AN INTRUSION DETECTION SYSTEM IN MOBILE ADHOC NETWORKS

S.Madhavi,
Research Scholar, Dept of Computer, Science & Engineering,
Acharya Nagarjuna University, India.

Dr. Tai Hoon Kim,
Professor , Dept. of Multimedia, Hannam University, Korea.

Abstract

Networks are protected using many firewalls and encryption software's. But many of them are not sufficient and effective. Therefore an intrusion detection system (IDS) is required that monitors the network, detects misbehavior or anomalies and notifies other nodes in the network to avoid or punish the misbehaving nodes. Numerous schemes have been proposed for Intrusion Detection and Response Systems, for Ad hoc networks. The ultimate goal of the security solutions for wireless networks is to provide security services, such as authentication, confidentiality, integrity, anonymity, and availability, to mobile users. In this paper, we examine the vulnerabilities of wireless networks and argue that we must include intrusion detection in the security architecture for mobile computing environment. We propose an mIDS (Mobile Intrusion Detection System) suitable for multi-hop ad-hoc wireless networks, which detects nodes misbehavior, anomalies in packet forwarding, such as intermediate nodes dropping or delaying packets. mIDS does rely on overhearing packet transmissions of neighboring nodes. Simple rules are designed to identify the misbehavior nodes. A special node called a monitor node carries out the process of identifying the misbehavior node.

1. Introduction

The nature of mobility for mobile networks needs additional mechanisms for providing security. These vulnerabilities do not exist in a fixed wired network. Therefore, the traditional way of protecting networks with firewalls and encryption software is no longer sufficient. We need to develop new architecture and mechanisms to protect the wire-less networks and mobile computing applications. Hence, in this paper, we discuss how to identify the intrusion after an anomaly is reported. Simple rules are applied to identify the intruder information and detect the type of the attack. A node called the Monitor node carries the identification process. This node overhears the channel and detects the misbehavior nodes. There may be more than one monitor node in the whole network. Periodically the monitor nodes are elected in the network.

The rest of the paper is organized as follows Section 2 outlines various challenges. Section 3 gives an overview of the attack models. Section 4 presents the Intruder Detection Methods and identification rules. Sections 5 gives a set based intrusion detection method. Section 6 outlines the possible responses to the identified attacks. Finally, we conclude our paper with some plans for future research.

2. Challenges

Unlike in fixed networks the mobile Ad Hoc networks needs more security mechanisms. Attackers may intrude into the network through the subverted nodes. The network topology is highly dynamic as nodes frequently join or leave the network, and roam in the network. In spite of its dynamic nature, mobile users request security services as they move from one place to another. Hence, a powerful security solution is required to achieve protection and high network performance.

The security solution should protect each node in the network and the security of the entire networks relies on the collective protection of all the nodes. The security solution should protect the network from both the inside and outside intruders into the system. The security scheme adopted by each device has to work within its own resource limitations in terms of energy supply, communication capacity, and memory and computation capability.

Each security solution encompasses all three components of prevention, detection, and reaction. However, an attacker succeeds in infiltrating the security system and causes them to misbehave. Node misbehavior can result in degradation of network performance.

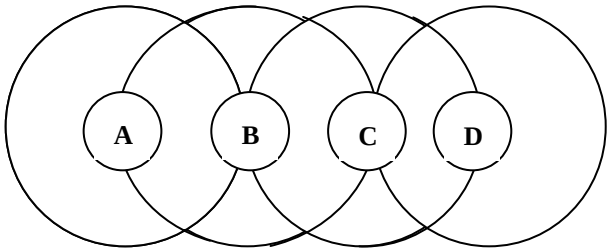

Figure. 1. Nodes in the network and their transmission ranges.

Hence, the system should be monitored for any anomalies and take necessary actions if an anomaly is detected. A system performing these tasks is known as an intrusion detection system (IDS). Ideal IDS should be able to set thresholds for its detection schemes dynamically so that misbehaving nodes cannot easily work around the detection scheme. An attacker may find certain loopholes in the current IDS and tries to attack.

Hence these types of flaws in the basic operations must be recognized and raise the security level. The attacker identity must be reported by the IDS. Each monitor node should invoke the security mechanisms when ever necessary and possible.

3. mIDS Attack

Attacks in Mobile Adhoc networks can be categorized as follows.

1. Unfair use of the transmission channel (ATTACK1).
2. Anomalies in Packet Forwarding (ATTACK2).

3.1.1 Unfair use of the transmission channel (ATTACK1)

A node can prevent other nodes in its neighborhood from getting fair share of the transmission channel. This misbehavior can be considered as DoS attacks against the competing neighbors in a contention-based network since the competing neighbors are deprived of their fair share of the transmission channel. Some of the possible methods for unfair use of the transmission channel are as follows:

1. Ignoring the MAC protocol.

Protocols like 802.11 uses RTS and CTS to notify the immediate neighbors of how long the transmission channel will be reserved for the successful transmission. Such methods minimizes collisions among competing neighbors and try to ensure that all the competing neighbors can get some share of the common channel. But a misbehaving node can generate RTS/CTS at an unacceptable rate by ignoring the back off mechanism. Hence the competing neighbors cannot get an adequate share of the transmission channel. This imposes a long delay at the output queues and they finally time out and get removed.
If the indicated duration (Ti) is less than the actual duration (Ta) taken for successful transmissions, the transmission channel will remain occupied for an additional period, Ta - Ti. The competing neighbors may not be aware of this additional hidden period. Therefore, neighbors trying to access the channel within the hidden period are likely to face unexpected collisions, increase their backoff intervals and hence may not get their share of the channel.

2. Jamming the transmission channel with garbage.

Garbage can consist of packets of unknown formats, violating the proper sequence of a transaction (e.g. sending a data packet without exchanging RTS and CTS) or simply random bits used as static noise by misbehaving nodes. Garbage data may result in too many collisions, may consume a significant part of the available Channel capacity or both. Consequently, legitimate neighbors may not be able to access the channel properly when needed.

3. Ignoring the bandwidth reservation scheme.

Nodes in a multi-hop wireless network reserves a slot for transmission channel before initiating a flow. If there is not enough bandwidth, new flows should not be admitted so that existing flows are not choked. A misbehaving node may not abide by this rule and try to push out packets when there is not enough bandwidth left. As a result legitimate nodes may not get fair share of the transmission channel.

4. Malicious flooding: Deliver unusually large amount of data of control packets to the whole network or some target nodes.

5. Network Partition: A connected network is partitioned into k (k >= 2) sub networks where nodes in different sub networks cannot communicate even through a route between them actually does exist.

6. Sleep Derivation: A node is forced to exhaust its battery power.

3.1.2. Anomalies in packet forwarding (ATTACK2)

Anomalies in packet forwarding take the following forms:

- Drop packets: A node may disrupt the normal operation of a network by dropping packets [22]. This type of attack can be classified into two types: (a) Black hole attack and (b) Gray hole attack.
  Black hole : All traffic are redirected to a specific node X. X will never forward any traffic at all. In a black hole attack, a misbehaving node drops all types of packets (both data and control packets).
  Gray hole attack: An attacker selectively drops data packets. We refer to this attack as ATTACK2a.
- Delay packet transmissions: A node can give preference to transmitting its own or friends' packets by delaying others' packets. As a result, some flows may be not being able to meet their end-to-end delay and jitter requirements. ATTACK2b and ATTACK2c refer to the attacks related to delay and jitter requirements respectively. If these anomalies are not detected, nodes may still use the offending node/nodes in their routes to connect to the remote parts of the network and may not achieve required QoS.
- Wormhole: A tunnel is created between two nodes that can be utilized to secretly transmit packets.
- Packet dropping. A node drops data packets that are supposed to forward.
- Routing Loop : A loop is introduced in a route path
- Denial – of - Service: A node is prevented from receiving and sending data packets to its destinations.
- Fabricated route messages: Route messages with malicious contents are injected into the network.
- False Source Route : An incorrect route is advertised into the network , setting the route length to be 1 regardless where the destination is.
- Maximum sequence: Modify the sequence field in control messages to the maximum allowed value.
- Cache Poisoning : Information stored in routing tables is either modified , deleted or injected with false information
- Selfishness: A node is not serving as a relay to other nodes.
- Rushing: This can be used to improve fabricated route messages.
- Spoofing : Inject data or control packets with modified source addresses.

mIDS proposes an attack model for the mobile adhoc networks that primarily concentrates on the above mentioned attacks It continuously monitors for these attacks. Whenever the anomaly is reported , mIDS protects the system, without compromising its effectiveness.

4. Intrusion detection methods and related work.

Intrusion detection systems can be classified broadly into two classes:

- Reputation based schemes.
- Incentive based approaches.

Reputation based schemes detect misbehaving nodes and notify other nodes of the misbehaving nodes. Incentive based approaches aims to promote positive behavior to foster cooperation instead of relying on participants to report and punish misbehaving nodes. mIDS is a reputation-based system. Zhang et al. [8][12] have developed a distributed and cooperative intrusion detection system (IDS). The authors have detailed intrusion detection methods for the following attacks:

(a) Identifying False route entry in a node's route and

(b) Random packet dropping by intermediate nodes.

The random packet dropping detection scheme relies on overhearing transmissions of neighboring nodes. Bhargava and Agrawal [13] have extended the IDS model described in [8] to enhance the security in AODV (Ad-hoc on demand Distance Vector [14]) routing protocol. Watchdog [18] proposes to monitor packet forwarding on top of source routing protocols like DSR. Watchdog has the limitations of relying on overhearing packet transmissions of neighboring nodes for detecting anomalies in packet forwarding. It assumes symmetric bidirectional connectivity: if A can hear B, B can also hear A. Since the whole path is specified, when node A forwards a packet to the next hop B, it knows B's next hop C. It then overhears the channel for B's transmission to C. If it does not hear the transmission after a timeout, a failure threshold associated with B is increased. If the threshold exceeds a maximum value, A sends a report packet to the source notifying B's misbehavior. Reference [19] follows the same concept but works with distance vector protocols such as ADOV. Each node knows about its correct next hop neighbors. It also considers more types of attacks, such as packet modification, packet duplication, and packet-jamming DoS attacks. Bal Krishnan [21] has proposed a way to detect packet dropping in ad-hoc networks that addresses the problems of receiver collisions, limited transmission power and directional antennas.

4.1 Detection methods.

Various methods are proposed to detect the intrusion identity.

The following are the notations used in such methods

Numberof_in(m): the number of incoming packets on the monitored node m.

Numberof_out(m): the number of outgoing packets from the monitored node m.

Numberof_out([m]): the number of outgoing packets of which the monitored node m is the source.

Numberof_in([m]): the number of incoming packets of which the monitored node m is the destination.

Numberof_in([s];m): the number of incoming packets on m of which node s is the source.

Numberof_out(m;[d]): the number of outgoing packets from m of which node d is the destination.

Numberof_out(m;n): the number of outgoing packets from m of which n is the next hop.

Numberof([s];M;m), the number of packets that are originated from s and transmitted from M to m.

Numberof([s];M;[m]), the number of packets that are originated from s and transmitted from M to m, of which m is the final destination.

Numberof([s];[d]), the number of packets received on the monitored node (m) which is originated from s and destined to d.

The detection methods are as follows.

1) Unconditional Packet Dropping.

FP (Forward Percentage) $FPm = \frac{\text{packets actually forwarded}}{\text{packets to be forwarded}} \rightarrow (1)$

FP determines the ratio of forwarded packets over the packets that are transmitted from M to m and that m should forward. If the denominator is not zero and FPi = 0, the attack is detected as Unconditional Packet Dropping and m is identified as the attacker.

2) Random Packet Dropping. If the denominator is not zero and FPm is less than a chosen threshold TH_FP (TH_FP < 1) but not zero, the attack is detected as Random Packet Dropping and node m is identified as the attacker. TH_FP is chosen so that 1 - TH_FP is equal to upper bound of the dropping rate that can be tolerated.

3) Selective (Random) Packet. LFP (Local Forward Percentage)

$LFPs\ m = \frac{\text{packets from s actually being forwarded}}{\text{packets from source s to be forwarded}} \rightarrow (2)$

If the denominator is not zero and the statistics is zero (un-conditional dropping), the attack is unconditional Packet Dropping targeted at s. Likewise, if the LFP is less than TH_LFP (TH_LFP < 1), the attack is random Packet Dropping targeted at s. In either case, m is identified as the attacker.

4) Black hole Monitor the statistics GFP (Global Forward Percentage)

GFPm as the ratio of the total number of packets that are received by M and M should forward to the total number of packets sent by M's 1-hop neighborhood (N(M)) and are not

destined for another neighbor or M over a time period of L.. If all such packets are being absorbed by M for a sufficiently long period, or more precisely, if the denominator is not zero and GFP = 1, then an black hole is detected and M is identified as the attacking or misbehaving node. The detection of black hole may be infeasible if M is malicious and the attacker has total control of M so that the detection modules can be disabled.

4.2. Detecting attack1

mIDS makes the following assumption to detect ATTACK1.

In mobile adhoc networks, the transmission time is divided into contention period and transmission period. Nodes in a multi-hop wireless network use TDMA/S-TDMA to reserve a slot for transmission channel before initiating a flow. Each node gets a chance to transmit at least once during a frame time. A security mechanism can be found where a node has to digitally sign before reserving the slot. Hence the intruders cannot reserve the slot. If there is not enough bandwidth, new flows should not be admitted so that existing flows are not choked. This enables existing flows to achieve their desired Qos. After the contention period, the nodes are allowed to transmit in the same order of their reservation. An intruder may attack a node X and allow it to misbehave. Due to this misbehavior, the performance of the network decreases. Hence a node X after completion of transmitting in its slot time, have to send a special packet identifying its completion of transmission. The predecessor of node X overhears this. If the predecessor node do not hear this special packet after a duration time from X, thinks that X is misbehaving and increments the misbehaving count by one. If the misbehaving count reaches to certain threshold value, then the X is identified as a misbehaving node. The neighbors of node X are reported of this misbehaving node

Alternatively let us assume that for each period T, a node X knows that p% of the available link capacity has been allocated by its neighboring nodes where p = L where L is the total link capacity. L should be less than 100% since no system can work at 100% capacity. Now for each period T, X measures the percentage of link capacity r% being used by the neighboring nodes for the admitted flows. It also measures the percentage of link capacity s% being wasted due to collisions, garbage data and flows that did not reserve bandwidth.

$$\text{If} \quad (r + s) >= L \quad \rightarrow \quad (3)$$

X assumes that, a neighbor or a group of neighbors is accessing the channel unfairly. X increases a non-negative misbehavior counter m_c each time X detects ATTACK1 and decrements it if there is no such misbehavior. If m_c reaches a threshold, X declares its neighborhood misbehaving. Sometimes a neighbor of X may not utilize the whole part of link capacity allocated to an admitted flow. This can happen if the flow does not send packets at a constant bit rate. Hence, r can be less than p. Therefore, r < p does not mean that neighbors are not getting fair share of the channel. However, r < p can also be true if a neighbor does not get fair share of the channel.

4.3 Detecting attack2.

Each node measures the rate Rt [f, h] at which it processes packets , where h denotes the hop distance a node is away from the source. The destination finds the Rt[f, h=destination] and adds it to the a packet and sends it to the source through all the intermediate nodes off.

Each intermediate node appends the rate to RSP and the Rt [f, h] can be digitally signed by its respective node. When RSP reaches the source node, it contains R [f, h] values of all the downstream nodes off. Now we can estimate the forwarding ratio of a node h hops away from the source by the following expression:

Forwarding ratio, $F[f, h] = R[f, h + 1] / R[f, h - 1]$ $\rightarrow$ (4)

If Delivery ratio, $R [f, h = \text{destination}] / R [f, 0] < Rthres [f]$ $\rightarrow$ (5)

Where Rthres [f] is the allowable minimum end-to-end delivery ratio for the flow f, the source suspects the intermediate node, h hops away from the source with the highest F [f, h], is dropping packets at an intolerable rate. The source Nodes towards the destination of a flow are called the downstream nodes.

Forwarding ratio = Data received by the neighboring downstream node/Data sent from the neighboring upstream node

Delivery ratio = Data received successfully/Data Sent

If the misbehavior counter MBC [ATTACK2a, f, h] for each downstream node reaches a threshold, the source declares that node to be misbehaving. The packet dropping can also be detected through contact scheduling. Contact scheduling is assumed while proposing a solution for mIDS. That is a node before transmitting knows the address of all the nodes in the path to the destination. Using this path a node transmits the data. The source node encrypts the message in such a way that the decryption is possible only for the destination node and not to the intermediate nodes.

The intermediate node may be a destination node, hence if a malicious intermediate node misbehaves and drops the packets, it losses ultimately. Hooks and snakes are used to forward the message from source to the destination. This type of scheduling algorithms is to make the intermediate nodes unaware of the address of the destination node's message decryption methods.

As an example a hook shaped path from nodes A, B, C, D, C using onion routing can be constructed as follows

mid;DATAK; [[[[K; eop]PKC ;C]PKD;D]PKC ;C]PKB ;B

Mid stands for the identifier of the message

Eop stands for the end of the path

Pki stands for the public key for the node i

DATA k denots the message encrypted using the secret key K

The message is sent from A to B . The node B decrypts the outer onion layer using the public key pk B and obtains

mid;DATAK; [[[[K; eop]PKC ;C]PKD;D]PKC ;C]

The message is sent from B to C . The node C decrypts the onion layer using the public key pk c and obtains mid;DATAK; [[[[K; eop]PKC ;C]PKD;D]
The message is sent from C to D . The node C decrypts the onion layer using the public key pk c and obtains mid;DATAK; [[[[K; eop]PKC ;C]
The message is sent from D to C , the final destination . The node D decrypts the onion layer using the public key pk D and obtains mid;DATAK; [[[[K; eop]
The node C decrypts the message in DATA k using the secret key K.

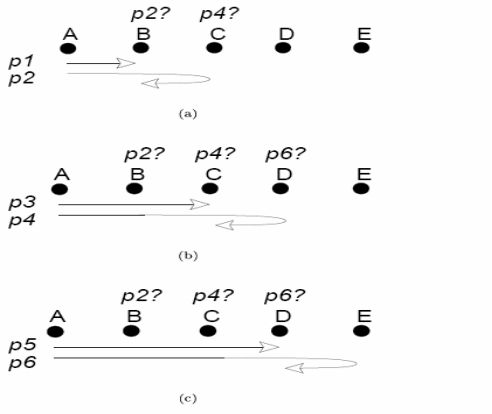

Figure 2.0 A Hook shaped path.

Hence, only the final destination node can know the secret key and decrypts the message. The intermediate nodes do not know the secret key. They only decrypt the message using the public key and forward the data to the next neighbor in the path. However, hooks are advantageous they suffer with a problem of revealing the destination address. Hence, the snakes can be used for forwarding the message to the nodes. In the snake shaped path, the address of the final destination cannot be revealed. The size of the onion should not be revealed to the nodes. Each time the node decrypts the message using its public key, the size of the onion is decremented.

4.4. Detecting attack2b

To detect this ATTACK there are two schemes. The first scheme depends on time synchronization among all nodes. Before sending packets of a flow f the sender puts time-stamps in each packet. A node calculates the delay of each packet from source to destination by subtracting the time-stamp in the packet from its current time. Let the average value be Tavg [f, h] at a downstream node, h hops away from the source. Similar to the detection scheme for ATTACK2a, the destination periodically sends a RSP containing Tavg [f, h=destination] to the source through all the intermediate nodes of “f”. Each intermediate node also appends Tavg [f, h] to the RSP with a digital signature. When RSP reaches the source node, it contains Tavg [f, h] values of all the downstream nodes of “f”. With this information

the source can estimate the average delay Tavg d [f, h], the flow has encountered at an intermediate node h hops away from the source,

I.e. Tavg d [f, h] = Tavg [f, h + 1] - Tavg [f, h - 1] → (6)

If Tavg [f, h = destination] > Tthres [f] → (7)

where Tthres[f] is the allowable maximum end-to-end delay of f, the source assumes the intermediate node, h hops away from the source with the highest Tavg d [f, h], is not complying with the end-to-end delay requirement.

If the misbehavior counter m_c [ATTACK2b, f, h] reaches a threshold, the source declares the node h hops away from the source is misbehaving.
The second scheme is based on measuring round trip delays with probe packets for each intermediate node.

This scheme relies on the following assumptions:
(a) Links are bidirectional,
(b) Transmission and queuing delays in both directions should be almost the same, and
(c) Probe packets should be encrypted on end-to-end basis so that intermediate nodes cannot detect their types and hence cannot treat them differently to remain undetected.

After initiating a flow f, the source node sends periodic probe packets to each of the associated downstream nodes. Once the probe packet reaches its destination X with hop count h, it is sent back to the source with the time Tprocess [f, h] indicating the processing delay at X. The source also measures the round trip delay of the probe packet sent to X.

Let the average round trip delay of probe packets sent to X be TroundTrip [f, h].

Tavg [f, h] denote let the average end-to-end delay up to node X. Tavg [f, h] can be computed as follows:

Tavg [f, h] = TroundTrip [f, h] - Tprocess [f, h] → (8)

Now we can substitute Eq. (8) into Eq. (6) and Eq. (7) in order to detect ATTACK2b.

5. Monitor identification method.

The mobile adhoc network is organized as a collection of such sets and each set has a monitor node. Each monitor node performs intrusion detection. There are many set-based intrusion detection schemes and set formation algorithms. Set formation using various algorithms in 18-node topology (Head Nodes are shown in Red, Gateway Nodes in Yellow and Member Nodes in Black) Consider the figure 4.0 For comparative analysis the 18 node

base topology has been taken from [8]. It is observed that a scheme proposed in [4] results in 9 sets as shown in figure 1b and 5 sets by using scheme proposed in [5] as shown in figure 4c.

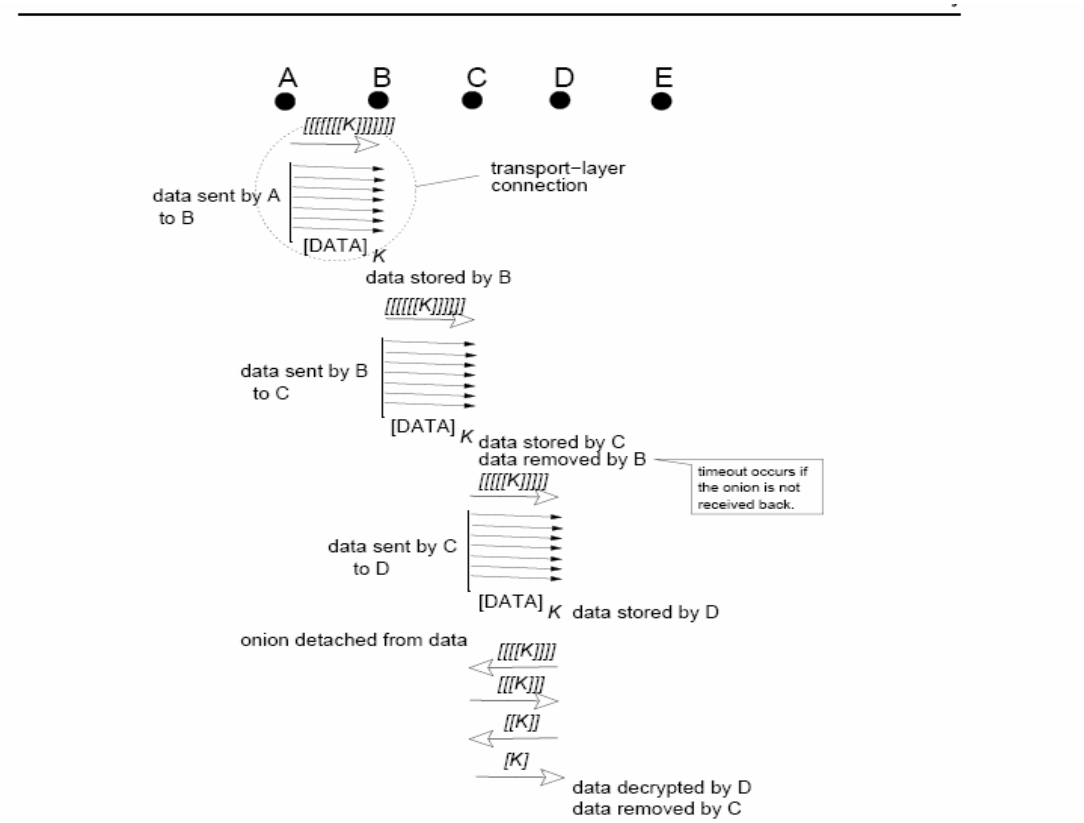

Figure 3.0 Optimized snake shaped path for long messages

Each set have the set head. A node that cannot be reached by anyone else forms a single node set, or SNC. The size of a set is defined as the number of nodes in the set and is denoted as SC. In addition, the monitor electing algorithm is fair, where every node has a chance to serve as a set head.

Firstly, each node i contributes a random value Ri to the input. Then a common selection function is used by all nodes to compute a integer from 0 to SC _ 1 from a total of SC inputs. The output of the election function must have a uniform distribution in [0, SC _ 1].

The selection function we use is simply the modular Exclusive OR (or XOR) function, i.e.,

Sf(R0;R1;R2; :::;RSC_1) = (LSC_1 i=0 Ri) MOD SC.

The random values are fully exchanged within the set (SET) and the selection function is computed in a distributed manner, i.e., on each node, to decide the set head..

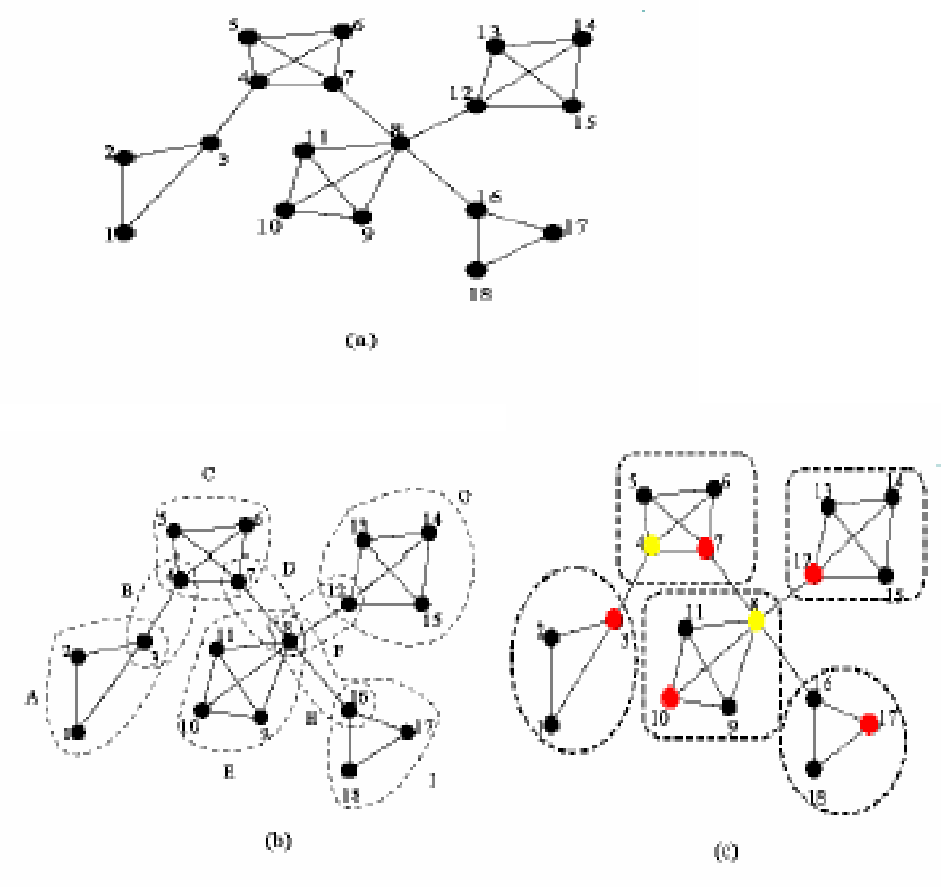

Figure 4.0. Monitor identification method.

5.1 Monitor Election Protocol

The purpose of this protocol is to randomly select one node in the set as the set head. we discuss the behavior on the i-th node.

1. Generate a random integer RI
2. Broadcast a message ELECTION_START to CL0i.
3. On Receiving all ELECTION_START from CL0i, broadcast the message ELECTION=(IDi;Ri) to set CL0i.
4. If T1 is timeout, every node for which ELECTION START has not be received is excluded from CLi.
5. On Receiving ELECTION from node j, verify its hash value matches the value in the ELECTION START message from j. Store Rj locally.
6. If all Rj from CL0i have arrived, compute H=SEL(R0; R1;R2; :::;RSC_1) where SEL is the selection function. Determine the set head H as the h-th node in the SET since all IDs are ordered.
7. If H 6= i (i.e., as a citizen), do the following.
(a) Send ELECTION DONE to H.

(b) Wait for ELECTION REPLY from H, then enter DONE state.
8. Otherwise, as a set head, H performs following.
(a) Setup a timer T2.
(b) On Receiving ELECTION DONE, verify it is from CL0i.
(c) If T2 is timeout, citizens from whom ELECTION DONE has not be received are excluded from CLi. Broadcast ELECTION REPLY to CL0 i and enter DONE state. Once the set head is determined, it copies the SET member list to a citizen list CTC. The sufix C denotes Set Valid Assertion Protocol All nodes should perform the current set controlled by the set head. this assertion periodically in DONE state.

There are two parts in this protocol.

1. Since the network topology of an ad hoc network changes dynamically, connections between the elected set head and some citizens' nodes may be broken from time to time. If a link between a citizen C and a set head Hd has been broken, C will check if it is in another set. If not, it enters LOST state and activates the Set Recovery Protocol. Also, C is removed from Hd's citizen list CTC. If there is no more citizens in set C, Hd becomes a citizen if it belongs to another set. Otherwise, Hd enters LOST state and activates the Set Recovery Protocol.

2. Even if no membership change has occurred, the cluster head cannot function forever. Because it is neither, fair in terms of service and unsafe in terms of the long time single-point control and monitoring.
We enforce a mandatory re-election timeout, Tr. Once the Tr expires, all nodes in the set enters the INITIAL state and start a new set head setup round.
If the SET property still holds, the SET Computation step can be skipped. Set Recovery Protocol In the case that a citizen loses its connection with previous set head or a set head loses all its citizens, it enters LOST state and initiates Set Recovery Protocol to re-discover a new set head.

Again, without loss of generality, we discuss the behavior on the i-th node.

1. A request message ADD REQUEST=(IDi) is broadcast with a timer T3.
2. A set head H receives the request and replies ADD REPLY=(IDH) only after a short delay Td (0.5s in our implementation). The delay is introduced in hope that a connection has been stable for Td can remain to be stable for a long time.
3. Node i replies the _rst ADD REPLY it received, i.e., ADD ACK=(IDi). And enters DONE state. Additional ADD Reply's are ignored.
4. On Receiving ADD ACK, H adds i into its CTC.
5. If T3 is timeout and no ADD REPLY is received, there is no active set head nearby. Node i enter INITIAL state to wait for other lost citizens to form new SETs and elect their new set heads.

6. Reactions .

There are two types of the reaction schemes known as global reaction and end-host reaction. In the former scheme, the malicious node is excluded from the network. On the

other hand, in the end-host reaction scheme, each node may make its own decision on how to react to a malicious node .

Global reaction. The reaction scheme in [19] falls into the global reaction category. It is based on the URSA certification framework [20]. Once multiple nodes in a local neighborhood have reached consensus that one of their neighbors is malicious, they collectively revoke the certificate of the malicious node. Consequently, the malicious node is isolated in the network, as it cannot participate in the routing or packet forwarding operations in the future.

End-host reaction. A node gives ratings to each of its neighbors and slowly increases the rating of well-behaved nodes and decreases the rating of a malicious node. Each node may have a different rating about whether another node is malicious, and each has its independent reaction accordingly. Reference [17] extends this idea with security protection of the routing messages, as discussed earlier.

6.1. Responding to attack1

If the detection module of P succeeds in identifying the misbehaving node M or the direction of misbehavior (DoM), it takes the following actions:

• P notifies the routing module that it should avoid M or the neighbors towards the DoM in the route discovery process for certain duration.

• P notifies the underlying security module to reauthenticate M or all the neighbors towards DoM.

• If P forwards any packet for an ongoing flow to M or towards the DoM, it notifies the source of the flow to find an alternate route that avoids M or the neighbors of P towards DoM. The detection module of P may detect ATTACK1 but may not identify the misbehaving node or the direction of misbehavior. In this case the responses are as follows:

• P notifies the routing protocol to avoid all its neighbors in the route discovery process a certain duration.

• P notifies the underlying security module to reauthenticate all its neighbors.

• If P forwards any packet for an ongoing flow to any of its neighbors, it notifies the source of the flow to find an alternate route that avoids all the neighbors of P.

6.2 Responding to attack2.

If a node P detects a node Q is responsible for ATTACK2 , it takes the following actions:

• Notifies the routing module to find an alternate route avoiding Q for the flow f..

• Notifies its routing module and those of the neighbors of Q to avoid Q for a certain period for any new flow that has similar QoS requirements as f.

7. Conclusion .

We have proposed IDS, referred to as mIDS, for wireless networks. mIDS can detect if nodes are getting their fair share of the transmission channel. It also detects packet drops or delays that violate the respective flow requirements. mIDS rely on overhearing packet transmissions of neighboring nodes that makes it an effective system in networks where nodes use different transmission power and directional antennas for different neighbors. mIDS does not require setting up various thresholds manually; rather it can select them dynamically. In future we want to implement all the causes of ATTACK1 and ATTACK2. We would also like to implement a fully functional response system that would incorporate all the features outlined in Sections 4,5,6.

8. References

[1]. S.Madhavi , "An Intrusion Detection System for Mobile AdHoc Networks". In Procedings of IEEE Xplore International conference on Security And Applications(ISA) , Korea, April 2008.
[2]. S.Madhavi and Dr I. Ramesh Babu , " Security in Mobile AdHoc Netorks : Challenges and Solutions", in Journal of Computer Science , Karpagam Publications , India, Vol 1 Issue 6 Sept-Oct 2007.
[3] D. Denning. An intrusion detection model. IEEE Transactions on Software Engineering, 13(2), Feb 1987.
[4] Yi-an Huang, Wenke Lee, "A Cooperative Intrusion Detection System for Ad Hoc Networks", in Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks (SASN), Fairfax, Virginia, October 31, 2003.
[5] Kashan Samad, Ejaz Ahmed, Waqar Mahmood, "Simplified Clustering Scheme for Intrusion Detection in Mobile Ad Hoc Networks", 13th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, September 15-17, 2005.
[6] P. Krishna, N. H. Vaidya, M. Chatterjee, D. K. Pradhan: "A cluster-based approach for routing in dynamic networks", ACM SIGCOMM Computer Communication Review, 27(2):49-64, 1997.
[7] S. Cheung and K. Levitt. Protecting routing infrastructures from denial of service using cooperative intrusion detection. In New Security Paradigms Workshop, 1997.
[8] Y. Zhang and W. Lee. Intrusion detection in wireless ad-hoc networks. In Mobile Computing and Networking, pages 275–283, 2000.
[9] R. Ramanujan, A. Ahamad, J. Bonney, R. Hagelstrom and K. Thurber. Techniques for Intrusion-Resistant Ad hoc Routing Algorithms (TIARA). In MILCOM, volume 2, pages 660–664, Oct 2000.
[10] S. Marti and T. J. Giuli and K. Lai and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In Mobile Computing and Networking, pages 255–265, 2000.
[11] L. Butty´an and J. P. Hubaux. Enforcing service availability in mobile ad-hoc WANs. In IEEE/ACM MobiHOC, pages 87–96, Aug 2000.
[12] Y. Zhang and W. Lee and Y. Huang. Intrusion detection techniques for mobile wireless networks. In ACM/Kluwer Mobile Networks and Applications (MONET), volume 8, pages 545–556, Sep 2003.
[13] S. Bhargava and D. P. Agrawal. Security Enhancements in AODV protocol for Wireless Ad Hoc Networks. In VTC, volume 4, pages 2143–2147, fall 2001.
[14] C. Perkins and E. Royer. Ad-hoc On-Demand Distance Vector Routing. In 2nd IEEE Workshop on Mobile Computing Systems and Applications, pages 90–100, Feb 1999.
[15] P. Krishna, N. H. Vaidya, M. Chatterjee, and D. K.Pradhan. A set-based approach for routing in dynamic networks. ACM SIGCOMM Computer Communication Review, 27(2):49{64, 1997.
[16] C. Krugel and T. Toth. Flexible, mobile agent based intrsuion detection for dynamic networks. In European Wireless, 2002.
[17] D.B. Johnson and D.A. Maltz. Dynamic Source Routing in Ad-hoc Wireless Networks, chapter 5, pages 153–181. 1996.
[18] S. Marti et al., "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," ACM MOBICOM, 2000.
[19] H. Yang, X. Meng, and S. Lu, "Self-Organized Network Layer Security in Mobile Ad Hoc Networks," ACM Wise, 2002.
[20] J. Kong et al., "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," IEEE ICNP, 2001.
[21] K. BAL Krishnan, J. Deng, P. K. Varhney. TWOACK: Preventing Selfishness in Mobile Ad Hoc Networks. In IEEE WCNC, Mar 2005.

This work was supported by a grant from Security Engineering Research Center of Mini stry of Knowledge Economy, Korea

Authors

S.Madhavi is pursuing PhD from Acharya Nagarjuna University. She received the M.S. Degree from B.I.T.S, Pilani in 1999. She is presently working as an Asst Professor, Department of Computer Science and Engineering, P.V.P.Siddhartha Institute of Technology, affiliated to Jawaharlal Nehru Technological University. She has guided 40 postgraduate student projects. She has Published 9 papers (International & National Journals / Conference proceedings) and had Academic Participation in 11 International workshops / conferences. Her area of interest is including Network security and channel scheduling in Mobile Ad hoc Networks.

Dr. Tai Hoon Kim received his M.S. degrees and Ph.D. in Electrics, Electronics & Computer Engineering from the Sungkyunkwan University, Korea. He worked as researcher at Technical Institute of Shindoricoh , as a senior researcher at the Korea Information Security Agency , at the DSC (Defense Security Command) , as a research professor at E-wha Woman University and now he is currently a professor of Hannam University. He wrote sixteen books about the software development, OS such as Linux and Windows 2000, and computer hacking & security. And he published about 150 papers by 2007. He was a General Chair of ICHIT 2006, MUE 2007 and ISA 2008, Steering Committee Chair of FBIT 2007, IPC 2007, FGCN 2007 and MUE 2008, and Publicity Chair of JRS 2007. Now he is a Steering Committee Chair of FGCN 2008, ASEA 2008, SecTech 2008, BSBT 2008 and UNESST 2008. He was a Guest Editor of AJIT and FGCS Journal.