# Improved Intrusion Detection System using Quantal Response Equilibrium-based Game Model and Rule-based Classification

Manjula C Belavagi[1] and Balachandra Muniyal[2]

[1]Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India
[2]Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India

***Abstract*:** Wireless sensor network has large number of low-cost tiny nodes with sensing capability. These provide low cost solutions to many real world problems such as such as defence, Internet of things, healthcare, environment monitoring and so on. The sensor nodes of these networks are placed in vulnerable environment. Hence, the security of these networks is very important. Intrusion Detection System (IDS) plays an important role in providing a security to such type of networks. The sensor nodes of the network have limited power and, traditional security mechanisms such as key-management, encryption decryption and authentication techniques cannot be installed on the nodes. Hence, there is a need of special security mechanism to handle the intrusions. In this paper, intrusion detection system is designed and implemented using game theory and machine learning to identify multiple attacks. Game theory is designed and used to apply the IDS optimally in WSN. The game model is designed by defining the players and the corresponding strategies. Quantal Response Equilibrium (QRE) concept of game theory is used to select the strategies in optimal way for the intrusion's detection. Further, these intrusions are classified as denial of service attack, rank attack or selective forwarding attacks using supervised machine learning technique based on different parameters and rules. Results show that all the attacks are detected with good detection rate and the proposed approach provides optimal usage of IDS.

***Keywords*:** Wireless Sensor Networks, Machine Learning, Game Theory, Intrusion Detection System, Quantal Response Equilibrium, Rule-based Classifier**.**

## 1. Introduction

Wireless Sensor Networks (WSN) comprises of large number of small sensor nodes. These nodes have low power with sensing capability and are used to observe the physical and environmental situations. These nodes forms a topology by organizing themselves. WSNs provides low cost solutions to many real world applications such as military, Internet of Things, health, business, environment surveillance etc [1]. Hence, they are becoming more and more popular. These nodes are deployed in not secure environments. Hence, they face different challenges of security like Sybil attack, Routing attacks, Denial of Service (DoS) attacks etc. Traditional security techniques such as authentication techniques, encryption, decryption, security protocols and key-management techniques cannot be implemented on sensor nodes due the lack of power and data [2]. Hence it is highly challenging to provide security to WSN due to their resource constraints. They require unique security technique such as an Intrusion Detection System (IDS). A Security mechanism used to monitor the abnormal behaviour of the WSN's is an IDS. Actions that violate integrity, confidentiality and availability of information and resources are called intrusion. Currently to handle insider and external attacks different IDS techniques are proposed by [3] [4] [5][6]. Misuse, anomaly and specification-based IDS techniques are computationally costly.

In order to use the resources efficiently, optimal strategies are designed using game theory to identify the intrusions [7], [8], [9] [10]. Game model is designed by considering Quantal Response Equilibrium (QRE) [11] [12]. It works based on the assumption that players are not going to select the strategy with highest payoff instead they select the strategies which gives better payoffs not the one which gives best payoffs. It is probability based strategy selection. Machine learning techniques are used to identify the intrusions of the WSN [13]. It is possible to identify multiple attacks by generating the rules using these techniques. In this paper Random Forest algorithm is used to generate the rules and to identify the multiple attacks.

Based on the literature, it is observed that there is a limited research available in intrusion detection using both machine learning and game theory. Hence, this paper focuses on stage wise intrusion detection using game theory and machine learning. Initially QRE is used to select the optimal strategy by an IDS agent based on the behavior of the sensor node. If the behavior is malicious then the generated rules are used to identify the multiple attacks.

Following are the contributions of this paper:

- Design a repeated intrusion detection game model by defining the players, strategies, and payoffs.
- Identification of multiple attacks using rule-based machine learning technique.

Remaining sections of the paper are organized as follows: in section 2 related work of intrusion detection is discussed. Data preparation and methodology is discussed in section 3 and results are discussed in section 4. Paper concludes in section 5.

## 2. Literature Review

Various intrusion detection mechanisms have been proposed by the researchers for WSN security. An intrusion detection model to handle denial of sleep attacks is proposed by Salmon et al. [14]. This method makes the sensor nodes to

awaken for longer times and increase the packet collisions to drain the resources. This is implemented using TOSSIM simulator. The technique uses a signal to identify an antigen or attacker belonging to the body or not. Dendritic cells process these signals to identify antigens as normal or not. The WSN on Contiki operating System using Cooja simulator is developed by the authors [15], [16] and [17]. They have compared the simulated network on different performance metrics and suggested that Routing Protocol for Low Power and Lossy Networks (RPL) is best suited protocol for WSN. Thombre et al [15] have verified simulated performance with the physically deployed network also.

Specification based IDS to identify the attacks in low power and lossy networks is proposed by Le et al [18]. Simulation of the network is carried out using Cooja simulator. They have observed that the deviation in the behaviour of each node to identify the malicious activities. They have concluded that the proposed method has good accuracy, overhead in case of large networks. The network traffic cation on real time data traffic is proposed by Jun et al [19]. Which is unsupervised machine learning approach to detect application based network traffic. Internet Protocol payload and some statistical properties are used as the parameters. Content of the clusters are represented using bag of word model. They categorized the similar traffic based on the payload content. Hummen et al [20] proposed packet fragmentation based intrusion detection in 6LowPAN and lossy networks. They have considered fragment duplication attack and buffer reservation attack. The cost of detection is less whereas detection rate is moderate.

Anomaly based method in wireless clusters architecture is proposed by Yassine et al [21]. They have experimented using Support Vector Machine (SVM) with the assumption that cluster head is known node, which forwards the packets to the base station. The paper depicts high detection rate and false positive rate as low. In RPL networks sinkhole defence mechanisms are evaluated based on rank verification and parent fail over techniques [22]. Results show that the combination of the above-mentioned methods can be used to improve the performance. Optimal strategies are defined using game theory based on the concept of puzzle is proposed by fallah et al [23]. They also handled flooding attacks by defining different strategies. They used the concept of Nash equilibrium and adjusted the difficulty level of puzzles and many other parameters. The method does not result in the exhaustion of defender's resources, gives maximum possible payoffs for the defender and very effective. Hence the defence mechanism works fine in detecting flooding attacks for an unknown number of sources.

Game theoretic defence mechanisms against Denial of Service (DoS) attacks is proposed by Narasimhan et al [24]. Important parameters are selected to avoid the overloading of the server by the attacker. Defence mechanism and the proposed model is based on the game theoretic model. They also designed an improved difficult puzzle that should not determine by the intruder. They concluded that if the puzzle difficulty is hidden from the attacker, then the game defence mechanism is very effective. Non-cooperative game theory with fuzzy Q learning to handle DoS attacks is proposed by Shamsi et al [10]. They concluded that the method has good detection rate and accuracy. However, they have not taken care of other attacks.

## 3. Methodology

To achieve efficient network level security, the basic IDS is improved by integrating game model and machine-learning algorithm. The integration framework of IDS is shown in the Figure 1.

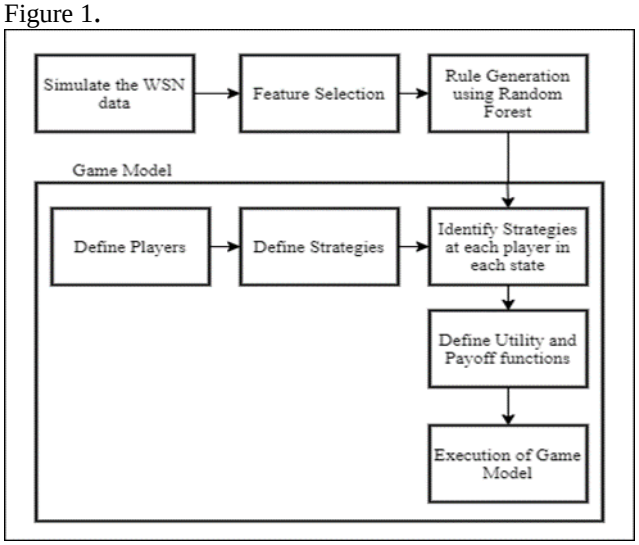


Figure 1 Proposed Framework of Efficient IDS

The framework consists of capturing the network data from the simulated WSN, extracting the relevant features from the generated network data which is in the form Packet Capture Format (pcap) file. These features will be useful to train the machine learning algorithm for generating the appropriate rules. The generated rules further classify the captured malicious behaviour. To run the IDS in optimized way in the WSN, Game model is designed to obtain the strategies for the intrusion detection.

### 3.1 WSN Simulation

Initially, WSN network traffic is simulated on Contiki operating system using Cooja simulator [25]. The routing protocol used for such networks is RPL [26] [27] based WSN is considered for the experimentation. The initial set of parameters used for the simulation are shown in the Table 1 to capture the network traffic data. Communication between the nodes is observed using 1) Simulation Visualizer 2) Timeline and 3) Radio Logger.

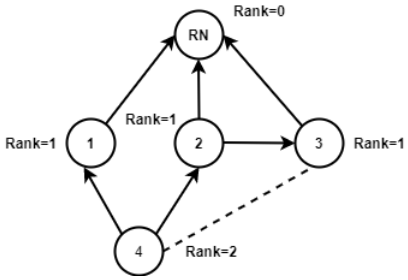


**Figure 2.** WSN topology –DODAG

The topology used for such networks is Destination Oriented Directed Acyclic Graph (DODAG), which is shown in the Figure 2. The topology construction depends on the control messages DODAG Advertisement Object (DAO), DODAG Information Object (DIO) and DODAG Information Solicitation (DIS). DAO forwards the routing information about the destination towards the Root (unicast). DIO

identifies the RPL Instance (multicast) and DIS message is used to maintain the overall topology. In the figure 2, root node "RN" is a Sink Node. Sink Node is considered as a Base Station. Based on the position of the node, rank is assigned to each node for managing the hierarchy of the network.

**Table 1.** Simulation Parameters

| Parameters Used | Value |
|---|---|
| **RPL Mode of Operation(MOP)** | NO-DOWNWARD-ROUTE |
| **Transmission-Range** | 50 m |
| **Packet-Transmission and Reception Ratio** | 100 % |
| **Objective Function** | ETX |
| **DIO Min** | 12ms |
| **Number of Client (Sensor) Nodes** | 50 |
| **Simulation Time Duration** | 20 minutes |
| **Interference-Range** | 55m |
| **$\delta_{PDR}$** | 60-70% |
| **$\delta_{DPR}$** | 80-85% |
| **$\delta_{PFR}$** | 90-95% |

### 3.2 Feature Extraction

Feature extraction is necessary step in the framework as the pcap file captured several features during simulation and relevant features need to be extracted for further processing. The pcap file contains source IP address, destination IP address, timestamp, protocols used for communication, and packet formats of the protocols. Hence, the relevant features extracted by executing a python script. The python scripts first check the source and destination IPs along with the protocol. Based on protocol packet format, the required fields are extracted with the specific data. The extracted features are stored in .csv file.

### 3.3 Generation of Rules for Multiple Attack Detection

The extracted relevant features are used to generate the rules. Since, the number of features is more and cannot be used directly to generate the rules, required computations have been done as a part of pre-processing of the data. The final set of features are fed to Random Forest algorithm. This algorithm creates separate Decision Trees based features identified for various attacks and generates separate rules. Steps to generate rules using Random Forest are shown in Algorithm 1.

**Algorithm 1: Random Forest Rule Generation (RFRG)**

**Input:** Training set D = {($X_1$, $y_1$). ($X_2$, $y_2$), = {($X_2$, $y_3$)….. ($X_n$, $y_n$)} // Where X is the data and y is the label
**Output:** Rule set R = {$R_1$, $R_2$, ……$R_P$}
**Begin**
dtN= Number of decision trees to construct in random forest
for i=1 to dtN
{
B = BootStrapSampling(training set D) // Selecting subset from D without replacement
$D_i$ = Decision tree using B
$R_i$ = All the rules generated by $D_i$
R = R U $R_i$ // R is the set containing all the rules generated
}
OR = φ // OR : Set containing optimized rules
for each sample j in the test set
{
$PV^j$ = Prediction using majority voting // Prediction using ensemble approach using random forest
If ($PV^j$== $y_i$) // Correct Prediction using Ensemble majority voting
CP++; // Count of correct prediction using Ensemble majority voting
for each Rule i in R
{
$PR^j_i$ = Prediction using Rule i // Prediction for sample j using Rule i
If( $PR^j_i$==$y_i$) // Correct Prediction using Rule i
CR++; // Count of correct prediction using Rule i
}
If (CR> 0.5*CP)
OR = OR U $R_i$ // OR: Optimized Rule Set
}
**End**

In Algorithm 1, the labelled WSN Intrusion Data is divided into training and test data as D_Training and D_Test. D_Training is used for building Random Forest, which constructs dtN number of decision trees. The rules generated by decision tree DTi are saved in set Ri. All the rules are merged and saved in set R. Once the Rule set is constructed, next step is to identify the optimal rules for intrusion detection. Optimal rules are identified using test set D_test. For that 3 steps are used, first for each sample in the test set the prediction is obtained using ensemble approach of random forest using majority voting. Subsequently it is checked if it is correct prediction if CP count is incremented. In the second step each sample in the test set the prediction is obtained using $Rule_i$. Subsequently it is checked if it is correct prediction if CR count is incremented. In the third step $Rule_i$ is checked for optimality considering the condition if correct predictions count using $Rule_i$ is at least better than 50% of the predictions using ensemble approach, then it is added to the Optimal rule set (OR).

The Algorithm 1 is executed at the base station periodically so that rules are generated and updated dynamically. Then the rules are executed to identify the malicious node behaviour at the base station. The generated rules are also used by the IDS agent during monitoring state to identify the intrusions as malicious communication.

### 3.4 Game Model for IDS

Based on the computing locations, IDS Agents are classified as centralized, distributed, and hierarchical. In case of centralized, the IDS agent is installed on sink node or Base Station to monitor the behaviour of the sensor nodes and the entire network. In distributed, the IDS Agent is installed on every sensor node to monitor the network which in turns increases the computation cost of the entire network. Hierarchical model consists of monitoring sensor nodes as Cluster Heads (CH). These nodes work as monitoring agents and normal sensing nodes. In this paper hierarchical model is used in the WSN structure which is shown in the Figure 3.

IDS Agent is installed on CH to monitor all the sensor nodes and communications of the network. IT also interacts with the BS. To design the game model for optimal execution of the IDS agent, the sensor nodes and IDS agents

are considered as two players. The game model for IDS (GIDS) is defined as two players-noncooperative-repeated game. The main aim of GIDS is to select optimal strategies for IDS agents against the sensor node as an Attacker. The structure of the GIDS is shown in Figure 4. Sensor Node ($SN_i$) is a Player 1 which can show "Normal" or "Malicious" behaviour in the network. Hence, Player 1 can have two pure strategies: "Attack (A)" and "No- Attack (NA)". IDS agent ($IA_j$) is player 2 which can have two pure strategies: "Monitor (M)" and "No-Monitor (NM)".

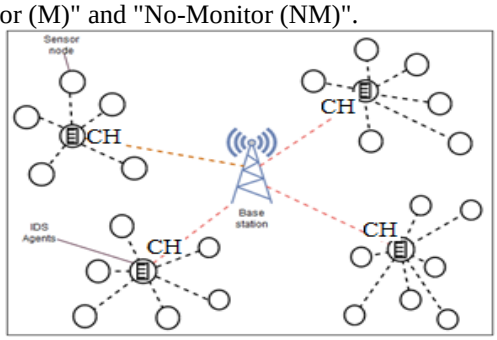


**Figure 3.** WSN Architecture

$IA_j$ = {Monitor (M), No-Monitor (NM)} (1)

$SN_i$ = {Attack(A), No-Attack (NA)} (2)

Interaction between the players $SN_i$ and $IA_j$ is basically provides incomplete information to the game model as the IDS agent is not sure about the behaviour of the sensor node. Hence, the IDS agent should intelligently identify the malicious behaviour of the sensor node during execution of the game model to define the strategies as A or NA.

Game is defined by specifying players (P), Strategies (S) and Payoff Utilities (U) as follows:

$G = \{P, S, U\}$ (3)

$P = \{SN_i, IA_j\}$ (4)

Where $SN_i$ is sensor node, $IA_j$ is IDS Agent,

$S = \{S_i X S_j\}$ (5)

Where S is strategy space, it is Cartesian product of strategies of two players $SN_i$ and $IA_j$

$U = \{U_i X U_j\}$ (6)

U is the payoff utility based on the strategy space S. Where $U_i$ and $U_j$ are payoffs of players $SN_i$ and $IA_j$. Game model updates the payoff utility values for each player based on the selected strategies by the players. These values decide the gains of each player and ultimately specifies the winner. The more weightage is given to $IA_j$ for selecting the best strategies in the network based on possible actions of the SNi. Player SNi has two possible actions: NA and A which can be identified based on behaviour of the respective sensor node. Usually, IAj will always choose action M to monitor the communication in the network. This leads to energy loss in the network and increases computation cost. Hence, IAj should select action NM to save the energy if the player SNi has normal behaviour. If SNi as an attacker, it tries to attack and based on strategy of the game it gains its profit. In response to this, player IAj identifies SNi with malicious behaviour and defends by selecting appropriate action M.

Member node may be malicious or normal, it plays different actions A or NA. IDS agent decides an event is normal or not by monitoring the events. Then these results stored temporarily. IDS agent calculates QRE probabilities and send this information to the sink node. After this round the game parameters are updated. The payoffs of the players are also updated based on the strategy selection. This process is repeated until the IDS Agent selects monitoring mode which is shown in Figure 4. Most profitable action for the player SNi is A, if it is not monitored by the IDS Agent by selecting the action NM. Hence, the strategies of both players are (A, NM). Based on these kinds of strategies, the Payoff matrix is designed as shown in Table 2.

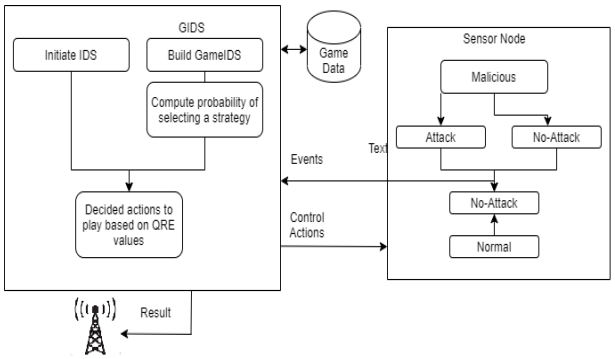


**Figure 4.** Structure of GIDS

**Table 2.** Payoffs – IDS Agent and Attacker

| | | IDS Agent | |
|---|---|---|---|
| | | Monitor (M) | No-Monitor (NM) |
| **Attacker** | No-Attack (NA) | 1,3 | 0,1 |
| | Attack (A) | 3,0 | 1,4 |

In the defined GIDS, the players play their actions continuously. Hence, GIDS is designed as repeated game (RGIDS).

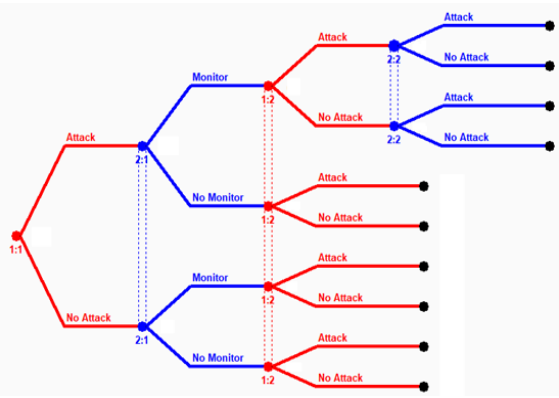


**Figure 5.** Repeated Game Representation – Extensive Game Form

Following definitions are used for the same:

$G = \{\infty, \delta\}$ (7)

Set of players defined are IAj and SNi.

Overall strategy at nth stage of GIDS for each player $a \in \{SN_i, IA_j\}$ is

$s^n_a = [\, s^n_a(h_0), s^n_a(h_1), \ldots s^n_a(h_n)]$ (8)

where ($h_y$) is $y^{th}$ history stage, $y \in \{0,1, \ldots n\}$.

Strategy selected by the player at $y^{th}$ history stage is $s^n_a(h_y)$. Overall payoff of each player, $a \in \{SN_i, IA_j\}$, is the average of instant payoffs of each round in RGIDS. Extensive form of RGIDS represented in Figure 5. Attacker is aware of past actions of IDS agent, whereas player $IA_j$ is

imperfectly aware of past actions player $SN_i$. It judges actions of $SN_i$ with uncertainty

From the Figure 5 it can be observed that, initially attacker may select attack or no-attack action. In the next level IDS agent responds to this by selecting the actions as monitor or no-monitor. As soon as IAj selects monitor, the game ends otherwise the game played repeatedly. The players try to maximize their payoffs over multiple rounds, hence the total number of strategy profiles at $n^{th}$ stage is computed depends on history profile strategies i.e. 0, 1,...,n-1.

Players $IA_j$, $SN_i$ attempt to maximize their estimated payoffs over the multiple rounds of GIDS. The expected payoff is the sum of payoffs of all the rounds. Where δ is the discount factor δ ∈ [0, 1). The total payoff for player a, a ∈{$IA_j$, $SN_i$}is defined as follows:

$$u_a^n(s_a) = \sum_{y=0}^{n} \delta^y \; u_a^y(s_a) \quad (9)$$

Where $u_a^y(s_a)$ is payoff obtained by the player by selecting the strategy $s_a$ at y=0,1,…n.

In the repeated game with infinite rounds, the total payoff is the average of the value obtained in Equation 9. Hence it is formulated as follows:

$$u_a^{-n}(s_a) = (1-\delta) \sum_{y=0}^{n} \delta^y \; u_a^y(s_a) \quad (10)$$

Further, total number of strategy profiles at $n^{th}$ state depends on the stages 0,1,..n-1. Hence, it is the product of history profiles of these stages. In the designed game model (GIDS), game ends by selecting a strategy Monitor (*M*) by the player $IA_j$. So the total number of actions is the number of combined actions (excluding the terminal action *M*) i.e. 2*(1) . Hence, total number of strategy profiles at $n^{th}$ stage is computed using the equation 11.

$$t_n = 2 * t_{n-1} \quad (11)$$

Where n=1,2,3…. and $t_{n0} = 4$

Strategies defined in RGIDS will increase with respect to increase in number of stages of GIDS. So complexity of identifying the player's behaviour by computing Nash Equilibrium also increases. This can be overcome by QRE model.

Quantal Response Equilibrium (QRE) based strategies are considered for the game implementation. This is suitable for the games with separate strategies. Whereas the strategy selection is probabilistic and is not deterministic. The rationality parameter λ is used for the payoffs.

If it is '0' both the players are irrational, i.e players select strategy not given by Nash Equilibrium. Hence, they cannot obtain greater payoffs. If the λ is ∞ then they show the opposite behaviour. So, Equation 12 is used to calculate the QRE.

$$\prod_{s_a}^{n} = \frac{\exp(\lambda u_a^{-n}(s_a))}{\sum_{y \in s} \exp(\lambda u_a^{-n}(y))} \quad (12)$$

Where $\prod_{s_a}^{n}$ represents probability of player a, a ∈ {$IA_j$, $SN_i$} selecting strategy $s_a$ . QRE based strategies for the defined players are obtained from Equation 12.

### 3.5 Communication Models between Base Station and IDS Agent

WSN has narrow bandwidth radio channels hence there is a need to minimize the information exchange between the IDS agents and the base station to reduce the congestion in the network due to intrusion detection traffic. This can be addressed using game theory based probabilistic monitoring method.

After the selection of strategies from the game model further processing is done in the sink node. Figure 5 shows communications between IDS Agent and Base Station. IDS agent monitors the group of sensor nodes using set of specification rules. Initially these rules are generated by Random Forest Classifier based on the historical data at the base station and are updated periodically and set to the IDS agent.

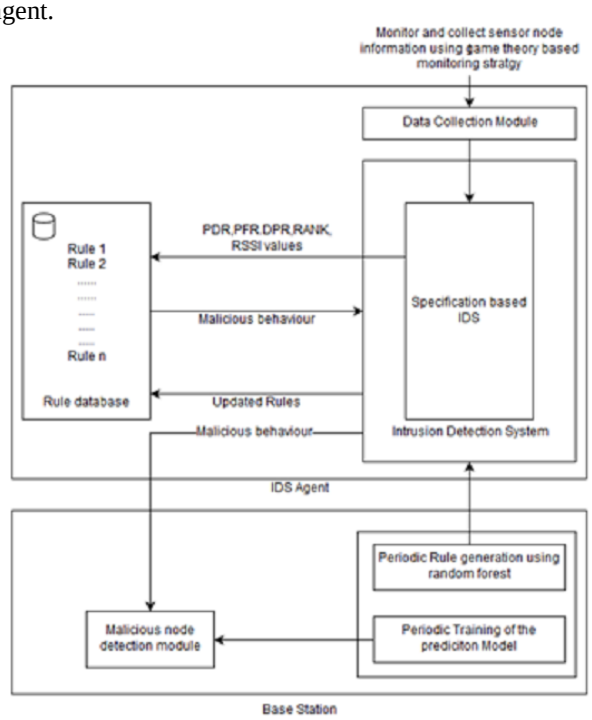


**Figure 5.** Communication Modules of Base Station and IDS Agent

A sensor node can behave as normal or malicious, hence it has two functionalities as listed as follows:

- No-Attack:
  - Sensor node is normal node
- Attack:
  - Selective forwarding - Malicious node forwards only certain packets and drops other. This can be identified by observing the Packet Drop Rate (PDR). This should not exceed the threshold $\Delta_{pdr}$.
  - Denial of Service - Malicious node disturbs the network by sending redundant messages. This can be identified by observing Duplicate Packet Rate (DPR) and Packet Forwarding Rate (PFR) rates. $\Delta_{DPR}$ and $\Delta_{PFR}$ are the threshold ranges considered for the simulation.
  - Rank Inconsistency (RI) - In case of rank attack a malicious node advertises a better rank value than the actual value it has. This also leads to sinkhole attack.

In order to detect the above-mentioned attacks, rules are generated by considering parameters such as PDR, RI, DPR and PFR at the Base Station using Random Forest Rule Generation (RFRG) algorithm.

IDS Agent uses its strategies to observe the behaviour of the sensor nodes. Observed result is forwarded to the base station. As shown in Algorithm 2. At the base station actual identification is carried out using generated rules. The model

is also trained periodically to incorporate new behaviour of malicious nodes.

**Algorithm 2: Game IDS**

Procedure Game-IDS (Strategies, Player)

1. Initialize necessary game parameters
2. while Not end of interactions do
3. **If** IDS Agent **then**
4. Monitor the events as normal or malicious
5. **If** Event is malicious **Then**
6. **If** the computations of Repeated GIDS not completed Then
7. Construct the first stage RGIDS
   **Else**
8. Use the stored data for further processing
   Endif
9. Compute $\Pi_{s_a}^{n}$
10. Compute $u_a^{-n}(s_a)$ and store it for next stage
11. Send the combine result to Sink
12. End While

Following are the rules generated for multiple attack detection:

if $PDR_{Node\_ID} > \Delta_{PDR}$ then
send Message (selective_forwarding, Node_ID) to Sink
if $(DPR_{Node\text{-}ID} > \Delta_{DPR})$ and $(PFR_{Node\text{-}ID} > \Delta_{PFR})$ then
send Message (DoS_attack, Node_ID) to Sink
end if
if Node_rank Mismatch then
send Message (Rank_attack, Node_ID) to Sink
end if

## 4. Results and Discussion

The efficiency of IDS for WSN is evaluated by conducting the simulation. The traffic of simulated WSN is shown in Figure 6. The screenshot of 'pcap' file of Cooja simulator consists of features such as id, source and destination IP, timestamp, protocol and various fields of packet formats. After applying feature extraction, the final set of features are shown in the Table 3.

The implementation of game model for IDS is done using open source Gambit tool. This tool has packages to design and analyse strategic and extensive forms of games. Working of GIDS with QRE based strategies are implemented using Gambit by considering WSN structure.

Initially all the actions of each player begin with equal probability. In the designed game model, two actions for each player are considered, hence each action starts with a probability of 0.5. Rationality parameter(λ) starts with 0 value. It indicates the selection probability of each action or strategy.

In Figure 7 and 8, y-axis represents the probability of selection of a certain strategy at given λ. If the sensor node behaves as an attacker, then the probability of selecting the action as A is high. From the figure 7, it can be observed that probability of selecting the action NA is gradually decreasing, whereas the probability of selecting the action A is increasing. The usual behaviour of IDS Agent is to monitor the nodes and communication in the WSN. From the figure 8, it can be identified that with increase in probability of λ, increases the probability of selecting the action M and decreases the selection probability of NM.

In the Table 4, when λ=203 probability of selecting action NA becomes 0. Action NA is eliminated from this step. When λ > 147.0421 the attacker always selects A action. When the value of λ reaches 203 the probability of selecting the action as *NA* becomes zero approximately. Similarly, when λ is 3.153901 IDS Agent always selects Monitor strategy.

**Table 3.** Features Selected

| Source IP | Destination IP | Time |
|---|---|---|
| **DODAGID** | RPL Sequence no. | Flags |
| **MOP** | Checksum | Payload |
| **Sequence No.** | Data | Frame Control |
| **Checksum** | Frame Length | Time Delta |
| **FCS** | RPL Instance ID | Source Port |
| **Destination Port** | UDP payload | Lifetime limit |
| **Hop Count** | ETX | Objective Function |
| **Rank** | Parent Node Rank | Control Message DIO |

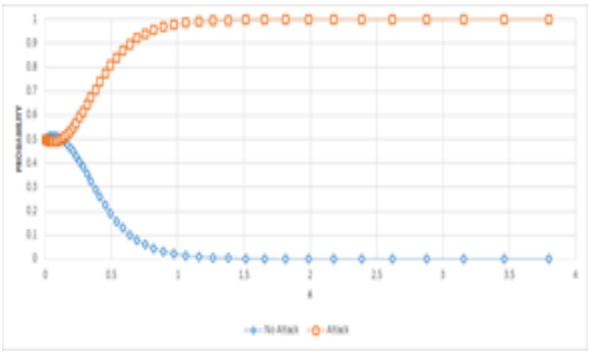

**Figure 7.** QRE – Probability values for Attacker

The result of multiple attacks detection is shown in Table 5. It shows the detection rate and accuracy of rank attack, selective forwarding, and denial of service attack. The structure of WSN as per simulation parameters consists of 50 client (sensor) nodes and 5 cluster heads which work as IDS Agents. The detection rate and accuracy are calculated by considering the parameters of confusion matrix. From the table it can be observed that, the detection rate and accuracy of rank attack is good.

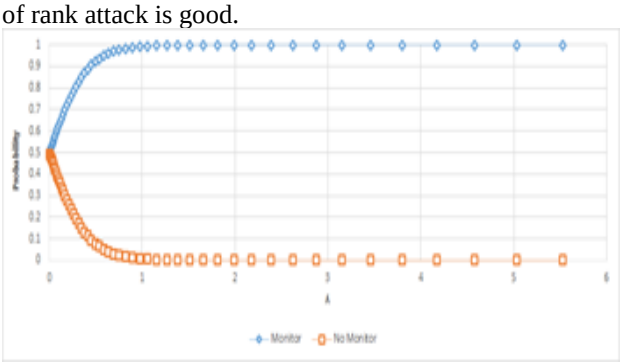

**Figure 8.** QRE - Probability Values for IDS Agent

**Table 4.** QRE-Result for Senor Node and IDS Agent

| λ | No-Attack | Attack | Monitor | No-Monitor |
|---|---|---|---|---|
| **0** | 0.5 | 0.5 | 0.5 | 0.5 |
| **0.00540** | 0.50193 | 0.49806 | 0.50675 | 0.49324 |
| **0.01137** | 0.50386 | 0.49613 | 0.51421 | 0.48578 |
| **0.01796** | 0.50572 | 0.49427 | 0.52243 | 0.47756 |
| **0.02523** | 0.507474 | 0.49252 | 0.53149 | 0.4685 |
| **0.03324** | 0.50901 | 0.49098 | 0.54146 | 0.45853 |
| **0.042079** | 0.510262 | 0.48973 | 0.55240 | 0.44759 |
| **0.051795** | 0.511078 | 0.48892 | 0.56438 | 0.43561 |
| **....** | .... | .... | .... | .... |
| **....** | .... | .... | .... | .... |
| **....** | .... | .... | .... | .... |
| **1.814589** | 0.000709 | 0.99929 | 0.99988 | 0.000114 |
| **1.988079** | 0.0003531 | 0.99964 | 0.99995 | 4.82E-05 |
| **2.17898** | 0.00016425 | 0.99983 | 0.999981 | 1.86E-05 |
| **2.38902** | 7.08E-05 | 0.99992 | 0.99999 | 6.49E-06 |
| **2.620095** | 2.81E-05 | 0.99997 | 0.99999 | 2.04E-06 |
| **2.874285** | 1.02E-05 | 0.99999 | 0.99999 | 5.74E-07 |
| **3.153901** | 3.32E-06 | 0.999997 | 1 | 1.42E-07 |
| **....** | .... | .... | .... | .... |
| **....** | .... | .... | .... | .... |
| **153.20857** | 7.07E-267 | 1 | 1 | 0 |
| **168.52162** | 1.77E-293 | 1 | 1 | 0 |
| **185.36597** | 9.38725e-323 | 1 | 1 | 0 |
| **203.89476** | 0 | 1 | 1 | 0 |

**Table 5.** Result of Multiple Attack Detection

| **Parameters** | **Value** | |
|---|---|---|
| **Number of Sensor Nodes** | 50 | |
| **Number of IDS Agents** | 5 | |
| **Attack Types** | Detection Rate (%) | Accuracy (%) |
| **Selective Forwarding** | 88.4 | 87.1 |
| **DoS** | 84.7 | 85.2 |
| **Rank** | 89.3 | 88.4 |

## 5. Conclusion

Recently WSN with RPL protocol has grown rapidly and has a variety of applications in various fields. Security of this WSN is a major concern. Hence, IDS is the focus of the research. IDS for WSN is designed and implemented using machine learning and game theory. The QRE based extensive form of repeated game is designed to select the best strategies for intrusion detection. Further, random forest algorithm is used to generate the rules to identify multiple attacks. From the results, it can be concluded that the proposed game theory and machine learning model can be used effectively to identify the intrusions in WSN. The detection rate for three considered attacks is around 87%. Game model reduces the energy and computation cost. However, the comparison of IDS with and without Game model can be experimented. The model can be tested with the other possible attacks.

## References

[1] D. Li, Z. Cai, L. Deng, X. Yao, and H. H. Wang, "Information security model of block chain based on intrusion sensing in the iot environment," Cluster computing, vol. 22, no. 1, pp. 451-468, 2019.

[2] M. Carlos-Mancilla, E. L_opez-Mellado, and M. Siller, "Wireless sensor networks formation: approaches and t techniques," Journal of Sensors, vol. 2016, no. 1, 2016.

[3] S. M. Sajjad, S. H. Bouk, and M. Yousaf, "Neighbor node trust-based intrusion detection system for wsn," Procedia Computer Science, the 6th InternationalConference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2015), Germany, vol. 63, pp. 183 -188, 2015.

[4] B.-S. Kim, H. Park, K. H. Kim, D. Godfrey, and K.-I. Kim, "A survey on real-time communications in wireless sensor networks," Wireless Communications and Mobile Computing, vol. 1, no. 1, pp. 1-14, 2017.

[5] F. Bao, I.-R. Chen, M. Chang, and J.-H. Cho, "Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection." IEEE Transactions on Network and Service Management, vol. 9, no. 2, pp. 169-183, 2012.

[6] Rheo Malani, Arief Bramanto Wicaksono Putra, Muhammad Rifani "Implementation of the Naive Bayes Classifier Method for Potential Network Port Selection" International Journal of Computer Network and Information Security(IJCNIS), Vol. 12, No.2, pp 32-40, 2020

[7] B. Subba, S. Biswas, and S. Karmakar, "Intrusion detection in mobile adhoc networks: Bayesian game formulation," Engineering Science and Technology, an International Journal, vol. 19, no. 2, pp. 782-799, 2016.

[8] M. C. Belavagi and B. Muniyal, "Game theoretic approach towards intrusion detection," in 2016 International Conference on Inventive Computation Technologies (ICICT), India, vol. 1, no. 1, pp. 1-5. 2016.

[9] L. Chen and J. Leneutre, "A game theoretical framework on intrusion detection in heterogeneous networks," IEEE Transactions on Information Forensics and Security, vol. 4, no. 2, pp. 165-178, 2009.

[10] S. Shamshirband, A. Patel, N. B. Anuar, M. L. M. Kiah, and A. Abraham, "Cooperative game theoretic approach using fuzzy q-learning for detecting and preventing intrusions in wireless sensor networks," Eng. Appl. of AI, vol. 32, pp. 228-241, 2014.

[11] S. Shen, K. Hu, L. Huang, H. Li, R. Han, and Q. Cao, "Quantal response equilibrium-based strategies for intrusion detection in WSNs," Mobile Information Systems, vol. 2015, no 1 pp. 1-10, 2015.

[12] J. K. Goeree, C. A. Holt, and T. R. Palfrey, Front Matter *"Quantal Response Equilibrium: A Stochastic Theory of Games"* Princeton; Oxford: Princeton University Press.,2016.

[13] Sumit S. Lad, Amol C. Adamuthe "Malware Classification with Improved Convolutional Neural Network Model" International Journal of Computer Network and Information Security (IJCNIS), Vol. 12, No. 6, pp 30-43, 2020

[14] H. M. Salmon, C. M. De Farias, P. Loureiro, L. Pirmez, S. Rossetto, P. H.d. A. Rodrigues, R. Pirmez, F. C. Delicato, and L. F. R. da Costa Carmo, “Intrusion detection system for wireless sensor networks using danger theory immune- inspired techniques," International Journal of Wireless Information Networks, vol. 20, no. 1, pp. 39-66, Mar 2013.

[15] T. Sumeet, I. R. Ul, A. Karl, and H. M. Shahadat, “Ip based wireless sensor networks : performance analysis using simulations and experiments," Journal of Wireless Mobile Networks Ubiquitous Computing and Dependable Applications, vol. 7, no. 3, pp. 53-76, 2016.

[16] A. Mayzaud, A. Sehgal, R. Badonnel, I. Chrisment, and J. Schononwalder, “Mitigation of topological inconsistency attacks in rpl based low power lossy networks," International Journal of Network Management, vol. 25, no. 5, pp. 320-339, 2015.

[17] T. Zhang and X. Li, “Evaluating and analyzing the performance of rpl in contiki," in Proceedings of the First International Workshop on Mobile Sensing, Computing and Communication, New York, USA, pp. 19-24, 2014.

[18] A. Le, J. Loo, K. K. Chai, and M. Aiash,”A specification-based ids for detecting attacks on rpl-based network topology," Information, vol. 7, no. 2, 2016.

[19] J. Zhang, Y. Xiang, W. Zhou, and Y. Wang, “Unsupervised traffic classification using ow statistical properties and ip packet payload," Journal of Computer and System Sciences, vol. 79, no. 5, pp. 573-585, 2013.

[20] R. Hummen, J. Hiller, H. Wirtz, M. Henze, H. Shafagh, and K. Wehrle, “6lowpan fragmentation attacks and mitigation mechanisms," in Proceedings of the Sixth ACM Conference on Security and Privacy in Wireless and Mobile Networks, Hungary, pp. 55-66., 2013.

[21] Y. Maleh, A. Ezzati, Y. Qasmaoui, and M. Mbida, “A global hybrid intrusion detection system for wireless sensor networks," Procedia Computer Science,6th International Conference on Ambient Systems, Networks and Technologies (ANT-2015), the 5th International Conference on Sustainable Energy Information Technology (SEIT-2015), United Kingdom vol. 52, pp. 1047 - 1052, 2015.

[22] K. Weekly and K. Pister, “Evaluating sinkhole defense techniques in rpl networks," in Proceedings of the 2012 20th IEEE International Conference on Network Protocols (ICNP), USA, pp. 1-6, 2012.

[23] M. Fallah, “A puzzle-based defense strategy against flooding attacks using game theory," IEEE Transactions on Dependable and Secure Computing, vol. 7, no. 1, pp. 5-19, 2010.

[24] H. Narasimhan, V. Varadarajan, and C. P. Rangan, “Game theoretic resistance to denial of service attacks using idden difficulty puzzles," in Information Security, Practice and Experience, 6th International Conference, ISPEC 2010, Seoul, Korea, pp.359-376, 2010.

[25] M. Q. Imed Romdhani, Ahmed Yassin Al-Dubai and B. Ghaleb, “Cooja simulator manual," Edinburgh Napier University, Tech. Rep., 2016.

[26] G. Montenegro, N. Kushalnagar, J. Hui, D. Culler et al.,”Transmission of ipv6 packets over ieee 802.15.4 networks," Internet proposed standard RFC 4944, PP1-130, 2007.

[27] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, R. Alexander et al., “Rpl: Ipv6 routing protocol for low-power and lossy networks.”, RFC 6550, pp. 1-157, 2012.

| | | | | | | |
|---|---|---|---|---|---|---|
| 22 | 10.677000 | fe80::212:7401:1:101 | fe80::212:7 | ICMPv6 | 68 | Neighbor Solicitation for fe80::212:7402:2:202 from 00:12:74:01:00:01:01:01 |
| 23 | 12.800000 | fe80::212:7403:3:303 | fe80::212:7 | ICMPv6 | 68 | Neighbor Solicitation for fe80::212:7401:1:101 from 00:12:74:03:00:03:03:03 |
| 24 | 12.826000 | | | IEEE 802 | 7 | Ack |
| 25 | 12.902000 | fe80::212:7401:1:101 | fe80::212:7 | ICMPv6 | 68 | Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01 |
| 26 | 12.931000 | fe80::212:7401:1:101 | fe80::212:7 | ICMPv6 | 68 | Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01 |
| 27 | 12.932000 | fe80::212:7401:1:101 | fe80::212:7 | ICMPv6 | 68 | Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01 |
| 28 | 12.933000 | fe80::212:7401:1:101 | fe80::212:7 | ICMPv6 | 68 | Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01 |
| 29 | 12.934000 | | | IEEE 802 | 7 | Ack |
| 30 | 13.526000 | 2002:db8::212:7405:5:505 | 2002:db8::1 | UDP | 61 | Source port: ultraseek-http Destination port: rrac |
| 31 | 13.555000 | | | IEEE 802 | 7 | Ack |

**Figure 6.** Simulated WSN traffic