

Ein weiteres Merkmal von Dual Stack Lite ist eine B4-Komponente im Endkunden-Router (CPE), die den IPv4-Datenverkehr per IPv6 zu einem AFTR-Device (Address Family Transition Router) transportiert, der den Datenverkehr ins IPv4-Internet weiterleitet. Wenn der Provider private IPv4-Adressen an seine Endkunden vergibt, dann wird beim Tunnel-Endpunkt des Providers per Carrier-Grade-NAT ins IPv4-Internet geroutet.

Probleme mit DS-Lite

Bei DS-Lite bekommt der Netzzugangsrouter des Kunden keine öffentliche IPv4-Adresse, sondern eine private IPv4-Adresse. Das bedeutet, der Kunde bekommt erst über Carrier Grade NAT eine öffentliche IPv4-Adresse. Das hat erhebliche Nachteile. Einige Netzwerk- und Internet-Dienste funktionieren bei Dual-Stack Lite nicht, wenn keine öffentliche IPv4-Adresse vergeben wurde. Ein echtes Problem ist das dann, wenn IPv6 von diesem Dienst noch nicht unterstützt wird.

Beispielsweise funktionieren IPv4-Port-Weiterleitungen am eigenen Internet-Router nicht mehr. Im LAN laufende Server lassen sich so nicht mehr per IPv4 aus dem Internet erreichen. Außerdem können auch Probleme bei VoIP (SIP), VPNs und Online-Gaming auftreten.

Das Problem dabei ist, dass keine öffentliche IPv4-Adresse am Internet-Anschluss zur Verfügung steht. Die konsequente Lösung für dieses Problem wäre, dass der jeweilige Dienst auf IPv6 umstellt oder der Provider seinem Kunden eine öffentliche IPv4-Adresse für seinen Internet-Zugang zur Verfügung stellt.

Dual Stack oder Dual Stack Lite?

Um zu wissen, ob man "Dual Stack" oder nur "Dual Stack Lite" hat, muss man neben der Frage, ob und wie IPv4 und IPv6 auf der Verbindung transportiert werden, auch klären, wo geNATet wird und wie die Adresszuweisung funktioniert. Ohne sich mit der Technik genau auszukennen ist es kaum möglich zu beurteilen ob man einen Dual-Stack-Lite-Anschluss hat oder nicht.

Was ist Dual Stack? Bei Dual Stack wird jeder Netzknoten mit öffentlicher oder privater IPv4-Adresse zusammen mit einem globalen IPv6-Präfix, also IPv4 und IPv6 parallel betrieben.

Was ist Dual Stack Lite? Dual Stack Lite ist eine Betriebsart für einen Breitband-Anschluss mit globalem IPv6-Präfix und Carrier-Grade-NAT-IPv4-Adresse, wobei der IPv4-Datenverkehr in IPv6-Paketen getunnelt wird.

Grundsätzlich bedeutet "Dual Stack", dass IPv4 und IPv6 auf der selben Verbindung parallel genutzt werden kann. Ob dabei NAT im Endkunden-Router stattfindet (typischerweise beim DSL-Anschluss) oder ob das NAT beim Mobilfunkprovider erfolgt, ist dabei völlig unerheblich. Da bei IPv4 in der Regel immer geNATet wird, macht man bei Dual Stack keine Unterscheidung, ob ein Anschluss eine private oder öffentliche IPv4-Adresse hat.

Als man für das Verfahren "Dual Stack Lite" den Namen gewählt hat, da hat man leider ignoriert, dass es eine große Vielfalt an Transition-Techniken und -Strategien gibt. Das hat dazu geführt, dass man irrtümlich davon ausgeht, dass wenn man per privater IPv4-Adresse über CG-NAT ins IPv4-Internet kommt, dass es sich dabei um "Dual Stack Lite"

handelt. Doch so einfach ist das nicht. Viele Internet-Zugänge werden über private IPv4-Adresse realisiert. Doch das ist dann nicht zwangsläufig DS-Lite.

Der Irrtum bzw. das Missverständnis hängt mit CG-NAT und der damit verbundenen Zuteilung einer privaten IPv4-Adresse zusammen. Egal wie der Endkunde zu seiner privaten IPv4-Adresse gekommen ist, die Auswirkungen für den Endkunden sind immer die selben. Fälschlicherweise nennt man das immer Dual Stack Lite.

Ob man jetzt Dual Stack Lite oder einfach nur Dual Stack hat, dafür gibt es eine Faustregel, die allerdings auch Ausnahmen hat. Damit die Bedingung für DS-Lite geben ist, muss das Endgerät beim Kunden IPv4 durch IPv6 tunneln. Das macht es nicht von sich aus. Dafür muss das Endgerät entsprechend ausgelegt und eingestellt sein. In der Regel überlassen Festnetz-Provider ihren Kunden die Endgerätewahl. Die oft billigen DSL-Router beherrschen diese Tunnel-Technik nicht und die Provider bieten es dementsprechend netzseitig auch nicht an. Die machen höchstens CG-NAT. Teilen also private IPv4-Adressen zu. Machen aber kein DS-Lite.

Im Mobilfunknetz gibt es in der Regel auch kein DS-Lite, weil hier vom Nutzer kein Router (CPE), sondern direkt ein Endgerät eingesetzt wird. Zum Beispiel ein Smartphone, ein Mobilfunk-Stick oder ein UMTS/LTE-Router. Egal welches Mobilfunkgerät, die beherrschen die Tunnel-Technik nicht. Und die Mobilfunknetzbetreiber bieten es in der Regel auch nicht an.

Im Mobilfunkbereich kommt typischerweise 464XLAT zur Anwendung, was in gewisser Weise Ähnlichkeiten zu DS-Lite aufweist. 464XLAT tunnelt IPv4-Datenverkehr ebenfalls vom mobilen Endgerät durch das IPv6-Mobilfunknetz. Das ist aber eine ganz andere Technik, die mit DS-Lite nichts zu tun hat, obwohl das so ähnlich aussieht und deshalb irrtümlich DS-Lite genannt wird.

Bleiben noch die TV-Kabelprovider, die in der Regel ihren Kunden das Endgerät stellen (mit oder ohne Router-Zwang). Entweder ein TV-Kabelmodem und/oder ein TV-Kabelrouter. Systembedingt muss der TV-Kabelprovider das Endgerät für den Kunden parametrieren und kann im gleichen Schritt das Endgerät mit der Tunnel-Technik ausstatten lassen.

Demnach können nur TV-Kabelprovider "echtes" Dual Stack Lite anbieten. Festnetz- oder Mobilfunk-Provider dagegen arbeiten nur mit Dual Stack. Auch dann, wenn sie ein private IPv4-Adresse an den Endkunden vergeben.

Teredo

von Microsoft

Teredo (RFC 4380) ist eine Tunnel-Technik von Microsoft, die IPv6-Datenpakete in IPv4-UDP-Datenpakete kapselt, damit Windows-PCs und die XBox (Spielekonsole) die NAT-Schranke von Internet-Zugangsroutern von innen überwinden und so miteinander reden können. Teredo kommt dabei ohne Anwendereingriffe aus. Als IPv6-Tunnel-Technik soll Teredo den Übergang von IPv4 auf IPv6 vereinfachen und an Rechnern in lokalen (IPv4-)Netzen eine global gültige IPv6-Adresse vergeben. Dazu baut Teredo einen IPv6-Tunnel über eine IPv4-Sitzung auf.

Teredo ist seit Windows XP Servicepack 1 in allen Microsoft-Betriebssystemen enthalten. Bei neueren Windows-Betriebssystemen kommt Teredo für eine IPv6-Nutzung über eine IPv4-Infrastruktur standardmäßig zum Einsatz. Beispielsweise auch in der XBox One. Die Spielekonsole setzt einen Teredo-Tunnel grundsätzlich voraus. Sonst funktioniert der Multiplayer-Modus nicht mehr.

Teredo ist nicht nur auf Windows beschränkt. Für Unix-Systeme gibt es das auch mit dem Namen Miredo.

Funktionsweise von Teredo

Der Client kontaktiert für den Verbindungsaufbau einen voreingestellten Teredo-Server über IPv4. Der Server hilft beim Analysieren der NAT-Umgebung (Network Address Translation). Der eigentliche IPv6-Verkehr läuft anschließend über das Teredo-Relay. Der Teredo-Server und das Teredo-Relay werden von Microsoft betrieben.

Bei der Paketübertragung verpackt Teredo die ausgehenden IPv6-Datenpakete in UDP-Pakete (Port 3544) und sendet sie per IPv4 an einen Teredo-Server, der gleichzeitig IPv4- und IPv6-Verbindungen herstellen kann. Teredo tunnelt also IPv6-Pakete aus einer IPv4-NAT-Umgebung heraus.

Auf diese Weise lassen sich aber auch Firewalls durchtunneln, die nur TCP, UDP und ICMP erlauben. Deshalb funktioniert Teredo auch dann, wenn die Verfahren 6in4/6to4/6over4 aus gutem Grund geblockt werden. Doch damit verstößt Microsoft gegen alle Sicherheitskonventionen.

Teredo vs. 6in4/6to4/6over4/DSlite

Neben Teredo gibt es noch andere Tunneltechniken, die IPv6-Pakete über IPv4 übertragen. Im wesentlichen 6in4, 6to4 und 6over4.

Der Unterschied zwischen Teredo und 6in4/6to4/6over4 ist, dass Teredo dem Benutzer die Konfiguration weitgehend aus der Hand nimmt. Während man bei 6in4/6to4/6over4

bestimmte Voraussetzungen erbringen muss, funktioniert Teredo in vielen Fällen ohne weitere Eingriffe. Es benötigt lediglich eine öffentlich erreichbare IPv4-Adresse. Beispielsweise über NAT. Um alles andere kümmert sich die Software. Tunnelverfahren wie zum Beispiel 6to4 funktionieren nur mit einer öffentlichen IPv4-Adresse.

Teredo und CG-NAT

Weil Internet-Provider und Mobilfunknetzbetreiber nicht genug öffentliche IPv4-Adressen für alle ihre Kunden haben, wickeln sie den Datenverkehr über CG-NAT und private IPv4-Adressen ab. Das bedeutet, dass neben einer global gültigen IPv6-Adresse nur eine private IPv4-Adresse pro Kunde zur Verfügung steht. Der IPv4-Verkehr wird dabei behelfsmäßig über ein Provider-seitiges NAT abgewickelt (CG-NAT). Wenn das Betriebssystem für IPv6-Verbindungen nicht die global gültige IPv6-Adresse, sondern Teredo verwendet, dann laufen die IPv6-Verbindungen über IPv4 und NAT. Dabei halten die IPv4-Verbindungen nicht dauerhaft und somit auch der Teredo-Tunnel nicht.

Teredo und private IPv4-Adressen direkt am Internet-Zugang sind generell keine gute Kombination und äußerst instabil.

Alternativen zu Teredo

Im Vergleich zu anderen VPN- und Tunneltechniken ist Teredo eher als Notlösung zu sehen. Anfangs war Teredo ein guter Ansatz von Microsoft. Teredo hat sich aber nie wirklich durchgesetzt. Im Zuge dessen, dass es immer öfter Dual Stack oder natives IPv6 gibt, wäre es besser komplett auf Teredo zu verzichten. Alternativen gibt es einige. Leider sind die technisch viel aufwändiger und erfordert manuelles Eingreifen des Benutzers (Port-Weiterleitungen) oder es fehlen die Voraussetzungen (UPnP oder NAT-PMP) in den Routern.

Wie sicher ist Teredo?

Die Kommunikation mit Teredo wird als Sicherheitsrisiko angesehen. Denn Teredo hebelt das Konzept des per NAT abgeschotteten lokalen Netze aus. Durch Teredo sind IPv4-Rechner wegen der globale gültigen IPv6 direkt erreichbar. Selbst dann, wenn die Router-Firewall den IPv6-Verbindungsaufbau blockiert. Der Teredo-Tunnel schleust die IPv6-Datenpakete am NAT-Router und der Firewall vorbei ins IPv4-Netzwerk. Auf diese Weise können Zugriffe auf die Computer kommen, die es gar nicht geben sollte.

Wie man es von NAT kennt, muss immer der Client eine Verbindung aus dem lokalen Netzwerk ins öffentliche Netz aufbauen, damit anschließend eine bidirektionale Verbindung möglich ist. Erst dann dürfen alle Datenpakete des Teredo-Servers den NAT-Router und die Firewall passieren. Und dann packt der Teredo-Client alle IPv4-Datenpakete aus, verarbeitet die IPv6-Pakete weiter und leitet sie ins lokale Netz möglicherweise weiter. Vorausgesetzt, ein Angreifer kann die Pakete manipulieren, dann könnte er das lokale Netzwerk von innen scannen, nach Angriffsmöglichkeiten suchen,

Router und Firewall manipulieren und vielleicht sogar Verbindungen von innen nach außen initiieren, um Software nachzuladen. Und das alles ohne das es der Benutzer merkt.

Auch aus Sicht des Datenschutzes ist Teredo als äußerst kritisch anzusehen. Da alle Windows-Betriebssysteme standardmäßig einen Teredo-Server von Microsoft kontaktieren, wenn eine Verbindung zu einem IPv6-Server aufgebaut werden soll. Anschließend läuft sämtlicher Datenverkehr über die Microsoft-Server, die möglicherweise in den USA stehen. Hier ist Vorsicht geboten. Insbesondere deshalb, weil hier die Daten im Klartext übers Netz übertragen werden.

Es ist durchaus sinnvoll Teredo abzuschalten (empfohlen) oder im Zweifelsfall IPv6 gleich ganz zu deaktivieren (nicht empfohlen).

Windows: netsh interface ipv6 set teredo disable

Umstieg von IPv4 auf IPv6

Für viele ist IPv6 einfach nur ein IPv4 mit längeren Adressen. Doch diese Ansicht ist völlig falsch. IPv6 ist ein Protokoll mit vielen neuen Funktionen. Die Erfahrungen, die jemand aus der IPv4-Welt mitbringt, lassen sich nur bedingt auf IPv6 übertragen.

Die Anzahl der IPv6-User nimmt in Zukunft stark zu. Vor allem in Lateinamerika und Afrika. Dort befindet sich die Internet-Infrastruktur noch im Ausbau. Gleichzeitig nimmt die Anzahl der User stark zu. Und gleichzeitig gibt es dort die geringste Verfügbarkeit von IPv4-Adressen.

Für diejenigen, die Netzwerke betreiben oder administrieren ist deshalb dringend angeraten sich intensiv mit IPv6 zu beschäftigen. Wer Wissen und Erfahrung mit IPv4 mitbringt, hat aber nur geringe Vorteile. Denn IPv6 bringt zwar viele Dinge mit, die auch von IPv4 bekannt sind, trotzdem verfolgt IPv6 ganz andere Ideen und Ziele.

Beim Umstieg oder der Umstellung von IPv4 auf IPv6 geht es nicht darum, IPv4 durch IPv6 auszutauschen, sondern neben IPv4 zusätzlich IPv6 zu betreiben. Der Parallelbetrieb treibt dabei den Aufwand für Netzplanung, Sicherheit und Konfiguration nach oben. Im Prinzip muss man jede Arbeit auf IP-Ebene doppelt machen. Einmal für IPv4 und dann auch noch für IPv6. Das zusätzliche Protokoll führt erst einmal zu mehr Arbeit und Komplexität. Dabei muss man berücksichtigen, dass der Parallelbetrieb einige Jahrzehnte dauern wird. IT-Verantwortliche müssen das bei der Personalplanung und den Kosten berücksichtigen.

Übergangsverfahren von IPv4 auf IPv6

Es ist praktisch unmöglich alle veralteten Geräte auf einmal zu aktualisieren und durch IPv6-fähige zu ersetzen. Deshalb gibt es eine Reihe von Übergangsverfahren, die IPv6-Pakete über die bestehende IPv4-Infrastruktur übertragen können.

IPv6-Fähigkeit

Prinzipiell kommt man um IPv6 nicht mehr herum. Deshalb ist es erforderlich, dass beim Einkauf von Hardware und Software auf IPv6-Fähigkeit geachtet werden muss. Doch während man zu IPv4 in Ausschreibungen, Bedienungsanleitungen und Dokumentationen seitenweise Informationen findet, wird in einer Zeile gerade mal "IPv6-Fähigkeit" erwähnt. Doch was heißt das? Im Prinzip sagt "IPv6-Fähigkeit" oder "IPv6-fähig" gar nichts aus.

Viele Komponenten der Netzinfrastruktur, wie z. B. Router und Firewalls, können noch nicht korrekt mit IPv6 umgehen. Sie haben nur einen Teil von IPv6 implementiert.

Doch selbst wenn Hardware und Software IPv6 vollumfänglich unterstützen, dann sind in den Unternehmen und bei den Anwendern bei Weitem noch nicht alle

Herausforderungen bei der Anschaffung, Installation und Betrieb für IPv6-geeignete Komponenten gelöst.

Verantwortliche und Administratoren betreten mit IPv6 zwangsläufig Neuland. Trotzdem sollte man die Migration kontinuierlich vorantreiben. Dazu gehört auch, bei Neuanschaffungen konsequent auf IPv6-Fähigkeit zu achten und diese auch in einer Teststellung zu überprüfen. Nicht nur bei Routern und der Firewall. Auch bei jeder anderen Hardware und Software.

IPv6 in lokalen Netzwerken (LAN)

Der Wechsel von IPv4 auf IPv6 in einem LAN gelingt in der Regel problemlos. In Windows Vista, Windows 7 und 8, MacOS und Linux ist IPv6 bereits enthalten. Anwender mit einem aktuellen Betriebssystem können bereits in ihrem lokalen Netzwerk IPv6 nutzen. Verteilende Geräte, wie Hubs und Switche, ist es egal ob IPv4 oder IPv6 zum Einsatz kommt. Sie kümmern sich um die Netzwerk-Kommunikation unterhalb des Internet Protokolls.

Nur bei den DSL- und WLAN-Routern im Privat-Bereich gibt es so gut wie keine IPv6-Unterstützung. Das bedeutet, innerhalb eines lokalen Netzwerks sind IPv6-Verbindungen möglich. Um auch IPv6-Verbindungen ins Internet aufbauen zu können, bedarf es der Unterstützung von IPv6 im Internet-Zugangs-Router und einer "nativen" IPv6-Anbindung des Internet-Providers. Das bedeutet, der Internet-Zugangs-Router muss einen globalen Präfix vom Provider-Netz bekommen und diesen auch im lokalen Netzwerk per Router Advertisement verteilen.

IPv6 im Zugangsnetz (Netzbetreiber)

Seit Ende 2012 werden Internet-Anschlüsse Schritt für Schritt auf IPv6 umgestellt. Wobei es sich bei der Umstellung um einen Parallelbetrieb von IPv4 und IPv6 (Dual-Stack-Modus) handelt. Die Deutsche Telekom als größter Netzbetreiber hat damit begonnen. Weitere Netzbetreiber folgen.

Leider erschweren ausschließlich IPv4-taugliche DSL- und WLAN-Router auf der Kundenseite die IPv6-Einführung erheblich. Das erfordert umständliche und teure Übergangsverfahren.

IPv6 im Internet

Damit ein Server im Internet per IPv6 erreichbar ist, muss im DNS-Server die IPv6-Adresse eingetragen sein. Zusätzlich müssen die Netze und kontaktierten Stationen IPv6 beherrschen.

Als nächste Schwierigkeit folgen Router, die IPv6 unzureichend unterstützen. Wer IPv6 ernsthaft nutzen will, der benötigt sowohl auf der LAN- wie auch auf der WAN-Seite IPv6-Fähigkeit.

Das Hauptproblem sind jedoch Plattformen, die auf Basis von IPv4 entwickelt wurden und nur mit IPv4 funktionieren. Ohne spezielle IPv6-Anwendungen wird es über eine lange Zeit kein reines IPv6-Internet geben. Server, die nur mit IPv4 adressierbar sind, sind für reine IPv6-User unerreichbar. Aus Kompatibilitätsgründen muss IPv4 noch eine lange Zeit unterstützt werden.

DNS und IPv6

Damit ein Server von außen über IPv6 erreichbar ist, müssen im DNS neben dem bestehenden A-Record für die IPv4-Adresse auch der AAAA-Record für die IPv6-Adresse eingetragen sein.

```
example.com. MX 10 mx1.example.com
example.com. MX 20 mx1.example.com
mx1.example.com. A 192.0.2.10
mx1.example.com. AAAA 2001:db8::2
mx2.example.com. A 192.0.2.20
```

Server-Dienste auf IPv6 umstellen

Das Augenmerk bei allen Serverdiensten ist die parallele Nutzung von IPv6 und IPv4.

Beim Einrichten eines neuen Servers muss man sich die Frage stellen, wie dieser zu seiner IPv6-Adresse kommt. Entweder per Hand, DHCPv6 oder eine andere zentrale Adressverwaltung. Auch der zuständige DNS-Server, der für einen Host-Namen die entsprechende IPv4-Adresse ausliefern muss per IPv6 ansprechbar sein und IPv6-Adressen ausliefern können. Erst danach ist die folgende Vorgehensweise in drei Schritten möglich.

1. Konfiguration von IPv6-Adressen auf dem Server
2. IPv6 für die einzelnen Dienste aktivieren
3. Dienste per IPv6 über DNS erreichbar machen (AAAA-Records mit IPv6-Adressen versehen)

Nach dem Einrichten der IPv6-Adressen und aktivieren von IPv6 sollte man versuchen eine Verbindung über die IPv6-Adresse aufzubauen. Am einfachsten mit einem Ping. Hierzu sollte man zuerst einen Server anpingen, der nur per IPv6 erreichbar ist.

Wenn der Ping-Test geklappt hat, kann man die Erreichbarkeit von außen prüfen. Hierzu pingt man von einem externen IPv6-fähigen Internet-Anschluss den neu konfigurierten Server an.

Abschließend sollte in jedem Fall ein Port-Scan auf die neu eingerichteten IPv6-Adressen erfolgen, um zu überprüfen, ob eventuell Ports offen sind, die es nicht sein sollten.

Troubleshooting

Während einer Übergangszeit wird man mit Aussagen konfrontiert sein wie "Der Server ist nicht erreichbar". Um Fehlerquellen durch IPv4 und IPv6 berücksichtigen zu können, sollte man sowohl einen IPv4- und einen separaten IPv6-Test-Server installieren, damit Kunden, Anwender und Supporter Dual-Stack-Probleme und in der kombinierten IPv4/ IPv6-Anbindung erkennen können. Oder anders ausgedrückt, um herauszufinden, ob der Fehler in der fremden oder eigenen Infrastruktur zu suchen ist.