

Network Working Group
Internet-Draft
Expires: December 7, 2006

A. Melnikov
Isode Ltd
C. Wilson
Nokia
June 5, 2006

**IMAP4 Extensions for Quick Reconnect
draft-ietf-lemonade-reconnect-07.txt**

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on December 7, 2006.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This document defines a quick reconnect IMAP4 extension, which gives a mobile client the ability to resume a previously abandoned session, without the need to perform a full synchronization sequence. The goal is to minimize the amount of data transferred at login after a dropped connection.

Changes since [draft-ietf-lemonade-reconnect-06](#)

- o Multiple spelling and editorial corrections from Arnt, Randy and Dave Cridland.
- o Renamed FOLDER response to SELECTED and replaced "()" with nothing (i.e. the mailbox name is now optional).
- o Deleted the requirement to support at least 5 per-user resumable sessions. This is a deployment choice and the client can't rely that there is always a number of sessions available to it.
- o Define generic syntax for LOGOUT parameters consistent with [RFC 4466](#).

Changes since [draft-ietf-lemonade-reconnect-05](#)

- o Fixed some formatting issues.
- o Clarified what state is to be associated with a session. Clarified interaction with IMAP ACL extension.
- o Clarified that creation of a new session when there is an active one makes the older one inactive. Added DELETEDSID response in order to convey this information.
- o Clarified that the server can still return NO if per-user limit on number of sessions is reached and all of them are active.
- o Added paragraph clarifying that any session can only be active on a single TCP connection at a time.
- o Clarified that if (on session resumption) UIDVALIDITY provided by the client doesn't match the server's value, than the mailbox is selected in the same read-write mode as it was originally selected, but all other state must be forgotten.
- o Specified that the second (and subsequent) request to create a new session is treated as the first one. (Behavior change from -05)
- o Added the DELETESID command that allows to terminate a no longer needed resumable session without the need to close the connection.

Changes since [draft-ietf-lemonade-reconnect-04](#)

- o Spelling/wording corrections from Zoltan.

- o Replaced extended Login/Authenticate with a new SID command. This made the document much simpler.
- o Removed the NOSID untagged response as the SID command can return the tagged NO response.
- o Made UIDVALIDITY/modification sequence parameters to the SID command required if the SID parameter is also specified.
- o Made UNSEEN untagged OK response optional.
- o Specified that LOGOUT (PRESERVE) should be just treated as LOGOUT if there is no active session. [Zoltan]
- o Specified that the second (and subsequent) request to create a new session is treated as NOOP.
- o Clarified that when the server reaches the global limit on number of sessions, then it returns NO to the SID command.
- o Clarified that when the server reaches the per-user limit on number of sessions, then it expires the oldest inactive session belonging to the same user.
- o Updated examples and ABNF.
- o Changed [section 3](#) around to be clearer.
- o Added new text to security section on New SID flooding.
- o Completed IANA Considerations section.

Changes since [draft-ietf-lemonade-reconnect-03](#)

- o Clarified which response codes/responses are required on successful AUTHENTICATE/LOGIN.
- o FOLDER untagged response code is not needed if NEWSID parameter was specified.
- o One of tagged READ-WRITE/READ-ONLY is required if resuming a session in selected state. Updated examples as the result.
- o Spelling/wording corrections from Eric.

Changes since [draft-ietf-lemonade-reconnect-02](#)

- o Removed minimal session expiration timeout.
- o Moved EXPUNGED response to a separate draft. Updated the document as a result of this change.
- o Updated references and completed IANA considerations.
- o Added description of the updated synchronization sequence.

Changes since [draft-ietf-lemonade-reconnect-01](#)

- o Added NOSID response.
- o Added UIDVALIDITY argument to the SID parameter.
- o Added EXPUNGED response.
- o Added the following example: an opened mailbox was deleted and another was renamed on top, while the client was offline.
- o Various editorial changes, mostly rewriting text for clarity.

Changes since [draft-ietf-lemonade-reconnect-00](#)

- o Added example protocol traces.
- o Added interaction with SASL-IR (SASL initial response extension).
- o SIDs are now server generated. Added NEWSID parameter to LOGIN/AUTHENTICATE and NEWSID response.
- o Removed SESSION response. The FOLDER response can now tell if there is a mailbox selected or not.
- o Added LOGOUT (PRESERVE).

Table of Contents

| | | |
|----------------------|--|--------------------|
| 1. | Requirements notation | 6 |
| 2. | Introduction and Overview | 6 |
| 2.1. | Typical IMAP Session without Reconnect | 7 |
| 2.2. | Reconnect IMAP Session | 9 |
| 3. | Session Control | 10 |
| 3.1. | SID command | 10 |
| 3.2. | Creating a new session | 12 |
| 3.3. | Resuming an existing session | 13 |
| 3.4. | DELETESID command | 17 |
| 4. | PRESERVE parameter to the LOGOUT command | 18 |
| 5. | New responses | 18 |
| 5.1. | SELECTED Response | 18 |
| 5.2. | NEWSID Response | 19 |
| 5.3. | DELETEDSID Response | 19 |
| 6. | Formal Syntax | 19 |
| 7. | Security Considerations | 20 |
| 8. | IANA Considerations | 21 |
| 9. | References | 21 |
| 9.1. | Normative References | 21 |
| 9.2. | Informative References | 22 |
| | Authors' Addresses | 23 |
| | Intellectual Property and Copyright Statements | 24 |

1. Requirements notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

"Resumable session" (or just "session") refers to the entire sequence of client/server interaction from the initial session creation during SID command (with no parameters) until the session is explicitly terminated (or deleted) upon client request or until the session expires some time after disconnect. A resumable session can be terminated by the [[RFC3501](#)] LOGOUT or DELETESID command. Note that a resumable session doesn't terminate when the connection is lost or closed with LOGOUT (PRESERVE) command (see [Section 4](#)). Also note that this term has a different meaning from the term "session" used by [[RFC3501](#)].

A resumable session is said to be "inactive" if it is not associated with any active TCP connection.

The term "event" is as defined in [[C2S](#)].

2. Introduction and Overview

The IMAP4 protocol does not distinguish between a session that has been disconnected for 15 seconds and one that has been disconnected for 1 week. The assumption is the same: The client and server may be completely out of sync and the client must ensure that it is completely synchronized with the server.

In today's mobile environments, accidental disconnects due to environmental factors (battery life, signal strength, etc.) can be quite common. In these situations, the time between the disconnect and reconnect can be quite short. It is these situations which we are interested in.

If the time between connections is sufficiently short, an assumption can be made that the state of the last connection is still valid and that the client can avoid having to query much of the state data at reconnect.

[[anchor3: Note to RFC editor: This document is compliant with "transitional IMAP capabilities" document [TRANS-CAPA]. Please change the capability name below to "RECONNECT"]]

The quick reconnect IMAP extension is present if an IMAP4 server returns "X-DRAFT-W07-RECONNECT" as one of the supported capabilities to the CAPABILITY command. Note, that this extension **REQUIRES** support for the [[CONDSTORE](#)] and the [[EXPUNGED](#)] IMAP extensions, so it **MUST** be announced in the CAPABILITY response as well.

The text below outlines how the quick reconnect extension is to be used. The description assumes that the quick reconnect is mainly used when the client closes the TCP connection (intentionally or accidentally) and tries to reconnect within a preset amount of time.

1. The client authenticates and asks the server to create a new session by sending a SID command. The server generates a new Session ID (SID) and returns it to the client.
2. The client performs operations as normal, typically selecting a mailbox and accessing messages.
3. At some point the client loses connectivity. The server retains all information about the client's session, including information about the currently selected mailbox, for some amount of time.

(case a) The client reconnects before the session timeout.

It authenticates as the same user as before and specifies the previously used session ID in the SID command. The client and server begin at the same mailbox as before the connection loss.

(case b) The client reconnects after the session timeout.

It authenticates as the same user and specifies the previously used session ID in the SID command. Since the session has timed out on the server, the server has no record of this session and creates a new session. This behaviour falls back to a normal IMAPv4 login sequence.

2.1. Typical IMAP Session without Reconnect

A typical IMAP client session looks as follows: (Modified dump from network sniffer)

```
S: * OK [CAPABILITY IMAP4REV1 LOGIN-REFERRALS STARTTLS
    AUTH=DIGEST-MD5] imap.mail.com IMAP4rev1 2001.315rh at Thu,
    15 Jul 2004 11:47:49-0400 (EDT)
C: 2 AUTHENTICATE DIGEST-MD5
S: + cmVhbG09ImVsd29vZC5pbm5vc29mdC5jb20iLG5vbmNlPSJPQTZNRz10
    RVFhbTJoaCIscW9wPSJhdXRoIixhbGdvcm10aG09bWQ1LXNlc3MsY2hh
    cnNldD11dGYtOA==
C: Y2hhcnNldD11dGYtOCx1c2VybmFtZT0iY2hyaXMiLHJlYWxtPSJlbHdvb2
```



```
Quaw5ub3NvZnQuY29tIixub25jZT0iT0E2TUc5dEVRR20yaGgiLG5jPTAw
MDAwMDAXLGNub25jZT0iT0E2TUhYaDZWcVRyUmsiLGRpZ2VzdC11cmk9Im
ltYXAvZWx3b29kLmlubm9zb2Z0LmNvbSIscmVzcG9uc2U9ZDM4OGRhZDkw
ZDRiYmQ3NjBhMTUyMzIxZjIxNDNhZjcscW9wPWF1dGg=
S: + cnNwYXV0aD1lYTQwZjYwMzM1YzQyN2I1NTI3Yjg0ZGJhYmNkZmZmZA==
C:
S: 2 OK [CAPABILITY IMAP4REV1 NAMESPACE] User chris
authenticated
C: 3 namespace
S: * NAMESPACE (("" "/" )("#mhinbox" NIL)("#mh/" "/" ) (( "~" "/" )
  (( "#shared/" "/" )("#ftp/" "/" )("#news." ".")("#public/" "/" )))
S: 3 OK completed
C: 4 lsub "" "*"
S: 4 OK LSUB completed
C: 5 lsub "" "#mhinbox*"
S: 5 OK LSUB completed
C: 6 lsub "" "#mh/*"
S: * NO /home/test8/.mh_profile not found, mh format names disabled
S: 6 OK LSUB completed
C: 7 lsub "" "~*"
S: 7 OK LSUB completed
C: 8 lsub "" "#shared/*"
S: 8 OK LSUB completed
C: 9 lsub "" "#ftp/*"
S: 9 OK LSUB completed
C: 10 lsub "" "#news.*"
S: 10 OK LSUB completed
C: 11 lsub "" "#public/*"
S: 11 OK LSUB completed
C: 12 list "" "INBOX"
S: * LIST (\NoInferiors) NIL INBOX
S: 12 OK LIST completed
C: 15 select "INBOX"
S: * 464 EXISTS
S: * 3 RECENT
S: * OK [UNSEEN 12] Message 12 is first unseen
S: * OK [UIDVALIDITY 3857529045] UIDs valid
S: * OK [UIDNEXT 550] Predicted next UID
S: * FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
S: * OK [PERMANENTFLAGS (\Deleted \Seen \*)] Limited
S: 15 OK [READ-WRITE] SELECT completed
C: 16 UID fetch 1:500 (FLAGS)
S: * 1 FETCH (UID 41 FLAGS (\Seen))
S: * 2 FETCH (UID 43 FLAGS (\Seen \Flagged))
S: ...
S: * 464 FETCH (UID 541 FLAGS (Junk))
S: 16 fetch completed
C: ...
```


(Note, that the client has recognized the CAPABILITY response code in the initial OK response, it has omitted the CAPABILITY command.)

The above dump was generated using Ethereal and it reported that the above process generated just under 20 KB of data.

The above protocol sequence simply logged the user into the account, selected the INBOX mailbox and downloaded the updated flags for a subset of messages.

2.2. Reconnect IMAP Session

Now let's look at the same mailbox, but see how the protocol would react if the client had dropped the connection and was reconnecting within a few minutes.

As in the previous section, the client has recognized the CAPABILITY response code in the initial OK response, so it has omitted the CAPABILITY command.

```
S: * OK [CAPABILITY IMAP4REV1 LOGIN-REFERRALS STARTTLS
    AUTH=DIGEST-MD5] imap.example.com IMAP4rev1 2001.315rh at Thu,
    15 Jul 2004 11:47:49-0400 (EDT)
C: 2 AUTHENTICATE DIGEST-MD5
S: + cmVhbG09ImVsd29vZC5pbm5vc29mdC5jb20iLG5vbmNlPSJPQTZNRz10
    RVFHbTJoaCIscW9wPSJhdXRoIixhbGdvcml0aG09bWQ1LXNlc3MsY2hh
    cnNldD11dGYt0A==
C: Y2hhcnNldD11dGYt0Cxc2VybmFtZT0iY2hyaXMiLHJlYWxtPSJlbHdvb2
    QuaW5ub3NvZnQuY29tIixub25jZT0iT0E2TUc5dEVRR20yaGgiLG5jPTAw
    MDAwMDAxLGNub25jZT0iT0E2TUhYaDZwcVRyUmsiLGRpZ2VzdC11cmk9Im
    ltYXAvZWx3b29kLmlubm9zb2Z0LmNvbSIscW9wPSJhdXRoIixhbGdvcml0aG09bWQ1LXNlc3MsY2hh
    ZDRiYmQ3NjBhMTUyMzIxZjIxNDNhZjcscW9wPWF1dGg=
S: + cnNwYXV0aD1lYTQwZjYwMzM1YzQyN2I1NTI3Yjg0ZGJhYmNkZmZmZA==
C:
S: 2 OK [CAPABILITY IMAP4REV1 NAMESPACE MAILBOX-REFERRALS
    CONDSTORE X-DRAFT-I01-EXPUNGED X-DRAFT-W07-RECONNECT] User chris
    authenticated
C: 3 SID P1234567890 56789 20010715194045000 41,43:211,214:541
S: * SELECTED INBOX
S: * 464 EXISTS
S: * 3 RECENT
S: * OK [UIDVALIDITY 3857529045] UIDVALIDITY
S: * OK [UIDNEXT 550] Predicted next UID
S: * OK [HIGHESTMODSEQ 20010715194045007]
S: * FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
S: * OK [PERMANENTFLAGS (\Deleted \Seen \*)] Limited
S: * 1 FETCH (UID 1 FLAGS (\Seen))
S: * EXPUNGED 205,207,209,215:321
S: 3 OK [READ-WRITE] session continued
C: ...
```

That's it! And since no new messages arrived, no headers were downloaded. The total byte count is about 500 bytes, significantly less than the amount of data transferred during a normal connect request.

3. Session Control

3.1. SID command

Arguments: optional SID parameter
 optional mailbox UIDVALIDITY
 optional last known modification sequence
 optional list of known UIDs

Responses: OPTIONAL untagged responses: NEWSID (REQUIRED if no SID parameter was specified; also required if SID parameter was specified but the given session ID is invalid)
 OPTIONAL untagged responses: SELECTED (REQUIRED if SID parameter was specified)
 The following responses are REQUIRED if the client remains in the selected state upon reconnect (See [Section 3.3](#)).
 Untagged responses: FLAGS, EXISTS, RECENT
 OK untagged responses: PERMANENTFLAGS, UIDNEXT, UIDVALIDITY, HIGHESTMODSEQ
 OPTIONAL OK untagged responses: UNSEEN
 OK tagged responses: READ-WRITE or READ-ONLY (as described in [section 6.3.1 \[RFC3501\]](#))

Result: OK - session resumed or created; see [Section 3.3](#)
 NO - session creation failure, e.g. session number limit exceeded
 BAD - arguments invalid

This section defines the new SID command. This command is only available in authenticated and selected IMAP states.

If the optional session identifier (SID) parameter is not provided, the server will generate a new session as described in [Section 3.2](#).

If the optional SID parameter is defined, the server MUST attempt to locate and resume the session with the given SID as described in [Section 3.3](#).

Whether a session was created or resumed, the server is now required to remember the session state associated with the (username, SID) pair.

At minimum this state includes:

- o The currently selected mailbox
- o If mailbox is selected read-only or read-write
- o The number of recent messages (RECENT)
- o Non persistent flags created while mailbox was opened, if the server supports not persistent IMAP flags

- o Any state associated with the current IMAP connection, but is not the state of the selected mailbox

Note that if the server also supports [\[ACL\]](#) extension it can either cache the MYRIGHTS value or can refresh it on reconnect. In the latter case the server SHOULD send the untagged MYRIGHTS response on reconnect.

Note that other IMAP extensions can define additional state that has to be preserved.

The following state doesn't have to be preserved by the server, because it is either supplied by the client on reconnect, or the server is required to send it to the client on reconnect:

- o The mailbox UIDVALIDITY value
- o The mailbox HIGHESTMODSEQ value [\[CONDSTORE\]](#)
- o The highest message UID used in the mailbox
- o The total number of messages (EXISTS)

An implementation may choose to keep all session state in memory and expire sessions some time after client disconnects due to dropped TCP connection or explicit execution of LOGOUT (PRESERVE) command. Upon session expiration the session ID becomes invalid and its associated state is deleted. While there is no required minimum expiration timeout, server implementations are encouraged to support as long a time as possible, subject to denial-of-service and related concerns.

Note that SIDs are chosen by the server and are opaque to the client. The same SID used with two different usernames refers to two different sessions, unless the two usernames refer to the same user account, e.g. one username is an alias for another.

[3.2.](#) Creating a new session

This section describes the server behavior if the client has not specified a session identifier parameter to the SID command. This instructs the server to create a new session regardless if a previous, active resumable session is available. If an active session is already available and a new one can be created, the existing session becomes inactive and can be resumed at a later time.

This operation can be performed at any point during the IMAP session.

Example: The client issues a SID command without a SID parameter, the server creates a new session and responds with a new NEWSID response:

```
C: A01 SID
S: * NEWSID ABC412423rs
S: A01 OK session created
```

In order to limit Denial of Service attacks a server implementation SHOULD have a per-user limit on maximum number of resumable sessions. When this limit is reached the server SHOULD expire the oldest inactive (not associated with an active TCP connection) session associated with the same user. If no inactive session associated with the same user is available, the server MUST return the tagged NO response to the SID command.

A server implementation might also have a per-server limit on maximum number of resumable sessions. This document doesn't define this limit. If the server has reached the limit, it MUST return the tagged NO response to the SID command.

3.3. Resuming an existing session

This section describes server behaviour if the client has specified the SID parameter to a SID command.

The SID command takes four arguments: the session identifier, the mailbox UIDVALIDITY, the last known modification sequence and the optional list of known UIDs.

The server verifies all arguments for syntactic validity. If any parameter is not syntactically valid, the server return the tagged BAD response.

After that the server checks the session identifier (SID). If the SID is not recognized (for example, the SID is invalid because it is either not known to the server or has expired), the server creates a new session as described in [Section 3.2](#).

If the SID is valid, the server MUST inform the client of the state of the session by sending an untagged SELECTED response. The SELECTED response contains the name of the currently selected mailbox name or nothing if there is no mailbox selected. Note that the mailbox name MUST be reported verbatim as originally specified by the client in SELECT/EXAMINE. I.e. before server applies any mailbox name canonicalization (like converting "Inbox" or "users.andrew" to "INBOX"). If, before the disconnect, the session had a mailbox selected and the mailbox has been deleted since then (or the user no

longer has permissions to access it), the server MUST return a special SELECTED untagged response that contains no mailbox name. This tells the client that the session has returned to the unauthenticated state.

Note that a session can only be active on a single TCP connection. An attempt to resume an active session on a different TCP connection forces the server to close the session on the other connection and to resume it on the current connection. The other connection is notified about this with the untagged DELETEDSID response (See [Section 5.3](#)).

If at this point there is no mailbox selected, the remaining arguments MUST be ignored by the server.

Example: Resuming an old session.

```
C: A02 SID ABC412423rs 67890007 20050715194045000
S: * SELECTED
S: ...
S: A02 OK session resumed, but no selected mailbox
```

UIDVALIDITY Parameter:

Once the session has successfully resumed and there is a mailbox selected, the server checks the UIDVALIDITY value provided by the client. If the provided UIDVALIDITY doesn't match the UIDVALIDITY for the currently selected mailbox, the server MUST send the NEWSID response containing the SID of the resumed session. This is done in order to force the client to perform full resynchronization. In this case the server SHOULD [[anchor7: (MUST? What about non permanent flags?)]] clear all state associated with the mailbox, except for whether the mailbox was selected read-only or read-write and the remaining arguments to the SID command are ignored. After that the server behaves as if SELECT/EXAMINE command for the mailbox was issued (depending on whether the current mailbox was originally selected read-write or read-only), i.e. it returns other untagged responses like EXISTS, UIDVALIDITY untagged OK response, etc.

Example: Attempting to resume an old session with an invalid UIDVALIDITY parameter.

```
C: A02 SID ABC412423rs 67890007 20050715194045000 41,43:
211,214:541
S: * SELECTED INBOX
S: * NEWSID ABC412423rs
S: * 464 EXISTS
S: * 3 RECENT
S: * OK [UIDVALIDITY 3857529045] UIDVALIDITY
```



```
S: * OK [UIDNEXT 550] Predicted next UID
S: * OK [HIGHESTMODSEQ 20060128194045007]
S: A02 OK [READ-WRITE] resync required
```

Note that even though the UIDVALIDITY is invalid, the mailbox remained selected and the untagged SELECTED response is returned as the result.

Modification Sequence and UID Parameters:

If the provided UIDVALIDITY matches that of the selected mailbox, the server then checks the last known modification sequence.

The server sends the client any pending flag changes (using FETCH responses that MUST contain UIDs) and expunges that have occurred in this mailbox since the provided modification sequence.

If the list of known UIDs was also provided, the server should only report flag changes and expunges for the provided messages. If the client doesn't provide the list of UIDs, the server acts as if the client has specified "1:*". Thus, the client can process just these pending events and need not perform a full resynchronization. The result of this step is semantically equivalent to the client issuing:

```
tag1 UID FETCH "known-uids" (FLAGS) (CHANGEDSINCE
"mod-sequence-value" REPORTEXPUNGES)
```

If the server was unable to cache all events since the MODSEQ and hence is unable to send them to the client, the server MUST return the NEWSID response to signal that the client should perform a state-comparison based synchronization. Note, that the NEWSID response MAY contain the same SID as specified in the SID parameter. [Section 3.2](#) describes server behavior in this case, i.e. the server behaves as if the session can't be resumed.

Example: Resuming an old session in selected state with the INBOX selected.

```
C: A02 SID ABC412423rs 67890007 20060115194045000 41,43:211,214:
541)

S: * SELECTED INBOX

S: * 314 EXISTS

S: 15 RECENT
```



```
S: * OK [UIDVALIDITY 67890007] UIDVALIDITY
S: * OK [UIDNEXT 567] Predicted next UID
S: * OK [HIGHESTMODSEQ 20060115205545359]
S: * 49 FETCH (UID 117 FLAGS (\Seen \Answered))
S: * 50 FETCH (UID 119 FLAGS (\Draft $MDNSent))
S: ...
S: * 100 FETCH (UID 541 FLAGS (\Seen $Forwarded))
S: * EXPUNGED 41,43:116,118,120:211,214:540
S: A02 OK session resumed
```

If, while the client was offline, the selected mailbox was deleted and another mailbox was renamed to the same name, the server MAY do either:

1. return the NEWSID response, as well as the new UIDVALIDITY and all other responses as described in [Section 3.1](#).
2. return "SELECTED" response with no mailbox in order to return the client to the authenticated state.

Example: Resuming an old session in selected (read-only) state with the INBOX selected.

```
C: A02 SID ABC412423rs 67890007 20010715194045000 41,43:211,214:541
S: * SELECTED INBOX
S: * OK [UIDVALIDITY 67890007] Current UIDValidity
S: * OK [UIDNEXT 782] Next available UID
S: * 114 EXISTS
S: * 5 RECENT
S: * OK [HIGHESTMODSEQ 20010715235047007] S: * FLAGS (\Answered
\Flagged \Draft \Deleted \Seen)
S: * OK [PERMANENTFLAGS (\Answered \Flagged \Draft
\Deleted \Seen \*)] Permanent flags
S: * 49 FETCH (UID 117 FLAGS (\Seen \Answered))
S: ...
S: A02 OK [READ-ONLY] session resumed
```

Example: Resuming an old session in selected state with the INBOX selected, but where the server could not cache all the events since the specified modification sequence. The server sends the NEWSID

response, which means that the client should perform full resynchronization.

```
C: A02 SID ABC412423rs 67890007 20010715194045000 41,43:211,214:541
S: * NEWSID jwejei9qe3r
S: * SELECTED INBOX
S: * OK [UIDVALIDITY 67890007] Current UIDValidity
S: * OK [UIDNEXT 1598] Next available UID
S: * 117 EXISTS
S: * 20 RECENT
S: * OK [HIGHESTMODSEQ 20010715235047019] S: * FLAGS (\Answered
\Flagged \Draft \Deleted \Seen)
S: * OK [PERMANENTFLAGS (\Answered \Flagged \Draft
\Deleted \Seen \*)] Permanent flags
S: A02 OK [READ-WRITE] session created
```

Example: Resuming an old session in selected state with the mailbox Drafts selected, but the mailbox was deleted and another one was renamed to become Drafts, while the client was offline. The server chooses to keep the Drafts mailbox open, however it sends the NEWSID response, which means that the client should perform full resynchronization.

```
C: A02 SID Frt-egf-779 6896834 11 20:35
S: * NEWSID Frt-egf-779
S: * SELECTED Drafts
S: * OK [UIDVALIDITY 1098183472] Current UIDValidity
S: * OK [UIDNEXT 12] Next available UID
S: * OK [HIGHESTMODSEQ 54] S: * 7 EXISTS
S: * 0 RECENT
S: * FLAGS (\Answered \Flagged \Draft \Deleted \Seen)
S: * OK [PERMANENTFLAGS (\Answered \Flagged \Draft
\Deleted \Seen \*)] Permanent flags
S: A02 OK [READ-WRITE] session created
```

Note that as one of READ-WRITE/READ-ONLY tagged response codes is required when a session is resumed in selected state, the server is unable to send the CAPABILITY response code in the tagged OK response.

3.4. DELETESID command

Arguments: none

Responses: no specific responses for this command

Result: OK - session successfully terminated
 NO - session deletion failure
 BAD - arguments invalid or no active session

The DELETESID command tells the server to terminate the current session. If there is no current session, the server MUST respond with the tagged BAD response.

4. PRESERVE parameter to the LOGOUT command

[[anchor8: This section clearly assumes voluntarily disconnect]]

This section updates the description of the LOGOUT command found in [Section 6.1.3 of \[RFC3501\]](#). A LOGOUT command with no parameters tells the server to terminate the current session (if any).

This document also adds a new parameter "PRESERVE" to the LOGOUT command. This tells the server that it should not terminate the current resumable session. If the client doesn't want to terminate the current session on logout it SHOULD use "LOGOUT (PRESERVE)" instead of just dropping the TCP connection. This helps to avoid unnecessary resource consumption on the server caused by a TCP disconnect.

Note that if "LOGOUT (PRESERVE)" is issued when no resumable session was created, the server should treat this as if the LOGOUT command was issued with no parameters.

Example: C: A023 LOGOUT (PRESERVE)

S: * BYE IMAP4rev1 Server logging out, state preserved

S: A023 OK LOGOUT completed

(Server and client then close the connection)

5. New responses

5.1. SELECTED Response

Contents: name of the selected mailbox or nothing if there is no
 selected mailbox

The SELECTED response tells the client that a session was successfully resumed and returns the name of the selected mailbox. If the response doesn't contain a mailbox name, this means that there

is no mailbox selected.

5.2. NEWSID Response

Contents: generated session ID

The NEWSID response tells the client the unique identifier that can be used to resume the created session. It also tells the client to perform a full state resynchronization.

5.3. DELETEDSID Response

Contents: session ID

The DELETEDSID response tells the client that the current active session is closed because it was resumed (transferred) on another TCP connection.

6. Formal Syntax

The following syntax specification uses the Augmented Backus-Naur Form (ABNF) notation as specified in [\[ABNF\]](#).

Non-terminals referenced but not defined below are as defined by [\[RFC3501\]](#), [\[CONDSTORE\]](#), [\[EXPUNGED\]](#) or [\[IMAPABNF\]](#).

Except as noted otherwise, all alphabetic characters are case-insensitive. The use of upper or lower case characters to define token strings is for editorial clarity only. Implementations **MUST** accept these strings in a case-insensitive fashion.

```
sid           = "SID" [SP session-id SP uidvalidity SP
                      mod-sequence-value [SP known-uids]]

uidvalidity   = nz-number

known-uids    = sequence-set
               ;; sequence of UIDs, "*" is not allowed

deletesid     = "DELETEDSID"

uidvalidity   = nz-number

known-uids    = sequence-set
               ;; sequence of UIDs, "*" is not allowed
```



```
command-auth      =/ sid / deletesid

logout            = "LOGOUT" [logout-params]

logout-params     = SP "(" logout-param-data ")"
                  ;; modifies the original IMAP4 LOGOUT
                  ;; command to accept optional parameters

logout-param-data= logout-param *(SP logout-param)

logout-param      = logout-param-name [SP logout-param-value]
                  ;; parameters to LOGOUT may contain one or
                  ;; more atoms atoms and/or strings and/or lists.

logout-param-name= tagged-ext-label

logout-param-value= tagged-ext-val
                  ;; This non-terminal shows recommended syntax
                  ;; for future extensions.

logout-preserve   = "PRESERVE"
                  ;; conforms to logout-param syntax

mailbox-data      =/ selected-resp / newsid-resp / deletedsid-resp

selected-resp     = "SELECTED" [SP folder]
                  ;; lack of the mailbox parameter denotes
                  ;; that there is no mailbox selected

newsid-resp       = "NEWSID" SP session-id

deletedsid-resp   = "DELETEDSID" SP session-id

session-id        = string ;; a unique session identifier,
                  ;; implementation specific.
```

7. Security Considerations

In order to mitigate DoS attacks caused by a client that tries to generate numerous resumable sessions (for example by issuing multiple SID commands with no parameters on the same TCP connection), a server implementation should consider imposing a limit on the maximum number of resumable sessions associated with a particular user and/or coming from a particular IP address/network. Once this limit has been reached, the oldest inactive session must be destroyed and the new one can be created. Inactive sessions SHOULD have an expiration timeout.

The Quick Reconnect extension does not make any modifications to the authentication sequence of the current IMAP4 protocol [[RFC3501](#)].

8. IANA Considerations

IMAP4 capabilities are registered by publishing a standards track or IESG approved experimental RFC. The registry is currently located at:

<http://www.iana.org/assignments/imap4-capabilities>

This document defines the X-DRAFT-W07-RECONNECT [[anchor15: Fix capability name upon publication]] IMAP capability. IANA is requested to add this capability to the registry.

9. References

9.1. Normative References

- [ABNF] Crocker, D., Ed. and P. Overell, Ed., "Augmented BNF for Syntax Specifications: ABNF", [RFC 4234](#), October 2005.
- [ACL] Melnikov, A., "IMAP4 Access Control List (ACL) Extension", [RFC 4314](#), December 2005.
- [CONDSTORE] Melnikov, A. and S. Hole, "IMAP Extension for Conditional STORE", November 2003.
- [EXPUNGED] Melnikov, A., "IMAP4 extension to CONDSTORE for reporting messages expunged since last synchronization", January 2005.
- [IMAPABNF] Melnikov, A. and C. Daboo, "Collected Extensions to IMAP4 ABNF", [RFC 4466](#), April 2006.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", [RFC 3501](#), March 2003.
- [RFC4234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", [RFC 4234](#), October 2005.

9.2. Informative References

- [C2S] Maes, S. and C. Wilson, "Lemonade Requirements for Server to Client Notifications", February 2005.

Authors' Addresses

Alexey Melnikov
Isode Ltd
5 Castle Business Village
36 Station Road
Hampton, Middlesex TW12 2BX
UK

Email: Alexey.Melnikov@isode.com

Corby Wilson
Nokia
5 Wayside Rd.
Burlington, MA 01803
USA

Email: corby@computer.org

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2006). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

