Autonomous Position Verification and Authentication for on Demand Routing Protocol for Ad Hoc Network

Sudhakar Sengan[1], Dr.S.Chenthur Pandian[2]
[1] Lecturer, Department of CSE, Nandha College of Technology , Erode -TamilNadu – India
[2] Principal, Selvam College of Technology, Namakkal -TamilNadu – India
sudhasengan@gmail.com, chenthur@rediffmail.com

Abstract:

An ad hoc network is a group of wireless mobile computers, in which individual nodes cooperate by forwarding packets for each other to allow nodes to communicate beyond direct wireless transmission range. Attacks on ad hoc network routing protocol affects network performance and reliability. Traditional routing protocols have no security mechanism and can be attacked by malicious nodes. In this paper, we present secure on demand position based routing protocol for ad hoc network based on basic operation of AODV protocol. The protocol makes use of protected position information to make routing decisions, resulting in improved efficiency and performance. In AODV protocol route selection is a function of hop count and destination sequence number. In our proposed model, the route selection is a function of following parameters: hop count, trust level of node and security level of application. In this paper, we focus on secure neighbor detection, trust factor evaluation, operational mode, route discovery and route selection. The paper mainly addresses the security of geographic routing.

Keywords—Ad hoc Network, Geographic Routing, Trust Factor Evaluation, Secure Neighbor Detection, Security, AODV, Hop Count.

1. Introduction

Most traditional mobile ad hoc network routing protocols were designed focusing on the efficiency and performance of the network. Ad hoc network are wireless network with no fixed infrastructure in which nodes depend on each other to keep the networked connected. Topology based routing protocols use the information about links for packet forwarding. Position based routing protocols use node's geographical position to make routing decisions, resulting in improved performance under extremely dynamic network condition.

Attacks on AODV protocol

In AODV protocol the main design issue is to achieve efficiency in ad hoc network environment while disregarding security issues. Known attacks on AODV are traffic redirection by modification, replay attacks, loop formation by spoofing, false route error.

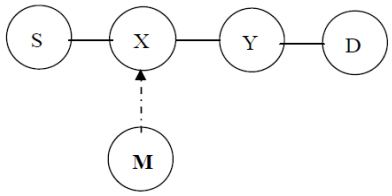

Fig. 1: Attacks using modification

Suppose node S in Figure 1 sends a RREQ with destination D. A malicious node M can receive it and read the destination sequence number as it is not encrypted. So M can send a RREP with greater sequence number to X. M can redirect traffic to itself. Node S will drop original copy of RREP, as it already has received a RREP with greater sequence number. In AODV protocol, the attacker can reset the value of hop count field so that it can later include itself with the route. There are two replay attacks in ad hoc network: RREQ flooding attack and wormhole attack [4][5].

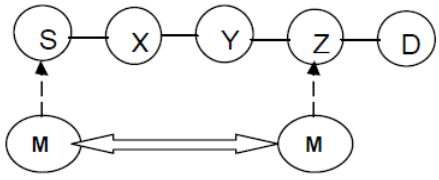

Fig. 2: Wormhole attack

In AODV protocol when a node needs to communicate with another node broadcasts RREQ to it's neighbors. The process continues until a route to the destination is found. S

wants to communicate with D, so it broadcasts a RREQ packet to it's neighbor X. Attacker M1 records the request and tunnels it through a fast channel to another attacker M2. Node Z will get the request from M2 and process it. Thus the attackers force to use the route via M1 and M2 to reach D.

AODV Protocol

In AODV protocol a source node wishing to communicate with a destination node first broadcasts a RREQ packet to its neighbors. On receiving, the desired destination node send reply packet RREP back to the source. Each node maintains only the next hop information to reach to destination.

In AODV protocol the route selection is based on following factors: hop count, destination sequence number. Hop count determines the length of the route and sequence number represents the freshness of the route information.The route selection metric is independent of trust factor of node and security level of application. By summarizing the attacks on AODV routing protocol, it is evident that secure neighbor detection and verification of node's position is the basic building block of our proposal. In RREQ some fields need to be secured. Hence some security mechanism for encryption/decryption must be adopted. In our proposed model, an additional parameter is added to determine the suitable route for any application: security level required by application [4].

Assumptions and Scenarios

The following figure represents the modules involved in our proposal.

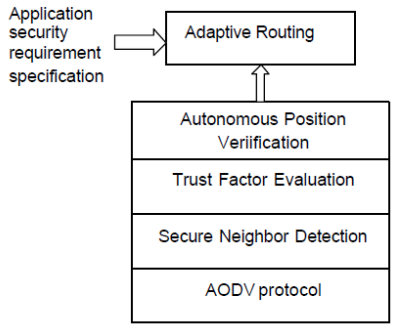

Fig. 3: Conceptual Framework

Scenario: The partners of company communicate through ad hoc network to exchange different ideas, policies and personal information. We classify different application with specific security requirement as follows.

Application	*Security requirement*
Exchange of new business ideas	Very high
Review of financial details of the company	High
Review of employee's performance	Low
Exchange of unofficial information	Very low

Fig. 4: Assumed security requirement of applications

2. Secure Ad Hoc Routing Protocol

Setup

Most of attacks on routing protocol are due to absence of encryption for some fields in the routing packets. Unauthorized modification of such fields could case serious security threats. We use DES for encryption mechanism. Each node in the network maintains a public/private key pair, certificate for public key identity signed by trusted certificate server and public key of trusted certificate server T. The certificate is to be valid for certain time period. Each node has T's public key, so it can decrypt certificates of other nodes. Each node maintains a neighbor table that contains TUSN (time stamped sequence number), neighbor ID, neighbor public key, location coordinates, neighbor group key, trust value of neighbor. Each initiator node maintains a node status table that contains destination ID, packet ID,forwarded (y/n) and unaltered (y/n). Each initiator node maintains recent destination list that contains destination ID, number of hops and time. Each node maintains a trust table that contains neighbor ID, trust value, trustworthy (y/n).

Secure Neighbor Detection

A node N broadcasts a hello message M1 with it's certificate. The target node receiving the message M1 decrypt N's certificate to verify and obtain N's public key. The target node sent the reply through message M2. After receiving the response, N stores the nodes public key and recent location coordinates of the target node in it's neighbor table. Node N records the sending time of M 1 at t0 and receiving time of M2 at t1 [6].

Total delay $d = t1 - t0$

Distance between the nodes must be less than $(d/2) * c$,

Where c is the speed of light. Thus node N can check that the other party is within its transmission range.

Trust Factor Evaluation

Each node maintains a database of it's neighbors with dynamically updated trust factor [2].

Neg_ID	*Trust value*	*Trustworthy*
X	6	Yes
Y	5	Yes
Z	3	No

Fig. 5: Trust table

Each node is assigned a trust value based on it's reliability. The trust value of the node can be –1 (malicious), 0 (not trusted), 1 to 3 (low trust level), 4 to 7 (standard trust level), 8 and 9 (high trust level). In our protocol, as long as the node's trust value = 4 it is assigned 'yes' meaning trustworthy otherwise it is 'no' meaning untrustworthy. Node1 authenticates it's neighbor Node2 using it's trust value. If Node2's trust value is in trust table and the value is 'yes', then Node2 is trusted. If the value is 'no', then Node2 is not trusted. If Node2 is not in the table, then Node1 will send a trust_request to all other trusted nodes for Node2's trust value.

Node Status Maintenance

The trust value of each node is selected based on node status. Each initiator node maintains node status information of it's neighbor nodes in form of table.

Neg_ID	*Packet_ID*	*Forwarded (Y/N)*	*Unaltered (Y/N)*
X	101	1	1
Y	102	1	0

Fig. 6: Node status table

Degrade Mechanism: The trust table is updated periodically for a predefined time period 't'. A threshold value 'P' is predefined used to detect a node as malicious. To evaluate the trust value of the node, we should count the number of successful forwards by the neighbor node. This can be done by applying logical AND operation to the last two fields and summing up all 1's generates the number of successful packet forwards [4].

Upgrade Mechanism: It uses the same algorithm for building the transfer string as explained in the previous paragraph. The success rate is computed by summing up the number of consecutive 1s from the LSB. If the success rate exceeds the threshold 'P' the trust factor of the node is incremented by 1.

Mode Selection

Additional routing fields are added in both RREQ and RREP packets. In RREQ field a two bit mode selection field is added. The mode field represents the required security level for the application. In general, the protocol consists of two operational modes [4].

Mode 0: No Encryption

In this mode, the protocol functions as a simple AODV protocol. The initiator can select this mode when the application does not require any security.

Mode 1: With Encryption & Trusted Path

In this mode, the protocol applies encryption mechanism to authenticate packets and packets are routed only along the trusted path.

Mode 2: With Encryption & Minimum Hop Count

In this mode, the protocol applies encryption mechanism to authenticate packets and packets are routed only along the shortest path.

Route Discovery

Route Request: A node wishing to communicate with destination node broadcasts the RREQ packet to its trusted neighbors. A RREQ contains the following fields: RREQ sequence number, destination ID, N's distance to D, D's position coordinates and TUSN, all encrypted with group encryption key [3][4]. The sequence number is incremented each time a node initiates a RREQ. TUSN represent the freshness of location information. The receiving node attaches the trust level of it's neighbor. The process repeats to all intermediate nodes until it reach the destination.

Route Reply: Upon receiving the RREQ the destination node respond with RREP packet containing RREQ sequence number, it's coordinates and TUSN. It signs the RREP with private key and encrypt it using group encryption key of it's neighbor. The reply propagates along the reverse path of RREQ. While receiving the RREP packet intermediate nodes decrypt it with their private key and verify the signature. Each intermediate node update the location field in neighbor table based on recent RREP packet [6].

An example: Suppose that a network is consisting of the nodes labeled S(source), D (destination) and from alphabet A to I. The source wishes to communicate with the destination. At first, the source selects the mode as 1 based on the required security level of application.

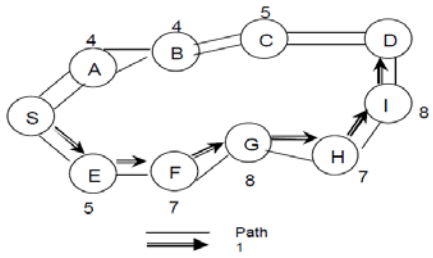

Fig. 7: Modified routing protocol

Neg_ID	*Trust value*	*Trustworthy*
A	4	Yes
E	5	Yes

Fig. 8: Trust table of source node S

The numbers shown closer to each node indicate their corresponding trust level. Node S to communicate with node D broadcasts RREQ to it's neighbors A and E. There are two possible paths from node S to D: S-A-B-C-D (path1), S-E-F-G-H-I (path2). Node A tries to authenticate the source node S. It checks it's trust table. If S is trusted, A accepts the RREQ message, update the location field and TUSN in it's neighbor table and broadcast the RREQ to the next node. If S cannot be trusted, A drops the RREQ. If S is not in A's table, A send a trust_request to S. If the response is 'yes', A stores the information in it's trust table and rebroadcasts the RREQ. When the response is not received within a limited time, node A drops the RREQ. As a result node A forwards to B, B forwards to C and C forwards to destination D. Similarly in path 2, E forwards to F, F forwards to G, G forwards to H, H forwards to I and I to destination D.

The destination D unicasts the RREP to C and I separately. Node C send the reply to node B. Node B forward the packet to A. But before sending, each node attaches the trust level of the node from where it just received the RREP. Upon receiving the RREP, each node update the recent destination list. The node attaches the trust level of C to trust string. So the trust string now contains the value 5. Node B forwards the RREP to A. Now the value of trust string is 4. The process continues until it reaches the source node. So the final value of trust string for the path 1 is 544. Similarly in path 2 node I forwards the RREP to I. The process will be similar as in path 1. The final value of trust string for the path 2 is 87875.

Now the source waits for a predefined time period to select the best route. The application requires trusted path for communication. The average trust weight of path 1 is 4.33 and trust weight of path 2 is 7. Hence path 2 is selected.

Autonomous Position Verification

The location based routing protocol require that a node be able to identify it's own position and position of destination node. This information is obtained via global positioning system (GPS) and location services. In the routing protocol,location information is distributed between nodes by means of position beacons.

All network used in MANETs have a maximum communication range. Based on these properties, we define acceptance range threshold 'T'. Position beacons received from nodes that are at position larger than 'T' away from current position of receiving nodes can be discarded. Position can also be verified based on the mobility of the node. It is assumed that all nodes move at well defined speed. When receiving a beacon the node records the arrival time of beacon. On receiving subsequent beacons, the node checks the average speed of nodes between two positions in two beacons. If the average speed exceeds mobility grade T, the position beacon is discarded [1].

Results and Future Work

```
A receives beacon from B
if distance(A's position, B's position) = T
 if B is in A's neighbor table
 update the position information of B
 else
 add B's ID, position details in A's table
else
 reduce trust value of B
 drop beacon
```

Fig. 9: Algorithm for position verification based on transmission range

```
A receives beacon from B
t=time of last beacon from B
if B is not in A's neighbor table
 add B's ID, position details in A's table
else
 old=position of B in A's table
 new=position information in beacon
 speed=distance(new,old)/(current time-t)
 if speed=Max.speed
 update position and time details
 else
 reduce trust level of B
 drop beacon
```

Fig. 10: Algorithm for position verification based on mobility

The protocol discussed overcomes all known vulnerabilities of the existing protcocols. It uses DES encryption mechanism to secure the fields in routing packets. The most severe attacks on MANETs is warm hole attack. The presented solution overcomes the attack by applying efficient secure neighbor detection mechanism. To enhance the security level of discovered path, route selection is done based on trust level of nodes along the path. In order to secure position coordinates of each node, we employ a position verification system. The proposed protocol can be simulated using network simulator like ns2.

Conclusions

In this paper proposed a secure routing protocol with autonomous position verification. The protocol follows different routing mechanism based on the security level required by application. In mode 1, the packets are routed along the trusted path based on the trust factor of the nodes. In mode2, the packets are routed along the shortest path based on hop count. The protocol uses a mechanism to detect and overcome the effect of falsified position information in geographic routing position. The protected position information reduces the routing overhead and increase the security of routing.

References

1 Elmar Schoch and Frank Kargl, Improved security in Geographic Ad hoc Routing through Autonomous Position Verification, Ulm University, Germany.

2 Huaizhi Li and Mukesh Singhla, A secure Routing protocol for Wireless Ad hoc Networks, Proceedings of the 39th Hawwaii International Conference on System Science, 2006.

3 Stephen Carter and Alec Yasinsac, Secure Position Arded Ad hoc Routing, Florida State Univeristy.

4 Abu Raihan Mostofa Kamal, Adaptive Secure Routing Protocol for Ad hoc Mobile Network, KTH, Sweden.

5 Yih-chun-hu and Adrian Perrig, A Survey of Secure Wireless Ad hoc Routing, University of California, Berkeley.

6 Yih-chun-hu and Adrian Perrig, Rushing Attacks and Defense in Wireless Ad hoc network Routing Protocols, ACM Conference on Wireless Security, September 2003

Authors

Sudhakar S is working as a Lecturer in Department of Computer Science and Engineering, Nandha College of Technology, Erode, TamilNadu, India. He received his M.E.(CSE) in 2007 from Anna University , Chennai , TamilNadu, India and pursing Ph.D in Anna University–Coimbatore. His areas of research interests include Mobile Computing and Network Security. He has published 4 papers in National Conferences, 2 International Conferences and 1 International Journals.

DR. S. Chenthur Pandian Ph.D is the Principal in Selvam College of Technology, Namakkal , Tamil Nadu, India.He received his B.Sc.(Maths) from Madurai Kamaraj University, Madurai in 1983 and AMIE from Institute of Engineers (India), Calcutta in 1984 and M.E. from Punjab University, Chandigarh in 1994 and Ph.D. in Fuzzy Application for Power Systems from Periyar Univeristy, Salem in 2005. He published National Journals 3, International Journals 4, and Conferences 34.And he published around 3 books. He is a member of MIEEE (USA), MISTE. His research interests include Fuzzy Application for Power Systems, Power Electronics, Neuro-Fuzzy System.