RFC 9788 – Header Protection for Cryptographically Protected Email

by Bernie Hoeneisen

(Co-Author of RFC 9787 / RFC 9788)

Senior Technical Consultant & CEO

Ucom Standards Track Solutions GmbH

ISOC-CH Head of Public Policy

ISOC-CH Happy Hour - 29. October 2025 - Zurich

Table Of Contents

- Background, Motivation & Issues
- Pre-Existing Implementations
- New in RFC 9788
  - MIME Format
  - Legacy Display (Backward Compatibility)
  - Header Confidentiality Policy (HCP)
  - Protection against Tampering in Transit
  - Summary of further Content
- Summary of RFC 9787
- Q & A / Discussion

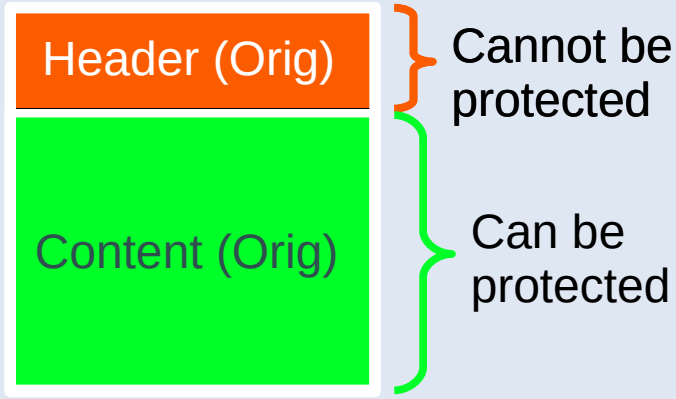
Background & Motivation

- Background
  - Cryptographically Protected Email has a lot of challenges
  - Such challenges include
    - Not widely implemented
    - Usability & rendering issues
    - Privacy issues
    - Security Issues
- Main Motivation
  - Address some of these challenge (mostly Privacy-related)
  - Backward compatibility with legacy Email implementations

Basic Issue

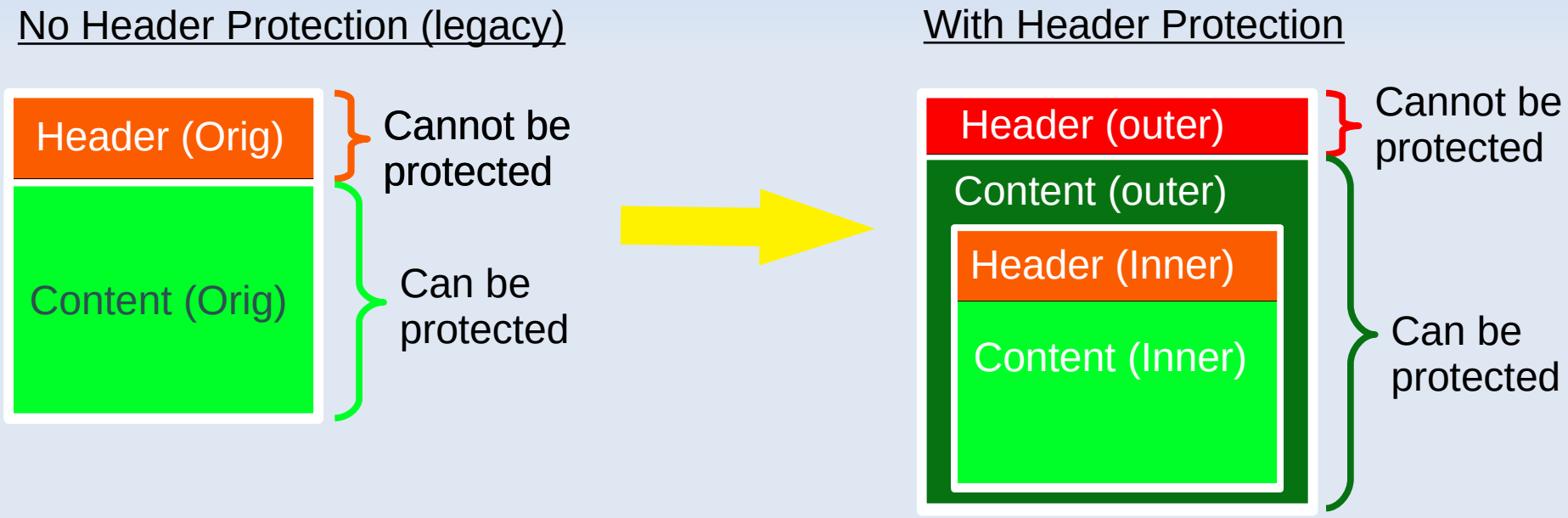
- Signing and/or encryption of email typically only included body
  - Header – containing e.g. *Subject* – remained **un**protected

No Header Protection (legacy)



Basic Issue

- Signing and/or encryption of email typically only included body
  - Header – containing e.g. *Subject* – remained **un**protected



Comparison of Pre-Existing Solutions

	RFC 8551	pEp (PEF-1)[1]	pEp (PEF-2)[2]	Autocrypt
Protection of Header Section	Whole message wrapped	*Subject* only	Whole message wrapped	**"Injected" to (top of) body**
Wrapper Envelope MIME Type	message/rfc822 || message/global	N/A	message/rfc822	N/A
Header Confidentiality Policy	Undefined	Subject only	Non-standard	Non-standard
Prevent tampering with Outer Header in transit	No	No	No[3]	No
Backward compatibiliy help for legacy rendering clients	Nothing specific	**Put *Subject* to body (first line)**	Information as first MIME node[3]	Legacy Display part as first MIME node
Rendering issues	Most clients	None	Most non-pEp clients[3]	Few clients
Reason (for rendering issues)	Wrapping	N/A	Wrapping	MIME structure

[1] Designed for compatibility with non-pEp clients (minimal solution)

[2] Designed for interoperability with pEp clients only

[3] Only relevant in special cases (e.g. mixture of pEp and non-pEp clients used)

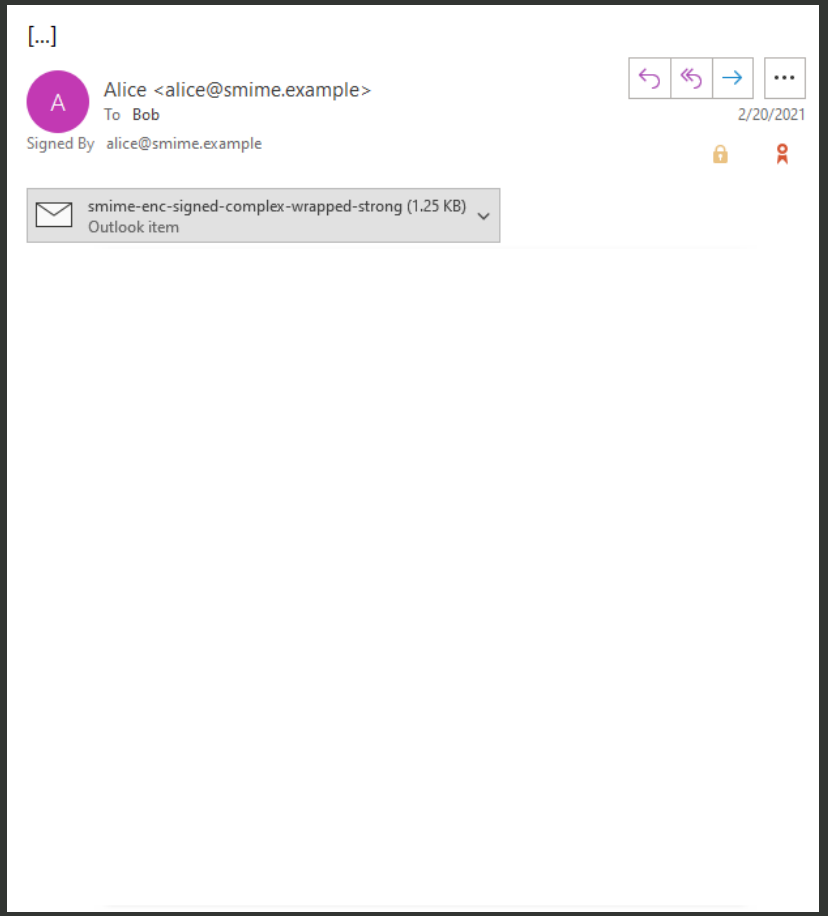
Rendering Issues

- Most legacy (crypto-capable) email clients render Header Protected messages badly, e.g.
  - Render protected message as attachment or forwarded message
  - Don’t render protected message at all
  - Show nuisanse-warnings
- Issues
  - Bad user experience or even message cannot be read at all
    → May result in avoidance of using encryption
  - Unclear protection states (after opening attachment)
  - Leaking of private parts at Reply or Forward

Outlook – Rendering Signed & Encrypted (1/2)

Wrapped vs. Injected

Wrapped (open attachment first)



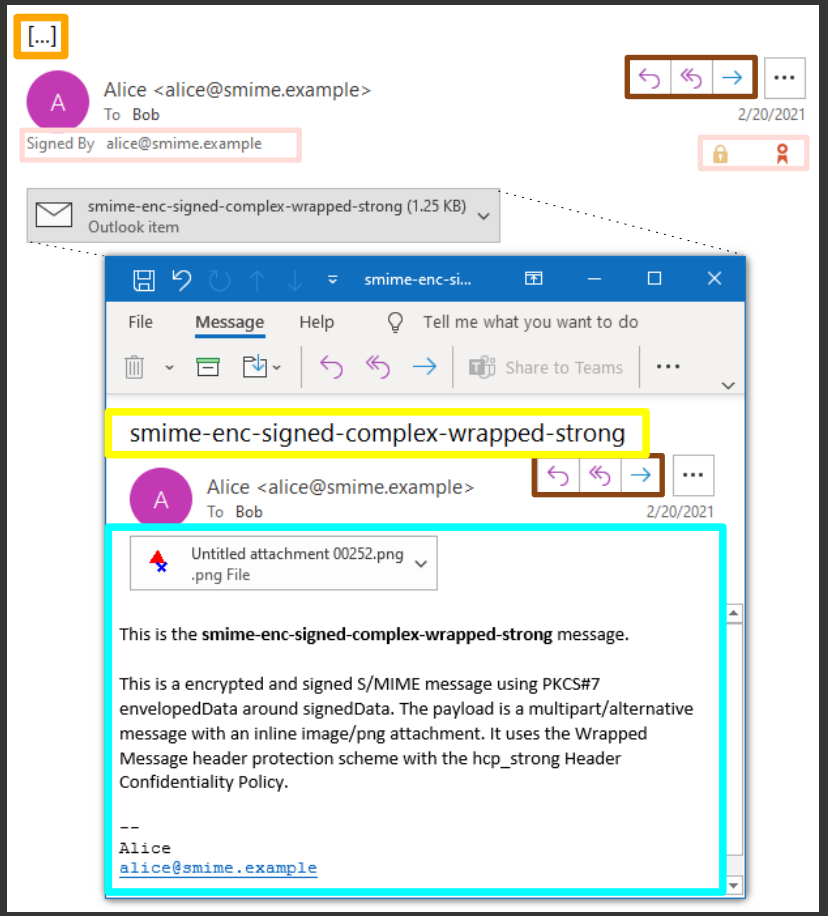
Injected (missing Subject)

MS Outlook 365 (2101 / 13628.20448)

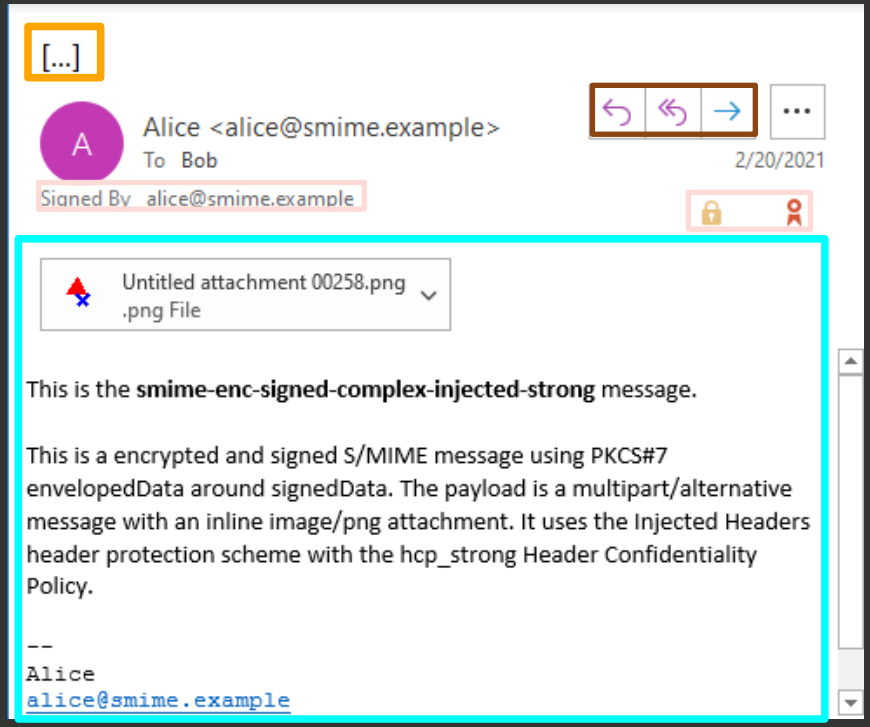
Outlook – Rendering Signed & Encrypted (1/2)

Wrapped vs. Injected

Wrapped (open attachment first)



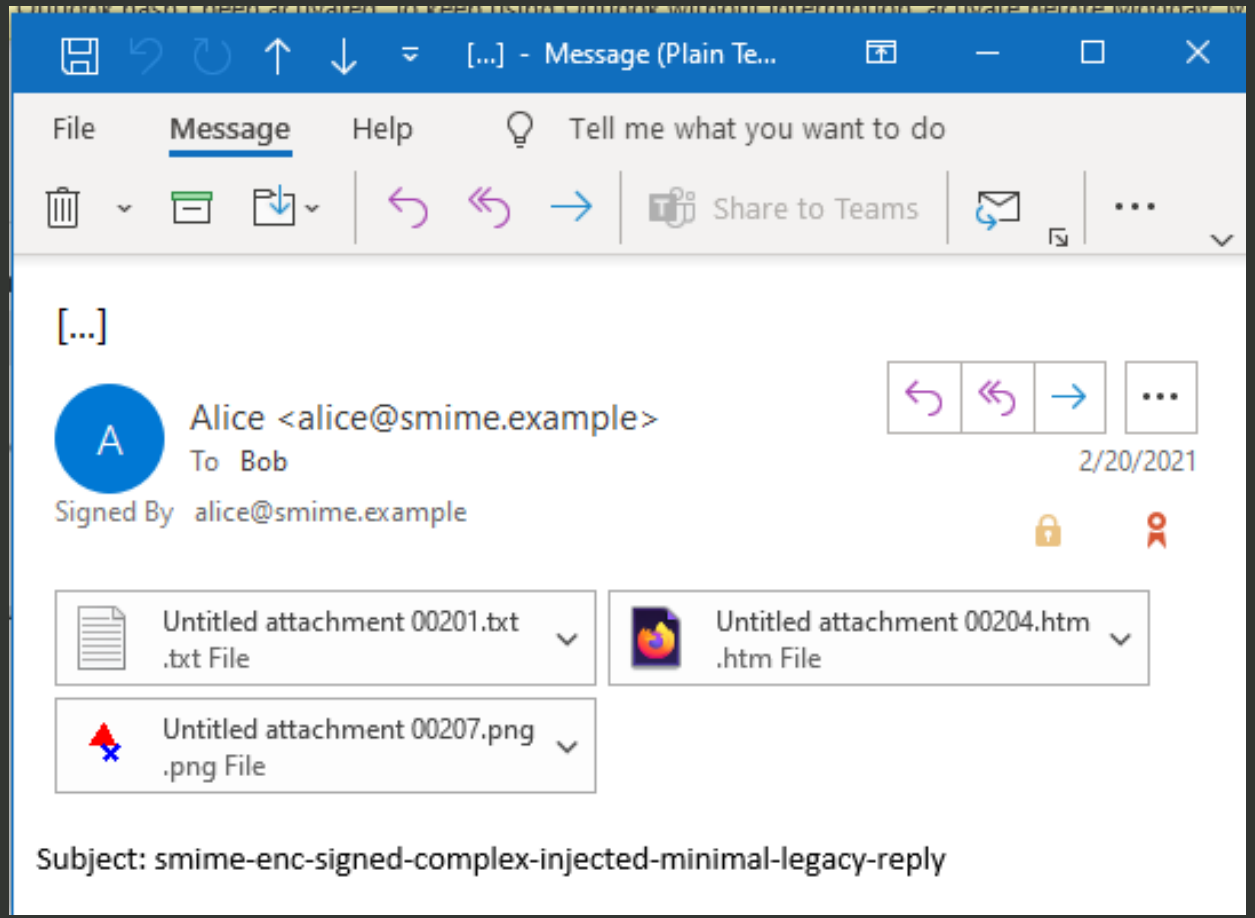
Injected (missing Subject)



MS Outlook 365 (2101 / 13628.20448)

Outlook – Rendering Signed & Encrypted (2/2)

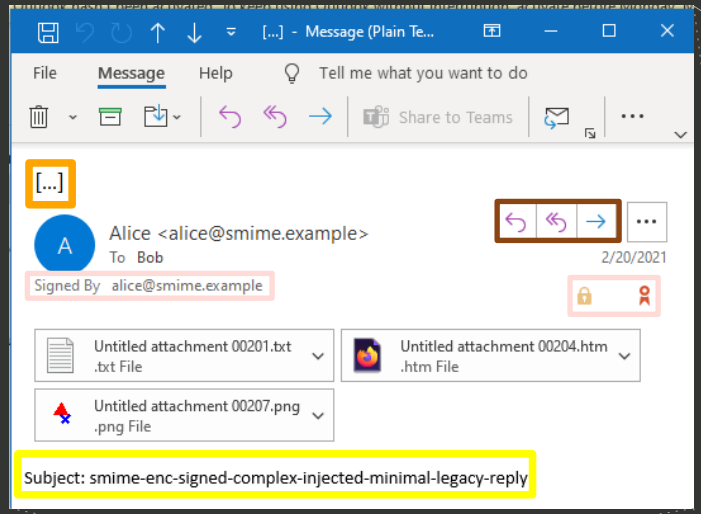
Injected Legacy Display MIME part



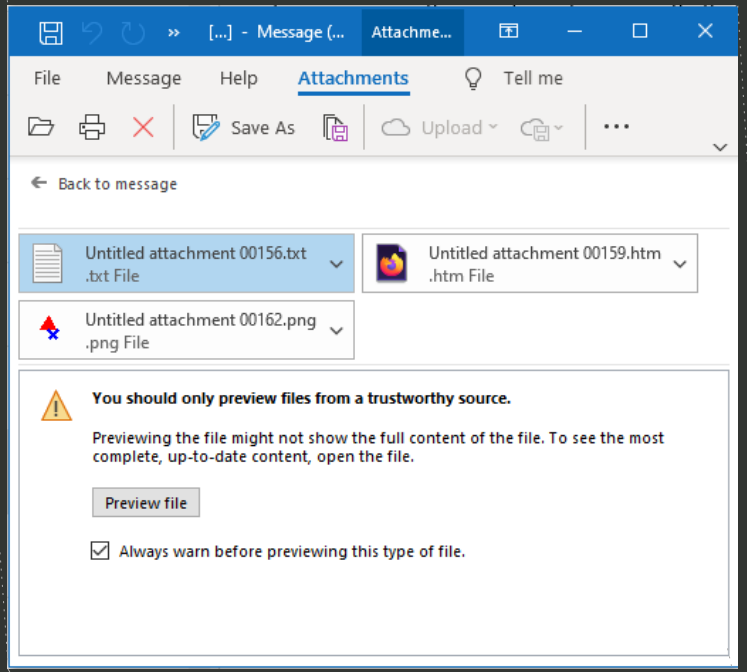
MS Outlook 365 (2101 / 13628.20448)

Outlook – Rendering Signed & Encrypted (2/2)

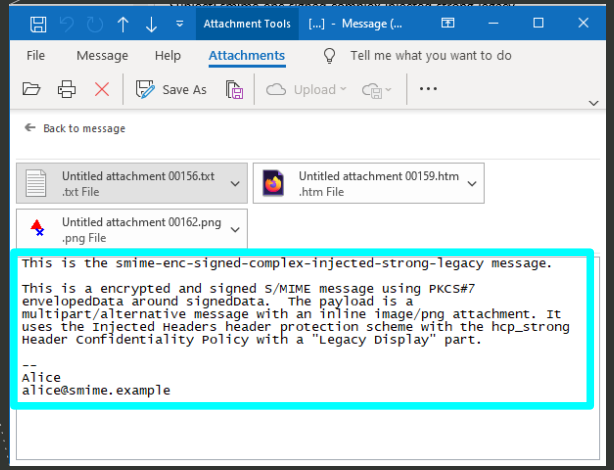
Injected Legacy Display MIME part

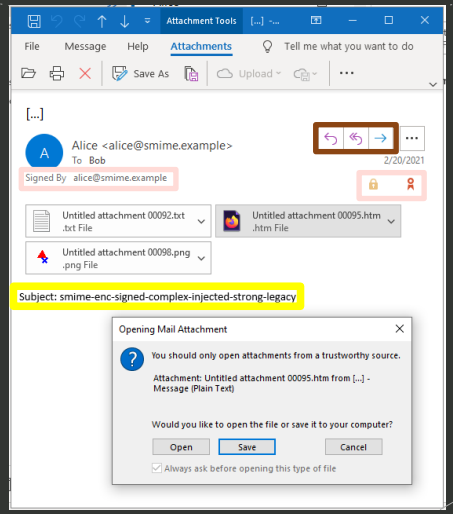


Injected Legacy Display MIME part (displays 3 attachments)



Injected Legacy Display, MIME Part (opens as text file, no reply button)





Injected Legacy Display MIME part (trying to open attachment leads to nuisance security warning)

Issues Addressed in RFC 9788

- Protect Header (not only Body)
- Ensure best possible backward compatibility with legacy Email implementations
- Avoid false sense of privacy and leaking private information into Reply or Forward, e.g.
  - If message is composed in an unpredictable manner
  - After tampering (with Outer Header) in transit, e.g.
    - Signed-only message encrypted in transit
    - Header Fields altered or removed in transit

MIME Format

- MIME Format of RFC 9788 (cryptographic envelope) (unchanged), e.g.

```
└─ multipart/alternative
   ├─ text/plain
   └─ text/html
```

- Comparison to RFC 8551 with wrapping (cryptographic envelope)

```
└─ message/rfc822
   └─ multipart/alternative
      ├─ text/plain
      └─ text/html
```

MIME Format Example

Outer Message Header Section (unprotected)

```
O: Date: Mon, 25 Sep 2017 17:31:42 +0000 (UTC)
O: From: john.doe@example.net
O: To: somebody@example.org
O: MIME-Version: 1.0
O: Subject: [...]
```

Outer Message Body (protected)

```
I: Date: Mon, 25 Sep 2017 19:31:42 +0200 (CEST)
I: From: "John Doe" <john.doe@example.net>
I: To: "Alice Wonder" <alice.wonder@example.org>
I: Message-ID: <e4a483cb-1dfb@m.example.net>
I: Subject: Meeting at my place
I: MIME-Version: 1.0
I: X-Mailer: iPhone Mail (23A355)
I: Content-Type: text/plain; charset=us-ascii

   This is an important message that I don't
   want to be modified.
```

Inner Message Header Section

Inner Message Body

Legacy Display Element (1/2)

- Most legacy (crypto-capable) clients render Outer Header Section
- Issue
  - If Header Fields in the Outer Header Section were hidden or obfuscated, those are invisible to the user, e.g.
    - *Subject: […]*
  - Only relevant if encrypted
- Means (transitional)
  - Add hidden or obfuscated (User Facing) Header Fields to top of the Inner Body as Legacy Display Element (LDE)
  - Mark (text, html) MIME parts containing LDE with *hp-legacy-display="1"*
    - Allows Header Protection aware rendering clients to safely remove them

Legacy Display Element (2/2)

Outer Message Header Section (unprotected)

```
O: Date: Mon, 25 Sep 2017 17:31:42 +0000 (UTC)
O: From: <john.doe@example.net>
O: To: somebody@example.org
O: MIME-Version: 1.0
O: Subject: [...]
```

Outer Message Body (protected)

```
I: Date: Mon, 25 Sep 2017 19:31:42 +0200 (CEST)
I: From: "John Doe" <john.doe@example.net>
I: To: "Alice Wonder" alice.wonder@example.org
I: Message-ID: <e4a483cb-1dfb@m.example.net>
I: Subject: Meeting at my place
I: MIME-Version: 1.0
I: X-Mailer: iPhone Mail (23A355)
I: Content-Type: text/plain; hp-legacy-display="1"

   Subject: Meeting at my place

   This is an important message that I don't
   want to be modified.
```

Inner Message Header Section

Inner Message Body

Header Confidentiality Policy (HCP)

- Means to define how the Outer Header Fields are constructed
  - Typically based on the Inner Header Fields
  - Only relevant if encrypted
- Example: Obfuscate *Subject* and hide *Cc*:

```
hcp_example_hide_cc(name, val_in) → val_out:
    if lower(name) is 'subject':
        return '[...]'
    else if lower(name) is 'cc':
        return null
    else:
        return val_in
```

Protection against Tampering in Transit (1/3)

- The Outer Header Section is still exposed to tampering in transit, e.g.
  - Outer Header Fields may be added, removed or modified
  - Signed-Only Message may be encrypted
- Issues
  - False sense of Privacy
  - Originally “private” Header Fields may be leaked into Reply or Forward Messages

Protection against Tampering in Transit (2/3)

- Means
  - Add Outer-Header Fields to Inner Header Section as Header Fields *HP-Outer*
    - Only relevant if encrypted
  - Specify the protection level of the Inner Message with Content Type Parameter *hp*:
    - *hp="cipher"* (if signed and encrypted)
    - *hp="clear"* (if signed only)

Protection against Tampering in Transit (3/3)

Outer Message Header Section (unprotected)

```
From: alice@example.org
To: bob@example.net
Date: Sat, 20 Feb 2024 15:09:02 +0000
Message-ID: <j4jjkl43-3n4@example.org>
Subject: [...]
```

Outer Message Body (protected)

```
MIME-Version: 1.0
Subject: smime-signed-enc-hp-baseline
Message-ID: <y74fkl35-9h5g@example.org>
From: Alice Wonderland <alice@example.org>
To: Bob Racer <bob@example.net>
Date: Sat, 20 Feb 2024 10:09:02 -0500
HP-Outer: From: alice@example.org
HP-Outer: To: bob@example.net
HP-Outer: Date: Sat, 20 Feb 2024 15:09:02 +0000
HP-Outer: Message-ID: <j4jjkl43-3n4@example.org>
HP-Outer: Subject: [...]
Content-Type: text/plain; charset="utf-8"; hp="cipher"

This is the smime-signed-enc-hp-baseline message.
```

Inner Message Header Section

Inner Message Body

RFC 9788 – ToC (1/2)

- Introduction
- Internet Message Format Extensions
  - Content-Type Parameters
    - Content-Type Parameter: hp
    - Content-Type Parameter: hp-legacy-display
  - HP-Outer Header Field
- Header Confidentiality Policy
- Rendering Guidance (Receiving Side)
- Composing Guidance (Sending Side)
- Replying and Forwarding Guidance

RFC 9788 – ToC (2/2)

- Unprotected Header Fields Added in Transit
- Email Ecosystem Evolution
- Usability Considerations
- Security Considerations
- Privacy Considerations
- IANA Considerations
- References
- Appendix
- Acknowledgements
- Authors' Addresses

Summary of RFC 9787

- Guidance on End-to-End Email Security (RFC 9787)
- Background
  - Standards for e2e email protection extremely flexible
    - Can trap users and cause surprising failures
- Goals
  - Help implementers to mitigate such risks
  - Make e2e protected email simple & secure for end user
  - Provides a useful set of vocabulary
  - Recommendations to avoid common failures
- Also lists a number of currently unsolved usability and interoperability problems

Q & A / Discussion

Further Information

- RFC 9788 – Header Protection for Cryptographically Protected Email
- RFC 9787 – Guidance on End-to-End Email Security
- RFC 8551 – Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification
- pEp – pretty Easy privacy
- Autocrypt – Convenient End-to-End Encryption for E-Mail
- Ucom.ch – Technical Consulting for Internet Technology

Contact

- Email: Bernie Hoeneisen <bernie_NOSPAM_@ucom.ch>
- WWW: https://ucom.ch/
- Phone: +41 44 500 52 40

Ucom.ch - Tech Consulting
for Internet Technology and Standardization