Security Analysis of ePrint Report 2016/500 ”Efficient Identity-Based Encryption and Public-Key Signature from Trapdoor Subgroups”

Lucjan Hanzlik, Kamil Kluczniak

Wrocław University of Technology
firstname.secondname@pwr.edu.pl

Abstract. In this short report we analyse the security of three schemes proposed by J. H. Park et al. in ”Efficient Identity-Based Encryption and Public-Key Signature from Trapdoor Subgroups”. The schemes make use of trapdoor subgroups of $\mathbb{Z}_n^*$ and are secure under new assumptions called q-Trapdoor Subgroup Diffie-Hellman (TSDH) and q-Trapdoor Subgroup Exponent Inversion (TSEI). We show that given several secret keys in case of IBE or several signatures in case of PKS, one can easily extract the trapdoor and break security of the proposed schemes.

Keywords: attack, trapdoor subgroup, congruential equations

1 Introduction

Recently J. H. Park et al. presented two efficient Identity-Based Encryption (IBE) schemes based on trapdoor subgroups [1]. IBE is a cryptosystem, where the public key is a string, e.g. the email address of the user. In order to work, secret keys must be generated by a trusted party. The schemes are secure under a new assumption called q-Trapdoor Subgroup Diffie-Hellman (TSDH). The author's also present a public key signature scheme that is strongly unforgeable against chosen message attacks under a new assumption called q-Trapdoor Subgroup Exponent Inversion (TSEI). Both assumptions were shown to hold in the generic group model [2].

Our Contribution

In this short report we show that both IBE schemes and the signature scheme from [1] are not secure. In particular, we show that:

1. given secret keys for several identities in the IBE scheme, one can decrypt messages encrypted for any identity,
2. given several message-signature pairs in the signature scheme, one can create signatures for arbitrary messages.

2 The Schemes

2.1 Identity-Based Encryption

The original paper [1] proposes two IBE schemes: a CPA secure and a CCA secure scheme. Here we focus on the CPA secure scheme as the attack simply transfers to the other case. Let M be the message space of the encryption scheme. The CPA secure scheme from [1] is defined as:

$\mathsf{Setup}_{\mathsf{IBE}}(\lambda)$ Given a security parameter λ, the setup algorithm chooses two Sophie Germain prime numbers $p = 2 \cdot p_1 + 1$ and $q = 2 \cdot q_1 + 1$, computes modulus $n = p \cdot q$, chooses random generator g of order $p_1 \cdot q_1$, chooses a random secret $x \xleftarrow{\$} \mathbb{Z}_{p_1 \cdot q_1}$ and computes $g_1 = g^x \mod n$. Select two hash functions $h : \{0,1\}^* \to \{0,1\}^\ell$, where $\ell < \log_2(p_1 \cdot q_1)$ and $H : \mathbb{Z}_n \to M$. Output the public parameters $\mathsf{pp} = (n, g, g_1, h, H)$ and the master secret key $\mathsf{msk} = (x, p_1 \cdot q_1)$.

$\mathsf{KeyGen}_{\mathsf{PBS}}(\mathsf{msk}, ID)$ Given a master secret key msk and identity ID, the key generation algorithm computes and returns the secret key sk_{ID}, such that $(x + h(ID)) \cdot \mathsf{sk}_{ID} \equiv 1 \mod (p_1 \cdot q_1)$.

$\mathsf{Encrypt}_{\mathsf{PBS}}(\mathsf{pp}, ID, m)$ To encrypt message m for identity ID, the algorithm chooses random $s \in \mathbb{Z}_n$, $C_0 = g^s \mod n$, computes $C_1 = (g_1 \cdot g^{h(ID)})^s \mod n$ and $C_2 = H(C_0) \bigoplus m$. Output the ciphertext $\mathsf{ct} = (C_1, C_2)$.

$\mathsf{Decrypt}_{\mathsf{PBS}}(\mathsf{ct}, \mathsf{sk}_{ID})$ To decrypt ciphertext $\mathsf{ct} = (C_1, C_2)$, the decryption algorithm computes $C_0 = (C_1)^{\mathsf{sk}_{ID}} \mod n$ and returns message $m = C_2 \bigoplus H(C_0)$.

2.2 Signature Scheme

In this subsection we recall the signature scheme presented in [1].

$\mathsf{Setup}_{\mathsf{PKSS}}(\lambda)$ Given a security parameter λ, the setup algorithm chooses two Sophie Germain prime numbers $p = 2 \cdot p_1 + 1$ and $q = 2 \cdot q_1 + 1$, computes modulus $n = p \cdot q$, chooses random generator g of order $p_1 \cdot q_1$, chooses a random secret $x \xleftarrow{\$} \mathbb{Z}_{p_1 \cdot q_1}$ and computes $g_1 = g^x \mod n$. Select hash functions $h : \{0,1\}^* \to \{0,1\}^\ell$, where $\ell < \log_2(p_1 \cdot q_1)$. Output the public key $\mathsf{pk} = (n, g, g_1, h)$ and the secret key $\mathsf{sk} = (x, p_1 \cdot q_1)$.

$\mathsf{Sign}_{\mathsf{PKSS}}(\mathsf{sk}, m)$ Given a secret key sk and message m, compute and return the signature σ, such that $(x + h(m)) \cdot \sigma \equiv 1 \mod (p_1 \cdot q_1)$.

$\mathsf{Verify}_{\mathsf{PKSS}}(\mathsf{pk}, m, \sigma)$ To verify signature σ on message m, the verification algorithm accepts if $(g_1 \cdot g^{h(m)})^\sigma = g \mod n$ holds. Otherwise, it rejects the signature.

3 The Attack

The crucial observation is, that using corruption queries in case of IBE and signature queries in case of the signature scheme, we are able to receive values h_1, s_1 such that $(x + h_1) \cdot s_1 \equiv 1 \pmod{q_1 \cdot p_1}$ (for unknown secret key x and order $q_1 \cdot p_1$). What is more, we are allowed to make several queries. Thus, let us assume we have k such values, i.e. $h_1, \ldots, h_k, s_1, \ldots, s_k$.

We will now show that we can reconstruct $q_1 \cdot p_1$, i.e. the secret trapdoor and order of g. First note that by definition we have:

$$\begin{aligned} x \cdot s_i + h_i \cdot s_i &\equiv 1 \pmod{q_1 \cdot p_1} \\ x \cdot s_j + h_j \cdot s_j &\equiv 1 \pmod{q_1 \cdot p_1} \end{aligned}$$

for some $i, j \in \{1, \ldots, k\}$. Multiplying the first congruential equation by s_j and the second one by s_i, we receive:

$$\begin{aligned} x \cdot s_i \cdot s_j + h_i \cdot s_i \cdot s_j &\equiv s_j \pmod{q_1 \cdot p_1} \\ x \cdot s_j \cdot s_i + h_j \cdot s_j \cdot s_i &\equiv s_i \pmod{q_1 \cdot p_1}. \end{aligned}$$

We now subtract the second equation from the first one and receive:

$$(h_i - h_j) \cdot s_i \cdot s_j \equiv (s_j - s_i) \pmod{q_1 \cdot p_1}.$$

It follows that:

$$(h_i - h_j) \cdot s_i \cdot s_j - (s_j - s_i) \equiv 0 \pmod{q_1 \cdot p_1}.$$

Thus, by computing the value

$$((h_i - h_j) \cdot s_i \cdot s_j - (s_j - s_i)) \tag{1}$$

in $\mathbb{Z}$ we receive a multiple of the hidden order $q_1 \cdot p_1$. What is more, this value is, with high probability, not equal to 0.

We can apply the same computations for different pairs of values receiving a different multiple of the hidden order. Given several multiples we can with high probability reconstruct the hidden order by applying the greatest common divisor algorithm and taking the absolute value. Note that there exist $\binom{k}{2}$ distinct pairs, thus we can compute, with high probability, $\binom{k}{2}$ unique multiples of the hidden order.

Success Probability

We begin the discussion with the following lemma.

Lemma 1. *k random integers are coprime with probability $\frac{1}{\zeta(k)}$, where ζ is the Riemann zeta function.*

Proof. The probability that all k integers have a given prime p as a factor is $1/p^k$. So the probability that at least one of them does not have p as a factor is $1 - 1/p^k$. Therefore, the probability that k integers have no common prime factor is:

$$P_k = \prod_{p \in \mathsf{primes}} (1 - 1/p^k).$$

Using $\frac{1}{1-x} = 1 + x + x^2 + x^3 + \ldots$, this can be rewritten as

$$P_k = \left(\prod_{p \in \mathsf{primes}} (1 + 1/p^k + 1/p^{2 \cdot k} + 1/p^{4 \cdot k} + \ldots) \right)^{-1}.$$

Now by the fundamental theorem of arithmetic (every integer greater than 1 either is prime itself or is the unique product of prime numbers), we get:

$$P_k = \left(1 + 1/2^k + 1/3^k + 1/4^k + \ldots\right)^{-1},$$

where $1/P_k$ is by definition the Riemann zeta function. Thus, $P_k = \frac{1}{\zeta(k)}$.

Now let $m_1, \ldots, m_k$ be the k multiples of $q_1 \cdot p_1$ computed using equation 1 and let $n_i = m_i/(q_1 \cdot p_1)$. Note that $\gcd(m_1, \ldots, m_k) = q_1 \cdot p_1$ if, and only if $\gcd(n_1, \ldots, n_k) = 1$. It now follows by lemma 1, that the probability that $\gcd(m_1, \ldots, m_k) = q_1 \cdot p_1$ is $\frac{1}{\zeta(k)}$.

Thus, for $k = 3$ we already have a non-negligible success probability of $\frac{1}{\zeta(3)} \approx 0.83$. In other words, given secret keys for two identities in case of IBE or two message-signature pairs in case of the signature scheme, we might break the scheme with probability 0.83 (i.e. CPA and CCA or EUF-CMA).

Remark 1. The series $\frac{1}{\zeta(1)}, \frac{1}{\zeta(2)}, \frac{1}{\zeta(3)}, \ldots$ quickly converges to 1, e.g. $\frac{1}{\zeta(4)} \approx 0.92$ and $\frac{1}{\zeta(8)} \approx 0.996$.

4 Dummy Example

In this section we will show that our attack works by giving a simple example. Since the IBE scheme is more complicated (i.e. involves more computations), we will describe the attack on the proposed signature scheme.

$\mathsf{Setup}_{\mathsf{PKSS}}(\lambda)$

Let $p = 227 = 2 \cdot p_1 + 1 = 2 \cdot 113 + 1$ and $q = 263 = 2 \cdot q_1 + 1 = 2 \cdot 131 + 1$. Thus, the hidden order is $q_1 \cdot p_1 = 14803$ and the modulus is $n = p \cdot q = 59701$. We choose the generator $g = 33413$ of order $q_1 \cdot p_1$ and the secret key $x = 11819$. Compute the value $g_1 = g^x \mod n = 2127$. By definition $\ell = 13 < \log_2(14803)$ and we define the hash function h as SHA-256 truncated to ℓ least significant bits. The public key is $\mathsf{pk} = (59701, 33413, 38048, h)$ and the secret key is $\mathsf{sk} = (11819, 14803)$.

The Attack

Let m_1 ="Hello World", m_2 ="My second message" and m_3 ="My third message" be three messages signed by a user knowing the secret key sk. By computing the hash values of this messages we get: $h_1 = h(m_1) = 5230$, $h_2 = h(m_2) = 6031$ and $h_3 = h(m_3) = 1455$. The signatures for this messages are $s_1 = 3737$, $s_2 = 5791$ and $s_3 = 8055$. We can check that this are valid signatures by computing $\hat{g}_i = (g_1 \cdot g^{h_i})^{s_i}$. Thus, we have (computations in $\mathbb{Z}_n^*$):

$$\hat{g}_1 = (2127 \cdot 33413^{5230})^{3737} = (2127 \cdot 10563)^{3737} = 19925^{3737} = 33413 = g$$
$$\hat{g}_2 = (2127 \cdot 33413^{6031})^{5791} = (2127 \cdot 7398)^{5791} = 34183^{5791} = 33413 = g$$
$$\hat{g}_3 = (2127 \cdot 33413^{1455})^{8055} = (2127 \cdot 48073)^{8055} = 43159^{8055} = 33413 = g$$

so s_1, s_2, s_3 are indeed valid.

Now, we compute (computations in integers):

$$\delta_{1,2} = (h_1 - h_2) \cdot s_1 \cdot s_2 - (s_2 - s_1) = -17334416621$$
$$\delta_{1,3} = (h_1 - h_3) \cdot s_1 \cdot s_3 - (s_3 - s_1) = 113633290307$$
$$\delta_{2,3} = (h_2 - h_3) \cdot s_2 \cdot s_3 - (s_3 - s_2) = 213454404616$$

and the greatest common divisor of those values, i.e. $\gcd(\delta_{1,2}, \delta_{1,3}, \delta_{2,3}) = 14803 = q_1 \cdot p_1$. Thus, we reconstructed the hidden order $q_1 \cdot p_1$ only using public values, i.e. the hash values of the signed messages h_1, h_2, h_3 and the corresponding signatures s_1, s_2, s_3. It follows that we can forge a signature for any message.

5 The TSDH and TSEI Problems

In this section we discuss the assumption under which the authors of [1] prove their schemes.

Definition 1 (q-Trapdoor Subgroup Diffie-Hellman Problem). *The q-TSDH problem is defined as follows: given $\left(n, g, g^x, g^{(x+r^*)y}, r^*, \{1/(x+r_i), r_i\}_{i=1}^q\right)$ as input, under the condition that g is the generator of order $q_1 \cdot p_1$ trapdoor subgroup of $\mathbb{Z}_n^*$, $g, g^x, g^{(x+r^*)y}$ are in $\mathbb{Z}_n^*$, $r^*, r_1, \ldots, r_q \in \{0,1\}^\ell$ for some ℓ (less than $\log_2(q_1 \cdot p_1)$) and $(1/(x+r_i))$ are in $\mathbb{Z}_{q_1 \cdot p_1}$, to output $g^y \in \mathbb{Z}_n^*$.*

Definition 2 (q-Trapdoor Subgroup Exponent Inversion Problem). *The q-TSEI problem is defined as follows: given $(n, g, g^x, \{1/(x+r_i), r_i\}_{i=1}^q)$ as input, under the condition that g is the generator of order $q_1 \cdot p_1$ trapdoor subgroup of $\mathbb{Z}_n^*$, g, g^x are in $\mathbb{Z}_n^*$, $r_1, \ldots, r_q \in \{0,1\}^\ell$ for some ℓ (less than $\log_2(q_1 \cdot p_1)$) and $(1/(x+r_i))$ are in $\mathbb{Z}_{q_1 \cdot p_1}$, to output a new pair $(1/(x+r^*), r^*) \in \mathbb{Z}_{q_1 \cdot p_1} \times \{0,1\}^\ell$.*

The advantage of the adversary in solving those problems is defined as the probability that he returns the required output. It should be obvious from section 3, that for $q > 2$ those problems are not hard. In fact, there exists an adversary that has non-negligible advantage in solving those problems.

In both cases, the adversary first uses the attack described in section 3 to reconstruct the hidden order $q_1 \cdot p_1$ and then uses this knowledge to compute x. Knowing those values, he can easily compute $1/(x + r^*) \in \mathbb{Z}_{q_1 \cdot p_1}$ and g^y in case of the q-TSDH problem and $(1/(x + r^*), r^*) \in \mathbb{Z}_{q_1 \cdot p_1} \times \{0, 1\}^\ell$, for a random r^*, in case of the q-TSEI problem.

Remark 2. The authors investigated the hardness of those problems in the generic group model. We argue that this analysis is not correct as the adversary is not limited to computations in $\mathbb{Z}_n^*$ (or it's subgroup). In fact, our attack uses computations in integers, which are not considered in their analysis.

6 Conclusions

In this short report, we have shown an attack against the scheme in [1]. It completely breaks the security of the IBE and signature schemes proposed in this paper. The attack uses computations in integers which are not considered by the author's in their generic group model analysis.

References

1. Park, J.H., Lee, K., Lee, D.H.: Efficient identity-based encryption and public-key signature from trapdoor subgroups. Cryptology ePrint Archive, Report 2016/500 (2016), http://eprint.iacr.org/
2. Shoup, V.: Advances in Cryptology — EUROCRYPT '97: International Conference on the Theory and Application of Cryptographic Techniques Konstanz, Germany, May 11–15, 1997 Proceedings, chap. Lower Bounds for Discrete Logarithms and Related Problems, pp. 256–266. Springer Berlin Heidelberg, Berlin, Heidelberg (1997), http://dx.doi.org/10.1007/3-540-69053-0_18