Authenticated Routing for Ad Hoc Networks Protocol and Misbehaving Nodes

Abdalla Mahmoud Ahmed Sameh Sherif El Kassas
Department of Computer Science
The American University in Cairo, Egypt.
P.O. Box 2511, Cairo
EGYPT

Abstract: - In this paper, we analyze one of the secure mobile ad hoc networks protocols, which is Authenticated routing for ad hoc networks (ARAN). Such protocol is classified as a secure reactive routing protocol, which is based on some type of query-reply dialog. That means ARAN does not attempt to continuously maintain the up-to-date topology of the network, but rather when there is a need, it invokes a function to find a route to the destination. Here, we detail how ARAN works, criticize how an authenticated misbehaving node can abuse the bandwidth and propose different solutions for this flaw in the protocol.

Key-Words: - Mobile Ad hoc Networks, Routing Security, ARAN, Misbehaving nodes, Cryptographic certificates, Reputation Systems.

1 Introduction

The ARAN secure routing protocol proposed by Sanzgiri, Laflamme, Dahill, Levine, Shields and Belding-Royer is seen as an on-demand routing protocol that detects and protects against malicious activities caused by other nodes and peers in the ad hoc environment. This protocol introduces authentication, message integrity and non-repudiation as part of a minimal security policy for the ad hoc environment. Our evaluation of ARAN shows that it has minimal performance costs for the increased security in terms of processing and networking overhead. However, by analyzing the protocol thoroughly, we figured out that an authenticated misbehaving node can consume the whole bandwidth of the network.

This paper is organized as follows. Section 2 presents the secure ad hoc routing protocol, ARAN. Section 3 discusses the problem that an authenticated misbehaving node can infer to this protocol performance. Section 4 presents the solution to this problem and the simulation results of the original protocol and our modified version of the protocol to account for authenticated and misbehaving nodes participation in the network. Finally Section 5 offers concluding remarks and future work.

2 Authenticated Routing for Ad Hoc Networks

ARAN uses cryptographic certificates to prevent and detect most of the security attacks that an ad hoc network can face. These attacks include: remote redirection, tunneling, spoofing and fabrication.

ARAN consists of a preliminary certification process followed by a route instantiation process that guarantees end-to-end authentication. Thus, the routing messages are authenticated end-to-end and only authorized nodes participate at each hop between source and destination.

2.1 Certification Process

ARAN requires the uses of a trusted certificate server T, whose public key is known to all valid nodes. Keys are a priori generated and exchanged through an existing out of band relationship between T and each node. Before entering the ad hoc

network, each node must request a certificate from T. Each node receives exactly one certificate after securely authenticating their identity to T.
So a node A receivers a certificate from T as follows:

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

The certificate contains the IP address of A, the public key of A, a timestamp t of when the certificate was created and a time e at which the certificate expires. These variables are concatenated and signed by T. Nodes uses these certificates to authenticate themselves to other nodes during routing messages exchange [2].

2.2 Authenticated Route Discovery

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

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

The RDP includes a packet type identifier ("RDP"), the IP address of the destination X, a nonce N_A, A's certificate and all signed by A's private key, K_{A-}. The purpose of the nonce is to uniquely identify an RDP coming from a source. Each time A performs route discovery, it monotonically increases the nonce. This nonce variable is large enough so that not to need to be recycled throughout the lifetime of the network.
When a node receives an RDP message, it setups up a reverse path back to the source by recording the neighbor from which it received the RDP. Therefore, it is ready, upon receiving a reply message, to forward back to the source. Furthermore, the receiving nodes 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. And it also checks the $\{N_A, IP_A\}$ tuple to verify that it has not already processed this RDP, since nodes do not forward messages with already-seen tuples.
Then the receiving node signs the content 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:

$B \rightarrow brdcast: [[RDP, IP_X, N_A]K_{A-}], K_{B-}, cert_A, cert_B$

Upon receiving the RDP, B's neighbor C validates the signatures for both A, the RDP initiator, and B, the neighbor it received the RDP from, using the certificates in the RDP. C then removes B's certificate and signature, records B as its predecessor, signs the contents of the messages originally broadcast by A and appends its own certificate. C then rebroadcast the RDP[2]:

C->brdcast:[[RDP,IPX,NA]KA-]KC-,certA, certC

Each intermediate node in the path repeats the same steps as C [2].

2.3 Authenticated Route Setup

Afterwards, the message is received by the destination, X, who replies to the first RDP that it receives for a source and a given nonce. There is no guarantee that the first RDP received traveled along the shortest path from the source. Because RDPs do not contain a hop count or specific recorded source route and since messages are signed at each hop, malicious nodes have no opportunity to redirect the traffic.
So by 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 X to be node D:

$X->D:[REP,IP_A,N_A]K_{X-},cert_X$

The REP includes a packet type identifier ("REP"), the IP address of A, the nonce sent by A, the X's certificate and all signed by X's private key, K_{X-}. Nodes that receive the REP forward the packet back to the predecessor from which they received the original RDP. Each node along the reverse path back to the source signs the REP and appends its own certificate before forwarding the REP to the next hop.
Now let D's next hop to the source be node C:

$D->C:[[REP,IP_A,N_A]K_{X-}]K_{D-},cert_{X,}cert_D$

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

$C->B:[[REP,IP_A,N_A]K_{X-}]K_{C-},cert_{X,}cert_C$

Each node checks the nonce and signature of the previous hop as the REP is returned to the source. This avoids the attacks where malicious nodes instantiate routes by impersonation and replay of X's message. Finally, when the source receives the REP, it verifies the destination's signature and the nonce returned by the destination[1].

2.4 Route Maintenance

When no traffic has occurred on an existing route for that route's lifetime, the route is simply deactivated in the routing table. Data received on an inactive route causes nodes to generate an Error (ERR) message. Also, nodes use ERR messages to report links in active routes that are broken due to node movement. Of course, all ERR messages is signed.
On the other hand, it is extremely difficult to detect when ERR messages are fabricated for links that are truly active and not broken. That's why having messages signed prevents impersonation and enables non-repudiation. So a node that transmits a large number of ERR messages, whether the ERR messages are valid or fabricated should be avoided.

2.5 Key Revocation

In the event that a certificate needs to be revoked, the trusted certificate server, T, sends a broadcast message to the ad hoc networks announcing the revoked node. And any node receiving this message rebroadcasts it to its neighbors. Moreover, revocation notices need to be stored until the revoked certificate expire normally [2].

3 Authenticated misbehaving node problem

In the previous section, we have introduced the Authenticated Routing for Ad hoc Networks protocol and discussed the different phases of it. And we showed how unauthenticated nodes, spoofed route signaling, fabricated routing messages, alteration of routing messages, and replay attacks are not prevented by the usage of certificates in ARAN routing communications.
In this section, we will be discussing the problem that we have discovered in this protocol. It is mainly in the part that we presented in Section 2.C. In this earlier section, we mentioned that after receiving the RDP, the destination unicasts a Reply packet, REP, to the source. And every intermediate node in this path verifies the signature of its forwarder, removes the signature and certificate and then signs using its own private key. So the problem is that it is only when the source receives the REP, it verifies the destination's signature! So that brings us to the problem of having an authenticated and misbehaving node in the ad hoc network. If this misbehaving node generated a fake REP and signed it twice, once as the destination and once as an intermediate node, then this misbehaving node will go undiscovered until the source is reached, and bandwidth will be consumed. Thus, this

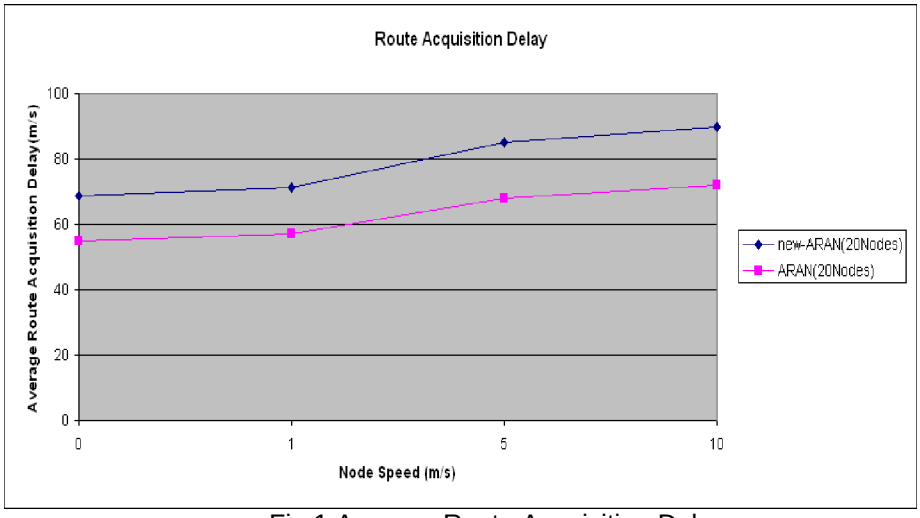

Fig.1 Average Route Acquisition Delay

problem makes ARAN protocol open to such an attack and the network can end up flooded with fake REPs from the authenticated and misbehaving nodes in it.

4 Proposed Solution and simulation Results

To solve the above mentioned problem, we needed to add in the ARAN algorithm another signature verification phase. This phase is mainly to check at each hop two signatures and not just one as the original ARAN protocol was handling it. So at each hop we check the signature of the forwarder node and do check the signature of the sender of this REP packet.

We have performed our evaluations with the new change using the Global Mobile Information Systems Simulation Library (Glomosim) [3]. We used an IEEE 802.11 MAC layer and CBR traffic over UDP.

We did the simulation with the following configuration: We simulated 20 nodes distributed over a 670m X 670m terrain. The initial positions of the nodes were random. Node mobility was simulated according to the random waypoint model. The node transmission range was 250m. We ran the simulations for constant node speeds of 0,1,5,10 m/s, with pause time fixed at 30 seconds. We simulated five CBR sessions in each run, with random source and destination pairs. Each session generated 1000 data packets of 512 bytes each at the rate of 4 packets per second. In addition, ARAN was simulated using 512 bit key and 16 byte signature. These values are reasonable to prevent compromise during the short time nodes spend away from the certificate authority and in the ad hoc network [1].

In order to compare the performance of ARAN and the new-ARAN, modified to face authenticated and misbehaving nodes, both protocols were run under identical mobility and traffic scenarios.

We mainly evaluated the average route acquisition latency delay. This performance metric is defined as the average delay between the sending of a route request/discovery packet by a source for discovering a route to a destination and the receipt of the first corresponding route reply. If a route request timed out and needed to be retransmitted, the sending time of the first transmission was used for calculating the latency.

In fig.1, we show the observed simulation result for both ARAN and new-ARAN. In this figure, we show that the average route acquisition latency for new-ARAN is approximately 25% more than that for the

ARAN. While each node has to verify one digital signature of the previous node in ARAN, each node is required to verify two digital signatures in our new-ARAN, one signature of the sender and the other of the previous node. This explains the additional delays at each hop, and so the route acquisition latency increases.

5 Conclusion and Future Work

According to our analysis and study of the ARAN protocol, we have concluded that it is an excellent secure routing protocols as it provides authentication, integrity and non-repudiation services using cryptographic certificates that guarantees end-to-end authentication. However, we have got to discover that an authenticated and misbehaving node can flood the ad hoc network with fake REPs and go undiscovered until the source node discover the fake authenticity of this packet . So that will render the network congested and consume its whole bandwidth. So we presented our solution by introducing the new-ARAN that prevents such type of misbehavior attack.

As for our future work, we are in the phase of studying the applicability of adding a reputation system like the ones discussed in [4], [5] and [6]. These reputation systems will help our new-ARAN protocol to detect and stop the several kinds of misbehaviors that authenticated nodes can cause to the ad hoc environment.

References

[1] Kimaya Sanzgiri, Daniel Laflamme, Bridget Dahill, Brian Neil Levine, Clay Shields, Elizabeth M. Belding-Royer. Authenticated Routing for Ad Hoc Networks. MobiCom 2005.

[2] Refik Molva and Pietro Michiardi. Security in Ad hoc Networks Personal Wireless Communication, September 23-25 2003.

[3] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla, "GlomoSim: A Scalable Network Simulation Environment," UCLA, Tech. Rep. CSD Technical Report #990027, 1997.

[4] Zygmunt J. Haas, Jing Deng, Ben Liang, Panagiotis Papadimitratos, and S. Sajama, "Wireless ad hoc networks," in Encyclopedia of Telecommunications, John G. Proakis, Ed. Wiley, 2002.

[5] Manel Guerrero Zapata. Secure Ad hoc On-Demand Distance Vector Routing. Mobile Computing and Communications Review, Vol. 6 Number 3.

[6] Prashant Dewan, Partha Dasgupta and Amiya Bhattacharya. On Using Reputations in Ad hoc Networks to Counter Malicious Nodes. QoS and Dynamic Systems, (in conjunction with IEEE ICPADS), Newport Beach, USA, 2004.