LEMONADE Working Group   S. Maes
Internet-Draft   Oracle
Expires: June 7, 2006   G. Parsons
                                                               Nortel
                                                     December 4, 2005

Realization of OMA Mobile Email (MEM) Architecture using Internet Mail
draft-ietf-lemonade-oma-mem-realization-00.txt

Status of this Memo

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

Copyright Notice

Abstract

This document specifies a realization of the architecture for the mobile email enabler (MEM) as specified by the OMA, using Internet Mail protocols.

Conventions used in this document

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 RFC 2119 [1].

Table of Contents

1. Introduction

This document describes a realization of OMA mobile email enabler (MEM) using Internet Mail protocols defined by the IETF. Many of these protocols have been enhanced by the LEMONADE work group for use in the mobile environment and are summarized in the LEMONADE profile [4]. This document shows how the requirements captured in OMA MEM Requirement document [3] and mechanisms of the OMA MEM Architecture [2] are satisfied.

This document contains the current view of the work. It refers to stable specifications and work in progress. As the work progress, it is expected that this document will evolve and be updated accordingly.

<Editor's note: As a result some of the statements about some specification may not yet supported by the references. When it is the case, these specifications are expected to be be updated.>

<Editor's note: Caveat to be removed when work stabilizes.>

2. OMA MEM Requirement document

The OMA MEM activity has collected a set of use cases and derived requirements for a mobile email enabler (MEM). the resulting work is summarized in OMA MEM Requirement document [3]. Some requirements relates to email protocols, some involve other OMA technologies outside the scope of IETF and some relate to implementations and normative interoperability statements for clients and servers.

In the OMA MEM RD, Mobile email is scoped as an email enabler (i.e. an OMA specification) optimized to support email usage in mobile devices and wireless networks. The RD focuses on requirements for the enabler specifications rather than for particular implementations of those. The RD also does not design the solution.

So, mobile email is defined as the set of enabling technologies that facilitate end-to-end application level interoperable email transactions (e.g. submission, retrieval, notification etc) to and from mobile devices. It targets both consumer and corporate mobile e-mail. The focus of the OMA MEM enabler is between the client and the server; not beyond. Interworking / interoperability among email clients is ensured by ensuring interworking with the internet email [8] as defined by IETF.

2.1. OMA MEM use cases

OMA RDs do not expect an exhaustive list of use cases. The one covered in the OMA MEM Requirement document [3] include:

- o Receiving an Email on the go. Some highlights:

  - * This use case describes a situation where new e-mail arrives in e-mail servers

  - * It is received on mobile client quasi-instantaneously based on settings preferences (whole e-mail or portions of e-mail; adapted or not)

  - * The use case targets enterprise and Non-enterprise users

  - >

- o Receiving a email server event on the go. Some highlights:

  - * The use case describes a situation where email events are received from the server

- Viewing email attachments on the go. Some highlights:

  - This use case describes a situation involving attachment adaptation and transcoding (known by server or requested by client)

  - It also discusses viewing the attachment (in whole or portion by portion)

- Sending emails on the go. Some highlights:

  - This use case describes a situation where the user sends an e-mail from teh correct SMTP server

  - It also discusses offline and intermittent connectivity

- Filtering rule changes while mobile. Some highlights:

  - This use case describes a situation where the user dynamically changes a filter (e.g. adds a sender from who notifications / new e-mails should be received)

- Data synchronization between clients. Some highlights:

  - This use case analyzes a situation of 3-way pair wise synchronization (e.g. Phone, laptop and server)

- Email with Attachment. Some highlights:

  - This use cases provides details about attachments manipulation including:

    - Selectively downloading part or all of attachments if not yet downloaded

    - Downloading more

- Forwarding Email without Downloading Attachments. Some highlights:

  - This case discusses forwarding without download with server-side re-composition including edits

  - It includes selection of what attachment to forward

- Configuring additional email accounts to be accessed. Some highlights:

  - This case discusses supporting multiple e-mail accounts possibly from different e-mail service providers.

- Replying to messages that are retrieved from different accounts. Some highlights:

  - This case discusses determining what account / server to use to send e-mail

- Client Email Events. Some highlights:

  - This use case describes a situation where e-mail client events are reflected to the server

  - It also discusses offline and intermittent connectivity

- Filtering Rules. Some highlights:

  - It discusses filtering rules on emails and events.

- Replying or Forwarding to Emails 'On the Go'. Some highlights:

  - This use cases describes how the user can edit text and attachments

  - It includes editing any email portion including addresses

  - It considers sending to teh server only the edit differences

- Configuring Auto-Reply Message. Some highlights:

  - This use case describes setting auto-reply for accounts while mobile.

2.2. OMA MEM requirements

The OMA MEM Requirement document [3] contains requirements about:

- High-Level Functional Requirements

- Security Requirements

- Charging Requirements

- Administration and Configuration Requirements

- Usability Requirements

- Interoperability Requirements

- Privacy Requirements

- Overall System Requirements

3. OMA MEM Architecture

This section gives a brief introduction to the OMA MEM Architecture.

3.1. OMA MEM logical Architecture

The OMA MEM activity has derived a logical architecture from the requirements and use cases described in [3]. It is represented in Figure 1, where arrows indicate content flows.

```
                       __________
                      | Other    |
                  ----| Mobile   |<---
                  |   | Enablers |   |
                  |   |__________|   |
                  |ME-4              |ME-3
                 _v____           ___v____        ________
                |      |         |        |      |        |
                | MEM  |ME-1     |  MEM   |  I2  |  Email |
                |Client|<------->| Server |<---->| Server |
                |______|     ME-2|________|      |________|
                                     ^
                                     |ME-5
                                     |
```

Figure 1: OMA MEM logical architecture

It identifies the following elements:

- The MEM client which implements the client-side functionality of the OMA Mobile Email Enabler. It is also responsible for providing the mobile email user experience and interface to the user and storing the email and data to be sent to the MEM server when not connected.

- The MEM server which implements the server-side functionality of the OMA Mobile Email Enabler (MEM).

- The MEM protocol between the MEM Client and MEM Server. It is responsible for all the data exchanges other than server to client event notifications that take place between the MEM client and server in order to update the MEM client with email server changes, the email server with changes in the MEM client and to send new email from the email server. Server to client notifications of email server events can be transported via the MEM protocol. We then speak of inband notifications. Note that there are no client to server notifications.

- o Other OMA enablers are needed to directly support the mobile email enabler. They are out of scope of IETF but they may include support for:

  - * Client provisioning and management for over the air installation of the MEM client on the device, provisioning of its settings and revocation

  - * Messaging enablers for outband notification, where outband notifications that are server to client event exchanges not transported by the MEM protocol but via other channels. Such channels may involve:

    - + SMS including GSMSMS or WAP WDP a la EMN

    - + MMS

    - + WAP Push

    - + SIP Push

    - + Additional outband notifications like SIP push or SIP event notify

    - + UDP notifications

    - + ...

OMA identifies different interfaces:

- o ME-1: MEM client interface to interact via the MEM protocol with the MEM server

- o ME-2: Corresponding interface of the MEM server

- o ME-3: Outband MEM server interfaces (e.g. to support generation of server to client notifications).

- o ME-4: Outband MEM client interfaces (e.g. to receive server to client notifications).

- o ME-5: Interface for management of MEM enabler server settings, user preferences and filters (globally and per account). <Editor's note: ME-5 is proposed in input contribution and not yet agreed by OMA. Mention of ME-5 will be removed or updated based on the disposition of teh proposal.>

It should be noted a single logical MEM server is identified. OMA

MEM activity may support realizations not based on Internet Mail protocols defined by the IETF that may not require distinguishing email submissions from other MEM server functions.

In addition, the MEM server and MEM client may interfaces to other mobile enablers.

Non-intrinsic functions can be provided by other enablers to enforce service providers policies like:

- Charging for the traffic or usage

- Content conversion / transcoding

The MEM server enables an email server. In a particular implementation, the email server may be packaged with (internal to it) the MEM server or be in a separate component. In such cases, interfaces to the email server are out of scope of the OMA MEM specifications. In the present document, we focus on the case where the backend consists of IETF LEMONADE enhanced IMAP and Submit servers. However, relationship to other cases are also discussed. The I2 interface is an OMA notation to designate protocol / interfaces that are not specified by the MEM enabler but may be standardized elsewhere.

3.2. OMA MEM Deployment Issues

The OMA MEM Architecture document [2] further identifies deployment models.

Certain of these deployment models are not what IETF has conventionally modeled. They require special attention to end-to-end security aspects and may warrant introduction of additional security measures (e.g. object level encryption).

3.2.1. OMA MEM proxy

The OMA MEM Architecture document [2] identifies OMA MEM server proxies as server components that may be deployed ahead of firewalls to facilitate traversal of firewalls.

Both IMAP and SMTP generally are compatible with proxies between the client and the server. Such proxies may disrupt end-to-end encryption, with the transport-level encryption ending at the proxy and re-generating from the proxy to the server. Again this may require additional security measures like object level encryption.

3.2.2. OMA MEM deployment cases

OMA MEM identifies that each component (MEM client, MEM servers, Other enablers and email server) may be deployed in different domains, possibly separated by firewalls and other network intermediaries. MEM proxies may be involved in front of firewall that protects the MEM server domain.

OMA MEM target support of configurations where:

- All components are within a same domain (Mobile operator)

- MEM client and other enablers are in the mobile operator domain. A MEM proxy is involved, deployed by the email service provider (in front of its domain)and MEM server and email server are in the domain of the email service provider

- MEM client and other enablers as well as a MEM proxy are in the mobile operator domain, MEM server and email server are in the domain of the email service provider

- MEM client and other enablers are in the mobile operator domain, a MEM proxy is in a third party service provider domain and MEM server and email server are in the domain of the email service provider

- MEM client, other enabler and MEM server are in the mobile operator domain and email server is in the domain of the email service provider

- MEM client and other enablers are in the mobile operator domain, MEM server is in a third party service provider domain and the email server is in the domain of the email service provider

3.3. OMA MEM Required Technical Capabilities

The OMA MEM Architecture document [2] analyzes the OMA MEM requirement [3] and presents a list of technical capabilities required to satisfy them.

In Section 7 of this document, after review of how OMA MEM can be realized with OMA and the resulting architecture, we describe how each technical feature within scope of IETF are supported. We also identify items left to OMA for specification

<Editor's note: Support of some of this is based on work in progress. The document tracks the current thoughts on how features will be supported. This may change as works evolves.>

4. IETF LEMONADE Architecture

This section gives a brief introduction to the LEMONADE Architecture.

The IETF LEMONADE activity has derived a LEMONADE profile [4] with the logical architecture represented in Figure 2, where arrows indicate content flows.

```
                            ______________
                           |              |
                  _________| Notification |
                 |         | Mechanism    |
                 |         |______________|
                 |Notif.              ^
                 |Protocol            |
                 |                 ___|______
                 |                |          |                 _____
               __v__    IMAP      | LEMONADE |      ESMTP     |     |
              |     |<----------->| IMAP     |<---------------| MTA |
              | MUA |-            | Store    |                |_____|
              |_____| \           |__________|
                       \               |
                        \              |URLAUTH
                         \SUBMIT       |
                          \        ____v_____
                           \      |          |                 _____
                            \     | LEMONADE |      ESMTP     |     |
                             ---->| Submit   |--------------->| MTA |
                                  | Server   |                |_____|
                                  |__________|
```

Figure 2: LEMONADE logical architecture

The LEMONADE profile [4] assumes:

- IMAP protocol [5] including LEMONADE profile extensions [4]
- SUBMIT protocol (SMTP [7], ...) including LEMONADE profile extensions
- LEMONADE profile compliant IMAP store. connected to MTA (Mail Transfer Agent) via ESMTP [6]
- LEMONADE profile compliant Submit server. connected to MTA via ESMTP
- Lemonade profile message store / submit server protocols (URLAUTH) (see [4]).

- o Outband server to client notifications relying on external notification mechanisms (and notification protocols) that may be out of scope of the LEMONADE profile.

- o A LEMONADE aware MUA (Mail User Agent). While use of outband notification is described in the LEMONADE profile, support for the underlying notifications mechanisms/protocols is out of scope of the LEMONADE specifications.

Further details on the IETF email protcol stack and architecture can be found in [8]

Note that in Figure 2 the IMAP server and Submit server are represented connected to MTAs (Mail Transfer Agents) via ESMTP [6]. This is not really essential. It could as well be X.400 so long as the message is in the store in an onternet form.

OMA MEM identifies other functionalities. These are considered as out of scope of the LEMONADE work and will need to be specified by OMA MEM.

5. Realization of OMA MEM Architecture with LEMONADE

This section details the use of the LEMONADE profile [4] to support the OMA MEM Architecture.

5.1. Relationship between the OMA MEM and LEMONADE logical architectures

Figure 3 illustrates the mapping of the IETF LEMONADE logical architecture on the OMA MEM logical architecture.

```
                          _____________________
                         | Other_Mob. Enablers |
                         | |--------------|    |
                  _________| Notification |    |
                 |       | | Mechanism    |    |
                 |       | |______________|    |
                 |Notif. |____________^________|
                 |Protocol      ______|__________
            ME-4 |             |   ___|_ME-3_    |
              ___|____         |  |          |   |         _____
             | __v__ |  IMAP   |  | LEMONADE |   |  ESMTP |     |
             ||     |<----------->| IMAP     |<-----------| MTA |
             || MUA ||   ME-2a |  | Store    |   |        |_____|
             ||_____||\ME-1    |  |__________|   |
             | MEM   | \       |       |         |
             | Client|  \      |       |URLAUTH  |
             |_______|   \SUBMIT       |         |
                          \    |   ____v_____    |
                           \   |  |          |   |         _____
                            \  |  | LEMONADE |   |  ESMTP |     |
                             ---->| Submit   |----------->| MTA |
                         ME-2b |  | Server   |   |        |_____|
                               |  |__________|   |
                               |MEM        Email |
                               |Server     Server|
                               |_________________|
                                        ^
                                        |ME-5
                                        |
```

Figure 3: Mapping of LEMONADE logical architecture onto the OMA MEM logical architecture.

As described in Section 4, the LEMONADE profile assumes LEMONADE profile compliant IMAP stores and Submit servers. Because the LEMONADE profile extends the IMAP store and the submit server, the

mobile enablement of email provided by the LEMONADE profile is directly provided in these server. Mapped to OMA MEM logical architecture, for the case considered and specified by the LEMONADE profile, the MEM server and email server logically combined. They are however split into distinct LEMONADE message store and LEMONADE submit server. ME-2 consists of two interfaces ME-2a and ME-2b associated respectively to IMAP extended according to the LEMONADE profile and SUBMIT extended according to the LEMONADE profile.

The MUA is part of the MEM client.

External notifications mechanism can be part of the other OMA enabler specified by OMA (or other activities).

5.2. LEMONADE realization of OMA MEM with non-LEMONADE compliant servers

The OMA MEM activity is not limited to enabling Lemonade compliant servers. It explicitly identifies the need to support other backends.

5.2.1. LEMONADE realization of OMA MEM with non-LEMONADE enhanced IMAP servers

Figure 4 illustrates the case of IMAP servers that are not (yet) LEMONADE compliant / enhanced with LEMONADE. In such case, the I2 interface between the MEM server components and the IMAP store and submit server are IMAP and SUBMIT.

```
                 ______________
                |              |
       _________| Notification |
      |         | Mechanism    |
      |         |______________|
      |Notif.           ^
      |Protocol         |
      |               ___|______         _____________
      |              | LEMONADE |        |             |        _____
    __v__    IMAP    | MEM      |  IMAP  |NON-LEMONADE | ESMTP |     |
   |     |<--------->|Enabler   |<------>|IMAP         |<----->| MTA |
   | MUA |\   ME-2a  | Server   |        |Store        |       |_____|
   |_____| \         |__________|        |_____________|
            \             |
             \            |URLAUTH
              \SUBMIT     |
               \      ____v_____         _____________
                \    |          |        |             |        _____
                 \   | LEMONADE | SUBMIT |NON-LEMONADE | ESMTP |     |
                  -->|  MEM     |        |Submit       |       |     |
                     | Enabler  |------->|Server       |------>| MTA |
              ME-2b  | Server   |        |             |       |_____|
                     |__________|        |_____________|
```

Figure 4: Architecture to support non-LEMONADE enhanced IMAP servers with a LEMONADE realization of OMA MEM enabler.

5.2.2. LEMONADE realization of OMA MEM with non-IMAP servers

Figure 5 illustrates the cases where the message store and submit servers are not IMAP store or submit servers. They may be POP3 servers or other proprietary message stores.

```
                 ______________
                |              |
       _________| Notification |
      |         | Mechanism    |
      |         |______________|
      |Notif.            ^
      |Protocol          |
      |               ___|______          _____________
      |              | LEMONADE |        |             |        _____
    __v__    IMAP    | MEM      |    I2  |Proprietary  | ESMTP |     |
   |     |<--------->|Enabler   |<------->|Message      |<----->| MTA |
   | MUA |\   ME-2a  | Server   |        |Store        |       |_____|
   |_____| \         |__________|        |_____________|
            \             |
             \            |URLAUTH
              \SUBMIT     |
               \      ____v_____          _____________
                \    |          |        |             |        _____
                 \   | LEMONADE |    I2  |Proprietary  | ESMTP |     |
                  -->| MEM      |        |Submit       |       |     |
                     | Enabler  |------->|Server       |------>| MTA |
              ME-2b  | Server   |        |             |       |_____|
                     |__________|        |_____________|
```

Figure 5: Architecture to support non-IMAP servers with a LEMONADE realization of OMA MEM enabler.

I2 designates proprietary adapters to the backends. They may invoved functions performed in the message stores or submit server as well as in the MEM enabler server.

6. Filters and server to client notifications and LEMONADE

OMA MEM RD [3] and AD [2] emphasize the need to provide mechanisms for server to client notifications of email events and filtering. Figure 6 illustrates how notification and filterings are introduced in LEMONADE profile [4].

```
                   ______________
                  |              |
         _________| Notification |
        |         | Mechanism    |
        |         |______________|
        |Notif.              ^
        |Protocol -------\  _|_
        |   ______|    ___\>|NF|____
        |  |          |     ----    |                 _____
      __v__|   IMAP   |__  LEMONADE |___   ESMTP   __|     |
     |     |<-------->|VF| IMAP     |DF |<--------|AF| MTA |
     | MUA |\   ME-2a |--  Store    |-^-           --|_____|
     |_____| \        |_____________| |
            \_\_______________|_______|
               \              |URLAUTH
                \SUBMIT       |
                 \        ____v_____
                  \      |          |                 _____
                   \     | LEMONADE |      ESMTP     |     |
                    ---->| Submit   |--------------->| MTA |
                ME-2b    | Server   |                |_____|
                         |__________|
```

Figure 6: Filtering mechanism defined in LEMONADE architecture

In Figure 6, four categories of filters are defined:

- AF: Administrative Filters - Set up by email service provider. AF are typically not configured by the user and set to apply policies content filtering, virus protection, spam filtering etc...

- DF: Deposit Filters - Filters that are executed on deposit of new emails. They can be defined as SIEVE filters [9]. They can include vacation notices.

- VF: View Filters - Filters that define which emails are visible to the MUA. View filters can be defined as virtual folders [10] as described in [11].

- o NF: Notification Filters - Filters that define for what email server event an outband notification is sent to the client.

The filters are manageable from the MUA:

- o NF and DF: via SIEVE management protocol <Editor's note: Still to be defined>

- o VF: via LFILTER as virtual folder as defined in [11]

7. Analysis of Lemonade Technology Support of OMA MEM enabler

The OMA MEM AD [2] analyzes the technical features needed to support its requirements [3].

In the present section we explain how these features extracted from the OMA MEM AD are supported with a LEMONADE realization.

Items maked as ** indicate how it is currently thought that this is addressed by LEMONADE. ++ refers to OMA aspects. -- refers to issues that are not within the scope of LEMONADE specifications or that are still unresolved at the level of LEMONADE.

1. Mechanisms to align, fetch and update email messages between the MEM client and the email server via the MEM server. The mobile email enabler focuses solely on the interaction between the MEM client and MEM server.

   * ** This is supported by IMAP base capabilities.

2. Mechanisms for event-based server to client alignment:

   * Defines the relationship between notification mechanisms and MEM protocol.

   * ** This is supported by server to client notifications [11]

     + To minimize the latency observed for email events on the email server to be reflected in the MEM client.

     + To avoid unnecessary polling and requests from the MEM clients:

       - To reduce the total amount of data to be exchanged between MEM server and client, e.g. by allowing the MEM client to select which messages to align.

       - To reduce the amount of transactions.

   * Needs to cope with possible lost or delayed notifications

   * ** This is supported by ensuring that the LEMONADE protocol does not require that the notification have been received by the MUA.

   * Support in-band (ME-1/ME-2 exchanges) and out-band notifications (Exchanged via ME-3/ME-4 via other enablers).

* ** In-band notifications are supported by IMAP IDLE [12] acting on the virtual folder defined by VF

* ** Outband notifications are generated by SIEVE bound to message store events [13] and acting on the virtual folder defined by VF.

    + Specified in ways that are network transport independent but may contain some bindings to particular notification channels (e.g. SMS binary, WAP Push, SIP Notification, ...)

    + When the MEM client is connected to the MEM server, only inband notifications shall take place

* Defines notification payload for inband and outband mechanisms.

* ** For inband this is defined by IMAP IDLE

* ++ For outband this is specified by server to client notifications [11]. It is expected to be based on OMA EMN server to client notifications [14] plus possible extensions.

* Server-side filtering to decide which messages will be accessible by the MEM client.

    + Filtering results into the following logical types:

    + ** This is achieved by usage of view filters defined as virtual folders, SIEVE binding to message store events.

        - Type A: Messages filtered out and not accessible by the MEM client (no notification, no header access, no access)

        - ** Achieved via VF as view filter defined as virtual folder. No notification will affect events related to these messages by restricting that NF applies only on the view. Otherwise, NF and VF must be synchronized.

        - Type B: Messages that are accessible by the MEM client but no outband notification takes place. Inband notification might however take place if MEM client is already connected to MEM server.

        - ** Achieved via VF as view filter defined as virtual folder. NF is then set to suppress (i.e. not produce)

        notifications even when tehse messages are made available by VF.

        - Type C: Messages that are accessible by the MEM client for which notifications (outband or inband) are always sent to the MEM client.

        - ** Achieved via NF using SIEVE and bindings to messages store events.

    + Notions of Filters:

        - View filters: Filters that determine which email messages are of type B and C or A

        - ** VF is defined as as virtual folder. This may be updated from the client via LFILTER mechanisms [11].

        - Notification filters: Filters that determine which email messages are of type C or B

        - Event filters: Filters that determines what events are to be notified to the client

    + Mechanisms to allow the user to update the filters from the MEM client

    + ** This can be achieved by LFILTER.

3. Client-side download and storage preferences:

    * Manage which of the accessible messages are maintained on MEM client

    * ** This is achieved via the view filter defined as virtual folder.

    * -- From a LEMONADE point of view, client specific additional aspects are left to specifications of the MEM client implementations or to OMA MEM specifications (out of scope of LEMONADE): the user may set local client preferences to cache less than their view filter. For example, the client may cache only the last N days while receiving view updates on all messages within the "urgent" view.

    * Manage which parts are maintained on MEM client

- ** This is achieved by IMAP

- -- Aspects are left to specifications of the MEM client implementations or to OMA MEM specifications (out of scope of LEMONADE).

- Configurable by user

- -- This is left to specifications of the MEM client implementations or to OMA MEM specifications (out of scope of LEMONADE).

- MEM client may support encrypting and password protecting the messages.

- -- This is left to specifications of the MEM client implementations or to OMA MEM specifications (out of scope of LEMONADE).

4. Client-side event filtering:

   - Local message store delete: ability to delete email message from the MEM client view while retaining the message on the email server. Some information may be passed to the MEM server.

   - -- This is left to specifications of the MEM client implementations (out of scope of LEMONADE).

   - Attachment local delete: Ability to delete from the MEM client the attachment while maintaining the view that an attachment is available for download from the email server.

   - -- This is left to specifications of the MEM client implementations (out of scope of LEMONADE).

   - Remote delete: ability to delete email messages both on the MEM client and on the email server.

   - ** Support by IMAP

5. Mechanisms for media conversion

   - ** This is supported by CONVERT [15].

   - Allows the MEM client to request conversion - including transcoding - of a body part or attachment from the MEM server when the email message part is fetched from the

server.

  + The client may request conversion to a specific format/size, or

  + The client may request conversion to a server-selected format/size - where the server decides the format/size credentials based on any knowledge (e.g. client capabilities, user preferences) it may have.

* Conversion does not alter the messages in the email server.

6. Mechanisms for MEM client to submit email to the MEM server.

   * ** This is achieved by SMTP - Submit

   * Mechanism to support remote message assembly on the MEM server based on email parts (body, address fields and attachments) that may not have been downloaded and others that may have been locally created or may have been downloaded and edited.

   * This is supported by LEMONADE profile trio: BURL, CATENATE, URLAUTH [4]

   * It may be desirable to support just uploading the differences of the body parts (e.g. address fields).

   * -- This may be achievable by IMAP URL extended with byte range (partial) [16].

   * -- Differences on address fields may require SMTP extensions. This is under discussion.

7. Mechanisms to allow configuration and exchange of settings between the client and the server inband or outband:

   * Server to client: e.g. server ID, account name, policies, ...

   * ++ This is achievable by OMA DM / CP <Editor's note: Add references>

   * ** Most configuration beyond bootstrap is also achievable via mailbox annotations <Editor's note: Add references>

   * Client to server: e.g. rules filters vacation notices, notification channel, ...

- ** A remote management protocol is to be defined for DF and NF that provide SIEVE management.

- ** SETPREFS/GETPREFS for server to client notifications [11]

8. Mechanisms to optimize bandwidth and/or delays on any data exchanges

   - ** This is addressed by IMAP BINARY, TLS Compression and LZIP compression. Exact usage guidelines are still under development

9. Mechanisms for encryption of the email data exchanged between the email server and the MEM client.

   - ** This is supported by TLS. Note TLS just protects between the MEM client and the MEM server (or may be just to the MEM proxy in front of the MEM server if there is one).

   - The enabler shall support data remaining encrypted at all times even if the MEM server is deployed outside the email server domain.

   - -- This may require object level encryption under some circumstances (e.g. to support deployment models where proxy / MEM server outside email service provider domain). This is still under discussion.

   - The mechanism should also be applicable to notifications if they carry information worth protecting.

   - ** The current thought is to reduce the notifications to the exchange of information that may not have to be encrypted. This is still work in progress.

10. Mechanisms for the MEM client to determine the capabilities of the server.

    - ** This can be supported by CAPABILITY, Mailbox annotations. The notion of Sieve Capabilities description is also work in progress.

11. Mechanisms to manage sessions:

    - ** Covert by Quick Reconnect [17]

    - Handling connectivity issues

        - E.g. dealing with IP address changes

        - E.g. re-establish secure connection

    - E.g. suspend and resume minimizing data exchange duplication

12. Capability to support the different deployment models in appendix of OMA MEM AD [2]

    - Mobile email must be usable in the presence of firewalls and intermediaries found in mobile networks.

    - ** This is discussed in [20]

    - ** Best practices exist [19].

    - -- This may require presence of LEMONADE proxy MEM servers (i.e. Lemonade enhanced IMAP or Submit proxies)

    - -- This may require the use of HTTP binding [18]

13. Mechanisms to ensure integrity of the email data exchanged between the email server and the MEM client.

    - ** This is supported by TLS with the TLS limitations pointed out above for confidentiality <Editor's note: Add references>.

14. Mechanisms for mutual authentication of the MEM client and the MEM server.

    - ** This is supported by TLS <Editor's note: Add references>.

15. Mechanism to allow the MEM client to send recall request to the email server via the MEM server.

    - -- This is a feature that we believe can not easily be satisfied for internet email. We note that some proposals have been made for addressing this with extensions to the MSGTRAK work <Editor's note: Add references>.

16. Mechanisms to sign data exchanged between MEM client and MEM server.

    - ** This is supported by TLS with the TLS limitations mentioned above <Editor's note: Add references> and providing that having a signature of the MEM server or MEM proxy is somehow meaningful.

17. Mechanisms to allow the MEM client to work off line or in intermittent connectivity:

    * ** This is already built in IMAP

    * ** This is addressed by QuickReconnect <Editor's note: Add references>

    * -- This are also aspects left to specifications of the MEM client implementations (out of scope of LEMONADE).

    * Store email and client email event

    * Detect network availability

    * Send emails and email client events when network connectivity is available

8. Security considerations

This specification provides no security measures beyond those in the referenced Internet Mail and LEMONADE documents.

We note however the security risks associated to:

- Outband notifications

- Server configuration by client

- Client configuration by server

- Presence of MEM proxy servers

- Presence of MEM servers as intermediaries

- In general the deployment models considered by OMA MEM that are not conventional IETF deployment models.

- Measures to address the need to traverse firewalls

9. IANA considerations

No specific IANA considerations have been identified that are not covered by the different drafts and RFCs included in the realization described in this document.

10. Acknowledgements

The authors acknowledge and appreciate the work and comments of the IETF LEMONADE working group and the OMA MEM working group.

This text was partially co-authored with G. Vaudreuil (Lucent) and Eric Burger (Excel) and reviewed in detail by them as well as by Fan Xiaohui (China Mobile - CMCC).

11. References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirements Levels", RFC 2119, BCP 14, March 1997.

[2] "Mobile Email Architecture Document", OMA (Work in Progress), http://www.openmobilealliance.org/, October 2005.

[3] "Mobile Email RequirementS Document", OMA http://www.openmobilealliance.org/release_program/docs/RD/OMA-RD-MobileEmail-V1_0_20051018-C.pdf, Oct 2005.

[4] Maes, S. and A. Melnikov, "LEMONADE profile", draft-ietf-lemonade-profile-0x (work in progress).

[5] Crispin, M., "IMAP4, Internet Message Access Protocol Version 4 rev1", RFC 3501, March 2003.

[6] Klensin, J., "SMTP Service Extensions", RFC 1861, November 1995.

[7] Klensin, J., "Simple Mail Transfer Protocol", RFC 2821, April 2001.

[8] Crocker, D., "Internet Mail Architecture", draft-crocker-email-arch-0x (work in progress).

[9] "SIEVE WG", http://www.ietf.org/html.charters/sieve-charter.html.

[10] Maes, S. and et Al., "Persistent Search Extensions and Virtual Folder to the IMAP Protocol", draft-maes-lemonade-vfolder-0x (work in progress).

[11] Maes, S. and et Al., "Server to Client Notifications and Filtering", draft-maes-lemonade-notifications-server-to-client-0x (work in progress).

[12] Leiba, B., "IMAP4 IDLE command", RFC 2177, June 1997.

[13] Newman, C., "Internet Message Store Events", draft-newman-lemonade-msgevent-0x (work in progress).

[14] "Open Mobile Alliance Email Notification Version 1.0", OMA http://www.openmobilealliance.org/tech/docs/EmailNot/OMA-Push-EMN-V1_0-20020830-C.pdf, August 2002.

[15] Maes, S. and et Al., "CONVERT", draft-ietf-lemonade-convert-0x (work in progress).

[16] Melnikov, A. and et Al., "IMAP URL Scheme", draft-ietf-lemonade-rfc2192bis-0x (work in progress).

[17] Melnikov, A. and et Al., "IMAP4 extension for quick reconnect", draft-ietf-lemonade-reconnect-0x (work in progress).

[18] Maes, S. and et Al., "Lemonade HTTP Binding", draft-maes-lemonade-http-binding-0x (work in progress).

[19] Freed, N., "Behavior of and Requirements for Internet Firewalls", RFC 2979, October 2000.

[20] Maes, S., "Lemonade and the challenges of Intermediaries", Internet-Draft ddraft-smaes-lemonade-intermediary-challenges-0x.

Authors' Addresses

Stephane H. Maes
Oracle
MS 4op634, 500 Oracle Parkway
Redwood Shores, CA 94539
USA

Phone: +1-203-300-7786
Email: stephane.maes@oracle.com

Glenn Parsons
Nortel
3500 Carling Avenue
Ottawa, ON K2H 8E9
CA

Phone: +1 613 763 7582
Email: gparsons@nortel.com

Intellectual Property Statement

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

Acknowledgment

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