ISSN No. 0976-5697

Volume 4, No. 1, January 2013 (Special Issue)

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Generation of an Electronic Signature Suite based on Cryptography Algorithms

Prof. Satyajit S. Uparkar*, Prof. Vidhi A. Mehta
Department of Computer Application
Shri Ramdeobaba College of Engineering and Management
Nagpur, India
ssuparkar@yahoo.co.in*, idhiamehta@rediffmail.com

Ms. Shruti C. Gola
Department of Electronics Engineering
Tulsiramji Gaikwad-Patil College of Engineering and Technology Mohgaon, Nagpur, India
golarshruti11@gmail.com

Abstract: The process of cryptography basically deals with protecting the information by transforming it (*encrypting* it) into an unreadable format, called cipher text. Only those who possess a secret *key* can decipher (or *decrypt*) the message into plain text. Electronic signature is one of the applications of the Cryptography algorithms. The electronic signature suite is a collection of various components such as, hash functions key generation algorithm, signing algorithms, verification algorithms and hash functions. This paper mainly deals with the maintenance activities for implementing cryptographic hash functions like Sha-1, RIPEMD-160, and WHIRLPOOL etc. and signature algorithms like RSA, DSA ,EC-GDSA etc.

Keywords: Cryptography, Electronic Signature Suite, Hash functions, key generation algorithms.

I. INTRODUCTION

A. *Cryptography:*

Cryptography (or cryptology) is the practice and study of hiding secret information by *encryption*. **Encryption** is the conversion of data into a form, called a cipher-text. **Decryption** is the reverse, in other words, moving from the unintelligible cipher-text back to plaintext. Modern cryptography intersects the disciplines of mathematics, computer science, and electrical/electronic engineering [1]. Applications of cryptography include credit cards, laptop data security, computer passwords, computer network security solutions & software systems, and online/internet data security in electronic commerce.

Cryptographic techniques are needed for privacy and authentication of digital data. There are two types of encryption algorithms used in cryptography, namely **Symmetric-Key Encryption** (also known as symmetric-key encryption, single-key encryption, one-key encryption and private key encryption) and **Asymmetric Encryption** (Public Key Encryption).are very problematic with regards to maintaining integrity and security, as there is nothing to prevent one individual from typing another individual's name. Due to this reality, an electronic signature that does not incorporate additional measures of security (similar to a digital signature, described above) is considered an insecure way of signing documentation.

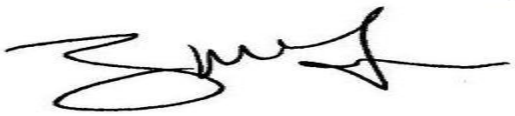

Figure. 1. Electronic Signature

Figure. 2. Electronic device capturing Electronic Signature

B. *Electronic Signature:*

The term "electronic signature" or e-signature means a method of signing an electronic message that—

(i) Identifies and authenticates a particular person as the source of the electronic message; and

(ii) Indicates such person's approval of the information contained in the electronic message.

An electronic signature is defined as an electronic sound (e.g., audio files of a person's voice), symbol (e.g., a graphic representation of a person in JPEG file), or process (e.g., a procedure that conveys assent), attached to or logically associated with a record, and executed or adopted by a person with the intent to sign the record [2]. An electronic signature is easy to implement, since something as simple as a typed name can serve as one. Consequently, e-signatures

C. *Electronic Signing Process:*

Electronic Signature transpose in the electronic world, the rich semantics of handwritten signatures. As a signature is a symbolic representation of an individual, there is a strong intertwining between electronic signatures, authentication and identification. An e-signature as data in electronic form logically associated with other electronic data and which serve as a method of authentication.. The intertwining is reinforced by the reliance of e-signature and e-identification on the same technology. There is a necessity of trust building instruments equally suitable for identification. Therefore, both are best addressed simultaneously.

The identification systems are needed to be designed in such a way that the embedded signatures should also work across borders reaping the benefits of the cross border legal recognition of e-signatures [3]. The following figure describes the Electronic Signature process.

CONFERENCE PAPER
National Level Conference on
"Trends in Advanced Computing and Information Technology (TACIT - 2012)"
on 2nd September 2012
Organized by
St. Vincent Pallotti College of Engineering and Technology, Nagpur, India

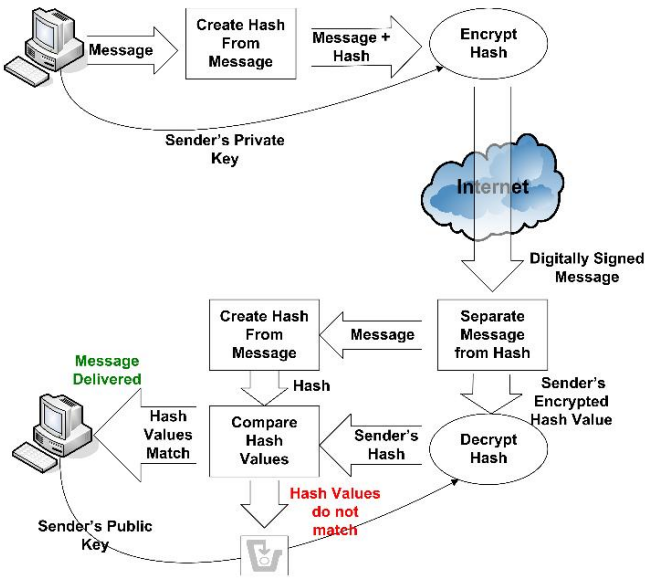

Figure. 3. Electronics signing process

II. CONCEPT OF ELECTRONIC SIGNATURE SITTE

Electronic Signature suite is here defined as consisting of the following components (H, *K*, *S*, *V*):

- A Hash Function *H*.
- A Key generation algorithm *K*;
- A Signing Algorithm S with parameters and padding method;
- A Verification algorithm *V*;

The objective and scope of this paper is to reflect the various types of Hash functions and the signing algorithms.

A. Hash Function[4]:

A Hash Function takes as input a variable-length message and produces as output a fixed-length hash value. Hash Functions may be used in a variety of cases, such as:

- Advanced Electronic Signatures include the identifier of the hash function used to compute the digital signature.
  Time- Stamp tokens include the identifier of the hash algorithm used to compute the hash value for the time-stamped data.
- Public key certificates include the identifier of a signature suite which defines the hash function used to compute the digital signature.

For the purpose of generating signatures the following (informally defined) three properties are required from the hash function h:

(a). Pre-image resistance: Given y = h(m) (but not m) it is practically infeasible to find m. Without this property, a signature scheme may otherwise be vulnerable to an attack based on generating the signature "backwards", applying the verification function to a randomly chosen signature value.

(b). 2nd pre-image resistance: Given h(m) and m, it is practically infeasible to find another m≠ m such that h(m) = h(m'). For signatures, this property protects from re-using an already existing signature for another message.

(c). Collision resistance: It is practically infeasible to find any pair of distinct values m, m' such that h(m) = h(m').

This property is obviously needed to protect signature against chosen message attacks.

a) Recommended One way Hash Functions [5]:

a. SHA-1:

SHA-1 may be used to hash a message, *M*, having a length of up to 2^{64}-1 bits. The main drawback is, several attacks against SHA-1 have been discovered. All known collision attacks on SHA-1 require full control of certain substrings within the data to be hashed and knowledge of the data bits prior to these strings. This is being considered as a realistic attack scenario for documents signed by signers (in particular, when a kind of "active" program element may be hidden in the document). On the other hand for X.509 certificates such attacks can be prevented by the CA by including a reasonable amount of entropy (i.e. data bits neither known to nor predictable by the attacker) in the certificate string prior to any data bits controllable by the attacker. This method leads to a considerably higher resistance of certificates against collision attacks.

b. RIPEMD-160:

RIPEMD-160 may be used to hash a message. RIPEMD-160 is a 160-bit cryptographic hash function, designed by Hans Dobbertin, Antoon Bosselaers, and Bart Preneel. It is replacing the 28-bit hash function RIPEMD. The maximal message size is 2^{64}-1 bits.

RIPEMD-320 is constructed from RIPEMD-160 by initializing the two parallel lines with different initial values, omitting the combination of the two lines at the end of every application of the compression function, and exchanging a chaining variable between the 2 parallel lines after each round. The security level of the 320-bit extension of RIPEMD-160 is the same as that of RIPEMD-160 itself. Similarly the 256-bit extension of RIPEMD-128, i.e. RIPEMD-256 is the same as that of RIPEMD-128.

c. WHIRLPOOL:

WHIRLPOOL is a hash function designed by Vincent Rijmen and Paulo S. L. M. Barreto that operates on messages less than 2^{256}-1 bits in length, and produces a message digest of 512 bits. Whirlpool may be used to compute the imprint of a message placed in a time-stamp token. Whirlpool may only be used with a secure signature scheme supporting key sizes that match the Whirlpool output, i.e. 512 bits. DSA and ECDSA cannot be used with Whirlpool. However, it may be used with the RSA algorithm.

The Whirlpool output, i.e. 512 bits, is more than what may be needed, but there is currently no Whirlpool algorithm variant defined by an OID/URN with an output less than 512 bits, beside the general rule to take the leftmost bits of the output. Whirlpool has been included as an alternative to the SHA-2 family and can be used either to compute a hash value (for a time-stamp token) or with the RSA algorithm.

d. SHA-224& SHA-256:

SHA-224 may be used to hash a message, *M*, having a length of up to 2^{64}-1 bits and the output size is 224 bits. The function is defined in the exact same manner as SHA-256 except for the initial value and the truncation of the final hash value.

The specification for SHA-224 is identical to SHA-256, except that different initial values are used, and the final hash value is truncated to 224 bits. Therefore it is not

CONFERENCE PAPER
National Level Conference on
"Trends in Advanced Computing and Information Technology (TACIT - 2012)"
on 2nd September 2012
Organized by
St. Vincent Pallotti College of Engineering and Technology, Nagpur, India

recommended to use SHA-224, if SHA-256 can be used instead without truncation. The final result of SHA-256 is a 256-bit message digest.

e. SHA-384 & SHA-512:

SHA-384 may be used to hash a message, *M*, having a length of up to 2^{128-1} bits and the output size is 384 bits. The function is defined in the exact same manner as SHA-512, except for the initial value and the truncation of the final hash value.

The specification for SHA-384 is identical to SHA-512, except that different initial values are used, and the final hash value is truncated to 384 bits. Therefore it is not recommended to use SHA-384, if SHA-512 can be used instead without truncation. SHA-512 may be used to hash a message, *M*; having a length of up to 2^{128}-1bits.The final result of SHA-512 is a 512-bit message digest.

B. Signature Algorithms:

A signature scheme consists of three algorithms: a key generation algorithm and a signature creation algorithm and a signature verification algorithm

a) Recommended signature algorithms:

a. RSA:

The RSA algorithm's security is based on the difficulty of factoring large integers. To generate the key pair two prime numbers, *p* and *q*, are generated randomly and independently, satisfying the following requirements:

- The bit length of the modulus $n = p\ q$ must be at least MinModLen; its length is also referred to as ModLen;
- Here *p* and *q* should have roughly the same length, e.g. set a range such as $0{,}1 < |\log 2p - \log 2q| < 30$;
- The set of primes from which p and q are (randomly and independently) selected SHALL be sufficiently large and reasonably uniformly distributed.

The private key consists of a positive integer *d* (the private exponent) and the modulus *n*. The public key consists of a positive integer *e* (the public exponent) and the modulus *n*.CRT (Chinese Remainder Theorem) [6] implementations are also allowed, in which case the private key will contain more values derived from the factorization of the modulus n. For RSA signatures also a padding method has to be specified.

b. DSA:

The DSA algorithm's [7] security is based on the difficulty of computing the discrete logarithm in the multiplicative group of a prime field *Fp*. The public parameters *p, q* and *g* may be common to a group of users.

The bit length α of the prime modulus *p* shall be at least p MinLen bits long. The bit length β of *q*, which is a prime divisor of (*p*-1), shall be at least q MinLen bits long. Only the following choices for α and β are specified:

α = 1024, β = 160;
α = 2048, β = 224;
α = 2048, β = 256;
α = 3072, β = 256.

The value of β determines the defined in hash function to be used. This requires for β = 160 the function SHA-1, which should not be used for new applications.SHA-224 does not provide security advantages over SHA-256. If it is not required by a signature length restriction, since a signature with β =224 occupies 448 bits whereas a signature with β =256 needs 512 bits, it is recommended to use parameters with β =256.

The private key consists of:

(a). The public parameters *p, q* and *g*;
(b). A statistically unique and unpredictable integer *x*, 0 $< x < q$, which is signatory-specific; and
(c). A statistically unique and unpredictable integer *k*, 0 $< k < q$, which must be regenerated for each signature.

If the distribution of *k* is significantly different from uniform within the interval then there may be weaknesses. Bleichenbacher has presented an attack which can be sub-exhaustive depending on the size of the bias and the number of signatures produced using a single secret key.

The value of *k* must be kept secret as well as the private key, even if *k* is only partially known there exists an attack (Nguyen/Shparlinski).The public key consists of *p, q, g* and an integer *y* computed as $y = gx \bmod p$. When computing a signature of a message M, no padding of the hash-code is necessary.

c. Elliptic curve analogue of DSA based on a group E (F_p):

This signature algorithm is referred to as ecdsa-F_p. The security of the ecdsa-F_p algorithm is based on the difficulty of computing the elliptic curve discrete logarithm [8].

The public parameters are as follows:

(a). *p* prime;
(b). *q* large prime at least qMinLen bits long, $p \neq q$;
(c). *E* elliptic curve over a finite field F_p whose order *n* is divisible by *q*; and
(d). *p* point on $E(F_p)$ of order *q*.

The public parameters may be common to a group of users. The quotient *h* of the group order *n* divided by *q* may be considered as a public parameter too. The class number of the maximal order of the endomorphism ring of *E* shall be at least MinClass = 200.

The value *r*0: = min (*r*: *q* divides *pr*-1) shall be greater than r0Min=104.*h* = *n*/*q* must be less or equal 4. The ECC Brainpool paper on standard curves and curve generation contains an alternative set of curves over prime fields with 160 192, 224, 256, 320, 384 and 512 bits. All these curves fulfill the above requirements.

The private key consists of:

(a). The public parameters *E, m, q* and *P*;
(b). A statistically unique and unpredictable integer *x*, 0 $< x < q$, which is signatory-specific; and
(c). A statistically unique and unpredictable integer *k*, 0 $< k < q$, which must be regenerated for each signature.

The public key consists of *E, q, P* and *Q*, a point of *E*, which is computed as *Q* = *xP*.

d. Elliptic curve analogue of DSA based on a group E(F_2m) :

This signature algorithm is referred to as ecdsa-F_2m **[9]**. The security of the ecdsa-F_2m algorithm is based on the difficulty of computing the elliptic curve discrete logarithm. The public parameters are as follows:

(a). *m* prime number;
(b). *q* large prime at least q MinLen bits long;
(c). *E* elliptic curve over a finite field F_2m whose order *n* is divisible by *q*;
(d). It must not be possible to define *E* over *F*2; and
(e). *P* point on *E* (F_2m) of order *q*.

h = *n*/*q* must be less or equal 4.The class number of the maximal order of the endomorphism ring of *E* shall be at

CONFERENCE PAPER
National Level Conference on
"Trends in Advanced Computing and Information Technology (TACIT - 2012)"
on 2nd September 2012
Organized by
St. Vincent Pallotti College of Engineering and Technology, Nagpur, India

least MinClass = 200. The value $r0:=\min(r: q \text{ divides } 2^{mr}-1)$ shall be greater than r0Min = 104.

A field representation is required, common to both the signatory and the verifier, so that signatures can be interpreted correctly. Thus if a polynomial basis is required then an irreducible trinomial of the form $x^m + x^a + 1$ with minimal a should be used. If such a polynomial does not exist then an irreducible pentanomial of the form $x^m + x^a + x^b + x^c + 1$ should be used; a should be minimal, b should be minimal given a and c should be minimal given a and b.

The private key consists of:

(a). The public parameters E, m, q and P;

(b). A statistically unique and unpredictable integer x, $0 < x < q$, which is signatory-specific;

(c). A statistically unique and unpredictable integer k, $0 < k < q$, which must be regenerated for each signature.

The public key consists of E, q, P and Q, a point of E which is computed as $Q=xP$.

e. EC-GDSA based on a group E (Fp):

This signature algorithm is referred to as ECGDSA-Fp. The ECGDSA-Fp algorithm is a variant of the ECDSA-Fp algorithm with a modified signature creation equation and verification method. The parameters are the same as for ECDSA-Fp. The basic difference between ECDSA and EC-GDSA is that during signature creation k does not need to be inverted for ECGDSA.

III. CONCLUSION

To meet this security requirement and to allow signing of more or less arbitrary long messages, a signature suite requires a hash function, so that the signing/verification algorithms operate on a fixed-size hash of the message. An important issue is to tie the hash function to the signature scheme [10]. Without this, the weakest available hash function could define the overall security level.

Due to possible interactions which may influence security of electronic signatures, algorithms and parameters for secure electronic signatures can be used only in predefined combinations referred to as the signature suites. A signature suite consists of the following components:

(a). a hash function;

(b). a signature algorithm and its associated parameters.

(c). a padding method;

If any of the components of a suite is modified, then the suite must be modified accordingly.

IV. FUTURE SCOPE

The paper has discuss the various hash functions and the signing algorithms. The other associated parameters and the combinations effect of key generation method can be studied. The final part to complete the suite is to develop the strategies for padding methods. The effect of time parameter and the length of the message can also be combining verified for generating an efficient electronic signature suite.

V. REFERENCES

[1] —Robust Encryption‖ , by M. Abdalla, M. Bellare and G. Neven in Proceedings of the 7th Theory of Cryptography Conference (TCC 2010), Vol. 5978, D. Micciancio ed, Springer-Verlag, 2010.

[2] APPLIED CRYPTOGRAPHY AND DATA SECURITY, Prof. Christof Paar (version 2.5 — January 2005)

[3] —A Signature System Based on Trust Computing‖ by Chi Ya Ping , Li Zhi Peng ; Wei Zhan Zhen ; Fang Yong ,in International Conference on —Computational Intelligence and Software Engineering (CiSE), 10-12 Dec. 2010 .

[4] An Improved Scheme for E-signature Techniques Based on Digital Encryption and Information Hiding Huang Tao; Zhou Qihai; Zhang Le; Li Zhongjun; Lin Xun in International Symposiums on Digital Object Identifier: 10.1109/ISIP.2008.47 Publication Year: 2008 , Page(s): 593 – 597.

[5] —Software adds e-signature capability." , by Fleet Owner in General One File. Web. 1 Oct. 2011.

[6] —Asymmetric cryptography algorithm with Chinese remainder theorem‖, by Zhang Yun -peng; Lin Xia; Wang Qiang in IEEE 3rd International Conference on Communication Software and Networks (ICCSN), 2011, Digital Object. Identifier: 10. 1109/ ICCSN. 2011. 6014606 Publication Year: 2011, Page(s): 450 – 454.

[7] —Integrating Diffie-Hellman key exchange into the digital signature algorithm (DSA)‖, by Harn, L.; Mehta, M .; Wen-Jung Hsin in Communications Letters, IEEE Volume:8, Issue:3 Digital Object Identifier: 10.1109/ LCOMM.2004. 825705 Publication Year: 2004, Page(s): 198 – 200.

[8] "A Unified Approach to the Discrete Logarithm Problem for the Multiplicative Group and for Elliptic Curves over finite Fields". September 20, 2004.

[9] ‖The Key Exchange Research of Chaotic Encryption Chip Based on Elliptic curve Cryptography Algorithm‖, by Ping Zhou; Qun Ding in Second International Conference on Intelligent Computation Technology and Automation, 2009. ICICTA '09. Volume: 4 Digital Object Identifier: 10.1109/ ICICTA. 2009.880 Publication Years: 2009, Page (s): 689-693

[10] "The New United States Uniform Electronic Transactions Act: Substantive Provisions, Drafting History and Comparison to the UNCITRAL Model Law on Electronic Commerce". by Gabriel, Henry,.

CONFERENCE PAPER
National Level Conference on
"Trends in Advanced Computing and Information Technology (TACIT - 2012)"
on 2nd September 2012
Organized by
St. Vincent Pallotti College of Engineering and Technology, Nagpur, India