A Blind Signature Scheme Based On ElGamal Signature

Elsayed Mohammed, A. E. Emarah and Kh. El-Shennawy, Senior Member, IEEE
Arab Academy for Science and Technology, Military Technical College
Email: elsamoha@hotmail.com

Abstract **This paper presents a new generalized blind signature scheme based on ElGamal signatures. This new scheme has a valuable property that assures that if a message is signed multiple times the corresponding signatures will be different. This adds to the anonymity of the blinded signatures. Public Key Cryptosystems are beneficial in encryption as well as signing which plays an essential role in electronic banking and financial transactions. The current research introduces a generalized signature scheme that could be used to generate blinded signatures as well as ordinary ElGamal signatures. The new scheme is found to be comparable to the RSA blinding. Moreover, the new scheme has the advantage of having less computational complexity and is faster than RSA in the blinding procedure.**

I. INTRODUCTION

Recently the use of public key cryptosystems to provide secure network communication has received considerable attention because of their usage in e-commerce. The digital signature of a person uniquely identifies that person in a transaction. On the other hand this identity verification reveals the person's identity when engaging in any activity. In order to obtain a service or make a purchase (using something other than cash), organization require that you identify yourself. This compromises the privacy of persons participating in these transactions. While advances in information and communications technology have enhanced the ability of organizations to keep massive amounts of personal data, this has increasingly jeopardized the privacy of those whose information is being collected. Minimizing the identifying data would restore privacy considerably, but would still permit the collection of needed information.

Blind signature technologies have been introduced to provide a way of conducting electronic transactions anonymously but securely.

Mathematical Foundation:

In 1976, Diffie and Hellman introduced the concept of public key cryptography. First, the Diffie-Hellman key distribution protocol is reviewed. Suppose that A and B want to share a secret K_{AB} where A has a secret x_A and B has a secret x_B. Let p be a large prime and α be a primitive element mod p, which are both known.

A computes:

$$y_A = \alpha^{x_A} \bmod p$$

and sends y_A.

Similarly, B computes:

$$y_B = \alpha^{x_B} \bmod p$$

and sends y_B.

Then the secret K_{AB} is computed as:

$$K_{AB} = \alpha^{x_A x_B} \bmod p$$
$$= y_B^{x_A} \bmod p$$
$$= y_A^{x_B} \bmod p$$

Therefore both A and B are able to compute K_{AB}. But, for an intruder, computing K_{AB} appears to be difficult. The intruder will need to compute discrete logarithms modulo a prime, which is known to be a difficult problem [1].
In all cryptographic systems based on discrete logarithms, p must be chosen such that $p-1$ has at least one large prime factor. If $p-1$ has only small prime factors, then computing discrete logarithms is easy [3].

Section II introduces the blinding concept and uses. Blinding using RSA is explained in section III. ElGamal signature scheme is explained in section IV. Section V shows the new blinding scheme.

II. BLINDING

A blind signature is a special form of digital signature. Just as in any digital signature scheme, only signers can create blind signatures using their private keys, while anyone can verify a blind signature using the public key of the signers. Unlike a normal digital signature scheme, however, the signer does not learn which messages he is actually signing. Moreover, the signer does not know which blind signatures he is actually creating.

Creating a blind signature for a message involves two parties, which we call the signer and the receiver. The receiver only needs to know the public key, while the signer knows both the private key and the public key. An important example of a blind signature scheme is David Chaum's Blind Signature Protocol for the RSA cryptosystem, which is explained in the next section.
It is interesting to know that ecash coins are, in fact, blind signatures. In this case, ecash issuers play the role of signers, while ecash users play the role of receivers. Each time an issuer executes the blind signature protocol with a user, the user receives an ecash coin. The properties of the blind signature protocol are used to assure that users are not able to create ecash coins on their own and that each execution of the blind signature protocol will give rise to the creation of at most one ecash coin. If it were not so, then the system could not be secure.
The implications of the blind signature method are that, on the one hand, ecash issuers are assured that ecash coins are not forged, while, on the other hand, ecash users are assured that ecash coins are not traced. For this reason, when ecash coins are used in anonymous purchases, an ecash shop, possibly in co-operation with the ecash issuer, is able to verify the authenticity of the coins but is unable to identify the origin of the coins.

III. BLINDING USING RSA DIGITAL SIGNATURES

A. *RSA Signatures:*

Assume a standard RSA setting, in which the public key is denoted as a pair (e,n) and the private key is denoted as a number d. Here, the modulus n is a product of two large (secret) primes p, q and the private key d is the multiplicative

0-7803-6323-X/00/$10.00 (c) 2000 IEEE

inverse of e modulo $(p-1)(q-1)$. For the security of the RSA system it is assumed that both p and q are sufficiently large (e.g., 200 digit numbers), such that it is infeasible to either find the factorization of n or to find the private key d, given only the public key (e,n).

Let a message m be given for which an RSA signature is to be produced. m corresponds to an integer between 0 and n. The signature is produced in one step by the signer:

• *Signing:*

The signer uses its private key d to compute the signature:

$$s = m^d \bmod n.$$

Anyone can verify that s is signature on the message m with respect to public key (e,n) by performing the following step.

• *Verifying:*

Given a pair (m,s), the signature s is correct for message m if and only if the equation:

$$m = s^e \bmod n$$

is satisfied [7].

B. Chaum's Blind Signature Protocol:

This blind signature protocol is based on the RSA digital signature algorithm [4].
Assume that the receiver wants to get a signature on a message m that corresponds to an integer between 0 and n, Chaum's Blind Signature Protocol then consists of the following three steps:

• *Blinding*:

The receiver picks a blinding factor r, which is a random integer between 0 and n, and computes the value:
$m' = m^* r^e \bmod n$. The receiver sends m' to the signer. The m' is the message to be signed by the signer and not the original message m.

• *Signing:*

The signer uses its private key d to compute the value:
$s' = m'^{\,d} \bmod n$. The signer returns s' to the receiver.

• *Unblinding*

The receiver extracts the signature:

$$s = s' / r \bmod n$$

So, the receiver ends up with a pair (m,s) satisfying the equation $s = m^e \bmod n$. This is exactly the verification relation for standard RSA signatures. It should be noted that the signer doesn't know which message m has actually been signed. Because of the random blinding factor r, the message m' is statistically independent of the actual message m. Strictly speaking, we should require that the blinding factor r is co-prime with n, but we will not be concerned with such details here.

IV. ELGAMAL SIGNATURE SCHEME

ElGamal signature scheme was first introduced in 1985 and is described in this section (see [2] for details). In this signature scheme the public key is used for encryption and signature verification. For each user, there is a key pair, which consists of a secret key x, and a public key y where:

$$y = \alpha^x \bmod p.$$

The public key y is published in a public file and known to everybody while the secret key x is kept secret.

Let m be a document to be signed, where:

$$0 \le m \le p-1 \text{ and } p \text{ is a large prime}$$

Thepublic file consists of the public key $y = \alpha^x \bmod p$ for each user. To sign a document, a user A uses the secret key x_A to compute a signature for m so that any user can verify that this message has been signed by A, using the public key y_A together with α and p. No one can forge a signature without knowing the secret x_A. The signature for m is a pair (r,s), where $0 \le r, s < p-1$, chosen such that :

$$\alpha^m = y^r r^s \bmod p \qquad (1)$$

A. Signing Procedure

The following three steps are done to compute the signature:

1. Choose a random number k, uniformly distributed between 0 and $p-1$, such that :
   $\gcd(k,p-1) = 1$
2. Compute $r = \alpha^k \bmod p$
3. Now (1) can be written as

$$\alpha^m = \alpha^{xr} \alpha^{ks} \bmod p$$

which can be solved for s by using:

$$m = xr + ks \bmod (p-1)$$

B. Verification Procedure

Given m, r, and s, it is easy to verify the authenticity of the signature by computing both sides of (1) and checking that they are equal.

V. A NEW BLINDING SCHEME

A. Scheme Explanation

A new blinding scheme is described in this section, which depends on ElGamal signature scheme. ElGamal digital signature scheme has a valuable property, which is the randomness of k. This randomness assures that if the same message is signed twice the two signatures will be different. This is not achieved in RSA. The mathematics of the new scheme will be explained through the following example. Suppose that Alice wants the Bank to sign a message for her. The blinding procedure goes as follows:

■ Alice should do the following:

1. Choose a random number k, uniformly distributed between 1 and $p-1$, such that:
   $\gcd(k,p-1) = 1$
2. Compute $r = \alpha^k \bmod p$
3. Take a blinding factor h such that:
   $\gcd(h,p-1)=1$
4. Compute $m' = h^*m \bmod (p-1)$
5. Send m' and r to the Bank

■ The Bank should do the following:

1. The Bank receives m' from Alice and treats it as any ordinary message since the Bank does not recognize the blinding.
2. The Bank computes s' from the relation:
   $m' = (xr + ks') \bmod (p - 1)$
   $\Rightarrow s' = (m' - x*r)*k^{-1} \bmod (p - 1)$
   where $s`$ is the blinded signature of m.
3. The Bank sends $s`$ to Alice.

■ Alice should do the following to recover the real signature s after receiving the blinded signature $s`$ from the Bank:

1. To find the unblinded signature s for m :

$$m = h^{-1} * m' \bmod (p - 1)$$
$$m = h^{-1}(xr + ks`) \bmod (p - 1)$$
But: $m = (xr + ks) \bmod (p - 1)$
$$\therefore xr + ks = h^{-1}(xr + ks`) \bmod (p - 1)$$
$$ks = h^{-1}(xr + ks`) - xr \bmod (p - 1)$$
$$ks = xr(h^{-1} - 1) + h^{-1}ks` \bmod (p - 1)$$
$$\therefore s = xrk^{-1}(h^{-1} - 1) + h^{-1}s` \bmod (p - 1) \quad (2)$$

which is easily computed by Alice.

2. Now the complete signature pair of the message m is : (r,s) which are both known to Alice but not to the Bank.

B. Complexity and Properties

For this blinding scheme only one member of the signature pair is affected by signing, which is s. The member r is kept the same. This is comparable to the RSA blinding in which only one number is changed by the blinding, which is the only number that comprises the signature.

The blinding procedure here is easier and faster than that of the RSA since it requires only a single multiplication ($m*h$) while the RSA blinding requires multiplication and exponentiation ($m*r^e$). On the other hand, the unblinding procedure for the RSA signatures is simpler since it requires only a single division ($s`/r$) while the unblinding in the new scheme requires evaluating expression (2) above.

The new scheme can also generate the ordinary (non-blinded) signatures by assuming $h = 1$, a property that can be used to generate both types of signature using a single scheme.

B. Application

The following procedure introduces an untraceable off-line electronic payment protocol assuming that the consumer Alice wants to purchase some goods from the merchant Bob and that they have bank accounts with Bank:

• *Withdrawal:*

1. Alice creates an electronic coin and blinds it.
2. Alice sends the blinded coin to the Bank with a withdrawal request.
3. Bank digitally signs the blinded coin.
4. Bank sends the signed-blinded coin to Alice and debits her account.
5. Alice unblinds the signed coin.

• *Payment:*

1. Alice gives Bob the coin.
2. Bob verifies the Bank's digital signature. (optional)
3. Bob gives Alice the merchandise.

• *Deposit:*

1. Bob sends coin to the Bank.
2. Bank verifies Bank's digital signature.
3. Bank verifies that coin has not already been spent.
4. Bank enters coin in spent-coin database.
5. Bank credits Bob's account.

VI. CONCLUSION

A new generalized blind signature scheme is presented, which could be utilized to generate blinded as well as normal ElGamal digital signatures. The new scheme has the advantage of being efficient. It is faster and simpler than RSA in the blinding procedure. The anonymity of the signatures generated by the new method is higher than that of RSA because of its inherent property that varies the output signatures for the same message, even without blinding. This new scheme can be utilized to offer privacy enhancement and unlinkability in electronic banking operations and e-commerce transactions.

REFERENCES

[1] W. Diffie and M. Hellman, "New directions in cryptography", *IEEE Trans. Inform. Theory*, vol. IT-30, 1976.

[2] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", *IEEE Trans. Inform. Theory*, vol. IT-31, 1985.

[3] S. Pohlig and M. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance", *IEEE Trans. Inform. Theory*, vol. IT-24, 1978.

[4] D. Chaum, A. Fiat and M. Naor, "Untraceable electronic cash", *Advances in Cryptology, Crypto 88*, S. Goldwasser (Ed.), Springer-Verlag.

[5] J.Orlin Grabbe, "Cryptography and Number Theory for Digital Cash".

[6] Beker and Piper, "Cipher Systems", Northwood Publications, 1982.

[7] William Stallings, "Cryptography and Network Security: Principles and Practice", Second Edition, *Prentice Hall*, 1999.

[8] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, "Handbook of Applied Cryptography",*CRC Press*, 1997.