World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:10, No:7, 2016

An Efficient Proxy Signature Scheme Over a Secure Communications Network

H. El-Kamchouchi, Heba Gaber, Fatma Ahmed, Dalia H. El-Kamchouchi

Open Science Index, Computer and Information Engineering Vol:10, No:7, 2016 waset.org/Publication/10004918

***Abstract*—**Proxy signature scheme permits an original signer to delegate his/her signing capability to a proxy signer, and then the proxy signer generates a signing message on behalf of the original signer. The two parties must be able to authenticate one another and agree on a secret encryption key, in order to communicate securely over an unreliable public network. Authenticated key agreement protocols have an important role in building secure communications network between the two parties. In this paper, we present a secure proxy signature scheme over an efficient and secure authenticated key agreement protocol based on the discrete logarithm problem.

***Keywords*—**Proxy signature, warrant partial delegation, key agreement, discrete logarithm.

I. Introduction

THE proxy signature scheme is a variation of the ordinary digital signature scheme. It was first presented by Mambo et al. in 1996. Their proxy signature scheme allows an original signer to delegate his/her signing right to a proxy signer to sign the message on behalf of an original signer [1]. Later, the verifier, which knows the public keys of the original signer and a proxy signer can check a validity of a proxy signature issued by a proxy signer.

The classification of the proxy signature is dependent on the basis of delegation, namely full delegation, partial delegation, and delegation by warrant, and presents a well-organized strategy.

In the full delegation, the proxy signer signs document using the same secret key by the original signer. The drawback of proxy signature with the full delegation is the difficulty to distinct/differentiate between the original signer and the proxy signer. In the partial delegation, the proxy key is derived from the secret key of the original signer and hands it over to the proxy signer as a delegation capability. Due to the partial delegation, the proxy signer's signing capability cannot be restricted, so he/she can misuse the delegation capability.

The weaknesses of full delegation and partial delegation are eliminated by the partial delegation with warrant. A warrant explicitly states the signer's identity, delegation period, and the qualification of messages on which the proxy signer can sign period and the types of a message on which a proxy signer can sign.

There are two types of partial delegation; with warrant protected and unprotected proxy signature schemes. In the unprotected proxy signature scheme, a proxy signature is generated by both proxy signer and original signer. In this case, the verifier cannot distinguish the identity of a signer. In the protected proxy signature scheme, a proxy signature is generated by the proxy signature key of an original signer and also with a private key of a proxy signer.

In 1997, Kim et al. [2] proposed a scheme using the concept of partial delegation with a warrant to restrict the proxy signer signing capability. In 1999, Okamoto et al. [3], for the first time, proposed a proxy unprotected signature scheme based on RSA scheme. A proxy-protected signature scheme based on the RSA assumption was proposed by Lee, et al. in 2001 [4], [5]. In 2009, Shao [6] proposed the proxy-protected signature scheme based on RSA. In 2011, Popescu [7] introduced a secure proxy signature scheme with delegation by warrant, and the scheme is based on the difficulty of solving the discrete logarithm problem (DLP).

The two parties must authenticate one another and agree on a secret encryption key to communicate together securely over an unreliable public network. To achieve this, key establishment protocols are applied at the beginning of a communication session in order to verify the identities of both parties and build a common session key. Authenticated key agreement protocols have an important role in establishing secure communications between the two parties over the open network. The most famous protocol for key agreement was proposed by Diffie and Hellman which is based on the concept of public-key cryptography (DL) [8]. There are two types of the Diffie-Hellman protocol, namely static and ephemeral. In the first one, the parities exchange static public keys, and in the second, they exchange ephemeral public keys [9]. The important feature of the designed protocol is that the established session key is formed as a combination of static and ephemeral private keys of two parties.

This paper demonstrates the effect of an efficient and secure authenticated key agreement protocol on a proxy protected signature scheme based on DLP. The designed protocol for the authenticated key agreement is secure, efficient, and provides authentication between two entities before exchanging the session keys. The remaining parts of this paper are organized as follows: In Section II, we elaborate security properties of the proxy signature scheme. Next, we discuss the designed protocol in Section III. In Section IV, we proposed our proxy signature scheme. We analyze the security properties and common attacks of our proposed scheme in Section V. Finally, in Section VI, we give our conclusion

H. El-Kamchouchi (Prof.), Dr Fatma Ahmed, and Dalia H. El-Kamchouchi (Dr.) are with the Electrical Engineering Department, University of Alexandria, Egypt (e-mail: helkamchouchi@ieee.org, moonyally@yahoo.com, Daliakamsh@yahoo.com).

Open Science Index, Computer and Information Engineering Vol:10, No:7, 2016 waset.org/Publication/10004918

II. Security Requirements of Proxy Signature

The security requirements for any proxy signature are first studied in [1] and later were improved in [4], [5]. According to them, a secure proxy signature scheme is expected to satisfy the following five requirements:

- Verifiability: A verifier can be confident of the original signer's agreement on the signed message from a proxy signature.
- Strong unforgeability: Only the designated proxy signer can generate a valid proxy signature.
- Strong identifiability: The identity of the proxy signer can be determined by any verifier from a proxy signature.
- Strong undeniability: The proxy signer cannot repudiate the signature creation against anyone else, once he/she creates a valid proxy signature on behalf of an original signer.
- Prevention of misuse: The responsibility of the proxy signer should be determined explicitly if he/she misuses the proxy key for the purposes other than generating a valid proxy signature.

III. New Key Agreement Protocol

The used protocol for authenticated key agreement [10] provides authentication between the two parties *A* and *B* before exchanging the session keys. The protocol consists of three phases; The Registration Phase, The Transfer and Substantiation Phase, and The Key Generation Phase. Fig. 1 shows the overall operation of the new protocol.

A		B
	Step 1	
$a_1 \equiv (y_B)^{-r_A} \equiv g^{-x_B r_A}$	d_1	$a_2 \equiv (y_A)^{-r_B} \equiv g^{-x_A r_B}$
$b_1 \equiv (x_A - r_B t_B) mod\ p1$	$\rightleftharpoons$	$b_1 \equiv (x_A - r_B t_B) mod\ p1$
$d_1 = a_1 b_1$	d_2	$d_2 = a_2 b_2$
	Step 2	
$a_{22} \equiv (t_B)^{x_A} \equiv g^{r_B x_A}$		$a_{11} \equiv (t_A)^{x_B} \equiv g^{r_A x_B}$
$b_{22} \equiv a_{22}.d_2 \equiv (x_B - r_B t_A)$		$b_{11} \equiv a_{11}.d_1 \equiv (x_A - r_A t_B)$
$v_2 = {t_B}^{t_A}.g^{b_{22}} \equiv y_B$		$v_1 = {t_A}^{t_B}.g^{b_{11}} \equiv y_A$
	Step 3	
$K_{AB} = {y_B}^{x_A} \cdot {t_B}^{r_A} \equiv g^{x_B x_A + r_A r_B}$		$K_{AB} = {y_A}^{x_B} \cdot {t_A}^{r_B} \equiv g^{x_B x_A + r_A r_B}$

Fig. 1 Overall operation of the proposed protocol

The system picks short-term private key r_A, r_B, they are random integers $2 \le r_A, r_B < p1$ and $GCD(r, p1) = 1$. $p1 = (p-1)$ where p is a large safe prime $p = n'p' + 1$ (n' is a small prime number, usually taken by 2 and p' is a large prime number usually at least 1024 bits). t_A, t_B are short-term public keys where $t_A = g^{r_A} mod\ p$ and $t_B = g^{r_B} mod\ p$. g is a generator of Z_p^*. Furthermore, the system picks long-term private keys x_A, x_B they are random integer where $2 \le x_A, x_B < p1$ and $GCD(x, p1) = 1$ then, computes long-term public key y_A, y_B where $y_A = g^{x_A} mod\ p$ and $y_B = g^{x_B} mod\ p$, K_{AB} is the shared secret key calculated by the new secure protocol between the two parties A and B.

In the first step, the number of scalar multiplications required is one, the number of exponentiation required is one, and the total number of sending message is one. In the second step, each user will be verified from the other one because in the first step each user uses the short-term private key which belongs to him/her in calculation.

IV. The Proposed Proxy Signature Scheme

The proposed proxy scheme is focused on the proxy protected proxy signatures with the new authenticated key agreement protocol based on the DLP. The system is divided into four phases: System setup, Proxy key generation, Proxy key verification, Proxy signature generation and Proxy signature verification.

A. System Setup

It is supposed that the original signer *A* invites the proxy signer *B* to perform signing on behalf of him/her, and the verifier entity *V* verifies the validity of the generated signature. Also, suppose that *p* is a large prime number, and *g* is a generator for Z_p^*. ID_A and ID_B *are* the identity of the original signer and the proxy signer, respectively. $x_A, x_B \in Z_p^*$ are the private key of the original signer and the proxy signer, respectively, then compute public key y_A and y_B where, $y_A = g^{x_A} mod\ p$ and $y_B = g^{x_B} mod\ p$ are the public keys of the original signer and proxy signer, respectively.

B. Proxy Key Generation

(1) The original signer entity *A* should do the following:

- Selects an arbitrary integer value $k_A \in Z_{p-1}$

Open Science Index, Computer and Information Engineering Vol:10, No:7, 2016 waset.org/Publication/10004918

- Find $r_A = g^{k_A} \bmod p$
- Calculate warrant m_w where, m_w must be created from ID_A, ID_B and other data on the delegation.
- Compute $h(m_w \| r_A)$
- Find $\sigma_A = k_A + x_A * (h(m_w \| r_A) \oplus K_{AB}) \bmod p - 1$
- Send $(m_w, r_A, K_{AB}, \sigma_A)$ to the proxy signer in the secure channel.

(2) The proxy signer checks the validity of $(m_w, r_A, K_{AB}, \sigma_A)$ by verifying whether or not the following equation holds $g^{\sigma_A} \equiv r_A y_A^{h(m_w \| r_A) \oplus K_{AB}}$. If the verification is successful, the proxy signer then computes an alternative proxy private/public key pair σ_p and y_p, respectively, such that

$$\begin{aligned} \sigma_P &= \sigma_A + x_B * (h(m_w \| r_A) \oplus K_{AB}) \bmod p - 1 \\ y_P' &= g^{\sigma_P} \bmod p \end{aligned} \quad (1)$$

C. Signature Generation

Now, the proxy signer *B* will sign a message *m* on behalf of the original signer, he uses σ_P to perform an ordinary signing operation. The proxy signature on the message *m* is then $(m, m_w, r_A, Sign_{\sigma_P}(m), K_{AB}, \sigma_A)$.

D. Signature Verification

Any verifier first uses the same verification procedures of the original signing scheme to check $Sign_{\sigma_P}(m)$. Furthermore, the verifier has to check whether or not the following equations hold:

$$y_P' = r_A (y_A y_B)^{h(m_w \| r_A) \oplus K_{AB}} \bmod p \quad (2)$$

V. Security Analysis

In the following, we show that the proposed schemes satisfy the security features, namely, verifiability, strong unforgeability, strong, undeniability, strong identifiability, and prevention of misuse.

A. Verifiability

The verifier of proxy signature, can check the verification equation:

$$\begin{aligned} y_P' &= g^{\sigma_P} \bmod p \\ &= g^{\sigma_A + x_B * (h(m_w \| r_A) \oplus K_{AB})} \bmod p \\ &= g^{\sigma_A} g^{x_B * (h(m_w \| r_A) \oplus K_{AB})} \bmod p \\ &= g^{k_A + x_A * (h(m_w \| r_A) \oplus K_{AB})} g^{x_B * (h(m_w \| r_A) \oplus K_{AB})} \bmod p \\ &= g^{k_A} g^{x_A * (h(m_w \| r_A) \oplus K_{AB})} g^{x_B * (h(m_w \| r_A) \oplus K_{AB})} \bmod p \\ &= g^{k_A} (g^{x_A} g^{x_B})^{h(m_w \| r_A) \oplus K_{AB}} \bmod p \\ &= r_A (y_A y_B)^{h(m_w \| r_A) \oplus K_{AB}} \bmod p \end{aligned}$$

B. Strong Unforgeability

In this scheme, from (1) the proxy signature is created with the proxy signer's secret key x_B and delegated proxy key σ_A. The proxy key is bound with the original signer's secret key x_A and the session key K_{AB}. No one (including the original signer) can construct the proxy signature. If the original signer tries to construct the proxy private key from a proxy public key, he/she will need to solve the DLP. However, the DLP is difficult. Moreover, from (1), the verification of $h(m_w \| r_A) \oplus K_{AB}$ with the signed message prevents the dishonest party from the creation of forged proxy signatures. Therefore, any party, including the original signer cannot forge a valid proxy signature, and thus the proposed scheme satisfies the unforgeability property.

C. Strong Identifiability

Any verifier can determine the identity of the proxy signer from the proxy signatures created by the proxy signer. Therefore, in the proposed scheme, any verifier can identify the identity of the proxy signer from the proxy signature generated by him $(m, m_w, r_A, Sign_{\sigma_P}(m), K_{AB}, \sigma_A)$ on the message *m.*

D. Strong Undeniability:

In the proposed scheme, from (1), the involvements of both original signer and proxy signer are determined by the secret keys x_B and x_A from the proxy signature. Thus, the proxy signer and the original signer cannot deny their involvement in a valid proxy signature. Consequently, the scheme satisfies the undeniability property.

E. Prevention of Misuse

In the proposed scheme, the proxy signer cannot forge the delegated rights. The responsibility of the proxy signer is determined from the warrant m_w in the case of the proxy signer's misuse. Therefore, the original signer's misuse is also prevented because he cannot compute a valid proxy signature against the proxy signer.

Next, we show that our scheme is heuristically secured by considering the following most common attacks.

(1) *Known-Key Security (K-KS)*:In the proposed scheme, if an established session key between original signer and proxy signer is disclosed, the adversary is unable to learn the other established session keys. In each run of the proposed scheme between the two parties, a unique session key which depends on r_A and r_B should be produced. Therefore, the adversary cannot compute K_{AB} and cannot calculate $\sigma_A = k_A + x_A * (h(m_w \| r_A) \oplus K_{AB}) \bmod p - 1$.

(2) *(Perfect) Forward Secrecy:* If both secret keys of two parties are compromised, the adversary is unable to derive the old session keys established by two parties. The protocol also possesses forward secrecy. Suppose that adversary compromises the private keys x_A and he/she

Open Science Index, Computer and Information Engineering Vol:10, No:7, 2016 waset.org/Publication/10004918

cannot calculate $\sigma_A = k_A + x_A * (h(m_w \| r_A) \oplus K_{AB}) \bmod p - 1$. However, the secrecy of previous session keys established by the honest parties is not affected, because an adversary who captured the private key x_A should extract the ephemeral keys r_A or r_B from the exchanged values to know the previous or next session keys between them. Thus, he/she still fails to produce σ_A send to proxy signer. However, this is DLP.

(3) *Key-Compromise Impersonation (K-CI):* When the private key of original signer is compromised, it may be desirable that this event does not enable an adversary to impersonate other entities to the original signer. Suppose that x_A is disclosed. Now an opponent who knows this value can clearly impersonate the original signer. In the proposed scheme, the opponent cannot impersonate the proxy signer to the original signer and compute $\sigma_P = \sigma_A + x_B * (h(m_w \| r_A) \oplus K_{AB}) \bmod p - 1$ without knowing the proxy signer's private key x_B. From the success of the impersonation, the opponent must know the original signer's ephemeral key r_A. So, in this case, the opponent should extract the value r_A from $t_A \equiv g^{r_A} \bmod n$; however, he/she cannot calculate the sharing key, this is DLP.

(4) *Unknown Key-Share (UK-S):* The original signer A cannot be coerced into sharing a key with the proxy signer B without the knowledge of the original signer, i.e., A believes that the key is shared with some entity $C \neq B$, and B believes that the key is shared with A. The used protocol prevents unknown key-share. Corresponding to the proxy signer's public static and ephemeral keys y_B, t_B, an adversary cannot register proxy signer's public keys y_B, t_B as its own, and according to the assumption of this protocol that d_2 has verified that B possesses the private static and ephemeral keys x_B, r_B, respectively. So an adversary cannot deceive the original assuming that $\sigma_P = \sigma_A + x_B * h(m_w \| r_A) \oplus K_{AB} \bmod p - 1$ was originated from him. Therefore, the original signer cannot be coerced into sharing K_{AB} with the proxy signer without his/her knowledge.

VI. Conclusion

In this paper, we proposed a new secure proxy protected signature with a new key agreement protocol based on DLP. Our scheme does not consider the proxy revocation mechanism. The proposed scheme satisfies the necessary security requirements of proxy signature and has a secure channel to deliver the proxy key, through the designed new protocol that meets the security attributes under the assumption of DLP.

References

[1] M. Mambo, K. Usuda, E. Okamoto, Proxy signature: delegation of the power to sign the message, IEICE Trans. Fundamentals E79-A (9) (1996) PP. 1338 - 1353.

[2] S. Kim, S. Park and D. Won, "Proxy signatures", In: ICICS97, LNCS 1334, Springer-Verlag, (1997), pp. 223-232.

[3] T. Okamoto, M. Tada and E. Okamoto, "Extended proxy signatures for smart card", In: Proceedings of Information Security Workshop 99, LNCS 1729, Springer-Verlag, (1999), pp. 247-258.

[4] B. Lee, H. Kim and K. Kim, "Secure mobile agent using strong non-designated proxy signature", In: Information security and private (ACISP01), LNCS 2119, Springer-Verlag, (2001), pp. 474-486.

[5] B. Lee, H. Kim and K. Kim, "Strong proxy signature and its applications", In: Proceeding of the 2001 symposium on cryptography and information security (SCIS01), vol. 2, no. 2, (2001), pp. 603-608

[6] Hwang S., Chen C.: A new proxy multi- signature scheme, International workshop on cryptology and network security, Tamkang University Taipei, Taiwan, pp 26–28 (2001).

[7] Constantin Popescu: A Secure Proxy Signature Scheme with Delegation by Warrant, SIC: Volume 20, issue 4, pp. 373- 380 (2011).

[8] W. Diffie and M. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, vol. IT-l 22, no. 6, PP. 644-654, November, 1976.

[9] K. Chalkias, F. Mpaldimtsi, D. H. Varsakelis, and G. Stephanides, "On the Key-compromise impersonation vulnerability of one-pass key establishment protocols," in Proc. International Conference on Security and Cryptography (SECRYPT 2007), Barcelona, Spain, July 28-31, 2007.

[10] Fatma Ahmedand Dalia Elkamchouchi," A New Efficient Protocol for Authenticated Key Agreement," IACSIT International Journal of Engineering and Technology, Vol. 2, No. 4, April 2013,pp.510-512.