Network Working Group
Request for Comments: 5409
Category: Informational

L. Martin
Voltage Security
M. Schertler
Axway

January 2009

Using the Boneh-Franklin and Boneh-Boyen Identity-Based Encryption Algorithms with the Cryptographic Message Syntax (CMS)

Status of This Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document describes the conventions for using the Boneh-Franklin (BF) and Boneh-Boyen (BB1) identity-based encryption algorithms in the Cryptographic Message Syntax (CMS) to encrypt content-encryption keys. Object identifiers and the convention for encoding a recipient's identity are also defined.

Table of Contents

1. Introduction

This document defines the way to use the Boneh-Franklin [IBCS] and Boneh-Boyen [IBCS] identity-based encryption (IBE) public-key algorithms in the Cryptographic Message Syntax (CMS) [CMS]. IBE is a public-key technology for encrypting content-encryption keys (CEKs) that can be implemented within the framework of the CMS: the recipient's identity is incorporated into the EnvelopedData CMS content type using the OtherRecipientInfo CHOICE in the RecipientInfo field as defined in Section 6.2.5 of [CMS]. This document does not describe the implementation of the BF and BB1 algorithms, which are described in detail in [IBCS].

IBE algorithms are a type of public-key cryptographic algorithm in which the public key is calculated directly from a user's identity instead of being generated randomly. This requires a different set of steps for encryption and decryption than would be used with other public-key algorithms, and these steps are defined in Sections 4 and 5 of this document, respectively.

This document also defines the object identifiers and syntax of the object that is used to define the identity of a message recipient.

CMS values and identity objects are defined using ASN.1 [ASN1].

1.1. 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 [KEYWORDS].

1.2. IBE Overview

In addition to the client components that are described in this document, the following additional components are required for a complete IBE messaging system.

- A Private-Key Generator (PKG). The PKG contains the cryptographic material, known as a master secret, for generating an individual's IBE private key. A PKG accepts an IBE user's private key request and, after successfully authenticating them in some way, returns their IBE private key.

- A Public Parameter Server (PPS). IBE system parameters include publicly sharable cryptographic material, known as IBE public parameters, and policy information for the PKG. A PPS provides a well-known location for distribution of IBE public parameters and policy information for the IBE PKG.

The interactions of senders and receivers of IBE-encrypted messages are described in [IBE]. All communications between users of an IBE system and the PPS or PKG MUST be protected using TLS [TLS] as described in [IBE]. This provides confidentiality and integrity of all information that is delivered to users as well as authentication of the PPS and PKG.

2. Using Identity-Based Encryption

To use IBE, the ori field in RecipientInfo MUST be used. The fields are set as follows: oriType is set to ibeORIType; oriValue is set to ibeORIValue.

These fields have the following meanings:

ibeORIType defines the object identifier (OID) that indicates that the subsequent ibeORIValue is the information necessary to decrypt the message using IBE. This field MUST be set to the following:

```
ibeORIType OBJECT IDENTIFIER ::= {
  joint-iso-itu(2) country(16) us(840)
  organization(1) identicrypt(114334)
  ibcs(1) cms(4) ori-oid(1) version(1)
}
```

ibeORIValue defines the identity that was used in the IBE algorithm to encrypt the CEK. This is an IBERecipientInfo type, which is defined as follows:

```
IBERecipientInfo ::= SEQUENCE {
  cmsVersion         INTEGER { v3(3) },
  keyFetchMethod     OBJECT IDENTIFIER,
  recipientIdentity  IBEIdentityInfo,
  serverInfo         SEQUENCE SIZE (1..MAX) OF
    OIDValuePairs OPTIONAL,
  encryptedKey       EncryptedKey
}
```

The fields of IBERecipientInfo MUST be set as follows.

The cmsVersion MUST be set to 3.

The keyFetchMethod is the OID that defines the method of retrieving the private key that the recipient MUST use. This SHOULD be set to uriPPSOID [IBE], which is defined to be the following:

```
uriPPSOID OBJECT IDENTIFIER ::= {
  joint-iso-itu-t(2) country(16) us(840)
  organization(1) identicrypt(114334)
  pps-schemas(3) ic-schemas(1) pps-uri(1) version(1)
}
```

The recipientIdentity is the data that the sender used to calculate the IBE public key that the sender used to encrypt the content-encryption key. This recipientIdentity is used to calculate IBE public and private keys as described in [IBCS]. This MUST be a DER-encoded [DER] IBEIdentityInfo type [IBE], which is defined as follows:

```
IBEIdentityInfo ::= SEQUENCE {
  district        IA5String,
  serial          INTEGER,
  identityType    OBJECT IDENTIFIER,
  identityData    OCTET STRING
}
```

The identityType defines the format that is used to encode the information that defines the identity of the recipient. This MUST be set to cmsIdentityOID to indicate that identityData contains an EmailIdentityData type. The value of cmsIdentityOID is the following:

```
cmsIdentityOID OBJECT IDENTIFIER ::= {
  joint-iso-itu-t(2) country(16) us(840)
  organization(1) identicrypt(114334)
  keyschemas(2) icschemas(1) email(1) version(1)
}
```

The identityData MUST be an EmailIdentityData type, which is defined as follows:

```
EmailIdentityData ::= SEQUENCE {
  rfc822Name   IA5String,
  time         GeneralizedTime
}
```

The rfc822Name field is the email address of the recipient in the format defined in Section 4.2.1.6 of [PKIX] for the rfc822Name subjectAltName variant. Rules for encoding Internet mail addresses that include internationalized domain names are specified in Section 7.5 of [PKIX].

The value of the time field is the UTC time after which the sender wants to let the recipient decrypt the message, so it may be called the "not-before" time. This is usually set to the time when the message is encrypted, but MAY be set to a future time. The value of "time" MUST be expressed in Greenwich Mean Time (Zulu), MUST include seconds (i.e., times are always YYYYMMDDHHMMSSZ), even where the number of seconds is equal to zero, and MUST be expressed to the nearest second.

The sender of an IBE-encrypted message may want to express this time rounded to a time interval to create a key lifetime. A key lifetime reduces the number of IBE private keys that a recipient needs to retrieve, but still forces the IBE user to periodically re-authenticate. Based on the time interval chosen a recipient would only have to retrieve a new IBE key once during the interval. To do this, follow the steps below. Let "time-interval" be the number of seconds in this larger time interval.

1. Find the GeneralizedTime for the not-before value.
2. Convert this GeneralizedTime into the number of seconds since January 1, 1970. Call this "total-time".
3. Calculate reduced-time = (floor (total-time / time-interval)) * time-interval.
4. Convert reduced-time to a GeneralizedTime to get the not-before "time" value.

An example of this algorithm for computing a one-week time interval is as follows.

1. Suppose that the GeneralizedTime is 20020401000000Z.
2. Then the total-time is 1017612000.
3. A time-interval of 1 week is 604800 seconds. So the reduced-time = (floor (1017612000 / 604800)) * 604800 = 1017273600.
4. This gives the GeneralizedTime form of the reduced-time of 20020328000000Z.

When issuing IBE private keys, a PKG SHOULD NOT issue them too far into the future. This restriction is to prevent an adversary who obtains an IBE user's authentication credentials from requesting private keys far into the future and therefore negating the periodic IBE user re-authentication that key lifetime provides. For example, if a one-week period is chosen for the key lifetime, then IBE private keys should not be issued more than one week in advance. Otherwise, once an adversary gains access to the PKG via the stolen IBE user credentials, they can request all future keys and negate the IBE user authentication restraints in place.

The serverInfo is an optional sequence of OID-value pairs that are defined to be the following:

```
OIDValuePairs ::= SEQUENCE {
  fieldID     OBJECT IDENTIFIER,
  fieldData   OCTET STRING
}
```

These can be used to convey any other information that might be used by a PKG. Examples of such information could include the user interface that the recipient will experience. Differences in the user interface could include localization information or commercial branding information. A client MUST ignore any part of serverInfo that it is unable to process.

The encryptedKey is the result of encrypting the CEK with an IBE algorithm using recipientIdentity as the IBE public key.

3. Key Encryption Algorithm Identifiers

The BF and BB1 algorithms as defined in [IBCS] have the following object identifiers. These object identifiers are also defined in the ASN.1 module in [IBCS].

```
bf OBJECT IDENTIFIER ::= {
  joint-iso-itu-t(2) country(16) us(840)
  organization(1) identicrypt(114334)
  ibcs(1) ibcs1(1) ibe-algorithms(2) bf(1)
}
```

This is the object identifier that MUST be inserted in the keyEncryptionAlgorithm field in the CMS when the BF algorithm is used to encrypt the CEK.

```
bb1 OBJECT IDENTIFIER ::= {
  joint-iso-itu-t(2) country(16) us(840)
  organization(1) identicrypt(114334)
  ibcs(1) ibcs1(1) ibe-algorithms(2) bb1(2)
}
```

This is the object identifier that MUST be inserted in the keyEncryptionAlgorithm field in the CMS when the BB1 algorithm is used to encrypt the CEK.

4. Processing by the Sender

The sender of a message that uses IBE to encrypt content-encryption keys performs the following steps:

1. Selects a set of IBE public parameters to use in the subsequent steps in accordance with his local security policy. He then determines the URI where the public parameters can be obtained using the process described in [IBE]. This information MUST be encoded in the IBEIdentityInfo as described in Section 2.

2. Sets the fields of an OtherRecipientInfo object to their appropriate values as described in Section 2.

3. Calculates an IBE public key as defined in [IBCS] using this IBEIdentityInfo as the identity information.

4. This IBE public key is then used to encrypt the content-encryption key (CEK), using the algorithms that are defined in [IBCS].

5. Sets encryptedKey to the IBE-encrypted CEK.

6. Within the CMS, keyEncryptionAlgorithm MUST then be set to the appropriate OID for the IBE algorithm that was used (see Section 3).

5. Processing by the Receiver

Upon receiving a message that has a CEK encrypted with IBE, the recipient performs the following steps to decrypt the CEK:

1. Determines that the CEK is IBE-encrypted by noting that the oriType of the OtherRecipientInfo type is set to ibeORIType.

2. Determines that the recipientIdentity was used as the identity in IBE encryption of the CEK.

3. Determines the location of the IBE public parameters and the IBE Private Key Generator as described in [IBE].

4. Obtains the IBE public parameters from the location determined in Step 3 using the process defined in [IBE].

5. Obtains the IBE private key needed to decrypt the encrypted CEK using the process defined in [IBE].

6. Decrypts the CEK using the IBE private key obtained in Step 4 using the algorithms described in [IBCS].

6. ASN.1 Module

The following ASN.1 module summarizes the ASN.1 definitions defined by this document.

```
IBECMS-module {
  joint-iso-itu-t(2) country(16) us(840)
  organization(1) identicrypt(114334)
  ibcs(1) cms(4) module(5) version(1)
}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS IBEIdentityInfo, uriPPSOID FROM

   IBEARCH-module { joint-iso-itu-t(2) country(16)
     us(840) organization(1) identicrypt(114334) ibcs(1)
     ibearch(5) module(5) version(1)
   };

IBEOtherRecipientInfo ::= SEQUENCE {
  oriType   OBJECT IDENTIFIER,
  oriValue  IBERecipientInfo
}
```

```
   ibeORIType OBJECT IDENTIFIER ::= {
     joint-iso-itu-t(2) country(16) us(840)
     organization(1) identicrypt(114334)
     ibcs(1) cms(4) ori-oid(1) version(1)
   }

   IBERecipientInfo ::= SEQUENCE {
     cmsVersion         INTEGER { v3(3) },
     keyFetchMethod     OBJECT IDENTIFIER,
     recipientIdentity  IBEIdentityInfo,
     serverInfo         SEQUENCE SIZE (1..MAX) OF
       OIDValuePairs OPTIONAL,
     encryptedKey       EncryptedKey
   }


   OIDValuePairs ::= SEQUENCE {
     fieldID     OBJECT IDENTIFIER,
     fieldData   OCTET STRING
   }

   EncryptedKey ::= OCTET STRING

   EmailIdentityData ::= SEQUENCE {
     rfc822Name   IA5String,
     time         GeneralizedTime
   }

   cmsIdentityOID OBJECT IDENTIFIER ::= {
     joint-iso-itu-t(2) country(16) us(840)
     organization(1) identicrypt(114334)
     keyschemas(2) icschemas(1) email(1) version(1)
   }

   END
```

7. Security Considerations

This document is based on [CMS], [IBCS], and [IBE], and the relevant security considerations of those documents apply.

7.1. Attacks outside the Scope of This Document

Attacks on the cryptographic algorithms that are used to implement IBE are outside the scope of this document. Such attacks are detailed in [IBCS], which defines parameters that give 80-bit, 112-bit, 128-bit, and 256-bit encryption strength. We assume that capable administrators of an IBE system will select parameters that provide a sufficient resistance to cryptanalytic attacks by adversaries.

Attacks that give an adversary the ability to access or change the information on a PPS or PKG, especially the cryptographic material (referred to in this document as the master secret), will defeat the security of an IBE system. In particular, if the cryptographic material is compromised, the adversary will have the ability to recreate any user's private key and therefore decrypt all messages protected with the corresponding public key. To address this concern, it is highly RECOMMENDED that best practices for physical and operational security for PPS and PKG servers be followed and that these servers be configured (sometimes known as hardened) in accordance with best current practices [NIST]. An IBE system SHOULD be operated in an environment where illicit access to the PKG or the ability to modify the information distributed by the PPS is infeasible for attackers to obtain.

Attacks that require administrative access or IBE-user-equivalent access to machines used by either the client or the server components defined in this document are also outside the scope of this document.

We also assume that all administrators of a system implementing the protocols that are defined in this document are trustworthy and will not abuse their authority to bypass the security provided by an IBE system. This is of particular importance with an IBE system, for an administrator of a PKG could potentially abuse his authority and configure the PKG to grant him any IBE private key that the PKG is capable of calculating. To minimize the possibility of administrators doing this, a system implementing IBE SHOULD implement n-out-of-m control for critical administrative functions and SHOULD maintain auditable logs of all security-critical events that occur in an operating IBE system.

Similarly, we assume that users of an IBE system will behave responsibly, not sharing their authentication credentials with others. Thus, attacks that require such assumptions are outside the scope of this document.

7.2. Attacks within the Scope of This Document

Attacks within the scope of this document are those that allow an adversary to:

- passively monitor information transmitted between users of an IBE system and the PPS and PKG

- masquerade as a PPS or PKG
- perform a denial-of-service (DoS) attack on a PPS or PKG
- easily guess an IBE user's authentication credential

7.3. Attacks to Which the Protocols Defined in This Document Are Susceptible

All communications between users of an IBE system and the PPS or PKG are protected using TLS [TLS]. The IBE system defined in this document provides no additional security for the communications between IBE users and the PPS or PKG. Therefore, the described IBE system is completely dependent on the TLS security mechanisms for authentication of the PKG or PPS server and for confidentiality and integrity of the communications. Should there be a compromise of the TLS security mechanisms, the integrity of all communications between an IBE user and the PPS or PKG will be suspect.

The protocols defined in this document do not explicitly defend against an attacker masquerading as a legitimate IBE PPS or PKG. The protocols rely on the server authentication mechanism of TLS [TLS]. In addition to the TLS server authentication mechanism, IBE client software can provide protection against this possibility by providing user interface capabilities that allows users to visually determine that a connection to PPS and PKG servers is legitimate. This additional capability can help ensure that users cannot easily be tricked into providing valid authorization credentials to an attacker.

The protocols defined in this document are also vulnerable to attacks against an IBE PPS or PKG. Denial-of-service attacks against either component can result in users unable to encrypt or decrypt using IBE, and users of an IBE system SHOULD take the appropriate countermeasures [DOS, BGPDOS] that their use of IBE requires.

The IBE user authentication method used by an IBE PKG SHOULD be of sufficient strength to prevent attackers from easily guessing the IBE user's authentication credentials through trial and error.

8. References

8.1. Normative References

[ASN1] ITU-T Recommendation X.680: Information Technology - "Abstract Syntax Notation One (ASN.1): Specification of Basic Notation", July 2002.

[CMS] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004.

[DER] ITU-T Recommendation X.690: OSI Networking and System Aspects: Abstract Syntax Notation One (ASN.1), July 2002.

[DOS] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.

[IBCS] Boyen, X. and L. Martin, "Identity-Based Cryptography Standard (IBCS) #1: Supersingular Curve Implementations of the BF and BB1 Cryptosystems", RFC 5091, December 2007.

[IBE] Appenzeller, G., Martin, L., and M. Schertler, "Identity-Based Encryption Architecture and Supporting Data Structures", RFC 5408, January 2009.

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

[PKIX] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

[TLS] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

8.2. Informative References

[BGPDOS] Turk, D., "Configuring BGP to Block Denial-of-Service Attacks", RFC 3882, September 2004.

[NIST] M. Souppaya, J. Wack and K. Kent, "Security Configuration Checklist Program for IT Products - Guidance for Checklist Users and Developers", NIST Special Publication SP 800-70, May 2005.

Authors' Addresses

Luther Martin
Voltage Security
1070 Arastradero Rd, Suite 100
Palo Alto, CA 94304
USA

Phone: +1 650 543 1280
EMail: martin@voltage.com

Mark Schertler
Axway
1600 Seaport Blvd, Suite 400
Redwood City, CA 94063
USA

Phone: +1 650 216 2039
EMail: mschertler@us.axway.com