An Efficient Anonymous Routing Protocol for Mobile Ad Hoc Networks

Xiaoqing Li, Hui Li, Jianfeng Ma, Weidong Zhang
Key Lab of Computer Networks and Information Security of Ministry of Education
Xidian University, Xi’an, P.R.China
{xqli, lihui, jfma, wdzhang}@mail.xidian.edu.cn

Abstract—Providing anonymous routing in mobile ad hoc networks (MANET) has been a hot issue for the purpose of security and privacy concerns. But there are very few have been done about providing a valid method to detect malicious node and providing a trustworthy protection over whole network. In this paper, we introduce a localized trust management which can primarily remove malicious nodes, and propose an efficient anonymous routing protocol by node that participates in the protocol encrypts entire message with trust key and says Hello to its ancestor within expiration time. It makes malicious node can be detected and isolated from the network. In this way, an anonymous and secure route path can be established in a hostile environment. Meanwhile, it is able to efficiently against the Denial-of-Service (DoS) attack.

Keywords—Anonymous; Routing protocol; DoS attacks; ad hoc network;

I. INTRODUCTION

A mobile ad hoc network (MANET) is a collection of nodes that are connected through a wireless medium and is lack of any kinds of fixed infrastructure that provide network management operations like the traditional fixed network. In this network each node both acts as a router and a host at the same time. In addition, every node moves irregularly and network topology changes frequently[1].

Therefore, routing in MANET faces additional challenges when compared to routing in traditional wired networks. The main assumption of the previously presented ad hoc routing protocols is that disregarding the existence of malicious entities in the network. There are several attacks that can target the operation of a routing protocol in MANET, such as location disclosure, eavesdropping, replay attack, and Denial of Service (DoS) attack. DoS attack is a serious attack for entire network. It aims at the complete disruption of the routing function and therefore the whole operation of MANET [2], [3]. So it is very necessary to produce a secure and reliable routing protocol for either router discover or data transfer. And anonymous routing based on the onion routing is an efficient method for security and privacy in MANET. SDAR [4] is a secure distributed anonymous routing protocol, but not against DoS attack. TEAR (Trust Enhanced Anonymous Routing) [5] is a trust-aware anonymous routing protocol, but cannot provide the secure and lightweight features as claimed. Other works for defending DoS attack and onion routing have been appeared in [6], [7], [8].

In this paper, we propose an efficient secure and anonymous routing protocol, we refer it as EARP. The typical features of this protocol are that each node encrypts entire message with trust key and notes participate in the protocol can detect malicious node and isolate them from the network, by saying Hello to its ancestor within expiration time. So it makes every node can choose a reliability intermediate node to establish a path in a hostile environment, meanwhile, it is able to defend against many attacks efficiently, such as DoS attack in MANET. Additionally, the source does not need to collect and maintain the topological information of the network. In this paper the anonymity properties [9], [5] include three parts: identity anonymous, path anonymous and location anonymous.

The rest of the paper is organized as fellows: In section 2 we introduce a method to identity trusty nodes, which is localized trust management. In section 3 we present our protocol. The feature of security and defending attacks of our protocol is analyzed and discussed in section 4. Finally, section 5 concludes the paper.

II. LOCALIZED TRUST MANAGEMENT

Since MANET is in open environments, it is necessary to produce a mechanism to identify malicious nodes and isolate them from network. In this paper, we introduce a localized trust management [4], [10], [11]. We choose some central nodes in the network and nodes are identified as its neighbors that only one hop away from the central nodes. The central node manages the topologies information of its neighbor nodes and distributes keys to them. The central node keeps track of its neighbors simply by listening for a Hello message which is broadcasted periodically by its neighbors, and confirms which node can be identified as trust by the node’s behavior. Then central node distributes the trust key to them and others considered as malicious node cannot get it. The key is distributed periodically since nodes can move freely in MANET.

The localized trust management identifies malicious nodes primarily and avoids using them during the route establishment. Meanwhile, it can help every node chooses a reliable node to establish a path or transfer data through checking if the node has trust key and the security and reliability of route is increased.
### III. AN EFFICIENT ANONYMOUS ROUTING PROTOCOL (EARP)

The proposed secure anonymous routing protocol consists of three parts: the route discovery phase, the route reverse phase and the data transfer phase. The main notations used in the paper are presented below.

- $ID_S, ID_D, ID_i$: The identity of source node, destination node and the $i$-th intermediate node.
- $r_i$: Random number of the $i$-th node.
- $K$: Trust key, it is a symmetric key.
- $K_{rd}$: A symmetric key, encrypting with $r_i$.
- $(pk, sk)$: A one-time public-private key pair of source node $S$.
- $e$: Message expiration time.
- $N$: Hop count.

#### A. Route discovery phase

When source node $S$ needs to communicate with intended destination node $D$, it must establish a secure anonymous path between them in the network and then data traffic can be safely transferred on the path.

1) The $RREQ$ in source node: Source node $S$ generates a key pair ($pk$, $sk$) and random number $r_S$ before it communicates with destination node $D$. We presume that the localized trust management has differentiated trust nodes and distributed trust key $K$ to them. Then node $S$ broadcasts a route request ($RREQ$) message to all of its neighbors, and fills "null" in ancestor node field of its route table. The message has the format as follow:

$$RREQ = \{Req, seq, pk, E_{K_{SD}}(ID_D, sk), E_{K_{rd}}(ID_S), E_{pk}(ID_S, r_S), \text{Hello}, e, N\}_K$$

There are two features in our protocol:

- The entire $RREQ$ message is encrypted with trust key before node forwards the $RREQ$.

It makes malicious node cannot participate in the route discovery. Whereas the localized trust management has isolated malicious nodes from the network, but the situation must be considered that a node that conformed as trust before is captured by adversary and its neighbors or central node do not know about that. It is serious whenever on the processes of route establishment or data transfer. However, since trust key is distributed periodically and malicious nodes do not know the trust key $K'$ of this session, it cannot decrypt received message and participate in the protocol.

- Hello message and $e$.

When a trust node receives the $RREQ$, it broadcasts Hello message to its ancestor within the message expiration time $e$. The purpose of that is to confirm it is a trust node to its ancestor. This phase combines with encrypting the entire $RREQ$ message can availably exclude malicious nodes and defend attacks.

$E_{K_{rd}}(ID_S)$ is used to fill in the route table for the next node in the route discovery phase.

2) The $RREQ$ in intermediate node: As show in the fig.1, when the node $M$ receives the $RREQ$, it goes through the following procedure:

- Check if $M$ has the trust key to decrypt the message, if so, it is a trust node as $T$, it goes to 3. Otherwise, it goes to 2.
- Since without the trust key, node $M$ would be a malicious node and cannot broadcast Hello message to its ancestor node $N/C$ within the time $e$. After the waiting time, $N/C$ marks $M$ as malicious node and informs to its central node, then $M$ is isolated from the network. And then $N/C$ forwards the $RREQ$ to node $T$.
- Check if $T$ is the intended receiver by trying to decrypt $E_{K_{SD}}(ID_D, sk)$. If not, it is an intermediation node and broadcasts Hello message to $N/C$. $T$ appends its identity and a random number $r_T$ produced by itself to the message by the onion algorithm and encrypts them with $pk$. Then $T$ fills $E_{K_{rd}}(ID_T)$ in ancestor node field of its route table. Adding its hidden identity $E_{K_{rd}}(ID_T)$ to the message as show in fig.1 and adding one to $N$. Encrypting the entire message with trust key finally. $T$ forwards the message and the format is:

$$RREQ = \{Req, seq, pk, E_{K_{SD}}(ID_D, sk), E_{K_{rd}}(ID_T), E_{pk}(...E_{pk}(ID_S, r_S)...), ID_T, r_T, \text{Hello}, e, N\}_K$$

Otherwise, it is the intended destination node $D$.

3) The $RREQ$ in destination node: Decrypting $E_{K_{SD}}(ID_D, sk)$, node $D$ finds it is the intended destination node and forwards Hello message to its ancestor. Filling $E_{K_{rd}}(ID_{N-1})$ in ancestor node field of its route table. Using the private key $sk$ to decrypt

$$E_{pk}(...E_{pk}(E_{pk}(ID_S, r_S))...), ID_T, r_T, \text{Hello}, e, N)$$

And all of the identity and random number of intermediation nodes can be obtained. Additionally, $D$ affirms that the $RREQ$ comes from the right source node $S$, and $D$ is the $N$-th node. The path discovery ends.

#### B. Route reverse phase

$D$ could receive several $RREQ$ since all the messages are broadcasted, it chooses the shortest path by $N$ and the path
1 in the fig.1 is the best. Then forwarding the route replay (RREP) phase to S along the selected path.

1) The RREP in source node: The format of RREP in D is:

\[
RREQ = \{REP, seq, E_{KSD}(ID_S), E'_{D}(ID_D), E_{r_{N-1}}, \ldots (E_{r_1}(\ldots (E_{r_2}(ID_D, E_{KSD}(ID_R))))), Hello, e \}_K
\]

Where, \( ID = \{ID_S, ID_1, \ldots , ID_i, \ldots , ID_{N-1}, ID_D \} \), and \( R = \{r_S, r_1, \ldots , r_i, \ldots , r_{N-1}, r_D \} \) are the identity and random number of all the nodes on the path.

\( E_{KSD}(ID, R) \) and the identity of D are layer encrypted and the key is the random number of each node. Since it is on the reverse path so the ancestor node is the successor node on route discovery path, D fills “null” in successor field of its route table and its identity in reverse path is \( E'_{r_D}(ID_D) \). Then node D broadcasts the RREP to all of its neighbors.

2) The RREP in intermediate node: When the i-th node receives the RREP, it runs the following procedure:

- Check if it is trust node with trust key. The operation is similar to the RREQ in intermediate node.
- If it is a trust node, broadcasts Hello message to its ancestor and checks if it can decrypt \( E_{KSD}(ID_S) \) with right key which is found in its key table. If not, it is not the intended receiver. It fills \( E'_{r_{i+1}}(ID_{i+1}) \) in successor field of its route table, and its identity \( E'_{r_i}(ID_i) \) is filled in the RREP message. Peeling the onion layer, encrypting the entire message with trust key and broadcasting the RREP forwards to all of its neighbors. The format of the RREP is:

\[
RREQ = \{REP, seq, E_{KSD}(ID_S), E'_{r_i}(ID_i), E_{r_{i+1}}(\ldots (E_{r_2}(ID_D, E_{KSD}(ID_R))))), Hello, e \}_K
\]

Otherwise, it is the destination node.

3) The RREP in destination node: The node S retrieves the identity and random number of all the nodes with the \( K_{SD} \) and \( K_{rs} \), respectively, and fills \( E'_{r_i}(ID_i) \) in successor field of its route table. In this way, the route from the source to the destination, the set of random number for the session and the route table of each node can be established. Furthermore, another trust key \( K' \) is distributed to the selected nodes for the next session.

C. Data transfer phase

Our data transfer protocol uses a similar approach to the router discover or reverse protocol. According to the route table, nodes transfer message M in the established path. The format is about to forward in the node S is:

\[
M = \{F_M, seq, E_{r_{i+1}}(\ldots (E_{r_1}(\ldots (E_{r_2}(M))))), Hello, E \}_K
\]

The format is about to forward in the i-th node is:

\[
M = \{F_M, seq, E_{r_{i+1}}(\ldots (E_{r_D}(M))), Hello, E \}_K
\]

And the format is received in the node D is:

\[
M = \{F_M, seq, E_{r_D}(M), Hello, E \}_K
\]

Since nodes can move freely, the topology of the network would be changed. When the i-th node detects the change of the network, it forwards route updating message to its neighbors. The format in the i-th node is:

\[
NEW = \{NEW, seq, E_{r_{i}}(newpath), Hello, e \}_K
\]

Following the route table, the message is forwarded in reverse path until S receives it. Then S broadcasts a new route discovery message.

IV. Security and attacks analysis

A. EARP is secure and anonymous

Our protocol bases on the localized trust management which can primarily remove malicious nodes. And every node can identity malicious node by encrypting the entire message with trust key and saying Hello to its ancestor, those operations can ensure that without malicious node can participate in the path and the nodes on the path are all trusty. Hence, the route can be established in security and the network environment can be considered as friendly.

In section 1, we have mentioned the anonymity properties include three parts: identity anonymous, path anonymous and location anonymous. In our protocol no matter it is source node, intermediate node or destination node the identity of them are layer encrypted with itself public-private key or random number key. Hence the node participates in the protocol and communicates with others nodes but cannot know the identity and location of them except the source and destination. The adversary observes the packets cannot infer the relationship between the nodes and deduce weather a node on the path, so the anonymity properties are provided by the protocol.

B. EARP can defend DoS attack

In our protocol, trust key is distributed periodically and intermediate node only say one Hello message to its ancestor node within the expiration time. If a node does not encrypt messages that it forwards with the right trust key and frequently forwards a mass of arbitrary messages within the time e, it can be detected by its neighbors immediately through its malicious behavior. Hence the node is confirmed as malicious node and isolated from the network by its central node. It has no chance to disrupt the routing function, and against servers who are required to interact with many clients. It availably protects the operation of MANET and defends the DoS attack.

C. EARP is against eavesdropping

Malicious nodes mainly eavesdrop the data packets in the network without doing any active operation. In our scheme, no matter the sender, forwarder and the receiver are all hidden by encrypting their identity with random number key or public-private key. So an eavesdropper can not get any identity information from the data packet.
D. EARP is low overhead

A better tradeoff between security and performance is an important issue in MANET. In our protocol, the main computation is forwarding the Hello message and onion routing algorithm. The Hello message is only forwarded once in single process. And although it is non-symmetric encrypting in the routing discovery phase, other operations are symmetric key cryptographic algorithm. And the mainly encrypting operation is put on the node $S$ and $D$ rather than all of nodes on the path. The time complexity of computation is acceptable for MANET. Additionally, the source does not need to collect and maintain the topological information of the network. So EARP is good in low overhead.

V. CONCLUSIONS

In this paper, we present an efficient secure anonymous routing protocol for MANET. Our protocol employs onion routing algorithm to achieve the anonymity and malicious node can be detected and isolated from the network by every node encrypts the entire message with trust key and says Hello to its ancestor. Hence, an anonymous and secure route path can be established. Additionally, it has better characteristic for defending against many attacks.

ACKNOWLEDGMENT

The work was supported by National Natural Science Foundation of China Under Grant No. 60772136, No. 60633020 and No. 60803151. 863 Hi-Tech Research and Development Program of China Under Grant No. 2007AA01Z435 and No. 2007AA01Z429. And the Key Program of NSFC-Guangdong Union Foundation of China Under Grant No.U0835004.

REFERENCES


