International Journal of Scientific & Engineering Research, Volume 3, Issue 3, March-2012 1
ISSN 2229-5518

Anomaly detection using clustering for ad hoc networks -behavioral approach-

M.Belacel

Abstract— Mobile ad hoc networks (MANETs) are multi-hop wireless networks of autonomous mobile nodes without any fixed infrastructure. In MANETs, it is difficult to detect malicious nodes because the network topology constantly changes due to node mobility. Intrusion detection is the means to identify the intrusive behaviors and provide useful information to intruded systems to respond fast and to avoid or reduce damages. The anomaly detection algorithms have the advantage because they can detect new types of attacks (zero-day attacks).In this paper, we present a Intrusion Detection System clustering-based (ID-Cluster) that fits the requirement of MANET. This dissertation addresses both routing layer misbehaviors issues, with main focuses on thwarting routing disruption attack Dynamic Source Routing (DSR). To validate the research, a ca se study is presented using the simulation with GloMoSum at different mobility levels. Simulation results show that our proposed system can achieve desirable performance and meet the security requirement of MANET.

Index Terms— MANET, ad hoc, Intrusion Detection System, routing disruption attack, intrusion, anomaly detection, DSR, IDS, cluster.

1 Introduction

We present an approach to anomaly detection based on clustering. In this paper, we describe a two-level Cluster-based Intrusion Detection System (AD-CLUSTER) which meets the requirement of MANETs. In the system aspect of AD-CLUSTER, an IDS agent is attached to each node. The network is logically divided into clusters which enable these agents to cooperate with each other to perform the intrusion detection task. In the high level of AD-CLUSTER, the gateway nodes (also called inter-cluster nodes, those nodes which have physical connections to different clusters) of each cluster are responsible for aggregating and correlating the locally generated alerts inside the cluster. An algorithm is presented to aggregate the locally generated alerts based on their attribute similarities in order to further improve the performance of AD-CLUSTER. In AD-CLUSTER, only the gateway nodes can utilize alerts to generate alarms.

2. OVERVIEW OF DYNAMIC SOURCE ROUTING (DSR)

The DSR protocol is composed of two mechanisms that work together to allow the discovery and maintenance of source routes in the ad hoc network [1], route Discovery Is the mechanism by which a node S wishing to send a packet to a destination node D obtains a source route to D. Route Discovery is used only when S attempts to send a packet to D and does not already know a route to D. and Route Maintenance Is the mechanism by which node S is able to detect, while using a source route to D, if the network topology has changed such that it can no longer use its route to D because a link along the route no longer works. When Route Maintenance indicates a source route is broken, S can attempt to use any other route it happens to know to D, or can invoke Route Discovery again to find a new route. Route Maintenance is used only when S is actually sending packets to D.

3. ROUTING DISRUPTION ATTACKS AGAINST DSR

As briefly overviewed in Section II, nodes process route requests by sending back cached route replies or by re-broadcasting route requests in DSR route discovery phase whereas dropping some redundant ones, this paper will focus on the detection of attacks targeted at MANET routing protocols, more specifically on detecting one of the most important active attacks: *routing disruption* attack.

4. CLUSTER-BASED INTRUSION DETECTION SYSTEM

An intrusion detection agent is attached to each node. The agents cooperate with each other to form a complete MANET IDS. These IDS agents run independently and monitor local activities to detect abnormal behaviors. For the local IDS agent, we implement a Markov chain based anomaly detection algorithm. The anomaly-based method is adopted because it is expected that more types of attacks will be launched against MANETs in the future. It is also difficult to obtain the complete trace of attacks, which are often required in designing a misuse detection algorithm. We logically divide the network into clusters to manage locally generated alerts. By integrating the net information from a wider area, this management framework could reduce false alarms and improve the detection ratio. In AD-CLUSTER, an algorithm utilizing attribute similarities is also presented to aggregate the locally generated security related information.

4.1 AD-CLUSTER Framework

We adopt a cluster-based intrusion detection framework based on the following considerations:

- Due to the dynamic nature of MANETs, *alert flooding* is expected in such an environment. Attacks are likely to generate multiple related alerts. By creating some alert concentration points, we can logically group related alerts

together and reduce the false alarms generated for various reasons. Note that *alert* and *alarm* are different concepts as stated before.

- Flat architecture is undesirable in managing alerts.

When mobility is high, the introduction of the message overhead to create and maintain the hierarchy is unbearable. We thus adopt a cluster-based framework. It also requires little mobility management efforts. Actually, AD-CLUSTER requires no extra control messages propagated within the cluster in order to maintain the framework. Whether it is an inter-cluster node or intra-cluster node. A node may change its role over time due to mobility.

4.2 Internal Model of IDS Agent

The local IDS agent we use in AD-CLUSTER is shown in Fig.1. The data collection module is mainly responsible for collecting the security related data from various audit sources. The detection engine will use the data which are parsed, filtered and formatted by the data collection module to perform intrusion detection locally. The Local Aggregation and Correlation Engine (LACE) will locally aggregate and correlate the detection results from different detection engines in the IDS agent. No detection model stands alone as a catchall for network penetrations. In an environment with high security requirements, it is desirable to have multiple detection engines, which enable the use of different detection techniques.

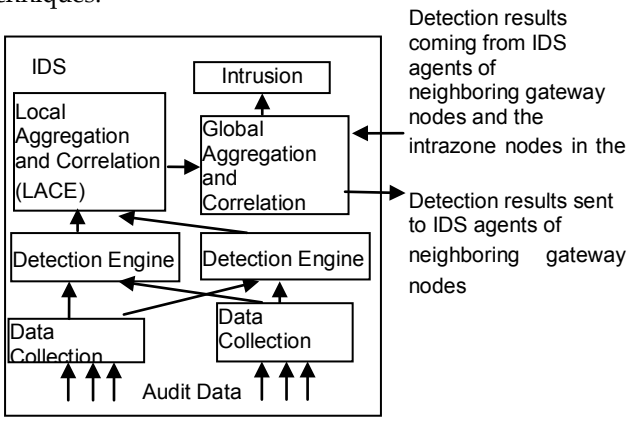


Figure 1: Diagram of an IDS agent.

They will complement each other to improve the detection performance. The functionality of LACE is to combine the detection results of different local detection engines. The functionality of Global Aggregation and Correlation Engine (GACE) depends on the type of the mobile nodes. If the node is an intra-cluster node, GACE is mainly responsible for transmitting the locally generated alerts to the gateway nodes in the same cluster; if the node is a gateway node, GACE is to aggregate and correlate the detection results from the LACE of its own agent and the LACEs of the intra-cluster nodes in the same cluster, and to cooperate with the GACEs of the gateway nodes with which it has physical connections. The intrusion response module is to handle the generated alarms.

5. LOCAL ANOMALY DETECTION

We use the following features that are sensitive to routing disruption attacks: PCR -Percentage of the change in route entries, and PCH -Percentage of the change in number of hops, which reflect the mobility of the network, to construct a Markov chain as the normal profile. Vector quantization (VQ) approach is used in this process to convert continuous raw audit data to categorized data items with minimum errors. In order to mitigate the impact of dynamics of MANETs on the construction of the normal profile, for those data items whose probability is below some threshold, we convert them to a common "rare" symbol. The output of the VQ is then used to construct a Markov chain model, which employs conditional probabilities in its transition probability matrix to represent the temporal profile of normal behavior.

6. ALERT AGGREGATION

Due to the lack of detailed analysis of attacks in MANETs, alert aggregation is very challenging in MANET environment. Potential complex attack scenarios could make the aggregation very complicated. Moreover, there are not many efforts that have been devoted to the local detection of MANET attacks. In this section, we provide our initial work in this respect. Because we lack detailed analysis of MANET attacks in the literature and sophisticated attacks may make the situations very complex, we only consider the same occurrence of attacks. Specifically, we still target at the *routing disruption* attack. Because there are not many efforts that have been devoted to the local detection of MANET attacks, we use the Markov chain based anomaly detection model described in Section 5 as the local detection model.

6.1 Collaboration Mechanism

In mobile ad hoc networks, the attackers can launch attacks when they are close to victims. Thus, in general cases, through the local IDS agent attached to the node itself and/or neighboring nodes, these attackers can be detected. There are other kinds of attacks, however, which the attackers may launch far away from the victims. What's more, two or more attackers may collude to launch more complicated attacks. In these situations, it is very difficult for the local IDS agent itself to detect the attacks. In addition, the results based on the local IDS agent could lead to a very high false positive ratio. There may exist two possible mechanisms for the gateway nodes to collaborate. One is the subscription-based mechanism. It is not necessary that the gateway nodes collect all of the security related information from the IDSs of intra-cluster nodes in order to draw some conclusions. Based on its own status, the IDS of the gateway node can send a subscription message to its intra-cluster nodes to subscribe security related information. The subscription message could contain information that is related to the required data. The intra-cluster nodes can thus generate corresponding messages to fit the subscribed requirement. This mechanism introduces low communication overhead. However, the gateway node needs to carefully analyze the messages in order to determine what information is needed. The other one is the local broadcast

mechanism for IDS agents to collaborate. When the IDS of the intra-cluster node generate a local alert, it could locally propagate the detection results to its gateway nodes. When nothing is suspicious in the last period, there is no need for the local IDS to propagate security information. The neighboring gateway nodes could further collaborate with each other through the transmission of the security-related information. In this way, we avoid the use of global broadcast. It is also unnecessary to propagate local alerts inside the cluster every period. All these strategies can result in less communication overhead. This is important because message sending and receiving is very expensive in terms of energy consumption. This mechanism could also enable gateway nodes to collect enough information to make final decisions. For simplicity, we adopt the local broadcast mechanism in our implementation. In the cluster-based intrusion detection framework, only gateway nodes could generate alarms. The local IDS attached to local nodes could only generate alerts based on their local information and propagate these alerts inside the cluster. The gateway nodes, having gathered the alert information periodically, could make better final decisions.

6.2 Aggregation Mechanism

The main objective of the aggregation algorithm and the cluster-based framework is to reduce false positive ratios and increase detection ratios by aggregating local alerts. Global alerts could be generated to provide more diagnostic information of attacks. By grouping alerts together, aggregation will allow a better evaluation of the progress of the attack. In order to do so, we need the definition of a data model in the form of a class hierarchy to describe the alerts.

Class Hierarchy of the Alerts: We use the definition and implementation method recommended by the Intrusion Detection Working Group (IDWG) to describe the alert classes in AD-CLUSTER. The Intrusion Detection Message Exchange Format (IDMEF) [2] proposed by IDWG aims at wired IDSs. Its purpose is to define common data formats and data exchange procedures for sharing information of interest to intrusion detection systems. Due to the unique characteristics of MANETs and because we focus on the intrusion detection targeted at the network layer, we modify the IDMEF data model when designing the alert class. This includes adding some new classes (Cluster class, for example) and attributes related to MANETs, deleting some unwanted classes (User class, Process class, etc.) and attributes, and modifying the definition of some classes and attributes (Location attribute, etc.). The alert class hierarchy for AD-CLUSTER using the UML notation. The alert class hierarchy is general in AD-CLUSTER. That is, it can be used as both the input and output of the LACE and GACE for better interoperability. When generating an alert, the detection engine formats it according to the class hierarchy. We generate all of our local alerts compliant with this format.

Aggregation Algorithm: The performance of the aggregation algorithm depends heavily on the performance of the local detection model, the amount of information and the accuracy of the information it provides. In MANET environment, the possible alert burst may crash gateway nodes. Our cases are different. First, due to the lack of misuse based MANET IDS; we cannot assume the accurate identification of attackers provided by local IDSs. Second, the gateway nodes execute the aggregation algorithm periodically. At each time period, the aggregation algorithm aggregates the received local alerts and makes final decisions. If there is no alert received in the last period, no action is taken. This is computationally efficient since it avoids executing the algorithm every time an alert is received. Old gateway nodes can locally broadcast historical records. This can lead to the quick learning of new gateway nodes and thus quick response to intrusion, but requirs more bandwidth cost. Also, a new gateway node can obtain information quickly from local IDSs from their locally broadcasted alerts. Each node has the LACE and GACE module. They use different sources as the alert inputs: the input of the LACE is the local detection engines, while the input of the GACE is either the local LACE (to intra-cluster nodes) or the intra-cluster nodes in the same cluster and the neighboring gateway nodes (to gateway nodes). When a local node detects an anomaly, it generates an alert based on the proposed MANET IDMEF data model. This alert contains the identification of the node, the alert classification, the time information, and the information of the routing control packets in the recent history that could contribute to local alerts. The routing control packets in a given time interval are not sufficient for the intrusion detection. In our attack model, the local broadcast message also includes the local history of the aggregated routing control packets, i.e., how many routing control packets are received and from which node these control packets are sent out. This could help the gateway node make the final decision. Much information could be provided by the local alerts. We introduce a parameter P-routing abnormal. If the proportion of routing control packets from a certain address exceeds P-routing abnormal, it is abnormal and deserves further investigation. However, in normal cases, it is still possible that a node receives a high percentage of routing control packets from some certain node in a period, such as in the initial routing discovery period. This is the main reason to cause the false positive alarms of our aggregation algorithm. When there exist attackers in the network, things are different. The attacker would send many falsified routing control packets into the network to effectively disrupt the routing logic of the network. The local IDSs of the victims, using the Markov chain detection model described in the previous section, could generate the alerts and record the source and destination distribution of the routing control packets in the last period. The attacker's address would dominate the source distribution of the routing control packets. Having gathered this information in the last period, the gateway nodes could know the source address distribution of the routing control packets. If the probability of a particular source address exceeds some predefined threshold P, this address is then identified as the attacker's address. Note that an attacker cannot use different IP addresses to send out fake messages. Otherwise, it can be detected easily by its neighbors. We now discuss how to decide P . The selection of P depends on attack intensity, attack time, node placement, etc. If the threshold P is low, the gateway nodes could identify

the attack more accurately, thus achieving higher detection ratios. However, this could lead to high false positive ratios. If the threshold P is high, the gateway nodes could miss the attack, but reduce the false positive ratio. We propose a simple approach to decide P in the following way. In normal cases, for a given gateway node, if local alerts are received in a given time period, we first pick those source addresses whose aggregated probability is larger than the parameter P-routing abnormal. We denote these probabilities as $P_{t_i} (i = 1, 2, \ldots, n_t)$. Suppose for a given gateway node G, it has m_t time periods in which it receives local alerts, we compute the average of P_{t_i}, $(i = 1, 2, \ldots, n_t)$ over these m_t periods as:

$$P_{G_t} = \sum_{j=1}^{m_t} \sum_{i=1}^{n_t} P_{t_i} / m_t$$

P_{Gt} represents, to gateway node G, the irregularity of the source address distribution of the routing control packets when the system is at normal status. Given a test trace, we compute its average over all gateway nodes:

$$P_{test} = \sum_{\forall\ gateway\ \ nodes} \frac{P_{G_t}}{the\ number\ of\ \ gatewey\ nodes}$$

Given the trace of intrusive activities, we first compute the attack address distributions contained in the routing control packets. We denote these probabilities as $P_{a_i} (i = 1, 2, \ldots, n_a)$.

Suppose for a given gateway node G, it has m_a time periods in which it receives local alerts, we compute the average of P_{a_i}, $(i = 1, 2, \ldots, n_a)$ over these m_a periods as:

$$P_{G_a} = \sum_{j=1}^{m_a} \sum_{i=1}^{n_a} P_{a_i} / m_a$$

P_{G_a} represents the source address distribution of the routing control packets in the gateway node G during the attack time. Given the trace of intrusive activities, we compute the average of P_{G_a} over all gateway nodes:

$$P_{attack} = \sum_{\forall\ gateway\ \ nodes} \frac{P_{G_a}}{the\ number\ of\ \ gateway\ nodes}$$

$$P = h_t * P_{test} + h_a * P_{attack} \quad , h_t > 0, h_a > 0\ and\ h_a + h_t = 1.$$

7. SIMULATION STUDY

7.1 Simulation Model

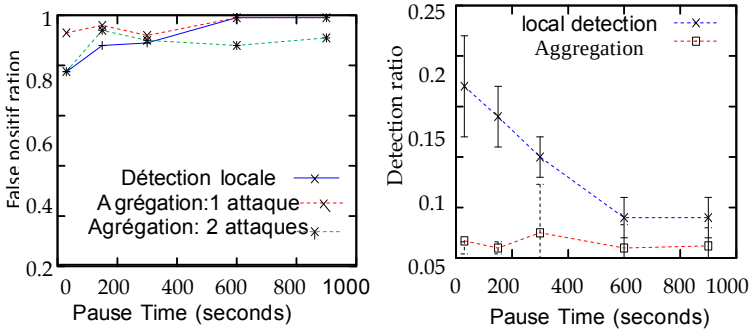
Simulation Platform and Parameter Settings: We use a simulation model based on GloMoSim to investigate the performance of the proposed approaches. In the radio model, capture effects are taken into account. We use the Distributed Coordination Function (DCF) of IEEE 802.11 for wireless LANs as the MAC layer protocol. It has the functionality to notify the network layer about link failures. We enable the promiscuous receive mode of nodes, which enables every received packets delivered to the network layer. In the simulation, 30 mobile nodes move in a 1000 meter X 500 meter rectangular region. In our simulation, the minimal speed is 3 m/s, and the maximal speed is 5 m/s. We change the pause time from 30 seconds to 900 seconds to investigate the performance influence at different mobilities. 8 source-destination pairs are selected randomly to generate Constant Bit Rate (CBR) traffic as the background traffic. The interval time for data transmission is 0.25 second. The size of all data packets is set to 512 bytes. A packet is dropped when no acknowledgement is received after seven retransmissions or when there is no buffer to hold the packet. The buffer size is set to 128 packets. 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.

7.2 Simulation Results

We show that the local IDS constructed using feature PCH demonstrates better performance results than that constructed using feature PCR. Therefore, we use the local IDS constructed using PCH in this section to illustrate the simulation result.

False positive ratio: We compute the false positive ratio of the aggregation algorithm based on the same test data used by the local Markov detection model for the purpose of comparison. If in the last time period, a gateway node receives no local alerts, it will take no action. As shown in Fig. 2(a), the aggregation algorithm achieves much lower false positive ratios compared to that of the local IDS.

Detection ratio: We measure the detection ratio from dividing the number of gateway nodes that actually generate alarms by the number of gateway nodes that should generate alarms. The result is illustrated in Fig. 2(b).



(a) False positif ration (b) Detection ratio

Figure 2: performance of AD-Cluster

8. CONCLUSION AND FUTURE WORK

This paper presents the design of a cluster-based intrusion detection system for mobile ad hoc networks. Based on a local Markov chain based anomaly detection engine, an aggregation algorithm for AD-CLUSTER is presented and a suitable MANET IDS alert data model is described. Using the routing disruption attack as the threat model, we have carried out extensive simulation studies and demonstrated the effectiveness of our system.

9. REFERENCES

[1] Özleyiş Ocakoğlu, Burak Bayoğlu, Albert Levi, Özgür Erçetin and Erkay Savaş, *A Probabilistic Routing Disruption Attack on DSR and Its Analysis,2004.*

[2] D. Curry and H. Debar, *"Intrusion Detection Message Exchange Format Data Model and Extensible Markup Language (XML) Document Type Definition,"* Internet Draft, June, 2002.