Symmetric Key Package Content Type
draft-ietf-keyprov-symmetrickeyformat-01.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on May 18, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

This document defines the symmetric key format content type. It is transport independent. The Cryptographic Message Syntax [RFC3852] can be used to digitally sign, digest, authenticate, or encrypt this content type.
1. Introduction

This document defines the symmetric key format content type. It is transport independent. The Cryptographic Message Syntax [RFC3852] can be used to digitally sign, digest, authenticate, or encrypt this content type.

1.1. Requirements 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 [RFC2119].

1.2. ASN.1 Syntax Notation

The key package is defined using the ASN.1 [X.680,X.681].
2. Symmetric Key Package Content Type

The symmetric key package content type is used to transfer one or more plaintext symmetric keys from one party to another. A symmetric key package MAY be encapsulated in one or more CMS protecting content types. This content type must be DER encoded [X.690].

The symmetric key package content type has the following syntax:

```plaintext
PKCS7-CONTENT-TYPE ::= TYPE-IDENTIFIER

symmetric-key-package PKCS7-CONTENT-TYPE ::= 
  { SymmetricKeyPackage IDENTIFIED BY id-ct-KP-sKeyPackage }

id-ct-KP-sKeyPackage OBJECT IDENTIFIER ::= |
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
    smime(16) ct(1) 25 }

SymmetricKeyPackage ::= SEQUENCE {
  version          KeyPkgVersion DEFAULT v1,
  sKeyPkgAtts  [0] SEQUENCE SIZE (1..MAX) OF Attribute OPTIONAL,
  sKeys            SymmetricKeys }

SymmetricKeys ::= SEQUENCE SIZE (1..MAX) OF OneSymmetricKey

OneSymmetricKey ::= SEQUENCE {
  sKeyAttrs  SEQUENCE SIZE (1..MAX) OF Attribute OPTIONAL,
  sKey       OCTET STRING }

KeyPkgVersion ::= INTEGER  { v1(1), ... }
```

The SymmetricKeyPackage fields are used as follows:

- **version** identifies version of the symmetric key package content structure. For this version of the specification, the default value, v1, MUST be used.

- **sKeyPkgAtts** optionally provides attributes that apply to all of the symmetric keys in the package. If an attribute appears here it MUST NOT also be included in sKeyAttrs.

- **sKeys** contains a sequence of OneSymmetricKey values. This structure is discussed below.
The OneSymmetricKey fields are used as follows:

- sKeyAttrs optionally provides attributes that apply to one symmetric key. If an attribute appears here it MUST NOT also be included in sKeyPkgAttrs.

- sKey contains the key value encoded as an OCTET STRING.

3. Security Considerations

The symmetric key package contents are not protected. This content type can be combined with a security protocol to protect the contents of the package.

4. IANA Considerations

None: All identifiers are already registered. Please remove this section prior to publication as an RFC.

5. References

5.1. Normative References


5.2. Non-Normative References

APPENDIX A: ASN.1 Module

This appendix provides the normative ASN.1 definitions for the structures described in this specification using ASN.1 as defined in X.680 and X.681.

SymmetricKeyPackageModulev1

\{ iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
  smime(16) modules(0) 33 \}

DEFINITIONS IMPLICIT TAGS ::= 

BEGIN

-- EXPORTS ALL

-- IMPORTS NOTHING

PKCS7-CONTENT-TYPE ::= TYPE-IDENTIFIER

KeyPackageContentTypes PKCS7-CONTENT-TYPE ::= { 
  symmetric-key-package |
  ... -- Expect additional content types -- 
}

symmetric-key-package PKCS7-CONTENT-TYPE ::= 
  { SymmetricKeyPackage IDENTIFIED BY id-ct-KP-sKeyPackage }

id-ct-KP-sKeyPackage OBJECT IDENTIFIER ::= 
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) 
    smime(16) ct(1) 25 }

SymmetricKeyPackage ::= SEQUENCE { 
  version           KeyPkgVersion DEFAULT v1,
  sKeyPkgAttrs  [0] SEQUENCE SIZE (1..MAX) OF Attribute OPTIONAL,
  sKeys             SymmetricKeys }

SymmetricKeys ::= SEQUENCE SIZE (1..MAX) OF OneSymmetricKey

OneSymmetricKey ::= SEQUENCE { 
  sKeyAttrs        SEQUENCE SIZE (1..MAX) OF Attribute OPTIONAL,
  sKey             OCTET STRING }

KeyPkgVersion ::= INTEGER  { v1(1), ... }
Attribute ::= SEQUENCE {
  type ATTRIBUTE.&id ((SupportedAttributes)),
  values SET SIZE (1..MAX) OF ATTRIBUTE.&Type
    ((SupportedAttributes){@type}) }

SupportedAttributes ATTRIBUTE ::= { ... }

ATTRIBUTE ::= CLASS {
  &derivation ATTRIBUTE OPTIONAL,
  &Type OPTIONAL,
  -- either &Type or &derivation required
  &equality-match MATCHING-RULE OPTIONAL,
  &ordering-match MATCHING-RULE OPTIONAL,
  &substrings-match MATCHING-RULE OPTIONAL,
  &single-valued BOOLEAN DEFAULT FALSE,
  &collective BOOLEAN DEFAULT FALSE,
  -- operational extensions
  &no-user-modification BOOLEAN DEFAULT FALSE,
  &usage AttributeUsage DEFAULT userApplications,
  &id OBJECT IDENTIFIER UNIQUE }

WITH SYNTAX {
[ SUBTYPE OF &derivation ]
[ WITH SYNTAX &Type ]
[ EQUALITY MATCHING RULE &equality-match ]
[ ORDERING MATCHING RULE &ordering-match ]
[ SUBSTRINGS MATCHING RULE &substrings-match ]
[ SINGLE VALUE &single-valued ]
[ COLLECTIVE &collective ]
[ NO USER MODIFICATION &no-user-modification ]
[ USAGE &usage ]
ID &id }

MATCHING-RULE ::= CLASS {
  &AssertionType OPTIONAL,
  &id OBJECT IDENTIFIER UNIQUE }

WITH SYNTAX {
  [ SYNTAX &AssertionType ]
  ID &id }

AttributeType ::= ATTRIBUTE.&id

AttributeValue ::= ATTRIBUTE.&Type
AttributeUsage ::= ENUMERATED {
    userApplications (0),
    directoryOperation (1),
    distributedOperation (2),
    dSAOperation (3) }

END

Author's Address

Sean Turner
IECA, Inc.
3057 Nutley Street, Suite 106
Fairfax, VA 22031
USA

Email: turners@ieca.com

Russ Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA 20170
USA

Email: housley@vigilsec.com