International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Cryptanalysis of Identity Transmission Authentication System

Sattar J. Aboud

University of Bedfordshire, Department of Computer Science & Technology, Luton, United Kingdom

Abstract: ***Recently, Shm et al. introduced the efficient identity-typed transmission authentication scheme relied on some system in order to reach security properties in wireless sensor networks. They claim that their system can attain security properties and alleviated* denial-of-service attack *by preventive the times of signature verification nonsuccess in wireless sensor networks. But, we discovered that a scheme does not achieve the security requirements as they claimed. We will show that in this paper.***

Keywords: identity signature scheme, transmission authentication scheme, Wireless Sensor Networks, denial-of-service attack.

1. Introduction

In 2007, Tso *et al.* [1] introduced an identity signature scheme with message recovery, whilst a message can be improved by someone without any secret data, to decrease the total size of the broadcast message in wireless sensor networks in which the communication efficiency is the main concern. But, in Barreto *et al.* system [2], the size of the broadcasted information is 88 bytes, while it is only 68 bytes in Tso *et al.* system, supposing the length of message and identity are 20 and 2 bytes respectively. This is because an original message is not broadcasted. In 2013, Shim *et al.* [3] relied on Tso *et al.* system presented an efficient identity-based broadcast authentication scheme, and claimed that their system can fulfill the security properties. Such as entity authentication and the message integrity; reducing overhead communication. However, they concentrate on decreasing the communication overhead to guarantee minimum power use. But, in the discussion it found that their system generally as $2^{n/2}$ secure. We will show the causes in this paper.

2. Review of Shim *et al.* System

Shim *et al.* scheme [3], relies on Tso *et al.* system, and contains four protocols. We are going to list the dissimilarities in every protocol.

1. Initialization Protocol: the key $b = (p, p)^{-1}$ rather than $b = (p, p)$ in Tso *et al.* system.
2. Extraction Protocol: this protocol is the same as in Tso *et al.* system.
3. Signature Generation Protocol:
   - Selects the present timestamp T_i
   - Selects x_1
   - Finds b^{x_1}
   - Finds $z = h_1(Id_i, T_i, b^{x_1})$
   - Finds $c = f_1(m) \| f_2(f_1(m)) \oplus m)$
   - Finds $x_2 = [z \oplus c]_{10}$
   - Finds $d = (x_1 + x_2)w_i$.
   - Finds $s_i = (x_2, d)$ the signature on m for Id_i.
   - Transmits (Id_i, T_i, s_i) to the wireless network were Id_i and T_i are taken two bytes.
4. Signature Verification Protocol
   - Finds $z' = h_1(Id_i, T_i, g(d, h(Id_i)P + P_e)b^{x_2})$
   - Finds $c' = [x_2]_2 \oplus z'$
   - Decrypt the message $m' = c'_{11} \oplus f_2({}_{12}c')$ and accept s' as a valid signature of the transmit message $m'(= m)$ if and only if ${}_{12}c' = f_1(m)$.

3. The Vulnerability

Upon intercepting transmissions messages $(Id_i, T_i, s_i), (Id_j, T_j, s_j)$ from some sensor nodes, a hacker can start an offline hash value search attack by arbitrarily picking the message v and find $c_r = f_1(v) \| (f_2(f_1(v)) \oplus v)$. Then, he starts hash value search by two steps which are as follows:

Step 1: The User

1. Finds $z_r = x_{2i} \oplus c_z$
2. Selects arbitrarily some timestamps, with every $T_k > T_i$
3. Finds $z_r = h_1(Id_i, T_k, g(d_i, h(Id_i)P + P_e)\ b^{x_{2i}})$
4. Transmits (Id_i, T_k, s_i) to the sensor nodes for checking the correctness.
5. Sum the d_i part of user $i's$ any two signatures of the transmission messages
6. Finds $z_r = (x_{2i} + x_{2i'}) \oplus c_r$
7. Selects arbitrarily some timestamps, with every $T_k > T_i$
8. Finds $z_r = h_1(Id_i, T_k, g(d_i + d_i', h(Id_i)P + P_e)b^{x_{2i} + x_{2i}'})$
9. Transmits $(Id_i, T_k, s_i'(= ((x_{2i} + x_{2i}'), (d_i + d_i'))))$ to sensor nodes for checking correctness.

Step 2: The User

1. Finds $z_r = x_{2j} \oplus c_r$.
2. Fakes arbitrarily a timestamp T_k
3. Finds $z_r = h_1(Id_j, T_k,\ g(d_j, h(Id_j)P + P_e)\ b^{x_{2j}})$

4. Transmits (Id_j, T_k, s_j) to the sensor nodes for checking the correctness.
5. computes $z_r = (x_{2j} + x'_{2j}) \oplus c_r$
6. Fakes arbitrarily a timestamp T_k
7. Computes $z_r = h_1(Id_j, T_k, g(d_j + d'_j, h(Id_j)P + P_e)b^{x_{2j}+x'_{2j}})$
8. Transmits $(Id_j, T_k, s'_j (= ((x_{2j} + x'_{2j}), (d_j + d'_j)))$ to sensor nodes for checking correctness.

While the above two steps are not essentially find the collision. Since a protocol runs for sufficient times, it will certainly increase the broken opportunity.

The Shim *et al*. system conceals a pairing calculation into a hashing function to check the signature and create the string z, but we found that it cannot completely remove the possibility of getting hash collision. With the above two steps of hash search, we can state that the security of their system is reduced to the power of a hash value, which constructs their system but not secure sufficient, particularly if there are many authors researching in the field of finding collisions on the hash functions global [4, 5]. Because of this and the birthday attack [6], we can state that a security label of their system is about $O(2^{n/2})$, if a size of a hash function is n.

4. Modifications

We observe from the vulnerability described in section 3 the main point is a message m was not straight bound into a signature and its verification is not done on a signature; instead it is embedded in a hash function. This causes it suffer from a hash function collision attack. To improve, we separate a signature verification operation from the hash value and bind message m into verification. Thus, a signature generation, and the signature verification processes are slightly customized as follows:

A. Signature Generation

1. Selects a present timestamp T_i
2. Finds $c = f_1(m) \| (f_2(f_1(m)) \oplus m)$
3. Finds $h(c)$
4. selects $x_1 \in Z_q$
5. Finds $b^{x_1+h(c)}$
6. Finds $z = h_1(Id_i, T_i, b^{x_1+h(b)})$
7. Finds $x_2 [z + c]_{10}$
8. Finds $d = (x_1 + h(c))w_i$
9. Finds $s_i = (b^{x_1+h(b)}, x_2, d)$ is a signature on m for Id_i
10. Finds $y = h(b^{x_i+h(c)}, h(c, x_2, T_i)P$
11. Transmits the message (Id_i, T_i, y, s_i) in a sensor node, with Id_i, and T_i are taken two bytes.

B. Signature Verification

Upon receiving a transmission message (Id_i, T_i, y, s_i), every sensor node checks its validity. First, verifies if timestamp T_i is valid or not. When it is valid, a sensor node researches the revocation list to decide if Id_i is not in a revocation list. The sensor node continues with the following steps of the signature verification:

1. Finds $n = g(d, h(Id_i)P + P_e$
   If $n = b^{x_1+h(c)}$ then
   Finds $z' = h_1(Id_i, T_i, b^{x_1+h(c)})$
   Finds $c' = [x_2]_2 \oplus z'$
   Finds $y' = h(b^{x_1+h(c)}, h(c', x_2, T_i))P$
2. If $y' = y$ then
   Decrypts a message $m' = | c' |_{l1} \oplus f_2(_{12} | c' |)$
   Determine s' as the valid signature of the transmission message m.

When the verification succeeds, a validity of the received message is certain. Then, compare it with the original system. However, the signature verification in the proposed protocol needs two hash operations $h()$, two computations, and one multiplication, but does not need $f_1()$ hash operation.

C. Discussion

In this section, we discuss the analysis of the scheme in term of security and computing cost.

1. Security

In the alteration, n approved that $Id_i, x_1 + h(c)$ has not been changed and y approved that c', x_2, T_i are the same as in the transmitting node which entirely guarantee that message m is properly built. However, a message relevant key c cannot be altered. Thus, if a hacker starts an attack altering c and x_2 to get the fake z, then using hash function to get the pre-image of this fake z on an alteration, like an original system. It is fated to be worsening, as the transmitting node committed two values, s_i and y, in the transmit message which can be then studied by the received node in the transmission authentication protocol. However, the security of the alteration does not simply base on the power hash value but also bases on a strength of a signature scheme. Also, the hash value of c is concealed in the exponents of $b^{x_1+h(c)}$, and rehashed and concealed in the coefficient of the point y. Though a hash value is found, the proposed system remains secure.

2. Computing Cost

Compared with an original scheme, the proposed modification require one hash operation on c in the signature generation protocol, and one hash operation and one

multiplication in the creation of y in the broadcast verification protocol. In total, it requires two hash operations and one multiplication. Though, it reduces the calculations of one modulo exponentiation b^{x_2} and one modulo multiplication $g(d, h(Id_i)P + P_e)b^{x_2}$, in G_2 in step one of the broadcast verification protocol, and does not need $f_1()$ hash operation in a broadcast verification protocol. According to Chou *et al.* scheme [7], it observes that the bilinear pairing is about 218 times the cost of 1024 modulo multiplication and that p is 1024-bit prime, operation is expected as $1.5|k|$ times the cost of 1024-bit modular multiplication, using square-and-multiply method. When it uses the operation modular multiplication as the basis, it observes that the proposed modification requires one w which is about 29.1 modular multiplication and the two hash operations. But, an original scheme requites one modulo exponentiation b^{x_2} which is about $1.5|x_2|(=1.5(11+12)$ modular multiplication. Clearly, when it discounts the cost of the two hash operations, a modification computing cost is about $29.1/1.5(11+12)(=29.1/(1.5*252))=0.077)$ times an original scheme when q is 1024-bit prime. While, we do not know an exact number of times if $q's$ size is reduced, it is clear that the scale must be reduced in some amount to $q's$ bit size (q is 252 bits.). However, the proposed system is more efficient than an original one.

5. Conclusions

In this paper, we verified that the power of Shim *et al.*'s scheme is relied on a hash function. So we tailored it to improve its security and raise its efficiency. From the discussion illustrated in section 5, we observe that we have achieved the objective.

References

[1] Tso R., Gu C., Okamoto T., Okamoto E., "Efficient ID-based digital signatures with message recovery", Proceedings of CANS ' 07, pp. 47–59, LNCS 4856, Springer-Verlag, 2007.
[2] Barreto P.S.L.M., Libert B., McCullagh N., Quisquater J., "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps", Proceedings of Asiacrypt'05, LNCS 3778, pp. 515–532, Springer-Verlag, 2005
[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, pp. 182-189, 2013.
[4] Guneysu T., Paar C., Schage S., "Efficient Hash Collision Search Strategies on Special-Purpose Hardware", LNCS, 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, pp. 70-89, 2009.
[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, pp. 56-75 2010.
[7] 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, pp. 3113-3123, 2008.

Author Profile

Sattar J. Aboud, received his Master degree in 1982 and a PhD in 1988 in the area of computing system. The two degrees were awarded from U.K. In 1990, he joined the Institute of Technical Foundation in Iraq as an assistant professor. In 1995 he joined the Philadelphia University in Jordan as a chairman of computer science department. Then, he moved as a professor at the Middle East University for Graduate Studies, Amman-Jordan. Currently, he is a visiting professor at university of Bedfordshire in UK. His research interests include areas like public key cryptography, digital signatures, identification and authentication, and networks security. He has supervised numerous PhDs and Masters Degrees thesis. He has published more than 60 research papers in a multitude of international journals and conferences.