Abstract

This document contains a set of updates to the base syntax of Certificate Management Protocol (CMP) version 2. This document updates RFC 4210.

Specifically, the CMP services updated in this document comprise:
Enable protection of server-side key generation using elliptic curve algorithms and the definition of an extended key usage to identify certificates of CMP endpoints on registration authorities and certification authorities.

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 January 8, 2020.

Copyright Notice

Copyright (c) 2019 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.
Internet-Draft                 CMP Updates                     July 2019

While using CMP [RFC4210] in industrial and IoT environments and developing the Lightweight CMP Profile [I-D.brockhaus-lamps-industrial-cmp-profile] some limitations were identified in the original CMP specification. This document updates RFC 4210 [RFC4210] to overcome these limitations.

In general this document aims to improve the crypto agility of CMP to be flexible to react on future advances in cryptography.

This document also introduces a new extended key usage to identify CMP services on registration and certification authorities.

< TBD: While implementing CMP we identified some wording that could be more precise. It can be discussed if such wording issued need to be addressed in the context of this document or not. >

1. Introduction

While using CMP [RFC4210] in industrial and IoT environments and developing the Lightweight CMP Profile [I-D.brockhaus-lamps-industrial-cmp-profile] some limitations were identified in the original CMP specification. This document updates RFC 4210 [RFC4210] to overcome these limitations.

In general this document aims to improve the crypto agility of CMP to be flexible to react on future advances in cryptography.

This document also introduces a new extended key usage to identify CMP services on registration and certification authorities.

< TBD: While implementing CMP we identified some wording that could be more precise. It can be discussed if such wording issued need to be addressed in the context of this document or not. >
1.1. Convention and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying significance described in RFC 2119.

Technical terminology is used in conformance with RFC 4210 [RFC4210], RFC 4211 [RFC4211], and RFC 5280 [RFC5280]. The following key words are used:

CA: Certification authority, which issues certificates.

RA: Registration authority, an optional system component to which a CA delegates certificate management functions such as authorization checks.

EE: End entity, a user or device or service that holds a PKI certificate. An identifier for the EE is given as the subject of the certificate.


2.1. New Section 1.1. - Changes since RFC 4210

The following subsections describe feature updates to RFC 4210 [RFC4210]. They are always related to the base specification. Hence references to the original sections in RFC 4210 are used whenever possible.

Insert this section at the end of the current Section 1.

The following updates were made since RFC 4210:

- Offering envelopedData as another choice next to EncryptedValue to extend crypto agility in CMP. Note that according to RFC 4211 [RFC4211] section 2.1.9 the use of the EncryptedValue structure has been deprecated in favor of the EnvelopedData structure. For reasons of completeness and consistency the exchange of EncryptedValue with EncryptedKey is performed not only where required for the needed crypto agility for protection of centrally generated private key, but also for other purposes like encryption of revocation passphrases.
Add new extended key usages for different CMP server types, e.g. Registration authority and certification authority.

2.2. Replace Section 5.2.2. - Encrypted Values

Section 5.2.2 of RFC 4210 [RFC4210] describes the usage of EncryptedValue to transport encrypted data.

Replace the text of the section with the following text.

Where encrypted data (restricted, in this specification, to be either private keys, certificates or passwords) are sent in PKI messages, the EncryptedKey data structure is used.

EncryptedKey ::= CHOICE {
  encryptedValue        EncryptedValue, -- deprecated
  envelopedData     [0] EnvelopedData }

See CRMF [RFC4211] for EncryptedKey and EncryptedValue syntax.

Using this data structure, it offers the choice to either use EncryptedValue or EnvelopedData. As EncryptedValue offers only key transport, e.g. using RSA or symmetric encryption, EnvelopedData offers further key management techniques, e.g. key agreement, and more crypto agility. Note that according to RFC 4211 [RFC4211] section 2.1.9 the use of the EncryptedValue structure has been deprecated in favor of the EnvelopedData structure. Therefore, it is recommended to use EnvelopedData.

See CMS [RFC5652] for EnvelopedData syntax.

Using envelopedData within CMP contains only one recepientInfo structure because the content is encrypted only for one recipient.

< TBD: Explain the detailed usage of the different key management techniques (e.g. key transport, key agreement, and previously distributed symmetric key-encryption keys) for encryption of the respective private key, certificate or revocation passphrase as specified for EnvelopedData in section 6 in CMS [RFC5652]. >

Use of either EnvelopedData or EncryptedValue (for backward compatibility only) requires that the creator and intended recipient be able to encrypt and decrypt, respectively. Typically, this will mean that the sender and recipient have, or are able to generate, a shared secret key.

< TBD: Description of EnvelopedData structure parts which are used and filled to support application in CMP >
If EncryptedValue is used by the sender and the recipient of the PKIMessage already possesses a private key usable for decryption, then the encSymmKey field MAY contain a session key encrypted using the recipient’s public key.

2.3. Update Section 5.3.4. - Certification Response

Section 5.3.4 of RFC 4210 [RFC4210] describes the Certification Response. This document updates the syntax by using EncryptedKey instead of EncryptedValue as described in Section 2.1 above.

Replace the ASN.1 syntax of CertifiedKeyPair and CertorEncCert with the following text.

CertifiedKeyPair ::= SEQUENCE {
    certOrEncCert CertOrEncCert, 
    privateKey [0] EncryptedKey OPTIONAL, 
    -- see [CRMF] for comment on encoding
    publicationInfo [1] PKIPublicationInfo OPTIONAL
}

CertOrEncCert ::= CHOICE {
    certificate [0] Certificate, 
    encryptedCert [1] EncryptedKey
}

< Question to Jim Schaad: Simply exchanging EncryptedValue with EncryptedKey is what I understood from our discussion back in April. Is it completely backwards compatible to current ASN.1 Syntax of RFC4210? >

Add the following paragraphs to the end of the section.

In case EnvelopedData is the choice used for EncryptedKey, the encrypted private key or certificate MUST be placed in the envelopedData encryptedContentInfo encryptedContent OCTET STRING. Note that according to RFC 4211 [RFC4211] section 2.1.9 the use of the EncryptedValue structure has been deprecated in favor of the EnvelopedData structure.

2.4. Update Section 5.3.19.9. - Revocation Passphrase

Section 5.3.19.9 of RFC 4210 [RFC4210] describes the provisioning of a revocation passphrase for authenticating a later revocation request. This document updates the handling by using EncryptedKey instead of EncryptedValue to transport this information as described in Section 2.1 above.
The revocation passphrase MAY be used by the EE to send a passphrase to a CA/RA for the purpose of authenticating a later revocation request (in the case that the appropriate signing private key is no longer available to authenticate the request). See Appendix B for further details on the use of this mechanism.

GenMsg:    {id-it 12}, EncryptedKey
GenRep:    {id-it 12}, < absent >

In case EnvelopedData is the choice used for EncryptedKey, the encrypted revocation passphrase MUST be placed in the envelopedData encryptedContentInfo encryptedContent OCTET STRING. Note that according to RFC 4211 [RFC4211] section 2.1.9 the use of the EncryptedValue structure has been deprecated in favor of the EnvelopedData structure.

2.5. Update Appendix B - The Use of Revocation Passphrase

Appendix B of RFC 4210 [RFC4210] describes the usage of the revocation passphrases. As this document updates RFC 4210 [RFC4210] to utilize EncryptedKey in favor of EncryptedValue as described in Section 2.1 above, the description is updated accordingly.

Replace the first bullet point of this section with the following text.

- The OID and value specified in Section 5.3.19.9 of RFC 4210 [RFC4210] MAY be sent in a GenMsg message at any time, or MAY be sent in the generalInfo field of the PKIHeader of any PKIMessage at any time. (In particular, the EncryptedKey may be sent in the header of the certConf message that confirms acceptance of certificates requested in an initialization request or certificate request message.) This conveys a revocation passphrase chosen by the entity (i.e., and for use of EnvelopedData this is in the decrypted bytes of encryptedContent of the EnvelopedData structure and for use of EncryptedValue this is in the decrypted bytes of the encValue field) to the relevant CA/RA; furthermore, the transfer is accomplished with appropriate confidentiality characteristics. Note that according to RFC 4211 [RFC4211] section 2.1.9 the use of the EncryptedValue structure has been deprecated in favor of the EnvelopedData structure.

Replace the third bullet point of this section with the following text.
When using EnvelopedData the contentType of EncryptedContentInfo and when using EncryptedValue the valueHint field MAY contain a key identifier (chosen by the entity, along with the passphrase itself) to assist in later retrieval of the correct passphrase (e.g., when the revocation request is constructed by the entity and received by the CA/RA).

2.6. Update Appendix C - Request Message Behavioral Clarifications

Appendix C of [RFC4210] provides clarifications to the request message behavior. As this document updates [RFC4210] to utilize EncryptedKey in favor of EncryptedValue as described in Section 2.1 above, the description is updated accordingly.

Replace the note coming after the ASN.1 syntax of POPOPrivKey of this section with the following text.

--- **********
--- * the type of "thisMessage" is given as BIT STRING in RFC 4211
--- * [RFC4211]; it should be "EncryptedKey" (in accordance with
--- * Section 5.2.2, "Encrypted Values", of this specification).
--- * Therefore, this document makes the behavioral clarification of
--- * specifying that the contents of "thisMessage" MUST be encoded
--- * either as EnvelopedData or EncryptedValue (only for backward
--- * compatibility) and then wrapped in a BIT STRING. This allows
--- * the necessary conveyance and protection of the private key
--- * while maintaining bits-on-the-wire compatibility with RFC 4211
--- * [RFC4211].
--- **********

2.7. Update Appendix D.4. - Request Message Behavioral Clarifications

< TBD, add further details on crc[1].certifiedKeyPair, e.g. refer to usage of EncryptedValue protected with the pre-shared symmetric MACing key using SYM_PENC_ALG. >

2.8. Insert section for EKUs like kp-id-cmpRA definition at an appropriate location.

< Details need to be defined later >

3. IANA Considerations

<Add any IANA considerations>
4. Security Considerations

No changes are made to the existing security considerations of RFC 4210 [RFC4210].

5. Acknowledgements

We would like to thank the various reviewers of this document.

6. References

6.1. Normative References


6.2. Informative References

[I-D.brockhaus-lamps-industrial-cmp-profile]

Appendix A. ASN.1 Modules

Changes to the following parts are needed

- Import from PKIXCRMF-2005

CertTemplate, PKIPublicationInfo, EncryptedKey, CertId, CertReqMessages
FROM PKIXCRMF-2005 {iso(1) identified-organization(3)
dod(6) internet(1) security(5) mechanisms(5) pkix(7)
   id-mod(0) id-mod-crmf2005(36)}

- In CertifiedKeyPair, CertOrEncCert and id-it-revPassphrase

CertifiedKeyPair ::= SEQUENCE {
certOrEncCert [0] CertOrEncCert,
privateKey [0] EncryptedKey OPTIONAL,
-- see [CRMF] for comment on encoding
   publicationInfo [1] PKIPublicationInfo OPTIONAL
}

CertOrEncCert ::= CHOICE {
certificate [0] CMPCertificate,
   encryptedCert [1] EncryptedKey
}

-- id-it-revPassphrase OBJECT IDENTIFIER ::= {id-it 12}
-- RevPassphraseValue ::= EncryptedKey

< TBD: If needed the complete ASN.1 Module from RFC 4210 section needs to be copied here to be changed. >

Author’s Address