Presented in International Conference on Information Technology, 2009 (**ICIT 2009**), Bhubaneswar, India, December 21-24, 2009

A Secure AODV Routing Scheme with Reduced Data Packet Delay in MANET

Alekha Kumar Mishra[1], Bibhudatta Sahoo[2]
Department of Comp. Sc. and Engg., NIT Rourkela, Orissa India
[1]E-mail: alekha@gmail.com
[2]E-mail: bdsahu@nitrkl.ac.in

ABSTRACT

Among the various security mechanisms that has been proposed for Ad hoc On-Demand Distance Vector (AODV) routing protocol, the secure extension of AODV (SAODV) is most popular and efficient. Since SAODV based on digital signature mechanism for authenticating routing packet of AODV, it consumes heavy computation time while generating and verifying a signature. The condition becomes worse for an intermediate node in ad hoc network and in turns degrades the performance. Adaptive-SAODV mechanism is a step towards enhancing the performance with respect to data packet delay of a node in the above mentioned scenario. In this paper we have proposed an algorithm that based on the adaptive decision of an intermediate node that depends on its load state of current node and neighbors. The performance of the proposed algorithm with A-SAODV and SAODV has been presented with data packet delay and throughput as metric.

Keywords
SAODV, A-SAODV, TTL, routing queue length, RREQ, RREP

1. INTRODUCTION

Considering various security issues of AODV[1,2] routing protocol several secure AODV routing protocol has been proposed featuring variety of advanced mechanism for securing data and control information. secure Ad hoc On Demand Vector routing (SAODV) [3,4] is one of the popular existing secured mechanism which takes help of digital signature and hash chain techniques to secured AODV packets. SAODV enables each node to sign an outgoing message with its own secret key and verify all incoming message with the public key shared by other nodes. Since, digital signature technique is based on asymmetric key cryptographic method [9], heavy amount of computational time is required for signature and verification mechanism [5], and hence it affects the performance of SAODV protocol.

Since SAODV has been proved to be free of most of the security issues of AODV protocol, our objective is to propose some changes in routing behavior of SAODV which in turn will improve its performance. In a recent work called Adaptive-SAODV (A-SAODV) [5], an adaptive mechanism that tunes the behavior of SAODV to improve its performance. It makes an adaptive decision whether to reply an incoming request based on the load threshold value of the current node provided it has a valid and fresh route to the requested destination. This decision helps to balance the load of intermediate nodes which are over-burdened by signing and verification task of incoming messages.

In our paper we have proposed an extension to Adaptive-SAODV, which includes further filtering strategies aimed at further improving its network performance parameters like first data packet delay and average throughput. We then tired to analyze and simulate the proposed algorithm to see help in further reduction of data packet delay in adaptive SAODV and also compared its performance with existing mechanisms using simulation.

The remaining sections of this paper are organized as follows. Section 2 briefly explains Secure AODV protocol with its message securing mechanisms like digital signature and hash chain. Section 3 describes performance issues of SAODV followed by the adaptive mechanism used in Adaptive SAODV to tune its performance in section 4. The proposed work has been discussed in section 5. It includes the algorithm and mechanism of proposed modification followed by analysis and simulation results in section 6.

2. SECURE AD HOC ON-DEMAND DISTANCE VECTOR ROUTING (SAODV)

SAODV[3,4] is based on public key cryptography and it extends the AODV message format to include security parameters for security the routing messages.

Considering Route Request (RREQ) and Route Reply (RREP) message in SAODV protocol there are two alternatives for ensuring secured route discovery; first, the basic one where only destination is allowed to reply a RREP and the second, any intermediate node which has valid routing information allowed to reply a RREP. Two mechanisms are used to secure the routing message. Digital Signature [9,10] is used to authenticate and preserve integrity of non-mutable fields' data in RREQ and RREP messages. For non-mutable field the authentication is done in an end-to-end manner. Hash chain is used to secure mutable field like hop count information. The two mechanisms have been discussed in brief in following sections.

2.1 Hash Chain

The hash chain mechanism [10] helps any intermediate node to verify that the hop count has not been decreased by any malicious node. A hash chain is formed by applying a one-way hash function repeatedly to a seed (random number). When a node needs to send a RREQ or RREP message, the following operations are performed.

i. A random number ‘s is generated called seed
ii. Value of the maximum hop count(MHC) field is set equal to time to leave value from IP header
iii. The value of s is stored in a field say hash.
iv. A Hash function is chosen, say HF
v. Another field top hash (TH) field is calculated as TH= HF^{MHC}(s), i.e., the hash function is applied to s exactly MHC times.

Now every time a node receive a RREQ or RREP from its neighbor node, it verify whether TH = = HF^{MHC} (hash).
HF is applied to hash before re-broadcasting a RREQ or forwarding a RREP message. All above mentioned fields are transmitted with the AODV messages in the signature extension so that intermediate node can verify the message using them.

2.2 SAODV Digital Signature

As mentioned earlier that SAODV use two way for performing verifying authentication of message. Therefore, signing and verifying mechanism by sender and receiver also differ up to some extent.

In first method, where only destination is allowed to reply, every time a RREQ is sent, the sender signs the message with its private key. An intermediate node verifies the signature before creating or updating the reverse path to the source and stores it only if verification is successful. For RREP message the final destination node sign the message using its private key. Intermediate and final node again verifies the signature before creating a route to that host.

In second method the signing and verifying process is almost similar to first one i.e. the sender signs the message with its private key and an intermediate node verifies the signature before creating or updating the reverse path to the source and stores it only if verification is successful. But the difference is that the RREQ message also has a second signature that is always stored with the reverse path route. The second signature is needed to be added in the gratuitous reply of that RREQ and in regular RREPs to future RREQs that node might reply as an intermediate node. An intermediate node that wants to reply a RREP needs not only the correct route, but also the signature corresponding to that route to add in the RREP and the lifetime and the originator IP address fields that work with that signature. All the nodes that receive the RREP and those update the route; store the signature, the lifetime and originator IP address with that route.

If a node want to have the feature of replying as an intermediate node for a route, it has to store the ‘RREQ Destination’ or ‘RREP Originator’ IP address, the lifetime and the signature. Since Hello messages of AODV are nothing but a reply messages, so they are signed and verified the same as mentioned above. Also every node generating or forwarding a RERR message uses digital signature to sign the whole message and is verified by the neighbor who receives it. SAODV does not take help of any extra message for security operations. Since a digital signature of any arbitrary node x can be created only by x using its private key, the SAODV mechanism prevents attacks like active forge, forged reply etc. using digital signature and prohibits malicious node from illegally modifying mutable fields like hop count. In our work we are more concerned about the performance of SAODV rather about securing mechanism. SAODV messages are significantly larger and require heavy computation time because of digital signatures.

3. PERFORMANCE ISSUES OF SAODV

As we mentioned earlier that SAODV extension protocol is the most successful secured protocol extension for AODV and already it has been proven better than AODV by [6] experimentally. It has been found that all securing proposal including SAODV consists of two kinds of techniques; one emphasizing on guaranteeing authenticity and integrity of routing messages and other to monitor the behavior of other nodes in routing operation. Both this techniques results in consumption of some additional resources of mobile ad hoc network like bandwidth, processing power etc. Considering constraints on limited resources of a mobile node in MANET the main issue of our concern is the trade-off between security and performance of secure AODV protocol. Though SAODV mechanism does not require any additional message in addition to routing messages of AODV, SAODV messages are significantly larger and require heavy computation time because of digital signatures especially for double signature mechanism. So, its performance may degrade significantly in heavy traffic scenarios of MANET.

4. ADAPTIVE SAODV (A-SAODV)

Cerri and Ghioni proposed an adaptive mechanism [5] that tunes its behavior for optimizing the performance of routing operation. They developed a prototype called Adaptive SAODV (A-SAODV) which is a multithreaded application. Cryptographic operations are performed by a dedicated thread to avoid blocking the processing of other message and other thread to all other functions.

The promising feature of A-SAODV which is called adaptive reply decision is to optimize SAODV performance with respect to double signature option. Allowing intermediate node to reply on behalf of destination node in AODV has a positive impact on its performance it do not require any additional computation. But, the case is different in SAODV as node may spend much time in computing these signatures and becomes overloaded. If only destinations are allowed reply then the performance becomes even worse than SAODV. This tends to make double signature mechanism adaptive i.e.

the intermediate nodes are allow to reply only if they are not overloaded.

Each node has a queue of routing messages to be signed or verified, and the length of this queue is used to check the current load state of the routing operations. When a node receives a RREQ message and has the information to generate a RREP on behalf of the destination, it checks the queue length and compares it with a threshold. If the queue length is lower than the threshold, the node generates a RREP; otherwise it forwards the RREQ without replying. Figure 1 shows this adaptive behavior of an intermediate node in A-SAODV. The same mechanism can be applied when generating a RREQ message in order to decide between a single signature and a double signature. In the simplest case, the threshold can be a fixed value; however, this value may be adjusted taking some external factors into account.

Experiments and simulation shows that Adaptive-SAODV is better than both variations [5] (single and double signature) of SAODV with respect to performance metrics like first data packet delay, number of successful connection etc. In our proposed work we have tried to further modify the adaptive behavior of an intermediate node to enhance its performance especially with first data packet delay metrics. The following section discusses our proposed work in detail.

```
receiveRREQ(Packet  rreq){
   if(isRouteExist(rreq.destination_address)
    &&!(rreq.destination_only_flag )){
          L= length(routing_packet_queue);
      if(L >= queue_threshold){
         for(each node n in neighbor list)
             forward(rreq) to n;
      }
      else
         generateRouteReply(rreq);
  }
…
```

Figure 1: A-SAODV algorithm

5. PROPOSED WORK

Our objective is to extend adaptive-SAODV with a modification in the behavior of an intermediate node using double signature mechanism. The proposed prototype intend to relax the overloading of a node with heavy cryptographic computations like signing and verifying routing packet up to a possible extent. The adaptive reply decision in A-SAODV depends mostly on the routing queue length of the current node which it uses to determine its load state. Our work further look for the load state of immediate neighbor of a current node which has fresh route to destination so that if it is found that the neighbor node is not overloaded then the replying job is left to it.

5.1 Modified Adaptive Reply Decision

In our proposed work, when an intermediate node that receives RREQ, finds that it has a fresh enough route to the destination and it is allowed to reply if it has them same, first it checks time to leave field (TTL) field of the packet, if its below some predefined time to leave threshold then the packet is simply forwarded to its neighbor nodes assuming that either the packet is going to be dropped after TTL hops or the packet going reach its destination with in this number of hops. When the above condition is not true then the node follows the steps of A-SAODV i.e. if the node has fresh route to destination and queue length is lower than the threshold, the node generates a RREP on behalf of destination node. If it is already over loaded with the job of singing or verifying of routing messages then the node do not simply forward as mentioned in A-SAODV rather it looks for its immediate neighbor that has a fresh route to destination. This can be easily found by looking at the next hop field of the fresh route entry to the destination in the routing table. Now the node checks for the load state of its neighbor in the path to the destination and if finds that the next hop neighbor node's routing packet queue length is less than the threshold value then it simply forward RREQ only to this neighboring node, otherwise, it again broadcast the route request message to all its neighbor since this condition shows that both the current node and the neighboring node in the path to destination are overloaded. Figure 2 shows the modification to behavior of an intermediate node in A-SAODV.

This modified adaptive reply mechanism has two advantages (i) relax the load of a node in term of signing and verifying task and (ii) reduces the traffic of the network by simply avoiding flooding (when a node in the path to destination has load state less then the threshold value).

```
receiveRREQ(Packet  rreq){
   if(isRouteExist(rreq.destination_address)
    &&!(rreq.destination_only_flag )){
      node_L= length(routing_packet_queue);
      if(rreq.ttl <= threshold_ttl){
         for(each node n in neighbor list)
            forward(rreq) to n;
      }
      elseif(node_L >= queue_threshold){
         rt_entry=lookup(rreq.destination_address);
         nbd_next= rt_entry.next_hop;
         if(nbd_next.queue_len <= queue_threshold)
            forward(rreq) to nbd_next;
         else{
            for(each node n in neighbor list)
               forward(rreq) to n;
         }
      }
      else
        generateRouteReply(rreq);
   }
…
```

Figure 2: Modified A-SAODV

5.2 Neighbors Load State Maintenance

Since our algorithm takes help of the load state of immediate neighborhood node for adaptive reply decision so it is necessary for a node to maintain the load state all the current immediate neighbors so that it can take the decision based on this. According to our proposed modification each node maintains an additional queue length field apart from its common routing information for all neighboring node. This field is associated with the information of each neighbors of a node in the routing table. One issue arises with this field is that how often we should update this load state field? The longer is update interval the lesser is freshness of the load state and this may lead to make an incorrect decision by an intermediate node when it receives a route request packet. On the other part shorter update interval may help each node to have fresh load status of each neighbor but more frequent information sharing may lead to increase in traffic overhead of the network. So to obtain a trade-off between these to extremes we have proposed to utilize the hello packet broadcast interval as the update interval for load state of neighbors. Each node may update and exchange their load state with their neighbors using hello message periodically. Since this information can be sent along with the hello messages, our modified prototype do not requires an additional message for this purpose.

5.3 Analysis of Proposed Algorithm

As we know that the time to leave (TTL) field is the number of hops to be traveled by the packet before being discarded by an arbitrary router. A small value of TTL say ‘t’, implies that either the packet going to reach its destination within t hops or going to be discarded after t hops. So, choosing a sufficiently small TTL value as TTL threshold field, any intermediate node is allow to reply a route request only if TTL field of the RREQ packet is larger than the TTL threshold value. Otherwise, the request packet is simply forwarded to all neighboring nodes assuming that either destination is within TTL threshold hop neighborhood of it or packet is to be dropped after TTL hops. This may significantly reduce the queue length of any intermediate node in the path to destination.

Secondly, in A-SAODV an intermediate node having a route to destination simply forward a route request for same without sending reply if it founds that its current routing message queue length is more than threshold queue length. If an intermediate node has a valid path to destination then among all the copy of forwarded packets to all neighboring nodes, the packet which has been forwarded to the next hop node of route entry for destination will follow the optimal path to destination. Our proposed modification is an additional checking to see that the whether next hop to the destination’s load factor is less than the threshold level. If yes, then the request packet is simply forwarded to next hop node instead of forwarding to all neighboring nodes. This may in turn reduce the delay associated with data packets and relax the load of all neighboring nodes which are not an active member of the optimal path to the destination.

6. SIMULATION AND RESULTS

In order to validate our analysis results, we have implemented all the misuses and performed a series of experiments through simulation. We have used ns2[7,8] network simulator version 2.33. Since the real performance of an intermediate node is more crucial in longer routes, we have tested the protocol under more critical conditions using a rectangular scenario of 1500 X 50 m, The network topology consists of 100 mobile nodes with each node establishing maximum 100 connections. Initially, the nodes are placed randomly in the grid. The random waypoint mobility model is used. The maximum node’s speed is kept at 20 m/sec with 0 pause time. Simulation time for each test is 200 seconds. We have used Constant Bit Rate (CBR) to generate UDP packets. CBR transmission rate is 4 packets/sec. Our prototype is implemented by modifying the original AODV source code in ns-2.

During each run we have measured first data packet delay and average throughput metrics of existing and proposed prototypes using different signing time. The figure 3 and 4 shows the comparison of our prototype with SAODV and Adaptive-SAODV protocols with respect to the first data packet delay metric. The average throughput of all three mechanisms has been shown in figure 5 and 6

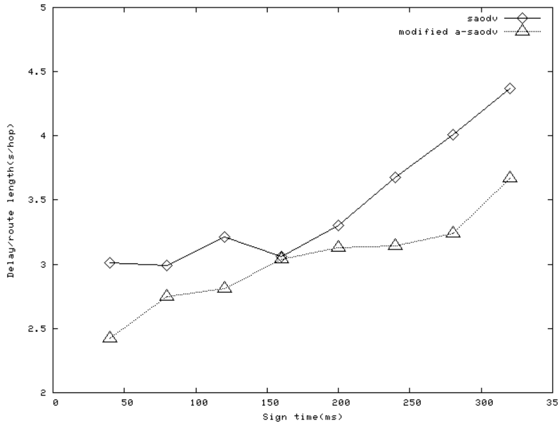

Figure 3: first data packet delay comparison between SAODV and modified A-SAODV.

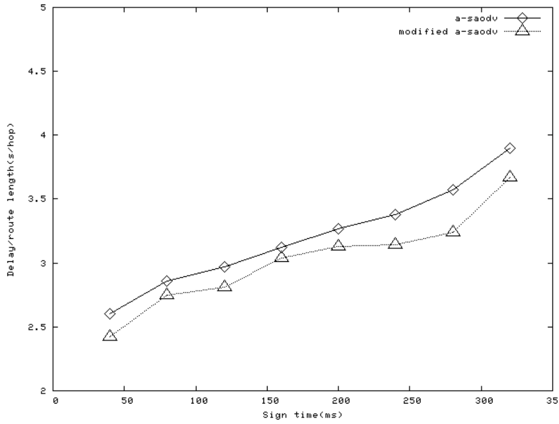

Figure 4: first data packet delay comparison between

A-SAODV and modified A-SAODV.

Although the improvement of our prototype is not significant because of other MANET constraints, the modified prototype behaves better than the other two, having shorter data packet delay and better throughput in the given scenario. From the simulation results we can say that our modification to adaptive reply decision of A-SAODV is contributing further improvement in the performance of SAODV. Other parameters, such as routing packet overhead and packet delivery fraction do not show significant differences between the three considered strategies.

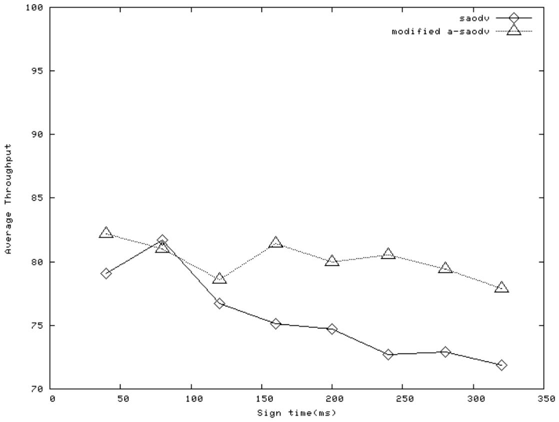

Figure 5: Average throughput comparison between SAODV and modified A-SAODV.

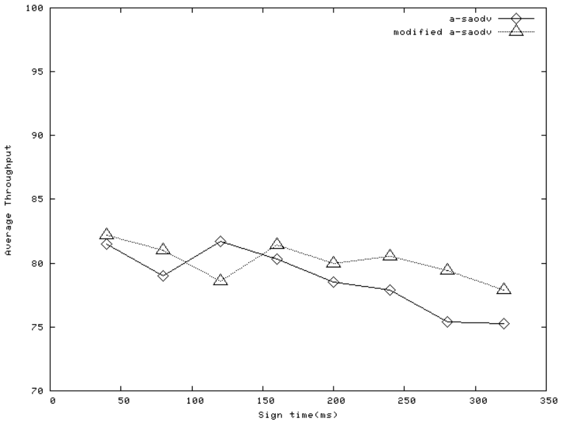

Figure 6: Average throughput comparison between A-SAODV and modified A-SAODV.

7. CONCLUSION AND FUTURE WORK

Securing AODV still an open area for research work. The existing mechanisms like SAODV able to secured the protocol with its signature extensions. But the overhead of cryptographic computation still persist in the SAODV mechanisms. A-SAODV is one of the steps towards optimizing the routing performance of secured protocols with help of a threshold mechanism. The adaptive reply decision by an intermediate node helps to balance the load of intermediate nodes which are over-burdened by signing and verification task of incoming messages. Our proposed extension to Adaptive-SAODV includes further filtering strategies aimed at further improving its network performance. The proposed mechanism has two advantages (i) relax the load of a node in term of signing and verifying task and (ii) reduces the traffic of the network by simply avoiding flooding (when a node in the path to destination has load state less then the threshold value).

We have analyzed and simulated our proposed algorithm to measure its ability in further improvement of performance with respect to reduce first data packet delay and also compared it with existing mechanisms using simulation. So, we can conclude that strength of a secured protocol for AODV not only depend on the strength of the cryptographic mechanism but also on the routing performance metrics.

The work is also open for a way to provide intermediate hop authenticity verification which still lack in existing literatures. To avoid the unnecessary flow of packet in the network one may also use selectively broadcasting instead of flooding. A mechanism for minimizing time involved in computation and verification of security fields will definitely boost the performance of AODV hence can be a nice work to proceed.

8. ACKNOWLEDGEMNT

This research work was supported by department of Computer Science and Engineering, NIT Rourkela, as a part of post graduation research 2008-2009 and being carried out at Advanced Database Laboratory, department of Computer Science and Engineering, NIT Rourkela.

9. REFERENCES

[1] Perkins C. E. and E. M. Royer, “Ad Hoc On-Demand Distance Vector Routing”, Proc. 2nd IEEE Wksp. Mobile comp. Sys. And Apps. Feb,1999, pp. 90-100

[2] Perkins C. E., E. M. Belding-Royer, and S. R. Das, “Ad Hoc On-Demand Distance Vector Routing”, IETF RFC 3561, July 2003

[3] Zapata M. G. and N. Asokan, “Securing Adhoc Routing Protocols”, Proceeding 1st ACM Workshop. Wireless Sec., 2002, pp. 1-10.

[4] Zapata M. G, “Secure Ad hoc On-Demand Distance Vector (SAODV) Routing, IETF Internet draft, September 2006, pp-1-22

[5] Cerri D., Alessandro Ghioni, “Securing AODV: The A-SAODV Secure Routing Prototype”, IEEE Communication Magazine, Volume 46, Issue 2, February 2008 pp.120 – 125

[6] Lin Y., A. Hamed Mohsenian Rad, Vincent W.S. Wong and Joo-Han Song, “Experimental Comparisons between SAODV and AODV Routing Protocols”, Proceedings of the 1st ACM workshop on Wireless multimedia networking and performance modeling, 2005, pp. 113 - 122

[7] http://www.isi.edu/nsnam/ns/ns-documentation.html.

[8] http://www-sop.inria.fr/maestro/personnel/ Eitan.Altman / COURS-NS/n3.pdf.

[9] Koblitz N. and Alfred J. Menezes, “A Survey of Public-Key Cryptosystems”, August 2004, pp. 1 – 47

[10] Menezes A.J., P.C. van Oorschot, and S.A. Vanstone, “Handbook of applied cryptography", CRC Press, 1997.