American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

http://asrjetsjournal.org/

Cryptanalysis and Modification of an Improved Self-Certified Digital Signature Scheme with Message Recovery

Mahshid Sadeghpour*

Iran University of Science and Technology, Tehran, Iran
Email: m_sadeghpour@ieee.org

Abstract

Digital signature plays a key role in bringing authenticity to cryptographic communications. A signature scheme with message recovery has two characteristics. The public key of the signer can be authenticated while verifying the signature; and the receiver is able to obtain the message. In 2013, Wu and Xu presented a self-certified digital signature scheme with message recovery by combining the two concepts of digital signature with message recovery and self-certified public key. They also claimed that their scheme provides provable security against man-in-the-middle attack, forgery attack, and message leakage. This paper first reviews the scheme of Wu and Xu, and then presents an insider forgery attack to this scheme. It will be shown that this scheme is not secure against insider forgery attack. A modification is proposed in order to overcome this weakness.

Keywords: Authenticated encryption; Digital signature; Message recovery; Self-certified public key.

1. Introduction

In cryptographic cryptosystems, digital signature is of paramount importance. It can provide integrity, authenticity, and non-repudiation. In public key cryptosystems, the integrity of the public keys must be kept. In classical digital signature schemes, the system authority ***SA*** is used to bind users to their own public keys. However, the management of the certificates of public keys costs a lot. In 1991, Girault [1] introduced the notion of self-certified public key cryptosystem. In his scheme, the features of discrete logarithm problem and factoring problem are combined.

---

*Corresponding author.

Each user is able to choose his own private key, and the user's public key is derived from user's identity and his own private key. In this kind of cryptosystem, the user's public key is generated by $\boldsymbol{SA}$ whereas his private key is only known by user himself. The user's public key and identity can be verified implicitly in self-certified public key cryptosystems.

In 1994, Neyberg and Ruppel [2] introduced a novel digital signature scheme that provides the property of message recovery based on discrete logarithm problem. Later, at the same year, Horster and his colleagues [3] proposed a modification of the scheme of Neyberg and Ruppel and called it authenticated encryption scheme. Based on Horster and his colleagues' scheme, in 2003, Tseng and his colleagues [4] designed an authenticated encryption scheme. They also introduced two variants of their scheme. In 2004, Shao [5] demonstrated that their scheme is insecure against insider forgery attack, and proposed an improvement of their scheme. However, Chen and Jan [6] showed that the scheme of Shao is still not secure against insider forgery attack. In 2005, Chang and his colleagues [7] proved that the scheme of Tseng and his colleagues cannot resist $\boldsymbol{SA}$'s forgery attack. They also designed an improvement to overcome this flaw. However, in 2006, Yoon and Yoo [8] showed that their proposed improvement is vulnerable to known plaintext-ciphertext attack. In their paper, they introduced a new signature scheme which could resist the known plaintext-ciphertext attack. In 2013, Wu and Xu [9] pointed out that the scheme of Yoon and his colleague is vulnerable to man-in-the-middle attack. They also proposed a new self-certified digital signature scheme with message recovery, and provided its provable security in eight aspects.

In this paper, it will be demonstrated that the scheme of Wu and Xu is vulnerable to insider forgery attack, and the specified verifier is able to forge the signature if another verifier fully collaborates with him, in terms of providing the specified receiver with his own private key. The rest of the paper is organized as follows. Section 2 is the review of the scheme of Wu and Xu. In section 3, it will be shown that this scheme is insecure against insider forgery attack. Section 4 is the modification of the scheme of Wu ad Xu. Finally, the conclusion is given in section 5.

2. The scheme of Wu and Xu

This scheme consists of four phases: system initialization phase, signature generation phase, message recovery phase, and signature conversion phase. The process of the signature is given in Figure 1.

2.1. System initialization phase

In the system initialization phase, there exists a system authority ($\boldsymbol{SA}$) that is responsible for generating system parameters. $\boldsymbol{SA}$ first chooses two large primes $\boldsymbol{p}$ and $\boldsymbol{q}$ of almost the same size such that $\boldsymbol{p = 2p' + 1}$ and $\boldsymbol{q = 2q' + 1}$, where $\mathbf{p}'$ and $\boldsymbol{q}'$ are also primes. Then, he computes $\boldsymbol{N = p.q}$. Afterwards, he chooses an integer $\boldsymbol{g}$ which is a base element of order $\boldsymbol{p'.q'}$.

The system authority keeps $\boldsymbol{p, q, p'}$ and $\boldsymbol{q'}$ secret and publishes $\boldsymbol{g, n}$, the group $\mathbb{Z}_{\boldsymbol{n}}^{*}$, and four collision free one-way hash functions as follows:

$$\boldsymbol{F, H}: \{\mathbf{0},\mathbf{1}\}^{*} \longrightarrow \{\mathbf{0},\mathbf{1}\}^{\boldsymbol{k}}$$

$$\boldsymbol{F_1}: \{\mathbf{0},\mathbf{1}\}^{*} \longrightarrow \{\mathbf{0},\mathbf{1}\}^{\boldsymbol{l_1}}$$

$$\boldsymbol{F_2}: \{\mathbf{0},\mathbf{1}\}^{*} \longrightarrow \{\mathbf{0},\mathbf{1}\}^{\boldsymbol{l_2}}$$

Where $\boldsymbol{l_N}$ is of the length of $\boldsymbol{N}$ in binary, and $\boldsymbol{l_N = l_1 + l_2}$. The scheme can deal with $\boldsymbol{M} \in \{\mathbf{0},\mathbf{1}\}^{\boldsymbol{l_2}}$.

In this scheme, $\boldsymbol{k} = \lceil \mathbf{log}(\boldsymbol{P}) \rceil$,and $\boldsymbol{P} = \mathbf{min}(\boldsymbol{p'}, \boldsymbol{q'})$.

When the user $\boldsymbol{U_i}$ wants to join the system, he chooses his private key $\boldsymbol{x_i}$ at random and computes $\boldsymbol{p_i} = \boldsymbol{g^{x_i}\ mod n}$. Then, he submits $(\boldsymbol{p_i}, \boldsymbol{ID_i})$ to the $\boldsymbol{SA}$. After receiving $(\boldsymbol{p_i}, \boldsymbol{ID_i})$, the system authority computes $\boldsymbol{y_i} = (\boldsymbol{p_i} - \boldsymbol{ID_i})^{\boldsymbol{F(ID_i)^{-1}}} \boldsymbol{mod\ n}$, and publishes $\boldsymbol{y_i}$. The user $\boldsymbol{U_i}$ can easily check the validity of $\boldsymbol{y_i}$.

2.2. Signature generation phase

When $\boldsymbol{U_i}$ decides to sign a message $\boldsymbol{M}$ with the length $\boldsymbol{k}$ to the verifier $\boldsymbol{U_j}$, he chooses a number $\boldsymbol{k_i}$ at random and computes the following variables:

$$\boldsymbol{e_i = g^{k_i}\ mod\ n}$$

$$\boldsymbol{R_i = (y_j^{F(ID_j)} + ID_j)^{k_i}\ mod\ n}$$

$$\boldsymbol{f_i = F_1(M) \parallel F_2\big(F_1(M)\big) \oplus M}$$

$$\boldsymbol{T_i = R_i \oplus f_i}$$

$$\boldsymbol{t_i = H(M, e_i)}$$

$$\boldsymbol{s_i = k_i - x_i . t_i . f_i}$$

Where "⊕" denotes the exclusive-or operator, and "‖" denotes the concatenation operator.

At last, $\boldsymbol{U_i}$ sends $(\boldsymbol{e_i}, \boldsymbol{s_i}, \boldsymbol{T_i})$ to $\boldsymbol{U_j}$ as the signature of $\boldsymbol{M}$.

2.3. Message recovery phase

After receiving $(\boldsymbol{e_i}, \boldsymbol{s_i}, \boldsymbol{T_i})$, the verifier $\boldsymbol{U_j}$ computes $\boldsymbol{R_i = e_i^{x_j}\ mod\ n}$, and calculates $\boldsymbol{f_i = R_i \oplus T_i}$.

Then, he can achieve the message $\boldsymbol{M}$ by the following equation

$$\boldsymbol{M = F_2([f_i]^{l_1}) \oplus [f_i]_{l_2}}$$

where $[\boldsymbol{f_i}]^{\boldsymbol{l_1}}$ is the left $\boldsymbol{l_1}$ bits of $\boldsymbol{f_i}$, and $[\boldsymbol{f_i}]_{\boldsymbol{l_2}}$ is the right $\boldsymbol{l_2}$ bits of $\boldsymbol{f_i}$.

In order to check the validity of the signature, $\boldsymbol{U_j}$ first computes $\boldsymbol{t_i} = \boldsymbol{H}(\boldsymbol{M}, \boldsymbol{e_i})$ and then, he checks whether the following equation holds or not.

$$\boldsymbol{e_i} = \boldsymbol{g}^{\boldsymbol{s_i}}(\boldsymbol{y_i}^{\boldsymbol{F(ID_i)}} + \boldsymbol{ID_i})^{\boldsymbol{t_i f_i}} \boldsymbol{mod\ n}$$

If the above equation holds, he accepts the signature as a valid signature for $\boldsymbol{M}$. Otherwise, he rejects the signature.

2.4. Signature conversation phase

$\boldsymbol{U_j}$ can publish $(\boldsymbol{M}, \boldsymbol{t_i}, \boldsymbol{e_i})$, and anyone can confirm the validity of the signature by the following equation:

$$\boldsymbol{t_i} = \boldsymbol{H}(\boldsymbol{M}, \boldsymbol{e_i})$$

phase	SA	U_i	U_j
System initialization phase	Computes $y_i = (p_i - ID_i)^{F(ID_i)^{-1}} mod\ n$, and publicize y_i.	Chooses x_i randomly and computes $p_i = g^{x_i} mod n$. Submits (p_i, ID_i) to the SA.	
Signature generation phase		Computes $e_i = g^{k_i} mod\ n$, $R_i = (y_j{}^{F(ID_j)} + ID_j)^{k_i} mod\ n$, $f_i = F_1(M) \parallel F_2(F_1(M)) \oplus M$ $T_i = R_i \oplus f_i$ $t_i = H(M, e_i)$ $s_i = k_i - x_i . t_i . f_i$ U_i sends (e_i, s_i, T_i) to U_j	
Message recovery phase			Calculates $R_i = e_i{}^{x_j} mod\ n$, $f_i = R_i \oplus T_i$, and obtains $M = F_2([f_i]^{l_1}) \oplus [f_i]_{l_2}$. Computes $t_i = H(M, e_i)$. Then, checks whether $e_i = g^{s_i}(y_i{}^{F(ID_i)} + ID_i)^{t_i f_i} mod\ n$

Figure 1: Scheme of Wu and Xu

3. Insider forgery attack to scheme of Wu and Xu

In this section, an insider forgery attack to the aforementioned scheme will be presented. In the insider forgery attack, the specified receiver U_j forges the received signature of a given message for other system users. The other user is denoted by U_O. If $U_O$is willing to provide U_j with his secret key x_o, then U_j is able to forge a valid signature for him. This cooperation between U_O and U_j makes sense in many cases of applications of digital signature scheme with message recovery, e.g. in electronic election or in electronic commerce. As an instance, consider that U_j and U_O are two friends who want to purchase e-tickets for some mp3 files. They decide to buy just one e-ticket and forge another e-ticket cooperatively. Thus, U_j buys the ticket (which is $U_i's$ signature with message recovery) and offers U_O a forged e-ticket. If U_O trusts U_j, he might reveal his secret key to U_j in order to obtain a forged e-ticket.

Follows demonstrate how this forgery attack operates.

Assume that U_j has recovered M from the signature (e_i, s_i, T_i) signed by U_i . He wants to forge a signature (e_i, s_i, T'_i) of M for U_O. The claim is that (e_i, s_i, T'_i) is a valid signature of M signed by U_i for U_O.

First, U_O reveals his own private key $x_o$to U_j. Thereafter, U_j performs the following steps to forge a valid signature:

1) Compute $R'_i = {e_i}^{x_0} \ mod \ n$.

2) Compute $T'_i = R'_i \oplus f_i$. (U_j has obtained f_i in the message recovery phase.)

3) Send the forged signature (e_i, s_i, T'_i) to U_O.

The signature (e_i, s_i, T'_i) can be verified easily since

$$M = F_2([f_i]^{l_1}) \oplus [f_i]_{l_2}$$

$$t_i = H(M, e_i)$$

, and $e_i = g^{s_i}({y_i}^{F(ID_i)} + ID_i)^{t_i f_i} \ mod \ n$, therefore; U_O can claim that (e_i, s_i, T'_i) is the signature of M signed by U_i for him.

In case of any disputes over the validity of the signature, U_O publishes (M, t_i, e_i), to satisfy whomever questions the validity of (e_i, s_i, T'_i).

The reason of vulnerability of Wu and Xu's scheme to this attack is that the verification equations $t_i = H(M, e_i)$, and $e_i = g^{s_i}({y_i}^{F(ID_i)} + ID_i)^{t_i f_i} \ mod \ n$ cannot detect the change in the amount of T_i which is replaced by T'_i in the forged signature.

The process of the attack is illustrated in Figure 2.

phase	U_i	U_j	U_O
Signature generation phase	Computes $e_i = g^{k_i} \bmod n$, and $R_i = (y_j{}^{F(ID_j)} + ID_j)^{k_i} \bmod n$, and $f_i = F_1(M) \parallel F_2(F_1(M)) \oplus M$, $T_i = R_i \oplus f_i$, $t_i = H(M, e_i)$, and $s_i = k_i - x_i . t_i . f_i$ $U_i \xrightarrow{(e_i, s_i, T_i)} U_j$		
Message recovery phase		Calculates $R_i = e_i{}^{x_j} \bmod n$, $f_i = R_i \oplus T_i$, and obtains $M = F_2([f_i]^{l_1}) \oplus [f_i]_{l_2}$. Computes $t_i = H(M, e_i)$. Then, checks whether $e_i = g^{s_i}(y_i{}^{F(ID_i)} + ID_i)^{t_i f_i} \bmod n$	
Insider forgery attack		Computes $R'_i = e_i{}^{x_0} \bmod n$, $T'_i = R'_i \oplus f_i$ forges the signature and $U_j \xrightarrow{(e_i, s_i, T'_i)} U_O$	Chooses x_O as his secret key $U_j \xleftarrow{x_O} U_O$ Claims that U_i signed (e_i, s_i, T'_i) for him.

Figure 2: Insider forgery attack to the scheme of Wu and Xu

4. Modification of Wu and Xu's self-certified digital signature scheme with message recovery

This section represents a slight modification to overcome the security flaw in the scheme of Wu and Xu.

Given the fact that the reason of insecurity of Wu and Xu's scheme against insider forgery attack is that the verification equation cannot find the change in T_i, it is sufficed to make a slight change in the equation $t_i = H(M, e_i)$. With changing $t_i = H(M, e_i)$ to $t_i = H(M^*, e_i)$ where $M^* = M \parallel T_i$, the insider forgery attack is not successful against the modified scheme because any changes in term of T_i would cause changes in the amount of M^* and t_i. The modified scheme has also four phases: system initialization phase, signature generation phase, message recovery phase, and dispute arbitration phase. The modified scheme is represented in Figure 3.

4.1. System initialization phase

This phase is exactly the same as system initialization phase in Wu and Xu's scheme.

4.2. signature generation phase

When U_i wants to sign a message M with the length k to the verifier U_j, he chooses a number k_i randomly and calculates the following variables:

$$e_i = g^{k_i} \ mod \ n$$

$$R_i = ({y_j}^{F(ID_j)} + \ ID_j)^{k_i} \ mod \ n$$

$$f_i = F_1(M) \parallel F_2\big(F_1(M)\big) \oplus M$$

$$T_i = R_i \oplus f_i$$

$$M^* = M \parallel T_i$$

$$t_i = H(M^*, e_i)$$

$$s_i = k_i - x_i . t_i . f_i$$

Where "⊕" denotes the exclusive-or operator, and "‖" denotes the concatenation operator.

Then, U_i sends (e_i, s_i, T_i) to U_j as the signature of M.

4.3. Message recovery phase

Upon receiving (e_i, s_i, T_i), the verifier U_j computes $R_i = {e_i}^{x_j} \ mod \ n$,and $f_i = R_i \oplus T_i$. Then, he can achieve the message $M = F_2([f_i]^{l_1}) \oplus [f_i]_{l_2}$, where $[f_i]^{l_1}$ is the left l_1 bits of f_i, and $[f_i]_{l_2}$ is the right l_2 bits of f_i. In order to check the validity of the signature, U_j first computes $M^* = M \parallel T_i$, and

$t_i = H(M^*, e_i)$. Then, he checks whether the following equation holds or not.

$$e_i = g^{s_i}({y_i}^{F(ID_i)} + ID_i)^{t_i f_i} \ mod \ n$$

If the above equation holds, he accepts the signature as a valid signature for M. Otherwise, he rejects the signature. If the signature is valid and signed by U_i, then the above equation must hold because

$g^{s_i}({y_i}^{F(ID_i)} + ID_i)^{t_i f_i} \ mod \ n = g^{s_i}(g^{x_i})^{t_i f_i} = g^{s_i} . g^{k_i - s_i} = g^{k_i} \ mod \ n$, and as mentioned in signature generation phase $g^{k_i} = e_i \ mod n$.

4.4. Dispute arbitration phase

In case of any dispute over the validity of the signature, U_j can publish (M, t_i, e_i), and anyone can confirm the

validity of the signature by the following equation:

$$t_i = H(M^*, e_i)$$

phase	SA	U_i	U_j
System initialization phase	Computes $y_i = (p_i - ID_i)^{F(ID_i)^{-1}} mod\ n$, and publicize y_i.	Chooses x_i randomly and computes $p_i = g^{x_i} modn$. Submits (p_i, ID_i) to the SA.	
Signature generation phase		Computes $e_i = g^{k_i} mod\ n$, $R_i = (y_j{}^{F(ID_j)} + ID_j)^{k_i} mod\ n$, $f_i = F_1(M) \parallel F_2(F_1(M)) \oplus M$, and $T_i = R_i \oplus f_i$ Then, Computes $M^* = M \parallel T_i$, $t_i = H(M^*, e_i)$ $s_i = k_i - x_i . t_i . f_i$ U_i sends (e_i, s_i, T_i) to U_j	
Message recovery phase			Calculates $R_i = e_i{}^{x_j} mod\ n$, $f_i = R_i \oplus T_i$, and obtains $M = F_2([f_i]^{l_1}) \oplus [f_i]_{l_2}$. Computes $M^* = M \parallel T_i$, $t_i = H(M^*, e_i)$. Then, checks whether $e_i = g^{s_i}(y_i{}^{F(ID_i)} + ID_i)^{t_i f_i} mod\ n$

Figure 3: Modification of an improved self-certified digital signature scheme with message recovery

5. Conclusion

In the present paper an improved self-certified digital signature scheme with message recovery is reviewed. A security analysis for the scheme is presented by proposing an insider forgery attack against this scheme, and it is demonstrated that this scheme cannot withstand the attack. It is also argued that in modern electronic environments, the aforementioned attack can cause negative consequences, particularly in electronic commerce and electronic voting systems. A slight improvement for the scheme is suggested in order to remove the weakness. The propose modification is secure against insider forgery attack.

Acknowledgement

The author would like to thank the anonymous reviewers for their comments.

References

[1] M. Girault, "Self-certified public-keys", Advances in Cryptology —EUROCRYPT'91, Berlin: Springer, 1991, pp. 491-497.

[2] K. Neyberg and R. Ruppel, "Message recovery for signature schemes based on the discrete logarithm problem", Advances in Cryptology—Erocryt'94, LNCS 950, Berlin: Springer, 1994, pp. 175-190.

[3] P. Horster and M. Michels and H. Petersen, (1994, Jul.), "Authenticated encryption schemes with low communication costs.", Electronic Letters, [On-line], 30(15), pp. 1212-1213. Available: http://ieeexplore.ieee.org/document/311896 [Aug. 2, 2002].

[4] Y.-M. Tseng and J.-K. Jan and H.-Y. Chien, (2003, Mar.), "Digital signature with message recovery using self- certified public keys and its variant.", Applied Mathematics and Computation, [On-line], 136(2-3), pp.203-214. Available: http://www.sciencedirect.com/science/article/pii/S0096300302000103 .

[5] Z. Shao, "Improvement of digital signature with message recovery using self-certified public keys and its variants", Applied Mathematics and Computation, vol. 159, pp.391-399, Dec. 2004.

[6] Y.-H. Chen and J.-K. Jan, "An authenticated encryption scheme for securely signing a signature with message linkages", International Conference on Innovative Computing, Information and Control, in Proc ICICIC, 2007, pp. 77-80.

[7] Y.F. Chang, C.C. Chang, F.F. Huang, (2005, Feb.). "Digital signature with message recovery using self-certified public keys without trustworthy system authority." Applied Mathematics and Computation. [On-line]. 161(1), pp. 211–227. Available: http://www.sciencedirect.com/science/article/pii/S0096300303012220.

[8] E.J. Yoon and K.Y. Yoo, "An improved digital signature with message recovery using self-certified public keys without trustworthy system authority", Theory and Practice of Computer Science, Czech Republic: Merin, 2006, pp. 548–555.

[9] F. Wu, L. Xu, (2013, Sep.). "An improved and provable self-certified digital signature scheme with message recovery", International Journal of Communication Systems. [On-line]. 28(2), pp. 344-357. Available: http://onlinelibrary.wiley.com/wol1/doi/10.1002/dac.2673/full .