ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Performance Evaluation of Protocols for Secure Routing in MANETs

S.Ravi Kumar, M.Tech (NN)
Assistant Professor,
Department of CSE
Gokul Institute of Technology And Sciences,
Piridi, Bobbili, Vizianagaram, Andhrapradesh.

Inuganti N V A Pratyusha
M.Tech (CSE)
Department of CSE
Gokul Institute of Technology And Sciences,
Piridi, Bobbili, Vizianagaram, Andhrapradesh.

Abstract

Mobile Ad-hoc Networks (MANETs) can form a network by allowing the wireless nodes without any fixed infrastructure. For MANET's, early routing protocols are not accounted because of security issues. Further, cryptographic methods were revised to secure the routing information. These protocols had created new avenues for denial of service (DoS) as a part of the process. Thus, the trade-off between security strength and DoS vulnerability has become an emerging area, which required further investigation. Therefore, Different trust methods can be used for the protocol development at various levels during trade-off. To gain the knowledge on this, real world testing which evaluates the cost of existing proposals is mandatory. Without this, the protocol design in the future will be difficult. Thus, in the present paper, the comparison of two MANET routing protocols, namely SAODV and TAODV are carried. This addresses the routing security through cryptographic and trust-based means, and also the performance comparisons on actual resource-limited hardware. Based on this, the design decisions are evaluated for routing protocols in near future.

Keywords— ***ad-hoc, security, routing, trust-based, performance.***

I. INTRODUCTION

A base station or an access point is the main source for the communication between nodes in a network and the destinations outside the network. In general, MANETs can form the network without any fixed infrastructure [1]. These only require that nodes having interoperable radio hardware, which only use the same routing protocol to route traffic over the network. Thus MANET's became popular in all types of application areas, because of their ability in using small, resource-limited devices [2]. The best example is usage of MANET in battle field. As there is no requirement of fixed infrastructure, the traffic from one to another will be carried out by the nodes in the network which allows the communication within physical radio range [3]. Nodes by themselves should be able to change, over the network to forward data as individual nodes while moving around and acquire and lose the neighbors, i.e., nodes within radio range. Determination of the routing protocols is required to know how to forward the data as well as how to adapt to topology changes, resulting from mobility [4].

Initially, AODV (MANET) protocols were not designed to withstand the malicious nodes within network or outside attackers which are nearby with malicious intent [5]. Subsequent protocol and its extensions are proposed to address the issue of security. Many of these protocols used the cryptographic methods, in order to secure the information in the routing packets [6]. Observations are concluding, but an approach like this prevents tampering with the routing information. It also makes a very simple denial of service (DoS) attack [7]. This attack is very effective in MANETs as the devices have limited battery power along with the limited computational power. Consequently, this type of DoS attack allows the effective shutdown of nodes or otherwise it may disrupt the network. The trade-off between cryptographic security and DoS has become increasingly important, since each MANET

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

application is being developed such a way that it require a protocol with reasonable security and reasonable resistance to DoS. Therefore, suggestions are made for various trust mechanisms, where it could be used to develop new protocols with a unique security assurance at different levels in the trade-off [8].

The arguments for this are only purely theoretical or simulation-based. Determining the actual span of this trade-off in real world implementations, it is a paramount importance in directing future research and protocol design [9]. In the present context, two proposed protocol extensions are considered to secure MANET routing. The SAODV, which uses cryptographic methods to secure the routing information in the AODV protocol [10] and TAODV, which uses trust metrics for better routing decisions and to penalize the uncooperative nodes. Some applications accept SAODV's vulnerability to DoS or TAODV's weak preventative security. But most of them require an intermediate protocol tailored to the specific point on the DoS/security trade-off which fits in the application [11]. The tailored protocols for these applications also need performance, which falls in-between the SAODV and TAODV. Thus understanding how the performance of SAODV and TAODV protocols on real hardware and up to what extent a performance gap exists is a prerequisite to develop the intermediate protocols [12]. Such evaluation is not only required to develop intermediate protocols, but also to determine the direction for development of new trust metrics for ad-hoc network. In this paper we provide the first performance evaluations for these protocols on real world hardware.

II. EXPERIMENTAL SETUP

Since most promising applications of ad-hoc network's use small, resource constrained devices; a special attention is required for the trade-off between strong cryptographic security and DoS. The theoretical analysis or simulation may give hints based on the relative efficiency of different approaches. Only real world implementation and performance testing can give a clear idea of the actual width of this spectrum. Such measurements are required to provide the necessary information of protocols required for suitable or specific applications. In addition, the results can also be used to guide the design of novel protocols suited for particular deployment situations. In order to understand the real world performance of the AODV, SAODV, and TAODV protocols, the implementation has been carried on real hardware and their performance. In this section we detailed the experimental setup used to acquire these measurements. The supporting hardware and software setup for our implementations are discussed first, and then the actual implementations for each of the three protocols were discussed. Finally the design of the experiments used to evaluate the protocols is discussed with the explanation why these tests are more relevant than other more common metrics.

2.1 Hardware and Software Setup

For our testing we used the Sharp Zaurus SL-5500 model palmtops. The SL-5500 contains a 206MHz Intel Strong ARM processor, 64MB of DRAM, a 16MB Flash ROM, a 950 mAH lithium ion battery, Compact Flash and Secure Digital card slots. Each Zaurus was equipped with a LinksysWCF11 compact flash card for wireless communication. The Zauruses ran Open Zaurus v3.5.4 which is an embedded version of Linux. In order to compile programs for Zaurus, a cross- compiler tool chain based on GCC v3.3.4 is used. In addition, as given in Section 4.2, our code requires the Open SSL libraries. For this purpose, Open SSL v0.9.7j was cross- compiled and statically linked with the executables where necessary. All the cross-compiling are carried on a desktop running Slack-ware Linux 11.0.

2.2 Implementation

The AODV implementation is designed to run on a Linux operating system. Many AODV implementations for Linux separate the functionality into a kernel module and a user space daemon. The

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

kernel module use hooks in the net filter interface to send the packet headers from wireless interface to user space daemon. The daemon then determines the packet handling. If it is a routing control packet, the daemon processes the packet in accordance with the AODV specification or else if it is a data packet, the daemon determines the existence of route to the necessary destination. If a suitable route is found, the packet will be flagged and the kernel module queues it to send out. If no route exists, then daemon begins a search for route. Once a route is found, the daemon pushes the route into the kernels routing table. It then flags the packet to be queued for transmission. The implementation is carried out completely in C. The SAODV is implemented where there is necessary for a library of cryptographic operations. We have used Open SSL for this purpose and a security library is developed to wrap the most Open SSL's functionalities into the components appropriate for ad-hoc routing purposes.

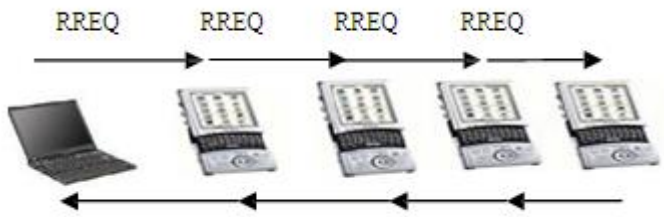

Fig. 1. Network setup for round trip timing tests.

One particularly useful feature of the security library is that it allows easy use of several different OpenSSL contexts at once. For SAODV, this was useful as nodes must switch between signing, verifying, and hash chain operations rapidly to both send and receive routing messages. New data structures are added for extension of SAODV's single signature and the necessary code was added to the message processing functions for RREQ, RREP, HELLO, and RERR messages. The design of the AODV implementation allowed SAODV functionality to be implemented while maintaining one binary with the ability to run both protocols.

Implementing TAODV required many additions similar to those involved in SAODV. New data structures were used for the NTT as well as the extended messages and the new R ACK message. Similarly, message handling functions were updated to use the extensions and take the appropriate actions. One challenge in implementing TAODV was counting packets sent, forwarded, or received for a particular route. While it intuitively seems to be something that should be implemented in the kernel module that is already tied into the net filter framework, this would require extra data exchange between the kernel module and the daemon.

Since our implementation already passes packet headers to the daemon for route discovery initiation and flagging, it was simply necessary to place the counting mechanism in the daemon.

The original implementation doesn't support any multi-path entries in routing table. Modifying it to support TAODV setup, rewriting significant amounts of the base AODV code is required. Instead, we have implemented a multi-path capable routing table used by the TAODV protocol. When a node initially finds a route, or changes the active route to destination, it merely copies the necessary entry to the daemon's local routing table and marks it as altered and updated in the kernel's routing table at the next sync. This simplifies the implementation using only a negligible amount of memory.

2.3 Testing Setup

Two performance factors are considered for the comparison. The first is the per-packet processing overhead; this is important to measure the CPU time. This overhead reflects the use of processor by each protocol. In these tests, AODV is considered as a baseline. Thus, for SAODV, the time for the generation of an SSE for RREQ, RREP and HELLO messages are estimated. Also measurement has been carried for the time of a node to verify an SSE for the same messages. For TAODV, measurement is carried out for a node to generate or process and update RREP

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

and R ACK messages. Due to the fact that some of the operations measured in this study has a runtime less than the resolution of the timer used (10ms as per the Linux kernel). A large number of operations are performed back-to-back per measurement and repeated for multiple times. Second performance metric is a round trip time for route discovery. The justification for metric lies at security of the routing control packets. Once a route is established, data will be forwarded with the same efficiency regardless of the routing protocol. Therefore, it is important to understand how the per-packet overhead along with the increased packet size affects the time for route discovery. For this, a measurement has been carried out to check the performance of AODV in addition to that of SAODV and TAODV. This is necessary since both AODV and TAODV will generate RREPs after fewer hops, when there is a response from destination's neighbor. SAODV requires response from the destination from itself. For our experiments, we have used a five node network consisting of one laptop and four Zauruses as illustrated in Figure 1. The network sniffer ethereal running on the laptop will measure the time elapsed from the sending of the RREQ to the receipt of the RREP. Individual measurements are performed repeatedly as explained.

TABLE 1
SAODV PER-PACKET OVERHEAD TIMES

Operation	Proc.Time (ms)	Std. Dev.
SSE generation	30.8	0.028
SSE validation	3.81	0.006

III. Results

Per-Packet Overhead

For the per-packet overhead tests, the amount of processing time of a node spends above and beyond that required for conventional AODV. All tests were performed with the Zauruses with only the necessary software running (i.e., no graphical login manager, no X server, etc.). In the SAODV tests, generation and validation of the SSE which requires hash computation and a digital signature/verification is computed. The hash function used for these tests are MD5 and the digital signature/verification with a 512-bit RSA key pair. Almost 1000 operations run per measurement and total 1000 measurements, overall. Table 1 shows the results carried for SAODV tests. Consequently, in order to send a RREQ, RREP, or HELLO message, the node spends 31.8 milliseconds generating the SSE. The significant impact on performance occurs while generating the SSE for HELLO messages as they are sent periodically. According to AODV specification, a node should send a "HELLO" message at every "HELLO INTERVAL" milliseconds unless; it has broadcast any messages during the previous interval. This means, only RREQ and RERR messages can prevent sending a HELLO message, as all other messages are unicast. Obviously, this can place significant burden on each node.

TABLE 2
TAODV PER-PACKET OVERHEAD TIMES

Operation	Proc. Time (ms)	Std. Dev.
RREP/HELLO send	0.0453	0.002
RREP/HELLO processing	0.0452	0.002
R ACK send	0.193	0.004
R ACK processing	0.297	0.005

Since SAODV requires that each message with a validated SSE, before any further processing. Each RREQ and RREP gets delayed 3.8 milliseconds at each hop which forwards it. In addition, HELLO messages take same amount of time, which is to be validated. While nodes are supposed to let ALLOWED HELLO LOSS *HELLO INTERVAL milliseconds pass before deciding when a link is broken and a neighbour should be removed from its routing table. It is conceivable that, on a node with several neighbors and a large amount of data to

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

forward, route status may fluctuate for some neighbors whose HELLO packets will get delayed while validation.

In TAODV, the per-packet overhead for RREP, HELLO, and R ACK messages are measured. The system- wide parameters discussed in the overhead of TAODV (not influenced) for any of the tests are also performed. However, it is necessary for anyone to fix these values to allow the calculation of RSV. For all TAODV tests, the following system-wide parameter values are considered: i = 0.8, p_ = 0.6, ph = 0.4, pc = 0.2, α1 = 0.4, α2 = 0.4, and α3 = 0.2. Due to the very small running time of the operations, one million operations are performed per measurement and total a number of 5000 measurements are obtained. Table 2 shows the results for the TAODV tests.

As the results show, there is much less per-packet overhead for TAODV when compared to SAODV. The main source of overhead involved the R ACK packets. Since the R ACK packets are new packets rather than packet extensions, it is necessary to allocate a packet buffer in the message sending to system during the implementation. Each time a R ACK packet is to be sent along with other messages that were extended. The packet buffer is allocated already and the extension is simply written into free space at the end. This difference contributed significantly to the 0.193ms overhead for sending the R ACK message. The overhead for processing the R ACK messages are completely due to the recalculation of the OTV and RSV values. The TAODV implementation used a double primitive for all calculations in order to keep protocol description. However, this affects the performance, as the SA- 1110 processor in the Zaurus has only a integer arithmetic unit. For systems with less computational power than Zaurus, suggests that it may be necessary to rewrite trust-based metrics into their equivalent using integer arithmetic instead.

Round Trip Results

The round trip tests for route discovery are performed for all the three protocols. This is important due to the differences in which node sends the RREP as described in Section 4.3. Due to the nature of the measurements, only one route discovery operation can be executed per measurement. Overall 5000 of these individual measurements are given in the Table 3.

TABLE 3
ROUND TRIP TIMES

Protocol	Round Trip Time (ms)	Std. Dev.
AODV	138.177	0.765
SAODV	324.732	7.22
TAODV	152.780	0.863

Table 3 shows the results of the tests and clearly indicates that SAODV is indeed a significantly expensive protocol. Specifically, SAODV takes 2.35 times as long as conventional AODV to get a RREP back to a RREQ originator. This is due, in part, to the added cryptography and increased message size. This is also due to the inability of intermediate nodes to respond to RREQs. Traversing the additional hop in both the directions adds latency. DSE is not implemented, because this has a large effect on the average route discovery. A destination now has to generate two digital signatures for a RREP. In addition, DSE only addresses the overhead incurred by intermediate nodes and it doesn't respond to RREQs. There is still overhead from the added cryptography and increased message size which implements DSE, which is not able to solve.

The results have also concluded that the use of SAODV needs adjustments for configurable parameters in AODV. This is missing from the current draft standard for SAODV. For example, the current suggested "NODE TRAVERSAL TIME" is 40ms which results in "NET TRAVERSAL TIME" being set to 1400ms. The value of "NET TRAVERSAL TIME" serves as the timeout for RREQ messages. Consequently, as per the results, if these parameters are not adjusted, nodes would have problems in

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

discovering the routes of length greater than seventeen hops. In some applications this may not cause problems. However, in certain applications such as large area sensor networks, routes of this length are not reasonable to expect. TAODV, on the other hand, takes only 1.11 times as long as AODV. This shows that the trust-based calculations and additional information exchange can be done without incurring the overhead of SAODV. While there is some expense for the trust calculations, as it is not nearly as expensive as the cryptographic operations. The results show that TAODV is indeed at the opposite end of the trade-off from SAODV. This is due to the fact that the TAODV information itself in each packet is not secured.

Overall, the results show that there is indeed a wide spectrum in the trade-off between cryptographic security and DoS. By adding an appropriate lightweight security mechanism for security of trust information in the routing packets, a hybrid protocol can be created which is less expensive than SAODV and more secure than TAODV. In Future, protocol designs should seek to use various new combinations of smarter, trust- based metrics and lightweight security mechanisms in order to develop hybrid protocols across the spectrum.

IV. CONCLUSIONS

In this paper, the performance of SAODV and TAODV protocols are tested for adhoc network routing security. The results of implementation and evaluation of both protocols are presented on a real resource-limited hardware. The expected differences between the two protocols are to be consistent with real world scenario. Also, these experiments showed that there is significant room between the two protocols for a secure hybrid protocol to be developed with an advantage of the strongest points of both. Future work needs to delve further into the extensive body of work on various trust metrics. This includes the testing of other trust metrics for use in ad-hoc routing as well as developing the aforementioned hybrid protocols, by testing their performance against the results presented in this paper. In addition, it is necessary to test the quality of the routing decisions produced by protocols in a malicious environment.

REFERENCES

[1] C. N.-R. Baruch Awerbuch, David Holmer and H. Rubens. An on-demand secure routing protocol resilient to byzantine failures. In ACM Workshop on Wireless Security (WiSe), September 2002.

[2] S. Buchegger and J.-Y. L. Boudec. Nodes Bearing Grudges, Towards Routing Security, Fairness, and Robustness in Mobile Ad Hoc Networks. In Proceedings of the Tenth Euromicro Workshop on Parallel, Distributed and Network-based Processing. IEEE Computer Society, January 2002.

[3] H. Deng. Routing security in wireless ad hoc networks, 2002.

[4] P. Dewan and P. Dasgupta. Trusting routers and relays in ad hoc networks. In ICPPW ’03, Proceedings of the 2003 International Conference on Parallel Processing Workshops, pages 351–358, 2003.

[5] L. Eschenauer, V. Gligor, and J. Baras, On trust establishment in mobile ad-hoc networks. Technical Report MS 2002-10, Institute for Systems Research, University of Maryland, MD, USA, October 2002.

[6] Ethereal - A Network Protocol Analyzer. http://www.ethereal.com/.

[7] T. Ghosh, N. Pissinou, and K. Makki. Collaborative trust- based secure routing against colluding malicious nodes in multi-hop ad hoc networks. In LCN ’04: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN’04). IEEE Computer Society, 2004

[8] Y. Hu, D. Johnson, and A. Perrig. SEAD, Secure efficient distance vector routing for mobile wireless ad hoc networks. Ad Hoc Networks, I:175– 192, 2003.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[9] Y. Hu, A. Perrig, and D. Johnson. Packet leashes, A defense against wormhole attacks in wireless ad hoc networks. Technical report, Department of Computer Science, Rice University, December 2001.

[10] N. Pissinou, T. Ghosh and K. Makki, Collaborative trust-based serure routing in multihop ad hoc netwroks. In NETWORKING 2004, Networking Technologies, Services, and Protocols, Performance of Computer and Communications Networks; Mobile and Wireless Communications, 2004.

[11] A. Perring, R. Szewczyk, V. Wen, D. E. Culler, and J.D. Tyar. SPINS , security protocols for sensor networks. In Mobile Computing and Networking, 2001.

[12] A. Pirzada and C. McDonald, Establishing trust in pure ad-hoc networks. In Twenty-Seventh Australasian Computer Science Conference (ACSC2004), 2004.