             Use of the CAST-128 Encryption Algorithm in CMS

Status of this Memo

Copyright Notice

Abstract

   This document specifies how to incorporate CAST-128 (RFC2144) into
   the S/MIME Cryptographic Message Syntax (CMS) as an additional
   algorithm for symmetric encryption. The relevant OIDs and processing
   steps are provided so that CAST-128 may be included in the CMS
   specification (RFC2630) for symmetric content and key encryption.

   The key words "MUST", "MUST NOT", "REQUIRED", "SHOULD", "SHOULD NOT",
   "RECOMMENDED", "MAY", and "OPTIONAL" in this document (in uppercase,
   as shown) are to be interpreted as described in [RFC2119].

1. Motivation

   S/MIME (Secure/Multipurpose Internet Mail Extensions) [SMIME2,
   SMIME3] is a set of specifications for the secure transport of MIME
   objects. In the current (S/MIME v3) specifications the mandatory-
   to-implement symmetric algorithm for content encryption and key
   encryption is triple-DES (3DES). While this is perfectly acceptable
   in many cases because the security of 3DES is generally considered to
   be high, for some environments 3DES may be seen to be too slow. In
   part to help alleviate such performance concerns, S/MIME has allowed
   any number of (optional) additional algorithms to be used for
   symmetric content and key encryption.

   The CAST-128 encryption algorithm [RFC2144, Adams] is a well-studied
   symmetric cipher that has a number of appealing features, including
   relatively high performance and a variable key size (from 40 bits to
   128 bits). It is available royalty-free and license-free for

commercial and non-commercial uses worldwide [IPR], and therefore is
widely used in a number of applications around the Internet. It thus
seems to be a suitable optional encryption algorithm for S/MIME.

This document describes how to use CAST-128 within the S/MIME CMS
specification.

2. Specification

This section provides the OIDs and processing information necessary
for CAST-128 to be used for content and key encryption in CMS.

2.1 OIDs for Content and Key Encryption

CAST-128 is added to the set of optional symmetric encryption
algorithms in CMS by providing two unique object identifiers (OIDs).
One OID defines the content encryption algorithm and the other
defines the key encryption algorithm. Thus a CMS agent can apply
CAST-128 either for content or key encryption by selecting the
corresponding object identifier, supplying the required parameter,
and starting the program code.

For content encryption the use of CAST-128 in cipher block chaining
(CBC) mode is RECOMMENDED. The key length is variable (from 40 to
128 bits in 1-octet increments).

The CAST-128 content-encryption algorithm in CBC mode has the
following object identifier:

```
  cast5CBC OBJECT IDENTIFIER ::= {iso(1) member-body(2)
      us(840) nt(113533) nsn(7) algorithms(66) 10}
```

The parameter associated with this object identifier contains the
initial vector IV and the key length:

```
  cast5CBCParameters ::= SEQUENCE {
      iv          OCTET STRING DEFAULT 0,
      -- Initialization vector
      keyLength   INTEGER
      -- Key length, in bits
  }
```

Comments regarding the use of the IV may be found in [RFC2144].

The key-wrap/unwrap procedures used to encrypt/decrypt a CAST-128
content-encryption key with a CAST-128 key-encryption key are
specified in Section 2.2. Generation and distribution of key-
encryption keys are beyond the scope of this document.

   The CAST-128 key-encryption algorithm has the following object
   identifier:

      cast5CMSkeywrap OBJECT IDENTIFIER ::= { iso(1)
          member-body(2) us(840) nt(113533) nsn(7)
          algorithms(66) 15}

   The parameter associated with this object identifier contains only
   the key length (because the key wrapping procedure itself defines how
   and when to use an IV):

      cast5CMSkeywrapParameter ::= INTEGER
        -- key length, in bits

2.2 Key Wrapping and Unwrapping

   CAST-128 key wrapping and unwrapping is done in conformance with CMS
   [RFC2630].

2.2.1 CAST-128 Key Wrap

   Key wrapping with CAST-128 is identical to [RFC2630], Sections 12.6.1
   and 12.6.4, with "RC2" replaced by "CAST-128" in the introduction to
   12.6.4. Only 128-bit CAST-128 keys may be used as key-encryption
   keys, and they MUST be used with the cast5CMSkeywrapParameter set to
   128. It is RECOMMENDED that the size of the content-encryption key
   and the size of the key-encryption key be equal (since the security
   of the content will be at most the smaller of these two values).

2.2.2 CAST-128 Key Unwrap

   Key unwrapping with CAST-128 is identical to [RFC2630], Sections
   12.6.1 and 12.6.5, with "RC2" replaced by "CAST-128" in the
   introduction to 12.6.5.

3. Using CAST-128 in S/MIME Clients

   An S/MIME client SHOULD announce the set of cryptographic functions
   it supports by using the S/MIME capabilities attribute. This
   attribute provides a partial list of OIDs of cryptographic functions
   and MUST be signed by the client. The functions' OIDs SHOULD be
   logically separated in functional categories and MUST be ordered with
   respect to their preference. If an S/MIME client is required to
   support symmetric encryption with CAST-128, the capabilities
   attribute MUST contain the cast5CBC OID specified above in the
   category of symmetric algorithms. The parameter associated with this
   OID (see above) MUST be used to indicate supported key length. For
   example, when the supported key length is 128 bits, the

SMIMECapability SEQUENCE representing CAST-128 MUST be DER-encoded as
the following hexadecimal string:
301106092A864886F67D07420A300402020080.

When a sending agent creates an encrypted message, it has to decide
which type of encryption algorithm to use. In general the decision
process involves information obtained from the capabilities lists
included in messages received from the recipient, as well as other
information such as private agreements, user preferences, legal
restrictions, and so on. If users require CAST-128 for symmetric
encryption, it MUST be supported by the S/MIME clients on both the
sending and receiving side, and it MUST be set in the user
preferences.

4. Security Considerations

This document specifies the use of the CAST-128 symmetric cipher for
encrypting the content of a CMS message and for encrypting the
symmetric key used to encrypt the content of a CMS message. Although
CAST-128 allows keys of variable length to be used, it must be
recognized that smaller key sizes (e.g., 40, 56, or 64 bits) may be
unacceptably weak for some environments. The use of larger key sizes
(e.g., 128 bits) is always RECOMMENDED (when relevant import, export,
or other laws permit). It is also RECOMMENDED that the size of the
content-encryption key and the size of the key-encryption key be
equal (since the security of the content will be at most the smaller
of these two values).

References

   [Adams]   C. Adams, "Constructing Symmetric Ciphers using the CAST
               Design Procedure", Designs, Codes, and Cryptography,
               vol.12, no.3, November 1997, pp.71-104.

   [IPR]   See the "IETF Page of Intellectual Property Rights
               Notices", http://www.ietf.cnri.reston.va.us/ipr.html

   [RFC2144] Adams, C., "The CAST-128 Encryption Algorithm", RFC 2144,
               May 1997.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC2630] Housley, R., "Cryptographic Message Syntax", RFC 2630, June
               1999.

   [SMIME2]   Dusse, S., Hoffman, P., Ramsdell, B., Lundblade, L. and L.
              Repka, "S/MIME Version 2 Message Specification", RFC 2311,
              March 1998.

              Dusse, S., Hoffman, P., Ramsdell, B. and J. Weinstein,
              "S/MIME Version 2 Certificate Handling", RFC 2312, March
              1998.

   [SMIME3]   Ramsdell, B., "S/MIME Version 3 Certificate Handling", RFC
              2632, June 1999.

              Ramsdell, B., "S/MIME Version 3 Message Specification", RFC
              2633, June 1999.

Author's Address

   Carlisle Adams
   Entrust Technologies
   1000 Innovation Drive,
   Kanata, Ontario, Canada K2K 3E7

   EMail: cadams@entrust.com

Full Copyright Statement

Acknowledgement