New ID-Based Proxy Signature Scheme with Message Recovery

Eun-Jun Yoon[1,*], YongSoo Choi[2], and Cheonshik Kim[3,*]

[1] Department of Cyber Security, Kyungil University, Republic of Korea
ejyoon@kiu.ac.kr
[2] BK21 Ubiquitous Information Security, Korea University, Republic of Korea
ciechoi@korea.ac.kr
[3] Department of Computer Science, Sejong University, Republic of Korea
mipsan@paran.com

Abstract. In 2012, Singh-Verma proposed an ID-based proxy signature scheme with message recovery. Unfortunately, by giving two concrete attacks, Tian et al. showed that Singh-Verma's scheme is not secure. This paper proposes an improvement of Singh-Verma's scheme to eliminate the security problems.

Keywords: Proxy signature, Cryptanalysis, ID-based cryptography, Mobile.

1 Introduction

An ID-based message recovery signature scheme is a kind of useful lightweight signature, in which the message itself is not required to be transmitted together with a signature [1-3]. A Proxy signature scheme allows an original signer to delegate a proxy signer to sign messages on its behalf, which has found numerous practical applications such as grid computing and mobile agent systems [4,5]. In 2012 combining the advantages of ID-based message recovery signatures and proxy signature, Singh-Verma [4] proposed the first ID-based proxy signature scheme with message recovery. They proved its security in the random oracle model and believed that it can be used in wireless e-commerce, mobile agent systems and mobile communication. Unfortunately, by giving two concrete attacks, Tian et al. [5] showed that Singh-Verma's scheme is not secure. This paper proposes an improvement of Singh-Verma's scheme to eliminate the security problems.

2 Review of Singh-Verma's Signature Scheme

This section reviews the Singh-Verma's ID-based proxy signature scheme with message recovery [4]. Throughout the paper, notations are employed in Table 1.

* Corresponding authors.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 945–951, 2013.

Table 1. Notation used in scheme

$a\|\|b$	a concatenation operation of two strings a and b.
$\oplus$	a bit-wise exclusive-or computation in the binary system.
$[x]_{10}$	the decimal representation of $x \in \{0,1\}^*$.
$[y]_2$	the binary representation of $y \in Z$.
${}_{l1}\|\beta\|$	the first l_1 bits of β from the left side.
$\|\beta\|_{l_2}$	the first l_2 bits of β from the right side.
$G_1,\ G_2$	two cyclic groups of the same order q, where $\|q\| = l_1 + l_2$.
H_0, H_1, H_2	three cryptographic hash functions $\{0,1\}^* \rightarrow G_1^*,\ \{0,1\}^* \times G_2 \rightarrow Z_q,\ G_2 \rightarrow Z_q^*$.
F_1, F_2	two cryptographic hash functions $\{0,1\}^{l_2} \rightarrow \{0,1\}^{l_1},\ \{0,1\}^{l_1} \rightarrow \{0,1\}^{l_2}$.

There are eight phases in Singh-Verma's scheme: (1) Setup, (2) Extract, (3) Delegate, (4) DVerify, (5) PKGen, (6) PSign, (7) Verify, and (8) ID phases.

(1) Setup: Given a security parameter λ, the PKG(Private Key Generator) does the following steps:

1. Choose a random generator P of G_1 and the master secret key $s \in Z_q^*$.
2. Set $P_{pub} = sP$ as his/her public key.
3. Publish the public parameters $PP = (G_1, G_2, e, P, P_{pub}, H_0, H_1, H_2, F_1, F_2, l_1, l_2)$.

(2) Extract: On input the master secret key s and a user's identity $ID_i \in \{0,1\}^*$, the PKG computes the user's private key $d_i = sH_0(ID_i)$ and sets its public key as $q_i = H_0(ID_i)$.

(3) Delegate: The original signer ID_A does the following steps:

1. Take as input his/her private key d_A and a delegation warrant m_w.
2. Pick a random value $k_A \in Z_q^*$.
3. Compute $r_A = e(P,P)^{k_A}$, $h_A = H_1(m_w, r_A)$ and $S = h_A \cdot d_A + k_A P$.
4. Output the delegation $W_{A\rightarrow B} = (m_w, r_A, S)$.

(4) DVerify: Upon receiving $W_{A\rightarrow B} = (m_w, r_A, S)$, the proxy signer ID_B does the following steps:

1. Compute $q_A = H_0(ID_A)$, $h_A = H_1(m_w, r_A)$.
2. Check if $e(S,P) = r_A \cdot e(q_A, P_{pub})^{h_A}$.
3. If so, ID_B accepts the delegation; otherwise, he/she requests a valid one from ID_A or terminates the protocol.

(5) PKGen: After accepting $W_{A\rightarrow B}$, ID_B sets $d_p = S + h_A \cdot d_B$ as his/her proxy signing key.

(6) PSign: Given a message $m \in \{0,1\}^*$ which conforms to the warrant m_w, the proxy signer ID_B with the proxy signing key d_p does the following steps:

1. Select a random value $k_B \in Z_q^*$ and set $r_B = e(P,P)^{k_B}$.
2. Set $\beta = F_1(m)||(F_2(F_1(m)) \oplus m)$ and $\alpha = [\beta]_{10}$.
3. Compute $v = r_A \cdot r_B$ and $V_B = H_2(v) + \alpha$
4. Compute $U = k_B P + d_p$.
5. Output the proxy signature $\delta = (m_w, r_A, V_B, U)$.

(7) Verify: On input a proxy signature $\delta = (m_w, r_A, V_B, U)$, a verifier does the following steps:

1. Set $h_A = H_1(m_w, r_A)$ and $\alpha = V_B - H_2(e(U,P)e(q_A + q_B, P_{pub})^{-h_A})$.
2. Compute $\beta = [\alpha]_2$ and $m = F_2({}_{l1}|\beta|) \oplus |\beta|_{l_2}$.
3. Accept the proxy signature δ if m conforms to m_w and ${}_{l1}|\beta| = F_1(m)$.

The correctness of the scheme is justified as follows:

$$\begin{aligned} e(U,P)e(q_A + q_B,\ & P_{pub})^{-h_A} = e(k_B P + d_p, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(k_B P + h_A \cdot d_B + S, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(k_B P + h_A \cdot d_B + h_A \cdot d_A + k_A P, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e((k_B + k_A)P, P)e(h_A \cdot (d_B + d_A), P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e((k_A + k_B)P, P) \\ &= r_A \cdot r_B \\ &= v \end{aligned} \tag{1}$$

Hence, we can obtain $V_B - H_2(e(U,P)e(q_A + q_B, P_{pub})^{-h_A}) = VB - H_2(v) = \alpha$. Since $\beta = F_1(m)||(F_2(F_1(m)) \oplus m) = [\alpha]_2$, we can obtain $m = F_2({}_{l1}|\beta|) \oplus |\beta|_{l_2}$. Finally, the integrity of m is justified by ${}_{l1}|\beta| = F_1(m)$.

(8) ID: On input a valid proxy signature $\delta = (m_w, r_A, V_B, U)$ the proxy signer's identity ID_B can be revealed by m_w.

3 Cryptanalysis on Singh-Verma's Signature Scheme

Tian et al. [5] demonstrated that Singh and Verma's ID-based message recovery proxy signature scheme is insecure to two forgery attacks as follows.

(1) Forgery Attack 1: Assume that an adversary A has obtained a valid proxy signature $\delta = (m_w, r_A, V_B, U)$ on message m. To produce a valid proxy signature δ' on a new message m', A does the following steps:

1. Pick a random value $t \in Z_q^*$.
2. Compute $U' = U + tP$ and $v' = e(U,P)e(q_A + q_B, P_{pub})^{-h_A} \cdot e(P,P)^t = v \cdot e(P,P)^t$.
3. Set $\beta' = F_1(m')||(F_2(F_1(m')) \oplus m')$ and $\alpha' = [\beta']_{10}$.
4. Compute $V'_B = H_2(v') + \alpha'$.
5. Output the proxy signature $\delta' = (m_w, r_A, V'_B, U')$.

We can easily see that $\delta' = (m_w, r_A, V'_B, U')$ is a valid proxy signature on the message m' as follows:

$$\begin{aligned} e(U',P)e(q_A + q_B,\ & P_{pub})^{h_A} = e(U + tP, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(tP, P)e(U, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(tP, P) \cdot v \\ &= v' \end{aligned} \tag{2}$$

Therefore, Singh-Verma's ID-based proxy signature scheme with message recovery is not secure to the above forgery attack 1.

(2) Forgery Attack 2: Assume that A is an adversary who aims to forge a proxy signature δ on any message m, but he/she has not yet obtained a valid proxy signature. Then A does the following steps:

1. Produce a delegation warrant m_w such that m conforms to it.
2. Select two random values $r_A, U \in G_1$, and set $h_A = H_1(m_w, r_A)$ and $v = e(U,P)e(q_A + q_B, P_{pub})^{-h_A}$.
3. Compute $\beta = F_1(m)||(F_2(F_1(m)) \oplus m)$ and $\alpha = [\beta]_{10}$.
4. Compute $V_B = H_2(v) + \alpha$
5. Output the proxy signature $\delta = (m_w, r_A, V_B, U)$.

We can easily see that $\delta = (m_w, r_A, V_B, U)$ is a valid proxy signature on the message m. Since $v = e(U,P)e(q_A + q_B, P_{pub})^{-h_A}$, we can obtain $V_B - H_2(e(U,P)e(q_A + q_B, P_{pub})^{-h_A}) = V_B - H_2(v) = \alpha$. As $\beta = F_1(m)||(F_2(F_1(m)) \oplus m) = [\alpha]_2$, hence we can obtain $m = F_2({}_{l1}|\beta|) \oplus |\beta|_{l_2}$. Finally, we can find out that ${}_{l1}|\beta| = F_1(m)$. Consequently, $\delta = (m_w, r_A, V_B, U)$ is indeed a valid proxy signature on m. Therefore, Singh-Verma's ID-based proxy signature scheme with message recovery is not secure to the above forgery attack 2.

4 Proposed Signature Scheme

This section proposes an improved Singh-Verma's ID-based proxy signature scheme with message recovery. The proposed scheme also consists on eight phases: (1) Setup, (2) Extract, (3) Delegate, (4) DVerify, (5) PKGen, (6) PSign, (7) Verify, and (8) ID phases. The proposed scheme works as follows.

(1) Setup: Given a security parameter λ, the PKG does the following steps:

1. Choose a random generator P of G_1 and the master secret key $s \in Z_q^*$.
2. Set $P_{pub} = sP$ as his/her public key.
3. Publish the public parameters $P = (G_1, G_2, e, P, P_{pub}, H_0, H_1, H_2, F_1, F_2, l_1, l_2)$.

(2) Extract: On input the master secret key s and a user's identity $ID_i \in \{0,1\}^*$, the PKG computes the user's private key $d_i = sH_0(ID_i)$ and sets its public key as $q_i = H_0(ID_i)$.

(3) Delegate: The original signer ID_A does the following steps:

1. Take as input his/her private key d_A and a delegation warrant m_w.
2. Pick a random value $k_A \in Z_q^*$.
3. Compute $r_A = k_A P$.
4. Compute $H_w = H_0(ID_A, m_w, r_A) \in G_1^*$.
5. Compute $S = k_A H_w + d_A$.
6. Output the delegation $W_{A \to B} = (m_w, r_A, S)$.

(4) DVerify: Upon receiving $W_{A \to B} = (m_w, r_A, S)$, the proxy signer ID_B does the following steps:

1. Compute $q_A = H_0(ID_A)$.
2. Compute $H_w = H_0(ID_A, m_w, r_A) \in G_1^*$.

3. Check if $e(S,P) = e(r_A, H_w) \cdot e(q_A, P_{pub})$.
4. If so, ID_B accepts the delegation; otherwise, he/she requests a valid one from ID_A or terminates the protocol.

(5) PKGen: After accepting $W_{A\to B}$, ID_B sets $d_p = S + d_B$ as its proxy signing key.

(6) PSign: Given a message $m \in \{0,1\}^*$ which conforms to the warrant m_w, the proxy signer ID_B with the proxy signing key d_p does the following steps:

1. Select a random value $k_B \in Z_q^*$ and set $r_B = k_B P$.
2. Set $\beta = F_1(m)||(F_2(F_1(m)) \oplus m)$ and $\alpha = [\beta]_{10}$.
3. Compute $H_m = H_0(ID_B, \alpha, r_B) \in G_1^*$.
4. Compute $U = k_B H_m + d_p$.
5. Output the proxy signature $\delta = (m_w, r_A, r_B, U)$.

(7) Verify: On input a proxy signature $\delta = (m_w, r_A, r_B, U)$, a verifier does the following steps:

1. Compute $H_w = H_0(ID_A, m_w, r_A) \in G_1^*$.
2. Compute $\beta = [\alpha]_2$ and $m = F_2(_{l1}|\beta|) \oplus |\beta|_{l_2}$.
3. Compute $H_m = H_0(ID_B, \alpha, r_B) \in G_1^*$.
4. Accept the proxy signature δ if m conforms to m_w and

$$H_2(e(U,P)e(q_A + q_B, P_{pub})^{-1}) \equiv H_2(e(H_m, r_B)e(H_w, r_A)) \tag{3}$$

The correctness of the scheme is justified as follows:

$$\begin{aligned} e(U,P)e(q_A + q_B, \ & P_{pub})^{-1} = e(k_B H_m + d_p, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + d_B + S, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + d_B + k_A H_w + d_A, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + k_A H_w, P)e(d_B + d_A, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + k_A H_w, P) \\ &= e(k_B H_m, P)e(k_A H_w, P) \\ &= e(H_m, k_B P)e(H_w, k_A P) \\ &= e(H_m, r_B)e(H_w, r_A) \end{aligned} \tag{4}$$

(8) ID: On input a valid proxy signature $\delta = (m_w, r_A, r_B, U)$ the proxy signer's identity ID_B can be revealed by m_w.

5 Security Analysis

This section demonstrates a concrete security proof of our proposed scheme.

(1) Unforgeability [6]: Only a designated proxy signer can create a valid proxy signature for the original signer. In other words, nobody can forge a valid proxy signature without the delegation of the original signer. It means that any entity (other than the real proxy signer ID_B), including the original signer ID_A, cannot generate a valid proxy signature. Only an authorized proxy signer ID_B can create a valid proxy signature δ. If any attacker wants to forge a proxy signature, he/she must be

authorized by the original signer signing on a warrant m_w and use the proxy signer's proxy secret key $d_p = S + d_B$ to compute δ. However, this is impossible since the identity of the attacker was not in m_w signed by the original signer. Not to mention, the attacker does not know $d_p = S + d_B$. Under this situation, even if the attacker want to (1) fake the proxy signer key as $d_{p\prime}$, (2) change value $U = k_B H_m + d_p$ to U', or (3) randomly select $k'_B \in Z_q^*$, trying to counterfeit the proxy signature, his/her attempt deems to fail without knowing the proxy secret key $d_p = S + d_B$. Therefore, the proposed scheme provides the unforgeability property.

(2) Verifiability [6]: After checking and verifying the proxy signature, a verifier can be convinced that the received message is signed by the proxy signer authorized by the original signer. In the proposed Verify phase, after checking and verifying the proxy signature δ, the verifier can calculate to check whether the verification equation $H_2(e(U,P)e(q_A + q_B, P_{pub})^{-1})? = H_2(e(H_m, r_B)e(H_w, r_A))$ holds. If it does, the verifier can be convinced that the received message is signed by one of the proxy signer members authorized by the original signer because $U = k_B H_m + d_p$ and $e(H_m, r_B)e(H_w, r_A)$ are used in the verification equation. Therefore, the proposed scheme provides the verifiability property.

6 Conclusion

In 2012, combining the advantages of ID-based message recovery signatures and proxy signatures, Singh-Verma proposed an ID-based proxy signature scheme with message recovery that can be used in wireless e-commerce, mobile agent systems and mobile communication. Unfortunately, Tian et al. showed that Singh-Verma's scheme is not secure against two forgery attacks. For this reason, Singh-Verma's scheme is insecure for practical application. This paper proposed an improvement of Singh-Verma's scheme to eliminate the security problems. The proposed scheme also requires smaller bandwidth in contrast to previous ID-based proxy signature schemes. Hence the proposed scheme can be a good alternative for certificate based proxy signatures used for mobile agent.

Acknowledgements. This work was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2010-0010106) and partially supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program (NIPA-2012- H0301-12-2004) supervised by the NIPA(National IT Industry Promotion Agency).

References

1. Mambo, M., Usuda, K., Okamoto, E.: Proxy signature for delegating signing operation. In: Proc. 3rd ACM Conference on CCS, pp. 48–57 (1996)
2. Zhang, F., Susilo, W., Mu, Y.: Identity-based partial message recovery signatures (or how to shorten ID-based signatures). In: Proc. 9th Conference on FC, pp. 45–56 (2005)

3. Tso, R., Gu, C., Okamoto, T., Okamoto, E.: Efficient ID-Based Digital Signatures with Message Recovery. In: Proc. 6th International Conference on CANS, pp. 47–59 (2007)
4. Singh, H., Verma, G.K.: ID-based proxy signature scheme with message recovery. Journal of Systems and Software 85, 209–214 (2012)
5. Tian, M., Huang, L., Yang, W.: Cryptanalysis of an ID-based proxy signature scheme with message recovery. Applied Mathematics & Information Sciences 6(3), 47–59 (2012)
6. Chou, J.: A Novel Anonymous Proxy Signature Scheme. Advances in Multimedia 427961, 1–10 (2012)