TELKOMNIKA, Vol. 11, No. 6, June 2013, pp. 3349 ~ 3354
e-ISSN: 2087-278X

A Security Mechanism based on Authenticated Diffie-Hellman for WSN

Xin Yu*, JianJun Fang, ZhaoLi Zhang
College of Automation, Beijing Union University, 100010, Beijing, China
*Corresponding author, emails: zdhtyuxin@buu.edu.cn

Abstract

Wireless sensor network (WSN) has been widely used in industrial technology, national defense, robotic system [1], medical and health field. But it is hedged about in various fields due to its security problem. Through the analysis of the characteristics of Bluetooth WSN and its security architecture, this paper gives out a security scheme for Bluetooth WSN including piconet and scatternet. This scheme based on improved Diffie-Hellman achieves authentication in the Bluetooth WSN and defeats threats derived from Bluetooth link-level.

Keywords: *wireless sensor network, security mechanism, ad hoc, User Authentication*

1. Introduction

Bluetooth network is one of the most widely used wireless sensor network. It was designed as a low-cost, low-power wireless networking technology. However, the current Bluetooth specification limits its usage because of its security problems. Though Bluetooth specification prescribes link-level security, its piconet and scatternet are confronted with great threats such as eavesdropping, replay attack. This paper gives out an integrated security mechanism based on authenticated Diffie-Hellman to resolve problems mentioned above for Bluetooth WSN.

A complete security mechanism for Bluetooth WSN should include Bluetooth device security, piconet security and scatternet security.

2. Bluetooth Device Security

a. Analysis of Bluetooth device security

Bluetooth security includes link key establishment, authentication, E0 Stream Cipher and hopping frequency and so on. It provides four kinds security entities: BD_ADDR, private Link Key, Private Encryption Key and random number. Figure 1 describes the process of Bluetooth device authentication.

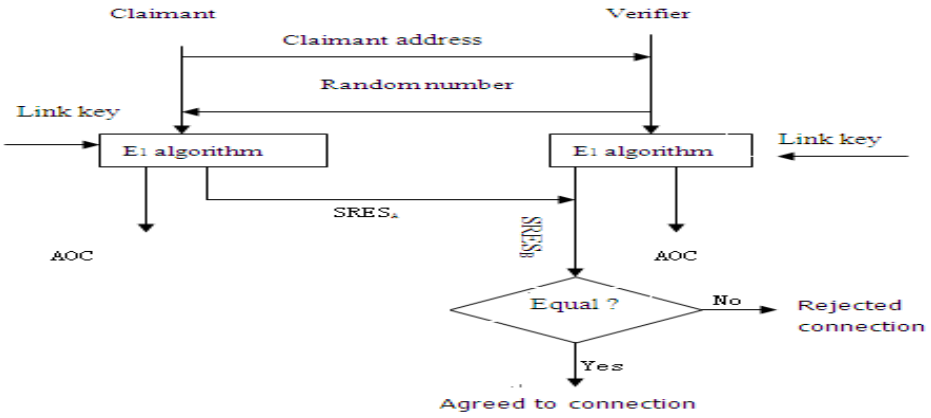


Figure 1. Bluetooth Device Authentication

Received December 24, 2012; Revised April 16, 2013; Accepted April 26, 2013

The verifier device A sends a 128 bit RAND message to the claimant device B. After the receipt of RAND message from A, the B responds with the message SRESB, which is calculated as the first 32 bit of E1 (KEY, BD_ADDRB, RANDA). Then the verifier calculates SRESA= E1 (KEY, BD_ADDRB, RANDA) and if the output matches the received SRESB the device authentication is complete.

Bluetooth devices are susceptible to threats such as denial of service attacks, eavesdropping, man-in-the-middle attacks, message modification [1]. These security vulnerabilities stem from its inherent security mechanism, which the Bluetooth authentication is only in level of device. It require a security strategy in high level such as RFCOMM level.

b. Improved security mechanism for Bluetooth

RFCOMM is a serial line emulation protocol. It emulates RS-232 control, providing transport capabilities for upper level services. References [2] has put forward the solutions based on Diffie-Hellman in the level of RFCOMM. It can achieve user authentication, in stead of device authentication, but it can't avoid replay attack. This paper gives out a improved Diffie-Hellman algorithm. It can avoid replay attack and achieve user authentication.

Step 1: A: EK1 (g^{X_A}); B: EK1 (g^{X_B})

Step 2: A sends message EK1 (g^{X_A}) to B

Step 3: B: DK1 (EK1 (g^{X_A})) →KeyB= $g^{X_A X_B}$ mod q

Step 4 B sends message EKeyB (RANDB) EK1 (g^{X_B}) to A

Step 5 A: DK1(EK1(g^{X_B})) KeyA= $g^{X_A X_B}$ mod q DKeyA (EKeyB (RANDB))

Step 6 A sends message EKeyA (RANDB+1, RANDA) to B

Step 7 B: DKeyB (EKeyA(RANDB+1, RANDA))

Step 8 B sends message EKeyB (RANDA+1) to A:

Step 9 A: DKeyA (EKeyB (RANDA+1))

Parameter integer g is a primitive root of q; XA (XB) is random integer generated in unit A (B). K1 is the current session key. It may be Bluetooth Unit Key or Combination Key. The unit key K is generated with the key-generating algorithm E21 when the Bluetooth device is in operation for the first time. After it has been created, it will be stored in the non-volatile memory of the device and is rarely changed. It is generated as follows: Ki =E21 (RAND, BD_ADDRi). The combination key is generated during the initialization process and generated by both devices at the same time. Figure 2 describes the process of combination key generation. Obviously, combination key K=KA□KB.

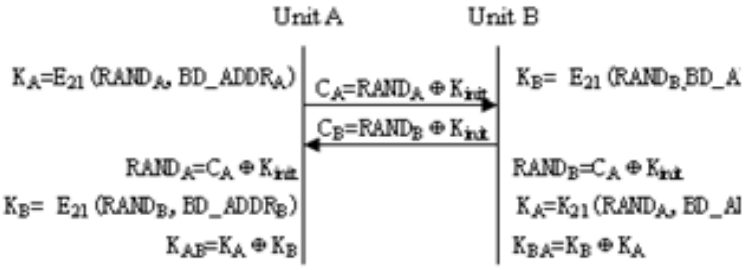


Figure 2. Generation of Combination Key

Step 1 to step 5 is the process of original Diffie-Hellman key agreement protocol. Key (key=KeyA= KeyB) is the private key for communication parties, which prepare to generate the group key for Bluetooth piconet. In step 7, if value obtained from calculating DKeyB (EKeyA (RANDB, RANDA)) is equal to the value of RANDB, unit A has been authenticated by unit B; similarly, unit B is authenticated by unit A in step 9. At the same time, it can avoid replay attack through verifying the value changes of RANDA and RANDB from step 6 to step 9.

3. Bluetooth Piconet Security

a. Bluetooth piconet

There are two types of Bluetooth WSN, piconet and scatternet. Piconet is a basic unit of Bluetooth WSN. But Bluetooth specification is no unified security standard for Bluetooth network. Bluetooth network is confronted with great threats such as eavesdropping and masquerade attack.

Piconet is single-hop ad hoc networks. One of the devices is a master device which determines the timing and the hopping sequence in the piconet, and the others are called slaves. Figure 3 depicts a piconet with three slaves. A piconet is consisting of up to seven slaves and a master.

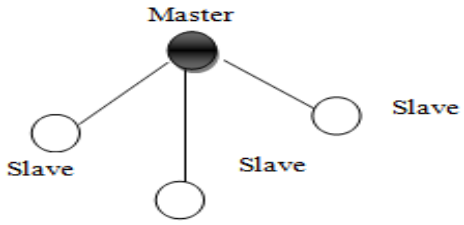


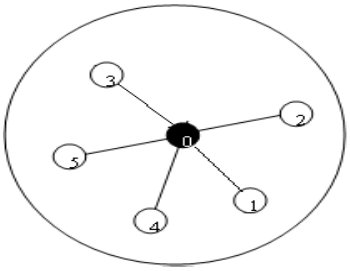
Figure 3. A Bluetooth Piconet

b. The security mechanism for piconet

The main problem of piconet security is how to produce key which can achieve authentication in each other and avoid foreign node jointing into the network illegally. References [2] put forward a method based on group key for piconet. We improve this algorithm to eliminate its weakness which is vulnerable to replay attack. A piconet group key generation divides into three stages.

Step 1: piconet initialization

Once a piconet has been established, each active Bluetooth device obtains a 3 bit value of AM_ADDR [3] to identify members of a piconet. Figure 4 is the result of piconet initialization with six units.



Unit Number	AM_ADDR
0	000
1	001
2	010
3	011
4	100
5	101

Figure 4. The Piconet Initialization with Six Units

Step 2: Generation adjacent key

Because units in a piconet are dynamic changes, the group key agreement must achieve forward security and backward security for piconet. There are two types of adjacent key: backward key and forward key.

Firstly, according to the value of AM_ADDR, the piconet arranges a pattern of logical circle.

Next, each unit performs Diffie-Hellman key exchange between adjacent forward unit and adjacent backward unit, using improved D-H algorithm mentioned above. Where BKi designates a backward key generated between unit i and unit i-1 by Diffie-Hellman key exchange protocol; FKi designates a forward key generated between unit i and unit i+1.

Obviously, forward key of unit i is equal to backward key of unit i+1, for example, FK3 is equal to BK4.

Step 3: Generation group key

The group key generation depends on adjacent key transferring order by unit AM_ADDR value. The algorithm of transmission process is described as follows:

FOR (start unit0; end unitn; i++)
// tow adjacent backward keys using the XOR operator
{
keyi +1← BKi⊕ BKi+1}
// keyi is Temporary key
}
Key← BKn⊕ BK0
// The final group key has been generated in start node
FOR (start unit0; end unitn; i++)
// transmission group key
{
unit i →unit i+1: $E_{FK_i}(Key)$
//encryption by backward forward key of unit i
unit i+1: $D_{BK_{i+1}}(E_{FK_i}(Key))$
//decryption by backward key of unit i+1 and achieved group key
}

Obviously, transfer process of group key is encrypted and each unit offers forward security, backward security.

The claimant security interface can be embedded in function rfcomm_send_data ( ); the receiver security interface can be embedded in function rfcomm_recv_data (), both two functions belonging to RFCOMM layer.

4. Bluetooth Scatternet Security

a. Bluetooth Scatternet

Two or more piconets connect into a scatternet. In the process of forming a scatternet, a Bluetooth unit can simultaneously joint into several piconets. Because a Bluetooth unit can transmit and receive data only in one piconet at a time, its participation in multiple piconets has to be on a time division multiplex basis [4]. Figure 5 depicts a scatternet consisting of three piconets.

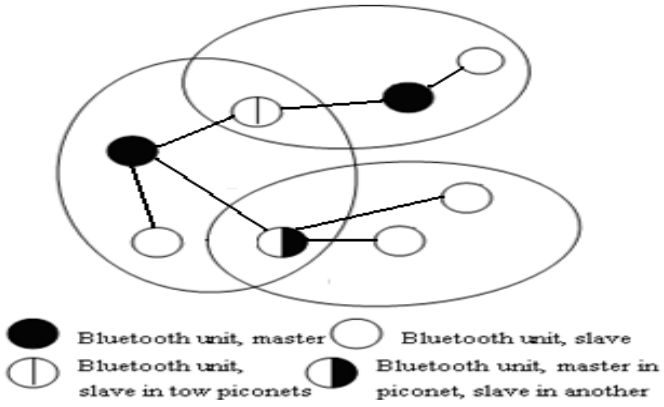


Figure 5. A Bluetooth Scatternet

Some piconets are interconnected via bridge nodes into a larger ad hoc network so-called a scatternet. A bridge node can be a master in one piconet and slave in others (M/S bridge node) or a slave in multiple piconets (S/S bridge node), as shown in Figure 5.

The current Bluetooth specification has already defined scatternets. However, it doesn't focuses on specify details of scatternet operation including security in process of scatternet formation. Protocols of scatternet formation based on Bluetooth ad hoc include BTCP [5], BlueStars [6], and so on.

The BTCP algorithm is based on the process of leader election. The mechanism doesn't indicated masters and slavers in advance. Each node random enters into INQUIRY state or INQUIRY SCAN state. If node hasn't connected, each node enters the opposite state respectively. This process will repeat until all the connections have established. This mechanism based on the symmetrical connections of random state. The algorithm is divided into three key steps [7].

Phase 1: coordinator election

Two connected nodes compare their VOTES (original value is 1) and the node with larger VOTES becomes win node (If the two values are equal, the node with bigger address is winner.). Then two nodes disconnect. Then the defeated node enter PAGE SCAN state and winner' VOTES changes into the sum of the two at the same time. After comparing n-1 times, the winner with the biggest VOTES will be the final coordinator node and all the other nodes will enter PAGE SCAN state waiting for being aroused.

Phase 2: role determination

After the coordinator has calculated p(number of generation piconet), coordinator chooses p-1 nodes as master in addition to itself and p(p-1)/2 nodes becaome bridge nodes. The rest of the nodes become pure slave r[7].

The coordinator call these designated master nodes to build an temporary piconet. As a master, the coordinator send a list with slaverslist and bridgelist.when every master has received this list, the temporary piconet is dismissed.

Phase 3: scatternet establishment

Designated masters page respective slaves according to the slaverslist distributed by the coordinator to built a respective piconet. When a node has been noticed as a bridge node, it waits for the second master pageing. Once paged by the second master, the other connection will be establishe and the bridge nodes send a connected notice to two master. A scatternet has produced when each master has received a connected notice from designated bridge node which is in its piconet.

b. The security mechanism for scatternet

The original BTCP is not involved the security measures. This makes scatternet more vulnerable to attack, such as personating attack. Reference [6] has given out a security scheme, but this method introduces the public key certificate and lead to consume vast systemresources. In this paper, the security algorithm was improved.

In the process of each forming piconet, we can adopt the group key scheme based on the Diffie-Hellman as described above. It guarantees the nodes authentication each other in the same piconet. But the scatternet working in different frequency hopping, the above strategy of group key is no longer viable. We improve scheme as follow.

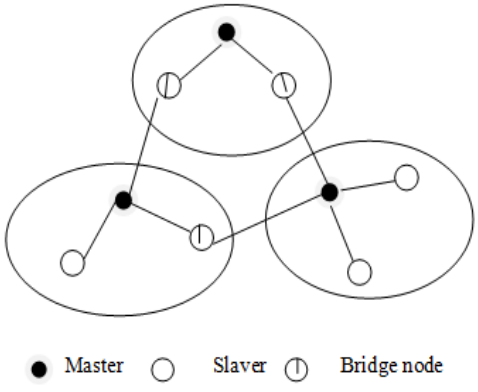


Figure 6. Group Key Strategy of Scatternet

According to BTCP algorithm, masters and bridge nodes can connecte into a ring (at least one node participated in this ring) and coordinator must be in it, depicted in figure Figure 6. The coordinator organizes consultation to produce a work frequency hopping. So these nodes can use group key strategy and realize authentication among different piconets. The system encrypts data with scatternet group key when nodes communicate via different piconets.

5. Conclusion

This paper gives out a scheme of group key based on authenticated Diffie-Hellman in Bluetooth WSN, including piconet or scatternet. It overcomes the inherent safety problems of bluetooth WSN. The claimant security interface can be embedded in function rfcomm_send_data ( ); the receiver security interface can be embedded in function rfcomm_recv_data (). these functions belonging to RFCOMM layer and easy to realize the user authentication And complex operation. What we will do next is how to improves the response speed when the structure of bluetooth WSN has changed, such as nodes joining or exiting the system.

Acknowledgments

This work has been supported by a planning grant from “Funding Project for Academic Human Resources Development in Institutions of Higher Learning Under the Jurisdiction of Beijing Municipality, PHR(IHLB)”.

References

[1] Suzuki Tsuyoshi, Sugizaki Ryuji, Kawabata Kuniaki, Hada Yasushi, Tobe Yoshito. Autonomous deployment and restoration of Sensor Network using mobile robots. *International Journal of Advanced Robotic Systems*. 2010; 7(2): 105-114.
[2] Markus Jacobson, Susanne Wetzel. *Security weaknesses in Bluetooth*. In Proc. RSA Security Conf. - Cryptographer's Track, LNCS. 2002: 76–191.
[3] Yu Xin, Wang ZhaoShun, Chong RongGang. *Application of group key agreement based on authenticated Diffie-Hellman for bluetooth piconet*. International Conference on Information Engineering. 2009: 125-128.
[4] Bluetooth SIG. Specification of the Bluetooth System. 1999; 1(10B).
[5] P Johansson, M Kazantzidis, R Kapoor, M Gerla. Bluetooth: an Enabler for Personal Area Networking. *IEEE Network*. 2001; 15(5): 28-37.
[6] T Salonidis, P Bhagwat, L Tassiulas, R LaMaire. *Distributed topology construction of Bluetooth personal area networks.* Proceedings of the Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. 2001: 1577-1586.
[7] S Basagni, C Pereioli. *Multihop Scatternet Formation for Bluetooth Networks.* IEEE Vehicular Technology Conference. 2002; 1: 424-428.
[8] Yu Xin, Wang yuping. *Secure constructing bluetooth scatternet based on BTCP*. 5th International Conference on Information Assurance and Security. 2009; 2: 200-203.