

Hash-based Secure Sensor Network Programming Method without Public Key Cryptography

Sokjoon Lee

RFID/USN Security Research Team
ETRI, 161 Gajeongdong
Yuseong-gu, Daejeon, S. Korea
+82-42-860-5455

junny@etri.re.kr

Howon Kim

RFID/USN Security Research Team
ETRI, 161 Gajeongdong
Yuseong-gu, Daejeon, S. Korea
+82-42-860-6288

khw@etri.re.kr

Kyoil Chung

Information Security Research
Division
ETRI, 161 Gajeongdong
Yuseong-gu, Daejeon, S. Korea
+82-42-860-1920

kyoil@etri.re.kr

ABSTRACT

Network programming or over-the-air programming is very important function for wireless sensor networks (WSN). Because sensor nodes are updated with wireless connection, there could be many security threats, so we need cryptographically strong protocol. Some researches solve this problem by adapting digital signature and hash function. But digital signature based PKI needs many computational overhead, therefore these may not be acceptable for wireless sensor node.

In some typical circumstances, the sensor network can have a few kinds of restriction. For example, sensor nodes can synchronize the time among them. Or broadcast message can be received over single hop. In this paper, we propose new secure network programming method using only hash function instead of PKI-based digital signature, which can be applied in those circumstances.

Categories and Subject Descriptors

C.2.2 [Computer-Communication Networks]: Network Protocols; D.4.6 [Operating Systems]: Security and Protection

General Terms

Algorithms, Design, Reliability, Security

Keywords

Wireless Sensor Networks, Network Programming, Deluge, TinyOS, Security, Broadcast, Hash Chain

1. INTRODUCTION

Network programming or over-the-air programming is very important function for wireless sensor networks (WSN). Deluge[1], Infuse[2] and Multi-hop Network Programming[3] are some examples of network programming protocol. They are used to update sensor

nodes over wireless connection, but do not provide any security, making the sensor network vulnerable to some kinds of simple attacks.

Some researches[4,5,6] solve this problem by adapting or modifying Gennaro and Rohatgi's scheme[7], which uses digital signature and hash function. But digital signature based PKI needs many computational overhead and large memory size, therefore these may not be acceptable for wireless sensor node.

In some circumstances, the sensor network can have a few kinds of restriction. First, we think the environment where sensor nodes can synchronize the time among them. The base station may send the message for time synchronization or the nodes on mobile phones can have ability of that. For the second example, broadcast message is sent so powerfully from the base station that the message can be received over single hop. In two examples, the broadcast message could be for one-time use. Sensor nodes can ignore the fake or modified message based on original one from the base station.

In this paper, we propose new secure network programming method using only hash function instead of PKI-based digital signature. By eliminating public key, our method is more efficient than the past researches.

2. RELATED WORK

2.1 Network Programming & Deluge

Wireless sensor network has been applied to various fields, from military application to monitoring system of forest fire. In most sensor network applications, nodes in the networks must be active in a long time (ex. some months to years) and able to be updated dynamically. The protocol of dynamic program update is called as network programming or over-the-air programming. In network programming, the program binary image is generally fragmented to be transported from base station to sensor nodes.

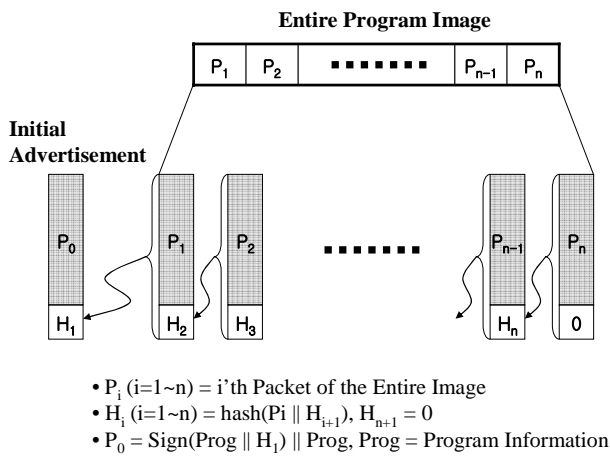
Deluge[1], Infuse[2] and Multi-hop Network Programming[3] are some examples of the protocol. Among them, Deluge is included in TinyOS project[9] and

used most popularly in the TinyOS based networks. It supports multi-hop network and permits different applications on different nodes in the network.

2.2 Secure Network Programming

Network programming models such as Deluge, etc didn't have any security consideration. So there have been some researches[4,5,6] to solve this problem recently. They adapt or modify Gennaro and Rohatgi's scheme[7], which uses digital signature and hash function for signing digital streams efficiently.

[Figure 1] shows the process used to transform a program image to a stream for secure network programming. The process of each research is different from [Figure 1] in detail, but not in many points.



[Figure 1] Process for secure network programming

2.3 Hash Chain

Hash chain[8] is made by applying a one-way hash function $hash()$ such as MD5 or SHA recursively to an initial secret s .

$$h_1 = s, h_2 = hash(h_1), \dots, h_n = hash(h_{n-1}) = p$$

If anyone knows only the public information $p(=h_n)$, he cannot compute h_{n-1} because of one-way characteristic of the hash function. So we can think h_n is the public value and h_{n-1} is the secret value. Once a trusted entity who knows the secret presents h_{n-1} publicly, h_{n-1} will be the public value and h_{n-2} will be the secret value and so on.

Due to its PKI-like feature and simplicity, hash chain has been employed in a variety of applications such as password based authentication, micropayments, and sensor network security protocols.

3. REQUIREMENTS

3.1 Threat Model

In wireless network, any attacker can access, eavesdrop on and counterfeit the packets if no security protocol is

applied to the network. Sensor nodes can be compromised or fake nodes can be created to be as normal ones by the attacker. Sensor node is generally regarded as no tamper-proof device because it must be very cheap and simple. If the attacker compromised a node, he could get private information easily from the node.

Base station would not be spoofed or damaged physically. It stores the hash chain securely and opens one secret value when it distributes new program.

3.2 Security Requirements

Broadcast message for the distribution of new program must be received without any modification. If the message was modified or counterfeited, sensor nodes should be able to notice it. Even if an attacker compromises some nodes, he should not be able to forge the broadcast message using the information from the nodes.

3.3 Assumption for the Network Environment

As we stated in Chapter 1, our solution needs some kinds of restriction. First, we think the environment where sensor nodes can synchronize the time among them. The base station may broadcast the message for time synchronization or the nodes on mobile phones can have ability of that. For the second example, broadcast message is sent so powerfully from the base station that the message can be received over single hop.

In Duttal et al[4]'s scheme, we don't need these assumptions because of the digital signature and program image is tightly coupled. The digital signature would be made from the program image and private key of base station, therefore the signature cannot be reused for fake message in the case. But there is no method that secret information in hash chain and program image is highly dependent. So, our solution needs assumptions where the broadcast message has to be for one-time use.

4. OUR SOLUTION

Our Solution is similar to the previous researches[4,5,6], except for not using digital signature. They need PKI-based digital signature to authenticate the advertisement packet, such as RSA. For example, Dutta et al. use the following notation for the packet.

$$M_{adv} = [S, X^{pid}, X^{ver}, N, hash(N, M_1)]_{SK_S}$$

where, S is a base station(or a trusted server), X^{pid} is the program identifier, X^{ver} is the version number, N is a nonce, $hash()$ is a hash function, M_1 is the first packet of the program image($M_1 = P_1 || H_2$ in 2.2), SK_S is the sign key of S and $[X]_{SK}$ is the signed message of X with SK .

Now, we will eliminate the digital signature with hash chain. In Dutta et al's scheme, the public key of the base station is pre-installed on each node. Similarly, in our

scheme, the base station must perform the following processes before each node is distributed in the field.

- The base station computes a hash chain starting s . The chain must be remained secretly in it except p .
- It installs the public information p and hash function $hash()$ in its nodes.

Then, when the base station updates the program of the nodes, it makes the 2-phase advertisement packets as follows.

$$M_{adv1} = S, X^{pid}, X^{ver}, N, n-1, hash(S, X^{pid}, X^{ver}, N, n-1, h_{n-1}, M_1)$$

$$M_{adv2} = h_{n-1}, hash(S, X^{pid}, X^{ver}, N, n-1, h_{n-1}, M_1)$$

If a node gets M_{adv1} , it saves the counter $n-1$ which means the order of hash chain. It will ignore any M_{adv1} with the counter $n-1$. Subsequently, when the node receives M_{adv2} , it verifies h_{n-1} by comparing $hash(h_{n-1})$ with h_n . It can verify M_{adv1} when it gets M_1 .

In the environment where the nodes can synchronize time among them, if program update message will be sent periodically and appropriate time interval is needed between M_{adv1} and M_{adv2} , the attacker cannot counterfeit them. Also, in single hop application, the attacker cannot counterfeit M_{adv1} . He cannot know h_{n-1} in the first phase. He comes to know h_{n-1} in the second phase, but he cannot go back to the first phase because the nodes will not be affected by M_{adv1} with $n-1$ any more.

The base station needs to update the nodes again, $n-1$, h_{n-1} will be replaced by $n-2$, h_{n-2} , and so on. The possible number of updates depends on the value of n .

5. CONCLUSION

We proposed new secure network programming method using only hash function instead of PKI-based digital

signature. In some environments, by eliminating public key, our method would be more efficient than the past researches. In the near future, we will implement our solution and the previous researches on the Deluge protocol for the comparison of them.

There is no consideration of secure re-initialization of hash chain in this paper. If we apply this concept to our work, the number of updates will be unlimited.

6. REFERENCES

- [1] J. W. Hui and D. Culler, The dynamic behavior of a data dissemination protocol for network programming at scale, SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems, 2004
- [2] S. S. Kulkarni and M. Arumugam, INFUSE: A TDMA based data dissemination protocol for sensor networks, Technical report, Michigan State Univ., 2004
- [3] S. S. Kulkarni and L. Wang, MNP: multihop network reprogramming service for sensor networks, ICDCS '05, 2005
- [4] P. K. Dutta, J. W. Hui, D. C. Chu and D. Culler, Securing the Deluge Network Programming System, IPSN '06, 2006
- [5] J. Deng, R. Han and S. Mishra, Secure Code Distribution in Dynamically Programmable Wireless Sensor Networks, IPSN '06, 2006
- [6] P. E. Lanigan, R. Gandhi and P. Narasimhan, Secure Dissemination of Code Updates in Sensor Networks, SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems, 2005
- [7] R. Gennaro and P. Rohatgi, How to sign digital streams, Crypto '97, LNCS 1294, 1997
- [8] L. Lamport, Password Authentication with Insecure Communication, Communications of the ACM 24.11, 1981
- [9] TinyOS, <http://www.tinyos.net/>