American Journal of Applied Sciences 2 (4): 873-876, 2005
ISSN 1546-9239

Blind Decryption and Privacy Protection

Mohammed Al-Fayoumi and Sttar Aboud
Department of Computer Science, Applied Science University, Jordan, Amman

Abstract: Blind decryption is an efficient way of protecting customer's privacy in online marketing over the Internet (i.e. Hiding information about which goods a user purchases from the vendor). In this study, the RSA based blind decryption is simply transposed from an identical protocol as the Chaum's blind signature scheme and the blind decryption protocol for the Elgamal encryption scheme is suggested. In addition, the difference between the known RSA based blind decryption protocol and our proposed protocol is examined in applications to protect copyright subjects of e-commerce documents over the internet.

Key words: Blind decryption, privacy protection, on-line marketing, RSA scheme, Elgamal scheme

INTRODUCTION

The blind decryption scheme was introduced by Sakurai and Yamane[1], it is defined for a public key encryption scheme. It is a protocol between two entities, A as a sender and B as a receiver. In which entity A has a document encrypted by entity B's public key and entity A wants entity B to decrypt the document without disclosing neither decrypted original document nor knowing B's private key.

Chaum[2] reported an analogous idea for signature mechanism, as a blind signature scheme, in which entity A receives a legal signature for a document from a signer entity B without perceiving the document or acquired signature. The original blind signature developed by Chaum is dependent on RSA scheme[3]. In the example of the RSA scheme, decrypting an encrypted document has the same procedure as signing a document; then we simply transpose the blind signature protocol to a blind decryption protocol. Moreover, Micali [4] implemented the blind decryption protocol depending on the RSA scheme to a fair public Key crypto-system for making trustees oblivious. Carmenisch *et al.*[5] introduced an efficient method on a blind signature protocol dependent on Elgamal encryption scheme [6-8], which is a different form. RSA scheme, the blind signature scheme proposed by Carmenisch *et al.* [5] cannot be straighten used in blind decryption. Absdi *et al.*[9] conceptually examined a typical example of blind computation. They described a technique of finding blindly the discrete logarithm. So, calculating the exponents of the results supposes to be extensive and consequently their manner does not solve the practical issue on the decryption Elgamal scheme.

In this study, we introduce a blind decryption protocol for Elgamal public key encryption algorithm [10]. The suggested protocol employs an identical scheme of the discrete logarithm as employed in the metal poker protocol suggested by Shamir *et al.*[11]. The difficulty of the blind decryption protocol for Elgamal public key encryption scheme is that entity A cannot prove the accuracy of decrypted document. Whilst in the example of RSA scheme the accuracy of the decrypted document can be verified by any person by using the encrypted document and the public key. However, In the example of Elgamal public key encryption scheme, entity A requires the entity B's aid for the verification to recover the document.

We can employ blind decryption in e-commerce and on-line marketing over the Internet for protecting purchasers' privacy (i.e. Hiding information about which goods a user purchases from the vendor). We also consider the difference between the RSA blind decryption protocol and proposed Elgamal blind decryption protocol. The RSA blind decryption protocol provides a transitive self-certificate on the decrypted document, which is considered as a digital signature protocol whilst the Elgamal blind decryption shows no self-certificate property. This difference reflects the different applications in e-commerce systems over the Internet[1].

An application of the blind decryption protocol is a "payment goods method" over the Internet by protecting customer's privacy. The producer assigns to the individuals various e-commerce messages; every message is encrypted using the producer's private key. Subsequently receiving these encrypted messages and in case, the individual need to see and understand the whole message, the individual requests the producer to decrypt the encrypted message. So, this clear demand show which message the individual wants. The blind decryption protocol is significant to protect customer's privacy. Initially, the producer assigns encrypted messages with the producer's identical private key to every customer. Secondly, the customer requests the producer to decrypt the encrypted message that the customer wishes to see by way of a blind decrypting algorithm. This discloses no data, which message the customer needs. In addition, the customer cannot see additional messages than the customer wants since the customer cannot reach the producer's private key.

The suggested blind decryption protocol performs entire invisibility of the decrypted document verses the decrypter, which has a negative side. This side is considered as the spotting difficulty of the oracle[12].

Blind Decryption Implementation:

Chaum's Blind Signature Protocol: The idea of the blind signature was inverted by Chaum[2, 13], who also developed their first implementation[14], it uses the RSA mechanism. Let entity B have a public key (e), a secret key (d) and a public modulus (n). Entity A wants entity B to sign the message (m) blindly (i.e. the signature on message miss = m^d mod n). Entity A checks whether the signature (s) on a message (m) convinces $s^e \equiv m \pmod n$.

* Entity A Randomly Picks r such that $1 < r < n$ and gcd(r, n) =1 Then Blinds m by Finding: $t = m*r^e$ mod n
* Entity B sign t as follows: $y = (t)^d$ mod n
* Entity A unbinds y by finding: $s = r^{-1} * y$ mod n
* The result is: $s = m^d$ mod n.

This can simply be reflected: $y \equiv (m*r^e)^d \equiv m^d * r$ mod n.

So, $y * r^{-1} \equiv m^d * r * r^{-1} \equiv m^d$ mod n

The blind signature algorithm permits objectify e-payment systems protecting user's privacy and, other crypto-system schemes protecting the user's anonymity same as e-voting systems.

Example: Suppose entity B has a public key (e = 19), a secret key (d =139) and a public modulus (n = 1403). Assume entity A has a message (m = 41) and wants entity B to sign the message blindly. Then entity A checks whether the signatures convince $s^e \equiv m$ mod n.

Suppose entity A picks r = 21, then entity A blinds m by finding:

$t = 41* 21^{19} \text{ mod } 1403 = 176$

Entity B signs t as follows:

$y = (176)^{139} \text{ mod } 1403 = 1161$

Entity A un blinds y by finding:

$s = 21^{-1} * 1161 \text{ mod } 1403 = 857$

The results are:

$s = 41^{139} \text{ mod } 1403 = 857$

This can be reflects:

$176^{139} \equiv (41*21^{19})^{139} \equiv 41^{139} *21 \text{ mod } 1403$

$1161 \equiv 1161 \equiv 1161$

So $176^{139}*21^{-1} \equiv 41^{139} * 21*21^{-1} \equiv 41^{139} \text{ mod } 1403$

$857 \equiv 857 \equiv 857$

Now entity A checks whether the signatures on a message m convinces:

$857^{19} \equiv 41 \text{ mod } 1403$

$41 \equiv 41$

RSA Based Protocol: A blind decryption can be applied employing the RSA public key encryption scheme by an identical mechanism as in the RSA based blind protocol introduced by Chaum[2]. Suppose that n is the public RSA modulus of entity B, e is the public key for encryption and d is the private key for decryption (i.e. Encryption of document misses = m^e mod n and the decryption is $m = s^d$ mod n). Assume that entity A has a message m, which is encrypted using the public key e of entity B.

* Entity A randomly, secretly chooses an integer r where $1 <= r < n$, gcd (n, r) =1 Then computes $x = r^e * m$ mod n and sends this to entity B.
* Entity B finds $y = x^d$ mod n and sends y to entity A.
* Entity A finds $z = r^{-1} * y$ mod n, which is an entity's b's signature on m.

Example: Suppose entity B public modulus (n = 1403), the public key (e = 19) and the private key (d = 139). Assume entity A has a message (m = 41), which is encrypted using the public key e of entity B.

$s = 41^{19} \text{ mod } 1403 = 430$

$m = 430^{139} \text{ mod } 1403 = 41$

Suppose entity A picks (r = 21) then entity A computes:

$x = 21^{19} *41 \text{ mod } 1403 = 176$

Entity B finds $y = 176^{139} \text{ mod } 1403 = 1161$

Entity A finds $z = 1069 * 1161 \text{ mod } 1403 = 857$ which is the entity's signature for m.

Actually, Micali[4] employed the blind decryption protocol depending on RSA mechanism for a fair crypto-system for making trustees oblivious. However, Micali's fair crypto-system is dependent on the Diffee-Hellman key exchange scheme[15], which employs the discrete logarithm problem. So, if we would have an Elgamal based blind decryption protocol, we could produce a fair crypto-system with making trustees oblivious by employing the unique crypto-system taking strength of the discrete logarithm problem.

Elgamal Based Protocol: In the Elgamal public key encryption scheme[15], entity B generate a random prime p and a generator g of the multiplicative group z_p^*, chooses a random integer x where, $1 <= xgx= p-2$ and finds $y = g^x$ mod p. Entity B determines (y, g, p) as a public key whilst holds x as the private key. Assume that entity A send a message m to entity B. Now the protocol as follows:

* Entity A randomly select an integer r less than p - 2, then finds $c_1 = g^r$ mod p and $c_2 = m * (y)^r$ mod p. Then sends (c_1, c_2) to entity B.

* Entity B employs the private key to compute $d = c_1^{p-1-x} \bmod p$ and then recovers m by finding $m = d * c_2 \bmod p$.

Example: Suppose entity B chooses the prime p = 2357 and a generator g = 2 of Z_{2357}. Entity B selects the secret key x = 1751 and computes: $y = g^x \bmod p = 2^{1751} \bmod 2357 = 1185$, B's public key is (p = 2357, g = 2, y = 1185).

To encrypt a message m = 2035, entity A chooses a random integer r = 1520 and finds:

$c_1 = 2^{1520} \bmod 2357 = 1430$
and $c_2 = 2035 * 1185^{1520} \bmod 2357 = 697$
Entity A sends $c_1 = 1430$ and $c_2 = 697$ to entity B.

To decrypt, entity B should computes:

$d = c_1^{p-1-x} \bmod p = 1430^{605} \bmod 2357 = 872$
and recover m by finding:
$m = 872 * 697 \bmod 2357 = 2035$

The Proposed Blind Decryption:
The Protocol We Suggest: Assume that entity B has a public key (y, g, p) and a private key x. Also suppose that entity A sends a message m to entity B. Entity A randomly selects an integer r less than p - 2 and finds $c_1 = g^r \bmod p$ and $c_2 = m * (y)^r \bmod p$. Then send (c_1, c_2) to entity B. Now the protocol is as follows:

* Entity A randomly picks e less than p -1, finds $x^- = c_1^e \bmod p$ and sends x^- to entity B.
* Entity B finds $y^- = (x^-)^x \bmod p$ and sends y^- to entity A.
* Entity A employs the private key e to recover m as follows:
* Compute $z = (y^-)^{-1} \bmod p$
* Compute $z^- = (y^{-1})^{y^-} \bmod p$
* Compute $m = z^- * c_2 \bmod p$

Example: Suppose entity B chooses the prime number p = 2357 and a generator g = 2 of Z_{2357}. Entity B selects the secret key x = 1751 and computes: $y = g^x \bmod p = 2^{1751} \bmod 2357 = 1185$, B's public key is (p = 2357, g = 2, y =1185).

To encrypt a message m = 2035, entity A chooses a random integer r = 1520 and finds:

$c_1 = 2^{1520} \bmod 2357 = 1430$
and $c_2 = 2035 * 1185^{1520} \bmod 2357 = 697$
Entity A sends $c_1 = 1430$ and $c_2 = 697$ to entity B.

To compute x^- entity A chooses the secret key e = 21, then finds:

$X^- = 1430^{21}$ modes 2357 = 1881 and send this to entity B.

Entity B should find:

$y^- = 1881^{1751} \bmod 2357 = 313$ and send this to entity A.

Entity A recovers m as follows:

* Compute $z = (313)^{-1} \bmod 2357 = 313 * 1860$ modes 2357 = 1
* Compute $z^- = (1860)^{313} \bmod 2357 = 872$
* Compute $m = 872 * 697 \bmod 2357 = 2035$

Note that a same approach to producing a discrete logarithm based cryptosystem blind is employed[1, 16].

Though, we choose a generator g of the multiplicative group of z^*_p, the set: $s(r) = \{(g^r)^e \bmod p : e \in z_{p-1}\}$ may be a smaller set than z^*_p for a randomly picked r. This could release some data on entity A's private key. A simple technique to prevent this difficulty is to select the prime modulus p such that p = 2q + 1, q is also prime and additional causes the generator gas the prime order q.

Preventing Deceiving If Any: In the RSA based blind decryption mechanism, the accuracy of the decrypted document is verified by any individual with the encrypted document and the public key, as it has a self-verification matter. But in the example of the Elgamal public key encryption scheme, entity A cannot check the accuracy of decrypted document, on account the encrypted document is randomized therefore being not unique in the Elgamal public key encryption scheme. However, in the protocol suggested were entity B has an opportunity to deceive entity A through sending $Y = (x^-)^t \bmod p$ where $t \neq x$. To prevent such a deceiving by entity B, we use an extra sub-protocol, in which entity B proves that indeed accurately calculate y^- from x^-, in which the verifier calculate that $y^- = x^s \bmod p$ by employing public key (g, p, $y = g^x \bmod p$). Assume that the prime modulus p such that p = 2q + 1, q is also prime and the generator g has the prime order q. The steps are as follows:

* Entity A chooses $j_1, j_2 \in z^*_p$ rand) mly (a) d finds $w = (y^-)^{j_1} * (p)^{j_2} \bmod p$ and sends w to entity B.
* Entity B finds $f = w^{x^{-1} \bmod q} \bmod p$ and sends f to Entity A.
* Entity A accepts y as an accurate answered, if and only if the formula $f \equiv (x^-)^{j_1} * g^{j_2} \bmod p$.

DISCUSSION

The difference between the RSA based blind decryption and our proposed Elgamal depending on the Elgamal mechanism, is that in the example of RSA based protocol any individual can check the accuracy of the decrypted message by the encrypted document with the public key (i.e. self-verification), whilst in our proposed Elgamal based blind decryption entity A cannot check the accuracy of the decrypted message.

In addition, in the example of an RSA based protocol, Entity B can transfer entity as certification, which is the encrypted document, on the decrypted document to any trusted authority as the general scheme. Nevertheless, our proposed Elgamal based protocol has no such possibility; even entity A declares a pair of encrypted and decrypted document. This means that there is no individual can verify the validity without entity bs aids through the protocol. The proposed Elgamal based protocol has a positive implementation is to limit unauthorized distribution of copyright on e- documents.

Also, in the blind decryption, entity B uses his private key to a random number j that is provided to him from entity A without any authentication. If entity A is genuine, the number j should be transformed from a decrypted document with entity bs private key. However, an entity A has an opportunity to deceive by obtaining the entity bs private key by certain computation on any document. This is a general difficulty is called hiding information from an oracle[12].

A technique to control such problem could be that entity B demands certain authentication on entity as provided a document, though this solution loss entire invisibility verses entity B in the blind decryption. We must remind that entire untraced of blind decryption would allow right crime[17-19]. Unfortunately, until now, the authors have no concept to find the key to such difficulty and finding the right answer is left as a visible difficulty.

CONCLUSION

This study considered a cryptography idea and blind decryption. We suggest a blind decryption protocol based on Elgamal public key encryption algorithm. Thus, we build an efficient scheme with making trusts oblivious[4], by employing the unique cryptography assumption of difficulty of the discrete logarithm problem.

Additionally, we conclude that the Elgamal blind decryption has a privilege compare with the RSA blind decryption in the application for protecting copyright subjects of e-commerce documents. The future areas are to develop several applications of blind decryption on e-voting, digital money and other similar applications for protecting privacy.

REFERENCES

1. Sakurai, K. and Y. Yamane, 1996. Blind Decoding, Blind Undeniable Signature and Their Application to Privacy Protection. Lecture Notes in Computer Science 1174, Information Hiding, Springer-Verlag, pp: 257-264.
2. Chaum, D., 1983. Blind signatures for untraceable payments. Advances in Cryptology Proceeding of CRYPTO' 82: 199-203.
3. Rivest, R., A. Shamir and L. Adlman, 1978. A method for obtaining digital signatures and public key cryptosystems. Communication of ACM, 21: 120-126.
4. Micahi, S., 1993. Fair public key cryptosystems. Proceeding Crypto'92, pp: 113-138.
5. Carmenisch, J., J. Piveteau and M. Stadler, 1995. Blind signature schemes based on the discrete logarithm problem. Proceeding of Eurocrypto 94, pp: 428-432.
6. Schnorr, C. and M. Jakobsson, 2000. Security of Signal Elgamal Encryption. Asiacrypt Proceeding, pp: 73-89, Springer-Verlag.
7. Verheul, E. and V. Tilbory, 1998. Blind elgamal: A fraud detectable alternative to key-escrow proposals. EUROCRYPT '97, May 11-15, 1997, pp: 119-133.
8. Lee, H. and T. Kim, 2000. Message recovery fair blind signature. Public Key Cryptography, second International Workshop on Practice and Theory in Public Key Cryptography, Japan, March 1-3, 1999. Proceeding, pp: 97-111, Springer.
9. Abadi, M., J. Feigenbaum and J. Kilian, 1989. On hiding information from an oracle. JCSS., 39: 21-50.
10. Elgamal, T., 1985. A Public Key Cryptosystems and a signature Scheme based on Discrete Logarithms. IEEE Transaction on Information Theory Based, IT, 31: 469-472.
11. Shamir, A., L. Rivest and L. Adlman, 1979. Mental Poker. MIT/LCS, TM-125.
12. Anderson, R. and R. Needham, 1995. Robustness principles of public key protocols. Advances in Cryptology-CRYPTO'95, LNCS 963, pp: 236-247.
13. Chaum, D., 1988. Blind signature systems. U.S. Patent# H4, 759, 063.
14. Chaum, D., 1985. Security without identification: Transaction systems to make big brother obsolete. Communication on the ACM, 28: 1030-1044.
15. Diffe, W. and M. Hellman, 1976. New directions in cryptography. IEEE Transaction on Information Theory, IT 6: 644-654.
16. Chaum, D. and T. Pederson, 1993. Wallet databases with observers. Advances in Cryptology, CRYPTO' 92, pp: 89-105.
17. Aktas, E. and U. Mitra, 2003. Adaptive blind decoding of unitary space-time constellations. 26: 2598-2602.
18. Solms, S. and D. Naccache, 1992. On blind signatures and perfect crimes. Computers and Security, 11: 6.
19. Ma, W., P. Ching, T. Davidson and X. Xia, 2004. Blind maximum likelihood decoding of orthogonal space-time block codes. Proceeding of the IEEE Global Communication Conference, San Francisco, CA, USA, Dec. 2003, USA., 2004.