Internet Draft
draft-ietf-ediint-compression-12.txt
Expires: February 27, 2009
Intended Status: Informational

Editor: Terry Harding
Axway

August 27, 2008

Compressed Data within an Internet EDI Message

Status of this memo

Abstract

This document explains the rules and procedures for utilizing compression (RFC 3274) within an Internet EDI (Electronic Data Interchange) 'AS' message, as defined in RFCs 3335, 4130, and 4823.

1. Introduction

Historically, electronic messages produced by systems following the guidelines as outlined in the IETF EDIINT working group specifications AS1[AS1], AS2[AS2] and AS3[AS3], did not have a way to provide a standardized transport neutral mechanism for compressing large payloads. However, with the development of RFC 3274 - Compressed Data Content Type for Cryptographic Message Syntax (CMS), we now have a transport neutral mechanism for compressing large payloads.

A typical EDIINT 'AS' message is a multi-layered MIME message, consisting of one or more of the following, payload layer, signature layer and/or the encryption layer. When an 'AS' message is received a Message Integrity Check(MIC) value must be computed based upon defined rules within the EDIINT 'AS' RFCs and returned to the sender

of the message via an Message Disposition Notification(MDN).

The addition of a new compression layer will require this document to outline new procedures for building/layering 'AS' messages and computing a MIC value that is returned in the MDN receipt.

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].

2. Compressed Data MIME Layer

The compressed-data CMS(Cryptographic Message Syntax) MIME entity as described in [COMPRESSED-DATA] may encapsulate a MIME entity which consists of either an unsigned or signed business document.

Implementers are to follow the appropriate specifications identified under "References" in [MIME-TYPES], for the type of object being packaged. For example, to package an XML object, the MIME media type of "application/xml" is used in the Content-type MIME header field and the specifications for enveloping the object are contained in [XMLTYPES];

MIME entity example:

```
Content-type: application/xml; charset="utf-8"

<?xml version="1.0" encoding="UTF-8"?>
<!-- sample xml document -->
```

The MIME entity will be compressed using [ZLIB] and placed inside a CMS compressed-data object as outlined in [COMPRESSED-DATA]. The compressed data object will be MIME encapsulated according to details outlined in [S/MIME3.1], RFC 3851, Section 3.5.

Example:

```
Content-Type: application/pkcs7-mime; smime-type=compressed-data;
        name=smime.p7z
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7z

MIAGCyqGSIb3DQEJEAEJoIAwgAIBADANBgsqhkiG9w0BCRADCDCABgkqhkiG9w0BBwGg
Hnic7ZRdb9owFIbvK/k/5PqVYPFXGK12YYyboVFASSp1vQtZGiLRACZE49/XHoUW7S/0
fU5ivWnasml72XFb3gb5druui7ytN803M570nii7C5r8tfwR281hy/p/KSM3+jzH5s3+
P3VT3QbLusnt8WPIuN5vN/vaA2+DulnXTXkXvNTr8j8ouZmkCmGI/UW+ZS/C8zP0bz2d
UEk2M8mlaxjRMByAhZTj0RGYg4TvogiRASROsZgjpVcJCb1KV6QzQeDJ1XkoQ5Jm+C5P
v+ORAcshOGeCcdFJyfgFxdtCdEcmOrbinc/+BBMzRThEYpwl+jEBpciSGWQkI0TSlREm
SGLuESm/iKUFt1y4XHBO2a5oq0IKJKWLS9kUZTA7vC5LSxYmgVL46SIWxIfWBQd6Adrn
```

vGxVibLqRCtIpp4g2qpdtqK1LiOeolpVK5wVQ5P7+QjZAlrh0cePYTx/gNZuB9Vhndtg
W9ogK+3rnmg3YWygnTuF5GDS+Q/jIVLnCcYZFc6Kk/+c80wKwZjwdZIqDYWRH68MuBQS

3CAaYOBNJMliTl0X7eV5DnoKIFSKYdj3cRpD/cK/JWTHJRe76MUXnfBW8m7Hd5zhQ4ri
+kV1/3AGSlJ32bFPd2BsQD8uSzIx6lObkjdz95c0AAAAAAAAAAAAAAAA

Note: Content-Transfer-Encoding of base64 would only be required if the compressed-data MIME bodypart is transferred via a 7-bit protocol like SMTP and is visible in the outer layer of the MIME message. If the compressed-data MIME bodypart is placed inside of an encrypted MIME bodypart, content-transfer-encoding would not be required on the compressed-data MIME bodypart, but would be required on the encrypted MIME bodypart.

3. Structure of an EDI MIME compressed message

When compressing a document which will be signed, the application MAY compress the inner most MIME body before signing, see Section 3.2 and 3.5 or MAY compress the outer multipart/signed MIME body, see Section 3.3 and 3.6 but MUST NOT do both within the same document. The receiving application MUST support both methods of compression when unpackaging an inbound document.

Note: The following sections 3.1 - 3.6 show the individual layers of a properly formatted EDIINT MIME message with a compressed data layer. Please refer to the appropriate RFCs for the proper construction of the resulting MIME message.

3.1 No encryption, no signature

```
-RFC2822/2045
  -[COMPRESSED-DATA](application/pkcs7-mime)
    -[MIME-TYPES](application/xxxxxxx)(compressed)
```

This section shows the layers of an unsigned, unencrypted compressed message. The first line indicates that the MIME message conforms to RFCs 2822 and RFC 2045 with a Content-Type of application/pkcs7-mime. Within the pkcs7-mime entity is a compressed MIME entity containing the electronic business document.

3.2 No encryption, signature

```
-RFC2822/2045
  -[RFC1847] (multipart/signed)
    -[COMPRESSED-DATA](application/pkcs7-mime)
      -[MIME-TYPES](application/xxxxxxx)(compressed)
    -RFC3851 (application/pkcs7-signature)
```

This section shows the layers of a signed, unencrypted compressed message where the payload is compressed before being signed.

3.3 No encryption, signature

```
  -[COMPRESSED-DATA](application/pkcs7-mime)
    -[RFC1847] (multipart/signed)(compressed)
      -[MIME-TYPES](application/xxxxxxx)(compressed)
      -RFC3851 (application/pkcs7-signature)(compressed)
```

This section shows the layers of a signed, unencrypted compressed message where a signed payload is compressed.

3.4 Encryption, no signature

```
-RFC2822/2045
  -RFC3851 (application/pkcs7-mime)
    -[COMPRESSED-DATA](application/pkcs7-mime) (encrypted)
      -[MIME-TYPES](application/xxxxxxx)(compressed)(encrypted)
```

This section shows the layers of an unsigned, encrypted compressed message where payload is compressed before it is encrypted.

3.5 Encryption, signature

```
-RFC2822/2045
  -RFC3851 (application/pkcs7-mime)
    -[RFC1847] (multipart/signed) (encrypted)
      -[COMPRESSED-DATA](application/pkcs7-mime) (encrypted)
        -[MIME-TYPES](application/xxxxxxx) (compressed)(encrypted)
      -RFC3851 (application/pkcs7-signature) (encrypted)
```

This section shows the layers of an signed, encrypted compressed message where the payload is compressed before being signed and encrypted.

3.6 Encryption, signature

```
-RFC2822/2045
  -RFC3851 (application/pkcs7-mime)
    -[COMPRESSED-DATA](application/pkcs7-mime) (encrypted)
      -[RFC1847] (multipart/signed) (compressed)(encrypted)
        -[MIME-TYPES](application/xxxxxxx) (compressed)(encrypted)
        -RFC3851 (application/pkcs7-signature)(compressed)(encrypted)
```

This section shows the layers of an signed, encrypted compressed message where the payload is compressed before being signed and encrypted.

4. MIC Calculations For Compressed Messages Requesting Signed Receipts

4.1 MIC Calculation For Signed Message

For any signed message, the MIC to be returned is calculated over the same data that was signed in the original message as per [AS1].

The signed content will be a mime bodypart that contains either compressed or uncompressed data.

4.2 MIC Calculation For Encrypted, Unsigned Message

For encrypted, unsigned messages, the MIC to be returned is calculated over the uncompressed data content including all MIME header fields and any applied Content-Transfer-Encoding.

4.3 MIC Calculation For Unencrypted, Unsigned Message

For unsigned, unencrypted messages, the MIC is calculated over the uncompressed data content including all MIME header fields and any applied Content-Transfer-Encoding.

5. Error Disposition Modifier

For a received message where a receipt has been requested and decompression fails, the following disposition modifier will be returned in the signed MDN.

"Error: decompression-failed" - the receiver could not decompress

6. EDIINT Version Header Field

Any application that supports the compression methods outlined within this document MUST use a version identifier value of "1.1" or greater within the AS2 or AS3 Version header field as describe in [AS2] and [AS3].

7. Compression Formats

Implementations MUST support ZLIB [ZLIB] which utilizes DEFLATE[DEFLATE].

8. Security Considerations

This document is not concerned with security, except for any security concerns mentioned in the referenced RFCs.

9. IANA Considerations

This document has no actions for IANA.

Author's Addresses

Terry Harding
Axway
Scottsdale, Arizona, USA
tharding@us.axway.com

References

Normative References

[AS1] T. Harding, R. Drummond, C. Shih, MIME-based Secure Peer-to-Peer Business Data Interchange over the Internet, RFC 3335, Sept 2002

[AS2] D. Moberg, R. Drummond, MIME-Based Secure Peer-to-Peer Business Data Interchange Using HTTP, Applicability Statement 2 (AS2), RFC 4130, July 2005.

[AS3] T. Harding, R. Scott, FTP Transport for Secure Peer-to-Peer EDI over the Internet, RFC 4823, May 2007.

[ZLIB] RFC1950 ZLIB Compressed Data Format Specification version 3.3, P.Deutsch and J-L Gailly, May 1996.

[DEFLATE] RFC1951 DEFLATE Compressed Data Format Specification version 1.3, P.Deutsch, May 1996.

[MIME-TYPES] "Media Types," http://www.isi.edu/in-notes/iana/assignments/media-types/media-types.

[RFC1847] Security Multiparts for MIME: Multipart/Signed and Multipart/Encrypted, J. Galvin, S. Murphy, S. Crocker, N. Freed RFC 1847, October 1995.

[RFC2119] Key Words for Use in RFC's to Indicate Requirement Levels, S.Bradner, March 1997.

[S/MIME3.1]S/MIME Version 3.1 Message Specification, B.Ramsdell, July 2004. RFC 3851

[XMLTYPES] M. Murata, S. St.Laurent, D. Kohn, "XML Media Types", RFC 3023, January 2001.

[COMPRESSED-DATA] Compressed Data Content Type for Cryptographic Message Syntax (CMS), P. Gutmann, RFC 3274, June 2002.

Acknowledgements

A number of the members of the EDIINT Working Group have also worked very hard and contributed to this document. The following people have made direct contributions to this document.

David Fischer, Dale Moberg, Robert Asis and everyone involved in the AS1, AS2 Interop testing during 2002.

Copyright Statement

Intellectual Property

Expires February 27, 2009