Security Enhancement of AODVjr Routing Protocol for ZigBee Network

Tao Shang, Jianwei Liu
School of Electronic and Information Engineering, BeiHang University
Beijing, 100191, China
{shangtao, liujianwei}@buaa.edu.cn

***Abstract*—ZigBee network is a kind of flexible wireless network technology and the performance of the network might be severely affected in the presence of malicious attack. Hence, security is critical issue and new techniques of security measures are essential for high-survivability network. Based on that AODVjr routing protocol is suitable for ZigBee network, in this paper, we have enhanced the security of reactive routing protocol, specially Ad Hoc On-Demand Distance Vector junior (AODVjr) protocol, by means of hop-by-hop identification verification. We also discuss the optimization of security scheme on real platform. Through practical experiments, the improved AODVjr routing protocol demonstrates a better routing flexibility and stronger security to malicious attack. We can verify that the AODVjr routing protocol with security enhancement has larger flexible application in ZigBee network.**

Keywords- AODVjr; ZigBee; security; routing

I. INTRODUCTION

ZigBee wireless technology based on the IEEE 802.15.4 standard is a kind of flexible wireless network technology, which offers low power consumption, interoperability, reliability and security for control and monitoring applications with low to moderate data rates. The ZigBee Alliance promotes world-wide adoption of ZigBee as the leading wirelessly networked, sensing and control standard for use in consumer electronics, energy, home, commercial and industrial areas, many of which are security sensitive. If the network is not secured, an attacker could modify and inject messages to cause a network error or industrial harm. Meanwhile, many applications also require confidentiality and most have a need for integrity protection.

The 802.15.4 specification addresses secure needs only through a link-layer security package[1]. In general, the implemented protocol is co-operative; there is no fixed infrastructure or central concentration point for security check; practical application is important and the consideration of hardware resource is necessary. The nature of ZigBee network makes it vulnerable to various forms of attacks such as passive eavesdropping, active interfering, leakage of secret information, data tampering, impersonation and denial of service. ZigBee network needs harder security than conventional wired and static internet. Hence, to create a highly secured ZigBee network, we need to implement a secure network protocol.

The work is funded by the Research Fund for the National High Technology Research and Development Program of China (2009AA012418).

Sastry[2] highlighted a lot of security considerations for IEEE 802.15.4 networks, and Zheng[3] presented a systematic analysis of the threats faced by low rate wireless personal area networks with respect to the protocol stack defined by IEEE 802.15.4 and the ZigBee Alliance. As a main research content, routing protocol is important in the context of ZigBee networks and has been proved to be very useful[4-6], however, few secure routing schemes for ZigBee network are proposed. A malicious node could make use of the flaws and inconsistencies in the routing protocol to create faked routing message and advertise nonexistent links, provide the incorrect link state information and flood other nodes with routing traffic. Hence the security enhancement of routing protocol need be further investigated.

In this paper, to enhance the security of AODVjr routing protocol in a ZigBee network, we address to propose an AODVjr routing protocol with security scheme by means of encrypting the immutable information of routing packets for hop-by-hop identification verification. Such processing can enhance the ZigBee network to defend some usual attacks.

II. BACKGROUND

Considering ZigBee network construction, AODVjr + Cluster Tree routing protocol was used to solve this problem[6]. AODVjr is based on the Ad hoc On-demand Distance Vector (AODV) routing protocol and is a trimmed down AODV specification which removes all but the essential elements of AODV and has nearly the same performance as AODV[7]. AODVjr removes the following items from the AODV specification: sequence numbers; gratuitous Route Reply (RREP); hop count; hello messages; Route Error (RERR); precursor lists. The AODVjr routing protocol is a reactive routing protocol, and routes are determined only when needed.

From the description of routing discovery of AODVjr, we can know that although source originates route on-demand, the destination, not source, finally determines the route due to the unique resulting route. The source is also not involved in route discovery like AODV. In AODV, RREQ (Route Request) and RREP play a role of route discovery. When multiple RREPs are received by the source, the chosen route changes with the updating of succeeding RREPs. This characteristic can improve the routing performance in term of network, therefore it is necessary to combine multiple feedback information to AODVjr, which can change the routing decision commander

from destination to source and make a proactive routing decision on the basis of multiple feedback information. Thus a improved AODVjr with multiple feedback policy was proposed[8]. Based on all RREQ messages from destination node, which contains all routes, source node processes multiple RREP messages for routing selection. Such improvement of AODVjr changes the decision commander from destination to source and make a proactive routing decision on the basis of multiple feedback information. Figure 1 shows the message exchanges of new AODVjr protocol. From Figure 1, we can know that two optional paths from S to D is S>1>4>D and S>1>2>3>D. After data flows from the source to the destination, the acknowledge message is sent according to D>3>2>1>S. Consequently, the chosen route can vary in need of network status by judging from multiple feedback information.

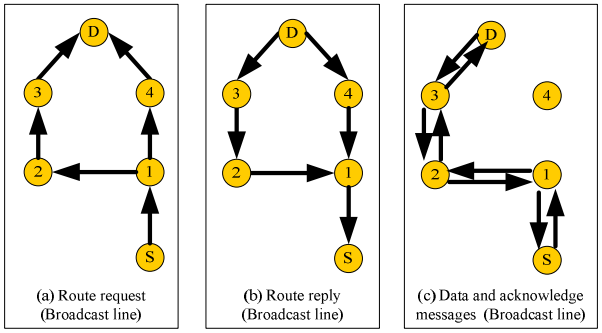

Figure 1 Routing discovery of AODVjr with multiple feedback policy

The results show that AODVjr with multiple feedback policy provides more flexible routing policy and could easily be extended to incorporate optimizations that are standard in AODV. However, one immeasurable quality of AODVjr is its simplicity. Another significant feature of AODVjr is that control packets contain immutable fields, which allows security to be added easily. The part of routing information needs to be defended by some secure measure.

III. PROPOSED APPROACH

A. *Assumption*

1) All legal nodes join network during the initial period of network construction. Afterwards, the network does not permit a joining request of new node any more, but may revoke request of old node;
2) There does not exist any malicious node during the initial period of network construction;
3) All nodes do not join the network at the same time, but in succession. Afterwards, we shall discuss the least time interval for practical platform.

B. *Security enhancement scheme*

We shall extend the improved AODVjr routing protocol with security scheme. As main ideas of security scheme, the scheme encrypts the immutable information of routing packets by means of symmetric encryption method, and implements hop-by-hop identification verification by means of multiple keys. Of course, the key distribution is very important and will be achieved during the initial period of network construction.

Source node *S* will send a RREQ message to destination *D*. The immutable information of RREQ message is denoted by *M*, *k1* is the distributed key, and I_1 is the first intermediate node. Source node *S* shares the key *k1* with I_1, while I_1 share another *k2* with next-hop node. When a node sends or receives a routing message, it will use the corresponding key to encrypt or decrypt the immutable information of routing message. Since the receiver and sender share the same symmetric key and the immutable information of routing message (node address) is used for cipher, the end-to-end identification verification of node is realized by means of hop-by-hop identification verification, the security of routing process can be guaranteed, just as described in the following part.

$$\begin{aligned} &S \to I_1 : E_{k1}\left[M \parallel H(Counts)\right] \\ &I_1 \to I_2 : E_{k2}\left[M \parallel H(Counts')\right] \\ &\cdots \\ &I_{n-2} \to D : E_{kn-1}\left[M \parallel H(Counts^{(n-2)})\right] \\ &I_{n-1} \to D : E_{kn}\left[M \parallel H(Counts^{(n-1)})\right] \end{aligned} \quad (1)$$

According to the hop-by-hop identification verification, each node owns one key for symmetric encryption and another key for symmetric decryption, and the keys between each pair of nodes are different.

In order to distribute key between each pair of nodes according to Cluster-Tree relationship, it is convenient to generate key between the network and new node when a node prepares to join network. Considering that hand-shake protocol is necessary for node joining ZigBee network, we adopt Diffie–Hellman (D–H) algorithm for key exchange with more security. D–H algorithm allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure communications channel. Thus we can combine the parameter exchanging of D–H algorithm into the hand-shake protocol for node joining ZigBee network.

D–H algorithm exchanges four parameters by two hand-shakes, and joining ZigBee network also requires two hand-shakes by means of beacon request frame, beacon frame, associate request frame, and associate response frame. Therefore it is feasible to exchange key parameters by means of two hand-shake protocol. By improving the payload field of related frames, the four parameters of *n, g, A, B* for key exchange are attached into beacon request frame, beacon frame, associate request frame, associate response frame, respectively.

Concretely, when node M tries to join ZigBee network by node N, (1) it firstly broadcasts a beacon request frame carrying with parameter *n*. (2) node N receiving the beacon request frame generates a large prime number *g* and broadcasts a beacon frame carrying with parameter *g*. (3) Node M receives the beacon frame carrying with parameter *g* and identifies the node address sending the beacon frame. According to the received parameter *g*, the parameter *A* can be calculated. Node M sends a associate request beacon carrying with the parameter *A* to node N. (4) Node N receives the associate request beacon, calculates the parameter *B* and symmetric key, and then broadcasts a associate response frame carrying with parameter

B. (5) Node M receives the associate response beacon, calculates the symmetric key, and successfully joins the network. The hand-shake protocol is shown in figure 2.

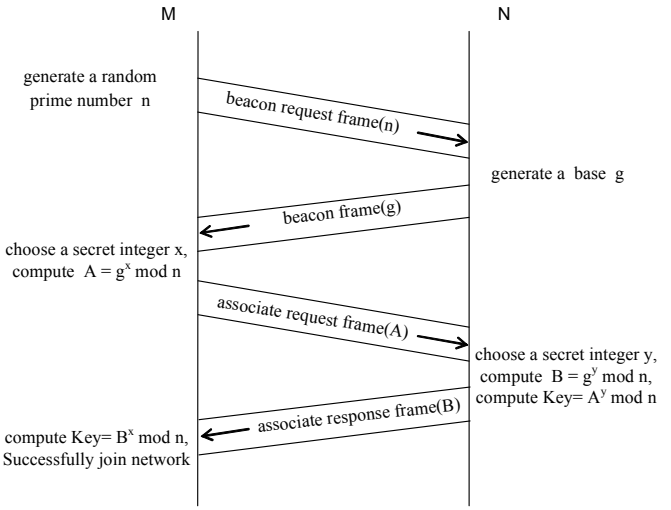

Figure 2 Hand-shake protocol of joining network with key exchange

According to the above protocol, we need to further make the related details clear:

1) Since *g* is a random number, different joining nodes probably use different value of *g*.
2) Since node M initially broadcasts a beacon request frame for network joining, and will probably receive multiple beacon frames returned from neighbor nodes. According to the hand-shake protocol, node M finally will probably receive multiple associated response frames. Considering that one node belongs to an unique parent node in ZigBee network, node M will build network associate with the node whose associate response frame firstly arrives. For other received associate response frames, node M only calculates symmetric key for routing.
3) During the process of joining network, node will construct key pairs with all neighbor nodes in the communication scope. Each node stores a key table for every pair of nodes.

The improved hand-shake protocol of joining network makes full use of the characteristics of ZigBee network, implements safe key distribution during the period of network construction, and guarantees that each joining node owns symmetric keys with multiple neighbor nodes. Meanwhile, it is also obvious that key distribution will fail if two nodes simultaneously apply to join network. After two nodes broadcast beacon request frames simultaneously, the network node broadcasts the returned beacon frame. Due to the broadcast of beacon frame, the node applying to join the network can not judge whether the received beacon frame is sent to itself or not, furthermore, not judge whether the carried parameter in beacon frame is sent to itself or not. Consequently the key distribution is mistaken. Thus in order to avoid such problem, node should be guaranteed to join network at intervals, so that key exchange can be achieved successfully.

C. Security analysis

According to the assumption 2 of section III, external attack node can not legally join the network and acquire the symmetric key for hop-by-hop identification verification. Based on such conditions, we shall discuss the security of routing protocol.

The protocol encrypts/decrypts the source and destination address of RREQ and RREP with symmetric key so that the identification verification between nodes can be implemented hop by hop. If an attacker fakes a new RREQ message or tampers with the source and destination address of received RREQ message for forwarding, the faked RREQ message can not be certificated and subsequently discarded, because the attacker has no correct symmetric key for identification verification. If the attacker wants to tamper with the mutable information (the number of hops), AODVjr has ignored this part so that the node sending RREQ will make a route from multiple returned RREPs. According to ZigBee routing protocol, the number of hops is not the metrics. so this case also can not affect the network. Since external nodes can not acquire symmetric key for identification verification, the malicious node also can not act as other node identification. Although the attacker does not acquire the related symmetric key for identification verification and the extra RREP does not response, DoS attack may still affect the routing protocol. Thus the security of protocol still need be enhanced further.

IV. PRACTICAL OPTIMAZTION

Considering the application of ZigBee network, the proposed security scheme is oriented to practical hardware platform, so we should combine the characteristic of resource constrained condition into our security scheme.

A. Selection of encryption algorithm

We shall use symmetric encryption method to encrypt the immutable part of routing message. AES, RC5 and RC6 are three kinds of practical symmetrical encryption algorithm with relatively high security.

In cryptography, RC6 is a symmetric key block cipher derived from RC5. It was designed to meet the requirements of the AES competition. RC6 is very similar to RC5 in structure, using data-dependent rotations, modular addition and XOR operations; in fact, RC6 could be viewed as interweaving two parallel RC5 encryption processes. However, RC6 does use an extra multiplication operation not present in RC5 in order to make the rotation dependent on every bit in a word, and not just the least significant few bits.

For memory usage, The subkey group of RC5 and RC6 will mainly occupy the memory of hardware, and the size of subkey group depends on the number of encryption rounds. In the case of 32 rounds, RC5 and RC6 occupy the memory of 200 Bytes. The transforming constant array of AES mainly occupies the memory of 1500 Bytes. Besides the basic network protocol stack, no more large memory is left for a usual sensor node with only several thousands bytes SRAM. So AES is not suitable for ZigBee network of limited resource.

Here we chose RC6 as symmetrical encryption method for routing message. RC6 has a block size of 128 bits and supports key sizes of 128, 192 and 256 bits, but it can be parameterized to support a wide variety of word-lengths, key sizes and number of rounds.

B. *Improvement of Diffie-Hellman algorithm*

In order to apply Diffie-Hellman algorithm to ZigBee wireless network for key distribution, the algorithm needs to be improved by means of optimizing computing operation.

Random number generator is kernel component of key generation. Pseudo-random function usually generates a series of predictive number. If the seed of series is same, same random number is generated. According to this characteristics, we frequently change a seed by means of sampling the ADC volt. The lowest bit of volt is respectively sampled five times for units, tens, hundreds, thousands, ten thousands of target seed, then use the seed to generate a random number [0, 0xffffffff].

Diffie-Hellman algorithm needs large prime number. For a large integer n, normal algorithm checks whether n is a prime number by means of judging whether n is divided by all number from 2 to $sqrt(n)$, and its order of complexity is $O(sqrt(n))$. If n is larger, computing efforts is larger. Here we use Miller Rabin algorithm based on Fermat theory. Considering the algorithm can not completely guarantee to generate a prime number, we can enhance performing times to obtain a large prime number n.

For larger x and y, the computing unit of $x^y\%n$ is usual and its calculation is very difficult, specially for sensor node with limited resource. According to binary format, this calculation is transformed by (2).

$$x^y \bmod n = [\prod_{i=0}^{i=k-1} (x^{(y_i * 2^i)} \bmod n)] \bmod n \qquad (2)$$

Where $y_{k-1} y_{k-2} \cdots y_0$ is binary format of y.

V. EXPERIMENTAL RESULTS

A. *Experimental platform*

The ZigBee network was implemented on the GainZ wireless sensor node. Chipcon CC2420 RF transceiver is integrated in the GainZ node and supports 2.4GHz band. SRAM memory is 4KB. The MAC provides network association and disassociation, has an optional superframe structure with beacons for time synchronization, and a guaranteed time slot (GTS) mechanism for high priority communications. The channel access method is carrier sense multiple access with collision avoidance (CSMA-CA). The maximum communication speed is up to 250kbps.

Coordinator, Router and End Device were all implemented on GainZ nodes. Router and Coordinator used improved AODVjr route protocol to implement routing, while End Device just continually obtained the data of light intensity and send these data to destination.

B. *Encryption parameter setup*

Due to the limited resource of platform, although the operation of $x^y\%n$ has been optimized, the node can only support long integer (32 bit), so Diffie-Hellman algorithm only generates the key sizes r of 28 bits length, which can not satisfy the normal requirement of 128bits in RC6 algorithm. As a compensation of security, the number of rounds r for RC6 algorithm is 32. Because of word-lengths of GainZ nodes, the word-lengths w for RC6 algorithm is 16.

Based on the practical optimization, security scheme can be realized on sensor node. The average time that RC6 encryption and decryption are collaboratively finished at a time on a node is about 30ms. The average time that D-H algorithm is collaboratively finished at a time on a node is about 600ms.

C. *Experimental procedure*

To examine the performance of AODVjr routing protocol with security scheme, a simple cluster-tree topology of a hybrid star/mesh was utilized, just as shown in Figure 3. The scenario of 5m X 5m field with 5 nodes was used. From the viewpoint of type, there are one Coordinator, three Router, and one End Device. For convenient debugging, Coordinator is connected with PC computer as monitor. Since the communication scope is larger than experimental field, black list is used to filter the unnecessary neighbor node in MAC layer.

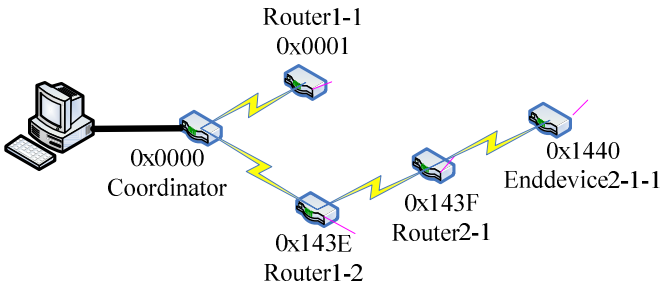

Figure 3 Cluster-tree topology

Firstly, the Coordinator initialized ZigBee network, configured the network parameters such as the maximum depth (Lm=5), maximum number of routers (Rm=6), maximum number of childs (Cm=20), working channel(13, 2.4GHz). Then the Coordinator added Router and End device into network, and distributed the network address to these devices. Once the End Device successfully joined the network, it continued to send data packets to the Coordinator per 3s. The resulting network is shown as Figure 3, and related key parameters are listed in Table 1.

Table 1 Key parameter

	Coordinator	Router1-1	Router1-2	Router2-1
Coordinator	none	0435fed5	085f443e	none
Router1-1	0435fed5	none	none	00a8e75d
Router1-2	085f443e	none	none	03683eb1
Router2-1	none	00a8e75d	03683eb1	none

Table 1 shows the first level router, Router1-1and Router1-2, joined network and acquired the symmetric keys; the second level router Router2-1 joined the network by Router1-2 and belonged to Router1-2, but it owned two keys with Router1-1and Router1-2; the third level Enddevice2-1-1 joined network by Router2-1 and did not own key with its parent.

When Enddevice2-1-1 successfully joined the network, it continued to send data packets to the Coordinator. During the process of route discovery, Firtsly, Enddevice2-1-1 sent the packet to Router2-1. Then Router2-1 searched routing table. If destination is listed in routing table, the packet can be directly sent to Coordinator, otherwise, it sent encrypted RREQ to neighbor nodes(Router1-1and Router1-2). When Router1-1and Router1-2 received RREQ, they decrypted the RREQ and continued to forward the packet after encrypting by owned key. Finally the Coordinator received the RREQ from Router1-1and Router1-2, then sent encrypted RREP to Router2-1. After two encrypted RREP arrived at Router2-1, routing decision was finished and data from Enddevice2-1-1 was sent to Coordinator. Such information is completely identical to key distribution.

After security scheme is integrated into routing algorithm, network construction period will be delayed and routing time will be enlarged. Diffie-Hellman algorithm enlarged the time of joining network. Figure 4 shows the variation of the time of joining network. It takes average 750ms, larger than normal joining time and offline D-H calculation time, but this delay can be accepted since it happens in network construction initialization, and used as the least interval for node joining.

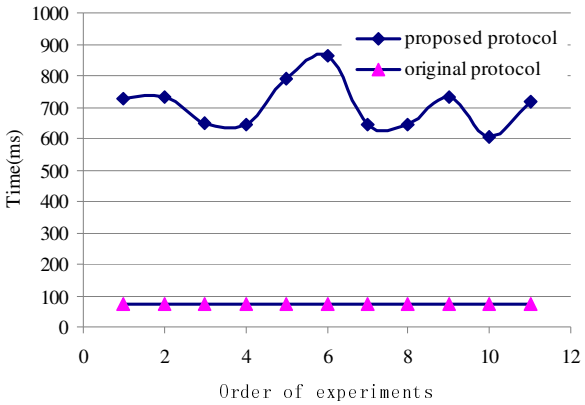

Figure 4 Time of joining network

On the other hand, RC6 encryption method also enlarged the delays incurred by all the packets that are successfully transmitted. When encryption is used, just as shown in Figure 5 average end-to-end delay is 180ms at the case of 1 level, larger than 25ms without encryption. As routing depth increases, the delay also increases a lot and can be described as (3):

$$\Delta t \approx (180-25) \times d \qquad (3)$$

Above all, we can know that AODVjr routing protocol with security scheme performs well, have nearly identical packet delivery performance with original AODVjr. Though improved AODVjr adds partial network delay, it outperforms original AODVjr when the network is confronted with malicious attack.

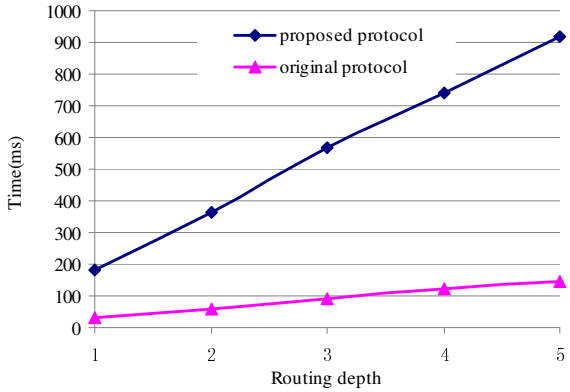

Figure 5 Average end-to-end delay

VI. CONCLUSION AND FUTURE WORK

In this paper, we have proposed an AODVjr routing protocol with security scheme and verified its performance on the practical platform. The solution proposed is for on-demand routing protocols, specially AODVjr. It has been shown that improved AODVjr has nearly the same performance as AODVjr. Such improvement can prevent false routing information and Sybil attack. Despite the performance of network delay maybe increase, the routing method necessarily result in a better routing security. The results of our implementation show that the overhead induced by security scheme has negligible effects on network performance while making the protocol secure. In fact, we plan to introduce security scheme for external attacks and incorporate those into ZigBee application.

REFERENCES

[1] ZigBee Alliance. ZigBee Specification (Version 1.1)[S]. 2006.

[2] N Sastry, D Wagner, Security considerations for IEEE 802.15.4 networks, Proceedings of the 3rd ACM workshop on Wireless security, Philadelphia, PA, USA, 2004:32 – 42.

[3] J Zheng, MJ Lee, M Anshel, Toward Secure Low Rate Wireless Personal Area Networks, IEEE Transactions on mobile computing, Vol. 5, No.10, 2006.

[4] Peng Ran, Mao-heng Sun, You-min Zou: ZigBee Routing Selection Strategy Based on Data Services and Energy-Balanced ZigBee Routing[C], Proceedings of the 2006 IEEE Asia-Pacific Conference on Services Computing(APSCC'06), Guang zhou, China, 2006:400-404.

[5] Francesca Cuomo, Sara Della Luna, Ugo Monaco, Tommaso Melodia, Routing in ZigBee: benefits from exploiting the IEEE 802.15.4 association tree, Proceedings of IEEE International Conference on Communications 2007, Glasgow, Scotland , 2007:3271-3276.

[6] Qiu F., Wang J-M., Leng J., Design and Implementation of a Wireless Personal Area Network Based on AODVjr Routing, Proceedings of Wireless Mobile&Multimedia Networks[C], Beijing: The Institution of Engineering and Technology, London(IET), 2006:424-427.

[7] Ian D. Chakeres, Luke Klein-Verndt. AODVjr, AODV Simplified[J]. Mobile Computing and Communication Review, 2002, 6(3):100-101.

[8] Tao Shang, Wei Wu, Xudong Liu, and Jianwei Liu. AODVjr Routing Protocol with Multiple Feedback Policy for ZigBee Network, The 13th IEEE International Symposium on Consumer Electronics, Kyoto, Japan, 2009:483-487.