Comments on "EIBAS: an efficient identity broadcast authentication scheme in wireless sensor networks"

Yalin Chen[1], Jue-Sam Chou[2,*]

[1] Institute of information systems and applications, National Tsing Hua University, Taiwan

yalin78900@gmail.com

[2,] Department of Information Management, Nanhua University, Taiwan

[2,*]: corresponding author, jschou@mail.nhu.edu.tw

Tel: 886+ (0)5+272-1001 ext.56536

Abstract

Recently, Shm et al. Proposed an efficient identity-based broadcast authentication scheme based on Tso et al.'s IBS scheme with message recovery to achieve security requirements in wireless sensor networks. They claim that their scheme can achieve security requirements and mitigated DOS attack by limiting the times of signature verification failures in wireless sensor networks (WSN). However, we found that the scheme cannot attain the security level as they claimed. We will demonstrate it in this article.

1. Introduction

In 2007, Tso et al. [1] based at Barreto et al.'s scheme [2] proposed an ID-based signature scheme with message recovery, where the message can be recovered by anyone without any secret information, to reduce the total length of the transmitted message in wireless sensor networks in which the communication efficiency is a major concern. In Barreto et al.'s scheme, the length of the transmitted data is 88 bytes, while it is only 68 bytes in Tso et al.'s IBS scheme, assuming the size of message and identity are 20 and 2 bytes, respectively. This is because the original message is not transmitted. In 2013, Shim et al. [3] based on Tso et al.'s IBS scheme proposed an efficient ID-based BA scheme, EIBAS, and claimed that their scheme can satisfy the following security and performance requirements: (1) user authentication and the message integrity. (2) minimization of communication overhead. Especially, they focus on minimizing the communication overhead to assure minimum energy consumption. However, after analysis we found that their scheme at most can be termed as $2^{n/2}$ secure. We will demonstrate the reasons in this article.

2. Review of Shim et al.'s IBS scheme

Shim et al.'s IBS scheme Shim et al. [3], based on Tso et al.'s IBS scheme, consists of four phases: System Initialization, Private Key Extraction, Signature

Generation and Message Broadcast, and Broadcast Authentication (Signature Verification). We only list the differences in each phase.

(1). System Initialization : the parameter $u=(P, P)^{-1}$, rather than $u=(P, P)$ in Tso et al.'s scheme.

(2). Private Key Extraction: this phase is the same as in Tso et al.'s scheme.

(3). Signature Generation and Message Broadcast:

1. The user picks a current timestamp tt_i, chooses r_1, computes u^{r1} and $\alpha = H_1(ID_i, tt_i, u^{r1})$.
2. Computes $\beta = F_1(M) \| (F_2(F_1(M)) \oplus M)$, $r_2 = [\alpha \oplus \beta]_{10}$, and $U=(r_1+r_2)Sk_i$. Then, $\sigma_i = (r_2, U)$ is the signature on M for IDi. The user then broadcast $< ID_i, tt_i, \sigma_i >$ in the wireless network, were ID_i and tt_i are taken to be two bytes.

(4). Broadcast Authentication (Signature Verification)

1. The user computes $\alpha' = H1(ID_i, tt_i, e(U, H(ID_i)P+P_{pub}). u^{r2})$ and $\beta' = [r_2]_2 \oplus \alpha'$.
2. Recover the message $M' = \beta'_{l1} \oplus F_2(_{l2}\beta')$ and accept σ 'as a valid signature of the broadcast message M'(=M) if and only if $_{l2}\beta' = F_1(M)$.

3. The weakness found

After intercepting several broadcast messages $< ID_i, tt_i, \sigma_i >$, $< ID_j, tt_j, \sigma_j >$ from several sensor nodes, an attacker can launch an offline hash collision search attack by randomly choosing a message Ma and computing $\beta_a = F_1(M_a) \| (F_2(F_1(M_a)) \oplus M_a)$. Then, he launches hash collision search by the following two ways:

(1) computes $\alpha_a = r_{2i} \oplus \beta_a$. He then randomly chooses several timestamps, with each $tt_k > tt_i$, such that $\alpha_a = H_1(ID_i, tt_k, e(U_i, H(ID_i)P+P_{pub}). u^{r2i})$. He then broadcasts $< ID_i, tt_k, \sigma_i >$ to the sensor nodes for verifying the correctness. Even, he may sum the U_i part of user i's any two signatures of the broadcast messages, computes $\alpha_a = (r_{2i} + r_{2i}') \oplus \beta_a$, then randomly chooses several timestamps, with each $tt_k > tt_i$, such that $\alpha_a = H_1(ID_i, tt_k, e(U_i + U_i', H(ID_i)P+P_{pub}). u^{r2i+r2i'})$ and then broadcasts $< ID_i, tt_k, \sigma_i' (= ((r_{2i} + r_{2i}'), (U_i + U_i'))) >$ to the sensor nodes for verifying the correctness.

(2) computes $\alpha_a = r_{2j} \oplus \beta_a$. He then randomly fakes a timestamp tt_k, such that $\alpha_a = H_1(ID_j, tt_k, e(U_j, H(ID_j)P+P_{pu}b). u^{r2j})$. He then broadcasts $< ID_j, tt_k, \sigma_j >$ to the sensor nodes for verifying the correctness. Certainly, he also can computes $\alpha_a = (r_{2j} + r_{2j}') \oplus \beta_a$, then randomly fakes a timestamp tt_k, such that $\alpha_a = H1(ID_j, tt_k, e(U_j + U_j', H(ID_j)P+P_{pub}). u^{r2j+r2j'})$, and then broadcasts $< ID_j, tt_k, \sigma_j' (= ((r_{2j} + r_{2j}'), (U_j + U_j'))) >$ to the sensor nodes for verifying the correctness.

Although the above two ways doesn't necessarily find a collision; however, as the protocol runs for enough times, it will inevitably increase the broken possibility.

Formally speaking, Shim et al.'s IBS scheme hides the pairing computation into the hashing function to verify the signature and produce the string α simultaneously, but we found doing so cannot entirely remove the possibility of finding hash collision. Using the above two ways of hash collision search, to some extent, we can say that the security of their scheme is reduced to the strength of the hash function, which makes their scheme not secure enough; especially, when there are many researchers working in the area of finding collisions on the hashing functions worldwide, such as [4, 5, 6]. Due to this and the birthday attack [7], we can say that the security label of their scheme is approximately $O(2^{n/2})$, if the length of the hash function is n and the protocol has run a specific times.

4. Modification

From the weakness found in section 3, we see that the key point is that the message M was not directly bound into the signature and its verification is not performed on the signature, rather it is embedded in the hash value. This makes it suffer from the hash value collision attack. To enhance, we isolate the signature verification process from the hash function and bind message M into the verification. Hence, the Signature Generation and Message Broadcast, and the broadcast authentication (Signature Verification) procedure are slightly modified as follows:

Signature Generation and Message Broadcast

1. Pick a current timestamp tt_i, Compute $\beta = F_1(M) \| (F_2(F_1(M)) \oplus M)$ and $H(\beta)$.
2. Choose $r_1 \in_R Z_q$, and compute $\mu^{r1+H(\beta)}$ and $\alpha = H_1(ID_i, tt_i, \mu^{r1+H(\beta)}) \in \{0, 1\}^{l1+l2}$.
3. compute $r_2 = [\alpha \oplus \beta]_{10}$ and $U = (r_1 + H(\beta))SK_i$. Then, $\sigma_i = (\mu^{r1+H(\beta)}, r_2, U)$ is a signature on M for ID_i. Then, compute $HP = H(\mu^{r1+H(\beta)}, H(\beta, r_2, tt_i)) \cdot P$.

The user then broadcasts *messg*=<ID_i, tt_i, HP, σ_i> in the WSN, where ID_i, and tt_i are taken to be two bytes.

Broadcast Authentication (Signature Verification)

After receiving the broadcast message *messg*, each sensor node verifies its authenticity. It first checks whether the timestamp tti is valid or not. If it is valid, the sensor node looks up the revocation list to determine that IDi is not in the revocation list. The sensor node proceeds with the following signature verification:

1. Compute $VS = e(U, H(ID_i)P + P_{pub})$. If $VS = \mu^{r1+H(\beta)}$, compute $\alpha' = H_1(ID_i, tt_i, \mu^{r1+H(\beta)})$, $\beta' = [r_2]_2 \oplus \alpha'$, and $HP' = H(\mu^{r1+H(\beta)}, H(\beta', r_2, tt_i)) \cdot P$.

2. If HP'=HP, recover the message $M' = |\beta'|_{l1} \oplus F_2(_{l2}|\beta'|)$ and accept σ_i as a

valid signature of the broadcast message *M*.

If this verification succeeds, the authenticity of the received message is guaranteed. Compared to the original scheme, the signature verification in this phase requires other two computations, two hash operations H() and one point multiplication in G_1, but does not require the $F_1()$ hash operation to see if $_{l2}|\beta'| = F_1(M')$.

Analysis

(1). Security

In our modification, VS confirmed that ID_i, r_1+ H(β) has not been alerted and HP confirmed that β', r_2, tt_i are the same as in the sending node which totally assures that message M is correctly constructed. In other words, the message relevant parameters β cannot be changed. Therefore, if an attacker launches an attack (changing β and r_2 to find the fake α**,** then using hash collision to find the pre-image of this fake α) on the modification, like ours on the original scheme. He is doomed to be failing, because the sending node committed two values, σ_i and HP, in the sent message which will be subsequently examined by the received node in the broadcast authentication phase. In other words, the security of our modification does not simply rely on the strength hashing function but also depends on the robustness of the signature scheme. In addition, the hash value of β is hidden in the is exponents of $_{\mu}{}^{r1+\ H(\beta)}$, and rehashed and hidden in the coefficient of the point HP. Even if the hash collision is found, our scheme remains secure still.

(2) Computational cost

Compared to the original scheme, our modification extra need one hash operation on β in the Signature generation phase, and one hash operation and one point multiplication in the formation of HP in the broadcast authentication phase. Totally, it needs two hash operations and one point multiplication (We denote this scalar multiplication as SM.). However, it eliminates the computations of one modulo exponentiation u^{r2} (ME) and one modulo multiplication (MM), e(U, $H(ID_i)P+P_{pub}$). u^{r2} , in G_2, in step one of the broadcast authentication phase, and but does not require the $F_1()$ hash operation in the broadcast authentication phase. According to [8], we see that a bilinear pairing is approximately 218 times the cost of a 1024-MM and that a g^k mod p (where p is a 1024-bit prime) operation is estimated as 1.5 |k| times the cost of a 1024-bit modular multiplication (1024-MM in brief) by using square-and-multiply algorithm. If we use the operation MM as the basis, we see that our modification needs one SK which is approximately 29.1 MM and the two hash operations. However, the original scheme needs one ME u^{r2}

which is approximately 1.5 | r_2| (= 1.5(l1+l2)) MM. Obviously, if we ignore the cost of the two hash operations, the modification's computational cost is approximately only 29.1/ 1.5(l1+l2) (= 29.1/ (1.5∗252)) =0.077) times the original scheme if q is a 1024-bit prime. Although, we cannot know the exact number of times when q's length is decreased, it is clear that the scale should be decreased in some proportion to q's bit length (Here, q is 252 bits.). In other words, our scheme is more efficient than the original one.

5. Conclusion

In this paper, we demonstrated that the strength of Shim et al.'s EIBAS is based on the hash function. We therefore modified it to enhance its security and promote its efficiency. From the analysis shown in section 5, we see that we have attained the goal.

References

[1] R. Tso, C. Gu, T. Okamoto, E. Okamoto, “Efficient ID-based digital signatures with message recovery”, in: *Proceedings of CANS’07, LNCS 4856*, Springer-Verlag, 2007, pp. 47–59.

[2] P.S.L.M. Barreto, B. Libert, N. McCullagh, J. Quisquater, Efficient and provably-secure identity-based signatures and signcryption from bilinear maps, in: *Proceedings of Asiacrypt’05, LNCS 3778*, Springer-Verlag, 2005, pp. 515–532.

[3] Shim, Kyung-Ah, Young-Ran Lee, and Cheol-Min Park. "EIBAS: An efficient identity-based broadcast authentication scheme in wireless sensor networks", Ad Hoc Networks 11.1 (2013): 182-189.

[4] Guneysu, T. ; Paar, C. ; Schage, S., “Efficient Hash Collision Search Strategies on Special-Purpose Hardware”, *LECTURE NOTES IN COMPUTER SCIENCE*; 4945; 39-51, Western European workshop on research in cryptology, WEWoRC 2007

[5] Aoki, Kazumaro, and Yu Sasaki. "Meet-in-the-middle preimage attacks against reduced SHA-0 and SHA-1." *Advances in Cryptology-CRYPTO 2009*. Springer Berlin Heidelberg, 2009. 70-89.

[6] Guo, Jian, et al. "Advanced meet-in-the-middle preimage attacks: First results on full Tiger, and improved results on MD4 and SHA-2." *Advances in Cryptology-ASIACRYPT 2010*. Springer Berlin Heidelberg, 2010. 56-75.

[7] Stinson, Douglas R. Cryptography: *theory and practice*. Vol. 36. CRC press, 2006.

[8] Chou, Jue-Sam, Yalin Chen, and Tsung-Heng Chen. "An efficient session key generation for NTDR networks based on bilinear paring." *Computer Communications* 31.14 (2008): 3113-3123.