Intrusion Detection in Sensor Networks: A Non-cooperative Game Approach

Afrand Agah*, Sajal K. Das*, Kalyan Basu*
Mehran Asadi**
Department of Computer Science and Engineering
University of Texas at Arlington
Arlington, TX 76019-0015
{agah,das,basu,asadi}@cse.uta.edu

Abstract— **Insufficiency of memory and battery power of sensors makes the security of sensor networks a hard task to do. This insufficiency also makes applying the existing methods of securing other type of networks on the sensor networks unsuitable. We propose a game theoretic framework for defensing nodes in a sensor network. We apply three different schemes for defense. Our main concern in all three schemes is finding the most vulnerable node in a sensor network and protecting it. In the first scheme we formulate attack-defense problem as a two-player, nonzero-sum, non-cooperative game between an attacker and a sensor network. We show that this game achieves Nash equilibrium and thus leading to a defense strategy for the network. In the second scheme we use Markov Decision Process to predict the most vulnerable senor node. In the third scheme we use an intuitive metric (node's traffic) and protect the node with the highest value of this metric. We evaluate the performance of each of these three schemes, and show that the proposed game framework significantly increases the chance of success in defense strategy for sensor network.**

I. Introduction

It is crucial that the security of sensor networks be monitored and diagnosed to ensure correct behavior. However, this is a challenging task in an environment where the network is designed to be flexible. Due to resource scarcity (battery power, memory, and processing power) of sensors, securing sensor networks is quite different from traditional schemes that generally involve management and safe keeping of a small number of private and public keys [14]. Disclosing a key with each packet requires too much energy [18]. Storing one-way chain of secret keys along a message route requires considerable memory and computation of the nodes on that rout [19]. The key management using a trusted third party requires an engineered solution that makes it unsuitable for sensor network applications [7]. Although the asymmetric key cryptography does not require a trusted server, key revocation becomes a bottleneck [8] as it involves an authority maintaining a list of revoked keys on a server or requesting the public key directly from the owner.

Recently game theory has been used extensively to model networking problems [15], where different players may have different strategies for network usage. Game theory is a formal way to analyze interaction among a group of rational players who behave strategically. A game is the interactive situation, specified by the set of players (sensor nodes), the possible actions of each node, and the set of all possible payoffs. In order to detect intrusions we introduce a scheme based on a foundation of game theory, where we define a game between an attacker and the sensor network. In this game each player tries to maximize its own payoff. Attacker as a player decides to attack or to wait for a better time and attacks later. Sensor network uses an intrusion detection system (IDS) to defend the sensor nodes from intrusions. We formulate the attack-defense game as a two-player, nonzero-sum, non-cooperative game. We show that this game achieves Nash Equilibrium, thus leading to a defense strategy for the IDS. Then in order to defend the most vulnerable nodes, we elaborate two other approaches. First approach is learning mechanism (MDP), which predicts the most vulnerable node that can be attacked in the future. IDS will incorporate this learning mechanism and based on that, makes a decision to defend one node. Second approach is an intuition based logic, where IDS defends the sensor node with the maximum individual activity load. We compare the performance of these three schemes while facing attacks.

The rest of the paper is organized as follows. Section II talks about related works. Section III is our proposed framework, which discusses non-cooperative game scheme, the Markov Decision Process (MDP) and intuitive metric. Section IV represents simulation and performance evaluation. Section V concludes the paper.

II. Related Work

The μ TESLA protocol [17] uses a symmetric key mechanism. To generate one-way key chain, the sender chooses the last key randomly and generates the remaining values by successively applying a one-way function. The protocol discloses the key once per time interval (rather than one key per packet), and restricts the number of authenticated senders. To bootstrap, each receiver needs one authentication key of one-way function key chain. The base station can also broadcast disclosed key and perform initial bootstrapping for new receivers to conserve energy. The periodic key disclosure of μ TESLA ensures compromising a single sensor does not reveal the keys of all the sensors in the network.

Authors in [5] proposed CONFIDANT protocol, which consists of the following components: (i) monitor, (ii) reputation system, (iii) path manager and (iv) trust manager. In this approach neighborhood watch is proposed and nodes locally look for deviating nodes. As a component within each node, the monitor registers these deviations from normal behavior. Reputation system provides a quality rating of participants of transactions. Path manager ranks paths according to security metric. A trust table managing trust levels for nodes to determine the trustworthiness of paths will be managed by the trust manager. It is obvious that this approach would not fit sensor networks due to their limited memory.

A game theoretic approach for detecting network intrusions via sampling is presented in [13], in which an intrusion detection game is played between two players: the service

*Center for Research in Wireless Mobility and Networking (CReWMaN)
**Artificial Intelligence Laboratory

Proceedings of the Third IEEE International Symposium on Network Computing and Applications (NCA'04)
0-7695-2242-4/04 $ 20.00 IEEE

provider and the intruder. The intruder minimizes chances of detection and the provider maximizes it. This is a min-max approach, where the solution is a max-flow problem from which the stable points will be obtained. But in real life, sampling can be expensive. The authors have considered that adversary has considerable information about the network and is able to pick a path that minimizes chances of detection. The user cooperation in ad hoc networks has been studied in [23], in which an acceptance algorithm that each node uses, to decide whether to accept or reject a relay request is proposed. The system is proved to converge to an equilibrium point. The authors assume each user has sufficient information about the system, like number of users in each energy class, and hence users exchange their view of the system. However, they do not consider malicious users. A framework to study the existence of cooperation in packet forwarding in a wireless network is proposed in [10], in which a model is defined and the conditions under which cooperative strategies can form an equilibrium are identified. This approach does not require each node to keep track of the behavior of other nodes, but it is assumed all routes are static.

III. Proposed Framework

In this paper we present three different schemes to detect intrusions. In all three schemes we refer to IDS. IDS is an intrusion detection system, where its task is protecting sensor nodes. Due to system limitations, IDS would not be able to protect all sensor nodes, and so based on one of the three schemes, it will choose one sensor node for protection, later on we will refer to these sensor nodes as clusterheads. In the first scheme we define one non-cooperative game between attacker and sensor nodes. By using game theory framework we show that the game achieves Nash equilibrium for both attacker and IDS, thus leading to the defense strategy for IDS. In the second scheme we introduce an intrusion detection system based on MDP, which incorporates a learning mechanism. In the beginning IDS observes the system and learns the behavior of the attacker, and tries to decide which node needs protection. If it protects the same node that attacker wanted to attack, the attack is unsuccessful, which has a large reward for IDS. But if attacker attacks different node than the the sensor node that IDS is protecting then attack is successful. In the third scheme we use an intuitive metric. Traffic load is our metric and IDS chooses to protect the node which has the highest amount of traffic load. At the end we compare the performance of each of these approaches.

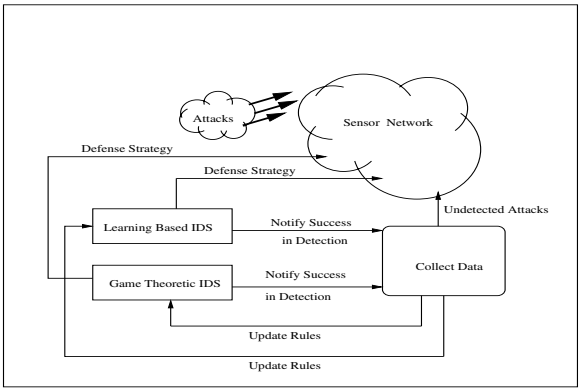

Fig. 1. A System View

As sensor network nodes are very limited in memory and power, we do not want to waste these limited resources. We need a way that sensor nodes can communicate with each other but would not consume too much energy. Also as nodes are memory limited, it would be infeasible for each node to remember all of its neighboring nodes. It is also very possible that sensor nodes do not have global identification (ID) because of the large amount of overhead and large number of sensors. So we divide all sensor nodes into clusters, in each cluster we choose one node as the cluster head. A positive point of this approach is that as IDS can not control all the nodes simultaneously it only has to monitor the chosen nodes which will be known as cluster heads, instead of monitoring all the nodes in the network. So one node in each cluster will be chosen as the cluster head, this choice can be random or based on some known clustering approaches like weight clustering approach (WCA) [9], Low-Energy Adaptive clustering Hierarchy (LEACH) [11], etc. We used WCA protocol, because cluster head election procedure is not periodic and is invoked as rarely as possible, which reduces system updates and hence computation and communication costs.

In the second scheme we use a learning mechanism, which is based on Markov Decision Process (MDP) [4]. The advantage of using MDP is the $reward$ concept. IDS gains the reward for choosing the right cluster to protect. And IDS tries to predict as correct as possible to maximize its own reward. In the third scheme we use an intuitive metric. At each time slot IDS should decide to protect one cluster, either the one that it was protecting during previous time slot or finding a more vulnerable one to protect. IDS will consider the activity load or traffic of each cluster and chooses a cluster for protection. At the end we compare the performance of these three schemes. Fig.1 depicts a high level view of the intrusion detection schemes.

A. Game Formulation

With respect to one fixed cluster, say k, the attacker has three strategies: (AS_1) attack cluster k, (AS_2) do not attack at all, (AS_3) attack a different cluster. The IDS has two strategies as well: (SS_1) defend cluster k, or (SS_2) defend a different cluster. We are considering the fact that at each time slot, IDS is defending one cluster. The payoffs of these two players are expressed in the form of 2×3 matrices, A and B where a_{ij} and b_{ij} denote IDS's and the attacker's payoff respectively. We define some notations:

- *U(t)*: the utility of sensor network's on-going sessions.
- AL_k: the average loss by loosing cluster k.
- C_k: the average cost of defending cluster k.
- N_k: the number of nodes in a cluster k.

The IDS's payoff matrix $A = [a_{ij}]_{2\times 3}$ is defined as follows:

$$A_{ij} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$$

Here $a_{11} = U(t) - C_k$ represents (AS_1, SS_1), which is when attacker and the IDS choose the same cluster (k) to attack and to defend respectively, so for IDS, its original utility value of $U(t)$ will be deducted by the cost of defense. $a_{12} = U(t) - C_k$ represents (AS_2, SS_1), which is when attacker does not attack at all, but IDS defends one cluster k, so we have to deduct the cost of defense. $a_{13} = U(t) - C_k - \sum_{i=1}^{N_{k'}} AL_{k'}$ represents (AS_3, SS_1), which is when attacker attacks cluster k', but IDS defends cluster k. In this case we subtract the average cost of defending one cluster, from original utility, as well as deducting the average loss of losing another cluster. $a_{21} = U(t) - C_{k'} - \sum_{i=1}^{N_k} AL_k$ represents (AS_1, SS_2), which is when attacker and the IDS choose two different clusters to

attack and to defend. $a_{22} = U(t) - C_{k'}$ represents (AS_2, SS_2), which is when attacker does not attack at all, but IDS defends one cluster k', so we have to deduct the cost of defense. $a_{23} = U(t) - C_{k'} - \sum_{i=1}^{N_{k''}} AL_{k''}$ represents (AS_3, SS_2), which is when IDS defends cluster k', but attacker attacks to a different cluster. In this case we subtract the average cost of defending one cluster, from original utility, as well as deducting the average loss of losing another cluster.

We define the attacker's payoff matrix $B = [b_{ij}]$ as follows:

$$B_{ij} = \begin{bmatrix} \text{PI(t)-CI} & \text{CW} & \text{PI(t)-CI} \\ \text{PI(t)-CI} & \text{CW} & \text{PI(t)-CI} \end{bmatrix}$$

where,

- CW: the cost of waiting and deciding to attack in the future.
- CI: the cost of intrusion for attacker.
- $PI(t)$: the average profit of each attack.

Here b_{11} and b_{21} are representing attacks to cluster k, b_{13} and b_{23} are representing attacks to different clusters than cluster k. We subtract average cost of attack from average profit of concurring a cluster. Also b_{12} and b_{22} represent non attack mode, and as attacker in these two modes decides to attack in future CW is the price to pay for waiting. We refer interested reader to our previous work [3] for more detailed definitions and descriptions of the above parameters. Now, we study the equilibrium solution for this game. Let us first introduce the concept of *dominant strategy* in the game theory. Given a bimatrix game defined by two $m \times n$ matrices, A and B, which are the payoffs of player p_1 and p_2 respectively. We say that "row i"dominates "row k" if $a_{ij} \geq a_{kj}$, for $j = 1, ..., n$. "row i" is called a dominant strategy for player p_1. For p_1, selecting the dominating "row i" is at least as good as selecting the dominated "row k". So "row k" actually can be removed from the game because p_1 as a rational player would not consider this strategy at all.

THEOREM 1: The game has Nash equilibrium at strategy pair (AS_1, SS_1).

Proof: Due to the limited space, we refer intrested reader to our previous work [3]. ■

The intuition behind the above discussion is that for IDS the best strategy is finding the best cluster to defend, which is the one with maximum value of $U(t) - C_k$, and for attacker the best strategy is finding the right cluster to attack, and as $PI - C$ is always more than CW, so attacker is always encouraged to attack.

B. MDP

Consider a stochastic process $\{X_n, n = 0, 1, 2, ...\}$ that takes on a finite number of possible values. If $X_n = i$, then the process is said to be in state i at time n. We suppose that whenever the process is in state i, there is a fixed probability P_{ij} that it will next be in state j. Such a stochastic process is known as a $Markov\ chain$ [21]. It basically states that the conditional distribution of any future state given the past states and the present state is independent of the past states and depends only on the present state. IDS's ability in choosing the right cluster to defend is a key factor in formulation this learning algorithm. We need to have a reward concept for IDS that only if it chooses the right cluster for protection, it will gain that reward. Basically in this section, we want to predict the future behavior of the attacker. We suppose that we know the past behavior of the attacker and so past states of the system, and now our task is predicting the most vulnerable node, which attacker most probably will attack.

A Markov Decision Process (MDP) is a model for sequential stochastic decision problems [4]. It is a four tuple (S, A, R, tr), where S is a set of states, A is a set of actions, R is a reward function, and tr is the state-transition function. A state $s \in S$ encapsulates all the relevant information about the state of the world. The actions change the states and the effect of the actions on the states is captured by the transition function. The transition function assigns a probability distribution to every (state, action) pair. Thus, $tr(s, a, s')$ is the probability of making a transition from state s to state s' when a is performed. Finally, the reward function assigns a real value to each (state, action) pair, which describes the immediate reward (or cost) of executing this action in that state. The states of the MDP for our intrusion detection system correspond to the states of the predictive model. For example the state (x_1, x_2, x_3) denotes the intrusion on the states x_3 where $\{x_1, x_2\}$ have been attacked in the past. This correspondence may not be optimal, and, in fact, it requires a large amount of training data (i.e., time on-line) to gain accuracy. Each action of the MDP corresponds to one intrusion detection of a sensor node. One can consider multiple intrusion detection systems based on MDP, but, to keep our model simple and computationally tractable, we consider only one intrusion detection. When we detect an intrusion on node x', the MDP can either accept this detection and thus state (x_1, x_2, x_3) will be transfered to state (x_1, x_2, x'), or it selects another node. The rewards in our MDP encode the utilities of detecting an intrusion. For example, the reward for state (x_1, x_2, x_3) can be the total benefit of keeping the node x_3. For simplicity we assign a constant value for the reward if the intrusion is detected. The transition function for the MDP model, $tr((x_1, x_2, x_3), x', (x_2, x_3, x''))$, is the probability that intrusion on node x'' is detected, given that node x' has been attacked in past. For the purpose of learning we use a learning method, which is known as Q-learning [16]. The objective, here, is to maximize the expected value of received reward over time. This can be done by learning a (possibly stochastic) mapping from states to actions, which is called a policy and defined as $\Pi : S \rightarrow A$, i.e. a mapping from states $s \in S$ to actions $a \in A$. The criterion used in selecting the action in every state is the maximization of its future reward. More precisely, the objective is to choose each action to maximize the expected return, $R = E\left[\sum_{i=0}^{\infty} \lambda^i w_i\right]$, where $\lambda \in [0, 1)$ is a discount-rate parameter and w_i refers to the reward at step i. The expected discounted sum of future rewards if action a is taken in state s is defined by the Q-function, $Q : S \times A \rightarrow \Re$, as:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[w_{t+1} + \lambda \max_{a \in A} Q(s_{t+1}, a) - Q(s_t, a)].$$

Hence Q-function, once learned, allows the learner to maximize R by picking actions greedily with respect to Q, such that, $\Pi(s) = argmax_{a \in A} Q(s, a)$. The Q-function is learned on-line through experimentation.

C. Intuitive Metric: Traffic

For the third scheme we use an intuitive metric. At each time slot IDS should decide to protect one cluster, either the one that it was protecting during previous time slot or finding a more vulnerable one to protect. We use activity load, which indicates traffic of each cluster. IDS will use this parameter and based on this value chooses a cluster for defending. So at each time slot the cluster with the highest value of traffic is the most vulnerable cluster and should be defended against attacks.

IV. Performance Evaluation

We developed a simulation platform to evaluate the performance of the three schemes. Assuming that there is only one

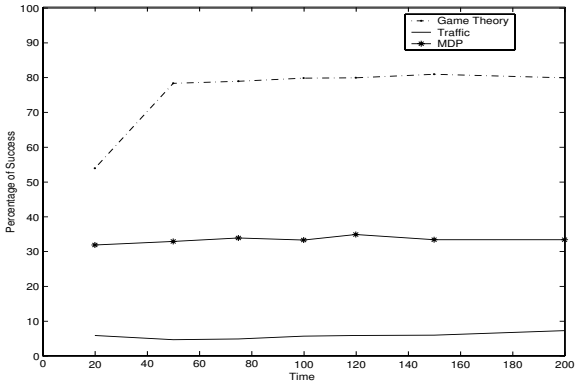

Fig. 2. N=50

attacker, one IDS, and the fact that at each time unit IDS is protecting only one cluster. In our simulations N gets different values ranging from 20 clusterhead nodes upto 200 clusterhead nodes. Figures 2 and 3 depict the success rate for 50 and 200 cluster head nodes in all three schemes, game theoretic framework, MDP and intuitive metric. Any learning approach needs some time to learn the past behavior and gets ready to predict the future behavior. Table 1. depicts the time that IDS needs to spend in order to learn and be able to predict the next most vulnerable node. This values are calculated based on the performance of MDP approach on a Pentium III computer, cpu model 1133 MHz. As the result indicate, by using the game theoretic frame work, performance of IDS is almost two times better than the overall performance of the MDP approach, also it does not need extra time for learning.

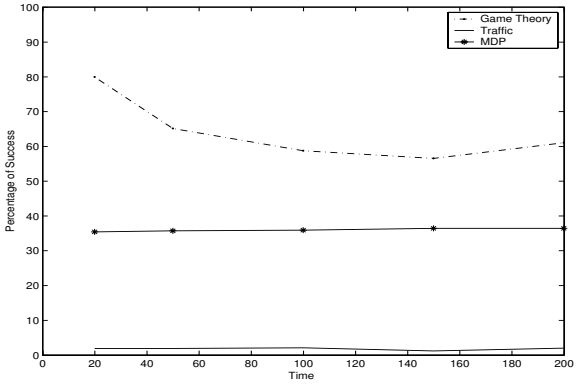

Fig. 3. N=200

V. Conclusion

In this paper, we proposed a game theoretic framework for intrusion detection in sensor networks. We formulated the intrusion detection problem as a non-cooperative two-player nonzero-sum game between the sensor network and the attacker. We show that Nash equilibrium can be established. We have considered many risk factors like reliability of a sensor node, different type of attacks, and past behavior of the attacker. We also implemented an MDP learning approach in order to predict the future behavior of the attacker. Simulation results show that by using game theoretic framework we can significantly improve the chance of intrusion detection.

There are many directions this work can be extended in the future. Finding a smart algorithm that knows when to use MDP algorithm and when to use game theoretic framework to have the maximum possible rate of success in detecting intrusions, is one of them that we plan to investigate.

TABLE I
Learning Time for MDP

N	20	50	100	200
Time in ms	243	590	1434	3245

Acknowledgments

This work is supported by NSF ITR grant IIS-0326505.

References

[1] I. F. Akyldiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol.38, 2002, pp:393-422.

[2] A. Agah, S. K. Das and K. Basu, "A Game theory based approach for security in wireless sensor networks," *International Performance Computing and Communications Conference (IPCCC)*, April 2004.

[3] A. Agah, S. K. Das and K. Basu, " A Non-cooperative Game Approach for Intrusion Detection in Sensor Networks," *VTC 2004*, Fall 2004.

[4] R. Bellman, "Dynamic Programming," *Princeton University Press*, 1957.

[5] S. Buchegger and J. L. Boudec, *Performance Analysis of the CONFIDANT Protocol Cooperation Of Nodes-Fairness In Dynamic Ad-hoc NeTworks*, MobiHoc 2002.

[6] N. Bulusu, D. Estrin, L. Girod and J. Heidemann, "Scalable coordination for wireless sensor networks: self-configuring localization systems," *International Symposium on Communication Theory and Applications (ISCTA)*, Ambleside, UK, July 2001.

[7] L. Buttyan, J. P. Hubaux and S. Capkun, "A Formal Analysis of Syversons Rational Exchange Protocol," *CSFW*, June, 2002.

[8] S. Capkun, L. Buttyan, and J. P. Hubaux, "Self-Organized Public Key Management for Mobile Ad hoc Networks," *MobiHoc*, 2002.

[9] M. Chatterjee, S. K. Sajal and D. Turgut, "WCA: A Weighted Clustering Algorithm for mobile Ad Hoc Networks," Cluster Computing , Vol.5, Kluwer Academic Publishers, 2002, pp:193-204.

[10] M.Felegyhazi, L. Buttyan and J.P.Hubaux, "Equilibrium Analysis of Packet Forwarding Strategies in Wireless Ad Hoc Networks- the Static Case," *Proceedings of Personal Wireless Communications (PWC '03)*, Venice, Italy, September 2003.

[11] W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-efficient communication protocol for wireless micro sensor networks," *IEEE Proceedings of the Hawaii International Conference on System Sciences*, January, 2000, pp:1-10.

[12] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," *Proc.6th Ann.Int'l Conf.Mobile Computing and Networking (MobiCom 2000)*, ACM Press, New York, 2000, pp:243-254.

[13] M.Kodialam and T.V.Lakshman, "Detecting Network Intrusions via Sampling: A Game Theoretic Approach," *INFOCOM* , 2003.

[14] A. Menezes, P. Oorschot and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.

[15] P.Michiardi and R.Molva, "A Game Theoretical Approach to Evaluate Cooperation Enforcement Mechanisms in Mobile Ad hoc Network," *WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2003.

[16] R. E. Parr, "Learning for Markov Decision Processes," *PhD Thesis University of California, Berkeley*, 1998.

[17] A. Perrig, R. Canetti, J. Tygar and D. Song, "Efficient Authentication and Signing for Multicast," NDSS, 2001.

[18] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," *ACM MobiCom*, July 2001, pp:189-199.

[19] A. Perrig and J. D. Tygar, "Secure Broadcast Communication in Wired and Wireless Networks," *Kluwer Academic Publisher*, 2003.

[20] S. M. Ross, *Introduction to Probability Models*, 6th edition, Academic Press, 1997.

[21] S. Russell and P. Norvig, *Artificial Intelligence A Modern Approach*, 2nd Ed., Prentice Hall, 2003.

[22] E. Shih, S. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, A.Chandrakasan, "Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks," *Proceedings of ACM MobiCom*, Italy, July 2001, pp:272-286.

[23] V.Srinivasan, P.Nuggehalli, C.F.Chiasserini and R.R.Rao, "Cooperation in Wireless Ad Hoc Networks," *INFOCOM*, 2003.

[24] A. D. Wood and J. A. Stankovic, " Denial of service in Sensor Networks," *IEEE Computer*, 2002, pp:54-62.