FORWARD-SECURE PROXY SIGNATURE AND REVOCATION SCHEME FOR A PROXY SIGNER WITH MULTIPLE ORIGINAL SIGNERS

B. B. Amberker
Department of Computer Science & Engg., National Institute of Technology, Warangal, Andhra Pradesh, India

N. R. Sunitha
Department of Computer Science & Engg., Siddaganga Institute of Technology, Tumkur, Karnataka, India

Keywords: Proxy Signature, Forward-Security, Digital Signature, Proxy Key Pair, Key Evolution, Proxy Revocation.

Abstract: On many occasions it is required for a single person to take up the responsibilities of many persons for some duration and work on their behalf so that the regular work goes on smoothly. For example in a bank, when majority of the employees need to attend an important meeting during working hours, to avoid disrupting any of the regular activities, one employee may need to play the role of many employees. An accountant may need to play the role of a cashier, asst. manager and cheque clearing officer. In such situations the employee working on behalf of other employees need to be delegated with signing power from the employees who perform the activities regularly. Also, this delegation must be only for some specified time period T and after the elapse of that time period the signing capability must be revoked. The concept of proxy signatures is used here.
A proxy signature scheme allows one user to delegate his/her signing capability to another user called a proxy signer in such a way that the latter can sign messages on behalf of the former. After verification the verifier is convinced of the original signer's agreement on the signed message. Forward-Secure signatures enable the signer to guarantee the security of messages signed in the past even if his secret key is exposed today. We have come up with a forward secure proxy signature and revocation scheme for a proxy signer who is delegated with signing power from multiple original signers. This scheme is based on the popular Bellare-Miner Forward-secure scheme.

1 INTRODUCTION

A proxy signature (M. Mambo and Okamoto, 1996; M. Mambo, 1996) allows one user Alice, called the original signer, to delegate her signing capability to another user Bob, called the proxy signer. After that, the proxy signer Bob can sign messages on behalf of the original signer Alice. Upon receiving a proxy signature on some message, a verifier can validate its correctness by the given verification procedure. By this the verifier is convinced of the original signer's agreement on the signed message. Proxy signatures can be used in a number of applications like e-cash, electronic commerce, mobile computing distributed shared object systems etc.

The basic working of most proxy signature schemes is as follows. The original signer Alice sends a specific message with its signature to the proxy signer Bob, who then uses this information to construct a proxy private key. With the proxy private key, Bob can generate proxy signatures by employing a specified standard signature scheme. When a proxy signature is given, a verifier first computes the proxy public key and then checks its validity according to the corresponding standard signature verification procedure.

Mambo, Usuda and Okamoto introduced the concept of proxy signatures and proposed several constructions in (M. Mambo and Okamoto, 1996). Based on the delegation type, they classified proxy signatures as full delegation, partial delegation and delegation by warrant schemes. In full delegation, Alice's private key is given to Bob so that Bob has the same signing capability as Alice. But such schemes are obviously impractical and insecure. In a partial delegation scheme, a proxy signer has a new key called proxy private key, which is different from Alice's private key. So, proxy signatures generated by using

B. Amberker B. and R. Sunitha N. (2008).
FORWARD-SECURE PROXY SIGNATURE AND REVOCATION SCHEME FOR A PROXY SIGNER WITH MULTIPLE ORIGINAL SIGNERS.
In *Proceedings of the International Conference on Security and Cryptography*, pages 245-252
DOI: 10.5220/0001919302450252

proxy private key are different from Alice'e standard signatures. However the proxy signer can sign any message of his choice i.e there is no limit on the range of messages he can sign. This limitation is eliminated in delegation by warrant schemes by adding a warrant that specifies what kind of messages are delegated and may contain the identities of Alice and Bob, the delegation period, etc.

Followed by the first constructions given in (M. Mambo and Okamoto, 1996; M. Mambo, 1996), a number of new schemes and improvements have been proposed (S. Kim and Won., 1997; Zhang., 1997; Zhang, 1997; N.-Y. Lee and Wang, 1998; Ghodosi and Pieprzyk, 1999; T. Okamoto and Okamoto, 1999; B. Lee and Kim, 2001b; B. Lee and Kim, 2001a; Park and Lee, 2001; J.-Y. Lee and Kim, 2003; Wang and Pieprzyk., 2003; A. Boldyreva and Warinschi, 2003); however, most of them do not fully meet the security requirements of a proxy signature scheme (see Section 2) . In (S. Kim and Won., 1997), Kim, Park and Won proposed a threshold proxy signature, in which the original signing power is shared among a delegated group of n proxy signers such that only t or more of them can generate proxy signatures cooperatively. In (B. Lee and Kim, 2001b), Lee, Kim and Kim proposed non-designated proxy signature in which a warrant does not designate the identity of a proxy signer so any possible proxy signer can respond to this delegation and become a proxy signer. Furthermore, their scheme is used to design secure mobile agents in electronic commerce setting (B. Lee and Kim, 2001a). One-time proxy signatures are studied in (Ai-Ibrahim and Cerny, 2003; Wang and Pieprzyk., 2003). In (J.-Y. Lee and Kim, 2003), Lee, Cheon, and Kim investigated whether a secure channel for delivery of a signed warrant is necessary in existing schemes. Their results show that if the secure channel is not provided, the MUO scheme (M. Mambo and Okamoto, 1996) and the LKK scheme (B. Lee and Kim, 2001b; B. Lee and Kim, 2001a) are insecure. To remove the requirement of a secure channel and overcome some other weaknesses, they revised the MUO and LKK schemes (M. Mambo and Okamoto, 1996; M. Mambo, 1996; B. Lee and Kim, 2001b). In contrast to the above mentioned schemes, which all are based on discrete logarithm cryptosystems, several RSA-based proxy signature schemes are proposed in (T. Okamoto and Okamoto, 1999; B. Lee and Kim, 2001a). In (Zhen Chuan Chai, 2004) a factorisation based forward-secure proxy signature scheme is proposed. The scheme is based on the forward-secure scheme of Abdalla and Reyzin.

In (Guilin Wang, 2004) a proxy signature scheme with multiple original signers suitable for wireless electronic commerce applications is proposed. When compared to this scheme, our scheme has the property of forward-security (this enables the proxy signer to guarantee the security of messages signed in the past even if his secret key is exposed today) and the proxy signer will be delegated with signing power only for a time period T. After the elapse of this time period, the proxy signer will automatically be revoked.

We consider a scenario where there is need for a single person to take up the responsibilities of many persons for some duration and work on their behalf so that the regular work goes on smoothly. For example in a bank, an accountant may need to play the role of a cashier, asst. manager and cheque clearing officer. In such situations the employee working on behalf of other employees need to be delegated with signing power from the employees who perform the activities regularly. This can be addressed using regular proxy signatures. In terms of proxy signatures, the problem we have considered requires a single proxy signer to sign on behalf of multiple original signers. Regular proxy signature force the proxy signer to generate separate proxy key pair for each original signer. In the scheme we propose the proxy signer just computes a single proxy key pair for n original signers. Also, as digital signatures, proxy signatures are also vulnerable to leakage of proxy secret key. If the proxy secret key is compromised, any message can be forged. To prevent future forgery of signatures, the concept of forward-security (Anderson, 1997) can be used (see section 3). We use the property of forward-security and apply it to proxy signatures. We therefore propose a new Forward-secure proxy signature and revocation scheme for a proxy signer with multiple original signers which is based on the popular Forward-secure Bellare-Miner scheme (Bellare, 1999). The scheme has the following features:

- The scheme is based on Forward-secure Bellare-Miner scheme.
- Multiple original signers can delegate signing power to a single proxy signer.
- Proxy signer is capable of signing on behalf of original signers only for a time period T, after which he is revoked as a proxy signer.
- Identity of the proxy signer is available in the information sent by original signer to proxy signer.
- Secure channel is not required to send the information to proxy signer.
- There is a facility to send warrant messages to proxy signer and verifier.
- Original signer cannot play the role of proxy signer.

- Verifier can determine when the proxy signature was generated.
- Both the original signer's signature and proxy signer's signature are made Forward-secure.

The organisation of our paper is as follows: In Section 2, we discuss the basic security requirement of any proxy signature scheme. In Section 3, we describe briefly the properties of forward-secure signature schemes. In Section 4, we describe our proxy signature scheme. In Section 5, we discuss the security of our scheme and in Section 6, we conclude the paper.

2 SECURITY REQUIREMENTS OF A PROXY SIGNATURE SCHEME

Any secure proxy signature scheme should satisfy the following five requirements:

1. Verifiability. From the proxy signature, a verifier is convinced of the original signer's agreement on the signed message.
2. Strong Uunforgeability. Only 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 corresponding proxy signer from the 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. Proxy Signer's Deviation. A proxy signer cannot create a valid signature not detected as a proxy signature.

3 FORWARD SECURE SIGNATURE SCHEME

Digital signatures are vulnerable to leakage of secret key. If the secret key is compromised, any message can be forged. To prevent future forgery of signatures, both public key and secret key must be changed. Notice, that this will not protect previously signed messages: such messages will have to be re-signed with new pair of public key and secret key, but this is not feasible. Also changing the keys frequently is not a practical solution.

To address the above problem, the notion of forward security for digital signatures was first proposed by Anderson in (Anderson, 1997), and carefully formalised by Bellare and Miner in (Bellare, 1999) (see also(Abdalla, 1997; Krawczyk, 2000; Itkis, 2001; Kozlov, 2002)). The basic idea is to extend a standard digital signature scheme with a key updation algorithm so that the secret key can be changed frequently while the public key stays the same. Unlike a standard signature scheme, a forward secure signature scheme has its operation divided into time periods, each of which uses a different secret key to sign a message. The key updation algorithm computes the secret key for the new time period based on the previous one using a one way function. Thus, given the secret key for any time period, it is hard to compute any of the previously used secret keys. (It is important for the signer to delete the old secret key as soon as the new one is generated, since otherwise an adversary breaking the system could easily get hold of these undeleted keys and forge signatures.) Therefore a receiver with a message signed before the period in which the secret key gets compromised, can still trust this signature, for it is still hard to any adversary to forge previous signatures.

To specify a forward-secure signature scheme, we need to (i) give a rule for updating the secret key (ii) specify the public key and (iii) specify the signing and the verification algorithms.

4 FORWARD-SECURE PROXY SIGNATURE AND REVOCATION SCHEME FOR A PROXY SIGNER WITH MULTIPLE ORIGINAL SIGNERS

As digital signatures, proxy signatures are also vulnerable to leakage of proxy secret key. If the proxy secret key is compromised, any message can be forged. To prevent future forgery of signatures, both proxy public key and proxy secret key must be changed which forces the original signer to change the proxy information. But this will not protect previously signed messages: such messages will have to be re-signed with new pair of proxy public key and secret key which is not feasible. To address this problem, we use the concept of forward security for proxy signatures.

The basic idea behind the construction of our scheme is as follows: There are n persons in the role of original signer who wants Bob to be the

proxy signer for T time periods. Each original signer A_x(where $x = 1,\ldots,n$) computes $P_{A_x,j,info}$ in every time period j (which ranges from 1 to T) and sends it along with other proxy information to Bob. Bob verifies that he is designated as a proxy signer by each of the original signers. If the verification holds Bob computes the proxy key pair $(x_{p,0}, y_p)$. Bob divides the time period j into T' time periods and signs any messages in these j' time periods (which ranges from 1 to T') using Forward-secure signatures (see Figure 1). Thus the proxy signatures generated by Bob are Forward-secure proxy signatures. Bob will be able to generate proxy signatures on behalf of the original signers only for T time periods. After the elapse of this time period, he is automatically revoked as a proxy signer. On receiving the proxy signatures, the verifier first computes the proxy public key using the available proxy information. Using this public key later verifies the signature using the verification equation of the forward-secure proxy signature scheme.

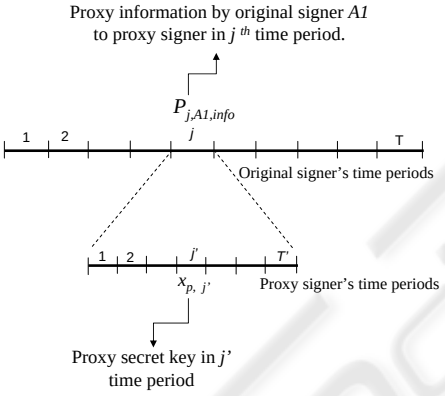

Figure 1: Basic idea of the scheme.

4.1 Initial Setup

Let p,q be two large primes each congruent to 3 mod 4. Let $N = p.q$.

T is the time period in which the signing power is delegated by original signers to proxy signer.

The initial secret key of original signer A_1 is $SK_{A_1,0} = (SK_{A_1,(1,0)},\ldots,SK_{A_1,(l,0)},N,0)$ where $S_{A_1,(i,0)} \overset{R}{\leftarrow} Z_N^*$.

Secret key $SK_{A_1,j} = (SK_{A_1,(1,j)},\ldots,SK_{A_1,(l,j)})$ for any time period j is obtained by updating the secret key $SK_{A_1,j-1} = (SK_{A_1,(1,j-1)},\ldots,SK_{A_1,(l,j-1)})$ of the previous time period via the update rule

$$SK_{A_1,(i,j)} = SK_{A_1,(i,j-1)}^2 \bmod N, \quad (1)$$

where $i = 1,\ldots,l$.

The public key is $U_{A_1} = (U_{A_1,1},\ldots,U_{A_1,l})$, is calculated as the value obtained on updating the base secret secret key $T+1$ times:

$$U_{A_1,i} = SK_{A_1,(i,0)}^{2^{T+1}} \bmod N. \quad (2)$$

Let there be n number of original signers. The initial secret key of any original signer A_x is $SK_{A_x,0} = (SK_{A_x,(1,0)},\ldots,SK_{A_x,(l,0)},N,0)$ where $S_{A_x,(i,0)} \overset{R}{\leftarrow} Z_N^*$ and $x = 1,\ldots,n$.

Secret key $SK_{A_x,j} = (SK_{A_x,(1,j)},\ldots,SK_{A_x,(l,j)})$ for any time period j is obtained by updating the secret key $SK_{A_x,j-1} = (SK_{A_x,(1,j-1)},\ldots,SK_{A_x,(l,j-1)})$ of the previous time period via the update rule

$$SK_{A_x,(i,j)} = SK_{A_x,(i,j-1)}^2 \bmod N, \quad (3)$$

where $i = 1,\ldots,l$.

The public key is $U_{A_x} = (U_{A_x,1},\ldots,U_{A_x,l})$, is calculated as the value obtained on updating the base secret secret key $T+1$ times:

$$U_{A_x,i} = SK_{A_x,(i,0)}^{2^{T+1}} \bmod N. \quad (4)$$

Bob is a proxy signer. His initial secret key is $SK_{B,0} = (SK_{B,(1,0)},\ldots,SK_{B,(l,0)},N,0)$ where $SK_{B,(i,0)} \overset{R}{\leftarrow} Z_N^*$.

Secret key $SK_{B,j} = (SK_{B,(1,j)},\ldots,SK_{B,(l,j)})$ for any time period j is obtained by updating the secret key $SK_{B,j-1} = (SK_{B,(1,j-1)},\ldots,SK_{B,(l,j-1)})$ of the previous time period via the update rule

$$SK_{B,(i,j)} = SK_{B,(i,j-1)}^2 \bmod N, \quad (5)$$

where $i = 1,\ldots,l$.

The public key $U_B = (U_{B,1},\ldots,U_{B,l})$, is calculated as the value obtained on updating the base secret secret key $T+1$ times:

$$U_{B,i} = SK_{B,(i,0)}^{2^{T+1}} \bmod N \quad (6)$$

4.2 Proxy Generation

Each original signer A_x generates the proxy information in time period j as follows:

$$Y_x = R_x^{2^{T+1-j}} \bmod N$$

where $R_x \overset{R}{\leftarrow} Z_N^*$ and $x = 1,\ldots,n$.

$$P_{j,A_x,info} = R_x.\prod_{i=1}^{l} U_{B,i} \prod_{i=1}^{l} SK_{A_x,(i,j)}^{c_i} \bmod N$$

where $c_1,\ldots,c_l \leftarrow H(M_w,Y,j)$, H is a collision resistant hash function and M_w is the message for proxy signer and the verifier (which may include warrants).

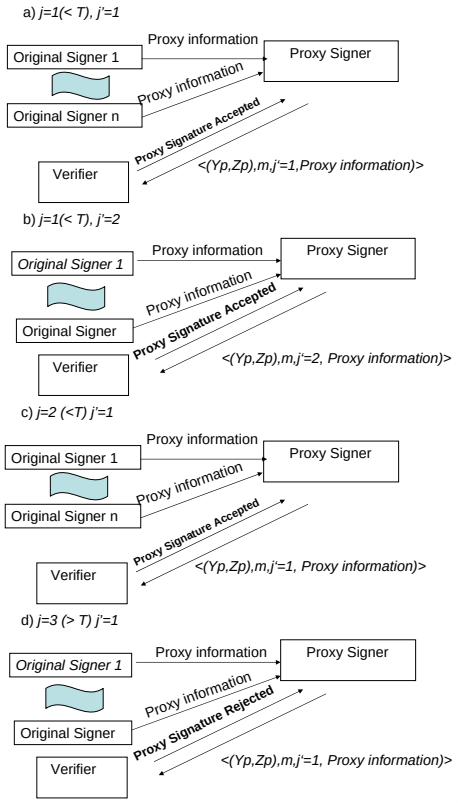

Figure 2: Proxy Signature Model for a single proxy signer with multiple original signers.

4.3 Proxy Delivery

Each original Signer A_x delegates his/her signing capability to Bob by giving the following information, $(M_w, Y_x, P_{j,A_x,info}, U_{A_x}, U_B)$.

This information also helps to identify the original signer and the proxy signer. Once the verification of this signature for a given message passes with the computation of proxy public key, the identity of the original signer and the proxy signer is confirmed. Thus the third requirement, Strong identifiability, of a secure proxy signature is satisfied.

4.4 Proxy Verification

For each original signer Bob checks whether he is the proxy signer of original signer A_x by using the following equation:

$$P_{j,A_x,info}^{2^{(T+1-j)}} = Y_x . \prod_{i=1}^{l} U_{A_x,i}^{c_i} . \prod_{i=1}^{l} U_{B,i}^{2^{(T+1-j)}} \mod N \qquad (7)$$

where $x = 1, \ldots, n$.

Notice that since

$$\begin{aligned} LHS &= (R_x . \prod_{i=1}^{l} U_{B,i} \prod_{i=1}^{l} SK_{A_x,(i,j)}^{c_i})^{2^{(T+1-j)}} \mod N \\ &= R_x^{2^{(T+1-j)}} . (\prod_{i=1}^{l} U_{B,i})^{2^{(T+1-j)}} . (\prod_{i=1}^{l} SK_{Ax,(i,j)}^{c_i})^{2^{(T+1-j)}} \mod N \\ &= Y_x . (\prod_{i=1}^{l} U_{B,i})^{2^{(T+1-j)}} . (\prod_{i=1}^{l} SK_{A_x,(i,0)}^{c_i . 2^j})^{2^{(T+1-j)}} \mod N \\ &= Y_x . (\prod_{i=1}^{l} U_{B,i})^{2^{(T+1-j)}} . (\prod_{i=1}^{l} SK_{A_x,(i,0)}^{c_i})^{2^{(T+1)}} \mod N \\ &= Y_x . (\prod_{i=1}^{l} U_{B,i})^{2^{(T+1-j)}} . (\prod_{i=1}^{l} U_{A_x,i}^{c_i}) \mod N \\ &= RHS \end{aligned}$$

Bob accepts the tuple $(M_w, Y_x, P_{j,A_x,info}, U_{A_x}, U_B)$ as valid proxy sent by an honest signer.

4.5 Proxy Key Generation

If the above verification is correct, Bob sets his initial proxy secret key $x_{p,0} = (x_{p,(1,0)}, \ldots, x_{p,(l,0)}, N, 0)$ in any time period j as

$$x_{p,(i,0)} = P_{j,A_1,info} \ldots P_{j,A_n,info} . (U_{A_1,i} \ldots U_{A_n,i}) . SK_{B,(i,j)} \mod N \qquad (8)$$

where $i = 1, \ldots, l$.

Proxy secret key $x_{p,j'} = (x_{p,(1,j')}, \ldots, x_{p,(l,j')})$ for any time period j' is obtained by updating the proxy secret key $x_{p,j'-1} = (x_{p,(1,j'-1)}, \ldots, x_{p,(l,j'-1)})$ of the previous time period via the update rule

$$x_{p,(i,j')} = x_{p,(i,j'-1)}^2 \mod N, \qquad (9)$$

where $i = 1, \ldots, l$.

The proxy public key is $y_p = (y_{p,1}, \ldots, y_{p,l})$, is calculated as the value obtained on updating the initial proxy secret key $T+1$ times:

$$y_{p,i} = x_{p,(i,0)}^{2^{T'+1}} \mod N \qquad (10)$$

where T' is the number of sub time periods in time period j. Note that in equation (12) the proxy private key used to generate the proxy signature is computed using the private key of the proxy signer and the public key of the original signer. This ensures that the proxy signer is creating a valid proxy signature on behalf of the original signer. He therefore cannot repudiate his signature. Thus the fourth requirement Strong undeniability, of a secure proxy signature is satisfied.

4.6 Proxy Signature Generation

The proxy signer uses Bellare-Miner Forward-secure signature scheme to generate proxy signature on any message m. The signature in time period j',

$((Y_p, Z_p), m, j, j', U_{A_1}, \ldots, U_{A_n}, P_{j,A_1,info} \ldots P_{j,A_n,info}, M_w)$ is generated as follows:
The proxy secret key is $x_{p,j'}$.

$$Y_p = R_p^{2^{(T+1-j')}} \bmod N \tag{11}$$

where $R_p \overset{R}{\leftarrow} Z_N^*$

$$Z_p = R_p . \prod_{i=1}^{l} x_{p,(i,j')}^{c_i} \bmod N \tag{12}$$

where $c_1, \ldots, c_l \leftarrow H(m, Y, j')$ and H is a collision resistant hash function.

The proxy signer will be able to generate proxy signatures on behalf of the original signers only for T time periods. After the elapse of this time period, he is automatically revoked as a proxy signer.

4.7 Proxy Signature Verification

The verifier receives the proxy signature $((Y_p, Z_p), m, j, j', U_{A_1}, \ldots, U_{A_n}, P_{j,A_1,info} \ldots P_{j,A_n,info}, M_w)$. He computes the proxy public key y_{p_j} for the j^{th} time period as $y_{p_j} = (y_{p,1}, \ldots, y_{p,l})$, where

$$y_{p,i} = (P_{j,A_1,info} \ldots P_{j,A_n,info} . U_{A_1,i} \ldots U_{A_n,i})^{2^{T'+1}} . U_{B,i}^{2^j} \bmod N.$$

$$\begin{aligned}
LHS &= x_{p,(i,0)}^{2^{T'+1}} \bmod N \\
&= (P_{j,A_1,info} \ldots P_{j,A_n,info} . (U_{A_1,i} \ldots U_{A_n,i}) . SK_{B,(i,j)})^{2^{(T'+1)}} \bmod N \\
&= (P_{j,A_1,info} \ldots P_{j,A_n,info} . (U_{A_1,n} \ldots U_{A_n,i}))^{2^{(T'+1)}} . (SK_{B,(i,j)})^{2^{(T'+1)}} \bmod N \\
&= (P_{j,A_1,info} \ldots P_{j,A_n,info} . (U_{A_1,i} \ldots U_{A_n,i}))^{2^{(T'+1)}} . (SK_{B,(i,0)})^{2^j . 2^{(T'+1)}} \bmod N \\
&= (P_{j,A_1,info} \ldots P_{j,A_n,info} . (U_{A_1,i} \ldots U_{A_n,i}))^{2^{(T'+1)}} . (SK_{B,(i,0)})^{2^{(T'+1)} . 2^j} \bmod N \\
&= (P_{j,A_1,info} \ldots P_{j,A_n,info} . (U_{A_1,i} \ldots U_{A_n,i}))^{2^{(T'+1)}} . U_{B,i}^{2^j} \bmod N \\
&= RHS
\end{aligned}$$

If the above equation holds then the received signature is verified as follows:

$$Z_p^{2^{(T'+1-j')}} = Y_p . \prod_{i=1}^{l} (y_{p,i})^{c_i} \bmod N \tag{13}$$

Notice that since

$$\begin{aligned}
LHS &= (R_p . \prod_{i=1}^{l} x_{p,(i,j')}^{c_i})^{2^{(T'+1-j')}} \bmod N \\
&= R_p^{2^{(T+1-j')}} . (\prod_{i=1}^{l} x_{p,(i,j')}^{c_i})^{2^{(T'+1-j')}} \bmod N \\
&= Y_p . \prod_{i=1}^{l} (x_{p,(i,0)}^{c_i})^{2^j . 2^{(T'+1-j')}} \bmod N \\
&= Y_p . \prod_{i=1}^{l} (x_{p,(i,0)}^{c_i})^{2^{(T'+1)}} \bmod N \\
&= Y_p . \prod_{i=1}^{l} y_{p,i}^{c_i} \bmod N \\
&= RHS
\end{aligned}$$

the signature $((Y_p, Z_p), m, j, j', U_{A_1}, \ldots, U_{A_n}, P_{j,A_1,info} \ldots P_{j,A_n,info}, M_w)$ sent by an honest proxy signer will be accepted.

If this check passes, the verifier is convinced of the original signer's agreement on the signed message as the public key used to verify the signature is calculated using the public key and the proxy information sent by the original signer. Thus the first requirement, Verifiability, of a secure proxy signature is satisfied.

The verification passes only for the signatures signed within the time period T. If the proxy signer tries to sign after the time period T, the signature is rejected as he is now a revoked signer.

Also, the signature is identified as a proxy signature and not as an ordinary signature as it is verified only by the proxy public key (y_p) and not by the public key of the proxy signer (U_B). Thus the fifth requirement, that a proxy signer cannot create a valid proxy signature not detected as a proxy signature, of a secure proxy signature is satisfied.

In Figure 2.a., all the n original signers send the proxy information to the proxy signer in time period $j = 1$. This time period j is divided into T' time periods and in the first time period *i.e* $j' = 1$, the proxy signer generates Forward-secure proxy signatures on message m and sends it along with other information to verifier. The verifier first computes the proxy public key using the available proxy information and later verifies the validity of the signature. The proxy signature generated by a honest proxy signer is always accepted.

In Figure 2.b., in time period $j = 1$ & $j' = 2$, the proxy signer generates Forward-secure proxy signature on message m using the same proxy information received earlier and sends it to the verifier for acceptance. This procedure repeats until $j' = T'$ after which j gets incremented by 1.

In Figure 2.c., we observe the communication

among the players of the proxy signature model in time period $j = 2$ & $j' = 1$ and in Figure 2.d., we observe the working of the model in time period $j = 3$ & $j' = 1$, where $j > T$. Here the proxy signer is generating proxy signatures after the time period T. The verification fails and thus the verifier rejects the proxy signature.

5 SECURITY OF OUR SCHEME

1. Forgery by the Original Signer: The proxy secret key is dependent on both the proxy information sent by the original signer as well as the secret key of the proxy signer. Therefore the original signer cannot generate the proxy secret key. He also cannot derive the proxy secret key from the proxy public key given by equation (14) as it is difficult to factorise the Blum William's integer N . Thus the original signer is unable to sign like the proxy signer. Therefore forgery by original signer is computationally not possible.
2. Impersonating attack: Let us assume that Bob is not designated as a proxy signer by the original signer $A1$. Though Bob can generate a proxy key pair (x'_p, y'_p) satisfying equations (12 and 14) and sign a message on behalf of a original signer, the verifier on receiving the signatures, can first verify the proxy information using the public key of $A1$ and later use it to compute the proxy public key. If Bob changes the proxy information, the verification of proxy information will fail and the verifier rejects the proxy signature. Thus Bob cannot become the proxy signer unless he is designated by the original signer Alice.
3. Framing attack: In this attack, a third party Charlie forges a proxy private key and then generates valid proxy signatures such that the verifier believes that these proxy signatures were signed by the proxy signer Bob on behalf of the original signer Alice. When such a proxy signature is presented, Alice cannot deny that she is the original signer of the proxy signer Bob. The result is that Alice and Bob will be framed.

   To accomplish this attack, Charlie needs to forge Bob's proxy key pair (x_p, y_p). As forward-secure signatures are used by proxy signer it is computationally difficult to forge the proxy secret key. Knowing the proxy public key y_p Charlie cannot generate the proxy private key given by equation (14) as it is difficult to factorise the Blum William's integer N.

   Thus our scheme withstands the above attacks. By this we can say that only the designated proxy signer can create a valid proxy signature on behalf of the original signer. In other words, the original signer and other third parties who are not designated as proxy signer cannot create a valid signature. Thus the second requirement, Strong unforgeability, of a secure proxy signature is satisfied.

6 CONCLUSIONS

We have considered a scenario where there is need for a single person to take up the responsibilities of many persons and work on their behalf. We propose a forward-secure proxy signature scheme which helps many original signers to delegate their signing power to one proxy signer. These forward-Secure proxy signatures guarantee the security of messages signed in the past even if the proxy signer's secret key is exposed today. Also, the proxy signer is required to generate just a single proxy key pair to sign new messages belonging to any of the original signers. The proxy signer will be able to generate proxy signatures on behalf of the original signers only for T time periods. After the elapse of this time period, he is automatically revoked as a proxy signer. The new scheme proposed is based on the popular forward-secure Bellare-Miner scheme. The scheme meets the basic requirements of a proxy signature scheme and certain additional properties which make the system more flexible and secure.

REFERENCES

A. Boldyreva, A. P. and Warinschi, B. (2003). Secure proxy signature schemes for delegation of signing rights. In *Fourth Annual Conference on Computer and Communications Security*. Available at http://eprint.iacr.org/2003/096.

Abdalla, M., R. L. (1997). A new forward-secure digital signature scheme. In *ASIACRYPT 2000*. LNCS 1976, pp. 116-129. Springer-Verlag, (2000),116-129.

Ai-Ibrahim, M. and Cerny, A. (2003). Proxy and threshold one-time signatures. In *11th International Conference Applied Cryptography and Network Security (ACNS03)*. LNCS 2846, Springer-Verlag.

Anderson, R. (1997). Invited lecture. In *Fourth Annual Conference on Computer and Communications Security*. ACM.

B. Lee, H. K. and Kim, K. (2001a). Secure mobile agent using strong non-designated proxy signature. In *Information Security and Privacy (ACISP01)*. LNCS 2119, pp. 474-486. Springer-Verlag.

B. Lee, H. K. and Kim, K. (2001b). Strong proxy signature and its applications. In *2001 Symposium on Cryptography and Information Security (SCIS01)*. Vol. 2/2, pp. 603-608. Oiso, Japan.

Bellare, M., M. S. (1999). A forward-secure digital signature scheme. In *Advances in Cryptology-Crypto 99 proceedings*. Lecture notes in Computer Science, Vol. 1666. Springer-Verlag.

Ghodosi, H. and Pieprzyk, J. (1999). Repudiation of cheating and non-repudiation of zhangs proxy signature schemes. In *Information Security and Privacy (ACISP99)*. LNCS 1587, pp. 129-134. Springer-Verlag.

Guilin Wang, Feng Bao, J. Z. D. R. (2004). Invited lecture. In *Proxy signature scheme with multiple original signers for wireless e-commerce applications*. Vehicular Technology Conference, Vol. 5, pp 3249-3253, IEEE.

Itkis, G., R. L. (2001). Invited lecture. In *Forward-secure signatures with optimal signing and verifying*. CRYPTO'01, LNCS 2139, Springer-Verlag, 332-354.

J.-Y. Lee, J. H. C. and Kim, S. (2003). An analysis of proxy signatures: Is a secure channel necessary ? In *Topics in Cryptology - CT-RSA 2003*. LNCS 2612, pp. 68-79. Springer-Verlag.

Kozlov, A, R. L. (2002). Forward-secure signatures with fast key update. In *Security in Communication Networks (SCN 2002)*. LNCS 2576, Springer-Verlag, (2002), (241-256).

Krawczyk, H. (2000). Simple forward-secure signatures from any signature scheme. In *Proc. of the 7th ACM Conference on Computer and Communications Security (CCS 2000)*. ACM.

M. Mambo, K. Usuda, E. O. (1996). Proxy signatures for delegating signing operation. In *3rd ACM Conference on Computer and Communications Security (CCS96)*. pp. 48-57. ACM Press.

M. Mambo, K. U. and Okamoto, E. (1996). Invited lecture. In *Proxy signature: Delegation of the power to sign messages*. IEICE Trans. Fundamentals, Vol. E79-A, No. 9, pp. 1338-1353.

N.-Y. Lee, T. H. and Wang, C.-H. (1998). Nonrepudiable proxy signature schemes. In *Information Security and Privacy (ACISP98)*. LNCS 1438, pp. 415-422. Springer-Verlag.

Park and Lee, I.-Y. (2001). A digital nominative proxy signature scheme for mobile communications. In *Information and Communications Security (ICICS01)*. LNCS 2229, pp. 451-455. Springer- Verlag.

S. Kim, S. P. and Won., D. (1997). Proxy signatures, revisited. In *Information and Communications Security (ICICS97)*. LNCS 1334, pp. 223-232. Springer-Verlag.

T. Okamoto, M. T. and Okamoto, E. (1999). Extended proxy signatures for smart cards. In *Information Security Workshop (ISW99)*. LNCS 1729, pp. 247-258. Springer-Verlag.

Wang, H. and Pieprzyk., J. (2003). Efficient one-time proxy signatures. In *Asiacrypt03*. Springer-Verlag.

Zhang, K. (1997). Nonrepudiable proxy signature schemes. In *Manuscript, 1997*. Available at http://citeseer.nj.nec.com/360090.html.

Zhang., K. (1997). Threshold proxy signature schemes. In *Information Security Workshop (ISW97)*. LNCS 1396, pp. 282-290. Springer-Verlag.

Zhen Chuan Chai, Z. C. (2004). Factoring-based proxy signature schemes with forward-security. In *First International Symposium on Computational and Information Science*. LNCS 3314, pp 1034-1040, Springer Verlag.