

CS 580 Client-Server Programming
Spring Semester, 2009
Doc 13 Protocol, HTTP, POP
12 March, 2009

Copyright ©, All rights reserved. 2009 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/opl.shtml>) license defines the copyright on this document.

References

Hypertext Transfer Protocol - HTTP/1.0, Berners-Lee, Fielding, Nielson, rfc1945, <http://www.w3.org/Protocols/rfc1945/rfc1945>

Hypertext Transfer Protocol -- HTTP/1.1, Fielding, Gettys, Mogul, Masinter, Leach, Berners-Lee, rfc2616, <http://www.w3.org/Protocols/rfc2616/rfc2616.html>

Uniform Resource Identifiers (URI): Generic Syntax, Berners-Lee, Fielding, Masinter, rfc2396
<http://www.ietf.org/rfc/rfc2396.txt>

RFC 1939, <http://www.ietf.org/rfc/rfc1939.txt>

Reading

HTTP/1.0 rfc1945, <http://ftp.ics.uci.edu/pub/ietf/http/rfc1945.html>

Post Office Protocol RFC 1939, <http://www.ietf.org/rfc/rfc1939.txt>

Protocol

Requirements for a "good protocol"

Well defined

Complete

Parsable

Extendable

Available protocol document

Assignment 2 Protocol

Client Command	Server Response
login;screenName:foo;password:bar;;	ok:success;;
transmitMessage>Hello World;;	ok:success;;
transmitMessage>Hello \\2;;	ok:success;;
messages;block: l;;	ok:2; text>Hello \\2:sender:foo:time:02/03/2009 13\\:29\\:45; text>Hello World:sender:foo:time:02/03/2009 13\\:29\\:42;;
fuss;;	error:Invalid command f;;
quit;;	ok:quit;;

Well defined

Every bit of data sent in either direction has to have its place in the protocol description.

Protocol is a Language

Common formal description:

BNF and Augmented BNF

Format of the description language needs to be part of the protocol document.

Examples are important

Complete

The protocol must cover all possible situations.

Garbage data

Old client or server (different protocol versions)

Illegal requests

Boundary conditions

Etc.

Parsable

Both clients and servers are computer programs.

A computer program's IQ is generally 0.

Design goals

Distinct information packets or messages

Allow parsing independent of semantics

Consistency

Allow for code reuse

Flexibility

Allow parsing independent of semantics

Client Command	Server Response
login;screenName:foo;password:bar;;	ok:success;;
transmitMessage:Hello World;;	ok:success;;
transmitMessage:Hello \2;;	ok:success;;
messages;block:1;;	ok:2; text:Hello \2:sender:foo:time:02/03/2009 13\29\45; text:Hello World:sender:foo:time:02/03/2009 13\29\42;;
quit;;	ok:quit;;

How does
the server parse each set of commands?

The client parse each response

Available

Different groups may write clients and servers at different times.

Central registry for Internet protocols

Self regulating:

RFC - Request For Comment

IETF - Internet Engineering Task Force

Official:

ISO

ANSI

Protocol Types

Typical **synchronous**

Client sends request to server

Server responds with a reply

HTTP, POP, SMTP, GOPHER, XMODEM

Typical **asynchronous**

Client and server both send information to each other concurrently.

TELNET, RLOGIN, ZMODEM

A hybrid protocol is also possible

Protocol Design Issues

Protocol design is difficult!

Learn from examples

Some issues

Protocol extensibility and versioning

Byte order used for sending values

ASCII vs. Binary protocol

Synchronous vs. Asynchronous

State

Timeouts

HTTP

Stateless (http 1.0)

Assigned port 80

Basic Server-Client Interaction (http 1.0)

Client: Open connection

Server: Accept/Reject connection

Client: Send request

Server: Send response to request

Connection closed

HTTP Message Format

HTTP-message = Simple-Request (HTTP/0.9 messages)
| Simple-Response
| Full-Request (HTTP/1.0 messages)
| Full-Response

Full-Request = Request-Line
*(General-Header | Request-Header | Entity-Header)
CRLF
[Entity-Body]

Full-Response = Status-Line
*(General-Header | Request-Header | Entity-Header)
CRLF
[Entity-Body]

HTTP-header = field-name ":" [field-value] CRLF

Entity-Body = *OCTET

HTTP Full Request

Request-Line = Method SP URI SP HTTP-Version CRLF

rohan 13-> **telnet www.eli.sdsu.edu 80**

Trying 130.191.226.80...

Connected to www.eli.sdsu.edu.

Escape character is '^['.

GET /courses/fall00/cs580/index.html HTTP/1.0

2 CRLF's end the full request

HTTP/1.1 200 OK

Date: Tue, 05 Sep 2000 19:31:14 GMT

Server: Apache/1.3.9 (Unix) PHP/3.0.12

Last-Modified: Mon, 04 Sep 2000 21:03:56 GMT

ETag: "14c199-7e8-39b40e3c"

Accept-Ranges: bytes

Content-Length: 2024

Connection: close

Content-Type: text/html

X-Pad: avoid browser bug

<HTML>

<HEAD>

 <TITLE>CS 580: Course Web Site</TITLE>

... stuff removed here...

Connection closed by foreign host.

Positional Data verses Name-Value Pairs

1.0; CERN/3.0; Thursday, 21-Mar-96
17:00:45 GMT; text/html; 2686; Tuesday,
27-Feb-96 05:34:12 GMT

MIME-Version: 1.0
Server: CERN/3.0
Date: Thursday, 21-Mar-96 17:00:45 GMT
Content-Type: text/html
Content-Length: 2686
Last-Modified: Tuesday, 27-Feb-96 05:34:12 GMT

Which is more error prone?

Name-Value Pairs & Orderer

MIME-Version: 1.0

Server: CERN/3.0

Date: Thursday, 21-Mar-96 17:00:45 GMT

Content-Type: text/html

Content-Length: 2686

Last-Modified: Tuesday, 27-Feb-96 05:34:12 GMT

Server: CERN/3.0

Content-Type: text/html

MIME-Version: 1.0

Content-Length: 2686

Last-Modified: Tuesday, 27-Feb-96 05:34:12 GMT

Date: Thursday, 21-Mar-96 17:00:45 GMT

Adding new Fields

MIME-Version: 1.0

Server: CERN/3.0

Date: Thursday, 21-Mar-96 17:00:45 GMT

Content-Type: text/html

Forwarded: by <http://rohan.sdsu.edu/> for
cs.sdsu.edu

Content-Length: 2686

WhitneyInfo: Hi Mom

Last-Modified: Tuesday, 27-Feb-96 05:34:12 GMT

Name-Value Pairs are your Friends
Don't Program without them

How to Indicate the End of a Message

Use termination sequence

Make the length of the message known

HTTP uses both

Header ends in CRLF

Header contains length in bytes of message body

HTTP/1.0 200 Document follows

MIME-Version: 1.0

Server: CERN/3.0

Date: Thursday, 21-Mar-96 17:00:45 GMT

Content-Type: text/html

Content-Length: 2686

Last-Modified: Tuesday, 27-Feb-96 05:34:12 GMT

Detecting End of a Message

What if the terminating sequence is part of the message?

What if a HTTP header contains CRLF CRLF

POP3

Post Office Protocol

Purpose: Allow PC's, Macs, etc. to download mail from server

Port number 110

Protocol uses ASCII only

Stateful protocol

Multiple requests & responses on same connection

Format of commands to server

keyword blank argument1 [blank argumentk] CRLF

| keyword | = 3, 4 characters, no spaces

| argument | <= 40 characters, no spaces

keyword and arguments are separated by single space character

Server Response

Status keyword additionalInfo

Status is either "+OK" or "-ERR0.3."

A single line response ends in CRLF

If response requires more than one line:

- Each line ends in a CRLF

- The response ends in CRLF.CRLF

- If a line starts with a "." preprend a "." to it

When Client reads the first CRLF how does it know it is at the end of message?

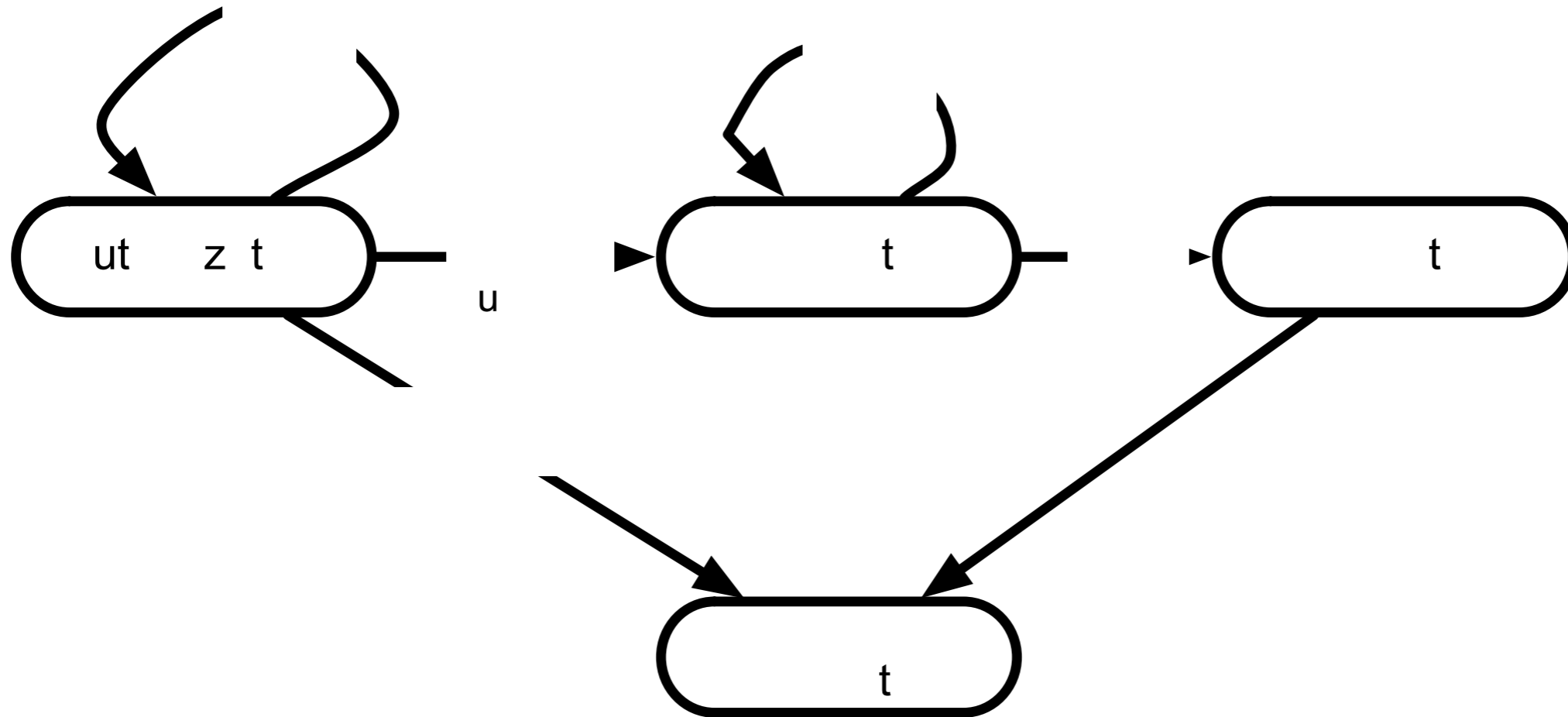
Timeouts

A POP3 server may have an autologout timer

A server must wait at least 10 minutes before timing out an idle client

The POP3 server on `cs.sdsu.edu` times out in 2 minutes

Client Connect States



Authorization State

Server acknowledges connection from client with

+OK "message"

+OK UCB Pop server (version 2.1.2-R3) at sciences.sdsu.edu starting.

Commands: USER, PASS, APOP, QUIT

USER PASS

Combination is used to progress to transaction state

USER must come first

PASS or QUIT must come after USER

Example

Ti 38->**telnet cs.sdsu.edu 110**

Trying 130.191.226.116...

Connected to cs.sdsu.edu.

Escape character is '^]'.

+OK QPOP (version 3.1.2) at sciences.sdsu.edu starting.

USER whitney

+OK Password required for whitney.

PASS typeYourPasswordHere

+OK whitney has 116 visible messages (0 hidden) in 640516 octets.

Transaction State

Commands: STAT, LIST, RETR, RSET, QUIT

STAT

Arguments: none

Returns "+OK" numberOfMessages SizeOfMail

STAT

+OK 22 45595

LIST

Arguments: a message-number (optional)

Returns: size of message in octets

Examples

LIST 2

+OK 2 3064

LIST

+OK 116 visible messages (640516 octets)

1 2980

2 3064 (message 3 - 116 deleted to save space)

116 1290

.

Transaction State

RETR 21

+OK 825 octets

Received: from [130.191.9.18] (ebb2p9.sdsu.edu [130.191.9.18]) by sciences.sdsu.edu (4.1/8.6.10) with SMTP id UAA29486 for <whitney@saturn.sdsu.edu>; Mon, 11 Mar 1996 20:16:07 -0800 (PST)

X-Sender: whitney@cs.sdsu.edu (Unverified)

Message-Id: <v02110100ad6aaaf097b6@[130.191.9.70]>

Mime-Version: 1.0

Content-Type: text/plain; charset="us-ascii"

Date: Mon, 11 Mar 1996 20:16:50 -0800

To: whitney@saturn.sdsu.edu

From: whitney@saturn.sdsu.edu (Roger Whitney)

Subject: Sample Mail

X-UIDL: 826604201.000

this is a test

..

the end

Roger Whitney

whitney@cs.sdsu.edu

<http://www.eli.sdsu.edu>

(619) 594-3535

(619) 594-6746 (fax)

Math & Computer Science Dept.

San Diego State University

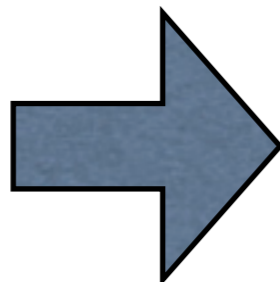
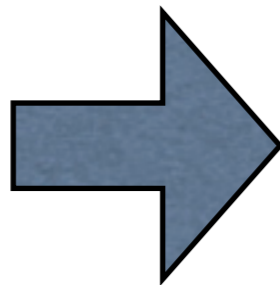
San Diego, CA 92182-7720

.

RETR

Arguments: a message-number

Returns: the message



Transaction State

DELE

Arguments: a message-number to delete

Returns: a confirmation of deletion

Marks a message to be deleted

NOOP

Arguments: none

Returns: a positive response

Does nothing

Why NOOP?

QUIT

Arguments: none

Returns: a positive response

Send POP3 server to UPDATE state

Update State

Updates mail box to reflect transactions taken during the transaction state, then logs user out

If session ends by any method except the QUIT command during the transaction state, the update state is not entered