Zusammenfassung Computernetzwerk

Inhalt

[1 Einstieg Netzwerke 2](#_Toc438541693)

[2 Layer 1 – Medien- und Zugriffverfahren 2](#_Toc438541694)

[3 Layer 2 – Switching 2](#_Toc438541695)

[4 Layer 3 – Vermittlungsschicht 3](#_Toc438541696)

[4.1 Adressen 3](#_Toc438541697)

[4.1.1 Spezielle Adressen 3](#_Toc438541698)

[4.1.2 Adressklassen 3](#_Toc438541699)

[4.2 Subnetze 4](#_Toc438541700)

[4.3 IP-Protokoll „Das Internet – Teil 1“ 4](#_Toc438541701)

[4.4 Routing „Das Internet – Teil 2“ 5](#_Toc438541702)

[4.4.1 Statisches Routing 5](#_Toc438541703)

[4.4.2 Dynamisches Routing 6](#_Toc438541704)

[4.5 Hierarchie der Routingstruktur 7](#_Toc438541705)

[5 Namenauflösen, IP-Vergabe und Rundsprüche 8](#_Toc438541706)

[5.1 DNS 8](#_Toc438541707)

[5.1.1 Iterative Auflösung 8](#_Toc438541708)

[5.1.2 Rekursive Auflösung 9](#_Toc438541709)

[5.1.3 DNS Caching 9](#_Toc438541710)

[5.2 DHCP 10](#_Toc438541711)

[5.3 Multicast 10](#_Toc438541712)

[6 Layer 4 – Ports, Sockets, TCP, UDP, Firewall 11](#_Toc438541713)

[7 VLAN-Konzepte 11](#_Toc438541714)

[8 VPN-Konzepte 12](#_Toc438541715)

[8.1 Split Tunnels 12](#_Toc438541716)

[8.2 Verschlüsselung 12](#_Toc438541717)

[8.3 IPSec 13](#_Toc438541718)

[9 WLAN, Funknetze, VoIP 14](#_Toc438541719)

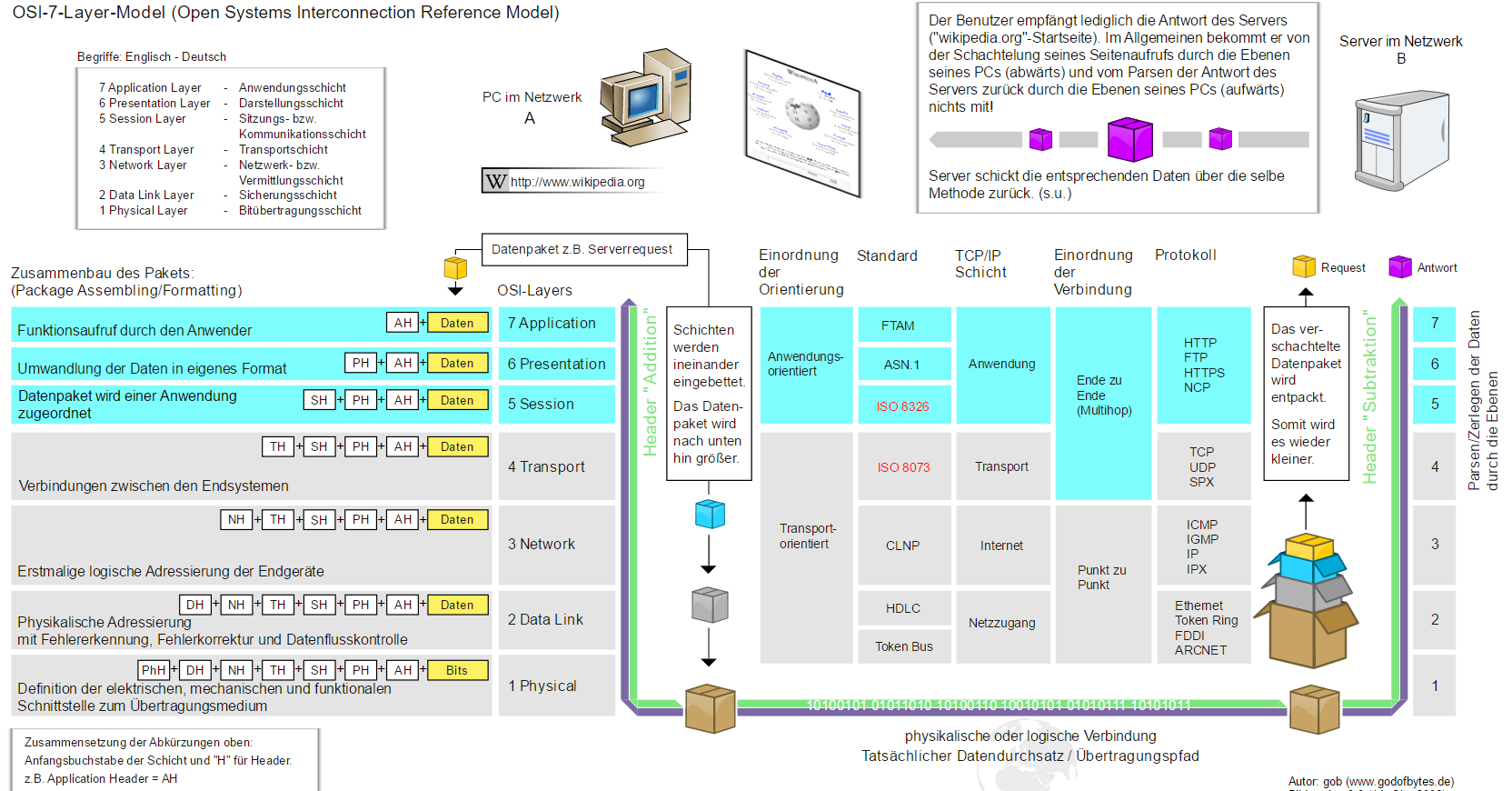
[10 Szenarien von Netzzugängen 14](#_Toc438541720)

[11 IPv6 15](#_Toc438541721)

[11.1 weitere Ipv6 Protokolle 17](#_Toc438541722)

[11.2 Ipv4 to Ipv6 und umgekehrt 18](#_Toc438541723)

# Einstieg Netzwerke



# Layer 1 – Medien- und Zugriffverfahren

## Koaxialkabel

## CU-Kabel

## TP Kabel

## Glasfaser

### LWL

### Single und Multimode

## Zugriffsverfahren

### CSMA/CD

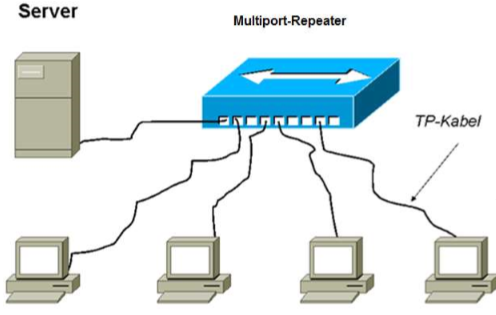
### Token Ring und Token-Passing

# Layer 2 – Sicherungsschicht

Aufgabe der Sicherungsschicht ist es, eine zuverlässige, das heißt weitgehend fehlerfreie Übertragung zu gewährleisten und den Zugriff auf das Übertragungsmedium zu regeln.

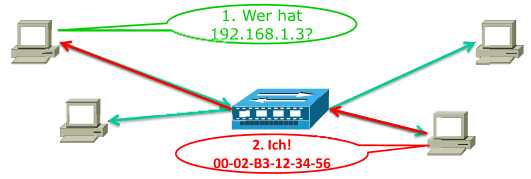
## Repeater

Ein Repeater ist in der Kommunikationstechnik ein elektrischer oder auch optischer Signalverstärker/aufbereiter zur Vergrösserung der Reichweite eines Signals.



## ARP

Das **Address Resolution Protocol** (**ARP**) ist ein Netzwerkprotokoll, das zu einer Netzwerkadresse der Internetschicht die physikalische Adresse (Hardwareadresse) der Netzzugangsschicht ermittelt und diese Zuordnung gegebenenfalls in den so genannten ARP-Tabellen der beteiligten Rechner hinterlegt. Es wird fast ausschließlich im Zusammenhang mit IPv4-Adressierung auf Ethernet-Netzen, also zur Ermittlung von MAC-Adressen zu gegebenen IP-Adressen verwendet, obwohl es nicht darauf beschränkt ist. Für IPv6 wird diese Funktionalität nicht von ARP, sondern durch das Neighbor Discovery Protocol (NDP) bereitgestellt.



Ein Rechner A will einem Rechner B Daten versenden. Für das braucht er die physikalische Adresse (MAC) des Empfängers. Bekannt ist nur die logische Adresse (IP) des Empfängers. Rechner A prüft als erstes ob die Empfänger IP im gleichen Netz ist. Wenn ja, versendet Rechner A ein Paket mit seiner IP- und MAC-Adresse als Broadcast und fragt wer die IP-Adresse 192.168.1.3 hat. Der Rechner mit der angefragten IP-Adresse speichert die Daten des Anfragenden Rechners in seiner ARP-Tabelle und füllt seine MAC-Adresse ins Paket und schickt sie zurück. Nun ist beiden, Sender und Empfänger, alles bekannt, sie können Daten austauschen.

## Hub

Ein Hub verbindet mehrere Stationen in einem Netzwerk miteinander. Somit verbinden Hubs Netzsegmente auf der physikalischen Schicht (Schicht 1), wodurch eine gemeinsame Kollisionsdomäne entsteht. Der Hub hat also nur eine Verteilerfunktion, indem er Datenpakete entgegen nimmt und an alle anderen Ports weiterleitet. Das heißt insbesondere auch, dass ein Hub Datenpakete auch an Stationen weiterschickt, die eigentlich nicht Empfänger der Daten sind.

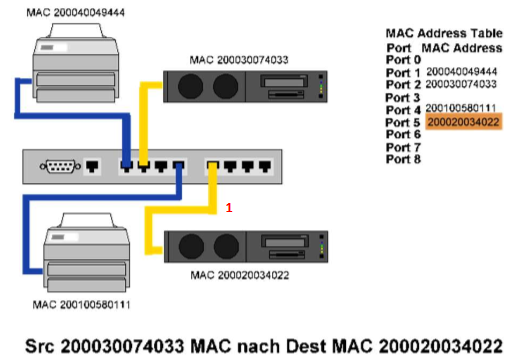
## Bridges

Brücken oder Bridges sind dazu da, 2 Computersegmente im Netzwerk zu verbinden. Brücken unterbrechen hierbei Kollisionsdomänen, indem Datenpakete nur dann in das jeweils andere Segment weitergeleitet werden, wenn sich der jeweilige Empfänger in diesem Segment befindet. Im Gegensatz zu Hubs, werden also Datenpakete nicht einfach an alle Ports weitergeleitet, sondern die Weiterleitungsentscheidungen werden auf Basis der der Ziel-Mac-Adresse getroffen. Im Schichtenmodell, arbeiten Brücken im Gegensatz zu Hubs auf Schicht 2 (Sicherungsschicht).

Switch  
Switches sind, ähnlich wie Brücken, Kopplungselemente die auf der Sicherungsschicht arbeiten und die Kollisionsdomäne unterbrechen. Im Gegensatz zu Brücken haben Switches aber mehr als zwei Ports. Switches und Hubs werden häufig miteinander verwechselt, nicht zuletzt weil die Geräte fast identisch aussehen (können), während ein Hub aber nur ein stupider  Sternverteiler ist, kann ein Switch auch eine Direktverbindung zwischen angeschlossenen Computern schalten.

Braucht ein Switch eine IP-Adresse? Jein: Nur wenn man den Switch über eine Management-Oberfläche konfigurieren möchte.

Switches „lernen“ während der Benutzung Ihr Netzwerk:



### Arten von „Switching“

**Cut Trough**

Beim Cut-Trough Verfahren wird nicht gewartet, bis das eintreffende Paket vollständig geladen ist, sondern nur bis die ersten Bytes, die die Zieladresse beinhalten, geladen sind. Nach dem auswerten der Zieladresse (interne Adresstabelle) wird der Datenstrom bereits auf den entsprechenden Ausgangsport geleitet. Das Datenpaket wird ohne Fehlerkorrektur weitergeitet. Auf diese Art ist die Latenzzeit extrem gering.

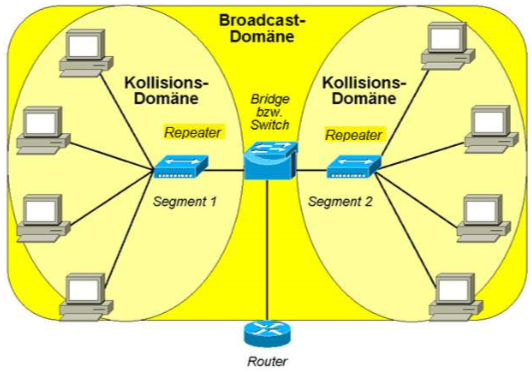
**Store and forward**

Wie der Name dieses Verfahrens bereits sagt, wird das ganze Datenpaket (bzw. „Frame“) als erstes gespeichert. Danach wird die Prüfsumme berechnet und mit dem CRC-Wert des Pakets überprüft. Besteht keine Differenz, wird das Paket der entsprechenden Zieladresse weitergeleitet. Falls eine Differenz und somit ein Fehler aufgetreten ist, wird das Paket verworfen. Store and forward ist somit die sicherste Methode im Switching, um die Datenströme fehlerfrei zu behalten, jedoch auch mit Abstand das langsamste. (Nicht für Streaming möglich, aber dafür für Short Message Services wie Whatsapp etc.)

## Kollisions- und Broadcast-Domänen

Eine Broadcast-Domäne ist ein logischer Verbund von Netzwerkgeräten in einem lokalen Netzwerk, der sich dadurch auszeichnet, dass ein Broadcast alle Domänenteilnehmer erreicht.

Die Kollisionsdomäne ist ein Teilbereich bestehend aus Teilnehmerstationen auf Layer 1 des OSI-Schichten Modells. Sie umfasst alle Netzwerkgeräte, die gemeinsam um den Zugriff auf ein Übertragungsmedium (geteilte Ressource) konkurrieren (Kabel oder Funknetz). Beginnen zwei Teilnehmerstationen gleichzeitig zu senden, kommt es zu Kollisionen. Die Signale (Spannungsimpulse) werden im Übertragungsmedium vermischt/überlagert und die Informationen dadurch zerstört.



## Spanning Tree Protocol (STP)

|  |  |
| --- | --- |
| Der Algorithmus ist in der IEEE 802.1d Norm spezifiziert, der auch die Bridge im Allgemeinen beschreibt. Er wird eingesetzt um bei Verknüpfungen von Netzwerken redundante Pfade (sog. Loops) durch einen deterministischen logischen Pfad im Netz zu ersetzen. Im Beispiel oben sind verschiedene LANs durch Bridges (Pfeile) miteinander verknüpft. Wie leicht zu sehen ist, würden alle Bridge-Links zusammen redundante Pfade im Netz ermöglichen. Dieses hätte sog. endlos kreisende Pakete zur Folge. Mit dem *Spanning Tree -Algorithmus* wird einer der möglichen logischen Pfade im Netz ausgewählt, der dann keine mehr Loops enthät. Das Ergebnis sind hier die blauen (ungebrochenen) Pfeile die einem Baum gleichen. Im Extremfall kann hierdurch auch eine Bridge total aus dem Netzverkehr herausfallen. |  |

# Layer 3 – Vermittlungsschicht

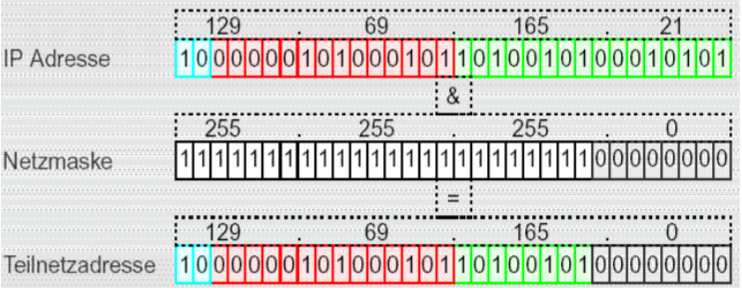
Kommunikation zwischen verschiedenen Netzwerken

## Adressen

In Layer 3 gelten andere Adressen als in Layer 2 (physische Adressen, MAC-Adressen). Durchgesetzt hat sich das IP-Adressierungssystem. IP-Adressen sind weltweit gültig, eindeutig, einheitlich und werden deshalb zentral verwaltet. Die Koordination dabei übernimmt die Organisation IANA – Internet Assigned Numbers Authority.

Eine IPv4 Adresse ist 32 Bit breit (IPv6 128 Bit).





Netzmaske zeigt an, wie viele Bits die Netzwerkkennung definieren. Im obigen Netz werden 24 Bits für die Netzwerkkennung benutzt („24-Bit-Netz“), 8 Bits sind frei verfügbar für Geräteteile. Die Teilnetzadresse ist die Netzerkennung für diese eine Teilnetz (immer am Schluss eine „0“).

### Spezielle Adressen

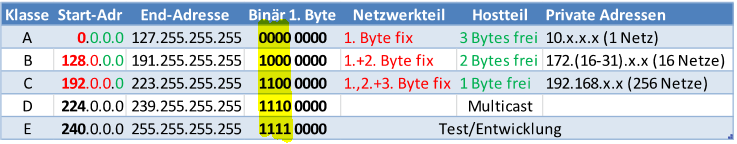
In Subnetzen gibt es auf Layer 3 (IP) fest reservierte Adressen, die keinem Host zugewiesen werden dürfen. Das sind:

|  |  |
| --- | --- |
| **Subnetzadresse** | Die kleinste Verfügbare Adresse ist die Bezeichnung des Subnetzes. |
| **Broadcast-Adresse** | Die grösste Verfügbare Adresse ist für die Rundspruchsendung.  Datenpakte für diese Adresse müssen von jedem Netzwerkgerät entgegen genommen werden. |
| **Auto Private IP Adress (APIPA)** | 169.254.0.1 – 169.254.255.255, kommen dann zum Einsatz wenn kein „DHCP“ Dienst vorhanden ist. |
| **Superprivate Adressen** | 127.x.x.x |

Wichtig: Durch die Segmentierung von Netzen gehen immer Minimum zwei Adressen für Endgeräte verloren. Beachtet man noch den benötigten Router für Netzwerk-Netzwerk Kommunikation, sind es sogar drei Adressen die „verloren“ gehen.

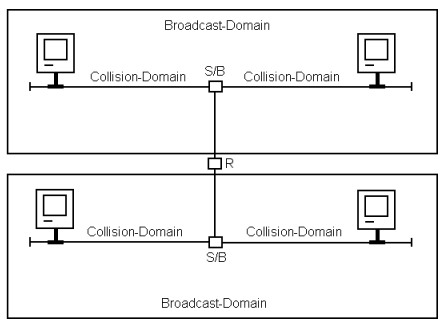
### Adressklassen

Man unterscheidet zwischen fünf Klassen von Adressen. Grosser Unterschied zwischen ihnen ist die Anzahl Hosts, welche adressiert werden können -> Grösse der möglichen Netze.



## Subnetze

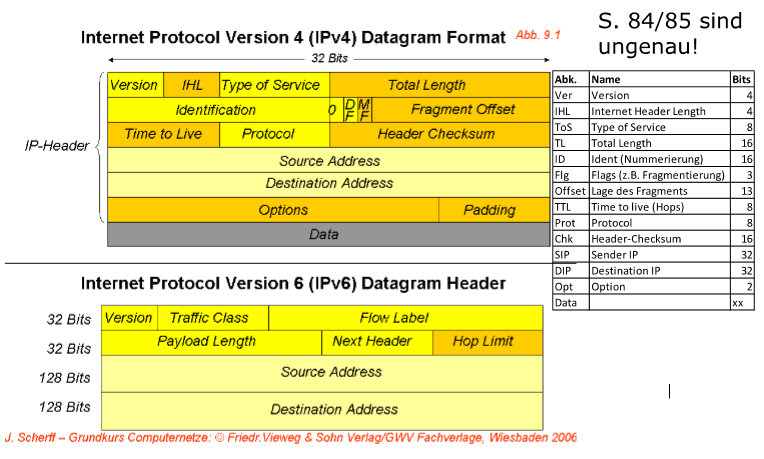
Alle 65536 Hosts einer B-Klasse Adresse auf Layer 1 oder 2 zu kommunizieren zu lassen, macht keinen Sinn. Alleine die Rundsendungen (Broadcasts) würden das Netzwerk zum Erliegen bringen. Deshalb unterteilt man die Netze in so genannte Subnetze.



|  |  |
| --- | --- |
| **1.Layer** | Hubs und Medium Konverter |
| **2.Layer** | Switches und Bridges -> trennen Kollisionsdomänen |
| **3.Layer** | Router -> trennen Broadcast-Domänen |

Die Kommunikation erfolgt innerhalb von Broadcastdomänen auf Layer 2 mithilfe von **ARP** (Address Resolution Protocol) und zwischen ihnen über Router auf Layer 3.

## IP-Protokoll „Das Internet – Teil 1“



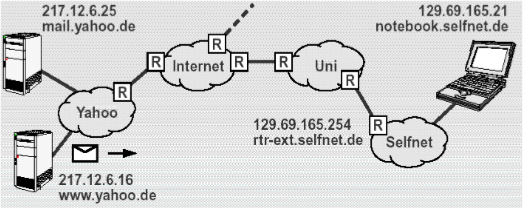
Traffic Class: Was gebe ich durch

Payload Length: Länge der Daten

Time To Live: wie viele "hops" das IP-Paket macht

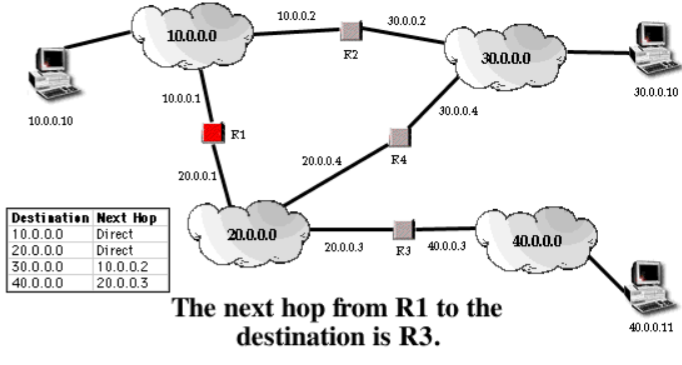
## Routing „Das Internet – Teil 2“

Jedes IP-Paket enthält eine Ziel- und Herkunftsadresse. Falls ein Rechner sich nicht im gleichen Netz befindet, wird das Paket in Richtung des Zielnetzes weitergeschickt. Für das Weiterschicken sind die Geräte an den Netzübergängen zuständig – die Router.



### Statisches Routing

Dieses Verfahren ist *nicht adaptiv*, sehr einfach und kommt daher eher in sehr kleinen Netzen zum Einsatz. Jeder Router („Knoten“) unterhält eine Tabelle mit einer Zeile für jeden möglichen Zielrouter („Zielknoten). Grösster Nachteil ist, dass sich die Routing-Tabellen nicht verändern. Um eine fehlerfreie Datenkommunikation zu ermöglichen, darf sich das Netzwerk nicht verändern.

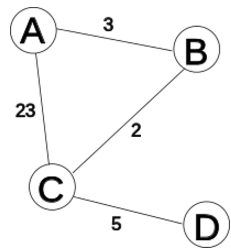


### Dynamisches Routing

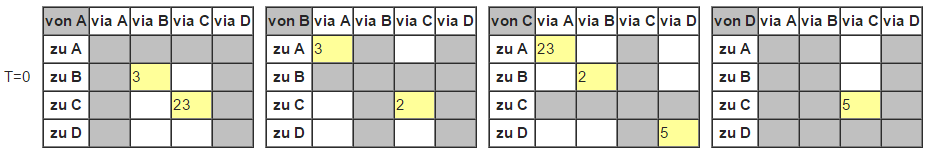
Der grosse Vorteil beim dynamischen Routing ist, das die Routing-Tabellen automatisch aktualisiert werden. Nachfolgend werden zwei bekannte Methoden auf möglichst einfache Art und Weise erklärt.

#### Distance Vector Routing

Situation: Ein Netz mit vier Routern ist geben. Die Zahlen zwischen den Routern beschreiben die „Zeitkosten“ für Datenpakete.



1. Erzeuge eine Kostenmatrix, welche Router über welche Nachbarn und zu welchen Kosten erreichbar sind und anfangs nur die (bekannten) Kosten zu direkten Nachbarn enthält.



1. Erzeuge eine Aufstellung mit Informationen, welche Router wir zu welchen Kosten am besten erreichen können und schicke sie an alle Nachbarn.
2. Warte auf Aufstellungen dieser Art von anderen Routern, rechne diese dann in die eigene Kostenmatrix ein.
3. Ändern sich dadurch die minimalen Kosten, zu denen wir einen Router erreichen können: fahre mit Schritt 2 fort, sonst mit Schritt 3.

Tiefe Konvergenz: Veränderungen werden nur von den erhaltenen Datenpakte übermittelt und jeweils nur für die eine Strecke die sie durchlaufen haben. Beispiel Protokoll ist das RIP

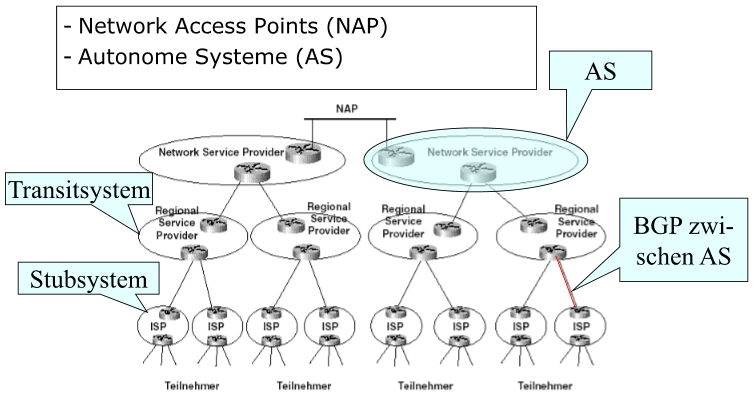
#### Link State Routing

Beim Link-State-Routing werden nur die Informationen über die direkten Nachbarn verschickt, dafür aber gleich an alle Router des Netzwerks. Mit diesen Informationen kann dann jeder Router seine Routing-Tabelle berechnen. Da die Änderungen verbindungsorientiert an die benachbarten Router propagiert werden, besitzen Routing-Protokolle mit dem LSA eine gute Konvergenz.

Wenn es viele Veränderungen in der Routingtabelle gibt und die Routingtabelle oft oder regelmäßig aktualisiert werden muss, empfiehlt es sich, ein Link-State-Routingprotokoll zu verwenden. Dabei werden nur die jeweiligen Änderungen unter den Routern ausgetauscht.

## Hierarchie der Routingstruktur

Alte was sell ech öber das schriebe….

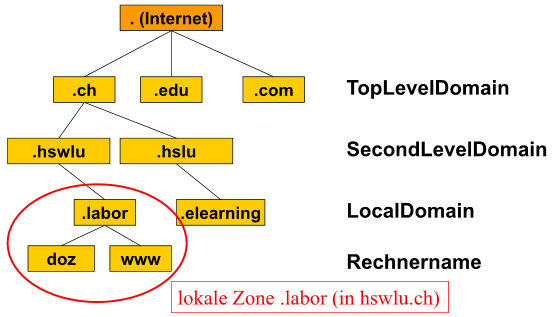
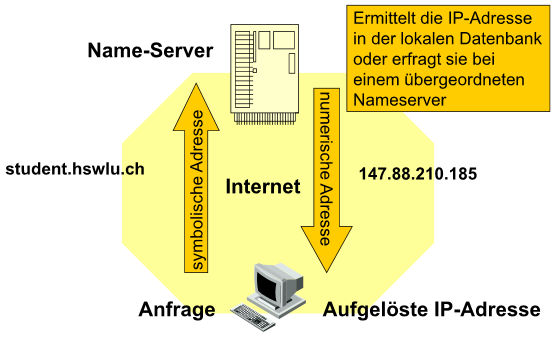


# Namenauflösen, IP-Vergabe und Rundsprüche

|  |  |
| --- | --- |
| Subnetzmaske gibt vor das die ersten drei Bytes gleich sein müssen. Entweder ist die IP-Adresse oder der Default gateway falsch. |  |

## DNS

Das Domain Name System (**DNS**) ist einer der wichtigsten Dienste in vielen [IP](https://de.wikipedia.org/wiki/Internet_Protocol)-basierten [Netzwerken](https://de.wikipedia.org/wiki/Rechnernetz). Seine Hauptaufgabe ist die Beantwortung von Anfragen zur [Namensauflösung](https://de.wikipedia.org/wiki/Namensaufl%C3%B6sung).

[www.labor.hswlu.ch](http://www.labor.hswlu.ch). -> Von links nach rechts, von unten nach oben. Ein bekanntes Programm für die Namensauflösung ist **nslookup** („name server look up“).

Es gibt folgende zwei Namens-Server

|  |  |
| --- | --- |
| **TLD Server (Top-level domain)** | Sind zuständig für com, edu, org, uk, ch und alle anderen Top Level Domains. |
| **Authoritative DNS** | Ist der Namensserver einer Organisation, welche eine Domäne betreibt. Dieser liefert Namensauflösungen zu allen Hosts innerhalb dieser Organisation. (Wird entweder von der Organisation selber betrieben oder vom Internet Service Provider) |

Es werden zwei Auflösungsarten unterschieden:

### Iterative Auflösung

|  |  |
| --- | --- |
| Iterative Query:  Server antwortet mit zuständigem Server.  „Ich kenne diesen Namen selber nicht, aber frag doch den da!“  Bspl. www.gaia.cs.usmass.edu   1. Zum eigenen DNS 2. Dieser kennt den Zuständen TLD-Server für „edu“ 3. „Edu“ TLD kennt „usmass“ 4. „Usmass“ kennt alle Hosts innerhalb von seinem Netz, also auch „gaia.cs“ |  |

### Rekursive Auflösung

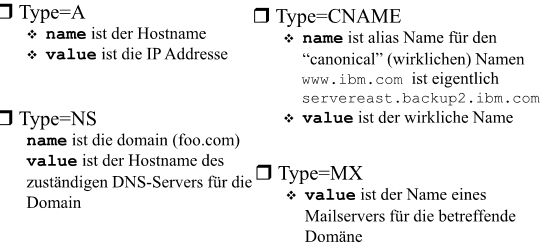
|  |  |
| --- | --- |
| Rekursive Query:  Der angefragte Server trägt die Last, deshalb nur noch intern bei Autoritativen DNS Server verwendet.  Funktioniert „fast“ wie iterative, grosser Unterschied ist das die „Anfragelast“ beim angefragten Server hängen bleibt. Dieser darf nicht nur einfach eine Antwort geben und weiterleiten, sondern muss auf eine Antwort warten und darf sie erst dann weiterleiten. |  |

### DNS Caching

Jeder Namensserver speichert („caching“) seine gelernten Namensauflösungen aus Performance Gründen. So ein Eintrag kann folgendermassen aussehen:



Im Type-Feld wird unterschieden, was für ein Namensauflösung gemacht wird (Host, Domäne, Mailserver, etc.)



## DHCP

DHCP steht für Dynamic Host Configuration Protocol. Es ermöglicht die Zuweisung der Netzwerkkonfiguration an [Clients](https://de.wikipedia.org/wiki/Client) durch einen [Server](https://de.wikipedia.org/wiki/Server_(Software)). Ziel vom DHCP ist eine dynamische Vergabe der IP-Konfiguration.

**Ablauf**

|  |  |
| --- | --- |
| 1. Neuer Client schickt eine **DHCPDISCOVER** Nachricht (mit seiner MAC-Adresse) als Broadcast (255.255.255.255, UDP Port 67) und weil er noch keine IP hat, ist die Absender IP 0.0.0.0 mit UDP Port 68. 2. Nun schicken alle DHCP’s im Netz eine **DHCPOFFER** (entweder als Broadcast oder als Unicast an 0.0.0.0+Mac-Adresse. 3. Client entscheidet nun anhand von selbstdefinierten Regeln welcher DHCPOFFER er annimmt und schickt eine **DHCPREQUEST**. 4. Der DHCP-Server bestätigt in einer *DHCPACK*-Nachricht (DHCP-Acknowledged) die IP-Adresse mit den weiteren relevanten Daten. |  |

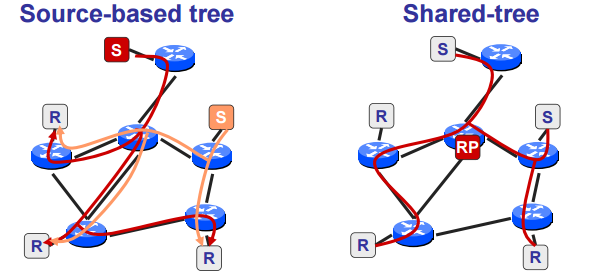
Der Client kann am Schluss mit einer ARP-Request im Netz nachfragen, ob er wirklich der einzige ist mit dieser IP. Falls ein anderes Gerät antwortet, kann die vorgeschlagene Adresse mit einer DHCPDECLINE-Nachricht zurückgewiesen werden.

## Multicast Routing

Das Ziel von Multicast-Routing ist, einen Baum von Verbindungen zu ermitteln, der alle Router mit angeschlossenen Hosts enthält, die Mitglieder der Multicast-Gruppe sind. „Baum“, weil nicht alle Verbindungen zwischen den Routern geflutete werden sollen. Zwei Varianten werden unterschieden:

**Source-based** jeder Sender baut seinen eigenen Baum. Merkmale sind: effiziente Bäume, kleine Verzögerungen und gleichmässige Last Verteilung

**Shared-tree** gleicher Baum für alle Gruppenmitgleder. Merkmale sind: grössere Verzögerungen, schlechtere Lastverteilung (RP-Router trägt hohe Last)

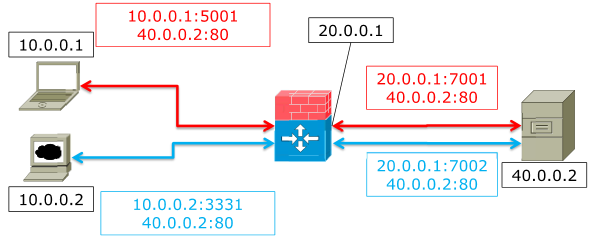


RP: „Rendevouz Point“; Router am RP verteilt die Datenpakete weiter.

## NAT und PAT

**Port and Address Translation** (PAT) oder **Network Address Port Translation** (NAPT) ist eine Technik, die in Computernetzwerken verwendet wird. Sie ist eine spezielle Form von NAT (1 zu n NAT). Dabei werden im Gegensatz zu NAT nicht nur die IP-Adressen, sondern auch Port-Nummern umgeschrieben. **PAT** wird eingesetzt, wenn mehrere private IP-Adressen aus einem LAN zu einer öffentlichen IP-Adresse übersetzt werden sollen.

Gegeben sei folgendes Netzwerk: die beiden Clients (linke Seite vom Router) haben die gleiche IP-Adresse, sobald sie nach aussen (rechts vom Router) kommunizieren



Der „Firewall-Router“ muss nun eine Tabelle pflegen, um die Verbindungen zu den Clients unterscheiden zu können. Zur Unterscheidung werden zu den IP-Nummern noch Portnummern hinzugefügt (Portnummern bis 1024 sind „Well-Known ports“ und sollten nicht genutzt werden).

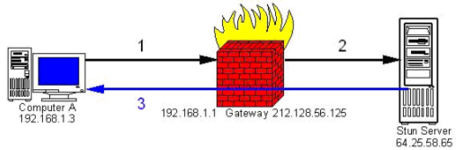


Wieso „Firewall-Router“? Firewalls leiten nur Antworten auf eine Anfrage wieder ins eigene Netz hinein, somit ist eine höhere Sicherheit gewährleistet.

### STUN

Session Traversal Utilities for NAT; Ermöglicht es NAT-Clients (Computer hinter einer Router-Firewall wie im NAT-Beispiel) die Kommunikation mit einem VoIP-Provider ausserhalb des lokalen Netzwerks aufzubauen.

Mit Hilfe des STUN-Servers können Clients ihre öffentliche IP-Adresse, das NAT-Gerät, hinter dem sie sich befinden, und den nach außen veröffentlichten, Internet-seitigen Port ermitteln, dem per NAT ein bestimmter lokaler Port zugewiesen wurde. Diese Informationen werden zur UDP-basierten Kommunikation zwischen dem Client und dem VoIP-Provider verwendet, um einen Anruf aufzubauen.



UDP: User Datagram Protocol. „Verbindungslose Übertragung von Daten über das Internet“ -> Stun macht eine virtuelle Verbindung für UDP (siehe blauer Pfeil).

# Layer 4 – Transportschicht

Die Transportschicht ist die erste Schicht, die direkt mit bestimmten Services kommuniziert. Das IP-Protokoll bietet einen verbindungslosen Transport der Daten an, das heißt ohne jegliche Sicherung.

## Ports und Sockets

|  |  |
| --- | --- |
| **Port** | Wollen zwei Prozesse miteinander kommunizieren, dann identifizieren sich die Prozesse gegenüber TCP/IP mit einer *Port-Nummer*. Das ist eine 16-Bit Zahl, somit gibt 65535 Ports für jedes Transport-Protokoll (UDP und TCP). Die Port-Nummer sagt also aus, an welchen Prozess ein bestimmtes Paket weitergereicht werden möchte. |
| **Sockets** | IP-Adresse + Port Nummer = Socket  Ein Socket ist ein Kommunikationsendpunkt (ein Objekt), durch das Datenpakete sowohl gesendet als auch empfangen werden (bidirektionaler Datenfluss). Man unterscheidet dabei zwei verschiedene Typen von Sockets:   * + **Streamsockets** (TCP Sockets = Transmission Control Portocol)   + **Datagrammsockets** (UDP Sockest = User Datagramm Protocol) |

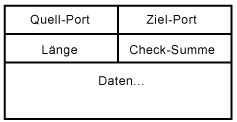
## TCP

Das Transmission Control Protocol („Übertragungssteuerungsprotokoll“) ist ein [Netzwerkprotokoll](https://de.wikipedia.org/wiki/Netzwerkprotokoll), das definiert, auf welche Art und Weise Daten zwischen Computern ausgetauscht werden sollen. Die zentrale Aufgabe von TCP/IP ist dafür Sorgen zu tragen, dass Datenpakete innerhalb eines dezentralen Netzwerks beim Empfänger ankommen. Dafür stellt TCP/IP die folgenden zentralen Funktionen bereit.

* + Logische Adressierung / Logical Addressing (IP)
  + Wegfindung / Routing (IP)
  + Fehlerbehandlung und Flussteuerung / Error Control and Flow Control (TCP)
  + Anwendungsunterstützung / Application Support (TCP)
  + Namensauflösung / Name Resolution (DNS)

## UDP

UDP (User Datagram Protocol) ist ein verbindungsloses Transport-Protokoll. Es hat damit eine vergleichbare Aufgabe, wie das verbindungsorientierte TCP. Allerdings arbeitet es verbindungslos und damit unsicher. Das bedeutet, der Absender weiß nicht, ob seine verschickten Datenpakete angekommen sind. Während TCP Bestätigungen beim Datenempfang sendet, verzichtet UDP darauf. Das hat den Vorteil, dass der Paket-Header viel kleiner ist und die Übertragungsstrecke keine Bestätigungen übertragen muss. Einfacher Aufbau eines UDP-Headers:



In der Regel wird UDP für Anwendungen und Dienste verwendet, die mit Paketverlusten umgehen können oder sich selber um das Verbindungsmanagement kümmern. Typisch sind DNS-Anfragen, VPN-Verbindungen, Audio- und Video-Streaming.

## Firewall

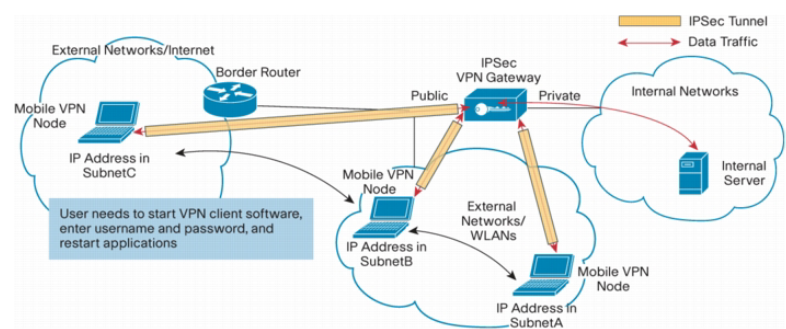
Die Firewall ist eine Art Filter, zwischen dem Computer und dem Internet oder jeder anderen Form von Netzwerk. Sie prüft unter anderem, ob Programme auf das Internet oder aus dem Internet auf den Computer zugreifen wollen und dürfen. Die Firewall schützt vor unberechtigten Zugriffen von innen und von außen.

### Unterschied zwischen Router und Firewall

Eine Firewall ist im Prinzip nichts anderes als ein Router, jedoch mit einer invertierten Weiterleitungsphilosophie. Ein Router ist maximal offen, bestrebt, hat also das Ziel, so viel als möglich Daten weiter zu transportieren. Hier müssen gewünschte Beschränkungen bewusst konfiguriert werden (sogenannte Access-Control-Lists, ACL). Unerwünschter Verkehr muss also bewusst durch Konfiguration verboten werden. Eine Firewall verhält sich genau andersherum. Hier ist jeglicher Verkehr von vornherein verboten. Will man bestimmte Daten weiterleiten, muss dies explizit durch Konfiguration erlaubt werden.

# VLAN-Konzepte

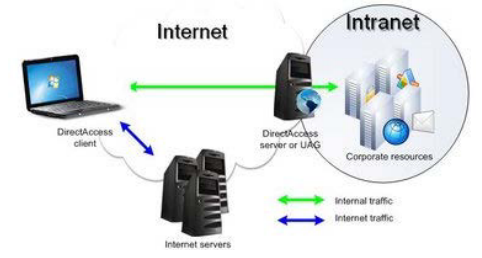
# VPN-Konzepte



VPN Gatewy (nicht Server) routet Subnetze A, B, C der Client.

Bei IP-Änderung bricht VPN-Verbindung zusammen.

## Split Tunnels



Diese geschossenen Tunnel verursachen viel Traffic.

Die Split Tunnels bieten Angriffspunkte für Angreifer aus dem Internet.

Der Client dient dabei als Hop-Node.

Als Gegenmassnahme ist nur eine Verbindung auf dem Client erlaubt.

## Verschlüsselung

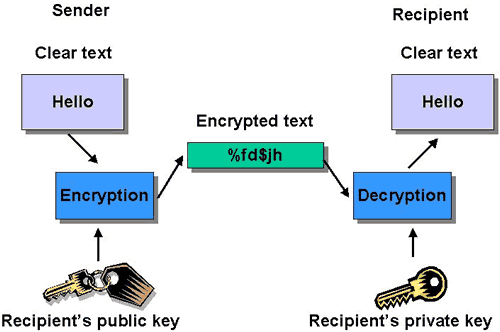
Verschüsselungsverfahren:

* DES
* 3DES
* IDEA
* AES -> aktueller Standard.

Symmetrisch:

Gleiche Schlüssel werden zum Verschlüsseln und Entschlüsseln verwendet. Problem dabei ist die Schlüsselverteilung, aso wir erhält mein Gegenüber den Schlüssel zur Entschlüsselung einer Nachricht auf sichere Art und Weise.

Asymmetrisch:



Mathematisch „verwandtes″ Schlüsselpaar (siehe Diffie-Helmann). Prinzip ist einfach, Nachrichten die mit dem öffentlichen Schlüssel verschlüsselt werden können nur mit dem Private-Key entschlüsselt werden. Verteilung des public Key ist daher unproblematisch, er kann veröffentlicht werden. Einzige Problematik ist die Frage der Authentizität (wirklich Bob’s Schlüssel?). Aus diesem Grund gibt es im Internet Certificate Authorities (CA).

Hybrid:

Mit asymmetrischem Verfahren einen symmetrischen Schlüssel für die gemeinsame Benutzung übertragen

Authentizität:

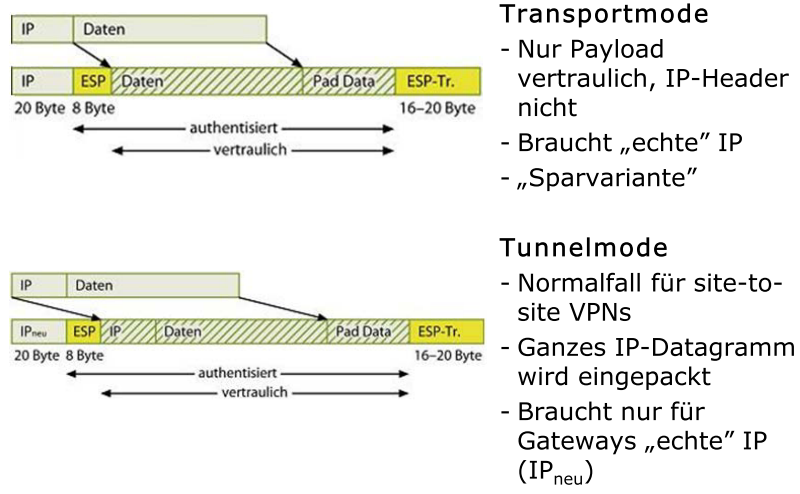
Sicherstellen, dass Gegenstelle authentisch ist, z.B. durch Prüfen der Kenntnis eines gemeinsamen Geheimnisses (PSK = Pre Shared Key)

Datenintegrität:

Den einzelnen Datenpaketen wird ein MAC (Message Authentication Code) angehängt, eine komplizierte Prüfzahl, die am Ziel erneut berechnet wird und den Originalwert ergeben muss. Ein HMAC kombiniert das Verfahren mit dem Hashwert eines gemeinsamen Geheimnisses (Authentizität)

## IPSec

IPSec kann in zwei verschienden Modis betrieben werden.

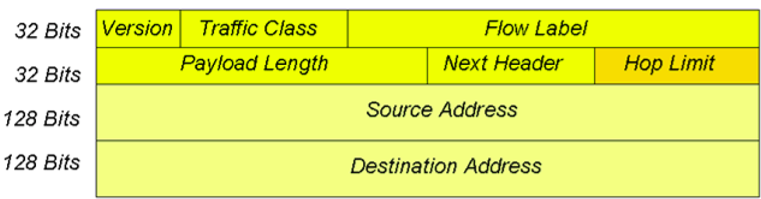


# WLAN, Funknetze, VoIP

# Szenarien von Netzzugängen

# IPv6

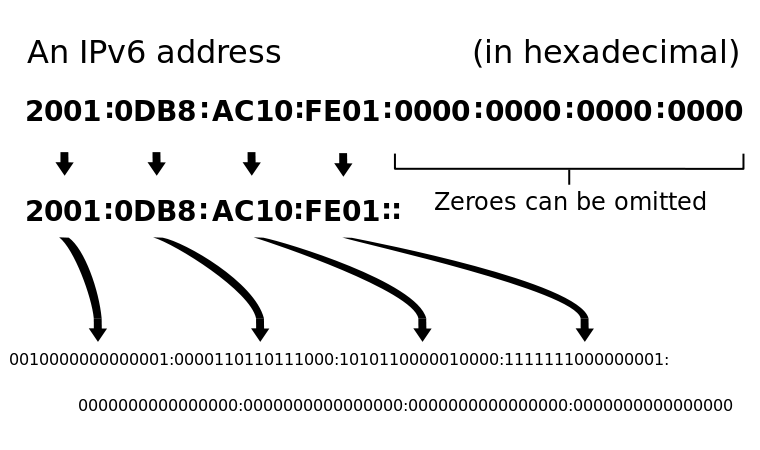
* Grösserer Adressraum: 2^128
* Variable und minimaler Header



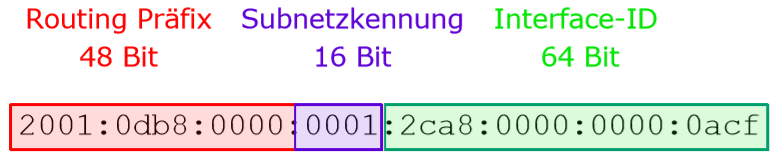
* Provider-Hierarchie in der Adresse ist Konzeptionellunverändert
* Routing Präfix und Host Suffix

**Adresse**

Ipv6 Adressen sind weitaus komplizierter als Ipv4 Addressen.



Bestimmte Felder in der Adresse haben unterschiedliche Funktionen.



Service Priovider erhalten 32 Bit Präfixe. Diese geben 48 Bit Netze weiter an die Kunden.

Beispiel 1

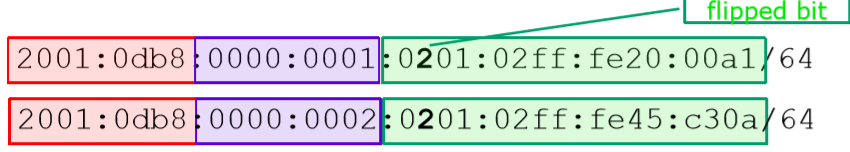
Kunde erhält das Netz 2001:0db8:0002/48 und definiert einen Host im subnetz 0001



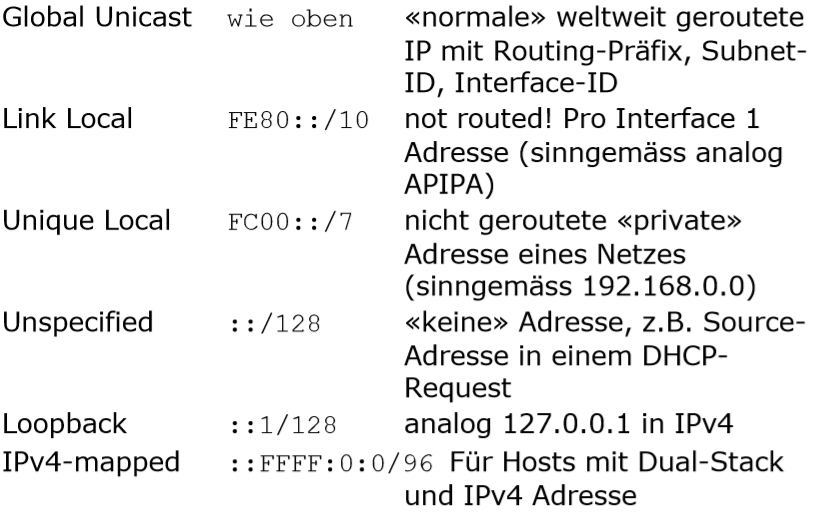
Das Minimum für eine Kunde-IP sind 64 Bit. 64 Bit weil der Kunde kein Subnetting machen darf.

Beispiel 2

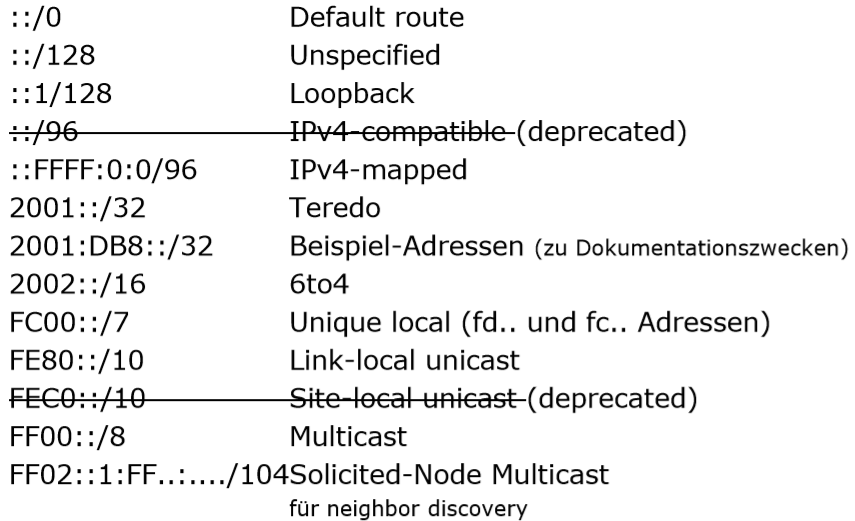
benachbarte /64er Kunden mit NICs von 3Com mit Herstellercode 00:01:02



**Unicast Adressen**



**Reservierte Adressen**



**Beispiel**

Lösung für oben mit «EUI Formation»:

Loopback: ::1

Link-Local: fe80::250:56ff:fec0:8

Global Unicast:2001:620:110: c101:250:56ff:fec0:8

All nodes Multicast: ff02::1 (link local wegen «im LAN»)

Solicited Node Multicast: ff02::1:ffc0:8

Weitere Multicast: ff00::/8 (diese Antwort könnte auch für die oberen 2 gegeben werden!)

Lösung für «von Hand» zugewiesene fixe IP: 2001:638:d:c101:acdc:1979:3:1008 (AAAA Eintrag von www.dfn.de)

Loopback: ::1

Link-Local: fe80::acdc:1979:3:1008

Global Unicast: 2001:638:d:c101:acdc:1979:3:1008

All nodes Multicast: ff02::1 (link local wegen «im LAN»)

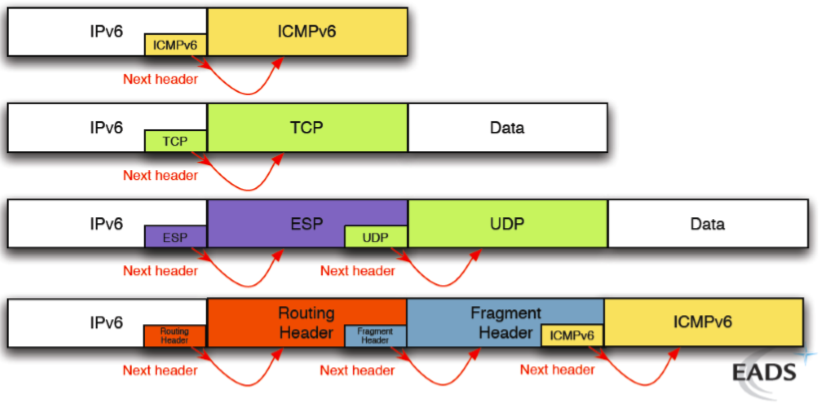
Solicited Node Multicast: ff02::1:ff03:1008

Weitere Multicast: ff00::/8 (diese Antwort könnte auch für die oberen 2 gegeben werden!)

**Extension Header**

Der Header von Ipv6 hat im Vergleich zu Ipv4 viel weniger Felder.

Der Extension-Header gibt an welchers Protokoll folgt.



## weitere Ipv6 Protokolle

**Stateless Address Autoconfiguration – SAA**

Netzwerkkonfiguration von Hand.

1. Link-Local mit EUI-64 Formation erzeugen
2. Die zugehörige Solicited-Node Multicast auf Einmaligkeit prüfen
3. Router Solicitation an FF02::2
4. Router Advertiment mit Netzparametern empfangen
5. Neighbor-Advertisment an FF02:1

Stateful Autoconfiguration mit DHCPv6

1. Nach SAA DHCPv6 Request via UDP an Port 546; Identifiziert durch DUID und IAIDs (eindeutige IDs, die bei Neustart nicht ändern = Rolle der MAC-Adresse in IPv4)
2. Antwort enthält DNS, NTP, SIP, NIS, … mit Ablaufdatum

**Neighbor Detection Protocol**

Ipv4 equivalent ist ARP

1. Neighbor Solicitation Message an Solicited Multicast Adresse des Partners z.B. FF02::1:ff12:3456 senden
2. Diese Solicitation enthält die Layer-2 Adresse des Absenders
3. Partner antwortet mit Neighbor Advertisment, das seine Layer-2 Adresse enthält
4. Beide können im lokalen Netz via MAC auf Layer 2 kommunizieren

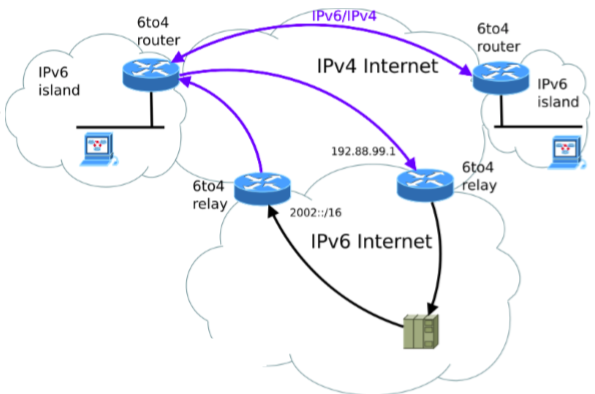
Wenn ein Host seine IP ändert, teilt er das allen Hosts über die All-Nodes Multicast Adresse (FF02::1) mit. Alle aktiven Hosts im Netz übernehmen diese Änderung und können die neue IP der MAC Adresse richtig zuordnen.

## Ipv4 to Ipv6 und umgekehrt

**6to4**

Verbindung von Ipv6 Netzen über ein Ipv4 Netz.

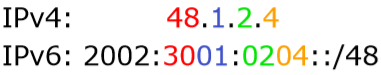
2002::/16 ist der Präfix für 6to4 Adressen, bei 6in4 globale Präfixe.



**6in4**

IPv6 wird auf IPv4-Strecke encapsulated

Beispiel der Zuweisung:



**Intra-Site Automatic Tunnel Addressing Protocol - ISATAP**

* In erster Linie für die Kommunikation reiner IPv6 Hosts in einem IPv4 Subnetz gedacht
* Die IPv4 Adresse wird direkt in die Interface-ID eingebaut: Interface ID = 0000:5EFE:<IPv4-Adresse>
* Sie ist automatisch Unique, wenn echte (ebenfalls eindeutige) IPv4 Adressen benutzt werden
* Das Präfix der IPv6 Adresse des Hosts kann global oder linklocal sein
* Bei globalen Präfixen ist Routing (oder relaying) ins IPv6 Internet möglich

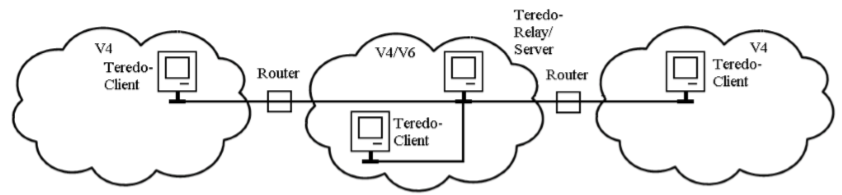
Beispiel:

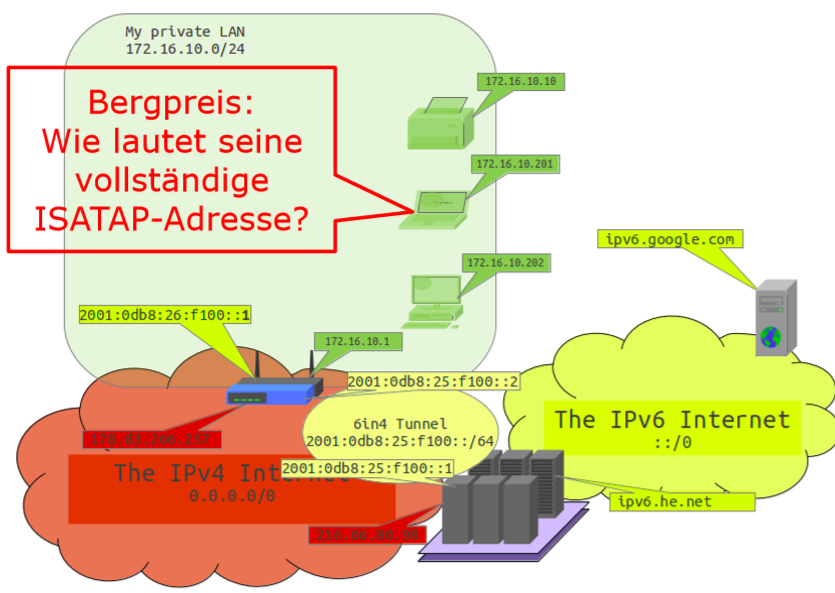
HSLU Adresse: 147.88.210.185

Interface-ID: 0000:5EFE: 9358D2B9

**Teredo**

* Workaround bei «dual-stack» Konfiguration, wenn z.B. 6to4 wegen NAT nicht möglich ist
* Der Protokolltyp 41 (IPv6 in IPv4 getunnelt) wird von den meisten NAT-Routern nicht geroutet (in der Payload befindet sich nämlich kein eigentliches IPv4)
* Der Transport durchs IPv4 Internet geschieht bei Teredo über UDP, wobei alle NAT-Infos mitgegeben werden. Es braucht allerdings einen Teredo Relay Server (3544/udp)!



Beispiel Tunnel Broker

Antwort:

Interface ID: 2001:0db8:26:f100::