A Novel IDS Agent Distributing Protocol for MANETs[1]

Jin Xin, Zhang Yao-Xue, Zhou Yue-Zhi, and Wei Yaya

Key Laboratory of Pervasive Computing,
Department of Computer Science & Technology,
Tsinghua University, Beijing, China
jx01@mails.tsinghua.edu.cn

Abstract. Intrusion Detection Systems (IDSs) for Mobile Ad hoc NETworks (MANETs) is becoming an exciting and important technology in very recent years, because the intrusion prevention techniques can not satisfy the security requirements in mission critical systems. The proposed IDS architecture can be divided into two categories by the distributing form of IDS agents: fully distributed IDS and cluster-based IDS. The former has a high detection ratio, but it also consumes a cascade of energy. The latter has considered energy saving, but some hidden troubles of security exist in it. In this paper, we have proposed a novel IDS Agent Distributing (IAD) protocol for distributing IDS agents in MANETs. IAD protocol divides the whole network into several zones, selects a node subset from each zone, and runs IDS agent on the node in this subset. At the same time, IAD protocol can rectify the number of nodes running IDS agent according to the threat level of the network. Compared with the scheme that each node runs its own IDS, our proposed scheme is more energy efficient while maintaining the same level of detection rate. While compared with the cluster-based IDS scheme, our scheme is more flexible when facing the emergent situations. Simulation results show that our scheme can effectively balance the security strength and energy consuming in practice.

1 Introduction

With rapid development of MANET applications, security becomes one of the major problems that MANET faces today. MANET is much more vulnerable to attacks than wired networks, because the nature of mobility creates new vulnerabilities that do not exist in fixed wired networks. Intrusion prevention measures, such as encryption and authentication, can be used in MANET to reduce intrusions, but cannot eliminate them. In mission critical systems, which require strict secure communication, intrusion prevention techniques alone cannot satisfy the security requirements. Therefore, intrusion detection system (IDS), serving as the second line of defense, is indispensable for MANET with high security requirements.

In this paper, we present our progress in developing an IDS Agent Distributing (IAD) protocol for distributing IDS agents in MANET. In wired network, traffic monitoring is usually done at traffic concentration points, such as switches or routers. But in mobile ad hoc environment, there is no such traffic concentration point.

[1] Supported by the National 863 High-Tech plan (No. 2002AA111020).

V.S. Sunderam et al. (Eds.): ICCS 2005, LNCS 3515, pp. 502 – 509, 2005.

Therefore, the IDS agents need to be distributed on the nodes in MANET. In addition, battery power is considered being unlimited in wired network, but MANET nodes typically have very limited batter power, so it is not efficient to make every node always run its IDS agent. The purpose of our scheme is to reduce the number of nodes running IDS agent, while maintain the same level of detection.

The rest of the paper is organized as follows. In section 2, we illustrate the motivation why we propose such an approach and some specific assumptions we rely on. Section 3 describes IAD protocol in detail. Simulation results are shown in Section 4. Section 5 concludes the paper.

2 Motivations and Assumptions

Extensive research has been done in this field and efficient IDS architectures have been designed for MANET. These architectures can be classified into two categories.

The first category is a fully distributed IDS architecture proposed in [1]. In this architecture, an intrusion detection module is attached to each node and each node in the network uses the local and reliable audit data source to participate in intrusion detection and response.

The second category is a cluster-based architecture. In [2], in order to address the run-time resource constraint problem, a cluster-based detection scheme is proposed. The whole network is organized as several clusters, each cluster elects a node as the clusterhead, and the clusterhead performs IDS functions for all nodes within the cluster.

There are some demerits on both of the two architectures. The first kind of architecture has a high detection ratio but it consumes a lot of power of each node. For the second kind of architecture, if a malicious node by any chance has been elected as the cluster head, it can launch certain attacks without being detected because it is the only node running IDS in the cluster. In addition, when mobility is high, the introduction of control overhead to create and maintain the cluster is unbearable. In one word, none of these two architectures considers both the effectiveness of IDS itself and the resource constrains of each mobile node at the same time.

3 IAD protocol

Aiming at solving this problem, we put the focus of our research on the combination of these two architectures and proposed the IAD (IDS Agent Distributing) protocol.

IAD protocol is based on the following assumptions:

1. Each node contains a unique and ordered identifier.
2. Neighbor information is always available.
3. Each node can overhear traffic within its transmission range.

IAD protocol consists of three sub-protocols: Neighbor Information Transmission (NIT) protocol, Cover Set Selection (CSS) protocol, and Monitoring Nodes

Adjustment (MNA) protocol. Node changes its state among the following states shown in Figure 1.

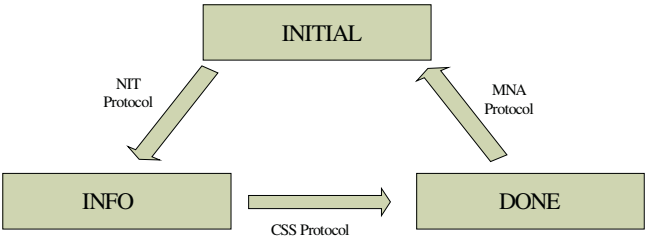

Fig. 1. State Changing in IAD Protocol

Initially, all nodes are in an INITIAL state. In that state, each node doesn't know the neighbor information of other nodes in the same zone. And, each node runs its own IDS agent. Once NIT protocol has finished, all nodes change their state from INTIAL to INFO, that is, each node has the information of other nodes' neighbor number. After CSS protocol has finished, a subset of nodes is selected, and the nodes in the subset continue running IDS agents while other nodes which are not included in the subset stop running IDS agent. All nodes enter into DONE state. If a node running IDS agent has detected an intruder, it will broadcast an Alarm message. Any node receiving this Alarm message will change its state back to INITIAL state again.

3.1 Neighbor Info Transmission Protocol

NIT protocol is responsible for transmitting a node's neighbor information to all other nodes in the same zone by a Neighbor Info Message (NIMsg), which data structure is defined in Table 1. For recording the NIMsg from other nodes, each node stores a Neighbor Info Table (NITbl) whose data structure is defined in Table 2.

Table 1. Data Structure of NIMsg

Field	Meanings
NodeAddr	Node address
ZoneID	Zone ID
SeqNum	Sequence number
NbrNum	Neighbor number of a node (in the same zone)
NbrChain	A chain that records the neighbors of a node

Table 2. Data Structure of NITbl

Field	Meanings
NodeAddr	Node address
NbrNum	Neighbor number of a node (in the same zone)
NbrChain	A chain that records the neighbors of a node

Without loss of generality, we discuss some zone *Z*. The topology map of zone *Z* is assumed as Figure 2. The number in the circle represents the neighbor number of the node. In the initial state, all nodes run their own IDS agent.

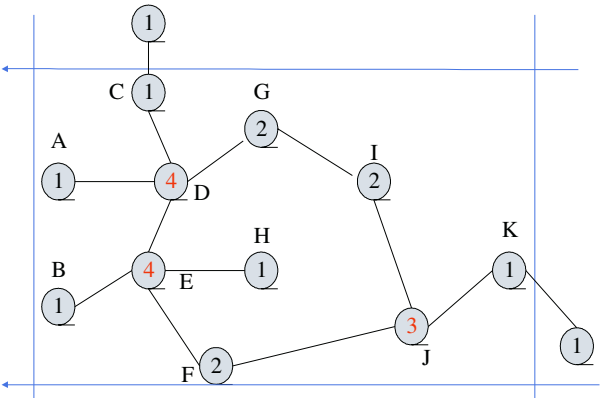

Fig. 2. Topology Map of Zone Z

Assume that *i*, *j* are two arbitrary nodes in zone *Z*. The concrete process of NIT protocol is described as following:

1. If none of the nodes has detected any intruder in a period T_1, every node will broadcast NIMsg.
2. When node *j* receives a NIMsg from node *i*, If $ZoneID_i = ZoneID_j$ and $SeqNum_i >= SeqNum_j$, then node *j* will do the following:
   (a) Restore the information of node *i* or update the old information of node *i* in its NITbl.
   (b) Forward this NIMsg by a probability *P* (*P*=0.7)[3].
3. Otherwise node *j* will discard this message.

Table 3. NITbl in node D

Node Address	Neighbor Number	Neighbor Chain
A	1	D
B	1	E
C	1	D
D	4	A->C->E->G
E	4	B->D->F->H
F	2	E->J
G	2	D->I
H	1	E
I	2	G->J
J	3	F->I->K
K	1	J

After NIT protocol has finished, all nodes in the zone have constructed a NITbl. For example, in Figure 2, node D will construct a NITbl shown as Table 3.

3.2 Minimal Cover Set Selection Protocol

We want to find out a subset of nodes, which satisfy that the nodes in the subset can overhear all the traffic in the zone. Then if only IDS agents run on these nodes, they can monitor all the traffic in the zone.

Assume that there is a set $A=\{a_1,a_2,\ldots\ldots,a_m\}$, a_i represents the node in network. And S_i represents all the neighbor nodes of node a_i, $S_i=\{a_i^1,a_i^2,\ldots\ldots,a_i^n\}$. If we can find out k nodes, which satisfy $\bigcup_{i=1}^{k} S_i = A$, then these nodes can overhear all the traffic in the zone. The key point is how to select this node subset. The above problem can be mapped to the classical minimal cover set problem in graph theory.

Minimal Cover Set Problem is a widely used problem among NP hard problems. There are many effective heuristic algorithms to solve it by now. Common heuristic algorithm is made up of one or more heuristic strategies. In general, the more the number of strategy, the more optimal the solution is. But more strategies also take more time to compute. Because of limited battery power in mobile nodes, and the high requirement for real time, we choose the greedy algorithm due to its simplicity and low complexity. Greedy algorithm always takes the best immediate, or local, solution while finding an answer. It finds the overall, or globally, optimal solution for some optimization problems, but may find less-than-optimal solutions for some instances of other problems. It never reconsiders this decision, whatever situation may arise later.

The concrete steps for finding a node subset by using greedy algorithm are in the following way:

1. Select a node from NITbl which has the most neighbors. If several nodes have the same neighbors, then the one which has the smallest address is selected.
2. Record the chosen node into the subset and delete the row which has the node in the NITbl.
3. Repeat step 1 and step 2, until the neighbors of nodes in the subset can cover all the nodes in the zone.

3.3 Monitoring Nodes Adjustment Protocol

When some node detects an intruder in the network, it will adjust the number of monitoring nodes using MNP protocol. The concrete process is as followings:

1. Once a node has detected an intruder, it will broadcast an Alarm message. The data structure of the Alarm message is shown in Table 4.
2. When any node receives this Alarm message, it will first determine the freshness of the Alarm message by SeqNum. If this Alarm message has already been received, the node will discard it. Otherwise, the node will perform following.
   - Record the AttackAddr and SeqNum field.
   - Start to run its IDS agent.
   - Forward this Alarm message.
   - Stop broadcasting NIMsg.

After all the nodes start their respective IDS agent, the network goes back to the fully distributed IDS architecture. We can use the method in [1] as the following intrusion respond method. If there is no intrusion in next period of T_1, the IAD protocol will begin its subset selection process again.

Table 4. Data Structure of Alarm message

Field	Meanings
AttackAddr	Address of the attacker
SeqNum	Sequence number

4 Simulation Results

We use a simulation model based on GloMoSim [4] to investigate the performance of the proposed approaches. In our simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. We assume all nodes have the same transmission range of 250 meters. A free space propagation model with a threshold cutoff is used as the channel model. We use the Distributed Coordination Function (DCF) of IEEE 802.11 for wireless LANs as the MAC layer protocol. It has the functionality to notify the network layer about link failures.

In the simulation, nodes move in a 1000*1000 meter region and we change the number of nodes from 30 to 100. The mobility model is the random waypoint model. The minimal speed is 5 m/s, and the maximal speed is 10 m/s. The pause time is 30 seconds. 5 source-destination pairs of CBR traffic and 3 source-destination of TCP traffic are used as the background traffic. All traffic is generated, and the statistical data are collected after a warm-up time of 300 seconds in order to give the nodes sufficient time to finish the initialization process.

Fig.3 compares the total number of nodes and the number of monitoring nodes under the same mobility level. From this figure, it can be seen that only half of the nodes are responsible to monitor neighbors in IAD protocol.

Fig.4 shows the average consumed power. In common conditions, the mode of wireless card can be divided into four kinds by the order of energy consumption: doze, idle, receiving and transmitting. Except doze, we call the other three modes as active state. In doze mode, Network Interface Card (NIC) neither sends nor receives signals, so this kind of mode is not fit for MANET. Feeney etc have tested the energy consumption of IEEE802.11 WaveLAN wireless network card produced by Lucent Corp. The result is shown in Table 5.

In GloMoSim simulator, wireless NIC is always under active state. The energy consumption consists of three parts: energy for sending data, energy for receiving data and energy in idle. When the amount of data to be sent and the sending power are fixed, the energy used to send data is fixed. In the fully distributed IDS architecture, when there is no data to be sent, the node is under receiving mode because every node needs to monitor the network, while in IAD protocol, when there is no data to be sent, the node will change its mode between idle and receiving according to its role. From the figure, we can conclude that IAD protocol can save about 10% of energy than that of fully distributed IDS architecture. When time goes on, IAD protocol will save more and more

energy. Especially, when traffic load is light, energy for sending data only accounts for a small ratio of energy consumption, then the power saving effect is more obvious.

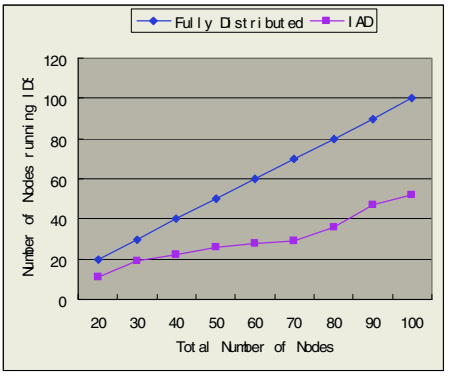

Fig. 3. Comparison of nodes

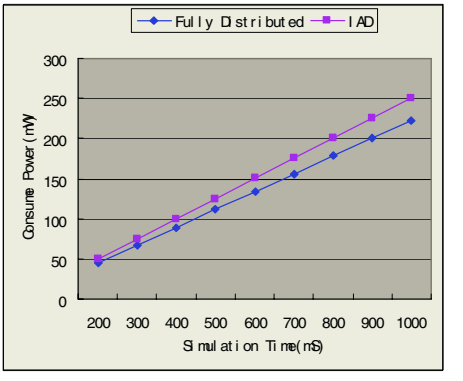

Fig. 4. Comparison of consumed power

Table 5. Data Structure of Alarm message

Mode	Actual Current	Actual Voltage	Referenced Current	Referenced Voltage
Doze	14mA	4.74V	9mA	5V
Idle	178mA		Null	
Receiving	204mA		280mA	
Transmitting	280mA		330mA	

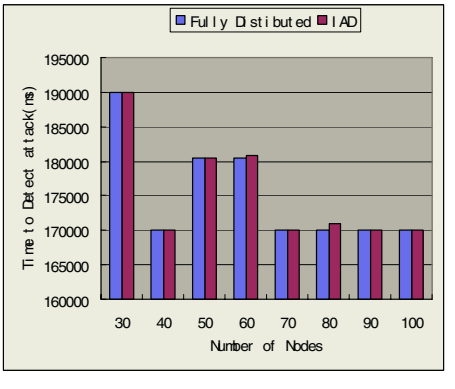

Fig. 5. Comparison of the time

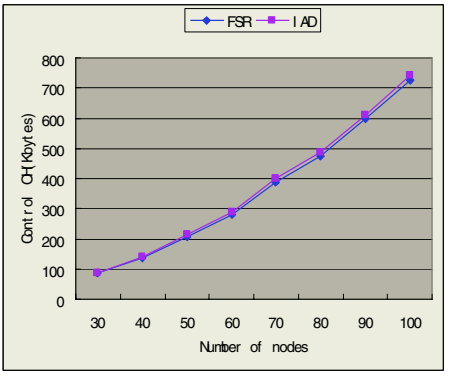

Fig. 6. Comparison of Control Overhead

Fig.5 compares the detection time between IAD protocol and the fully distributed scheme. We assume that there is one intruder sending a sequence of consecutive packets

constituting an attack to the destination. These packets are sent in a flow consisting of normal packets. Further, we assume that the nodes, which are a part of the intrusion detection subsystem, know this sequence of packets constituting the intrusion. The intrusion is considered being detected if this subsequence of attack packets pass through any of the nodes that constitute the intrusion detection subsystem.

In IAD protocol, since the nodes in the subset can overhear all the traffic of the network, the time spent by the IAD protocol to detect an intruder should be equal to the fully distributed scheme. Only when the attacked node moves out of the scope that the detecting nodes can overhear, the intruder may be detected later than fully distributed scheme. The figure shows that the IAD protocol can detect an attack almost as quickly as the fully distributed scheme. Even at the worse case, the IAD protocol only costs several more millisecond than the fully distributed scheme.

Figure 6 shows the additional control overhead introduced by IAD protocol. Because we implemented IAD protocol on FSR routing protocol, we compared the control overhead of original FSR routing protocol with that of IAD protocol. As shown in figure 6, IAD protocol only brings a small amount of additional control overhead. This part of control overhead is introduced by broadcasting NIMsg. Because 1) the period for broadcasting a NIMsg is much longer than routing update period, 2) the NIMsg is only broadcasted within the zone, 3) each node forwards this message only with a probability *P*, we can effectively control this part of overhead within a small scope.

5 Conclusion

Intrusion detection is an indispensable second wall of defense especially in any high-survivability network. Considering the limited computational and energy resources of mobile nodes, it is not efficient to make every mobile node always run IDS agent on itself. In this paper, we have proposed an IAD protocol for MANET. Its goal is to minimize the consumption of battery power and at the same time maintains an acceptable level of monitoring. It divides the whole network into several zones, selects a node subset that can overhear all the traffic from each zone, and all the nodes in the subset run IDS agents. In addition, it can rectify the detection level if intruders emerge. Simulation results show that the IAD protocol can implement the goals above efficiently.

Reference

[1] Y. Zhang and W. Lee, “Intrusion Detection in Wireless Ad Hoc Networks,” the 6th Annual Int’l Conf. on Mobile Computing and Networking (MobiCom’00), Boston, MA, Aug., 2000, pp. 275-283.

[2] Yian Huang and Wenke Lee,“A Cooperative Intrusion Detection System for Ad Hoc Networks,“In Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '03), Fairfax VA, October 2003, pp. 135 – 147

[3] Z. Haas, J. Halpern, and L. Li, "Gossip-based ad hoc routing," in IEEE InfoCom Proceedings 2002, vol. 3, pp. 1707--1716, June 2002.

[4] X. Zeng, R. Bagrodia, and M. Gerla, “GloMoSim: a Library for Parallel Simulation of Large-Scale Wireless Networks,” Proc. of the 12th Workshop on Parallel and Distributed Simulations (PADS ’98), Banff, Canada,May 26-29, 1998, pp. 154-161.