Hochschule für Technik, Wirtschaft und Kultur Leipzig (FH)

Fachbereich Informatik, Mathematik und Naturwissenschaften

Diplomarbeit

Entwicklung und Bewertung eines Einsatzes des IPv6-Protokolls in einem IP-Intranet unter allgemeinen Gesichtspunkten einer Migration mit Schwerpunkt der Ersteinführung des Protokolls für Mobile Services

vorgelegt von

André Böttcher

Matrikelnummer: 24864

Betreuung durch: Dipl. Ing. Thomas Sprey (Volkswagen AG)

Verantwortlicher Hochschullehrer: Prof. Dr. Klaus Hänßgen

(Wolfsburg, August – Oktober 2002)

E r k l ä r u n g

Ich versichere wahrheitsgemäß, die Diplomarbeit selbständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Wolfsburg, den

Inhalt

1 Einleitung

Das Internet-Protokoll Version 6 – IPv6 – auch als das Internet-Protokoll der nächsten Generation – IPnG – bezeichnet, wurde entwickelt, um das nun schon über 20 Jahre im Einsatz befindliche Internet-Protokoll der Version 4 – IPv4 – abzulösen.

IPv4 wurde Ende der 70er Jahre entwickelt, um einige wenige Rechner des amerikanischen Militärs oder universitäre Einrichtungen miteinander zu vernetzen. Unter den damaligen Voraussetzungen schien eine auf 32 Bit bezogene Adressierungsarchitektur auszureichen um ein paar Tausend Rechner des amerikanischen Verteidigungsministeriums und später ein paar Millionen Rechner von auf der ganzen Welt verteilten wissenschaftlichen Einrichtungen miteinander zu verbinden. Diese Annahme wurde Anfang der 90er Jahre ungültig. Denn als sich der Begriff Internet nicht mehr nur auf Militär- und Forschungseinrichtungen beschränkte, sondern durch eine ständig wachsende Anzahl von Anwendungen, die auf eine weltweit vernetzte Datenbasis zugreifen können, neu definiert wurde, schien eine somit explosionsartig ansteigende Anzahl von Hosts mit 32 Bit langen IP-Adressen in absehbarer Zeit nicht mehr adressierbar zu sein.

Hieraus entstand als wichtigster Grund für die Entwicklung von IPv6 die Bereitstellung eines größeren Adressraumes, um das zukünftige Wachstum von IP-Geräten und -Benutzern zu unterstützen. Die Entwickler von IPv6 wollten jedoch nicht nur mehr Adressen bereitstellen, sondern ein neues Protokoll entwickeln, welches sowohl das Adressproblem löst als auch die aus IPv4 bekannten Schwächen beseitigt. So definierten sich die Ziele der Entwickler des neuen Internet-Protokolls aus den Defiziten von IPv4 und Möglichkeiten eines Wechsels zwischen den Protokollversionen.

- Bereitstellung eines Adressierungsschemas das neben der Lösung des Adressproblem ein effizienteres Routing und eine Reduzierung von Routingtabellen auf Routern ermöglicht.
- Eine Vereinfachung des Protokolls, um die Performancezeiten von Routeroperationen zu verbessern.

- Bereitstellung von Authentifizierungsmechanismen und Verschlüsselungsverfahren die eine Sicherheit ohne zusätzliche Applikationen ermöglichen.
- Bildung eines Adressierungskonzeptes, welches eine flächendeckende Multicastadressierung zur Verfügung stellt.
- Integration von Quality of Service Funktionen, die eine konstant verfügbare Mindestbandbreite für unterschiedliche Applikationen zwischen Kommunikationspartnern garantieren.
- Bereitstellung von Funktionen die den Anforderungen mobiler IP-Geräte gerecht werden.
- Bildung von Mechanismen, die eine Autokonfiguration von IP-Geräten ohne zusätzliche Server ermöglichen.
- Die Möglichkeiten von zukünftigen Weiterentwicklungen muss gewährleistet werden.
- Bereitstellung von Möglichkeiten die eine Koexistenz der Internet-Protokollversionen 4 und 6 garantieren.

Der Anfangs große Boom bei der Entwicklung und dem vorhergesagten Einsatz von IPv6 ist Ende der 90er Jahre zurückgegangen. Dies resultiert daraus, dass der drohende Adresskollaps durch eine Definition des klassenlosen Routings (CIDR) und einer Einführung der Nutzung privater Adressbereiche in Unternehmen, welche über Network Address Translation (NAT) mit dem Internet verbunden werden können, aufgeschoben wurde. Des Weiteren wurden Entwicklungen wie beispielsweise IPSec von IPv6 auf IPv4 zurück portiert. Ebenso wurden Lösungen vorgestellt, mit denen das Problem von Anforderungen an Quality of Service Möglichkeiten mit IPv4 zumindest in hauseigenen Netzwerken gelöst werden kann.

Dennoch gibt es Funktionen in IPv6, welche mit IPv4 nicht oder nur ansatzweise gelöst werden können. Hier sei neben der automatischen Konfiguration von Netzwerkknoten, welche in IPv4 nur durch zusätzliche Server (DHCP) bereitgestellt werden kann oder die Unterstützung von Mobilen Endgeräten durch Mobile IPv6 genannt. Denn durch die Arbeitsweise von Mobile IPv6 kann eine Kommunikation mit mobilen Endgeräten unabhängig deren Aufenthaltsorte ohne umständliches Triangle Routing über einen Heimatagenten, wie es bei einer Lösung für IPv4 nötig war, gewährleistet werden. Unter dem Stichwort Mobile wird

auch der oben erwähnte Aufschub eines Adresskollapses wieder hinfällig. Denn auch durch den Einsatz einer klassenlosen Routingstruktur können die durch die Mobilkommunikation entstehenden Endgeräte nicht mehr adressiert werden.

So kann IPv4 trotz aller vorgenommenen Rettungsversuche heutige und zukünftige Anforderungen nicht erfüllen, wodurch die Kombination aus Wireless und Sprach-Daten-Konsolidierung eine langersehnte Einführung IPv6 zur Realität gemacht hat. Denn bei großen Mobilfunk-Netzbetreibern ist die Einführung von IPv6 für 2.5G und zukünftiges 3G schon längst im Gange.

Trotz dieser Forderung nach IPv6 besteht für Unternehmen kein Grund zur Beunruhigung. Denn IPv6 setzt keine Umstellung oder Erweiterung der vorhanden Infrastruktur voraus, da für eine Koexistenz beider Protokollversionen garantiert wird. Um dennoch die Möglichkeiten und Funktionen, welche sich durch IPv6 bieten für zukünftige Entwicklungen in Unternehmen zu nutzen, stehen für die Einführung von IPv6 zahlreiche Techniken zur Verfügung, die eine schrittweise Migration von IPv4 zu IPv6 gewährleisten.

1.1 Motivation für diese Diplomarbeit

Für ein Unternehmen, was wie die Volkswagen AG mit 10.0.0.0 ein privates Klasse A Netz, mit 143.163/16 ein öffentliches Klasse B Netz sowie mit 194.114/23 128 öffentliche Klasse C Netze zur Verfügung hat, scheint das bekannte Adressproblem von IPv4 in den Hintergrund zu treten. Diese Aussage ist jedoch nur richtig, wenn davon ausgegangen wird, dass in zukünftigen Entwicklungen weiterhin auf klassische Datennetze, welche sich auf die Vernetzung von Computern beschränken, gesetzt wird. Dies hätte jedoch zur Folge, dass der Einsatz neuer Techniken erheblich eingeschränkt ist. Denn allein unter dem Stichwort „IP everywhere“, wobei unter anderen an den Einsatz des Internet-Protokolls auf Endgeräten wie IP-Telefone, Handhelds, Fahrzeuge und Entwicklungsstraßen gedacht wird, wird auch der bereitstehende IPv4-Adressraum im Unternehmen knapp werden.

Neben der Bereitstellung einer auf absehbaren Zeit ausreichenden Anzahl von IP-Adressen bietet IPv6 auch eine Anzahl neuer Funktionen um die genannten IP-Geräte sinnvoll einzusetzen. Zu diesen Funktionen zählt neben der Fähigkeit einer

automatischen Konfiguration, was beispielsweise die Inbetriebnahme eines IP-Telefons erheblich erleichtert, die Unterstützung Mobiler Geräte durch Mobile IPv6. Mobile IPv6 wird derzeit für das Unternehmen der Volkswagen AG als das interessanteste Feature von IPv6 gesehen. Aus diesem Grund liegt es nahe, dass weit umfassende Thema IPv6 auf eine Untersuchung von Mobile IPv6 zu beschränken. Hieraus ergab sich die Aufgabenstellung dieser Diplomarbeit, die sich neben dem besagten Mobile IPv6 mit einer damit erforderlichen Einführung von IPv6 beschäftigt.

1.2 Gliederung dieser Diplomarbeit

Als Grundlage für die Untersuchung einer Einführung von IPv6 für Mobile IPv6 wird im Kapitel 2 dieser Arbeit das neue Internet-Protokoll beschrieben. Dabei wird mit einer Veranschaulichung der Veränderungen von Adressarchitektur und Datagram-Aufbau gegenüber IPv4 begonnen. Danach wird gezeigt, wie die mit IPv6 mögliche Autokonfiguration von Netzwerkknoten funktioniert. Des Weiteren wird ein Überblick über Veränderungen und Neuerungen in Protokollen höherer Ebenen, die sich mit der Umstellung von IPv6 ergeben, gegeben. So werden Änderungen im DNS, ICMP für IPv6 (ICMPv6), DHCP für IPv6 (DHCPv6) sowie das Routing in Verbindung mit IPv6 veranschaulicht. Abschließend zu diesem Kapitel werden die mit IPv6 definierten Techniken zur Migration von IPv4 zu IPv6 erläutert.

Da die Sicherheit der Datenkommunikation in einem Unternehmen besonders in Mobilen Endgeräten, welche das gesicherte Heimatnetz verlassen könnten, eine nicht unwesentliche Rolle spielt, werden im Kapitel 3 die mit IPv6 bereitgestellten Sicherheitsmechanismen beschrieben.

In Kapitel 4 wird ausführlich beschrieben, wie eine Kommunikation mit mobilen Endgeräten verläuft, und wie Mobile IPv6 dabei hilft, das Routing zwischen dem mobilen Rechner und dem Kommunikationspartner effizient zu gestalten. Hierbei wird auf die Unterschiede und Verbesserungen gegenüber einer Mobilitätslösung mit IPv4 eingegangen. Nachdem das Funktionsprinzip von Mobile IPv6 erläutert wurde, wird auf Sicherheitsanforderungen, welche an die bei einer Kommunikation mit einem Mobilen Host beteiligten Instanzen gestellt werden müssen, eingegangen und Möglichkeiten für eine Erfüllung dieser Anforderungen erläutert.

In einem weiteren Abschnitt dieses Kapitels werden mögliche Einsatzgebiete von Mobile IPv6 vorgestellt. Abschließend wird ein Überblick über den derzeitigen Entwicklungsstand von Mobile IPv6 in möglichen Anwendungen gegeben.

Um eine bestehende IPv4 Netzwerk Infrastruktur nach IPv6 zu migrieren, muss geklärt werden, ob die eingesetzten Systeme die neue Internet-Protokollversion unterstützen und welche Funktionen bereits implementiert wurden. Hierzu wird in Kapitel 5 ein Überblick über den derzeitigen Implementationsstand von IPv6 auf den in der Volkswagen AG eingesetzten Systeme gegeben und diskutiert, aus welchen Gründen auf vielen Systemen noch Defizite bezüglich der Implementation von IPv6 vorherrschen.

In Kapitel 6 wird diskutiert, wie die in Kapitel 2.7 beschriebenen Migrations-techniken für eine mögliche Einführung von IPv6 genutzt werden können. Dabei werden nach einer Beschreibung der von einer Umstellung betroffenen Komponenten zwei Wege einer Migration dargestellt, welche sich in der Ersteinführung von IPv6 mit dem Beginn am Backbone bzw. in einem Teilnetz unterscheiden. Da nicht nur die Protokolle der Internet Schicht von einer Umstellung betroffen sind, sondern auch Protokolle höherer Ebenen, wird in einem weiteren Abschnitt dieses Kapitels auf Auswirkungen einer IPv6 Einführung auf diese eingegangen. Im abschließenden Abschnitt dieses Kapitels wird zu den von möglichen Einsatzgebieten unabhängigen beschriebenen Migrationsvorschlägen auf eine Ersteinführung für Mobile Services eingegangen. Hierbei wird diskutiert, welche beschriebenen Möglichkeiten genutzt werden können und welche Möglichkeiten aus geforderten Anforderungen an Mobile IPv6 nicht genutzt werden können.

Nach den theoretischen Ausführungen wird in Kapitel 7 der Aufbau eines IPv6 Testnetzwerkes beschrieben, wobei die vorgenommen Konfigurationen auf den Routern und Hosts erläutert werden. An dieser Testumgebung werden Untersuchungen, welche sich auf die Funktion der Autokonfiguration von Netzwerkknoten und einer Implementierung des IPv6 fähigen DNS Servers Bind 9 beziehen, vorgenommen. Dabei wird untersucht, ob die IPv6 Implementierungen auf den eingesetzten Nortel, Cisco, Linux, und Windows Systeme untereinander kompatibel sind.

2 Grundlagen zu IPv6

Dieses Kapitel soll einen Überblick des neuen Internet-Protokolls IPv6 geben. Es werden sowohl der Aufbau des IPv6-Datagram als auch der Aufbau und die Architektur von IPv6-Adressen erläutert.

Es wird gezeigt, wie die mit IPv6 mögliche Autokonfiguration von Routern und Hosts funktioniert. Des Weiteren wird ein Überblick über Veränderungen und Neuerungen in Protokollen höherer Ebenen, die sich mit der Umstellung von IPv6 ergeben, gegeben. So werden Änderungen im DNS, ICMP für IPv6 (ICMPv6), DHCP für IPv6 (DHCPv6) sowie das Routing in Verbindung mit IPv6 veranschaulicht.

2.1 IPv6-Datagram

Das IPv6-Datagram ist in [RFC2460] definiert und besteht wie das Datagram der Internet-Protokollversion 4 aus einem IP-Header und den Daten (Payload) des IP-Pakets. Der IPv6-Header setzt sich aus einem Basisheader konstanter Länge und optionalen Erweiterungsheadern zusammen.

2.1.1 IPv6-Basisheader

Der Basisheader hat sich gegenüber dem Header von IPv4 auf 7 Felder reduziert, wodurch ein schnelleres Weiterleiten des IP-Pakets in Routern ermöglicht wird.

Trotz der Reduzierung der Felder des Basisheader ist dieser auf 40 Byte angestiegen, was jedoch auf die 128 Bit langen Adressen zurückzuführen ist. Eine Reduzierung der Felder wurde erreicht, indem Felder, welche in IPv4 noch fest verankert waren, in IPv6 optional sind und in Erweiterungsheadern realisiert werden.

Zum anschaulichen Vergleich der Veränderungen vom IPv4- zum IPv6-Packet-Header sollen Abbildung 2.1 und Abbildung 2.2 dienen.

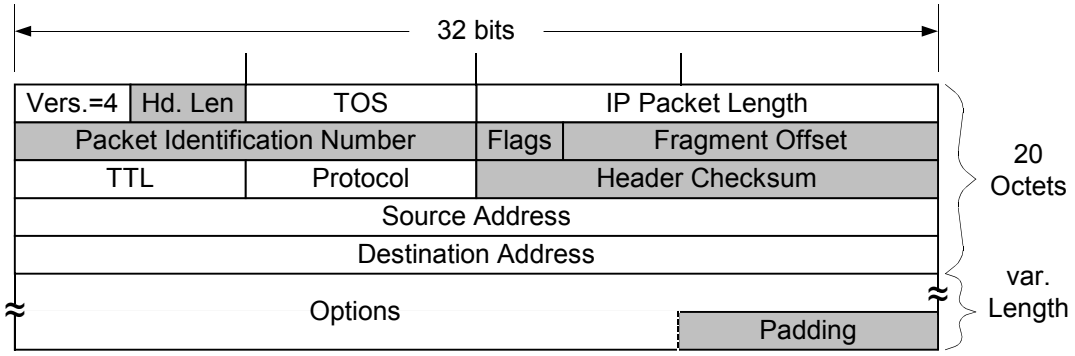

Abbildung 2.1: schematische Darstellung des IPv4-Paket-Header

Zur Verdeutlichung der Änderungen zwischen den Paket-Headern sind die Felder, welche nicht mehr vorhanden bzw. neu hinzugekommen sind grau dargestellt.

Auf eine Erläuterung der Felder im IPv4-Header wird an dieser Stelle mit einem Verweis auf [RFC791] verzichtet.

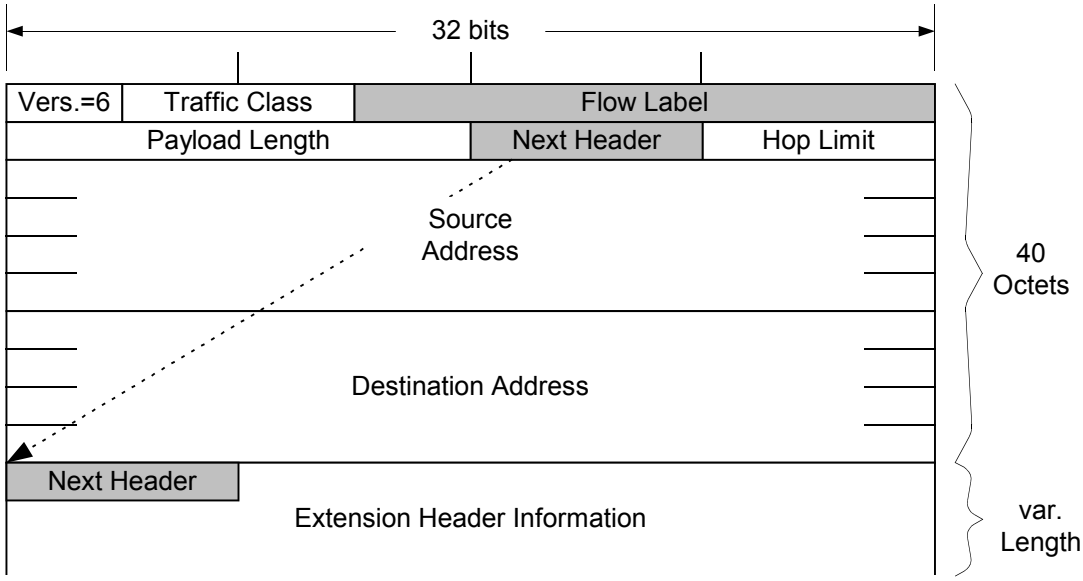

Abbildung 2.2: schematische Darstellung des IPv6-Paket-Header

Das 4 Bit große Feld *Version* im IPv6-Header gibt die Internet-Protokollversion des IP-Pakets an und hat im IPv6-Datagram den Wert 6. Durch dieses Feld ist eine Abwärtskompatibilität der Protokollversionen gewährleistet, da Router anhand dieses Wertes das IP-Paket gesondert behandeln können. Das Feld Version ist analog dem Versionsfeld im IPv4 Header.

Das Feld *Traffic Class* (8 Bit) ist ähnlich dem TOS-Feld im IPv4-Header. Mit diesem Feld kann in Routern eine Priorisierung des Datenpakets vorgenommen werden, wodurch eine Flusssteuerung von Paketen ermöglicht wird. Der Unterschied zum TOS-Feld besteht darin, dass das Feld in IPv4 nur optional zu behandeln war, d.h. es wurde in den meisten Router-Implementierungen gar nicht beachtet. In IPv6 muss dieses Feld von Routern ausgewertet werden, wodurch die Übertragungen von Realtime-Anwendungen erst möglich werden.

Mit Hilfe der 20 Bit des *Flow Label* Feldes (neu in IPv6), können Verbindungen aufgebaut werden, bei denen die Router anhand des Wertes Datenströme identifizieren und somit gesondert behandeln können. Datenströme werden von Paketen mit gleicher Quell- und Zieladresse sowie einen von Null unterschiedlichen Flow-Label-Wert gebildet. Das Flow-Label-Feld wird vom Resource Reservation Protocol (RSVP) genutzt, um eine Verbindung für von einem Datenfluss gewünschte Serviceeigenschaften wie beispielsweise garantierte Bandbreite zwischen Quelle und Ziel bereitzustellen. Das erste zu einer Verbindung gehörende Paket teilt den Routern auf dem Weg zum Ziel die gewünschten Serviceeigenschaften mit und die Router kennzeichnen die Verbindung anhand des Flow Label Wertes. In nachfolgenden zum Datenstrom gehörende Pakete brauchen die Router dann nur das Flow-Label-Feld auszuwerten und das Paket auf der vorher gewählten Route weiterzuleiten. Hierdurch können zum Beispiel Quality of Service Funktionen oder Realtime-Anwendungen gekennzeichnet werden. Zur Zeit befindet sich dieses Feld jedoch noch im Experimentierstatus und wird von Hosts und Routern die diese Funktion nicht unterstützen Null gesetzt, was bedeutet, dass das Feld ignoriert wird.

Das Feld *Payload Length* ist ein 16 Bit unsigned Integer und ist ähnlich dem IP-Packet-Length-Feld in IPv4. Es gibt an, wie viele Bits dem Basisheader folgen. Hieraus resultiert eine maximale Paketgröße von 64 KByte, welche jedoch durch die Jumbo-Payload-Option [RFC2675] im Hop-by-Hop-Options Header auf über 4 GByte erhöht werden kann.

Das neue Feld *Next Header* (8 Bit) enthält einen Code (vgl. Kapitel 2.1.2), der den dem Basisheader folgenden Header identifiziert.

Das Feld *Hop Limit* (8 Bit) entspricht dem Time-To-Live-Feld im IPv4-Protokoll. Der Wert dieses Feldes wird von jedem durchlaufenden Router dekrementiert und bei erreichen des Wertes Null wird das Paket verworfen.

Source Address und *Destination Address* enthalten jeweils die 128 Bit IPv6-Adresse des Senders bzw. des Empfängers des IP-Pakets.

Diesen 40 Bytes, welche den Basisheader des IPv6-Headers darstellen, können optionale Informationen im Erweiterungsheader (*Extension Header Information*) folgen.

Durch eine konstante Headerlänge ist eine Optimierung der Hardware von IPv6-Protokollstacks möglich, wodurch z.B. ein besserer Datendurchsatz gegenüber IPv4 auf Routern zu erwarten ist.

Des Weiteren tragen die gegenüber IPv4 weniger gewordenen Basisheader-Felder zu einer Reduzierung von Routeroperationen bei.

Eine weitere Entlastung der Router wird durch das Wegfallen von Fragmentierung und einer Berechnung von Prüfsummen ermöglicht. Was in Bezug auf den wachsenden Leitungsdurchsatz eine wesentliche Rechenersparnis für Router darstellt.

2.1.2 Erweiterungsheader

Optionen, welche nicht zwingend für den Transport von Daten-Paketen benötigt werden, wurden aus dem Basisheader entfernt und können optional in Erweiterungsheadern eingebunden werden. Durch das Einbetten der Optionen in Erweiterungsheader können die Optionen wesentlich umfangreicher sein und von IPv6 besser genutzt werden.

Bei der Verwendung eines korrekten Erweiterungsheaders für IPv6 kann ein Paket optionale Informationen transportieren, die nur für den Empfänger oder nur für dazwischenliegende Router bestimmt sind.

Ebenso wird durch die Einführung von Erweiterungsheadern eine Flexibilität mit Blick auf zukünftige Erweiterungen gewährleistet. [WER00]

Die IPv6-Erweiterungsheader sind im Datagram zwischen Basisheader und Nutzdaten platziert.

Abbildung 2.3 zeigt schemenhaft das Format des IPv6-Erweiterungsheaders.

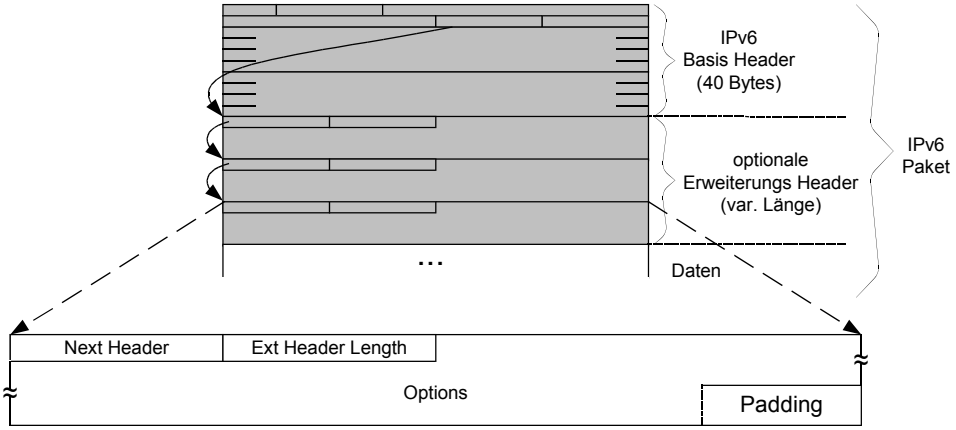

Abbildung 2.3: schematische Darstellung des IPv6 Erweiterungsheaders

Zur Identifizierung eines Erweiterungsheaders wird das Feld Next Header, welches sowohl im Basisheader als auch im Erweiterungsheader auf den nächsten Header verweist, genutzt.

Das Feld *Next Header* ist zu vergleichen mit dem Protokoll-Feld aus IPv4, welches das transportierte Protokoll angibt. Das bedeutet, dass das Next-Header-Feld nicht nur IPv6-Erweiterungsheader identifiziert, sondern auch Protokolle höherer Ebenen.

Das Feld *Ext Header Length* (8 Bit) gibt die Länge des Erweiterungsheaders in Einheiten von jeweils 8 Bytes (ohne die ersten 8 Bytes) an.

Das Feld *Options* ist je nach Header Typ individuell gestaltet. Wenn durch die Optionen die Länge des Erweiterungsheaders kein ganzzahliges vielfaches von 64 Bits erreicht, wird der Header mit Padding-Bytes auf diese Länge aufgefüllt.

Zur Verdeutlichung der Verkettung von Erweiterungsheadern in IPv6, ist in Abbildung 2.4 eine beispielhafte Verkettung von Headern dargestellt.

Für die Identifizierung von Erweiterungsheadern bzw. Header von Protokollen höherer Ebenen werden die in [RFC1700] für IPv4 definierten Werte genutzt, zu

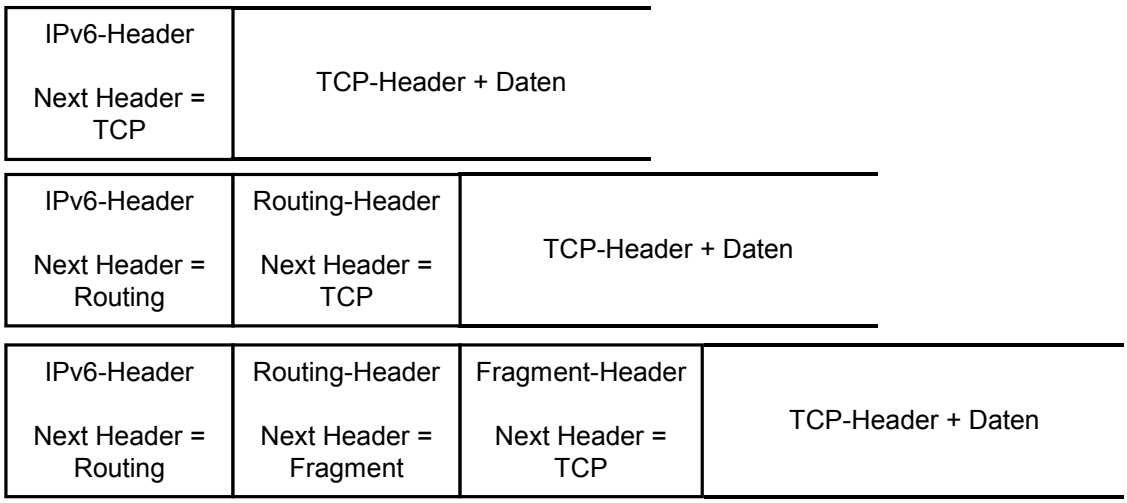

Abbildung 2.4 schemenhafte Darstellung der Verkettung von IPv6-Erweiterungsheadern und Headern höherer Protokollschichten

denen die in Tabelle 2.1 dargestellten Werte hinzugekommen sind [RFC2460]. Von diesen Erweiterungsheadern können die Header Authentication und Encapsulation Security Payload auch in IPv4 verwendet werden (IPSec).

Erweiterungs-Header	***Wert (Hex.)***	***Beschreibung***
Hop by Hop Options	0	Wird von allen Knoten, die ein IP-Paket von Quelle zum Ziel durchläuft ausgewertet.
Routing	2B	Der Routing Header wird für Source Routing genutzt.
Fragment	2C	Bei verwendet, wenn aufgrund einer zu geringen MTU zwischen Sender und Empfänger das IPv6-Paket fragmentiert werden muss. Der Fragment-Header wird in jedem fragmentierten Paket benutzt.
Encapsulation Security Payload	32	In diesem Header wird unter anderem die Verschlüsselung der Nutzdaten bestimmt.
Authentication	33	Dieser Header bestimmt die Methode der Authentisierung des Senders genutzt wird.
ICMPv6	3A	Inter Control Message Protocol für IPv6
No Next	0	Kein weiterer Header
Destination Options	3C	Steht diese Header vor dem Routing Header, so müssen Router diesen auswerten, ansonsten wird es nur vom Zielrechner ausgewertet.

Tabelle 2.1 Next Header Codes und deren Bedeutung

Wenn in einem IPv6-Paket mehrere Erweiterungsheader enthalten sind, ist folgende Reihenfolge einzuhalten:

- IPv6 Header
- Hop-by-Hop Options Header
- Destination Options Header
- Routing Header
- Fragment Header
- Authentication Header
- Encapsulating Security Payload Header
- Destination Options Header[1]
- upper-layer Header (TCP [RFC793], UDP[RFC768], ICMP [RFC792], ...)

Ein Beispiel für den Grund der Reihenfolge ist, dass der Routing-Header nicht verschlüsselt werden darf und somit vor dem Authentication-Header stehen muss.

Des Weiteren sei an dieser Stelle darauf hingewiesen, dass in Fällen, in den beispielsweise Firewalls und QoS Filter zum Einsatz kommen, eine Verschlüsselung auf IP-Ebene zu Problemen führen kann, da für diese Funktionen notwendige Informationen ebenfalls verschlüsselt werden.

2.1.3 Unterschiede im Header zwischen IPv4 und IPv6

Abschließend zu diesem Kapitel werden noch einmal zusammenfassend die wesentlichen Veränderungen zwischen den Headern der IPv4- und IPv6-Datagrame aufgelistet.

Verkürzung des Headers

- Fragmentation-Feld entfernt
- IP-Optionen entfernt
- Header-Checksum entfernt
- Header-Length-Feld entfernt
- Length-Feld ohne IPv6 Header

Überarbeitung

- Time-to-Live → Hop-Limit

1 Der Destination Options Header ist der einzige Header, der zweimal in einem IPv6-Header vorkommen darf. Steht er vor einem Routing Header, muss er von den im Routing Header stehenden Routern ausgewertet werden; steht er am Ende der Header-Erweiterungen, muss er nur vom Zielknoten ausgewertet werden.

- Protocol → Next-Header
- Precedence & TOS → Traffic-Class
- Adressen von 32 Bits auf 128 Bits verlängert

Erweiterung

- Flow-Label-Feld hinzugefügt

Ein weiterer wesentlicher Unterschied ist, dass die Anzahl der Optionen durch das 8 Bit lange Feld Header-Length auf 40 Byte begrenzt waren. In IPv6 werden die Optionen bzw. Erweiterungsheader erst durch die maximale Paket Größe von 64K bzw. 4G oder einer von Protokollen der Layer 1 und Layer 2 Ebenen vorgeschrieben Größe -MTU- begrenzt.

2.2 IPv6-Adressarchitektur

Alle IPv6-Adressen haben eine Länge von 128 Bit und identifizieren, auch wie in IPv4, entweder ein oder mehrere Interfaces.

Es gibt drei Typen von IPv6-Adressen. Diese sind Unicast-, Anycast- und Multicast-Adressen. Die aus IPv4 bekannten Broadcast-Adressen sind in IPv6 nicht mehr vorhanden, ihre Funktionen werden durch Multicast-Adressen ersetzt.

Eine Unicast-Adresse identifiziert ein eindeutiges Interface eines Netzknotens.

Anycast-Adressen werden genutzt, um eine Gruppe von Interfaces bzw. Netzwerk-Knoten zu adressieren. Wobei zu beachten ist, dass ein IP-Paket, welches an eine Anycast-Adresse gesendet wird, von nur einem Mitglied, z.B. der nächste dieser Gruppe empfangen wird. Dabei wird die Auswahl des Gruppenmitglieds, welches das Paket empfängt, von Routern bzw. den eingesetzten Routing-Protokollen vorgenommen. Da eine Anycast-Adresse von einer Unicast-Adresse nicht zu unterscheiden ist, muss bei der Konfiguration die Eigenschaft „Anycast“ mit angegeben werden. Wird dies nicht getan, werden doppelt vergebene Unicast-Adressen von den Routern erkannt und entsprechende Fehlernachrichten generiert.

Multicast-Adressen adressieren ebenfalls eine Gruppe von Interfaces, jedoch wird ein an eine Multicast-Adresse adressiertes IP-Paket von allen Mitgliedern dieser Gruppe empfangen.

Die 128 Bit lange IPv6-Adresse setzt sich aus 8 Blöcken mit jeweils 16 Bits, welche durch „:“ voneinander getrennt sind zusammen. Hierbei werden alle 16-Bit-Blöcke als 4 Hexadezimalzahlen dargestellt, wobei eine groß-klein-Schreibung der vorkommenden Buchstaben in den Hexadezimalzahlen nicht beachtet werden muss. Eine Ausnahme dieser Schreibweise bilden sogenannte Ipv4-kompatible-IPv6-Adressen.

Als Beispiel für die Darstellung soll folgende IPv6-Adresse dienen.

4030:00BC:0000:0000:0267:01FF:FE01:7352

Zur Vereinfachung der Schreibweise bzw. Darstellung einer IPv6-Adresse können führende Nullen in den durch „:“ getrennten Blöcken gestrichen werden. Die als Beispiel angegebene IPv6-Adresse kann somit auf folgende Darstellung verkürzt werden.

4030:BC:0:0:267:1FF:FE01:7352

Des Weiteren können Nullfolgen in einer Adresse durch „::“ ersetzt werden. Sollten in einer Adresse mehrere Nullfolgen auftreten, so ist eine Ersetzung nur einmal möglich. Zur Verdeutlichung sind folgend einige Beispiele dargestellt.

FE80:0:0:0:0:0:0:0057	–>	FE80::57
0:0:0:0:0:83C:993	–>	::83C:993
FE80:0:0:2:0:0:0:5	–>	FE80::2:0:0:0:5
bzw.	–>	FE80:0:0:2::5
0:0:0:0:0:0:0:0	–>	::

In Abbildung 2.5 ist noch einmal Zusammenfassend die Darstellung einer IPv6-Adresse und derer verkürzten Schreibweise beispielhaft dargestellt.

In IPv6-Adressen, die aus IPv4-Adressen durch Voranstellen von 96 Null-Bits gebildet werden, so genannte IPv4-kompatible-IPv6-Adressen (vgl. Kapitel 2.7.2), können die letzten 32 Bits in dezimaler Schreibweise belassen werden.

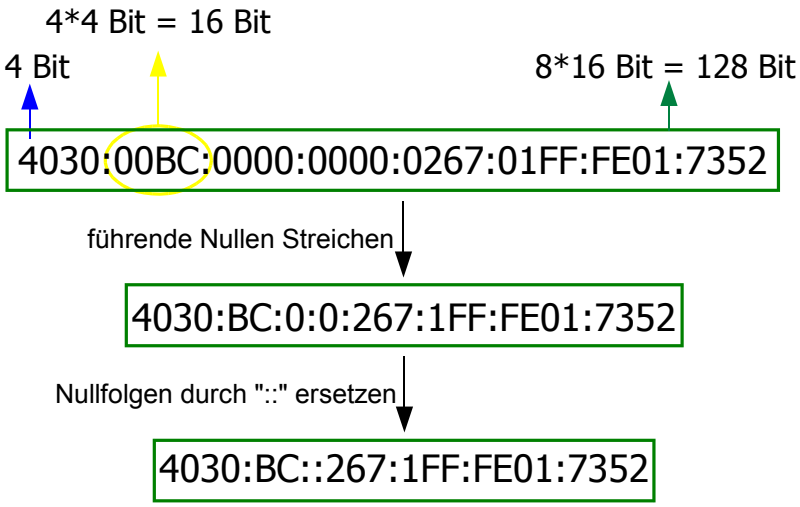

Abbildung 2.5: Beispielhafte Darstellung einer IPv6-Adresse und die Möglichkeiten einer verkürzten Schreibweise

Beispiel:

::129.187.214.42

In IPv6 gibt es zwei Adressen, welche als Sonderadressen bezeichnet werden. Diese zwei Adressen sind zum einen die Loopback-Adresse und zum anderen die unspezifizierte Adresse.

Die Loopback-Adresse hat das Format „::1“ und kann für lokale Software-Rückkopplungstest verwendet werden. Hierfür adressiert die Loopback-Adresse ein virtuelles Interface eines Hosts, d.h. es braucht keine Netzwerkkarte vorhanden sein um z.B. TCP-Verbindungen zu testen. Ein Paket, welches an eine Loopback-Adresse adressiert ist, darf niemals einen Host verlassen.

Die unspezifizierte Adresse mit dem Format „::“ wird von einem Absender als Source-Adresse in ein IP-Paket eingetragen, wenn dieser seine eigene IP-Adresse noch nicht kennt.

Neben den genannten Adresstypen wurden für die Migration von IPv4 zu IPv6 noch eine Reihe von Sonderadressen definiert. Eine Beschreibung dieser Adressen und deren Anwendung wird in Kapitel 2.7.2 gegeben.

Bei der Struktur von IPv6-Adressen wird zwischen einer unstrukturierten 128 Bit langen IP-Adresse und einer Strukturierten IP-Adresse unterschieden.

Die strukturierte Adresse setzt sich aus n Bits für die Netzidentifikation und (128-n) Bits für das Knoten-Interface zusammen. Der n Bit lange Netzidentifikator dient sowohl der Zustellung eines IP-Pakets in ein bestimmtes Netzwerk, als auch der Erkennung, ob es sich bei dieser Adresse um eine Uni-, Multi- oder Anycast-Adresse handelt. Die Bits des Netzidentifikators bzw. die Bits, welche zur Kennzeichnung des Netzwerkes genutzt werden, werden auch Adresspräfix genannt. Die Anzahl der Bits des Adresspräfixes werden als Präfixlänge bezeichnet und wie in IPv4, durch „/“ getrennt an die IP-Adresse angehängt. Zur Verdeutlichung soll folgendes Beispiel dienen.

Interface-Adresse	=	2030:BC:1B24:A4C7:267:1FF:FE01:7367
Adress-Präfix-Länge	=	56 Bits
Netzbezeichnung	=	2030:BC:1B24:A400::/56
Interface-Adresse mit Präfix	=	2030:BC:1B24:A47C:267:1FF:FE01:7367/56

Wie bei IPv4 sind auch bei IPv6 bestimmte Adressen für spezielle Anwendungen vorgesehen. Hierfür wurde eine Aufteilung des IPv6-Adressraum in verschiedene Adressbereiche vorgenommen. In Tabelle 2.2 sind die Adressbereiche mit ihren dazugehörigen Adresspräfixen und ihrem Anteil am gesamten IPv6-Adressraum dargestellt [RFC2373]. Wie aus der Tabelle deutlich zu ersehen ist, ist bisher erst ein geringer Teil des gesamten IPv6-Adressraumes einer Verwendung zugeordnet. Durch den großen Anteil der noch nicht zugewiesenen Adressräume wird eine Anpassung an zukünftige Entwicklungen gewährleistet.

Adressbereichs-Aufteilung	*Adresspräfix (binär)*	*Adresspräfix (hexadez.)*	*Anteil am gesamten Adressraum*
Reserviert	0000 0000	00	1 / 256
Nicht zugewiesen	0000 0001	01	1 / 256
Reserviert für NSAP Allokation	0000 001	02 ... 03	1 / 128
Reserviert für IPX Allokation	0000 010	04 ... 05	1 / 128
Nicht zugewiesen	0000 011	06 ... 07	1 / 128
Nicht zugewiesen	0000 1	08 ... 0F	1 / 32
Nicht zugewiesen	0001	10 ... 1F	1 / 16
Aggregierbare globale Unicast-Adresse	001	20 ...3F	1 / 8
Nicht zugewiesen	010	40 ... 5F	1 / 8
Nicht zugewiesen	011	60 ... 7F	1 / 8
Nicht zugewiesen	100	80 ... 9F	1 / 8
Nicht zugewiesen	101	A0 ... BF	1 / 8
Nicht zugewiesen	110	C0 ... BF	1 / 8
Nicht zugewiesen	1110	E0 ... EF	1 / 16
Nicht zugewiesen	1111 0	F0 ... F7	1 / 32
Nicht zugewiesen	1111 10	F8 ... FB	1 / 64
Nicht zugewiesen	1111 110	FC ... FD	1 / 128
Nicht zugewiesen	1111 1110 0	FE00 ... FE7F	1 / 512
Link-Lokale Netzadressen	1111 1110 10	FE80 ... FEBF	1 / 1024
Site-Lokale Netzadressen	1111 1110 11	FEC0 ... FEFF	1 / 1024
Multicast-Adressen	1111 1111	FF	1 / 256

Tabelle 2.2: Aufteilung der IPv6 Adressbereiche

Aggregierbare globale Unicast-Adressen ersetzen das in IPv4 benutzte Klassen-Modell und ermöglichen ein effizientes Routing durch eine Adresshierarchie (vgl. Kapitel 2.2.1).

Adressen mit dem Präfix FE80::0/10 werden als Link-Lokal bezeichnet und sind nur innerhalb eines Subnetzes eindeutig. Das bedeutet, dass IP-Pakete mit diesem Präfix von Routern nicht weitergegeben werden dürfen. Jedes Knoten-Interface bekommt bei der Aktivierung eine solche Adresse zugewiesen.

Site-Lokal sind Interfaces, welche sich im gleichen Netzwerk befinden. Da Site-Lokale Adressen im Internet nicht geroutet werden, können diese mit den privaten Adressbereichen von IPv4 verglichen werden.

Multicast-Adressen adressieren eine Gruppe von Knoten. Wird eine Nachricht an eine Multicast-Adresse gesendet, so empfangen alle Mitglieder dieser Gruppe diese Nachricht (vgl. Kapitel 2.2.2).

2.2.1 Aggregierbare globale Unicast-Adresse

Aggregierbare globale Unicast-Adressen sind in [RFC2374] definiert und bezeichnen die Adressen, welche global gültig sind und somit im Internet geroutet werden. In Abbildung 2.6 ist der Aufbau einer solchen Adresse dargestellt. Die 128 Bit der Adresse werden in 64 Bit Netz-Anteil und 64 Bit Host-Anteil aufgeteilt. Der Host-Anteil der Adresse wird auch als Interface-Identifier (Interface-ID)bezeichnet.

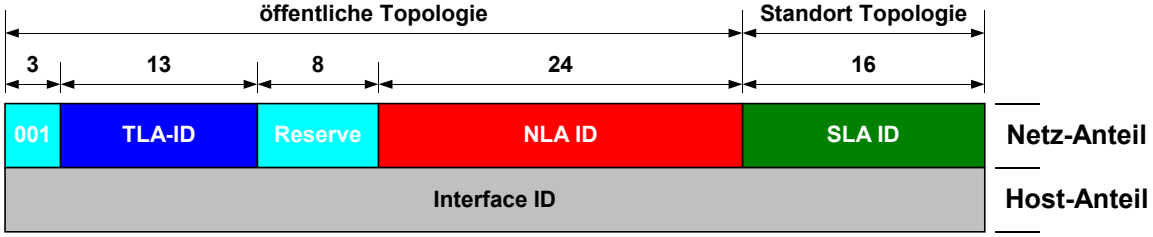

Abbildung 2.6: schematische Darstellung der Aufteilung von aggregierbaren globalen Unicast-Adressen [BRA01]

Die Identifizierung dieses Adresstyps erfolgt in den ersten 3 Bit (*FP* - Format Präfix) der Adresse und lauten 001_2 (vgl. Tabelle 2.2).

Die nächsten 13 Bit ergeben den Top-Level-Aggregator (*TLA*) welche die oberste Stufe in der Routinghierarchie darstellt.

Das Feld *Reserve* (8 Bit) ist für zukünftige Zwecke reserviert und muss Null gesetzt werden.

Die folgenden 24 Bit definieren den Next-Level-Aggregator (*NLA*).

Diese führenden 48 Bit beschreiben die öffentliche Topologie und stellen den Präfix einer Organisation oder eines Unternehmens dar. Die restlichen 16 Bit des Netzanteils der Adresse werden als Site-Level-Aggregator (*SLA*) bezeichnet und dienen der Bildung von individuellen Adressierungshierarchien einer einzelnen Organisation (Standort Topologie).

Die 64 Bit lange Interface-ID der Adresse wird gemäß den „Guidelines for 64-Bit Global Identifier“ [EUI64] aus den eindeutigen MAC-Adressen der Knoten-Interfaces gebildet. Zur Veranschaulichung ist in Abbildung 2.7 die Abbildung einer 48-Bit MAC-Adresse auf eine Interface-ID dargestellt.

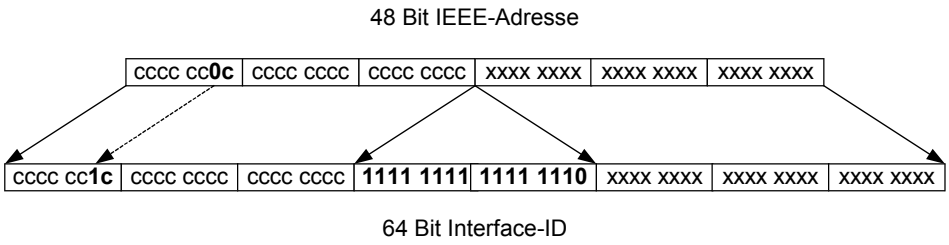

Abbildung 2.7: Abbildung einer 48-Bit MAC-Adresse auf eine 64 Bit Interface-ID

Zur Abbildung einer 48 Bit IEEE-MAC-Adresse auf den 64 Bit Interface-Identifier werden zwischen den ersten 3 Bytes (Hersteller des Netzadapters) und den hinteren 3 Byte der MAC-Adresse die 2 Byte FF:FE_{Hex} eingefügt.

Das 7. Bit der Interface-ID dient der Kennzeichnung auf Eindeutigkeit der Adresse und wird auch als u-Bit (universal/local-Identifier) bezeichnet. Bei eindeutigen Adressen wird das u-Bit auf 1 gesetzt. Nichteindeutige Adressen entstehen, wenn z.B. einer seriellen Schnittstelle, welche keine MAC-Adresse besitzt, manuell eine Interface-ID zugewiesen werden muss oder eine MAC-Adresse einer Ethernet Schnittstelle geändert wird.

Die Bildung der Interface-ID einer IPv6-Adresse aus der MAC-Adresse hat neben dem Vorteil, das sie von jedem Host automatisch gebildet werden kann, einen Nachteil,

2.2.2 Multicast-Adressen

Multicast-Adressen identifizieren eine Gruppe von Hosts. Wird ein IP-Paket an eine Multicast-Adresse und somit an eine Multicast-Gruppe gesendet, dann empfangen alle Mitglieder dieser Gruppe dieses Paket.

Eine Multicast-Adresse wird gebildet aus 8 Bit Adresspräfix gefolgt von 4 Bit Flags und 4 Bit Scope sowie 112 Bit Group-ID (vgl. Abbildung 2.8).

Abbildung 2.8: schemenhafte Darstellung der Struktur einer Multicast-Adresse

Die Identifizierung einer Multicast-Adresse geschieht anhand der ersten 8 Bit der Adresse (Adresspräfix), welche immer „FF_{Hex}“ (vgl. Tabelle 2.2) lautet.

Die oberen drei der vier Bit *flags* sind reserviert und müssen 0 gesetzt werden. Das unterste Flagbit gibt an, ob die Adresse permanent (Wert=1) oder nicht permanent (Wert=0) vergeben ist.

Mit dem *Scope*-Wert kann angegeben werden, wie weit das Multicast-Paket geroutet werden soll, wodurch eine Unterteilung der Multicast-Adresse in Link-Lokale, Site-Lokale und globale Bereiche stattfindet. In Tabelle 2.3 sind die zulässigen Werte, welche in [RFC2373] definiert sind, aufgelistet.

Wert (Hex.)	***Bedeutung***	***Wert (Hex.)***	***Bedeutung***
0	reserved	8	organization-local scope
1	node-local scope	9	(unassigned)
2	link-local scope	A	(unassigned)
3	(unassigned)	B	(unassigned)
4	(unassigned)	C	(unassigned)
5	site-local scope	D	(unassigned)
6	(unassigned)	E	global scope
7	(unassigned)	F	reserved

Tabelle 2.3: Scope-Werte einer Multicast-Adresse und deren Bedeutung

Durch den Scope-Wert wird sichergestellt, dass sich das Multicast-Paket nicht weiter als angegeben ausbreiten kann. In IPv4 wird das Eingrenzen der Ausbreitung von Paketen mit dem Hop-Limit realisiert. Diese Methode hat jedoch den Nachteil, dass der Hop-Limit- Wert mit jeder Netzwerkänderung aktualisiert werden muss. Ebenso kann auch eine Anwendung anhand des Scope-Wertes feststellen, wieweit ein Paket geroutet wird.

Das Feld *Group-ID* gibt die Multicast-Gruppe innerhalb des angegebenen Scopes an. Um einen Überblick über die derzeit vergebenen Multicast-Gruppen bzw. Multicast-Adressen zu erhalten empfiehlt sich ein Blick in [RFC2375].

Zu beachten ist, dass die Felder Flag und Scope zur Adresse gehören und somit verschiedene Multicast-Adressen bilden, so werden z.B. mit der Adresse FF0x::2 immer Router erreicht. Jedoch spricht die Adresse mit x=2 nur alle Router im gleichen link-lokalen Bereich (Layer 2) an, wobei bei einem Wert x>2 auch Router über diesen Bereich hinaus adressiert werden.

Zur Verdeutlichung soll folgendes Beispiel dienen:

Ein DHCP-Server für eine Firma registriert sich an der Multicast-Gruppe mit der Adresse FF08::1:3. Der Scope-Wert ist 8 (organization local scope). Damit empfängt er jedoch nicht die Nachrichten von Rechnern, die einen DHCP-Server in ihrem site-local Bereich (also mit Scope 5) suchen. Soll er diese dennoch empfangen, so muss sich der Server zusätzlich für die Multicast-Adresse FF05::1:3 registrieren. [SCH00]

2.2.3 Adressauflösung

Das Domain Name System (DNS) wird dazu verwendet, Rechnernamen sowohl zu IPv6-Adressen als auch zu IPv4-Adressen aufzulösen. Dabei werden IPv4-Adressen weiterhin als *A*-Resource Records dargestellt. Für IPv6-Adressen wurde in [RFC2874] der neue[2] *A6*-Resource Recordtyp eingeführt.

Der Datenbereich eines A6-Records hat das in Abbildung 2.9 dargestellte Format.

Präfixlänge (1 Oktett)	Adress-Suffix (0...16 Oktetts)	Präfixname (0...255 Oktetts)

Abbildung 2.9: schematische Darstellung des Aufbaus eines A6-Resource Records

Um die neu definierten A6-Records in die DNS-Datenbank einzubringen, ist folgende Syntax definiert:

<Präfixname> A6 <Präfixlänge> <Adress-Suffix> <Domain>

2 Die zuerst eingeführte Resource Recordtyp gemäß [RFC1886] lautete *AAAA* und wird aus Kompatibilitätsgründen weiter unterstützt.

oder

<Präfixname> IN A6 <Präfixlänge> < Adress-Suffix > <Domain>

Für Duale IPv6/IPv4-Knoten (vgl. Kapitel 2.7.1) werden Einträge mit Recordtypen für IPv6 und IPv4 im DNS geführt. Ihre Resolverbibliotheken müssen ebenfalls beide Recordtypen unterstützen.

Zur Adressauflösung in umgekehrter Richtung, also um zu einer gegebenen IPv6-Adresse den Knotennamen zu erhalten, wurde in [RFC2673] eine neue Notation eingeführt, da die Übertragung des bei IPv4 verwendeten Prinzips auf IPv6 zu extrem langen und somit fehleranfälligen Texten geführt hätte.

Die neue Schreibweise wird als „Binary Label“ für DNS bezeichnet. Es werden ganze oder nur Teile einer IPv6-Adresse als zusammenhängende Zahlenfolgen dargestellt, wobei eine Binär-, Oktal-, Dezimal- oder Hexadezimalschreibweise zulässig ist. Ein Binary Label besteht aus einer digit-sequence, welcher optional ein Schrägstrich „/“ (slash) mit nachfolgender Längenangabe folgen kann. Hieraus ergibt sich die Syntax :

„\[<digit-sequence>]“ oder „[<digit-sequence>/<length>]“

Für die <digit-sequence> kann z.B. die Hexadezimalschreibweise ähnlich wie in der Programmiersprache „C“ mit einem vorangestellten „x“ oder die punktierte Dezimalschreibweise, d.h. hinter jeder Dezimalziffer folgt ein Punkt („.“), benutzt werden.

Für die Abbildung von Ipv6-Adressen auf Namen ist in [RFC3152] unter der Top-Level-Domain „arpa“ speziell für IPv6 die Sub-Domäne „ip6“ reserviert. Das bedeutet, dass die Abbildungen von IPv6-Adressen auf Namen immer in der Domäne „ip6.arpa“ stehen.

Da eine Adressierung von Hosts über IPv6-Adressen für einen Anwender „unzumutbar“ sein wird, ist eine rasche Implementierung von DNS-Systemen, welche die neuen Resource Records unterstützen, in einem wachsenden IPv6-Netzwerk unverzichtbar. Bereits heute unterstützt die DNS-Server Software BIND in späten 8.x-Version und 9.x-Versionen die notwendigen Funktionen, um sie in einem IPv6 Netzwerk einzusetzen.

In folgendem Beispiel wird abschließend ein DNS-Eintrag in alter Notation bei IPv4 und neuer Notation bei IPv6 dargestellt:

Alte Notation bei IPv4:

Eintrag für einen Rechnernamen in die DNS-Hauptdatenbank in alter Notation für IPv4

bb-service02.wob.vw.de IN A 10.182.26.20

Zugehöriger Eintrag in der Reverse-Address Datei

71.26.182.10.in-addr.arpa. IN PTR bb-service02.wob.vw.de

Neue Notation für IPv6

IPv6-Adresse = FE80::10:5A30:114D

Eintrag für einen Rechnernamen in der DNS-Hauptdatenbank in neuer Notation für IPv6

bb-service02.wob.vw.de IN A6 0 FE80::10:5A30:114D

Zugehöriger Eintrag in der Reverse-Address Datei

\[xFE80 0000 0000 0000 0000 0010 5A30 114D].ip6.arpa IN PTR

bb-service02.wob.vw.de

2.3 Internet Control Message Protocol für IPv6

Das Internet Control Message Protocol für IPv6 (ICMPv6) ist im [RFC2463] definiert und wird von IPv6-Knoten genutzt um Fehlernachrichten, Diagnosereports und Konfigurationsdaten zu transportieren. ICMPv6 enthält Funktionen, die in IPv4 mit den Protokollen ICMP, IGMP (Inter Group Management Protocol) und ARP (Address Resolution Protocol) realisiert wurden.

Alle ICMP-Nachrichten sind als Nutzlast von IP-Paketen zu verstehen und werden durch den Wert $3A_{Hex}$ im Next Header Feld (siehe Tabelle 2.1) eines IP-Headers gekennzeichnet.

Der grundsätzliche Aufbau eines ICMPv6-Pakets ist in Abbildung 2.10 dargestellt.

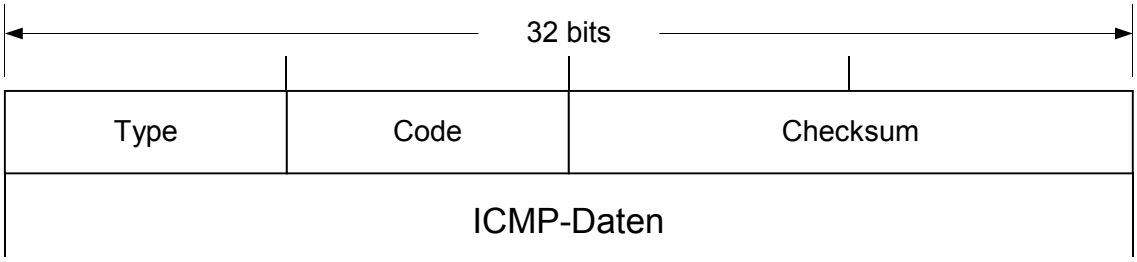

Abbildung 2.10: schematische Darstellung des grundsätzlichen Aufbaus eines IPv6-ICMP-Pakets

Das Feld *Type* gibt den Typ der ICMP-Nachricht an. In Tabelle 2.4 sind die derzeit definierten Nachrichtentypen mit ihrem *Type-Wert* und einer kurzen Erläuterung aufgelistet. Die Werte 0-127 (höherwertiges Bit = 0) definieren Fehlermeldungen und die Werte 128-255 (höherwertiges Bit = 1) stehen für Steuerungs- und Statusmeldungen.

Das *Code*-Feld ist abhängig vom Nachrichtentyp und wird für mögliche Unterfunktionen der Nachricht verwendet.

Im Feld *Checksum* wird eine Prüfsumme über das gesamte ICMP-Paket eingetragen.

Die *ICMP-Daten* variieren mit den ICMP-Nachrichtentypen, welche an dieser Stelle nicht näher erläutert werden sollen. Eine genaue Definition und die Funktions- und Arbeitsweise der Nachrichtentypen wird in den jeweiligen Referenzen beschrieben.

Type (dez.)	*Bedeutung*	*Reference*
1	Destination Unreachable - angegebene Zieladresse nicht erreichbar	[RFC2463]
2	Packet to Big - Paketgröße überschreitet MTU einer Teilstrecke	[RFC2463]
3	Time Exceeded - Zeitüberschreitung bei Hop Limit oder beim defragmentieren von Paketen	[RFC2463]
4	Parameter Problem	[RFC2463]
128	Echo Request	[RFC2463]
129	Echo Replay	[RFC2463]
130	Group Membership Query - Bewerbung um Mitgliedschaft in einer Multicast-Gruppe	[RFC2710]
131	Group Membership Report - Antwort auf Bewerbung um eine Mitgliedschaft in einer Multicast-Gruppe	[RFC2710]
132	Group Membership Reduction - Abmelden aus einer Multicast-Gruppe	[RFC2710]
133	Router Solicitation - Anfrage nach Router-Werten	[RFC2461]
134	Router Advertisement - Bekanntgabe von Routerwerten	[RFC2461]
135	Neighbor Solicitation	[RFC2461]
136	Neighbor Advertisement	[RFC2461]
137	Redirect - Aufforderung eines Routers an einen Knoten, einen besser geeigneten Router als den Standard-Router zu verwenden	[RFC2461]
138	Router Renumbering	[RFC2894]
139	ICMP Node Information Query	[Draft01]
140	ICMP Node Information Response	[Draft01]
141	Inverse Neighbor Discovery Solicitation Message	[RFC3122]
142	Inverse Neighbor Discovery Advertisement Message	[RFC3122]
150	Home Agent Address Discovery Request	[Draft03]
151	Home Agent Address Discovery Reply	[Draft03]
152	Mobile Prefix Solicitation	[Draft03]
153	Mobile Prefix Advertisement	[Draft03]

Tabelle 2.4: Festlegung des Typ-Feldes von ICMP-Nachrichten

2.4 Autokonfiguration

Bei IPv4 wurden die für eine Netzwerkverbindung notwendigen Konfigurationsdaten, wie IP-Adresse, Subnetzmaske, DNS-Server und Standard-Router manuell eingestellt oder über ein DHCP-Server vergeben.

Bei der Definition des neuen Internetprotokolls wurde Wert darauf gelegt, dass die Konfiguration von Internethosts ohne manuelle Einstellungen und zusätzliche Server möglich ist. Ziel war es, einen Host zu starten und sofort einen Zugang zum Intranet und Internet zur Verfügung zu haben. Diese „Plug and Play“ Arbeitsweise wird als stateless Autokonfiguration bezeichnet. Als Grundlage der

automatischen Konfiguration wurde Neighbor-Discovery („Nachbarschafts-Entdeckung“) für IPv6 in [RFC2461] spezifiziert.

Neben der stateless Autokonfiguration bleibt auch weiterhin die Möglichkeit einer statefull Autokonfiguration, welche Netzwerkmanagern über die IPv6 Variante des Dynamic-Host-Configuration-Protocol (DHCPv6) umfangreichere Eingriffs-möglichkeiten bei der Vergabe von Netzwerkparametern bietet. Eine nähere Beschreibung zu DHCPv6 wird in Kapitel 2.5 gegeben.

2.4.1 IPv6 Neighbor Discovery

Das IPv6-Neighbor-Discovery-Protokoll definiert Mechanismen, die es Systemen am selben Link erlauben, Netzwerkinformationen wie MAC-Adressen von Nachbarn, MTU oder Hop-Limit zu bestimmen und doppelt vergebene IP-Adressen zu erkennen und zu vermeiden. Um diesen Informationsaustausch zu ermöglichen, wurden die folgend aufgeführten ICMPv6-Nachrichten definiert:

- Router Solicitation
- Router Advertisement
- Neighbor Solicitation
- Neighbor Advertisements
- Redirect

Unter Verwendung dieser ICMPv6-Nachrichten werden die in Tabelle 2.5 dargestellten Funktionen ermöglicht [RFC2461].

Um die beschrieben Nachrichten zu adressieren, sind auf allen IPv6-Hosts die folgenden IPv6-Adressen vordefiniert.

FF02::1 (link-node Multicast-Adresse)
: Mit dieser Adresse werden alle am lokalen Link angeschlossenen Knoten erreicht

FF02::2 (link-Router Multicast-Adresse)
: Mit dieser Adresse werden alle am lokalen Link angeschlossenen Router erreicht

FF02:0:0:0:0:1:FF00::/104 (solicited-node Multicast-Adresse)

Jeder IPv6-Knoten tritt zu Beginn seiner Interface-Konfiguration der Solicited-Node Multicast-Gruppe bei. Die Adresse setzt sich aus dem 104 Bit langen Multicast-Präfix FF02:0:0:0:0:1:FF00::/104 sowie aus den letzten 3 Byte der IPv6-Adresse zusammen.

FE80::/64 (link-lokale Adresse)

vgl. Kapitel 2.2

:: (unspezifizierte Adresse)

vgl. Kapitel 2.2

Neighbor Discovery Funktion	***Beschreibung***
Router Discovery	Lokalisieren von Routern im lokalen Netz
Prefix Discovery	Rechner erkennen, welche Ziele (Präfixe) vom lokalen Netz aus erreichbar sind und ob Ziele direkt oder nur über Router zu erreichen sind
Parameter Discovery	Erfragen von Netzwerkparametern
Address Autokonfiguration	automatische Konfiguration der IP-Adresse eines Interfaces
Address Resolution	Erfragen der MAC-Adresse eines Nachbarknotens (gleiche Layer2 Ebene), von dem nur die IP-Adresse bekannt ist
Next Hop Determination	Algorithmus um Zieladresse auf Adresse eines Nachbarn abzubilden, Next-Hop ist entweder das Ziel selbst oder ein Router
Neighbor Unreachability Detection	Rechner erkennen, wenn ein Ziel nicht mehr erreichbar ist und ein möglicher alternativer Router wird gesucht, Address Resolution
Duplicate Address Detection	Vergewissern, ob zu benutzende IP-Adresse nicht schon vergeben ist
Redirect	Router informiert Host, dass es einen besseren First-Hop zum angegebenen Ziel gibt

Tabelle 2.5 Auflistung der Funktionen des Neighbor Discovery Protokolls und deren Beschreibung

2.4.2 Vergleich mit IPv4

Das IPv6 Neighbor-Discovery-Protokoll entspricht einer Kombination der IPv4-Protokolle ARP [RFC826], ICMP-Router-Discovery [RFC 1256] und ICMP-Redirect [RFC1256].

In IPv4 gibt es kein im Allgemeinen vereinbartes Protokoll für „Nachbar-unerreichbar-Erkennung“.

Das Neighbor-Discovery-Protokoll liefert eine Menge Verbesserungen gegenüber den Kombinationen von IPv4. So enthalten beispielsweise Redirect-Nachrichten die Link-Layer-Adresse des besseren First-Hop Routers und nicht wie bisher seine IP-Adresse, was zusätzliche Nachrichten (ARP) unnötig macht, wodurch der Datenverkehr reduziert wird.

Die Router-Advertisements verbreiten die Präfixinformationen, wodurch auch die bisher benutzte Netzmaske ersetzt wird.

Vollkommen neu ist die statuslose Autokonfiguration, für die es in IPv4 keine analogen Mechanismen gibt. Auch die Möglichkeit zur Übergabe von Netzwerkparametern wie MTU oder Hop Limit durch den Router stellt eine Neuerung dar.

Des Weiteren lernen die Rechner alle On-Link-Präfixe von den Routern, so dass Pakete an Hosts, welche sich in lokalen Subnetzen am gleichen Lokal-Link befinden, direkt zugestellt werden können. Bisher wurden alle Pakete, welche sich laut Netzwerkmaske nicht on-link befanden, zuerst zum Router gesendet. Hierdurch wird eine Kommunikation in Netzen, welche Multinetting[3] unterstützen, effizienter.

Da Neighbor Discovery Nachrichten immer mit einem Hop Limit von 255 initialisiert werden, ist ein wirksame Schutz vor spoofing Attacken gewährleistet, denn Neighbor Discovery Nachrichten, welche ein HOP-Limit-Wert kleiner 255 haben, werden verworfen.

3 Multinetting bezeichnet den Betrieb verschiedener logische Subnetze auf einem physikalischen Netz.

2.5 Dynamic Host Configuration Protocol für IPv6

Im Zusammenhang mit der Entwicklung von Pv6 wurden auch Überlegungen angestellt, wie das Dynamic-Host-Configuration-Protocol (DHCP) an die neue IP-Version angepasst werden kann. Diese Entwicklung ist zum Zeitpunkt dieser Diplomarbeit noch nicht abgeschlossen, d.h. DHCP für IPv6 (DHCPv6) ist noch in keinem Internetstandard spezifiziert sondern lediglich in einem Internetdraft beschrieben.

In diesem Abschnitt soll die Funktions- und Arbeitsweise von DHCPv6 dargestellt werden. Die beschriebenen Erläuterungen beziehen sich auf den aktuellen Internetdraft [Draft02] zu DHCPv6. Dabei darf nicht vergessen werden, dass in zukünftigen Definitionen von DHCPv6 Änderungen gegenüber den beschriebenen Funktionen und Arbeitsweisen enthalten sein können.

Die Hauptaufgaben von DHCPv6 haben sich gegenüber denen von DHCPv4 nicht geändert. So gehören zu diesen weiterhin:

- Zuordnung von IP-Adressen
- Rechnernamen von einer zentralen Stelle verteilen
- von einem Knoten vorgeschlagene Rechnernamen prüfen und zuweisen
- Aktualisierung der DNS-Datenbank bei Namensvergabe und Adresszuteilung durch Dynamisches DNS (DDNS)
- IP-Parameter wie Adresspräfixe bzw. Subnetzmaske konfigurieren
- Bekanntgabe von Internetservern wie z.B. time-Server, NIS-Server, DNS-Server, ...

Bei den von DHCP genutzten Kommunikationsinstanzen haben sich einige Namen und deren Funktionen geändert. Die von DHCPv4 gewohnten Relay Agents heißen jetzt nur noch *Relays*. Auch ihr Verhalten hat sich geändert. Während bei DHCPv4 die Relay Agents eine relativ passive Rolle spielen und vor allem bei Beginn der Kommunikation zwischen DHCP-Server und DHCP-Client wichtig sind, übernehmen sie bei DHCPv6 einige Aufgaben der DHCP-Server, die auch in Hinsicht auf das Netzwerkmanagement interessant sind. Weiterhin gibt es im neuen Protokoll den Begriff der *DHCP-Agents*. Diese spielen eine Vermittler- und kontrollierende Rolle bei der Kommunikation zwischen DHCPv6-Clients und

DHCPv6-Servern. Die Rolle von DHCP-Agents kann sowohl von DHCPv6-Servern als auch Relays ausgefüllt werden. [RIE96]

Für DHCP-Nachrichten wird weiterhin UDP genutzt. Wobei die DHCPv6-Clients auf dem Port 546 und die DHCPv6-Server auf dem Port 547 auf Nachrichten warten. In DHCPv4 wurden hierfür die Ports 67 und 68 genutzt [RFC2131].

Die Adressierung erfolgt laut [Draft02] über die in Tabelle 2.6 aufgelisteten Multicast-Adressen sowie die Link-Lokalen Adressen der Clients.

Adresse	***Bezeichnung***	***Erläuterung***
FF0**2**::1:2	All_DHCP_Agents	Multicast-Adresse für alle Relay-Agenten und DHCP-Server im gleichen Link-Segment
FF0**5**::1:3	All_DHCP_Servers	Gemeinsame Multicast-Adresse für alle DHCP-Server der Domain

Tabelle 2.6: von DHCPv6 verwendete Multicast-Adressen

In [RFC2375] ist neben der All_DHCP_Servers-Adresse noch eine zweite Site-Lokale Multicast-Adresse für DHCP reserviert, welche ALL_DHCP_RELAYS-Adresse (FF05::1:4) genannt wird. Diese Adresse wird laut [Draft02] jedoch nicht mehr verwendet.

Clients senden DHCP-Nachrichten an die reservierten Multicast-Adressen, so dass diese nicht mit Adressen der DHCP-Server konfiguriert werden müssen. Außerdem brauchen keine Broadcast-Nachrichten mehr versendet werden, um einen DHCP-Server ausfindig zu machen. Wenn ein Client Nachrichten an einen DHCP-Server außerhalb des lokalen Links senden will, so wird ein DHCP-Relay benötigt der die Nachrichten an den Server weiterleitet. Sobald ein Client die Adresse eines DHCP-Servers kennt, sendet er seine DHCP-Nachrichten direkt an die Unicast-Adresse des Servers.

Für DHCPv6 sind derzeit die in Tabelle 2.7 angegebenen Nachrichtentypen definiert. Die Nachrichten werden anhand des Wertes im Type-Feld, welches in jeder DHCP-Nachricht enthalten ist identifiziert.

Type	*Name*	*Beschreibung*
1	Solicit	Suche nach DHCP-Server
2	Advertise	Bekanntgabe eines DHCP-Servers
3	Request	Client-Anfrage nach Konfigurationsdaten
4	Confirm	Anfrage an Server ob Client-Daten noch gültig
5	Renew	Client sendet die vom Server erhalten Daten zum Server
6	Rebind	wenn, keine Antwort auf Renew wird Rebind gesendet um neue Daten zu erhalten
7	Reply	Server-Antwort mit Konfigurationsdaten
8	Release	Client-Freigabe von Ressourcen
9	Decline	Client teilt Server mit, dass die Zugewiesene Adresse im Link schon verwendet wird
10	Reconfigure	Rekonfigurationsaufforderung des Server an Client
11	Information-Request	Anfrage nach Konfigurationsdaten ohne IP-Adresse
12	Relay-Forward	Weiterleitung einer Anfrage
13	Relay-Reply	Antwort auf Relay-Forward

Tabelle 2.7: mögliche Nachrichten-Typen in DHCPv6

2.6 Routing in IPv6

Das Internet besteht aus einem dynamischen Verbund unterschiedlichster Rechnernetze, die als autonome Systeme (AS[4]) bezeichnet werden. Zur Verdeutlichung der komplexen Struktur des Internets wurden einzelne organisatorische Bestandteile typitisiert und das Internet in drei Bereiche aufgeteilt.

- organisationsweite Netzwerke
- regionale (Provider-) Netzwerke, die angeschlossenen Organisationen als Zugang zum Internet dienen
- Transit-Netzwerke (backbones bzw. exchanger), die ausschließlich Daten zwischen Provider-Netzwerken vermitteln, also keinen Zugangsmöglichkeiten für Organisationen bieten

4 AS ist ein Netzwerk-Bereich, der genau einer Kontrollinstanz untersteht

Die aufgeführten Bereiche bilden eine Hierarchie, die sich in der Adressierung von IPv6 widerspiegeln (vgl. Kapitel 2.2). Um Daten über das Internet zu senden, müssen evtl. mehrere solche Bereiche (AS) passiert werden. Ein autonomes System bezeichnet ein Netz, was aus einem oder mehreren Subnetzen bestehen kann und unter einer zentralen Verwaltungsinstanz steht. In IPv6 werden AS auch als Routing-Domänen bezeichnet. [MIC99]

Innerhalb eines Subnetzes können Daten direkt (ohne Router) zugestellt werden. Hierfür werden die Link-Lokal-Adressen der Hosts genutzt. Sollen Daten an einen Empfänger außerhalb des eigenen Subnetzes gesendet werden, wird mindestens ein Router benötigt. Router verwenden Routing-Protokolle, mit denen Wege zwischen dem Quell- und Zielhost ermittelt werden. Anhand dieser Wege werden die Datenpakete zugestellt bzw. zum nächsten Router weitergeleitet. Das Weiterleiten von Datenpaketen aus einem Netz in ein anderes wird als Routing bezeichnet.

Informationen, die ein Router benötigt, um ein Paket zum Ziel weiterzuleiten, sind zum einem Routingtabellen und zum anderen Quell- und Zieladresse des Datenpakets. Routingtabellen enthalten unter anderem folgende Informationen:[HÜB96]

- eine Zieladresse (kann eine einzelner Rechner oder ein ganzes Netz sein)
- nächster Router oder ein direkt verbundenes Netz
- Routerinterface, auf dem das Datagram abzuschicken ist

Die Routingtabellen eines Routers können manuell eingetragen oder vom Router selbst durch Routing-Protokolle ermittelt werden. Hierbei verständigt sich ein Router mit anderen Routern um sich gegenseitig über vorhandene und bekannte Wege zum Ziel zu informieren.

Die Quell- und die Zieladresse des Datenpakets entnimmt der Router dem Datagram-Header des Protokolls, welches zur Layer3-Vermittlung eingesetzt wird. Da in Netzwerken meist mehr als nur eine Protokoll für die Kommunikation verwendet wird, sind Router in der Regel multiprotokollfähig. Das bedeutet sie können z.B. IPv4, IPv6 und IPX übertragen.

Entscheidend für die Wegewahl sowohl bei internen als auch bei externen Routing-Protokollen ist die Adressarchitektur des Netzwerkes. Bei der Entwicklung der IPv6-Adressarchitektur wurde Wert darauf gelegt, dass zum einem die Wegewahl relativ einfach vorgenommen werden kann (kleine Routingtabellen in Routern) und zum anderen der Wechsel des Providers einfach möglich ist.

Alle Routingentscheidungen werden aufgrund des Adresspräfixes vorgenommen. Das Datagram wird zu dem Netz weitergeleitet, bei dem die längste Übereinstimmung zwischen Ziel-Adresse und Adresspräfix vorliegt (longest prefix match). Zur Veranschaulichung soll folgendes Beispiel dienen:

Netzpräfix_1: 2ABC::0/16
Netzpräfix_2: 2ABC:34::0/32
Zieladresse: 2ABC:34:5678:3::25 –> Paket wird zum Netz_2 weitergeleitet

2.6.1 Intra Domain Routing

Intra Domain Routing-Protokolle (IRP) werden zum Routing innerhalb privater Netze bzw. innerhalb Autonomer Systeme genutzt. Es haben sich zwei IRP-Protokolle etabliert. Diese sind das Routing Information Protocol (RIP) und das Open Shortest Path First Protocol (OSPF). In den folgenden Abschnitten sind die grundsätzlichen Arbeitsweisen der Protokolle beschrieben und die Erweiterungen bzw. Veränderungen, welche für IPv6 vorgenommen wurden dargestellt. Da eine detaillierte Beschreibung von Routing-Protokollen nicht Bestandteil dieser Diplomarbeit ist, wird an dieser Stelle auf weiterführende Literatur wie beispielsweise [TAN00] verwiesen.

RIP - Routing Information Protocol

RIP tauscht in festen Zeitintervallen Informationen aus, indem es die ihm bekannten Daten einfach ins Netz sendet. Für das RIP ist ein Weg dann ideal, wenn er wenige Hops (Router, die das Paket passiert) enthält. Eine Gewichtung der Leitungskapazität findet nicht statt. Tritt zum Beispiel eine langsame Strecke mit 4 Hops gegen eine schnelle mit 8 Hops in Konkurrenz, verliert die schnellere. Ein weiteres Defizit von RIP ist, dass es auftretende Netzprobleme erst nach dem Ablauf seines Zeitintervalls (30 Sekunden pro Hop) erkennt. Die Nodes kennen bei der Initialisierung nur die direkt benachbarten Router. Dadurch breiten sich die

Routing-Informationen recht langsam aus, bei 15 Hops beträgt die Zeitspanne schon sieben Minuten. Da RIP auf dem Distance-Vector-Algorithmus basiert, kann das ‘Count-to-Infinity-Problem’ auftauchen, dadurch wird eine ausgefallene Strecke nicht erkannt. RIP ist in großen Netzen nicht einsetzbar, da es nach dem 15. Hop keine Segmente mehr erreichen kann. Es bietet in kleinen Installationen eine bequeme Lösung, um beispielsweise eine Default-Route zu verbreiten. [GRZ01]

RIP wurde für IPv6 angepasst und mit RIPnG[5] benannt. RIPnG ist im [RFC2080] definiert. Die grundsätzliche Arbeitsweise von IPv4 wurde praktisch unverändert übernommen. Lediglich die IPv6-Adressbezeichnungen und IPv6-Netzbeschreibungen mit den Adresspräfixen und die Präfixlängen wurden angepasst.

OSPF - Open Shortest Path First

OSPF ist ein Link-State-Protokoll und hält in jedem Knoten die Topologie-Informationen des Netzes bereit, um den kürzesten Weg zum anderem Knoten selbst zu berechnen. Diese Linkinformationen aktualisieren die Knoten regelmäßig mittels des Flooding-Algorithmus. Um nun die einzelnen Topologie-Datenbanken klein zu halten, wird das Netz in Teilbereiche unterteilt. Diese Areas sind Gruppen von Routern, welche exakt die gleiche Datenbank mit Routing-Informationen vorhalten. Der nächstübergeordnete Bereich ist der Backbone, der die Areas zu einem Autonomen System verbindet. Der Backbone kann nur einmal im gesamten OSPF-Netzbereich vorkommen und trägt immer die Area-Nummer 0.

OSPF hat sich in wegen seiner wesentlich leistungsfähigeren Arbeitsweise gegenüber RIP durchgesetzt, und das trotz seiner erheblich komplizierten Struktur. [GRZ01]

OSPF für IPv6 arbeitet nach den gleichen Prinzipien wie OSPF für IPv4, welche in [RFC2328] beschreiben sind. Für IPv6 mussten jedoch Anpassungen und Erneuerungen vorgenommen werden, welche in [RFC2740] spezifiziert sind. Die Änderungen gegenüber IPv4 sind im Wesentlichen folgende:

5 RIPnG: RIP next Generation, Routing Information Protokoll für IPv6

- OSPF für Vermittlung von Link-Segmenten und nicht wie bei IPv4 zwischen Subnetzen eingesetzt, Knoten am gleichen Link-Segment immer direkt erreichbar auch wenn nicht im gleichen Subnetz (Link-Lokale Adresse)
- IPv6-Adressbezeichnungen und IPv6-Netzbeschreibungen mit Adresspräfix und Präfixlänge wurden angepasst
- Multicastunterstützung nicht nur für IPv4 sondern auch IPv6
- kein Feld für Authentifikation im OSPF-Paket mehr, sondern Verschlüsselung über Authentification Header (AH) sowie optional über den Encapsulating Security Payload Header (ESP) [RFC2460]

2.6.2 Inter Domain Routing

BGP - Border Gateway Protocol

BGP ‘spricht’ in der Regel der gesamte Internet-Backbone zwischen den einzelnen Providern. Eine Ausnahme können zwei Provider bilden, die ihre Netze z.B. über OSPF miteinander vernetzen.

Zwischen den BGP-Partnern besteht eine permanente TCP-Verbindung. Zu Beginn einer Sitzung tauschen die Router zunächst ihre gesamten Topologie-Informationen aus, danach nur noch die Änderungen sowie einmal pro Minute ein ‘Keep-Alive’-Paket. Fällt nun ein Uplink aus, ist BGP in der Lage eine alternative Anbindung an die restlichen Netze im Internet zu suchen und seine Netze bei den anderen Backbone-Routern bekannt zu machen. BGP benutzt den Pathvector-Algorithmus. [GRZ01]

Die ursprüngliche Version von BGP wurde ausschließlich für IPv4 entworfen. In [RFC2858] und [RFC2545] wurde es um Multiprotokollfähigkeit erweitert und an IPv6 angepasst. Das erweiterte Protokoll wird auch als BGP+4 bezeichnet

2.6.3 Multicast Routing

Wie in Kapitel 2.2.2 beschrieben identifizieren Multicast-Adressen eine Gruppe von Empfängern. Hierdurch kann ein sendender Host ein Datagram in einer einzigen Operation an eine Menge von Empfängern senden, indem er es an eine entsprechende Multicast-Adresse adressiert. Die Menge der Empfänger eines Multicast-Paketes muss dem Sender nicht bekannt sein.

Für das Senden und Empfangen von Multicast-Paketen müssen zwei wesentliche Fragen beantwortet werden:

1. Wie wird ein Host Mitglied in einer Multicast-Gruppe und wie wird diese im Internet bekannt gegeben?
2. Wie werden Multicast-Pakete zu den Empfängern geroutet?

Um eine Mitgliedschaft in einer Multicast-Gruppe zu bekommen, wurde in IPv4 das Internet Group Management Protocol (IGMP) [RFC3228] verwendet. Bei IPv6 werden die im Multicast Listener Discovery Protocol (MLD) [RFC2710] definierten IPv6-ICMP-Nachrichten verwendet. Um eine Mitgliedschaft in einer Multicast-Gruppe zu beantragen werden ICMP Group Membership-Nachrichten gesendet (vgl. Kapitel 2.3). Wenn ein Router eine Anfrage für Mitgliedschaft erhält, trägt er die betreffende Multicast-Adresse in eine Liste ein. Das bedeutet, dass jeder Router eine Liste mit allen in seinem Netz genutzten Multicast-Adressen und dem entsprechenden Interface vorhält, wodurch er ankommende Multicast-Pakete gegebenenfalls duplizieren und auf den passenden Interfaces ausgeben kann.

Für das Routen von Multicast-Paketen werden spezielle Routing-Protokolle verwendet. Für IPv4 wurden die Protokolle DVMRP (Distance Vector Multicast Routing Protocol), MOSPF (Multicast für OSPF) und PIM (Protocol Independent Multicast) entwickelt. Für PIM gibt es zwei Varianten, die sich im Umgang mit den zu empfangenden und zu sendenden Multicast-Paketen unterscheiden. Zum einen gibt es PIM im Dense Mode[6] und zum anderen im Sparse Mode[7]

OSPF für IPv6 unterstützt die Erweiterungen für Multicast in OSPF [RFC1584] ebenfalls, dass heißt MOSPF kann auch für IPv6 genutzt werden. Neben MOSPF kann auch PIM im Sparse Mode [RFC2362] für Multicast Routing über IPv6 eingesetzt werden.

Die Mechanismen der verschieden Multicast-Protokolle werden an dieser Stelle nicht weiter untersucht. Für weiterführende Literatur sei auf angegebenen RFCs verwiesen.

6 Dense Mode: Verkehr wird überall hin zugestellt, wer ihn nicht haben will, muß ihn explizit abbestellen.
7 Sparse Mode: Verkehr wird nirgends zugestellt, ohne vorher beantragt zu sein

2.7 Migrationstechniken

Ein wesentliches Ziel bei der Entwicklung von IPv6 war es, eine möglichst einfache und fließende Migration von IPv4 zu IPv6 zu gewährleisten. Hierzu wurden in [RFC2893] Transitions Mechanismen für IPv6-Hosts und -Router definiert.

Fließende Migration bedeutet, dass während einer Übergangsphase beide Internet-Protokolle mit einander operieren können. Dies ist notwendig, da es praktisch unmöglich ist ein Netzwerk auf einem Schlag komplett umzustellen. Des Weiteren soll die vorhandene IPv4-Infrastruktur bei einem Übergang zu IPv6 mit genutzt werden.

In den folgenden Abschnitten werden Techniken vorgestellt, welche für eine Migration von IPv4 zu IPv6 entworfen wurden.

2.7.1 Dual IP-Layer

Eine Kompatibilität zur bestehenden IPv4-Infrastruktur ist besonders in der Anfangsphase von IPv6 nötig. So werden IPv4-Protokolle genutzt werden müssen, wenn zwei IPv6-Knoten nicht direkt über eine IPv6-Infrastruktur miteinander verbunden sind, aber dennoch miteinander kommunizieren wollen. Um dies zu ermöglichen kann eine Dual-IP-Layer-Implementation auf den Knoten eingesetzt werden. Die Dual-IP-Layer-Technik wird auch als Dual-Stack bezeichnet.

Dual-Stack bedeutet, dass auf den Knoten jeweils eine eigenständige Implementation von IPv4 und von IPv6 zur Verfügung steht, d.h. es ist sowohl IPv4 als auch IPv6 als Internet-Protokoll implementiert (vgl. Abbildung 2.11).

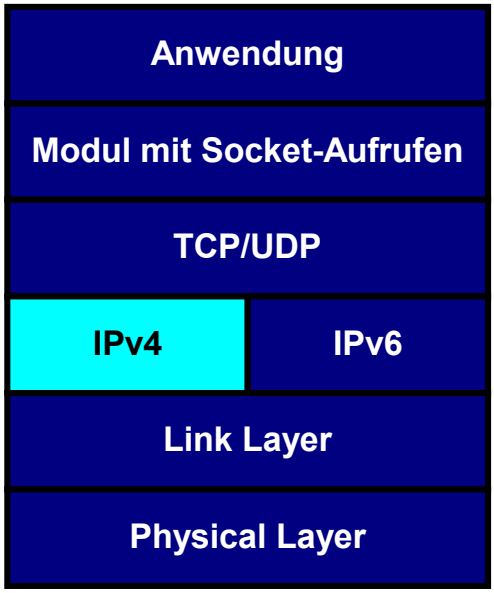

Abbildung 2.11: schematische Darstellung der Dual Stack Architektur am TCP/IP-Schichten Modell

Diese Knoten, auch als IPv4/IPv6-Knoten bezeichnet, können sowohl IPv4- als auch IPv6-Pakete senden und empfangen. Folglich können IPv4/IPv6-Knoten mit IPv6-Knoten über IPv6 und mit IPv4-Knoten über IPv4 kommunizieren. Das Routen der Pakete erfolgt dabei mittels des jeweiligen Routing-Protokolls.

In Tabelle 2.8 sind die möglichen Kommunikationsszenarien und das dabei verwendete Internet-Protokoll angegeben.

Source	***Destination***	***Protokoll***
IPv4/IPv6-Knoten	IPv6-Knoten	IPv6
IPv4/IPv6-Knoten	IPv4-Knoten	IPv4
IPv6-Knoten	IPv4/IPv6-Knoten	IPv6
IPv4-Knoten	IPv4/IPv6-Knoten	IPv4

Tabelle 2.8: durch Dual IP-Layer mögliche Kommunikationsszenarien und das verwendete Protokoll

Alle IPv4/IPv6 konfigurierten Knoten erhalten sowohl eine IPv4- als auch eine IPv6-Adresse.

Die Dual IP-Layer-Technik kann in Verbindung des Tunneling[8] von IPv6-Paketen über IPv4-Netze (vgl. Kapitel 2.7.3) angewendet werden. Ein IPv4/IPv6 Knoten der

8 Tunneling: Kapselung eines Datagram in den Datenteil eines anderen Datagram

das Tunneling unterstützt, kann entweder ausschließlich manuell konfiguriertes Tunneling unterstützen oder beides - manuell konfiguriertes und automatisches Tunneling. Es gibt demnach drei Implementierungsmöglichkeiten für einen IPv6/IPv4-Knoten: [RAU97]

- es wird kein Tunneling unterstützt
- Tunneling erfolgt ausschließlich über manuell voreingestellte Tunnelpfade
- sowohl manuell konfiguriertes, als auch automatisches Tunneling ist möglich

2.7.2 Sonderadressen für Übergang von IPv4 zu IPv6

Um eine fließende Migration von IPv4 zu IPv6 zu gewährleisten, wurden unter anderem die in Tabelle 2.9 aufgelisteten Adressen definiert.

Sonder-Adresse	***Adress-Präfix***	***Adressbildung***	***Bedeutung***
6to4-Adresse	2002/16	2002 + IPv4Addr +::+ Interface-ID	automatisches Tunneln von IPv4-Paketen über ein IPv6-Netz
IPv4-kompatible IPv6-Adresse	::/96	:: + IPv4Addr	Übertragung von IPv6-Adressen über ein IPv4-Netz
IPv4-mapped IPv6-Adresse	::FFFF/96	:FFFF + IPv4Addr	Übertragung von IPv4-Adressen über ein IPv6-Netz
IPv4-translated IPv6-Adresse	::FFFF:0/96	::FFFF:0 + IPv4Addr	Adresse wird automatisch in IPv4- bzw. IPv4-translated IPv6-Adresse umgewandelt

Tabelle 2.9: Sonderadressen für Migration von IPv4 zu IPv6

6to4-Adressen sind in [RFC3056] beschrieben und werden verwendet, um IPv4-Pakete automatisch über ein IPv6-Netzwerk zu tunneln (vgl. Kapitel 2.7.3). Die Adresse setzt sich aus dem Präfix 2002::/16, der sich aus dem Format Präfix 110_2 (vgl. Tabelle 2.2) und dem TLA-Wert 2_2 welcher für 6to4-Systeme reserviert ist bildet, sowie einer anschließenden 32 Bit IPv4-Adresse und den abschließenden 64 Bit Interface-Identifier zusammen.

Bei einer IPv4-kompatible IPv6-Adresse handelt es sich um eine Adresse eines ansonsten IPv6 fähigen Hosts. Sie wird genutzt, um IPv6-Adressen in einem IPv4-

Netzwerk zu transportieren, wobei ebenfalls Tunnel verwendet werden. Die oberen 96 Bit der Adresse sind Null, woraus sich der Adresspräfix ::/96 ergibt. Die restlichen 32 Bit werden durch die IPv4-Adresse gefüllt.

IPv4-mapped IPv6-Adressen werden für die Kommunikation zwischen einem IPv6/IPv4-Host und einem IPv4-Host genutzt. Die Adresse setzt sich aus dem Adresspräfix ::FFFF/96 und einer IPv4-Adresse zusammen und findet ihre Anwendung bei der in Kapitel 2.7.4 beschriebenen Protokollübersetzung.

Eine IPv4-translated IPv6-Adresse wird von einem Protokollübersetzer (vgl. Kapitel 2.7.4) erkannt und automatisch in eine IPv4-Adresse oder auch umgekehrt umgewandelt. Die unteren 32 Bit der Adresse bestehen aus einer IPv4-Adresse und die oberen 96 Bit bilden sich aus dem Adresspräfix ::FFFF:0/96.

2.7.3 IPv6-over-IPv4 Tunneling

In der Einführungsphase von IPv6 in eine bestehende IPv4-Umgebung werden IPv6-Netzwerke innerhalb des IPv4-Netzes implementiert. Hierdurch entstehen sogenannte IPv6-Inseln, die miteinander über das IPv4-Netz kommunizieren.

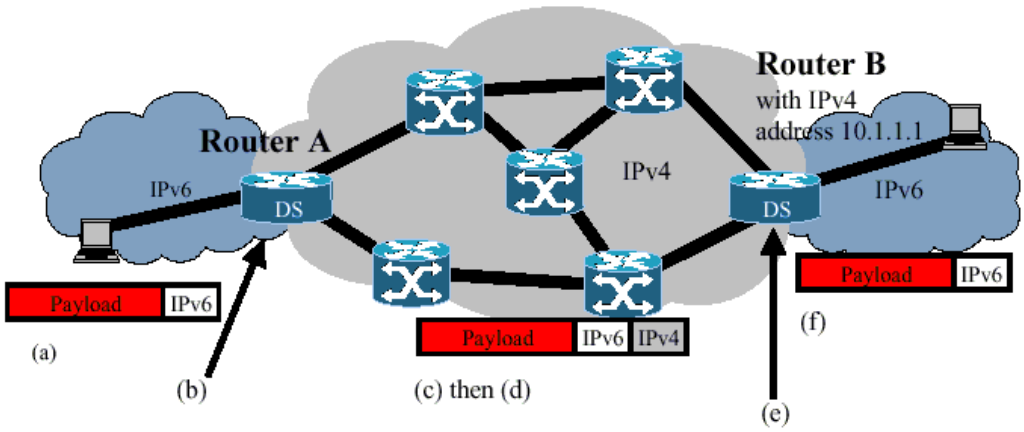

Abbildung 2.12: beispielhafte Darstellung des Ablaufs beim IPv6-over-IPv4-Tunneling [JMA02]

Im folgenden Absatz ist anhand der Abbildung 2.12 beschrieben, wie ein Daten-Paket aus einem IPv6-Netz über das IPv4-Netz in ein anderes IPv6-Netz getunnelt wird.

Router A und Router B sind Dual-Stack-Router.

(a) Ein Paket mit einer IPv6-Zieladresse erreicht Router A

(b) Router A erkennt anhand seiner Routingtabelle, dass das Paket an Router B weitergeleitet werden muss. Ebenfalls findet Router A die IPv4-Adresse 10.1.1.1 von Router B heraus (konfigurierte oder automatische Tunnel)

(c) Das IPv6-Paket ist in ein IPv4-Paket gekapselt und wird über das IPv4-Netz an Router B gesendet

(d) Das IPv4-Netz routet das Paket mit der Zieladresse 10.1.1.1, als wäre es ein IPv4-Paket, bis zum Router B

(e) Router B schaut sich das Paket an und erkennt, dass eine IPv6-Paket enthalten ist. Er entfernt den IPv4-Header und nutzt den IPv6-Header um die IPv6-Zieladresse mit seiner Routingtabelle zu vergleichen. Er erkennt, dass das Ziel innerhalb seines Netzes liegt.

(f) Router B sendet das IPv6-Paket zu seinem Ziel

Wie eben dargestellt, tunneln IPv4/IPv6-Hosts und -Router IPv6-Pakete über ein IPv4-Netz, indem die IPv6-Pakete in ein IPv4-Datagram gekapselt werden.

Neben dem dargestellten Szenario wäre es auch denkbar, dass nur ein einzelner IPv4/IPv6-Host in einer IPv4-Umgebung vorkommt. Hierbei gibt es keinen Router der die Pakete kapselt bzw. entkapselt, d.h. der Host selber muss die Aufgaben des Tunnel-Endpunktes übernehmen.

Bei einem Einsatz von IPv6-over-IPv4 Tunneling wird zwischen zwei Arten von Tunneln unterschieden.

- konfigurierte Tunnel
- automatische Tunnel

Konfigurierte Tunnel

Um zwei IPv6-Netze über ein IPv4-Netz durch einen Tunnel zu verbinden, jedoch aus der IPv6-Zieladresse eines Paketes nicht automatisch eine IPv4-Adresse abgeleitet werden kann, so muss ein Tunnel fest konfiguriert werden. Hierfür muss

auf beiden Tunnelendpunkten eine Konfiguration erfolgen. Ebenso müssen beide Endpunkte eine gültige eindeutige IPv4-Adresse besitzen.

Automatische Tunnel

Neben der Möglichkeit Tunnel manuell zu konfigurieren, können Tunnel auch automatisch eingerichtet werden, wenn aus der IPv6-Zieladresse der Tunnelendpunkt abgeleitet werden kann. Damit ein Router erkennen kann an welche IPv4-Adresse das Paket gesendet werden soll, werden IPv4-kompatible-IPv6-Adressen und 6to4-Adressen eingesetzt.

2.7.4 IPv4/IPv6 Protokollübersetzung

Neben dem Einsatz von Dual-IP-Architekturen ist bei der Einführung von IPv6 auch der Fall zu betrachten, dass IPv6-Knoten in einem neu entstandenen reinem IPv6-Netz weiterhin mit IPv4-Knoten, welche sich in einem reinen IPv4-Netz befinden, kommunizieren wollen. Dabei bedeutet reines IPv4- bzw. IPv6-Netz, dass innerhalb dieser Netze nur mit den jeweiligen Netzprotokollen gearbeitet und nur für das jeweilige Netzprotokoll eine Routing Infrastruktur aufgebaut wird.

Um die beschriebenen Kommunikationsszenarien zu ermöglichen, kann eine Protokollübersetzung auf Netzebene (Ebene 3 des OSI-Modells) vorgenommen werden. Hierfür wurden die Mechanismen Stateless IP/ICMP Translation (SIIT) [RFC2765] und Network Address Translation – Protocol Translation (NAT-PT) [2766] definiert.

Bei beiden Mechanismen werden IPv4-Header in IPv6-Header und umgekehrt transformiert. Dabei können aufgrund der erheblichen Unterschiede zwischen den beiden Protokollversionen (vgl. Kapitel 2.1) die folgenden Felder nicht übersetzt werden:

- Alle IPv4-Optionen im IPv4-Header
- IPv6-Routing-Header
- IPv6-Hop-by-Hop-Erweiterungsheader
- IPv6-Destination-Options-Header

Von den Erweiterungsheadern für IPSec (vgl. Kapitel 3) ist nur der ESP-Header im Transport-Mode[9] transformierbar. ESP im Tunnel-Mode und der Authentification-Header AH sind nicht übersetzbar, da einige Felder, welche übersetzt werden, in die Berechnung der verschlüsselten bzw. authentifizierten Daten einbezogen sind.

Des Weiteren sind von den ICMP-Nachrichten nur die ICMP-Typen transformierbar, welche in beiden Protokollversionen vorhandenen sind.

Die für beide IP-Versionen elementaren Felder wie z.B. Quell- und Zieladresse oder Next-Header bzw. Protokoll sind in beide Richtungen übersetzbar.

Stateless IP/ICMP Translation (SIIT)

Mit SIIT wird eine zustandslose Protokollübersetzung von IP- und ICMP-Protokollen bezeichnet. Dabei bedeutet zustandslos, dass jedes Paket für sich – ohne Speicherung eines Kontextes – übersetzbar ist.

Die Voraussetzung für SIIT ist eine dynamische Allokation von IPv4-Adressen. Um dies zu ermöglichen werden IPv4-translated IPv6-, IPv4-mapped IPv6- und IPv4-kompatible IPv6-Adressen (vgl. Kapitel 2.7.2) verwendet. Neben der Allokation der IPv4-Adresse ist als weitere Voraussetzung für eine Kommunikation eine öffentliche IPv4-Adresse für den IPv6-Host bereitzustellen.

Die Aufgaben bzw. Funktionen der Protokollübersetzung werden in einer so genannten SIIT-Box bzw. im Translator, einem mit Zusatzsoftware ausgestattetem Router, übernommen. Die SIIT-Box bzw. der Translator ist am Rand eines IPv6-Netzes zum Übergang an ein IPv4-Netz stationiert.

9 Transport Mode: Bezeichnet die Verschlüsselung von Endknoten zu Endknoten

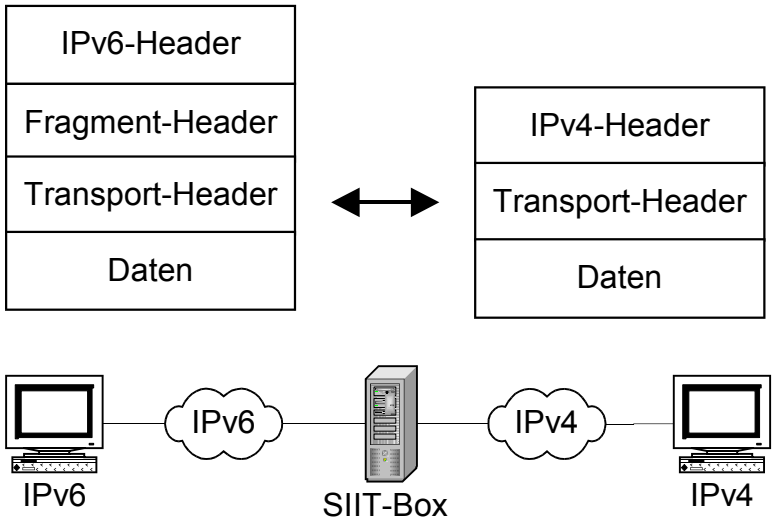

Abbildung 2.13: schematische Darstellung von SIIT zwischen einem IPv6-und einem IPv4-Host über eine SIIT-Box

Die Kommunikation zwischen einem IPv6-Host und einem IPv4-Host mit Hilfe einer SIIT-Box ist schemenhaft in Abbildung 2.13 dargestellt.

Network Address Translation – Protocol Translation (NAT-PT)

Mit NAT-PT wird die Voraussetzung einer IPv4-Adresse für den IPv6-Host in SIIT übergangen. Hierbei wird der aus IPv4 bekannte Mechanismus Network Adress Translation (NAT) mit der Protokollübersetzung gemäß SIIT kombiniert.

Der Vorteil gegenüber SIIT ist, dass für eine Kommunikation zwischen einem IPv6- und einem IPv4-Host für den IPv6-Host keine IPv4-Adresse mehr bereitgestellt werden muss, was in Anbetracht einer Adressknappheit von großem Nutzen sein kann.

Nachteil gegenüber SIIT ist, dass es sich bei NAT-PT um keine zustandslose Protokollübersetzung mehr handelt. Denn die Identifikation einer Kommunikationsbeziehung wird bei Einrichten einer Verbindung im NAT-PT Gateway gespeichert und für eine spätere Protokollübersetzung gebraucht. Hierfür müssen größere Rechen- und Speicher-Ressourcen auf dem NAT-PT-Gateway als für eine SIIT-Box bereitgestellt werden.

Das bekannte Problem von NAT, dass eine Verbindung nur von einem Host innerhalb des mit NAT betrieben Netzes aufgebaut werden kann, wird durch die als Bi-Directional-NAT-PT bezeichnete Betriebsart von NAT-PT umgangen. Hierdurch ist es möglich, auch eine Kommunikation aus einem IPv4-Netz in ein IPv6-Netz aufzubauen.

2.7.5 Transport Relay Translator (TRT)

Neben einer Protokollübersetzung auf Netzebene, wie sie im vorangegangen Abschnitt beschrieben wurde, bietet TRT die Möglichkeit einer Kommunikation zwischen reinen IPv6- und reinen IPv4-Netzen, durch eine Protokollübersetzung auf Transportebene (Ebene 4 des OSI Modells).

Diese Technologie ist vor allem für die Fälle vorgesehen, in denen IPv6-Knoten Dienste von IPv4-Servern in Anspruch nehmen wollen. In [RFC3142] wurde dafür ein so genannter „IPv6-to-IPv4-Transport-Relay-Translator“ (TRT) definiert.

Bei TRT findet eine Übersetzung von TCP/UDP-IPv6 nach TCP/UDP-IPv4 und umgekehrt statt. Hierbei können sich die Kommunikationspartner in reinen IPv6- bzw. IPv4-Netzen befinden, dass heißt es ist nur eine Routing-Infrastruktur für die entsprechende IP-Version notwendig.

Voraussetzung für TRT ist ein Modifizierter Name-Server, der netzlokale Pseudo-IPv6-Adressen mit Präfix des TRT-Systems erzeugt, welches diese Adresse bei Erhalt von Paketen durch IPv4-Adressen ersetzt.

Der wesentliche Vorteil dieses Verfahrens ist, dass keine Programme geändert werden müssen. Es ist lediglich die Installation eines TRT-Systems, ein speziell modifizierter DNS-Server und eine Routing-Konfiguration, welche dafür sorgt, dass die für das TRT-System bestimmten Pakete zu diesem geroutet werden, notwendig.

3 Sicherheit

Die Sicherheitsmechanismen von IPv6 umfassen die zwei grundlegenden Funktionsbereiche Authentifizierung und Verschlüsselung, welche durch die Erweiterungsheader Authentication und Encapsulation Security Payload (vgl. Tabelle 2.1) implementiert wurden.

Die Konzepte der Verschlüsselung und Authentisierung wurden ursprünglich als Bestandteil von IPv6 entwickelt, später jedoch herausgelöst und mit IPSec selbstständig weiterentwickelt.

Mittlerweile steht IPSec auch für das Internet-Protokoll der Version 4 zur Verfügung. Der große Unterschied zwischen beiden Protokollversionen liegt darin, dass IPSec in IPv4-Implementierungen rückwirkend eingebunden werden muss, während es bei IPv6 von Anfang an vorhanden ist.

3.1 Authentifikation mit AH

Die Authentifikation mit dem Authentication-Header (HA) von Datenpaketen schützt einen Kommunikationsbeziehung zwischen Partnern gegen Angriffe, bei denen ein Angreifer durch IP-Spoofing[10] eine falsche Identität vorgibt. Der sendende Partner weist durch Authentifikationsdaten seine Identität nach. Eine Authentifizierung des Absenders ist vor allem wichtig bei Aufgaben wie Konfigurationsdiensten (z.B. DHCP), Routing-Protokollen und mobilen Rechnern.

Für die Berechnung der Authentifizierungsdaten sind mindestens die beiden Hash Verfahren HMAC-MD5 gemäß [RFC2403] und HMAC-SHA-1 [RFC2404] in IPSec bereitzustellen. Zusätzlich werden die Daten des IP-Pakets, welche authentifiziert werden sollen, mit einem geheimen Schlüssel verkettet.

10 Beim IP-Spoofing verfälscht der Angreifer die IP-Adresse des Absenders, d.h. er täuscht einen anderen Absender vor.

3.2 Verschlüsselung mit ESP

Neben dem Nachweis der Identität des Kommunikationspartners durch Authentifikation ist für eine Übertragung sensibler Daten wie z.B. Passwörter oder Kundendaten eines Unternehmens eine Verschlüsselung dieser notwendig. Durch eine Verschlüsselung ist gewährleistet, dass die übertragenen Daten vor Dritten verborgen bleiben.

Um eine schnelle Verschlüsselung zu realisieren kommen in den meisten Fällen symmetrische Verschlüsselungsverfahren mit gleichen Schlüsseln beim Sender und Empfänger zum Einsatz. Eine Schwachstelle symmetrischer Verschlüsselung ist der Austausch der verwendeten Schlüssel.

Der in [RFC2406] definierte Encapsulation Security Payload (ESP) ist offen für verschieden Verfahren der Verschlüsselung. Ebenfalls können zusätzlich auch die verschlüsselten Daten authentifiziert werden. Wobei die Daten, welche vor dem ESP-Header liegen nicht verschlüsselt werden. Zu diesen Daten gehören z.B. Routing-Informationen und IP-Adressen des Senders und Empfängers der Datenpakete.

Sollen auch die Quell- und Zieladresse geheim gehalten werden, so wird das Paket samt Headerinformationen verschlüsselt, in ein neues Datenpaket gepackt und in einem sogenannten IP-in-IP-Tunnelmodus übertragen. Die Übertragung im Tunnelmodus bietet sich beispielsweise an, wenn private Netze über das Internet verbunden werden. Hierbei würden nur Informationen über die Tunnelendpunkte, welche beispielsweise die Router eines Netzwerks sein können, im Klartext über das Netz übertragen und Informationen über Adressen in diesem Netzwerk werden verschlüsselt. [MIK00]

Um eine Kompatibilität bei der Verschlüsselung zwischen Sender und Empfänger zu garantieren, müssen alle ESP-IPSec-Implementierungen mindestens den Data Encryption Standard (DES) mit 56 Bit langen Schlüsseln unterstützen. Das auf DES basierende IPSec-Verfahren ist in [RFC2405] beschrieben. Weitere Verfahren für ESP müssen in entsprechenden RFCs festgelegt werden, in denen die anzuwendenden Schlüsselalgorithmen und ergänzende Vorschriften zum Erzeugen der verschlüsselten Daten, z.B. zusätzliche Initialisierungsvektoren, und Längenangaben der Schlüssel, beschrieben sind.

3.3 Schlüsselaustausch

Eine Sicherheitslücke bei den Verschlüsselung von Daten ist der vorherige Austausch von Schlüsselinformationen zwischen den Kommunikationspartnern. Denn was nützt der beste Verschlüsselungsalgorithmus, wenn die verwendeten Schlüssel in die Hände nicht autorisierter Dritter gelangen.

Neben den Schlüsseln müssen sich Sender und Empfänger vor einer Übertragung verschlüsselter Daten auch noch über den verwendeten Verschlüsselungs-algorithmus (Security Association) einigen.

Für einen sicheren Schlüsselaustausch und die Übertragung der zu verwendenden Security Association wurden die Verfahren ISAKMP (Internet Security Association and Key Management Protocol) in [RFC2408] und IKE (Internet Key Exchange) in [RFC2409] beschrieben. Diese Varianten beruhen auf dem von W. Diffie und M.E. Hellmann 1976 vorgestellten Verfahren zum Austausch von Schlüsseln über unsichere Strecken.

Weiterführende Literatur zu den genannten Verfahren finden sich z.B. in [LIP01].

4 Mobile IPv6

Mit der wachsenden Anzahl an mobilen Endgeräten wie Notebooks, Sub-Notebooks, palm-sized PCs, Personal Digital Assistants (PDAs) und Handys, gewinnt die Datenübertragung zu diesen mobilen Endgeräten an immer zunehmender Bedeutung. Um den sprunghaften Anstieg von Internetbenutzern, von denen viele die oben genannten tragbaren Geräte benutzen gerecht zu werden, ist die Einführung einer Erweiterung zum Internet-Protokoll sinnvoll. Diese Erweiterung soll das Internet-Protokoll an die Mobilitätsbedürfnisse anpassen.

Ein wesentliches Problem besteht darin, dass wenn ein mobiler Host seinen Standort ändert, die ihm betreffenden IP-Konfigurationen geändert werden oder neue Routen propagiert werden müssen. Mit manueller Konfiguration ist dies nahezu unmöglich. Aber auch bei dynamischer Konfiguration, beispielsweise mit DHCP, entstehen Probleme wie z.B. offene Verbindungen auf Transportebene, welche bei Wechsel der IP- und Netzadresse nicht gehalten werden können.

Es existieren eine Reihe von Ansätzen und Ideen, welche diese Beschränkungen aufheben sollen. An erster Stelle ist hier Mobile IP als eine IP-Protokollerweiterung zu nennen. Auf Transportebene ermöglichen Protokollergänzungen wie Indirect-TCP (I-TCP) das unterbrechungsfreie Mitführen von TCP-Verbindungen über Adressänderungen hinweg. Auf Applikationsebene unterstützen Zero Configuration Networking-Protokolle das automatische Finden und Konfigurieren von lokalen Parametern wie IP-Adresse/Subnetzmaske, Service-Discovery, etc. [WIT02]

4.1 Überblick - Mobile IPv6

Um die Möglichkeiten, welche Mobile IPv6 (MIPv6) für mobile Endgeräte bietet zu zeigen, werden in den folgenden Abschnitten die Funktionsweise von MIPv6 und die mit MIPv6 entstandenen Erweiterungen für IPv6 beschrieben. Als Quelle für die dargestellten Inhalte wurde vorwiegend der aktuelle Internetdraft für Mobility Support in IPv6 [Draft03] genutzt. Hierbei ist zu erwähnen, dass es sich dabei nur um vorgeschlagene Standards handelt, welche sich in evtl. folgenden Drafts bzw. RFCs zu MIPv6 ändern können.

4.1.1 Terminologie in Mobile IPv6

Für das Verständnis von MIPv6 und den damit verbundenen Formulierungen werden an dieser Stelle einige der mit MIPv6 in [Draft03] definierten Begriffe erläutert.

Mobile Node (MN):	Ein Host, der seinen Zugangspunkt zu einem Netzwerk ändern kann und trotzdem über seine Heimatadresse erreichbar bleibt.
Home Address:	Die Heimat-IP-Adresse des Mobile Nodes. Diese Adresse wird dem Mobilen Knoten permanent zugeteilt. Das Netzpräfix dieser Adresse ist identisch mit dem des Heimatnetzes.
Home Link:	Der Link in dem das Subnetzpräfix des Mobile Nodes definiert ist.
Foreign Link:	Analog zum Home Link - jeder Link, der nicht der Heimatlink des Mobile Nodes ist.
Care-Of Address:	Ist die für den Aufenthalt eines Knotens in einem fremden Netz temporär vergebene IP-Adresse. Subnetzpräfix dieser Adresse ist das fremde Subnetzpräfix.
Home Agent (HA):	Ein Router im Heimatnetz des Mobile Nodes, dem der Mobile Node seine Care-Of-Adresse mitteilt. Pakete die an die Heimatadresse adressiert sind, werden vom Router angenommen und an die Care-Of-Adresse weitergeleitet (getunnelt).

Binding: „Bindung“ zum Heimatnetz; bedeutet die Assoziation zwischen Home-Adresse und Care-Of-Adresse.

Correspondent Node (CN): Als Correspondent Node wird der Kommunikationspartner des Mobile Nodes bezeichnet.

Foreign Subnet Prefix: Jedes Subnetzpräfix, das nicht das Heimat-Subnetzpräfix des Mobile Nodes ist.

Home Registration: Registrierung des Mobile Nodes mit seiner Care-Of-Adresse am Home Agent.

Home Subnet Prefix: Das Subnetzpräfix im Heimatnetz des Mobile Nodes

4.1.2 Vergleich mit Mobile IP für IPv4

Das Design von MIPv6 entstand sowohl aus den Erfahrungen, welche bei der Entwicklung von Mobile IP für IPv4 (Mobile IPv4) gesammelt wurden als auch den Möglichkeiten, welche das neue Internet-Protokoll bietet. MIPv6 teilt Funktionen mit Mobile IPv4, bietet dazu aber viele Erweiterungen und ist im Gegensatz zu IPv4 fest im Internet-Protokoll integriert und nicht nur aufgesetzt.

Ein Vergleich zwischen Mobile IPv4 und Mobile IPv6 zeigt, dass MIPv6 bezüglich Routing, Effizienz, Sicherheit, Robustheit und Performance wesentliche Verbesserungen bietet.

Das Routing von Datenpaketen zwischen einem MN und einem CN wurde bei Mobile IPv6 dahingehend optimiert, dass ein ineffizientes Triangle Routing zwischen MN, HA und CN, wie es bei Mobile IPv4 notwendig war, nur noch bei der Kontaktaufnahme zwischen CN und MN auftritt. Hierdurch wird die Gefahr, dass der HA zu einen möglichen Flaschenhals bei der Kommunikation mit einem MN wird, umgangen und ein wesentlich effizienteres Routing erreicht.

Neben den Performanceverbesserungen bietet MIPv6 bessere Voraussetzungen um die Sicherheitsanforderungen welche bei einer mobilen Kommunikation gefordert werden (vgl. Kapitel 4.2), zu gewährleisten. Denn durch die Sicherheitsmechanismen, welche das neue Internet-Protokoll bereits implementiert hat, sind keine zusätzlichen Installationen mehr vorzunehmen um beispielsweise IPSec einzusetzen.

Eine weitere Verbesserung von MIPv6 bieten die Erweiterungen von IPv6 (vgl. Kapitel 4.1.4), die einen kurzzeitigen Ausfall oder eine Umstrukturierung des Heimatnetzwerkes ohne Konsequenzen für bestehende Kommunikationen mit dem MN ermöglichen.

In Tabelle 4.1 sind noch einmal zusammenfassend die genannten Vorteile, welche Mobile IPv6 gegenüber Mobile IPv4 bietet dargestellt.

	Mobile IPv4	***Mobile IPv6***
Routing	optimales Routing nur wenn Mobile Node (MN) im Heimatnetzwerk, ansonsten uneffizientes „Triangle“-Routing	Prinzipiell immer optimales Routing möglich, falls Correspondent Node (CN) die Care-Of-Adresse kennt
Flaschenhälse	Home Agent (HA) ist möglicher Flaschenhals, da sämtlicher Verkehr zum MN über ihn abgewickelt wird	HA wird erheblich entlastet, da CNs nun direkt mit MNs kommunizieren können
Sicherheit	Beglaubigung nur bei der Registratur, und auch da nur zwischen HA und MN vorgeschrieben	Beglaubigung und Verschlüsselung theoretisch überall möglich, da durch IPv6 unterstützt
Robustheit	Benutzte Foreign Agents (FA) und HAs dürfen nicht ausfallen. Standard Mobile IP schwer zu erweitern, da selbst nur Aufsatz‘	Kurzzeitiger Ausfall / Umkonfiguration des HA wird Dank Automatic Home Agent Discovery gemeistert. IPv6 ist wesentlich einfacher zu erweitern, damit auch Mobile IPv6
Performance	Aufgrund der IPv4-Voraussetzungen und des nicht-optimalen Routings keine gute Performance	Aufgrund der wesentlich besseren Voraussetzungen durch IPv6 (einheitliche Header, weniger Overhead)

Tabelle 4.1: Vergleich von Mobile IPv4 und Mobile IPv6

4.1.3 Funktionsweise von Mobile IPv6

Mit Mobilität eines Hosts wird dessen Möglichkeit bezeichnet seinen Standort zu wechseln. Wie jeder „Reisende“ hat auch ein Mobile Node ein zu Hause, was in diesem Fall als das Heimatnetz bezeichnet wird. In Abbildung 4.1 ist beispielhaft die Mobilität eines Mobile Node und die bei einer mobilen IP-Kommunikation beteiligten Partner dargestellt.

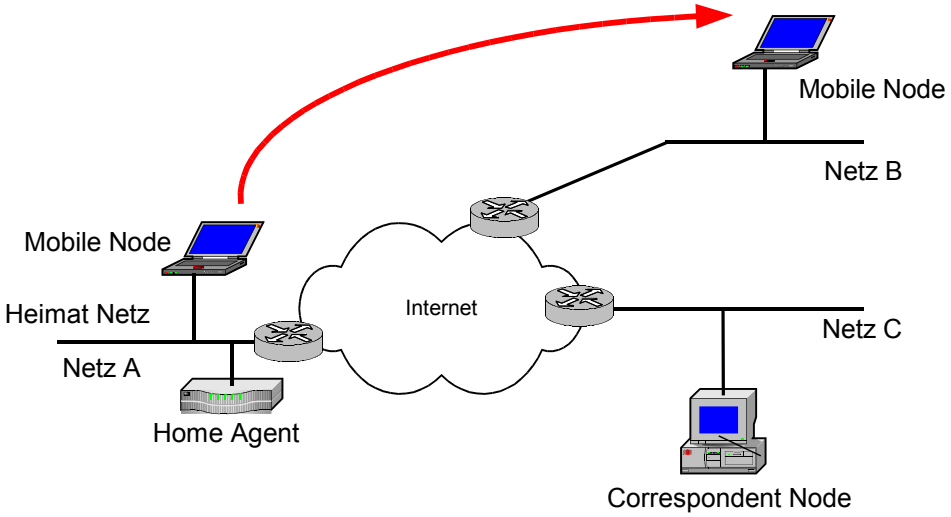

Abbildung 4.1: schematische Darstellung eines Mobile-IP-Szenario mit den dabei beteiligten Instanzen

Ein Mobile Node (MN) ist immer über seine statische Heimatadresse adressierbar, egal ob sich dieser in seinem Heimatnetz befindet oder in einem fremden Netzwerk angemeldet ist. Während der MN zu Hause ist, werden Pakete, welche an seine Heimatadresse adressiert sind durch herkömmliche Routingmechanismen an diesen gesendet, als wäre er ein „ganz normaler“ stationärer Host.

Wenn ein MN an einem fremden Netz angeschlossen ist, ist er über eine oder mehrere sogenannte Care-Of-Adressen erreichbar. Eine Care-Of-Adresse ist die Adresse, die ein MN in einem fremden Netzwerk beispielsweise durch Autokonfiguration oder einen DHCP-Server erhält. Das Subnetzpräfix der Care-Of-Adresse ist eines der Präfixe welche das fremde Netz identifizieren. So lange sich der MN im fremden Netz befindet, werden alle Pakete die an dessen Care-Of-Adresse adressiert sind an ihn weitergeleitet.

Die Verbindung zwischen der Heimat-Adresse und der Care-Of-Adresse wird als Binding (Bindung) des MN bezeichnet.

Während der Abwesenheit eines MN von seinem Heimatnetz registriert er eine seiner Care-Of-Adressen an einem Router in seinem Heimatnetz mit der Aufforderung an den Router die Funktionen eines ome Agentgent (HA) für den MN

zu übernehmen. Diese Binding-Operation wird durch eine Binding-Update Nachricht des MN an den HA eingeleitet. Der HA antwortet darauf mit einer Binding-Acknowledgement-Nachricht (vgl. Abbildung 4.2). Die Care-Of-Adresse, welche mit dem Binding-Update bekannt gegeben wurde, wird als Primäre Care-Of-Adresse des MN bezeichnet. Von nun an nutzt der HA des MN stellvertretend Neighbor-Discovery, um alle Pakete, welche an die Heimat-Adresse(n) des MN gesendet werden, abzufangen. Jedes abgefangene Paket wird an die Primäre Care-Of-Adresse des MN getunnelt. Das Tunneln wird durch IPv6-Encapsulation [RFC2473] durchgeführt, wobei als Destination-Adresse im äußeren IPv6-Header die Primäre Care-Of-Adresse des MN eingetragen wird.

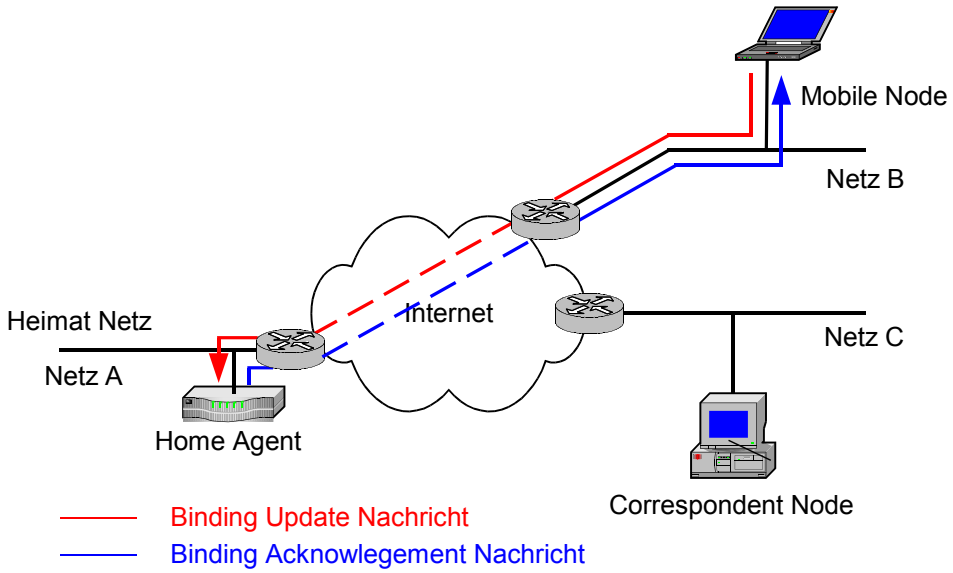

Abbildung 4.2: Veranschaulichung wie ein Mobile Node seine Care-Of-Adresse bei seinem Home Agent registriert

Jeder Host, der mit einem MN kommuniziert, wird als Correspondent Node (CN) bezeichnet, unabhängig davon ob sich der MN zu Haus oder in einem fremden Netz befindet. Dem MN ist es durch die Correspondent-Binding-Procedure möglich, dem CN seinen derzeitigen Standort mitzuteilen. Jeder Host besitzt sogenannte Binding-Cache-Einträge, welche die Care-Of-Adressen von Mobilen Hosts beinhalten. Bevor ein Host ein IP-Paket sendet, schaut er nach, ob er einen Binding-Cache-Eintrag für die Destination-Adresse besitzt, mit dem er gegebenenfalls das IP-Paket direkt an die Care-Of-Adresse des MN senden kann. Hierfür wird ein neu definierter Typ des IPv6-Routing-Header genutzt. Wenn kein

Eintrag vorhanden ist, sendet er das Paket an die Heimat-Adresse des MN, wobei das IP-Paket wie beschrieben vom HA abgefangen und zum MN getunnelt wird. In diesem Fall erhält der MN das getunnelte Paket, an welchem er erkennen kann, dass ein CN kein Binding-Eintrag für ihn besitzt. Der MN kann dann eine Bindung mit dem CN eingehen. Das bedeutet, der HA wird lediglich für eine Kontaktaufnahme des CN mit dem MN gebraucht und der weitere Datenaustausch kann direkt zwischen den beiden Partnern ablaufen.(vgl. Abbildung 4.3)

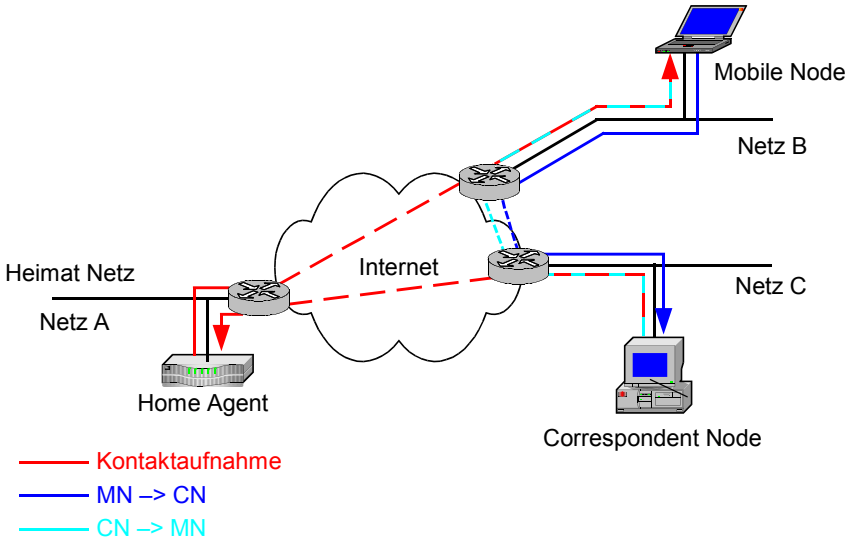

Abbildung 4.3: Veranschaulichung des Kommunikationsverlaufs zwischen einem Correspondent Node und einem Mobile Node

Es wird erwartet, dass CNs Pakete direkt an die Care-Of-Adressen von MNs senden, so dass die HAs selten in die Kommunikation zwischen Mobile und Correspondent Node involviert werden. Dies ist wichtig für die Skalierbarkeit und Ausfallsicherheit sowie für die Minimierung des gesamten Netzverkehrs. Pakete die also direkt an die Care-Of-Adresse gesendet werden verringern die Arbeitslast des HA und den Netzverkehr zum Heimatnetz. Ebenso werden mögliche Ausfälle des HA, Probleme im Heimatnetz oder Probleme mit dem Verkehr zum bzw. vom Heimatnetz nicht dazu führen, dass die Verbindung zwischen Mobile- und Correspondent Node gefährdet ist.

Wenn ein MN, während er nicht zu Hause ist, Nachrichten senden will, kann er generell einen Tunnel über den HA nutzen. Da dies aber wieder zu den bei einer über den HA laufenden Kommunikation beschriebenen Probleme führen könnte,

kann der MN, vorausgesetzt der CN besitzt einen passenden Binding-Cache-Eintrag, IP-Pakete auch direkt an den CN senden. Hierbei trägt der MN seine Care-Of-Adresse in das Source-Adress-Feld des IP-Headers ein und fügt die neue definierte Home-Address-Destination-Option, welche die Home-Adresse des MN beinhaltet, ein. Durch die Verwendung dieser Option kann ein MN einem CN die Heimatadresse mitteilen. Dies macht die Nutzung von Care-Of-Adressen für Schichten oberhalb der IP-Schicht transparent, da eine bestehende Verbindung nicht ständig mit einer neuen Adresse konfrontiert wird.

Für den Fall, dass während der Abwesenheit des MN vom Heimatnetz einige Links im Heimatnetz neu konfiguriert werden oder beispielsweise der Router, welcher die Aufgabe des HA übernommen hat durch einen anderen Router ersetzt wird und sich dessen IP-Adresse ändert, stellt Mobile IPv6 mit Dynamic-Home-Agent-Address-Discovery einen Mechanismus bereit, mit dem ein MN die IP-Adresse seines HAs herausfinden kann. Hierfür sendet der MN eine ICMP-Home-Agent-Address-Discovery-Request-Nachricht an die Anycast-Adresse der HAs mit dem Subnetzpräfix des Heimatnetzes. Mit dieser Nachricht erreicht der MN einen Router, welcher als HA eingesetzt wird. Dieser HA antwortet mit einer ICMP-Home-Agent-Address-Discovery-Reply-Nachricht, welche eine Liste aller verfügbaren Home Agents im Heimatnetz beinhaltet.

4.1.4 Erweiterungen von IPv6 für Mobile IPv6

Um die Funktionen von MIPv6 zu ermöglichen, sind einige Erweiterungen des IPv6-Protokolls notwendig geworden. In Kapitel 4.1.3 wurde der Einsatz dieser Erweiterungen bereits vorweggenommen. An dieser Stelle werden deshalb nähere Erläuterungen diese Erweiterungen, zu denen neue ICMP Nachrichten, ein neuer Erweiterungsheader und eine neue Address Destination Option gehören, gegeben.

Neue IPv6 ICMP Nachrichten

Zu den Erweiterungen von IPv6, welche mit MIPv6 definiert wurden, gehören vier neue ICMP Nachrichten Typen, welche in Tabelle 4.2 beschrieben werden.

ICMP-Type	ICMP-Nachricht	Beschreibung
150	Home Agent Address Discovery Request	genutzt für Home Agent Address Discovery
151	Home Agent Address Discovery Reply	
152	Mobile Prefix Solicitation	genutzt für Netzwerk-Renumbering und Adresskonfiguration der Mobile Nodes
153	Mobile Prefix Advertisement	

Tabelle 4.2: für Mobile IPv6 neu definierte ICMP Nachrichten

Neuer IPv6 Erweiterungsheader

Mit dem Mobility Header definiert Mobile IPv6 einen neuen Erweiterungsheader für IPv6. Ein schemenhafte Darstellung des Mobility Header ist in Abbildung 4.4 dargestellt.

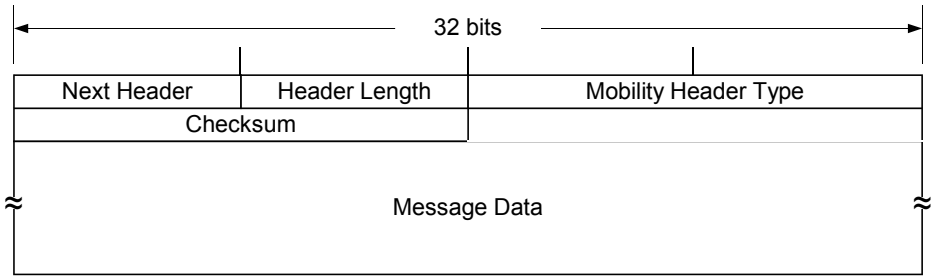

Abbildung 4.4: schematische Darstellung des IPv6 Mobility Header

Die Felder *Next Header* und *Header Length* sind analog den Feldern anderer Erweiterungsheader (vgl. Kapitel 2.1.2).

Das Feld *Mobility Header Type* gibt den Typ der Nachricht an. Hierfür sind die in Tabelle 4.3 aufgelisteten Werte definiert.

Type	*Nachricht*	*Beschreibung*
1	Home Test Init	Diese vier Nachrichten werden genutzt, um die Return Routability Procedure einzuleiten.
3	Home Test	
2	Care-Of Test Init	
4	Care-Of Test	
5	Binding Update	Die Binding Update Nachricht wird vom MN genutzt, um seinem Home Agent oder den CNs über seine neue Care-Of-Adresse zu informieren.
6	Binding Acknowledgement	Die Binding Acknowledgement Nachricht wird genutzt, um den Empfang eines Binding Updates zu bestätigen.
0	Binding Refresh Request	Die Binding Request Option dient der Anfrage eines Knotens an einen Mobilen Rechner seine derzeitige Care-Of-Adresse mitzuteilen.
7	Binding Error	Die Binding Error Nachricht wird von CN genutzt, um einen Fehler, der beim Versuch die Home Address Destination Option ohne passendes Binding zu nutzen, aufgetreten ist, mitzuteilen

Tabelle 4.3: Mobility Header Nachrichten

Neue IPv6 Destination Option für Mobile IPv6

Mobile IPv6 definiert mit der Home-Address-Destination-Option eine neue Destination-Option. Diese Option wird von einem MN während seiner Abwesenheit vom Heimatnetz in einer Nachricht genutzt, um den Empfänger dieser Nachricht über die Heimat-Adresse des MN zu informieren. Hierbei dürfen für die Heimat-Adresse keine Multicast-, Link-Lokal-, loopback- und IPv4-mapped-Adressen genutzt werden.

Für die Angabe der Home-Address-Destination-Option, dessen Format schemenhaft in Abbildung 4.5 dargestellt ist, wird der Routing-Erweiterungsheader genutzt.

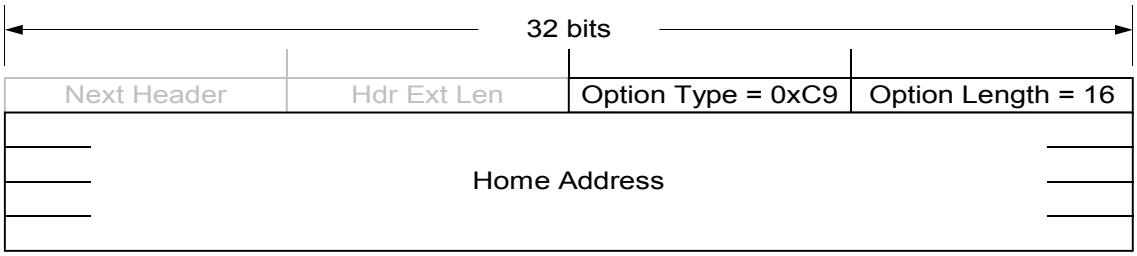

Abbildung 4.5: schematische Darstellung der Home Address Destination Option

Das Feld *Option Type* kennzeichnet die angegebene Option und hat für die Home Address Option den Wert $C9_{Hex}$.

Option Length gibt die Länger der Option ohne die Felder Type und Length in Oktetts an. Der Wert dieses Feldes muss immer 16 sein.

Im Feld *Home Address* (132 Bit) wird die mitzuteilende Heimatadresse des MN eingetragen.

4.2 Sicherheitsanforderungen bei Mobile IP

In diesem Abschnitt werden die Sicherheitsanforderungen, welche an die beteiligten Instanzen eines bei Mobile-IP-Szenario (vgl. Kapitel 4.1.3) gestellt werden müssen, erläutert. Die dafür zu fordernden Sicherheitsziele lassen sich in folgende Punkte unterteilen: [PSS01]

- Authentizität der beteiligten Instanzen (Mobile Rechner und deren Benutzer sowie Infrastrukturen von Fremd- und Heimatnetz)
- Authentizität und Integrität der übertragenen Daten
- Schutz vor unerwünschten Datenpaketen (in ortsfesten Netzen häufig durch die Paketfilterfunktionalität von Firewalls erbracht)
- Vertraulichkeit der übertragenen Daten
- Vertraulichkeit der Benutzeridentität und des Aufenthaltsortes

Authentizität der beteiligten Instanzen

Die Authentizität von Mobile Nodes bzw. deren Benutzern ist für ein Fremdnetz für eine verlässliche Zugangskontrolle, eine Abrechnung erbrachter Dienste und für die Rückverfolgung im Falle missbräuchlicher Nutzung unverzichtbar.

Die Authentizität des Heimatnetzes ist vor allem für Mobile Nodes wichtig, um die Verfügbarkeit der Mobilitätsunterstützung zu gewährleisten und den Missbrauch von Informationen über den Aufenthaltsort zu verhindern.

Die Authentizität des Fremdnetzes muss gewährleistet werden, wenn sich ein Mobiler Rechner nur in bestimmten Fremdnetzwerken aufhalten darf oder der Mobile Rechner bestimmte Dienste im fremden Netz in Anspruch nehmen möchte.

Um die genannten Ziele zu erreichen, muss bei der Mobile-IP-Registrierung eine kryptografische Authentisierung zwischen Mobile Node und Heimatnetz sowie zwischen Mobile Node und Fremdnetz vorgenommen werden. Eine Authentisierung zwischen Mobile Node und Fremdnetz stellt in der Praxis Schwierigkeiten dar, denn es ist nicht zu erwarten, dass ein Fremdnetz zu jedem potenziell anwesenden Mobile Rechner eine Sicherheitsassoziation vorhalten wird. Mögliche Lösungen dieses Problems liegen eventuell in einer Public Key Infrastruktur (PKI) oder durch eine indirekte Authentisierung über das Heimatnetz bzw. eine AAA-Infrastruktur (Authentisierung, Autorisierung, Accounting [RFC2977].

Authentizität und Integrität der übertragenen Daten

Die Authentizität und Integrität der übertragenen Daten ist sowohl für das Heimatnetz als auch den Mobile Node von großer Bedeutung. Deshalb muss für die Übertragung von Daten zwischen dem Mobilrechner und dem Heimatnetz eine kryptographische Sicherung erfolgen.

Für das Fremdnetz ist die Integrität von übertragenen Daten relativ unwichtig. Jedoch muss auf eine verlässliche Authentifizierung des Mobile Nodes Wert gelegt werden, um einen eventuellen Missbrauch des Netzes durch gefälschte Quelladressen zu verhindern.

Schutz vor unerwünschten Datenpaketen

Der Schutz vor unerwünschten Datenpaketen muss sowohl für das Heimatnetz als auch für das Fremdnetz gewährleistet sein.

Um das Fremdnetz vor dem Einbringen unerwünschter Daten durch anwesende Mobile Nodes zu sichern, könnten Mobile Nodes im Fremdnetz in einem virtuellen

Netz, welches nur wenige durch Firewalls geschützte Übergangspunkte zum Rest des Fremdnetz besitzt, isoliert werden. Die virtuellen Netze können durch dedizierte Subnetze realisiert werden, in denen keine ortsfesten Rechner stationiert sind. Zu dem muss sichergestellt werden, dass von entfernten Heimatagenten zu Mobile Nodes getunnelte Datenpakete im Fremdnetz nicht in die ortsfesten Teile des Fremdnetzes gelangen, sondern durch virtuelle Netze an den entsprechenden Mobile Node geroutet werden.

Das Heimatnetz ist durch Mobile Nodes gefährdet, denn diese stellen potenzielle Netzübergänge zu fremden Netzen dar. Deshalb müssen Mobile Nodes besonders sorgfältig konfiguriert werden und sollten nicht mit Routingfunktionalitäten ausgestattet sein.

Vertraulichkeit der übertragenen Daten

Neben der Authentizität und der Integrität spielt die Vertraulichkeit eine nicht so wesentliche Rolle. Dennoch müssen hierfür ähnliche Anforderungen gestellt werden. Der Mobile Node ist Teil des Heimatnetzes, so dass die ausgetauschten Daten vertrauliche Informationen enthalten können, welche durch Verschlüsselung gesichert werden müssen.

Für das Fremdnetz ist die Vertraulichkeit von Daten zwischen Mobile Node und Heimatnetz in der Regel von keiner Bedeutung.

Vertraulichkeit der Benutzeridentität und des Aufenthaltsortes

Dieses Schutzziel liegt in erster Linie im Interesse des Benutzers eines Mobilrechners der seinen Aufenthaltsort vor Dritten verbergen möchte. Aber auch für das Heimatnetz wird es von Interesse sein die Aufenthaltsorte seiner Mobile Nodes vor Dritten zu verbergen.

Um die Vertraulichkeit der Identität gegenüber dem Fremdnetz sowie gegenüber Dritten zu wahren, kann bei der Registrierung und beim Datenaustausch eventuell an Stelle der Heimatadresse ein temporäres Pseudonym verwendet werden. Dieses Pseudonym entspräche in etwa der TMSI[11] (Temporary Mobile Subscriber Identity) im GSM-Mobilfunkstandard. Die Abbildung zwischen echter und temporärer Heimatadresse wäre nur dem Heimatnetz bekannt. Dem Fremdnetz muss es lediglich möglich sein, anhand der Adresse das zuständige Heimatnetz zu bestimmen.

4.3 Einsatz von Mobile IPv6

Mit Mobile-IPv6 bieten sich eine Reihe von Einsatzmöglichkeiten, die in zwei wesentlichen Gebieten Verwendung finden werden. So setzt die Mobilfunk-industrie auf Funktionen und Programme die durch die mögliche Adressierung aller Handys und GSM-Geräte und deren ortsunabhängige Erreichbarkeit möglich werden. Neben dem Mobilfunkbereich ist das Thema Mobile-IPv6 auch für klassische[12] Netzwerkstrukturen von großer Bedeutung.

Für eine „grenzenlose“ Mobilität auch zwischen diesen beiden Gebieten haben die Mitglieder der Internet Engineering Task Force (IETF) die Arbeitsgruppe „Seamless Mobility“ (Seamoby) gegründet [SEAMO]. Sie erstellt derzeit ein Dokument, in dem die Faktoren aufgelistet sind, die beim Wechsel zwischen unterschiedlichen Netzwerktechniken, wie etwa beim Übergang von Wireless LANs zu Mobilfunknetzen auf Grundlage des General Packet Radio Service (GPRS) oder des Universal Mobile Telecommunications System (UMTS), zu berücksichtigen sind. Hierdurch wird es in Zukunft keine Trennung zwischen dem Medium Handy und dem Medium PC mehr geben, sondern es wird eine direkte Kommunikation zwischen diesen möglich sein.

In den folgenden Abschnitten werden einige Anwendungsmöglichkeiten von Mobile-IPv6 an verschiedenen Beispielen dargestellt.

Echtzeitanwendungen wie IP-Telefonie und Videokonferenzen werden durch Quality of Service Eigenschaften von IPv6 und der Anzahl an IP-Adressen zwischen allen Endgeräten möglich. Aber was nützt eine IP-Adresse, wenn diese

11 Die TMSI ist eine dem Teilnehmer temporär zugewiesene Teilnehmernummer.
12 bezieht in diesen Zusammenhang Wireless LAN (WLAN) mit ein

sich bei einem Standortwechsel eines Teilnehmers ändert, was zur Folge hat, dass dieser Teilnehmer nicht mehr adressiert werden kann und eventuell bestehende Verbindungen unterbrochen werden. In dieser Situation kommt Mobile-IPv6 zum Einsatz, wodurch die Erreichbarkeit eines solchen Teilnehmers weiterhin sichergestellt wird.

Durch die Anzahl der durch IPv6 bereitgestellten Adressen kommt der Gedanke nahe, nicht nur PCs und Handys mit einer solchen auszustatten, sondern auch alle anderen technischen Geräte wie Videorecorder und sogar Autos. Hierdurch lassen sich zum Beispiel Fernwartungen und Softwareupdates durchführen. Für solche Möglichkeiten muss eine Adressierung gewährleistet werden. Was bei einem Videorecorder der in der Regel Standort gebunden ist noch einfach zu realisieren ist, scheint bei einem Auto, dessen Aufenthaltsort nicht bekannt sein wird, unmöglich. Durch Mobile-IPv6 können aber beide Geräte ohne weiteres adressiert werden, unabhängig vom Aufenthaltsort und den für die Internetverbindung genutzten Provider. Hierfür müssen lediglich eine Home-Adresse bekannt sein und ein Heimatagent zur Verfügung stehen, was z.B. von Seiten der Hersteller gewährleistet werden kann.

In der Automobilindustrie wird unter dem Stichwort Telematik, unter der man im weiteren Sinne den interaktiven Datenaustausch über ein drahtloses Kommunikationsnetz versteht, Mobile-IPv6 ein weites Anwendungsspektrum finden.

4.4 Entwicklungsstand für Mobile IPv6 Anwendungen

Das neue Internet-Protokoll bietet durch seinen großen Adressraum die Möglichkeit eine auf absehbare Zeit ausreichende Anzahl von Geräten mit einer IP-Adresse zu versehen. Zu dem wird durch die Mobilitätsunterstützung von IPv6 eine Erreichbarkeit all dieser Geräte, unabhängig von ihrem Standort, gewährleistet. Eine Adressierung aller Geräte und die neuen Technologien für nicht drahtgebundene Kommunikationen wie GPRS, UMTS und WLAN bieten neue Möglichkeiten der Mobilen Kommunikation.

Wo sich eine mobile Kommunikation bisher auf Telefonieren und mit großen Konfigurationsaufwand versehenden Netzwerkverbindungen mit mobilen Geräten

wie Laptops beschränkte, kann mit den neuen Technologien ohne jeglichen Konfigurationsaufwand, abgesehen von Sicherheitsanforderungen (vgl. Kapitel 4.2), eine IP-Kommunikation mit mobilen Geräten geschaffen werden.

Den größten Nutzen von IPv6 und den damit gegebenen Möglichkeiten für Adressierung und Mobilität sehen die Hersteller von Telekommunikationsgeräten und die Betreiber von Mobilfunknetzen. Den Herstellern und Betreibern bieten sich neue Möglichkeiten die paketvermittelnden Mobilfunknetze wie GPRS und UMTS zu nutzen.

Auch die Nutzung und der Einsatz von WLANs wird effizienter und komfortabler. So konnte bisher ein Laptop, welcher über WLAN an das Netzwerk angeschlossen wurde nicht ohne eine Unterbrechung seiner bestehenden Verbindungen in ein anderes Subnetz wechseln.

Trotz der wesentlichen Verbesserungen und Neuerungen des neuen Internet-Protokolls gegenüber IPv4 lässt die Einführung von IPv6 im produktiven und kommerziellen Umfeld auf sich warten. So sind zwar erste Implementierungen auf Routern und Betriebssystemen namhafter Hersteller vorgenommen wurden, welche aber in den meisten Fällen von den Herstellern nur als Entwicklungsstatus freigegeben sind.

Ein wesentlicher Druck für den Durchbruch von IPv6 geht vom Einsatz des neuen Protokolls im Mobilfunkbereich aus. So sind Schlagzeilen wie „Mobilfunk hilft IPv6“ [SIE00], „Deutlicher Schub für das Internet-Protokoll IPv6 durch den Boom Mobiler Endgeräte“ [ORD01] und „IP-Adressen für jedes Handy“ [MAN02] seit über zwei Jahren in renommierten Fachzeitschriften zu finden. Durch den Mobilfunkboom und der Tatsache, dass per GPRS in wenigen Jahren fast jedes Handy Online sein kann werden allein in diesen Bereichen sehr viele IP-Adressen benötigt und somit der Einsatz von IPv6 unverzichtbar. Die Tatsache, dass Mobilfunkanbieter auf die Einführung von IPv6 angewiesen sind, führte in den letzten Jahren zu einem erheblichen Entwicklungsschub in den Technologiebereichen der Mobilen Kommunikation und ein reges Interesse neue Features und Funktionen auf den Markt zu bringen. So gründete Nokia im Mai 2000 ein unternehmensweites Programm zur Förderung des IPv6-Standards und präsentierte während des World Telecommunications Congress und des IPv6-Forums in Birmingham

(Großbritannien) eine IPv6 Live-Demonstration. Auch Ericsson stellte bereits im November 2000 zusammen mit BT (British Telecom) Wireless und SmarTone in Honkong eine erfolgreiche End to End Demonstration von IPv6 vor.

Leider ist aber auch im Bereich der Mobilkommunikation der anfangs starke Boom für den schnellstmöglichen Einsatz neuer Technologien zurückgegangen. So gehen regelmäßig Meldungen von Netzbetreibern ein, in denen der Aufschub des UMTS-Starts mitgeteilt wird. Hierdurch gewinnen die Entwickler und Visionäre von Anwendungen die IPv6 und Mobile-IPv6 im Bereich des Mobilfunks einsetzen wollen Zeit ihre Produkte zu entwickeln und zu veröffentlichen. Dies führte dazu, dass zum jetzigen Zeitpunkt noch keine Mobilfunklösungen angeboten werden, die das neue Internet-Protokoll nutzen.

Neben den Mobilfunkbereichen, werden auch die klassischen Netzwerkbereiche, beispielsweise durch den Einsatz von Wireless LAN oder den sogenannten Handelsreisenden, immer mobiler. Um den Einsatz von Mobile IPv6 in diesen Bereichen zu ermöglichen müssen die mobilen Hosts Mobile IPv6 unterstützen und Home Agents installiert und konfiguriert werden. Während die verschiedenen Betriebssysteme Mobile IPv6 zumindest für mobile Node Anforderungen unterstützen, sind von den verschiedenen Router Herstellern derzeit nur in vereinzelten Fällen Implementierungen bezüglich Mobile IPv6 vorgenommen worden (vgl. Kapitel 5).

5 Implementationsstand von IPv6

Um ein bestehendes IPv4-Netzwerk zu IPv6 zu migrieren, ist zu untersuchen, ob die vorhandenen Hardware- und Softwareprodukte das neue Protokoll unterstützen und wenn ja, welche Funktionen des neuen Protokolls implementiert sind. Hierzu ist eine Bestandsaufnahme der im Netzwerk eingesetzten Produkte notwendig. Anhand dieser kann untersucht werden, welche Funktionen unterstützt werden. Um die Funktionen und Implementationen der Soft- und Hardware zu analysieren, sind neben dem Studium von Referenzen der Hersteller auch entsprechende Testnetzwerke einzurichten, an denen die Funktion und Arbeitsweise des neuen Protokolls untersucht werden kann. Anhand der dabei gewonnen Kenntnisse kann ein Überblick über die notwendigen Veränderungen im Netzwerk gegeben werden. Diese Änderungen können sich vom einfachen Aktivieren der Funktionen bis zum Update bzw. Austausch von Hardware und Software steigern.

Für die Einführung von IPv6 muss ebenfalls diskutiert werden, an welcher Stelle mit dem Aufbau des IPv6-Netzwerkes begonnen werden kann und welche Möglichkeiten bestehen einen parallelen Betrieb von IPv6 und IPv4 zu unterhalten, ohne eine Gefährdung des Netzwerkbetriebes zu riskieren. Hierzu sind Einsatzmöglichkeiten der in Kapitel 2.7 beschrieben Migrationstechniken zu analysieren.

In den folgenden Abschnitten werden die derzeitigen Implementierungsgrade des neuen Internetprotokolls verschiedener Betriebssystem- und Router-Hersteller dargestellt. Die angegebenen Informationen beziehen sich auf Veröffentlichungen der Hersteller.

5.1 Unterstützung der Betriebssysteme

Windows 2000

Windows 2000 ist in der Standardinstallation nicht für IPv6-Funktionalitäten ausgestattet. Um IPv6 mit einem Windows 2000 Client nutzen zu können, muss zunächst das Service Pack 2 installiert sein. Zu diesem muss das von Microsoft bereitgestellte „IPv6 Technology Preview Tool“ installiert werden. Für diese

Software ist unter [NTIP6] eine Installations- und Konfigurationsanleitung zu finden. Vor der Installation der Software muss jedoch klar sein, dass Microsoft darauf hinweist, dass dieses Produkt nicht freigegeben ist und vor einer Installation in Produktionsumgebungen warnt.

Mit dem „IPv6 Technology Preview Tool“ werden folgende Funktionen von IPv6 unterstützt

- Neighbor Discovery, Autokonfiguration,
- DNSv6
- Router Advertisements
- IPSec
- Kommunikation über IPv4 Netzwerk
- Automatisches Tunneln und 6to4

Um einen Windows 2000 Client für Mobile IPv6 nutzen zu können, muss das auf dem „IPv6 Technology Preview Tool“ [MIPNT] basierende Paket für Mobile IPv6 installiert werden. Dieses Paket unterstützt die folgend dargestellten Leistungsmerkmale von Mobile IPv6:

- Implementation der durch die IETF beschriebenen Mobile Node, Correspondent Node Funktionalitäten
- Movement Detection basierend auf Router Advertisements
- IPSec Unterstützung zur Sicherung gegen Missbrauch von Mobile IPv6 Kontrollnachrichten
- permanentes Speichern von Mobile IPv6 Konfigurationen
- Unterstützung mehrerer Heimat-Adressen, und damit verbundenen mehreren Heimat Agenten in verschiedenen Heimat Netzwerken des Mobile Node
- Automatische Registratur beim Heimat Agenten durch IPv6 Router Advertisements
- Transparent Operationen für IPv6 Transport-Protokolle, wie TCP, UDP und ICMP

Windows XP

Windows XP enthält eine Entwicklerfreigabe für IPv6. IPv6 ist im Betriebssystem integriert und muss lediglich aktiviert werden. Unter [WINIP6] stellt Microsoft eine

ausführliche Referenz bezüglich IPv6 und Windows XP zur Verfügung. Die Leistungsmerkmale der IPv6 Implementation sind die gleichen wie die, die für Windows 2000 zu Verfügung stehen.

Für Mobile IPv6 stehen keine Implementierungen zur Verfügung, die es erlauben einen Windows XP Rechner als Mobile Node oder als Home Agent einzusetzen. Windows XP ist lediglich in der Lage mit Mobile Nodes zu kommunizieren, wobei es in einem Binding-Cache die Care-Of-Adressen der Mobilen Nodes vorhält. (vgl. Kapitel 4.1.3)

Linux

Linux unterstützt das neue Internet-Protokoll seit der Kernel Version 2.2.x und stellt somit alle Funktionen für den Einsatz in einem IPv6-Netzwerk zur Verfügung

Mobile IPv6 ist noch nicht Bestandteil der aktuellen Kernelversionen. Hierfür stehen jedoch Kernelpatches unter [MIPL] bereit, mit denen Mobile IPv6 auf Linuxsystemen implementiert werden kann. Diese Kernelpatches basieren auf den aktuellen Internet Drafts zu Mobile IPv6. Im Gegensatz zu den Betriebssystem Windows 2000 kann ein Linuxrechner mit Mobile IPv6 Implementierung auch als Home Agent eingesetzt werden.

Solaris

Für Sun und die „Solaris Software Organization“ hat die Unterstützung für IPv6-Netzwerke eine Schlüsselpriorität. Mit der Freigabe von Solaris 8 im Februar 2000 bietet SUN volle Unterstützung für das IPv4- und das IPv6-Protokoll. Ausführlich Informationen zu Funktionen von IPv6 und Solaris stellt Sun in [SUNIP6] zur Verfügung.

Eine Implementierung von Mobile IPv6 will SUN in zukünftigen Versionen seines Betriebssystems integrieren.

HP-UX

HP stellt in der HP-UX 11i Version eine IPv6 Implementation bereit. Eine Übersicht über die implementierten IPv6 Funktionen wird in [HPIP6] gegeben.

Für Mobile IPv6 gibt es derzeit noch keine Unterstützung.

IBM – OS/390

Für das Betriebssystem OS/390 von IBM steht ebenfalls eine Implementierung von IPv6 zur bereit.

Hierfür stellt IBM unter [IBMIP6] eine Installations- und Konfigurationsanleitung zur Verfügung.

5.2 Unterstützung der Router

In den folgenden Abschnitten wird der derzeitige Entwicklungsstand der Router von den Herstellern *Cisco*, *Nortel* und *Extreme* gezeigt.

Cisco

Cisco nimmt innerhalb der Internet Engineering Task Force (IETF) eine Führungsposition bei der Definition und Implementierung der IPv6-Architektur ein und ist in der Industrie auch weiterhin richtungsgebend bei der Protokollstandardisierung.

Cisco hat eine Entwicklungslinie in drei Phasen für IPv6 ausgearbeitet (vgl. Tabelle 5.1) [CISIP6].

Nortel

Nortel Networks unterstützt IPv6 seit der Herausgabe von Version 12.0 der BayRS Software im Jahr 1997. Derzeit stellt Nortel eine Dokumentation für die Einführung von IPv6 mit BayRS Version 14.00 Router Software zur Verfügung. [NORIP6]

Produkte, welche im Backbone Bereich eingesetzt werden, haben bei der Implementierung von IPv6 eine höhere Priorität als Produkte im Access Bereich eines Netzwerks. So unterstützt beispielsweise der BayRS Router von Nortel heute schon IPv6 und Geräte wie beispielsweise der Passport 8600 noch nicht.

Phase 1 - Cisco IOS-Software v12.2(1)T Q1 CY2001	Unterstützung für IPv6-Architektur und -Adressen Multiprotokollerweiterungen für Border Gateway Protocol 4 (MP-BGP4) Routing Information Protocol für IPv6 (RIPnG, RFC 2080) ICMPv6 und Neighbor Discovery (ND) Unterstützung für manuelle und automatische Tunnel 6to4-Tunnel Ping Traceroute Telnet Standard-Zugangskontrollisten (Access Control Lists, ACLs)
Phase 2 - Mitte 2001	Cisco Express Forwarding für IPv6 (CEFv6) Protocol Translation IPv6-IPv4 Zusätzliche Routing-Protokolle, wie IPv6-Unterstützung für Intermediate System-to-Intermediate System (i/IS-ISv6) IPv6 über Multiprotocol Label Switching (MPLS) Erweiterte Zugangskontrollliste (Extended Access Control List, EACL) IPv6-MIBs
Phase 3 - nach Mitte 2001	Open Shortest Path First für IPv6 (OSPFv3) Mobilität Multicast Dienstgüte (Quality of Service, QoS) Sicherheit Sprache über IPv6

Tabelle 5.1: Cisco IOS Software und IPv6 - Entwicklung in drei Phasen

Extreme

Extreme Networks hat IPv6 in den i-Serien Summit, Alpine and Black Diamond Layer 3 Switches implementiert. Extreme Networks IPv6 Software ist für Interoperabilitätstests für ausgewählte Kunden verfügbar.

Extreme Networks' IPv6 Software beinhaltet:

- IPv6 forwarding über Ethernet Interfaces
- IPv6 Static routes
- RIPnG
- Neighbor Discovery mit Duplicate Address Detection und Router Discovery

- ICMPv6

5.3 Zusammenfassung

Trotz der Tatsache, dass sich in den letzten zwei Jahren auf dem Weg zu IPv6 sehr viel getan hat, finden sich in den derzeitigen Implementierungen von IPv6 auf den Betriebssystemen und Routern noch viele Lücken die den Einsatz von IPv6 verhindern. Denn solange nicht alle Netzwerkbereiche mit IPv6 ausgestattet werden können und die Hersteller nur Entwicklungsumgebungen freigeben, kann von einem Einsatz im produktiven Umfeld nicht gesprochen werden. Dennoch bieten die derzeitigen Implementierungen durchaus Möglichkeiten Testumgebungen einzurichten um vorhandene Funktionen für spätere Verwendungen zu untersuchen und sich mit dem neuen Protokoll vertraut zu machen.

Gründe für noch fehlende Implementation des neuen Protokolls sind möglicherweise in der Tatsache zu finden, dass Funktionen die mit IPv6 definiert wurden teilweise auf IPv4 portiert wurden sind. Als Beispiel seien hier IPSec und QoS genannt. Des Weiteren konnte der vor einigen Jahren drohende Adresskollaps durch Network Adress Translation (NAT) [RFC2663] und Classless Inter Domain Routing (CIDR) [RFC1519] noch um einige Jahre verzögert werden. Ein weiterer Grund ist sicherlich, dass viele Entwicklungen wie DHCPv6 und Mobile IPv6 noch keine verabschiedeten Standards sind.

Aufgrund der Entwicklung in den letzten Jahren und einer IDC-Studie (International Data Corporation), welche prognostiziert, dass die IPv4-Adressen zwischen 2005 und spätestens 2011 vergeben sein werden [MAN02], kann davon ausgegangen werden, dass die Implementierung der vorhandenen Standards und die Verabschiedung neuer Standards in großen Schritten vorangetrieben wird.

6 Möglichkeiten einer Migration

In diesem Kapitel werden zwei grundlegende Möglichkeiten für die Einführung von IPv6 dargestellt. Die Ausführungen beschränken sich auf die Einführung des Protokolls auf Netzwerkebene. Auf notwendige Anpassungen für IPv6 in höheren Protokollschichten wird im Kapitel 6.3 daher nur kurz eingegangen.

Des Weiteren ist vorwegzunehmen, dass kein Konzept erstellt wird anhand dessen eine Migration durchgeführt werden kann, sondern es werden lediglich mögliche Migrationsszenarien und die dafür notwendigen Voraussetzungen erläutert. Das bedeutet, dass dieses Kapitel als Leitfaden für spätere Untersuchungen und einer eventuellen Migration dienen kann.

6.1 betroffene Komponenten

Die Planung einer Migration beginnt mit der Aufstellung der Komponenten, welche von einer Migration betroffen sind. Hierfür ist eine Aufteilung in die Netzwerkkomponenten und die Netzwerkanwendungen möglich.

Die betroffen Netzwerkkomponenten sind Geräte, welche oberhalb der Schicht 3 des OSI Modells angesiedelt werden. Hubs (Schicht 1) arbeiten transparent für die IP-Schicht (Schicht 3) und sind somit von einer Umstellung auf IPv6 nicht betroffen. Das gleiche gilt für Switches, die in der Regel auf Schicht 2 arbeiten. Eine Ausnahme hierzu bilden die Switches, welche zusätzlich mit Schicht-3-Eigenschaften ausgestattet sind und IP-Routing Funktionen übernehmen können. Diese Geräte werden als Layer3-Switches oder Routing-Switches bezeichnet. Neben den Routing-Switches sind Router (Schicht 3) ebenfalls von einer Umstellung auf IPv6 betroffen. Hierbei ist zu beachten, dass es nicht genügt die Router mit einem IPv6 Stack auszustatten, sondern auch die für das Routing verwendeten Protokolle müssen für die neue Internet-Protokoll Version ausgelegt sein.

Mit IPv6-Unterstützung der Betriebssysteme und der angesprochenen Netzwerkkomponenten wäre eine Kommunikation über das neue Protokoll

möglich, eine Migration jedoch noch lange nicht abgeschlossen, denn die Umstellung des Internetprotokolls selbst reicht nicht aus um IPv6 in einen produktiven Umfeld einzusetzen. So müssen auch die Protokolle höherer Schichten bis hin zu den Anwendungsprogrammen an IPv6 angepasst werden. In vielen Fällen erfordert die neue Länge der Adressen eine Änderung bis an die Benutzeroberfläche (vgl. Kapitel 6.3).

Ein weiterer wichtiger Punkt bei der Betrachtung der betroffenen Anwendungen sind Managementsysteme, welche für den funktionierenden Betrieb eines Netzwerks der Größe wie es im VW-Konzern vorherrscht unverzichtbar sind. Ebenso muss für die Migration zu IPv6 ein Domain Name Konzept erstellt werden, was einen IPv6-fähiges Domain Name System voraussetzt.

6.2 Möglichkeiten für eine Migration

Mit Hilfe der in Kapitel 2.7 beschriebenen Migrationstechniken lassen sich eine Reihe von Möglichkeiten für eine Migration des vorhanden Netzwerkes von IPv4 zu IPv6 realisieren. An dieser Stelle werden zwei Richtungen für eine Migration dargestellt. Zum einem kann mit einer Umstellung und Einführung in Teilnetzen begonnen werden und zum anderen wäre es möglich, ein IPv6 Backbone zu schaffen, bevor mit der Einführung von IPv6 in Teilnetzen begonnen wird.

Für die Entscheidung, welche der beiden Richtungen zur Migration zu IPv6 vorgenommen werden, muss geklärt werden, wie dringend die Umstellung auf IPv6 ist und welche Funktionen des neuen Protokolls wann und in welchem Maße eingesetzt werden sollen. So ist der Weg bei der Infrastruktur zu beginnen von Vorteil, wenn die Einführung von IPv6 für mögliche Anwendungen unter keinem zeitlichen Druck steht. Sollen jedoch die Möglichkeiten von IPv6 in nächster Zeit zur Verfügung stehen, sei es um diese zu testen oder in der Produktionsumgebung zu nutzen, so ist der Beginn einer Migration für diese Bereiche von größerer Bedeutung als eine konzernweite IPv6-Infrastruktur.

Neben den Gründen der Nutzung spielen auch die beteiligten Hardware und Software-Ressourcen eine wichtige Rolle bei der Entscheidung. Denn um z.B. eine IPv6-Infrastruktur zu schaffen, ist die Unterstützung der Hardware im Netzwerkbereich von größerer Bedeutung als die Software am Arbeitsplatz. Und

im anderen Fall ist für eine Einführung von IPv6 am Arbeitsplatz nicht nur die IPv6 Fähigkeit des Betriebssystem wichtig, sondern auch die eingesetzten Managementsysteme und Server-Dienste in diesen Bereichen müssen für IPv6 gerüstet sein.

In den folgenden Abschnitten werden die beschriebenen Möglichkeiten für die Migration von IPv4 zu IPv6 erläutert und die Ziele sowie die Voraussetzungen für diese beschrieben.

6.2.1 Beginn am Teilnetz

Das Ziel mit der Migration in Teilbereichen des Netzwerks zu beginnen ist es in diesen Bereichen Funktionen des neuen Protokolls zu nutzen ohne auf eine konzernweite IPv6-Infrastruktur angewiesen zu sein.

Als wichtigste Voraussetzung für dieses Vorhaben muss sichergestellt werden, das eine Kommunikation mit allen anderen Netzwerkbereichen, unabhängig der Protokollversion welche diese einsetzen, gewährleistet ist. Um diese Voraussetzung zu erfüllen muss geklärt werden, welche Möglichkeiten für die Kommunikation mit anderen Teilnetzen zur Verfügung stehen und welche eingesetzt werden sollen.

Im Kapitel 2.7 wurden eine Reihe von Migrationstechniken vorgestellt, mit denen die Voraussetzung einer protokollunabhängigen Kommunikation erfüllt werden kann. Die scheinbar einfachste Möglichkeit ein Teilnetz mit IPv6 auszustatten und eine Kommunikation mit IPv4-Partnern weiter zu gewährleisten wäre die Implementierung von Dual-Stack-Architekturen (vgl. Kapitel 2.7.1) auf den Hosts des IPv6-Netzes. Wenn es sich bei dem IPv6-Teilnetz lediglich um ein Subnetz handelt, wäre nicht einmal eine IPv6-Unterstützung der Router in diesem Netz notwendig. Besteht das Teilnetz jedoch aus mehreren Subnetzen, so wäre es ebenfalls möglich die beteiligten Router ohne IPv6 zu betreiben, was jedoch nicht sinnvoll wäre, da die IPv6-Subnetze untereinander nicht mit IPv6 kommunizieren könnten.

Um IPv6 Routerfunktionalitäten zu erhalten wäre es möglich, die Router mit einer Dual Stack Architektur auszurüsten. Hierbei ist jedoch zu beachten, dass der

Einsatz der Dual Stack Architektur einen höheren Speicherbedarf erfordert und die Rechenlast des Routers erheblich ansteigen wird, wodurch die IPv4 Kommunikation, welche als Voraussetzung gilt, gefährdet wäre. Aus diesem Grund sollte der Einsatz zusätzlicher Router, die lediglich mit IPv6 ausgestattet sind, bedacht werden. Das würde bedeuten, das Routing der unterschiedlichen Protokollversionen geschieht auf voneinander unabhängigen Routern. Bei einem Einsatz von somit parallel laufenden Routern entstehen natürlich Kosten, die bei der Planung der Migration von IPv4 zu IPv6 nicht außer acht gelassen werden dürfen.

Bei einer solchen Implementierung wäre ein Einsatz von Tunneln nicht notwendig, da immer eine Ende zu Ende Kommunikation mit nur einer Protokollversion stattfindet. Das heißt, eine Kommunikation innerhalb des IPv6 Netzes geschieht über IPv6 und eine Kommunikation aus dem IPv6 in ein IPv4 Netz (oder umgekehrt) geschieht über IPv4.

Nach einer erfolgreichen Implementierung von IPv6 in einem Teilnetz liegt der Gedanke nahe weitere Teilnetze mit IPv6 auszurüsten. Bei diesen weiteren Teilnetzen gelten die selben Voraussetzungen und Möglichkeiten wie bei dem ersten Teilnetz. Wenn nun mehrere Teilnetze mit der neuen Protokollversion ausgestattet sind, ist es sinnvoll die auch als IPv6-Inseln bezeichneten Netze miteinander zu verbinden, so das diese Inseln IPv6 für eine Kommunikation untereinander nutzen können. Hierfür besteht die Möglichkeit, auf dem Netz zwischen diesen Inseln IPv6 zu implementieren. Was jedoch im Falle großer Entfernungen bzw. einer großen Anzahl von zu überquerenden Netzwerken nicht sinnvoll scheint.

Um IPv6-Inseln über IPv4-Netze miteinander zu verbinden, wurden Tunnelmechanismen (vgl. Kapitel 2.7.3) entworfen. Hierbei bestehen die Möglichkeiten des Einsatzes von manuell zu konfigurierenden Tunneln oder der Einsatz von automatischen Tunneln. Ob automatische oder manuell zu konfigurierende Tunnel oder sogar beide Varianten eingesetzt werden wird von der Anzahl der untereinander verbundenen IPv6-Inseln anhängig sein. So lassen sich beispielsweise Tunnelendpunkte zwischen zwei Inseln noch manuell Konfigurieren, was jedoch mit einer steigenden Anzahl von IPv6-Inseln und einer

damit steigenden Anzahl von Tunneln aufgrund eines dabei entstehenden Konfigurationsaufwandes nicht mehr sinnvoll sein wird. In diesem Falle können automatische Tunnel unter Verwendung der in Kapitel 2.7.2 für den Übergang von IPv4 zu IPv6 beschrieben Sonderadressen eingerichtet werden.

Bei einem Einsatz von Tunneln wären mehrere Szenarien möglich, welche sich in den Tunnelendpunkten unterscheiden. So ist zu überlegen, ob das Tunneln und somit das Kapseln und Entkapseln von Paketen ausschließlich von den Routern, welche den Übergang der Netze mit verschiedenen Protokollversionen bilden, oder direkt an der Quelle bzw. am Ziel des Datenpaketes also an den Hosts selber vorgenommen werden soll.

Bei der Entscheidung der zu benutzenden Varianten ist zu berücksichtigen, wie groß die Netze sind aus denen bzw. zu denen getunnelt werden soll. So ist eine Variante bei der ein Host als Beginn oder Ende eines Tunnels eingesetzt wird nur sinnvoll, wenn nur wenige Hosts in einem Netz auf Tunnel angewiesen sind, und die Anzahl dieser sich in nächster Zukunft nicht erhöht. Sollen größere IPv6 Netze über Tunnel miteinander verbunden werden, so ist über einen Einsatz von Routern, welche die Aufgaben des Tunnelns übernehmen, nachzudenken. Der Vorteil des Einsatzes von Routern zur Einrichtung von Tunnelendpunkten ist im Konfigurationsaufwand zu finden, welcher somit nur auf den Routern und nicht auf allen Hosts vorgenommen werden muss. Ein weiterer Vorteil wäre, dass die Datenpakete bis zum Router IPv6-Routing-Protokolle nutzen und somit der IPv4-Verkehr innerhalb des Netzes abnimmt, wodurch eine Reduzierung der IPv4-Infrastruktur innerhalb des IPv6-Netzes zurück gebaut werden kann, was wiederum den Verwaltungs- und Konfigurationsaufwand innerhalb dieses Netzes reduziert.

Neben den genannten Vorteilen gibt es auch Nachteile, welche in den Ressourcen der verwendeten Router zu finden sind. Denn ein Router, der mit dem Packen und Entpacken von zu tunnelnden Paketen beschäftigt ist, verbraucht für diese Vorgänge Speicher- und Rechenressourcen, wodurch der Router an seine Grenzen bezüglich Speicher- und CPU-Bedarf kommen kann. Hierdurch entsteht ein zusätzlicher Kostenfaktor der gegen die einzusparenden Kosten beim Konfigurationsaufwand gerechnet werden muss.

Neben den bisher beschriebenen Verfahren, welche sich auf den Einsatz von Dual-Stack Implementierungen beschränken, ist mit Blick auf die Zukunft sicherlich nicht abwegig, dass zu den bestehenden Netzwerken neue hinzukommen. Für diese neue Netze ist zu überlegen, auf IPv4 zu verzichten und gleich mit einer IPv6-Implementierung zu beginnen. Für die dabei entstehenden reinen IPv6 Netze bieten die in Kapitel 2.7 beschrieben Protokollübersetzer die Möglichkeit einer Kommunikation mit reinen IPv4 Netzwerken.

Die Vorteile neue Netze ohne IPv4 zu betreiben sind wegfallende Planungen und Konfigurationen eines IPv4-Netzes und eine Kostenersparnis bei der Bereitstellung von IPv4 und IPv6 Hardware. Diesem Vorteil entgegen stehen umfangreiche Softwarekonfigurationen und Hardwareressourcen, welche bei einer Protokollübersetzung von Paketen aller Hosts eines Netzwerkes entsprechend anwachsen können. Ein weiterer Nachteil gegenüber Dual-Stack-Architekturen ist die Notwendigkeit der Unterstützung für IPv6 von allen Netzwerk-Anwendungen, was bei IPv4/IPv6-Implementierungen nicht zwingend erforderlich ist. Probleme die Internetanwendungen, welche für IPv4 geschrieben wurden, mit IPv6 haben und welche Änderungen an diesen Anwendungen vorgenommen werden müssen, sind im Kapitel 6.3 beschrieben.

Der Einsatz von Protokollübersetzern ist nicht die einzige Möglichkeit für reine IPv6- bzw. IPv4 -Clients Kontakt mit IPv4- bzw. IPv6-Servern aufzunehmen. So können beispielsweise die ohnehin schon eingesetzten Applikation-Gateways wie beispielsweise der HTTP-Proxy so modifiziert werden, dass diese Anfragen von IPv6- bzw. IPv4-Clients entgegen und an IPv4- bzw. IPv6-Server weiterreichen können. Hierdurch wird für den Client eine Anfrage ungeachtet dessen, welche Internet-Protokollversion die Gegenstelle benutzt, beantwortet.

6.2.2 Beginn am Backbone

Das Ziel mit der Einführung von IPv6 am Backbone zu beginnen ist die Schaffung einer konzernweiten IPv6-Infrastruktur, um darauf folgenden Implementierungen in Netzwerken einen direkten Anschluss an ein IPv6-Netz zu ermöglichen.

Die wichtigste Anforderung die an die Migration des Backbones von IPv4 zu IPv6 erfüllt sein muss ist die Gewährleistung, dass der laufende IPv4-Betrieb nicht

gefährdet wird. Das heißt, der bestehende Netzwerkverkehr darf zu keiner Zeit gefährdet werden und eine IPv4-Infrastruktur zu noch vorhandenen IPv4-Netzwerken muss gesichert sein.

Für die Planung des IPv6 Backbones wird es von Nutzen sein das IPv6-Netz auf das IPv4-Netz abzubilden. Hierdurch kann der Aufwand für die Planung des neuen Netzes verringert werden und Standorte des alten Netzes können bestehen bleiben.

Als Voraussetzungen für den Beginn der Einführung eines IPv6-Backbones gelten Anforderungen an die eingesetzte Hardware und genutzte Protokolle. Das bedeutet, dass IPv6 von der Hardware unterstützt werden muss und die eingesetzten Routing-Protokolle mit Erweiterungen für IPv6 zur Verfügung stehen müssen (vgl. Kapitel 2.6). Eine weitere Voraussetzung ist die Planung einer Adressierungsstruktur für die angebundenen bzw. anzubindenden Netzwerke.

Bei der Planung einer Migration von IPv4 zu IPv6 muss berücksichtigt werden, ob die bestehende Infrastruktur genutzt werden kann bzw. soll oder ob zu diesem ein paralleles IPv6-Netz aufgebaut wird, welches nur an einzelnen Punkten im Netzwerk mit dem IPv4-Netz verbunden wird.

Der Vorteil einer parallelen Struktur ist eine höhere Flexibilität bei der Planung des Netzwerks. Ebenfalls lassen sich die einzelnen Schnittpunkte zwischen den zwei Versionen des Internet-Protokolls in Bezug auf Sicherheit und Managebarkeit besser konfigurieren und protokollieren. Neben der Flexibilität ist bei Routern welche nur eine IP-Version verarbeiten müssen ein wesentlich besserer Durchsatz von Daten zu erwarten. Denn eine Dual-Stack Architektur ist speicherintensiver und beansprucht mehr Rechenleistung als eine Router auf dem nur eine IP-Version implementiert ist.

Der Nachteil eines parallelen Netzes sind ohne Zweifel die dafür anfallenden Kosten.

Eine Möglichkeit, um einen Kompromiss zwischen anfallenden Kosten und hoher Flexibilität zu arrangieren, wäre beispielsweise auf den Einsatz von verschiedenen Routern für die IP-Versionen zu bestehen und im Gegenzug dazu auf parallele Leitungen zu verzichten.

6.3 IPv6 in höheren Protokollschichten

Wie in Kapitel 6.1 bereits erwähnt sind neben den Netzwerkkomponenten auch Netzwerk-Anwendungen von einer Internet-Protokoll-Umstellung betroffen.

Die meisten Internet-Anwendungen benutzen – zumindest in einer Zwischenschicht – die Socket-Schnittstelle. Um diese Anwendungen für IPv6 vorzubereiten, wäre eine Neuübersetzung dieser Anwendungen mit einer Socket-Schnittstelle für IPv6 [RFC2553] notwendig.

Eine Neuübersetzung allein mit einem neuen Socket wird aber in der Regel nicht ausreichen, um eine Anwendung IPv6 fähig zu machen. Denn falls in den internen Datenstrukturen eines Programms die Größe einer IPv4-Adresse fest eingebaut ist, besitzen die Anwendungen in der Regel kaum genügend Buffer-Platz, um eine IPv6-Adresse einzulesen und auszuwerten. Um das Problem zu beheben, müsste der Programmierer im Idealfall nur eine Konstante in einer Header-Datei ändern, wobei er aber auch in diesem Fall erst einmal den Quell-Code finden und umschreiben, danach rekompilieren, testen und zu guter Letzt verteilen muss. Und jedes Programm, das vernünftigerweise prüft, ob die eintreffenden Daten dem Format x.x.x.x entsprechen, muss einen neuen Parser bekommen, der auf IPv6 ausgerichtet ist. Der gesamte Programmierungsaufwand kann durchaus komplex sein und hängt ab von der betroffenen Programmstruktur und den Stärken der Entwicklerabteilung.

Aufgrund dieser bei der Einführung von IPv6 entstehenden Probleme, müssen die Anwendungen in einem Netzwerk, welche davon betroffen sind, rechtzeitig ausfindig gemacht werden. Sind die betroffenen Anwendungen gefunden, müssen diese vor dem endgültigen Abschalten von IPv4 oder der Einführung eines reinen IPv6-Netzes umgeschrieben werden. Da zu erwarten ist, dass die bestehende IPv4-Struktur noch für Jahre, auch parallel zu IPv6 bestehen bleiben wird, sollten Entwickler und Programmierer von Internet-Anwendungen genügend Zeit haben, um die Programme um zuschreiben und neu zu übersetzen.

6.4 Einführung von Mobile IPv6

Nachdem in den vorangegangenen Abschnitten zwei mögliche Wege einer Migration und die bei einer Einführung von IPv6 entstehenden Notwendigkeiten von Änderungen für Internet-Anwendungen beschrieben wurden, wird in diesem Abschnitt erläutert, wie die beschriebenen Möglichkeiten bei einer Ersteinführung von IPv6 für Mobile IPv6 genutzt werden können und welche Voraussetzungen bereits beschriebene Möglichkeiten verhindern.

Als Grundvoraussetzung für Mobile-IPv6 müssen alle bei einer mobilen Kommunikation beteiligten Instanzen – Mobile Node, Correspondent Node und Home Agent (vgl. Kapitel 4) – die Internet-Protokollversion 6 unterstützen. Des Weiteren müssen Mobile Node-Funktionalitäten auf dem Mobile Node und Home Agent-Funktionalitäten auf dem Home Agent bereitgestellt werden.

Als weitere Voraussetzung für eine Mobile Kommunikation muss eine automatische Konfiguration des Mobile Node in den Netzen gewährleistet werden, in denen er sich aufhalten wird. Da eine manuelle Konfiguration des Mobile Node bei einem Standortwechsel in ein neues Subnetz nicht sinnvoll ist, muss beispielsweise ein Router, welcher die Autokonfiguration (vgl. Kapitel 2.4) unterstützt oder ein DHCPv6-Server in allen in Frage kommenden Subnetzen bereitgestellt werden.

Falls der Einsatz von Mobile-IPv6 vor einer kompletten Netzwerkumstellung auf IPv6 erfolgen wird, müssen entsprechende Vorkehrungen getroffen werden, die eine Kommunikation der beteiligten Instanzen über das IPv4-Protokoll ermöglichen. Von den in Kapitel 2.7 beschrieben Möglichkeiten ist der Einsatz von Protokollübersetzern nicht möglich. Die Gründe dafür sind, dass der Protokollübersetzer auf Transportebene (TRT) eine Übersetzung oberhalb der Internet-Schicht vornimmt. Da Mobile-IPv6 Bestandteil des Internet-Protokolls ist wird dies bei einer Übersetzung nicht beachtet und ist somit im IPv4-Datagram nicht mehr vorhanden. Das Problem bei einer Protokollübersetzung auf der Internet-Schicht durch SIIT oder NAT-PT liegt darin, dass nicht alle Felder, Optionen bzw. Erweiterungsheader und ICMP-Nachrichten übersetzt werden können. Da Mobile-IPv6 aber auf bestimmte ICMP-Nachrichten und

Erweiterungsheader angewiesen ist (vgl. Kapitel 4.1.4), kann auch dieses Verfahren nicht genutzt werden.

Nachdem die Möglichkeiten einer Protokollübersetzung nicht genutzt werden können, bleibt der Einsatz von Tunnel, um eine Kommunikation über IPv4-Netze zu ermöglichen. Hierfür gelten die selben Voraussetzungen und Möglichkeiten, welche in Kapitel 6.2.1 beschrieben wurden.

7 IPv6-Testnetzwerk

Vor einer Einführung von IPv6 im Unternehmen ist es notwendig, Tests durchzuführen, in denen die Interoperabilität zwischen IPv6 Implementierungen auf verschieden Systemen untersucht werden kann. In diesem Kapitel wird der Aufbau eines IPv6-Testnetzes, wie es in Abbildung 7.1 dargestellt ist, beschrieben und eine Untersuchung der damit verbundenen Anforderungen an die angeschlossenen Komponenten vorgenommen.

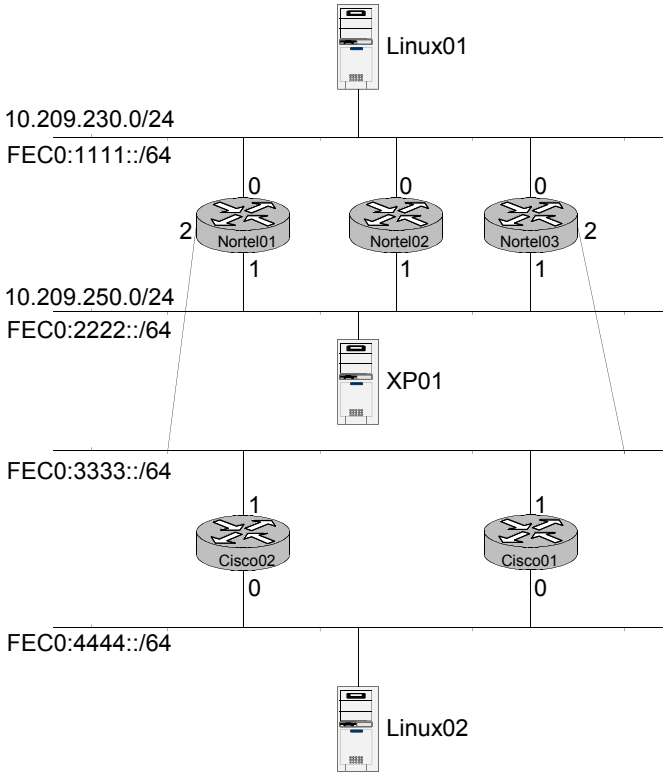

Abbildung 7.1: schemenhafte Darstellung des IPv6-Testnetzes

Zum Aufbau des Testnetzwerkes wurden drei Border-Link-Node-Router (BLN) von Nortel Networks, zwei Cisco-Router der Serie 4700 sowie drei PCs verwendet. Um die verwendeten Komponenten mit IPv6-Funktionalitäten auszustatten, wurden auf den Cisco-Routern das IOS 12.2 mit IPv6-Unterstützung und auf den Nortel-Routern das BayRS 15.0 installiert. Als Betriebssysteme für die PCs kommen auf

zwei Rechnern ein SuSE-Linux 8.0 mit Standardkernel 2.4.18 und auf einem Rechner Windows XP zum Einsatz.

Durch eine Konfiguration der Router wurden vier IPv6-Subnetze eingerichtet an denen die PCs angeschlossen werden (vgl. Tabelle 7.1).

IPv6-Subnetze	***Router / Interface***	***angeschlossene PCs***
FEC0::1111::/64	Nortel01 / 0	Linux01
	Nortel02 / 0	
	Nortel03 / 0	Linux02 (Home)
FEC0::2222::/64	Nortel01 / 1	XP01
	Nortel02 / 1	
	Nortel03 / 1	
FEC0::3333::/64	Nortel01 / 2	
	Nortel03 / 2	
	Cisco01 / 1	
	Cisco02 / 1	
FEC0::4444::/64	Cisco01 / 0	Linux02 (Mobile)
	Cisco02 / 0	

Tabelle 7.1: IPv6-Subnetze im Testnetz und die angeschlossenen Komponenten

Die Struktur des Testnetzes resultiert aus den Funktionen und Operationen, welche untersucht werden sollen. So soll das Netz einer Untersuchung der in Kapitel 2.4 beschrieben Möglichkeiten der Autokonfiguration von Netzwerkknoten durch Neighbor-Discovery Nachrichten und einer Kommunikationen über mehrere IPv6-Subnetze dienen. Dabei wird untersucht, ob eine Kompatibilität zwischen den verwendeten Komponenten besteht.

7.1 Router Konfiguration

In diesem Abschnitt wird beschrieben, welche Konfigurationen auf den eingesetzten Routern vorgenommen wurden. Die wesentlichen Anforderungen an die Router bestehen in Routing Funktionen und Funktionalitäten, welche den angeschlossenen Hosts eine Autokonfiguration ermöglichen sollen.

Das Routing im Testnetz wird durch das Routing-Protokoll RIPnG (vgl. Kapitel 2.6.1) vorgenommen, wobei auf eine statische Konfiguration von Routen verzichtet

wird. An dieser Stelle sei erwähnt, dass eine Untersuchung des Routing-Protokolls OSPF für IPv6 aufgrund des Einsatzes dieses Protokolls im VW-Konzern sinnvoller wäre, jedoch wegen einer fehlenden Implementation auf den eingesetzten Routern nicht möglich ist.

Um eine Autokonfiguration der angeschlossenen Hosts zu gewährleisten, musste auf den Routern die Neighbor Discovery Funktion aktiviert werden.

Im Gegensatz zu IPv4 ist eine explizite Vergabe einer IP-Adresse nicht notwendig, da diese automatisch durch das eingetragene Subnetz und der MAC-Adresse gebildet werden kann. So mussten lediglich die Subnetze gemäß Tabelle 7.1 auf den entsprechenden Interfaces eingetragen werden.

Eine Beschreibung der vorgenommen Konfigurationen eines Cisco-Routers ist in Anhang A dargestellt.

7.2 Host Konfiguration

Eine Konfiguration auf den Hosts musste nicht vorgenommen werden, denn durch die Möglichkeit der Autokonfiguration generieren sich die Hosts ihre IP-Adresse selbst und finden die Router in ihrem Subnetz, welche als Gateways in andere Netze dienen. Dies setzt allerdings voraus, dass IPv6 von den Hosts unterstützt wird. Auf den eingesetzten Linux-PCs ist IPv6 bereits installiert und aktiviert. Auch auf dem Windows XP Rechner ist IPv6 installiert, was nur noch aktiviert werden musste. Hierzu genügte es den Befehl *ipv6 install* aufzurufen.

Da ein Arbeiten - auch in einer Testumgebung - mit den 128 Bit langen IPv6-Adressen zu fehleranfällig und umständlich schien, wurde in dem Test-Netzwerk ein DNS-Server eingerichtet. Hierzu wurde auf einem Linuxrechner der Nameserver BIND 9.0 installiert, welcher in der eingesetzten Linux-Distribution vorhanden ist. Hierbei war zu beachten, dass Windows XP keine Möglichkeit bietet, eine IPv6-Adresse als DNS-Resolver einzutragen. Aus diesem Grund war es notwendig den DNS-Server und den Windows XP Client zusätzlich mit einer IPv4-Adresse auszustatten um eine DNS-Anfrage über eine IPv4-Adresse zu ermöglichen. Da sich der XP-Client und der DNS-Server in verschiedenen Subnetzen befinden, musste zusätzlich noch dafür gesorgt werden, dass ein IPv4-

Router die beiden Subnetze miteinander verbindet. Diese Aufgabe übernimmt im Testnetzwerk ein Nortel Router, welche nun ebenfalls mit zusätzlichen IPv4-Adressen ausgestattet wird. Der Betrieb der IPv4- und IPv6-Versionen auf einem Host ist dank der Dual-IP-Architektur, welche von allen Hosts unterstützt wird, unproblematisch. In Tabelle 7.2 sind die eingerichteten IPv4-Subnetze und die zu den Hosts vergebenen IPv4-Adressen dargestellt.

Des Weiteren musste beachtet werden, dass Windows XP den A6-Resource-Record-Type (vgl. Kapitel 2.2.3) nicht kennt. Deshalb müssen die DNS-Einträge neben den A6-Resource-Records auch die alten AAAA-Resource-Record-Typen zur Namensauflösung beinhalten. Ebenso müssen zur Auflösung der IP-Adressen in Namen die Einträge in der Reverse-Datenbank des DNS-Servers in der punktierten Schreibweise eingetragen werden, da Windows XP die in [RFC2673] eingeführte vereinfachte Notation nicht versteht. In Anhang A sind die Konfigurationsdateien des eingerichteten DNS-Servers dargestellt.

IPv4-Subnetz	***IPv4-Adresse***	***Hostname***
10.209.230.0/24	10.209.230.1	Nortel02 / 0 - Gateway
	10.209.230.100	Linux01 – DNS-Server
10.209.250.0/24	10.209.250.1	Nortel02 / 1 – Gateway
	10.209.250.100	XP01

Tabelle 7.2: verwendete IPv4-Adressen im Testnetz

Zusätzlich zum DNS-Server wurde auf dem Linux-PC noch der Apache Webserver in der Version 2.0.4 installiert. Hierfür wurde ein Quell-Paket des Servers von [APACHE] heruntergeladen, entpackt, mit dem Befehl *./configure –enable-rule=INET6* konfiguriert und mit den Befehlen *make* und *make install* installiert. Dieser Webserver ist in der Lage, Anfragen über IPv6 entgegen zu nehmen und zu beantworten. Als Web-Browser wird der Microsoft Internet Explorer, welcher mit Windows XP installiert wird, genutzt. Mit Hilfe dieses Browsers können IPv6-Anfragen an den Web-Server gestellt werden.

7.3 Beobachtungen und Protokollmitschnitte

Um zu untersuchen, ob die im IPv6-Testnetz beteiligten Komponenten miteinander kooperieren, wurde mit dem Netzwerkanalyzer Ethereal der Netzwerkverkehr

während Konfigurations- und Kommunikationsabläufen beobachtet. Hierfür wurde ein zusätzlicher Computer an die Subnetze, in denen der Verkehr beobachtet werden sollte, angeschlossen und das Programm Ethereal gestartet. Als Voraussetzungen für diese Untersuchungen müssen die in den vorangegangen Abschnitten beschriebenen Konfigurationen auf den Routern vorgenommen und IPv6 auf dem Windows XP Rechner aktiviert werden.

7.3.1 Autokonfiguration

In diesem Abschnitt wird beschrieben, wie die in Kapitel 2.4 beschriebene Autokonfiguration von Netzwerkknoten durch Neighbor-Discovery Nachrichten abläuft und Protokollmitschnitte von dabei gesendeten Nachrichten dargestellt.

Die IPv6-Adresse wird aus dem Netzwerkpräfix, welches von den angeschlossenen Routern mitgeteilt werden muss, und der Interface-ID, welche sich jeder Knoten aus seiner MAC-Adresse bilden kann, von allen Knoten im Netzwerk selbst generiert (vgl. Kapitel 2.2).

Um Pakete, welche mit dem Protokoll-Analyzer mitgeschnitten werden, den Quell- und Zielknoten zuordnen zu können, sind in Tabelle 7.3 die im Testnetzwerk vorhandenen Knoten-Interfaces mit der zugehörigen MAC-Adresse und den daraus gebildeten Interface-IDs aufgelistet. Die Namen der Knoten resultiert aus einer Durchnummerierung der Systeme und der Interfaces dieser Systeme (SystemnameSystemnummer_Systeminterface).

Für die automatische Adressvergabe und der Entdeckung von Default Routern werden Neighbor Discovery Nachrichten (vgl. Kapitel 2.4.1) zwischen den beteiligten Knoten versendet.

Der zu untersuchende Test wurde auf den Rechnern XP01 und Linux02 durchgeführt, wodurch sowohl ein Ablauf für die beiden Betriebssysteme als auch für die beiden Router-Systeme generiert wurde. Da sich die Ergebnisse von den verschiedenen Systemen nicht unterschieden, wird auf eine Darstellung der Protokollmitschnitte für den Rechner XP01 verzichtet.

Host-Name	*MAC-Adresse*	*Interface-ID*
XP01	00:02:B3:4B:76:DB	0202:B3FF:FE4B:76DB
Linux01	00:A0:C9:4D:04:EC	02A0:C9FF:FE4D:04EC
Linux02	00:90:27:A3:DA:18	0290:27FF:FEA3:DA18
nortel01_0	00:00:A2:CB:69:D0	0200:A2FF:FECB:69D0
nortel01_1	00:00:A2:CB:69:D1	0200:A2FF:FECB:69D1
nortel01_2	00:00:A2:CB:69:D2	0200:A2FF:FECB:69D2
nortel02_0	00:00:A2:CB:FA:B4	0200:A2FF:FECB:FAB4
nortel02_1	00:00:A2:CB:FA:B5	0200:A2FF:FECB:FAB5
nortel03_0	00:00:A2:F7:2B:A9	0200:A2FF:FEF7:2BA9
nortel03_1	00:00:A2:F7:34:28	0200:A2FF:FEF7:3428
nortel03_2	00:00:A2:F7:34:29	0200:A2FF:FEF7:3429
cisco01_0	00:E0:1E:A7:8B:18	02E0:1EFF:FEA7:8B18
cisco01_1	00:E0:1E:A7:8B:19	02E0:1EFF:FEA7:8B19
cisco02_0	00:E0:1E:BB:91:80	02E0:1EFF:FEBB:9180
cisco02_1	00:E0:1E:BB:91:81	02E0:1EFF:FEBB:9181

Tabelle 7.3 Auflistung der im Testnetzwerk eingesetzten Knoten mit zugehöriger MAC-Adresse und Interface-Identifier

Um den Rechner Linux02 dazu zu bewegen, seine IP-Adresse neu zu bilden und Netzwerkinformationen anzufordern, wurde der Befehl *rcnetwork restart* ausgeführt. Infolgedessen konnten die folgenden Nachrichten, welche von Linux01 und den Routern Cisco01_0 und Cisco02_0 versendet wurden, beobachtet werden.

- Neighbor Solicitation zur Duplicate Address Detection: (vgl. Abbildung 7.2)

```
Ethernet II
    Destination: 33:33:ff:a3:da:18 (33:33:ff:a3:da:18)
    Source: 00:90:27:a3:da:18 (Intel_a3:da:18)
    Type: IPv6 (0x86dd)
Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 24
    Next header: ICMPv6 (0x3a)
    Hop limit: 255
    Source address: ::
    Destination address: ff02::1:ffa3:da18
Internet Control Message Protocol v6
    Type: 135 (Neighbor solicitation)
    Code: 0
    Checksum: 0x9e1f (correct)
    Target: fe80::290:27ff:fea3:da18
```

Abbildung 7.2: Protokollmitschnitt einer Neighbor Solicitation Nachricht

Diese Nachricht wird von einem Host mit der Absenderadresse :: an dessen Solicited-Node-Multicast-Adresse hier *ff02::1:ffa3:da18* (vgl. Kapitel 2.4.1) gesendet bevor mit der Bildung einer Adresse begonnen wird, um eine Vergabe doppelter Adressen im Netzwerk zu vermeiden. Wenn diese Nachricht nicht beantwortet wird, ist keine gleiche Adresse im Netzwerk vorhanden.

- Router Solicitation zur Anfrage von Routerinformationen: (vgl. Abbildung 7.3)

```
Ethernet II
Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 16
    Next header: ICMPv6 (0x3a)
    Hop limit: 255
    Source address: fe80::290:27ff:fea3:da18
    Destination address: ff02::2
Internet Control Message Protocol v6
    Type: 133 (Router solicitation)
    Code: 0
    Checksum: 0x7696 (correct)
    ICMPv6 options
        Type: 1 (Source link-layer address)
        Length: 8 bytes (1)
        Link-layer address: 00:90:27:a3:da:18
```

Abbildung 7.3: Protokollmitschnitt einer Router Solicitation Nachricht

Diese Nachricht sendet ein Host an die Multicast-Adresse *FF02::2*, wodurch alle Router in seinem Netzwerk erreicht werden können um von diesen Routern Informationen über das Netzwerk zu erhalten. Dabei verwendet der Host als Absender-Adresse seine Site-Lokale Adresse *fe80::290:27ff:fea3:da18*, welche es ihm ermöglicht, mit Knoten in seinem Netzwerk zu kommunizieren (vgl. Kapitel 2.2). Gleichzeitig wird dem Router die Link-layer-Adresse *00:90:27:a3:da:18* des Absenders mitgeteilt, wodurch der Router direkt, also ohne vorher, wie es bei IPv4 nötig war, eine ARP-Anfrage zu senden, antworten kann.

- Router Advertisement zur Mitteilung von Routerwerten (vgl. Abbildung 7.3)

```
Ethernet II
Internet Protocol Version 6
    Version: 6
    Traffic class: 0x70
    Flowlabel: 0x00000
    Payload length: 56
    Next header: ICMPv6 (0x3a)
    Hop limit: 255
    Source address: fe80::2e0:1eff:febb:9180
    Destination address: fe80::290:27ff:fea3:da18
Internet Control Message Protocol v6
    Type: 134 (Router advertisement)
    Code: 0
    Checksum: 0x7f84 (correct)
    Cur hop limit: 64
    Flags: 0x00
        0... .... = Not managed
        .0.. .... = Not other
        ..0. .... = Not Home Agent
        ...0 0... = Router preference: Medium
    Router lifetime: 1800
    Reachable time: 0
    Retrans time: 0
    ICMPv6 options
        Type: 1 (Source link-layer address)
        Length: 8 bytes (1)
        Link-layer address: 00:e0:1e:bb:91:80
    ICMPv6 options
        Type: 3 (Prefix information)
        Length: 32 bytes (4)
        Prefix length: 64
        Flags: 0xc0
            1... .... = Onlink
            .1.. .... = Auto
            ..0. .... = Not router address
            ...0 .... = Not site prefix
        Valid lifetime: 0x00278d00
        Preferred lifetime: 0x00093a80
        Prefix: fec0:4444::
```

Abbildung 7.4: Protokollmitschnitt einer Router Advertisement Nachricht

Die Router im Subnetz von Linux02 antworten auf dessen Router Solicitation Nachricht mit einer Router Advertisement Nachricht, in der sie Routerinformationen mitteilen. Der dargestellte Protokollmitschnitt ist die Antwort von cisco1_0, wobei als Quell- und Zieladressen wieder Link-Lokale Adressen verwendet werden. Die gleiche Nachricht wird auch von cisco0_0 versendet, der die selben Informationen wie cisco1_0 mitteilt. Die Informationen, welche dem Host mitgeteilt werden, beziehen sich auf das Präfix des Subnetzes. Die dafür erforderlichen Informationen sind die Länge des Präfixes (*64*) und das Präfix selbst (*fec0:4444::*). Zusätzlich zu diesen Werten wird auch eine Preferred und eine Valid lifetime mitgeteilt, welche die Gültigkeitsdauer der mitgeteilten Informationen angeben. Ein weiterer Wert ist Router lifetime, welcher die maximal zulässige Gültigkeitsdauer, für die der Router als Default Router verwendet werden darf, angibt. In der Regel werden diese Zeiten durch regelmäßige Router Advertisement Nachrichten der Router aktualisiert. Dieser werden dann aber nicht an einen Host persönlich sondern an die Multicast-Adresse FF02::1 adressiert, wodurch alle angeschlossenen Hosts im Subnetz erreicht werden. Sollten dieser Nachrichten durch eventuelle Störungen oder einer Konfiguration auf dem Router ausbleiben, so müssen die angeschlossenen Hosts einer neue Router-Anfrage stellen.

Um zu kontrollieren, ob die Rechner Linux02 und XP01 ein Adresspräfix vom Router erhalten und daraus eine über das Subnetz hinaus gültige IP-Adresse gebildet haben, kann der Befehl *ifconfig auf* linux02 und *ipv6 if* auf XP01 ausgeführt werden.

Als Ergebnis dieses Tests kann gesagt werden, dass die verwendeten Systeme in der Lage sind eine Autokonfiguration durchzuführen.

Das einziges Problem bei diesem Test ist auf eine Fehlannahme bezüglich der standardmäßigen Aktivierung von Neighbor Discovery auf den Routern, welche zur Verteilung von Router Informationen notwendig ist, zurückzuführen. Denn wo eine Autokonfiguration von linux02 am Netz mit den Cisco-Systemen funktionierte blieb ein anfänglicher Erfolg auf XP01 mit den Nortel Routern aus. Die erste Frage bei der Suche nach der Lösung des Problems war, ob die angeschlossenen

Router von XP01 aus erreichbar sind. Diese Frage konnte mit einem Ping an alle Router im angeschlossenen Subnetz beantwortet werden. Denn als auf einen Ping an die Adresse FF02::2 alle angeschlossen Router antworteten (vgl. Abbildung 7.5) musste das Problem woanders liegen.

```
ping6 ff02::2 -t

Pinging ff02::2 wird angepingt
von fe80::202:b3ff:fe4b:76db%4 mit 32 Bytes Daten:
Antwort von fe80::200:a2ff:fecb:fab5%4: Bytes=32 Zeit=1ms
Antwort von fe80::200:a2ff:fecb:fab5%4: Bytes=32 Zeit=1ms
Antwort von fe80::200:a2ff:fecb:69d1%4: Bytes=32 Zeit=1ms
Antwort von fe80::200:a2ff:fef7:3428%4: Bytes=32 Zeit=1ms
Antwort von fe80::200:a2ff:fecb:69d1%4: Bytes=32 Zeit=1ms
Ping-Statistik für ff02::2
    Pakete: Gesendet = 5, Empfangen = 5, Verloren = 0 (0% Verlust)
Ungefähre Zeitangaben in Millisekunden:
    Minimum = 0ms, Maximum = 1ms, Mittelwert = 0ms
STRG-C
^C
C:\>
```

Abbildung 7.5: Protokollmitschnitt einer Antwort auf einen Ping an alle Router im Subnetz

Nach einer genaueren Beobachtung der Kommunikation zwischen den Routern und XP01 war schnell zu sehen, dass die Antwort auf eine Router Solicitation Nachricht von XP01 ausblieb. Damit wurde klar, dass der Router keine Router Advertisement Nachrichten versendet, was auf eine nicht vorgenommene Konfiguration bezüglich dieser Funktion zurückzuführen war. Nachdem die Neighbor Advertisement Funktion auf den Nortel Routern aktiviert wurde, bekam auch XP01 eine Antwort und konnte seine Adresse bilden.

7.3.2 Mobile IPv6

Nachdem das Testnetzwerk so installiert und konfiguriert war, dass Kommunikationsabläufe wie www-Anfragen, DNS-Anfragen und Telnet- bzw. SSH-Verbindungen mit Erfolg durchgeführt werden konnten, sollte eine Mobile IPv6-Installation im Testnetzwerk durchgeführt werden, um die in Kapitel 4 beschriebene Kommunikation mit einem Mobile Node zu untersuchen. Hierfür wurden auf den Rechnern Linux01 und Linux02 die Mobile IPv6 Software mipv6-0.9.4-v2.4.18 installiert. Da der eingesetzte Linuxkernel noch keine Mobile IPv6 Unterstützung enthält, musste der in der Software vorhandene Kernelpatch eingespielt und eine Neuübersetzung des Kernels vorgenommen werden.

Nachdem der Kernel für Mobile IPv6 vorbereitet war, wurde die Software für Mobile IPv6 installiert und konfiguriert. Eine Beschreibung der somit durchgeführten Schritte bezüglich der Installation und Konfiguration für Mobile IPv6 wird in Anhang A zusammengefasst.

Da Windows XP keine Möglichkeit bietet als Home Agent oder Mobile Node zu fungieren, wird XP01 als Correspondent Node eingesetzt. Hierfür sind keine weiteren Installationen auf XP01 mehr notwendig. Die Aufgabe des Home Agent übernimmt linux01 und als Mobile Node wird linux02 eingesetzt. Da der Home Agent und der Mobile Node ein gemeinsames Heimatnetz haben, ist linux02 laut Abbildung 7.1 bereits mobil. Damit linux02 merkt, dass er in einem fremden Netzwerk ist, muss dessen Heimatadresse fest konfiguriert werden. Hierfür wird der folgende Befehl ausgeführt:

```
ifconfig eth0 add fec0:1111::0290:27FF:FEA3:DA18
```

Nachdem linux02 eine Binding Update Nachricht versendet und der Home Agent diese empfangen und beantwortet hat, sollte linux02 über seine Heimatadresse erreichbar sein. Um dies zu testen, wurde von XP01, der als Correspondent Node fungiert, ein Ping an linux02 gesendet. Dabei war zu beobachten, das der Home Agent die an den Mobile Node adressierte Nachricht annahm und es in ein neues IPv6-Paket einpackte, um es an linux02 weiterzuleiten. In Abbildung 7.6 ist ein Protokollmitschnitt des getunnelten Ping Requests dargestellt. Wie in dieser Abbildung zu erkennen ist, besitzt das Paket zwei IPv6-Header, wobei der äußere Header als Source-Adresse die Adresse des Home Agent und der innere IPv6-Header als Source-Adresse die Adresse des Correspondent Node beinhaltet. Anhand dieses Pakets, bemerkt der Mobile Node, dass der ursprüngliche Sender der Nachricht XP01 ist. Die Antwort des Ping Requests sendet der Mobile Node direkt an XP01, wodurch eine Tunnel über den Home Agent nicht mehr notwendig ist. In dieser Antwort sendet linux02 dem Correspondent Node XP01 auch gleich ein Binding Update Nachricht, welche den XP01 auffordern sollte die Care-Of Adresse von linux02 in seinem Binding Cache zu speichern, um eventuell folgende Nachrichten an linux02 direkt an dessen Care-Of-Adresse zu senden. In Abbildung 7.7 ist zu Veranschaulichung der Protokollmitschnitt dieser Nachricht dargestellt.

```
Ethernet II
Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 80
    Next header: IPv6 (0x29)
    Hop limit: 254
    Source address: fec0:1111::2a0:c9ff:fe4d:4ec
    Destination address: fec0:4444::290:27ff:fea3:da18
Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 40
    Next header: ICMPv6 (0x3a)
    Hop limit: 62
    Source address: fec0:2222::202:b3ff:fe4b:76db
    Destination address: fec0:1111::290:27ff:fea3:da18
Internet Control Message Protocol v6
    Type: 128 (Echo request)
 ...
```

Abbildung 7.6: Protokollmitschnitt eines vom Home Agent an den Mobile Node getunnelten Paketes

Anders als erwartet, nutzt XP01 die Binding Update Aufforderung von linux01 nicht um die Care-Of-Adresse in seinem Binding-Cache aufzunehmen. Dies führt dazu, dass XP01 auch folgende Nachrichten für linux02 an dessen Heimat-Adresse adressiert. Der Grund für dieses unerwartete Verhalten konnte nicht festgestellt werden. Da sich jedoch die Implementierungen von Mobile IPv6 für die verschiedenen Betriebssysteme auf unterschiedliche Versionen des Drafts für Mobile IPv6 beziehen ist nicht auszuschließen, dass das Verhalten auf Inkompatibilitäten der Implementierungen zurückzuführen ist.

Ein weiterer Test sollte zeigen, ob mit Hilfe von Mobile IPv6 eine bestehende TCP-Verbindung zwischen einem Correspondent Node und einem Mobile Node während eines Standortwechsels des Mobile Node bestehen bleibt. Hierzu wurde eine SSH-Session zwischen XP01 und linux02 geöffnet. Anschließend wurde linux02 vom Subnetz 4 getrennt und an Subnetz 2 angeschlossen. Nach ca. 5 Sekunden konnte die SSH-Session fortgesetzt werden, ohne einen neuen Verbindungsaufbau zu starten. Als Ergebnis dieses Tests lässt sich sagen, dass unter der Voraussetzung, der Mobile Node und das fremde Netzwerk unterstützen die Autokonfiguration, eine nahezu konstante Verbindung auch über verschiedene Standorte des Mobile Node hinaus möglich ist.

```
Ethernet II
Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 80
    Next header: IPv6 destination option (0x3c)
    Hop limit: 62
    Source address: fec0:4444::290:27ff:fea3:da18
    Destination address: fec0:2222::202:b3ff:fe4b:76db
Destination Option Header
    Next header: IPv6 destination option (0x3c)
    Length: 2 (24 bytes)
    PadN: 4 bytes
    Option Type: 201 (0xc9) - Home Address
    Option Length : 16
    Home Address : fec0:1111::290:27ff:fea3:da18
Destination Option Header
    Next header: ICMPv6 (0x3a)
    Length: 1 (16 bytes)
    Option Type: 198 (0xc6) - Binding Update
    Option Length : 8
    0... .... = Acknowledge (A)
    .0.. .... = Home Registration (H) : No Home Registration
    ..1. .... = Router (R) : Router
    ...0 .... = Duplicate Address Detection (D)
    .... 0... = MAP Registration (M) : No MAP Registration
    .... .0.. = Bicasting all (B) : Do not request for bicasting
    Prefix Length : 0
    Sequence Number : 3
    Life Time : 1000
    PadN: 4 bytes
Internet Control Message Protocol v6
    Type: 129 (Echo reply)
...
```

Abbildung 7.7: Protokollmitschnitt eines vom Home Agent an den Mobile Node getunnelten Paketes

8 Zusammenfassung

Die Diplomarbeit befasste sich mit den Migrationsverfahren vom Internet-Protokoll Version 4 hin zu Version 6 und den durch die Unterstützung von Mobilität durch IPv6 gegeben Möglichkeiten eines Einsatzes des neuen Internet-Protokolls im IP-Intranet der Volkswagen AG.

Um die Frage des Nutzens einer Einführung von IPv6 für mobile Services beantworten zu können, wurden nach einer ausführlichen Beschreibung der Funktionsweise von Mobile IPv6 und einem Vergleich mit Mobile IPv4 verschiedene Möglichkeiten für den Einsatz von Mobile IPv6 gezeigt. Hierbei wurde deutlich, dass neben der zur Verfügung stehenden Adressanzahl der Vorteil von IPv6 gegenüber IPv4 unter anderen in den Möglichkeiten der Autokonfiguration, in den implementierten Sicherheitsfunktionen und dem optimierten Routing im Mobile IP- Szenario liegt, wodurch eine Einführung von IPv6 für Mobile Service gerechtfertigt und durch den zu erwartenden Verbrauch von IP-Adressen sogar notwendig wäre. Jedoch wurde in einem Überblick über den derzeitigen Entwicklungsstand von Anwendungen und der Unterstützung verschiedener Systemhersteller für Mobile IPv6 gezeigt, dass eine Einführung von Mobile IPv6 zum jetzigen Zeitpunkt aufgrund fehlender Systemunterstützung nicht möglich ist.

Aufgrund von Untersuchungen, welche bezüglich einer allgemeineren Systemunterstützung für IPv6, also unabhängig von Mobile IPv6, vorgenommen wurden, kann gesagt werden, dass hier ebenfalls Mängel bestehen, wodurch eine Einführung im unternehmensweiten IP-Netzwerk ebenfalls nicht sinnvoll ist. Dennoch lassen die derzeitigen Implementierungen einen Einsatz von IPv6 in Testumgebungen zu, in denen der Umgang mit dem neuen Protokoll geübt und zukünftige Implementierungen untersucht werden können. Des Weiteren kann aufgrund der von den Herstellern herausgebrachten Implementierungsstrategien für IPv6 davon ausgegangen werden, dass die Voraussetzungen für den Einsatz von IPv6 in naher Zukunft erfüllt sein werden.

Um die durch die IETF vorgeschlagenen Migrationsmechanismen zu veranschaulichen und einen Einsatz dieser aufzuzeigen, wurden zwei Migrationsszenarien entwickelt, welche als Leitfaden für zukünftige Entwicklungen genutzt werden können. Hierbei wurde gezeigt, dass der Beginn einer Einführung von IPv6 sowohl am einzelnen PC bzw. in einzelnen Subnetzen als auch am IP-Backbone vorgenommen werden kann. Diese unabhängig von einer bestimmten Anforderung an den Einsatz von IPv6 entwickelten Strategievorschläge wurden durch Hinweise für die Einführung von Mobile IPv6 ergänzt. Dabei wurde gezeigt, dass nicht alle vorgeschlagenen Möglichkeiten eingesetzt werden können, um eine Migration von IPv4 zu IPv6 für Mobile IPv6 vorzunehmen.

Im Testlabor wurde ein IPv6-Testnetzwerk eingerichtet, um erste Untersuchungen und Kompatibilitätstest zwischen verschiedenen Systemen vorzunehmen. Anhand der dabei gewonnen Kenntnisse und Erfahrungen lässt sich sagen, dass beispielsweise die Möglichkeit der Autokonfigurationen einen wesentlichen Vorteil gegenüber IPv4 darstellt. Der eingesparten Zeit bei der Konfiguration bzw. „Nicht-Konfiguration“ der Hosts standen Probleme beim Umgang mit den 128-Bit langen IPv6-Adressen, insbesondere bei der Konfiguration des Nameservers in ein für Windows XP verständliches Format, entgegen. Mit aus diesem Grund sind für einen sicheren und effizienten Umgang mit IPv6, insbesondere den Adressen, Programme erwünscht, welche zukünftig Automatismen für beispielsweise die DNS-Konfigurationen zur Verfügung stellen.

Ein Ausblick auf zukünftige Entwicklungen bezieht sich hauptsächlich auf die Fertigstellung von mit der Entwicklung des Internet Protokolls der Version 6 zusammenhängenden Protokollen und Anwendungen. So befinden sich derzeit Protokollerweiterungen wie das Dynamic Host Configuration Protocol für IPv6, die Protokolle für dynamische Updates des Domain Name Systems und die Unterstützung von Mobilität für IPv6 noch in Entwicklungsstadien, wodurch eine Implementierung dieser Protokolle von Systemherstellern nur in wenigen Fällen vorgenommen wird. Des Weiteren ist eine Untersuchung und Entwicklung von Konfigurations- und Managementsystemen, welche für den Einsatz von IPv6 gerade in Unternehmensnetzwerken unverzichtbar sind, vorzunehmen. Ebenso sind derzeit kaum Untersuchungen an Anwendungen vorgenommen worden, welche die Vorteile von IPv6 nutzen. So fehlen beispielsweise Programme, die das

prioritätsgesteuerte Routing, die Datenübertragung in Echtzeit oder die Übertragung von authentifzierten und verschlüsselten Paketen nutzen.

Abschließend lässt sich mit Blick eines Einsatz von IPv6 im IP-Intranet der Volkswagen AG sagen, dass trotz einer zum heutigen Zeitpunkt nicht notwendigen Umstellung die Untersuchung und Beobachtung der Entwicklung und Implementierung von IPv6 und damit verbundenen Protokollerweiterungen und Anwendungen vorgenommen werden sollte. Denn nicht nur aus innovativen Gründen des Unternehmens wird eine Umstellung auf einen lang gesehenen Zeitraum notwendig werden. Aus diesem Grund, sollten nicht nur die Netzwerkkomponenten auf eine Einsatzfähigkeit untersucht werden, sondern ebenso frühzeitig damit begonnen werden, die von einer Umstellung betroffenen und im Volkswagen-Netzwerk eingesetzten Internet-Anwendungen auf ihre Fähigkeit mit dem neuen Protokoll umzugehen zu untersuchen und notwendige Erweiterungen und gegebenenfalls notwendige Umprogrammierungen vorzunehmen. So soll diese Arbeit nicht nur dazu dienen auf das neue Internet-Protokoll aufmerksam zu machen und derzeit mögliche bzw. zukünftig notwendige Entwicklungen und Anforderungen aufzuzeigen sondern ebenso einen Leitfaden für weitere Untersuchungen bezüglich einer Einführung von IPv6 im Unternehmen der Volkswagen AG darstellen.

Anhang A: Installation des Testnetzwerkes

A.1 Cisco – Routerkonfiguration

Die Kommandos, welche auf cisco01 ausgeführt wurden, um auf dem Interface 0 eine IPv6-Adresse zuzuordnen und RIPnG zu aktivieren, sind der Eingabereihenfolge nach in Tabelle A.1 dargestellt:

Kommando	***Beschreibung***
conf t	Wechsel in den Konfigurationsmodus
ipv6 unicast-routing	aktiviert IPv6-Unicast-Routing
interface Fastethernet 0	Wechsel in Konfiguration für Fastethernet-Interface 0
ipv6 address fec0:4444::/64 eui-64	weist dem Interface IPv6-Adresse mit Präfix fec0:4444::/64 und Host-ID nach EUI-64 (MAC->Host-ID) zu
ipv6 enable	aktiviert IPv6 auf dem Interface
ipv6 nd dad attempts 1	aktiviert Duplicate Address Detection für das Interface
exit	verlässt Konfigurationsmodus für Interface
ipv6 router rip TESTIPV6	Konfiguriert einen IPv6-RIP-Routingprozess und Wechsel und Konfigurationsmodus für IPv6 RIP
exit	verlässt Konfigurationsmodus für IPv6 RIP
interface Fastethernet 0	Wechsel in Konfiguration für Fastethernet-Interface 0
ipv6 rip TESTIPV6 enable	aktiviert den angegebenen IPv6-RIP-Routingprozess auf dem Interface

Tabelle A.1: ausgeführte Kommandos für die Konfiguration des Interfaces 0 von cisco01 im IPv6-Testnetzwerk

A.2 Bind 9 Konfiguration

In die zentrale Konfigurationsdatei */etc/named.conf* wurden Einträge zu den relevanten Zonendateien vorgenommen. In Abbildung A.1 ist ein Ausschnitt aus dieser Datei dargestellt.

```
...
zone "ip6.lab" in {
          type master;
          file "ipv6.zone";
};
zone "\[xfec0111100000000/64].ip6.arpa."{
          type master;
          file "ipv6.rev.1111";
};
zone "\[xfec0222200000000/64].ip6.arpa."{
          type master;
          file "ipv6.rev.2222";
};
zone "\[xfec0333300000000/64].ip6.arpa."{
          type master;
          file "ipv6.rev.3333";
};
zone "\[xfec0444400000000/64].ip6.arpa."{
          type master;
          file "ipv6.rev.4444";
};
zone "0.0.0.0.0.0.0.0.1.1.1.1.0.c.e.f.ip6.int."{
          type master;
          file "ipv6.rev.1111";
};
zone "0.0.0.0.0.0.0.0.2.2.2.2.0.c.e.f.ip6.int."{
          type master;
          file "ipv6.rev.2222";
};
zone "0.0.0.0.0.0.0.0.3.3.3.3.0.c.e.f.ip6.int."{
          type master;
          file "ipv6.rev.3333";
};
zone "0.0.0.0.0.0.0.0.4.4.4.4.0.c.e.f.ip6.int."{
          type master;
          file "ipv6.rev.4444";
};
...
```

Abbildung A.1: Ausschnitt aus der Bind 9 Konfigurationsdatei /etc/named.conf

Die Konfigurationsdateien für die Namensauflösung und das *reverse mapping* stehen im Verzeichnis */var/named*.

In Abbildung A.2 ist der Inhalt der Datei: /var/named/ipv6.zone dargestellt.

```
$TTL 1W
@          IN SOA     localhost.   root.localhost. (
                      42         ; serial (d. adams)
                      2D         ; refresh
                      4H         ; retry
                      6W         ; expiry
                      1W )       ; minimum

@          IN NS                 linux01.

linux01    IN A                  10.209.230.100
           IN A6      0          FEC0:1111::02A0:C9FF:FE4D:04EC
           IN AAAA               FEC0:1111::02A0:C9FF:FE4D:04EC
; linux02 Home-Address
linux02    IN A6      0          fec0:1111::290:27ff:fea3:da18
           IN AAAA               fec0:1111::290:27ff:fea3:da18
xp01       IN A                  10.209.250.102
           IN A6      0          fec0:2222::0202:b3ff:FE4b:76db
           IN AAAA               fec0:2222::0202:b3ff:FE4b:76db
cisco1_1   IN AAAA               fec0:3333::02e0:1eff:fea7:8b19
           IN A6      0          fec0:3333::02e0:1eff:fea7:8b19
cisco1_0   IN AAAA               fec0:4444::02e0:1eff:fea7:8b18
           IN A6      0          fec0:4444::02e0:1eff:fea7:8b18
cisco2_1   IN AAAA               fec0:3333::02e0:1eff:febb:9181
           IN A6      0          fec0:3333::02e0:1eff:febb:9181
cisco2_0   IN AAAA               fec0:4444::02e0:1eff:febb:9180
           IN A6      0          fec0:4444::02e0:1eff:febb:9180
nortel01_0 IN AAAA               fec0:1111::0200:a2ff:fecb:69d0
           IN A6      0          fec0:1111::0200:a2ff:fecb:69d0
nortel01_1 IN AAAA               fec0:2222::0200:a2ff:fecb:69d1
           IN A6      0          fec0:2222::0200:a2ff:fecb:69d1
nortel01_2 IN AAAA               fec0:3333::0200:a2ff:fecb:69d2
           IN A6      0          fec0:3333::0200:a2ff:fecb:69d2
nortel02_0 IN A                  10.209.230.1
           IN AAAA               fec0:1111::0200:a2ff:fecb:fab4
           IN A6      0          fec0:1111::0200:a2ff:fecb:fab4
nortel02_1 IN A                  10.209.250.1
           IN AAAA               fec0:2222::0200:a2ff:fecb:fab5
           IN A6      0          fec0:2222::0200:a2ff:fecb:fab5
nortel03_0 IN AAAA               fec0:1111::0200:a2ff:fef7:2ba9
           IN A6      0          fec0:1111::0200:a2ff:fef7:2ba9
nortel03_1 IN AAAA               fec0:2222::0200:a2ff:fef7:3428
           IN A6      0          fec0:2222::0200:a2ff:fef7:3428
nortel03_2 IN AAAA               fec0:3333::0200:a2ff:fef7:3429
           IN A6      0          fec0:3333::0200:a2ff:fef7:3429
```

Abbildung A.2: Inhalt der Bind 9 Konfigurationsdatei /var/named/ipv6.zone

In Abbildung A.3 ist der Inhalt der Datei: /var/named/ipv6.rev.1111 dargestellt.

```
$TTL 1W
@          IN SOA               localhost.   root.localhost. (
                      42        ; serial (d. adams)
                      2D        ; refresh
                      4H        ; retry
                      6W        ; expiry
                      1W )      ; minimum

@          IN NS                linux01.
; Notation laut [RFC2673]
\[x02A0C9FFFE4D04EC/64]  14400 IN PTR    linux01.ip6.lab.
\[x029027fffea3da18/64]  14400 IN PTR    linux02.ip6.lab.
\[x0200a2fffecb69d0/64]  14400 IN PTR    nortel01_0.ip6.lab.
\[x0200a2fffecbfab4/64]  14400 IN PTR    nortel02_0.ip6.lab.
\[x0200a2fffef72ba9/64]  14400 IN PTR    nortel03_0.ip6.lab.
; Notation laut [RFC1886]
c.e.4.0.d.4.e.f.f.f.9.c.0.a.2.0  14400 IN PTR  linux01.ip6.lab.
8.1.a.d.3.a.e.f.f.f.7.2.0.9.2.0  14400 IN PTR  linux02.ip6.lab.
0.d.9.6.b.c.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel01_0.ip6.lab.
4.b.a.f.b.c.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel02_0.ip6.lab.
9.a.b.2.7.f.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel03_0.ip6.lab.
```

Abbildung A.3: Inhalt der Bind 9 Konfigurationsdatei /var/named/ipv6.rev.1111

In Abbildung A.4 ist der Inhalt der Datei: /var/named/ipv6.rev.2222 dargestellt.

```
$TTL 1W
@          IN SOA               localhost.   root.localhost. (
                      42        ; serial (d. adams)
                      2D        ; refresh
                      4H        ; retry
                      6W        ; expiry
                      1W )      ; minimum

@          IN NS                linx01.
; Notation laut [RFC2673]
\[x0202b3fffe4b76db/64]  14400 IN PTR    xp01.ip6.lab.
\[x0200a2fffecb69d1/64]  14400 IN PTR    nortel01_1.ip6.lab.
\[x0200a2fffecbfab5/64]  14400 IN PTR    nortel02_1.ip6.lab.
\[x0200a2fffef73428/64]  14400 IN PTR    nortel03_1.ip6.lab.
; Notation laut [RFC1886]
b.d.6.7.b.4.e.f.f.f.3.b.2.0.2.0  14400 IN PTR  xp01.ip6.lab.
1.d.9.6.b.c.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel01_1.ip6.lab.
5.b.a.f.b.c.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel02_1.ip6.lab.
8.2.4.3.7.f.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel03_1.ip6.lab.
```

Abbildung A.4: Inhalt der Bind 9 Konfigurationsdatei /var/named/ipv6.rev.2222

In Abbildung A.5 ist der Inhalt der Datei: /var/named/ipv6.rev.3333 dargestellt.

```
$TTL 1W
@          IN SOA     localhost.   root.localhost. (
                      42         ; serial (d. adams)
                      2D         ; refresh
                      4H         ; retry
                      6W         ; expiry
                      1W )       ; minimum

@                     IN NS                linux01.
; Notation laut [RFC2673]
\[x02e01efffea78b19/64]  14400 IN PTR     cisco1_1.ip6.lab.
\[x02e01efffebb9181/64]  14400 IN PTR     cisco2_1.ip6.lab.
\[x0200a2fffecb69d2/64]  14400 IN PTR     nortel01_2.ip6.lab.
\[x0200a2fffef73429/64]  14400 IN PTR     nortel03_2.ip6.lab.
; Notation laut [RFC1886]
9.1.b.8.7.a.e.f.f.f.e.1.0.e.2.0  14400 IN PTR  cisco1_1.ip6.lab.
1.8.1.9.b.b.e.f.f.f.e.1.0.e.2.0  14400 IN PTR  cisco2_1.ip6.lab.
2.d.9.6.b.c.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel01_2.ip6.lab.
9.2.4.3.7.f.e.f.f.f.2.a.0.0.2.0  14400 IN PTR  nortel03_2.ip6.lab.
```

Abbildung A.5: Inhalt der Bind 9 Konfigurationsdatei /var/named/ipv6.rev.3333

In Abbildung A.6 ist der Inhalt der Datei: /var/named/ipv6.rev.4444 dargestellt.

```
$TTL 1W
@          IN SOA               localhost.   root.localhost. (
                      42         ; serial (d. adams)
                      2D         ; refresh
                      4H         ; retry
                      6W         ; expiry
                      1W )       ; minimum

@          IN NS                linux01.
; Notation laut [RFC2673]
\[x02e01efffea78b18/64]  14400 IN PTR     cisco1_0.ip6.lab.
\[x02e01efffebb9180/64]  14400 IN PTR     cisco2_0.ip6.lab.
; Notation laut [RFC1886]
8.1.b.8.7.a.e.f.f.f.e.1.0.e.2.0  14400 IN PTR  cisco1_0.
0.8.1.9.b.b.e.f.f.f.e.1.0.e.2.0  14400 IN PTR  cisco2_0.
```

Abbildung A.6: Inhalt der Bind 9 Konfigurationsdatei /var/named/ipv6.rev.4444

A.3 Mobile IPv6 Konfiguration

Wie in Kapitel 7 bereits beschrieben wird im IPv6-Testnetzwerk die Linux-Distribution SuSE 8.0 verwendet. Dabei sind die für einen IPv6-Einsatz notwendigen Erweiterungen bereits enthalten. Für Mobile IPv6 müssen jedoch Modifikationen am Kernel vorgenommen werden. Hierfür werden die in der Distribution enthaltenen Kernel-Sourcen verwendet. Diesem Kernel der Version

2.4.18 muss ein Patch hinzugefügt werden, um Mobility Support für IPv6 zu ermöglichen. Hierfür wurde der unter [MIPL] angebotene Patch den genutzt.

```
patch -p1 < /usr/src/linux/mipv6-0.8.1-v2.4.2/mipv6-0.8.1-v2.4.2.patch
```

Nachdem der Patch eingespielt wurde, stehen in der Konfiguration des Kernels weitere Optionen zur Verfügung, welche Mobile IPv6 betreffen. Um Mobility Support für IPv6 zu ermöglichen müssen wenigstens die Optionen

```
CONFIG_IPV6_IPV6_TUNNEL=m
CONFIG_IPV6_MOBILITY=m
```

aktiviert werden. In der Testumgebung wurde eine weitere Option aktiviert, welche Ausgaben von Debuginformationen über Mobile IPv6 ermöglicht.

```
CONFIG_IPV6_MOBILITY_DEBUG=m
```

Nachdem diese Optionen aktiviert wurden, muss der Kernel neu übersetzt und installiert werden.

```
make dep clean bzImage modules modules_install bzlilo
```

Nach einem Neustart des Systems mit dem neuen Kernel muss die mipv6_dev Gerätedatei angelegt werden.

```
mknod /dev/mipv6_dev c 0xf9 0
```

Neben dem Kernelpatch finden sich in der Software von [MIPL] auch Konfigurations- und Startdateien um Mobile-IPv6 zu konfigurieren und zu starten. Nach der Installation dieser Software müssen die Konfigurationsdateien, welche sich unter /etc/sysconfig/network-mipv6.conf angepasst werden. In Abbildung A.7 ist ein Ausschnitt aus der Konfigurationsdatei des Home Agent und in Abbildung A.8 aus der Konfigurationsdatei des Mobile Node dargestellt, welche die vorgenommenen Einstellungen zeigen. Nachdem diese Konfigurationen vorgenommen wurden, muss die Funktion von Mobile IPv6, sowohl auf dem Home Agent als auch auf dem Mobile Node, gestartet werden.

```
/etc/rc.d/mobile-ipv6 start
```

```
# MIPL Mobile IPv6 Configuration file

# Should this node act as a home agent (ha), mobile node (mn) or
# correspondent node (cn). HA and MN both have CN functionality
# embedded. Default: cn.
FUNCTIONALITY=ha
# In error situations it may be desired to get more detailed
# information what is happening. Increase this value to get more
# messages from the module (default: 0).
DEBUGLEVEL=4
# Should unicasts to node's site-local address be tunneled when mobile
# node is not in its home network (default: yes).
TUNNEL_SITELOCAL=yes
# Minimum number of free tunnel devices kept in cache on MN or HA
# Must be set to at least 1 for MN and HA. To ensure successful
# bindings even during high work loads it could be set to a bigger
# value on the HA.
MIN_TUNNEL_NR=1
# Maximum number of free tunnel devices kept in cache on MN or HA
# Must be set to at least 1 for MN and HA. To improve performance
# set it higher than MIN_TUNNEL_NR
MAX_TUNNEL_NR=2
# Path for a file containing list of IPv6 addresses (and other
# information) of mobile nodes for which this node is allowed to act
# as a home agent.
MOBILENODEFILE=/etc/mipv6_acl.conf
```

Abbildung A.7: Ausschnitt aus der Konfigurationsdatei /etc/sysconfig/network-mip6.conf des Home Agent

```
# MIPL Mobile IPv6 Configuration file

# Should this node act as a home agent (ha), mobile node (mn) or
# correspondent node (cn).
FUNCTIONALITY=mn

# In error situations it may be desired to get more detailed
# information what is happening. Increase this value to get more
# messages from the module (default: 0).
DEBUGLEVEL=4

# Home address for mobile node with prefix length.
HOMEADDRESS=fec0:1111::0290:27ff:fea3:da18/64

# Home agent's address for mobile node with prefix length.
HOMEAGENT=fec0:1111::02a0:c9ff:fe4d:04ec/64

# Mobile nodes are allowed to send Router Solicitation messages at a
# shorter minimum interval than normal IPv6 nodes (default: 4
# seconds). This value controls the mobile nodes minimum Router
# Solicitation interval in seconds (default: 1).
RTR_SOLICITATION_INTERVAL=1

# After sending MAX_RTR_SOLICITATIONS (defined by IPv6, default: 3)
# mobile node reduces Router Solicitation send rate with binary
# exponential back-off mechanism until the maximum send time limit is
# reached. This option sets the maximum send time in seconds.
RTR_SOLICITATION_MAX_SENDTIME=5
```

Abbildung A.8: Ausschnitt aus der Konfigurationsdatei /etc/sysconfig/network-mip6.conf des Mobile Node

Tabellenverzeichnis

Abbildungsverzeichnis

Referenzen

Bücher

[HEI98] TCP/IP – Internet-Protokolle im professionellen Einsatz
Mathias Hein
mitp Verlag 1998

[HUI00]: IPv6 - die neue Generation
Christian Huitema
Addison Wesley Verlag, 2000

[LIP01]: Manfred Lipp
VPN - Virtuelle Private Netzwerke . Aufbau und Sicherheit
Addison-Wesley-Verlag, 2001

[TAN00]: Andrew S. Tanenbaum
Computernetzwerke
Pearson Studium-Verlag, 2000

[WER00]: J. D. Wegner, Robert Rockell
IP: Addressing & Subnetting
mitp-Verlag, 2000

[WIE02]: Herbert Wiese
Das neue Internetprotokoll IPv6
Hanser Verlag, 2002

Zeitschriften

[GRZ01]: Likas Grunwald, Alexandros Zachos
Selbst gestrickt
Heise Verlag, IX 5/2001

[MAN02]: Klaus Manhart
IP-Adressen für jedes Handy
WEKA Verlag, Funkschau 06/2002

[SIE00]: Richard Sietmann
Mobilfunk hilft IPv6
WEKA Verlag, Funkschau 4/2000

[ORD01]: Andreas Ordemann
Mobile machen mobil NET Verlag
NET 1-2/2001

Studien- und Diplomarbeiten

[MIK00]: Thomas Miklitz
Internet Protokoll Version 6
Technische Universität Darmstadt, 2000

[RIE96]: Gernot Riegert
Erstellung einer Managementschnittstelle für DHCP-Server
Technische Universität München, 1996

[SCH00]: Stefan Schlott
IPsec-Implementierung in IPv6 für Linux
Universität Ulm, 2000

Internet

[BRA01]: Computernetz - Internetworking
http://www.iam.unibe.ch/~rvs/lectures/cn/cn_6.pdf
Torsten Braun, 2001

[CISIP6]: Cisco IOS IPv6
http://www.cisco.com/ipv6
Cisco Systems, 2001

[EUI64]: Guidellines For 64-Bit Global Identifier (EUI-64™) Registration Authority
http://standards.ieee.org/regauth/oui/tutorials/EUI64.html
Institute of Electrical and Electronics Engineers, 1997

[HIN95]: IP Next Generation Overview
http://playground.sun.com/pub/ipng/html/INET-IPng-Paper.html
Robert M. Hinden, 1995

[HPIP6]: HP-UX 11i IPv6
http://www.software.hp.com/cgi-bin/swdepot_parser.cgi/cgi/displayProductInfo.pl?productNumber=T1306AA
hewlett-packard company

[HÜB96]: Das Internet-Protokoll (Teil A)
http://www.tu-chemnitz.de/iuk/probeold/kap1a-all/kap1a.htm
Uwe Hübner, 1996

[IBMIP6]: Introduction to IPv6 for OS/390
http://www-3.ibm.com/software/network/commserver/library/publications/ipv6.html
IBM

[JMA02]: IPv6 Transition Test Challenges
http://advanced.comms.agilent.com/RouterTester/member/appnotes/pdf/ipv6-tran.pdf
jmarx, 2002

[MIC99]: Routing in the Internet
http://www.uni-koblenz.de/~steigner/labor/seminar-routing/07-04-michels-grundlagen.pdf
Hartmut Michels, 1999

[NORIP6]: Nortel Networks: Corporate Information – Technology & Expertise – IPv6
http://www.nortelnetworks.com/ipv6
Nortrel Networks, 2001

[PSS01]: Multilateral sichere Mobilitätsunterstützung für IP-Netze: Paketfilter- und Tunnelkonfiguration
http://www-tkn.ee.tu-berlin.de/~schaefer/Publications/pik2001.pdf
Frank Pählke, Günter Schäfer, Jochen Schiller, 2001

[RAU97]: Transition Mechanisms and Packet Tunneling
http://www.uni-koblenz.de/~raue/ipng/main.htm
Heiko Raue, 1997

[SEAMO]: SEAMOBY WG
http://www.seamoby.org/seamoby_wg.htm
IETF Seamoby Working Group, 2001

[SUNIP6]: Internet Protocol Version 6 (IPv6) Networking For The Solaris 8 Operating Environment
http://wwws.sun.com/software/solaris/ds/ds-ipv6networking/ds-ipv6n.pdf
SUN, 2001

[WINIP6]: Microsoft Windows - IPv6
http://www.microsoft.com/windows.netserver/technologies/ipv6/default.asp
Microsoft Corporation, 2002

[WIT02]: Mobile IP
http://www.zib.de/schintke/lehre/mobile-computing/ausarbeitung-mobile-ip-witte.pdf
Jochen Witte, 2002

IETF - Drafts

[Draft01] draft-ietf-ipngwg-icmp-name-lookups-09.txt,
IPv6 Node Information Queries

[Draft02] draf-ietf-dhc-dhcpv6-26.txt
Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

[Draft03] draft-ietf-mobileip-ipv6-18.txt
Mobility Support in IPv6

IETF - Request for Comments

[RFC768] User Datatgramm Protocol

[RFC791] Internet Protocol // IP Version 4

[RFC792] Internet Control Message Protocol // ICMP Version 4

[RFC793] Transmission Control Protocol

[RFC826] Ethernet Address Resolution Protocol

[RFC1256] ICMP Router Discovery Messages

[RFC1519] Classless Inter-Domain Routing

[RFC1584] Multicast Extension to OSPF

[RFC1700] Assigned Numbers

[RFC1886] DNS Extensions to support IP version 6

[RFC2080] RIPnG for IPv6

[RFC2031] IETF-ISOC relationship

[RFC2328] OSPF Version 2

[RFC2362] Protocol Independent Multicast-Sparse Mode (PIM-SM)

[RFC2373] IP Version 6 Addressing Architecture

[RFC2374] An IPv6 Aggregatable Global Unicast Address Format

[RFC2375] IPv6 Multicast Address Assignments
[RFC2403] The Use of HMAC-MD5-96 within ESP and AH
[RFC2404] The Use of HMAC-SHA-1-96 within ESP and AH
[RFC2405] The ESP DES-CBC Cipher Algorithm With Explicit IV
[RFC2406] IP Encapsulating Security Payload (ESP)
[RFC2408] Internet Security Association and Key Management Protocol (ISAKMP)
[RFC2409] The Internet Key Exchange (IKE)
[RFC2460] Internet Protocol, Version 6 (IPv6)
[RFC2461] Neighbor Discovery for IP Version 6 (IPv6)
[RFC2462 IPv6 Stateless Address Autoconfiguration
[RFC2463] Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification
[RFC2464] Transmission of IPv6 Packets over Ethernet Networks
[RFC2473] Generic Packet Tunneling in IPv6 Specification
[RFC2545] Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing
[RFC2553] Basic Socket Interface Extensions for IPv6
[RFC2663] IP Network Address Translator (NAT) Terminology and Considerations
[RFC2673] Binary Labels in the Domain Name System
[RFC2675] IPv6 Jumbograms
[RFC2710] Multicast Listener Discovery (MLD) for IPv6
[RFC2740] OSPF for IPv6
[RFC2858] Multiprotocol Extensions for BGP-4
[RFC2874] DNS Extensions to Support IPv6 Address Aggregation and Renumbering
[RFC2893] Transition Mechanisms for IPv6 Hosts and Routers
[RFC2894] Router Renumbering for IPv6
[RFC2977] Mobile IP Authentication, Authorization, and Accounting
[RFC3056] Connection of IPv6 Domains via IPv4 Clouds
[RFC3122] Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification
[RFC3152] Delegation of IP6.ARPA
[RFC3228] IANA Considerations for IPv4 Internet Group Management Protocol

Software

[MIPL] MIPL - Mobile IPv6 for Linux
http://www.mipl.mediapoli.com/

[MIPNT] Microsoft Mobile IPv6 Implementation (MIPv6)
http://research.microsoft.com/programs/europe/projects/MIPv6.asp

[NTIP6] Microsoft IPv6 Technology Preview for Windows 2000
http://msdn.microsoft.com/downloads/sdks/platform/tpipv6.asp

[APACH] The Apache Software Foundation
http://www.apache.org

Glossar

ARP (Address Resolution Protocol)

Protokoll zur Ermittlung der MAC-Adresse (bzw. OSI-Ebene-2 Adresse) zu einer gegebenen IP-Adresse

Datagram

Bezeichnung einer Dateneinheit, die ohne vorherigen Verbindungsaufbau übertragen werden kann.

DES (Data Encryption Standards)

Ein standardisiertes Verfahren zur Verschlüsselung von Nachrichten. Das Verfahren arbeitet mit symmetrischen Schlüsseln mit einer Länge von 56 Bytes

DNS (Domain Name System)

Ein im Internet zur Übersetzung von Objektreferenzen in andere Referenzen verfügbarer Mechanismus. Die primäre Nutzung ist die Übersetzung des Namens eines Hosts in seine numerische IP-Adresse

DHCP (Dynamic Host Configuration Protocol)

Eine Industriestandard-Methode zur vereinfachten und dynamischen Konfiguration von IP-Adressen für Computer in TCP/IP-Netzwerken.

Gateway

Ein Rechner zur Übersetzung unterschiedlicher Protokolle zwischen verschiedenen Netzwerken oder Netzwerkkomponenten, im Sonderfall auch zur Verbindung zweier Teilnetze gleicher Technologie, speziell im Internet werden auch die Rechner Gateways genannt, die nur als Bridge zu einem anderen (Teil-)Netz im Internet dienen

GPRS (General Packet Radio Service)

Technologie für den Internetzugang via Mobilfunk. Dabei können Übertragungsgeschwindigkeiten von bis zu 115 KBit/s erreicht werden.

GSM (Global System for Mobile Communication)

GSM ist ein technischer Standard für die digitale Funktelefonie Mit Hilfe dieser Technik lassen sich auch Daten übertragen. Mittlerweile werden Handys mit integriertem PC samt Internet- und Faxsoftware angeboten. Die Übertragungsgeschwindigkeit ist mit 9.6 KBit/s sehr langsam.

HOP

Wörtlich:"Schaltstelle". Bei der Übertragung eines IP-Paketes von einem Router-Eingang zum Router-Ausgang wird genau ein "Hop" passiert.

IETF (Internet Engineering Task Force)

Eine zur Internet Society gehörende technische Arbeitsgruppe, die sich mit der Standardisierung von Entwicklungsstrukturen im Zusammenhang mit dem Internet befasst.

Internet

Bezeichnung für eine Ansammlung von Netzen (Netzverbund), die den gesamten Globus überspannen.

IPSec

IPSec ist ein Set von Protokollen zur Implementierung von sicheren Verbindungen und zum Schlüsselaustausch im Internet Protokoll.

LAN (Local Area Network)

Lokales Netz, das innerhalb eines Gebäudes oder Firmengeländes installiert wird und in dem private Übertragungsmedien und Vermittlungsanlagen benutzt werden.

Multicasting

Eine Telekommunikationstechnik, durch die ein Informationsfluss von einer Quelle an mehrere potentielle Empfänger verbreitet werden kann.

QOS (Quality of Service)

Ein Konzept in der Vernetzungstechnik, durch das Anwendungen dem zugrunde liegenden Netz ihre spezifischen Anforderungen anzeigen können, bevor mit der Übertragung von Daten begonnen wird.

Router

Bezeichnung für einen Knoten oder Vermittler in einem verbindungslosen paketvermittelten Netz, z.B. einem IP-Netz. Router dienen als Verbindungspunkte zwischen LAN-Segmenten.

RSVP (Resource reSerVation Protocol)

RSVP ermöglicht mehreren Sendern, an mehrere Empfängergruppen zu übertragen, und einzelnen Empfängern, nach Wunsch Kanäle zu wechseln. Außerdem optimiert es die Nutzung der Bandbreite bei gleichzeitiger Vermeidung von Überlastungen. In seiner einfachsten Form nutzt das Protokoll das Multicast-Routing mit Spanning-Trees

TCP (Transmission Control Protocol)

Zuverlässiges verbindungsorientiertes Protokoll, durch das ein Bytestrom von einer Maschine fehlerfrei im Internet einer anderen Maschine zugestellt wird.

Tunneling

Ein Verfahren, bei dem Datenpakete des einen Protokolls mit Hilfe eines anderen Protokolls übertragen werden. So können z.B. IPv6-Pakete in IPv4-Pakete eingepackt und über das Internet transportiert werden.

UDP (User Datagram Protocol)

Unzuverlässiges verbindungsloses Protokoll der Transportschicht für Anwendungen die anstelle der Abfolge oder Flusskontrolle von TCP diese Aufgaben lieber selbst bereitstellen.

UMTS (Universales Mobiles Telekommunikations System)

UMTS ist der neue Hochgeschwindigkeit-Mobilfunk-Standard für Europa und der Nachfolger des derzeitigen Standards GSM. Die neue Technologie ermöglicht mit bis zu 2 MBit/s (im Vergleich zu 9,6 KBit/s von GSM) eine wesentlich schnellere mobile Daten-Kommunikation.