Network Working Group
draft-crocker-wrap-02.txt
Expiration <12/02>

D. Crocker
Brandenburg InternetWorking

June 23, 2002

Wrapping MIME Objects: Application/MIME

STATUS OF THIS MEMO

ABSTRACT

MIME is a general-purpose mechanism for labeling and aggregating objects into complex hierarchies. Some MIME objects are discrete objects. If placed within another MIME hierarchy, they need to appear as a leaf node in the containing hierarchy, rather than seeming to be continuations of that hierarchy. This specification defines Content-type Application/MIME to provide an encapsulation mechanism for arbitrary MIME structures. This facilitates their treatment as a single unit, rather than causing them to be treated as part of any containing MIME hierarchy.

INTRODUCTION

MIME is a general-purpose mechanism for labeling and aggregating objects into complex hierarchies. Some MIME objects are discrete objects. If placed within another MIME hierarchy, they need to appear as a leaf node in the containing hierarchy, rather than seeming to be continuations of that hierarchy. One example of this requirement is to provide some protection against the translations that email gateways often make to MIME attachments. Further some uses of MIMEÆs aggregation mechanism, Content-type Multipart, can lead to requirements for processing an entire aggregated object as a single unit. This is in contrast to Content-type Multipart/Mixed, which specifies separate processing of constituent components, and to which multipart sub-types default if they are unknown to the processing software.

This specification defines Content-type Application/MIME to provide an encapsulation mechanism for arbitrary MIME structures. This facilitates their treatment as a single unit, rather than causing them to be treated as part of any containing MIME hierarchy. It is expected that Application/MIME will be especially useful in getting data past gateways.

The specification simply defines an object that contains MIME data. When the object is removed from its Content-type Application/MIME wrapper, what remains is an entire MIME object, beginning with a (new) Content-type header.

APPLICATION/MIME SPECIFICATION

MIME type name: Application

MIME subtype name: MIME

Required parameters: <contained>

Optional parameters: None

Encoding considerations: May need BASE64 or QUOTED-PRINTABLE transfer encoding, as appropriate for the <contained> data. Careful use of QUOTED-PRINTABLE will maintain clear text as robust against gateway processing yet still be readable without special processing.

NOTE: items carried inside this MIME object must be fully conformant MIME structure and data; this includes all of the usual rules concerning network standard canonical representation.

Security considerations: See separate section in this document.

Published specification: See detail, below.

Rationale: MIME objects can be discrete entities; when attached to another MIME object, the attachment needs to be represented as a leaf, rather than a part of that hierarchy. One use of this distinction is for gateways and other processing environments that can alter and destroy MIME data structure; the defined data type will permit separate handling of the structure inside an object. It also permits passing an aggregate object as a single entity, through processors that would otherwise separate the components.

Contact-info: See Contact section, below.

Detail specific to MIME-based usage:

This is a generic mechanism for encapsulating any MIME object structure. The object is self-defining, since it begins with a MIME Content-Type header and is then processed (recursively). The mechanism is intended for use when correct processing of the basic MIME structure is at risk, in effect allowing the MIME structure to be tunneled through such an environment.

The Content-Type parameter comprises the type and sub-type tokens of the Content-Type header for the contained MIME component, i.e.,

```
contained   =  c-t *(ô;ö *parameter)

c-t         =  ôContent-Type=ô <"> type "/" subtype <">

parameter   =  {as defined for MIME}
```

where the type and subtype tokens are defined by the MIME [2] specification. The value of the Content-Type parameter specifies the MIME Content-Type and subtype of the data structure contained inside the Application/MIME object.

The <contained> parameters replicate all of the parameters which are present in the Content-Type specification header referenced by the Content-Type parameter. That is, they replicate the contained objectÆs parameters, in their entirety. This is done to facilitate dispatch of a particular type to a handler, without parsing the contained MIME structure.

> If the contained object is, itself Application/MIME, then none of the <contained> parameters of that contained object shall be included.

(Using Application/MIME for doubly-wrapping MIME data may provide a necessary level of protection in some cases.)

GATEWAY AND PASS-THROUGH PROCESSING

Software which manipulates an Application/MIME component must do so ONLY when processing can be done fully and correctly. In the extreme, this may require full parsing of the contained MIME structure and its parameters, prior to deciding whether to take responsibility for the content. Typically, however, review of the Application/MIME type and <contained> parameters will suffice.

Any MIME-aware software that encounters an Application/MIME component must leave the component in its existing form, unless that software is able to fully and correctly process ALL of the component contents AND such processing is appropriate to the environment in which the software is operating.

IMPORTANT NOTE: Gateway implementers are specifically and strongly cautioned against modification of an Application/MIME component.

The question of whether to unwrap content that is embedded in an Application/MIME becomes very simple. Application/MIME is used to provide protection against mishandling by intermediaries. Hence, only end-system softwareûincluding gateways and regular email user agentsûshould even consider touching the content and then it should only do so when the recipient has a basis for believing that the processing will be correct.

SIMPLE USAGE IN MIME-BASED EMAIL

This section is intended as a simple example of the gist of the formatting required to encapsulate MIME objects within Internet mail, using Application/MIME:

```
To:
Subject:
From:
Date:
Mime-Version: 1.0
Content-Type: Application/MIME;
                 content-type=Multipart/signed;
                 protocol="application/somesigscheme";
                 boundary="//signatureboundary//"

Content-Type: Multipart/Signed;
                 boundary=//signatureboundary//;
                 protocol="application/somesigscheme"

--//signatureboundary//
Content-type:<<type of the user data>>

<<user data>>

--//signatureboundary//
Content-type:application/somesigscheme

<<signature control information>>

--//signatureboundary//--
```

DOUBLE-WRAPPED EXAMPLE

This section shows a contained object which is, itself, a contained object, double-wrapped for extra protection against decay:

```
To:
Subject:
From:
Date:
Mime-Version: 1.0
Content-Type: Application/MIME;
```

```
                 content-type=Multipart/signed;
                 protocol="application/somesigscheme";
                 boundary="//signatureboundary//"

Content-Type: Application/MIME;
                 content-type=Multipart/signed;
                 boundary="//signatureboundary//";
                 protocol="application/somesigscheme"

Content-Type: Multipart/Signed;
                 boundary=//signatureboundary//;
                 protocol="application/somesigscheme"

--//signatureboundary//
Content-type:<<type of the user data>>

<<user data>>

--//signatureboundary//
Content-type:application/somesigscheme

<<signature control information>>

--//signatureboundary//--
```

SECURITY CONSIDERATIONS

MIME content often includes sensitive data, so that transmission often needs to attend to authentication, data integrity, privacy, access control, and the like as appropriate.

IMPORTANT NOTE: The recursive processing required by Application/MIME requires use of whatever security checks are applied to newly received MIME data.

This specification does NOT, itself, provide any security-related mechanisms. As needed and appropriate, such mechanisms MUST be added, either via Internet MIME-based security services or any other services which are appropriate to the user requirements.

ACKNOWLEDGMENTS

The idea for Application/MIME first developed out of conversations with Einar Stefferud and Marshall Rose, in trying to find a way for exchanging valid Internet Mail,

complete with RFC822 headers and MIME content, through environments that provided no other Internet Mail technology besides a gateway between the proprietary environment and the Internet. Additional benefits of this mechanism then surfaced during discussions on the S/MIME development list.

A previous draft was co-authored by Laurence Lundblade and Jamie Zewinskie. They are not listed in the current draft merely to protect the innocent.

CONTACT

David H. Crocker <dcrocker@brandenburg.com>
Brandenburg InternetWorking
675 Spruce Drive
Sunnyvale, CA 94086 USA
Phone: +1.408.246.8253