J. Basic. Appl. Sci. Res., 2(3)2576-2581, 2012
© 2012, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

ARAN Immunization against Wormhole Attack

Marjan Kuchaki Rafsanjani[1] and Mahmoud Eshraghi Samani[2]

[1]Department of Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran
[2]Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Kerman, Iran

ABSTRACT

Ad-hoc networks refer to temporary networks which form for special purposes. Mobile ad-hoc networks (MANETs) are networks which use no network assisting element for routing, but every node has the role of router and actively participates in data forwarding. Security problems in MANETs are specified and investigated separately, because in these infrastructureless networks, in addition to all available problems in a wired or wireless network, there are some more problems. Since all connections are available in wireless form and they are audible, so they can be changeable. Also nodes themselves are involved in routing, so existence of a malicious node can lead to the destruction of the network and one of the most important challenges in these networks is providing of a secure routing protocol algorithm. Here we investigate one of the secure routing protocols (ARAN) and try to increase its security against wormhole attack which is one of the famous attacks .This kind of attack by cooperation of two or more nodes create for transferring of information from their private network or tunnelling of information. Then we will try to decrease wormhole attacks through this method and also minimize problems resulting from using packet leash with different methods and changes its structure.

KEY WORDS: ARAN; Component; Temporal leashes; Wormhole Attack; Secure routing protocol.

I. INTRODUCTION

Mobile ad-hoc networks are interim networks which use for special application and purposes. Actually they are wireless networks with mobile nodes. Major difference of ad hoc networks with common wireless 802.11 networks is that in ad-hoc networks there is a collection of wireless mobile nodes without any infrastructure like central station, router, switch or any other things which are used in other networks [2].

Mobile nodes are equipped with receiver and transmitter for making wireless connections. Mobile nodes cannot make contact with all nodes directly because of some limitations in receiver and transmitter. Therefore data transfer through other nodes when there is no direct connection between them. However because of mobile nodes, the network is constantly changing and different paths appear between two nodes. We can refer to personal applications among MANETs like connection of laptops together, public applications like communication of vehicles and taxies, military applications like communication of war ships, and emergency applications like rescue and relief operations. MANETs are networks which use no network assisting element, but they have cooperative nodes in the network responsible for routing. These networks may have various applications due to no using of predetermined infrastructure. These networks can be easily started up, used and finally removed. Advantage of this network is its speed and easy operation and also it has no dependency on predetermined infrastructures. So, only those nodes in effective range of other nodes can receive each other's message and recognize it from noise environment and each node also both are used as end-system and as routing for other nodes in the network [3, 4].

II. WORMHOLE ATTACK

One of famous special attacks of MANETs is Wormhole attack. During the attack, two malicious nodes make a short connection cooperatively in network's topology. Mentioned attacks with following order:
Requesting of routing through one node reaches to one of the malicious nodes .Then malicious node send this request to second node through one private network or through tunnelling. Now, if these two nodes do not change hop counter value then a long amount of path has been passed through the private network without increasing hop values. Thus it is possible to get to the destination just with two hopes rather than ten hopes. In this case certainly this path chooses as the shortest path. Therefore both are involved in created path. These two nodes cooperate together and force original node to accept relatively incorrect routing information [7, 8].

***Corresponding Author:** Marjan Kuchaki Rafsanjani, Department of Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran. Email: kuchaki@mail.uk.ac.ir

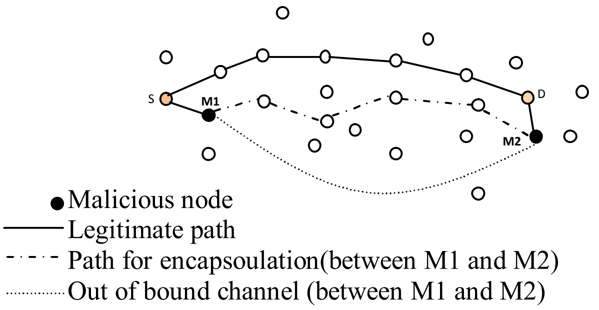

S
M1
D
M2
● Malicious node
Legitimate path
Path for encapsoulation(between M1 and M2)
Out of bound channel (between M1 and M2)

Figure 1. Sample (Wormhole attack)

Wormhole Attack Effects

It can be demonstrated that if the amount of malicious nodes to be n>1 then average amount of (1-1/n) connections are affected. Also wormhole attack leads to DoS (Denial of Service) attack with removing of data or removing of control packet. Wormhole attack can lead to Gray hole attack or Black hole attack and malicious nodes can perform statistical analysis of data flow [5].

III. ARAN PROTOCOL (AUTHENTICATED ROUTING FOR AD_HOC NETWORK)

A Secure Routing Protocol for Ad Hoc Networks [1] which is an on demand protocol designed for performing of secure communications in open environment. At first, before initiation of the work, nodes' connection is authenticated. This authentication performs through a secure T routing authentication server and nodes in ARAN receive a certification after their authentication in a secure manner for T routing authentication server. Nodes use this certification in transferring during message routing with other nodes in order to authenticate themselves and finally for authentication of end to end at the time of routing. In this protocol, encryption with public key is used. Routing in ARAN performs through a routing message from original broadcast node which is responded through destination node as unicast. ARAN protocol requires using a secure T certification issuer server which its public key is considered valid for all nodes. First, keys are made and transferred between each node and T server. Each node before entering the network should ask a certification from T. Each node receives a certification after its authentication. This certification includes node's IP address and its public key and certification issuing time and certificate expiration time in which these fields are connected together and signed by T server. One node like A receives a certification from *T* server as following:

$T \longrightarrow A\text{: } cert_A = [IP_A, K_{A+,}t, e]K_{T-}$

TABLE I. TABLE OF VARIABLES AND NOTATION

K_{A+}	**Public key of nod A**	**K_{A-}**	**Private key of nod A**
{d} K_{A+}	Encryption of data d with key K_{A+}	{d} K_{A+}	Data d digitally signed by node A
N_a	Nonce issued by node A	$cert_A$	Certificate belonging to node A
IP_a	IP address of node A	t	Time stamp
RDP	Route Discovery Packet identifier	e	Certificate expire time
REP	REPly packet identifier	ERR	ERRor packet identifier

The goal of end-to-end authentication is for the source to verify that the intended destination was reached. In this process, the source trusts the destination to choose the return path. Source node A begins route instantiation to destination X by broadcasting to its neighbors a *route discovery packet* (RDP):

$A \longrightarrow \text{ brdcast: } [[RDP, IP_X, cert_A, N_A, t]K_{A-}$

RDP Includes following:

Route discovery packet (RDP), IP address of destination (IP_X), A's certificate ($cert_A$), a nonce N_A and the current time t, all signed with A's private key. Other nodes save nonce along with timestamp seen for the last time for a special node. Hop count is not included with the message. When a node receives an RDP message, it setup a reverse path back to the source by recording the neighbor from which it received the RDP. This is in anticipation of eventually receiving a reply message that it will need to backward to the source. The receiving node uses A's public key, which it extracts from A's certificate, to validate the signature and verify that A's certificate has not expired. The receiving node also checks the (N_A,IP_A) tuple to verify that it has not already processed this RDP. Nodes do not forward messages for which they have already seen the tuple;

otherwise, the node signs the contents of the message, appends its own certificate, and broadcasts the message to each of its neighbors. The signature prevents spoofing attacks that may alter the route or form loops. Let B be a neighbor that has received from A the RDP broadcast, which it subsequently rebroadcasts.

$B \longrightarrow \text{brdcast: } [[\text{RDP},\text{IP}_X,\text{cert}_A,N_A,t]K_A\text{-}]K_B\text{-},\text{cert}_B$

Upon receiving the RDP, B's neighbor C validates the signature with the given certificate. C then removes B 's certificate and signature, records B as its predecessor, signs the contents of the message originally broadcast by A , appends its own certificate, and forward broadcasts the message. C then rebroadcasts the RDP again.

$C \longrightarrow \text{brdcast: } [[\text{RDP},\text{IP}_X,\text{cert}_A,N_A,t]K_A\text{-}]K_C\text{-},\text{cert}_C$

Each node along the path repeats these steps of validating the previous node's signature, removing the previous node's certificate and signature, recording the previous node's IP address, signing the original contents of the message, appending its own certificate and forward broadcasting the message.

Finally, the destination X receives the message; it replies to the first RDP that receives for a source and a given nonce. It does not guarantee that the first received RDP transferred from the source in the shortest path. An RDP that travels along the shortest path may be prevented from reaching the destination as the first RPD if it encounters congestion or network delay, either legitimately or maliciously. In this case, however, a non-congested, non-shortest path is likely to be preferred to a congested shortest path because of the reduction in delay. Because RDPs do not contain a hop counter specific recorded source route and due to messages are signed at each hop, malicious nodes have no opportunity to redirect traffic with the exploits.

After receiving the RDP, the destination unicasts a Reply (REP) packet back along the reverse path to the source. Let the first node that receives the REP sent by B be node D.

$X \longrightarrow D : [\text{REP},\text{IP}_A,\text{cert}_x,N_A,t]K_X\text{-}$

The REP includes:

A packet type identifier (REP), the IP address of A(IP.), the certificate belonging to X (cert.), the nonce and associated timestamp sent by A . Nodes that receive the REP, forward the packet back to the predecessor from which they received the original RDP. Before forwarding the REP to the next hop, each node along the reverse path back to the source signs the REP and appends its own certificate. Consider node C is the next hop of node D.

$D \longrightarrow C : [[\text{REP},\text{IP}_A,\text{cert}_x,N_A,t]K_X\text{-}]K_D\text{-},\text{cert}_D$

C validates D's signature on the received message, removes the signature and certificate, then signs the contents of the message and appends its own certificate before unicasting the REP to B.

$C \longrightarrow B : [[\text{REP},\text{IP}_A,\text{cert}_x,N_A,t]K_X\text{-}]K_C\text{-},\text{cert}_C$

The nonce and signature of the previous hop are checked by each node as the returned REP to the source. This prevents malicious nodes attacks where they instantiate routes by impersonation and replay of X's message. When the source receives the REP, it verifies the returned nonce and signature of the destination.

In ARAN nodes keep track of active routes. When no traffic has occurred on an existing route for that route's lifetime, the route is simply deactivated in the route table. Received data on an inactive route causes nodes to create an Error (ERR) message that transfers in the reverse path to the source. Nodes also use ERR messages to report links in active routes that are broken due to node movement. All ERR messages must be signed. For a route between source A and destination X, a node B generates the ERR message for its neighbor C as following:

$B \longrightarrow C : [\text{ERR},\text{IP}_A,\text{IP}_X,\text{cert}_b,N_b,t]K_B\text{-}$

This message is forwarded along the path toward the source without modification. A nonce and timestamp ensure that the ERR message is fresh.

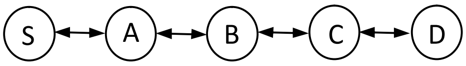
Unpredictable behaviors may originate from a malicious node. It can also originate from a friend node incorrectly. ARAN's response does not differentiate between the two and regards all erratic behavior as the same. Erratic behavior includes the use of invalid certificates, improperly signed messages, and misuse of route error messages.

If a certificate needs to be revoked, the *T* server sends a broadcast message to the ad hoc group that announces the revocation.

T ⟶ brdcast : [revoke, cert_r]K_{T}-

Each node that receives this message, broadcasts it to its neighbors. Revocation announcements require being stored until the revoked certificate would have expired normally [1].

Suppose that node S in this network needs to go to the path which directs in to D. The transmission performance protocol appears as following:



S → brdcast: (RDP,D,cer_s,N,t)$K_{s\text{-}}$
A → brdcast: (RDP,D,cer_s,N,t)$k_{s\text{-}}$)$K_{A\text{-}}$,cert_A
B → brdcast: ((RDP,D,cer_s,N,t)$k_{s\text{-}}$) $K_{B\text{-}}$,cert_B
C → brdcast: ((RDP,D,cer_s,N,t)$k_{s\text{-}}$) $K_{c\text{-}}$,cert_c
D → C: (REP,S,cert_D,N,t)$K_{D\text{-}}$
C → B: ((REP,S,cert_D,N,t)$K_{D\text{-}}$) $K_{c\text{-}}$,cert_c
B → A: ((REP,S,cert_D,N,t)$K_{D\text{-}}$) $K_{B\text{-}}$,cert_B
A → S: ((REP,S,cert_D,N,t)$K_{D\text{-}}$) $K_{A\text{-}}$,cert_A

One of the problems of this protocol is its non resistance against wormhole attack. Another problem of this protocol is that it uses Asymmetric cryptography system. Therefore it has high amount of energy consumption and processing and it makes too much problems in limited MANETs [9].

For example in figure 1 node S attempts to discover the path for making connection with node D, so performs a routing packet of broadcast. When this packet receives by node M1 changes it in to data and sends for node M2 (or sends it in to its neighbor node through the private network) according to this fact that using of the private network or sending of routing as data needs no authentication. Therefore this packet reaches to the destination faster than other paths and therefore wormhole attack happens.

IV. DECREASING OF WORMHOLE ATTACKS

Applicable technique is by using packet leash .This technique influences on the limitation of packet life's time and according to packets speed influences on passed distance by packets which equals maximum amount of speed of light.

To construct a temporal leash, in general, all nodes must have tightly synchronized clocks, such that maximum difference between any two nodes' clocks is Δ. The value of the parameter Δ must be known by all nodes in the network and for temporal leashes, generally must be on the order of a few microseconds or even hundreds of nanoseconds. This level of time synchronization can be achieved now with off the shelf hardware based on LORAN-C, WWVB or GPS. Esoteric hardware such as cesium-beam clocks, rubidium clocks, and hydrogen maser clocks, could also be used in special applications today to provide sufficiently accurate time synchronization for months.

We can find this fact in temporal leash of t_s sending packet and time of receiving t_r packet .Therefore according to speed (maximum speed of light), we can find that whether packet has passed the path more than enough or not. If transmitter wants to prevent packet sending for paths more than L (m) then it must be more than $L_{min}=c.\Delta$ (It is supposed that the speed of light in the air equals light speed in the void) [6].

Temporal leashes analysis

According to research conducted in [10], one of the problems of using temporal leashes is determination of the amount of the path as a limitation for the packet. This packet strictly prevents the passing of all packets from routes longer than this path and allows all shorter paths of packets to pass. For example, if we consider as L then nodes S and D in the L space are L'>L. Therefore, S node never can send any message to the node D, because node D removes the packet as it receives the packet and according to the limitation from packet leash. On the other hand, if there are two nodes D and S between M2 and M1 as $L''=d_{SM1}+d_{M1M2}+d_{M2D}<L$ then the packet leash cannot prevent occurrence of Wormhole.

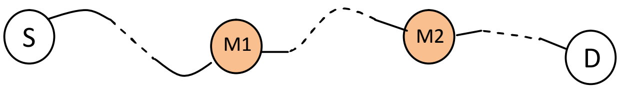


Figure 2. Wormhole prevention

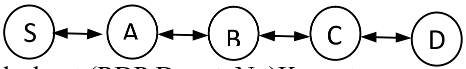
Therefore long routing pass of the packet cannot be considered as occurrence of Wormhole. So receiver node should calculate the real time for passing of the packet and subtract packets' delay in middle nodes from this time. Total amount of delays imposed on the packet can be calculated by following formula:

D_t=h.D_h

Where D_h is delay in one step, h is the number of passed steps by the packet D_h is separate to 3 sections; each of them determines delay in one of the layers; layers of network, MAC or Radio. These delays influence routing, but the delay of upper layers is not effective in the amount of delay. In the sender node, the timestamp is located in the network's layer in packet header. Each middle node brings the packet up to the network layer and then sends it again to the lower layer after necessary investigations. Finally receiver node receives packet and records time after entering packet to network layer. Calculation of delay values related to each layer is dependent on algorithm used in it. Delay in the layer as used DSR algorithm is different from delay value in the layer while we use AODV routing protocol. The main problem in calculation happens when for example in MAC layer we use 802.11 algorithm, This layer sense media for data sending and preventing of the collision and it also waits when the occupied value equals n slot and then sends the packet (n is a random number). For solving this problem we can consider different values based on kind of network and severity of Wormhole prevention and network traffic anticipated. If we consider low amount of delay then we can assure that no Wormhole attack happens. But it is also possible that some admissible packets to be removed if we consider high amount of delay, then we can assure that no valid packet will be removed but there is possibility of Wormhole attacks and we try to use average amount of delays.

ARAN Immunization against wormhole

We create some changes in routing and add a hop counter to it. Each of the middle nodes adds to the hop counter and the remained part of process performs as before. For clear understanding of mentioned issue please consider new changes.



S $\longrightarrow$ brdcast:(RDP,D,cert$_s$,N,t)K$_{s-}$,cert$_s$
A$\longrightarrow$ brdcast:(RDP,D,cert$_s$,N,t)K$_{s-}$,cert$_s$,h) K$_{A-}$,cert$_A$
B $\longrightarrow$ brdcast:(RDP,D,cert$_s$,N,t)K$_{s-}$,cert$_s$,h) K$_{B-}$,cert$_B$
C $\longrightarrow$ brdcast:(RDP,D,cert$_s$,N,t)K$_{s-}$,cert$_s$,h) K$_{C-}$,cert$_C$
D $\longrightarrow$C: ((REP,S, cert$_D$,N,t) K$_{D-}$
C $\longrightarrow$B: ((REP,S, cert$_D$,N,t) K$_{D-}$) K$_{C-}$,cert$_C$
B $\longrightarrow$A: ((REP,S, cert$_D$,N,t) K$_{D-}$) K$_{B-}$,cert$_B$
A$\longrightarrow$ S: ((REP,S, cert$_D$,N,t) K$_{D-}$) K$_{A-}$,cert$_A$

Time of packet creation exist inside the time stamp routing that we show it with t. we show the time of reaching the destination with t_r sign, h is the number of hopes passed by routing packet and D shows the total amount of delay and emphasizes the amount of time that this packet delays in the nodes .

t_{hmax} shows maximum time required for transferring of message from one node to the other node. So, just those routings are accepted that have the following condition:

t_r-t-D<=h.t_{hmax}

Above relation indicates that if the real time of passing for packet is less than required time for passing with h step then routing packet will be accepted. However, we should consider this fact that may be nodes are not in maximum amount of distance from each other. So it is possible to use t_{havg} instead of t_{hmax} which shows average required time for passing from the distance between two nodes and increase the confidence rate of preventing from Wormhole incidence. Any way in performance period, we can obtain appropriate value of t_{havg} according to environmental conditions. We can rewrite this relation as following:

t_r-t-D<=h. t_{havg}

This condition provides more ideal situation when there is a large number of nodes in the network. We can obtain maximum amount of delay for the packet through following relation:

D=[2(h+1)-1].t_{ds}

t_{ds} is required time for digital signature.

Considering the fact that in ARAN the first packet is responded by the destination. Therefore, attacker can escape from destination calculations by increasing h and also show it admissible. In order to remove this problem, the destination node controls if the number of registered hops in the routing package is reasonable or not. To do so the destination node investigates whether the number of registered hops in the routing packet is based on the following relation or not:

h<=c.(tr-t-D)/average distance of nod (c: Light Speed)

Current relation obtains the amount of distance that routing packet can pass and divides it to nodes distance. The more the distance average is the less possibility of Wormhole will be. If this distance too be excessive and more than enough then allowed packets will be removed. Any way, it is better to make this decision based on environment and conditions of related ad-hoc network.

V. DISCUSSION

In ARAN routing protocol the cryptography of public key is used. Therefore, we should use nodes with good processing capacity in performing. Also time distance between two nodes to the required time for performing of calculations should not be negligible.

REFERENCES

[1] Kimaya Sanzgiri, Bridget Dahill, Brian Neil Levine, Clay Shields, and Elizabeth M. Belding-Royer, A secure routing protocol for ad hoc networks, Proceedings of the 10th IEEE International Conference on Network Protocols(ICNP'02), CA, USA, pp. 78-87, 2002.

[2] Mohammad O. Pervaiz, Mihaela Cardei, and Jie Wu, Routing security in ad hoc wireless networks, In: Network Security, Springer, 2005.

[3] Siddhartha Gupte, and Mukesh Singhal, Secure routing in mobile wireless ad hoc networks, Ad Hoc Networks, pp.151–174, 2003.

[4] Patroklos G. Argyroudis, and Donal O'Mahony, Secure Routing for Mobile Ad hoc Networks, IEEE Communications, vol. 7, no. 3, 2005.

[5] Majid Khabbazian, Hugues Mercier and Vijay K. Bhargava, NISO2-1: Wormhole Attack in Wireless Ad Hoc Networks: Analysis and Countermeasure, Proceedings of the IEEE Global Telecommunications Conference (LGLOBCOM '06), San Francisco, CA, USA, pp. 1-6, 2006.

[6] Yih-Chun Hu , Adrian Perrig, and David B. Johnson Packet leashes: A defense against wormhole attacks in wireless ad hoc networks, Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications (INFOCOM 2003), Pittsburgh, PA, USA , vol. 3, pp. 1976-1986, 2003.

[7] Issa Khalil, Saurabh Bagchi, and Ness B. Shroff - MOBIWORP: Mitigation of the wormhole attack in mobile multihop wireless networks, Ad Hoc Networks, vol. 6, pp. 344–362, 2008.

[8] Marianne A. Azer, Sherif M. El-Kassas, and Magdy S. El-Soudani, Immuning routing protocols from the wormhole attack in wireless ad hoc networks, Proceedings of the Fourth International Conference on Systems and Networks Communications, Egypt, pp. 30-36, 2009.

[9] Stefano Basagni, Marco Conti, Silvia Giordano, and Ivan Stojmenovic, Mobile Ad-hoc Networking, IEEE press, John Wiley and Sons publication, New York, USA, 2004.

[10] Masih Moosapoor, Design and Evaluate a Light-Weight Secure Routing Algorithm in Ad-Hoc Networks, Master of Science, Amirkabir University of Technology, Tehran, 2007.