Protecting Biometrics Using Signcryption

Abstract

Organizations should protect biometric information to manage security risk within the firewall perimeter and on public networks such as the Internet. Digital signatures and encryption combined in a single cryptographic operation can provide assurance of the confidentiality, integrity, and availability of biometric information. Signcryption provides data integrity, origin authentication, and confidentiality simultaneously in a single efficient cryptographic function.

This paper introduces the signcryption cryptographic primitive and describes its benefits. Current practice that relies on other hybrid cryptographic safeguards for mitigating security threats is described. Specific attention is given to authenticated encryption, a hybrid technique similar to signcryption. Abstract schema definitions are provided that illustrate how signcryption can be implemented in commonly used cryptographic message protocols to protect biometric information. Digital signatures and signature types are defined, and a brief history of their invention, evolution, implementation, and use is presented.

Biometric Protection

The X9.84 biometric information security management standard describes how messages containing biometric information can be cryptographically bound to set of security attributes (X9 Financial Services [X9], 2010b). This binding under a digital signature provides data integrity and origin authenticity of biometric data and management information. A digital signature does not provide biometric data confidentiality.

X9.84 requires biometric data elements in a message to be kept confidential. The standard suggests encrypting biometric data elements, then signing the message. This approach requires symmetric key management in addition to the certificate and asymmetric key management needed to support a digital signature.

As an alternative, this paper describes using the signcryption cryptographic primitive (Smith, 2005) to signcrypt biometric data elements in a message, then sign the message using the same keys. With this approach, message processing is improved, there are fewer keys to manage, and the need to establish a shared secret with a relying party is eliminated.

Signcryption simultaneously signs and encrypts information in a single operation to achieve origin authentication, data integrity, and confidentiality. Signcryption can be performed faster than traditional signature followed by encryption techniques, and has the advantage over symmetric, shared key encryption of making non-repudiation possible (Dent, 2004). The same asymmetric key pair used to protect the biometric data elements can be used to bind a message and any attributes.

Efficient Applications

Signcryption schemes “achieve confidentiality and authentication simultaneously by combining public-key encryption and digital signatures” and offer better overall performance and security (Barbosa & Farshim, 2008). These schemes provide “shorter cipher text and/or lower computational cost” (Li & Wong, 2009). The efficiencies of signcryption make it ideal for

protecting biometric information in environments constrained by bandwidth limitations (e.g., wireless mobile devices), high volumes of transactions (e.g., Internet commerce), or size or cost of storage (e.g., smart cards). Applications based on signcryption techniques “range from efficient security solutions for mobile communications and secure and authenticated message delivery to economical electronic payment systems” (AVIPAC, 2010).

Signcryption provides the high-level cryptographic key protection required to support the biometric matching operations of the Department of Defense (DoD) Biometrics Identity Management Agency (BIMA), while meeting the performance demands of biometric collection devices in the mobile battlefield. In implementing security for “Mobile Ad-hoc NETwork (MANET)” applications, designers have used a “signcryption type key exchange scheme Direct Key Exchange Using Time Stamp (DKEUTS)” to ensure “fair key exchange among high level tiers of military MANETs” (Yavuz, Alagöz, & Anarim, 2010).

Cross-domain security solutions needed to support efficient biometric exchange can also be provided using signcryption. Signcryption has been used to implement an “Elliptic Curve Pintsov-Vanstone Signature Scheme (ECPVSS)” to provide security for “bandwidth/computational resource limited Regular Ground Nodes (RGN)” (Yavuz, Alagöz, & Anarim, 2010). This yielded a security mechanism that provided “sufficient security for each tier”, while preventing “the network from being overloaded due to the unnecessary cryptographic operations” (Yavuz, et al., 2010).

Signcryption schemes require “only a single key pair for each user" while tradition sign-then-encrypt schemes "require two: one for encrypting and one for signing” (Smith, 2005). A scheme recently proposed for use in firewalls, allows a third party to verify the authenticity of a signcrypted message without having to reveal the plaintext to the third party (Mohamed & Elkamchouchi, 2009). This property could be used to address privacy concerns for biometric data.

Hybrid Cryptography

Signcryption is an efficient, relatively new hybrid cryptographic primitive. Informally, hybrid cryptography is the “branch of asymmetric cryptography that makes use of convenient symmetric techniques to remove some of the problems inherent in normal asymmetric cryptosystems” (Dent, 2004). Problems include those encountered “when trying to process long messages quickly” (Dent, 2004), such as large iris scan or fingerprint sets.

Hybrid cryptography is not new technology. Authenticated encryption (AE), a family of hybrid cryptographic techniques, is currently used to secure network communications. AE is “a shared-key based transform” that relies on a symmetric encryption scheme “to provide both privacy and integrity” (Bellare & Namprempre, 2008). Signcryption is the asymmetric analog of AE.

Some versions of Secure Sockets Layer (SSL) use the AE method “MAC-then-encrypt (MtE)” (Bellare & Namprempre, 2008) to “provide privacy and reliability” (Freier, Karlton, & Kocher, 1996) services. The Encapsulating Security Payload (ESP) protocol provides “confidentiality, data origin authentication”, and connectionless integrity (Kent, 2005) using an “Encrypt-then-MAC (EtM)” AE method (Bellare & Namprempre, 2008). The Transport Layer of Secure Shell protocol (SSH) uses “Encrypt-and-MAC (E&M)” (Bellare & Namprempre, 2008).

Security services provided by “symmetric encryption schemes and MAC algorithms” such as authenticated encryption rely on shared key, symmetric approaches, so non-repudiation is not

possible (Dent, 2004). While several signcryption schemes provide non-repudiation, there is no standardized signcryption schema. This paper defines an abstract schema for a new cryptographic message type, `SigncryptedData` that can protect the integrity, confidentiality, and origin authenticity of biometric information.

Signature History

Whitfield Diffie and Martin Hellman invented public key cryptography by establishing a theory for practical key distribution and digital signatures (1976). A practical implementation of their theory would occur two years later with the arrival of the RSA signature scheme developed by Rivest, Shamir, and Adleman (1978). RSA would become "the first practical signature scheme based on public-key techniques" (Menezes, et al., 1996, p. 482).

In their 1978 publication on digital signatures, there is no mention of protecting biometric information using the RSA scheme, though a wide variety of applications are called out. The authors predict an "era of 'electronic mail'" and note that digital signatures have "obvious applications in 'electronic mail'"(Rivest, et al., 1978, p. 120). The trio forms RSA Data Security to provide cryptographic solutions to help implement applications described in their paper.

As a means of securing electronic mail, they created the Public Key Cryptography Systems #7 (PKCS#7) Cryptographic Message Syntax (CMS) standard (Kaliski, 1998), a cryptographic message schema needed to send and receive secure messages over an open network. In CMS, each cryptographic type is treated as a complete message, ready for use by an application without modification. An extensibility feature allows these CMS message types to be extended.

Implementation

Signcryption can be implemented using the Cryptographic Message Syntax (CMS) defined in U. S. standard, X9.73:2010 *Cryptographic Message Syntax - ASN.1 and XML*, using a new CMS type defined in this paper. This new type, `SigncryptedData`, is based on the CMS type `SignedData` (X9, 2010a). Type `SignedData` is currently used to sign electronic mail, biometric enabled watch lists, biometric reference templates, and to provide security for biometric elements in the DoD Electronic Biometric Transmission Specification (EBTS) standard (BIMA, 2011).

The data content in the `SigncryptedData` type, which can be any type of information object in any format, is protected by a signature and encryption. In this paper, the protected content is some type of biometric information object, such as a DoD EBTS file or a biometric reference template. Either the entire content or specified components of the content are signcrypted.

There are three processing modes for this new CMS type: *signcrypted-content* mode, *signcrypted-attributes* mode, and *signcrypted-components* mode. In the *signcrypted-content* mode data content of any type is signcrypted. In the *signcrypted-attributes* mode, data content and associated attributes of any type are signcrypted. In the *signcrypted-content* mode, components of the data content are signcrypted, and then the resulting content is signed along with a set of associated attributes.

A schema is provided for using signcryption in Cryptographic Message Syntax (CMS) applications. This schema can be used to implement key management techniques to protect biometric and other information. It is defined using Abstract Syntax Definition One (ASN.1), a

formal notation specified in a series of international standards for information exchange. The schema is presented as an ASN.1 module, a form suitable for input to ASN.1 programming language code generation and syntax checking tools.

ASN.1 is used today in telecommunications, financial services, and electronic commerce protocols, and in military and commercial applications to facilitate successful information exchange, and to improve the chances of interoperability between applications. ASN.1 values can be represented in compact binary formats and as XML (Extensible Markup Language) markup. Every value of every ASN.1 type can be represented in both binary and XML formats.

The following ASN.1 module has been proposed to the U.S. Accredited Standards Committee (ASC) X9 as a contribution towards the next revision of the X9.73 CMS standard:

```
SigncryptedCMS {
   iso(1) identified-organization(3) tc68(133) country(16) x9(840)
      x9Standards(9) x9-73(73) module(0) signcrypt(7) v2012(1) }

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- EXPORTS All --

IMPORTS

   -- X9.73 CMS Object Identifiers --

   id-signcryptedData, id-signcryptedParts, id-cms-XPath
      FROM CMSObjectIdentifers {
         iso(1) identified-organization(3) tc68(133) country(16) x9(840)
            x9Standards(9) x9-73(73) module(0) oids(1) v2009(1) }

   -- X9.73 Cryptographic Message Syntax (CMS) - ASN.1 and XML --

   ATTRIBUTE, Certificates, CMSVersion, CONTENTS, CRLs,
   DigestAlgorithmIdentifier,DigestAlgorithmIdentifiers,
   EncapsulatedContentInfo, SignedAttributes,SignatureAlgorithmIdentifier,
   SignatureValue, UnsignedAttributes
      FROM CryptographicMessageSyntax {
         iso(1) identified-organization(3) tc68(133) country(16) x9(840)
            x9Standards(9) x9-73(73) module(0) cms(2) v2009(1) };

SigncryptedData ::= SEQUENCE {
   version           CMSVersion,
   digestAlgorithms  DigestAlgorithmIdentifiers,
   encapContentInfo  EncapsulatedContentInfo,
   certificates      [0] Certificates  OPTIONAL,
   crls              [1] CRLs  OPTIONAL,
   signcrypterInfos  SigncrypterInfos
}

SigncrypterInfos ::= SET SIZE(0..MAX) OF SigncrypterInfo

SigncrypterInfo ::= SEQUENCE {
   version             CMSVersion,
   sids                SigncrypterIDs,
   digestAlgorithm     [0] DigestAlgorithmIdentifier  OPTIONAL,
   signedAttrs         [1] SignedAttributes  OPTIONAL,
```

```
   signatureAlgorithm  SignatureAlgorithmIdentifier,
   signature           SignatureValue,
   unsignedAttrs       [2] UnsignedAttributes  OPTIONAL
}

SigncrypterIDs ::= SEQUENCE {
   sender     KeyPairIdentifier,
   recipient  KeyPairIdentifier
}

KeyPairIdentifier ::= CHOICE {
  issuerAndSerialNumber  IssuerAndSerialNumber,
   subjectKeyIdentifier   [0] SubjectKeyIdentifier,
   certHash               [1] CertHash
}

-- Signcrypted components signed attribute --

SigncryptedParts ::= Signcrypted {{ Manifest }}

Manifest SIGNCRYPTED::= {
   xPathManifest,
   ... -- Expect additional manifest objects  --
}

xPathManifest SIGNCRYPTED ::= {
   OID id-cms-XPath PARMS XPathSet
}

XPathSet ::= SEQUENCE SIZE(1..MAX) OF XPath

XPath ::= UTF8String (CONSTRAINED BY { -- XML Path Language 2.0 -- })

signcryptedParts ATTRIBUTE ::= {
   WITH SYNTAX SigncryptedParts ID id-signcryptedParts
}

-- SigncryptedData content type --

signcryptedData CONTENTS ::= {
   SigncryptedData IDENTIFIED BY id-signcryptedData
}

-- Supporting defintions --

SIGNCRYPTED ::= CLASS {
   &id    OBJECT IDENTIFIER  UNIQUE,
   &Type  OPTIONAL
}
  WITH SYNTAX { OID &id [ PARMS &Type ] }

Signcrypted { SIGNCRYPTED:IOSet } ::= SEQUENCE {
   name   SIGNCRYPTED.&id({IOSet}),
   parts  SIGNCRYPTED.&Type({IOSet}{@name})  OPTIONAL
}

END  -- SigncryptedCMS –
```

A new SigncryptedData CMS type is defined in the module. This type differs from type SignedData by only one component, signcrypterInfos, which is a set of per-message-recipient information. Each element in the set, a value of type SigncrypterInfo, provides information for the entity whose public key is used to perform the cryptographic operations.

The key pairs of the message sender and recipient are used to sign and signcrypt data, and to verify signatures and recover encrypted content. These two key pairs are identified in a value of type SigncrypterIDs. Each key pair is typically associated with a public key identity certificate using one of the choice alternatives of type KeyPairIdentifier.

In the *signcrypted-content* mode, content of any type or format is signcrypted using the signcryption algorithm identified by the value of the signatureAlgorithm component of type SigncrypterInfo. The message sender applies this algorithm to signcrypt the content using the public and private keys of the sender and the public key of the recipient. These keys are identified by the sids component of type SigncrypterInfo, a value of type KeyPairIdentifier. The signcryption results are placed in the signature component of type SigncrypterInfo.

The plaintext content is not carried in the encapContentInfo component of type SigncryptedData. Only the signcrypted content is available to the message recipient. The recipient uses the provided signcryption algorithm, the public key of the sender, and their own public- private key pair to verify the signature and recover the plaintext from the cryptogram in the signature component of type SigncrypterInfo.

In the *signcrypted-attributes* mode, content of any type or format, together with any number of attributes of any type or format are signcrypted as described for the *signcrypted-content* mode. For the *signcrypted-attributes* mode, the result of concatenating a complete ASN.1 encoding of the signedAttrs component of type SigncrypterInfo, a value of type SignedAttributes, to a complete encoding of the content is signcrypted by the sender, is signcrypted. The signcryption results are placed in the signature component of type SigncrypterInfo. The optional signedAttrs component of type SigncrypterInfo is not included in the message.

In the *signcrypted-content* mode, components of content of any type or format are signcrypted as described for the *signcrypted-content* mode. The resulting content containing signcrypted components together with at least three required attributes are then signed following the processing requirements for type SignedData (X9, 2010a) [1]. A list of signcrypted components in the form of the signcryptedParts attribute defined in this paper must be included in the signed attributes, to ensure they are bound to the content under a digital signature.

The signcryptedParts attribute carries a value of type SigncryptedParts. The definition of type SigncryptedParts is based on a parameterized type Signcrypted {{ Manifest }}, whose components are constrained by the elements in the information object set Manifest. Any number of objects can be added to the Manifest object set, and these can be used to locate signcrypted components in any document or file of any type. Only one element is defined in this paper, the information object xPathManifest.

[1] When there are attributes in type SignedData, the messageDigest and contentType attributes are required.

The `xPathManifest` object carries a value of type `XPathSet`, a series of values of type `XPath`. These XPath expressions can be used to locate any signcrypted element in any XML instance document, such as an element in a DoD EBTS 3.0 XML transaction, or an element in a financial transaction based on the ISO 20022 UNIversal Financial Industry (UNIFI) message scheme (International Organization for Standardization [ISO], 2004).

When the contents of an XML element are signcrypted, the sender includes the outer markup tags in the signcryption. The markup between these tags is then replaced with a character string representation of the signcryption results, a value of XML type *base64Binary*, which represents arbitrary Base64-encoded binary data. A message recipient uses an XPath expression to locate the tags in an XML instance document. The signature can then be verified and the plaintext content can be recovered. The recovered plaintext can then be used to replace the cryptogram with the recovered XML markup.

Conclusion

Cryptographic safeguards are needed to protect the confidentiality, integrity, and availability of biometric and other information assets on unprotected networks, such as the Internet. Authenticated encryption is the symmetric key analog to signcryption that has proven itself to be a reliable cryptographic safeguard in security protocols such as SSL, IPsec, and SSH. Signcryption provides a way for secure applications in a single cryptographic function to "integrate encryption and signature schemes in an efficient way without sacrificing each scheme's security" (Baek, Steinfeld, & Zheng, 2007). A new `SigncryptedData` CMS type can be used to protect the integrity, origin authenticity, and confidentiality of DoD EBTS transactions, ISO 20022 financial messages, and other types of information.

References

AVIPAC. (2010). Staff Profile - Yuliang Zheng. Retrieved March 31, 2012, from http://www.csse.monash.edu.au/research/avipac/staff_profiles/profile_yz.htm

Baek, J., Steinfeld, R., & Zheng, Y. (2007). Formal Proofs for the Security of Signcryption. *Journal of Cryptology*, *20*(2), 203-235. doi:10.1007/s00145-007-0211-0. Retrieved February 24, 2011, from International Security & Counter Terrorism Reference Center database.

Barbosa, M., & Farshim, P. (2008). Certificateless signcryption. *Asia CCS '08*. Retrieved April 12, 2012, from ACM Digital Library database.

Bellare, M., & Namprempre, C. (2008). Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. *Journal of Cryptology*, *21*(4), 469-491. doi:10.1007/s00145-008-9026-x. Retrieved April 12, 2012, from International Security & Counter Terrorism Reference Center database.

BIMA. (2011). Electronic Biometric Transmission Specification (EBTS). Retrieved April 12, 2012, from http://www.biometrics.dod.mil/Files/Documents/Standards/DoD_EBTS_v3_0.pdf

Dent, Alexander W. (2004). Hybrid cryptography, Cryptology ePrint Archive Report 2004/210.

Diffie, W. & Hellman, M. E. (1976). New directions in cryptography [Electronic version]. *IEEE*

Transactions on Information Theory, 22, 644-654.

Freier, A., Karlton, P., & Kocher, P. (1996). *The SSL protocol version 3.0*. Retrieved March 31, 2012, from http://www.mozilla.org/projects/security/pki/nss/ssl/draft302.txt

International Organization for Standardization. (2004). *ISO 20022 (All parts) Financial services – UNIversal Financial Industry message scheme*.

Kaliski, B. (1998, March). *PKCS #7: Cryptographic message syntax version 1.5.* Retrieved March 31, 2012, from http://www.ietf.org/rfc/rfc2315.txt

Kent, S. (2005). *IP encapsulating security payload (ESP).* Retrieved March 31, 2012, from http://www.faqs.org/rfc/rfc4303.txt

Li, Chung Ki, & Wong, Duncan S. (2009). Signcryption from randomness recoverable public key encryption. Retrieved April 12, 2012, from ScienceDirect database.

Menezes, A., Oorscot, P. van, Vanstone, S. (1996). *Handbook of applied cryptography* [Electronic version], 321-482.

Mohamed & Elkamchouchi. (2009). Elliptic curve signcryption with encrypted message authentication and forward secrecy. Retrieved March 31, 2012, from http://paper.ijcsns.org/07_book/200901/20090155.pdf

Rivest, R. L., Shamir, A., Adleman, L. (1978, February). A method for obtaining digital signatures and public-key cryptosystems [Electronic version]. *Communications of the ACM*, 21, 120–126.

Smith, C. (2005). Digital signcryption. Retrieved March 31, 2012, from http://www.signcryption.org/theses/pdffiles/Smith-MstrThesis-Digital_Signcryption.pdf

X9 Financial Services. (2010a). *ANSI X9.73:2010 Cryptographic Message Syntax - ASN.1 and XML*. U.S.A.: American National Standards Institute (ANSI).

X9 Financial Services. (2010b). *ANSI X9.84:2010 Biometric Information Management and Security for the Financial Services Industry*. U.S.A.: American National Standards Institute (ANSI).

Yavuz, A., Alagöz, F., & Anarim, E. (2010). A new multi-tier adaptive military MANET security protocol using hybrid cryptography and signcryption. *Turkish Journal of Electrical Engineering & Computer Sciences*, *18*(1), 1-21. doi:10.3906/elk-0904-6. Retrieved April 12, 2012, from Academic Search Premier database.

Zheng, Yuliang. (1998). Shortened Digital Signature, Signcryption and Compact and Unforgeable Key Agreement Schemes. Retrieved March 31, 2012, from http://grouper.ieee.org/groups/1363/StudyGroup/contributions/signcr.pdf