A Scheme for obtaining a Warrant Message from the Digital Proxy Signatures

Sunder Lal[•] and Amit Kumar Awasthi

Institute of Basic Sciences,
Khandari, Agra – 282002 (UP), INDIA
Hindustan College of Science and Technology
Farah, Mathura – 281122 (UP), INDIA

Email: awasthi_hcst@yahoo.com

ABSTRACT

Mambo et al [6-7] introduced a proxy signature scheme. Neuman [8] extended the scheme for delegation by warrant, which was further extended by Kim et al [4] to partial delegation with a warrant. In this paper we propose a new type of digital proxy signature scheme in which the warrant message can be recovered from the proxy signature. In this scheme the warrant message is conveyed within the proxy signature and recovered by the verifier, i.e., the warrant need not be hashed or sent along with the proxy signature. It saves both communication bandwidth and storage space.

Keywords: *Proxy signature, Warrant message, Message recovery, Hash function*

INTRODUCTION

Mambo et al. [6-7] developed a systematic approach to proxy or delegated signatures. Neuman [8] introduced the scheme for delegation by warrant, which was further extended by Kim et al [4] to partial delegation by warrant. Till now we have the following types of delegations in literature -

Full delegation – This happens when the original signer Alice gives its private key 'x_{Alice}' to the proxy signer Bob. In this case a proxy signature by Bob is indistinguishable from the original signature of Alice.

Partial delegation – This happens when the original signer 'Alice' computes a proxy key 'σ' from her private key 'x_{Alice}' and gives it to the proxy signer in a secure way. In this case the proxy signature by Bob is distinguishable from the original signature created by Alice.

Delegation by warrant – warrant is a certificate composed of a message part that the proxy signer is authorised to sign and a public key which ensures the involvement of original signer.

Partial delegation with warrant – in this delegation proxy key is computed by original signer Alice's private key 'x_{Alice}' and warrant message issued to her. The proxy signature scheme of Kim et al.[4] included a warrant in their proxy signature key.

In this paper we propose a scheme by which the warrant message can be recovered directly from the proxy signature. There is no need to append warrant message with signed message by proxy signer 'Bob'. This scheme saves storage space and communication bandwidth both.

[•] Research partially supported by UGC grant No. 8-9/98(SR-I)

Throughout this paper, it is assumed that an original signer Alice delegates a proxy key to a designated proxy signer Bob to sign on her behalf. Carol is the verifier.

RELATED WORK

Throughout this paper p denotes a large prime with $2^{511} < p < 2^{512}$ and g denotes a generator for Z_p^*. Each user selects a secret key $x_u \in Z_q$ and computes a public key $y_u = g^{x_u} \bmod p$, where Before introducing our proposed scheme it is necessary to introduce a following two related works. i.e. Kim et al.' scheme [4] and Nyberg and Reuppel's scheme [9]

Nyberg Reuppel's Signature Scheme **[9]**

(*Signature Generation*)

To sign a message $M \in Z_p$ the signer selects a random number $k \in Z_q$ and computes

$$R = M \cdot g^k \bmod p, \text{ and}$$

$$S = R x + k \bmod q$$

where x is the secret key of signer.The pair (R, S) is the signature on the message 'M'.

(*Message recovery & Verification*)

To verify the validity of a signature one checks that following equality holds.

$$M = g^{-S} y^R R \bmod p$$

Because this scheme provides message recovery, the signature need not to be accompanied by the message.

*Kim's scheme for Partial Delegation***[4]**

Basic protocol consists of the following:

(*Proxy key generation phase*)

1. (Proxy generation) – An original signer Alice chooses a random number $k \in Z_q$, $k \neq 1$ and computes

   $$K = g^k \bmod p \quad \text{and} \quad e = h(m_w, K).$$

   where m_w is warrant message having the information about delegation and h is publicly known hash function. Now she computes

   $$\sigma = e\, x_{Alice} + k \bmod q$$

2. (Proxy delivery) – Alice gives (m_w, σ, K) to a proxy signer, Bob, in a secure way.

3. (Proxy verification) – Bob checks, that

   $$e = h(m_w, K), \text{ and } g^{\sigma} = y^{e}_{Alice} K \bmod p$$

   he accepts σ as proxy secret key if above congruence holds.

(*Signing by the proxy signer*)

To sign a message M, Bob uses 'σ' as proxy key and executes the ordinary signing operation. Then the proxy signature on M is $(m, m_w, Sign_{\sigma}(M), K)$. where $Sign_{\sigma}(M)$ refers to signing a message M with private key σ.

(*Verification of the proxy signatures*)

The verification is similar to verification in the original signature scheme except for extra computation

$$e = h(m_w, K) \qquad \text{and} \qquad y' = y^e_{Alice} \cdot K \bmod p.$$

The value y' is dealt with as new public key, which shows the involvement of Alice.

PROPOSED SCHEME

In this section, we propose a new digital proxy signature scheme with message recovery. This scheme allows an original signer to sign some warrant and compute a proxy key and send it to a proxy signer. With this proxy key proxy signer is able to create a valid proxy signature. The receiver can verify the validity of the proxy signature and he is also able to recover the warrant message from this proxy signature. System parameters are same as given previously –

Basic protocol:

1. (Proxy generation) – The original signer Alice chooses a random number $k \in Z_q$ and computes $r = m_w\, g^k \bmod p$, where m_w is warrant message having the information about delegation. After that she computes

   $$\sigma = r\, x_{Alice} + k \bmod q$$

2. (Proxy delivery) – Alice gives (σ, r) to the proxy signer, Bob, in a secure way.

3. (Proxy verification) – Bob accepts proxy iff he confirms

   $$m_w = g^{-\sigma} (y_{Alice})^r\, r \bmod p$$

4. (Signing by the proxy signer) – For signing a message m, Bob uses 'σ' as proxy key and executes the ordinary signing operation. He computes-

   $$R = g^K \bmod p$$

   $$R' = h\,(m,\, \mathrm{R})$$

   $$Sign_\sigma(m) = (\sigma + \mathrm{K}\, \mathrm{R}' \bmod q\ ,\ \mathrm{R})$$

   The proxy signature on m is $(m, Sign_\sigma(m), r)$.

5. (Verification and warrant recovery of the proxy signatures) – The verifier Carol first computes $\mathrm{R}' = h(m, \mathrm{R})$ and then verifies $m_w = g^{-S}\, Y\, R^{R'} \bmod p$, where Y is new public value. The value Y $\left(= [(y_{Alice})^r\, r]\right)$ is deals with as new public key, which shows the involvement of Alice.

Because this scheme provides message recovery, the proxy signature needs not to be accompanied by the warrant message. The previous proxy signature schemes were not able to recover warrant message from a proxy signature. In those schemes proxy signed message is appended with warrant message. Verification of the equation is possible only if the warrant message is known.

In the verification procedure the hash of message is computed first then the hash value is entered in to verification equation. The validity of the proxy signature is established through checking the verification equation.

For our warrant recovery scheme the process runs differently. The verification equation recovers the warrant message itself.

As discussed in Nyberg's scheme to ensure that recovered message is the correct one, the original signer adds some redundancy to the warrant message, before it is signed, and for verification the redundancy after recovery checked.

This scheme as other message recovery schemes can be used in text hashing mode. i.e. warrant message is hashed and hash value is signed. At Carol's end the hash value is recovered (using warrant recovery feature of our scheme) and the authenticity of the message

is verified through comparison of such recovered hash value with the locally computed hash value of the message.

ILLUSTRATION

Let the system parameters be p = 23, q = 11, g = 3, x_{Alice} = 3, y_{Alice} = 4

Proxy generation Phase

1. (Proxy generation) – The original signer Alice chooses a random number

   $9 \in Z_q$ and computes $r = m_w\, g^k \bmod p$

   Let $m_w = 7$ (say). Then $r = 7 \times 3^9 \bmod 23 = 11$

   Here m_w is warrant message having the information about delegation. Now Alice computes $\sigma = r\, x_{Alice} + k \bmod q$

   $= 11 \times 3 + 9 \bmod 11 = 42 \bmod 11 = 9$

2. (Proxy delivery) – Alice gives (9, 11) to the proxy signer Bob in a secure way.
3. (Proxy verification) – Bob computes $g^{-\sigma} (y_{Alice})^r r \bmod p$

   $= 3^{-9} \times 4^{11} \times 11 \bmod 23 = 7$

   and checks that is equal to m_w (= 7). He, therefore accepts σ = 9 as proxy secret key.

Signing by the proxy signer

For signing a message *m*, Bob uses '9' as proxy key and executes the ordinary signing operation. He chooses a random number K = 3

$R = g^K \bmod p = 3^3 \bmod 23 = 4$

$R' = h(m, R) = h(m, 4) = 11$ (say)

Finally he computes

$Sign_\sigma(m) = (9 + 3 \times 11 \bmod 11, 4) =$ **(9, 4)**

Thus the proxy signature on *m* is **(*m*, (9, 4), 11)**.

Verification and warrant recovery of the proxy signatures –

To verify Carol checks $R' = h(m, R) = h(m, 4) = 11$

and computes $= g^{-S}\, Y R^{R'} \bmod p = g^{-S} [(y_{Alice})^r r]\, R^{R'} \bmod p$,

$= 3^{-9} [4^{11} \times 11]\, 4^{11} \bmod 23 = 7$

where (g = 3, S = 9, y_{Alice} = 4, r = 11, R = 4, R′ =11) and $Y \left(= [(y_{Alice})^r r]\right)$

the warrant message '7' recovered. Hence the scheme.

SECURITY DISCUSSION

To forge a signature for a given message without the knowledge of the private key is same as to solve (*r, s*) from the verification equation. Hence the security depends on the difficulty of the following problem.

For given $g \in Z_p$, $y \in Z_p$ and $m_w \in Z_p$ find $r \in Z_p$ and $s \in Z_q$ such that warrant recovery equation $m_w = g^{-S}\, Y R^{R'} \bmod p$ is satified.

Let q be prime divisor of $(p-1)$. Then for a warrant message $m_w \in Z_p$, it is as difficult as solving DLP to obtain $r \in Z_p$ *and* $s \in Z_q$ such that (r, s) is a valid proxy signature giving warrant recovery of the message [9].

CONCLUSION

We proposed a new type of digital proxy signature scheme, which allows an original signer to sign some warrant and compute a proxy key and send it to a proxy signer. With this proxy key a proxy signer can sign any message restricted with warrant. The receiver can verify the validity of the proxy signature and can recover the warrant message from this proxy signature.

By using this scheme communication cost may be reduced efficiently. Computational load in our scheme is not much different from Kim et al.'s scheme. In our scheme one multiplication is more than that of Kim's during verification phase, However, our scheme gives warrant message recovery, which reduces storage space and communication bandwidth.

REFERENCE

1. Chang C. C., Leu J. J., Huang P.C., Lee W.B. : " A scheme for obtaining a message from the digital multi-signature", LNCS #1431, pp 154-163.
2. ElGamal, [1985] "A public-key cryptosystem and a signature scheme based on discrete logarithms," IEEE Trans. Inf. Theory, vol. IT–31, no. 4, pp 469-472.
3. Harn, L. [1994]: "New digital signature scheme based on discrete logarithms" Electronic Letters, Vol 30, No.5, pp 396-397.
4. Kim S., Park S., Won D, [1997], "Proxy Signatures, Revisited" in proceedings of ICICS'97, Springer Verlag, LNCS 1334, pp 223 – 232.
5. Lee N., Hwang T., Wang C., [1998]: "On Zhang's nonrepudiable proxy signature scheme" in proceeding of ACISP'98 (Australasian Conference on Information Security & Privacy), (C Boyd and E Dawson, eds.), Vol 1438, LNCS, pp 415-422.
6. Mambo, M., Usuda K. and Okamoto E. [1996], "Proxy signatures for delegating signing operation". Proc. 3rd ACM Conference on Computer and Communications Security. pp 48 – 57.
7. Mambo M, Usuda K, Okamoto E, [1996] "Proxy Signature: Delegation of the power to sign messages," IEICE Trans. Fundamentals, E79 – A: 9 1338 – 1353
8. Neuman B C, [1993] "Proxy-based authorization and accounting for distributed systems," Proc. 13th International conference on Distributed Computing systems, pp 283 – 291.
9. Nyberg K., Rueppel R.,[1994] "Message recovery for signature schemes based on discrete logarithm problems," Pre-proceeding of Eurocrypt'94. pp 175-190.
10. Yen S. M. and Laih C. S.,[1993] " New digital signature scheme based on discrete logarithm", Electronic Letters, Vol 29, No. 12, pp 1120-1121.
11. Zhang K.[1997]: "Threshold proxy signature schemes", Information security workshop, Japan, pp 191-197.