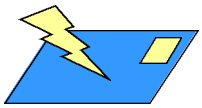
Internet: Dienste und Sicherheit

Beitrag von Bernhard Gross
für die Liste TV-Technik

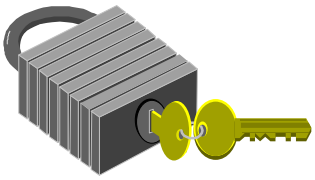
Domain Name Service



Email, News

WWW

http, URL, HTML



Security in Netzwerken

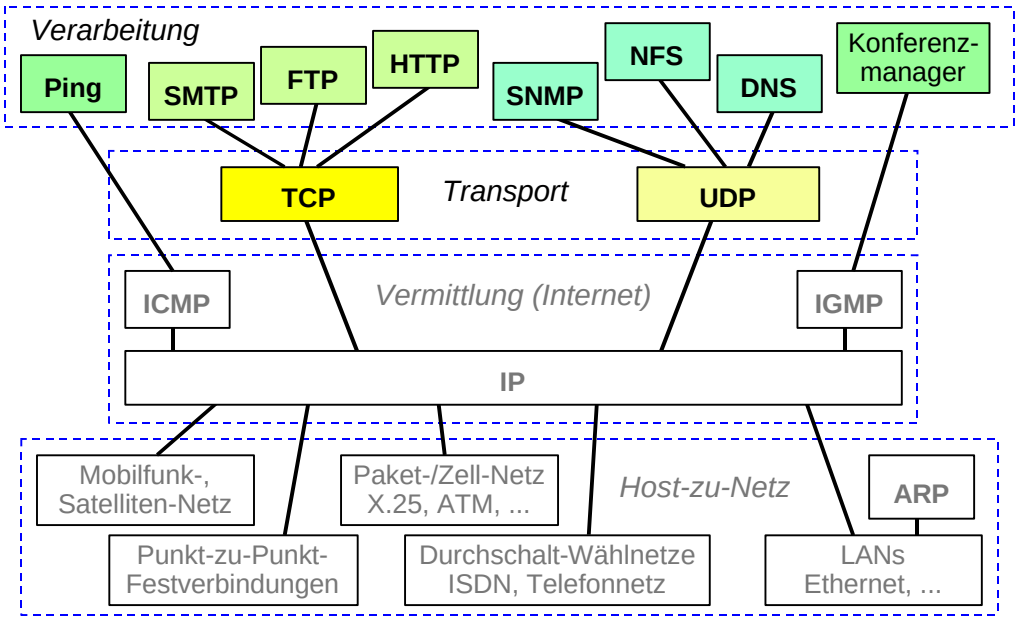
Gliederung

1. DNS-Adressierung mit Klartextnamen
2. Technik des Email-Dienstes (MIME, SMTP, POP3)
3. Das World Wide Web
4. Die Web Beschreibungssprache HTML
5. Sicherheit im Internet
6. Diskussionsforen im Internet (Listen, News)

Begleitbuch:
Tanenbaum: Computer Networks. Prentice Hall

Mein besonderer Dank gilt Herrn Prof. Dr. Werner Winzerling, FH Fulda für die freundliche Überlassung eines Powerpoint-Foliensatzes zum Thema.

Anwendungen im Internet mit Protokollstapeln



Übersicht der Internet Anwendungen

Folgende (wichtige) Verarbeitungsdienste (-protokolle) werden hier näher vorgestellt:

DNS (Domain Name Service)

Namenbildung und -verwendung im Internet.

SMTP (Simple Mail Transfer Protocol)

Das E-Mail-Protokoll war ursprünglich nur eine Art Dateitransfers.

NNTP (News Network Transfer Protocol)

Austausch von Nachrichtenartikeln.

HTTP (Hyper-Text Transfer Protocol)
HTML (Hyper-Text Markup Language)

World Wide Web Protokoll einschließlich der Web-Beschreibungssprache.

Adressierung im Internet

Jede Schicht hat üblicherweise ihr ***eigenes Adressierungsschema***.

Wichtigsten Adressen im Internet sind die ***IP-Adresse*** der ***Vermittlungsschicht***. Diese Adressen sind rein ***numerisch*** und dadurch nur schwer handhabbar.

Aus diesem Grund werden in der ***Verarbeitungsschicht*** die Adressen für Hosts, Mailboxen, und anderen Anwendungsressourcen nicht aus Binärzahlen gebildet, sondern es werden ***Namen*** aus ***ASCII-Zeichenketten*** verwendet; z.B.:

MMN-1

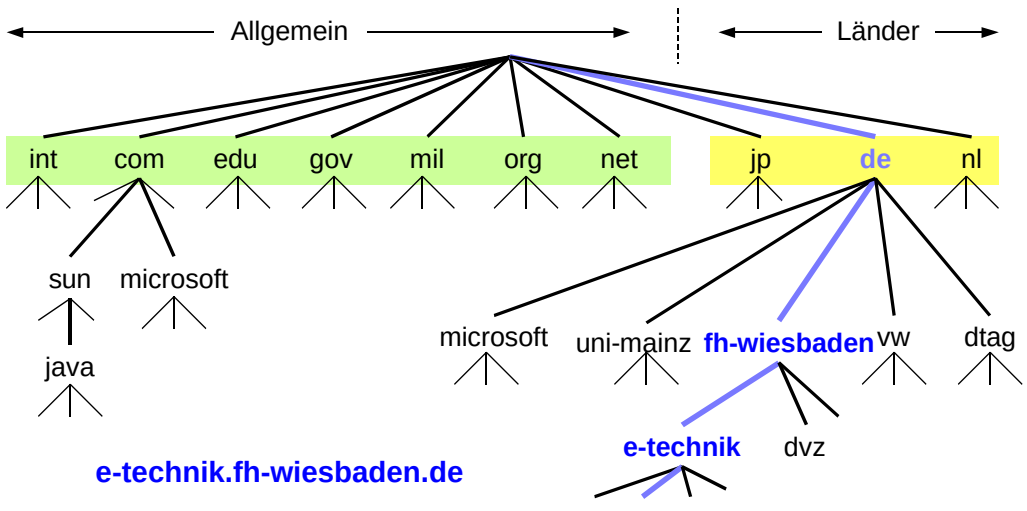
gross@e-technik.fh-wiesbaden.de

e-technik.fh-wiesbaden.de

Im Internet kann dazu mittels des ***DNS-Protokolls*** der Verarbeitungsschicht (Domain Name System) eine Abbildung von (ASCII)-Namen auf die numerischen IP-Netzadressen erfolgen.

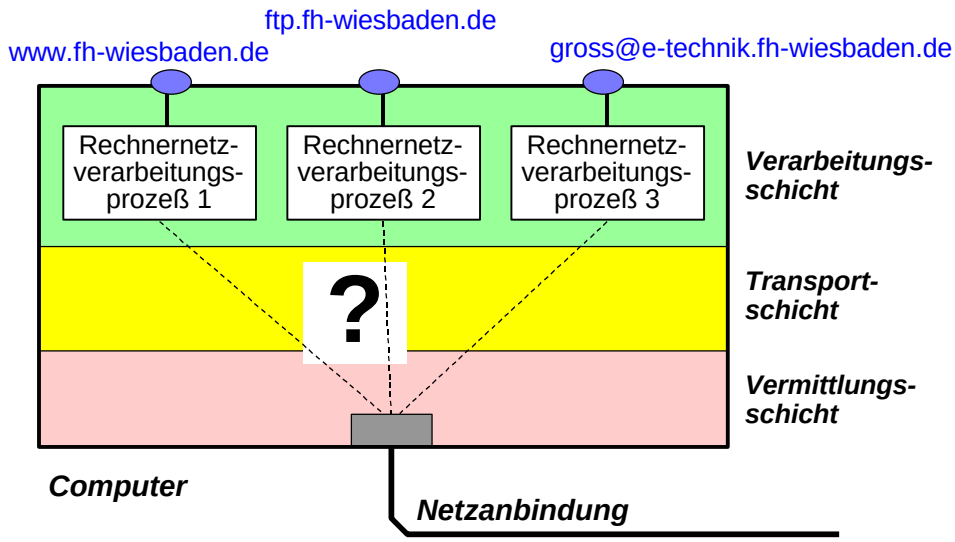
Folie nach Prof. Dr. W. Winzerling, FH Fulda

Teil der Internet-Domänen



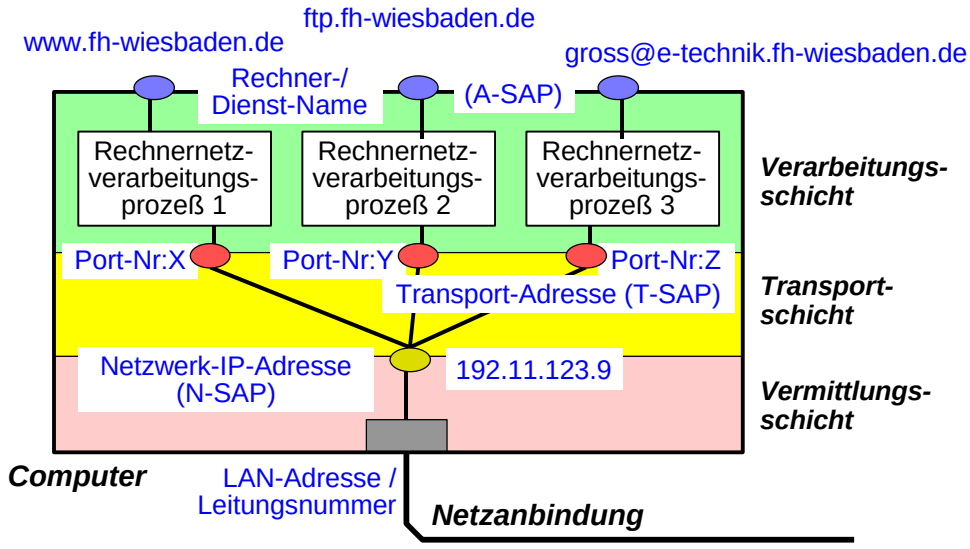
Folie nach Prof. Dr. W. Winzerling, FH Fulda

Adreßabbildung im Internet (1)



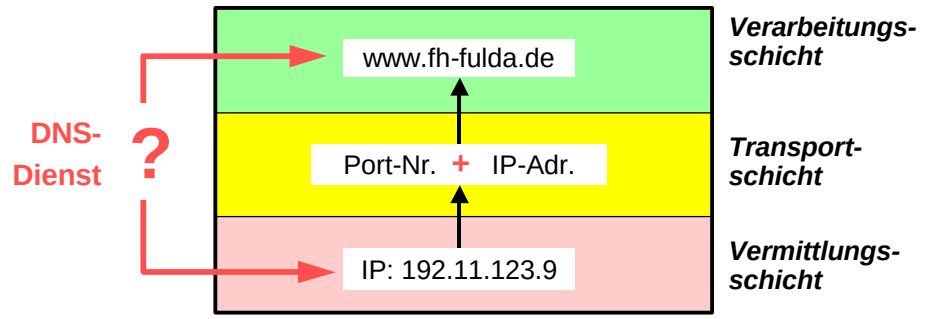
Folie nach Prof. Dr. W. Winzerling, FH Fulda

Adreßabbildung im Internet (2)



Folie nach Prof. Dr. W. Winzerling, FH Fulda

Adreßauflösung im Internet



Folie nach Prof. Dr. W. Winzerling, FH Fulda

DNS-Namenserver

Am ***Anfang*** des Internets, zu ARPA-Zeiten:

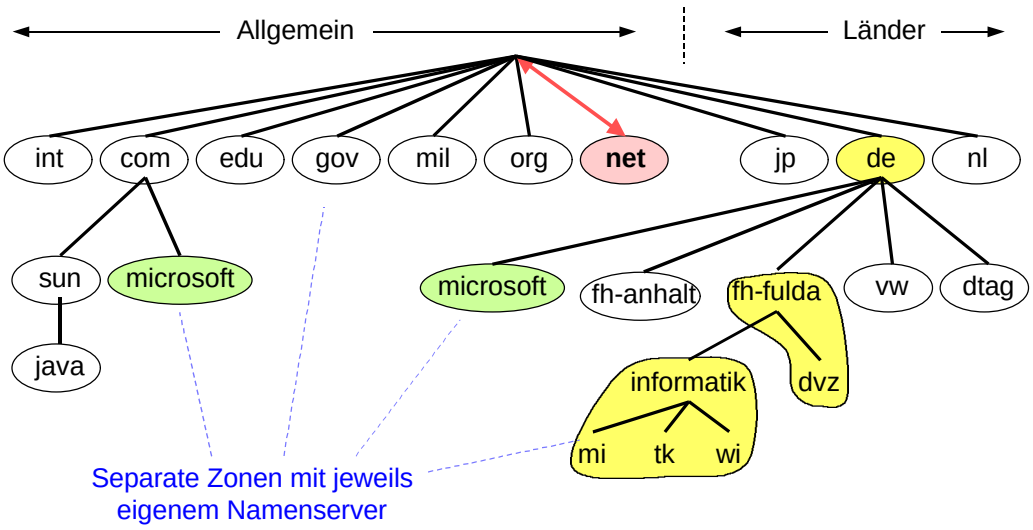
- Auf jedem Host befand sich die Datei ***host.txt***
- sie war überall gleich und enthielt die ASCII-Namen sowie die zugehörige IP-Adresse ***aller (!)*** Hosts im weltweiten Netz.

Heute existieren hierfür spezielle, verteilte ***DNS-Namenserver***:

- Antwortet auf eine ASCII-Namen-Anfrage mit der zugehörigen IP-Adresse;
- Ein weltweit einzelner DNS-Server könnte alle Adressen enthalten. Aus praktischen Gründen wird jedoch eine verteilte DNS-Datenbank weltweit auf viele dezentrale DNS-Servern verteilt.
- Dazu wird der DNS-Namenbereich in nicht-überlappende ***Zonen*** aufgeteilt.

Folie nach Prof. Dr. W. Winzerling, FH Fulda

In Zonen aufgeteilter DNS-Namenbereich



Folie nach Prof. Dr. W. Winzerling, FH Fulda

Zonen enthalten eigene DNS-Namenserver

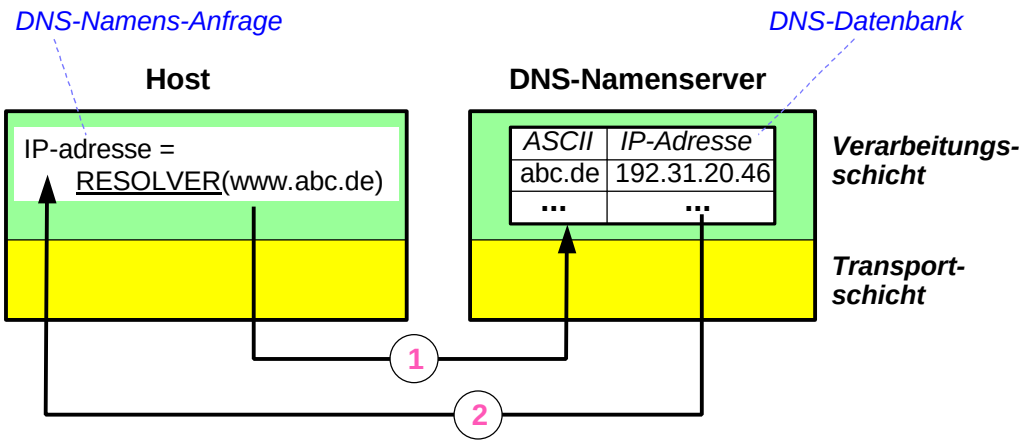
- Jede **Zone** enthält:
  Für seinen Teil des Baums einen Namenserver, der „autoritative“ Informationen über die Zone enthält, insbesondere die den ASCII-Namen zugeordneten IP-Adressen.

- **DNS-Namenserver** werden unterschieden in einen ***primären*** Namenserver und zur Verbesserung der Zuverlässigkeit einen oder mehrere ***sekundäre*** Namenserver, die ihre Informationen regelmäßig vom primären Namenservern erhalten.

- Die **Zonengrenzen** innerhalb einer Zone (unterhalb der aktuellen Domäne) wird vom ***Zonenverwalter*** festgelegt:
  Jede Zone muß einen eigenen Namenserver enthalten.

Folie nach Prof. Dr. W. Winzerling, FH Fulda

Namenauflösung mit dem DNS-Protokoll (1)



RESOLVER : Bibliotheksprozedur

Folie nach Prof. Dr. W. Winzerling, FH Fulda

Ablauf Namenauflösung

Jedem ***Client*** ist ein ***lokaler DNS-Namenserver*** (primärer und sekundärer) zugewiesen.

Liegt die gesuchte Domäne ***in der Zone*** des lokalen DNS-Servers, gibt dieser die gewünschte IP-Adresse zurück.

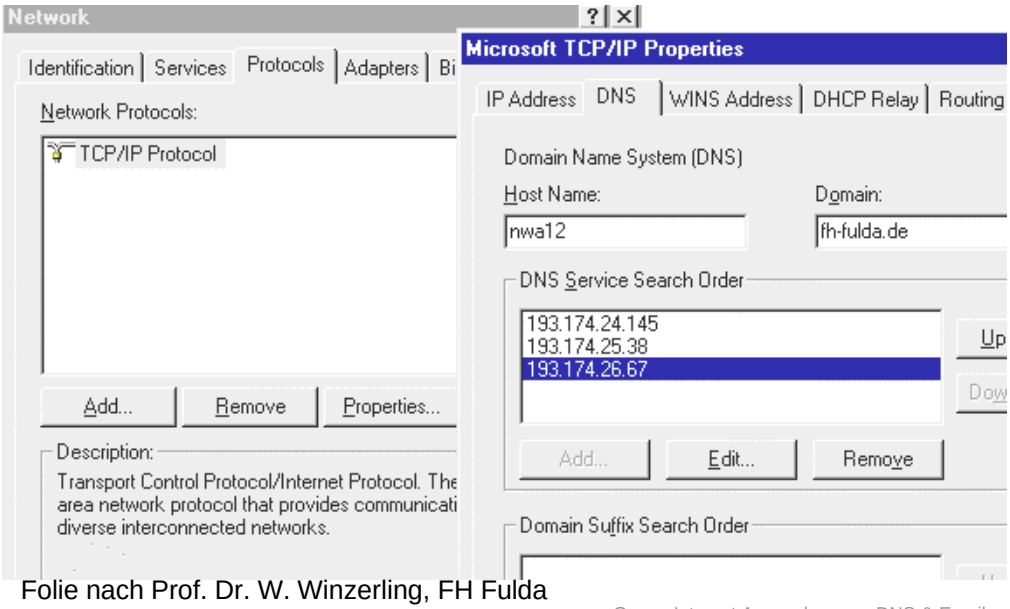
Liegt die gesuchte Domäne ***entfernt***, dann wendet sich der lokale DNS-Server an den DNS-Namenserver der obersten Ebene (DNS-Server der TOP-Level-Domäne). Diese stehen in der Datenbank ***xxx-server.net.***

Jeder DNS-Server muß alles über ***seine Kinder*** (die nächste darunter liegende Hierarchiestufe des Baums) wissen, aber nicht unbedingt alles über seine Enkel und Ur-Enkel usw. Deshalb wird eine Anfrage ggfl. ***Von oben nach unten*** durchgereicht.

Ein einmal aufgelöster ASCII-Name wird von den DNS-Servern für eine gewisse Zeit zwischengespeichert (***Caching***).

Folie nach Prof. Dr. W. Winzerling, FH Fulda

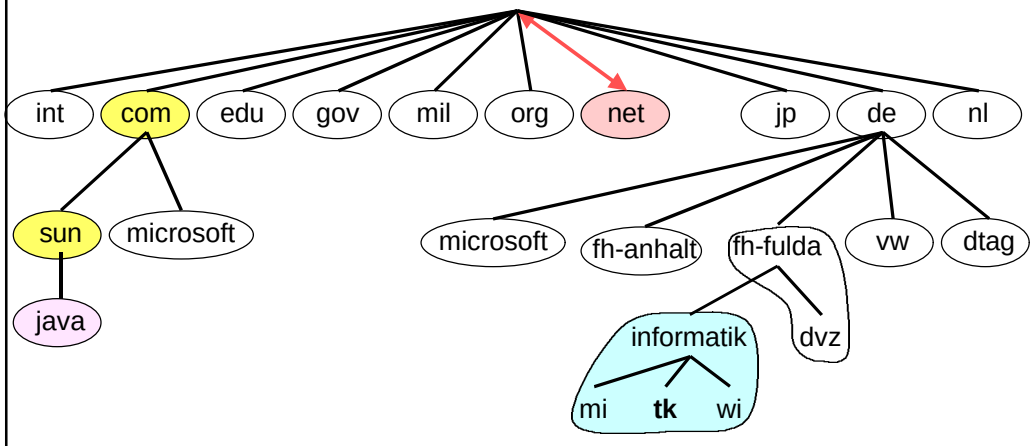
Eintrag des lokalen DNS-Server im Client



Folie nach Prof. Dr. W. Winzerling, FH Fulda

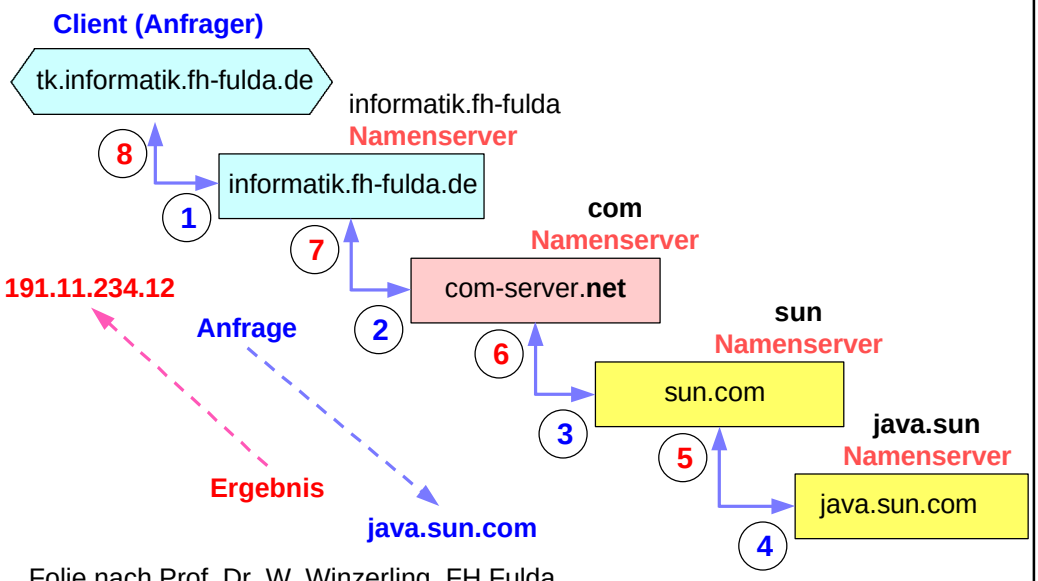
Beispiel einer Namenauflösung (1)

Der **Client** ***tk.informatik.fh-fulda.de*** sucht den **Server** ***java.sun.com***



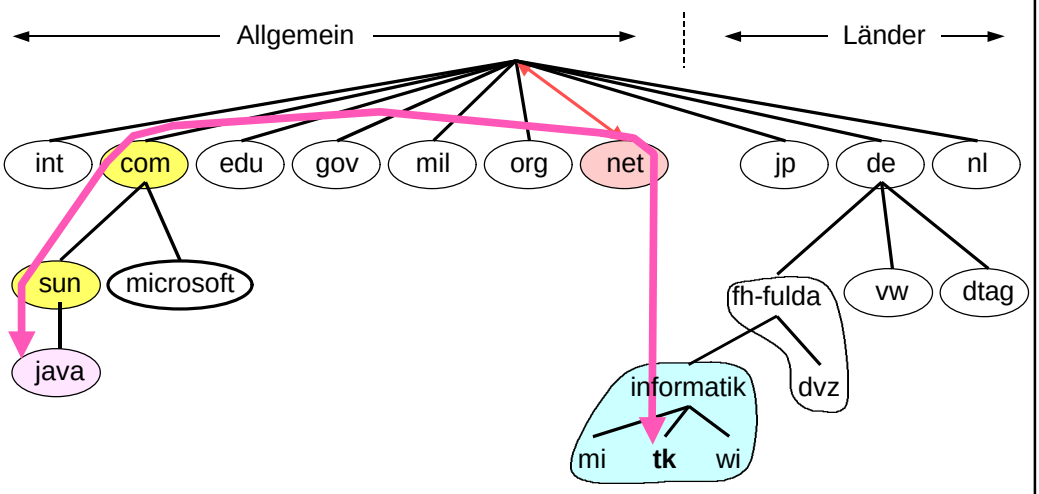
Folie nach Prof. Dr. W. Winzerling, FH Fulda

Beispiel einer Namenauflösung (2)



Folie nach Prof. Dr. W. Winzerling, FH Fulda

Beispiel einer Namenauflösung (3)



Folie nach Prof. Dr. W. Winzerling, FH Fulda

DNS - Zusammenfassung

Jede Rechnernetz-Softwareschicht verwendet i.a. ein eigenes Adressierungsschema.

Während beispielsweise in der **Vermittlungsschicht binäre Adressen** genutzt werden, wird in der **Verarbeitungsschicht** mit leichter zu merkende **Namen** gearbeitet.

Die Abbildung von Namen auf binäre Netzadressen erfolgt mittels **Namenserver**.

Im Internet wird dazu der **DNS-Dienst** (Domain Name Service) genutzt. Er ist als weltweit verteilte, dezentrale Datenbank organisiert.

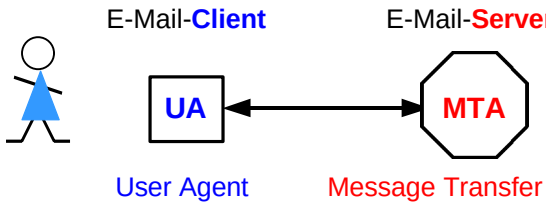
Um z. B. die IP-Adresse für einen speziellen Host aus dessen Namen zu ermitteln, werden entlang der hierarchischen Struktur des Namenbaums die zuständigen DNS-Namenserver befragt, bis ein Namenserver gefunden wird, in dessen Zone sich der gesuchte Host befindet.

Folie nach Prof. Dr. W. Winzerling, FH Fulda

E-Mail-Standards

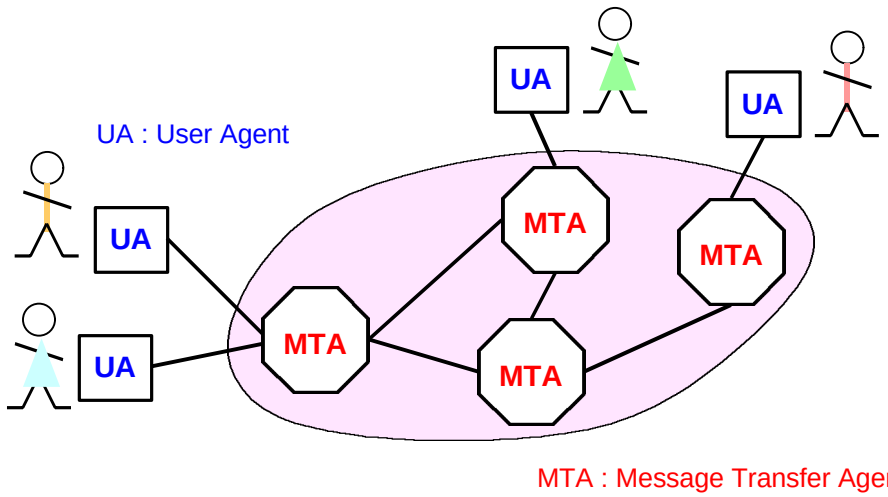
1. Empfehlung der CCITT (heute ITU) die für OSI übernommen wurde: **X.400**
   - schlecht ausgelegt
   - sehr komplex
   - nie zufriedenstellend implementiert
   - *abnehmende Bedeutung*
2. Internet-Empfehlung: **RFC 822** (u. a.)
   - (zu ?) einfach
   - aber es funktioniert **!**
   - *mit wachsender Bedeutung, der wichtigste E-Mail-Standard*
3. Verschiedene proprietäre Firmenstandards
   - *abnehmende Bedeutung*

Architektur eines E-Mail-Systems (1)

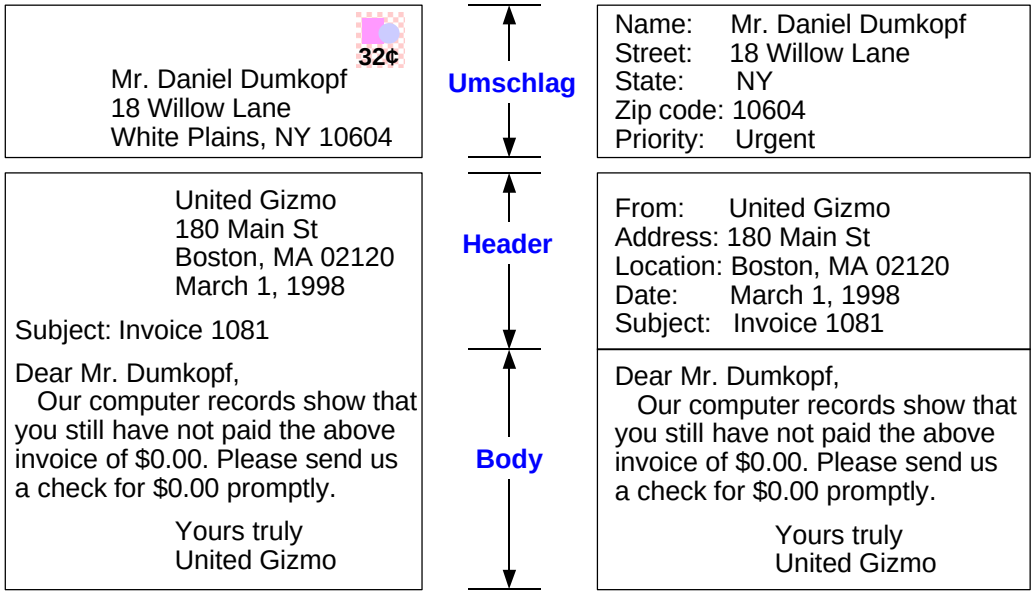


- Der **Benutzer** interagiert mit dem E-Mail-System über einen **Benutzer-Agenten** (**UA**).
- Jeder UA ist mit einem **Nachrichten-Übertragungs-Agenten** (**MTA**) verbunden, auf dem der Benutzer (der UA) ein Konto besitzen muß.
- Der MTA des Senders überträgt die E-Mails an den MTA des Empfängers. Dort wird die Nachricht **zwischen gespeichert**, bis sie vom UA des Empfängers abgeholt wird.
- Ein UA muß **keine ständige** Verbindung zum MTA unterhalten.

Architektur eines E-Mail-Systems (2)



Vergleich des Aufbaus eines Briefes und einer E-Mail



E-Mail Adressen nach RFC 822

Allgemeines **Internet**-Adressierungsschema:

➔ mittels **DNS-Adressen** (Domain Name Service)

mailbox@location

Benutzername *DNS-Adresse*

z.B.: gross@e-technik.fh-wiesbaden.de

Nachrichtenformat RFC 822 (ASCII-Version)

Aufbau: primitiver Umschlag (in RFC 821 beschrieben)
einige Header-Felder
Leerzeile
Nachrichtentext

- Keine klare Unterscheidung zwischen Umschlag und Header
- Mischung zwischen Umschlag und Header
- alle Angaben reiner ASCII-Text (7 bit)
- jedes Header-Feld besteht aus einer Zeile (Feldname, Doppelpunkt, Wert)

RFC 822 Header-Felder (für Nachrichtentransport)

Header	Bedeutung
To:	E-Mail-Adressen der primären Empfänger
Cc:	E-Mail-Adressen der sekundären Empfänger
Bcc:	E-Mail-Adressen für blinde Kopien an Dritte
From:	Verfasser der Nachricht
Sender:	Absender der Nachricht
Receive:	Zeile, die von jedem MTA auf der Route eingefügt wird
Return-Path:	Kann benutzt werden, bei speziellem Pfad zurück zum Sender

Cc - Carbon Copy
Bcc - Blind Carbon Copy
Return-Path - gibt an, auf welchem Weg eine Nachricht an den Sender zurückgeschickt werden soll (selten genutzt)

Weitere RFC 822 Header-Felder

Header	Bedeutung
Date:	Datum und Urzeit, wann die Nachricht gesendet wurde
Reply-To	E-Mail-Adresse, an die Antworten gesendet werden können
Message-Id:	Eindeutige Nummer der Nachricht
In-Reply-To:	Kennung der Nachricht, der diese Antwort gilt
Reference:	Andere relevante Nachrichtenkennungen
Keywords:	Vom Benutzer gewählte Schlüsselwörter
Subject:	Kurzer einzeiliger Betreff der Nachricht

Reply-To - Empfänger einer Antwort, wenn nicht der Verfasser (und auch nicht der Absender) die Antwort erhalten sollen

Nachrichtenformat MIME

Frühere E-Mail-Systeme übertrugen nur ASCII-Text (**7-bit !**)

Probleme: Länderspezifische Sonderzeichen (z. B. deutsche Umlaute)

Sprachen ohne Alphabet (Chinesisch, Japanisch)

Multimedia-Daten (Bilder, Video, Audio, ...)

Lösung: MIME-Erweiterung (Multipurpose Internet Mail Extensions)

Beibehaltung der ASCII-Kodierung (Kodierung in 7 bit)

dadurch alte Mail-Systeme und MTA weiterverwendbar

MIME-Header-Felder

Header	Bedeutung
MIME-Version:	Bezeichnet die verwendete MIME-Version
Content-Description:	Beschreibt für Empfänger den Inhalt der Nachricht z. B. *„Foto von Barbaras Meerschwein“*
Content-Id:	Eindeutiger Bezeichner, analog zu Message-Id
Content-Transfer-Encoding:	Beschreibt Kodierung des Nachrichteninhalts
Content-Type:	Bezeichnet die Art der Nachricht

Content-Transfer-Encoding Optionen (1)

Zur Auswahl stehen 5 (6) Codierungsmöglichkeiten:
ASCII, 8-bit, Binärcodierung, Base64, Quoted-Printable, (HTML)

1. ASCII-Text:

- Nur **7-bit** Zeichen, das 8. bit wird nicht berücksichtigt
- max. Zeilenlänge: 1000 Zeichen
- Ursprünglicher Standard, „ä, ü, ß“ müssen als „ae, ue, ss“ geschrieben werden.

2. 8-bit-Zeichen:

- Nutzt **8-bit**-Zeichen (alle Werte 0 .. 255)
- max. Zeilenlänge: 1000 Zeichen
- Verletzt das Kodierschema des ursprünglichen Standards
- Funktioniert manchmal sogar

Content-Transfer-Encoding Optionen (2)

3. Binärkodierung:

- Nutzt **8-bit**-Zeichen (alle Werte 0 .. 255)
- **Keine** Begrenzung der Zeilenlänge
- Verletzt das Kodierschema des ursprünglichen Standards
- Funktioniert noch seltener als die 8-bit-Zeichen Kodierung

4. Base64 Encoding (auch: ASCII Armor oder B Encoding):

- Eine *"richtige"* Art Binärdaten zu kodieren.
- Zerlegt Gruppen von 24 bit (von 3 * 8 bit) in 4 * 6 bit Einheiten
- Jede Einheit ergibt ein zulässiges ASCII-Zeichen
- Oft als *"3-4"* oder *"3-zu-4-Kodierung"* bezeichnet
- Empfänger mail Client muß diese aber decodieren können !

Content-Transfer-Encoding Optionen (3)

base64-Kodierungs-Regeln:

6-bit-Einheit	0	1	...	25	26	27	...	51	52	53	...	61	62	63
Kodierzeichen	"A"	"B"	...	"Z"	"a"	"b"	...	"z"	"0"	"1"	...	"9"	"+"	"/"

Nicht benutzte 6-bit-Einheiten werden mit "=" bzw. "==" angezeigt

Beispiel für base64 Encoding:

➔ aus "**Grüße**" wird für die Übertragung "**R3KB4WU=**"

```
G          r          ü
47         72         81
01000111   01110010   10000001
↑     ↑       ↑        ↑
010001 110111 001010   000001
R      3      K        B
```

```
ß          e
E1         65
11100001   01100101   --------
↑     ↑       ↑        ↑
111000 010110 0101(00) ------
4      W      U        =
```

Content-Transfer-Encoding Optionen (4)

5. Quoted-Printable Encoding (auch Q Encoding):

- Eine weitere *"richtige"* Art Binärdaten zu kodieren.
- Effizient, wenn die Nachricht fast nur aus (7-bit)-ASCII-Zeichen besteht und nur wenige ASCII-fremde Zeichen enthält, z. B. nur wenige Umlaute

Codierungs-Regeln:

Wenn Zeichen > 128 (d. h. das 8. Bit wird benutzt):

dann Darstellung als Hexadezimalzahl in der Form: **"=**hexa**"**

Beispiel für Quoted Encoding:

➔ aus "**Grüße**" wird für die Übertragung "**Gr=FC=DFe**"

Schlußfolgerung zum Content-Transfer-Encoding

8-bit-Zeichen- und Binärkodierung:

Sollten **nie** verwendet werden.

Base64:

Sollte nur dann verwendet werden, wenn vom Empfänger bekannt ist, daß er auch wirklich **MIME-kompatibel** ist.

ASCII-Text:

Wenn vom Empfänger nichts bekannt ist, oder wenn man an **Newsgruppen** schreibt.

Quoted-Printable Encoding:

Der goldene **Kompromiß**, wenn vom Empfänger nichts bekannt ist. (Aber **nicht für Newsgruppen** verwenden!)

HTML:

Sollte für E-Mail nicht verwendet werden, da (noch ?) wenig verbreitet.

Beispiel Content-Transfer-Encoding

```
[...]
MIME-Version: 1.0
Content-Type: text/plain
Content-Transfer-Encoding: Quoted-Printable

Das Meeting f=E4llt leider aus. Viele Gr=FC=DFe an das Team.

[...]
```

```
[...]
MIME-Version: 1.0
Content-Type: text/plain
Content-Transfer-Encoding: base64

RGFzIE1lZXRpbmcgZuRsbHQgbGVpZGVyIGF1cy4gVmllbGUgR3L832UgYW4gZGFzIFRlY
W0uDQoNClByb2YuIERyLiBXZXJuZXIgV2luemVybGluZw0KRkggRnVsZGEgIC0gIEFuZ2V
3YW5kdGUgSW5mb3JtYXRpaw0KDQo=

[...]
```


MIME-Header-Feld: Content-Type (RFC 1521)

Wichtigster MIME-Parameter, beschreibt wie die Nachricht zu interpretieren / darzustellen ist.

Typ	Subtyp	Beschreibung
Text	**Plain**	Unformatierter Text
	Richtext	ASCII-Text mit einfachen Formatierungen
Image	**GIF**	Bild im GIF-Format
	JPEG	Bild im JPEG-Format
Audio	**Basic**	Klangdaten
Video	**MPEG**	Videodaten im MPEG-Format
Application	**Octet-Stream**	Bytefolge (nicht direkt interpretierbar)
	Postscript	Druckbares Dokument im Postscript-Format

Content-Type Optionen (2)

Typ	Subtyp	Beschreibung
Message	**RFC822**	MIME-Nachricht nach RFC822
	Partial	Nachricht wurde zur Übertragung zerlegt
	External-Body	Nachricht muß noch über das Netz geladen werden
Multipart	**Mixed**	Unabhängige Teile in angegebener Reihenfolge
	Alternativ	Gleiche Nachricht in verschiedenen Formaten
	Parallel	Teile müssen gleichzeitig ausgegeben werden
	Digest	Jeder Teil ist eine vollständige RCF822-Nachricht

SMTP - Simple Mail Transfer Protocol

- Übertragungsprotokoll nach RFC 822
- Quellmaschine baut zum Port 25 der Zielmaschine eine TCP-Verbindung auf
- Protokoll läuft im ASCII-"***Klartext***" ab;
- Auf ein ***Kommando*** (z.B. HELO, RCPT, DATA, QUIT) folgt eine ***Antwort***
- ***Ablauf:***

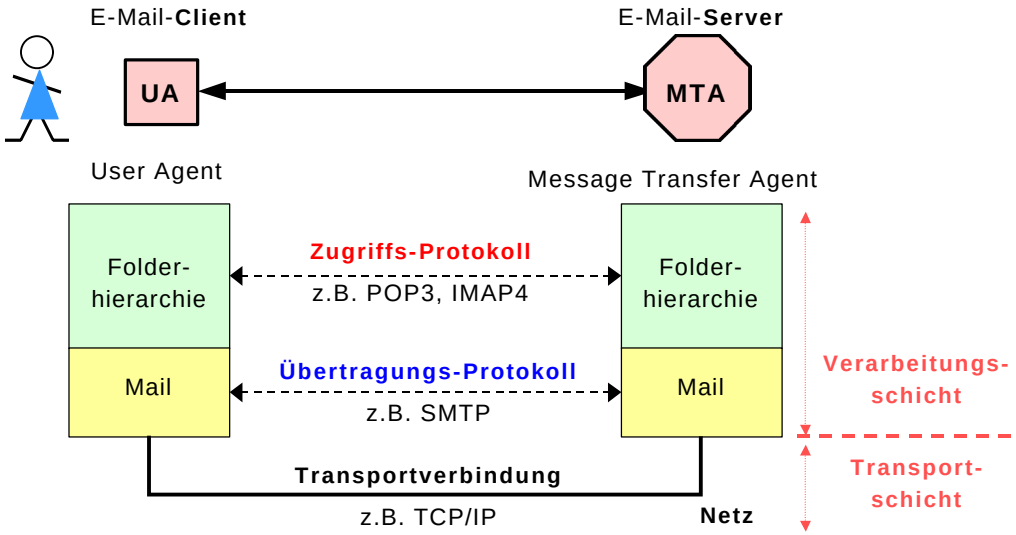
  Client (UA) baut Verbindung zum Server (MTA) auf und wartet, bis Server mitteilt, ob er bereit ist Mails anzunehmen,

  wenn nicht, beendet Client die Verbindung und versucht es später erneut,

  wenn Server bereit ist, beginnt Client mit dem Mail-Austausch

Mail-Zugriffsprotokolle (POP3, IMAP4)

Das **Zugriffs-Protokoll** setzt auf dem **Übertragungs-Protokoll** auf.



POP3 (Post Office Protocol, Version 3)

Offline Bearbeitungsmodus Prinzip:

- Nutzt die Verabeitungsfähigkeit ("Intelligenz") des Clients
- Im Internet ist **POP3** heute sehr verbreitet;
- Während einer kurzen und intensiv genutzten Online-Verbindung (Online-Kosten !) werden alle Nachrichten vollständig vom Server zum Client übertragen und anschließend werden die Mails auf dem Server gelöscht.
- Mails liegen anschließend (ausschließlich) auf dem Client vor und können dann dort offline gelesen und bearbeitet werden.
- Einfacher Befehlssatz erlaubt einfache Implementierung
- Aber: Oft kann nicht selektiert werden, welche Mails übertragen werden sollen (ungünstig z.B. bei unerwünschten Mails mit großen Attach-Files). Manche POP3 Clients (z.B. T-Online zeigen Subject-Zeile vor der Übertragung an.

IMAP4 (Internet Message Access Protocol Version 4)

Replikations-Modus Prinzip:

- Erweitert Offline-Modus um Synchronisation und Replikation
- Zentraler (Mail)-Datenbestand liegt auf dem Server. Ein oder mehrere Clients gleichen während der Online-Verbindung ihren (Mail)-Datenbestand mit dem Server ab (Replikation). Abgeglichen werden zu sendende und zu empfangende Mails.
- Übertragung von Teilen einer zu empfangenden Mail wählbar (z.B. nur Anschreiben, keine Attach-Files usw.).
- Im Internet gilt **IMAP4** als künftiger Standard.
- Aber: Umfangreicher Befehlssatz, dadurch aufwendigere Implementierung und noch Kompatibilitäts-Probleme zwischen verschiedenen Herstellern- Dies verzögert die Einführung von IMAP4.