A System for Power-aware Agent-based Intrusion Detection (SPAID) in Wireless Ad Hoc Networks

T.Srinivasan[1], Jayesh Seshadri[2], J.B.Siddharth Jonathan[3], Arvind Chandrasekhar[4],

Department of Computer Science and Engineering,
Sri Venkateswara College of Engineering,
Sriperumbudur, India 602105
tsrini@svce.ac.in[1], jayeshs2000@yahoo.co.in[2],
jonathansiddharth@yahoo.co.in[3], arvindcac@gmail.com[4]

Abstract. In this paper, we propose a distributed hierarchical intrusion detection system, for ad hoc wireless networks, based on a power level metric for potential ad hoc hosts, which is used to determine the duration for which a particular node can support a network monitoring node. We propose an iterative power-aware, power-optimal solution to identifying nodes for distributed agent-based intrusion detection. The advantages that our approach entails are several, not least of which is the inherent flexibility SPAID provides. We consider minimally mobile networks in this paper, and considerations apt for mobile ad hoc networks and dynamism issues are earmarked for future research. Comprehensive simulations were carried out to analyze and clearly delineate the variations in performance of our approach with changing density of wireless networks, and the effect of parametric variations such as hop-radius.

1 Introduction

An intrusion is defined as "any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource" [1]. Several algorithms have been published in recent years to deal with intrusion detection, which incorporated the essence of the *wireless* nature of wireless ad hoc networks. Intrusions in wireless networks amount to *interception, interruption, or fabrication* of data transmitted across nodes. Intrusion into a wireless network is possible if an intruder node attempts to access data unauthorized for itself. *Ad hoc networks* are particularly prone to such dangers, considering the *dynamism* and *essentially geographically distributed* nature of the nodes. Ad hoc networks can be hence classified on their dynamism, i.e., as minimally mobile or highly mobile. In this paper, we primarily focus on minimally mobile networks, where power levels of the nodes are absolutely critical in determining the kinds of processes they can run in a sustainable fashion.

We briefly discuss PLANE, a metric we suggest for comparing power-levels across nodes for running agent-based network monitoring processes. Agent-based systems are inherently reconfigurable, since the agents can easily be migrated to other hosts, and are by themselves lightweight, and thus suit the power sensitive nature of the networks, such as wireless sensor networks. For a complete analysis of possible network threats to general ad-hoc networks, refer [2]. We adopt the hierarchical model proposed in [5], and extend the model to include power awareness of individual nodes in SPAID.

2 Existing Approaches

Tackling the intrusion problem can typically be done by adding additional intrusion detection layers on top of the protocol, or through alterations to the wireless protocol itself. For the former style of enforcing security, typically two types of intrusion detection systems (IDS) are used, as a reminiscence of wired intrusion detection techniques [5].

2.1 Network based systems

Network-based systems (NIDS) can be passive or active systems, listening in on network traffic, and capture and examining individual packets flowing through a network NIDS can analyze across all layers of the network protocol and are able to look at the payload within a packet, to see which particular host application is being accessed, and with what options, and raise alerts when an attacker tries to exploit a bug in such code, by detecting known attack signatures. NIDS often required dedicated hosts or special equipment, and thus can be prone to network attacks. For further considerations please refer [6,7].

2.2 Host based systems

Host-based intrusion detection systems [8, 9] monitor each individual host, by running on each host. They are able to detect actions such as repeated failed access attempts or changes to critical system files, and normally operate by accessing log files or monitoring real-time system usage [5].To ensure effective operation, host IDS clients have to be installed on every host on the network, tailored to specific host configuration. Host-based systems require dedicated processes to run for network monitoring, and, as their name suggests, are not bandwidth dependent. The disadvantage such comprehensive host-based systems is that they can considerably slow down the hosts that have IDS clients installed.

To circumvent these problems agent-based lightweight models were proposed for wireless networks, which are more bandwidth efficient, and provide an heuristic approach to intrusion detection. Our approach combines the approach in [5] of providing a hop-based hierarchical agent -based model with our own approach for power-awareness in selection of nodes.

2.3 "Secure" Protocols for wireless ad hoc networks

Protocol-based security measures provide for encryption mechanisms and other extensions such as one-way hash chains using in [4], to deal with routing updates attacks. Some approaches also suggest symmetric cryptographic methods to alter the MAC sub-layer to improve security [10]. Research on secure versions of existing protocols such as link state protocols [13] and variations of distance vector protocols [4] have been performed. It is clear that such "secure" additions correspond to an increase in rigidity of wireless ad hoc networks, which in essence, curtail their usefulness. Certain protocols such as that in [4] seem to improve on the base protocol but comparisons with other approaches is still in its infancy and increased cryptographic overhead in some applications are not be justified. For examples of protocol-based security extensions and measures refer [3], [4], [11], [12] and [13].

3 Preliminary Considerations for SPAID

The agent-based model proposed in [9] approaches the IDS problem with an approach that handles intrusions with an agent running on each system. Further their suggestion of statistical methods for classifying network data seems to have been proven to be inappropriate, in light of the Support Vector Machine (SVM) based model suggested in [14]. Further, the model in [9] is not suitable for a power-aware IDS, since such a system warrants energy consumption in systems irrespective of their current battery levels, i.e., it suggests an IDS without considering the feasibility of the assumption that network monitoring and analysis is justified in nodes with minimal power, such as robust wireless sensor networks (WSN).

3.1 Modular IDS Architecture

The IDS we propose is built on a mobile agent framework. It is a non-monolithic system and employs several sensor agents that perform certain functions, such as:

- *Network monitoring*: Only certain nodes will have sensor agents for network packet monitoring, since we are interested in preserving total computational power and battery power of mobile hosts.
- *Host monitoring:* Every node on the mobile ad hoc network will be monitored internally by a host-monitoring agent. This includes monitoring system-level and application-level activities.
- *Decision-making:* Every node will decide on the intrusion threat level on a host-level basis. Certain nodes will collect intrusion information and make collective decisions about network level intrusions.
- *Action:* Every node will have an action module that is responsible for resolving intrusion situation on a host (such as locking-out a node, killing a process, etc).

A hierarchy of agents has been devised in order to achieve the above goals. We will adapt the hierarchy for our purposes. There are three major agent classes as used in [5], are categorized as monitoring, decision-making and action agents. Some are present on all mobile hosts, while others are distributed to only a select group of nodes,

as discussed further. Monitoring agent class consists of packet, user, and system monitoring agents. The following diagram shows the hierarchy of agent classes.

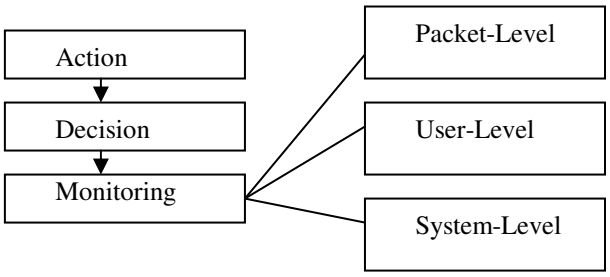

Fig. 1. Typical agent hierarchy, depicting the multi-level decision making process for intrusion detection.

3.2 Agent distribution

As mentioned above, not all the nodes on a wireless ad-hoc network will host all types of IDS agents. To save the resources, some of the functionality must be distributed efficiently to a (small) number of nodes. The modular architecture we use mimics the architecture [5]. The decision making module incorporates the energy metric, Power Loss/Availability for Network-monitoring Estimate (PLANE), a node-specific measure of the mean power loss per node for running the network monitoring agent. PLANE is directly related to the wireless protocol used, mean number of wireless links for the specific node, average node maintenance energy consumption, and finally the battery power (energy) remaining. PLANE ultimately estimates the duration the node can last on the same power without replenishment if the network monitoring agent were run on it. To calculate the power consumption metrics such as those in [15] are often used. The reception costs are multiplied by the number links for the node to yield an average reception cost, to which we add the average sending cost of a message is added. These calculations though, are highly dependent on the density of the network, and the routing/link exchange protocols used.

3.3 Calculating PLANE

The calculation of PLANE involves calculating the duration for which the node can continue to support a network monitor along with its normal operations. We therefore calculate PLANE by calculating time for which node can last as the network monitoring node as follows:

$$PLANE = \frac{BPR}{TEC_{nm}} \quad (1)$$

Where BPR is the total battery power remaining at the instant of determination of the network monitoring node selection algorithm, i.e., SPAID (Section 4), and TEC_{nm} is the total energy consumption with network monitoring node processes running. In the absence of measurement of exact networking monitoring energy consumption, we assume PLANE as PLANE'. The value PLANE' is typically available directly from most distributed wireless networks, such as sensor networks, and hence finds presence in the above calculation.

$$PLANE' = \frac{BPR}{TEC} \tag{2}$$

TEC is the total energy consumption before the node being selected for network monitoring. It is to be noted that PLANE can be tailored to suit the needs of the type of network monitoring required and the nature of the actual node on which it runs. We shall not deal in further detail with PLANE in this paper, but rather focus our attention on the iterative algorithm for network monitoring node selection. TEC values are represented by idle wireless nodes running in ad hoc mode which consume between 741 mW and 843 mW[15].

4 The SPAID Algorithm

In SPAID, we deal with multi-hop network monitoring clustered node selection. This type of a node selection has its inherent advantages in allowing complete coverage of all nodes and links in a network, but with a factor of redundancy incorporated in the collection of intrusion detection data. Additionally, by varying the hop-radius of the algorithm and the PLANE/Topology constraints, sufficient redundancy in overlap of monitored nodes can be achieved, which allows us to prune the set of nodes selected for network monitoring. Considering that we are dealing with minimally mobile wireless ad hoc networks, topological changes shall not be considered in PLANE evaluation, and deemed to be constant during the process of selection of a network monitoring node.

4.1 SPAID Node Selection algorithm for Network monitoring nodes

The SPAID algorithm uses the agent hierarchy presented in Fig.1, with a significantly adapted node selection to incorporate power-awareness, and is best detailed by the following 6 steps.

Step 1: Set PLANE Constraint/ Topology Constraint .Set a constraint on the PLANE value of nodes which are allowed to compete for becoming a Network monitoring node. These depend upon the duration for which the topology is expected to be unchanged, and IDS active duration. Further, certain nodes which have very small number of adjacent nodes may be discarded by setting the Topology Constraint.

Step 2: PLANE Calculation and PLANE Ordered List (POL). Arrange the different nodes in increasing values of PLANE as calculated previously, for all nodes which satisfy the PLANE Constraint. This implies that nodes that can last longer as a network monitoring node takes higher precedence in consideration for selection.

Step 3: Hop Radius. Set the hop radius to one initially, and increment for each insufficient node selection with the current hop radius.

Step 4: Expand Working Set of Nodes. Consider node selection incrementally, initially from the first node, (node with highest PLANE), to finally the set of all nodes in the network, incrementing the set of nodes under consideration by one node each time. We call this set the working set (WS) of nodes. The WS is expanded only if the addition leads to an increase in number of represented nodes.

Step 5: Voting. We use the voting system for Node Selection, as used in [5], except that, we limit the candidates to just the nodes which are part of WS. Under this voting system, each node votes for that node within the hop radius which it feels is the best-connected node in the network. Connectivity indices used in [5] are not necessary to be calculated in our approach.

Step 6: Check acceptability of nodes. If all links/nodes are not represented by the set of nodes covered by the voting scheme, then we expand the WS and repeat from Step 4. If WS equals the POL, then increment the hop radius, and repeat from Step 3. It is suggested that the increment in hop radius be considered a final resort, as it effectively increases the amount of processing per monitoring node.

4.2 An Example

Let us consider a network given below in Fig. 2, listed with the PLANE values for different nodes.

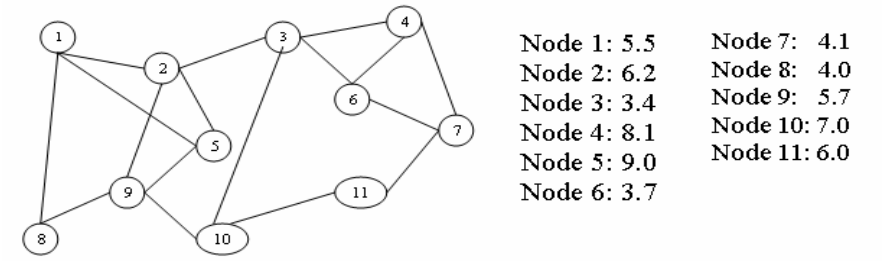

Fig. 2 An example network with a D = 3. The PLANE values (in relative time) for the different nodes are shown.

The POL is therefore given by{5,4,10,2,11,9,1,7,8,6,3} where each number represents the node number. We initially set the Hop radius to 1, in case an allocation is not possible, SPAID continues with higher hops. We depict the Working Set as WS{<node list> }, and iteratively augment the list with nodes from the PLANE Ordered List. Hence for this example, we begin with WS{5},i.e., we take the first node, Node 5, which has the highest PLANE.

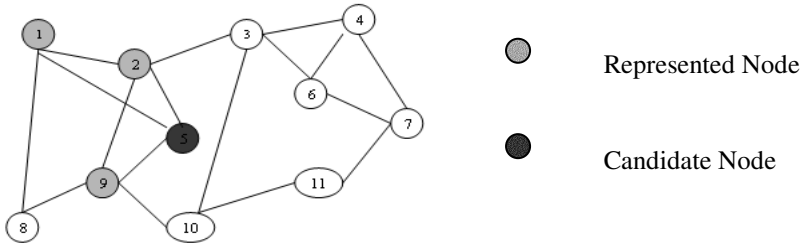

Fig 3. Example 1 with WS {5}.

Next, considering that all nodes have not been covered, we choose the next node, in this case node 4, and so on.

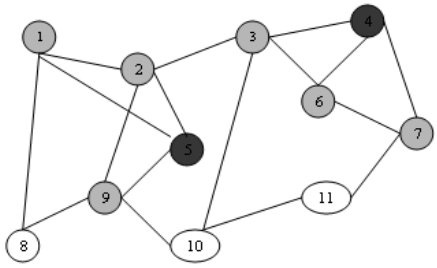

Fig 4. Example 1 with WS{5,4}.

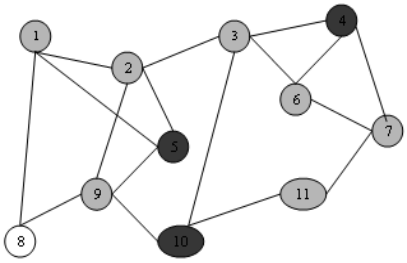

Fig 5. Example 1with WS{5,4,10}.

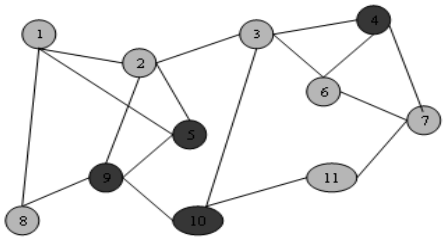

Fig 6. Example1. Final node selection {5,4,10,9}. WS{5,4,10,2} and WS{5,4,10,2,11} were skipped since no new nodes are represented.

The current WS {5,4,10} has the next choice in accordance with SPAID, as node 2, followed by node 11. The addition of these nodes, though provides no additional information from any node which cannot be established from existing nodes. Thus we have:

WS{5,4,10,2} – skipped , since no new nodes are represented.

WS{5,4,10,2*,11}-skipped , since no new nodes are represented.

All nodes are represented and hence the solution set WS reached is {5,4,10,9}. It is clear from the above example that the percentage of packet monitors varies inversely with increase in number of connections per node for a particular hop radius.

4.3. Rerunning SPAID

Dynamism in SPAID is a very important concept, considering that power levels drop considerably if a node persistently runs as a network monitoring node. The SPAID algorithm needs to be run, when a change in the power level of the current WS indicates that another node has a better chance of lasting longer as a network monitoring node.

In Example 1, after current WS {5,4,10,9} has run for about 200 seconds(assuming idle power consumption is minimal), the power level in node 9 would have dropped below that in node 1.In this case, the SPAID algorithm needs to be run again, to ensure a power-optimal solution to the multiple-sensor network monitoring problem is maintained as the power levels change.

5 Performance Comparisons

Comparisons between single-hop and multiple-hop radius for allocating network monitoring nodes provides a neat comparison of the tradeoff involved vis-à-vis the number of nodes needed. As the node density (D) increases drastically, the percentage of nodes allocated for network monitoring increases gradually and then stabilizes. The performance of SPAID can be appraised using the percentage of nodes selected as network monitors as a metric.

The density of the network clearly plays a major role, since more the number of adjacent nodes per node, fewer the network monitors needed to verify their authenticity. For high density wireless (D greater than or equal to 8) ad hoc networks, such as wireless sensor networks, we find that the density of network monitors stabilizes to near constant levels, and mimics the values presented in [5], which represent the performance of a non-power-aware node selection algorithm. As evident from the succeeding graphs, increasing node density and adjacency reduces the percentage of nodes to be selected as network monitoring nodes. IDS systems adapt quite efficiently to SPAID when using high-density ad-hoc networks with a large number of nodes.

A practical limit of 2 hops is practical, so as to limit the amount of network monitoring traffic to be transferred through intermediate nodes, as the amount of traffic varies as a quadratic of the hop-limit.

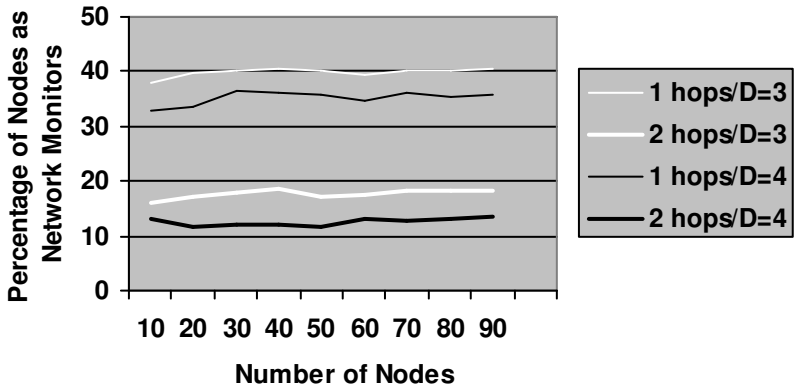

Fig 7. Performance of Sparse Wireless Networks with low average number of adjacent nodes (D) per node using SPAID. The near constant percentage of nodes used is to be noted.

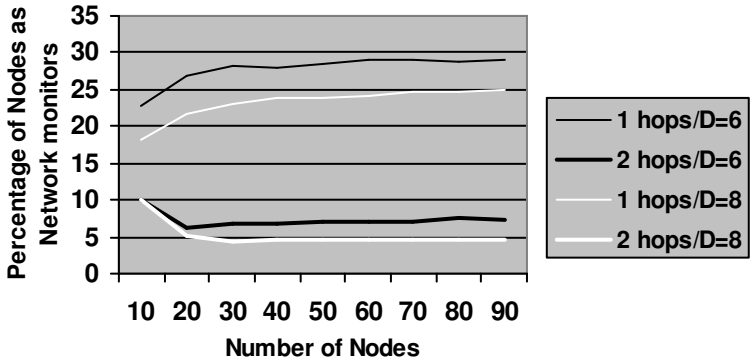

Fig 8.Performance of Dense Wireless Networks with high average number of adjacent nodes (D) per node using SPAID. The gradual drop in percentage of nodes towards stable levels is to be noted.

6 Conclusion

In this paper, we have suggested an iterative algorithm SPAID, that culminates from our consideration of individual node power-levels using PLANE. SPAID, as explained earlier, provides a capable trade off between strength of intrusion detection and the suitability of certain nodes to act as network monitoring nodes in an agent-based distributed network. Selection of network monitoring nodes plays a key role in determining the effectiveness of coverage of any intrusion detection techniques which run on each node, and through this paper we propose an adaptive scheme that connects power-awareness and agent-based node selection.

References

1. Heady, R., Luger, G., Maccabe, A., and Servilla, M.: The architecture of a network level intrusion detection system. Technical report, Computer Science Department, University of New Mexico, August 1990.
2. Zhou L., Haas, Z.J.: Securing Ad Hoc Networks, IEEE Networks Special Issue on Network Security. November,1999.
3. Zapata, M.G.: Secure Ad hoc On-Demand Distance Vector (SAODV) Routing. IETF MANET Mailing List, Message-ID 3BC17B40.BBF52E09.
4. Hu, Y.C., Johnson, D.B., Perrig A.: SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks Fourth IEEE Workshop on Mobile Computing Systems and Applications \mbox\rm (WMCSA '02).
5. Kachirski O., and Guha R.: Efficient Intrusion Detection using Multiple Sensors in Wireless Ad Hoc Networks", 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 2 , January 06 - 09, 2003.
6. Dasgupta D. and Brian, H.; "Mobile Security Agents for Network Traffic Analysis", Proceedings of DARPA Information Survivability Conference & Exposition II,2001.
7. Tao, J., Ji-ren, L.,Yang, Q.: "The Research on Dynamic Self-Adaptive Network Security Model Based on Mobile Agent", Proceedings of 36th International Conference on Technology of Object-Oriented Languages and Systems,2000.
8. Bernardes, M.C., and Moreira, E.S.: "Implementation of an Intrusion Detection System based on Mobile Agents", Proceedings of International Symposium on Software Engineering for Parallel and Distributed Systems, 2000, pp. 158-164.
9. Zhang, Y., and Lee, W.: "Intrusion Detection in Wireless Ad-Hoc Networks", Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, MobiCom'2000, pp. 275-283.
10. Perrig,A., Hu Y.C., and Johnson, D.B.: Wormhole Protection in Wireless Ad Hoc Networks. Technical Report TR01-384, Department of Computer Science, Rice University, December 2001.
11. Awerbuch B., Holmer D., Nita-Rotaru C., and Rubens, H.: An On-Demand Secure Routing Protocol Resilent to Byzantine Failures In ACM Workshop on Wireless Security (WiSe), Atlanta, Georgia, September 28 2002.
12. Bhargava, S., and Agrawal D.P.: Security enhancements in aodv protocol for wireless ad hoc networks. Vehicular Technology Conference, 2001.
13. Papadimitratos, P., and Haas, Z.J.: Secure Link State Routing for Mobile Ad Hoc Networks, IEEE Workshop on Security and Assurance in Ad hoc Networks, in conjunction with the 2003 International Symposium on Applications and the Internet, Orlando, FL, 2003.
14. Deng H., Zeng Q.A., and Agrawal, D.P.: SVM-based Intrusion Detection System for Wireless Ad Hoc Networks Proceedings of the IEEE Vehicular Technology Conference (VTC'03), Orlando, 2003.
15. Feeney L.M., and Nilsson, M.: Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment, Proceedings of IEEE INFOCOM 2001.