Jacek POMYKAŁA

Warsaw University

Enhanced ID-based signature scheme

Abstract*. In the paper we present the design of new digital signature protocol with the secretly hidden warning in the Gap Diffie-Hellman group. The proposed scheme is the extended Id based protocol applying the idea of Schnorr signature and the subliminal channel defined by Simmons.*

Streszczenie. W pracy przedstawiono projekt nowego protokołu cyfrowego podpisu z *ukrytym ostrzeżeniem wykorzystujący grupę Diffiego-Hellmana z luką obliczeniowo-decyzyjną. Proponowany schemat* jest oparty na rozszerzonym protokole podpisu bazującego na tożsamości i wykorzystuje schemat C. P Schnorra i ideę kanału podprogowego zainicjowaną przez G. Simmonsa. *(**Zaawansowany schemat podpisu cyfrowego opartego na tożsamości**)*.

Keywords: gap Diffie-Hellman groups, bilinear pairing, Schnorr digital signature, coercion secretly embedded warning, subliminal channel.
Słowa kluczowe: grupa Diffie-Hellmana z luką obliczeniowo-decyzyjną, odwzorowanie dwuliniowe, podpis cyfrowy Schnorra, ukryte ostrzeżenie o wymuszeniu, kanał podprogowy.

doi:10.12915/pe.2014.02.30

Introduction

The main purpose of the paper is to extend the standard digital signature scheme to the protocol resistant on the possible coercion of the signature. We consider the model enhancing the Id based signature protocol. To be more precise we consider a signature scheme with the core property of being blackmail secure. Namely, the signer should be able to transfer a message to some trusted authorities subliminally alerting that a signature has been coerced. Such embedded alert must be completely hidden from a point of view of the adversary, which can be a quite powerful entity. Thus, we consider the notion of an embedded secret signature defined in [8]. Our protocol is however dedicated to the extended Id-based model in which there are 3 entities taking part in the protocol: Signer, Key Verification Party and the Trustee responsible for the blackmail recovery. The Key Verification Party acknowledges the verification key used to in the signature verification process. It consists from the keys one generated by the Signer X, while the other by the Verification Key Generator Y. One can view the first one as the short term verification key while the other as the long term verification key. The secret key in distinction the standard Id based signature is known only by the signer. The Trustee *T* is a party that is able to resolve the existence of the embedded coercion warning in the signature. Our solution is based on the Gap Diffie-Hellman group and the idea of Schnorr signature scheme. The protocol may be viewed as the advanced Id based signature scheme with the underlying subliminal transfer contained in the signature. The strong point of the proposed signature is its simplicity and relatively low communication and storage complexity. Below we will deal with some basic notions and definitions required to introduce and compare our solution with other existed within this area of subject and point out their possible applications.

Gap Diffie–Hellman groups

Let (G, +) and (G', .) be additive and multiplicative groups of prime order q. By P, Q let us denote any elements of G. Let $e: G \times G \to G'$ be a bilinear pairing i.e. the map satisfying the following conditions:

- bilinearity: for any P, $Q \in G$ we have $e(aP, bQ) = e(P,Q)^{ab}$ for any integers a and b;
- non-degeneracy: if $P \in G$ is such that for all $Q \in G$, $e(P,Q) = 1$, then P is neutral element of the group G;
- computability: there exist a polynomial time deterministic algorithm to compute $e(P,Q)$ for any $P, Q \in G$.

To construct the bilinear pairing we can use the Weil pairing or Tate pairing related with the supersingular elliptic curves. Assume that the discrete logarithm problems are hard in G and G'. Then with the group G and G' we can associate the following cryptographic problems:

Discrete Logarithm (DL Problem):
Given any random $P, Q \in G$, find (if exists) an integer a such that $Q=aP$.

Computational Diffie-Hellman (CDH Problem):
Given a triple (P, aP, bP) with $a, b \in Z_q$ find the element abP.

Decision Diffie-Hellman (DDH Problem):
Given a quadruple (P, aP, bP, cP) decide whether $c = ab \pmod q$.

Gap Diffie-Hellman (GDH Problem):
A class of problems where CDH problem is hard but DDH problem is feasible.

We frequently call the related group the Gap Diffie-Hellman group (GDH group).

The bilinear pairing provides us with the good tool for the construction and verification of the corresponding secret commitments. Namely if we want to prove that the two elements P' and Q' of the group G are the same multiplicities of group elements P and Q respectively, it is sufficient to check if $e(P', Q) = e(P, Q')$.

The hardness of the Computational Diffie-Hellman problem means that given the random multiplicities $aP \in G$ and $bP \in G$ it is hard to compute the value of abP. In other words given $P \in G$, $aP \in G$ and $bP \in G$ it is hard to compute the value of $Q \in G$ so that the quadruple [P, aP, bP, Q] is the Diffie-Hellman quadruple. Obviously in any group with defined above bilinear pairing the DDH problem is feasible. The example of the deterministic algorithm satisfying the above computability condition was given in [15].

Id based signatures

The critical point joining the functionality of the digital signatures is the management and authentication of the corresponding public keys. The potential way of cheating of (certificated) public key causes the risk that the identity of the user may be stolen. The concept of the ID-based public key cryptography introduced by Shamir implied the significant simplification of the corresponding management and authentication process. In this concept the role of public

key has been replaced by the user identity on the network (user-ID) like e-mail address, phone number, etc. More precisely there is the secret key y of the Private Key Generator (PKG), called the master key that is involved in the creation of the entity's secret key $sk = sk(ID, y)$, by means of some trapdoor function. In order to keep the consistency, the public key of PKG (known by each entity) should be related to the secret key $sk(ID, y)$, so that the proof of knowledge of $sk(ID, y)$ could be checked by any verifier. When comparing with the certificate based cryptosystems the elimination of the public key certificates results here with the evident drawback. It implies that PKG knows the user secret key $sk(ID, y)$. Moreover, loosing the master key y would compromise the secret keys of all entities. This obstacle makes favourable the application of the ID-based schemes only in the systems with intermediate level of security. In the next section we shall propose some modification of the above concept in which we replace the party PKG by Y playing the role of Acknowledgement Authority (confirming the verification key related to the secret key of the signer).

Advanced Id based signatures

We encounter the problem of improving the weak point of the standard Id based signatures suggesting some further ideas towards enhancing their security. The first idea is to include in the entity's secret key its private part $k = k_{ID}$ generated by the signer. The corresponding public part $K = K_{ID}$ sent to PKG allows him to approve the corresponding verification key $vk = vk(ID,K, y)$ related to the signer X. As a result, in the subsequent step the signer is able to compute the new secret key $sk = sk(ID, y, k)$ related to the assigned verification key. There are at least two possible approaches to such modifications of the classical ID-based cryptosystem. One basing on the idea of credentials relies on the generation and acknowledgement by Y the new verification key of the signer X corresponding to the computed secret signer's key $sk = sk(ID, y, k)$. The other one which is developed here is based on the two term verification key vk, where one may be viewed as a long term key computed by Y and the other as a short term key computed by the signer X.

We will deal with the additional device which still improves the security of the above scheme against the so called coercion attacks. This concept is based on the notion of the subliminal channel first considered by G. Simmons. More precisely many Id-based signature schemes use a random parameter r in the signature generation process. This admits to hide the suitable private value $k=k_{ID}$ in the corresponding pseudorandom value $r = r(m, k)$. Certainly the corresponding commitment $R = R(r, m, ID)$ depends implicitly on k. Recovering this dependence allows the suitable party T (sharing e.g. the key k with the signer) to read the hidden (in the subliminal channel) information, with the aid of some trapdoor information t (known only by T). There exist at least two approaches to design a suitable subliminal channel to warn the verifier that the signature was coerced. One basing on the idea of sharing the secret key between X and the trustee T, k_{XT} and applying the notion of deniable encryption and the other based on the Diffie-Hellman key exchange idea.

In application the map $R = R(r, m, ID)$ is usually a one way function of r, so extraction of r from R is rather unrealistic task. However it is possible to replace one-way function by a suitable trapdoor function depending on the parameters k and t, that allows to recover the hidden information by the trustee from the corresponding signature. The above idea can be further enhanced in order to protect the signer against quite powerful adversary (that forces the signer to unveil his secret key $sk = sk(ID, y, k)$. In such approach the value of $R = R(m, k, r)$ can be verified only on the basis of some trapdoor information t known only by T. In case of attack the signer show the "fake" values k' and r' instead of k and r leading to the same value $r = r(m, k', r')$.

Deniable encryption approach and possible applications

In this paper we address problem that is quite similar to the one solved by a deniable encryption. But it exists in the other part of the public key cryptography, i.e., in the world of signature schemes. We consider what could happen if, at a certain moment, a signer, would be somehow forced to hand over her private signing key to an adversary a party that knows the signature scheme and all former signatures issued by the signer or to issue valid signatures on messages of the adversary's choice. The main purpose of ours is to give a signature scheme with the core property of being blackmail secure. Namely, the signer should be able to transfer a message to some trusted authorities subliminally alerting that a signature has been coerced. Such embedded warning should be completely hidden from a point of view of the adversary, which can be a quite powerful entity. Thus, following [8] let us recall a notion of an embedded secret signature. It should satisfy the following conditions:

- an ordinary signature receiver is able to verify correctness of a signature, but both types of signatures are considered valid by him: these assembled voluntarily and these coerced by an adversary
- when the signer is forced by the adversary to create a signature, she can leak information subliminally that the signature is coerced
- nobody, even the signer, except some fixed trusted authority is able to extract the information about coercion from a signature; in particularly, the adversary cannot distinguish between both types of signatures even under the assumption that he possesses signatures issued in the past by the signer
- with an overwhelming probability the coercer cannot craft a signature that is considered as a voluntary one by the trusted party (however the adversary might be able to produce a correct signature, which is accepted by the signature receiver, on an arbitrary message) even having access to previously intercepted signatures issued by the signer.

Clearly, no cryptographic solution can prevent the adversary from blocking the entire communication between the signer and the trusted party. However, it is reasonable to assume that the adversary wants to make use of coerced signatures and will present them to some third party (e.g. a bank). In this case that party can contact the trusted authority to verify whether these signatures are legitimate.

The idea leading to the scheme that meets requirements sketched above is to establish a new, shared key K_{XT} between the signer X and the trusted authority T. The ability of hiding an embedded secret in a signature is assured if only K_{XT} is kept secret. Still, the adversary can demand K_{XT} from the signer but now he may present a fake y instead of the real one. If the data the signer gives to the adversary is coherent, then by no means can he tell whether the given key matches K_{XT} or not.

The defined above cryptographic scheme has a natural appearance in the signature schemes with the trusted party involved in the verification process. A typical example concerns the situation when the trusted party legitimates a voluntary signature, or is able to discover the embedded secret in a signature. In some applications the corresponding trusted authority should be involved in the

preparation of a suitable "proof of coercion". Let us consider as an example the e-delegation of signing rights (the corresponding proxy signature primitive has been defined in the work by Mambo et al. [16]). Assume that the original signer who delegates his signing ability to the proxy signer, is equipped with a suitable verification algorithm V*.Then, in the case when the proxy signer is forced to sign a given message, or simply to expose his private key to the adversary, the corresponding signature would be discovered by the designator as a coerced one. Obviously, the strong unforgeability condition (see e.g. [17]) should imply that the designator is not able to generate voluntary signatures on behalf of the proxy signer. Another application towards the group signature schemes, introduced by Chaum and van Heyst [18] may regard the manager as being equipped with the suitable verification algorithm V*, since he is the party that can recognize the identity of signer being coerced. Here the full traceability condition (see e.g.[19]) should imply that the manager is not able to compute the voluntary signature on behalf of a group member. A similar functionality could be also adopted in a more involved context of fair exchange protocols. The hybrid solution [7] joining the idea of anonymous-signer signatures [20] and verifiably encrypted signature [21] is another useful appearance of the signature scheme with the trusted party involved in the verification process V*.The concept of the subliminally embedded warning can be also adopted to the certificateless systems [12]. Unlike the typical Id-based digital signature scheme, this approach avoids to regard the Private Key Generator (PKG) as the trusted party *T*. The shared knowledge between *T* and the signer allows *T* to detect an embedded secret in a corresponding signature (cf. [6]). Another solution is presented in [10]. Though this scheme is more simple and efficient one, it does not apply to some well known signature schemes like in particular the adaptation to universal padding scheme for RSA [22] and Feige-Fiat-Shamir signature scheme [23]. In the first case the corresponding scheme is safe even if the same pair private/public keys are used for signing and encrypting. In the second one the additional (subliminal) information is created on the basis of the two additional primes dividing the modulus which are known only by the trusted party *T* (see [12], [24]). Summing up the deniable encrypting approach might be useful even in the case of the partial leakage of signer's private key.

Related work and our result

The idea of ID-based cryptosystem was introduced by Shamir [1]. The idea of Gap Diffie-Hellman group based on the Weil pairing has its origin in the paper [2]. Boneh and Franklin [3] have proposed the first provably secure ID-based cryptosystem relating to GDH groups. The proxy ID-based digital signature with derandomized Weil pairing computation was proposed in [4]. The general concept of transforming the standard signature schemes into the corresponding identity-based signatures (IBS) was the subject of paper [5]. In this paper we investigate the extensions of ID-based signature schemes having in mind the security requirements. The suitable improvements are based on the idea of subliminal channels investigated by Simmons [6] and applied in [7] for IBS scheme from the bilinear pairing. This approach was then enhanced in [8] for the standard certificate-based signature schemes, referring to the concept of deniable encryption [9] and unconditionally basing on the Diffie-Hellman key exchange idea in [10]. On the other hand the approach focusing on the certificateless (see e.g. [11]) systems was investigated in [12].

In this paper we present the new proposal of digital signature scheme principally based on the idea of the Schnorr signature scheme [13], applying the bilinear pairing based cryptosystem model (cf. e.g. [14]). The secretly embedded warning ties in the discussion of the related papers [8] and [10]. Our proposal focusing on the certificateless systems is therefore the practically required one in view of the possible forgery threats.

Communication model

We distinct 4 parties taking part in the protocol: Signer *X*, Verification Key Generator *Y*, Trusted Party *T* and Verification Party *V*. The Verification Key Generator Y is a party (usually a trusted local center) that assigns and acknowledges the verification key corresponding to the signer *X*. The signer *X* generates the secret key corresponding to the acknowledged verification key published by *Y*. The trusted party *T* is responsible for the validation (using the *Ver**) algorithm whether the signature contains the secretly embedded warning or not. The verification party *V* is any user of the system that may check (using *Ver* algorithm) the correctness of the given signature. First the party *X* contacts *T* to establish the way of notifying about the potential coercion in the signature computing (sending subliminally the secretly embedded warning). Next X applies for the assigning and approval of the corresponding verification key related to the computed secret key. Finally in case of the coercion the party *T* can show any party the proof of existence of the embedded warning in the signature.

The protocol

The protocol is the tuple (*Setup, Establish, Keygen, Signvol, Signcoerc, Ver, Ver**) which are described below. We remark that the verification key *vk* is actually a pair, one part of which is generated by the Signer *X*, while the other by the Verification Key Generator *Y*. One can view the first one as the short term verification key while the other as the long term verification key.

Setup

Having as input the security parameter the algorithm returns the bilinear structure (G, G′, e, P), where P is a generator of *G* of order *q*, while h, h' and H are suitable one-way hash functions such that:

$h : \{0,1\}^* \rightarrow Z_q$

$h': Z_q \rightarrow \{0, 1, \ldots, q-1\}$

$H : \{0,1\}^* \rightarrow G.$

We assume that the parties *Y* and *T* are equipped with the pair of private/public keys *Y*: (y, yP), *T*: (t, $(P, P)^t$), where y, $t \in Z_q$.

Establish

Given the identity *ID* the signer *X* selects randomly an element $k = k_{ID} \in Z_q$ and the corresponding commitment $K_{ID} = k_{ID} P$ sends to *Y*. If accepted the party *Y* computes $Q_{ID}=H(ID)$ and then publish and authorize the pair of verification keys assigned to X as: $vk = vk(ID)=(y\ Q_{ID}, K_{ID} = k_{ID}\ P)$ together with the corresponding signature authorizing the *relevance* of the published pair of the type $yH(vk)$.

Keygen

The algorithm is performed by the signer. Having as input the tuple (k_{ID}, $y\ Q_{ID}$) the signer computes the secret key equal to $sk(ID)=sk(vk, k_{ID}) =(k_{ID}\ y)Q_{ID}$.

Signvol

The algorithm is performed by the signer. It has as input the message m, the public key T of the party T and signer's secret key $sk = sk(\mathrm{ID}) = k_{ID}\, yQ_{ID}$. It returns the signature of the message m of the form $[\rho, S]$. First the signer selects a random element $r \in Z_q$ and then computes the corresponding commitment $\rho = e(P,P)^r$. Next he computes the least significant bit of the hash value $h'[e(T,P)^r] = h'[e(tP,P)^r] = h'[e(P,P)^{rt}] = h'(\rho^t)$. If the resulted bit is equal to 1 then the random choice of $r \in Z_q$ is repeated unless the corresponding bit is equal to 0. Finally the signature of the message m is equal to $[\rho = e(P,P)^r,\ S = h(m, \rho)\ sk(ID) + rP] = [\rho,\ S = h(m, \rho)\ k_{ID}\, yQ_{ID} + rP]$.

Signcoerc

This algorithm differs from the previous one only in the phase of computing the random element $r \in Z_q$. We repeat its random selection until the least significant bit of the hash value $h'[e(T,P)^r]$ is equal to 1.

Ver

Any party having as input the (authorized) pair of verification keys $vk = (y\, Q_{ID},\ K_{ID} = k_{ID}\, P)$ first checks if $[P,\ yP,\ Q_{ID},\ y\, Q_{ID}]$ is a Diffie-Hellman quadruple and then whether $e(S, P) = \rho e(k_{ID}\, P,\ y\, Q_{ID})^{h(m,\ \rho)}$.

*Ver**

The Party T computes the lest significant bit of $h'(\rho^t)$. If it is 0 then the signature is regarded as voluntary, otherwise it is regarded as a coerced one.

Correctness and security remarks

It is clear that the correctly computed signature passes the basic verification process (algorithm *Ver*). This follows from the properties of the bilinear pairing since $e(S, P) = e(h(m, \rho)\ sk(ID) + rP,\ P) = e(P,P)^r\ e(h(m, \rho)\ k_{ID}\ yQ_{ID} = \rho e(k_{ID}\, P,\ y\, Q_{ID})^{h(m,\rho)}$, as required. Moreover the embedded warning is really resolved by the trusted party T since the value of the least significant bit of $h'(\rho^t)$ is the same as of the least significant bit of $h'[e(T, P)^r] = h'[e(P, rP)^t] = h'(\rho^t)$.

As concerns the security of the basic scheme the arguments are similar as in the original Schnorr's signature protocol. The security of the signer secret key is related to the Computational Diffie-Hellman problem since given the value $K_{ID} = k_{ID}\, P$ and the secret key y the party Y is unable to compute the signer secret key $sk(ID) = k_{ID}\, yQ_{ID}$. Actually this reduces to the problem of computing $k_{ID}\, yQ_{ID}$ on the basis of: P, $k_{ID}\, P$ and yQ_{ID} which is the underlying problem related to the Diffie-Hellman quadruple $[P,\ k_{ID}\, P,\ Q_{ID},\ k_{ID}\, yQ_{ID}]$. On the other hand the adversary that wants to resolve the problem of the embedded warning encounters the task of computing the value of $e(P,P)^{tr}$ on the basis of $\rho = e(P,P)^r$ and the Trustee T public key $e(P,\ P)^t$. In other words knowing the values $e(P,P)^r$ and $e(P,P)^t$ we encounter the classical Diffie-Hellman problem of computing $e(P,P)^{rt}$.

Acknowledgment.

This scientific research work is supported by NCBiR of Poland (grant No PBS1/B3/11/2012) in 2012-2015.

REFERENCES

[1] A. Shamir, Identity-based cryptosystems and digital signatures, in Proc. Crypto'87, Santa Barbara, USA, 1987, pp. 47–53.

[2] A. Joux, A one-round protocol for tripartite Diffie-Hellman, J. Cryptol., vol. 17, no. 4, pp. 263–276, 2004.

[3] D. Boneh and M. Franklin, Identity-based encryption from the Weil pairing, SIAM J. Comp., vol. 32, no. 3, pp. 586–615, 2003.

[4] J. Pomykała and B. Źrałek, A model of Id-based proxy signature scheme, in Proc. 6th Coll. Iberoam. Collab. Electron. Commun. eCommerce Tech. Res. Conf., Madrid, Spain, 2008.

[5] M. Bellare, C. Namprempre, and G. Neven, Security Proofs for Identity-Based Identification and Signature Schemes, LNCS,vol. 3027. Berlin: Springer, 2004, pp. 268–286.

[6] G. J. Simmons, The subliminal channel and digital signatures, in Proc. EUROCRYPT'84Worksh. Adv. Cryptol. Theory Appl., Paris,France, 1985, pp. 364–378.

[7] J. Pomykała and T. Trabszys, Blackmail warning verifiably encrypted signatures from bilinear pairing, Bull. WAT, vol. LVII,no. 4, pp. 167–182, 2008.

[8] K. Durnoga, J. Pomykała, and T. Trabszys, Digital signature scheme with secretly embedded warning, to appear in Control and Cybernetics, vol. 4 2013

[9] R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky, Deniable Encryption,LNCS, vol. 1294. Berlin: Springer, 1997, pp. 90–104.

[10] P. Kubiak, M. Kutyłowski, Litweight digital signature with secretly embedded warning, to appear in Control and Cybernetics, vol. 4(2013).

[11] T. Hyla, J. Pejaś, A practical certificate and identity based encryption scheme and related security architecture, LNCS, vol. 8104, 2013 , s. 178-193

[12] J. Pomykała, ID-based digital signatures with security enhanced approach, Journal of Telecommunication and information technology, no 4 (2009).

[13] Claus P. Schnorr. Efficient Identification and Signatures for Smart Cards, Proceedings of CRYPTO '89.

[14] R. Sakai, M. Kasahara, ID based cryptosystems with pairing on elliptic curve, in Symp. Cryptogr. Inform. Secur. SCIS'2003, Hamamatsu, Japan, 2003.

[15] J. Pomykała, B. Źrałek, Dynamic group threshold signature based on derandomized Weil Pairng. Metody Informatyki Stosowanej 17(4), s. 183-193, Polska Akademia Nauk Oddział w Gdańsku, 2008.

[16] M. Mambo, K. Usuda, and E. Okamoto, Proxy Signatures for Delegating SigningOperation, 3rd ACM Conference on Computer and Communications Security (CCS '96) (1996), 48–57.

[17] A. Boldyreva, A. Palacio, and B. Warinschi, Secure Proxy Signature Schemes for Delegation of Signing Rights (2003), available at http://eprint.iacr.org/2003/096.

[18] D. Chaum and E. van Heyst, Group Signatures, Advances in Cryptology – Eurocrypt'91 (2003), 257–265.

[19] M. Bellare, D. Micciancio, and B. Warinschi, Foundations of Group Signatures:Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions, Advances in Cryptology – Eurocrypt '03 2656 (2003),

[20] D Yao and R. Tamassia, Cascaded Authorization with Anonymous-Signer Aggregate Signatures, Information Assurance Workshop, 2006 IEEE (2006), 84–91.

[21] D. Boneh and C. Gentry, Aggregate and Verifiably Encrypted Signatures from Bilinear Maps, Advances in Cryptology – Eurocrypt '03 2656 (2003), 416–432.

[22] M. Bellare and P. Rogaway, Random oracles are practical: a paradigm for designing efficient protocols, Proceedings of the 1st ACM conference on Computer and communications security, 1993, pp. 62–73.

[23] U. Feige, A. Fiat, and A. Shamir, Zero Knowledge Proof of Identity, Journal of Cryptologyc1, 1988, 77-94.

[24] B. Hołyst and J. Pomykała, Electronic Signature and Biometric Methods of Identification (in Polish), WSM publications, ISSN 978-83-7520-042-3, Warsaw (2010).

***Authors**: dr hab. Jacek Pomykała, profesor nadzwyczajny Wydziału Matematyki Informatyki i Mechaniki Uniwersytetu Warszawskiego, ul. Banacha 2, 02-097 Warszawa, E-mail:pomykala@mimuw.edu.pl*

The correspondence address is:
e-mail: pomykala@mimuw.edu.pl