S/MIME Working Group
Internet Draft
Expires: December 2006

G. Appenzeller
Voltage Security

June 2006

# Identity-based Encryption
# Private Key Request Protocol

<draft-ietf-smime-ibepkg-00.txt>

## Status of this Document



## Abstract

This document describes a protocol to request private keys from a Private Key Generator (PKG) for an identity-based encryption system.

## Table of Contents

## 1. Introduction

An identity-based encryption system [IBEARCH] allows the encryption of messages using a user s identity plus a set of public parameters. For decryption users need a private key that is generated by a private key generator. This document defines a protocol to retrieve private keys from the private key generator (PKG) of an IBE system.

This document does not describe the actual algorithms used for encryption or the mathematical structure of the public parameters, they are described in [IBCS]. It also does not describe the communication protocol to retrieve public parameters, it is described in [IBEPPS].

### 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 [KEY].

## 2. Overview

In an identity-based encryption (IBE) system messages are encrypted using a public key that is locally calculated from public parameters and a user s identity and decrypted using a private key that corresponds to the user s public key. These private keys are generated by a private key generator (PKG) based on a global secret called a master secret.

When requesting a private key, a client has to transmit two parameters:
1. The identity it is requesting a key for

2. Authentication credentials
These two are often not the same as a single user may have access to multiple aliases. For example an email user may have access to the keys that correspond to two different email addresses, e.g. bob@example.com and bob.smith@example.com.

This document defines the protocol to request private keys, a minimum user authentication method for interoperability, and how to pass authentication credentials to the server. It assumes that a client has already determined the URL of the PKG. This can be done from hints included in the IBE message format [IBCMS] and the system parameters of the IBE system [IBEPPS].

## 3. Private Key Request

To request a private key, a client performs a HTTP POST method as defined in [RFC2616]. The request MUST happen over a secure protocol. The requesting client MUST support either SSL v 3.0 [SSL3] protocol or TLS v 1.1 [TLS]. When requesting the URL the client MUST abort the key request if the server certificate verification of the SSL or TLS connection fails [RFC2618]. Doing so is critical to protect the authentication credentials and the private key against man-in-the-middle attacks when it is transmitted from the key server to the client.

### 3.1. Request Structure

The POST method contains in its body the following XML structure:

```
<ibe:request xmlns:ibe=\"http://www.ietf.org/tbd/ibepkg\">
   <ibe:header>
      <ibe:client version="clientID"/>
   </ibe:header>
   <ibe:body>
   <ibe:keyRequest>
      <ibe:algorithm>
        <oid> algorithmOID </oid>
      </ibe:algorithm>
      <ibe:id>
      ibeIdentityInfo
      </ibe:id>
   </ibe:keyRequest>
   </ibe:body>
</ibe:request>
```

A <ibe:request> SHOULD include a <ibe:clientID> element that identifies the client type and client version.

A key request MUST contain a valid ibeIdentityInfo that the private key is requested for. This identity is the BASE64 encoding of the DER encoding of the ASN.1 structure IBEIdentityInfo as defined in [IBECMS].

A key request MUST contain a <ibe:algorithm> element that contains a XER encoded ASN.1 OBJECT IDENTIFIER that identifies algorithm for which a key is requested. OIDs for the BB1 and BF algorithms are listed in [IBCS].

A client MAY include optional additional XML elements in the <ibe:body> part of the key request.

### 3.2. Authentication

When a client requests a key from a PKG, the PKG SHOULD authenticate the client before issuing the key. Authentication may either be done through the key request structure or as part of the secure transport protocol.

A client or server implementing the request protocol MUST support HTTP Basic Auth as described in [RFC2617]. A client and server SHOULD also support HTTP Digest Auth as defined in [RFC2617].

For authentication methods that are not done by the transport protocol, a client MAY include additional authentication information in xml elements in the body part of the key request. If a client does not know how to authenticate to a server, the client MAY send a key request without authentication information. If the key server requires the client to authenticate externally, it MAY reply with a 201 response code as defined below to redirect the client to the correct authentication mechanism.

## 4. Server Response Format

The key server replies to the HTTP request with an HTTP response. If the response has a redirect, client error or server error status code, the client MUST abort the key request and fail.

If the PKG replies with a HTTP response that has a status code indicating success, the body of the reply MUST contain the following XML structure:

```
<ibe:response xmlns:ic="http://www.ietf.org/tbd/icsip">
   <ibe:responseType value="responseCode"/>
   <ibe:body>
      bodyTags
   </ibe:body>
</ibe:response>
```

The responseCode describes the type of response from the key server. The list of currently defined response codes is:

```
 100  KEY_FOLLOWS
 101  RESERVED
 201  FOLLOW_ENROLL_URL
 300  SYSTEM_ERROR
 301  INVALID_REQUEST
 303  CLIENT_OBSOLETE
 304  AUTHORIZATION DENIED
```

### 4.1. Response containing a Private Key

If the key request was successful, the key server responds with KEY FOLLOWS, and the <ibe:body> must contain a <ibe:privateKey> tag with a valid private key. An example of this is shown below.

```
  <ibe:response xmlns:ic=" http://www.ietf.org/tbd/icsip">
     <ibe:responseType value="100"/>
     <ibe:body>
        <ibe:privateKey>
          privateKey
        </ibe:privateKey>
     </ibe:body>
  </ibe:response>
```

The privateKey is the Base64 encoding of the DER encoding of the following ASN.1 structure:

```
IBEPrivateKeyReply ::= SEQUENCE {
   pkgIdentity    IBEIdentityInfo,
   pgkAlgorithm   OBJECT IDENTIFIER
   pkgKeyData     OCTET STRING
   pkgOptions     SEQUENCE OF Extensions
}
```

The pkgIdentity is an IBEIdentityInfo structure as defined in [IBECMS]. It MUST be identical to the IBEIdentityInfo structure that was sent in the key request.

The pkgAlgorithm is an OID that identifies the algorithm of the returned private key. The OIDs for the BB and BF algorithms are defined in [IBCS].

The pkgKeyData is a ASN.1 structure that contains the actual private key. Private key formats for the BB and BF algorithms are defined in [IBCS].

A server MAY pass back additional information to a client in the pkgOptions structure. The contents of the structure are defined in the ASN.1 module below.

### 4.2. Responses containing a Redirect

A Key Server MAY support authenticating user to external authentication mechanism. If this is the case, the server replies to the client with response code 201 and the body MUST contain a <ibe:location> element that specifies the URL of the authentication mechanism. An example is shown below.

```
<ibe:response xmlns:ic=" http://www.ietf.org/tbd/icsip">
   <ibe:responseType value="201"/>
   <ibe:body>
      <ibe:location url="http://www.example.com/enroll.asp"/>
   </ibe:body>
</ibe:response>
```

The client can now contact the authentication mechanism to obtain authentication credentials. Once the client has obtained the credential, it sends a new key request to the PKG with the correct authentication token contained in the request.

### 4.3. Responses indicating an Error

If the server replies with a 3xx error code, the client MUST abort the request and discard any data that is part of the response.

The meaning of the response codes for errors is as follows:

300 This indicates an internal server error of the PKG.

301 The request to the server is invalid or the server is not able to fulfill this type of request.

303 The server is not able to serve key requests for this type of client. A client with a newer version of the protocol is required.

304 The key request was processed correctly, but the authentication credentials provided by the user were invalid, could not be verified, or do not allow access to keys for this identity.

## 5. ASN.1 Module

This section defines the ASN.1 module for the encodings discussed in section 4.

```
IBEPKG { joint-iso-itu(2) country(16) us(840) organization(1)
   identicrypt(114334) ibcs(1) ibcs2(2) pks(1) module (5) version(1)
}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS IBEIdentityInfo
  FROM BFCMS
      { joint-iso-itu(2) country(16) us(840) organization(1)
        identicrypt(114334) ibcs(1) cms(4) module(5) version(1)
      };

ibcs OBJECT IDENTIFIER ::= { joint-iso-itu(2) country(16)
      us(840) organization(1) identicrypt(114334) ibcs(1)
}

-- Private Key Format

IBEPrivateKeyReply ::= SEQUENCE {
   pkgIdentity    IBEIdentityInfo,
   pgkKeyType     OBJECT IDENTIFIER,
   pkgKeyData     OCTET STRING,
   pkgOptions     Extensions
}

Extensions ::= SEQUENCE OF Extension

Extension  ::=  SEQUENCE  {
  id         OBJECT IDENTIFIER,
  value      OCTET STRING
}

ibeParamExt OBJECT IDENTIFIER ::= {
  ibcs ibcs2(2) pks(1) extensions(2)
}

END
```

## 6. Security Considerations

This entire document relates to security considerations.

## 7. IANA Considerations

No further action by the IANA is necessary for the protocols described in this document.

## 8. References

### 8.1. Normative References

[CMS] R. Housley, Cryptographic Message Syntax, RFC 3369, August 2002.

[KEY] S. Brander, Key Words for Use in RFCs to Indicate Requirement Levels, BCP 14, RFC 2119, March 1997.

[P1363] IEEE P1363.3, Standards Specifications for Public-Key Cryptography, 2001.

[SSL3] A. Frier, P. Karlton, and P. Kocher, "The SSL 3.0 Protocol", Netscape Communications Corp., Nov 18, 1996.

[TLS] T. Dierks, E. Rescorla, The Transport Layer Security Protocol Version 1.1, RFC 4346, April 2006.

[X509] ITU-T Recommendation X.509 (1997 E): Information Technology - Open Systems Interconnection - The Directory: Authentication Framework, June 1997.

[IBEARCH] L. Martin, M. Schertler, Identity-based encryption architecture, draft-ietf-smime-ibearch-00.txt.

[IBCS] X. Boyen, L. Martin, Identity-based cryptography standard (IBCS) #1: supersingular curve implementations of the BF and BB1 cryptosystems, draft-ietf-smime-ibcs-00.txt.

[IBECMS] M. Schertler, L. Martin, Using the Boneh-Franklin identity-based encryption algorithm with the Cryptographic Message Syntax (CMS), draft-ietf-smime-bfibecms-01.txt.

[IBEPPS] G. Appenzeller, Parameter and policy lookup for identity-based encryption, draft-ietf-smime-ibepkg-00.txt.

[RFC2396] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.Author's Address

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frysyk, H., Masinter, L., Leach, P. and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC2617] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., Sink, E. and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999. [jg646]

## Author's Address

Guido Appenzeller
Voltage Security
1070 Arastradero Rd Suite 100
Palo Alto CA 94304

Phone: +1 650 543 1280
Email: guido@voltage.com

## Intellectual Property Statement

## Disclaimer of Validity



## Copyright Statement



## Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.