Design and Implementation of Hybrid Broadcast Authentication Protocols in Wireless Sensor Networks

ZHAO Xin, WANG Xiao-dong
School of Computer Science, National University of Defense Technology, Changsha 410073, China
{xinzhao_remerci, xdwang}@nudt.edu.cn

Abstract

The proposed broadcast authentication protocols for wireless sensor networks can be divided into two categories: protocols based on digital signature and protocols based on improved message authentication code. This paper implements and evaluates the performance of two broadcast authentication: TinyECC, which is based on ECDSA, and GBA, which is based on improved MAC. Through analysis of the performance difference of the two protocols, this paper proposes a hybrid broadcast authentication protocol (HBA). HBA achieves an appropriate compromise of security and performance according to packet value. The analysis and experiments show that HBA is efficient and practical.

1. Introduction

A wireless sensor network (WSN) is a wireless network consisting of spatially distributed resource constrained sensor nodes which can cooperatively monitor physical or environmental conditions. The wide applications of WSN and the challenges in them have attracted many researchers to develop protocols and algorithms. Broadcasts are important network functions in WSN, they are used in many applications and protocols such as networks query, software updates, time synchronization, multihop routing, and so on. Broadcast authentication is an essential service in WSN that ensure the broadcast packets from a valid source were not altered in transmit.

Many broadcast authentication protocols have been proposed for WSN[1-6]. These protocols can be classified into two categories. One is based on digital signature technologies, such as TinyECC[1], TinyPK[7]. The Other is based on improved message authentication codes (MAC), such as μ TESLA[5], PBA[3]. It's considered by most papers that the former are more secure because digital signature supports non-repudiation. And the latter have better performance in authentication computation. Thus some works were proposed to design hybrid broadcast authentication protocols. These works commonly use digital signature in base station or cluster head, and use improved MAC in sensor nodes.

The idea about more computation efficiency of protocols based on improved MAC comes from the analysis of algorithm complexity, simulation and testbed where broadcast authentication is only applied to packets from base station. So the conclusion is limited. Consequently, the hybrid broadcast authentication protocols based on this conclusion are not practical. In fact, protocols based on improved MAC are computation efficient only when generating and verifying packets with MAC. However, a broadcast authentication protocol includes not only authentication parts, but also key initialization and update parts. Some protocols need further support from additional middle-ware protocols. For example, μ TESLA needs loose time synchronization between broadcast sender and receiver. It also

needs additional memory in receiver to cache packets received. The drawbacks of the current performance analysis for broadcast authentication protocols include:

- Lack of analysis about impact from supporting middle-ware protocols.
- Lack of analysis about memory cost and authentication delay.
- Lack of analysis about impact of network size.

This paper selects TinyECC and GBA to represent the tow categories of broadcast authentication protocols respectively. Through the performance analysis of the two protocols and their difference, this paper proposes a flexible hybrid broadcast authentication protocol, named HBA. HBA is implemented in sensor nodes, and achieves appropriate promise of security and performance of broadcast authentication protocol.

The rest of this paper is organized as follows. The next section gives a brief overview of TinyECC and GBA. Section 3 describes the implementation and experiments of TinyECC and GBA. Section 4 presents the mechanisms and performance of HBA. Section 5 concludes the paper and points the future work.

2. Overview of TinyECC and GBA

TinyECC is a software package providing ECC-based PKC operations that can be flexibly configured and integrated into sensor network applications. It provides a digital signature scheme (ECDSA), a key exchange protocol (ECDH), and a public key encryption scheme (ECIES) on the platform of MICAz, TelosB, Tmote Sky, and Imote2. TinyECC implemented several well known optimizations for ECC algorithm, making it both computation-efficient and storage-efficient. This paper takes ECDSA of TinyECC as a building block of broadcast authentication protocol based on digital signature. For simplicity, the protocol is also called TinyECC.

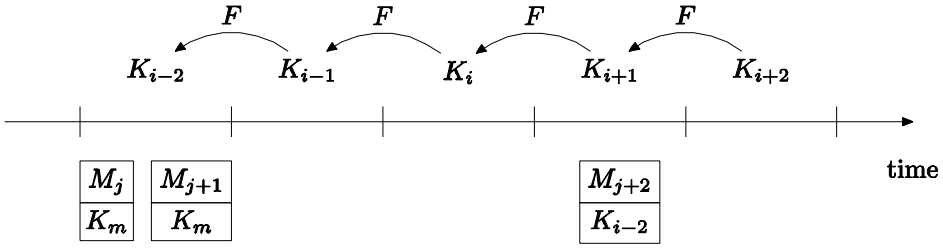

Figure 1 The mechanisms of key assignation and disclosure in μTESLA (d=2)

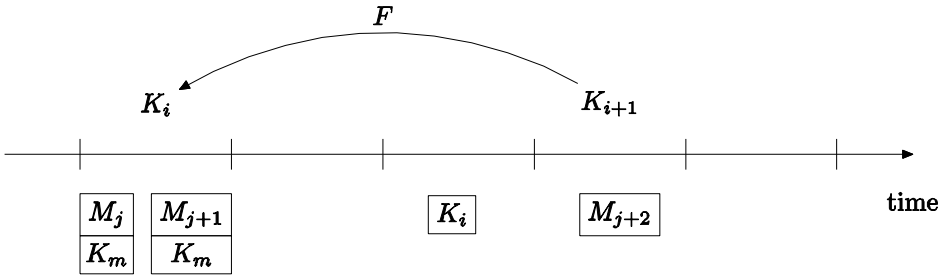

Figure 2 The mechanisms of the key assignation and disclosure in GBA (d=2)

GBA is an early work of the author. It's an improvement on μTESLA protocol. μTESLA is an efficient broadcast authentication protocol based on delayed disclosure of MAC key and one-way key chain technique. However, when applied to applications such as time synchronization and fire alarm in which broadcast messages are sent infrequently, μTESLA encounters problems of obsolete key resources and slow message

verification. With the improvement of the key assignation and key disclosure mechanisms, GBA behaves good performance when broadcast messages are sent both frequently and infrequently. It limits the delay to an acceptable extent and thus improves the robustness of GBA when facing DoS attacks.

For simplicity, this paper only introduces the improvements of GBA to μTESLA. The difference of the two protocols is shown in the figure 1 and figure 2. In μTESLA, the time period for broadcasting is divided into multiple time intervals. Each time interval is associated with a key, which is used to generate MAC for the broadcast messages sent in the time interval. The key is disclosed with broadcast messages sent after a period of time, typically *d* time intervals. If there is no broadcast message sent after *d* time intervals, the key can be got by the receiver only through one-way function computation on the follow-up disclosed keys. In GBA, Only time intervals with broadcast messages needed to send are associated with keys. Thus GBA takes full use of key resources when messages are sent infrequently. Another difference of GBA to μTESLA is the key disclosure mechanism. When there is no broadcast message to take disclosed key in the *d* delayed time interval, GBA will send separate key disclosure message at the beginning of *d+1* time interval. With this change, the receiver only need one one-way function computation to authenticate the key, the authentication delay is also limited to about *d* time intervals.

3. Performance analysis

In order to conduct a systematic performance analysis of TinyECC and GBA, we implement the two protocols on the MICAz platform. Through simulation and testbed, this paper compares the performance difference of them, and points out that their respective advantages and disadvantages. This analysis is important for constructing hybrid broadcast authentication protocol.

The testbed is composed of 15 MICAz motes and a base station. With the implementation on MICAz motes, the protocol design must meet the hardware and environment constraints, making the protocols more practical.

The simulation is conducted on the TOSSIM simulator. TOSSIM provides a scalable, high fidelity simulation of a complete TinyOS sensor network. TOSSIM simulates the network at the bit level, making it more suitable for wireless sensor networks than traditional network simulator such as NS2. Another advantage of TOSSIM is that most codes for simulation can be borrowed from real mote codes, avoiding external programming work. This paper mainly uses TOSSIM for multi-hop broadcast simulation and power consumption analysis. These experiments are difficult to conduct in the testbed due to environment or hardware restrict.

3.1 protocol parameters and underlying protocols

Table 1 shows some parameters of TinyECC and GBA taken in the experimental analysis. These parameters is related to the protocol performance such as authentication delay and computation cost. For example, a long time interval in GBA will cause a large authentication delay. While a short time interval will lead to more key resources consumption.

Both TinyECC and GBA need key initialization and update mechanisms. These mechanisms may be complicated for performance optimization. This paper utilizes the simplified code to ease the implementation. The initialized parameters are preset in the node before network deployment. This mechanism encounters a problem of extendibility. In our experiment, the number of broadcast sender is limited to below 20. So the mechanism is

appropriate. For TinyECC, because of the high security level of public key cryptography and the limited experiment time, there is no key update mechanism. For GBA, when a key chain is used up, the sender needs to generate a new key chain and update the related parameters to all possible receivers. In [2] and [3], Liu et al propose multi-level μTESLA and hash tree mechanisms for this purpose. In GBA implementation, we use a simpler mechanism: The sender generate and broadcast new key related parameters when there are two keys left in the key chain, using the current key for authentication.

Table 1 The protocol parameters

TinyECC	
key length	160bits
GBA	
key chain length	20
duration of a time interval	1s
number of delayed time intervals for key disclosure	2

GBA needs loose time synchronization between senders and receivers. A number of time synchronization protocols have been proposed for WSN. In TOSSIM simulator, it can be achieved by add a running parameter *-b* to the simulation program. In our testbed, the 15 MICAz motes are in the one-hop communication range. So the loose time synchronization problem turns out to be the basic issue of single-hop pairwise time synchronization. This paper utilizes RBS [8] and TPSN [9] mechanism for this purpose.

3.2 Performance analysis

This paper evaluates the performance of TinyECC and GBA in terms of code size, authentication delay, and power consumption.

Memory cost. Table 2 shows the code size of the implementation of TinyECC and GBA in TinyOS. It can be found that the rom size (the program memory cost) of the two protocol is similar. So we can conclude that the broadcast authentication protocols based on improved MAC isn't more efficient than those based on digital signature in terms of program memory cost.

Table 2 Code size requirements of TinyECC and GBA

code size (bytes)	TinyECC	GBA
rom size	27558	25164
ram size	2306	1186

The 4K ram of MICAz mote is used to store variables, including initial key related resources. For every sender, every receiver in TinyECC needs to store a public key of 40 bytes. While every receiver in GBA needs to store 11-byte key related resources for every sender. Both protocols have the scalable problem and it's more severe in TinyECC. If no appropriate mechanisms are designed to solve the problem, the application of TinyECC in large sensor networks will be limited to a few nodes with more memory support. In [6], Ren et al uses bloom-filter to reduce the memory cost of

public key storage. If the identity-based digital signature can be used for WSN, the storage problem may be erased in the broadcast authentication protocols based on digital signature.

Authentication delay. The authentication delay for TinyECC mainly includes delay introduced by signature signing and signature verification. The transmission time of broadcast messages can be ignored when comparing the above two. The experimental results from testbed are shown on table 3. We can find from the results that if the transmission of broadcast messages is a frequent event, and hence there are several messages needed to be signed at the same time, the execution time of TinyECC authentication operations become longer. So TinyECC is not suitable for very frequent broadcast authentication.

Table 3 Execution time of TinyECC

time between two successive transmission	sign	verify
0.8s	2.7s	3.1s
4s	2.0s	2.4s

Table 4 The message cache time of GBA in various message transmission rates

time between two successive transmission	message cache time
0.4s	1.69s
0.7s	1.96s
1.3s	2.2s
1.9s	1.88s
2.2s	2.13s
2.6s	2.32s
4.0s	1.96s
6.0s	1.96s

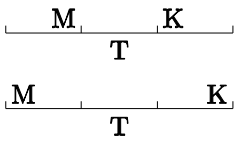

Figure 3 The extreme cases of the time interval between the transmission of messages and disclosed keys

The authentication delay in GBA is another case. It's mainly introduced by the message cache time in the receiver. The generation and verification of MAC is very computation efficient. The execution time in the experiment is typically tens of milliseconds. The message cache time is decided by the time interval between the transmission of broadcast messages and their related disclosed key. The time is variable with different message transmission rate; figure 3 shows the extreme cases. *M* means a broadcast message, *K* means the related disclosed key, *T* represents the time interval of GBA. We conduct a series of experiments according to various message transmission rates and the results are shown in the table 4. From table 3 and table 4 it can be concluded that the authentication delay of GBA is about 2 seconds, roughly a half of that of TinyECC.

Power consumption. This paper uses PowerTOSSIM component of TOSSIM to measure power consumption of TinyECC and GBA. The topology of simulation is shown in figure 4. Each node can only communicate with its neighbour nodes. Every node on the left side of the grid topology broadcast messages every 4 seconds. The duration of simulation is 60 seconds. The simulation results are shown in table 5. From table 5 we find the radio power consumption in GBA is larger than TinyECC. The reason is that in GBA protocol, the senders need broadcast external messages to release disclosed key and to update key related parameters. Due to the more complex algorithm, TinyECC protocol consumes more power than GBA.

Table 5 The power consumption (mJ) of TinyECC, GBA and HBA

	radio	cpu	led	total
TinyECC	698.22	721.32	142.45	1561.99
GBA	787.48	315.66	174.95	1278.09
HBA	735.24	452.18	152.39	1339.81

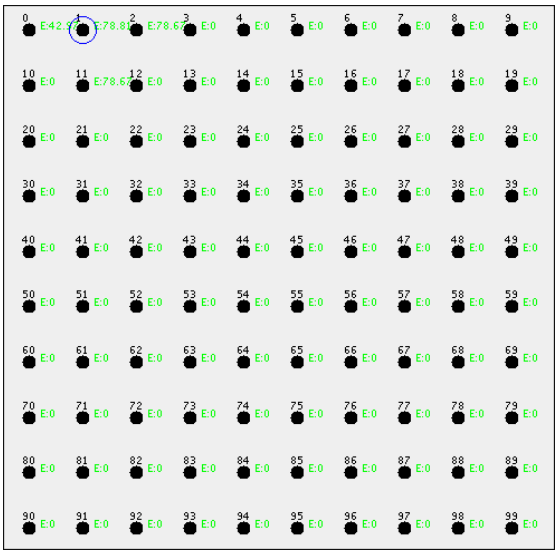

Figure 4 The topology of simulation for TinyECC and GBA

3.3 Conclusion

From the above performance evaluation of TinyECC and GBA through simulation and testbed, it's clear that broadcast authentication protocols based on digital signature can be applied to resource-constraint sensor nodes. The memory cost of TinyECC and GBA is similar. The power consumption of TinyECC can be reduced if we limit the number of packets authenticated by TinyECC. In fact, the time between two successive TinyECC authentications can't be too short. Otherwise the authentication delay will become longer.

4. Hybrid broadcast authentication protocol

With the performance analysis of TinyECC and GBA, this paper tries to utilize the two protocols as building blocks to design a hybrid broadcast authentication protocol named HBA (Hybrid Broadcast Authentication protocol). The design of HBA observes the following rules:

- Every sensor node is able to use TinyECC and GBA for broadcast authentication.

- The time between two successive TinyECC authentications is longer than 5s.
- The number of packets authenticated by TinyECC is limited.

Through component reuse (for example, SHA component can be used for both generation of TinyECC signature and generation GBA key chain), this paper implement both TinyECC and GBA on every MICAz mote. The main design issue of HBA is to choose proper authentication protocol at a moment. For proper use of more secure but more energy-consuming TinyECC protocol, we define a packet value for every broadcast packet. The packet value reflects the importance of the packet. For example, event alarm packets are more important than normal sensing packets.

We simulate HBA with the same simulation environment as section 3.1. Every broadcast packet is assigned a random packet value. The result is shown in the table 5. It can be found that the power consumption of HBA is between that of TinyECC and GBA.

5. Conclusion and future work

This paper implements and evaluates the performance of two broadcast authentication: TinyECC, which is based on ECDSA, and GBA, which is based on improved MAC. Through analysis of the performance difference of the two protocols, this paper proposes a hybrid broadcast authentication protocol (HBA). HBA can achieve an appropriate compromise of security and performance according to packet value.

HBA is a flexible hybrid protocol. If wireless sensor networks have the ability to decide the security situation of the network environment and correlate the security situation state with packet value, HBA will be self-adaptive to the network security situation. It's the future work to include the self-adaptive property into HBA.

References

[1] A. Liu and P. Ning, “TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks,” Department of Computer Science, North Carolina State University, Tech. Rep. TR-2007-36, Nov. 02 2007, mon,05 Nov 2007.

[2] D. Liu and P. Ning, “Multilevel μTESLA: Broadcast authentication for distributed sensor networks,” ACM Trans. Embedded Comput. Syst, vol. 3, no. 4, pp. 800–836.

[3] D. Liu, P. Ning, S. Zhu, and S. Jajodia, “Practical broadcast authentication in sensor networks,” in MobiQuitous. IEEE Computer Society, 2005, pp. 118–132.

[4] M. Luk, A. Perrig, and B. Whillock, “Seven cardinal properties of sensor network broadcast authentication,” in fourthACM workshop on Security of ad hoc and sensor networks, Alexandria, Virginia, USA, 2006, pp. 147–156.

[5] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, “Spins: Security protocols for sensor networks,” Wireless Networks, vol. 8, pp. 521–534, September 2002.

[6] K. Ren, W. Lou, and Y. Zhang, “Multi-user broadcast authentication in wireless sensor networks,” in Fourth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2007), San Diego, CA, June 2007.

[7] R. J. Watro, D. Kong, S. fen Cuti, C. Gardiner, C. Lynn, and P. Kruus, “TinyPK: securing sensor networks with public key technology,” in SASN, S. Setia and V. Swarup, Eds. ACM, 2004, pp. 59–64.

[8] J. Elson, L. Girod, and D. Estrin, “Fine-grained network time synchronization using reference broadcasts,” in ACM SIGOPS Operating Systems Review, 2002, pp. 147–163.

[9] S. Ganeriwal, R. Kumar, and M. B. Srivastava, “Timing-sync protocol for sensor networks,” in Proceedings of the first international conference on Embedded networked sensor systems (SenSys-03). New York: ACM Press, Nov. 5–7 2003, pp. 138–149.

Authors

Zhao Xin was born in 1979. He received the M.S. degree in Computer Science from National University of Defense Technology, and now is a PhD candidate at the university. His research interests include the security of mobile ad hoc networks and wireless sensor networks.

Wang Xiao-dong was born in 1973. He received the PhD degree in Computer Science from National University of Defense Technology in 2001. He is a associate professor and master's supervisor at National University of Defense Technology. His research interests include mobile computing technology.