

Rechnernetze I

Übungsblatt 7

Anne Martens, Felix J. Oppermann

10. Juli 2006

1 HTTP

- a) HTTP wird verwendet um Dateien im Internet zu übertragen. Das Protokoll dient dabei dem Herunterladen von Daten eines entfernten Rechners. Die häufigste Anwendung ist das Abrufen von Webseiten oder Elementen einer solchen.

- b) Das Anwendungsprotokoll HTTP setzt im TCP/IP-Stack auf dem TCP-Protokoll auf.

- c) `felix@lizzy ~ $ telnet felix-oppermann.de 80`

```
Trying 85.13.134.154...
```

```
Connected to felix-oppermann.de.
```

```
Escape character is '^]'.
```

```
GET /index.html HTTP/1.1
```

```
Host: felix-oppermann.de
```

```
HTTP/1.1 200 OK
```

```
Date: Sun, 02 Jul 2006 12:18:02 GMT
```

```
Server: Apache/1.3.27 (Linux/SuSE) mod_fastcgi/2.4.2 FrontPage/4.0.4.3 PHP/4.4.1 mod_perl/1.27  
mod_ssl/2.8.12 OpenSSL/0.9.6i
```

```
Last-Modified: Thu, 19 Jan 2006 19:45:26 GMT
```

```
ETag: "18076-2e8-43cfec56"
```

```
Accept-Ranges: bytes
```

```
Content-Length: 744
```

```
Content-Type: text/html
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  
<html>  
<head>  
<title>Felix Oppermann</title>  
</head>  
<body>  
<center>  
<br>  
<h2> Felix Oppermann</h2>  
Vielleicht wird hier irgendwann auch noch mal eine richtige Homepage entstehen.  
<br><br><br>  
<a href="/uni"> Einige L&ouml;nungen zu &Uuml;bungszetteln</a><br>  
<br><br>  
<a href="/perzeptronen/index.html">Vortrag zum Thema Perzeptronen</a><br>  
<br>  
<a href="/art/index.html">Vortrag zum Thema ART</a><br>  
<br><br>  
<a href="http://mentis.uni-oldenburg.de">Projektgruppe Mentis</a>  
<br><br>  
<br><br>  
<br><br>  
<a href="http://www.nosoftwarepatents.com"></a>  
</center>  
</body>  
</html>Connection closed by foreign host.
```

- d) **GET** Mit GET können Informationen vom Server angefordert werden.

POST Der Befehl POST kann ebenfalls zum Abrufen von Informationen verwendet werden. Zusätzlich zu GET sind die Möglichkeiten erweitert beim Abrufen Informationen an den Server zu übertragen.

HEAD Mit HEAD können Metainformationen zu einer Webseite abgerufen werden ohne die Datei zu laden. Diese kann zum Beispiel verwendet werden um den Inhalt eines Caches oder Proxis auf Aktualität zu prüfen.

- e) Andere Nachrichten als Text werden bei HTTP mit einem zum MIME ähnlichen Verfahren kodiert übertragen. Diese entspricht in den wesentlichen dem im E-Mail-Verkehr verwendeten MIME-Standard, weist jedoch in Details Abweichungen auf. Die Kodierung ist für HTTP/1.1 in RFC 2616 spezifiziert.

Beispiel:

```
felix@lizzy ~ $ telnet felix-oppermann.de 80
Trying 85.13.134.154...
Connected to felix-oppermann.de.
Escape character is '^]'.
GET /nswpat80x15.gif HTTP/1.1
Host: felix-oppermann.de

HTTP/1.1 200 OK
Date: Mon, 10 Jul 2006 15:39:18 GMT
Server: Apache/1.3.27 (Linux/SuSE) mod_fastcgi/2.4.2 FrontPage/4.0.4.3 PHP/4.4.1 mod_perl/1.27 mod_ssl/2.8.12
OpenSSL/0.9.6i
Last-Modified: Wed, 11 Jan 2006 21:29:29 GMT
ETag: "1807a-c7-43c578b9"
Accept-Ranges: bytes
Content-Length: 199
Content-Type: image/gif

GIF89aP33fyù,P@ÈiI?U°P?à?værBé
Åáo<*I;Içà)§B-2#Ään0?è² iÜÄq§Qn
ä»µ«-5$öFW?8f%Èö×0?H9(ö*è79C0
äü)jBJs²éäBüj!;K|<P;
```

- f) Der MIME-Standard, an den die in HTTP zu Übertragung von Binärdaten verwendete Kodierung angelehnt ist, wird vorrangig für E-Mail-Nachrichten verwendet. Die aktuelle Version von MIME ist in den Dokumenten RFC 2045, RTF 2046 und RFC 2047 spezifiziert. In folgendem Beispiel wurde im Gegensatz zu der in der vorigen Nachricht aufgeführten HTTP-Nachricht die base64-Kodierung verwendet

Beispiel:

```
--Boundary-00=_egnsETfBhjUaUi2
Content-Type: image/gif;
  name="nswpat80x15.gif"
Content-Transfer-Encoding: base64
Content-Disposition: attachment;
filename="nswpat80x15.gif"

R01G0D1hUAApAJEAAp9mADMzZpmZmf///yH5BAAQAAAAALAAAAABQAA8AAAKY1I+py+0Pz5i02ouz
3rybAYTiGHbmiXIIfyQLpC59r08b2ba1gWe5UEJgEB0BicUgUCofF5W8ZPCqBzKH05+Ipp8ktMpoE
I8PkbT8neiyIOyT3GTCj2enUW4M47uDtaut9ST2RgdXlgaGpjhMJQDI89fTpEg5KPY16JWINz1D
04MTCuP5KWqaQkPzUurhyPpqEyE7S1urUAAA0w==

--Boundary-00=_egnsETfBhjUaUi2--
```

2 Networking in Java

Die Java Klassen für diese Aufgabe liegen der Abgabe bei.

- Realisiert in der Klasse A_URLNetworking.java
- Realisiert in der Klasse B_TCPNetworking.java.

Um die Datei vom Server zu lesen muss der GET Befehl verwendet werden. Auch wenn die Schreibverbindung vom Client zum Server bereits geschlossen ist, kann der Client weiter vom Server gesendete Daten empfangen.

- Für diese Aufgabe haben wir zwei Programme geschrieben.
 - i. und ii.: C_MoreTCPNetworking_Client.java und C_MoreTCPNetworking_Server.java
Wenn nach dem schließen der Schreibverbindung vom Client zu Server trotzdem wieder Daten in dieser Richtung gesendet werden sollen, passiert nichts: Java erzeugt keine Exception, wenn der OutputStream angesprochen wird, es werden aber auch keine Daten übertragen.
 - iii.: C_MoreTCPNetworking_Client_iii.java und C_MoreTCPNetworking_Server_iii.java
Die Laufzeit beträgt 5548 Millisekunden mit einer Datei der Größe von 104542 Bytes.
- Realisiert in den Klassen D_UDPNetworking.java und D_UDPNetworking_Server.java.
Die Laufzeit kann hier nicht gemessen werden. Es wird eine Angabe von 0 Millisekunden vom Programm erzeugt, dabei ist zu bemerken, dass die Erzeugung der DatagramPacket Objekte bereits vor Beginn der Messung durchgeführt wurde, und nur das reine Senden gemessen wurde. Die Laufzeit, d.h. Dauer bis zum Empfang der Pakete, kann nicht von einem der beiden Teilnehmer gemessen werden. Es wurden ebenfalls 104542 Bytes übertragen.

3 NFS

a) Nutzen und Eigenschaften von NFS

Das Network File System (NFS) von Sun dient dazu, Datei auf anderen Rechners zu manipulieren, ohne sie auf den eigenen Rechner übertragen zu müssen. Die Dateien können genau wie lokale Daten verändert werden, dabei werden Abfragen und Änderungen gegebenenfalls im Dateisystem des entfernten Rechners durchgeführt. Ursprünglich war NFS ein UDP Protokoll, mittlerweile gibt es auch TCP Implementierungen.

b) Dateizugriff

Damit ein Datenaustausch zwischen NFS-Server und Client stattfinden kann, muss der NFS-Server gestartet und beim Portmapper registriert sein.

- Client kontaktiert Portmapper und fragt nach dem Port des Mount-Daemons (mountd)
- Portmapper gibt Portnummer für mountd heraus
- Client kontaktiert mountd und fragt nach einem Filehandle für ein Verzeichnis
- mountd gibt ein Filehandle 0 zurück
- Client kontaktiert Portmapper und fragt nach dem Port für NFS (nfsd)
- Portmapper gibt Portnummer für nfsd heraus
- Client führt LOOKUP-Prozedur aus mit den Parametern Filehandle 0 und dem Dateinamen
- nfsd gibt Filehandle 1 für Datei heraus
- Client führt READ-Prozedur aus mit dem Parameter Filehandle 1
- nfsd gibt Inhalt der Datei zurück (Daten)

c) In der Version 4 wurde NFS vor allem um Sicherheitsfunktionen erweitert. Diese spielten bei der Entwicklung ältere Versionen noch keine große Rolle. NFS unterstützt in der Version 4 einen Benutzerauthentifizierung über Kerberos. Neben einer Erweiterung der Sicherheitsfunktionen wurde das System stärker von UNIX unabhängig gemacht. In der neuen Version ist eine Verwendung mit anderen Betriebssystemen leichter möglich. In diesem Rahmen entfiel auch die Verwendung eines Portmappers und andere Hilfsprotokolle. Die Funktionalität zum einbinden des entfernten Dateisystems ist jetzt Teil des Protokolls. Diese Änderungen vereinfachen die Verwendung in Verbindung mit einer Firewall.

4 SMTP, POP, IMAP

a) **Textbasiert** Das Protokoll verwendet ausschließlich Zeichen das ASCII-Zeichensatzes.

Nur PUSH-Übertragung Ein expliziter Abruf von Nachrichten ist nicht möglich. Es kann nur der Versand angestoßen werden.

Keine Authentifizierung Das ursprüngliche SMTP sieht keine Authentifizierung der Versender von Nachrichten vor.

Setzt auf TCP auf Das Protokoll verwendet zur Datenübertragung TCP.

Verlässlich Das Protokoll stellt einen verlässlichen Nachrichtenaustausch zwischen zwei Rechnern sicher.

b) **Authentifizierung** IMAP ermöglicht die Verwendung eines Challenge-Response-Verfahrens zur Benutzerauthentifizierung. Das Passwort wird dann nicht wie bei POP im Klartext übertragen, so dass Angriffe erschwert werden.

Mailverwaltung IMAP ermöglicht eine Verwaltung der Mails auf dem Server. Nachrichten können direkt auf dem Server gelöscht, verschoben, etc. werden. Darüber hinaus ist es möglich auf dem Server weitere Verzeichnisse zum Sortieren der Nachrichten anzulegen.

Statusinformationen Bei Verwendung von IMAP können Statusinformationen einer Mail zentral auf dem Server verwaltet werden. Es steht dann z. B. allen Mailclients des Nutzers die Information zur Verfügung ob eine Nachricht bereits gelesen wurde. Dies ist bei Verwendung des POP-Protokolls nicht möglich.

Gemeinsame Nutzung IMAP ermöglicht die gemeinsame Nutzung einer Mailbox durch mehrere Nutzer. Hierzu stehen auch Methoden zur Vergabe von Benutzerrechten zur Verfügung.

- c) Das Übertragen nicht druckbarer Zeichen in E-Mails wird durch den MIME-Standard geregelt. Die Zeichen werden dabei während der Übertragung durch eine Ersatzdarstellung aus in der Mail verwendbaren Zeichen ersetzt. Die Codierung kann dabei nach verschiedenen Standards umgesetzt werden.
- d) Das Dateiformat der Nachrichtenteile eine E-Mail im MIME-Format kann durch die Angabe im Inhaltstypfeld nach RFC 1049 durch dem Empfänger ermittelt werden. Die Verwendung des Feldes sowie die möglichen Dateitypen werden durch den MIME-Standard festgelegt.