Enhanced Intrusion Detection System for DiscoveringMalicious Nodes in Mobile Ad hoc Networks

Dr.B. Prabhakar Reddy, P. Imran Khan, S. Mahaboob Basha

Professor[1], Assistant Professor[2,3]
1,2 Bheema Institute of Technology and Sciences, Adoni-518301.

[3]Geethanjali College of Engineering and Technology, Kurnool.

Abstract **–** Differentiating mobile wireless ad hoc networks from wired networks and even from conventional wireless networks presents new problems in terms of security. Many intrusion detection systems, such Watchdog/Pathrater and Routeguard, have been developed. These systems have close ties to routing protocols. Watchdog is the intrusion detection component, while Pathrater and Routeguard handle the reaction. Each node has its own overhearing watchdog system. Each node may overhear its neighbors' harmful behavior and report them. The speed of the network may be severely impacted, though, if the node that is listening in and reporting itself is malicious. In this article, we discuss how we overcame Watchdog's flaws and introduced our ExWatchdog intrusion detection solution. Discovering rogue nodes that potentially split the network by fraudulently reporting other nodes as misbehaving is the major aspect of the proposed system, and the system then continues to secure the network. Results from simulations demonstrate that our method significantly reduces overhead without clearly increasing throughput.

I. Introduction

In a Mobile Ad hoc Network (MANET), a collection of wireless mobile hosts (nodes) works together to build a temporary, self-configuring network that doesn't rely on a single server. The mobile nodes may be anything from a cell phone to a laptop computer, and they usually have many wireless connection options (802.11, IrDA, Bluetooth, etc.).

Wireless networks allow users to share data while still being mobile, which is a major benefit. However, the distance between peers is limited by the range of transmitters or their closeness to wireless central points. Each node in a mobile ad hoc network sends its own data and also routes and passes data on behalf of other nodes, which helps to alleviate the issue of nodes being out of range.

The MANET may function alone or as part of the wider Internet. Ad hoc networks are useful in situations when a permanent infrastructure cannot be set up immediately, such as during a natural or man-made catastrophe, a military battle, or a terrorist attack. cases of extreme medical urgency.

There have been major advancements in the areas of routing protocols, clustering protocols, position and mobility prediction, and other aspects and themes relevant to MANET. However, MANET's security features are seldom discussed. Accessibility, integrity, authentication, privacy, and resistance to forgery are all important parts of MANET security [9].

Secure communication between mobile nodes in an unfriendly environment is a major issue for ad hoc networks. Due to their dynamic nature, mobile ad hoc networks provide a number of difficulties in terms of security architecture. Among them include a highly adaptable topology, wireless communication, and decentralization. The primary issue with MANET security is this final one: users and malevolent attackers alike have easy access to the ad hoc networks. A mobile ad hoc network can readily exploited or even disabled if a hostile attacker gains access to the network. As the existence of a Certificate Authority or a key Distribution Center cannot be taken for granted, traditional means of identification and authentication are unavailable.

We present ExWatchdog, an intrusion detection system built on top of the original Watchdog framework provided in [3]. We designed our approach to address the following Watchdog-specific problems: When a malicious node is at fault, it falsely accuses other nodes of misbehavior. The total number of packets sent, forwarded, and received by each node is tracked in a database kept by each node. The node that is receiving the information about the misbehaving nodes might then send a message to the node it is communicating with to see whether their packet totals are equal. If both are equivalent, the malevolent node is the one that falsely accuses other nodes of wrongdoing. In any case, the observed malicious nodes indeed act badly.

Here is how the rest of the paper is structured. In Section II, we give the necessary context via a review of the relevant literature and previous research. In Section III, we discuss our ExWatchdog intrusion detection and response system. In Chapter Four, we detail the simulation results and discussion. Finally, conclusions drawnfrom the paper and future work are given in Section V.

II. Background and Related Work

In this part, we introduce several key ideas that will help set the stage for the rest of the article. First, a quick primer on MANET routing protocols. We then go on to detail the ways in which MANET's intrusion detection and response systems are distinct from those of more conventional networks like the Internet. We wrap off with a brief discussion of Watchdog, Pathrater, and Routeguard, three proposed intrusion detection and response systems. ExWatchdog is an intrusion detection system that builds on Watchdog's foundation by requiring a response system, such as Pathrater or Routeguard, to prevent attacks.

Ad hoc network routing protocols, type A

Due to the restricted wireless transmission range, it is common for pathways between source nodes and destination nodes in mobile ad hoc networks to have many hops. In this setup, every node may function as a router, passing data packets along from their origin to their final destination. The mobility of the nodes also necessitates periodic rerouting of established connections. In order to keep services for ongoing sessions to a minimum, a MANET routing protocol has to be able to identify and react promptly to such changes in state.

In a MANET, you may choose from a wide variety of routing protocols. Based on when routes are chosen, they may be broken down into two broad categories: proactive and reactive. To ensure that nodes with packet-sending responsibilities always have available pathways, proactive routing continually formulates routing decisions [6]. As soon as a node has a packet to send, it will query the network for a route using reactive routing [2, 4, 5, 7, 8].

DSR [2] stands for the Dynamic Source Routing Protocol, and it is a protocol for routing sources dynamically. When a packet is sent from a source to a destination for which the source does not yet have a path, the path to that destination is found "on-demand." Route Discovery and Route Maintenance are DSR's two main stages of operation. Ad hoc network nodes seek for potential routes to their destination in a cache table before actually sending the data packet. If the route in the cache has expired or does not exist, a procedure to find a new route is initiated. DSR's route maintenance is its second primary use. When forwarding a packet down a route, an intermediate node may detect whether a connection has been severed and alert the source node. The node that discovered the broken connection deletes all instances of the route or only the part of the route that relied on the link. The source must also try another path or do a new route discovery if it does not have another path.

A. Intrusion Detection and Response Systems

There must be innate and observable aspects of normal conduct that can be gathered and evaluated, and those qualities must be capable of being used to differentiate between normal and aberrant behavior for intrusion detection to be viable.

When it comes to spotting and stopping intrusion attempts, the majority of conventional IDSs either focus on the network or the host. Network-based IDS acts as an ear on the network, monitoring traffic in order to inspect each packet as it travels through. Host-based systems care only about the state of their own hosts.

On the other hand, MANET has its own unique characteristics that make host-based and network-based IDS inappropriate. Thus, it has been suggested for MANET IDS to operate in tandem with and as an integral component of the preexisting routing protocols.

To meet the requirements of mobile ad hoc networks, intrusion detection and response systems in MANET should be both dispersed and cooperative. According to the design put out in [10], all of the mobile ad hoc network's nodes take part in detecting and responding to intrusions. Each node must individually monitor its immediate vicinity for any indications of intrusion since it can trust none of its neighbors. However, in the event of a suspicious circumstance or a verified intrusion detection, surrounding nodes might work together to exchange messages.

B. Related Works

In order to increase performance in mobile ad hoc networks when hacked or malfunctioning nodes accept to forward messages but are unable to do so, Marti et al. [3] suggested two approaches (Watchdog and Pathrater). In mobile ad hoc networks [2], combining the DSR protocol with the Watchdog/Pathrater system significantly improved throughput.

Each node on the path from the source to the destination takes part in intrusion detection and response with the help of DSR by monitoring the node directly downstream from it to ensure that it has not tampered with the packet in any way before retransmitting it. This downstream node is acting badly if it does not forward the message. To lessen the impact of a bad node, Marti et al. offer Pathrater, an algorithm that uses ratings rather than distance to choose the best route between two points. Every computer in the system is running Pathrater. In [3], the benefits and drawbacks of the Watchdog approach are examined. By including the Watchdog, DSR is able to identify abnormal activity not only at the link level, but also at the forwarding level. The flaws in Watchdog include ambiguous collisions, receiver collisions, low transmission power, fake misbehavior, collusion, and partial dropping are all scenarios in which a misbehaving node can go undetected. Hasswa et al. [1] also talks about the problems with Pathrater. Five key flaws may be identified in the rating system itself: (1) rigidly defined states; (2) vulnerability to behavioral deception; (3) the invisibility of new nodes; (4) the potential for a previously hostile node to re-enter the system; and (5) the promotion of self-interest and greed.

(Second) Routeguard (1: Each network node operates its own copy of Routeguard, which is similar to the Pathrater. Every node keeps an evaluation of every other node it is connected to. However, Routeguard refines Pathrater by rating nodes and calculating a route measure more precisely. With Routeguard, you can categorize your network's nodes more precisely and naturally, into one of five groups: new, member, unstable, suspect, or malicious. The nodes are handled differently based on their ratings and status.

III. EXWATCHDOG INTRUSION DETECTION SYSTEM

The ExWatchdog intrusion detection and response system is discussed in this article. Similarly to Watchdog, ExWatchdog is able to report malicious node intrusions to the response system (Pathrater or Routeguard) and is therefore an extension of Watchdog.

Issues, Dialogue, and Drive Regarding Watchdog's Malicious Nodes A.

In both Watchdog and Routeguard, each node incorporates new information into its evaluations of other nodes it is familiar with.

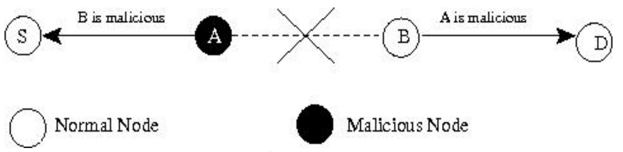

Figure 1: Malicious node A falsely report B as misbehaving in order to partition the network

Falsely accusing other nodes of bad behavior is a potential issue that might arise when a node acts deliberately or mistakenly. A rogue node might cause a split in the network by falsely asserting that nodes in its route are acting maliciously. For illustration, in Figure 1, node S is the initiator of communication and node D is the receiver. Even if node B is really forwarding packets, node A may mistakenly claim that it is not doing so. Because of this, S will blame B for bad behavior while A is really to blame.

The developers of Watchdog describe how to identify this kind of activity in [3]. If D sends an acknowledgment to S, it will be relayed via A, and S will be perplexed as to why it hasn't heard back from D directly. responses from D when it seems that packets were lost in transit from B. Furthermore, node B will identify this misconduct and report it to node D if A suppresses acknowledgements to disguise them from S.

The authors make the assumption that Watchdog would be able to spot this kind of issue. However, Watchdog is powerless if malevolent node A intends to split the network. The difficulty arises when both S and A label B as misbehaving and both D and B label A as misbehaving, as we can see by thinking about this more. Both nodes D and B will consider A malicious if node B communicates its suspicious behavior to node D. In the same way that A reports that B is harmful to S, S also considers B to be malicious. As a result, the network is effectively split in two: S, A on one side, and B, D on the other.

Such a situation is shown in Figure 1. After A reports B's inappropriate behavior to S, both S and A label B as hostile. If A is misbehaving and B reports it to D, then both D and B will label A as bad. This means that communication between points A and B has been severed.

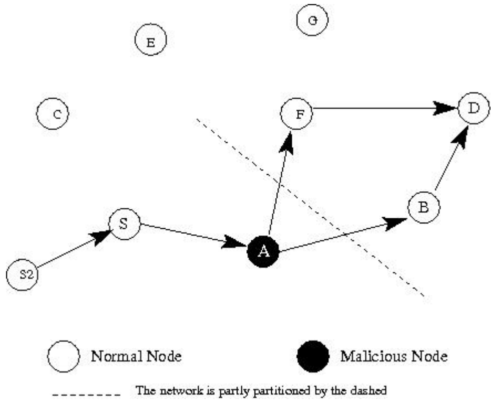

Figure 2: Malicious node A falsely reports all nodes on the path from source to destination as misbehaving in order to partition the network.

When node A, an intermediary in the network, indicates that every node along the route from S to D is hostile, this causes a severe split in the network. When a malicious node, A, is present on every route from S to D, we analyze this scenario. The outcome is seen in Figure 2 when a malicious node (A) claims that the other nodes (F and B) along the route from the source (S) to the destination (D) are also malicious.In Figure 2, there are two paths from S to D after Route Discovery:

S -> A -> B -> D,
andS -> A -> F ->
D.

If A then reports B and F for misbehaving, S will label B and F as hostile. Because A no longer provides S with acknowledgements, D, B, and F label A as evil. S will inform any other nodes that connect with D through S, such S2, and those nodes will respond accordingly. will also flag B and F as potentially harmful. And the network will be split in two, as in Figure 2's dashed line. It's possible that this is the worst-case scenario. Even if A is not part of the route from S to D (S > C > E > G > D), S will still be able to talk to D. However, even in this scenario, the malicious node A achieves some of its goals:

1) S will suspect B and F of malice and refuse to relay packets for them.

2) S incurs more expense by picking a less-than-ideal route from its route database or initiating a brand-new round of Route Discovery.

Node S, as an intermediary node, will not reply to a Route Discovery request with nodes B and F in the route path, even if this is the best possible route, while other nodes, such as S2, initiate a Route Discovery to node D. S will alert S2 that B and F are malicious, and S2 may either choose a suboptimal way from its route database or initiate a new Route Discovery if it already has routing routes to D that include paths S -> A -> B -> D and S -> A -> F -> D.

Another option is that D informs S that A is harmful by sending a message along the route D > F > E > C > S. In this case, S would be wise to adjust its routing table. If this occurs, S won't know which node to trust or which one is telling the truth.

Since Routeguard simply improves the rating system and doesn't have a method to verify the veracity of data given by nodes playing Watchdog, it faces the same issue.

B. A Description of the ExWatchdog System

As an addition to the Watchdog, our system looks for nodes that incorrectly identify other nodes as misbehaving. In most cases, these vertices are malignant rather than self-serving. They pose a greater threat to the network's efficiency. In the proposed system, the Routeguard is still used for emergency responses.

The following is an assumption made by our system: It is computationally prohibitive for a mobile device to deduce a private key from a public key alone since certain encryption algorithms are utilized and the key lengths are suitably lengthy. Assuming this is true, it would be impossible for hostile nodes to alter packets.

To accomplish the function of intrusion detection, we keep a table with the columns source, destination, sum, and path. This item is added to the database whenever a new packet is sent, forwarded, or received, regardless of whether the current node is the source, the destination, or an intermediate node. Each field's value is:

The location of the source is indicated.

The final destination's physical address.

cumulative: the sum of all packets sent, forwarded, or received by this node along this path. path The path may be the starting point, a connecting link, or the final destination.

path — The means via which two or more entities exchange information origin, final resting place>. To put it simply, a route is a collection of node addresses or a unique identifier for a path.

The source will not instantly lower the malicious node's rating when an intermediate node on a route path indicates that its next hop is malicious. Instead, it will use an alternate route in the route table to relay the message to its intended recipient. Source, target, total, and malicious node address are all included in the message. All of the aforementioned holds true for the origin, final resting place, and total. The reported malicious node's address is stored in malicious_node_address. The original node then looks in the routing database for a route that does not include the malicious node. If no such route exists, the origin will initiate Route Discovery to locate one. Once a route has been determined, the sending source will use that route to deliver the message.

When the destination node receives the message, it checks its own database for a possible match. If the destination node cannot find a matching item in the database, it confirms the malicious nature of the sending node by sending a message back to the source node. If so, the receiving node checks the sum field of the incoming message against the data in the database. If the two totals are the same, then the rogue node is harmless since it forwards all packets sent to it. The node may be dangerous if the two sums are not equal, though. After that, Routeguard will utilize this data to adjust the node's star rating.

Our solution has the same benefits as Watchdog. While doing so, it addresses a significant flaw related to pretend misconduct. It can tell whether one node is wrongly accusing another of bad behavior. As was previously said, erroneous reporting may cause a network to become divided, further reducing its performance.

However, our approach has several restrictions. It is hard for the source node to validate with the destination node that the report is true if the genuine malicious node is on all pathways from particular source and destination. Since we don't know who's lying and can't verify it, we have no choice except to do nothing at this time. Neither the reporting nor the reported node's reputation is lowered by the Routeguard.

Pseudo code for ExWatchdog's extra maintenance is shown in Figure 3.

IV. Performance Evaluation

In this part, we assess ExWatchdog's efficiency. Section IV.A describes the simulation model, whereas Section IV.B presents and discusses the findings.

Assumption Model for Simulation

Network Simulator (NS-2) [9] has been used to create a simulation model of the ExWatchdog system. Our simulations take place on a network consisting of fifty dispersed wireless mobile nodes in a flat area of 300 by 300 meters.

There are 10 CBR (constant bit rate) links between the nodes, and the maximum data rate is 4 bps. In random mode, the speeds of all the nodes may be anything from 0 meters per second to 3 meters per second. The duration of the simulation is 100 seconds. Nodes that agree to forward packets (without changing their contents) but later fail to do so owing to overload, selfishness, malice, or technical difficulties are considered misbehaving nodes in all studies. In our simulations, malicious nodes have a significant impact on network performance, in particular when they incorrectly label other, healthy nodes as malicious. We paid particular attention to the effect that this malicious conduct had on the network. In the simulated network with 50 nodes, a non-zero fraction of the nodes exhibit undesirable behavior. Misbehaving node percentages range from 0% to 40%, with 10% intervals in between. We begin by simulating a network where 30% of the nodes are acting maliciously by fraudulently reporting. Next, we do the same tests again, but this time with 80% of the nodes misbehaving by fraudulently reporting.

```
Nodes sending, forwarding, or receiving
packets do:

//search if the entry exists in
tableOne_entry = search_entry();

If one_entry = NULL
        // add new entry to
        tableadd_new_entry();
else
        // update the sum of existed
        entryUpdate_sum();

The destination node verifies if the node
ismalicious does:
One_entry = search_entry();

If one_entry = NULL
       return false;
else{
 sum_in_table = one_entry.getSum();
 sum_from_source =
 msg_from_source.getSum();if (sum_in_table
```

Figure 3: Abstract pseudo code of ExWatchdog operation

Simulation Results

Figures 4-7 show the simulation outcomes. Similar to [3], we utilize the Throughput and Overhead metrics to measure the success of our addition. The rate at which data packets are received by their intended recipients is known as throughput. In a computer simulation, the overhead is the amount of time spent sending and receiving information that is not actual data.

In Figures 4 and 5, we increase the number of badly behaved nodes from zero to a maximum of forty percent. with the percentage of falsely reporting misbehaving nodesfixed at 30%.

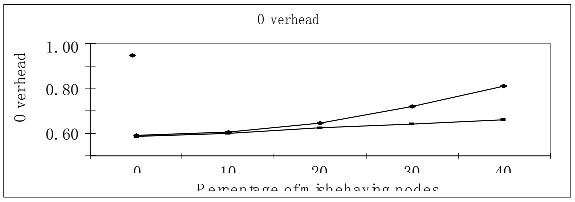

Figure 2: Overall Network Throughput As a Function of the Percentage ofMisbehaving Nodes

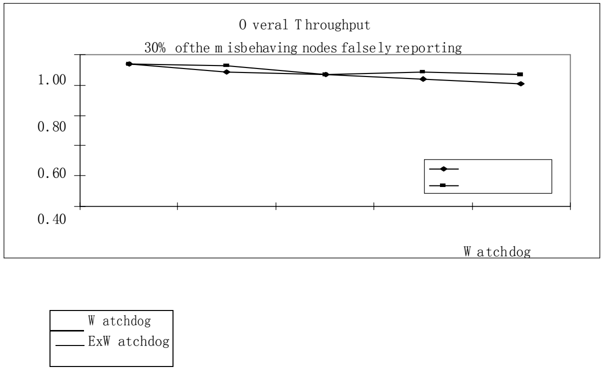

Figure 2: Network Overhead As a Function of the Percentage of Misbehaving Nodes

In Figure 1 we can see the overall network throughput. When there are no bad actors in the network, both curves may reach throughputs of 95 percent or higher. Due to the lack of the targeted misbehaving nodes that ExWatchdog seeks to identify, it is unclear what benefit ExWatchdog would provide in this scenario. The findings shift, however, if the 20% scenario of misbehaving nodes is taken into account.

Even when 40% of nodes are acting up, ExWatchdog's average throughput only drops by 7%, whereas Watchdog's average throughput drops by 18%. ExWatchdog can provide up to 11% more throughput than Watchdog. There isn't a huge jump in efficiency.

Even if the source cannot identify the erroneous reporting, Watchdog will force it to choose another way to deliver packets if a malicious node incorrectly claims that the next hop is harmful. Some malevolent nodes aren't interested in losing packets so much as they are in splintering the network. As a result, it has no impact on how many data packets reach their destinations.

Figure 5 depicts the ceiling height. When 40% of nodes are acting erratically, ExWatchdog reduces the overhead by up to 35% compared to Watchdog. That's because Watchdog will continue to overwhelm Route Discovery if every possible route between the source and the destination is erroneously reported to have at least one malicious node that is, in fact, benign. More control packets will need to be sent as a result of this. However, with ExWatchdog, the source will not change its behavior in response to the false report by choosing a different route or by broadcasting Route Discovery. It will make sure everything is right. If it turns out to be a fraudulent report, the source will merely lower the rating of the node that made the report.

When 80 percent of hostile nodes act maliciously by fraudulently reporting, the performance gap becomes readily apparent. As can be seen in Figure 6, there is no sudden dropoff in throughput for either Watchdog or ExWatchdog. This is the same conclusion we reached earlier when explaining how certain packets are utilized to find new routes but not many. As a result, the impact is negligible.

Figure 7 shows that there is a significant rise in Watchdog's communication overhead as the percentage of malicious nodes increases from 30% to 40%. The study of the trace file revealed that the Watchdog nodes have to broadcast Route Request many times, resulting in a substantial amount of communication overhead, when the 16 (50*40%*80 = 16) malicious nodes that falsely reports spread uniformly around the network. Once these malicious nodes are detected, no additional communication overhead will be incurred in ExWatchdog since the cost of detecting the specific harmful node remains constant.

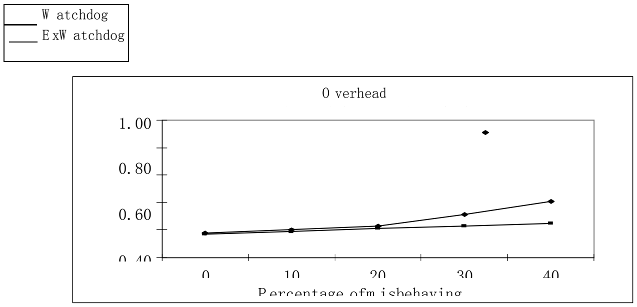

Figure 3: Network Overhead As a Function of the Percentage of Malicious Nodes 80% of Malicious Nodes Falsely Reporting

V. CONCLUSION AND FUTURE WORK

There are many real-world applications of the research being done on ad hoc networks, making it an exciting field of study. However, assaults on MANETs are common because of their fluid topology, lack of traditional security measures, and the insecure nature of their open channel of communication compared to that of their wired analogues.

In this work, we introduce ExWatchdog, an intrusion detection system built on top of a single suggested solution called Watchdog. By preventing a hostile node from fraudulently reporting other nodes as misbehaving and so dividing the network, ExWatchdog addresses a major flaw in Watchdog.

With some nodes acting maliciously to falsely report other nodes as misbehaving, we employ Throughput and Overhead as measures to assess ExWatchdog's performance. We evaluate Watchdog and our solution independently across all metrics. The simulation results demonstrate that our technique significantly reduces the overhead without visibly increasing the throughput.

Because our approach relies on the premise that the malicious nodes cannot modify the packet, we would want to investigate a more efficient and reliable technique to determine whether the reporting node is the genuine culprit in future work. In certain contexts, this may be an unrealistic assumption.

REFERE
NCES

[1] Routeguard: an intrusion detection and response system for mobile ad hoc networks, by A. Hasswa, M. Zulker, and H. Hassanein, published in Wireless And Mobile Computing, Networking And Communication 2005, Volume 3, pages 336–343, August 2005.

[2] Chapter 5: "Dynamic Source Routing in Ad Hoc Wireless Networks," in Mobile Computing, D. B. Johnson and D. A. Maltz (eds.), Kluwer Academic Publishers, 1996, pages 153–181.

[3] Mitigating Routing Misbehavior in Mobile Ad Hoc Networks, by S. Marti, T.J. Giuli, K. Lai, and M. Baker, 6th International Conference on Mobile Computing and Networking, MOBICOM'00, pp. 255-265, August 2000.

[4]

[5] Internet Draft, IETF, August 1998 [4] Temporally-Ordered Routing Algorithm (TORA), V. Park and S. Corson.

[6] Secure Routing and Intrusion Detection in Ad Hoc Networks, Proceedings of the Third International Conference on Pervasive Computing and Communications, Kauai Island, Hawaii, March 2005 [5] by A. Patwardhan, J. Parker, A. Joshi, M. Iorga, and T. Karygiannis.

[7] Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, ACM SIGCOMM 94 Conference on Communications Architectures, Protocols, and Applications, October 1994, pages 234-244. [6] Christopher Perkins and Pradeep Bhagwat.

[8] Ad hoc On demand distance vector routing, Proc. 2nd IEEE Wksp Mobile Comp. Sys. & Apps, February 1999, C. Perkins and E. Royer.

[9] Intrusion Detection in Wireless Ad Hoc Networks, 2003, F.H. Wai, Y.N. Aye, and N.H. James.

[10] Securing Ad Hoc Networks, IEEE Network Magazine, Special Issue on Network Security, Vol. 13, No. 6, November/December 1999 [9] - L. Zhou and Z. Haas.

[11] Intrusion Detection Techniques for Mobile Networks, Y. Zhang, W. Lee, and Y. Huang, Wireless Networks Journal, Volume 9, Issue 5, 2003.

[12] Source: [11] The NS2 Project, http://www.isi.edu/nsnam/ns/.