LAMPS Working Group
Internet-Draft
Obsoletes: 5273, 6402 (if approved)
Intended status: Standards Track
Expires: 23 April 2026

J. Mandel, Ed
AKAYLA, Inc.
S. Turner, Ed
sn3rd

20 October 2025

Certificate Management over CMS (CMC): Transport Protocols
draft-ietf-lamps-rfc5273bis-09

Abstract

This document defines a number of transport mechanisms that are used to move CMC (Certificate Management over CMS (Cryptographic Message Syntax)) messages. The transport mechanisms described in this document are HTTP, file, mail, and TCP.

This document obsoletes RFC 5273 and RFC 6402.

About This Document

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

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-lamps-rfc5273bis/.

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

Source for this draft and an issue tracker can be found at https://github.com/seanturner/cmcbis.

Status of This Memo

This Internet-Draft will expire on 23 April 2026.

Table of Contents

1. Introduction

This document defines a number of transport methods that are used to move CMC messages (defined in [CMC-STRUCT]). The transport mechanisms described in this document are HTTP, file, mail, and TCP.

This document obsoletes RFC 5273 [CMC-TRANSv1] and RFC 6402 [CMC-Updates]. This document also incorporates [erratum3593].

2. Requirements 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 Since 5273 and 6402

Merged [CMC-Updates] text.

IANA assigned TCP port 5318 for the use of CMC.

Clarified the file extensions for Full PKI Requests and Responses.

Replaced TLS 1.0 with TLS 1.2 or later, and added that implemetations are required to follow the recommendations in [BCP195].

Addressed [erratum3593].

4. File-Based Protocol

Enrollment messages and responses may be transferred between clients and servers using file-system-based mechanisms, such as when enrollment is performed for an off-line client. When files are used to transport Full PKI Request or Full PKI Response messages, there MUST be only one instance of a request or response message in a single file and the file MUST be binary encoded. The abbreviations crq and crp stand for Full PKI Request/Response, respectively; for clarity we define file extensions for them. The following file type extensions SHOULD be used:

Message Type	File Extension
Simple PKI Request	.p10
Full PKI Request	.crq
Simple PKI Response	.p7c
Full PKI Response	.crp

Table 1: File PKI Request/Response Identification

5. Mail-Based Protocol

MIME wrapping is defined for those environments that are MIME native. The basic mime wrapping in this section is taken from [SMIMEV4]. When using a mail-based protocol, MIME wrapping between the layers of CMS wrapping is optional. Note that this is different from the standard S/MIME (Secure MIME) message.

Simple enrollment requests are encoded using the "application/pkcs10" content type. A file name MUST be included either in a Content-Type or a Content-Disposition statement. The extension for the file MUST be ".p10".

Simple enrollment response messages MUST be encoded as content type "application/pkcs7-mime". A smime-type parameter MUST be on the Content-Type statement with a value of "certs-only". A file name with the ".p7c" extension MUST be specified as part of the Content-Type or Content-Disposition statement.

Full enrollment request messages MUST be encoded as content type "application/pkcs7-mime". The smime-type parameter MUST be included with a value of "CMC-Request". A file name with the ".p7m" extension MUST be specified as part of the Content-Type or Content-Disposition statement.

Full enrollment response messages MUST be encoded as content type "application/pkcs7-mime". The smime-type parameter MUST be included with a value of "CMC-Response". A file name with the ".p7m" extension MUST be specified as part of the Content-Type or Content-Disposition statement.

Item	MIME Type	File Extension	SMIME Type
Simple PKI Request	application/ pkcs10	.p10	N/A
Full PKI Request	application/ pkcs7-mime	.p7m	CMC-Request
Simple PKI Response	application/ pkcs7-mime	.p7c	certs-only
Full PKI Response	application/ pkcs7-mime	.p7m	CMC-Response

Table 2: MIME PKI Request/Response Identification

6. HTTP-Based Protocol

This section describes the conventions for use of HTTP [HTTP] as a data transfer protocol. Consult [HTTP-IMP] for additional information. The use of HTTPS [HTTP] provides any necessary content protection from eavesdroppers.

In order for CMC clients and servers using HTTP to interoperate, the following rules apply.

Clients are configured with sufficient information to form the server URI [RFC3986].

Client requests are submitted by use of the POST method.

Servers MUST use the 2XX response codes for successful responses.

Clients MAY attempt to send certification requests using HTTPS [HTTP], although servers are not required to support TLS/QUIC but a secure channel might be available regardless depending on the HTTP version implemented [HTTP_1.0], [HTTP_1.1], [HTTP_2], [HTTP_3], or later. If TLS is used by the HTTP version, then the implementation MUST follow the recommendations in [BCP195]. CMC implementations that support TLS 1.3 or QUIC MUST NOT use early data (i.e., 0-RTT) because POST is not idempotent.

Clients are not required to support any type of HTTP authentication (Section 11 of [HTTP]) nor Cookies [COOKIES]. Thus, servers can not rely on these features to be available.

Clients and servers are expected to follow other rules and restrictions in [HTTP]. Note that some of those rules are for HTTP methods other than POST; clearly, only the rules that apply to POST are relevant for this specification.

6.1. PKI Request

A PKI Request using the POST method is constructed as follows:

The Content-Type field MUST have the appropriate value from Table 2.

A Content-Type field for a request:

Content-Type: application/pkcs7-mime; smime-type=CMC-Request; name=request.p7m

The content of the message is the binary value of the encoding of the PKI Request.

6.2. PKI Response

The content of an HTTP-based PKI Response is the binary value of the BER (Basic Encoding Rules) encoding [X690] of either a Simple or Full PKI Response.

The Content-Type field MUST have the appropriate value from Table 2.

A Content-Type field for a response:

Content-Type: application/pkcs7-mime; smime-type=CMC-Response; name=response.p7m

7. TCP-Based Protocol

When CMC messages are sent over a TCP-based connection, no wrapping is required of the message. Messages are sent in their binary encoded form.

The client closes a connection after receiving a response, or it issues another request to the server using the same connection. Reusing one connection for multiple successive requests, instead of opening multiple connections that are only used for a single request, is RECOMMENDED for performance and resource conservation reasons. The client MUST wait for the full response before making another request on the same connection. A server MAY close a connection after it has been idle for some period of time; this timeout would typically be several minutes long.

CMC requires a registered port number to send and receive CMC messages over TCP. The Service Name is "pkix-cmc". The TCP port number is 5318.

Prior to [CMC-Updates], CMC did not have a registered port number and used an externally configured port from the Private Port range. Client implementations MAY continue to use a port chosen from the Private Port range. A TCP Server SHOULD use port 5318 assigned to the CMC service. It is expected that HTTP will continue to be the primary transport method used by CMC installations.

8. IANA Considerations

IANA has assigned a TCP port number in the Registered Port Number range for the use of CMC.

```
Service name: pkix-cmc
Port Number: 5318
Transport protocol: TCP
Description: PKIX Certificate Management using CMS (CMC)
Reference: [RFC-to-be]
Assignee: iesg@ietf.org
Contact: chair@ietf.org
```

IANA is requested to update the existing references to [CMC-TRANSv1] in the Media Type Sub-Parameter Registries for CMC-Request and CMC-Response to [RFC-to-be].

9. Security Considerations

Mechanisms for thwarting replay attacks may be required in particular implementations of this protocol depending on the operational environment. In cases where the Certification Authority (CA) maintains significant state information, replay attacks may be detectable without the inclusion of the (optional) CMC nonce mechanisms. [Implementers/Designers] of this protocol need to carefully consider environmental conditions before choosing whether or not to [implement/use] the senderNonce and recipientNonce attributes described in Section 6.6 of [CMC-STRUCT]. Developers of state-constrained PKI clients are strongly encouraged to incorporate the use of these attributes.

Initiation of a secure communications channel between an end-entity and a CA or Registration Authority (RA) -- and, similarly, between an RA and another RA or CA -- necessarily requires an out-of-band trust initiation mechanism. For example, a secure channel may be constructed between the end-entity and the CA via IPsec [IPsec] or TLS [TLS]. Many such schemes exist, and the choice of any particular

scheme for trust initiation is outside the scope of this document. Implementers of this protocol are strongly encouraged to consider generally accepted principles of secure key management when integrating this capability within an overall security architecture.

In some instances, no prior out-of-band trust will have been initiated prior to use of this protocol. This can occur when the protocol itself is being used to download onto the system the set of trust anchors to be used for these protocols. In these instances, the Enveloped Data content type (Section 3.2.1.3.3 of [CMC-STRUCT]) provides the same shrouding that TLS would have provided.

For the mail-based protocol, the Enveloped Data content type can also be used to apply confidentiality protection (content shrouding) to the conveyed messages. SMTP-over-TLS [RFC3207] does provide hop-by-hop security, but cannot guarantee that all hops are actually protected.

For the file-based protocol, an additional method of applying confidentiality protection (content shrouding) to the conveyed messages is usually availablle in the form of filesystem permissions. The local system may allow for read access to be limited to just a single user or group that corresponds to the entity authorized to read the request or response, respectively, and diligent use of these filesystem permissions can be a useful mechanism in multi-user environments.

10. References

10.1. Normative References

[BCP195] Sheffer, Y., Saint-Andre, P., and T. Fossati, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 9325, DOI 10.17487/RFC9325, November 2022, <https://www.rfc-editor.org/rfc/rfc9325>.

[CMC-STRUCT] Mandel, J. and S. Turner, "Certificate Management over CMS (CMC)", Work in Progress, Internet-Draft, draft-ietf-lamps-rfc5272bis-09, 20 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-rfc5272bis-09>.

[HTTP] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/rfc/rfc9110>.

[HTTP-IMP] Nottingham, M., "Building Protocols with HTTP", BCP 56, RFC 9205, DOI 10.17487/RFC9205, June 2022, <https://www.rfc-editor.org/rfc/rfc9205>.

[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>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/rfc/rfc3986>.

[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>.

[SMIMEV4] 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>.

[X690] ITU-T, "Information Technology -- Abstract Syntax Notation One (ASN.1): 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 8825-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.690>.

10.2. Informative References

[CMC-TRANSv1] 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/rfc/rfc5273>.

[CMC-Updates] Schaad, J., "Certificate Management over CMS (CMC) Updates", RFC 6402, DOI 10.17487/RFC6402, November 2011, <https://www.rfc-editor.org/rfc/rfc6402>.

[COOKIES] Barth, A., "HTTP State Management Mechanism", RFC 6265, DOI 10.17487/RFC6265, April 2011, <https://www.rfc-editor.org/rfc/rfc6265>.

[erratum3593]
"RFC 5273 erratum 3593", April 2013,
<https://www.rfc-editor.org/errata/eid3593>.

[HTTP_1.0] Berners-Lee, T., Fielding, R., and H. Frystyk, "Hypertext Transfer Protocol -- HTTP/1.0", RFC 1945, DOI 10.17487/RFC1945, May 1996, <https://www.rfc-editor.org/rfc/rfc1945>.

[HTTP_1.1] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP/1.1", STD 99, RFC 9112, DOI 10.17487/RFC9112, June 2022, <https://www.rfc-editor.org/rfc/rfc9112>.

[HTTP_2] Thomson, M., Ed. and C. Benfield, Ed., "HTTP/2", RFC 9113, DOI 10.17487/RFC9113, June 2022, <https://www.rfc-editor.org/rfc/rfc9113>.

[HTTP_3] Bishop, M., Ed., "HTTP/3", RFC 9114, DOI 10.17487/RFC9114, June 2022, <https://www.rfc-editor.org/rfc/rfc9114>.

[IPsec] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/rfc/rfc4301>.

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, DOI 10.17487/RFC3207, February 2002, <https://www.rfc-editor.org/rfc/rfc3207>.

[TLS] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/rfc/rfc5246>.

Acknowledgements

Obviously, the authors of this version of the document would like to thank Jim Schaad and Michael Myers for their work on the previous version of this document.

The acknowledgement from the previous version of this document follows:

The authors and the PKIX Working Group are grateful for the participation of Xiaoyi Liu and Jeff Weinstein in helping to author the original versions of this document.

The authors would like to thank Brian LaMacchia for his work in developing and writing up many of the concepts presented in this document. The authors would also like to thank Alex Deacon and Barb Fox for their contributions.

Contributors

Jim Schaad
August Cellars

Michael Myers
TraceRoute Security, Inc.

Authors' Addresses

Joseph Mandel
AKAYLA, Inc.
Email: joe@akayla.com

Sean Turner (editor)
sn3rd
Email: sean@sn3rd.com