IJCSMS International Journal of Computer Science and Management Studies, Vol. 12, Issue 03, Sept 2012
ISSN (Online): 2231-5268
www.ijcsms.com

An Implementation of Advanced Traffic Control Techniques in MANET

Babita[1], Dr. Nasib Singh Gill[2]

[1]M.tech Student, DCSA, MDU, Rohtak, Haryana, India
babymor900@gmail.com

[2]Professor, DCSA, MDU, Rohtak, Haryana, India
nasibsgill@gmail.com

Abstract

Mobile Ad hoc Networks (MANET) has become an exciting and important technology in recent years because of the rapid proliferation of wireless devices. A mobile ad-hoc network (MANET) is a self-configuring network of mobile routers (and associated hosts) connected by wireless links - the union of which form an arbitrary topology. Secured ad hoc routing protocols are a necessity for securing the routing of data. To have security in the routing, one should sacrifice the performance of the data transmission. This shows that in the secure routing protocols, the usage of security techniques like digital signatures, authentications and hash chains have major impacts on the performance since it will use more processing power and time. Secure routing protocols available today (such as SAODV) still need further optimizations to minimize the processing overhead, delays and to maximize the routing throughputs.

Keyword: MANET, Traffic Control, RREQ, AODV, SAODV.

Introduction

A mobile ad hoc network (MANET) is a dynamic distributed system of wireless nodes that move independently of each other. The operating transmission range of the nodes is limited and as a result, MANET routes are often multi-hop in nature. Ad hoc networks are emerging as the next generation of networks and defined as a collection of mobile nodes forming a temporary (spontaneous) network without the aid of any centralized administration or standard support services MANETs are usually set up in situations of emergency for temporary operations or simply if there are no resources to set up elaborate networks. These types of networks operate in the absence of any fixed infrastructure, which makes them easy to deploy, at the same time however, due to the absence of any fixed infrastructure, it becomes difficult to make use of the existing routing techniques for network services, and this poses a number of challenges in ensuring the security of the communication.

Ad Hoc on Demand Distance Vector (AODV)

AODV is perhaps the most well-known routing protocol for a MANET. It is a reactive protocol: nodes in the network exchange routing information only when a communication must take place and keep this information up-to-date only as long as the communication lasts. The Ad-hoc On Demand Distance Vector (AODV) classified under reactive protocols.

A third kind of routing message, called route error (RERR), allows nodes to notify errors, for example, because a previous neighbor has moved and is no longer reachable. If the route is not active (i.e., there is no data traffic flowing through it), all routing information expires after a timeout and is removed from the routing table.

AODV routing was created without taking security into major concern, which it should be the most important factor to be looked at. The analysis on the security threats that have been have been made to describe the requirements for AODV routing protocol to mitigate threats. A node is malicious if it is an attacker that cannot authenticate itself as a legitimate node due to the lack of valid cryptographic information. A node is compromised if it is an inside attacker who is behaving maliciously but can be authenticated by the network as a legitimate node and is being trusted by other nodes.

Attack can be occurred against the AODV routing protocol

Message replay (or wormhole) attack: Attackers can retransmit eavesdropped messages again later in a different place. One type of replay attacks is the wormhole attack. A wormhole attacker can tunnel an RREQ directly to a destination node. Since a wormhole attacker may not increase the hop-count field value, it prevents any other routes from being discovered. The wormhole attack can be combined with the message dropping attack to prevent the destination node from receiving packets.
Message tampering attack: An attacker can alter the content of routing messages and forward them with falsified information. For example, by reducing the hop-count field in either an RREQ or RREP packet, an attacker can increase its chance to be an intermediate node of the route. A selfish node can relieve the burden of forwarding messages for others by setting the hop-count field of the RREQ to infinity.

The security requirements for AODV routing protocol include:

1) Source authentication: The receiver should be able to confirm that the identity of the source is indeed who or what it claims to be Performance Comparisons of AODV, Secure AODV and Adaptive Secure AODV Routing Protocols in Free Attack Simulation Environment.
2) Neighbor authentication: The receiver should be able to confirm that the identity of the sender (i.e., one hop previous node) is indeed who or what it claims to be.
3) Message integrity: The receiver should be able to verify that the content of a message has not been altered either maliciously or accidentally in transit.
4) Access control: It is necessary to ensure that mobile nodes seeking to gain access to the network have the appropriate access rights. There are a number of secure protocols proposed especially for AODV to mitigate the attacks. The approaches included in this paper are SAODV (Secure AODV) and Adaptive SAODV.

Secure AODV (SAODV)

Secure AODV (SAODV) is a security extension of the AODV protocol, based on public key cryptography. SAODV routing messages (RREQs, RREPs, and RERRs) are digitally signed to guarantee their integrity and authenticity. Therefore, a node that generates a routing message signs it with its private key, and the nodes that receive this message verify the signature using the sender's public key.

Implementation Design Possibilities

Possible opportunities for obtaining the said events include
• Snooping
• Netfilter
• Kernel Modification

Snooping

In order to determine the needed events is to promiscuously snoop all incoming and outgoing packets. The code to perform snooping is built into the kernel and is available to user space programs. For e.g. An ARP packet is generated when a node does not know the MAC layer address of the next hop. Using this interface, if an ARP request packet is seen for an unknown destination and it is originated by the local host, then a route discovery needs to be initiated. Similarly, all other AODV events may be determined by monitoring incoming and outgoing packets. The most important advantage of this solution is it does not require any code to run in the kernel space. Hence it allows for simple installation and execution .But two disadvantages are overhead and dependence over ARP.

Netfilter

Netfilter is a set of hooks at a various points inside the Linux protocol stack. Netfilter redirects packet flow through user defined code, which can examine, drop, discard, modify or queue the packets for user space daemon. Using Netfilter is similar to snooping method however it does not have the disadvantage of unnecessary overhead or dependence on ARP. This solution has the strength such as there is no unnecessary communication; it is highly portable, it is easy to install and user space daemon can determine all the required events. On the other hand, the disadvantage of this solution is that it requires a kernel module. However kernel module is easier than kernel modifications. A kernel module is more portable than kernel modifications because it depends only on the Netfilter interface. This interface does not change from one kernel version to next.

Kernel Modification

In order to determine the AODV events is to modify the kernel. Code can be placed in the kernel to

communicate the events to an AODV user-space daemon. For example, to initiate route discovery, code is added in the kernel at the point where route lookup failures occur. Given this code in the kernel, if a route lookup failure happens, then a method is called in the user-space daemon.

Challenges

One of fundamental vulnerability of MANETs comes from their open peer-to-peer architecture.
Unlike wired networks that have dedicated routers, each mobile node in an ad hoc network may function as a router and forward packets for other nodes. The wireless channel is accessible to both legitimate network users and malicious attackers. As a result, there is no clear line of defense in MANETs from the security design perspective. The boundary that separates the inside network from the outside world becomes blurred. There is no well defined place/infrastructure where we may deploy a single security solution.
The above characteristics of MANETs clearly make a case for building multifence security solutions that achieve both broad protection and desirable network performance. First, the security solution should spread across many individual components and rely on their collective protection power to secure the entire network. The security scheme adopted by each device has to work within its own resource limitations in terms of computation capability, memory, communication capacity, and energy supply. Second, the security solution should span different layers of the protocol stack, with each layer contributing to a line of defense. No single-layer solution is possible to thwart all potential attacks. Third, the security solution should thwart threats from both outsiders who launch attacks on the wireless channel and network topology, and insiders who sneak into the system through compromised devices and gain access to certain system knowledge. Fourth, the security solution should encompass all three components of prevention, detection, and reaction that work in concert to guard the system from collapse. Last but not least, the security solution should be practical and affordable in a highly dynamic and resource constrained networking scenario.

A MULTIFENCE Security Solution

In this, the state-of-the-art security proposals for MANETs because multihop connectivity is provided in MANETs through distributed protocols in both the network and link layers, the ultimate multifence security solution naturally spans both layers. There are basically two approaches to securing a MANET: proactive and reactive. The proactive approach attempts to thwart security threats in the first place, typically through various cryptographic techniques. On the other hand, the reactive approach seeks to detect threats *a posteriori* and react accordingly.

NETWORK-LAYER SECURITY

The network-layer security designs for MANETs are concerned with protecting the network functionality to deliver packets between mobile nodes through multi hop ad hoc forwarding.
Therefore, they seek to ensure that the routing message exchanged between nodes is consistent with the protocol specification, and the packet forwarding behavior of each node is consistent with its routing states. Accordingly, the existing proposals can be classified into two categories: secure ad hoc routing protocols and secure packet forwarding protocols. Before we describe these security solutions in detail, we first introduce several cryptographic primitives for message authentication, the essential component in any security design, and analyze the trade-offs behind them.

Proposed System

We assume that cryptographic operations are performed by a dedicated thread to avoid blocking the processing of other messages. Therefore, there are two execution threads: one dedicated to cryptographic operations and the other to all other functions (routing message processing, routing table management, timeout management, message generation, and data packet forwarding).
The two threads communicate via a first input first output (FIFO) queue containing all the messages that must be signed or verified. The prototype developed includes an experimental feature, the adaptive reply decision, to respect to the double signature option. In AODV, allowing intermediate nodes to generate RREPs on behalf of the destination node has a positive impact on performance, because it does not require heavyweight operations by intermediate nodes themselves.
The situation is different in SAODV, because generating such a reply requires the intermediate node to generate a cryptographic signature: nodes may spend much time in computing these signatures and become overloaded.

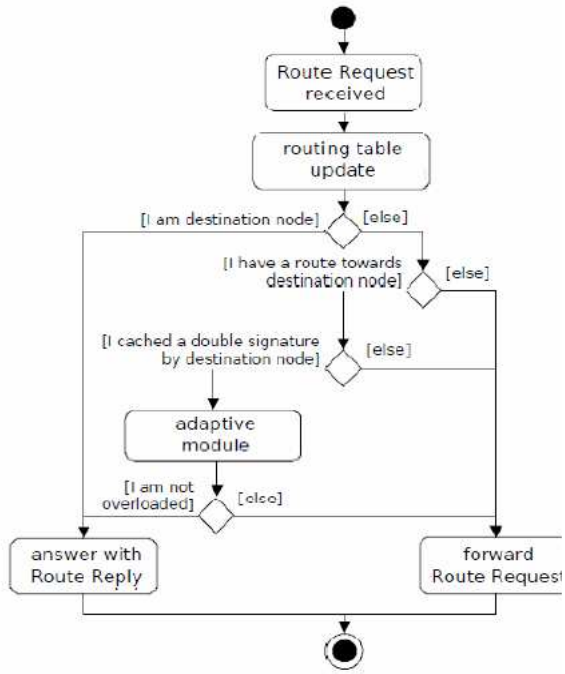

Figure 1.Route Request and Route Reply Processing

Moreover, if intermediate nodes have a long queue of routing messages that must be cryptographically processed, the resulting delay may be longer than if the request reaches the destination node. If the double signature mechanism removed [3], an un-collaborative protocol created, in which only the destination node is allowed to reply to a RREQ message. This is possible, where the simulation results show that if signing time is low, and routes are not very short, performance is worse than SAODV with double signatures. Therefore, the proposed approach makes the double signature feature adaptive: intermediate nodes reply to RREQs 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 (with different weights for signature operations and verification operations) can be used to evaluate the current load state of the routing daemon.
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 (collaborative behavior); otherwise it forwards the RREQ without replying (un-collaborative behavior).

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 would not be very flexible because the value maybe adjustable, depending on external factors (e.g., battery state). In this prototype, the threshold value can be changed during execution (two special values allow always *reply* behavior and *never reply* behavior). Other, more elaborate strategies could be defined to estimate the crypto queue delay and consequently decide whether to reply or forward the message. For example, a fixed threshold (based on the timeouts defined by the routing protocol) and a predictor of queuing times could be used.
In this way, the algorithm could adapt itself to the situation and the computing power of the node. An additional external parameter could be used to take into account the previously mentioned external factors (how much a node is willing to collaborate, e.g., depending on its battery state).Another little optimization included in this prototype is a cache of latest signed and verified messages, in order to avoid signing or verifying the same message twice.

Simulations and Results

Tool Used:

Ns-2 is an open source discrete event simulator used by the research community for research in networking. It has support for both wired and wireless networks and can simulate several network protocols such as TCP, UDP, multicast routing, etc. More recently, support has been added for simulation of large satellite and ad hoc wireless networks. The Ns-2 simulation software was developed at the University of Berkeley. It is constantly under development by an active community of researchers.

The standard Ns-2 distribution runs on Linux. However, a package for running Ns2 on Cygwin (Linux Emulation for Windows) is available.

NS uses two languages because simulator has two different kinds of things it needs to do. On one hand, detailed simulations of protocols require a systems programming language which can efficiently manipulate bytes, packet headers, and implement algorithms that run over large data sets. For these tasks run-time speed is important and turn-around time (simulation, find bug, fix bug, recompile, re-run) is less important.

Implementation Parameters for MANETs

- Number of Nodes = 50 nodes
- Maximum connections = 40 traffic sources
- Mobility Model = Random Waypoint
- Mobility Speed = 40 m/s
- Rate = 8kbps (2 packets per load)
- Topology Size = 500m x 500m
- Channel =Wireless
- Propagation = Two ray
- MAC = 802.11
- Queue = Drop tail/ pri queue
- Antenna = Omni directional
- Routing Protocol = AODV
- Simulation Time 100s

Simulation Results

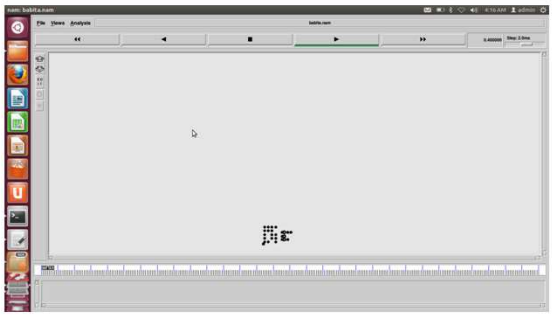

Figure 2.Given Node

Inputs that are given are the node id, number of mobile nodes and the output is the simulation of graphs giving the degradation of the performance based on packet delivery ratio, throughput.

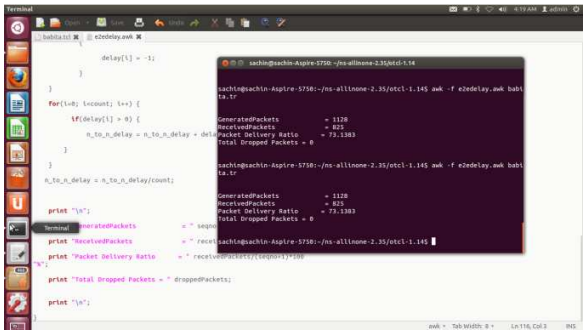

Figure 3 Performance Of Packet Delivered Ratio

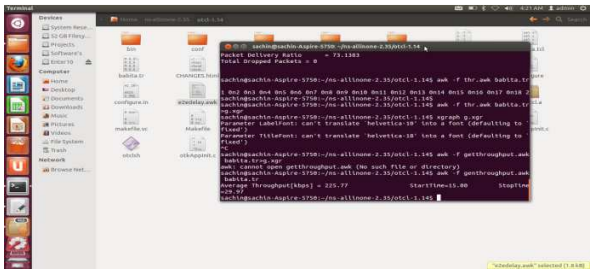

Figure 4 Performance of Throughput

Conclusion

Roughly, SAODV and Proposed performs such a like AODV with lower number of nodes and less mobility but when moving to more higher number of nodes and increased mobility, it tends to break down. Secured ad hoc routing protocols are a necessity for securing the routing of data. To have security in the routing, one should sacrifice the performance of the data transmission. This shows that in the secure routing protocols, the usage of security techniques like digital signatures, authentications and hash chains have major impacts on the performance since it will use more processing power and time. Secure routing protocols available today (such as SAODV) still need further optimizations to minimize the processing overhead, delays and to maximize the routing throughputs.

References

[1] M. F. Juwad, and H. S. Al-Raweshidy, "Experimental Performance Comparisons between SAODV & AODV", IEEE Second Asia International Conference on Modelling& Simulation, 2008

[2] JunaidArshad and Mohammad Ajmal Azad, "Performance Evaluation of Secure on-Demand Routing Protocols for Mobile Ad-hoc Networks", 1-4244-0626-9/06 © 2006 IEEE

[3] DavideCerri and Alessandro Ghioni, "Securing AODV: The A-SAODV Secure Routing Prototype", 0163-6804/08 © 2008 IEEE, IEEE Communications Magazine, February 2008

[4] Yuxia Lin, A. HamedMohsenian Rad, Vincent W.S. Wong, and Joo-Han Song, "Experimental Comparisons between SAODV and AODV Routing Protocols", WMuNeP'05, October 13, 2005, ACM

[5] Stephan Eichler and Christian Roman, "Challenges of Secure Routing in MANETs: A Simulative Approach using AODV-SEC", 1-4244-0507-6/06 © 2006 IEEE

[6] J. Martin Leo Manickam, R. Bhuvaneswari, M.A. Bhagyaveni and S.Shanmugavel, "Secure Routing Protocol for Mobile Ad-Hoc Networks", ISBN # 1-56555-316-0, SCSC 2007

[7] N.Shanthi and Dr. L.Ganesan, "Security In Multicast Mobile Ad-Hoc Networks", IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.7, July 2008

[8] Elizabeth M. Belding-Royer and Charles E. Perkins, "Evolution and future directions of the ad hoc on-demand distance-vector routing protocol", doi:10.1016/S1570-8705(03)00016-7@ 2003 Elsevier

[9] SonaliBhargava and Dharma P. Agrawal, "Security Enhancements in AODV protocol for Wireless Ad Hoc Networks", 0-7803-7005-8/01 © 2001 IEEE

[10] PengNing and Kun Sun, "How to Misuse AODV: A Case Study of Insider Attacks against Mobile Ad-Hoc Routing Protocols", Proceedings of 2003 IEEE Systems, Man and Cybernetics Society Information Assurance Workshop, pages 60–67, June 18-20, 2003

[11] Chin-Yang Tseng, Poornima Balasubramanyam, Calvin Ko, Rattapon Limprasittiporn, Jeff Rowe, Karl Levitt, "A Specification-based Intrusion Detection System for AODV" Computer Security Laboratory University of California, Davis and 2Network Associates Laboratories Network Associates, Inc.

[12] Jane Zhen and SampalliSrinivas, "Preventing Replay Attacks for Secure Routing in Ad Hoc Networks", Dalhousie University, Halifax, NS, Canada, Springer-Verlag Berlin Heidelberg 2003, ADHOC-NOW 2003, LNCS 2865, pp. 140–150, 2003.

[13] Gergely ´Acs, LeventeButty´an, and Istv´anVajda, "Provable Security of AODV Routing in Wireless Networks", Laboratory of Cryptography and Systems Security (CrySyS), Department of Telecommunications, Budapest University of Technology and Economics, Hungary, Springer-Verlag Berlin Heidelberg 2005, ESAS 2005, LNCS 3813, pp. 113–127, 2005.

[14] Hoda M. Hassan, Mohy Mahmoud, Sherif El-Kassas, "Securing the AODV Protocol Using Specification-Based Intrusion Detection", Q2SWinet'06, October 2, 2006, Torremolinos, Malaga, Spain, Copyright 2006 ACM 1-59593-486-3/06/0010.

[15] Patroklos G. Argyroudis, DonalO'Mahony, "Secure Routing for Mobile Ad hoc Networks", Department of Computer Science University of Dublin.

[16] Woei-JiunnTsaur, Haw-TyngPai, "A New Security Scheme for On-Demand Source Routing in Mobile Ad Hoc Networks", IWCMC'07, August 12–16, 2007, Honolulu, Hawaii, USA, Copyright 2007 ACM 978-1-59593-695-0/07/0008.

[17] Seung Yi, Prasad Naldurg, Robin Kravets, "A Security-Aware Routing Protocol for Wireless Ad Hoc Networks", Dept. of Computer Science, University of Illinois at Urbana-Champaign Urbana, Illinois.

[18] Satoshi Kurosawa, Hidehisa Nakayama, Nei Kato, Abbas Jamalipour, and Yoshiaki Nemoto, "Detecting Blackhole Attack on AODV-based Mobile Ad Hoc Networks by Dynamic Learning Method", International Journal of Network Security, Vol.5, No.3, PP.338–346, Nov. 2007.

[19] M. Guerrero Zapata, "Key Management and Delayed Verification for Ad Hoc Networks", J. High Speed Networks, vol. 15, no. 1, Jan. 2006, pp. 93–109.

[20] Manel Guerrero Zapata, "Secure Ad hoc On Demand Distance Vector (SAODV) Routing", Technical University of Catalonia (UPC), Mobile Ad Hoc Networking Working Group, Internet Draft, 15 September 2005.

[21] Manel Guerrero Zapata, N. Asokan, "Securing Ad Hoc Routing Protocols", WiSe'02, September 28, 2002, Atlanta, Georgia, USA, ACM 1-58113-585-8/02/0009 Copyright 2002.

[22] The NS Manual, http://www.isi.edu/nsnam/ns.

[23] The Network Simulator NS-2 tutorial homepage, http://www.isi.edu/nsnam/ns/ tutorial/index.html