none
Internet-Draft
Updates: 3156, 8551 (if approved)
Intended status: Informational
Expires: 1 November 2026

A. Gallagher, Ed.
PGPKeys.EU
D. K. Gillmor
ACLU
K. Engert
Thunderbird
30 April 2026

Unobtrusive End-to-End Email Signatures
draft-ietf-mailmaint-unobtrusive-signatures-02

Abstract

This document deals with end-to-end cryptographically signed email. It introduces a structure for signed email that is designed to avoid creating any disturbance in legacy email clients. This "unobtrusive" signature structure removes disincentives for signing email.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://andrewgdotcom.gitlab.io/unobtrusive-signatures/. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-mailmaint-unobtrusive-signatures/.

Discussion of this document takes place on the MAILMAINT Working Group mailing list (mailto:mailmaint@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/mailmaint/. Subscribe at https://www.ietf.org/mailman/listinfo/mailmaint/.

Source for this draft and an issue tracker can be found at https://gitlab.com/andrewgdotcom/unobtrusive-signatures/.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

Several different standard structures for end-to-end cryptographically signed email exist (see Sections 4.1.1.1, 4.1.1.2 and 4.1.2.1 of [RFC9787]). But the existing mechanisms have some undesirable properties which can make such mail difficult for the recipient to handle in some instances, particularly when read by legacy email clients that don't understand the signing structure. This document offers another signed email structure, which is designed to be transparent to legacy email clients.

The goal of this mechanism is to help email clients commit to signing every outbound message, which reduces complexity for the user of the mail client. The mechanism is capable of working with any signature mechanism, as well as transporting multiple signatures over a single message. It is specified initially for [OpenPGP] and [CMS], but can be extended to be used with other signature formats.

This mechanism is intended only for signed-only messages. A message that is encrypted-and-signed MUST NOT use this mechanism, since any existing MUA that can decrypt an encrypted-and-signed message already handles the signatures on such a message correctly.

This document updates [RFC3156] by providing an additional mechanism for producing and consuming OpenPGP-signed MIME email. This document also updates [RFC8551] by providing an additional mechanism for producing and consuming CMS-signed MIME email.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

* "Signed Mail" is used to refer to Internet Mail Messages that are cryptographically signed by the sender of the message, and expected to be validated by the recipient of the message. This document does not consider any cryptographic signature mechanism that is not end-to-end (such as [DKIM]), and should be agnostic to and non-interfering with any such mechanism.

* "OpenPGP Signature" refers to an OpenPGP Detached Signature as described by Section 10.4 of [OpenPGP].

* "CMS Signature" refers to an application/pkcs7-signature object as described by Section 3.5.3.1 of [RFC8551].

* "MUA" refers to a Mail User Agent, which is also known as an email client. For end-to-end signed mail, the sender's MUA performs message composition and injection into the mail system, and the receiver's MUA performs message ingestion from the mail system and rendering to the user.

* "Legacy MUA" refers to a MUA that does not know about this specification.

* "MTA" refers to a Message Transfer Agent, for example an SMTP server that relays Internet mail messages from one point to another.

* "CRLF" refers to "Carriage Return followed by Line Feed", the standard email line-ending sequence of two octets, CR (0x0D) and LF (0x0A).

3. Problems With Existing Signature Schemes

Existing end-to-end signature schemes for mail can trigger a set of annoyances for a recipient who uses a MUA that doesn't understand these structures. These annoyances can cause the recipient to complain to the sender. The easiest way for the sender to try to accommodate the recipient in this case is to simply not sign mail.

The Unobtrusive Signature scheme defined in this document is intended to minimize or eliminate all of these problems.

3.1. Unreadable Signed Mail

A signed mail message that uses the S/MIME PKCS #7 signed-data Cryptographic Layer described in Section 4.1.1.2 of [RFC9787] is unreadable by a receiving MUA that doesn't understand [CMS].

By contrast, a mail message signed with an Unobtrusive Signature should render normally by any legacy MUA.

3.2. Unknown Attachment

A signed mail message that uses the S/MIME Multipart Signed Cryptographic Layer described in Section 4.1.1.1 of [RFC9787] or the PGP/MIME Signing Cryptographic Layer (multipart/signed) described in Section 4.1.2.1 of [RFC9787] has a separate MIME part that contains the message signature.

A receiving MUA that doesn’t understand these structures will often render the signature as an "attachment". This can cause confusion and anxiety to the user of the MUA, and they will sometimes respond to the sender with the complaint "I can’t open your attachment".

By contrast, a mail message signed with an Unobtrusive Signature is merely encapsulated in a multipart/mixed outer layer. Legacy MUAs do not render such an encapsulation as an attachment.

3.2.1. Reducing Confusion with name Parameter

For existing end-to-end multipart signature schemes, one partial mitigation to this problem is to mark the signature part with an explicit filename that a legacy MUA is likely to display to the recipient. Concretely, some signing MUAs that generate multipart/signed messages using PGP/MIME ([RFC3156]) will add a name="openpgp-digital-signature.asc" parameter to the Content-Type header of the application/pgp-signature MIME part.

For recipients who understand what an OpenPGP digital signature is (even if their MUA can’t interpret it), this might reduce the amount of pushback they provide to the sender.

The Unobtrusive Signature scheme described in this document intends to offer even less friction to the recipient using a Legacy MUA by hiding the signature entirely.

3.3. Broken Signature

In some cases, mail is tampered with in transit, whether deliberately or maliciously. In this case, for a MUA that does understand these messages, some MUAs will visibly complain to the recipient that there is a failed signature.

If unsigned mail receives no comparable warning, then the act of adding a signature to a message that might traverse a modifiable path is risky. An MUA compliant with Section 6.4 of [RFC9787] will not create such a warning, but many MUAs do not yet comply with that guidance.

By contrast, a legacy MUA won’t render anything about the cryptographic status of an Unobtrusively Signed message at all. And an MUA compatible with this specification that encounters a message with a broken Unobtrusive Signature will never render an error that it wouldn’t have rendered on an unsigned message anyway, which removes this disincentive to sign.

4. Unobtrusively Signed Message

An Unobtrusively Signed Message has a specific MIME structure and uses a specific header field.

4.1. MIME structure

The top-level Content-Type of an unobtrusively signed message is multipart/mixed, and it has a single MIME subpart, which this specification refers to as the "Protected Part". The Protected Part’s header sections’ first header field is Sig, described in Section 4.2.

We hereby specify a third PGP/MIME format in addition to the two listed in Section 4.1.2 of [RFC9787]:

4.1.1. PGP/MIME Unobtrusive Signing Cryptographic Layer (multipart/mixed)

筋披澆笊エmultipart/mixed
 筋披楳笊エ[protected part]

This MIME layer offers authenticity and integrity if and only if the Protected Part contains one or more valid Sig headers.

This format is a Simple Cryptographic Envelope as specified in Section 4.4.1 of [RFC9787], and the Protected Part (with all leading Sig Header Fields removed) is the Cryptographic Payload.

This MIME structure MUST NOT be used as part of a Multilayer Cryptographic Envelope. If it is found anywhere but the outside of the message it MUST NOT be treated as a Cryptographic Layer.

4.2. Sig Header Field

This specification defines a new header field, named Sig. Sig is only meaningful if it appears in the Protected Part of an Unobtrusively Signed Message, before any non-Sig header field.

It contains parameters, only two of which are currently defined.

* The t parameter indicates the type of the signature with its value, and the only values currently defined are p (meaning an OpenPGP signature) and c (meaning a CMS signature). See Section 10.3.

* The b parameter contains a base64-encoded blob that contains the cryptographic signature object of the type described by t. Whitespace is ignored in this value and MUST be ignored when reassembling the original signature. In particular, the signing process can safely insert Folding White Space (Section 3.2.2 of [RFC5322]) in this value in arbitrary places to conform to line-length limits.

Note that if multiple Sig header fields appear in a single message, each Sig header field represents a signature over the Protected Part (with all leading Sig Header Fields removed). That is, each Sig signs the same content, and the order of the Sig header fields among themselves doesn't matter as long as every Sig header field precedes all non-Sig header fields in the Header Section of the Protected Part. Sig header fields MUST NOT appear in a non-leading position.

5. Sender Guidance

5.1. Always Use Header Protection

A message signed with an unobtrusive signature MUST always use [RFC9788], signing every header field known to the sending MUA at message composition time.

5.2. Message Composition

This updates the message composition function found in Section 5.1 of [RFC9787], using the same parameters.

* origbody: the traditional unprotected message body as a well-formed MIME tree (possibly just a single MIME leaf part). As a well-formed MIME tree, origbody already has structural header fields present.

* origheaders: the intended non-structural header fields for the message, represented here as a list of (h,v) pairs, where h is a header field name and v is the associated value.

* crypto: an indication that the message is to be signed with one or more Unobtrusive Signatures. This contains a list of one or more secret keys. Each key will make one signature.

The algorithm returns a MIME object that is ready to be injected into the mail system:

1. Create MIME tree inner as a copy of origbody

2. Ensure Content-Type Header Field of inner has parameter hp set to "clear"

3. For each header name and value (h,v) in origheaders:

    a. If h is Sig, skip that header, otherwise

    b. Add header h to inner with value v

4. Encode inner to reduce the risk of modification by MTAs (see Section 5.4)

5. Convert inner to bytestring innerbytes, applying canonicalization as per Section 5.5

6. For each signing key key in crypto:

    a. The signing system takes innerbytes and the signing key key, yielding a respective signature payload sig.

    b. Prepend a Header Field named Sig to inner with two parameters, t (set to the literal string p) and b (set to the base64-encoded value of sig).

7. Create new MIME tree output with Content-Type multipart/mixed, with a single subpart, set to inner

8. For each header name and value (h,v) in origheaders:

    a. Add header h to outer with value v

9. Return output

5.3. Do Not Use Unobtrusive Signature When Encrypting

In accordance with Section 5.2 of [RFC9787], when sending end-to-end encrypted messages an MUA MUST place end-to-end signatures inside the encrypted data. This mechanism is therefore not applicable to encrypted messages.

5.4. Formatting for Transit

The Protected Part (with all leading Sig Header Fields removed) SHOULD be formatted, for example by following the patterns described in Section 3 of [RFC3156], to make it less likely that it will be modified by MTAs:

* Content-Encoding is used to make the message 7-bit clean

* End of line trailing whitespace is stripped or encoded to non-whitespace

* If any line begins with the string "From ", either the Quoted-Printable or Base64 MIME encoding MUST be applied, and if Quoted-Printable is used, at least one of the characters in the string "From " MUST be encoded

5.5. Canonicalization

To ensure that changes to line endings and trailing empty lines made by MTAs in transit do not invalidate the signature, the formatted Protected Part MUST be canonicalized before signing or verification.

Line endings in the message MUST be converted to CRLF format. This ensures that an OpenPGP signature over the message will be invariant for both binary and text mode signatures.

All empty lines at the end of the message body are ignored. An empty line is a line of zero length after removal of the line terminator. If there is no body or no trailing CRLF on the message body, a CRLF is added. In more formal terms, *CRLF at the end of the body is converted to a single CRLF. Note that a completely empty or missing body is canonicalized as a single CRLF; that is, the canonicalized length will be 2 octets.

The latter transformation is equivalent to the "simple" body canonicalization algorithm defined in Section 3.4.3 of [RFC6376].

5.6. OpenPGP Signature Details

The OpenPGP Signature is made over the canonical bytestring, and binary mode (OpenPGP Signature Type 0x00) SHOULD be used.

5.7. CMS Signature Details

The CMS signature is a CMS ContentInfo containing a single CMS object of type SignedData. The SignedData encapContentInfo eContent field MUST be absent. The signerInfos field contains the signatures for the canonical bytestring.

This specification is directly aligned with the specification of the application/pkcs7-signature media type in Section 3.5.3.1 of [RFC8551].

6. Recipient Guidance

6.1. Detecting an Unobtrusive Signature

A receiving MUA detects the presence of an unobtrusive signature on a message by verifying that:

* the message Content-Type is multipart/mixed, and
* there is exactly one top-level subpart (though that subpart itself may be multipart), and
* the Content-Type of that top-level subpart has parameter hp="clear", and
* the first header field of the top-level subpart is named Sig, and
* the top-level subpart has a From header field, and its addr-spec matches the addr-spec in the message's From header field.

This last requirement (matching From addr-specs) is an anti-spoofing measure, by analogy with Section 4.4 of [RFC9788].

6.2. Validating an Unobtrusive Signature

When validating an unobtrusive signature, the signature data (that is, the value of the b field) is converted from Base64 to binary format. When t=p, this decoding yields a binary-format OpenPGP Detached Signature. When t=c, it yields a CMS signature in PKCS7 format. The signed object is extracted from the multipart/mixed part by selecting every octet that comes after the CRLF that terminates the last leading Sig header, and before the CRLF that immediately precedes the trailing MIME boundary. The signed object is then canonicalized as described in Section 5.5. The canonicalized data is then passed to the signature verification routine as a raw bytestream.

6.3. Message Rendering and the Cryptographic Summary

If the message has at least one Unobtrusive Signature which validates, then the MUA renders the message as though the top-level subpart is the message itself. The Cryptographic Summary of the message SHOULD indicate that the message is signed-only, and that all header fields present in the top-level subpart share that Cryptographic Status.

6.3.1. Example Rendering

For example, consider a message with this structure:

```
A 笏披滂笊ｴmultipart/mixed
B  笏披滂笊ｴmultipart/alternative; hp="clear" [Cryptographic Payload]
C   笏懌棋笊ｴtext/plain
D   笏披棋笊ｴtext/html
```

If at least one Unobtrusive Signature is present as a leading Sig header field in B, and it validates correctly, the message should be rendered the same way as this message:

```
B 笏披滂笊ｴmultipart/alternative
C  笏懌棋笊ｴtext/plain
D  笏披棋笊ｴtext/html
```

And its Cryptographic Status will be signed-only.

6.4. Consistency with Summary View for Tampered Messages

Many MUAs have two different views of a given message:

* A summary view, when rendering an overview of the contents of a mailbox, for example showing only From, Subject, Date, and "unread" status information for any given message, and

* A message view, for displaying the message itself, with its header context, cryptographic summary, and full contents.

The user reasonably expects that, for any given message, the information available in the summary view should match the message view.

Some MUAs render the summary view after ingesting the full message, but other MUAs might render the summary view without ever accessing anything other than the Outer Header Section. If the latter style of MUA gains access to a full message that has a valid unobtrusive signature, it can construct a candidate summary view using the signed

header field information. If the candidate summary view differs from the already displayed summary view, then the Outer Header Section has most likely been tampered with in transit.

The MUA MUST NOT render the message view in such a way that its header information does not match the summary view, as this will lead to user confusion about the message itself.

In such a situation, the MUA has two reasonable choices:

* The MUA MAY treat the unobtrusive signature as invalid, and show a message view that aligns with the already displayed summary view by rendering only the Outer Header Section. Such a message would have a cryptographic summary of unprotected.

* The MUA MAY accept the unobtrusive signature (yielding a cryptographic summary of signed-only), and update the summary view to use the candidate summary view instead. Such an updated summary view may surprise a user who is used to the summary view only sustaining minor changes (e.g., from "unread" to "read") upon rendering the message view.

A more complex approach in such a situation would be for the MUA to alert the user that the message may have been tampered with in transit, and allow them to choose to view either form of the message. This is similar to the approach described in Section 6.2.1.1 of [RFC9787].

Note that an MUA that renders the summary view only after evaluating the full message will never encounter this problem, as the summary view will be fully aligned with the message view from the start.

Note also that this concern applies to all forms of signed-only mail with header protection, not just to mail protected with an unobtrusive signature.

6.4.1. Unprotected Header Fields Added In Transit

As noted in Section 7 of [RFC9788], it’s possible that a MUA encounters some Header Fields on the outer message (in the Header Section of A in the example above) which could not have been known by the sender.

If any such fields would normally be rendered in some fashion by the MUA on an unsigned message, it MAY consider rendering them even on a signed-only Unobtrusively Signed message, but it should take care to indicate that they do not share the signed-only Cryptographic Status with the rest of the message.

6.5. Signature Failure Handling

Sometimes a receiving MUA encounters an unobtrusively signed message where all unobtrusive signatures fail to validate. The receiving MUA MUST NOT present the user with a cryptographic status that is different from a message with no signature at all. That is, the message's Cryptographic Status SHOULD be unprotected.

If a message gets tampered with in such a way that all unobtrusive signatures are broken, the recipient should see the message as though it were a normal unsigned message.

6.6. Handling Multiple Signatures

If more than one unobtrusive signature is present in a message, the receiving MUA MUST verify each signature against the known certificates associated with the indicated sender. As long as one of the signatures validates, the message should be treated as correctly signed, even if all the other signatures are invalid.

6.6.1. Multiple OpenPGP Signatures

Note that when a message is signed by multiple OpenPGP keys, the composer MAY structure the message with each OpenPGP signature packet (see Section 5.2 of [OpenPGP]) in its own Sig: t=p header field, or it MAY pack the OpenPGP signature packets together into a single OpenPGP Detached Signature (see Section 10.4 of [OpenPGP]) and place them in a single Sig: t=p header field. The verifying implementation MUST consider all appropriately placed Sig: t=p header fields, regardless of how many signature packets are included in each header field. It MAY coalesce the decoded b= data from multiple Sig: t=p header fields into a single OpenPGP Detached Signature (by simply concatenating the base64-decoded b values) before attempting verification.

Some MIME parsers have a fixed upper bound on the size of any MIME header field. A composer signing the message with more than one key should consider the size of the OpenPGP signatures when generating the Sig: t=p header fields to avoid breaking the message for recipients who use those constrained parsers. If the total size of the cumulative signature packets are very large, the composer MAY split up the OpenPGP Detached Signature at OpenPGP Signature packet boundaries, and place each disaggregated OpenPGP Detached Signature into a separate header field.

A good rule of thumb is to ensure each Sig: t=p header field is no larger than 50 KiB.

6.6.2. Multiple CMS Signatures

When a message is signed by multiple CMS keys, each CMS signature SHOULD be placed in its own Sig: t=c header field.

6.7. Ignore Out-of-place Unobtrusive Signatures

An unobtrusive signature Sig header field MUST NOT be evaluated unless:

* it is within the MIME headers of the only subpart of a multipart/mixed message, and

* it appears before any non-Sig header field

Evaluating a Sig header outside of this location might mean that a modified message could still appear to be successfully verified. For example, an unobtrusively signed message might be included as a sub-part of another multipart message, or be transformed into a non-MIME message with different message headers than the original email. This could conceivably be used by an attacker to make subtle changes to the meaning of a message without altering the content of the Protected Part.

7. MTA Guidance

An MTA or any other message relay service that observes a message with Content-Type multipart/mixed that is a single part MUST NOT alter the content of this message body in any way, including, but not limited to, changing the content transfer encoding of the body part or any of its encapsulated body parts. This corresponds to the guidance in Section 2.1 of [RFC1847] about the first section of multipart/signed messages.

8. Security Considerations

Based on the principle that "a broken signature is the same as no signature", a receiving MUA MUST NOT display any warnings if an Unobtrusive Signature fails to verify, unless the user has requested debugging output. This is because if an MITM can modify a message in transit, then they can choose whether or not to also remove the (now invalid) signature. If the receiving MUA displayed a more severe warning for a broken signature than for a missing one (or vice versa), the MITM could choose to modify the message in such a way that would result in the less-severe warning. The warning message is thus attacker-controlled.

Otherwise, the security properties are equivalent to those of a multipart/signed message.

9. Performance Considerations

9.1. Rationale for Signature in MIME Part

An alternate design considered for unobtrusive signatures was to simply place the Sig header in the Outer Header Section of the message itself, without requiring any additional MIME structure. This was rejected in favor of the MIME structure described in Section 4.1 for the following reasons:

* Unobtrusive signatures always offer Header Protection aligned with [RFC9788], so the signature needs to be able to cover those Header Fields generated by the sending MUA. But we know that most received messages contain a mix of Header Fields generated by the sending MUA and Header Fields injected by MTAs that touch the message in transit.

* Placing the signature as a Header Field in the Outer Header Section raises challenges in identifying which Header Fields are covered by the signature.

* An MTA is more likely to modify, reorder, or enforce limits on Header Fields in the message’s Outer Header Section than it is to corrupt Header Fields in the subpart.

* Any DKIM signature that includes the body of the message will cover the end-to-end signature if it is part of the message body. If the end-to-end signature was in the message’s Outer Header Section it would not normally be signed by DKIM, and would be vulnerable to inadvertent breakage by naive MTAs.

9.2. No One-pass Message Generation

Because the signature is included first in the message, it is not possible to generate the message in a single pass.

A sending MUA that needs to generate a signed outbound message in a single pass should use another end-to-end signing mechanism, like multipart/signed.

10. IANA Considerations

10.1. Register the Sig Header Field

IANA is requested to update the Permanent Message Header Field Names registry to add the following entry:

Header Field Name	Protocol	Status	Trace	Reference
Sig	MIME	informational	no	This document

Table 1: Permanent Message Header Field Names

The registration template called for in Section 4.2.1 of [RFC3864] is:

Template Field	Value
Header field name	Sig
Applicable protocol	MIME
Status	informational
Author/Change controller	IETF
Specification document(s)	This document
Related information	RFC9580 describes OpenPGP detached signatures, RFC8551 describes application/pkcs7-signature objects.

Table 2: Permanent Message Header Field Registration Template for Sig

10.2. Create Registry for Sig Message Header Parameters

IANA is requested to create a registry titled "Sig Message Header Parameters" in the "Message Headers" group of registries, with the following initial contents:

Name	Description	Reference
t	Type of Signature (see Section 10.3)	This document
b	Base64-encoded Signature Content (whitespace permitted and ignored)	This document

Table 3: Sig Message Header Parameters

((TODO: do we need a registry for this? Are we expecting any new parameters?))

10.3. Create Registry For t Parameter

IANA is requested to create a registry titled "Sig Message Header Signature Types" in the "Message Headers" group of registries, with the following initial contents:

Value	Description	Reference
p	An OpenPGP Detached Signature	This document
c	A CMS Signature	This document

Table 4: Sig Message Header Signature Types

10.4. Update multipart/mixed to Refer Here

IANA is requested to update the "multipart/mixed" entry in the Media Types registry, to add a reference to this document.

10.5. Registration Policies

IANA is requested to set all registries within this document to use the SPECIFICATION REQUIRED registration policy, see Section 4.6 of [RFC8126]. This policy means that review and approval by a designated expert is required, and that the IDs and their meanings must be documented in a permanent and readily available public specification, in sufficient detail so that interoperability between independent implementations is possible.

11. References

11.1. Normative References

[CMS] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <https://www.rfc-editor.org/rfc/rfc5652>.

[OpenPGP] Wouters, P., Ed., Huigens, D., Winter, J., and Y. Niibe, "OpenPGP", RFC 9580, DOI 10.17487/RFC9580, July 2024, <https://www.rfc-editor.org/rfc/rfc9580>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC3156] Elkins, M., Del Torto, D., Levien, R., and T. Roessler, "MIME Security with OpenPGP", RFC 3156, DOI 10.17487/RFC3156, August 2001, <https://www.rfc-editor.org/rfc/rfc3156>.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <https://www.rfc-editor.org/rfc/rfc5322>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/rfc/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8551] Schaad, J., Ramsdell, B., and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification", RFC 8551, DOI 10.17487/RFC8551, April 2019, <https://www.rfc-editor.org/rfc/rfc8551>.

[RFC9787] Gillmor, D. K., Ed., Melnikov, A., Ed., and B. Hoeneisen, Ed., "Guidance on End-to-End Email Security", RFC 9787, DOI 10.17487/RFC9787, August 2025, <https://www.rfc-editor.org/rfc/rfc9787>.

[RFC9788] Gillmor, D. K., Hoeneisen, B., and A. Melnikov, "Header Protection for Cryptographically Protected Email", RFC 9788, DOI 10.17487/RFC9788, August 2025, <https://www.rfc-editor.org/rfc/rfc9788>.

11.2. Informative References

[DKIM] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, DOI 10.17487/RFC6376, September 2011, <https://www.rfc-editor.org/rfc/rfc6376>.

[I-D.bre-openpgp-samples] Einarsson, B. R., "juga", and D. K. Gillmor, "OpenPGP Example Keys and Certificates", Work in Progress, Internet-Draft, draft-bre-openpgp-samples-03, 8 May 2025, <https://datatracker.ietf.org/doc/html/draft-bre-openpgp-samples-03>.

[RFC1847] Galvin, J., Murphy, S., Crocker, S., and N. Freed, "Security Multiparts for MIME: Multipart/Signed and Multipart/Encrypted", RFC 1847, DOI 10.17487/RFC1847, October 1995, <https://www.rfc-editor.org/rfc/rfc1847>.

[RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", BCP 90, RFC 3864, DOI 10.17487/RFC3864, September 2004, <https://www.rfc-editor.org/rfc/rfc3864>.

[RFC6376] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, DOI 10.17487/RFC6376, September 2011, <https://www.rfc-editor.org/rfc/rfc6376>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/rfc/rfc7942>.

[RFC9216] Gillmor, D. K., Ed., "S/MIME Example Keys and Certificates", RFC 9216, DOI 10.17487/RFC9216, April 2022, <https://www.rfc-editor.org/rfc/rfc9216>.

Appendix A. Test Vectors

These test vectors show different examples of unobtrusive signed messages.

A.1. From Alice to Bob

The message below is a common multipart/alternative email, signed with an unobtrusive signature. The signature should be verifiable using the "Alice" v4 certificate found in Section 2.1.1 of [I-D.bre-openpgp-samples].

```
Content-Type: multipart/mixed; boundary="5d6"
MIME-Version: 1.0
From: Alice Lovelace <alice@openpgp.example>
To: Bob Babbage <bob@openpgp.example>
Subject: This is a Test
Date: Thu, 01 May 2025 22:16:15 -0400
Message-ID: <uosig-0@openpgp.example>

--5d6
Sig: t=p; b=wnUEABYKAB0WIQTrhbtfozp14V6UTmPyMVUMT0fjjgUCaBQq
 7wAKCRDyMVUMT0fjjr+3AP4nGDsaptk9I6EePoXftyevyH6luB2aSAzrD8o
 xQVNWDQD/VQ/s85C3v6SAxtFDcBsn2H32Hd/yW5BsDx62gmpL7Aw=
MIME-Version: 1.0
From: Alice Lovelace <alice@openpgp.example>
To: Bob Babbage <bob@openpgp.example>
Subject: This is a Test
Date: Thu, 01 May 2025 22:16:15 -0400
Message-ID: <uosig-0@openpgp.example>
Content-Type: multipart/alternative; boundary="913"; hp="clear"

--913
Content-Type: text/plain; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit

Hi Bob,

This is Alice.  I need you to:

- read this message
- reply to it
- delete it promptly.

Thanks,
Alice
--913
Content-Type: text/html; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit

<html><head></head><body><p>Hi Bob,</p>
<p>This is Alice.  I need you to:</p>
<ul>
<li>read this message</li>
<li>reply to it</li>
<li>delete it promptly.</li>
</ul>
<p>Thanks,
```

```
   Alice</p></body></html>
   --913--

   --5d6--
```

A.2. From David to Alice

The message below is a simple text/plain email, signed with an unobtrusive signature. The signature should be verifiable using the "David" certificate found in Section 5.1 of [I-D.bre-openpgp-samples].

```
   Content-Type: multipart/mixed; boundary="a21"
   MIME-Version: 1.0
   From: David Deluxe <david@openpgp.example>
   To: Alice Lovelace <alice@openpgp.example>
   Subject: Checking in
   Date: Fri, 02 May 2025 13:01:07 -0400
   Message-ID: <uosig-1@openpgp.example>

   --a21
   Sig: t=p; b=wpIGABsIAAAAKSIhBkGZ2eqmaCp41aU09iv3YiKlTk3rx4Xb
    5qbFs0WGAm/iBQJoFPpTAAAACgkQQZnZ6qZoKnhDIRA9tgkt50eA1ckzilm
    9KndQt3t4iYlab66bvtP+kP9D7zaNzvC1vE+B6jPY1gUBOQMyF5CK3yC/xZ
    Ol2ww+x8Y3PZ7OpZ1dPUlshDL5gA7ZAw==
   MIME-Version: 1.0
   Content-Transfer-Encoding: 7bit
   From: David Deluxe <david@openpgp.example>
   To: Alice Lovelace <alice@openpgp.example>
   Subject: Checking in
   Date: Fri, 02 May 2025 13:01:07 -0400
   Message-ID: <uosig-1@openpgp.example>
   Content-Type: text/plain; charset="us-ascii"; hp="clear"

   Alice!

   So good to see you earlier.

   I hope you will have a chance to check out
   our new website: https://openpgp.example/
   and tell me what you think.

   All the best,

   David

   --a21--
```

A.3. From Alice to David

The message below is a multipart/alternative email with an image attached, signed with an unobtrusive signature. The signature should be verifiable using the "Alice" v4 certificate found in Section 2.1.1 of [I-D.bre-openpgp-samples].

```
Content-Type: multipart/mixed; boundary="3e4"
MIME-Version: 1.0
From: Alice Lovelace <alice@openpgp.example>
To: David Deluxe <david@openpgp.example>
Subject: Re: Checking in
Date: Fri, 02 May 2025 17:03:35 -0400
Message-ID: <uosig-2@openpgp.example>
In-Reply-To: <uosig-1@openpgp.example>
References: <uosig-1@openpgp.example>

--3e4
Sig: t=p; b=wnUEABYKAB0WIQTrhbtfozp14V6UTmPyMVUMT0fjjgUCaBUz
 JwAKCRDyMVUMT0fjjmnRAQDKnIfyPyvE2lVlVOQl+H99TK+VFCvBaTZyTAV
 xnKgJ1gEAjVDQ3idx4Z4wSN+pLhWS1LdpVbWdH7mW58gS0GBz5AM=
MIME-Version: 1.0
From: Alice Lovelace <alice@openpgp.example>
To: David Deluxe <david@openpgp.example>
Subject: Re: Checking in
Date: Fri, 02 May 2025 17:03:35 -0400
Message-ID: <uosig-2@openpgp.example>
In-Reply-To: <uosig-1@openpgp.example>
References: <uosig-1@openpgp.example>
Content-Type: multipart/mixed; boundary="d64"; hp="clear"

--d64
Content-Type: multipart/alternative; boundary="f4f"
MIME-Version: 1.0

--f4f
Content-Type: text/plain; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit

Hi David,

I think the attached logo might look good
on the website.

Thanks,
Alice
```

```
--f4f
Content-Type: text/html; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit

<html><head></head><body><p>Hi David,</p>
<p>I think the attached logo might look good
on the website.</p>
<p>Thanks,
Alice</p></body></html>
--f4f--

--d64
Content-Type: image/png
Content-Transfer-Encoding: base64
Content-Disposition: inline; filename="logo.png"

iVBORw0KGgoAAAANSUhEUgAAABQAAAAUCAYAAACNiR0NAAAAcElEQVR42uVTOxbA
MAgS739nO3TpRw20dqpbfARQEjOywiwYnCtkDKnbcLk66sqlT+zt9cidkE+6KwkZ
sgrzfcqVMpL2jo0447gYDpeArk+OnJHkIhAfTPRicihAf5YJrw7vjv0ZWRWM/uli
vdPf1QZ2kDD9xppd8wAAAABJRU5ErkJggg==

--d64--

--3e4--
```

A.4. Alice to David Followup

The message below is a multipart/alternative email that is a self-reply from about a week later. In the meantime, Alice has gotten a new OpenPGP certificate, so the message is signed with both her old key and her new key. This message's signatures should be verifiable respectively using either the "Alice" v4 certificate found in Section 2.1.1 of [I-D.bre-openpgp-samples] or the "Alice" v6 certificate found in Section 2.2.1 of [I-D.bre-openpgp-samples].

```
Content-Type: multipart/mixed; boundary="0cd"
MIME-Version: 1.0
From: Alice Lovelace <alice@openpgp.example>
To: David Deluxe <david@openpgp.example>
Subject: Re: Checking in
Date: Thu, 08 May 2025 18:41:05 -0400
Message-ID: <uosig-3@openpgp.example>
In-Reply-To: <uosig-2@openpgp.example>
References: <uosig-2@openpgp.example>

--0cd
Sig: t=p; b=wnUEABYKAB0WIQTrhbtfozp14V6UTmPyMVUMT0fjjgUCaB0z
```

```
    AQAKCRDyMVUMT0fjjtCJAQCLvEeDH/grJ9szJTEPumRz0lvQm1f3GHNuTnS
    W9+SV/wD/YpPK4oMy2Cbrzo9JagpO4uxXkbCWQIHI9HFlwkz8Hg0=
   Sig: t=p; b=wogGABsIAAAAKSIhBuRqR5oGQqpTb/U1uxxDl7NeiBI/TgFl
    Z9LvdROjBBHyBYJp1PpIAAAAAMXUEJmS8XYBIAZAiRxhUsVSePxF7QMK9CA
    V/4u2hOAfg/z5Tg/p9B8TB9ydm81DI/+ltru4grSnvFL2JKYMrdOxVMY40u
    NugvrgF+YyAGHahkMN
   MIME-Version: 1.0
   From: Alice Lovelace <alice@openpgp.example>
   To: David Deluxe <david@openpgp.example>
   Subject: Re: Checking in
   Date: Thu, 08 May 2025 18:41:05 -0400
   Message-ID: <uosig-3@openpgp.example>
   In-Reply-To: <uosig-2@openpgp.example>
   References: <uosig-2@openpgp.example>
   Content-Type: multipart/alternative; boundary="97a"; hp="clear"

   --97a
   Content-Type: text/plain; charset="us-ascii"
   MIME-Version: 1.0
   Content-Transfer-Encoding: 7bit

   Hey David,

   Also, please spell my name correctly in the
   website’s acknowledgements section.

   Kind regards,
   Alice

   --97a
   Content-Type: text/html; charset="us-ascii"
   MIME-Version: 1.0
   Content-Transfer-Encoding: 7bit

   <html><head></head><body><p>Hey David,</p>
   <p>Also, please spell my name correctly in the
   website’s acknowledgements section.</p>
   <p>Kind regards,
   Alice</p></body></html>
   --97a--

   --0cd--
```

A.5. From Carlos to Dana

The message below is a multipart/alternative email, signed with an unobtrusive CMS signature. The signature should be verifiable using the "Carlos" X.509 certificate from Section 7.1 of [RFC9216].

```
   Content-Type: multipart/mixed; boundary="b8f"
   MIME-Version: 1.0
   From: Carlos Turing <carlos@smime.example>
   To: Dana Hopper <dana@smime.example>
   Subject: Touching base on Project Scoop
   Date: Mon, 01 Dec 2025 20:41:05 -0400
   Message-ID: <uosig-4@smime.example>

   --b8f
   Sig: t=c; b=MIIDoAYJKoZIhvcNAQcCoIIDkTCCA40CAQExDTALBglghkgB
    ZQMEAgMwCwYJKoZIhvcNAQcBoIICCzCCAgcwggG5oAMCAQICEz9eH1Qk0bQ
    BQ3gPc8GKF4UedpYwBQYDK2VwMFkxDTALBgNVBAoTBElFVEYxETAPBgNVBA
    sTCExBTVBTIFdHMTUwMwYDVQQDEyxTYW1wbGUgTEFNUFMgRWQyNTUxOSBDZ
    XJ0aWZpY2F0aW9uIEF1dGhvcml0eTAgFw0yMDEyMTUyMTM1NDRaGA8yMDUy
    MTIxNTIxMzU0NFowOjENMAsGA1UEChMESUVURjERMA8GA1UECxMITEFNUFM
    gV0cxFjAUBgNVBAMTDUNhcmxvcyBUdXJpbmcwKjAFBgMrZXADIQDCzoAyLN
    5hyE2ETWDvkZznna6ufx7kB10j8gCmkvfIraOBsDCBrTAMBgNVHRMBAf8EA
    jAAMBcGA1UdIAQQMA4wDAYKYIZIAWUDAgEwATAfBgNVHREEGDAWgRRjYXJs
    b3NAc21pbWUuZXhhbXBsZTATBgNVHSUEDDAKBggrBgEFBQcDBDAOBgNVHQ8
    BAf8EBAMCBsAwHQYDVR0OBBYEFGSF4zucHVrN5gu6Gn8IvsSczIQ/MB8GA1
    UdIwQYMBaAFGuilX26FJvkLQTRB6TRguQua4y1MAUGAytlcANBAMFRkFm3c
    uhUCKUxbGlrxtv1Etn50ugfgc4Aq5BkCll9VoJE4+DBDqi/tHBVy6+2UNg0
    FKNol8kwLufAUem7XAkxggFbMIIBVwIBATBwMFkxDTALBgNVBAoTBElFVEY
    xETAPBgNVBAsTCExBTVBTIFdHMTUwMwYDVQQDEyxTYW1wbGUgTEFNUFMgRW
    QyNTUxOSBDZXJ0aWZpY2F0aW9uIEF1dGhvcml0eQITP14fVCTRtAFDeA9zw
    YoXhR52ljALBglghkgBZQMEAgOggYkwGAYJKoZIhvcNAQkDMQsGCSqGSIb3
    DQEHATAcBgkqhkiG9w0BCQUxDxcNMjUxMjAyMDA0MTA1WjBPBgkqhkiG9w0
    BCQQxQgRAKL1rWpLN627fT1QCjZRf+3Jhvr6SpbQREptLf0wBzIPktKLT4W
    nDO68KCYuZMD07gPaw9o47QxhC9C4CnXT3DDAFBgMrZXAEQOQGMe6IlrFn1
    owFikFVQ18/x9nqJ3FI2nJDOKv87VStnqsDOnMcjTLxSD5uGnKgUY15yD/Z
    p+oXksYt791AmQM=
   MIME-Version: 1.0
   From: Carlos Turing <carlos@smime.example>
   To: Dana Hopper <dana@smime.example>
   Subject: Touching base on Project Scoop
   Date: Mon, 01 Dec 2025 20:41:05 -0400
   Message-ID: <uosig-4@smime.example>
   Content-Type: multipart/alternative; boundary="12f"; hp="clear"

   --12f
   Content-Type: text/plain; charset="us-ascii"
   MIME-Version: 1.0
   Content-Transfer-Encoding: 7bit

   Ahoy Dana--

   Can you take a look at the most recent document on the W: drive
   related to Project Scoop?
```

```
   I am happy to talk with you about it Thursday.

   --Carlos

   --12f
   Content-Type: text/html; charset="us-ascii"
   MIME-Version: 1.0
   Content-Transfer-Encoding: 7bit

   <html><head></head><body><p>Ahoy Dana--</p>
   <p>Can you take a look at the most recent document on the W: drive
   related to Project Scoop?</p>
   <p>I am happy to talk with you about it Thursday.</p>
   <p>--Carlos</p></body></html>
   --12f--

   --b8f--
```

Appendix B. Document History

This section is to be removed before publishing as an RFC.

B.1. Changes Between draft-ietf-mailmaint-unobtrusive-signatures-01 and draft-ietf-mailmaint-unobtrusive-signatures-02

* Adjust canonicalization to align with DKIM simple.
* Fix one of the test vectors.
* Textual nits.

B.2. Changes Between draft-ietf-mailmaint-unobtrusive-signatures-00 and draft-ietf-mailmaint-unobtrusive-signatures-01

* Specified CMS signatures
* Added implementation status section

B.3. Changes Between draft-gallagher-email-unobtrusive-signatures-02 and draft-ietf-mailmaint-unobtrusive-signatures-00

* Working group adoption.

B.4. Changes Between draft-gallagher-email-unobtrusive-signatures-01 and draft-gallagher-email-unobtrusive-signatures-02

* Align IANA registration section with requests from IANA.

* Update references for drafts which are now RFCs.

* Permit multiple OpenPGP signature packets in each Sig header, aligning with OpenPGP "detached signature".

* Clarify signing process.

* Guidance for possible discrepancy between "summary" and "message" views if headers are not aligned.

B.5. Changes Between draft-gallagher-email-unobtrusive-signatures-00 and draft-gallagher-email-unobtrusive-signatures-01

* Made explicit that Sig MUST NOT appear in a non-leading position.

* Expanded design rationale section.

* Clarified use of Quoted-Printable encoding.

* Terminology and reference cleanup.

B.6. Changes Between draft-gallagher-email-invisible-signatures-00 and draft-gallagher-email-unobtrusive-signatures-00

* Updated sender canonicalization guidance from MUST to SHOULD.

* Registries changed to SPECIFICATION REQUIRED.

* Improved test vectors.

* Renamed to "Unobtrusive Signatures".

* Explicitly allow folding whitespace.

* Document existing convention re attachment filenames.

* Fixed references.

* Various clarifications to wording.

Appendix C. Implementation Status

This section is to be removed before publishing as an RFC.

RFC Editor: When this section is removed before publishing, please also remove the reference to [RFC7942].

As recommended in [RFC7942], this section describes the status of implementations known to the editors at the time of draft publication.

The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

C.1. emacs mml-mode

The mml MUA within emacs has a patch set for generating unobtrusive signatures (https://debbugs.gnu.org/78448). This code is distributed under the General Public License, version 3 or later.

C.2. Thunderbird

The Thunderbird MUA has a work-in-progress patch set (https://bugzilla.mozilla.org/show_bug.cgi?id=1958983) for generating and processing unobtrusive signatures. This code is distributed under the Mozilla Public License, version 2.0 or the General Public License, version 2.

Acknowledgments

The authors would like to thank the attendees of the 9th OpenPGP Email Summit for feedback and suggestions. Additionally, Anarcat and Barry Leiba offered useful feedback on the draft.

Authors' Addresses

Andrew Gallagher (editor)
PGPKeys.EU
Email: andrewg@andrewg.com

Daniel Kahn Gillmor
ACLU
Email: dkg@fifthhorseman.net

Kai Engert
Thunderbird

Email: kaie@thunderbird.net