Workgroup: LAMPS Working Group
Internet-Draft: draft-ietf-lamps-rfc6712bis-03
Obsoletes: 6712 (if approved)
Published: 10 February 2023
Intended Status: Standards Track
Expires: 14 August 2023
Authors: H. Brockhaus (Siemens), D. von Oheimb (Siemens), M. Ounsworth (Entrust), J. Gray (Entrust)

Internet X.509 Public Key Infrastructure -- HTTP Transfer for the Certificate Management Protocol (CMP)

Abstract

This document describes how to layer the Certificate Management Protocol (CMP) over HTTP.

It includes the updates on RFC 6712 specified in CMP Updates [RFCAAAA] Section 3 and obsoleted both documents. These updates introduce CMP URIs using a Well-known prefix.

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/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 14 August 2023.

Table of Contents

1. Introduction

[RFC Editor: please delete:

During IESG telechat the CMP Updates document was approved on condition that LAMPS provides a RFC6712bis document. Version -00 of this document shall be identical to RFC 6712 and version -01 incorporates the changes specified in CMP Updates Section 3.

A history of changes is available in Appendix A of this document.

The authors of this document wish to thank Tomi Kause and Martin Peylo, the original authors of RFC 6712, for their work and invite them, next to further volunteers, to join the -bis activity as co-authors.

]

[RFC Editor:

Please perform the following substitution.

  *RFCXXXX ---> the assigned numerical RFC value for this draft

  *RFCAAAA ---> the assigned numerical RFC value for
   [I-D.ietf-lamps-cmp-updates]

   Add this RFC number to the list of obsoleted RFCs.

  *RFCBBBB ---> the assigned numerical RFC value for
   [I-D.ietf-lamps-lightweight-cmp-profile]

  *RFCCCCC ---> the assigned numerical RFC value for
   [I-D.ietf-lamps-rfc4210bis]

]

The Certificate Management Protocol (CMP) [RFCCCCC] requires a well-defined transfer mechanism to enable End Entities (EEs), Registration Authorities (RAs), and Certification Authorities (CAs) to pass PKIMessage sequences between them.

The first version of the CMP specification [RFC2510] included a brief description of a simple transfer protocol layer on top of TCP. Its features were simple transfer-level error handling and a mechanism to poll for outstanding PKI messages. Additionally, it was mentioned that PKI messages could also be conveyed using file-, E-mail-, and HTTP-based transfer, but those were not specified in detail.

The second version of the CMP specification [RFC4210] incorporated its own polling mechanism and thus the need for a transfer protocol providing this functionality vanished. The remaining features CMP requires from its transfer protocols are connection and error handling.

In addition to reliable transport, CMP requires connection and error handling from the transfer protocol, which is all covered by HTTP. Additionally, delayed delivery of CMP response messages may be handled at transfer level regardless of the message contents. Since [RFCAAAA] extends the polling mechanism specified in the second version of CMP [RFC4210] to cover all types of PKI management transactions, delays detected at application level may also be handled within CMP, using pollReq and pollRep messages.

The usage of HTTP for transferring CMP messages exclusively uses the POST method for requests, effectively tunneling CMP over HTTP. While this is generally considered bad practice and should not be emulated, there are good reasons to do so for transferring CMP. HTTP is used as it is generally easy to implement and it is able to

traverse network borders utilizing ubiquitous proxies. Most importantly, HTTP is already commonly used in existing CMP implementations. Other HTTP request methods, such as GET, are not used because PKI management operations can only be triggered using CMP's PKI messages, which need to be transferred using a POST request.

With its status codes, HTTP provides needed error reporting capabilities. General problems on the server side, as well as those directly caused by the respective request, can be reported to the client.

As CMP implements a transaction ID, identifying transactions spanning over more than just a single request/response pair, the statelessness of HTTP is not blocking its usage as the transfer protocol for CMP messages.

1.1. Changes Since RFC 6712

CMP Updates [RFCAAAA] updated RFC 6712 [RFC6712], supporting the PKI management operations specified in the Lightweight CMP Profile [RFCBBBB], in the following areas:

- Introduce the HTTP URI path prefix '/.well-known/cmp'.

- Add options for extending the URI structure with further segments and to this end define a new protocol registry group.

1.2. Changes Made by This Document

This document obsoletes RFC 6712 [RFC6712]. It includes the changes specified by CMP Updates [RFCAAAA] Section 3 as described in Section 1.1.

2. Conventions Used in This Document

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. HTTP-Based Protocol

For direct interaction between two entities, where a reliable transport protocol like TCP is available, HTTP **SHOULD** be utilized for conveying CMP messages.

3.1. HTTP Versions

Implementations **MUST** support HTTP/1.0 [RFC1945] and **SHOULD** support HTTP/1.1 [RFC9112].

3.2. Persistent Connections

HTTP persistent connections [RFC9112] allow multiple interactions to take place on the same HTTP connection. However, neither HTTP nor the protocol specified in this document are designed to correlate messages on the same connection in any meaningful way; persistent connections are only a performance optimization. In particular, intermediaries can do things like mix connections from different clients into one "upstream" connection, terminate persistent connections, and forward requests as non-persistent requests, etc. As such, implementations **MUST NOT** infer that requests on the same connection come from the same client (e.g., for correlating PKI messages with ongoing transactions); every message is to be evaluated in isolation.

3.3. General Form

A DER-encoded [ITU.X690.1994] PKIMessage [RFCCCCC] is sent as the entity-body of an HTTP POST request. If this HTTP request is successful, the server returns the CMP response in the body of the HTTP response. The HTTP response status code in this case **MUST** be 200; other "Successful 2xx" codes **MUST NOT** be used for this purpose. HTTP responses to pushed CMP Announcement messages (i.e., CA Certificate Announcement, Certificate Announcement, Revocation Announcement, and Certificate Revocation List (CRL) Announcement) utilize the status codes 201 and 202 to identify whether the received information was processed.

While "Redirection 3xx" status codes **MAY** be supported by implementations, clients should only be enabled to automatically follow them after careful consideration of possible security implications. As described in Section 5, "301 Moved Permanently" could be misused for permanent denial of service.

All applicable "Client Error 4xx" or "Server Error 5xx" status codes **MAY** be used to inform the client about errors.

3.4. Header Fields

The Internet Media Type "application/pkixcmp" **MUST** be set in the HTTP Content-Type header field when conveying a PKIMessage.

The Content-Length header field **SHOULD** be provided, giving the length of the ASN.1-encoded PKIMessages.

3.5. Communication Workflow

In CMP, most communication is initiated by the EEs where every CMP request triggers a CMP response message from the CA or RA.

The CMP Announcement messages described in Section 3.7 are an exception. Their creation may be triggered by certain events or done on a regular basis by a CA. The recipient of the Announcement only replies with an HTTP status code acknowledging the receipt or indicating an error, but not with a CMP response.

If the receipt of an HTTP request is not confirmed by receiving an HTTP response, it **MUST** be assumed that the transferred CMP message was not successfully delivered to its destination.

3.6. HTTP Request-URI

Each CMP server on a PKI management entity supporting HTTP or HTTPS transfer **MUST** support the use of the path prefix '/.well-known/' as defined in RFC 8615 [RFC8615] and the registered name 'cmp' to ease interworking in a multi-vendor environment.

The CMP client needs to be configured with sufficient information to form the CMP server URI. This is at least the authority portion of the URI, e.g., 'www.example.com:80', or the full operation path segment of the PKI management entity. Additionally, **OPTIONAL** path segments **MAY** be added after the registered application name as part of the full operation path to provide further distinction. The path segment 'p' followed by an arbitraryLabel <name> could for example support the differentiation of specific CAs or certificate profiles. Further path segments, e.g., as specified in the Lightweight CMP Profile [RFCBBBB], could indicate PKI management operations using an operationLabel <operation>. A valid full CMP URI can look like this:

http://www.example.com/.well-known/cmp

http://www.example.com/.well-known/cmp/<operation>

http://www.example.com/.well-known/cmp/p/<name>

http://www.example.com/.well-known/cmp/p/<name>/<operation>

3.7. Pushing of Announcements

A CMP server may create event-triggered announcements or generate them on a regular basis. It **MAY** utilize HTTP transfer to convey them to a suitable recipient. In this use case, the CMP server acts as an HTTP client, and the recipient needs to utilize an HTTP server. As no request messages are specified for those announcements, they can only be pushed to the recipient.

If an EE wants to poll for a potential CA Key Update Announcement or the current CRL, a PKI Information Request using a General Message as described in Appendix E.5 of [RFCCCCC] can be used.

When pushing Announcement messages, PKIMessage structures are sent as the entity-body of an HTTP POST request.

Suitable recipients for CMP announcements might, for example, be repositories storing the announced information, such as directory services. Those services listen for incoming messages, utilizing the same HTTP Request-URI scheme as defined in Section 3.6.

The following PKIMessages are announcements that may be pushed by a CA. The prefixed numbers reflect ASN.1 numbering of the respective element.

[15] CA Key Update Announcement
[16] Certificate Announcement
[17] Revocation Announcement
[18] CRL Announcement

CMP Announcement messages do not require any CMP response. However, the recipient **MUST** acknowledge receipt with an HTTP response having an appropriate status code and an empty body. When not receiving such a response, it **MUST** be assumed that the delivery was not successful. If applicable, the sending side **MAY** try sending the Announcement again after waiting for an appropriate time span.

If the announced issue was successfully stored in a database or was already present, the answer **MUST** be an HTTP response with a "201 Created" status code and an empty message body.

In case the announced information was only accepted for further processing, the status code of the returned HTTP response **MAY** also be "202 Accepted". After an appropriate delay, the sender may then try to send the Announcement again and may repeat this until it receives a confirmation that it has been successfully processed. The appropriate duration of the delay and the option to increase it between consecutive attempts should be carefully considered.

A receiver **MUST** answer with a suitable 4xx or 5xx HTTP error code when a problem occurs.

3.8. HTTP Considerations

While all defined features of the HTTP protocol are available to implementations, they **SHOULD** keep the protocol utilization as simple as possible. For example, there is no benefit in using chunked Transfer-Encoding, as the length of an ASN.1 sequence is known when starting to send it.

There is no need for the clients to send an "Expect" request-header field with the "100-continue" expectation and wait for a "100 Continue" status as described in Section 8.2.3 of [RFC9112]. The CMP payload sent by a client is relatively small, so having extra messages exchanged is inefficient, as the server will only seldom reject a message without evaluating the body.

4. Implementation Considerations

Implementors should be aware that implementations might exist that use a different approach for transferring CMP over HTTP, because RFC 6712 [RFC6712] has been under development for more than a decade. Further, implementations based on earlier drafts of RFC 6712 [RFC6712] might use an unregistered "application/pkixcmp-poll" MIME type.

5. Security Considerations

The following aspects need to be considered by implementers and users:

1. There is the risk for denial-of-service attacks through resource consumption by opening many connections to an HTTP server. Therefore, idle connections should be terminated after an appropriate timeout; this may also depend on the available free resources. After sending a CMP Error Message, the server should close the connection, even if the CMP transaction is not yet fully completed.

2. Without being encapsulated in effective security protocols, such as Transport Layer Security (TLS) [RFC5246] or [RFC8446], there is no integrity protection at the HTTP protocol level. Therefore, information from the HTTP protocol should not be used to change state of the transaction.

3. Client users should be aware that storing the target location of an HTTP response with the "301 Moved Permanently" status code could be exploited by a man-in-the-middle attacker trying to block them permanently from contacting the correct server.

4. If no measures to authenticate and protect the HTTP responses to pushed Announcement messages are in place, their information regarding the Announcement's processing state may not be trusted. In that case, the overall design of the PKI system must not depend on the Announcements being reliably received and processed by their destination.

5. CMP provides inbuilt integrity protection and authentication. The information communicated unencrypted in CMP messages does not contain sensitive information endangering the security of

the PKI when intercepted. However, it might be possible for an eavesdropper to utilize the available information to gather confidential technical or business critical information. Therefore, users of the HTTP transfer for CMP might want to consider using HTTP over TLS according to [RFC9110] or virtual private networks created, for example, by utilizing Internet Protocol Security according to [RFC4301]. Compliant implementations **MUST** support TLS with the option to authenticate both server and client.

6. IANA Considerations

The IANA has already registered what is specified in CMP Updates [RFCAAAA].

No further action by the IANA is necessary for this document or any anticipated updates.

7. Acknowledgments

The authors of this document wish to thank Tomi Kause and Martin Peylo, the original authors of [RFC6712], for their work.

We also thank all reviewers of this document for their valuable feedback.

8. References

8.1. Normative References

[RFC1945] 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>.

[RFC8615] Nottingham, M., "Well-Known Uniform Resource Identifiers (URIs)", RFC 8615, DOI 10.17487/RFC8615, May 2019, <https://www.rfc-editor.org/rfc/rfc8615>.

[RFC9112] 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>.

[I-D.ietf-lamps-rfc4210bis] Brockhaus, H., von Oheimb, D., Ounsworth, M., and J. Gray, "Internet X.509 Public Key Infrastructure -- Certificate Management Protocol (CMP)", Work in Progress, Internet-Draft, draft-ietf-lamps-rfc4210bis-03, 24 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-rfc4210bis-03>.

[ITU.X690.1994]
International Telecommunications Union, "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, 1994.

[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>.

[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>.

8.2. Informative References

[I-D.ietf-lamps-cmp-updates] Brockhaus, H., von Oheimb, D., and J. Gray, "Certificate Management Protocol (CMP) Updates", Work in Progress, Internet-Draft, draft-ietf-lamps-cmp-updates-23, 29 June 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-cmp-updates-23>.

[I-D.ietf-lamps-lightweight-cmp-profile] Brockhaus, H., von Oheimb, D., and S. Fries, "Lightweight Certificate Management Protocol (CMP) Profile", Work in Progress, Internet-Draft, draft-ietf-lamps-lightweight-cmp-profile-20, 12 January 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-lightweight-cmp-profile-20>.

[RFC2510] Adams, C. and S. Farrell, "Internet X.509 Public Key Infrastructure Certificate Management Protocols", RFC 2510, DOI 10.17487/RFC2510, March 1999, <https://www.rfc-editor.org/rfc/rfc2510>.

[RFC4210] Adams, C., Farrell, S., Kause, T., and T. Mononen, "Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)", RFC 4210, DOI 10.17487/RFC4210, September 2005, <https://www.rfc-editor.org/rfc/rfc4210>.

[RFC4301] 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>.

[RFC5246] 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>.

[RFC6712] Kause, T. and M. Peylo, "Internet X.509 Public Key Infrastructure -- HTTP Transfer for the Certificate Management Protocol (CMP)", RFC 6712, DOI 10.17487/RFC6712, September 2012, <https://www.rfc-editor.org/rfc/rfc6712>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/rfc/rfc8446>.

[RFC9110] 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>.

Appendix A. History of Changes

Note: This appendix will be deleted in the final version of the document.

From version 02 -> 03:

* Fixing one formatting nit.

From version 01 -> 02:

* Updated Section 3.4 including the requirement to add the content-length filed into the HTTP header.

* Added a reference to TLS 1.3.

* Addressed idnits feedback, specifically changing the following RFC references: RFC2616 -> RFC9112; RFC2818 -> RFC9110, and RFC5246 -> RFC8446

From version 00 -> 01:

* Performed all updates specified in CMP Updates Section 3.

Version 00:

This version consists of the text of RFC6712 with the following changes:

* Introduced the authors of this document and thanked the authors of RFC6712 for their work.

* Added a paragraph to the introduction explaining the background of this document.

*Added the change history to this appendix.

Authors' Addresses

Hendrik Brockhaus
Siemens
Werner-von-Siemens-Strasse 1
80333 Munich
Germany

Email: hendrik.brockhaus@siemens.com
URI: https://www.siemens.com

David von Oheimb
Siemens
Werner-von-Siemens-Strasse 1
80333 Munich
Germany

Email: david.von.oheimb@siemens.com
URI: https://www.siemens.com

Mike Ounsworth
Entrust
1187 Park Place
Minneapolis, MN 55379
United States of America

Email: mike.ounsworth@entrust.com
URI: https://www.entrust.com

John Gray
Entrust
1187 Park Place
Minneapolis, MN 55379
United States of America

Email: john.gray@entrust.com
URI: https://www.entrust.com