MIME E-mail Encapsulation of Aggregate Documents, such as HTML (MHTML)

Status of this Document

This document is an Internet-Draft. 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.''

To learn the current status of any Internet-Draft, please check the
``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow
Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe),
munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or
ftp.isi.edu (US West Coast).

Abstract

Although HTML [RFC 1866] was designed within the context of MIME, more
than the specification of HTML as defined in RFC 1866 is needed for two
electronic mail user agents to be able to interoperate using HTML as a
document format. These issues include the naming of objects that are
normally referred to by URIs, and the means of aggregating objects that
go together. This document describes a set of guidelines that will allow
conforming mail user agents to be able to send, deliver and display
these objects, such as HTML objects, that can contain links represented
by URIs. In order to be able to handle inter-linked objects, the
document proposes to use the MIME type multipart/related and specifies
the MIME content-headers "Content-Location" and "Content-Base".

draft-ietf-mhtml-spec-03.txt   July 1996

Table of Contents

Mailing List Information

Further discussion on this document should be done through the mailing
list MHTML@SEGATE.SUNET.SE.

To subscribe to this list, send a message to
    LISTSERV@SEGATE.SUNET.SE

which contains the text
SUB MHTML <your name (not your e-mail address)>

Archives of this list are available by anonymous ftp from
    FTP://SEGATE.SUNET.SE/lists/mHTML/
The archives are also available by e-mail. Send a message to
LISTSERV@SEGATE.SUNET.SE with the text "INDEX MHTML" to get a list of
the archive files, and then a new message "GET <file name>" to retrieve
the archive files.

Comments on less important details may also be sent to the editor, Jacob
Palme <jpalme@dsv.su.se>.

More information may also be available at URL:
HTTP://www.dsv.su.se/~jpalme/ietf/jp-ietf-home.HTML

---

draft-ietf-mhtml-spec-03.txt   July 1996

1. Introduction

There are a number of document formats, HTML [HTML2], PDF [PDF] and VRML
for example, which provide links using URIs for their resolution. There
is an obvious need to be able to send documents in these formats in e-
mail [RFC821=SMTP, RFC822]. This document gives additional
specifications on how to send such documents in MIME [RFC 1521=MIME1] e-
mail messages. This version of this standard was based on full
consideration only of the needs for objects with links in the Text/HTML
media type (as defined in RFC 1866 [HTML2]), but the standard may still
be applicable also to other formats for sets of interlinked objects,
linked by URIs. There is no conformance requirement that implementations
claiming conformance to this standard are able to handle URI-s in other
document formats than HTML.

URIs in documents in HTML and other similar formats reference other
objects and resources, either embedded or directly accessible through
hypertext links. When mailing such a document, it is often desirable to
also mail all of the additional resources that are referenced in it;
those elements are necessary for the complete interpretation of the
primary object.

An alternative way for sending an HTML document or other object
containing URIs in e-mail is to only send the URL, and let the recipient
look up the document using HTTP. That method is described in [URLBODY]
and is not described in this document.

2. Terminology

2.1 Conformance requirement terminology

This specification uses the same words as RFC 1123 [HOSTS] for defining
the significance of each particular requirement. These words are:

MUST   This word or the adjective "required" means that the item is
          an absolute requirement of the specification.

SHOULD   This word or the adjective "recommended" means that there may
          exist valid reasons in particular circumstances to ignore this
          item, but the full implications should be understood and the
          case carefully weighed before choosing a different course.

MAY   This word or the adjective "optional" means that this item is
          truly optional. One vendor may choose to include the item
          because a particular marketplace requires it or because it
          enhances the product, for example; another vendor may omit the
          same item.

An implementation is not compliant if it fails to satisfy one or more of
the MUST requirements for the protocols it implements. An implementation
that satisfies all the MUST and all the SHOULD requirements for its
protocols is said to be "unconditionally compliant"; one that satisfies

---

all the MUST requirements but not all the SHOULD requirements for its
protocols is said to be "conditionally compliant."

2.2 Other terminology

Most of the terms used in this document are defined in other RFCs.

Absolute URI,   See RFC 1808 [RELURL].
AbsoluteURI

CID   See [MIDCID].

Content-Base   See section 4.2 below.

Content-ID   See [MIDCID].

Content-Location   MIME message or content part header with the URI of

	the MIME message or content part body, defined in [section 4.3](#) below.
Content-Transfer-Encoding	Conversion of a text into 7-bit octets as specified in [[MIME1](#)].
CR	See [[RFC822](#)].
CRLF	See [[RFC822](#)].
Displayed text	The text shown to the user reading a document with a web browser. This may be different from the HTML markup, see the definition of HTML markup below.
Header	Field in a message or content heading specifying the value of one attribute.
Heading	Part of a message or content before the first CRLFCRLF, containing formatted fields with attributes of the message or content.
HTML	See [RFC 1866](#) [[HTML2](#)].
HTML Aggregate objects	HTML objects together with some or all objects, to which the HTML object contains hyperlinks.
HTML markup	A file containing HTML encodings as specified in [HTML] which may be different from the displayed text which a person using a web browser sees. For example, the HTML markup may contain "<" where the displayed text contains the character "<".
LF	See [[RFC822](#)].
MIC	Message Integrity Codes, codes use to verify that a message has not been illegally modified.

MIME	See [RFC 1521](#) [[MIME1](#)], [[MIME2](#)].
MUA	Messaging User Agent.
PDF	Portable Document Format, see [[PDF](#)].
Relative URI, RelativeURI	See [RFC 1866](#) [[HTML2](#)] and [RFC 1808](#)[[RELURL](#)].

```
URI, absolute and      See RFC 1866 [HTML2].
relative

URL                    See RFC 1738 [URL].

URL, relative          See [RELURL].

VRML                   Virtual Reality Markup Language.
```

3. Overview

An aggregate document is a MIME-encoded message that contains a root
document as well as other data that is required in order to represent
that document (inline pictures, style sheets, applets, etc.). Aggregate
documents can also include additional elements that are linked to the
first object. It is important to keep in mind the differing needs of
several audiences. Mail sending agents might send aggregate documents as
an encoding of normal day-to-day electronic mail. Mail sending agents
might also send aggregate documents when a user wishes to mail a
particular document from the web to someone else. Finally mail sending
agents might send aggregate documents as automatic responders, providing
access to WWW resources for non-IP connected clients.

Mail receiving agents also have several differing needs. Some mail
receiving agents might be able to receive an aggregate document and
display it just as any other text content type would be displayed.
Others might have to pass this aggregate document to a browsing program,
and provisions need to be made to make this possible.

Finally several other constraints on the problem arise. It is important
that it be possible for a document to be signed and for it to be able to
be transmitted to a client and displayed with a minimum risk of breaking
the message integrity (MIC) check that is part of the signature.

4. The Content-Location and Content-Base MIME Content Headers

4.1 MIME content headers

In order to resolve URI references to other body parts, two MIME content
headers are defined, Content-Location and Content-Base. Both these
headers can occur in any message or content heading, and will then be
valid within this heading and for its content.

In practice, at present only those URIs which are URLs are used, but it is anticipated that other forms of URIs will in the future be used.

The syntax for these headers is, using the syntax definition tools from [RFC822]:

    content-location ::= "Content-Location:" (absoluteURI | relativeURI)

    content-base ::= "Content-Base:" absoluteURI

where URI is at present (June 1996) restricted to the syntax for URLs as defined in RFC 1738 [URL].

These two headers are valid only for exactly the content heading or message heading where they occurs and its text. They are thus not valid for the parts inside multipart headings, and are thus meaningless in multipart headings.

These two headers may occur both inside and outside of a multipart/related part.

4.2 The Content-Base header

The Content-Base gives a base for relative URIs occurring in other heading fields and in content which do not have any BASE element in its HTML code. Its value MUST be an absolute URI.

Example showing which Content-Base is valid where:

    Content-Type: Multipart/related; boundary="boundary-example-1";
                  type=Text/HTML; start=foo2*foo3@bar2.net
      ; A Content-Base header cannot be placed here, since this is a
      ; multipart MIME object.

    --boundary-example-1

    Part 1:
    Content-Type: Text/HTML; charset=US-ASCII
    Content-ID: foo2*foo3@bar2.net
    Content-Location: "http/www.ietf.cnir.reston.va.us/images/foo1.bar1";
      ; This Content-Location must contain an absolute URI, since no base
      ; is valid here.

    --boundary-example-1

    Part 2:
    Content-Type: Text/HTML; charset=US-ASCII

```
   Content-ID: foo4*foo5@bar2.net
   Content-Location: "foo1.bar1" ; The Content-Base below applies to
                                 ; this relative URI
   Content-Base: "http:/www.ietf.cnri.reston.va.us/images/"

   --boundary-example-1--
```

[4.3](4.3) The Content-Location Header

The Content-Location header specifies the URI that corresponds to the
content of the body part in whose heading the header is placed. Its
value CAN be an absolute or relative URI. Any URI or URL scheme may be
used, but use of non-standardized URI or URL schemes might entail some
risk that recipients cannot handle them correctly.

The Content-Location header can be used to indicate that the data sent
under this heading is also retrievable, in identical format, through
normal use of this URI. If used for this purpose, it must contain an
absolute URI or be resolvable, through a Content-Base header, into an
absolute URI. In this case, the information sent in the message can be
seen as a cached version of the original data.

The header can also be used for data which is not available to some or
all recipients of the message, for example if the header refers to an
object which is only retrievable using this URI in a restricted domain,
such as within a company-internal web space. The header can even contain
a fictious URI and need in that case not be globally unique.

Example:

```
Content-Type: Multipart/related; boundary="boundary-example-1";
                type=Text/HTML

   --boundary-example-1

   Part 1:
   Content-Type: Text/HTML; charset=US-ASCII

   ... ... <IMG SRC="fiction1/fiction2"> ... ...

   --boundary-example-1

   Part 2:
   Content-Type: Text/HTML; charset=US-ASCII
   Content-Location: "fiction1/fiction2"
```

--boundary-example-1--

4.4 Encoding of URIs in e-mail headers

Since MIME header fields have a limited length and URIs can get quite
long, these lines may have to be folded. If such folding is done, the
algorithm defined in [URLBODY] section 3.1 should be employed.

5. Base URIs for resolution of relative URIs

Relative URIs inside contents of MIME body parts are resolved relative
to a base URI. In order to determine this base URI, the first-listed
method in the following list applies.

   (a) There is a base specification inside the MIME body part
       containing the link which resolves relative URIs into absolute
       URIs. For example, HTML provides the BASE element for this.

   (b) There is a Content-Base header (as defined in section 4.2),
       specifying the base to be used.

   (c) There is a Content-Location header in the heading of the body
       part which can then serve as the base in the same way as the
       request URI can serve as a base for relative URIs within a file
       retrieved via HTTP [HTTP].

When the methods above do not yield an absolute URI the procedure in
section 8.2 for matching relative URIs MUST be followed.

6. Sending documents without linked objects

If a document, such as an HTML object, is sent without other objects, to
which it is linked, it MAY be sent as a Text/HTML body part by itself.
In this case, multipart/related need not be used.

Such a document may either not include any links, or contain links which
the recipient resolves via ordinary net look up, or contain links which
the recipient cannot resolve.

Inclusion of links which the recipient has to look up through the net
may not work for some recipients, since all e-mail recipients do not

have full internet connectivity. Also, such links may work for the
sender but not for the recipient, for example when the link refers to an
URI within a company-internal network not accessible from outside the
company.

Note that documents with links that the recipient cannot resolve MAY be
sent, although this is discouraged. For example, two persons developing
a new HTML page may exchange incomplete versions.

7. Use of the Content-Type: Multipart/related

If a message contains one or more MIME body parts containing links and
also contains as separate body parts, data, to which these links (as
defined, for example, in RFC 1866 [HTML2]) refers, then this whole set
of body parts (referring body parts and referred-to body parts) SHOULD
be sent within a multipart/related body part as defined in [REL].

The root body part of the multipart/related SHOULD be the start object
for rendering the object, such as a text/html object, and which contains
links to objects in other body parts, or a multipart/alternative of
which at least one alternative resolves to such a start object.
Implementors are warned, however, that many mail programs treat
multipart/alternative as if it had been multipart/mixed (even though
MIME [MIME1] requires support for multipart/alternative).

[REL] requires that the type attribute of the "Content-Type:
multipart/related" statement be the type of the root object, and this
value can thus be "multipart/alternative". If the root is not the first
body part within the multipart/related, [REL] further requires that its
Content-ID MUST be given in a start parameter to the "Content-Type:
multipart/related" header.

When presenting the root body part to the user, the additional body
parts within the multipart/related can be used:

    (a) For those recipients who only have e-mail but not full Internet
        access.

    (b) For those recipients who for other reasons, such as firewalls
        or the use of company-internal links, cannot retrieve the
        linked body parts through the net.

        Note that this means that you can, via e-mail, send HTML which

includes URIs which the recipient cannot resolve via HTTPor
other connectivity-requiring URIs.

(c) For items which are not available on the web.

(d) For any recipient to speed up access.

The type parameter of the Content-Type: multipart/related MUST be the
same as the Content-Type of its root.

When a sending MUA sends objects which were retrieved from the WWW, it
SHOULD maintain their WWW URIs. It SHOULD not transform these URIs into
some other URI form prior to transmitting them. This will allow the
receiving MUA to both verify MICs included with the email message, as
well as verify the documents against their WWW counterpoints.

This standard does not cover the case where a multipart/related contains
links to MIME body parts outside of the current multipart/related or in
other MIME messages, even if methods similar to those described in this
standard are used. Implementors who provide such links are warned that
mailers implementing this standard may not be able to resolve such
links.

Within such a multipart/related, ALL different parts MUST have different
Content-Location or Content-ID values.

8. Format of Links to Other Body Parts

8.1 General principle

A body part, such as a text/HTML body part, may contain hyperlinks to
objects which are included as other body parts in the same message and
within the same multipart/related content. Often such linked objects are
meant to be displayed inline to the reader of the main document; for
example, objects referenced with the IMG tag in HTML [RFC 1866=HTML2].

Palme-Hopmann   Do not implement based on this draft   [Page 9]

New tags with this property are proposed in the ongoing development of
HTML (example: applet, frame).

In order to send such messages, there is a need to indicate which other
body parts are referred to by the links in the body parts containing
such links. For example, a body part of Content-Type: Text/HTML often
has links to other objects, which might be included in other body parts
in the same MIME message. The referencing of other body parts is done in

the following way: For each body part containing links and each distinct URI within it, which refers to data which is sent in the same MIME message, there SHOULD be a separate body part within the current multipart/related part of the message containing this data. Each such body part SHOULD contain a Content-Location header (see section 8.2) or a Content-ID header (see section 8.3).

An e-mail system which claims conformance to this standard MUST support receipt of multipart/related (as defined in section 7) with links between body parts using both the Content-Location (as defined in section 8.2) and the Content-ID method (as defined in section 8.3).

8.2 Use of the Content-Location header

If there is a Content-Base header, then the recipient MUST employ relative to absolute resolution as defined in RFC 1808 [RELURL] of relative URIs in both the HTML markup and the Content-Location header before matching a hyperlink in the HTML markup to a Content-Location header. The same applies if the Content-Location contains an absolute URI, and the HTML markup contains a BASE element so that relative URIs in the HTML markup can be resolved.

If there is NO Content-Base header, and the Content-Location header contains a relative URI, then NO relative to absolute resolution SHOULD be performed when matching Content-Location headers (even if there is a BASE specification, such as the BASE element in HTML, in the body part containing the URI), and exact textual match of the relative URI-s in the Content-Location and the HTML markup is performed instead (after removal of LWSP introduced as described in section 4.4 above). Note that this only applies for matching Content-Location headers, not for URL-s in the HTML document which are resolved through network look up at read time.

If there is NO Content-Base header, and the Content-Location header contains a relative URI, then NO relative to absolute resolution SHOULD be performed. Matching the relative URI in the Content-Location header to a hyperlink in an HTML markup text is in this case a two step process. First remove any LWSP from the relative URI which may have been introduced as described in section 4.4. Then perform an exact textual match against the HTML URIs. For this matching process, ignore BASE specifications, such as the BASE element in HTML. Note that this only applies for matching Content-Location headers, not for URL-s in the HTML document which are resolved through network look up at read time.

The URI in the Content-Location header need not refer to an object which is actually available globally for retrieval using this URI (after resolution of relative URIs). However, URI-s in Content-Location headers (if absolute, or resolvable to absolute URIs) SHOULD still be globally unique.

8.3 Use of the Content-ID header and CID URLs

When CID (Content-ID) URLs as defined in RFC 1738 [URL] and RFC 1873 [MIDCID] is used for links between body parts, the Content-Location statement will normally be replaced by a Content-ID header. Thus, the following two headers are identical in meaning:

Content-ID: foo@bar.net
Content-Location: CID: foo@bar.net

Note: Content-IDs MUST be globally unique [MIME1]. It is thus not permitted to make them unique only within this message or within this multipart/related.

9 Examples

9.1 Example of a HTML body without included linked objects

The first example is the simplest form of an HTML email message. This is not an aggregate HTML object, but simply a message with a single HTML body part. This message contains a hyperlink but does not provide the ability to resolve the hyperlink. To resolve the hyperlink the receiving client would need either IP access to the Internet, or an electronic mail web gateway.

    From: foo1@bar.net
    To: foo2@bar.net
    Subject: A simple example
    Mime-Version: 1.0
    Content-Type: Text/HTML; charset=US-ASCII

    <HTML>
    <head></head>
    <body>
    <h1>Hi there!</h1>
    An example of an HTML message.<p>
    Try clicking here.<p>
    </body></HTML>

9.2 Example with absolute URIs to an embedded GIF picture:

```
From: foo1@bar.net
   To: foo2@bar.net
   Subject: A simple example
   Mime-Version: 1.0
   Content-Type: multipart/related; boundary="boundary-example-1";
                 type=Text/HTML; start=foo3*foo1@bar.net
```

```
--boundary-example 1
     Content-Type: Text/HTML;charset=US-ASCII
     Content-ID: foo3*foo1@bar.net

     ... text of the HTML document, which might contain a hyperlink
     to the other body part, for example through a statement such as:
     <IMG SRC="http://www.ietf.cnri.reston.va.us/images/ietflogo.gif"
      ALT="IETF logo">

     --boundary-example-1
     Content-Location:
         "http://www.ietf.cnri.reston.va.us/images/ietflogo.gif"
     Content-Type: IMAGE/GIF
     Content-Transfer-Encoding: BASE64
```

```
     R0lGODlhGAGgAPEAAP/////ZRaCgoAAAACH+PUNvcHlyaWdodCAoQykgMTk5
     NSBJRVRGLiBVbmF1dGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
     etc...
```

```
     --boundary-example-1--
```

9.3 Example with relative URIs to an embedded GIF picture

```
   From: foo1@bar.net
   To: foo2@bar.net
   Subject: A simple example
   Mime-Version: 1.0
   Content-Base: "http://www.ietf.cnri.reston.va.us"
   Content-Type: multipart/related; boundary="boundary-example-1";
                 type=Text/HTML

     --boundary-example 1
     Content-Type: Text/HTML; charset=ISO-8859-1
     Content-Transfer-Encoding: QUOTED-PRINTABLE

     ... text of the HTML document, which might contain a hyperlink
     to the other body part, for example through a statement such as:
     <IMG SRC="/images/ietflogo.gif" ALT="IETF logo">
```

```
Example of a copyright sign encoded with Quoted-Printable: =A9
Example of a copyright sign mapped onto HTML markup: &#168;

--boundary-example-1
Content-Location: "/images/ietflogo.gif"
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64

R0lGODlhGAGgAPEAAP/////ZRaCgoAAAACH+PUNvcHlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmF1dGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...

--boundary-example-1--
```

9.4 Example using CID URL and Content-ID header to an embedded GIF
picture

```
From: foo1@bar.net
To: foo2@bar.net
Subject: A simple example
Mime-Version: 1.0
Content-Type: multipart/related; boundary="boundary-example-1";
              type=Text/HTML

--boundary-example 1
Content-Type: Text/HTML; charset=US-ASCII

... text of the HTML document, which might contain a hyperlink
to the other body part, for example through a statement such as:
<IMG SRC="cid:foo4*foo1@bar.net" ALT="IETF logo">

--boundary-example-1
Content-ID: foo4*foo1@bar.net
Content-Type: IMAGE/GIF
Content-Transfer-Encoding: BASE64

R0lGODlhGAGgAPEAAP/////ZRaCgoAAAACH+PUNvcHlyaWdodCAoQykgMTk5
NSBJRVRGLiBVbmF1dGhvcml6ZWQgZHVwbGljYXRpb24gcHJvaGliaXRlZC4A
etc...

--boundary-example-1--
```

10. Content-Disposition header

Note the specification in [REL] on the relations between Content-Disposition and multipart/related.

11. Character encoding issues and end-of-line issues

For the encoding of characters in HTML documents and other text documents into a MIME-compatible octet stream, the following mechanisms are relevant:

- HTML [HTML2, HTML-I18N] as an application of SGML [SGML] allows characters to be denoted by character entities as well as by numeric character references (e.g. "Latin small letter a with acute accent" may be represented by "á" or "á") in the HTML markup.

- HTML documents, in common with other documents of the MIME content-type "text", can be represented in MIME using one of several character encodings. The MIME content-type "charset" parameter value indicates the particular encoding used. For the exact meaning and use of the "charset" parameter, please see [MIME-IMB section 4.2].

   Note that the "charset" parameter refers only to the MIME character encoding. For example, the string "á" can be sent in MIME with "charset=US-ASCII", while the raw character "Latin small letter a with acute accent" cannot.

The above mechanisms are well defined and documented, and therefore not further explained here. In sending a message, all the above mentioned mechanisms MAY be used, and any mixture of them MAY occur when sending the document via e-mail. Receiving mail user agents (together with the Web browser they may use to display the document) MUST be capable of handling any combinations of these mechanisms.

Also note that:

- Any documents including HTML documents that contain octet values outside the 7-bit range need a content-transfer-encoding applied before transmission over certain transport protocols [MIME1, chapter 5].

- The MIME standard [MIME1] requires that documents of "Content-
Type:text"
   MUST be in canonical form before Content-Transfer-Encoding, i.e. that
   line breaks are encoded as CRLFs, not as bare CRs or bare LFs or
   something else. This is in contrast to [HTTP] where section 3.6.1
   allows other representations of line breaks.

Note that this might cause problems with integrity checks based on
checksums, which might not be preserved when moving a document from the
HTTP to the MIME environment. If a document has to be converted in such
a way that a checksum integrity check becomes invalid, then this
integrity check header SHOULD be removed from the document.

Other sources of problems are "Content-Encoding" used in HTTP but not
allowed in MIME, and "charsets that are not able to represent line
breaks as CRLF. A good overview of the differences between HTTP and MIME
with regards to Content-Type:text" can be found in [HTTP], appendix C.

If the original document has line breaks in the canonical form (CRLF),
then the document SHOULD remain unconverted so that integrity check sums
are not invalidated.

A provider of HTML documents who wants his documents to be transferable
via both HTTP and SMTP without invalidating checksum integrity checks,
should always provide original documents in the canonical form with CRLF
for line breaks.

Some transport mechanisms may specify a default "charset" parameter if
none is supplied [HTTP, MIME1]. Because the default differs for
different mechanisms, when HTML is transferred through mail, the charset
parameter SHOULD be included, rather than relying on the default.

draft-ietf-mhtml-spec-03.txt   July 1996

12. Security Considerations

Some Security Considerations include the potential to mail someone an
object, and claim that it is represented by a particular URI (by giving
it a Content-Location: header). There can be no assurance that a WWW
request for that same URI would normally result in that same object. It
might be unsuitable to cache the data in such a way that the cached data

can be used for retrieval of this URI from other messages or message
parts than those included in the same message as the Content-Location
header. Because of this problem, receiving User Agents SHOULD not cache
this data in the same way that data that was retrieved through an HTTP
or FTP request might be cached.

URLs, especially File URLs, may in their name contain company-internal
information, which may then inadvertently be revealed to recipients of
documents containing such URLs.

One way of implementing messages with linked body parts is to handle the
linked body parts in a combined mail and WWW proxy server. The mail
client is only given the start body part, which it passes to a web
browser. This web browser requests the linked parts from the proxy
server. If this method is used, and if the combined server is used by
more than one user, then methods must be employed to ensure that body
parts of a message to one person is not retrievable by another person.
Use of passwords (also known as tickets or magic cookies) is one way of
achieving this. Note that some caching WWW proxy servers may not
distinguish between cached objects from e-mail and HTTP, which may be a
security risk.

In addition, by allowing people to mail aggregate objects, we are
opening the door to other potential security problems that until now
were only problems for WWW users. For example, some HTML documents now
either themselves contain executable content (JavaScript) or contain
links to executable content (The "INSERT" specification, Java). It would
be exceedingly dangerous for a receiving User Agent to execute content
received through a mail message without careful attention to
restrictions on the capabilities of that executable content.

13. Acknowledgments

Harald T. Alvestrand, Richard Baker, Dave Crocker, Martin J. Duerst,
Lewis Geer, Roy Fielding, Al Gilman, Paul Hoffman, Richard W. Jesmajian,
Mark K. Joseph, Greg Herlihy, Valdis Kletnieks, Daniel LaLiberte, Ed
Levinson, Jay Levitt, Albert Lunde, Larry Masinter, Keith Moore, Gavin
Nicol, Pete Resnick, Jon Smirl, Einar Stefferud, Jamie Zawinski, Steve
Zilles and several other people have helped us with preparing this
document. I alone take responsibility for any errors which may still be
in the document.

14. References

Ref.   Author, title
---------   --

[CONDISP]   R. Troost, S. Dorner: "Communicating Presentation
                Information in Internet Messages: The Content-
                Disposition Header", RFC 1806, June 1995.

[HOSTS]   R. Braden (editor): "Requirements for Internet Hosts --
                Application and Support", STD-3, RFC 1123, October 1989.

[HTML2]   T. Berners-Lee, D. Connolly: "Hypertext Markup Language
                – 2.0", RFC 1866, November 1995.

[HTML-I18N]   F. Yergeau, G. Nicol, G. Adams, & M. Duerst:
                "Internationalization of the Hypertext Markup
                Language". draft-ietf-html-i18n-04.txt, May 1996.

[HTTP]   T. Berners-Lee, R. Fielding, H. Frystyk: Hypertext
                Transfer Protocol -- HTTP/1.0. RFC 1945, May 1996.

[MD5]   R. Rivest, "The MD5 Message-Digest Algorithm", RFC 1321,
                April 1992.

[MIDCID]   E. Levinson: "Message/External-Body Content-ID Access
                Type", draft-ietf-mhtml-cid-00.txt, August 1996.

[MIME1]   N. Borenstein & N. Freed: "MIME (Multipurpose Internet
                Mail Extensions) Part One: Mechanisms for Specifying and
                Describing the Format of Internet Message Bodies", RFC
                1521, Sept 1993.

[MIME2]   N. Borenstein & N. Freed: "Multipurpose Internet Mail
                Extensions (MIME) Part Two: Media Types". draft-ietf-
                822ext-mime-imt-02.txt, December 1995.

[MIME-IMB]   N. Freed & N. Borenstein: "Multipurpose Internet Mail
                Extensions (MIME) Part One: Format of Internet Message
                Bedies". draft-ietf-822ext-mime-imb-07.txt, June 1996.

[NEWS]   M.R. Horton, R. Adams: "Standard for interchange of
                USENET messages", RFC 1036, December 1987.

[PDF]   Bienz, T., Cohn, R. and Meehan, J.: "Portable Document
                Format Reference Manual, Version 1.1", Adboe Systems
                Inc.

[REL]   Harald Tveit Alvestrand, Edward Levinson: "The MIME
                    Multipart/Related Content-type", <draft-ietf-mhtml-
                    related-00.txt>, May 1995.

[RELURL]   R. Fielding: "Relative Uniform Resource Locators", RFC
                    1808, June 1995.

[RFC822]   D. Crocker: "Standard for the format of ARPA Internet
                    text messages." STD 11, RFC 822, August 1982.

[SGML]   ISO 8879. Information Processing -- Text and Office -
                    Standard Generalized Markup Language (SGML),
                    1986. <URL:http://www.iso.ch/cate/d16387.html>

[SMTP]   J. Postel: "Simple Mail Transfer Protocol", STD 10, RFC
                    821, August 1982.

[URL]   T. Berners-Lee, L. Masinter, M. McCahill: "Uniform
                    Resource Locators (URL)", RFC 1738, December 1994.

[URLBODY]   N. Freed and Keith Moore: "Definition of the URL MIME
                    External-Body Access-Type", draft-ietf-mailext-acc-url-
                    01.txt, November 1995.

15. Author's Address

For contacting the editors, preferably write to Jacob Palme rather than
Alex Hopmann.

Jacob Palme   Phone: +46-8-16 16 67
Stockholm University and KTH   Fax: +46-8-783 08 29
Electrum 230   E-mail: jpalme@dsv.su.se
S-164 40 Kista, Sweden

Alex Hopmann
President
ResNova Software, Inc.   E-mail: alex.hopmann@resnova.com
5011 Argosy Dr. #13
Huntington Beach, CA 92649

Working group chairman:

Einar Stefferud <stef@nma.com>