

Hálózatok elmélet – Fábián Zoltán

Webes szolgáltatások

Hypertext Transfer protokoll

- ! RFC2616
- ! A TCP/IP-re épülő szolgáltatás, a default szolgáltatási címe 80-as port
- ! Szokásos további címek: 81, 82, 83, 1080, stb...
- ! Kliens-szerver architektúra
- ! Kérés/Válasz protokoll
- ! Állapot nélküli protokoll

A HTTP tranzakció lépései

- ! HTTP kliens lehet: böngésző, user agent, program (JAVA, PHP, C# alkalmazás, AJAX javascript alkalmazás)
- ! HTTP szerver (web szerver)

§ Microsoft termék:

- Internet Information Server
- Personal Web Server

§ Nem Microsoft Termék

- Apache (Majdnem minden Platformon)
<http://www.apache.org>
- Apache Tomcat (JAVA szerver oldali kiszolgáló)
- NGIX



Tranzakció jellemzői

- URL (Uniform Resource Locator)
`http://<host>[:port][<path>[?<query>]]`
- Szöveges metódusok (parancsok)
- Verziószám
- Szöveges információk a fejlécben
content-type, server, user-agent, referrer, location
- Logikailag csoportosított státusz kódok
(303 - átirányítás, 404 – Nem létező oldalhiba, 200 -ok)
- Átküldött tartalomtípusok MIME típus
-html: text/html, jpg:image/jpeg,

Példa egy Request üzenetre

- ! GET / HTTP/1.1 - Metódotus, Verziószám
- ! Host: www.petrik.hu - URL
- ! Accept: */* - Paraméterek
- ! Accept-language: hu
- ! Accept-Encoding: gzip, deflate
- ! User-Agent: Mozilla/4.0 (compatible;MSIE 6.0;Windows NT5.1
- ! Connection:Keep-Alive

Példa egy Response üzenetre

- ⚠ HTTP/1.1 200 OK <- Verziószám, státusz kód
- ⚠ Date: Fri, 06 Sep 2010 08:39:05 GMT
- ⚠ Server: Apache/2.1.2(Win32) PHP/5.1.15
- ⚠ Last-Modified: Thu, 22 May2010 12:03:45 GMT
- ⚠ ETag:"0-dc5-3eccbca1,,
- ⚠ Accept-Ranges: bytes
- ⚠ Content-Length: 3525 <- Tartalom hossza
- ⚠ Keep-Alive: timeout=15, max=100
- ⚠ Connection: Keep-Alive
- ⚠ Content-Type: text/html
- ⚠ <HTML><HEAD>... <- Tartalom

Főbb HTTP parancsok

- ! GET: Erőforrás kérés (oldal lekérés)
- ! POST : adatok küldése
 - § PUT: fájl feltöltés
 - § DELETE: fájl törlése
 - § HEAD: erőforrás paramétereinek lekérdezése

HTTP státuskódok

- 1xx: információs

- 2xx: sikeres

- 200 OK

- 3xx: átirányítás

 - § –301 Moved Permanently

- 4xx: kliens hiba

 - § –403 Forbidden, 404 Not Found

- 5xx: szerver hiba

 - § –500 Internal Server Error, 503 Service Unavailable,

- 505 HTTP Version Not Supported

Fontosabb Request információk

- ! Host: a kérdéses szerver
- ! If-Unmodified-Since: csak ha az adott dátum óta változott
- ! Referer: honnan történt a hivatkozás az erőforrásra
- ! User-Agent: böngésző típusa
- ! From: böngésző személy e-mail címe

Fontosabb response fejléc információk

- ! Server: webszervertípusa, verziója
- ! Cache-Control: no-cache/no-store/max-age
az erőforrás cache-elésiparaméterei
- ! Expires: az erőforrás mikor évül el
- ! Content-Length, Content-Type: a válasz
üzenet hossza és típusa
- ! Last-Modified: az erőforrás utolsó
módosításának dátuma

Főbb MIME típusok

i application/

§ –application/msword: Word dokumentum

§ –application/octet-stream: futtatható program

§ –application/zip: zipfájlok

i audio/

§ –audio/midi: midi fájlok

§ –audio/mpeg: mp3 fájlok

MIME típusok 2.

! image/

§ image/bmp: bmp fájlok

§ image/gif: gif fájlok

§ Image/jpeg: jpg fájlok

! text/

§ text/html: html fájlok

§ text/plain: txt fájlok

! video/

§ video/mpeg: mpg, mpeg fájlok

§ video/x-msvideo: avi fájlok

HTTP GET próba

- ! TELNET-tel
- ! Start/Futtatás/telnet.exe
- ! Set localecho
- ! Open localhost 80
- ! GET / HTTP/1.1
- ! Host: localhost
- ! vagy FIREFOX FIREBUG kiegészítésével

HTTP felhasználása

- ! Web böngészés
- ! WEB szolgáltatások
- ! Bújtatott szolgáltatások
 - Egy rendszerfelügyeleti vagy távoli elérési program a 80-as porton kapcsolódik egy külső modulhoz. Pl: TeamViewer

HTTP biztonsága

- ! GET – URL része a paraméterlista
- ! POST – a felhasználó elől eldugjuk
- ! Kérések szerver oldali feldolgozása PHP segítségével
 - § PHP: `$_GET`, `$_POST` globális tömbök.
 - § Típus és tartalomellenőrzés után használható csak fel a szerver oldali programban
 - § Minta alkalmazás...

Biztonságos változat HTTPS

- ! Hasolít a HTTP-hez
- ! 443-as TCP porton kommunikál
- ! HTTPS://
- ! SSL TLS titkosítás
- ! A Webszerveren nyilvános kulcs igazolást (public key certificate), pl OPENSSL segítségével
- ! Öntanúsító igazolás

HTTP Streaming – Új felhasználás

! HTTP - PUSH protokoll