

# ColdFusion Foundations: POP3



Mosh Teitelbaum

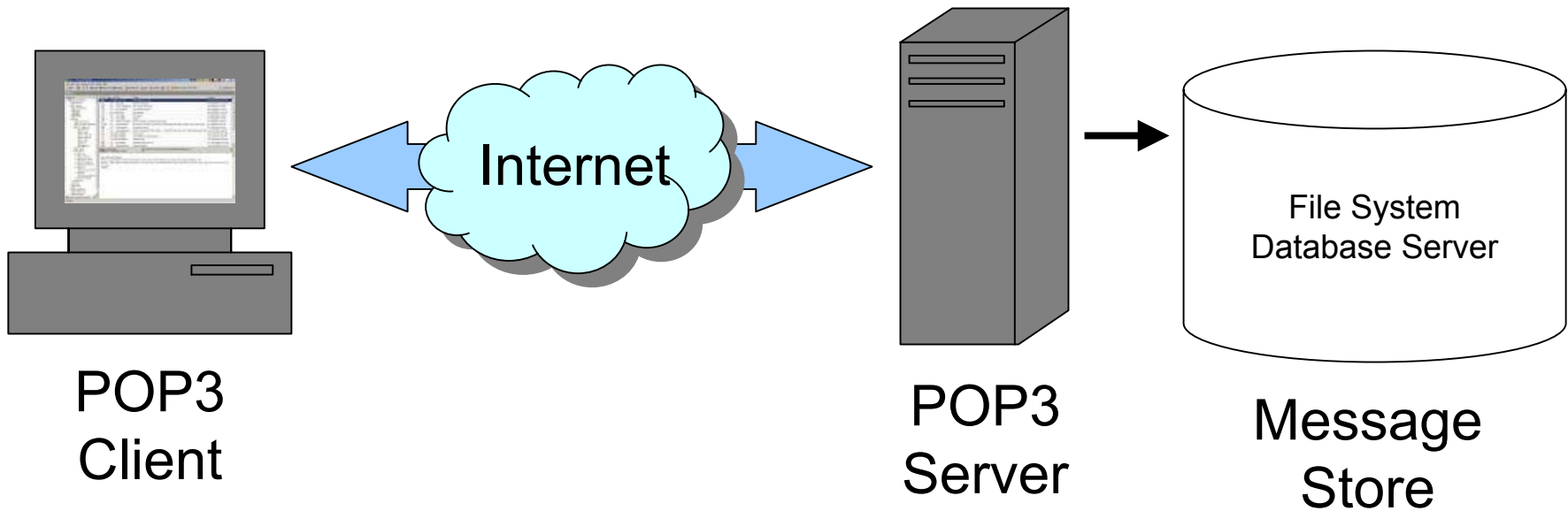
[mosh.teitelbaum@evoch.com](mailto:mosh.teitelbaum@evoch.com)

evoch, LLC

# POP3: Post Office Protocol – Version 3

- Purpose
  - To allow a workstation to retrieve mail - *RFC 1939*
- What is POP3?
  - A protocol that defines how email clients communicate with email servers to retrieve email messages.
- What does POP3 do?
  - It allows email messages to be retrieved from the email server. It does not support transmission of email messages by email clients.

# POP3 Involves Clients and Servers



# POP3 Communication Process

1. Session Initiation - Client establishes 2-way connection to server (port 110) which responds with welcome message
2. AUTHORIZATION State - Client sends identification and server responds with another message, acquires access to the user's mail store and enters the TRANSACTION state.
3. TRANSACTION State – Client initiates one or more transactions.
4. UPDATE State - Client initiates termination of connection and server updates mail store, sends a farewell message and terminates the connection.

# POP3 Commands and Responses

All client-server communication involves:

- Commands
  - Clients send commands to provide information and instructions to the server
  - Commands are usually 3-4 characters and are case-insensitive
- Responses
  - Servers respond with a status indicator and a keyword possibly followed by more information. Status indicators are “+OK” (positive) or “-ERR” (negative)
  - Single line responses end with a single CRLF
  - Multiple line responses end with a line consisting solely of a period and a CRLF

# POP3 Commands

Command	Description
QUIT	Initiates session termination
STAT	Requests a “drop listing” indicating number of messages and size of the mail store
LIST [msg]	Requests a “scan listing” for specified or all message(s) indicating message number and size of the message
RETR msg	Retrieves specified message
DELE msg	Marks specified message as deleted
NOOP	No operation
RSET	Resets initial state by unmarking deleted messages
TOP msg n	Retrieves specified message’s headers and top n lines of body
UIDL [msg]	Requests a “unique-id listing” for specified or all message(s) indicating message number and unique ID of the message
USER name	Specifies username for USER/PASS authentication
PASS password	Specifies password for USER/PASS authentication
APOP name digest	Specifies MD5-based authentication credentials



# POP3 Authorization State

- The Authorization state begins upon transmission of the 1-line welcome message:

```
+OK POP3 server ready
```

- Client must identify and authenticate itself:
  - USER/PASS – Plaintext authentication
  - APOP – MD5 digest “encryption”
  - AUTH – Alternate authentication mechanism (*RFC 1734*)
- If authentication fails, client can try again or may terminate the session via the QUIT command
- If authentication is successful, server enters Transaction state

# USER/PASS Authentication

- Plaintext authentication via username and password
- Simplest form of authentication but also the least secure

```
+OK POP3 server ready
USER cfugDemo@evoch.com
+OK cfugDemo@evoch.com
PASS cfugDemo123
+OK 0 messages 0 octets
```

```
+OK POP3 server ready
USER cfugDemo@evoch.com
+OK cfugDemo@evoch.com
PASS hack
-ERR Unknown user or incorrect password
```



# APOP Authentication

- Authentication via username and MD5 hashed password
- Server indicates APOP support by sending timestamp in welcome message

```
+OK POP3 server ready Wed, 18 Aug 2004 14:37:44 -0400  
➡ <20040818143744@email02.mywebmailserver.com>
```

- Digest is the password appended to the timestamp (including angle brackets) which is then run through the MD5 algorithm

```
+OK POP3 server ready Wed, 18 Aug 2004 15:05:27 -0400  
➡ <20040818150527@email02.mywebmailserver.com>  
APOP cfugDemo@evöch.com 786b5c12203b391c9a903b515ce65a12  
+OK 0 messages 0 octets
```

# AUTH Authentication

- Specified in RFC 1734, “POP3 AUTHentication Command,” to allow use of IMAP4 authentication mechanisms in POP3
- Client-specified authentication mechanism allowing for much more secure means of authentication

```
+OK POP3 server ready
AUTH KERBEROS_V4
+ AmFYig==
BAcAQU5EUkVXLkNNVS5FRFUAOCAsho84kLN3/IJmrMG+25a4DT
+nZImJjnTNHJUtxAA+o0KPKfHEcAFs9a3CL5Oebe/ydHJUwYFd
WwuQ1MWiy6IesKvjL5rL9WjXUb9MwT9bpObYLGOKi1Qh
+ or//EoAADZI=
DiAF5A4gA+oOIALuBkAAmw==
+OK Kerberos V4 authentication successful
```

# POP3 Transaction State

- The Transaction state begins when the client successfully authenticates and the server gains exclusive access to the mail store
- After gaining access, server assigns a message-number to each message which is good for the duration of the session
- Client may repeatedly issue any number of commands
- Each client command is followed by a server response
- After client issues the QUIT command, server enters UPDATE state.

# STAT Command

- The STAT command requests a “drop listing” of the server indicating number of messages and the size of the mail store
- Drop listings consist of a positive response code, a space, the number of messages, a space and the size of the maildrop

```
STAT  
+OK 2 2068
```

# LIST Command

- The LIST command requests a “scan listing” indicating message number and size of specified or all message(s)
- Drop listings consist of the message number, a space and the size of the message

```
LIST
+OK 2 messages 2068 octets
1 1015
2 1053
.
```

```
LIST 2
+OK 2 1053
```

# RETR Command

- The RETR command retrieves the specified message

```
RETR 1
+OK 1015 octets
From: "Mosh Teitelbaum" <mosh.teitelbaum@evoch.com>
To: <cfugDemo@evoch.com>
Subject: Test Message #1
Date: Wed, 18 Aug 2004 15:58:32 -0400
[... more headers ...]
```

```
12345
```

```
.
```

```
RETR 3
-ERR No such message
```



# DELE Command

- The DELE command marks the specified message for deletion
- The message is “deleted” from the current client session but is not actually removed from the message store until the UPDATE state
- Messages marked as deleted can be undeleted via RSET

```
DELE 1  
+OK Message deleted
```

# NOOP Command

- The NOOP command doesn't change anything
- Usually used to maintain an idle state without having the server terminate the connection from lack of activity

NOOP

+OK

# RSET Command

- The RSET command resets the session (i.e, undeletes all messages marked for deletion)

```
RSET
```

```
+OK
```

# QUIT Command

- The QUIT command terminates the session
- If issued in the Authorization state, server does not enter UPDATE state. If issued in the Transaction state, server enters UPDATE state.

```
QUIT
```

```
+OK POP3 server closing connection
```

# TOP Command

- The optional TOP command retrieves the headers and first  $n$  lines of the specified message

```
TOP 2 3
+OK 1053 octets
From: "Mosh Teitelbaum" <mosh.teitelbaum@evoch.com>
To: <cfugDemo@evoch.com>
Subject: Test Message #1
Date: Wed, 18 Aug 2004 15:58:32 -0400
[... more headers ...]
```

```
1st line
```

```
2nd line
```

```
3rd line
```

```
.
```

# UIDL Command

- The optional UIDL command requests a “unique-id listing” indicating current message number and permanent unique ID
- Unique-id listings consist of the message number, a space and the unique ID of the message

```
UIDL
+OK
1 20040818155839E5E3
2 20040818155912E640
.
```

```
UIDL 2
+OK 2 20040818155912E640
```



# POP3 Update State

- The Update state begins when the client issues the QUIT command from within the Transaction state
- In the Update state, the server deletes marked messages from the mail store, releases its exclusive access to the mail store, sends a farewell message to the client and terminates the connection

# <CFPOP>

Retrieves and/or deletes email messages from a POP mail server. Retrieved messages are placed in specified query. Most common attributes are below:

Attribute	Description
Server, Port	Optional. Specifies server and port to connect to.
Username, Password	Optional. Specifies username and password to use for authentication
Action	Optional. getHeaderOnly (default)   getAll   delete
Name	Required for getHeaderOnly and getAll. Name of the query object that contains the retrieved message info
messageNumber	Comma-delimited list of message numbers. Ignored if uid is specified.
uid	Comma-delimited list of Unique Ids. New in CFMX.
attachmentPath	Optional. Directory in which attachments should be saved

# <CFPOP> Query Columns

For actions *getHeaderOnly* and *getAll*, the query specified via the NAME attribute has the following columns:

Column	Description
Date, From, ReplyTo, Subject, CC, To	Standard email header values
MessageNumber	The message's MessageID
UID	The message's UniqueID
Body, TextBody, HtmlBody	First, text/plain, and text/html message parts, respectively. <i>getAll</i> only.
header	The message's header part. <i>getAll</i> only.
Attachments, AttachmentFiles	Tab delimited lists of the attachment names and locations (full path). <i>getAll</i> only.

# <CFPOP> Date Format

Date values returned via <CFPOP> are in UTC/GMT format

```
Thu, 19 Aug 2004 17:22:13 -0400
```

To convert to a standard ColdFusion date value in local time use:

```
<CFSCRIPT>
function getTimeStamp(httpTimeString) {
    // Build Time Stamp
    var tsParts = ListToArray(httpTimeString, " ");
    var timeStamp = "{ts '" & tsParts[4] & "-" &
        DateFormat("#tsParts[3]#/1/1970", "mm") & "-" & tsParts[2]
        & " " & tsParts[5] & "'}";

    // Convert to local time
    timeStamp = DateConvert("utc2Local", timeStamp);

    // Return timeStamp
    return timeStamp;
}
</CFSCRIPT>
```

# <CFPOP> Example: Get Message Headers

```
<CFPOP ACTION="getHeaderOnly"  
  NAME="getHeaders"  
  SERVER="server"  
  PORT="port"  
  USER="username"  
  PASSWORD="password">
```

# <CFPOP> Example: Get Message

```
<CFPOP ACTION="getAll"  
  NAME="getMessage"  
  SERVER="server"  
  PORT="port"  
  USER="username"  
  PASSWORD="password"  
  UID="20040818155912E640"  
  ATTACHMENTPATH="C:\Attachments\">
```



# <CFPOP> Example: Delete Message

```
<CFPOP ACTION="delete"  
SERVER="server"  
PORT="port"  
USER="username"  
PASSWORD="password"  
UID="20040818155912E640">
```

# POP Resources

- RFCs from <http://www.ietf.org/rfc/rfc#####.txt>:
  - rfc1939.txt – “Post Office Protocol - Version 3”
  - rfc2384.txt – “POP URL Scheme”
  - rfc2449.txt – “POP3 Extension Mechanism”
  - rfc1734.txt – “POP3 AUTHentication command”
  - rfc2195.txt – “IMAP/POP AUTHorize Extension for Simple Challenge/Response”
  - rfc3206.txt – “The SYS and AUTH POP Response Codes”
  - rfc2595.txt – “Using TLS with IMAP, POP3 and ACAP”
  - rfc1321.txt – “MD5 Algorithm”
  - rfc1521.txt – “MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies”
  - rfc2045.txt - “Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies”

# Closing

- Questions?
- Contact Info  
Mosh Teitelbaum  
evoch, LLC  
[mosh.teitelbaum@evoch.com](mailto:mosh.teitelbaum@evoch.com)  
<http://www.evoch.com/>