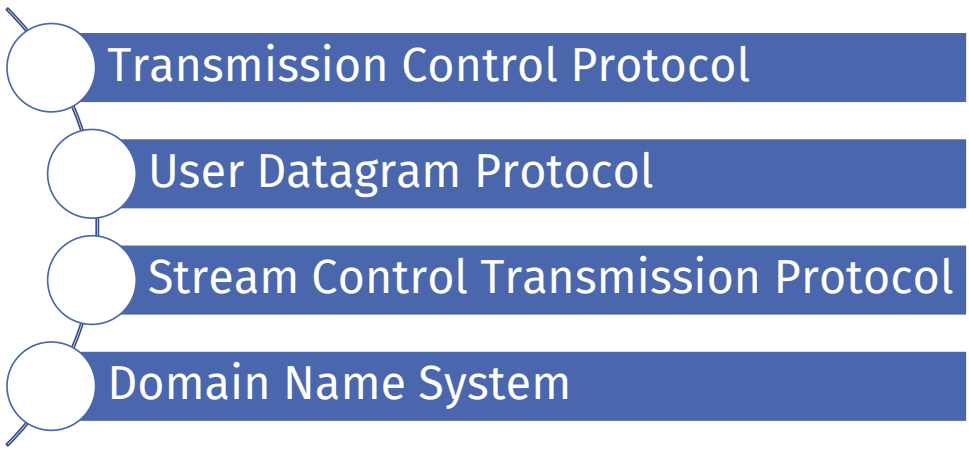
Die Internet-Protokollwelt

5. DIE TRANSPORTSCHICHT IM INTERNET

166

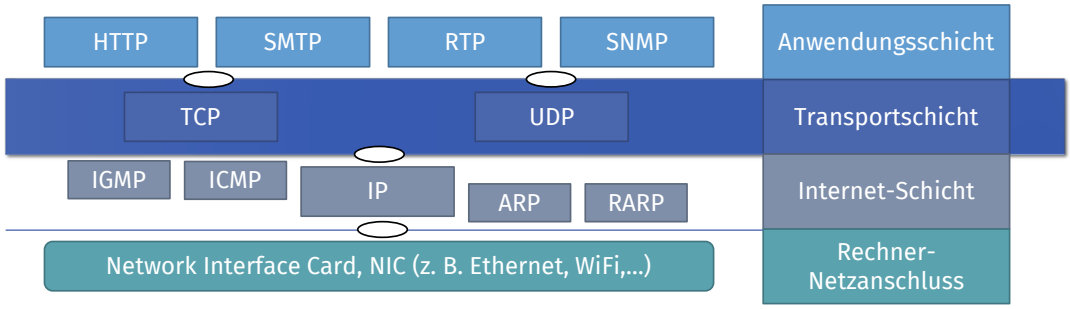
Übersicht



167

Die Internet-Protokollfamilie: Einordnung

TCP/IP häufig Synonym für die gesamte Protokollfamilie im Internet
Einordnung der Protokolle in das Schichtenmodell:



Transmission Control Protocol, TCP [RFC 793]

Verbindungsverwaltung
- Verbindungsaufbau zwischen zwei „Sockets" (entspricht CEP im T-SAP)
- Datentransfer über virtuelle Transportschichtverbindung (über verbindungslosen Vermittlungsdienst)
- Gesicherter Verbindungsabbau (alle Daten müssen quittiert sein)

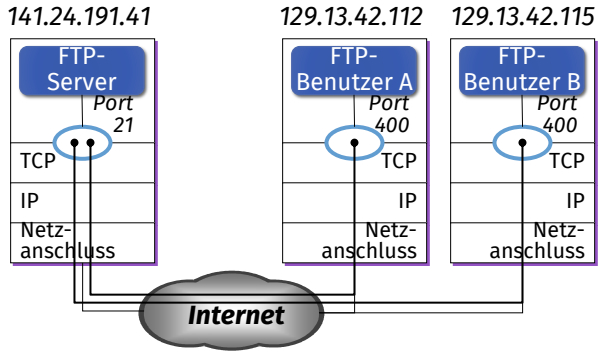
Multiplexen
- Mehrere Prozesse können gleichzeitig eine TCP-Instanz benutzen

Datenübertragung
- Vollduplex
- Reihenfolgetreue
- Flusskontrolle mit Fenstermechanismus
- Fehlerkontrolle durch Folgenummern (Sequenznummern), Prüfsumme, Quittung, Übertragungswiederholung, Rücksetzen

Fehleranzeige

TCP: Adressierung

- Identifikation von Anwendungen/Diensten über Ports
- Portnummern bis 1024 für häufig benutzte Dienste reserviert („*well-known ports*“, z. B. 20, 21 für FTP, 25 für SMTP, 80 für HTTP)
- *Socket* = IP-Adresse eines Rechners + Portnummer
- *Notation*: (IP-Adresse:Portnummer) → Internet-weit eindeutig
- *Beispiel* – FTP-Server der TU Ilmenau über Socket 141.24.191.41:21 erreichbar



TCP: fest vereinbarte Port-Nummern („well-known ports“)

Festgelegte Ports für viele Anwendungen:

- 13: Tageszeit
- 20: FTP Daten
- 25: SMTP (Simple Mail Transfer Protocol)
- 53: DNS (Domain Name Server)
- 80: HTTP (HyperText Transfer Protocol)
- 119: NNTP (Network News Transfer Protocol)

```
> telnet walapai 13
Trying 129.13.3.121...
Connected to walapai.
Escape character is '^]'.
Mon Aug  4 16:57:19 1997
Connection closed by foreign host
```

```
> telnet mailhost 25
Trying 129.13.3.161...
Connected to mailhost .
Escape character is '^]'.
220 mailhost ESMTP Sendmail 8.8.5/8.8.5;
Mon, 4 Aug 1997 17:02:51 +0200
HELP
214-This is Sendmail version 8.8.5
214-Topics:
214- HELO    EHLO    MAIL    RCPT    DATA
214- RSET    NOOP    QUIT    HELP    VRFY
214- EXPN    VERB    ETRN    DSN
214-For more info use "HELP <topic>".
...
214 End of HELP info
```

TCP: Verbindungsaufbau

Aufbau einer TCP-Verbindung
- **aktiv** (connect) oder
- **passiv** (listen/accept)

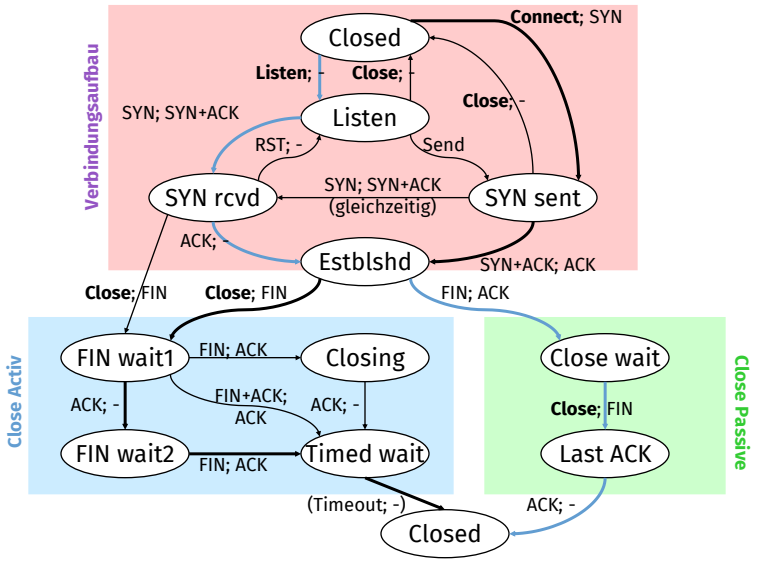
Aktiver Modus: Anforderung einer TCP-Verbindung mit dem spezifizierten Socket

Passiver Modus: Warten eines TCP-Benutzers auf eine eingehende Verbindung
- Spezifikation eines speziellen Sockets, von dem eine eingehende Verbindung erwartet wird (*fully specified passive open*) oder
- alle Verbindungen annehmen (*unspecified passive open*)
- Geht ein Verbindungsaufbauwunsch ein, wird ein neuer Socket erzeugt, der dann als Verbindungsendpunkt dient

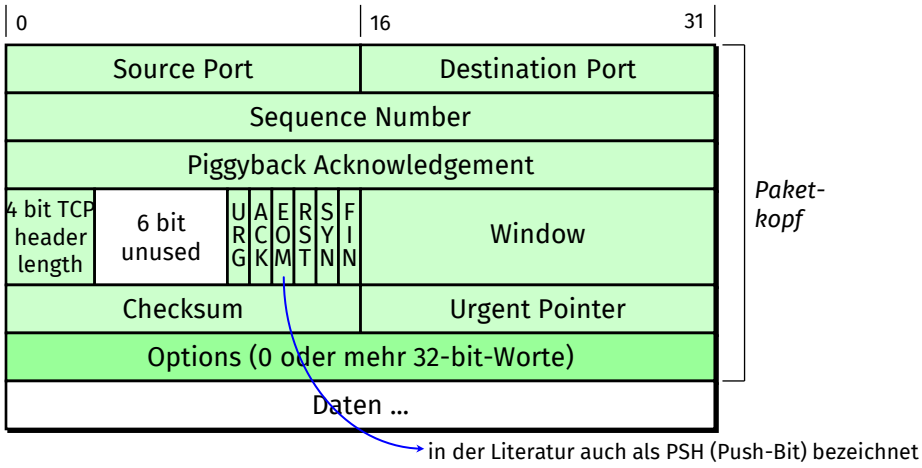
Anmerkung: Die Verbindung wird von den TCP-Instanzen ohne weiteres Eingreifen der Dienstbenutzer aufgebaut (es existiert z. B. kein Primitiv, das T-CONNECT.Rsp entspricht)

TCP: Verbindungs-management

Nach RFC 0793



TCP-Paketformat: Aufbau



174

TCP-Paketformat

Source und **Destination Port** – Endpunkte der TCP-Verbindung: well-known oder beliebige freie Portnummen
Sequence Number – *Byte*-Folgenummer
Piggyback Acknowledgement – Huckepackquittierung: nächste erwartete Folgenummer
TCP Header Length – Anzahl der 32-bit-Wörter im Paketkopf

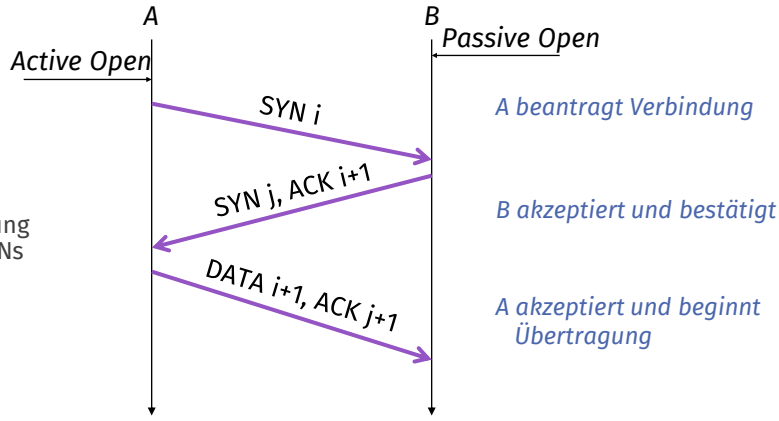
Flags
- **URG** – auf 1 gesetzt, falls der Urgent Pointer verwendet wird
- **SYN** – ausschließlich beim Verbindungsaufbau verwendet
- **ACK** – Gültigkeit des Acknowledgement-Feldes
- **FIN** – Verbindungsabbau: gibt an, dass der Sender keine Daten mehr senden möchte
- **RST** – zum Rücksetzen einer Verbindung (z. B. bei unerwarteten Paketnummern)
- **EOM** (bzw. **PSH**) – Ende einer Nachricht: Aufforderung zum Ausliefern der Daten an die Anwendungsschichtinstanz

Window – kreditbasierte Flusskontrolle: Anzahl der Bytes, die nach letztem bestätigten Byte gesendet werden dürfen
Checksum – Prüfsumme über Paketkopf und Daten
Urgent Pointer – Relativer Zeiger auf wichtige Daten
Options-Feld – Optionen variabler Länge

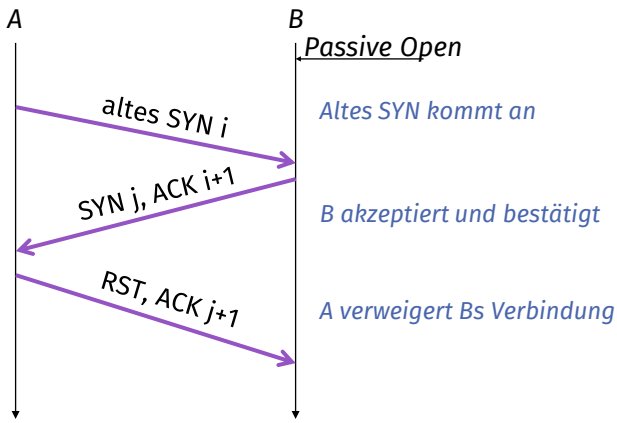
175

TCP-Verbindungsaufbau im Detail

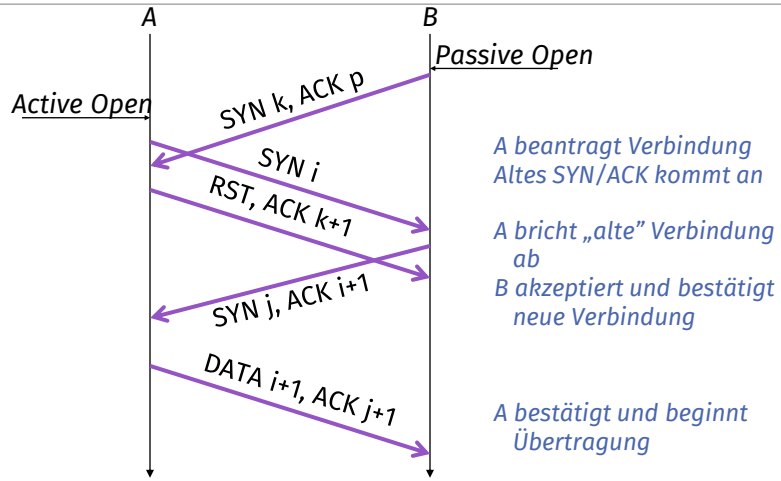
- **3-Wege-Handshake**
  - Beide SYNs müssen bestätigt sein
  - Für den Empfänger gilt die Verbindung erst nach der Bestätigung seines SYNs als aufgebaut



TCP-Verbindungsaufbau: Verwaistes SYN



TCP-Verbindungsaufbau: Verspätetes SYN/ACK



TCP: Duplikaterkennung

Datenpakete zur Duplikaterkennung durchnummeriert

Unter Umständen mehrfache Bestätigung eines Datenpakets (aufgrund von Huckepack-Quittierung)

➔Kein Anzeichen für Duplikate

Ausreichender Sequenznummernbereich, sodass zwei Pakete mit der gleichen Sequenznummer zeitlich genügend weit auseinander liegen

Allerdings:

- Datenpakete können Verbindungsabbau überstehen und irrtümlich einer neuen Verbindung zugeordnet werden
- Durch einen Systemzusammenbruch kann die Paketnummerierung verloren gehen

Problemvermeidung:

- Uhr-unterstützte Sequenznummer (*Clock-based initial sequence number*)
- Sendeverzögerung (*Quiet Time*)

TCP: Fenster-management

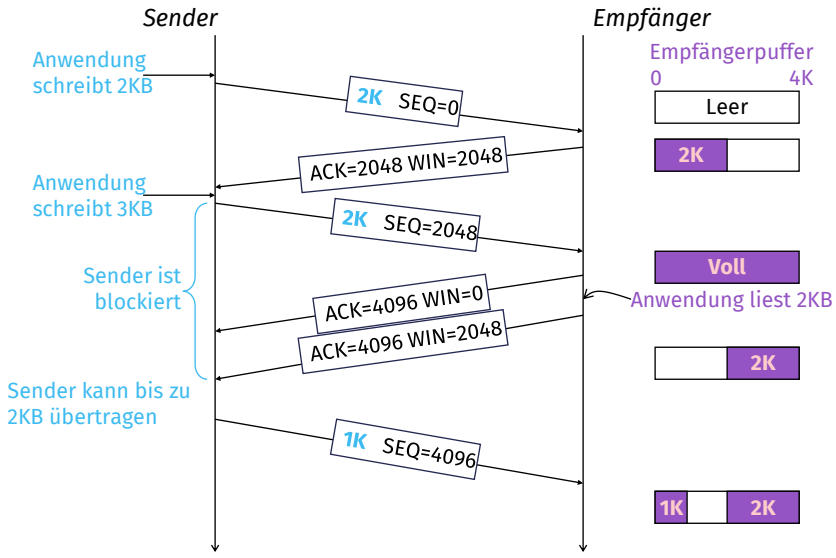
Flusskontrolle:

Sender darf nicht mehr Daten schicken, als der Empfänger verarbeiten kann

Regelung des Datenflusses zwischen den Endsystemen

Fenstermechanismus mit Kreditvergabe:

- Bestätigung der Daten mit niedriger Bytefolgenummer durch *ACK*-Feld im Paketkopf
- Kredit = empfangbare Menge an Bytes im *Window*-Feld



180

TCP: Staukontrolle [RFC 5681]

Staukontrolle → Umgang mit Stausituationen im Netz, d. h. in den Routern

Problem „*Congestion Collapse*":

- Stau in Zwischensystemen → Timeout → Paketwiederholungen → Verstärkung der Stausituation

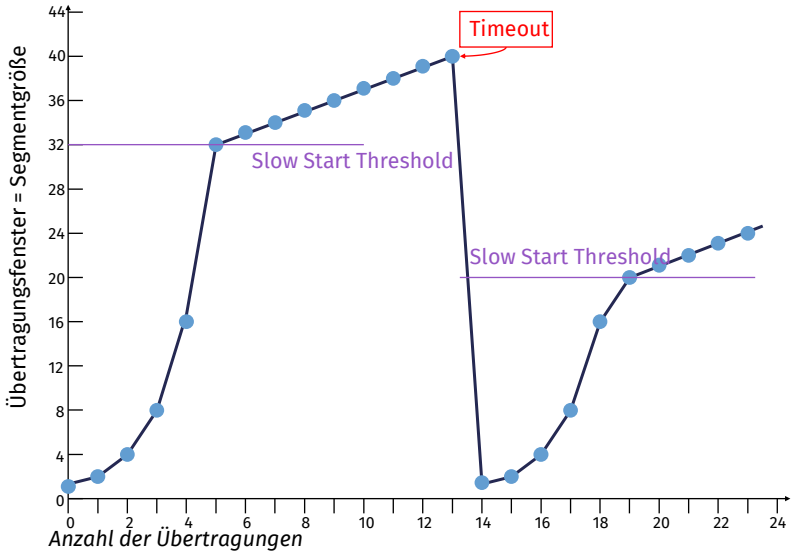
TCP: „*Slow Start*" und „*Multiplicative Decrease*"

- Zu Beginn schrittweises Eruieren der Netzkapazität mit Verdopplung der gesendeten Segmentgröße bei erfolgreichem Senden bis zu einem Schwellwert (*Slow Start Threshold*), danach lineare Steigerung der Senderate (*Congestion Avoidance*)
- Bei zu spätem ACK Verdacht auf Stau
  - *Slow Start Threshold* := aktuelle Segmentgröße / 2 (*Multiplicative Decrease*).
  - Weiter mit *Slow Start* (TCP Tahoe) oder *Congestion Avoidance* (TCP Reno)

181

TCP: Staukontrolle am Beispiel

Unter Annahme, dass Window-Size des Empfängers immer ausreichend groß

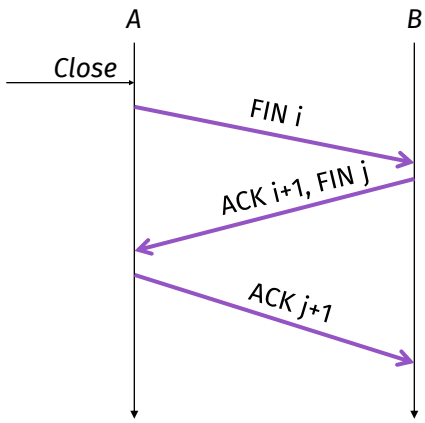


182

TCP: Verbindungsabbau

Bestätigter und nummerierter Verbindungsabbau

- Erkennung noch ausstehender Datenpakete
- Vollzug des Verbindungsabbaus erst mit Eintreffen des letzten Datenpakets



A sendet Verbindungsabbau-Anfrage

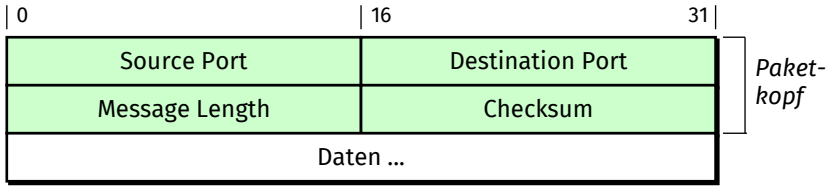
B bestätigt und sendet eigenes Verbindungsabbau-Signal

A bestätigt den Verbindungs-abbau von B. Ist dieses Signa bei B angekommen, gilt die Verbindung als abgebaut.

183

User Datagram Protocol, UDP [RFC 768]

- Unzuverlässig, verbindungslos, einfacher und schneller als TCP
- Demultiplexing der empfangenen Pakete basierend auf Port-Nummer
- Optionale Prüfsumme



- Wiederum „well-known" Ports:
  - 13: Daytime
  - 53: Domain Name Server
  - 123: Network Time Protocol
- UDP vor allem für Multimedia- oder Echtzeit-Anwendungen geeignet

Stream Control Transmission Protocol, SCTP [RFC 4960]

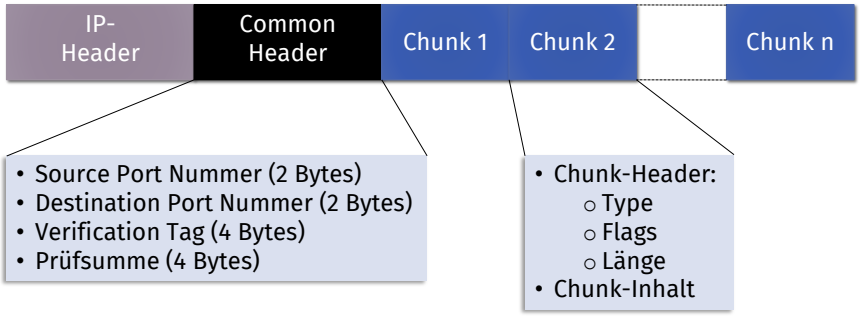
SCTP ist als Kompromiss zwischen TCP und UDP entwickelt worden:

- Verbindungsorientiert: SCTP-Assoziation
- Nachrichtenbasiert
- Ermöglicht Flusssteuerung
- Segmentieren und Blocken

SCTP-Assoziation:

- Zusammengesetzt aus mehreren Streams
- Ein Stream entspricht einer unidirektionalen Verbindung

SCTP-Paketaufbau

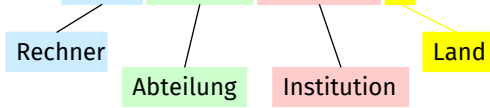


Anwendungsnahe Adressierung im Internet

Adressierung über logische Namen
- Einfacher zu merken
- Dienste einfacher auf andere Rechner übertragbar

Aufbau eines logischen Namens
- Weltweit eindeutig
- Hierarchische Struktur
- Gliederung in Domänen

Beispiel
- ikmcip1.e-technik.tu-ilmenau.de

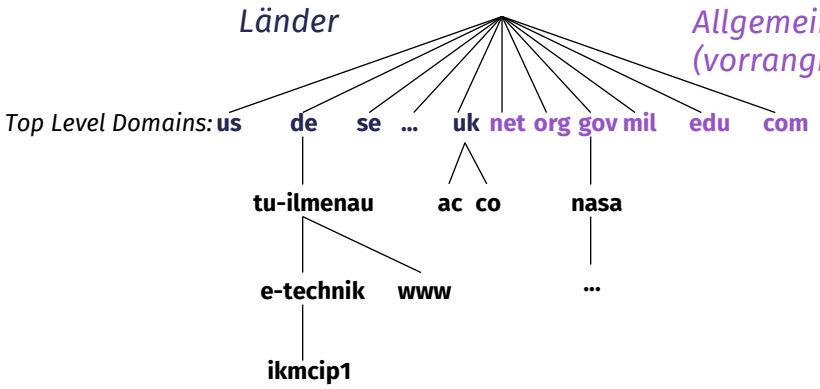


Benötigt:
- Abbildung: logischer Name → IP-Adresse
- Ursprünglich: Datei (hosts.txt), die jede Nacht vom Server geladen wurde
- Problem: steigende Anzahl der Namen ließ zentrale Datei nicht mehr zu

Domain Name System, DNS

[RFC 1591]

Namensraum in Zonen aufgeteilt:



Geplante, z. T. bereits realisierte weitere Zonen (Top Level Domains):

- *app* – Handy-Apps
- *farm* – Landwirtschaft
- *info* – Information
- *red* – für Leute, die rot mögen
- *singles* – für Online-Dating
- *vodka* – wie der Name schon sagt
- *work* – ???

DNS – Resource Records

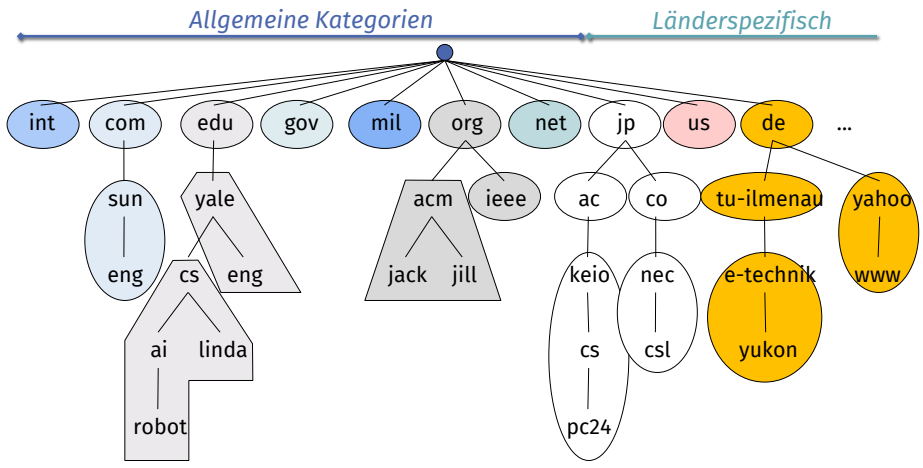
Fünf-Tupel, das einzelne Ressourcen näher beschreibt:

- Domain_name
- Time_to_live
- Class
- Type
  - A (IP-Adresse des Rechners)
  - MX (Mail Exchange)
  - HINFO (CPU und Betriebssystem des Rechners in ASCII)
  - CNAME (Canonical Name)
  - ...
- Value

DNS – Beispieldatenbank

Domain_name	Time_to_live	Class	Type	Value
cs.vu.nl	86400	IN	TXT	„Faculteit Wiskunde en Informatica"
cs.vu.nl	86400	IN	TXT	„Vrije Universiteit Amsterdam"
cs.vu.nl	86400	IN	MX	1. zephyr.cs.vu.nl
cs.vu.nl	86400	IN	MX	2. top.cs.vu.nl
flits.cs.vu.nl	86400	IN	HINFO	Sun Unix
flits.cs.vu.nl	86400	IN	A	130.37.16.112
flits.cs.vu.nl	86400	IN	A	192.31.231.165
flits.cs.vu.nl	86400	IN	MX	1. flits.cs.vu.nl
flits.cs.vu.nl	86400	IN	MX	2. zephyr.cs.vu.nl
www.cs.vu.nl	86400	IN	CNAME	star.cs.vu.nl
ftp.cs.vu.nl	86400	IN	CNAME	zephyr.cs.vu.nl
laserjet		IN	A	192.31.231.216
		IN	HINFO	„HP Laserjet IIISi" Proprietary

DNS – Name Servers

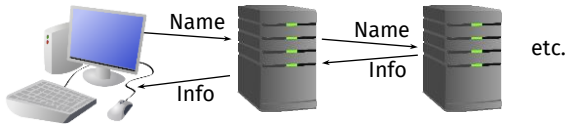


DNS – Anfragen an Name Server

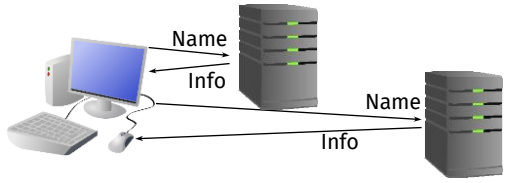
Je Zone ein primärer und beliebig weitere sekundäre Nameserver

Rekursive oder nicht-rekursive Beantwortung von Anfragen:

- *rekursiv:*



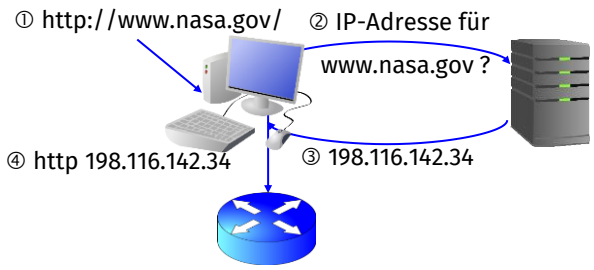
- *nicht-rekursiv:*



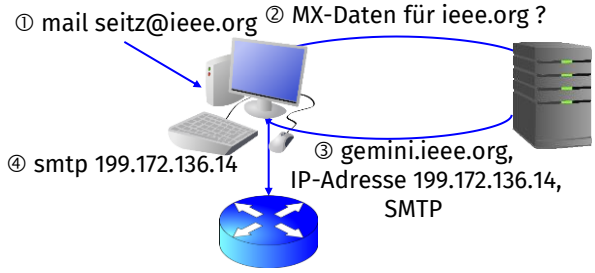
192

DNS – Beispiele

AUFLÖSUNG DER ADRESSE EINES WEB-SERVERS:



AUFLÖSUNG DER ADRESSE EINES MAIL-SERVERS:



193

Literatur

COMER, Douglas E. (2000): *Computernetzwerke und Internets*. München: Pearson Studium.

COMER, Douglas E. (2011): *TCP/IP - Studienausgabe. Konzepte, Protokolle, Architekturen*. Heidelberg: mitp.

KUROSE, James F.; ROSS, Keith W. (2017): *Computer Networking. A Top-Down Approach*. Seventh edition. Boston: Pearson.

PETERSON, Larry L.; DAVIE, Bruce S. (2012): *Computer Networks – A Systems Approach*. 5th edition. Amsterdam, Boston: Morgan Kaufmann.

STEVENS, W. Richard (2004): *TCP-IP. Der Klassiker: Protokollanalysen, Aufgaben und Lösungen*. 1. Auflage. Bonn: Hüthig.

TANENBAUM, Andrew S.; WETHERALL, David J. (2012): *Computernetzwerke*. 5., aktualisierte Auflage. München: Pearson (It Informatik).

Requests for Comments (RFC)

POSTEL, Jon (1980): *User Datagram Protocol*. Internet Engineering Task Force (IETF) (Request for Comments, 768).

POSTEL, Jon (1981): *Transmission Control Protocol*. DARPA Internet Program Protocol Specification. Internet Engineering Task Force (IETF) (Request for Comments, 793).

POSTEL, Jon (1994): *Domain Name System Structure and Delegation*. Internet Engineering Task Force (IETF) (Request for Comments, 1591).

STEWART, Randall R. (2007): *Stream Control Transmission Protocol*. Internet Engineering Task Force (IETF) (Request for Comments, 4960).

ALLMAN, Mark; Paxson, Vern; Blanton, Ethan (2009): *TCP Congestion Control*. Internet Engineering Task Force (IETF) (Request for Comments, 5681).

EASTLAKE, Donald E., 3rd (2013): *Domain Name System (DNS) IANA Considerations*. Internet Engineering Task Force (IETF) (Request for Comments, 6895).