ANALYSIS AND IMPLEMENTATION OF INTRUSION DETECTION USING WIRELESS NETWORKS

Ms. Megha Jain*

Ms. Surbhi Birani*

Abstract

A number of neighbor-monitoring, trust-building, and cluster-based voting schemes have been proposed in the research to enable the detection and reporting of malicious activity in ad hoc networks. The resources consumed by ad hoc network member nodes to monitor, detect, report, and diagnose malicious activity, however, may be greater than simply rerouting packets through a different available path. In this paper we present a method for determining conditions under which critical nodes should be monitored, describes the details of a critical node test Simulation, presents experimental results, and offers a new approach for conserving the limited resources of an ad hoc network IDS.

KEY WORDS

Mobile ad hoc network, MANET, intrusion detection, IDS, worm model,

* Asst. Prof., Computer Science Department, Career College, Bhopal

1. Introduction

Wireless communications offer organizations and users many benefits such as portability and flexibility, increased productivity, and lower installation costs. Wireless local area network (WLAN) devices, for instance, allow users to move their laptops from place to place within their offices without the need for wires and without losing network connectivity. Less wiring means greater flexibility, increased efficiency, and reduced wiring costs. Ad hoc networks, such as those enabled by Bluetooth, allow data synchronization with network systems and application sharing

between devices. Handheld devices such as personal digital assistants (PDA) and cell phones allow remote users to synchronize personal databases and provide access to network services such as wireless e-mail, Web browsing, and Internet access.

However, risks are inherent in any wireless technology. Some of these risks are similar to those of wired networks; some are exacerbated by wireless connectivity; some are new. Perhaps the most significant source of risks in wireless networks is that the technology's underlying communications medium, the airwave, is open to intruders, making it

the logical equivalent of an Ethernet port in the parking lot.

The loss of confidentiality and integrity and the threat of denial of service (DoS) attacks are risks typically associated with wireless communications. Unauthorized users may gain access to agency systems and information, corrupt the agency's data, consume network bandwidth, degrade network performance, and launch attacks that prevent authorized users from accessing the network, or use agency resources to launch attacks on other networks. Network traffic can be monitored on a wired network segment, but ad hoc nodes can only monitor network traffic within their observable radio transmission range [1].

MANETs have come into prominence due to potentially rapid infrastructure-less deployment in military and emergency situations. However, the unreliability of wireless links between nodes, possibility of mobile nodes being captured or compromised, break down of cooperative algorithms, all lead to increased vulnerability [2]. Unrelenting attackers will eventually infiltrate

any system. It is important to monitor what is taking place in a system and look for intrusions. Intrusion Detection Systems (IDS) do precisely that. An IDS forms the second wall of defense in a high-survivability network.

Intrusion prevention measures such as authentication and encryption are not guaranteed to work all the time, which brings out the need to complement them with efficient intrusion detection and response. If an intrusion is detected quickly enough, the intruder can be ejected before any damage is done or any data is compromised. Effective IDS can not only serve as a prevention to prevent intrusions but also provide information about intrusions to strengthen intrusion prevention measures.

The paper is organized as follows: Section 2 provides some background work on IDS, Section 3 provides the Introduction of Critical Path Detection and Worm Propagation Model, Section 4 Simulation Details, section 5 Simulation Results, Section 6 Conclusion, Section 7 Future Work, Section 8 References.

2. Background

A number of IDS techniques have been proposed in the research literature. Moreover, a number of trust building and cluster-based voting schemes have been proposed to enable the sharing and vetting of messages, and data, generated and gathered by IDS systems. Zhang and Lee describe a distributed and collaborative anomaly detection-based IDS for ad hoc networks [3,4]. Tseng et al. describe an approach that involves the use of finite state machines for specifying correct AODV routing behavior and distributed network monitors for detecting run-time violation of the specifications [5]. Pirzada and McDonald present a method for building confidence measures of route trustworthiness without a central trust authority. The authors also present a concise summary of previous work in the area of establishing trust in ad-hoc networks [6]. Theodorakopoulos and Baras present a method for establishing trust metrics and evaluating trust [7]. Michiardi and Molva assign a value to the “reputation” of a node and use this information to identify misbehaving nodes and cooperate only with nodes with trusted reputations [8]. Albers and Camp couple a trust-based mechanism with a mobile agent based intrusion detection system, but do not discuss the security implications or overhead needed to secure the network and individual nodes from the mobile agents themselves [9]. Sun, Wu and Pooch introduce a geographic zone-based intrusion detection framework that uses location-aware

zone gateway nodes to collect and aggregate alerts from intra-zone nodes. Gateway nodes in neighboring zones can then further collaborate to perform intrusion detection tasks in a wider area and to attempt to reduce false positive alarms [10].

3. Critical Path Detection and Worm Propagation Model

In this section first, we describe the definition of a critical path is a path whose failure or malicious behavior disconnects or significantly degrades the performance of the network. Once identified, a critical path can be the focus of more resource intensive monitoring or other diagnostic measures. If a path is not considered critical, this metric can be used to help decide if the application or the risk environment warrant the expenditure of the additional resources required to monitor, diagnose, and alert other path about the problem.

Worm propagation model is mainly described as detailed network and abstract network. The detailed network can be an enterprise network and run by ISPs, the rest of the part is known as abstract network.

3.1 Architecture of IDS

Figure 3.1 shows the architecture of intrusion detection system, we first generate the test traffic using tcl script and find out the critical link in the network and then we block the path, after that worm propagation model is injected in the network through critical link and find out the type of attack, tcp, udp and cbr comparison before intruder and after intruder and we also find the node who spread the malicious activity.

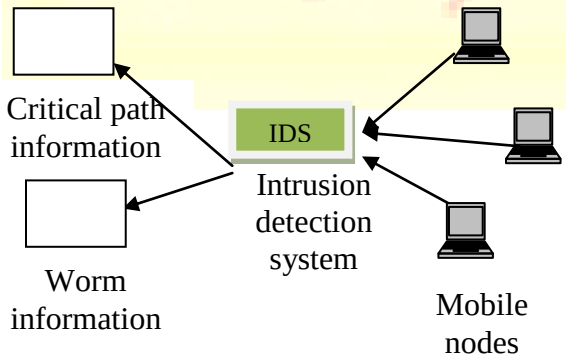

Figure 3.1 Architecture of IDS

3.2 Critical Path Detection

In this simulation module we use the trace file generated by ns-2, which is used as an input for C++ structure file, where we have created two linked lists. One link list stores incoming node numbers and other outgoing node numbers. In this file we used a count variable as global variable. These count variable stores the total number of pair's m-n, where m and n are some positive integer. Whenever first value pair comes then the count variable will increase and in the same manner we read all the incoming and outgoing node number and set the count for this path. After that we check that which incoming and outgoing node count value is greater in all pair's and set the path between these nodes as a critical path. After that we enter a worm propagation model in the network and check the status of the network and then we find out the intruder node.

3.3 Worm Propagation Model

Worm propagation model can be described as detailed network and abstract network. The detailed network could be an enterprise network, the whole network can be considered as detailed network in our simulation. Abstract network is also a part of the detailed network but the only difference is that it can be worm node where the infection occurrence can be assumed, so an abstract network can be called as susceptible infectious removal model. Worm node takes place in the network were we have blocked the path.

The communication between detailed network and abstract network is done through actual packet transmission that is the probing packet generated by compromised node in both parts. A vulnerable node is compromised upon receiving a probing packet. Then it chooses a target node to scan. Probing packet has no effect on invulnerable nodes. Figure 3.2 shows the worm propagation model

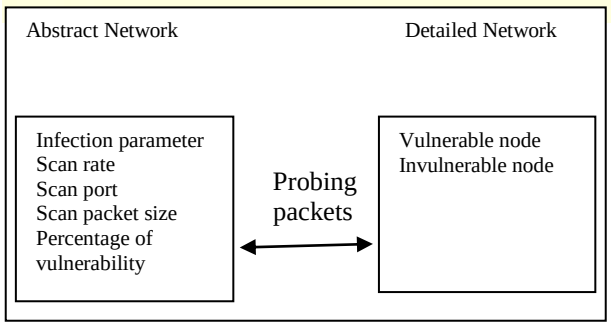

Figure 3.2 Worm Propagation Model

3.4 Intrusion Detection System

An IDS is shown below in figure:-3.3 which have three different modules namely Normal Profile, Worm node and Intrusion information. . Normal profile consists of TCP transmission, UDP transmission and CBR transmission; it also contains the path of packet flow in the network this information is before the worm node enters in the network. After that worm node enter the network in place of critical path, it captures the information of normal profile and infect the vulnerable node in network through message passing (probing packets) between abstract network and detailed network and then worm node set the scan rate, scan port, percentage of vulnerability and infection parameters. If probing port of detailed network and abstract network are same then worm node sends the infected packets to all the vulnerable nodes and infects the network. Intrusion Information takes the information from both normal profile and worm node and detects the intrusion by comparing both the information's. It checks for fields like worm node number, port number, time of intrusion and type of attack.

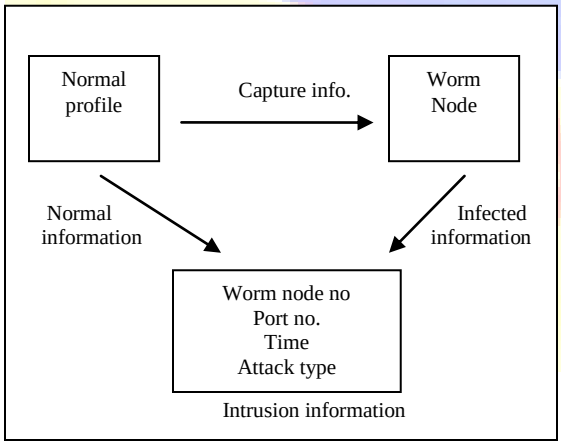

Figure 3.3 Model of Intrusion Detection System

4. Simulation Details

The simulation described in this paper was tested using the ns-2 test-bed that allows users to create arbitrary network topologies [11]. By changing the logical topology of the network, ns-2 users can conduct tests in an ad hoc network without having to physically move the nodes. ns-2 controls the test scenarios through a wired interface, while the ad hoc nodes communicate through a wireless interface.

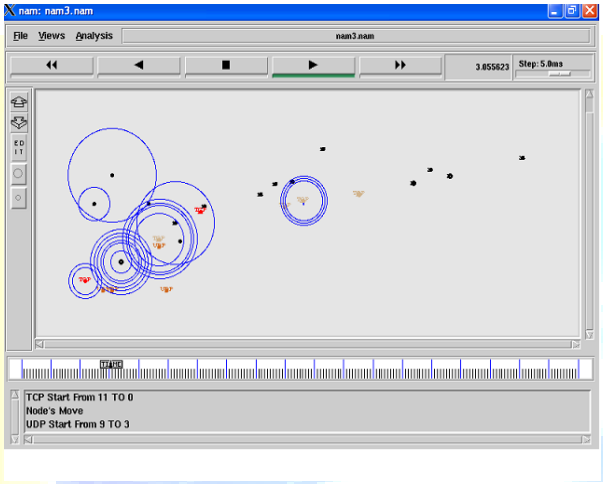

Figure 4.1 A sample topology generated by ns-2

The topology shown in Figure 4.1 is used to show how the IDS collects information and determines if a path and its incident communication links warrant the invocation of the critical path test. In order to illustrate the detection of critical path, we first generate some test traffic in the network. TCP socket servers are initiated at nodes 0, 11 and 19 to generate TCP traffic. Three TCP socket clients are initiated at nodes 0, 12 and 22. These clients send simple socket messages every 2 to 3 seconds to the servers. Node initiates a ping of node 3 and similarly node 5 initiates a ping of node 9 in order to create UDP packet traffic within the network. Finally two Secure Shell (SSH) sessions are initiated between node 0 and node 11, and node 22 and node 19.

The Simulation ends up with generation of a trace file and a nam file. We use the trace file to retrieve **Hs (**id for this source node) and **Hd (**id for next hop towards the destination). AWK utility is used to retrieve those fields from trace file and save them in a different file then the generated out put file can be use for calculating number of packet travel from each path by using C++, after that we find out the path from where the maximum data has travel and set the path as a critical path. Table1 shows the source to destination path and traffic

From_Node	To_Node	No_of_pkt_recorded
6	1	2726
5	3	1640
16	4	322
22	1	1712
0	2	744
From node	To Node	Maximum Traffic
6	1	2726
Total traffic from all path		7144

Table 1 shows the source to destination path and traffic

Then after detecting the critical path a worm model replaces the critical path. The test traffic is again generated and then we again monitor the network and record the changes in the network. Changes can be inspected by examining the trace file. If we found any change in the information in any field of the trace file or more losses in TCP, UDP and CBR or infected UDP packet then we can conclude that the path is critical and our assumption of critical path is true otherwise it is not the critical path and we keep on checking the same thing for different paths.

5. Experimental Results

UDP Comparison before Intrusion

Figure 5.1 shows UDP packet transmission which includes packet receive, packet loss and total packet transmission and comparison before intruder node enter in our network. Graph shows that the packet received is much more than the packet loss.

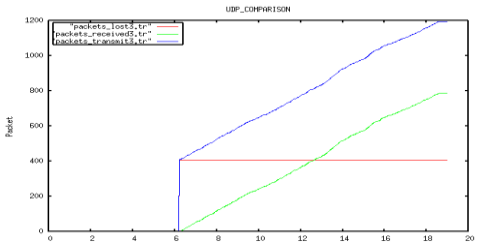

Figure 5.1 UDP packets before intrusion

UDP Comparison After intruder

Figure 5.2 shows UDP packet transmission which includes packet receive, packet loss and total packet transmission and comparison after intruder node enter in our network. Graph shows that the packet received is much less than the packet loss.

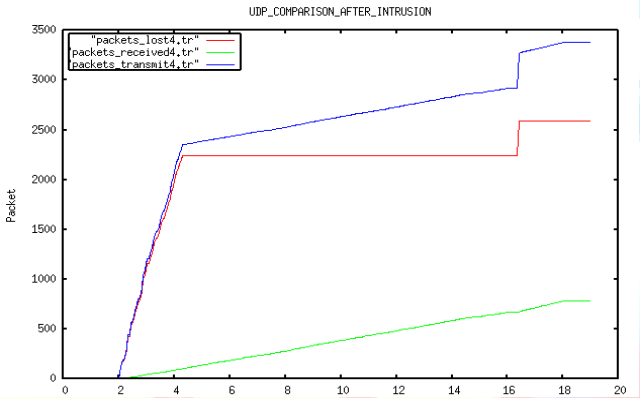

Figure 5.2 UDP packets after intrusion

TCP Comparison

Figure 5.3 shows TCP packet transmission which includes packet receives and total packet transmission and comparison before and after intrusion in network. Graph shows that the packet received is much large before intrusion than after intrusion.

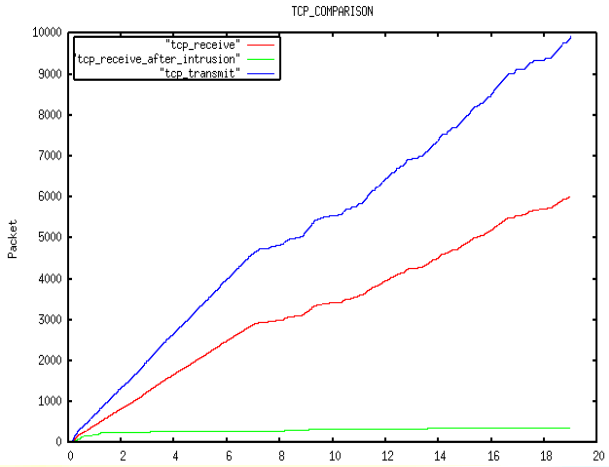

Figure 5.3 TCP packets before intrusion and after Intrusion

CBR Comparison

Figure 5.4 show CBR packet transmission which includes packet receives and total packet transmission and comparison before and after intrusion in network. Graph shows that the packet received is less before intrusion than after intrusion.

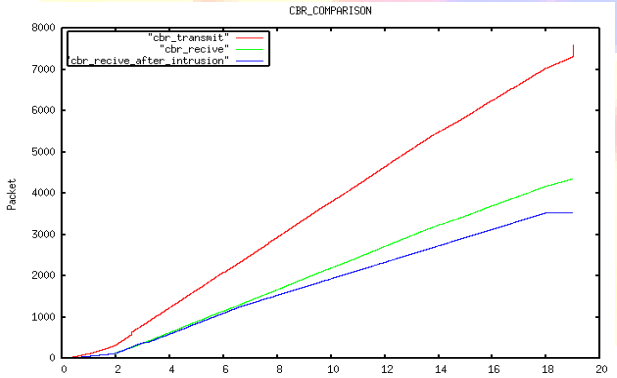

Figure 5.4 CBR packets before intrusion and After Intrusion

UDP Infected packet

Table 2 given below shows the sample of infected packet after the intrusion has occurred in the network

worm node	Time	Total Host	scan	Address range	Sport	Dport
A	2	159979	1	39996	0	0
A	3	159977	3	39996	0	0
A	4	159974	6	39996	0	0
A	5	159967	13	39996	0	0
A	6	159956	24	39996	0	0
A	7	159933	47	39996	0	0

Table 2 UDP infected packet

6. Conclusion

We perform number of test in ns-2 simulator and find out critical path after that we block the link and worn model infects the network. Here some result is shown. Table 3 conclude if the number of nodes are minimum than UDP packet received is much large before intrusion than after intrusion and TCP packet block in case of intruder node is also large. Because received percentage decreases after intruder node enters in network and the CBR packet reception is also decreased.

UDP				TCP		CBR	
before intrusion		after intrusion		before intrusion	after intrusion	before intrusion	after intrusion
received%	loss%	received%	loss%	received%	received%	received%	received%
66	34	24	76	62	17.2	71.4	56.2

Table 3 Packet Comparison before and after intrusion

Based on number of simulation analysis, first we find critical path in ad-hoc network. And after that we check the activities of critical node by injecting the worm packets in that critical node and than analyze the UDP, TCP and CBR packets. And we find out the following information that shows when intruder comes in the network

1) In TCP packets : more packets are blocked (more packet loss)
2) In UDP packets: more packets loss and some received packets are also infected.
3) In CBR packets: little bit more loss than without intruder.

7. Future work

In this project we detect only single critical path and single intruder node in ad-hoc network. In future we trace all the critical paths and nodes. We injected worm packet in critical node so that it act like an intruder and than analyze TCP, UDP, and CBR packets transmission. So we can also apply the other techniques like packet capturing, false route forwarding, changing source and destination addresses etc.

8. References

[1] A. Karygiannis, E. Antonakakis, and A. Apostolopoulos, Detecting Critical Nodes for MANET Intrusion Detection Systems". Available: {arygiannis@nist.gov }

[2] Ketan Nadkarni, Amitabh Mishra,"A Novel Intrusion Detection Approach for Wireless Ad hoc Networks". Available: {ketann@vt.edu, mishra@vt.edu }

[3] Y. Zhang and W. Lee. Intrusion detection in wireless ad hoc networks. In Proceedings of the 6th annual international conference on Mobile computing and networking, pp. 275–283. ACM Press, 2000.

[4] Y. Zhang, W. Lee, and Y. Huang. Intrusion detection techniques for mobile wireless networks. ACM/Kluwer Mobile Networks and Applications (MONET), 2002.

[5] C.-Y. Tseng, P. Balasubramanyam, C. Ko, R. Limprasittiporn, J. Rowe, and K. Levitt. A specificationbased
intrusion detection system for AODV. In Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks, pp. 125–134. ACM Press, 2003.

[6] Pirzada, Asad Amir, and McDonald, Chris. Establishing trust in pure ad-hoc networks. Proceedings of the 27th conference on Australasian Computer Science - Volume 26, pp 47-54, 2004

[7] Theodorakopoulos, George and Baras, John. Trust evaluation in ad-hoc networks. Proceedings of the 2004 ACM workshop on Wireless security, pp. 1-10, 2004.

[8] Michiardi, P. and Molva, R., "Core: A Collaborative Reputation mechanism to enforce node cooperation in Mobile Ad Hoc Networks", Communication and Multimedia Security 2002 Conference.

[9] Albers, Patrick and Camp, Olivier. Security in Ad hoc Networks: a general Intrusion detection architecture enhancing trust based approaches. Proceedings of the First International Workshop on Wireless Information Systems 2002.

[10] Sun, Bo, Wu, Kui and Pooch, Udo. Alert aggregation in mobile ad hoc networks. Proceedings of the 2003 ACM workshop on Wireless security, pp.69 – 78, 2003.

[11] The Network Simulator – ns-2 http://www.isi.edu/nsnam/ns/