
Stream: Internet Engineering Task Force (IETF)
RFC: [8951](#)
Updates: [7030](#)
Category: Standards Track
Published: November 2020
ISSN: 2070-1721
Authors: M. Richardson T. Werner W. Pan
Sandelman Software Works Siemens Huawei Technologies

RFC 8951

Clarification of Enrollment over Secure Transport (EST): Transfer Encodings and ASN.1

Abstract

This document updates RFC 7030: Enrollment over Secure Transport to resolve some errata that were reported and that have proven to cause interoperability issues when RFC 7030 was extended.

This document deprecates the specification of "Content-Transfer-Encoding" headers for Enrollment over Secure Transport (EST) endpoints. This document fixes some syntactical errors in ASN.1 that were present.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc8951>.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions

with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- 1. [Introduction](#)
 - 2. [Terminology](#)
 - 3. [Changes to EST Endpoint Processing](#)
 - 3.1. [White Space Processing](#)
 - 3.2. [Changes to Section 4 of RFC 7030](#)
 - 3.2.1. [Section 4.1.3](#)
 - 3.2.2. [Section 4.3.1](#)
 - 3.2.3. [Section 4.3.2](#)
 - 3.2.4. [Section 4.4.2](#)
 - 3.2.5. [Section 4.5.2](#)
 - 4. [Clarification of ASN.1 for Certificate Attribute Set](#)
 - 5. [Clarification of Error Messages for Certificate Enrollment Operations](#)
 - 5.1. [Updating Section 4.2.3: Simple Enroll and Re-enroll Response](#)
 - 5.2. [Updating Section 4.4.2: Server-Side Key Generation Response](#)
 - 6. [Privacy Considerations](#)
 - 7. [Security Considerations](#)
 - 8. [IANA Considerations](#)
 - 9. [References](#)
 - 9.1. [Normative References](#)
 - 9.2. [Informative References](#)
- [Appendix A. ASN.1 Module](#)
- [Acknowledgements](#)
- [Authors' Addresses](#)

1. Introduction

Enrollment over Secure Transport (EST) is defined in [RFC7030]. The EST specification defines a number of HTTP endpoints for certificate enrollment and management. The details of the transaction were defined in terms of MIME headers, as defined in [RFC2045], rather than in terms of the HTTP protocol, as defined in [RFC7230] and [RFC7231].

[RFC2616] and later Appendix A.5 of [RFC7231] have text specifically deprecating Content-Transfer-Encoding. However, [RFC7030] incorrectly uses this header.

Any updates to [RFC7030] to bring it in line with HTTP processing risk changing the on-wire protocol in a way that is not backwards compatible. However, reports from implementers suggest that many implementations do not send the Content-Transfer-Encoding, and many of them ignore it. The consequence is that simply deprecating the header would remain compatible with current implementations.

[BRSKI] extends [RFC7030], adding new functionality. Interop testing of the protocol has revealed that unusual processing called out in [RFC7030] causes confusion.

EST is currently specified as part of [IEC62351] and is widely used in government, utilities, and financial markets today.

This document, therefore, revises [RFC7030] to reflect the field reality, deprecating the extraneous field.

This document deals with errata numbers [errata4384], [errata5107], [errata5108], and [errata5904].

This document deals with [errata5107] and [errata5904] in Section 3. [errata5108] is dealt with in Section 5. [errata4384] is closed by correcting the ASN.1 Module in Section 4.

2. Terminology

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.

3. Changes to EST Endpoint Processing

Sections 4.1.3 (CA Certificates Response, /cacerts), 4.3.1 and 4.3.2 (Full CMC, /fullcmc), 4.4.2 (Server-Side Key Generation, /serverkeygen), and 4.5.2 (CSR Attributes, /csrattrs) of [RFC7030] specify the use of base64 encoding with a Content-Transfer-Encoding for requests and responses.

This document updates [RFC7030] to require the POST request and payload response of all endpoints using base64 encoding, as specified in Section 4 of [RFC4648]. In both cases, the Distinguished Encoding Rules (DER) [X.690] are used to produce the input for the base64 encoding routine. This format is to be used regardless of any Content-Transfer-Encoding header, and any value in such a header **MUST** be ignored.

3.1. White Space Processing

Note that "base64" as used in the HTTP [RFC2616] does not permit CRLF, while the "base64" used in MIME [RFC2045] does. This specification clarifies that despite what [RFC2616] says, white space including CR, LF, spaces (ASCII 32), and tabs (ASCII 9) **SHOULD** be tolerated by receivers. Senders are not required to insert any kind of white space.

3.2. Changes to Section 4 of RFC 7030

3.2.1. Section 4.1.3

Replace:

A successful response **MUST** be a certs-only CMC Simple PKI Response, as defined in [RFC5272], containing the certificates described in the following paragraph. The HTTP content-type of "application/pkcs7-mime" is used. The Simple PKI Response is sent with a Content-Transfer-Encoding of "base64" [RFC2045].

with:

A successful response **MUST** be a certs-only CMC Simple PKI Response, as defined in [RFC5272], containing the certificates described in the following paragraph. The HTTP content-type of "application/pkcs7-mime" is used. The CMC Simple PKI Response is encoded in base64 [RFC4648].

3.2.2. Section 4.3.1

Replace:

If the HTTP POST to /fullcmc is not a valid Full PKI Request, the server **MUST** reject the message. The HTTP content-type used is "application/pkcs7-mime" with an smime-type parameter "CMC-request", as specified in [RFC5273]. The body of the message is the binary value of the encoding of the PKI Request with a Content-Transfer-Encoding of "base64" [RFC2045].

with:

If the HTTP POST to /fullcmc is not a valid Full PKI Request, the server **MUST** reject the message. The HTTP content-type used is "application/pkcs7-mime" with an smime-type parameter "CMC-request", as specified in [RFC5273]. The body of the message is encoded in base64 [RFC4648].

3.2.3. Section 4.3.2

Replace:

The body of the message is the binary value of the encoding of the PKI Response with a Content-Transfer-Encoding of "base64" [RFC2045].

with:

The body of the message is the base64 [RFC4648] encoding of the PKI Response.

3.2.4. Section 4.4.2

Replace:

An "application/pkcs8" part consists of the base64-encoded DER-encoded [X.690] PrivateKeyInfo with a Content-Transfer-Encoding of "base64" [RFC2045].

with:

An "application/pkcs8" part consists of the base64-encoded, DER-encoded [X.690] PrivateKeyInfo.

Replace:

In all three additional encryption cases, the EnvelopedData is returned in the response as an "application/pkcs7-mime" part with an smime-type parameter of "server-generated-key" and a Content-Transfer-Encoding of "base64".

with:

In all three additional encryption cases, the EnvelopedData is returned in the response as an "application/pkcs7-mime" part with an smime-type parameter of "server-generated-key". It is base64 encoded [RFC4648].

3.2.5. Section 4.5.2

This section is updated in its entirety in [Section 4](#).

4. Clarification of ASN.1 for Certificate Attribute Set

[Section 4.5.2](#) of [RFC7030] is to be replaced with the following text:

4.5.2 CSR Attributes Response

If locally configured policy for an authenticated EST client indicates a CSR Attributes Response is to be provided, the server response **MUST** include an HTTP 200 response code. An HTTP response code of 204 or 404 indicates that a CSR Attributes Response is not available. Regardless of the response code, the EST server and CA **MAY** reject any subsequent enrollment requests for any reason, e.g., incomplete CSR attributes in the request.

Responses to attribute request messages **MUST** be encoded as the content-type of "application/csrattrs" and are to be "base64" [RFC4648] encoded. The syntax for application/csrattrs body is as follows:

```
CsrAttrs ::= SEQUENCE SIZE (0..MAX) OF AttrOrOID

AttrOrOID ::= CHOICE {
  oid          OBJECT IDENTIFIER,
  attribute    Attribute {{AttrSet}} }

AttrSet ATTRIBUTE ::= { ... }
```

An EST server includes zero or more OIDs or attributes [RFC2986] that it requests the client to use in the certification request. The client **MUST** ignore any OID or attribute it does not recognize. When the server encodes CSR attributes as an empty SEQUENCE, it means that the server has no specific additional information it desires in a client certification request (this is functionally equivalent to an HTTP response code of 204 or 404).

If the CA requires a particular cryptographic algorithm or use of a particular signature scheme (e.g., certification of a public key based on a certain elliptic curve or signing using a certain hash algorithm), it **MUST** provide that information in the CSR Attribute Response. If an EST server requires the linking of identity and POP information (see [Section 3.5](#)), it **MUST** include the challengePassword OID in the CSR Attributes Response.

The structure of the CSR Attributes Response **SHOULD**, to the greatest extent possible, reflect the structure of the CSR it is requesting. Requests to use a particular signature scheme (e.g., using a particular hash function) are represented as an OID to be reflected in the SignatureAlgorithm of the CSR. Requests to use a particular cryptographic algorithm (e.g., certification of a public key based on a certain elliptic curve) are represented as an attribute, to be reflected as the AlgorithmIdentifier of the SubjectPublicKeyInfo, with a type indicating the algorithm and the values indicating the particular parameters specific to the algorithm. Requests for descriptive information from the client are made by an attribute, to be represented as Attributes of the CSR, with a type indicating the [RFC2985] extensionRequest and the values indicating the particular attributes desired to be included in the resulting certificate's extensions.

The sequence is Distinguished Encoding Rules (DER) encoded [X.690] and then base64 encoded (Section 4 of [RFC4648]). The resulting text forms the application/csrattr body, without headers.

For example, if a CA requests that a client a) submit a certification request containing the challengePassword (indicating that linking of identity and POP information is requested; see Section 3.5), b) submit an extensionRequest with the Media Access Control (MAC) address [RFC2307] of the client, and c) use the secp384r1 elliptic curve to sign using the SHA384 hash function, then it takes the following:

```

OID:      challengePassword (1.2.840.113549.1.9.7)
Attribute: type = extensionRequest (1.2.840.113549.1.9.14)
           value = macAddress (1.3.6.1.1.1.1.22)
Attribute: type = id-ecPublicKey (1.2.840.10045.2.1)
           value = secp384r1 (1.3.132.0.34)
OID:      ecdsaWithSHA384 (1.2.840.10045.4.3.3)

```

and encodes them into an ASN.1 SEQUENCE to produce:

```

30 41 06 09 2a 86 48 86 f7 0d 01 09 07 30 12 06 07 2a 86 48 ce 3d
02 01 31 07 06 05 2b 81 04 00 22 30 16 06 09 2a 86 48 86 f7 0d 01
09 0e 31 09 06 07 2b 06 01 01 01 01 16 06 08 2a 86 48 ce 3d 04 03
03

```

and then base64 encodes the resulting ASN.1 SEQUENCE to produce:

```

MEEGCSqGSIB3DQEJBzASBgqhkJOPQIBMQcGBSuBBAAiMBYGCsqGSIB3DQEJDjEJ
BgcrBgEBAQEwBggqhkJOPQDAw==

```

TESTING -- For example, if a CA requests that a client a) submit a certification request containing the challengePassword (indicating that linking of identity and POP information is requested; see Section 3.5), b) submit an extensionRequest with the Media Access Control (MAC) address [RFC2307] of the client, and c) use the secp384r1 elliptic curve to sign using the SHA384 hash function, then it takes the following:

```

OID:      challengePassword (1.2.840.113549.1.9.7)

Attribute: type = extensionRequest (1.2.840.113549.1.9.14)
           value = macAddress (1.3.6.1.1.1.1.22)

Attribute: type = id-ecPublicKey (1.2.840.10045.2.1)
           value = secp384r1 (1.3.132.0.34)

OID:      ecdsaWithSHA384 (1.2.840.10045.4.3.3)

```

and encodes them into an ASN.1 SEQUENCE to produce:

```

30 41 06 09 2a 86 48 86 f7 0d 01 09 07 30 12 06 07 2a 86 48 ce 3d
02 01 31 07 06 05 2b 81 04 00 22 30 16 06 09 2a 86 48 86 f7 0d 01
09 0e 31 09 06 07 2b 06 01 01 01 01 16 06 08 2a 86 48 ce 3d 04 03
03

```

and then base64 encodes the resulting ASN.1 SEQUENCE to produce:

```

MEEGCSqGSIB3DQEJBzASBgqhkJOPQIBMQcGBSuBBAAiMBYGCSqGSIB3DQEJDjEJ
BgcrBgEBAQEWBggqhkJOPQDAw==

```

5. Clarification of Error Messages for Certificate Enrollment Operations

[[errata5108](#)] clarifies what format the error messages are to be in. Previously, a client might be confused into believing that an error returned with type text/plain was not intended to be an error.

5.1. Updating Section 4.2.3: Simple Enroll and Re-enroll Response

Replace:

If the content-type is not set, the response data **MUST** be a plaintext human-readable error message containing explanatory information describing why the request was rejected (for example, indicating that CSR attributes are incomplete).

with:

If the content-type is not set, the response data **MUST** be a plaintext human-readable error message containing explanatory information describing why the request was rejected (for example, indicating that CSR attributes are incomplete). Servers **MAY** use the "text/plain" content-type [RFC2046] for human-readable errors.

5.2. Updating Section 4.4.2: Server-Side Key Generation Response

Replace:

If the content-type is not set, the response data **MUST** be a plaintext human-readable error message.

with:

If the content-type is not set, the response data **MUST** be a plaintext human-readable error message. Servers **MAY** use the "text/plain" content-type [RFC2046] for human-readable errors.

6. Privacy Considerations

This document does not disclose any additional identities that either an active or passive observer would see with [RFC7030].

7. Security Considerations

This document clarifies an existing security mechanism. It does not create any new protocol mechanisms.

All security considerations from [RFC7030] also apply to the clarifications described in this document.

8. IANA Considerations

The ASN.1 module in [Appendix A](#) of this document makes use of object identifiers (OIDs).

IANA has registered an OID for id-mod-est-2019 (1.3.6.1.5.5.7.0.98) in the "SMI Security for PKIX Module Identifier" registry for the ASN.1 module.

The OID for the Asymmetric Decryption Key Identifier (1.2.840.113549.1.9.16.2.54) was previously defined in [RFC7030]. IANA has updated the Reference column for the Asymmetric Decryption Key Identifier attribute to also include a reference to this document.

9. References

9.1. Normative References

- [**errata4384**] RFC Errata, Erratum ID 4384, RFC 7030, <<https://www.rfc-editor.org/errata/eid4384>>.
- [**errata5107**] RFC Errata, Erratum ID 5107, RFC 7030, <<https://www.rfc-editor.org/errata/eid5107>>.
- [**errata5108**] RFC Errata, Erratum ID 5108, RFC 7030, <<https://www.rfc-editor.org/errata/eid5108>>.
- [**errata5904**] RFC Errata, Erratum ID 5904, RFC 7030, <<https://www.rfc-editor.org/errata/eid5904>>.
- [**IEC62351**] International Electrotechnical Commission, "Power systems management and associated information exchange - Data and communications security - Part 9: Cyber security key management for power system equipment", ISO/IEC 62351-9:2017, May 2017.
- [**RFC2045**] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, DOI 10.17487/RFC2045, November 1996, <<https://www.rfc-editor.org/info/rfc2045>>.
- [**RFC2046**] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, DOI 10.17487/RFC2046, November 1996, <<https://www.rfc-editor.org/info/rfc2046>>.
- [**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/info/rfc2119>>.
- [**RFC2986**] Nystrom, M. and B. Kaliski, "PKCS #10: Certification Request Syntax Specification Version 1.7", RFC 2986, DOI 10.17487/RFC2986, November 2000, <<https://www.rfc-editor.org/info/rfc2986>>.
- [**RFC4648**] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <<https://www.rfc-editor.org/info/rfc4648>>.
- [**RFC5272**] Schaad, J. and M. Myers, "Certificate Management over CMS (CMC)", RFC 5272, DOI 10.17487/RFC5272, June 2008, <<https://www.rfc-editor.org/info/rfc5272>>.
- [**RFC5273**] Schaad, J. and M. Myers, "Certificate Management over CMS (CMC): Transport Protocols", RFC 5273, DOI 10.17487/RFC5273, June 2008, <<https://www.rfc-editor.org/info/rfc5273>>.

- [RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the Public Key Infrastructure Using X.509 (PKIX)", RFC 5912, DOI 10.17487/RFC5912, June 2010, <<https://www.rfc-editor.org/info/rfc5912>>.
- [RFC6268] Schaad, J. and S. Turner, "Additional New ASN.1 Modules for the Cryptographic Message Syntax (CMS) and the Public Key Infrastructure Using X.509 (PKIX)", RFC 6268, DOI 10.17487/RFC6268, July 2011, <<https://www.rfc-editor.org/info/rfc6268>>.
- [RFC7030] Pritikin, M., Ed., Yee, P., Ed., and D. Harkins, Ed., "Enrollment over Secure Transport", RFC 7030, DOI 10.17487/RFC7030, October 2013, <<https://www.rfc-editor.org/info/rfc7030>>.
- [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/info/rfc8174>>.
- [X.680] ITU-T, "Information technology -- Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, ISO/IEC 8824-1:2015, August 2015, <<https://www.itu.int/rec/T-REC-X.680-201508-I/en>>.
- [X.681] ITU-T, "Information Technology - Abstract Syntax Notation One (ASN.1): Information object specification", ITU-T Recommendation X.681, ISO/IEC 8824-2:2015, August 2015, <<https://www.itu.int/rec/T-REC-X.681>>.
- [X.682] ITU-T, "Information Technology - Abstract Syntax Notation One (ASN.1): Constraint specification", ITU-T Recommendation X.682, ISO/IEC 8824-3:2015, August 2015, <<https://www.itu.int/rec/T-REC-X.682>>.
- [X.683] ITU-T, "Information Technology - Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications", ITU-T Recommendation X.683, ISO/IEC 8824-4:2015, August 2015, <<https://www.itu.int/rec/T-REC-X.683>>.
- [X.690] ITU-T, "Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, ISO/IEC 8825-1:2015, August 2015, <<https://www.itu.int/rec/T-REC-X.690>>.

9.2. Informative References

- [BRSKI] Pritikin, M., Richardson, M., Eckert, T., Behringer, M., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructures (BRSKI)", Work in Progress, Internet-Draft, draft-ietf-anima-bootstrapping-keyinfra-45, 11 November 2020, <<https://tools.ietf.org/html/draft-ietf-anima-bootstrapping-keyinfra-45>>.
- [RFC2307] Howard, L., "An Approach for Using LDAP as a Network Information Service", RFC 2307, DOI 10.17487/RFC2307, March 1998, <<https://www.rfc-editor.org/info/rfc2307>>.

- [RFC2616]** Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, DOI 10.17487/RFC2616, June 1999, <<https://www.rfc-editor.org/info/rfc2616>>.
- [RFC2985]** Nystrom, M. and B. Kaliski, "PKCS #9: Selected Object Classes and Attribute Types Version 2.0", RFC 2985, DOI 10.17487/RFC2985, November 2000, <<https://www.rfc-editor.org/info/rfc2985>>.
- [RFC7230]** Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <<https://www.rfc-editor.org/info/rfc7230>>.
- [RFC7231]** Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, DOI 10.17487/RFC7231, June 2014, <<https://www.rfc-editor.org/info/rfc7231>>.

Appendix A. ASN.1 Module

This annex provides the normative ASN.1 definitions for the structures described in this specification using ASN.1 as defined in [X.680], [X.681], [X.682], and [X.683].

The ASN.1 modules makes imports from the ASN.1 modules in [RFC5912] and [RFC6268].

There is no ASN.1 Module in [RFC7030]. This module has been created by combining the lines that are contained in the document body.

```
PKIXEST-2019
  { iso(1) identified-organization(3) dod(6)
    internet(1) security(5) mechanisms(5) pkix(7)
    id-mod(0) id-mod-est-2019(98) }

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

-- EXPORTS ALL --

IMPORTS

Attribute
FROM CryptographicMessageSyntax-2010 -- [RFC6268]
  { iso(1) member-body(2) us(840) rsadsi(113549)
    pkcs(1) pkcs-9(9) smime(16) modules(0)
    id-mod-cms-2009(58) }

ATTRIBUTE
FROM PKIX-CommonTypes-2009 -- [RFC5912]
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) id-mod(0)
    id-mod-pkixCommon-02(57) } ;

-- CSR Attributes

CsrAttrs ::= SEQUENCE SIZE (0..MAX) OF AttrOrOID

AttrOrOID ::= CHOICE {
  oid          OBJECT IDENTIFIER,
  attribute    Attribute {{AttrSet}} }

AttrSet ATTRIBUTE ::= { ... }

-- Asymmetric Decrypt Key Identifier Attribute

aa-asymmDecryptKeyID ATTRIBUTE ::=
  { TYPE AsymmetricDecryptKeyIdentifier
    IDENTIFIED BY id-aa-asymmDecryptKeyID }

id-aa-asymmDecryptKeyID OBJECT IDENTIFIER ::= { iso(1)
  member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
  smime(16) aa(2) 54 }

AsymmetricDecryptKeyIdentifier ::= OCTET STRING

END
```

Acknowledgements

Huawei Technologies supported the efforts of Wei Pan and Michael Richardson.

The ASN.1 Module was assembled by Russ Housley and formatted by Sean Turner. Russ Housley provided editorial review.

Authors' Addresses

Michael Richardson

Sandelman Software Works

Email: mcr+ietf@sandelman.ca

Thomas Werner

Siemens

Email: thomas-werner@siemens.com

Wei Pan

Huawei Technologies

Email: william.panwei@huawei.com