On the security of Joint Signature and Hybrid Encryption

M.Ayoub Khan and Y.P.Singh
Centre for Development of Advanced Computing
Ministry of Communications and Information Technology, Noida (INDIA)
email: khanayoub@yahoo.com, ypsingh@cdacnoida.in

Abstract—**This paper presents the security of joint signature and hybrid encryption .The proposed scheme combines the security of a document by hybrid encryption method and authenticity by digital signatures. IDEA-RSA algorithm is used for hybrid encryption and RSA digital signature algorithm is used to obtain digital signature (D). This joint signature scheme uses "encrypt-then-sign (ℰtS)" instead of, "sign-then-encrypt (Stℰ)". The security of RSA Digital Signature (𝒟) amplifies the privacy of ℰtS. The effectiveness and correctness of the proposed scheme is illustrated through implementation and its results. The proposed scheme achieved a speed of 2.8 Mbps**

Index Terms—**ANSI X9.31,DSS, IDEA, Hybrid, RSA,SHA-256**

I. Introduction

Electronic and mobile commerce is an emerging area involving applications, mobile devices, and middleware and wireless networks. Mobile financial applications are likely to be one of the most important components of m-commerce. These could involve various types of applications such as mobile-banking and mobile money transfer. The electronic commerce based on an open network is gradually being established as part of daily life. In the non-face-to-face transaction of electronic commerce, the authentication, encryption and integrity is very important.

The process of signing entails transforming the message and some secret information held by the entity into a tag called a digital signature [1]. Encryption and digital signatures techniques are fundamental in any cryptographic tools for privacy of the data and authenticity respectively. The cryptographic scheme providing both authenticity and privacy has been usually called an authenticated encryption, but was mainly studied in the symmetric setting [10, 11].

Digital signatures, like physical signatures, can verify a specific user affixed signature to a document and they can also verify that the document is the same as when the user has affixed the digital signature. A digital signature must be message dependent [2]. A major difference between physical and digital signature is that a digital signature cannot be constant. It must be a function of the entire document that it signs; changing even a single bit should produce a different signature. A digital Signature Standard (DSS) uses public-key cryptography methods to create digital signatures [3]. The integrity of the digital signature is tied to the security of the user's private-key. As long as the user's private-key is secure, then only the user can affix their digital signature to a document.

Every user has two special and related values, the public-key and the private-key that the DSS uses for creating and verifying a digital signature. Digital signature cannot be forged, and a signer cannot later deny the validity of his signature. This has obvious applications in electronic mail, e-cash and electronic fund transfer system [2].

In practical implementations, public-key algorithms are too inefficient to sign and encrypt long documents. To save signing time, digital signature algorithms are implemented with one-way hash function [3]. In this scheme, instead of signing a document, the hash of document is taken. After that, hash is encrypted by the private-key, thereby signing the document. This increases speed drastically. The security of hybrid-key setting depends on the method used. The security of base signature scheme can amplify the privacy of the ℰtS, while the security of base encryption scheme can do the same to the authenticity of Stℰ.

In this paper we concentrate on hybrid-key setting and digital signature which is more secure and the security relies on the problem of solving discrete logarithms and on factorization. The proposed scheme has all the features of symmetric, asymmetric algorithm and digital signature. This scheme eliminates the requirement of DSA which is 10-40 times slower in verification process [3]. The remainder of this paper is organized as follows: In section 2 a brief description of hybrid cryptographic system is given. In the next section the hashed-RSA algorithm is presented. Section 4 presents the proposed scheme and it is analyzed in detail. Section 5 gives design implementation and finally some conclusions will be given in the last section based on the implementation.

II. Hybrid Cryptosystem

The term "hybrid" is borrowed from the natural sciences, it means "crossbreed". In cryptography it refers to the combination of symmetric and asymmetric algorithms. The ISO/IEC JTC1/SC27 standardization committee [5] suggests that hybrid cryptography can be defined as the branch of asymmetric cryptography that makes use of convenient symmetric techniques to remove some of the problems inherent in normal asymmetric cryptosystem. The fast encryption speed of symmetric algorithm is coupled with the security of asymmetric cipher algorithm such as RSA. Certainly, breakable symmetric ciphers such as the Caesar, DES are not used. Rather, the so called IDEA (International Data Encryption Algorithm) will be used for hybrid

1-4244-0000-7/05/$20.00 ©2005 IEEE.

cryptographic system. IDEA was developed in 1991 by Professor Jim Massey, ETH Zurich, Switzerland, and his student Xuejia Lai [3, 4]. However, it is important to know that IDEA key consist of only 128 bits in comparison to 1024 bit of RSA as recommended for strong security applications. Such keys are encrypted even with the slow RSA Cipher in milliseconds. Thus, time is not an issue for the hybrid cryptographic systems. Traditionally, hybrid cryptography has been studied with building asymmetric encryption schemes. In these cryptosystems a symmetric encryption algorithm is used to solve the shortcomings, typically associated with encrypting long messages using asymmetric algorithms. Another development in hybrid cryptography is key encapsulation mechanism and symmetric data encapsulation mechanism (KEM-DEM) model for hybrid encryption algorithms [15, 16].

III. HASHED-RSA DIGITAL SIGNATURE ALGORITHM

The RSA public-key cryptosystem can be used to authenticate or identify another person or entity. In practice, the public exponent in the RSA algorithm is usually much smaller than the private exponent [13]. This means that verification of a signature is faster than signing. This is desirable because a message will be signed by an individual only once, but the signature may be verified many times. To make it faster modified Hashed-RSA Algorithm is presented as following. In practical implementation the data that is to be signed is first compressed by using unidirectional and collision-free transformation function that is called hash function. This algorithm takes SHA-256 hashed data instead of plaintext.

Step 1. Take Hash of the encrypted message $\varepsilon\mathcal{M}$, This will produce $\mathcal{EH}$

$$\mathcal{EH} = Sha\text{-}256\ (\varepsilon\mathcal{M}) \quad (1)$$

Step 2. Sign the $\mathcal{EH}$ by the Private-key (d) of the sender to produce digital signature $\mathcal{D}$

$$\mathcal{D} = (\mathcal{EH})^{d} \pmod{n} \quad (2)$$

Step 3. To verify the message, Senders Public-key (e) will be applied on $\mathcal{D}$ onto produce $\mathcal{EH}$

$$\mathcal{EH} = (\mathcal{D})^{e} \pmod{n} \quad (3)$$

IV. JOINT SIGNATURE AND HYBRID ENCRYPTION SYSTEM

Digital signatures are a recent development, the need for which has arisen with the rapid growth of digital communications. A digital signature algorithm authenticates the integrity of the signed data and identity of the signatory. Authentication in a digital signature is a process whereby the receiver of a digital message can be confident of the identity of the sender and/or the integrity of the message. Authentication protocols can be based on either conventional secret-key cryptosystems like DES or on public-key systems like RSA. Authentication in public-key systems uses digital signatures.

The security of the proposed scheme relies on the problem of solving discrete logarithms and on factorization of large numbers. So the presented system is more secure in comparison to any other joint signature and encryption schemes [12].

RSA encryption is quite slow because of large key size and modular exponentiation operations that have to be used to ensure security [7]. For the same reason, RSA's digital signature is slow as well. The length of transmitted signature equals the length of transmitted message. In other words: longer the message, the longer the digital signature. So, the proposed scheme uses Secure Hash Algorithm (SHA-256) to obtain condensed version of message, which will go as input for RSA Digital signatures algorithm [3] as shown in Figure- 1. The RSA Digital signature algorithm is defined in ANSI X9.31 [6].

First, a secret IDEA key of length 128 bits is generated. Using this key, the message is encrypted in a quick manner. Afterwards, the key itself is encrypted with the recipient's public encoding key using RSA encryption Algorithm. This key size is 1024 bit long which ensures the security [13]. The message is encrypted using the hybrid function which is the combination of IDEA and RSA algorithm.

$$\varepsilon\mathcal{M} = HybridEnc\ (\mathcal{M}) \quad (4)$$

The encrypted message $\varepsilon\mathcal{M}$ will go as input for Sha-256, which will produce 256-bit condensed version of data called Message digest [6]. After that message digest will be signed using RSA Digital signature algorithm. This digital signature ($\mathcal{D}$) will be sent along with the $\varepsilon\mathcal{M}$ and $\mathcal{E}_k$ as shown in Figure-1.

$$\mathcal{D} = RSA\text{-}Sign\ (\mathcal{EH}) \quad (5)$$

Encrypting document before signing ($St\mathcal{E}$) seems to be unnatural, but it is like putting an encrypted letter in envelope and putting seal on the envelope, which is another method of signing the document. Encrypted signed message($\mathcal{ES}$) will be transmitted to the party for whom it is intended.

$$\mathcal{ES} = \varepsilon\mathcal{M} + \mathcal{E}_k + \mathcal{D} \quad (6)$$

The verification process is just reverse of signature generation process. At the receiver end we have encrypted signed message $\mathcal{ES}$ which contains RSA encrypted IDEA key, IDEA encrypted cipher and Hashed RSA Digital signatures. Digital signature will go as input in RSA digital signature verification algorithm and encrypted message ($\varepsilon\mathcal{M}$) will go in SHA-256 algorithm which will produce encrypted message digest. The encrypted key ($\mathcal{E}_k$) will go in RSA decryption algorithm as shown in Figure-2.

$$\mathcal{EH} = RSA\text{-}Verify\ (\mathcal{D}) \quad (7)$$

After the key will be decrypted then encrypted message along with IDEA key will go as input IDEA algorithm. Now RSA Verification algorithm will match these two hashes. The signature is valid if document's hash matches the decrypted signature's hash. This can only occur when the document is the same as when the user created the digital signature and when the user is the one who created the digital signature. Only the user could create a valid signature since the user has access to their private-key [9]. This failure of matching implies that document is not authentic or a possible computation error has occurred.

V. DESIGN IMPLEMENTATIONS

The Joint signature and Hybrid Encryption system has been designed and developed using C#. Microsoft.NET has classes that extend the cryptographic services provided by Windows CryptoAPI. The *System.Security.Cryptography* namespace of .NET Framework Class Library (FCL) provides classes for Symmetric Encryption, Asymmetric Encryption, Hashing, Digital Signatures, Digital Certificates and XML Signatures [14]. For the rapid development and to demonstrate the correctness of proposed scheme, all the experimented has been done on ASCII encoded File structure.

System.Security.Cryptography.RSACryptoServiceProvider class provides methods for generating key, encryption and decryption. The key has been stored into XML format file. The key size of 1024 bit for RSA and 128 bit for IDEA algorithm is used in the development of the proposed scheme. The IDEA class has been developed in C# and it is combined to form Hybrid encryption. The RSA Digital signature algorithm utilizes the existing infrastructure of windows CryptoAPI.

VI. RESULTS

The .NET Framework provides implementations of many standard cryptographic algorithms. These algorithms are easy to use and have the safest possible default properties. In addition, the .NET Framework cryptography model of object inheritance, stream design, and configuration are extremely extensible. The System.Security.Cryptography namespace contains classes that allow you to perform both symmetric and asymmetric cryptography, create hashes, and provide random number generation. Successful cryptography is the result of combining these tasks. Here, ASCII encoding of 8-bit is assumed for experiment.

Algorithm	File Size (Bits)	Encrypted File Size(Bits)	Throughput (Mbps)
IDEA	1000	1000	3.53
RSA	1000	1024000	1.2
Proposed	1000	394216	2.8

Table 1. Throughput analysis

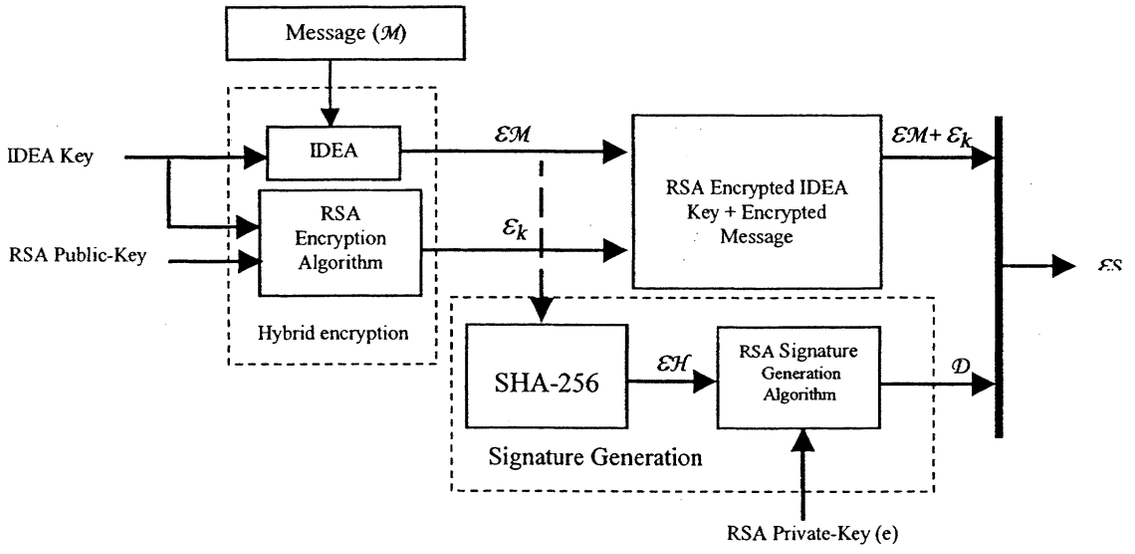

Figure 1. Joint Signature Generation and Hybrid Encryption scheme

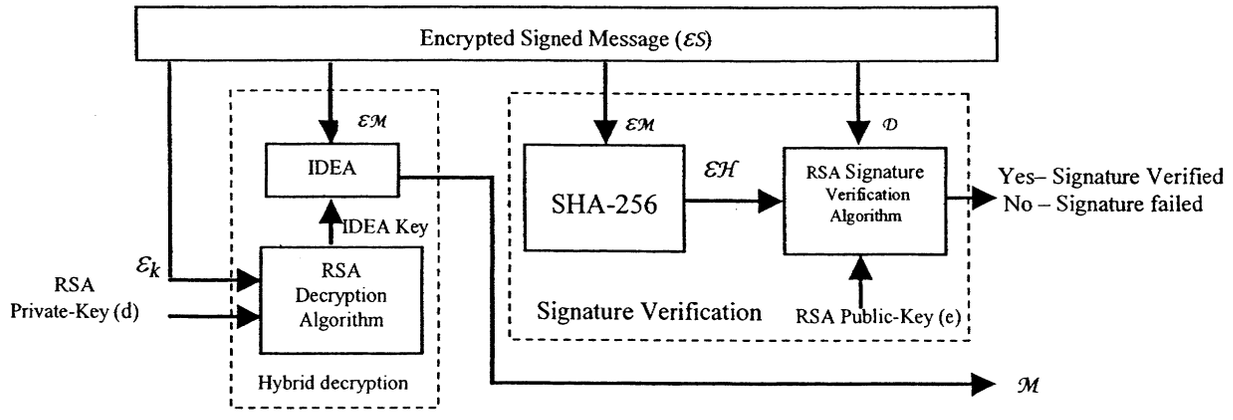

Figure 2. Joint Signature Verification and Hybrid decryption scheme

IDEA Algorithm developed at ETH Zurich has reported a speed of .88 Mbps on 33 MHz 386 machine [2]. The proposed scheme achieved a speed of 2.8 Mbps which is higher than existing implementations. Formula for calculating size of Joint Digital Signature and Hybrid Encryption is as following.

$$Total\ Size = S_{efs} + S_{ek} + S_{ds} \quad (8)$$

where
- S_{ef} = size of encrypted file
- S_{ek} = size of encrypted key
- S_{ds} = size of digital signature

Experiment was performed on 2.5 GHz machine with 256 MB RAM.

ALGORITHM VS. SIZE

The presented graph shows the size of encrypted message after applying all three algorithms. The graph shows that size of message after applying proposed Scheme is moderate.

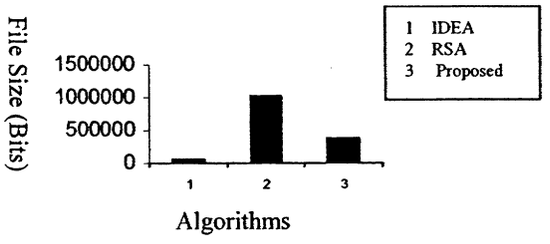

Figure 3. Algorithm Vs Size

ALGORITHM VS. THROUGHPUT

Following graph shows that throughput of various algorithms. The graph shows the proposed scheme has good throughput in contrast to RSA Algorithm, but relatively less than IDEA Algorithm.

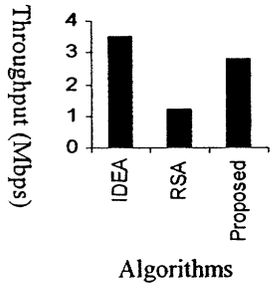

Figure 4. Algorithm Vs Throughput

VII. CONCLUSIONS

The proposed scheme is a more secure for hybrid encryption and digital signature. In particular, it is very useful paradigm for constructing signcryption scheme [17]. The signature verification process is 10-40 times slower in DSA [3], which has been removed by using RSA Digital signature algorithm along with SHA-256. We have shown that hybrid cryptographic scheme can be used for fast encryption and digital signature jointly and achieved speed of 2.8 Mbps which is faster than the existing implementations. This scheme applicable in secure internet computing[20], e-payment in distance education system[21] and as well in mobile environment, because overall computational cost is low. This scheme is also very useful for mobile devices like smart card based applications [8] and many other applications.

REFERENCES

[1] A.J. Menezes, P.C. van Oorschot and S. Vanstone, "Handbook of applied cryptography", CRC Press, 1997.

[2] R.L. Rivest, A.Shamir and L. Adleman, "A method for obtaining digital signature and public-key cryptosystem", Communication of the ACM, Vol. 21, pp. 120-126, Feb 1978

[3] Tanenbaum, Andrew S., "Computer networks", PHI, INDIA 3rd edition, pp.

[4] International Organization for standardization ISO/IEC 11770-3, Information technology-Security techniques-Key management- Part 3: Mechanism using asymmetric techniques, 1999

[5] National Institute of standards and technology (NIST), Secure Hash Standards FIPS PUB 180-2, www. itl.gov/fipspubs/

[6] David Jablon, "IEEE P1363:Standard Specification for Public-Key Cryptography" CTO Phoenix Technologies Treasurer, IEEE P1363 NIST Key Management Workshop, November 1-2, 2001

[7] Scott Campbell, "Supporting Digital Signature in Mobile Environments", Proc. of 12th IEEE international workshop on enabling technologies WETICE'03 1080-1383/03, 2003

[8] W. Stalling , "Network Security Essentials: Application and Standard", 2000

[9] M.Bellare, C.Namprempre, "Authenticated Encryption:Relations among Notions and Analysis of the Generic Composition Paradigm", In Asiacrypt'00, LNCS 1976, pp. 531-545

[10] H.Krawczyk, "The Order of Encryption and Authentication for Protecting Communications", In Crypto'01, pp. 310-331, LNCS

[11] Jee Hea An, Yevgeniy Dodis, Tal Rabin, "On the Security of Joint Signature and Encryption", In EUROCRYPT' 02 ,LNCS 2332, pp. 83-107

[12] www.rsasecurity.com/rsalabs/

[13] http://msdn.microsoft.com/library/system.security/cryptography/c/rsa/rsa.asp

[14] Yoshiaki Isobe, Yoichi Seto and Masanori Kataoka, "Development of Personal Authentication System using Fingerprint with Digital Signature Technologies", Proc. of 34th Hawaii International Conference on System Sciences", 2001

[15] R.Cramer and V.Shoup, "Design and Analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack" , SIAM Journal on Computing, 33(1):167-226, 2004

[16] V.Shoup, "Using Hash Functions as a hedge against chosen ciphertext attack", In B.Preneel, editor, *Advances in Cryptology-Eurocrypt 2000,* Volume 1807 of *Lecture Notes in Computer Science,* pp. 275-288. Springer-Verlag, 2000.

[17] Y.Zheng, "Digital signcryption or how to achieve cost (Signature & Encryption) << cost (signature) + cost (encryption)", In B.Kaliski, editor, *Advances in Cryptology- Crypto'97,* Volume 1294 of *Lecture Notes in Computer Science,* pp. 165-179. Springer-Verlag, 1997.

[18] S.Halevi and P.Rogaway, "A Parallelizable enciphering mode", In T.Okamoto, editor, *Topics in Cryptography – CT-RSA 2004,* volume.2964 of *Lecture Notes in Computer Science,* pp. 292-304. Springer-Verlag, 2004.

[19] S.Gurgens and C.Rudolph, "Security Analysis of (un-) fair non-repudiation protocols", In A. E. Abdallah, P. Ryan, and S. A. Schneider, editors, Formal Aspects of Security, volume 2629 of Lecture Notes in Computer Science, pages 97–114, London, UK, 2003. Springer-Verlag

[20] Uwe G. Wilhelm, Sebastian Staamann, and Levente ButtyBn. Introducing trusted third parties to the mobile agent paradigm. In Jan Vitek and Christian D. Jensen, editors, Secure Internet Programming, volume 1603 of Lecture Notes in Computer Science, pages 469-489. Springer, 1999

[21] Wei-Kuei Chen, "An Applicative Distance Education Payment Protocol", In Proc. of the 18th International Conference on Advanced Information Networking and Application (AINA'04), 2004