Electronic Seal Stamping Based on Group Signature

Girija Srikanth[1]

[1] **Department of CSE, Birla Institute of Technology,**
Al Dhait south, Ras Al Kaimah, UAE

Abstract

This paper describes a new electronic official seal stamping based on Group Signature, USB Key. Bill/Contract in E-commerce must be seal stamped to gain tamper proof and non-repudiation. The seal stamping control is designed based on the certificate-based public key. This technique is more efficient for generating and verifying individual/group signatures in terms of computational efforts and communication costs. Web page electronic seal-stamping system is implemented which has been adopted by CNBAB platform since Mar., 2008.

Keywords: *Digital Signature, Self certified public key, Seal Stamp, USB key*

1. Introduction

CNBAB [1][2][3] is an e-commerce platform which constructs a credit worthy trade environment and provides financing channels for Chinese small and medium-sized enterprise (SME) and even small enterprises. CNBAB has carried an operation in Shandong province and achieved a great success. CNBAB adopts a brand new business pattern called BAB (Business agent business) [4]. Business Agent is an agent who handles business affairs for another, especially one who deals with employers. An agent is a representative who acts on behalf of other persons/organizations. BAB pattern can provide credit guarantee and solve quickly transactional fund storage for SMEs. CNBAB constructs an aggregate called agent to guarantee reliable trading environment by combining banks, the government, the digital authentication centre and third party quality supervision institutions, in which every party undertakes different responsibility throughout the entire trading process. There are three kinds of users and eighteen kinds of agent staffs in CNBAB.

Enterprise user or individual user can become register by registering in CNBAB. Register user can apply for becoming contracted user. Contracted user can trade in CNBAB platform. Contracted user can apply for becoming core user by submitting appointed materials to CNBAB and banks. If these materials are materials are audited to pass, contracted user can become core user. CNBAB launches trade-currency service similar to short term loans for core user to resolve financing problem. Trade currency guarantees trade between users, banks guarantees the value of trade-currency to assure smooth trade steps.

After achieving a transaction between users, both sides need to sign a contract. And in sequence every stage of trade process, users need to fill in some bills and agent staffs need to audit these bills; some agent staffs need to fill in some bills and other agent staffs need to audit these bills. There are twenty-six kinds of contract templates and sixteen kinds of bills in CNBAB. Bill/contract must be seal-stamped to gain tamperproof and non-repudiation.

Seal-stamping on web page is a method that allows a person to 'seal' documents in a manner parallel to the traditional seal. Seal-stamping on web page can be regarded as electrification of the traditional seal and the handwritten signature. Combining the digital signature with the seal image prevent bill/contract from altering and denying.

2. Literature Survey

2.1 Web page seal-stamping and verify

CNBAB is developed in java language based on IBM Rational Application Developer IDE using JSF web framework and hibernate Middleware, adopts Oracle 10g release 2 as DBMS and IBM Web sphere as application server. CNBAB has 600,000 lines code approximately.

Seal-stamping control based on proposed digital signature scheme using self-certified public key is an ActiveX control on client which is available in IE browser. It is developed in C++ language based on Microsoft Visual Studio 2008 IDE. It provides JavaScript interface functions, the most important two functions are sign and verify. Internal specific cryptography operations of the two functions are described in "Signature generation and verification" section. Sign function executes seal-stamping operation. Verify function verifies the validity of public

key and signature, but the verification of the public key is accomplished within the signature verification Procedure.

As compared with seal-stamping control designed based on the certificate-based public key [5][6], this control is more efficient for generating and verifying signatures in terms of computational efforts and communication costs.

After Users/agent staff logins CNBAB, there appears a web page include many menu items according to their individual rights. And there is a session bean storing user information, including key-information. There is a table recording username and corresponding public key in database.

When user views a contract, user/agent staff views a bill by
clicking a menu item on web page, corresponding functional page is opened. According to the status of bill/contract and the privileges of user/agent staff, web page backend business logic judges whether there is a seal-stamping button on the page. If web page contains seal-stampings, control will verify the validity of every signature, then valid seal image is showed if passing verify, otherwise invalid seal image.

2.2 Web page Seal-Stamping

There is a processing step on server side before corresponding functional page is opened. Entire bill/contract page's html data is converted into XML data, stored as a property of page bean.

If bill/contract need to be seal-stamped by user/agent staff, a seal-stamping control and a seal-stamping button are inserted in the right position of the page. User/agent staff can trigger the seal-stamping button. After this button being triggered, control executes following steps accomplished on client.

(1) Examines whether there is a valid USB key on computer USB interface. If yes, require user /agent staff input USB key PIN; if no, prompt user to insert USB key.

(2) Examines whether this USB key is owned by login person according to public key information.

(3) Reads seal image in USB key, then sign organized XML data (mentioned at the beginning of this section) using private key in USB key, then seal image is inserted into the web page and floats above the web page automatically. Signature data include the signature value of organized XML data, seal image and public key. At the same time, signature data is assigned to a hidden html element in bill/contract web page whose value is corresponding to a property of page bean. The maximum size of signature data is 15K, commonly 4K.

Thus, seal-stamping finished. After saving bill/contract, organized XML data and signature data is saved into database and the status of bill/contract is updated. When agent staff needs to audit bill, only if all seal image is valid, there is a seal-stamping button in the right position of page.

2.3 Verify

When user/agent staff views seal-stamped web page, all seal-stamping controls execute verify operation. There is a processing step on server side before corresponding functional page is opened. Data before signature and after signature for every seal-stamping must be retrieved from database to verify the validity of signature, stored as two property of page bean. According to CNBAB SRS [7], at most there are three seal stamps in a web page, commonly two.

If signature passes verify, controls in web page show valid Seal image, otherwise invalid seal image.

2.4 Digital Signature

A digital signature or digital signature scheme is a mathematical scheme for demonstrating the authenticity of a digital message or document. A valid digital signature gives a recipient reason to believe that the message was created by a known sender, and that it was not altered in transit. Digital signatures are commonly used for software distribution, financial transactions, and in other cases where it is important to detect forgery or tampering.

Digital signatures are often used to implement electronic signatures, a broader term that refers to any electronic data that carries the intent of a signature, but not all electronic signatures use digital signatures. In some countries, including the United States, India, and members of the European Union, electronic signatures have legal significance. However, laws concerning electronic signatures do not always make clear whether they are digital cryptographic signatures in the sense used here, leaving the legal definition, and so their importance, somewhat confused.

Digital signatures employ a type of asymmetric cryptography. For messages sent through a nonsecure channel, a properly implemented digital signature gives the receiver reason to believe the message was sent by the claimed sender. Digital signatures are equivalent to traditional handwritten signatures in many respects; properly implemented digital signatures are more difficult to forge than the handwritten type. Digital signature schemes in the sense used here are cryptographically based, and must be implemented properly to be effective. Digital signatures can also provide non-repudiation, meaning that the signer cannot successfully claim they did not sign a message, while also claiming their private key remains secret; further, some non-repudiation schemes offer a time stamp for the digital signature, so that even if the private key is exposed, the signature is valid nonetheless. Digitally signed messages may be anything representable as a bit string: examples include electronic mail, contracts, or a message sent via some other cryptographic protocol.

As organizations move away from paper documents with ink signatures or authenticity stamps, digital signatures can provide added assurances of the evidence to provenance, identity, and status of an electronic document as well as acknowledging informed consent and approval by a signatory. The United States Government Printing Office (GPO) publishes electronic versions of the budget, public and private laws, and congressional bills with digital signatures. Universities including Penn State, University of Chicago, and Stanford are publishing electronic student transcripts with digital signatures.

Below are some common reasons for applying a digital signature to communications:

Authentication: Although messages may often include information about the entity sending a message, that information may not be accurate. Digital signatures can be used to authenticate the source of messages. When ownership of a digital signature secret key is bound to a specific user, a valid signature shows that the message was sent by that user. The importance of high confidence in sender authenticity is especially obvious in a financial context. For example, suppose a bank's branch office sends instructions to the central office requesting a change in the balance of an account. If the central office is not convinced that such a message is truly sent from an authorized source, acting on such a request could be a grave mistake.

Integrity: In many scenarios, the sender and receiver of a message may have a need for confidence that the message has not been altered during transmission. Although encryption hides the contents of a message, it may be possible to change an encrypted message without understanding it. (Some encryption algorithms, known as nonmalleable ones, prevent this, but others do not.) However, if a message is digitally signed, any change in the message after signature will invalidate the signature. Furthermore, there is no efficient way to modify a message and its signature to produce a new message with a valid signature, because this is still considered to be computationally infeasible by most cryptographic hash functions

Non-repudiation: Non-repudiation, or more specifically non-repudiation of origin, is an important aspect of digital signatures. By this property an entity that has signed some information cannot at a later time deny having signed it. Similarly, access to the public key only does not enable a fraudulent party to fake a valid signature.

2.5 Group Signature

Based on digital signature scheme, we develop an ActiveX control on client to accomplish seal-stamping and verify. As compared with seal-stamping control designed based on the certificate-based public key [8][9], this control is more efficient for generating and verifying signatures in terms of computational efforts and communication costs. Further, we propose an electronic seal stamping based on Group signature which overcomes the disadvantages and retains all merits of the original scheme.

Group signatures allow individual members to make signatures on behalf of the group while providing, all previously proposed schemes are not very efficient and are also not to secure.

Group signatures allow individual members to make signatures on behalf of the group. Group oriented signature is a method to distribute the ability to sign among a set of users in such a way that only certain subsets of a group of users can collaborate to produce a valid signature on any given message. A group signature scheme has the following three properties

(1) Only legal member of the group can sign messages.
(2) The receiver can verify that it is indeed a valid group signature, but cannot discover which group member made it.

(3) In the case of a later dispute, the signer can be identified by either the group members together or a group authority.

Group signature scheme with signature claiming and variable linkability is a digital signature scheme with three types of participants: A group manager, an open authority, and group members. It consists of the following procedures:

- Setup: For a given security parameters, the group manager produce system-wide public parameters and a group manager master key for group membership certificate generation.

- Join: An interactive protocol between a user and the group manager. The user obtains a group membership certificate to become a group member. The public certificate and the user's identity information are stored by the group manager in a database for future use.

- Sign: Using his group membership certificate and his private key, a group member creates an anonymous group signature for a message.

- Verify: A signature is verified to make sure it originates from a legitimate group member without the knowledge of which particular one.

- Open: Given a valid signature, an open authority discloses the underlying group membership certificate.

- Claim (Self-trace): A group member creates a proof that he created a particular signature.

- Claim Verify: A party verifies the correctness of the claiming transcript. Similar to a group signature, our signature scheme should satisfy the following properties:

- Correctness: Any valid signature can be correctly verified by the Verify protocol and a valid claiming proof can be correctly verified.

- Forgery-Resistance: A valid group membership certificate can only be created by a user and the group manger through Join protocol.

- Anonymity: It is infeasible to identify the real signer of a signature except by the open authority or if the signature has been claimed.

- Unlinkability: It is infeasible to link two different signatures of the same group member.

- Non-framing: No one (including the group manager) can sign a message in such a way that it appears to come from another user if it is opened.
- Non-appropriation: No one (including the group manager) can make a valid claim for signature which they did not create.

3. Proposed Signature Scheme using Self-Certified Public keys

3.1 System Model

In the system environments, there exists a DUC (Digital Authentication Centre). The responsibilities of digital authentication centre are to generate the system parameters and to issue users' public keys. Stages of the proposed signature scheme include the system setup, the registration, the signature generation and verification.

In the system setup stage, digital authentication centre generates system parameters, including digital authentication centre's private key and public key pair. In the registration stage, digital authentication centre deals with the registration requests submitted by a registering user for issuing self certified public keys. After that, digital authentication centre publishes all self-certified public keys and sends each user a witness. Note that digital authentication centre does not need to generate any certificates for these public keys. With the received witness and the secret shadow, each user can solely compute his private key.

Moreover, each user could directly verify the validity of his self-certified public key with his private key, which demands on any additional public key Certificate. It should be assured that digital authentication centre does not have any useful knowledge of any user's private key. Note that the validity of signature and the authenticity of the signer have self-certified public key can be simultaneously verified in the signature verification.

3.2 Realization of the Proposed Scheme

Following the system model as mentioned in the previous section, we propose a signature scheme using self-certified public keys in this section. The system setup, the registration, the signature generation and verification are described below in detail.

3.2.1 System Setup

Initially, digital authentication centre chooses a one-way hash function h, a large primes p such that $p-1$ has also a large prime factor (e.g. $(p-1)/2$) and a generator g of Z^*_p.

Then digital authentication centre randomly selects an integer a ($a \in [1, p-2]$) and computes

$$b = g^a \bmod p \quad (1)$$

The parameters b, g, p are published by digital authentication centre while a is kept secret.

3.2.2 Registration

When a user U_i with identity ID_i wants to join the system, the procedure for generating self-certified private-key/public-key pair is described below.

Step 1: U_i chooses a random integer j in Z^*_p ($j \in [1, p-2]$), j is co-prime with $p-1$, computes

$$u = g^j \bmod p \quad (2)$$

and U_i sends $\{ID_i, u\}$ to digital authentication center for registration. Then he proves to digital authentication center that she knows j without revealing it by using an interactive zero knowledge proof.

Step 2: Upon receiving $\{ID, u_i\}$ digital authentication center selects a random integer k, computes the public key for U_i as

$$P_i = u^k \bmod p \quad (3)$$

and solves x in the equation using extended Euclidean algorithm

$$aP_i + kx = ID_i \bmod (p-1) \quad (4)$$

Step 3: Digital authentication centre returns (P_i, ID_i, x) to U_i, who calculates:

$$s_i = xj^{-1} \bmod (p-1) \quad (5)$$

So that,

$$b^{Pi} P^{si}_i = g^{Idi} \bmod p \quad (6)$$

U's secret key is s_i and self-certified public key is P_i. U_i computes solely his private key, so level 3 [10] is reached. U_i can check the validity of P_i by verifying (6). The correctness of the verification for the self-certified public key is shown through the following theorems.

Theorem 1: The self-certified public key P_i is valid provided that (6) holds.

Proof: Substituting ID_i with (4), we can rewrite (6) as

$$b^{Pi} P^{si}_i = g^{(aPi+kx)} \bmod p \quad (7)$$

combining (5), (1), (2), (3), we can infer (6).

If U_i wants to prove his identity to some verifier, he can perform the following procedure:

Step 1: U_i sends $\{ID_i, P_i\}$ to the verifier, who computes

$$v_i\ b^{-Pi}\ g^{ID\,i} p \bmod p \quad (8)$$

Step 2: U_i selects a random integer r_i in Z^*_p computes

$$t_i = P^{ri}_i \bmod p \quad (9)$$

and sends t_i to the verifier.

Step 3: The verifier randomly selects an integer k in Z^*_p and sends it to U_i.

Step 4: U_i computes

$$x_i = r_i + s_i k \quad (10)$$

Step 5: The verifier checks the following verification equation:

$$P^{xi}_i = t_i\ v_i^k (\bmod p) \quad (11)$$

If it holds, then the verifier accepts the validity of the identity of U_i, otherwise rejects the identity claimed by U_i.

Note that no additional certificate is required when verifying the validity of the identity of U_i, since P_i is self-certified. Except for U_i, another user cannot infer s_i from P_i and all available public information, under the cryptographic assumptions that the discrete logarithm problems are hard [5].

Also note that digital authentication centre might impersonate U_i by randomly choosing a random integer j', computing public key P_i' and private key s_i' by (2), (3), (4), and (5). The forged public key P_i' *will* pass the verification check in. (6).

However, the existence of two valid public keys linked to U_i gives the proof that digital authentication centre is dishonest.

4. Signature Generation & Verification

4.1 Signature Generation

Let M be the signing message. To generate the signature for M , each user U_i performs the following procedure:
U_i first chooses an random integer w_i in Z^*_p and then computes the signature for M , i.e., (r_i, x_i) where

$$r_i = P_i^{wi} \bmod p \quad (12)$$

$$x_i = w_i + s_i h(M, r_i) \quad (13)$$

4.2 Signature Verification

Upon receiving M and its signature (r_i, x_i), the verifier checks the following signature verification equation:

$$P_i^{xi} = r_i (b^{-Pi} \; g^{IDi})^{h(M, ri)} \pmod p \quad (14)$$

If it holds, then the verifier accepts the validity of the signature, otherwise rejects the signature

Theorem 2: If (13) holds, then the signature of M is verified, and meanwhile, the public key of U_i is authenticated.
Proof: Raising both sides of (12) to exponents with the base P_i yields

$$P_i^{xi} = P_i^{wi} \cdot P_i^{si\,h(M,ri)} \pmod p \quad (15)$$

Thus, (r_i, x_i) are verified if P_i is authenticated

4.3 Group Signature Generation & verification

If all individual signatures are verified, then CLK computes

$$R = \prod_{i=1}^{t} r_i^{r_i} \bmod p \quad (16)$$

$$S = \sum_{i=1}^{t} si \bmod q \quad (17)$$

Thus (R, S) is the group signature of M with respect to G. To verify the group signature, any verifier checks the following equality:

$$g^s = R^{h(m\|R)} \; ((Y_G + h(GID)) \; \beta^{h(Y_G \| GID)})^R \bmod p \quad (18)$$

If it holds, then (R, S) is a valid group signature of M signed by G with the self certified public key Y_G [11], [12], [13].

5. USB Key

USB Key is a smart hardware of USB interface within CPU, memory and chip operating systems (COS) inside. It is used to store user's self certified private key/ public key pair and watermarked seal image. The procedure for generating self certified private-key/public-key pair is described in "REGISTRATION". User/Agent staff seal is scanned into computer to seal image. After Hollow processing, semitransparent processing, Gray Processing, Seal image is returned into USB key at the same time seal image is watermarked using User's private key. Inside USB key, there are algorithms to verify private-key/public-key pair and watermark seal image.

Each USB key has PIN protection [14]. Since PIN is input on the computer, then the attacker may get PIN by program. If the user does not take USB key in time, the attacker may pass the fake authentication through having gotten PIN. So there is dynamic password algorithm inside USB key to work out frequently changed, unpredictable and one time valid password, so that PIN may be produced dynamically. Even if the attacker can get the last PIN, it has been already disposable. Time stamp can be implemented with the USB key. This can be considered as future work.

6. Conclusions

In this paper, we present a group signature scheme using self certified key. Electronic commerce, commonly known as e-commerce or ecommerce, consists of the buying and selling of products or services over electronic systems such as the Internet and other computer networks. Bill/Contract in E-commerce must be seal stamped to gain tamper proof and non-repudiation. Non-repudiation refers to a state of affairs where the purported maker of a statement will not be able to successfully challenge the validity of the statement or contract. The term is often seen in a legal setting wherein the authenticity of a signature is being challenged. In such an instance the authenticity is being "repudiated".The seal stamping control is designed based on the certificate-based public key. This technique is more efficient for generating and verifying individual/group signatures in terms of computational efforts and communication costs. The security of the proposed scheme is based on the hash function.

References

[1] Shijin Yaun, Bin Mu and Xianing Zhang, "Implementation for electronic seal-stamping using self certified public key in e-commerce", Internet technology and application, wuhan, 20-22, Aug, 2010.

[2] http://www.cnbab.com/, last visited on 28th, April. 2011.

[3] Bin Mu, Shijin Yaun, "software analyze and design for resources operations and its supporting technologies project", unpublished

[4] Girault M, " self –certified public keys", In Advances in cryptology- EUROCRPYT'91, springer-verlag, Berlin, pp 491-497,1991.

[5] Shahrokh Saeednia, "A short note on Girault's self certified model", http: //eprint.iacr.org/2001/100.ps.gz., 2001.

[6] Tzong-Sun Wu, Chien-Lung Hsu, "Threshold signature scheme using sel certified public keys", The journal of systems, Vol.67, pp.89-97, 2003

[7] Li Guo, zang Jinmei, "Realization of electronic official seal system based on WORD", Proceedings-International conference on Networks Security, Wireless Communications and Trusted Computing, NSWCTC 2009,v 1, p 501-504, 2009

[8] http://www.bjca.org.cn/ last visited on 28th, Mar., 2011

[9] Cheng Zhen-bo, Xiao Gang Zhang Fei," Design and Implementaion on digitial stamp system for public document", Journal of Zheijiang University of Technology, Volume 36 issue 5, 2008

[10] Shi-yuan Zheng; Jun Liu "An USB-Key_based approach for software tamper resistance", Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on 20-22 Aug. 2010.

[11] Ueda, K. Mutoh, T. Matsuo, K. Dept. of Inf. & Computer. Eng., Nara Nat. Coll. of Technol. "Automatic verification system for seal imprints on Japanese bankchecks ", Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on IssueDate: 16-20, Aug, 1998; Volume: 1, On page(s): 629 - 632 vol.1

[12] Jianhong Zhang, Qin Geng;North China University of Technology;Beijing." On the Security of a Group Signature Scheme", networking, Sensing & Control, 2008. ICNSC 2008. IEEE International Conference,6-8 April 2008, Pages:1310-1314

[13] Popescu, C.; Noje, D.; Bede, B.; Mang, I.; Dept. of Math., Univ. of Oradea, Romania ,"A group signature scheme with revocation" Video/Image Processing and Multimedia Communications, 2003. 4th EURASIP Conference, Issue Date: 2-5 July 2003 On page(s): 245 - 250 Vol.1

[14] Park, , Haeryong; Kim, Hyun; Chun, Kilsoo; Lee, Jaeil; Lim, Seongan; Yie, Ikkwon; Cryptography Technol. Team, Korea Inf. Security Agency, Seoul , " Untraceability of Group Signature Schemes based on Bilinear Mapping and Their Improvement" Information Technology, 2007. ITNG '07. Fourth International Conference on Issue Date: 2-4April2007, on page(s): 747 - 753

Girija Srikanth has completed her BE Degree in Electronics & Communication Engineering [2005], M.Tech degree in Computer Science & Engineering with the specialization of Information Security [2008] from Pondicherry Engineering College. Currently she is working as Lecturer, Department of Computer Science & Engineering, Birla Institute of Technology, Ras-Al-Kaimah, Dubai. She presented two papers in National Conference and participated in an International Conference [IACITS]. Her research interests include Cryptography, Image Processing, Steganography, Network Security and Web Security.