Published in M. Gavrilova et al., Eds, *Computational Science and Its Applications – ICCSA 2006*, vol. 3982 of *Lecture Notes in Computer Science*, pp. 338–344, Springer, 2006.

On the TYS Signature Scheme

Marc Joye[1,⋆] and Hung-Mei Lin[2]

[1] Gemplus, Security Technologies Department
La Vigie, Avenue du Jujubier
13705 La Ciotat Cedex, France
[2] Traverse des Jardins
83640 Saint Zacharie, France

Abstract. This paper analyzes the security of a very efficient signature scheme proposed by C.H. Tan, X. Yi and C.K. Siew: the TYS signature scheme. We show that this scheme is universally forgeable; more specifically, we show that anyone is able to produce a valid TYS signature on a chosen message from an arbitrary valid message/signature pair. We also suggest modifications to the TYS signature scheme and relate the resulting scheme to the Camenisch-Lysyanskaya signature scheme.

Keywords. Cryptography, digital signature, standard model, TYS signature scheme, Camenisch-Lysyanskaya signature scheme.

1 Introduction

Designing secure yet efficient signature schemes is of central importance for cryptographic applications. The standard security notion for signature schemes is *existential unforgeability against adaptive chosen message attacks* (EUF-CMA) [12]. Informally, we require that an adversary getting access to a signing oracle and making adaptive signature queries on messages of his choice cannot produce a new valid signature. A scheme provably meeting this security notion is given in [12] with subsequent improvements in [7, 5]. More efficient, *stateless signature schemes* were later proposed. These include the Cramer-Shoup signature scheme [6] (revisited in [9]) and the Gennaro-Halevi-Rabin signature scheme [11], independently introduced in 1999.

More recently, C.H. Tan, X. Yi and C.K. Siew proposed a new signature signature scheme: the *TYS signature scheme* [16]. This scheme can be seen as an efficient variant of the Cramer-Shoup signature scheme, allowing faster signing and producing signatures roughly half the size.

Crosschecking the security proof offered in [16], we observed some inaccuracies. This illustrates once more that details are easily overlooked in security proofs [15] and this may have dramatic consequences.

⋆ The work described in this paper has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

We present in this paper an attack against the TYS signature scheme. We show how given a valid message/signature pair it is easy to produce a valid signature on a *chosen* message. Repairing the scheme does not seem possible. Instead, we suggest two simple modifications to the original scheme. Interestingly, the so-obtained scheme leads to an off-line/on-line variation [8, 14] of a previous scheme due to Camenisch and Lysyanskaya [4].

The rest of this paper is organized as follows. In the next section, we review the original TYS signature scheme. In Section 3, we show that the scheme is universally forgeable. Next, in Section 4, we suggest modifications to the original scheme. Finally, we conclude in Section 5.

2 Tan-Yi-Siew Signature Scheme

We review in this section the TYS signature scheme, as given in [16].

For a security parameter ℓ, let $H : \{0,1\}^* \to \{0,1\}^\ell$ be a collision-resistant hash function. The scheme consists of three algorithms: the key generation algorithm, the signing algorithm and the verification algorithm.

Key generation Choose two random primes $p = 2p' + 1$ and $q = 2q' + 1$, where p' and q' are prime and $\log_2 p'q' > 2\ell + 1$, and let $N = pq$. Choose two quadratic residues g and x in $\mathbb{Z}_N^*$ such that the order of g is $p'q'$. Finally, choose a random ℓ-bit integer z and compute $h = g^{-z} \bmod N$.

The public key is $\mathsf{pk} = \{g, h, x, N\}$ and the private key is $\mathsf{sk} = \{p, q, z\}$.

Signing Let $m \in \{0,1\}^*$ denote the message being signed. Randomly pick an ℓ-bit integer k and an ℓ-bit prime e. Next compute

$$y = (xg^{-k})^{1/e} \bmod N, \quad c = H(\mathsf{pk}, y, m), \quad t = k + cz \ .$$

The signature on message m is $\sigma = (t, y, e)$.

Verification Signature $\sigma = (t, y, e)$ on message m is accepted iff e is odd and

$$y^e g^t h^{c'} \equiv x \pmod N$$

with $c' = H(\mathsf{pk}, y, m)$.

Fig. 1. Original TYS signature scheme.

The TYS signature scheme is very efficient and allows the use of coupons. The value of y can be precomputed off-line; only the pair (c, t) needs to be computed on-line.

3 Security Analysis

The authors of the TYS signature scheme conclude that their signature scheme is existentially unforgeable against chosen-message attacks (EUF-CMA) under the strong RSA assumption. From the definition of a TYS signature, $\sigma = (t, y, e)$, we observe that $t = k + cz$ is not zero-knowledge if $k < cz$. We take advantage of this observation and present below a universal forgery.

Given a valid signature $\sigma = (t, y, e)$ on a message m, we compute

$$\hat{z} := \left\lfloor \frac{t}{c} \right\rfloor \quad \text{with } c = H(\mathsf{pk}, y, m)$$

as an approximation for z. Since $t = k + cz$, we obviously have $z = \hat{z}$ when $k < c$. Otherwise, we have $z = \hat{z} - \delta$ for some integer $\delta > 0$. The correct value of δ can be guessed by checking whether $g^{\delta} \equiv h g^{\hat{z}} \pmod{n}$ for $\delta = 0, 1, 2, \ldots$

By construction, we have $t = k + cz$ and so we obtain $\delta = \lfloor t/c \rfloor - z = \lfloor k/c \rfloor$. If we view function H as a random oracle then, with probability $1/2$, the most significant bit of c is 1. Therefore, with probability at least $1/2$, it follows that

$$\delta = \left\lfloor \frac{k}{c} \right\rfloor \leq \frac{k}{c} < \frac{2^{\ell}}{2^{\ell-1}} = 2$$

or equivalently that δ is equal to 0 or 1.

Once $z = \hat{z} - \delta$ is known, it is easy to forge the signature on a *chosen* message m' as

$$\sigma' = (t', y', e) \quad \text{with } \begin{cases} y' = y\, g^{-a} \bmod N \\ c' = H(\mathsf{pk}, y', m') \\ t' = t + (c' - c)z + ae \end{cases}, \tag{1}$$

for an arbitrary integer a. It is easy to see that $y'^{e} g^{t'} h^{c'} \equiv y^{e} g^{-ae} g^{t+(c'-c)z+ae} g^{-zc'} \equiv y^{e} g^{t} h^{c} \equiv x \pmod{N}$.

A closer look at the security proof in [16] shows that exponent e is assumed to be $\neq 1$. A signature with $e = 1$ is however valid as it is an odd value (e is not required to be prime in the verification algorithm). This may allow to mount further signature forgeries.

4 Modifying the Scheme

As usual, the security proof given in [16] is by contradiction. It is assumed that there exists a polynomial-time adversary $\mathcal{A}$ making q_S adaptive signature queries on chosen messages and then producing a forgery with non-negligible probability. The forgery is then used to solve some intractable problem — the strong RSA problem.

Definition 1 (Strong RSA problem [1, 10]). *Being given an RSA modulus N and an element $s \in \mathbb{Z}_N$, the* strong RSA problem *consists in finding $r \in \mathbb{Z}_{>1}$ and $u \in \mathbb{Z}_N$ such that $u^r \equiv s \pmod{N}$.*

For $i \in \{1, \ldots, q_S\}$, let m_i be the i^{th} signed message and let $\sigma_i = (t_i, y_i, e_i)$ be the i^{th} signature. Let $\sigma_* = (t_*, y_*, e_*)$ be the forgery on message m_*, produced by $\mathcal{A}$. Two types of forgeries are distinguished in [16]:

Type I: $e_* = e_j$ for some $j \in \{1, \ldots, q_S\}$;

Type II: $e_* \neq e_j$ for all $j \in \{1, \ldots, q_S\}$.

For the Type I forger, letting $r := e_* = e_j$ and $c_* = H(\mathsf{pk}, y_*, m_*)$, the authors of [16] obtain the relation

$$\left(\frac{y_*}{y_j}\right)^r \equiv g^{t_j - t_*}\, h^{c_j - c_*} \equiv s^{2(t_j - t_* - z(c_j - c_*))\prod_{i \neq j} e_i} \pmod{N} \ . \tag{2}$$

Next they incorrectly assume that

(A1) $t_j - t_* - z(c_j - c_*)$ (in absolute value) is of length ℓ bits, and
(A2) $t_j - t_* - z(c_j - c_*) \neq 0$,

and hence deduce that the probability of having

$$\gcd\left(r, 2(t_j - t_* - z(c_j - c_*)) \prod_{i \neq j} e_i\right) = \gcd\big(r, t_j - t_* - z(c_j - c_*)\big) = 1$$

is overwhelming. As a result, the extended Euclidean algorithm yields integers α and β satisfying $\alpha\, r + \beta\, 2(t_j - t_* - z(c_j - c_*)) \prod_{i \neq j} e_i = 1$ and (r, u) with $u := s^{\alpha}\, (y_*/y_j)^{\beta} \bmod N$ is a solution to the strong RSA problem.

Remark that the implications of Assumptions (A1) and (A2) are illustrated by the cases $a \neq 0$ and $a = 0$ in the forgeries described in the previous section (cf. Eq. (1)), respectively.

Assumption (A1) should be satisfied by proper parameter definitions and appropriate range verifications in the verification algorithm. Assumption (A2) is more intricate to handle. If $t_j - t_* - z(c_j - c_*) = 0$ then, from Eq. (2), we get $(y_*/y_j)^r \equiv 1 \pmod{N}$ and thus $y_* = y_j$ since $\gcd(r, 2p'q') = 1$. We therefore subdivide the Type I forger into:

Type Ia: $e_* = e_j$ and $y_* \neq y_j$ for some $j \in \{1, \ldots, q_S\}$;

Type Ib: $e_* = e_j$ and $y_* = y_j$ for some $j \in \{1, \ldots, q_S\}$.

It remains to get a scheme secure in the case corresponding to the Type Ib forger (only Type Ia and Type II are covered in [16]).

The presence of y in the definition of $c = H(\mathsf{pk}, y, m)$ in the original scheme (cf. Fig. 1) seems to indicate that we need to additionally rely on the random oracle model [2]. However, inspecting the security proof given in [16], we see that y is unnecessary in the definition of c for proving the security against Type Ia or Type II forger. We therefore decide to remove it.

Standard techniques for proving the security against the Type Ib forger would lead to defining larger parameters in the signing algorithm. To avoid this, we

suggest to exchange the roles of k and t in the original signature scheme. The signature $\sigma = (k, y, e)$ on a message m is then given by

$$y = (x\, g^{-t})^{1/e} \bmod N \quad \text{with} \begin{cases} t = k - cz \\ c = H(\mathsf{pk}, m) \end{cases} .$$

In more detail, we get the scheme depicted in Fig. 2.

Global parameters Let ℓ_N, ℓ_H, ℓ_E and ℓ_K be four security parameters, satisfying

$$\ell_E \geq \ell_H + 2 \quad \text{and} \quad \ell_K \gg \ell_N + \ell_H \ .$$

(Typically $\ell_N = 1024$, $\ell_H = 160$, $\ell_E = 162$, and $\ell_K = 1344$.)
Let also a collision-resistant hash function $H : \{0,1\}^* \to \{0,1\}^{\ell_H}$.

Key generation Choose two random primes $p = 2p'+1$ and $q = 2q'+1$, where p' and q' are primes of equal length, so that $N = pq$ is of length exactly ℓ_N. Choose at random two quadratic residues g and x in $\mathbb{Z}_N^*$. Finally, for a random integer $z \bmod p'q'$, compute $h = g^{-z} \bmod N$.

The public key is $\mathsf{pk} = \{g, h, x, N\}$ and the private key is $\mathsf{sk} = \{p, q, z\}$.

Signing Let $m \in \{0,1\}^*$ denote the message being signed. Randomly pick an ℓ_K-bit integer t and an ℓ_E-bit prime e. Next compute

$$y = (xg^{-t})^{1/e} \bmod N, \quad c = H(\mathsf{pk}, m), \quad k = t + cz \ .$$

The signature on message m is $\sigma = (k, y, e)$.

Verification Signature $\sigma = (k, y, e)$ on message m is accepted iff e is an odd ℓ_E-bit integer and

$$y^e g^k h^{c'} \equiv x \pmod N$$

with $c' = H(\mathsf{pk}, m)$.

Fig. 2. Modified TYS signature scheme.

It is interesting to note that the scheme we obtained is a disguised version of the Camenisch-Lysyanskaya signature scheme [4]. Actually, it can seen as an efficient off-line/on-line variant of it. See Appendix A.

5 Conclusion

We have shown that the TYS signature scheme does not meet the EUF-CMA security level. We have mounted an attack against it, yielding part of the secret key. This partial information was then used to produce universal signature

forgeries. In a second part, we analyzed why the security was flawed and suggested simple modifications, leading to an efficient off-line/on-line variant of the Camenisch-Lysyanskaya signature scheme.

References

1. N. Barić and B. Pfitzmann, *Collision-free accumulators and fail-stop signature schemes without trees*, Advances in Cryptology – EUROCRYPT '97, Lecture Notes in Computer Science, vol. 1233, Springer-Verlag, 1997, pp. 480–494.
2. M. Bellare and P. Rogaway, *Random oracles are practical: A paradigm for designing efficient protocols*, 1st ACM Conference on Computer and Communications Security, ACM Press, 1993, pp. 62–73.
3. D. Boneh and X. Boyen, *Short signatures without random oracles*, Advances in Cryptology – EUROCRYPT 2004, Lecture Notes in Computer Science, vol. 3027, Springer-Verlag, 2004, pp. 62–73.
4. J. Camenisch and A. Lysyanskaya, *A signature scheme with efficient protocols*, Security in Communication Networks (SCN '02), Lecture Notes in Computer Science, vol. 2576, Springer-Verlag, 2003, pp. 268–289.
5. R. Cramer and I. Damgård, *New generation of secure and practical RSA-based signatures*, Advances in Cryptology – CRYPTO '96, Lecture Notes in Computer Science, vol. 1109, Springer-Verlag, 1996, pp. 173–185.
6. R. Cramer and V. Shoup, *Signature schemes based on the strong RSA assumption*, ACM Transactions on Information and System Security **3** (2000), no. 3, 161–185.
7. C. Dwork and M. Naor, *An efficient existentially unforgeable signature scheme and its applications*, Journal of Cryptology **11** (1998), no. 3, 187–208.
8. A. Fiat and A. Shamir, *How to prove yourself: Practical solutions to identification and signature problems*, Advances in Cryptology – CRYPTO '86 (A.M. Odlyzko, ed.), Lecture Notes in Computer Science, vol. 263, Springer-Verlag, 1987, pp. 186–194.
9. M. Fischlin, *The Cramer-Shoup strong RSA scheme revisited*, Public Key Cryptography – PKC 2003, Lecture Notes in Computer Science, Springer-Verlag, 2003, pp. 116–129.
10. E. Fujisaki and T. Okamoto, *Statistical zero-knowledge protocols to prove modular polynomial equations*, Advances in Cryptology – CRYPTO '97, Lecture Notes in Computer Science, vol. 1294, Springer-Verlag, 1997, pp. 16–30.
11. R. Gennaro, S. Halevi, and T. Rabin, *Secure hash-and-sign signatures without the random oracle*, Advances in Cryptology – EUROCRYPT '99, Lecture Notes in Computer Science, vol. 1592, Springer-Verlag, 1999, pp. 123–139.
12. S. Goldwasser, S. Micali, and R. Rivest, *A digital signature scheme secure against adaptive chosen message attacks*, SIAM Journal of Computing **17** (1988), no. 2, 281–308.
13. G. Poupard and J. Stern, *On the fly signatures based on factoring*, 7th ACM Conference on Computer and Communications Security, ACM Press, 1999, pp. 37–45.
14. A. Shamir and Y. Tauman, *Improved online/offline signature schemes*, Advances in Cryptology – CRYPTO 2001 (J. Kilian, ed.), Lecture Notes in Computer Science, Springer-Verlag, 2001, pp. 355–367.
15. J. Stern, D. Pointcheval, J. Malone-Lee, and N.P. Smart, *Flaws in applying proof methodologies to signature schemes*, Advances in Cryptology – CRYPTO 2002, Lecture Notes in Computer Science, vol. 2442, Springer-Verlag, 2002, pp. 93–110.

16. C.H. Tan, X. Yi, and C.K. Siew, *A new provably secure signature scheme*, IEICE Trans. Fundamentals **E86-A** (2003), no. 10, 2633–2635.

A Camenisch-Lysyanskaya Signature Scheme

For the reader's convenience, we review below the (single-message) Camenisch-Lysyanskaya signature scheme. We use the same notations as in [4].

Global parameters Let ℓ_n, ℓ_m, ℓ_e, ℓ_s and ℓ be five security parameters, satisfying

$$\ell_e \geq \ell_m + 2 \quad \text{and} \quad \ell_s = \ell_n + \ell_m + \ell \ .$$

Key generation Choose two random primes $p = 2p' + 1$ and $q = 2q' + 1$, where p' and q' are primes of equal length, so that $n = pq$ is of length exactly ℓ_n. Choose at random three quadratic residues $a, b, c \in \mathbb{Z}_n^*$.

The public key is $\mathsf{pk} = \{n, a, b, c\}$ and the private key is $\mathsf{sk} = \{p, q\}$.

Signing Let $m \in \{0,1\}^{\ell_m}$ denote the message being signed. Randomly pick an ℓ_s-bit integer s and an ℓ_e-bit prime e. Next compute

$$v = (a^m \, b^s \, c)^{1/e} \bmod n \ .$$

The signature on message m is $\sigma = (s, v, e)$.

Verification Signature $\sigma = (s, v, e)$ on message m is accepted iff $v^e \equiv a^m \, b^s \, c \pmod{n}$ and $2^{\ell_e} > e > 2^{\ell_e - 1}$.

Remark that the change of variables

$$\begin{cases} n \leftarrow N \\ c \leftarrow x \\ b \leftarrow g^{-1} \\ a \leftarrow h^{-1} \\ s \leftarrow k \\ m \leftarrow c \\ v \leftarrow y \end{cases}$$

transforms the Camenisch-Lysyanskaya signature $\big(s, v = (a^m \, b^s \, c)^{1/e} \bmod n, e\big)$ into our modified TYS signature (k, y, e) with

$$\begin{aligned} y &= \big((h^{-1})^c \, (g^{-1})^k \, x\big)^{1/e} \bmod N \\ &= (g^{zc-k} \, x)^{1/e} \bmod N \\ &= (x g^{-t})^{1/e} \bmod N \end{aligned}$$

where $t = k - cz$.