A New Proxy-Protected Signature Scheme Based on DLP

Swati Verma* and Birendra Kumar Sharma**
* School of Studies in Mathematics, Pt. Ravishankar Shukla University Raipur (C.G.), India.
** School of Studies in Mathematics, Pt. Ravishankar Shukla University Raipur (C.G.), India.

Article Info

Article history:

Received Sept 03th, 2012
Accepted Sept 24th, 2012

Keyword:

Cryptography
Digital Signature
Discrete logarithm problem;
Proxy Signature
Security

ABSTRACT

During electronic communication, proxy signature helps the signer to sign message on behalf of the original signer. This type of signature scheme is used when the signatory is not available to sign this document. In this paper, we propose a new protected proxy signature scheme. In the proposed scheme, the proxy signer can sign message on behalf of the original signer with the different type of public key. The proposed scheme satis_es unforgeability, undeniability, non-transferability and distinguishability properties which are the most important security requirements for a proxy signature.

Corresponding Author:

Swati Verma
School of Studies in Mathematics,
Pt. Ravishankar Shukla University, Raipur (C.G.), India.
Email: swativerma15@gmail.com

1. INTRODUCTION

The notion of proxy signature was first introduced by Mambo et al. in 1996 [12, 13]. However, the concept of proxy signature was already recorded in 1989 [4]. A proxy signature scheme is an important investigation in the field of digital signature which involves three entities: an original signer, a proxy signer and a verifier. It provides tools to the original signer to delegate his signing right to a particular signer, known as proxy signer. Once the proxy signer signed the message on behalf of the original signer, the verifier, who knows the public keys of the original and proxy signers, verifies the validity of the proxy signature after receiving it.

In the literature, in 1997, Kim et al. [5] proposed a scheme by restricting proxy signer signing right using the concept of partial delegation with warrant. In 1999, Okamoto et al. [16], for the first time, proposed proxy signature based on RSA scheme, but they considered the proxy unprotected notion. In 2001, Lee et al. [6, 7] proposed a proxy-protected signature scheme based on the RSA assumption. In 2002, Shum and Wei [23] proposed another proxy signature scheme which was proxy protected. The first proxy signature scheme based on the factoring integer problem was proposed by Shao [21] in 2003. In 2005, Zhou et al. [27] proposed two efficient proxy-protected signature schemes. Their first scheme are based on RSA assumption and the second scheme was based on the integer factorization problem. Zhou et al. [27] claimed that their schemes are more efficient than other schemes. However, Park et al. [17] pointed out the shortcoming of their schemes. In 2006, Xue et al. [24] proposed the normal proxy signature scheme and multi-proxy signature scheme based on the difficulty of factoring of large integers but without giving their formal security proofs. In 2009, Shao [22] proposed proxy-protected signature scheme based on RSA. Many variants of RSA-based proxy signature scheme were proposed in the sequel [2, 9, 10, 11].

Most of the proxy signature schemes are based on discrete logarithm problems [5, 25]. Some proxy signature schemes are constructed using on pairings technique [15, 26]. A few proxy signature schemes are also constructed based on factoring problem [21, 22, 24, 27]. In this paper, we propose a new protected proxy signature scheme based on difficulty of solving discrete logarithm problem.

The algorithm for a proxy signature is as follows:

1. **Key generation:** The original signer and the proxy signer enter the system and select their private and public keys.

2. **Proxy key generation:** The original signer generates a proxy key using its secret key for the proxy signer. The proxy key generation step usually involves a two-party protocol to be run between the original and proxy signers.

3. **Proxy signature generation:** The signer (proxy signer) who has to use the signing right of original signer generates a valid signature (proxy signature) using the proxy key for the verifier.

4. **Verifying**: The verifier verifies the input (message and the proxy signature) using the public keys provided by original and proxy signer.

CLASSIFICATION OF PROXY SIGNATURE

Proxy signature schemes are classified mainly into following categories.

- Full delegation: In this category, original signer gives his private key to a proxy signer and the proxy signer signs the document using original signer's secret key. The drawback of this class of proxy signature is the absence of distinguishability between original signer and proxy signer.
- Partial delegation: In this category of proxy signature, the original signer derives a proxy key from his secret key and gives to the proxy signer as a delegation of his signing right. In this case, the proxy signer can misuse the delegation right, because partial delegation does not restrict the proxy signer's to misuse this signing right.
- Delegation by warrant: In this category, the original signer gives the proxy signer a warrant, which construct of a message part and public key. Then, the proxy signer can use the corresponding private key to sign. However, since it requires consecutive execution, it cannot provide faster processing speed.
- Unprotected proxy signature: In the category of proxy-unprotected signature scheme, the proxy signer generates proxy signatures with the proxy signature key given by the original signer only. As a result, both the original signer and the proxy signer can produce the same proxy signatures. When dispute occurs between the original signer and the proxy signer, no one can identify the real signer of the message.
- Protected proxy signature: In the category of proxy-protected signature scheme, only the proxy signer can generate valid proxy signatures. In other words, anyone else, including the original signer, is unable to produce the same proxy signatures.

Organization: The remaining parts of this paper an organized as follows. In Section 2, we elaborate security properties of the proxy signature scheme. Next, we proposed our proxy-protected signature scheme in Section 3. In Section 4, we analyze the security properties of the our proposed scheme. Finally, in Section 5, we give our concluding remarks.

2. SECURITY REQUIREMENTS OF PROXY SIGNATURE SCHEME

The security requirements for any proxy signature are first studied in [12, 13], and later those were improved in [6, 7]. According to them, a secure proxy signature scheme is expected to satisfy the following five requirements:

1. **Verifiability**: The verifier is convinced that the original signer has given consent to the proxy signer to sign a message.
2. **Strong unforgeability:** No body else other than the designated proxy signer can create a valid proxy signature on behalf of the original signer.
3. **Strong identifiability:** Anyone can determine the identity of the proxy signer of the corresponding proxy signature.
4. **Strong undeniability:** Once a proxy signer creates a valid proxy signature on behalf of an original signer, he cannot repudiate the signature creation against anyone else.
5. **Prevention of misuse:** The proxy signer cannot use the proxy key for the purposes other than generating a valid proxy signature. In case of misuse, the responsibility of the proxy signer should be determined explicitly.

3. THE NEW PROXY PROTECTED SIGNATURE SCHEME

In this section, we propose a new proxy protected signature scheme, which is based on discrete logarithm problem [14] having different form of public key and with high security. There are also three parties in our scheme: the original signer O, the proxy signer P and the verifier V. We assume that each user has a pair of private key and public key and their certificates. The system public parameters consist of a large prime number p, a large prime factor q of (p-1), elements $g \in Z_p^*$ with prime order q such that the discrete logarithm of g is unknown. The proposed scheme is divided into four phases: *Key generation, Proxy key generation, Signing phase and Signature verification phase.*

3.1 Key Generation

For the convenience of describing our work, we define the parameters as follows:

* O: the original signer
* P: the proxy signer
* p, q : two large prime number with q=(p-1)
* g: an element of order q in Z_p*
* H() : a secure one-way hash function
* ‖: denotes the concatenation of string.
* y_o: the public key of original signer with secret key of x_{o1} and x_{o2},

$$y_o = g^{x_{o1}+x_{o2}} \bmod p,$$

* y_p : the public key of proxy signer with secret key of x_{p1} and x_{p2},

$$y_p = g^{x_{p1}+x_{p2}} \bmod p,$$

* y_{pr} : the public key of P's proxy signature with secret key of x_{pr1} and x_{pr2},

$$y_{pr} = g^{x_{pr1}+x_{pr2}} \bmod p,$$

* m_w : a parameter which includes certificate of public key of the original signer and the proxy signer and the duration of delegation.

3.2 Proxy Key Generation

The original signer O interacts with the proxy signer P and a pair of private and public keys are generated for the proxy signer.

1. The original signer O chooses two random numbers $k_{o1,}k_{o2} \in Z_q *$ and $k_{o1,}k_{o2} \neq 1,$ and computes following equations:

$$r_o = g^{k_{o1}+k_{o2}} \bmod p \tag{1}$$

$$x'_{pr1} = x_{o1} + k_{o1} \times H(m_w \| r_o) \bmod q \tag{2}$$

$$x'_{pr2} = x_{o2} + k_{o2} \times H(m_w \| r_o) \bmod q \tag{3}$$

$$y_{pr} = g^{x'_{pr1}+x'_{pr2}} \bmod p \tag{4}$$

2. The original signer O sends (x'_{pr1}, x'_{pr2}) to the proxy signer P in a secure way and makes ($y_{pr,}$ m_w, r_o) public.

3. The proxy signer P computes the proxy keys as:

$$x_{pr1} = x'_{pr1} + x_{p1} \bmod q \tag{5}$$

$$x_{pr2} = x'_{pr2} + x_{p2} \bmod q \tag{6}$$

Then P checks the validity of the proxy keys with the following congruence:

$$y_{pr} = g^{(x_{pr1}+x_{pr2})} = y_o \times y_p \times r_o^{H(m_w \| r_o)} \bmod p \tag{7}$$

If (x_{pr1}, x_{pr1}, r_o, m_w) satisfies Eq. (7), she accepts them as valid proxy keys, otherwise rejects them.

3.3 Signing Phase

In this phase, the proxy signer P signs message m for the verifier V on behalf of the original signer O.

1. The proxy signer P chooses two random numbers $k_1, k_2 \in Z_q *$ and $k_{o1,} k_{o2} \neq 1,$ and computes Eq. (8-11).

$$R = g^{(k_1+k_2)} \bmod p \tag{8}$$

$$e = H(R \| m) \bmod q \tag{9}$$

$$y_1 = k_1 + e \times x_{pr1} \bmod q \tag{10}$$

$$y_2 = k_2 + e \times x_{pr2} \bmod q \tag{11}$$

2. The proxy signer P sends (e, y_1, y_2) along with message m to the verifier V.

3.4 Verification Phase

A verifier after receiving signature (e, y_1, y_2) on the message m, to verify the validity of the public key of the proxy signer and correctness of receiving signature as given below:

1. The verifier V checks the validity of the public key of proxy signature with the following congruence

$$y_{pr} = y_o \times y_p \times r_o^{H(m_w \| r_o)} \bmod p.$$

If it holds, V accepts it as a valid proxy signer ; otherwise, it is an invalid proxy signer.

2. The verifier V checks congruence

$$e = H(g^{(y_1+y_2)} \times y_{pr}^{-e} \bmod p \| m) \bmod q.$$

If it holds, V accepts it as a valid signature; otherwise, rejects.

4 SECURITY ANALYSES

We analyze the security of our scheme as follows.

4.1 Verifiability:

The verifier of proxy signature, can check whether verification equation

$$e = H(g^{(y_1+y_2)} \times y_{pr}^{-e} \bmod p \| m) \bmod q.$$

holds or not. We prove this as follows.

$$\begin{aligned} e &= \mathrm{H(R \| m) \bmod q} \\ &= \mathrm{H(g^{k_1+k_2} \bmod p \| m) \bmod q} \\ &= \mathrm{H(g^{y_1} g.^{-e \times x_{pr1}} g^{y_2} g^{-e \times x_{pr2}} \bmod p \| m) \bmod q} \\ &= \mathrm{H(g^{y_1} g^{y_2} g^{-e(x_{pr1}+x_{pr2})} \bmod p \| m) \bmod q} \\ &= H(g^{y_1} g^{y_2} y_{pr}^{-e} \bmod p \| m) \bmod q \\ &= \mathrm{H(g^{(y_1+y_2)} \times y_{pr}^{-e} \bmod p \| m) \bmod q.} \end{aligned}$$

4.2 Unforgeability:

- Nobody can forge the signature of proxy signer, the proposed proxy protected signature is based on difficulty of solving discrete logarithm problem and forger must have (x_{pr1}, x_{pr2}) to sign message instead of the proxy signer. If any one try to find (x_{pr1}, x_{pr2}) from y_{pr}, but they will also need to solve the discrete logarithm problem. The original signer has $(x'_{pr1,}\ x'_{pr2,}\ r_o)$, but he cannot obtain ($x_{pr1}$, x_{pr2}), because ($x_{p1,}$ x_{p2}) unknown in Eq. (5, 6).

- Nobody can delegate the original signer's signing right to himself. Suppose one of the people tries to forge a proxy signature, he must obtain the secret key of the original signer from Eq. (2, 3). It is difficult to derive the value of x_{o1} and x_{o2}, since k_{o1} and k_{o2} are two random elements of Zq* . If he tries to find k_{o1} and k_{o2} from r_o, he will need to solve the discrete logarithm problem.

4.3 Undeniability:

The original signer cannot repudiate her delegated signing right to proxy signer and P cannot deny a generated valid signature on a given message on behalf of the verifier, because we saw at strong unforgeability proof only the original signer can delegate her signing right to any one and only the proxy signer can compute the proxy signing's private key used in the signature also.

4.4 Not Transferebility :

For transferring the proxy signature key from proxy signer 1 to proxy signer 2, forger should change warrant mw in Eq. (2,3). But mw includes the public key certificate of proxy signer 1 and it must be converted to the public key certificate of the proxy signer 2. However, the forger does not know x_{o1}, x_{o2}, k_{o1} and k_{o2} , therefore he must find r_{o2} such that $H(m_{w1} \| r_{o1}) = H(m_{w2} \| r_{o2})$ and it is computationally infeasible, since the hash padding technique is utilized in the proxy signature generation.

4.5 Distinguishability:

Obviously, the verifier can easily distinguish the original signer's signature from proxy signer's because they use different keys, i.e., $(x_{o1,} x_{o2}, y_o)$ for original signer O and $(x_{pr1,} x_{pr2,} y_{pr})$ for proxy signer P, and they don't access to each other's secret key.

5 CONCLUSION

In this paper, we proposed a new protected proxy signature scheme based on discrete logarithm problem having different form of public key with more efficiency and security. Our scheme satisfies all the security properties of a proxy signature.

ACKNOWLEDGEMENTS

The author wishes to thank the anonymous referees for their very useful comments and suggestions

REFERENCES

[1] A. Bakker et al., “A law-abiding peer-to-peer network for free software distribution,” *In: IEEE International Symposium on Network Computing and Applications (NCA01),* pp. 60-67, (2001).

[2] H. Chien, “Extending RSA cryptosystems to proxy multi-signature scheme allowing parallel individual signing operation,” *J Chin Inst Eng,* 29(3), 527-532, (2006).

[3] J.Z. Dai et al., “Designated-Receiver Proxy Signature Scheme for Electronic Commerce,” *In: Proc. of IEEE International Conference on Systems, Man and Cybernetics,* Vol. 1, pp.384-389, (2003).

[4] M.Gasser et al., “The digital distributed system security architecture,” *In: Proceedings of National Computer Security Conference,* pp.305-319, (1989).

[5] S. Kim et al., “Proxy signatures. Revisited,” *In: ICICS97. LNCS 1334. Springer-Verlag,* 223-232, (1997).

[6] B. Lee et al., “Secure mobile agent using strong non-designated proxy signature,” *In: Information security and private (ACISP01), LNCS 2119, Springer-Verlag,* 474-486, (2001).

[7] B. Lee et al., “Strong proxy signature and its applications,” *In: Proceeding of the 2001 symposium on cryptography and information security (SCIS01),* vol. 2(2), 603-608, (2001).

[8] J. Leiwo et al., “Disallowing unauthorized state changes of distributed shared objects,” *In: Information Security forGlobal Information Infrastructures (SEC00),* pp. 381-390, (2000).

[9] Y. Liu et al., “Proxy-protected signature secure against the undelegated proxy signature attack,” *Comput Electron Eng* 33(3), 177-185, (2007).

[10] R. Lu et al., “Designated verifier proxy signature scheme with message recovery,” *Appl Math Comput* 169(2), 1237-1246, (2005).

[11] R. Lu et al., “Designing efficient proxy signature schemes for mobile Communication,” *In: Science in China,* 51(2), pp.183-195, (2008).

[12] M. Mambo et al., “Proxy signatures for delegating sign operation,” *In: Proceeding of the 3rd ACM conference on computer and communications security (CCS96), ACM press,* 48-57, (1996).

[13] M. Mambo et al., “Proxy signatures: delegation of the power to sign messages,” *IEICE Trans Fundam,* E79-A(9), 1338-1354, (1996)

BIOGRAPHY OF AUTHORS

Swati Verma received the B.Sc. and M.Sc. degree in Mathematics form Pt. Ravishankar Shukla University, Raipur. Chhattisgarh, India in 2005 and 2007. She joined School of Studies in Mathematics, Pt. Ravishankar Shukla University, Raipur, India for her research work. She is a life member of Cryptology Research Society of India (CRSI). Her area of interest is Public Key Cryptography and Digital Signature.

Birendra Kumar Sharma Professor, School of Studies in Mathematics, Pt. Ravishankar Shukla University Raipur (C. G.) India. He has been working for long time in the field of Non Linear Operator Theory and currently in Cryptography. He and his research scholars work on many branches of public key cryptography. He is a life member of Indian Mathematical Society and the Ramanujan Mathematical Society.