Post Office Protocol (POP3 Tutorial Sequence Diagram)		
Internet		**EventHelix.com/EventStudio 2.5**
POP3 Client	POP3 Server	
Mail Client	POP3 Port 110	01-Apr-05 07:56 (Page 1)

This sequence diagram was generated with EventStudio 2.5 (http://www.EventHelix.com/EventStudio).

This tutorial explains the operations of a POP3 (Post Office Protocol) e-mail exchange via a
sequence diagram. All the message interactions between the POP3 client and server for retrieving
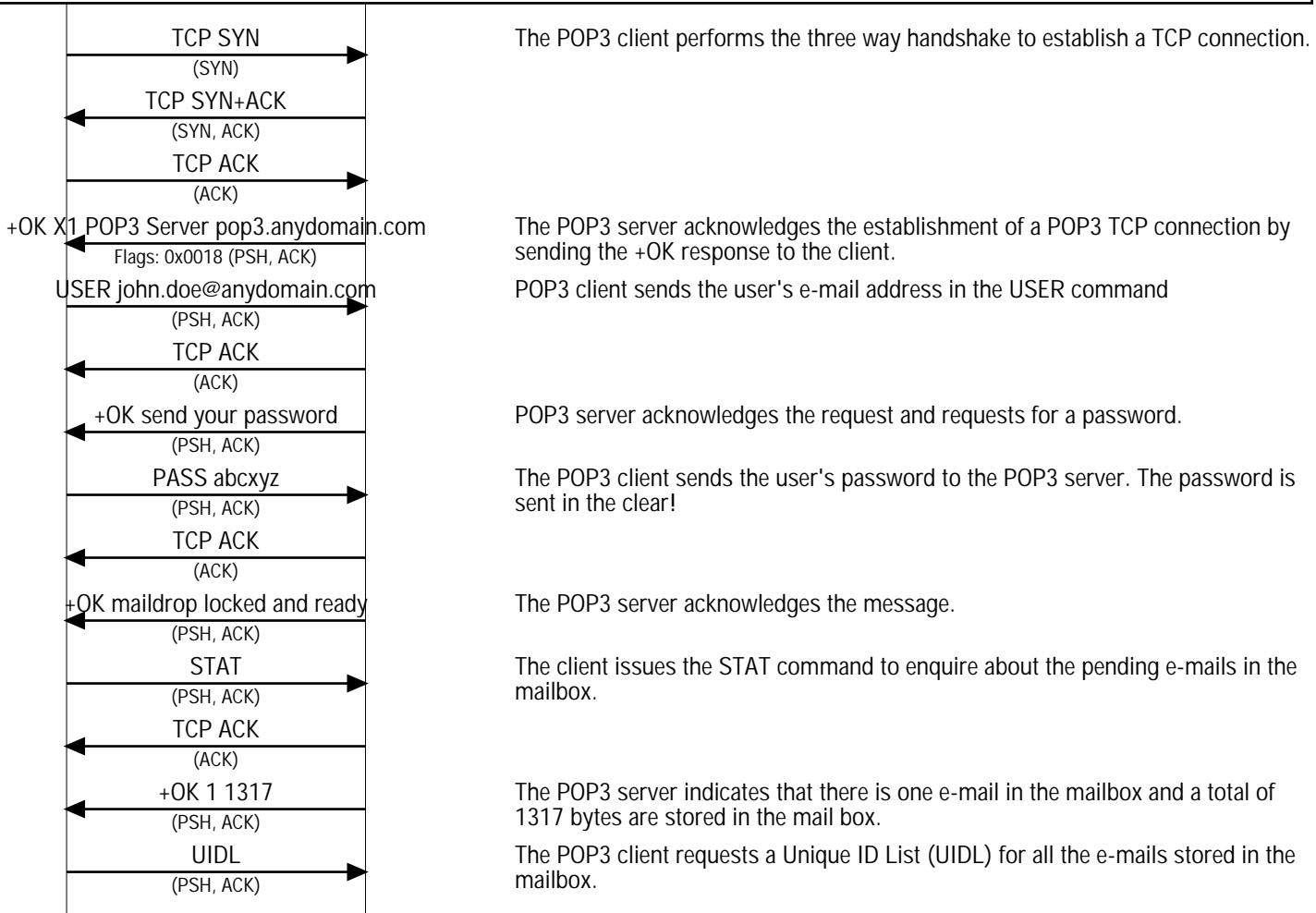one e-mail message from a POP3 mailbox have been described.

The message flow described here corresponds to the following communication between a POP3 client
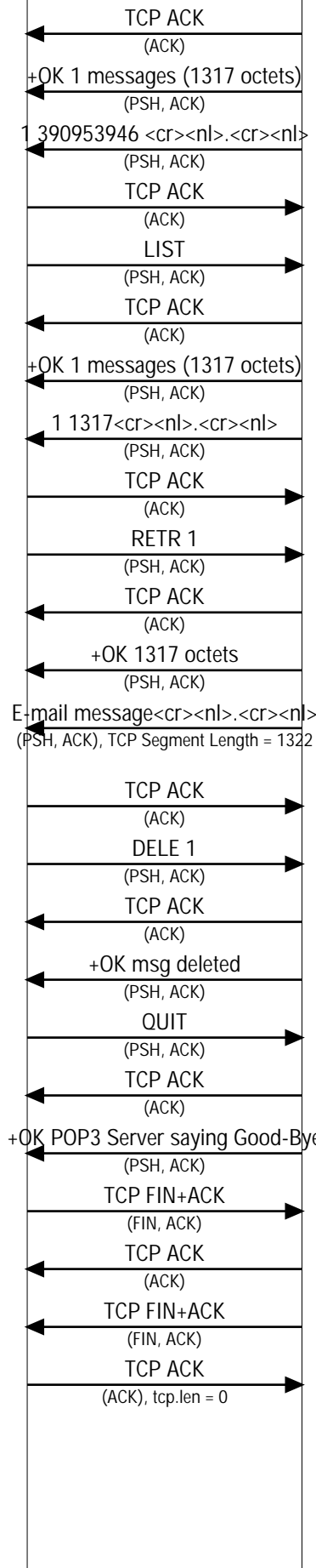and server:

```
Server: +OK X1 POP3 Server pop3.anydomain.com
Client: USER john.doe@anydomain.com
Server: +OK send your password
Client: PASS abcxyz
Server: +OK maildrop locked and ready
Client: STAT
Server: +OK 1 1317
Client: UIDL
Server: +OK 1 messages (1317 octets)
       1 390953946
       .
Client: LIST
Server: +OK 1 messages (1317 octets)
       1 1317
       .
Client: RETR 1
Server: +OK 1317 octets
Server: E-mail contents
       .
Client: DELE 1
Server: +OK msg deleted
Client: QUIT
Server: +OK POP3 Server saying Good-Bye
```

Copyright © 2005 EventHelix.com Inc. All Rights Reserved.

TCP SYN → (SYN)	The POP3 client performs the three way handshake to establish a TCP connection.
← **TCP SYN+ACK** (SYN, ACK)	
TCP ACK → (ACK)	
← **+OK X1 POP3 Server pop3.anydomain.com** Flags: 0x0018 (PSH, ACK)	The POP3 server acknowledges the establishment of a POP3 TCP connection by sending the +OK response to the client.
USER john.doe@anydomain.com → (PSH, ACK)	POP3 client sends the user's e-mail address in the USER command
← **TCP ACK** (ACK)	
← **+OK send your password** (PSH, ACK)	POP3 server acknowledges the request and requests for a password.
PASS abcxyz → (PSH, ACK)	The POP3 client sends the user's password to the POP3 server. The password is sent in the clear!
← **TCP ACK** (ACK)	
← **+OK maildrop locked and ready** (PSH, ACK)	The POP3 server acknowledges the message.
STAT → (PSH, ACK)	The client issues the STAT command to enquire about the pending e-mails in the mailbox.
← **TCP ACK** (ACK)	
← **+OK 1 1317** (PSH, ACK)	The POP3 server indicates that there is one e-mail in the mailbox and a total of 1317 bytes are stored in the mail box.
UIDL → (PSH, ACK)	The POP3 client requests a Unique ID List (UIDL) for all the e-mails stored in the mailbox.

TCP ACK
(ACK)

+OK 1 messages (1317 octets)
(PSH, ACK)

The POP3 server indicates that it has an e-mail with ID 1 and length 1317 bytes.

1 390953946 <cr><nl>.<cr><nl>
(PSH, ACK)

The unique id associated with the message is sent. A dot on the next line signifies end of UIDL list.

TCP ACK
(ACK)

LIST
(PSH, ACK)

The POP3 client issues a LIST command for all the messages.

TCP ACK
(ACK)

+OK 1 messages (1317 octets)
(PSH, ACK)

The POP3 server responds to the list command with the total number of messages and size of the mailbox.

1 1317<cr><nl>.<cr><nl>
(PSH, ACK)

The POP3 server sends the list entry number and the size of the message in the mailbox. The ASCII DOT (.) signifies the end of the listing.

TCP ACK
(ACK)

RETR 1
(PSH, ACK)

The client requests the server to retrieve message 1. This is a request to get message 1 from the mailbox.

TCP ACK
(ACK)

+OK 1317 octets
(PSH, ACK)

The POP3 server acknowledges the request and indicates that a 1317 byte message follows.

E-mail message<cr><nl>.<cr><nl>
(PSH, ACK), TCP Segment Length = 1322

The 1317 byte e-mail message is transported in one TCP segment. The actual segment length is 1322 as it includes the <cr><nl>.<cr><nl> sequence at the end. Again, the ASCII DOT (.) specifies the end of the message.

TCP ACK
(ACK)

DELE 1
(PSH, ACK)

The e-mail has now been stored in the e-mail client so it is requests a delete for e-mail message 1 from the mailbox.

TCP ACK
(ACK)

+OK msg deleted
(PSH, ACK)

The POP3 server acknowledges the delete the of the message to the client.

QUIT
(PSH, ACK)

The POP3 client sends QUIT command to initiate the release of the session.

TCP ACK
(ACK)

+OK POP3 Server saying Good-Bye
(PSH, ACK)

The server acknowledges the QUIT command.

TCP FIN+ACK
(FIN, ACK)

The POP3 client sends FIN to signal the release of the client side of the TCP connection.

TCP ACK
(ACK)

Server acknowledges FIN.

TCP FIN+ACK
(FIN, ACK)

The POP3 server sends FIN to signal the release of the server side of the TCP connection.

TCP ACK
(ACK), tcp.len = 0

Server acknowledges FIN.