Workgroup: Network Working Group
Internet-Draft: draft-gutmann-pkcs15-00
Published: 6 December 2023
Intended Status: Informational
Expires: 8 June 2024
Authors: P. Gutmann
University of Auckland

PKCS #15 Updates

Abstract

This document describes updates to the PKCS #15 standard made since the original publication of the standard.

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 8 June 2024.

Copyright Notice

Table of Contents

1. Introduction

After the publication of the original PKCS #15 standard it saw minor updates that were only published as drafts and circulated informally among implementers, but never finalised due to the dissolution of the organisation that published the standards. Since the standard, including the updates, remains in active use today, this document gathers the updates in a single location for reference by implementers.

The updates cover the extension of the original PKCS #15 identifiers to handle validFrom/validTo dates for certificates and PGP/OpenPGP keys and email addresses, the addition of support for CMS Authenticated-Enveloped-Data, and support for cryptographic binding of public-key components to private-key ones.

Since names and definitions have changed across the different drafts (for example v1.0 used PKCS15XXXAttributes while v1.1 and later used XXXXAttributes), this document uses the v1.1 ASN.1 module pkcs-15v1_1.asn as its baseline.

2. ValidFrom/ValidTo Dates

PKCS #15 v1.0 didn't support the validFrom/validTo dates that are required for certificates, these were added in the PKCS #15 v1.2 draft and extend the CommonCertificateAttributes to add:

```
     validFrom               GeneralisedTime OPTIONAL,
     validTo         [4] GeneralisedTime OPTIONAL,
```

For context, the full CommonCertificateAttributes are then:

```
CommonCertificateAttributes ::= SEQUENCE {
        iD                      Identifier,
        authority       BOOLEAN DEFAULT FALSE,
        identifier      CredentialIdentifier {{KeyIdentifiers}}
        certHash        [0] OOBCertHash OPTIONAL,
        ...,
        trustedUsage [1] Usage OPTIONAL,
        identifiers     [2] SEQUENCE OF CredentialIdentifier{{Ke
        implicitTrust [3] BOOLEAN DEFAULT FALSE
        validFrom       GeneralisedTime OPTIONAL,
        validTo         [4] GeneralisedTime OPTIONAL,
        }
```

3. Key Identifiers

PKCS #15 v1.0, designed for use with smart cards, didn't support PGP/OpenPGP or email use, making it difficult to implement PKCS #11 with PKCS #15 as the storage format. The PKCS #15 v1.2 draft extended CredentialIdentifier to include these additional IDs, which extend the existing KeyIdentifiers values to add pgp, openPGP, and uri identifiers:

```
KeyIdentifiers KEY-IDENTIFIER ::= {
        issuerAndSerialNumber |
        issuerAndSerialNumberHash |
        subjectKeyId |
        subjectKeyHash |
        issuerKeyHash |
        issuerNameHash |
        subjectNameHash |
        pgp |
        openPGP |
        uri
        ...
}

pgp KEY-IDENTIFIER ::=
        {SYNTAX OCTET STRING SIZE(8) IDENTIFIED BY 8}
        -- RFC 4880 V3 (PGP 2.x) key ID

openPGP KEY-IDENTIFIER ::=
        {SYNTAX OCTET STRING SIZE(8) IDENTIFIED BY 9}
        -- RFC 4880 V4 key ID

uri KEY-IDENTIFIER ::=
        {SYNTAX UTF8String IDENTIFIED BY 10}
        -- Typically email address but may be a more general URI
```

4. Authenticated-Enveloped-Data

PKCS #15 v1.0 predates the existence of CMS Authenticated-Enveloped-Data, which was added in the PKCS #15 v1.2 draft by extending the ObjectValue/ PathOrObjects CHOICE to include a new content type AuthEnvelopedData alongside the existing EnvelopedData. For ObjectValue this is:

```
direct-protected-auth [4] AuthEnvelopedData {Type},
```

For PathOrObjects this is:

```
direct-protected-auth [4] AuthEnvelopedData {SEQUENCE OF ObjectT
```

Note that the tags jump from the v1.1 'direct-protected [2] EnvelopedData' to 'direct-protected-auth [4] AuthEnvelopedData', the [3] tag was used for another object type whose purpose is now lost. For context, the full ObjectValue / PathOrObjects are then:

```
ObjectValue { Type } ::= CHOICE {
        indirect                        ReferencedValue {Type},
        direct                          [0] Type,
        indirect-protected      [1] ReferencedValue {EnvelopedDa
        direct-protected        [2] EnvelopedData {Type},
        direct-protected-auth [4] AuthEnvelopedData {Type}
        }

PathOrObjects {ObjectType} ::= CHOICE {
        path                            Path,
        objects                         [0] SEQUENCE OF ObjectTy
        ...,
        indirect-protected      [1] ReferencedValue {EnvelopedDa
        direct-protected        [2] EnvelopedData {SEQUENCE OF O
        direct-protected-auth [4] AuthEnvelopedData {SEQUENCE OF
        }
```

5. Public/Private Key Binding

An update to the PKCS #15 v1.2 draft provided for cryptographic binding between the private key and public key data. This protects the otherwise typically unprotected public-key objects from undetectable manipulation. This cryptographic binding is added by extending the existing privateXXXKey types with new privateXXXKeyExt types that include the cryptographic binding:

```
PrivateKeyType ::= CHOICE {
        privateRSAKey   PrivateKeyObject {PrivateRSAKeyAttribute
        privateECKey    [0] PrivateKeyObject {PrivateECKeyAttrib
        privateDHKey    [1] PrivateKeyObject {PrivateDHKeyAttrib
        privateDSAKey   [2] PrivateKeyObject {PrivateDSAKeyAttri
        privateKEAKey   [3] PrivateKeyObject {PrivateKEAKeyAttri
        privateRSAKeyExt [4] PrivateKeyObject {PrivateRSAKeyAttr
        privateECKeyExt [5] PrivateKeyObject {PrivateECKeyAttrib
        privateDSAKeyExt [6] PrivateKeyObject {PrivateDSAKeyAttr
        ...
        }
```

The Ext variants wrap the original XXXPrivateKeyObject in an additional SEQUENCE that adds an [ESSCertIDv2] field, with the ESSCertIDv2 restricted to contain only a SHA-2 hash of the public key data in SubjectPublicKeyInfo form. In other words the ESSCertIDv2:

```
ESSCertIDv2 ::= SEQUENCE {
        hashAlgorithm   AlgorithmIdentifier DEFAULT {algorithm i
        certHash                Hash,
        issuerSerial    IssuerSerial OPTIONAL
        }
```

is present as:

```
ESSCertIDv2 ::=  SEQUENCE {
        certHash                OCTET STRING SIZE(32),
        }
```

The resulting XXXPrivateKeyObject is then, in an ASN.1-like notation:

```
XXXPrivateKeyObject ::= SEQUENCE {
        spkiHash                ESSCertIDv2,
        -- Original PrivateXXXKeyObject
        }
```

For example for an ECC private key the original:

```
ECPrivateKey ::= INTEGER
```

would become in extended form with cryptographic binding:

```
ECPrivateKeyExt ::= SEQUENCE {
        spkiHash                ESSCertIDv2,
        value                   INTEGER
        }
```

For an RSA private key the original:

```
RSAPrivateKeyObject ::= SEQUENCE {
        modulus                 [0] INTEGER OPTIONAL, -- n
        ...
        coefficient             [7] INTEGER OPTIONAL -- inv(q) m
        }
```

would become in extended form with cryptographic binding:

```
RSAPrivateKeyObjectExt ::= SEQUENCE {
        spkiHash                ESSCertIDv2,
        value                   SEQUENCE {
                modulus         [0] INTEGER OPTIONAL, -- n
                ...
                coefficient     [7] INTEGER OPTIONAL -- inv(q) m
                }
        }
```

6. IANA Considerations

This document has no IANA actions.

7. Security Considerations

This document serves to document minor updates to the original PKCS #15 standard, there are no security considerations present beyond those in the original standard.

8. Normative References

[ESSCertIDv2] Schaad, J., "Enhanced Security Services (ESS) Update: Adding CertID Algorithm Agility", RFC 5035, August 2007, <http://www.ietf.org/rfc/rfc5035.txt>.

Author's Address

Peter Gutmann
University of Auckland
Department of Computer Science
Auckland
New Zealand

Email: pgut001@cs.auckland.ac.nz