Zusammenfassung - Advanced Communications

Marc Meier, CD

19. November 2015

Korrektheit und Vollständigkeit der Informationen sind nicht gewährleistet. Macht euch eigene Notizen oder ergänzt/korrigiert meine Ausführungen!

Inhaltsverzeichnis

1 Grundlagen

1.1 Grundprinzipien und Entwicklung des Internets

Das Internet entwickelte sich ab den 1960er Jahren. Es ging aus dem am Ende des Jahrzehnts entstandenen, vornehmlich militärisch und akademisch geprägten ARPA-Net hervor. Heutzutage wird es international kommerziell, industriell und auch akademisch (Katzenbilder) genutzt. Bei seiner Entstehung war vor allem eine dezentrale Struktur ohne zentrale Verwaltung von Interesse. Grund hierfür war die Angst des amerikanischen Department of Defense, dass eine atomarer Angriff zentrale Kommunikationspunkte außer Kraft setzen könnte. Die Kommunikation findet über hochgradig vernetzte Knoten mithilfe von Paketen statt.

Literatur: [1, 2]

1.2 Packet Switching

Paketvermittelte Übertragung bedeutet die Abkehr von der leitungsbasierten Vermittlung. Dabei werden längere Nachrichten in Datenpakete aufgeteilt und voneinander unabhängig versendet. Dies ermöglicht eine faire Verteilung der Leistungskapazität und redundante Wege bei einem Ausfall von Knoten oder Verbindungen. Im Gegenzug können konstante Bandbreiten nicht ohne Weiteres (Abschnitt 8) gewährleistet werden, ebenso ergeben sich unterschiedliche Laufzeiten von Paketen.

1.3 Dezentrale Verwaltung des Internets

1.3.1 Prinzipien

- Keine zentrale Verwaltung oder Behörde (trotz Einflussnahme)
- Demokratisches Zusammenwirken der Beteiligten / Wahlen
- Selbstorganisation
- Standards dort, wo sie erforderlich sind
- Dynamisch, offen für Neuigkeiten

1.3.2 Organisationen

ICANN: Vergibt IP-Adressen und betreibt die DNS-Rootserver.

IETF: Standardisierung von Protokollen in RFCs [18]

RIPE: Administration und technische Koordination

RIPE NCC: Adressvergabe in Europa und Zentralasien, Verwaltung der WHOIS-Datenbank.

DENIC eG: Domain-Verwaltung für die Zone .de

1.4 Standards

Standards ermöglichen die Kooperation im Netzwerk, nur durch sie können Geräte verschiedener Hersteller miteinander kommunizieren. Sie können textuell, mithilfe einer Referenzimplementierung oder anhand von Automaten (meist für zustandsbehaftete Protokolle) festgelegt werden.
Sie müssen verschiedenen Ansprüchen genügen: Vollständig, eindeutig (widerspruchsfrei) und stabil

Protokoll Standardisierte Regeln (Vorschriften) und Vereinbarungen zu Form, Ablauf, Steuerung und Sicherung (Fehler) der Datenübertragung in und zwischen Rechnernetzen, zwischen Einzel-Rechnern und zwischen Rechnern und Peripheriegeräten.

Standard Ein Standard wird von den verschiedensten internationalen und nationalen Organisationen sowie von großen Firmen erstellt. Ein Standard wird als verbindliche oder unverbindliche (empfohlene) Festlegungen schriftlich niedergelegt.

Ablauf einer Standardisierung bei RFCs:

1. **Proposed Standard**: Vollständige, konsistente Spezifikation vorhanden
2. **Draft Standard**: Mindestens 2 unabhängige, interoperable Implementierungen
3. **Standard**: Operationell stabil

Weitere Status: *Experimental*, *Informational* und *Historic*.

1.5 Netze, Autonome Systeme und Schichten

Große Teile des Internet-Backbones werden von wenigen Firmen bereitgestellt (Tier-1). Diese werden an einigen Knotenpunkten verbunden. Wichtiger Knotenpunkt in Deutschland ist DE-CIX in Frankfurt/Main. Man unterteilt folgende **Netzwerk-Schichten**:

Tier 1 : Ein Netzwerk, das mit allen anderen Tier-1-Netzwerken verbunden ist; *Internet-Backbone*; z.B. ATDN, GX, AT&T...

Tier 2 : Netzwerk, das mit vielen Netzwerken verbunden ist, aber Transit *einkauft*, um einige Bereiche des Internets zu erreichen; z.B. Deutsche Telekom

Tier 3 : Ein Netzwerk, das ausschließlich Transit *einkauft*, um das Internet zu erreichen

Autonome Systeme sind Ansammlungen von IP-Netzen, die als Einheit verwaltet werden. Innerhalb kommt ein einheitliches Routing-Protokoll zum Einsatz. Autonome Systeme sind untereinander verbunden und bilden das Internet.

1.6 Begriffe

Datendurchsatz Bla

Datenrate Bla

Routing Bla

2 Protokolle

Protokolle können als Vorschrift betrachtet werden, wie sich verhalten werden soll. Zur Interaktion wird beschrieben, welches Datenformat und wann etwas geschickt werden darf.

2.1 Zustandslose und zustandsbehaftete Protokolle

Bei **zustandslosen Protokollen** wird jede Anfrage in einer eigenständigen Transaktion ausgeführt, es existieren keine Vorbedingungen oder Sitzungsinformationen (UDP, HTTP, TFTP). **Zustandsbehaftete Protokolle** hingegen merken sich den aktuellen Zustand mithilfe einer Sitzung. Nachfolgende Anfragen können auf die Sitzungsinformationen zugreifen. Diese Zustandsübergänge können durch endliche Automaten dargestellt werden. Beispiele sind FTP, TCP und SMTP.

2.2 OSI-7-Schichten-Modell

1. Physical Layer / Bitübertragung
2. Data Link Layer / Sicherungsschicht / Datenübertragungsschicht
3. Network Layer / Vermittlungsschicht
4. Transport Layer
5. Session Layer / Sitzungsschicht
6. Presentation Layer / Darstellungsschicht
7. Application Layer / Anwendungsschicht

Gute **Eselsbrücken** sind:

- Alle deutschen Studenten trinken verschiedene Sorten Bier (deutsche Bezeichnungen, 7-1)
- An dem Samstag trug Verena 'nen String in Blau (deutsche Bezeichnungen, 7-1)
- Alle poppen Susis Tante nach der Party (deutsche Bezeichnungen, 7-1)
- Physiker, die nicht trinken sind potentielle Attentäter (deutsche/englische Bezeichnungen, 1-7)
- Alibaba präsentiert sich täglich nackt dem Personal
- Please Do Not Throw Salami Pizza Away (englisch, 1-7)

Jede Schicht n nutzt die darunterliegende Schicht $n-1$ um mit dem Kommunikationspartner zu kommunizieren. Daten höherer Schichten werden in niederen Schichten umkapselt. Die Bezeichnung der Pakete ist je nach Schicht unterschiedlich:

Data Link Layer : (Ethernet-)Frame

Network Layer : Paket

Transport Layer : Fragment

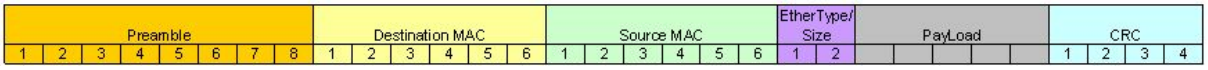
2.3 Ethernet

Das Ethernet-Protokoll wirkt auf den Layern 1 + 2 und wird im Standard **IEEE 802.3** definiert. Es kümmert sich um

- Elektrokrams (Physikalische Eigenschaften, Stecker, Stromversorung, Kabel etc.),
- Zugriffsverfahren auf das Medium,
- Adressierung (MAC),
- Protocol-Multiplexing,
- Flow Control (Logical Link Control),
- Fehlererkennung (CRC).

Es ähnelt den Standards **802.11** (WLAN), **802.15.1** (Bluetooth) und **802.16** (WiMAX).

Ein **Ethernet-Frame** hat eine Größe von 64 - 1518 Byte. Davon ausgenommen sind die Präambel und der SFD. Wird das VLAN-Tag genutzt, sind 1522 Byte möglich. Das **Ethernet-Paket** (Offensichtlich Präambel + SFD + Ethernet-Frame) umfasst folgende Felder:



Präambel : Zum Synchronisieren von Sender und Empfänger, *Einschwingphase* (8 Byte)

SFD : Festgelegte Sequenz 10101011 (1 Byte)

Ziel-Mac-Adresse : Adresse des Empfängers (8 Byte)

Quell-Mac-Adresse : Adresse des Senders (8 Byte)

VLAN-Tag : Nach IEEE 802.1q, optional (4 Byte)

Typ-Feld : Identifiziert die Art des nachfolgenden Inhalts, z.B. IP, ARP, etc...

Nutzlast

PAD-Füllfeld : Wird optional benötigt, um die Mindestlänge von 64 Byte einzuhalten [1]

CRC-Prüfsumme : Zur Fehlererkennung (4 Byte)

2.3.1 CSMA/CD

CSMA/CD regelt den Zugriff auf ein von mehreren Teilnehmern genutztes Medium (Kabel). Dazu prüft der sendende Host, ob das Medium frei ist, bevor er sendet. Beim Übertragen von Daten können Kollisionen erkannt werden. Der Sendevorgang wird dann nach einer zufälligen Zeit wiederholt. Aufgrund der verbreiteten Nutzung von Switches sind echte geteilte Medien inzwischen eher die Ausnahme.

⇒ Jeder Port am Switch bildet eine eigene *Kollisionsdomäne*. Die Bustopologie mit Koaxialkabeln (aber auch mit Hubs) wird nicht mehr genutzt.

2.3.2 Duplex / Half Duplex

Beim **Full Duplex** sind beide Seiten in der Lage, gleichzeitig zu Senden und zu Empfangen. Im Falle von **Half Duplex** ist dies nur wechselseitig möglich (vgl Walkie Talkie). Es sind verschiedene Realisierungen einer geteilten Nutzung eines Mediums möglich:

Zeitduplex (TDD) : Übertragung in verschiedenen Zeitschlitzen

Frequenzduplex (FDD) : Übertragung auf verschiedenen Frequenzen

Codeduplex : (nicht im Skript)

[1] Rausfinden, warum mindestens 64 Byte nötig. Vermutung: Kollisionserkennung

2.4 Switching

Switches sind Geräte auf dem OSI-Layer 2. Sie empfangen Ethernet-Frames und leiten sie anhand ihrer Empfänger-MAC-Adresse weiter. Im Gegensatz zum Hub wird dabei nur über den Port ausgegeben, hinter dem sich der Empfänger befindet. Die Ausnahme ist hierbei, wenn der Port des Empfängers nicht bekannt ist. Anhand der empfangenen Frames lernt ein Switch, wo sich Geräte befinden.

2.4.1 Realisierungsmöglichkeiten

2.4.2 Cut-Through und Store-and-Forward

Beim **Cut-Through** (auch *On The Fly Forwarding*) werden Pakete sofort nach Empfang der Empfängeradresse auf dem entsprechenden Port weitergeleitet, sofern dieser Frei ist. Diese Methode ist sehr schnell (Verzögerung ca. 40μs), leitet jedoch gegebenenfalls auch fehlerhafte Frames weiter, da CRC umgangen wird.

Store-and-Forward hingegen empfängt zuerst den gesamten Frame, prüft diesen und leitet ihn anschließend weiter. Offensichtlich werden keine fehlerhaften Pakete mehr in benachbarte Segmente weitergeleitet, dies wird jedoch durch erhöhte Latenz erkauft.

In der Praxis arbeiten Switches häufig im Cut-Through-Modus und schalten bei erhöhter Fehlerrate in den Store-and-Forward-Modus.

2.4.3 VLAN

Ermöglicht die Aufteilung von Switches in mehrere virtuelle LANs. Den Ports werden dabei einzelne VLANs zugeordnet. Auf diese Weise kann Hardware eingespart werden. Realisiert wird dies mit einem 4 Byte langen Feld im Ethernet-Frame:

- 2 Bytes **TPID** - Tag Protocol Identifier – Fester Wert 0x8100. Frame trägt die 802.1q/802.1p-Tag-Information
- 3 Bit **Priorität** (user_priority) – Benutzer-Prioritätsinformationen
- 1 Bit **CFI** - Canonical Format Indicator – Gilt für alle vorhandenen MAC-Adressinformationen im MAC-Datenpaket des Frames. Wert 0 das Format ist kanonisch (am wenigsten signifikante Bit zuerst); Wert 1 Format nicht-kanonisch. Benutzung im Token Ring/Source-Routed- FDDI-Media-Zugang, um die Bit-Order der Adressinformationen des verkapselten Frames festzulegen
- 12 Bit **VID** - VLAN Identifier – Identifizierung des VLANs zu dem der Frame gehört

Erleichtert die Arbeit eines Administrators, da es viele Probleme von physikalischen Verbindungen umgeht. (bspw. Viel Hardware, unflexibel, Anpassungen nur mit hohem Aufwand)

„Faulheit ist die Mutter der Ingenieurswissenschaften“

2.4.4 Trunking / Link Aggregation

Ermöglicht die Zusammenfassung mehrerer Ports zur Erhöhung des Datensatzes.

2.5 Asynchronous Transfer Mode

2.6 ATM

- ATM kann als Protokoll für Internettelefonie eingesetzt werden und bietet eine geringe Latenz (von unter 200ms).
- Im Gegensatz zu Ethernet bietet ATM Garantien(!) und besitzt einen geringen Header von 5Byte. Es wird eine Leitung für den Datenstrom geschaltet.

2.7 Internet Protocol

Beim Internet Protocol handelt es sich um ein Layer-3-Protkoll, weches auf die Layer-2-Protokolle Ethernet, ATM und FDDI aufsetzen kann. Es verwendet globale, logische Adressen. Aufgrund der Erschöpfung des IPv4-Adressraumes[2] (32 Bit) wird nach und nach IPv6 eingeführt (128 Bit)

[2] Weitere Maßnahmen, dem entgegenzuwirken sind etwa: NAT, CIDR, DHCP, Private Adressräume

2.7.1 IPv4

Wurde im RFC 791[36] definiert. Der Header eines IPv4 Paketes ist insgesamt 20 Byte lang. Davon sind insbesondere die folgenden von Interesse:

Version : In diesem Fall *4*, bei IPv6 offensichtlich *6* (4 Bit)

Header Length : Gesamtlänge des Headers kann 20 Byte überschreiben, wenn zusätzliche Optionen gesetzt werden. Angabe in 32-Bit langen Blöcken (4 Bit)

Total Length : Gesamtgröße des Pakets. Nach RFC muss jeder Host in der Lage sein, mindestens Pakete mit einer Länge von 576 Bytes zu verarbeiten. (16 Bit)

Type of Service : Type of Service nach RFC791(ursprünglich für Quality-of- Service-Anwendungen gedacht)

- bits 0-2: precedence
- bit 3: 0 = Normal Delay, 1 = Low Delay
- bit 4: 0 = Normal Throughput, 1 = High Throughput
- bit 5: 0 = Normal Reliability, 1 = High Reliability
- bits 6-7: Reserved for future use

Heute anders verwendet zur Servicebeschreibung durch Dienstklassen (DiffServ, 8 Bit)

Identification : Falls ein Paket fragmentiert wird, haben alle Fragmente die selbe Identification.

Flags : Reserved[3], Don't Fragment, More Fragments (3 Bit)

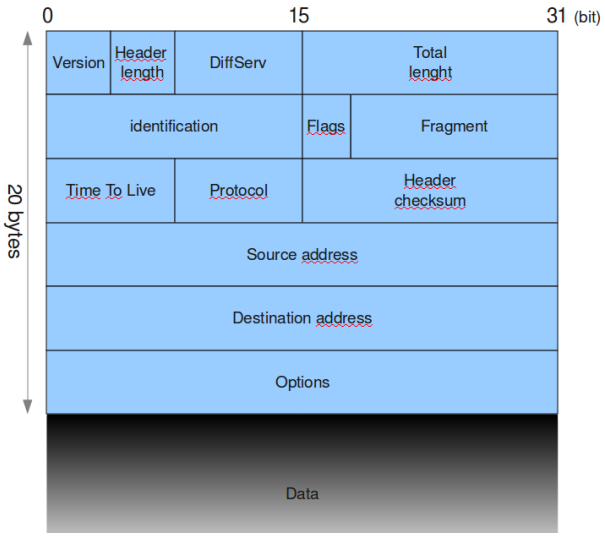
Fragment Offset : Kann ein Paket nicht auf einmal übertragen werden (z.B. bei kleinerer Maximum Transfer Unit, MTU), wird es fragmentiert. FO gibt an, ab welcher Stelle (gemessen in Blöcken von 8 Byte) dieses Paket die Daten enthält (MF Flag ist gesetzt) (13 Bit)

TTL : Anzahl der Hops, bis Paket verworfen wird (wird bei jedem Routingvorgang reduziert)

Protocol : Enthält für die darüber liegenden Layer Informationen, „sodass diese etwas damit anfangen können"

Checksum wird selbst als 0 betrachtet, und fließt so nicht in die Kalkulation mit ein

Options : Beispielsweise für Source Routing (Route ist im Paket vorgegeben); Sehr selten verwendet, häufig blockiert oder ignoriert



Nutzung von Adressklassen[3] aufgrund der Verknappung der Adressen durch CIDR[15] abgelöst. Dies ermöglichte Super- und Subnetting. Adressangabe bei CIDR im Format a.b.c.d/x, wobei x angibt, wie viele Bits zum Netz-Anteil der Adresse gehören. Subnetze dienen zur Aufspaltung von Netzen in Teile, um diese besser handhaben zu können (Broadcast-Domains, Logische Strukturierung, Dezentrale Verwaltbarkeit)

[3] Class A 1.x.y.z-126.x.y.z; Class B 128.0.y.z-191.255.y.z; Class C 192.0.0.z-223.255.255.z

2.7.2 IPv6

Die auffälligste Änderung von IPv4 zu IPv6 ist die vergrößerte Adressgröße (128 Bit). Damit ergeben sich $3,4 \cdot 10^{38}$ Adressen. IP - Adressen werden im Hexadezimalsystem zu je acht Word-Gruppen á 2 Bytes dargestellt. Verkürzte Darstellung möglich durch Verzicht auf „Nullen“ in einer Gruppe (einmal je Adresse). Es existieren IPv4-kompatible Adressen und die CIDR-Darstellung für Subnetze bleibt erhalten. Weiterhin wurde die Anzahl der Felder im Header reduziert und (optionale) Erweiterungsheader hinzugefügt. Wichtige Felder sind:

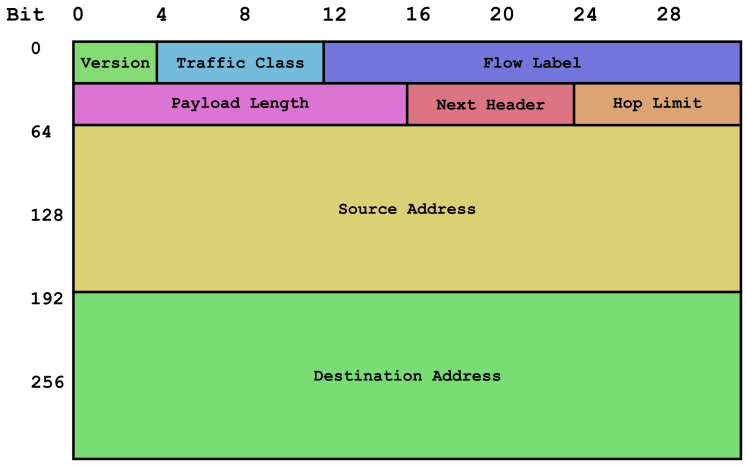
Traffic Class :

- 0 uncharakterisierter Verkehr
- 1 „Füllmaterial”, z.B. Newsgroups
- 2 zeitunkritischer Verkehr, z.B. EMail
- 3 reserviert
- 4 Mengendaten, z.B. FTP, NFS
- 5 reserviert
- 6 Interaktive Anwendungen, z.B. telnet
- 7 Steuerung, z.B. SNMP

Flow Label : Anwendung kann Datenstrom mit einem Flow-Label versehen, z. B. bei Streaming-Anwendungen. Flow nicht notwendigerweise an Verbindung gebunden (logisch, da IP nicht verbindungsorientiert arbeitet). Empfänger kann Datenstrom am Flow Label erkennen. [39, 38]

Next Header : Gibt an, dass ein weiterer Header folgt. In IPv6 sind viele Felder weggefallen. Next Header ermöglicht das Anfügen eines weiteren Headers. RFC 2460[8] bietet beispielsweise:

- Hop – by – Hop Options Header
- Routing Header
- Fragmentation Header
- Authentication Header
- Encapsulated Security Payload (ESP) Header
- Destination – Option – Header



2.7.3 Migration von IPv4 nach IPv6

Wie migriert man Millionen von Hosts im Internet auf IPv6? Langsam, nach und nach. Alle Hosts auf einmal sind nicht realisierbar. RFC 1933[16] schlägt drei Migrationsstrategien vor.

Tunneling : Zwischen zwei IPv6-Knoten wird ein virtueller Link aufgebaut. Die IPv6-Pakete werden (als Payload) in IPv4-Pakete verpackt und *normal* über das Internet geroutet.

Dual Stack : Auf Hosts und Routern werden sowohl IPv4 als acuh IPv6 eingerichtet. DNS kann A- oder AAAA-Records zurück geben, entsprechender Stack wird dann genutzt. Wird häufig mit Automatic Tunneling[4] genutzt.

[4] “IPv6-over-IPv4 tunneling where the IPv4 tunnel endpoint address is determined from the IPv4 address embedded in the IPv4-compatible destination address of the IPv6 packet.”

Außerdem besteht die Möglichkeit von **Assignment of IPv4 Global Addresses to IPv6 Hosts** (AIIH). Dabei wird eine IPv4-kompatible IPv6-Adresse genutzt. Diese entspricht dem 96-bit Präfix 0:0:0:0:0:0, gefolgt von der IPv4-Adresse. Ist der Client der Dual-Stack-Hosts und der Server IPv4-only, verlangt der Client für die Dauer der Kommunikation eine temporäre IPv4 Adresse beim AIIH Server (Kooperation von DNS und DHCPv6). Andernfalls (Client IPv4-only; Server Dual-Stack): DNS verlangt bei DHCPv6 eine temporäre IPv4 Adresse für Dual-Stack Host, welcher mit dieser rekonfiguriert wird.

Übersetzung der Header (Header Translation): Hierbei wird die IPv4 Unterstützung auf Systemen entfernt. Die IPv4-Pakete werden in IPv6-Pakete übersetzt; ein Translator übersetzt IP und ICMP Meldungen. Erweiterungsheader werden nicht, oder nur bedingt übersetzt. Probleme entstehen, weil sich einige Felder nicht immer übersetzen lassen und Adressumwandlung Datenbanken-Lookups erfordern.

2.8 User Datagram Protocol

Bei UDP handelt es sich um ein Layer-4-Protokoll.[34] Es dient zur Übermittlung kurzer Nachrichten an andere Systeme und garantiert weder Zuverlässigkeit noch Einhaltung der Reihenfolge der Pakete beim Empfänger. Die Adressierung geschieht über Ports (16 Bit). Der Header enthält 4 Felder (je 16 Bit): Quellport, Zielport, Datagram-Länge und eine Checksumme.

2.9 Transmission Control Protocol

TCP ist ebenfalls ein Layer-4-Protokoll.[35] Es garantiert eine Ankunft der Pakete in korrekter Reihenfolge. Clienten sehen die Verbindung als bidirektionalen Datenstrom, tatsächlich findet die Kommunikation über Pakete statt. Die Kommunikation erfolgt durch TCP zustandsbehaftet.

2.9.1 Paketstruktur

Der Header des TCP-Fragments ist 20 Byte groß. Wichtige Felder sind:

Sequence Number / Acknowledgement Number : Zerlegung des Datenstroms in nummerierte Blöcke. Größe der Blöcke ist variabel (Nagle Algorithmus). Verwerfen von Segmenten mit fehlerhafter Prüfsumme. Bestätigung empfangener Segmente. Nicht unbedingt für jedes Segment einzeln (Windowing). Erneuter Transport unbestätigter Segmente. Zusammensetzung des Datenstroms auf Empfängerseite

Flags : dienen unter anderem zur Steuerung des Verbindungsauf- und –abbaus.

URG - Urgent Flag (Urgent Pointer enthält Sequenznummer, die bevorzugt übertragen werden soll)

ACK – Acknowledgement

PSH – Push (Paket wird sofort an Anwendung weitergeleitet, ohne Zwischenpuffer)

RST – Reset (Unterbrechung der Verbindung)

SYN – Synchronized (Aufbau der Verbindung)

FIN – Finish (Beenden der Verbindung)

Window Size : Anzahl der Daten, die gesendet werden können bis ein Acknowledgement gesendet werden muss (in Bytes oder mit speziellem Option Header nach RFC1323 [19] auch bis zu 1GB, dann Linksverschiebung um bis zu 14 Bits, $2^{14} \cdot 64k = 1G$).

2.9.2 Zustände

Zum Aufbau einer Verbindung wird der **3-Way-Handshake** durchgeführt:

1. → SYN
2. ← SYN + ACK
3. → ACK

Ein Timeout findet typischerweise nach 75 Sekunden statt. Zum Abbau der Verbindung genügt das Senden und Quittieren eines FIN.

2.9.3 Sliding Window

2.9.4 Nagel-Algorithmus

3 Adressierung

3.1 MAC-Adressen

- Layer-2-Adressen für Ethernet
- 6 Byte / 48 Bit groß
- *Eigentlich* weltweit eindeutig
- *Eigentlich* in Hardware gegossen, trotzdem fälschbar
- Darstellung: Hexadezimal, Bits getrennt durch

  . : –

- Bit 3 bis Bit 24 an Hersteller gebunden (z.B. 00-60-2F-xx-xx-xx für Cisco)
- Broadcast: FF-FF-FF-FF-FF-FF

3.2 VPI und VCI bei ATM

3.3 IP-Adressen

Siehe Abschnitt 2.7.

3.4 Ports

Für die Layer-4-Protokolle UDP und TCP werden 16-Bit-Adressen verwendet. Diese werden als Ports bezeichnet. Ports 0-1023 sind dabei standardisiert. Wichtige Portnummern sind beispielsweise:

Port	TCP	UDP	Beschreibung
20	Ja	Nein	FTP - Datenübertragung
21	Ja	Nein	FTP - Kontrolle
22	Ja	(Ja)	SSH
23	Ja	Nein	Telnet
25	Ja	Nein	SMTP
53	Ja	Ja	DNS
80	Ja	Nein	HTTP
110	Ja	Nein	POP3
123	Nein	Ja	NTP
443	Ja	Nein	HTTPS

3.5 URIs, URNs und URLs

Uniform Resource Identifier : String; Benennt oder identifiziert eine Ressource z.B. \\139.30.3.23\iukp\beispiel.txt

Uniform Resource Name : Sonderform der URI, die eine Ressource in einem bestimmten Namespace benennt — urn:ip-addr:139.30.3.23

Uniform Resource Locator : Sonderform der URI, die zusätzlich das Protokoll angibt, über das die Ressource erreichbar ist z.B. http://139.30.3.23/beispiel.txt

3.6 *cast

Unicast : Einer sendet an einen, wie beispielsweise in einer TCP-Verbindung für HTTP. Adressierung durch Angabe des Empfängers.

Broadcast : Einer sendet an alle, wie beispielsweise in ARP oder DHCP. Hierzu werden spezielle Adressen (MAC: FF-FF-FF-FF-FF-FF; IP: höchste Adresse im Subnetz) verwendet. Broadcastbereiche sollten möglichst klein gehalten werden, um möglichst wenig Teilnehmer zu belästigen.

Multicast : Einer sendet an viele Mitglieder einer Gruppe. Siehe Abschnitt 9.

Concast : Viele senden an einen, Nachrichten werden so früh wie möglich zusammengefasst. Verhindert Implosion, Überlastung des Empfängers. Beispielswiese sinnvoll, wenn Fehlermeldungen zusammengefasst werden können. Keine direkte Unterstützung in IP, MAC, Ethernet

4 ARP, RARP

ARP wandelt Layer-3-Adressen in Layer-2-Adressen um. Es beschränkt sich nicht auf das Erfragen von MAC-Adressen zu IP-Adressen, sondern ermöglicht beispielsweise auch ATM ARP, IP over FDDI und IP over Token Ring.

- Erfragt für gegebene IP-Adresse eine MAC-Adresse.
- Host erzeugt Broadcast.
- Der angefragte Host darf darauf antworten (Unicast).
- Nachdem der Sender der Broadcast-Nachricht die Antwort erhalten hat, kann er die IP-Adresse der Ethernet-Adresse zuordnen.
- Diese Ethernet-Adresse wird dann für alle folgenden Pakete an diese Internet-Adresse verwendet, solange bis die Cache-Zeit abgelaufen ist.
- RFC 826 [33]

4.1 Einsatzfälle

1. Zwei Hosts möchten im selben Netzwerk (ausschließlich Layer 2) miteinander kommunizieren und kennen nur die Layer-3-Adresse (z.B. IP-Adresse) des Empfängers.
2. Ein Host benötigt die Layer-2-Adresse des Gateways, um andere Netze zu erreichen (Sonderfall des zuvor genannten)
3. Zwei Gateways wollen kommunizieren.

4.2 Paketstruktur

Wichtige Felder eines ARP-Requests sind:

Hardware type : Kennzeichen für die verwendeten Hardware-Adressen (1 für Ethernet, 2 Byte)

Protocol type : Kennzeichen für die verwendeten Protokoll-Adressen (L3) (0x0800 für IPv4, 2 Byte).

Hardware length : Länge der Hardware-Adresse in Bytes (6 für Ethernet, 1 Byte)

Protocol Length : Länge der Protokoll-Adressen (L3) (4 für IPv4, 1 Byte)

Operation : Unterscheidung von 1 Request und 2 Reply, da für beides gleiche Pakete verwendet werden (2 Byte, warum zur Hölle reserviert man für 2 Werte 2 Byte?)

Es folgen Sender hardware address, Sender protocol address, Target hardware address und Target protocol address

4.3 ARP-Caching

Um nicht für jedes IP-Paket einen neuen ARP Request zu stellen, werden die Ergebnisse zwischengespeichert. Typischerweise 10 Minuten lang. Kommt es während der Cache-Zeit zu einem Fehler (Host nicht erreichbar), wird erneut ein ARP-Request ausgeführt.

4.4 ARP-Announcements

Der anfragende Host sendet bekanntlich einen Broadcast, alle Host im gleichen Netz können folglich diese Information ebenfalls verwenden, und auf diese Weise eigene Anfragen sparen. Anwendungen:

- Unterbrechungslose Übernahme einer Protokoll-Adresse (z.B. IP-Adresse) in hochverfügbaren Systemen.
- Der anfragende Host kann auf diese Weise feststellen, ob eine Protokoll-Adresse bereits vergeben ist (Gratuitous ARP). Dieser Mechanismus wird bei IP-Autoconf verwendet.

IP-Autoconf [5] ist eine einfache Möglichkeit zur Adresskonfiguration ohne Server (DHCP, RARP).

- Host wählt zufällig eine Adresse.
- Überprüft mittels Gratuitous ARP, ob diese Adresse bereits vergeben ist.
- Falls ja, andere Adresse zufällig wählen und erneut überprüfen.
- Falls nein, Adresse benutzen und andere ARP Requests für diese Adresse beantworten.

4.5 ARP-Spoofing

Da ARP nicht kryptografisch abgesichert ist, kann jeder Host auf ARP Requests antworten oder ARP Announcements erzeugen. Ziel ist das Umleiten von Datenpaketen. Es gibt Programme, die Änderungen der Hardware-Adressen erkennen (z.B. arpwatch). Der Administrator muss dann die Ursache überprüfen

4.6 Proxy ARP

Wird von speziellen Routing-Protokollen verwendet. Jeder Nachbar (One-Hop-Neighbor), der auf der Route zum Ziel liegt, beantwortet einen eintreffenden ARP-Request mit seiner Hardware-Adresse. Pakete werden so von Host zu Host weitergeleitet

4.7 Reverse RP

Erfragt eine IP-Adresse bei bekannter MAC-Adresse und ist nicht unbedingt für die Funktionsfähigkeit von IP über Ethernet notwendig. RARP wird in RFC 903 [14] standardisiert. Haupteinsatzzweck ist die automatische Konfiguration der Protokoll-Adresse. Dabei sendet der Host einen RARP-Request mit der eigenen Hardware-Adresse (z.B. MAC). Ein RARP-Server beantwortet diesen Request und liefert die hinterlegte Protokoll-Adresse (z.B.IP). Da RARP Link-Local-Broadcasts verwendet, ist ein RARP-Server pro Netz notwendig. Benutzt gleiche Paketstruktur wie ARP, lediglich anderen EtherType[5] (0x8035)

5 DNS und WHOIS

Die Aufgabe des Domain Name Systems ist die Übersetzung von (Layer-4) IP-Adressen in, für den menschlichen Gebrauch besser verwendbare Namen. Der Name wird dabei aus hierarchischen Domains, getrennt durch Punkte (.) zusammengesetzt.
Direkt unter der Wurzel steht hierbei die sogenannte Top-Level-Domain. Bei Top-Level-Domains wird zwischen **länderspezifischen** (ccTLD) und **generischen** (gTLD) Domains unterschieden. Während erstere, bestehend aus zwei Buchstaben, durch lokal verantwortliche Institutionen[6] verwaltet werden, werden gTLD durch die **ICANN** oder beauftragte Institutionen vergeben. Bei der Vergabe gilt das *first come, first serve*-Prinzip. Für die Namensbildung gelten gewisse Regelungen, z.B. Ziffern 0-9, Bindestriche, Buchstaben, sowie einige ausgewählte lokale Buchstaben. Letztere müssen ASCII-kodierbar sein (ACE)[11]. Jede (Teil-) Domain ist maximal 63 Zeichen lang, insgesamt ist der vollständige Pfad jedoch 255 Zeichen lang. Folgende Daten sind bei der Domainanmeldung von Interesse:

Domaininhaber (Holder): Person, der die Domain *gehört*

Administrativer Ansprechpartner : Vom Inhaber Bevollmächtiger, darf alle Entscheidungen treffen (Admin-C)

Technischer Ansprechpartner, Zonenverwalter : Typischerweise Kontaktadresse der Firma, die die Domain betreibt (Tech-C, Zone-C)

Technische Daten : z.B. Nameserver

Hoster tragen oft nicht den Inhaber selbst als Admin-C, sondern sich selbst ein. Außerdem stellen sie in der Regel den Nameserver zur Verfügung. Die Registrierung einer Domain gilt immer für einen bestimmten Zeitraum. Ein Wechsel erfordert eine Freigabe beim alten Provider durch Inhaber oder Admin-C.

[5] Feld im Ethernet-Header, siehe Abschnitt 2.3
[6] Network Information Centers z.B. DENIC für .de

5.1 Weiteres Bla zu DNS

Beim Domain Name System handelt es sich um ein **sehr großes**, **hierarchisch strukturiertes**, **verteiltes**, **repliziertes** und **lokal verwaltetes System** [28, 29]. Aus dem hierarchischen Benennungsschema resultiert das verteilte Datenbanksystem des DNS. Jede Domain bestimmt selbst, wie die unter ihr liegenden Domains zugewiesen werden, hat dementsprechend eigene Verantwortlichkeiten. Ist ein Namensserver für eine solche Zone verantwortlich, kann er Anfragen autoritativ beantworten(Autoritäskonzept). Andernfalls wird die Anfrage aus dem Cache beantwortet oder an einen anderen Nameserver delegiert (Delegationskonzept). Das kann der Nameserver einer Subdomain oder (default) ein Root Name server sein.
Die Aufteilung in die Zonen erfolgt anhand von Punkten - Bsp: www.spiegel.de

5.2 Servertypen

Primary Server : Ist Hauptserver einer Domain und autorisiert, Anfragen zu seiner Domain verbindlich zu beantworten. Er verfügt über alle Daten dieser Domain, welche in Zonen-Dateien abgelegt sind, die der Verwalter des Servers erstellt

Secondary Server : Ist ebenfalls autorisiert, verbindliche Antworten zu seiner Domain zu liefern, lädt die Domain-Datenbank von einem Primary Server und aktualisiert sie bei Bedarf

Caching-only Server : Verfügt über keine eigenen Domain-Informationen. Fragt bei dem für die Domain zuständigen Primary oder Secondary Server nach und speichert die Antwort zwischen. Ist zur Beantwortung von Anfragen zu einer Domain „nicht autorisiert“ (kann die Genauigkeit und Aktualität nicht gewährleisten)

Slave Server : Reicht alle Anfragen, die er selbst nicht aus seinem eigenen Cache beantworten kann, an eine zuvor festgelegte Liste von anderen Servern (Forwarders) weiter. Forwarders fragen ihrerseits den zuständigen Server und speichern Ergebnis zwischen (Caching)

6 Timeouts, ACK, Bestätigungen

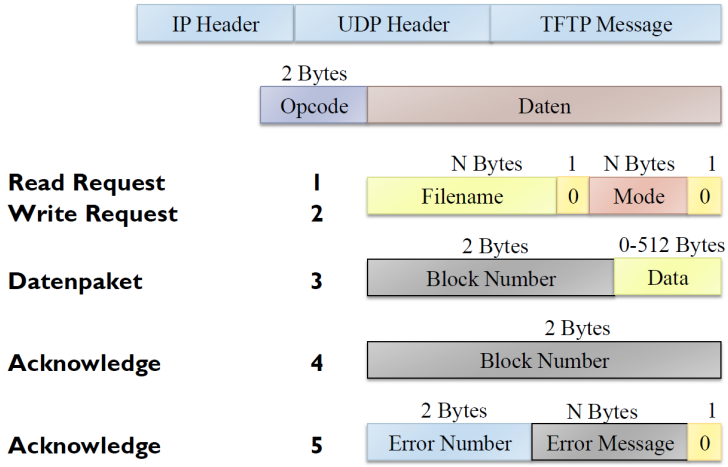
6.1 Begriffe

- **Positive Bestätigung**
  - Bestätigung, wenn ein Protokollschritt erfolgreich ausgeführt wurde.
  - Wartezeit ist ein Nachteil.
  - Reaktion auf verloren gegangene Bestätigungen muss vorgesehen werden.
- **Negative Bestätigung**
  - Fehlermeldung, wenn ein Protokollschritt nicht erfolgreich war.
  - Ausbleiben der Fehlermeldung ist nur eine notwendige Bedingung, keine hinreichende, dafür, dass Protokollschritt erfolgreich war.
- **Timeouts**
  - Ein Kontrollsignal in einem technischen System, das die Überschreitung der normalerweise für eine Aktion benötigten Zeit anzeigt. Damit sollen unkontrollierte Zustände (Warten, Endlosschleifen) verhindert und die Systemressourcen geschont werden.[44]
- **Acknowledgements**
  - Ein ACK dient der Bestätigung eines empfangenen Datenpakets.

6.2 TFTP - Trivial File Transfer Protocol

- TFTP ist ein vereinfachtes Datei-Übertragungs-Protokoll, das auf UDP aufbaut (FTP verwendet TCP) [42].
- Das Ziel von TFTP ist zum einen ein einfacher **Datentransfer** und zum anderen das Ermöglichen des **Bootens einer Workstation** ohne Diskettenlaufwerk.
- An TFTP gestellte Anforderungen sind, dass es kompakt sein soll (um ins ROM zu passen), keine Authentifikation.

- TFTP ist ein stop-and-wait Protokoll.
  - Kein neuer Protokollschritt, solange nicht der vorhergehende erfolgreich (korrekt) abgeschlossen ist.
  - Niedrige Protokoll-Leistung (Datendurchsatz).
  - Dafür einfache Implementierung.
  - Geänderte Reihenfolge stellen kein Problem dar, da stopand-wait Protokoll nur jeweils ein Paket hängig hat.
- Alternative
  - Positive Bestätigung kann verzögert werden (sliding window).
  - Übersendung mehrerer Pakete und dann erst warten auf eine Bestätigung.
  - Höherer Datentransfer (Beispiel TCP).
- Doppelte Pakete werden aufgrund der Blocknummer erkannt, während verlorene Pakete über Timeout beider Partner erkannt werden.
- TFTP Nachricht hat keine Checksumme, Datenveränderungen müssen durch die UDP-Checksumme abgefangen werden.
- Retransmission Timer
  - Wert sollte je timeout um bestimmten Faktor multipliziert werden (sog. Exponential backoff).
  - Retransmission timeout je Paket = 5 s.
  - Retransmission timeout für „Verbindung“= 25 s.



7 Routingkonzepte

7.1 Begriffe

7.2 Routing-Algorithmen

7.3 Link-State-Routing

7.4 Distance-Vector-Routing

7.4.1 Count-to-infinity-Problem

7.5 Routing in MANETs

8 Quality of Service

9 Multicast

10 Zeitsynchronisation

10.1 Notwendigkeit der Zeitsynchronisation

Authentisierungsverfahren müssen die Gültigkeit von Schlüsseln überprüfen können. Ebenso haben einige Abläufe eine beschränkte zeitliche Dauer. Andere Verfahren benötigen Time-Stamps, Vorher-Nachher-Beziehungen von Ereignissen müssen bekannt sein.

Uhren in verteilten Systemen sind nicht zwangsweise synchron. Sie können unterschiedliche Zeiten aufweisen, je nach Art der Uhr *ticken* sie unterschiedlich schnell (Quarzuhr, Atomuhr). Zusätzlich kann die physikalische Umgebung diesen Effekt verstärken (z.B. Temperatur einer Quarzuhr). Eigenschaften von Uhren nach Mühl sind: Drift, Auflösung und Abweichung (von der Realzeit).

10.2 Logische Zeit

Manchmal reicht es, nicht den genauen Zeitpunkt von Ereignissen, sondern lediglich die Vorher-Nachher-Beziehung zu kennen. Man spricht von **Lamport-Zeitstempel** oder **Vektoruhren**. Empfangene Lamport-Zeitstempel werden einfach bei jedem Kommunikationsschritt um eins inkrementiert und dann als eigener Zeitstempel versendet. Bei der Vektor-Zeit werden zusätzlich die Zeitstempel der anderen Kommunikationspartner mitgeführt. Dies hat den Vorteil, dass ein kausaler Zusammenhang immer erkennbar ist.

10.3 Herausforderungen

- Logische Zeitstempel (Lamport-Zeitstempel und Vektor-Zeit) genügen den Anforderungen nicht immer.
- Viele Uhren müssen synchronisiert werden.
- Die Übertragung eines Zeitstempels kostet Zeit – Latenz.
- Die Latenz ist nicht konstant – Jitter.
- Die Latenz ist nicht symmetrisch.

10.4 Algorithmen

Anforderungen an Algorithmen sind symmetrische oder bekannte Laufzeiten und eine konstante Latenz. Letzteres ist in paketvermittelnden Netzen nicht erreichbar.

10.4.1 Algorithmus von Cristian

Clients fragen beim Server nach der korrekten Zeit. Server kennt die Zeit aus zuverlässiger, externer Quelle.

10.4.2 Berkeley-Algorithmus

Server fragt alle verfügbaren Clients nach der Zeit und bildet den Mittelwert. Zeit wird anschließend an Clients verbreitet.

10.4.3 Marzullo's Algorithmus

Versucht den Einfluss des Jitters durch mehrfache Messungen zu eliminieren. Messungen werden solange durchgeführt, bis das Konfidenzinterval den Anforderungen entspricht.

10.5 NTP

NTP[26, 25] verwendet Marzullo's Algorithmus. Es löste den ICMP Timestamp Request (Abschnitt 10.7) ab und ist mittlerweile in Version 4 aktuell. NTP-Server lauschen auf UDP-Port 123. Im Internet sind Genauigkeiten von $\leq 10ms$ erreichbar.

10.5.1 Paketaufbau

Anmerkung: Paketaufbau von Version 4 scheint sich recht stark von Version 3 zu unterscheiden. Ich bleibe bei der in der Vorlesung besprochenen Version 3.

Leap Indicator : Gibt an, ob Schaltsekunde in dieser Minute hinzugefügt oder entfernt wird (2 Bit)

Status : Gibt mögiche Fehler an.

0. clock operating correctly
1. carrier loss
2. synch loss
3. format error
4. interface (Type 1) or link (Type 2) failure

Type : Typ[7] der Referenzuhr: 1 — Primärreferenz (z.B. Atomuhr) bis 4 — *Eyeball and wrist watch*

Precision : Angabe der Genauigkeit der Uhr (als Exponent zur Basis 2)

Estimated Error : Geschätzter Fehler zum Zeitpunkt der Synchronisation in Sekunden.

Estimated Drift Rate : Geschätzte Drift der Uhr (ohne Einheit).

Reference Timestamp : Zeit, auf den die Uhr bei der letzten Synchronisation eingestellt wurde.

Originate Timestamp : Zeit beim Senden des Pakets

Receive Timestamp : Lokale Zeit bei Ankunft des Pakets

Transmit Timestamp : Lokale Zeit beim Senden der Anwort

10.6 SNTP

Vereinfachung von NTP, bei der auf eine Wiederholung der Zeitsynchronisation verzichtet wird. So sind etwa Synchronisationen über Multi- und Broadcasts möglich. Es werden dasselbe Nachrichtenformat und derselbe UDP-Port, wie bei NTP genutzt[27].

10.7 ICMP - Timestamp Request und Reply

Timestamp Request[37] ermöglicht die Anfrage eines anderen Systems nach der aktuellen Zeit. Es handelt sich um ICMP-Nachrichten mit den Type-Werten 13 (Request) oder 14 (Reply)[8]. Empfohlener Rückgabewert sind die Millisekunden seit Mitternacht (Coordinated Universal Time - UTC). ICMP-Data-Bereich kennt Felder **Originate Timestamp**, **Receive Timestamp** und **Transmit Timestamp** (je 32 Bit); diese scheinen dieselbe Bedeutung wie bei NTP zu haben. Außerdem **Identifier** und **Sequence Number**, diese gestatten dem Sender eine Zuordnung von Replies bei mehreren Requests. Letztere werden vom Sender eingetragen und vom Empfänger in die Antwort kopiert.

11 Internet Control Message Protocol

ICMP ist ein Layer-3-Protokoll zur Steuerung des Nachrichtenaustausches im Internet[37][9]. Es dient häuptsächlich dem Austausch von Status- und Fehlermeldungen zwischen Gateways und Hosts. Die Nachrichten werden über IP-Datagramme übertragen, das Protokoll ist verbindungslos.

[7] Neuere Versionen bezeichnen dies als Stratum; Es gibt quasi eine Hierarchie bei den (Genauigkeiten der) NTP-Server
[8] Skript sagt hier 17 und 18, im RFC stehen allerdings 13 und 14. Habe Thomas drauf hingewiesen.
[9] Laut Folien RFC 950[30], aber das ist Quatsch (*Internet Standard Subnetting Procedure*)

11.1 Paketaufbau

Eine ICMP-Nachricht besteht aus den Feldern **Type**, **Code**, **Checksum** und **Data**, angeführt von einem 20 Byte IP-Header. Der IP-Header hat häufig vorbestimmte Feldwerte, z.B. Type of Service = 0, weitere Infos im RFC.

Type : Bestimmt das Format der weiteren Felder. (1 Byte)

Code : Inhalt Abhängig vom Type (1 Byte)

Checksum : Prüfsumme[10] (2 Byte)

Data : Nicht immer genutzt; Enthält beispielsweise bei Type *Redirect Message* (5) die *Gateway Internet Address* oder die in Abschnitt 10.7 beschriebenen Felder für Timestamps.

11.2 Beispiele

Im Skript existiert eine umfangreiche Tabelle von Funktionen und entsprechender Werte für Type- und Code-Felder. Diese auswendig zu lernen entspräche nicht 80-20. Daher beschränke ich mich hier nur auf zwei Beispiele (+ Zeitsynchronisation in Abschnitt 10.7)

11.2.1 Destination Unreachable

Wird beispielsweise gesendet, wenn er Empfänger laut den Routing-Tabellen eines Gateways nicht erreichbar ist, z.B. weil die Distanz zum Zielnetzwerk unendlich ist. Empfänger im IP-Header ist der ursprüngliche Absender einer (nicht zustellbaren) Nachricht. Der ICMP-**Type** ist 3, der Code gibt genauere Informationen zur Fehlerursache (z.B. 3 — port unreachable)

11.2.2 Ping

Beispiel nicht aus dem Skript. Beim Request wird ein **Echo-Request** (Type 8) versendet. Das Feld **Code** enthält den Wert 0[11]. **Identifier** und **Sequence Number** helfen bei Unterscheidung mehrerer Requests/Replys. Im **Data** Feld kann scheinbar noch unbestimmter Payload mitgesendet werden.

Bei der Antwort werden Sender- und Empfängeradresse im IP-Header vertauscht. Der **Type** ist 0 (Echo-Reply). Weitere Felder werden aus dem Request übernommen.

11.3 Regeln

ICMP-Fehler-Nachrichten werden nie als Folge auf

- eine ICMP Fehlermeldung („Teufelskreislauf“) erzeugt.
- ein Datagram, das an eine IP Broadcast Adresse oder eine IP Multicast - Adresse (class D) gesendet wird, erzeugt.
- ein Datagram, welches als Link - Layer Broadcast gesendet wird, erzeugt.
- ein anderes Fragment als des ersten erzeugt.
- ein Paket erzeugt, dessen Quelladresse nicht einen einzelnen Host definiert.

12 Internet Group Message Protocol

13 Voice over IP

14 World Wide Web und HTTP

Durchbruch des Internets erfolgte erst nach dem Erscheinen des ersten graphischen WWW-Browsers („Mosaic“). Dieser ermöglichte erstmals den komfortablen und effizienten Zugriff auf die Ressourcen des Internets. Das zuvor „beliebte“ Gopher wurde relativ schnell verdrängt. Bis zum Ende der 90er-Jahre vervielfachte sich die Zahl der Websites dramatisch (1993 ca. 50; 1994 ca. 800, 2000 über 22 Millionen)

[10] Scheinbar abhängig vom Type; häufig: *The checksum is the 16-bit ones's complement of the one's complement sum of the ICMP message starting with the ICMP Type.*

[11] Laut RFC scheint die Möglichkeit zu bestehen, dass hier etwas anderes steht. Warum das so ist, kann ich allerdings nicht finden

14.1 Anforderungen an Internet-Protokolle

- Weitestgehende Unabhängigkeit von der Netzwerk-Hardware.
  - Übertragungsmedium (Kupferkabel, Lichtwellenleiter usw.)
  - Typ des lokalen Netzwerks (Ethernet, Token Ring usw.)
- Erreichbarkeit aller Rechner im gesamten Internet.
- Fehlererkennung und Fehlerkorrektur.
- Logische Adressierung (Adresse eines Rechners sollte nicht von der Netzwerk-Hardware abhängen, da bei Defekt einer Ethernet-Karte beispielsweise eine ausgetauschte Karte zu einem Adresswechsel führen würde).
- Verbindungsherstellung.
- Datenübertragung (Versendung von Datenpaketen /Datagrammen).

14.2 Hypertext Transfer Protocol

HTTP wurde in RFC 2616[13] definiert. TLS[20] führte später einige Sicherheitsfeatures ein. Trotz des *Hypertext* im Namen ist es generisch und ermöglicht die Übertragung beliebiger Inhalte. HTTP ist **zustandslos** und nutzt den TCP-Port 80. Es handelt sich um ein **Request-Response-Protokoll**. Zusätzlich werden Nachrichten im ASCII-Klartext gesendet, solange sie nicht verschlüsselt werden. Ressourcen werden über URIs[4] adressiert.[12]

Eine einfachste Anfrage an den Server wäre beispielsweise `GET index.html HTTP/1.1`. Hierbei handelt es sich offensichtlich um eine Bitte, die Datei index.html auszuliefern und dabei HTTP 1.1 zu verwenden.

14.3 File Transfer Protocol

15 Peer-to-Peer

16 E-Mail

17 Autokonfiguration

18 Dateien und Drucken

19 Telnet, SSH und rlogin

20 Extensible Messaging and Presence Protocol (XMPP)

21 LDAP

22 Authentication Protocols

23 Simple Network Management Protocol

24 Mac-Sublayer

25 Mobile Netzwerke

26 HTTP2 und SCTP

27 Thomas Fragestunde

27.1 08.04

- Merkmale von Verbindungen

[12] Angeführt durch die Protokollangabe http:// oder https://, dabei handelt es sich dann allerdings um eine URL.

1. Latenz
2. Datenrate
3. Jitter
   (Änderung der Latenz über die Zeit)
4. Verfügbarkeit
5. Fehlerrate

- Bandbreite wird in Hz und nicht in Mbit/s angegeben (Wlan bspw.)
- Ursachen für Jitter
  - Es ist notwendig zu warten, bis gesendet werden kann
  - Sich ändernde Routen, durch bspw. Wegänderungen, Datenaufteilung durch den Router
  - Zeit, Route, Warteschlangenproblematik, gemeinsam genutzte Medien, CSMA
- Warum Begrenzung der Übertragungsraten?
  - Physikalisch - Frequenzen
- Was begrenzt den max. Durchsatz eines Glasfaseranschlusses
  - unterschiedlich lange Wege
  - Kapazität (Zeit bis Ladung angekommen ist)
  - Transistoren schalten nicht so schnell
  - Signal-Rausch-Verhältnis

27.2 09.04

- Was begrenzt die Bandbreite?
  - Kapazität → Grenzfrequenz
- CSACD
  - Warten auf freies Medium
  - Es gibt jedoch keine Garantien
  - Verursacht Jitter
- Jitter Gegenmaßnahmen
  - Puffer (Warteschlange, FiFo)
  - Pakete werden in gleichmäßiger Reihenfolge erhalten
- Leitungs- vs. Paketorientiert
  - ?
- Möglichkeiten zur Beschreibung eines Standards
  - textuell (schwer)
  - Referenzimplementation (bspw. Bittorrent - aber berücksichtigt ggf. nicht alle Fälle)
  - Automaten (5-Tupel, Mealy/Moore)
- Zustandsbehaftete und zustandslose Protokolle
  - Beispiel TCP (closed, SYN, etc.) besitzt einen Zustand
  - Ein Zustand merkt sich die „Vorgeschichte “und damit was passiert ist. Somit ist ein Login immer zustandsbehaftet.
  - Werden die Zwischenschritte nicht gespeichert, ist es zustandslos.
  - Cookies bieten die Möglichkeit einen Zustand nachzurüsten.
- Autonome Systeme sind „Provider “, die miteinander über *Peering Points* verbunden sind
- Minimierung der Aufreihungslatenz

  - Entsteht dadurch, dass das Paket erst versendet wird, wenn es vollständig da ist
  - Lösung, Weiterleiten, sobald die ersten 6 Byte (Adresse) bekannt sind
- Problem von CRC (cyclic redundancy check)?
  - Es entsteht eine unnötige Belastung, wenn fehlerhafte Pakete weitergeleitet werden
- Wann sollte umgeschaltet werden zwischen den verschiedenen Switch-Modi?

27.3 22.04

- Implemenationen von Switches
  - Shared Memorie
  - Schaltmatrix (crossPoint → komplexe Schaltungen)
  - Gemeinsamer Bus
- Welche Geräte arbeiten auf welchen Layern?
  - Layer 3: Router
  - Layer 2: Switches
- Welche Dienste werden von welchen Layern erbracht?
  - Layer 3: Routing, logische Adressierung
  - Layer 2: Media Access
- –

27.4 04.05

- Routing?
  - Erfolgt auf dem Network-Layer
  - Dient der Kommunikation zwischen Geräten (IP kann als Protokoll genutzt werden)
  - Ziele sind das Verbinden von Netzen miteinander und das Finden des richtigen Weges.
  - IP bietet die Möglichkeit Netze logisch einzuteilen.
- DNS (Domain Name System)
  - Zum Auflösen einer Domain zu einer IP-Adresse
  - Es gibt verschiedene Ebenen ...
  - Beschreibung der Struktur: hierarchisch, mit Root, etc.
  - Anfragen werden mitunter gecached
- Migrationsstrategien von IPv4 nach IPv6
  (Tunneling, Dual Stack, Header Translation)
- Nachteile von Tunneling?
  - mehr Header-Daten
  - äußeres Protokoll bspw. nicht von Admins kontrolliert werden
- Funktionsweise eines Routers
  - Statisch (fest eingegebene Liste)
  - Dynamisch (Routingprotkolle, Topologieermittlung, Pfaderkennung)
- Möglichkeit zur Topologiebestimmung
  - Es werden Testnachrichten (TTL=1) im Netzwerk versendet
  - Auf diese Weise werden die Nachbarn bestimmt
  - Anschließend erfolgt die Weitergabe dieser Informationen: n-Hob-Nachbarn
  - Eine wichtige Bedingung hierbei ist, dass kein häufiger Wechsel erfolgt

27.5 13.05

- Wie lange Routingverfahren brauchen um zu konvergieren hängt von der Größe der Netzwerke ab. Mögliche Lösungen sind
  - Näherungsverfahren (Es muss nicht alles bekannt sein)
  - pro- bzw. reaktives Routing: Abhängig davon, ob Kenntnisse über das Netz vorhanden sind, oder nicht
  - Reduzierung der Graphenkomplexität - Einführen eines Backbones als „minimal dominating set“. Eine Menge, aus der alle Knoten erreichbar sind
- Das Count-to-infinity-Problem kann durch einen Distanzvektor behoben werden
- Netzwerkparameter, die die Güte beschreiben: Datenrate, Datensatz, Fehlerrate
- Was ist zu tun, wenn mehr Bedarf als Ressourcen bestehen
  - Flusskontrolle
  - Prioritäten festlegen
    (Ein Gespräch sollte eine Latenz von unter 1/45 Sekunden haben, 64 kbit/s + Header)

27.6 18.05

- Gegen was wirkt die FiFo-Queue?
  → Zur Jitter-Bekämpfung, bewirkt dafür aber Latenz
- Ursachen für Verzögerungen auf Layer 2 (Ethernet)
  - Shared Medium
  - Alle Teilnehmer senden, dadurch kommt es Kollisionen, die Teilnehmer müssen warten/lauschen/nach einer zufälligen Zeit erneut senden
- Beispielhafte Übertragungsraten
  - Audio CD
    - 44,1 kHz Abtastfrequenz, 16 Bit Abtastung (geringer Quantisierungsfehler), *2 (Stereo)
    - 1,5 Mbit/s ohne Fehlerkorrektur
  - Video 10-100 Mbit/s
  - Tippgeschwindigkeit: 200-300 Anschläge die Minute
- Ursachen für Latenz
  - Entfernung, Router/Switches, Aufreihung, Pakete erst abgesendet, wenn voll

27.7 25.05?

- RTP (Real-Time Transport Protocol)?
  - zur kontinuierlichen Übertragung von audiovisuellen Daten
  - Normalerweise über UDP
  - ? Kommt es zu einem zu hohen Verlust, erfolgt ein Kodecwechsel

27.8 03.06

- SMTP/POP3
  - Via TCP (Layer 4)
  - Port 25,584
  - Thomas erwartete detailiertes Wissen (bspw. befindet sich am Ende ein Punkt, etc.)
  - Sicherheit, Authentizität,... gibt es nicht
- Zustandsbehaftete Protokolle sind: POP3, IMAP, TCP, FTP
- IMAP4

  - Port 143
  - neuer und bietet Verwaltung, Ordner, etc.
  - „Besser“als POP3 und arbeitet auf dem Server (suchen, kopieren, etc.)
- Sinn von Subnetzmasken?
  - Anhand der Ziel-, der eigenen Adresse und der Netzmaske kann entschieden werden, ob sich der Empfänger im eigenen Netzwerk oder außerhalb befindet
- Nutzung von UPnP: Fernseher, Smartphones, Playstation, DSL-Router, Drucker, etc.
- ICMP?
  - Steuerungsnachrichten für Geräte im Internet
  - Meldungen, wenn Nachricht nicht zugestellt wurde (Host, Port, etc. unreachable)
  - TTL abgelaufen (bei IP-Paketen)
  - EchoRequest & EchoReply
  - „Mach-mal-langsam“Nachrichten
- Routingprotokolle
  - Topologieerkennung
  - Topologieverbreitung
  - günstigste Wege finden (Kostenfunktion)
  - Distanzvektorproblem
  - CountToInfinity $\rightarrow \infty = 16$
- typische Protokollfragen
  - VLAN
  - HTTP
  - FTP
  - Die zwei Modi von Switches
  - Routing Protokoll

27.9 15.06

- Wie groß sollten LAN-Segmente gewählt werden?
  - Nachteile von groß: Broadcast Domain
  - Nachteile von klein: Mehr Routing, mehr verpacken, Aufreihungslatenz, Prüfsummen, mehr Overhead
- Anforderungen an das Networkfilesystem (NFS): sicher, transparent, Mehrbenutzerbetrieb, schnell, Latenz
- RFC Lebenszyklus: ?
- XMPP
  - basiert auf XML zum Nachrichtenaustausch
  - Ende-zu-Ende chatten
  - Geringe Datenübertragungsrate (Chat, ohne Bilder)
  - Character-Encoding
  - Anwesenheitserkennung (on, off, tipping)

27.10 17.06

- Was ist P2P?
  - direkte Kommunikation via Vermittlungsstellen
  - Verfügbarkeit: auf einem PC (Lösung: Replikation)
  - dezentral: keine legislativen/judikativen Entscheidungen
  - Lastverteilung
- WebDav
  - Web-based Distributed Authoring and Versioning
  - Standard zur Bereitstellung von Dateien im Internet
  - setzt auf HTTP auf - und fügt hinzu
  - Mit WebDAV können ganze Verzeichnisse übertragen werden.
  - Einsatz für CMS
- Angaben von RoutNameServern für Topleveldomains
  - Country: de,...
  - General: edu, org,...
- LDAP?
  - hierarchische Datenbank
  - Aktives Verzeichnis: Nutzergruppenverwaltung
- Aufgabe des Sessionlayers
  - Zusammenfassung der verschiedenen Kommunikationspfade
  - Nutzung der Zustände (-behaftete: SSH,FTP,TCP,IMAP4,POP3)
- Begriff Authentisierung
  - "Dem System klar machen wer man ist"
  - Biometrisch, Token, Schlüssel, wissensbasiert

27.11 24.06

- Kerberos
  - Kerberos ist ein verteilter Authentifizierungsdienst
  - Kerberos soll eine sichere und einheitliche Authentifizierung in einem ungesicherten TCP/IP-Netzwerk auf sicheren Hostrechnern bieten.
- Medienzugriff auf Layer 2
  - Aloa (Bei Zugriff wird einfach zugerufen)
  - Slotted Aloa (Zugriff nur zu bestimmten Zeiten, dadurch geringere Kollisionen)
  - Bei Ethernet: CSMACD
    - Carrier Sense (Lauschen auf dem Medium)
    - Multiple Access
    - Collision Detection
  - Pure Aloa
  - GSM als kollisionsfreies Zugriffsverfahren (sobald die Verbindung besteht - auf Grund des exklusiven Medienzugriffs??)
- Multiplexverfahren
  - Methoden zur Signal- und Nachrichtenübertragung, bei denen mehrere Signale zusammengefasst und simultan über ein Medium übertragen werden.
    (Raum, Frequenz, Zeit, Code)

27.12 29.06

- Faktoren die in eine Kostenfunktion beim Routing miteinbezogen werden können
  - Topologiekenntnisse
  - Durchsatz/Datenrate
  - Latenz
  - Reale Kosten
- Dijkstra Algorithmus
- Hidden Station Problem
  - RTC/CTS lohnt sich, wenn die Pakete klein genug sind - verglichen zu den Nutzerdaten
  - OLSR (Optimized Link State Routing)
    (Reduzierung der Graphenkomplexität - Backbone Bildung)

27.13 08.07

- MIB (bei SNMP)
  - Management Information Base (deutsch: Verwaltungsinformationsbasis) beschreibt die Informationen, die über ein Netzwerk-Management-Protokoll abgefragt oder modifiziert werden können.
  - Das Simple Network Management Protocol ist ein Netzwerkprotokoll, um Netzwerkelemente von einer zentralen Station aus überwachen und steuern zu können.
  - Es werden Schlüssel beschrieben, die zur Speicherung des „Gesundheitszustandes“von Geräten dienen.
  - Eine MIB-Datenbank erklärt, für welche Information ein Schlüssel steht.
- Hemming-Distanz
  - Der Hamming-Abstand ist ein Maße für die Unterschiedlichkeit von Zeichenketten.
  - Die Distanz zweier Blöcke mit fester Länge ist dabei die Anzahl der unterschiedlichen Stellen.
  - HD wird zur Fehlererkennung und zur Fehlerkorrektur benutzt.
  - Ob eine Fehlererkennung oder -korrektur stattfinden kann, hängt von der Hamming-Distanz ab.
  - FEC - forward error correction
    - Am einfachsten: 2-3 mal senden
    - Fehler erkennen: Distanz von 2
    - Fehler korrigieren: Distanz von 3
- Aktives vs. passives Scanning
  - Passiv: Der AP sendet Beacons mit Daten aus
  - Aktiv: Der Client fragt beim AP nach
  - Wenn ein Client wechseln möchte, wäre er solange offline bis er ein Beacon erhalten würde. Durch aktives nachfragen kann die „Hand off“Zeit reduziert werden.
- RTS/CTS (Ready to send / clear to send)
- Bluetooth
  - Musik, Staubsauger, Eingaben, Smartwatches, Datenübertragung,...
  - Kopfhörer Datenübertragung
    - Bits pro Sekunde = Samplerate * Samplebreite * Kanäle
    - 2*48 kHz (Nyquist) - Audio-CD: 44,1 kHz
    - 16 Bit
    - 2 Kanäle
    - $\approx$ 140/150 kBit/s

Literatur

[1] Janet Abbate. *Inventing the internet*. MIT press, 2000.

[2] Paul Baran et al. On distributed communications. *Volumes I-XI, RAND Corporation Research Documents, August*, pages 637–648, 1964.

[3] S Bellovin. The security flag in the ipv4 header. Technical report, RFC 3514, 2003. https://tools.ietf.org/html/rfc3514.

[4] Tim Berners-Lee, Roy Fielding, and Larry Masinter. Rfc 2396: Uniform resource identifiers (uri): Generic syntax, august 1998. *Status: Draft Standard*. http://tools.ietf.org/html/rfc2396.

[5] Stuart Cheshire, B Aboba, and E Guttman. Rfc 3927: Dynamic configuration of ipv4 link-local addresses. *IETF standard*, 2005. https://tools.ietf.org/html/rfc3927.

[6] T Clausen and P Jacquet. Rfc 3626. *Optimized link state routing protocol (OLSR)*, 2003. http://tools.ietf.org/html/rfc3626.

[7] Leslie Daigle. Whois protocol specification. 2004. http://tools.ietf.org/html/rfc3912.

[8] Stephen E Deering. Internet protocol, version 6 (ipv6) specification. 1998. https://tools.ietf.org/html/rfc2460.

[9] H Eidnes and G de Groot. P. vixie,"classless in-addr. arpa delegation. Technical report, BCP 20, RFC 2317, March, 1998. http://tools.ietf.org/html/rfc2317.

[10] Kevin R Fall and W Richard Stevens. *TCP/IP illustrated, volume 1: The protocols*. addison-Wesley, 2011. Verfügbar auf http://tinyurl.com/tcp-ip-illustrated (aus Uni-Netz).

[11] P Faltstrom, P Hoffman, and A Costello. Rfc 3490: internationalizing domain names in applications (idna). *Network Working Group, IETF*, 2003. http://tools.ietf.org/html/rfc3490.

[12] Adrian Farrel. *The Internet and its protocols: a comparative approach*. Morgan Kaufmann, 2004.

[13] R Fielding, J Gettys, J Mogul, H Frystyk, L Masinter, P Leach, and T Berners-Lee. Rfc 2616. *Hypertext Transfer Protocol–HTTP/1.1*, 2(1):2–2, 1999. http://tools.ietf.org/html/rfc2616.

[14] Ross Finlayson, Timothy Mann, Jeffrey C Mogul, and Marvin Theimer. A reverse address resolution protocol. Technical report, 1984. https://tools.ietf.org/html/rfc903.

[15] Vince Fuller, Tony Li, Jessica Yu, and Kannan Varadhan. Rfc 1519: Classless inter-domain routing (cidr): an address assignment and aggregation strategy, 1993. https://tools.ietf.org/html/rfc1519.

[16] Network Working Group et al. Rfc 1933, 1996. http://tools.ietf.org/html/rfc1933.

[17] C Hedrick, " Network Working Group Routing information protocol, et al. Rfc 1058. 1988. https://tools.ietf.org/html/rfc1058.

[18] Paul Hoffman and Scott Bradner. Defining the ietf. 2002. http://www.ietf.org/rfc/rfc3233.txt.

[19] Van Jacobson, Robert Braden, Dave Borman, M Satyanarayanan, JJ Kistler, LB Mummert, and MR Ebling. Rfc 1323: Tcp extensions for high performance, 1992. https://tools.ietf.org/html/rfc1323.

[20] Rohit Khare and Scott Lawrence. Rfc 2817: upgrading to tls within http/1.1, 2000. http://tools.ietf.org/html/rfc2817.

[21] Ed Krol. Hitchhikers guide to the internet. 1989. http://www.ietf.org/rfc/rfc1118.txt.

[22] Cricket Liu and Paul Albitz. *DNS and Bind*. Ö'Reilly Media, Inc.", 2006.

[23] G Malkin. Rfc 2453: Rip version 2. *Request for Comments*, 2453, 1998. https://tools.ietf.org/html/rfc2453.

[24] G Malkin and R Minnear. rfc 2080: Ripng for ipv6, 1997. https://tools.ietf.org/html/rfc2080.

[25] D Mills, J Martin, J Burbank, and W Kasch. Rfc 5905: Network time protocol version 4: Protocol and algorithms specification. *Internet Engineering Task Force*, 2010. http://tools.ietf.org/html/rfc5905.

[26] David L Mills. Rfc 1305: Network time protocol (version 3) specification. *Implementation and Analysis*, 1992. http://tools.ietf.org/html/rfc1305.

[27] David L Mills. Simple network time protocol (sntp) version 4 for ipv4, ipv6 and osi. 2006. http://tools.ietf.org/html/rfc4330.

[28] Paul Mockapetris. Rfc 1034: Domain names: concepts and facilities (november 1987). *Status: Standard*, 2003. https://tools.ietf.org/html/rfc1034.

[29] Paul Mockapetris. Rfc 1035—domain names—implementation and specification, november 1987. *URL http://www. ietf. org/rfc/rfc1035. txt*, 2004. https://tools.ietf.org/html/rfc1035.

[30] Jeffrey Clifford Mogul and Jon Postel. Internet standard subnetting procedure. Technical report, 1985. http://tools.ietf.org/html/rfc950.

[31] I Nazar. The hyper text coffee pot control protocol for tea efflux appliances (htcpcp-tea). 2014. http://www.ietf.org/rfc/rfc2324.txt.

[32] Charles Perkins, Elizabeth Belding-Royer, Samir Das, et al. Rfc 3561-ad hoc on-demand distance vector (aodv) routing. *Internet RFCs*, pages 1–38, 2003. http://tools.ietf.org/html/rfc3561.

[33] David C Plummer. Rfc 826: An ethernet address resolution protocol. *InterNet Network Working Group*, 1982. http://tools.ietf.org/html/rfc826.

[34] Jon Postel. Rfc 768: User datagram protocol, august 1980. *Status: Standard*, 1980. https://tools.ietf.org/html/rfc768.

[35] Jon Postel. Rfc 793: Transmission control protocol, september 1981. *Status: Standard*, 88, 2003. https://tools.ietf.org/html/rfc793.

[36] Jon Postel et al. Rfc 791: Internet protocol. 1981. http://tools.ietf.org/html/rfc791.

[37] Jon Postel et al. Rfc 792: Internet control message protocol. *InterNet Network Working Group*, 1981. http://tools.ietf.org/html/rfc792.

[38] Jarno Rajahalme, Shane Amante, Sheng Jiang, and Brian Carpenter. Ipv6 flow label specification. 2011. https://tools.ietf.org/html/rfc6437.

[39] Jarno Rajahalme, Alex Conta, Brian E Carpenter, and Steve E Deering. Rfc 3697: Ipv6 flow label specification, mar 2004. https://tools.ietf.org/html/rfc3697.

[40] Y Rekhter, T Li, and S Hares. Rfc 4271: Border gateway protocol 4, 2006. http://tools.ietf.org/html/rfc4271.

[41] M Slavitch. Definitions of managed objects for drip-type heated beverage hardware devices using smiv2. Technical report, 1998. http://www.ietf.org/rfc/rfc2325.txt.

[42] K Sollins. The tftp protocol (revision 2). 1992. http://tools.ietf.org/html/rfc1350.

[43] Andrew S Tanenbaum. Computer networks, 4-th edition. *ed: Prentice Hall*, 2003.

[44] wissen.de. Timeout. *Lexikon*. http://www.wissen.de/lexikon/timeout-informatik.

[45] Robert H Zakon. Hobbes' internet timeline. 1997. http://www.ietf.org/rfc/rfc2235.txt.

Glossar

ACE ASCII Compatible Encoding.

ARPA Advanced Research Projects Agency.

ATDN AOL Transit Data Network.

ATM Asynchronous Transfer Mode.

ccTLD Country Code Top Level Domain.

CFI Canonical Formati Indicator.

CIDR Classless Inter-Domain Routing.

CRC Cyclic Redundancy Check.

CSMA/CD Carrier Sense Multiple Acces / Collision Detection.

DE-CIX German Commercial Internet Exchange.

DENIC Deutsches Network Information Center.

DHCP Dynamic Host Configuration Protocol.

DNS Domain Name System.

FDD Frequency Divsion Duplex.

FTP File Transfer Protocol.

gTLD Generic Top Level Domain.

GX Global Crossing.

HTTP Hypertext Transfer Protocol.

ICANN Internet Corporation for Assigned Names and Numbers.

IETF Internet Engineering Task Force.

MAC Medium Access Control.

NAT Network Address Translation.

NIC Name Information Center.

NTP Network Time Protocol.

RIPE Réseaux IP Européens.

RIPE NCC RIPE Coordination Centre.

SFD Start Frame Delimiter.

SMTP Simple Mail Transfer Protocol.

SNMP Simple Network Management Protocol.

SNTP Simple Network Time Protocol.

TCP Transmission Control Protocol.

TDD Time Division Duplex.

TFTP Trivial File Transfer Protocol.

TLD Top Level Domain.

TLS Transport Layer Security.

ToS Type of Service.

TPID Tag Protocol Identifier.

TTL Time to Live.

UDP User Datagram Protocol.

URI Uniform Resource Identifier.

URL Uniform Resource Locator.

URN Uniform Resource Name.

VCI Virtual Channel Identifer.

VID VLAN Identifier.

VPI Virtual Path Identifier.

WWW World Wide Web.