A Stamped Blind Signature Scheme based on Elliptic Curve Discrete Logarithm Problem

Kalyan Chakraborty and Jay Mehta
(Corresponding author: Jay Mehta)

Harish-Chandra Research Institute, Chhatnag Road, Jhusi, Allahabad - 211 019, India.
(Email: jaymehta@hri.res.in)

(Received July 13, 2011; revised and accepted Nov. 28, 2011)

Abstract

Here we present a stamped blind digital signature scheme which is based on elliptic curve discrete logarithm problem and *collision-resistant* cryptographic hash functions.

Keywords: blind digital signature, discrete log problem, elliptic curves, hash function, protocol

1 Introduction

A blind signature scheme is a protocol allowing the recipient to obtain a valid signature for a message, from the signer without him or her seeing the message. Blind signature scheme is a digital signature scheme which satisfies *non-forgeability* and *unlinkability* properties. Non-Forgeability property means that only signer should be able to generate valid signatures. Every digital signature scheme should satisfy non-forgebility property. Unlinkability property means no one can derive a link between a protocol view and a valid blind signature except the requester or the author of the message.

The concept of blind signature was first introduced by Chaum (1983) [2, 3], which was a breakthrough in achieving the digitalization of signature services. But his scheme was vulnerable to chosen-plaintext attack. Many blind signatures that satisfy anonymity and unlinkability have been proposed [1, 5, 11]. Blind signatures are publicly verified by any third party and meet the requirements of privacy-oriented protocols that have a conflict of interest between the signer and message's author. Blind signature schemes helps in realizing secure electronic payment systems or voting systems protecting customer's or voter's privacy as well as other cryptographic protocols protecting the participants anonymity. Couple of stamped blind signatures are also given in [4, 7].

In this paper we propose a stamped blind signature scheme based on discrete logarithm problem for elliptic curves and on one-way, collision-resistant cryptographic hash functions.

2 Preliminary

2.1 Definitions

Definition 1. ***Elliptic Curve Discrete Logarithm Problem:***
Given an elliptic curve E over a finite field $\mathbb{F}_q$ and a point Q on E other than $\mathcal{O}$, the discrete logarithm problem on E to the base Q is the following:
Given a point P in $E(\mathbb{F}_q) \setminus \{\mathcal{O}\}$, find an integer n such that $nQ = P$, if such an integer exists.

Definition 2. ***Cryptographic Hash Function:***
A cryptographic hash function is a function that takes inputs of arbitrary length, sometimes a message of billions of bits, and outputs values of fixed length.
A hash function h should have the following properties:

1) *Given a message m, the value $h(m)$ can be calculated very quickly and easily.*

2) *Given y, it is computationally infeasible to find m with $h(m) = y$. (This says that h is pre-image resistant.)*

3) *It is computationally infeasible to find distinct messages m_1 and m_2 with $h(m_1) = h(m_2)$. (This says that h is strongly collision-free.)*

The second and third property of hash functions prevents an adversary from producing messages with a desired hash value, or two messages with the same hash value. This helps prevent forgery. There are several popular hash funtions available, for example MD5, due to Rivest [8]. A survey on hash functions is given by Preneel [6].

3 Domain Parameters

In this section, we describe the domain parameters for our proposed signature scheme.

Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$. Let P be any point on the elliptic curve E of large prime order p. We call P as the base-point. Let G be the elliptic curve subgroup generated by the point P such that the elliptic curve discrete log problem for G is hard to solve. In addition, our domain parameters also include a cryptographic hash function $h : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ which is *collision-resistant* (one-one).

4 Proposed Blind Digital Signature Scheme

The proposed blind digital signature scheme involves three parties, the Requester(R), the Signer(S) and the Verifier(V). It comprises of two protocols, the signing protocol and the verification protocol. The signing protocol is executed by the Requester R and the Signer S. The verification protocol is carried out by the Verifier V.

Before the signing protocol, the signer chooses his secret key $x \in \mathbb{Z}_p^*$ and computes $Q = xP \in G$, where P is the base point on the elliptic curve E. The signer makes Q public.

4.1 Signing Protocol

The signing protocol comprises of two algorithms, the blinding algorithm and the signing algorithm.The blinding algorithm is executed by the Requester (or the author of the message) and the signing algorithm is carried out by the Signer.

Blinding Algorithm:
The requester wishes to get signer's signature on the message without disclosing the content of the message. This involves blinding the message so that the signer cannot read the message. At the same time the requester wants to make sure that the signer is the designated recipient of the blinded message. This can be achieved by double blinding the message i.e., by putting two locks on it. One lock is put by the signer and he is the only one who can unlock it which assures that he is the only person who is receiving the requester's blinded message. This step uses signer's public key. The blinding algorithm runs as follows:

1) The requester computes $h(M) = m$, where M is the message and $h : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ is the hash function.

2) Then the requester calculates $r = mQ = mxP$ and sends r to the signer for signing.

Remarks:

1) The requester actually wants to send mP to the signer for signing. The only person who can compute mP from $mQ = mxP$ is the one who knows the inverse of x as it involves solving discrete log problem. This makes sure that signer is the recipient of the message mP from requester.

2) After receiving $r = mxP$ from the requester, the signer can compute mP by using the inverse of the secret key x. But knowing m from mP is hard as it involves solving a discrete log problem. This makes sure that the signer cannot view the content of the message sent by the requester, i.e., the message is blinded.

Signing Algorithm:

1) The signer receives $r = mQ$ and computes $r' = x^{-1}r = mP$.

2) The signer generates the signature parameter, called the stamp of the signature, $z = $ <nounce||date||place> and computes $h(z)$.

3) The signer computes an elliptic curve point $R = r' + h(z)P$ and $s = x - h(z)$.

4) The signature (R, s, z) is generated and send to the verifier for verification.

4.2 Verification Protocol

The verifier V verifies the signature as follows:

$$sP - Q + R \stackrel{?}{=} h(M)P$$

If the above expression holds then the signature is considered to be valid. The signature is verified as:

$$\begin{aligned} sP - Q + R &= (x - h(z))P - xP + r' + h(z)P \\ &= xP - h(z)P - xP + mP + h(z)P \\ &= h(M)P \end{aligned}$$

5 Security Analysis

In this section we first describe the security of blind signatures and hidden signature, the different type of possible attacks and the meaning of "breaking a signature scheme". Later we demonstrate the security aspects of the proposed scheme.

5.1 Security of blind and hidden signatures

We describe different attacks on a digital signature scheme and the attacks that lead to "breaking a signature scheme":

Attacks on a Digital Signature Scheme:-
There are two kinds of attacks on a digital signature scheme, *Key-Only Attacks* and *Message Attacks*. In Key-Only Attacks, the adversary knows only the signer's public key. In Message Attacks, the adversary is able to inspect some signatures corresponding to either a known message or some chosen-message before he attempts to

break the scheme. Goldwasser, Micali and Rivest [10] identified four kinds of message attacks grouped according to how the messages are chosen, and whose signatures the adversary sees. The following message-attacks are listed in ascending order of their severity:
Known Message Attack, Generic Chosen Message Attack, Directed Chosen Message Attack and Adaptive Chosen Message Attack.

Attacks that lead to breaking a signature scheme:-
An adversary is able to break signer's signature scheme, if his attack allows him to do any of the following with a non-negligible probability:

A Total Break Compute signer's secret trap-door information.

Universal Forgery Find an efficient signing algorithm which is equivalent to signer's signing algorithm.

Selective Forgery Forge a signature for a particular message chosen a priori by the adversary.

Existential Forgery Forge a signature for at least one message on which the adversary has no control. So the message for which signature is obtained may be random or does not make any sense.
Rompel showed that signatures secured against existential adaptive chosen-message attacks can be based on general *one-way functions* [9].

5.2 Security Aspects of Proposed Scheme

The security aspect of the proposed scheme is two fold. It first analyses *the blindness aspect* and then *the non-forgeability aspect* of the proposed scheme. The Blindness Aspect is important as the core goal of any blind signature scheme is to hide the message from the signer and the Non-Forgeability aspect is the a mandatory property of any digital signature scheme.

Blindness:
"Blindness" means that the signer cannot view the content of the message he is signing as long as m is unrevealed by the requester or the author of the message. In the proposed scheme, the blindness aspect depends on the elliptic curve discrete log problem which is hard to solve. We discuss the blindness aspect of the proposed scheme from signer's point of view and adversary's point of view:

From Signer's view point:
The signer receives $r = mQ = mxP$ from the requester. He can compute mP using the inverse of his secret key x. Calculation m from mP is hard as it is equivalent to solving an elliptic curve discrete logarithm problem in a group of large prime order. So the message is blinded for the signer.

From Adversary's view point:
An adversary sees only Q and $r = mQ$. Calculating m from mQ is equivalent to solving an instance of discrete log problem in an elliptic curve subgroup of larger prime order. Again the adversary sees $r = mxP$. So, even if adversary performs a total break of the system by figuring out signer's secret x then he gets mP. But computing m from mP is again elliptic curve discrete log problem. This shows that the message is hidden from the adversary too even if the signer's secret key is compromised, which results in total breakdown of the signature system. In case of a total break of the cryptosystem, the signature can be verified by comparing the signature parameter z with the signer's database.

Non-Forgeability: As the signer's public key Q is a point on the elliptic curve subgroup generated by P of large prime order p, an adversary can guess the signer's secret key x with a probability $\frac{1}{p}$ which is negligible as p is a large prime. So it is practically impossible for an adversary to guess a random signature.
The following are some non-forgeability aspects of the proposed scheme:

Theorem 1. *It is difficult to find any random message m_2, different from a given message m_1, that satisfies the signature (R_1, s_1) corresponding to m_1 for the stamp $z_2(\neq z_1)$ chosen by an adversary.*

Proof. The adversary wants to find a message m_2 that satisfies the signature (R_1, s_1) for the chosen stamp z_2. This implies $m_1P + h(z_1)P = m_2P + h(z_2)P$ and $x - h(z_1) = x - h(z_2)$. This gives $h(z_1) = h(z_2)$ and hence $m_2P = m_1P$. This is not possible because the hash function h is assumed to be *collision-resistant.* In addition $m_2P = m_1P$ implies that $m_2 = m_1 \mod p$. □

Theorem 2. *It is difficult to find any random stamp z_2, different from a given stamp z_1 corresponding to a message m_1, such that z_2 satisfies the signature (R_1, s_1) for the message m_2 chosen by an adversary.*

Proof. The adversary wants to find a stamp z_2 that satisfies the signature (R_1, s_1) for the chosen message m_2. This implies $m_1P + h(z_1)P - m_2P = h(z_2)P$ and $x - h(z_1) = x - h(z_2)$. The latter expression gives $h(z_1) = h(z_2)$. This is not possible because the hash function h is assumed to be *collision-resistant.* □

Other Attack Scenarios:
The following are some of the possible attack scenarios. We show that these attacks too fail for the proposed scheme.

Attack 1 In this attack an adversary requests the signer to sign the message $m = 1$. In this case $r = mQ = Q = xP$. The signer calculates $r' = x^{-1}r = P$. The signature generated is $(R, s) = (P + h(z)P, x - h(z))$. An adversary can compute $h(z)P$ as he knows P and R. To find signer's secret x from s, an adversary has to find $h(z)$ from $h(z)P$. This is hard as it is equivalent to solving an elliptic curve discrete log problem. Thus, this attack fails as the adversary fails to forge the signature or unable to know signer's secret key x.

Attack 2 The adversary sends $r = P$ to the signer to obtain the signature. In this case, signer computes $r' = x^{-1}P$ and the signature generated is $(R, s) = (x^{-1}P + h(z)P, x - h(z))$. Then the signature will not get verified and it will be considered as invalid. Also, finding x^{-1} is equivalent to knowing signer's secret key x. Hence, the attack fails.

5.3 Efficiency Performance

Before the protocol run, the signer and the requester perform following operations:
The signer chooses his private key x and computes its inverse x^{-1} modulo the order of the base point P. Then the signer computes his public key $Q = xP$ which involves multiplication. The requester performs only one hashing operation $h(M) = m$ prior to the protocol run. In all, there are three operations, namely, inverse operation, hashing operation and multiplication performed by signer and requester before the protocol run. All these operations are offline operations and do not contribute in the actual computation cost of the signature scheme.

The total computation cost of the proposed blind signature scheme is 3 multiplications (2 performed by signer and 1 by requester) and 1 hashing operation performed by signer. Two out of three multiplications are performed one each by signer and requester to blind the message.

6 Conclusion

The blind digital signature scheme proposed here is based on elliptic curve discrete logarithm problem and collision resistant hash functions. Blind digital signature are more preferable over the digital signatures because the message is hidden from the signer. In our blind digital signature scheme, the requester is sure that the message is blinded from the signer and that the signer is the designated recipient of the blinded message. Our scheme is efficient upto 3 multiplications and 1 hash operation.

References

[1] A. Boldyreva, "Efficient threshold signature, multisignature and blind signature schemes based on the gap-diffie-hellman-group signature scheme," *Proceedings of Practice and Theory in Public Key Cryptography - PKC 2003, LNCS 2567*, pp. 31–46, 203.

[2] D. Chaum, "Blind signatures for untraceable payments," *dvances in Cryptology - Crypto '82 Springer-Verlag*, vol. 10, pp. 199–203, 1983.

[3] D. Chaum, "Security without identification: Transaction systems to make big brother obsolete," *Communications of the ACM*, pp. 1030–1044, 1985.

[4] Nikolay A. Moldovyan, "Blind signature protocols from digital signature standards," *International Journal of Network Security*, vol. 13, no. 1, pp. 22–30, 2011.

[5] D. Pointcheval and J. Stern, "Provably secure blind signature schemes," *Advances in Cryptology - Asiacrypt 1992, LNCS 1163*, pp. 252–265, 1996.

[6] B. Preneel, "The state of cryptographic hash functions," *Lecture Notes in Computer Science*, pp. 158–182, 1999.

[7] Mohamed M. Rasslan, "A stamped hidden-signature scheme utilizing the elliptic curve discrete logarithm problem," *International Journal of Network Security*, vol. 13, no. 1, pp. 49–57, 2011.

[8] R. L. Rivest, "The md5 message digest algorithm," *Internet Network Working Group RFC 1321*, 1992.

[9] J. Rompel, "One-way functions are necessary and sufficient for secure signatures," *STOC 90: 22nd Annual ACM Symposium on Theory of Computing*, pp. 387–394, 1990.

[10] S. Micali S. Goldwasser and R. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM Journal of Computing*, vol. 17, no. 2, pp. 281–308, 1998.

[11] Z. Zhao, "D-based weak blind signature from bilinear pairings," *International Journal of Network Security*, vol. 7, no. 2, pp. 265–268, 2008.

Kalyan Chakraborty is an Associate Professor at Department of Mathematics, Harish-Chandra Research Institute, Allahabad, India. His research interests includes Algebraic number theory, Analytic number theory, Elliptic Curves, Cryptography, Automorphic forms. He has received his Ph.D. from Harish-Chandra Research Institute, Allahabad, India.

Jay Mehta is a Senior Research Fellow (Ph.D. Student) at Harish-Chandra Research Institute, Allahabad, India. He has received his M.Sc. from Sardar Patel University, Vallabh Vidyanagar, Gujarat, India. His research interests includes Elliptic Curves, Cryptography, Algebraic Number Theory.