Web Engineering

Inhaltsverzeichnis

Glossar

AJAX	Asynchronous JavaScript And XML
APIPA	Automatic Private IP Addressing
APOP	Authenticated Post Office Protocol
ASP	Active Server Pages
BA	Basic Authentication
CGI	Common Gateway Interface
CHAP	Challenge Handshake Authentication Protocol
DBMS	Database Management System
DHCP	Dynamic Host Configuration Pool
DNS	Domain Name Server
DOM	Document Object Model
ERP	Enterprise Resource Planning
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IMAP4	Internet Message Access Protocol
JSON	Java Simple Object Notation
MIME	Multipurpose Internet Mail Extension
MTA	Mail Transfer Agent
NAT	Network Address Translation
PAP	Password Authentication Protocol, simple
POP3	Post Office Protocol
SMTP	Simple Mail Transfer Protocol
TCP	Transmission Control Protocol
TP	Transaction Processing
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URL	Uniformed Resource Locator

1 Einführung

Grundlagen der Informatik, Kapitel 10

1.1 Internet

Um ein Teil vom Internet zu sein, muss man sich nur ans IP Protokoll halten und sich korrekt an andere Teilnetze anschliessen

1.2 Institutionen

ICANN Zuteilung von Nummern und Top Level Design, früher klein jetzt gross.

ISCOC Internet Society, Technologische Entwicklung des Internets

NIC CH/LI und SWITCH Vergabe von .ch Domains

W3C

1.3 Entwicklung

Zuerst wurde es zu militärischen Zwecken gebraucht ARPANET, es soll immer verfügbar sein. Danach verwendeten es Universitäten. Mit dem UI und den Namen wurde es auch für normale Nutzer attraktiv.

Router verbinden die Netze.

1.3.1 Befehle

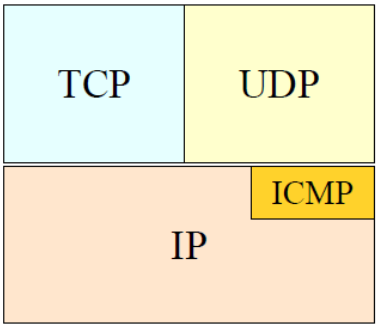
- ping: zum Test der TCP/IP-Installation arbeitet mit ICMP
- tracert
- telnet *IP port:* Aufbau einer Terminalverbindung

1.4 IP-Stack

Man müsste sagen, TCP via IP

Die IP-Schicht auf dem 3. Layer ist für die Adressierung zuständig und definiert das Adressformat. TCP, verbindungsorientiertes Protokoll, ist auf dem 4. Layer und stellt sicher, dass ein Packet wirklich ankommt und auch in der richtigen Reihenfolge. UDP prüft das nicht und ist ein verbindungsloses Protokoll. Es wird für Broadcast Nachrichten eingesetzt und ist schneller als TCP.

ICMP ist ein Satz von Befehlen, damit die Adressierung auch wirklich funktioniert, ist quasi Polizei.



1.5 IP Adressen

Jeder Rechner braucht eine eindeutige Nummer unter der er im Internet erreichbar ist. Jetzt inzwischen gibt es IPv6 *3ffc:675:53b:41:134:c35:ff:4*. Die Anzahl der IP-Adressen wird vergrössert. Das Problem ist, dass die Router jetzt die neue Adressierung lernen müssen. (Früher waren es 32 Bit)

Man hat meist keine eigene Adresse, sondern ist nur Teil eines Subnetzes, ist nur über dessen Gateway (oder Router erreichbar)

Ein Network Information Center (NIC) oder eine Domain Name Registry verwaltet eine oder mehrere Top-Level-Domains im Domain Name System.

Die Netzmaske definiert den Adressteil für das Netzwerk. Notation idR dotted decimal: 212.71.125.130

1.5.1 Arten

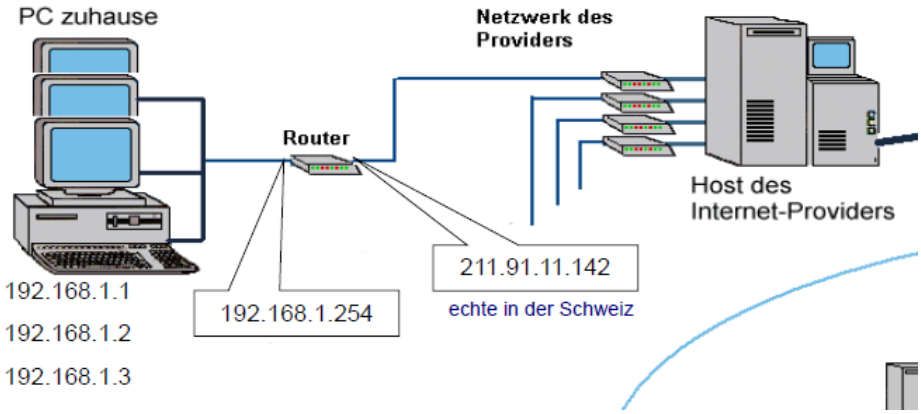
Lookback Adresse	127.0.0.1
statische Adresse	unveränderte Adresse für Rechner die ständig mit dem Internet verbunden sind.
dynamische Adresse	Adresse die vom DHCP-Server vergeben wird, wenn benötigt
Portnummber	Portnummer gehört zu einem Prozess – 0-1023 well known – 1024-49151 registrierte – 49152-65535 dynamische oder private
Netzwerkmaske und Broadcast	Wird bei lokalen Netzwerken verwendet: lokales Netz 192.168.1.n Netzwerkmaske: 255.255.255.0 Broadcast: 192.168.1.255
Host- und Domainname	Der Name der zur Nummer gehört *rechnername.domainname.netzwerk*
Nameserver DNS	Rechner, der den Namen oder IP Adresse auflöst

1.5.2 Netzklassen

Netz Klasse A	10.0.0.0 bis 10.255.255.255 1
Link-Local / APIPA	169.254.1.0 bis 169.254.254.255
Netze Klasse B	172.16.0.0 bis 172.31.255.255 16
Netze Klasse C	192.168.0.0. bis 192.168.255.255 256

1.6 NAT

Durch NAT können hinter einer öffentlichen Adresse mehrere private Adressen sein. Für die Verbindung mit dem Internet wird jeweils die öffentliche Adresse des NAT-Routers gebraucht.



1.7 DHCP

Durch DHCP bekommt ein Client eine Adresse aus einem Pool zugewiesen.

2 Client/Server Konzept

WWW ist eine Anwendung, die man über das Internet verwenden kann. Es ist ein graphischer Zugang zu einem Hypertextsystem. Es basiert auf dem Client/Server Konzept.

WWW ist nicht gleich Internet

Es braucht einen Client- und Serverprozess. Der aktive Serverprozess (Zustand: listen) nimmt Request entgegen und sendet Response. Zudem hält der Server die Informationen als HTML Dokumente.

Der Client schickt Anfragen an den Server, nimmt Informationen vom Server entgegen und stellt sie graphisch dar.

Die Links sind unidirektional?

Die 3 Grundkomponenten vom WWW sind HTML, http und URI.

2.1 HTML

HTML ist eine Layoutsprache, die speziell für das Web entwickelt worden ist. Das spezielle an ihr sind die Hyperlinks, Verweise auf ein Dokument auf einem anderen Rechner.

- zwingend sind <html>, <head>, <body>
- Dateiendung muss .html sein
- Element: alles zwischen einem Tag
- Attribut: **name="value"**
- Referenz: href="…"

2.2 HTTP

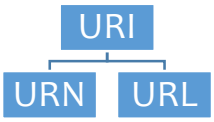
Die Anwendung *hyper text transfer protocol* ist zustandslos. Es wird für die Kommunikation zwischen Webserver und Webbrowser verwendet. Es gibt nur Anfrage und Antwort. Das Problem wird mit Cookies gelöst.

1. Aufbau mittels TCP/IP zum gewünschten Webserver

2. Anfordern der gewünschten Informationen über die Verbindung
3. Antwort des Servers durch Sendung der angeforderten Informationen
4. Abbau der Verbindung nach der Datenübertragung

2.3 URI

Ein URI ist ein Konzept zur Adressierung von Objekten im Internet. URL ist der Spezialfall vom URI.



2.3.1 URL Beispiel

schema://authority(DNS)/path?query#fragment

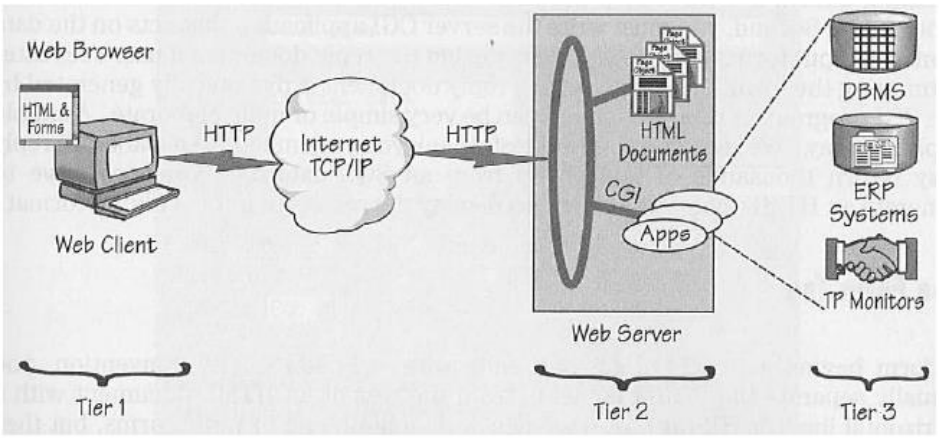
#fragment, damit nicht das ganze geladen werden muss

http://ww.hsw.fhz.ch/infos/hswinfo.htm#AbteilungIT

2.3.2 Muster complex

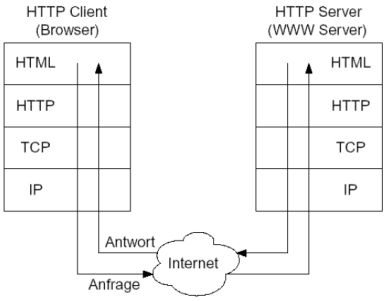
Protokoll://[user:passwort@]host.dom.ch[:Port][/verzeichnis[/Dokument[#AnkerImDokument]]]

3 3 Schicht-Modell



Presentation (GUI / Browser) **Business-Logic** (Application Server) **DBMS / ERP / TP** (Data Server)

1. Presentation- Browser Schicht:
   a. GUI Repräsentation – View
   b. Aufbereitung und Präsentation der Daten
2. Business Logic - Webserver-Schicht:
   a. Management der HTTP-Verbindungen
   b. koordiniert Anfragen und schiebt richtige Daten an die View
   c. Zugriff auf Tier-3 Applikationen via CGI
3. Data: Datenhaltungsschicht
   a. Verwaltung von Transaktionsdaten und Datenbanken



4 Webserver

Man kann auf das Filesystem mit file\\ zugreifen oder über http.

Apache wird als Service und nicht als Applikation installiert. Er muss immer unter einem Benutzer mit eingeschränkten Rechten laufen. (Applikationen für Menschen und Service für andere Programme und enthalten nur eine Funktionen, Applikationen mehrere)

4.1 Konfigurationsdateien

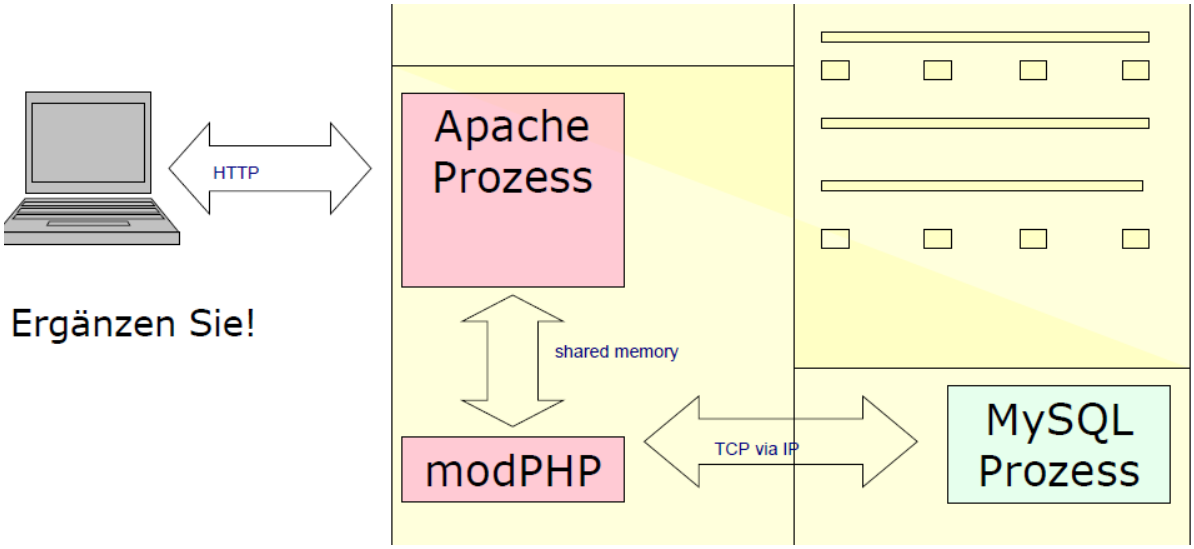
- httpd.conf
- php.ini
- perl.ini
- my.ini

4.2 Kommunikation

Der Browser auf der Clientmaschine kommuniziert mit dem Apache-Serverprozess *httpd.exe* auf der Servermaschine per HTTP (via TCP)

modPHP wird als Modul in Apache dynamisch eingebunden d.h. zur Laufzeit geladen. Es gehört zum Apache-Prozess und teilt mit ihm Speicherbereiche.

MySQL läuft als eigenständiger Serverprozess und kommuniziert i.d.R. mit Apache via TCP. In produktiven Umgebungen läuft der Datenbankprozess i.d.R. nicht auf der gleichen Maschine wie der Webserverprozess. Die Verbindung zur DB hat dann als Zieladresse nicht mehr «localhost» wie bei unseren Übungen.



4.3 Services

Wenn Webserver, Datenbankserver und FTP-Server als Service installiert würden, laufen sie unter dem Konto SYSTEM und bei Systemstart automatisch. Das ist ein Sicherheitsrisiko sonst könnte ein gewöhnlicher Benutzer die FW umkonfigurieren.

4.4 Datenbank

Eine Kollation definiert eine Sortierreihenfolge. Man kann testen ob Müller, Muller richtig sortiert werden.

5 Joomla

Ein CMS hat folgende Vorteile:

- Inhalt einfach zu verwalten
- Nicht-Informatiker können eine solche Site erstellen
- Funktionen müssen nicht selbst programmiert werden

Joomla besteht aus einem Grundgerüst (Core), mind. 1 Komponente, Modulen und einem Template.

Core	– bildet technische Basis für Darstellung von Inhalten – regelt Berechtigungen – stellt Strukturen zu Verfügung
Komponente	– befindet sich im Hauptbereich – bereitet Daten auf und stellt sie im Frontend dar
Template	– definiert Farben, Formen, etc. – besteht aus HTML, PHP, CSS
Modul	– wird irgendwo im Template platziert – hat nur eine bestimmte Aufgabe
Plugins	– kleine Skripte, die im Hintergrund auf Ereignisse reagieren

5.1 Konfiguration

- Eine DB brauch einen anderen Benutzer als das CMS
- Php konfigurieren *php.ini*
- Firewall konfigurieren, dass Emails erlaubt

- Super-User muss Mails erhalten können

6 Dokumenttypen

1. Statisch: Enthält HTML Code *Erste Übung mit Hund*
2. Dynamisch: Enthält Code (z.B. PHP), der serverseitig ausgeführt wird *SBB*
3. Aktiv: Enthält Code (z.B. Javascript), der clientseitig ausgeführt wird, braucht Aktion vom Client
4. Mischung zwischen dynamisch und statisch: AJAX (überträgt Daten als XML), DOM (Zugriff auf Dokumenteigenschaften) *SBB*
5. pseudo aktiv: statische mit <img … > Browser braucht doch was zu tun

6.1 Dynamische Dokumente mit CGI

Damit dynamische Dokumente entstehen, arbeitet der Webserver mit anderen Programmen zusammen. Dazu wird die Schnittstelle CGI Common Gateway Interface zusammen. Sie beschreibt die Funktionalität und Rahmenbedingungen der Schnittstelle.

Je näher Webserver und Programmierwerkzeuge "verwandt" sind, desto effizienter ist die Implementation.

6.2 DHTML und DOM

Eine DHTML Seite aktualisiert sich selbst bei der Interaktion mit einem Benutzer. DOM ist ein Synonym. DOM verwendet Objekte. DoM ist nicht bei allen Browsern gleich implementiert.

6.3 Datenübertragung

- POST, diskret: sieht Wert in URL nicht
- GET: Paramter werden sichtbar in der Adresszeile übertragen, enthält keine Umlaute
- Querystring http://formular.ch?name=Desiree

6.4 Mailserver

Heute erwarten Mailserver eine Authentisierung. Die PhP Installation muss dazu Verbindungen verschlüsseln können. Dazu wird SSL konfiguriert und der Serverprozess muss selbst eine Verbindung zum Mailserver aufbauen können. Die Firewall Regeln müssen ok sein.

6.5 Datenbankverbindung

Für die Verbindung zu DB müssen die Verbindungsangaben und der Primärschlüsselname angegeben werden. Dies geschieht in der include Datei *view_inc.php. HTMLEntities()* überträgt Werte als Objekt.

Für CRUD Operationen wird jeweils eine neue php-Datei erstellt.

7 Cookies

Cookies werden gebraucht damit verschiedene Requests einem Browser zugeordnet werden können.

Browser speichern die Cookies während der aktuellen Session oder bis zum Ablaufdatum. In der Regel wird nur der Session Key herum geschickt. Die Informationen speichert der Server jeweils ab. Auch wenn Server heruntergefahren wurde, hat er die Informationen noch.

7.1 Eigenschaften

Ein Cookie ist mindestens 4KB und hat folgende optionale Eigenschaften:

- Verfallsdatum
- Domäne (Server?)
- Pfad, da mehrere Cookies pro Domäne
- Sicherheit, nur wenn über SSL
- Nur HTTP, JS darf das Cookie nicht sehen

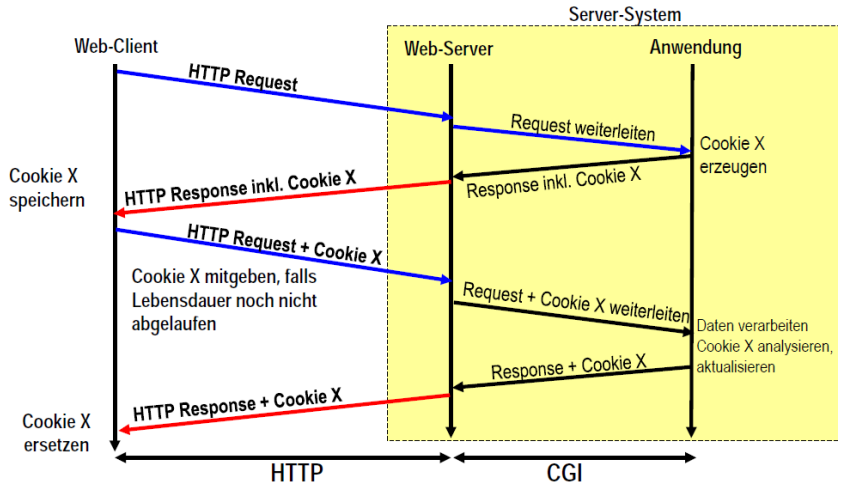
Untergrenzen für Browser

- mind. 20 Cookies pro Domäne
- mind. 3000 Cookies insgesamt
- mind. 4096 Bytes pro Cookie

Warum braucht es diese Untergrenzen?

7.2 Ablauf

1. 1. Anfrage des Browsers
2. Der Server erstellt ein Cookie und schickt es in der Response mit
3. Client speichert das Cookie
4. Bei einer Änderung schickt der Browser einen Request inkl. Cookie
5. Der Server berechnet den Inhalt des Cookies neu und schickt es in der Response mit



7.3 Fremdcookies

Betreiber eines Webserver schliessen sich zusammen, damit sie auch die Cookies der anderen lesen könne. Fremdcookies sind heute nicht mehr üblich. Also Cola speichert eines für Rivella. Wird herausgefunden und lebt nicht mehr lange.

7.4 Sessions

Im Unterschied zu Cookies werden die Informationen auf dem Server gespeichert und der Browser wird durch eine Session-ID identifiziert. Korrekt?

Sessions dienen zur Überwindung der Zustandslosigkeit von http. Die Session-ID gehört zu einem Browser und wird via Cookie oder Query-String übermittelt. $_SESSION[‚Warenkorb‘] = (Geht aber nicht für alle Datentypen)

Auf dem Server liegen sie im Verzeichnis /tmp. Timeout Parameter werden in der php.ini Datei gespeichert. Inhalte einer Session werden als Dateien gespeichert und nach Beenden wieder gelöscht.

7.5 Aufgaben

Cookie setzen: http://localhost/setcookie.php?Bisquit=Willisauerringli

Session-Hijacking (stehlen des Cookies):

1. Anzeigen des original Cookies: javascript:document.write(document.cookie)
2. Cookie in Zwischenablage kopieren
3. Im 2. Browser ein Cookie mit identischer PHPSESSID hinzufügen: javascript:document.cookie=“PHPSESSID=“lsjfs“
4. wieder Seite öffnen

8 Authentisierung

- Authentisierung: Wer bin ich?
- Autorisierung: Was darf ich?

Eine Authentisierung selbst zu Codieren bedeutet ein riesiger Aufwand. Am besten verwendet man die vom CMS, da jene vom Server bei einer Portierung Schwierigkeiten machen kann.

8.1 Konfiguration

Für die Konfiguration wird bei Apache **httpd.conf** benötigt. Dieses beinhaltet den Name der Datei, die über Benutzerrechte Auskunft gibt.

1. Zugriffsbeschränkungen für Container festlegen *<Directory>*
2. Benutzer- und Gruppennamen festlegen
3. Passwörter speichern, mit htpaswd.exe Hashwert speichern

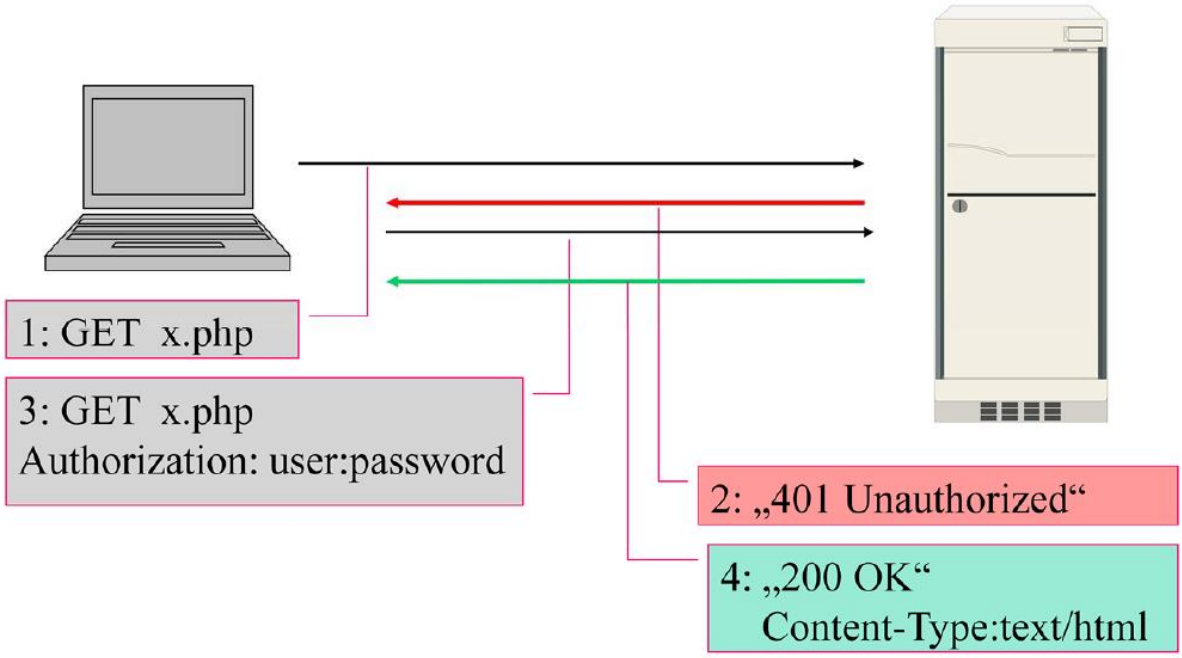
Dezentrale Konfiguration bedeutet, dass nicht die zentrale Konfiguration vom Server genommen wird. Dazu werden mit **.htaccess** die Zugriffsrechte oder auch benutzerdefinierte Error Pages definiert. Das File ist auch für die unterliegenden Verzeichnisse gültig bis ein weiteres .htaccess auftaucht. Bei Änderungen ist kein Restart erforderlich. *.htaccess* verweist auf *xampp/security/xampp.users*

Dieses File darf nicht vom Server herausgegeben werden. Dazu muss die <FilesMatch … > Direktive mit Regulären Ausdrücken angepasst werden. Die Dezentrale Konfiguration ist nicht ungefährlich.

Da für Windows die .htaccess Datei ungewohnt ist wird htaccess.sic verwendet.

8.2 Basic Authentication

BA übermittelt Benutzername und Passwort mit Base64 (unverschlüsselt aber codiert) über den http Header.



8.3 Digest Authentication

CHAP: Challenge Handshare Authentication Protocol, gibt keine Umkehrfunktion, Verwendet MD5-Hash vom Passwort.

Der Client bekommt eine Aufgabe die er mit Hilfe des Passwortes lösen muss.

username: Name, unter dem der Client sich anmeldet
nonce, realm, algorithm, opaque: die Werte der Challenge, zufälliger Wert vom Server.
uri: URI, die zur Challenge geführt hat (die Anfragezeile könnte von einem Proxy verändert worden sein)
response: Hash, der aus Nonce, Benutzername, Kennwort und möglicherweise weiteren Bestandteilen der Nachricht berechnet wurde.
cnonce: Zufälliger Text des Clients, der in den Hash einbezogen wird und Chosen Plaintext Angriffe verhindert.
qop: gewählte Sicherheitsstufe
nonce-count: Zähler, wie oft der nonce in Requests verwendet worden ist
auth-param: z.Zt. nicht benutzt

8.4 Replay Attacke

Wenn das Passwort einer Person abgefangen wird und von dieser Person verwendet wird.

8.5 Salt

Salt ist eine zufällig gewählte Zeichenfolge um das Passwort zu verschlüsseln. Sie wird an das Passwort in Klartext angehängt und danach wird alles zu einem Hashwert gemacht.

8.6 Authentication bei Joomla

In Joomla sind Nutzerdaten in Tabelle. Sind die Daten verschlüsselt?

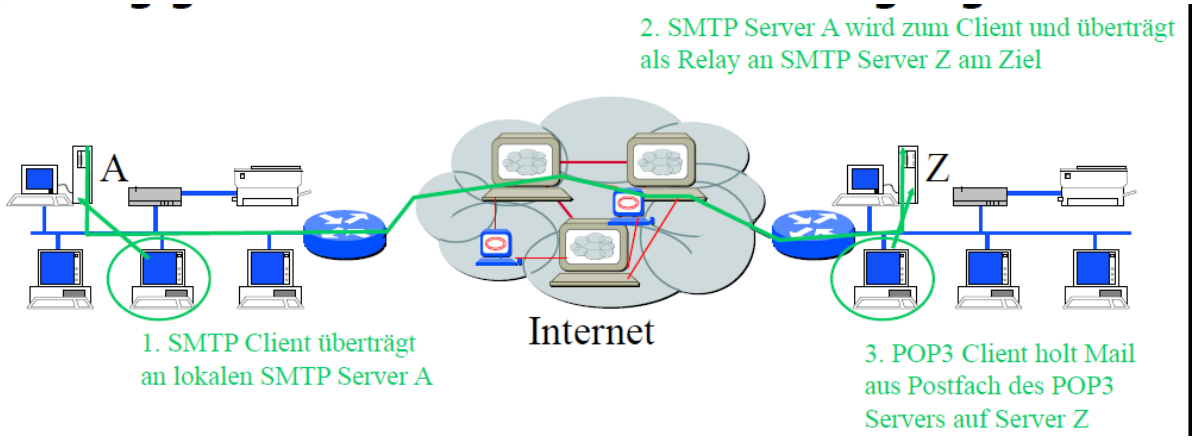
9 Mail

Emails sind wie Briefe:

- Authentisierung nur beim Lesen der Emails
- Emails können überall aufgegeben werden

Sie werden über SMTP übertragen.

9.1 SMTP



Ein Mail Transfer Agent ist ein SMTP Server, der die E-Mail weitergibt und seinen Header hinzufügt. Stimmt das? Eine Server Antwort beginnt immer mit einem 3 stelligen Code.

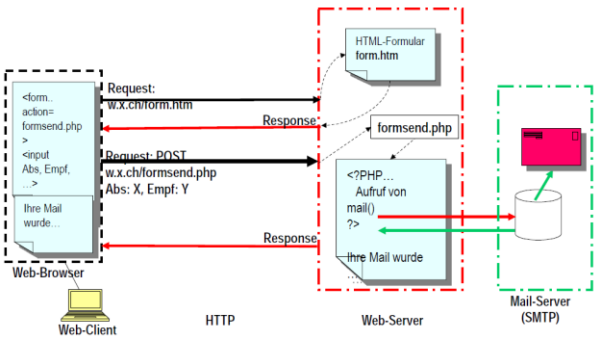
Eine Datei über SMTP statt über FTP zu übertragen braucht 33% Bandbreite mehr. Da 8 Bits für 6 Zeichen bei Base64 gebraucht werden.

9.2 Mail via Webserver

Optionen:

1. Hyperlink mit mailto: lokaler Mailclient wird verwendet
2. Formular mit Action sendmail: ruft Mailprogramm auf dem Server auf
3. Mailfunktion mit Skriptsprache: ruft externer Mailserver auf

Für Optionen 2 und 3 braucht es ein MTA oder SMTP-Client als Relay.



9.3 Mittel gegen SPAM

- keine offenen Relays

- nur authentisierte Benutzer senden lassen
- nur verschlüsselte Verbindungen zulassen
- Filterprogramme

9.4 Mail Header

Der jüngste Mail Header steht zuoberst.

- Received: von wem, alle Stationen MTA
- Return-path
- Date
- From
- To
- MIME-Version
- Subject
- Message-ID
- Content-Type
- Content-Transfer-Encoding

9.5 Mail-Bounces

Ein Mail hat zwei Umschläge. Im äusseren Umschlag steht MAIL FROM (Absender) und RCPT TO (Absender, der bei Fehler kontaktiert wird). Im Inneren stehen From, Reply-To und To.

9.6 MIME

Damit Dateien mitgeschickt werden können braucht es MIME. Soll keine Dateien unnötig anhängen.

9.7 POP3 vs. IMAP

Pop3 holt die Nachrichten vom Server, während bei IMAP4 die Nachrichten auf dem Server bearbeitet werden.

9.8 APOP

APOP ist ein Verfahren zur sicheren Übertragung des Passwortes beim Abruf von Emails mit POP.

Statt dem Passwort wird ein Hash aus dem Passwort und Zeitstempel übermittelt. Ein Spion sieht nur den Benutzernamen aber nicht das Passwort.

10 FTP

Browser können Teile des Protokolls abdecken: ftp://user:passwort@server.ch/startverzeichnis

Es werden keine Befehle und Daten verschlüsselt, das wird heute durch SFTP oder SCP gelöst.

Heutzutage werden FTP passiv gemacht (PASV). Es gibt zusätzlich eine Steuerverbindung. Der Client ist für Datenverbindung verantwortlich. Die Verbindung ist uni – statt bidirektional.

10.1 Active Mode

1. Client ruft Port 21 des Servers auf (Control Connection)
2. *PORT ip, portnummber* wird empfangsbereite Port des Clients angemeldet
3. Server überträgt Daten über Port 20 an angemeldeten Port vom Client

10.2 Passive Mode

Active Mode ist oft gar nicht möglich weil der Client gar keine eigene IP hat sondern hinter einem Router sitzt. Darum verlang der Client PASV Mode. Jede Verbindung geht dann vom Client aus.

Der Server muss sich den offenen Port merken **stateful inspection.** Das heisst die Firewall muss Bescheid. Der Port ist nur während des Datentransfers offen.

Damit der Server bestimmen kann, welche Ports er für passive Verbindungen öffnet muss folgendes Konfiguriert werden

IP	Port , + weitere
	1. Ziffer * 256, + weitere
10,9,35,2	100,1 -> 25601

Passive Mode: Entering PASV Mode 197, 187, 187, 198, 100,1

Beim Zugriff mit Fileserver werden auch gesperrte Dateien angezeigt. Dokumente und Bilder werden als binäre Dateien übertragen.

10.3 Passwort mitschicken

ftp//user:passwort@server.ch/verzeichnis

10.4 Checkliste für FTP Server

1. Benutzer anlegen
    a. Zugriffsrechte bestimmen
    b. Einstiegsverzeichnis bestimmen
2. Benutzer *Anonymous* löschen
3. Rollenkonzept erstellen
4. Sicherheitskonzept konfigurieren
    a. PASV verlangen
    b. Port für Datentransfer annoncieren
    c. Port muss im Firewall-Konzept zugelassen werden
    d. Definieren welche Dateien zugelassen werden

11 Applet

1. Anforderung eines Dokuments
2. Übermitteln des Dokuments und Applets in Java-Bytecode
3. Laden JVM und ausführen der Applets
    a. JVM packt JAR aus
    b. JVM lädt Startklasse (wird im HTML angegeben)
    c. JVM interpretiert Bytecode und führt ihn aus
4. Code verwerfen und Threads killen, wenn ungebraucht.

Die Java Sandbox regelt den Zugriff auf lokale Ressourcen. Ein Applet darf eine TCP-Verbindung nur zum Server aufbauen, von dem es geladen wurde.

11.1 Java Script

Wegen JS sind Applets überflüssig? JS war nicht zur Interaktion mit dem Server gedacht. Heute sollte das aber möglich sein. Die Datenstruktur vom Dokument (DOM) wird von JS optimiert. Es kann auf die Methoden eines Applets zugreifen

```
<script language="JavaScript"></script>
```

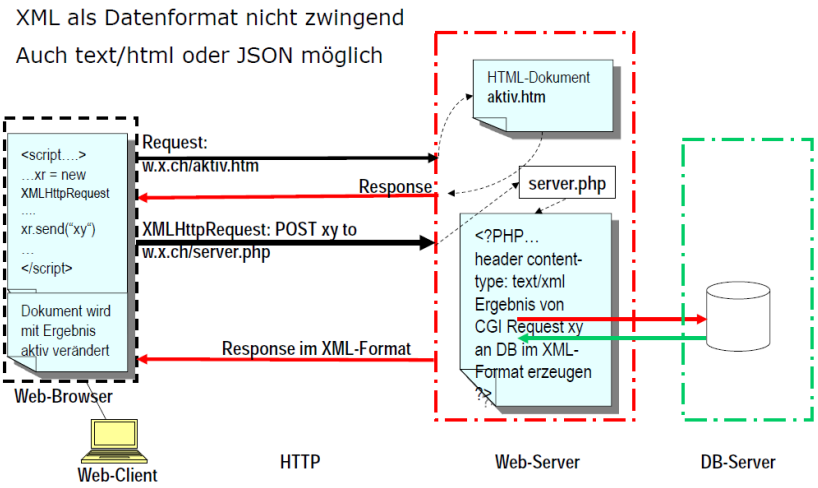
JS kann Methoden public Methoden vom Applet aufrufen.

11.2 Ajax

Durch `XMLHttpRequest();` kann JS Requests ausführen aber nur zum Server von dem das Applet stammt. Der Zustand der Verbindung muss laufen überprüft werden. Man sieht dies am Status 4. Es wird zum Beispiel gebraucht um Suchanfragen zu ergänzen.

Verbindungsaufbau vom Browser aus

1. Hole vom Server eine aktives Dokument
   Aktiv.htm enthält Javascript mit XMLHtppRequest (AJAX)
2. Mit send() werden Informationen zum Server gesendet
3. Ein Prozess auf dem Server wartet auf gesendet Informationen (Listend) interpretieren
4. SBB: Bahnhöfe suchen aus DB und per XML (oder JSON) zurück an den Client schicken



11.2.1 JSON

Es gibt Webservices die im JSON Format kommunizieren. Generell ist es einfacher zu handhaben als XML.

12 Fragen

- Je näher Webserver und Programmierwerkzeuge "verwandt" sind, desto effizienter ist die Implementation. Was ist mit effizient gemeint?