Independent Submission
Request for Comments: 6539
Category: Informational
ISSN: 2070-1721

V. Cakulev
G. Sundaram
I. Broustis
Alcatel Lucent

March 2012

IBAKE: Identity-Based Authenticated Key Exchange

Abstract

Cryptographic protocols based on public-key methods have been traditionally based on certificates and Public Key Infrastructure (PKI) to support certificate management. The emerging field of Identity-Based Encryption (IBE) protocols allows simplification of infrastructure requirements via a Private-Key Generator (PKG) while providing the same flexibility. However, one significant limitation of IBE methods is that the PKG can end up being a de facto key escrow server, with undesirable consequences. Another observed deficiency is a lack of mutual authentication of communicating parties. This document specifies the Identity-Based Authenticated Key Exchange (IBAKE) protocol. IBAKE does not suffer from the key escrow problem and in addition provides mutual authentication as well as perfect forward and backward secrecy.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6539.

Independent Submissions Editor Note

This document specifies the Identity-Based Authenticated Key Exchange (IBAKE) protocol. Due to its specialized nature, this document experienced limited review within the Internet Community. Readers of this RFC should carefully evaluate its value for implementation and deployment.

Table of Contents

1. Introduction

Authenticated key agreements are cryptographic protocols where two or more participants authenticate each other and agree on key material used for securing future communication. These protocols could be symmetric key or asymmetric public-key protocols. Symmetric-key protocols require an out-of-band security mechanism to bootstrap a secret key. On the other hand, public-key protocols traditionally require certificates and a large-scale Public Key Infrastructure (PKI). Clearly, public-key methods are more flexible; however, the requirement for certificates and a large-scale PKI have proved to be challenging. In particular, efficient methods to support large-scale certificate revocation and management have proved to be elusive.

Recently, Identity-Based Encryption (IBE) protocols have been proposed as a viable alternative to public-key methods by replacing the PKI with a Private-Key Generator (PKG). However, one significant limitation of IBE methods is that the PKG can end up being a de facto

key escrow entity (i.e., an entity that has sufficient information to decrypt communicated data), with undesirable consequences. Another limitation is a lack of mutual authentication between communicating parties. This document specifies an Identity-Based Authenticated Key Encryption (IBAKE) protocol that does not suffer from the key escrow problem and that provides mutual authentication. In addition, the scheme described in this document allows the use of time-bound public identities and corresponding public and private keys, resulting in automatic expiration of private keys at the end of a time span indicated in the identity itself. With the self-expiration of the public identities, the traditional real-time validity verification and revocation procedures used with certificates are not required. For example, if the public identity is bound to one day, then, at the end of the day, the public/private key pair issued to this peer will simply not be valid anymore. Nevertheless, just as with public-key-based certificate systems, if there is a need to revoke keys before the designated expiry time, communication with a third party will be needed. Finally, the protocol also provides forward and backward secrecy of session keys; i.e., a session key produced using IBAKE is always fresh and unrelated to any past or future sessions between the protocol participants.

2. Requirements Notation

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].

2.1. IBE: Definition

Identity-Based Encryption (IBE) is a public-key encryption technology that allows a public key to be calculated from an identity and a set of public parameters, and the corresponding private key to be calculated from the public key. The public key can then be used by an Initiator to encrypt messages that the recipient can decrypt using the corresponding private key. The IBE framework is defined in [RFC5091], [RFC5408], and [RFC5409].

2.2. Abbreviations

EC	Elliptic Curve
IBE	Identity-Based Encryption
IBAKE	Identity-Based Authenticated Key Exchange
IDi	Initiator’s Identity

IDr	Responder’s Identity
K_PUB	Public Key
PKG	Private-Key Generator
PKI	Public Key Infrastructure

2.3. Conventions

- E is an elliptic curve over a finite field F.
- P is a point on E of large prime order.
- s is a non-zero positive integer. s is a secret stored in a PKG. This is a system-wide secret and not revealed outside the PKG.
- sP is the public key of the system that is known to all participants. sP denotes a point on E, and denotes the point P added to itself s times where addition refers to the group operation on E.
- H1 is a known hash function that takes a string and assigns it to a point on the elliptic curve, i.e., H1(A) = QA on E, where A is usually based on the identity.
- E(k, A) denotes that A is IBE-encrypted with the key k.
- s||t denotes concatenation of the strings s and t.
- K_PUBx denotes a public key of x.

3. Identity-Based Authenticated Key Exchange

3.1. Overview

IBAKE consists of a three-way exchange between an Initiator and a Responder. In the figure below, a conceptual signaling diagram of IBAKE is depicted.

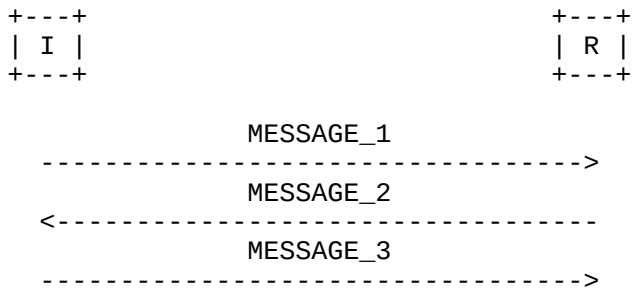

Figure 1: Example IBAKE Message Exchange

The Initiator (I) and Responder (R) are attempting to mutually authenticate each other and agree on a key using IBAKE. This specification assumes that the Initiator and the Responder trust a third party -- the PKG. Rather than a single PKG, different PKGs may be involved, e.g., one for the Initiator and one for the Responder. The Initiator and the Responder do not share any credentials; however, they know or can obtain each other's public identity (key) as well as the public parameters of each other's PKG. This specification does not make any assumption on when and how the private keys are obtained. However, to complete the protocol described (i.e., to decrypt encrypted messages in the IBAKE protocol exchange), the Initiator and the Responder need to have their respective private keys. The procedures needed to obtain the private keys and public parameters are outside the scope of this specification. The details of these procedures can be found in [RFC5091] and [RFC5408]. Finally, the protocol described in this document relies on the use of elliptic curves. Section 3.3 discusses the choice of elliptic curves. However, how the Initiator and the Responder agree on a specific elliptic curve is left to the application that is leveraging the IBAKE protocol (see [EAP-IBAKE], for example).

The Initiator chooses a random x. In the first step, the Initiator computes xP (i.e., P, as a point on E, added to itself x times using the addition law on E); encrypts xP, the IDi, and the IDr using the Responder's public key (e.g., K_PUBr=H1(IDr||date)); and includes

this encrypted information in MESSAGE_1 sent to the Responder. In this step, encryption refers to IBE as described in [RFC5091] and [RFC5408].

The Responder, upon receiving the message, IBE-decrypts it using its private key (e.g., a private key for that date), and obtains xP. The Responder further chooses a random y and computes yP. The Responder then IBE-encrypts the Initiator's identity (IDi), its own identity (IDr), xP, and yP using the Initiator's public key (e.g., K_PUBi=H1(IDi||date)). The Responder includes this encrypted information in MESSAGE_2 sent to the Initiator.

The Initiator, upon receiving and IBE-decrypting MESSAGE_2, obtains yP. Subsequently, the Initiator sends MESSAGE_3, which includes the IBE-encrypted IDi, IDr, and yP, to the Responder. At this point, both the Initiator and the Responder are able to compute the same session key as xyP.

3.2. IBAKE Message Exchange

Initially, the Initiator selects a random x and computes xP; the Initiator MUST use a fresh, random value for x on each run of the protocol. The Initiator then encrypts xP, the IDi, and the IDr using the Responder's public key (e.g., K_PUBr=H1(IDr||date)). The Initiator includes this encrypted information in MESSAGE_1 and sends it to the Responder, as shown below.

Initiator ----> Responder

MESSAGE_1 = E(K_PUBr, IDi || IDr || xP)

Upon receiving MESSAGE_1, the Responder SHALL perform the following:

- Decrypt the message as specified in [RFC5091] and [RFC5408].
- Obtain xP.
- Select a random y and compute yP. The Responder MUST use a fresh, random value for x on each run of the protocol.
- Encrypt the Initiator's identity (IDi), its own identity (IDr), xP, and yP using the Initiator's public key (K_PUBi).

Responder ----> Initiator

MESSAGE_2 = E(K_PUBi, IDi || IDr || xP || yP)

Upon receiving MESSAGE_2, the Initiator SHALL perform the following:

- Decrypt the message as specified in [RFC5091] and [RFC5408].
- Verify that the received xP is the same as that sent in MESSAGE_1.
- Obtain yP.
- Encrypt its own identity (IDi), the Responder's identity (IDr), and yP using the Responder's public key (K_PUBi).

Initiator ----> Responder

MESSAGE_3 = E(K_PUBr, IDi || IDr || yP)

Upon receiving MESSAGE_3, the Responder SHALL perform the following:

- Decrypt the message as specified in [RFC5091] and [RFC5408].
- Verify that the received yP is the same as that sent in MESSAGE_2.

If any of the above verifications fail, the protocol halts; otherwise, following this exchange, both the Initiator and the Responder have authenticated each other and are able to compute xyP as the session key. At this point, both protocol participants MUST discard all intermediate cryptographic values, including x and y. Similarly, both parties MUST immediately discard these values whenever the protocol terminates as a result of a verification failure or timeout.

3.3. Discussion

Properties of the protocol are as follows:

- Immunity from key escrow: Observe that all of the steps in the protocol exchange are encrypted using IBE. So, clearly, the PKG can decrypt all of the exchanges. However, given the assumption that PKGs are trusted and well behaved (e.g., PKGs will not mount an active man-in-the-middle (MitM) attack), they cannot compute the session key. This is because of the hardness of the Elliptic Curve Diffie-Hellman problem. In other words, given xP and yP, it is computationally hard to compute xyP.
- Mutually authenticated key agreement: Observe that all of the steps in the protocol exchange are encrypted using IBE. In particular, only the Responder and its corresponding PKG can decrypt the contents of MESSAGE_1 and MESSAGE_3 sent by the Initiator, and similarly only the Initiator and its corresponding

PKG can decrypt the contents of MESSAGE_2 sent by the Responder. Again, given the assumption made above -- that PKGs are trusted and well behaved (e.g., a PKG will not impersonate a user to which it issued a private key) -- upon receiving MESSAGE_2, the Initiator can verify the Responder's authenticity, since xP could have been sent in MESSAGE_2 only after decryption of the contents of MESSAGE_1 by the Responder. Similarly, upon receiving MESSAGE_3, the Responder can verify the Initiator's authenticity, since yP could have been sent back in MESSAGE_3 only after correct decryption of the contents of MESSAGE_2 by the Initiator. Finally, both the Initiator and the Responder can agree on the same session key. In other words, IBAKE is a mutually authenticated key agreement protocol based on IBE. The hardness of the key agreement protocol relies on the hardness of the Elliptic Curve Diffie-Hellman problem. Thus, in any practical implementation, care should be devoted to the choice of elliptic curve.

- Perfect forward and backward secrecy: Since x and y are random, xyP is always fresh and unrelated to any past or future sessions between the Initiator and the Responder.

- No passwords: Clearly, the IBAKE protocol does not require any offline exchange of passwords or secret keys between the Initiator and the Responder. In fact, the method is applicable to any two parties communicating for the first time through any communication network. The only requirement is to ensure that both the Initiator and the Responder are aware of each other's public keys and the public parameters of the PKG that generated the corresponding private keys.

- PKG availability: Observe that PKGs need not be contacted during an IBAKE protocol exchange, which dramatically reduces the availability requirements on PKGs.

- Choice of elliptic curves: This specification relies on the use of elliptic curves for both IBE and Elliptic Curve Diffie-Hellman exchange. When making a decision on the choice of elliptic curves, it is beneficial to choose two different elliptic curves -- a non-supersingular curve for the internal calculations of Elliptic Curve Diffie-Hellman values xP and yP, and a supersingular curve for the IBE encryption/decryption. For the calculations of Elliptic Curve Diffie-Hellman values, it is beneficial to use the curves recommended by NIST [FIPS-186]. These curves make the calculations simpler while keeping the security high. On the other hand, IBE systems are based on bilinear pairings. Therefore, the choice of an elliptic curve for

IBE is restricted to a family of supersingular elliptic curves over finite fields of large prime characteristic. The appropriate elliptic curves for IBE are described in [RFC5091].

- Implementation considerations: An implementation of IBAKE would consist of two primary modules, i.e., point addition operations over a NIST curve, and IBE operations over a supersingular curve. The implementation of both modules only needs to be aware of the following parameters: (a) the full description of the curves that are in use (fixed or negotiated), (b) the public parameters of the PKG used for the derivation of IBE private keys, and (c) the exact public identity of each IBAKE participant. The knowledge of these parameters is sufficient to perform Elliptic Curve Cryptography (ECC) operations in different terminals and produce the same results, independently of the implementation.

4. Security Considerations

This document is based on the basic IBE protocol, as specified in [BF], [RFC5091]), [RFC5408], and [RFC5409], and as such inherits some properties of that protocol. For instance, by concatenating the "date" with the identity (to derive the public key), the need for any key revocation mechanisms is virtually eliminated. Moreover, by allowing the participants to acquire multiple private keys (e.g., for duration of contract) the availability requirements on the PKG are also reduced without any reduction in security. The granularity associated with the date is a matter of security policy and as such is a decision made by the PKG administrator. However, the granularity applicable to any given participant should be publicly available and known to other participants. For example, this information can be made available in the same venue that provides "public information" on a PKG server (i.e., P, sP) needed to execute IBE.

4.1. General

Attacks on the cryptographic algorithms used in IBE are outside the scope of this document. It is assumed that any administrator will pay attention to the desired strengths of the relevant cryptographic algorithms based on an up-to-date understanding of the strength of these algorithms from published literature, as well as to known attacks.

It is assumed that the PKGs are secure, not compromised, trusted, and will not engage in launching active attacks independently or in a collaborative environment. Nevertheless, if an active adversary can fool the parties into believing that it is a legitimate PKG, then it can mount a successful MitM attack. Therefore, care should be taken

when choosing a PKG. In addition, any malicious insider could potentially launch passive attacks (by decryption of one or more message exchanges offline). While it is in the best interest of administrators to prevent such an issue, it is hard to eliminate this problem. Hence, it is assumed that such problems will persist, and hence the session key agreement protocols are designed to protect participants from passive adversaries.

It is also assumed that the communication between participants and their respective PKGs is secure. Therefore, in any implementation of the protocols described in this document, administrators of any PKG have to ensure that communication with participants is secure and not compromised.

Finally, concatenating the date to the identity ensures that the corresponding private key is applicable only to that date. This serves to limit the damage related to a leakage or compromise of private keys to just that date. This, in particular, eliminates the revocation mechanisms that are typical to various certificate-based public key protocols.

4.2. IBAKE Protocol

For the basic IBAKE protocol, from a cryptographic perspective, the following security considerations apply.

In every step, IBE is used, with the recipient's public key. This guarantees that only the intended recipient of the message and its corresponding PKG can decrypt the message [BF].

Next, the use of identities within the encrypted payload is intended to eliminate some basic reflection attacks. For instance, suppose we did not use identities as part of the encrypted payload, in the first step of the IBAKE protocol exchange (i.e., MESSAGE_1 of Figure 1 in Section 3.1). Furthermore, assume that an adversary has access to the conversation between the Initiator and the Responder and can actively snoop packets and drop/modify them before routing them to the destination. For instance, assume that the IP source address and destination address can be modified by the adversary. After the first message is sent by the Initiator (to the Responder), the adversary can take over and trap the packet. Next, the adversary can modify the IP source address to include the adversary's IP address, before routing it on to the Responder. The Responder will assume that the request for an IBAKE session came from the adversary, and will execute step 2 of the IBAKE protocol exchange (i.e., MESSAGE_2 of Figure 1 in Section 3.1) but encrypt it using the adversary's public key. The above message can be decrypted by the adversary (and only by the adversary). In particular, since the second message

includes the challenge sent by the Initiator to the Responder, the adversary will now learn the challenge sent by the Initiator. Following this, the adversary can carry on a conversation with the Initiator, "pretending" to be the Responder. This attack will be eliminated if identities are used as part of the encrypted payload. In summary, at the end of the exchange, both the Initiator and the Responder can mutually authenticate each other and agree on a session key.

Recall that IBE guarantees that only the recipient of the message can decrypt the message using the private key, with the caveat that the PKG that generated the private key of the recipient of the message can decrypt the message as well. However, the PKG cannot learn the public key xyP given xP and yP, based on the hardness of the Elliptic Curve Diffie-Hellman problem. This property of resistance to passive key escrow from the PKG is not applicable to the basic IBE protocols proposed in [RFC5091]), [RFC5408], and [RFC5409].

Observe that the protocol works even if the Initiator and Responder belong to two different PKGs. In particular, the parameters used for encryption to the Responder and parameters used for encryption to the Initiator can be completely different and independent of each other. Moreover, the elliptic curve used to generate the session key xyP can be completely different and can be chosen during the key exchange. If such flexibility is desired, then it would be required to add optional extra data to the protocol to exchange the algebraic primitives used in deriving the session key.

In addition to mutual authentication and resistance to passive escrow, the Diffie-Hellman property of the session key exchange guarantees perfect secrecy of keys. In other words, accidental leakage of one session key does not compromise past or future session keys between the same Initiator and Responder.

5. References

5.1. Normative References

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

5.2. Informative References

[BF] Boneh, D. and M. Franklin, "Identity-Based Encryption from the Weil Pairing", in SIAM Journal on Computing, Vol. 32, No. 3, pp. 586-615, 2003.

[EAP-IBAKE] Cakulev, V. and I. Broustis, "An EAP Authentication Method Based on Identity-Based Authenticated Key Exchange", Work in Progress, February 2012.

[FIPS-186] National Institute of Standards and Technology, "Digital Signature Standard (DSS)", FIPS Pub 186-3, June 2009.

[RFC5091] Boyen, X. and L. Martin, "Identity-Based Cryptography Standard (IBCS) #1: Supersingular Curve Implementations of the BF and BB1 Cryptosystems", RFC 5091, December 2007.

[RFC5408] Appenzeller, G., Martin, L., and M. Schertler, "Identity-Based Encryption Architecture and Supporting Data Structures", RFC 5408, January 2009.

[RFC5409] Martin, L. and M. Schertler, "Using the Boneh-Franklin and Boneh-Boyen Identity-Based Encryption Algorithms with the Cryptographic Message Syntax (CMS)", RFC 5409, January 2009.

Authors' Addresses

Violeta Cakulev
Alcatel Lucent
600 Mountain Ave.
3D-517
Murray Hill, NJ 07974
US

Phone: +1 908 582 3207
EMail: violeta.cakulev@alcatel-lucent.com

Ganapathy S. Sundaram
Alcatel Lucent
600 Mountain Ave.
3D-517
Murray Hill, NJ 07974
US

Phone: +1 908 582 3209
EMail: ganesh.sundaram@alcatel-lucent.com

Ioannis Broustis
Alcatel Lucent
600 Mountain Ave.
3D-526
Murray Hill, NJ 07974
US

Phone: +1 908 582 3744
EMail: ioannis.broustis@alcatel-lucent.com