PeerJ Computer Science

A multiplayer game model to detect insiders in wireless sensor networks

Ioanna Kantzavelou[1], Leandros Maglaras[2], Panagiotis F. Tzikopoulos[3] and Sokratis Katsikas[4]

[1] University of West Attica, Athens, Greece
[2] Cyber Technology Institute, De Montfort University Leicester, Leicester, UK, United Kingdom
[3] Banking and Financial Management, University of Piraeus, Piraeus, Greece
[4] Norwegian University of Science and Technology, Gjøvik, Norway

ABSTRACT

Insiders might have incentives and objectives opposed to those of the belonging organization. It is hard to detect them because of their privileges that partially protect them. In Wireless Sensor Networks (WSNs), significant security issues arise, including compromised nodes by insiders that disrupt the normal network operation. Immediate defensive actions to isolate malicious nodes would mitigate any related impacts. A multiplayer game model is proposed as a solution to the problem of insider attacks in WSNs, the Game of Wireless Sensor Networks (GoWiSeN). It is an imperfect information game, formulated with the use of non-cooperative game theory, holding the assumption that all players are rational. The model consists of several Local Intrusion Detection Systems (LIDSs), which are located to different nodes and communicate with a Global Intrusion Detection System (GIDS). Each LIDS gives suggestions whether the monitoring node is trusted or not. The game is being played between a potential attacker, the nodes and the GIDS. The GIDS is responsible for making a final decision and for isolating a compromised node in case of an internal attack. The theoretical model represents these interactions in an extensive form game. The formal elements of the game are specified, the outcomes of the game are quantified by first specifying players' preferences, and then, by using the von Neumann-Morgenstern utility function, and payoffs are obtained. The game is constructed and solved, by locating NE in pure and mixed strategies. Experimental evaluations conducted on real network datasets, using IDSs of different capabilities, simulate special cases and compromised nodes in a WSN, verify the model efficiency, and show how the game should be played.

Subjects Computer Networks and Communications, Security and Privacy
Keywords Intrusion detection, Game theory, Wireless sensor networks, Multiplayer game

INTRODUCTION

Wireless Sensor Networks (WSNs) were introduced some years ago as a new technology that combines wireless communication (*Alcaraz, 2019*), computation, and sensing (*Stankovic, 2008*). The great range of applications has made WSNs very popular and the need for simple and familiar interactions more essential than ever. As part of pervasive computing environments, WSNs raise fundamental security issues. Attacks on sensor networks routing (novel attacks-sinkhole and HELLO floods) have imposed new design for secure routing protocols (*Karlof & Wagner, 2003*). Detection of masquerade attacks on

Submitted 6 November 2020
Accepted 1 November 2021
Published 20 January 2022

Corresponding author
Leandros Maglaras,
leandros.maglaras@dmu.ac.uk

Academic editor
Kezhi Wang

Additional Information and Declarations can be found on page 28

DOI **10.7717/peerj-cs.791**

WSNs requires lightweight techniques with respect to important WSN properties, like coverage, connectivity, data aggregation and specific communication patterns (*Bhuse & Gupta, 2006*). Such characteristics have generated special attacking methods to WSNs, as the sleep deprivation attack (*Pirretti et al., 2005*), the time synchronization attack (*Manzo, Roosta & Sastry, 2006*), and the selective forwarding attack (*Yu & Xiao, 2006*). Several taxonomies of attacks on WSNs has been proposed in *Han et al. (2005)*, *Nawir et al. (2016)*.

Intrusion Detection plays an active role in cybersecurity and many different technologies, tools, and approaches have been used in conjunction with it, as blockchain technology (*Agarwal et al., 2021*), machine learning (*Dua & Du, 2016*), and fuzzy logic (*Elhag et al., 2015*). As an important area of research, Intrusion detection has been applied in WSNs to enhance their security, despite the fact that it is used as a second line of defense. Classical detection techniques have been employed over lightweight Intrusion Detection Systems (IDSs), like the anomaly intrusion detection technique in *Bhuse, Gupta & Al-Fuqaha (2007)*, to detect and deter attacks that affect the normal and uninterruptible operation of WSNs.

One of the main security problems in WSNs is the problem of compromised nodes, as defined in *Zhang, Yu & Ning (2006)*. Although cryptographic mechanisms are used to protect sensor networks from masquerading attacks, attackers might compromise a node by stealing a key, and introducing afterwards faulty data from this compromised node. There are two approaches to address this problem; either distinguish faulty data from real data, or detect which node is the compromised node and exclude it from the network. The first approach has been mainly employed by other related works, but without significant results. *Abdalzaher et al. (2016)* introduced the concept of evolutionary game using a group policy/authentication method to resist intelligent attacks which do not use pure strategies.

Since a great number of attacks against sensor network routing originated by outsiders can be evaded by the use of authentication and encryption mechanisms (*Karlof & Wagner, 2003*), insider attacks are the most challenging and demanding to be counteracted. The proposed work addresses the problem of detecting insider attacks in WSNs, by taking the advantages of intrusion detection when incorporating findings of game theory. The current manuscript is an extended version of the previously published article in *Kantzavelou, Tzikopoulos & Katsikas (2013)*. Compared to the previous publication, it incorporates evaluation results of the IDSs under a realistic Internet of Things (IoT) dataset, which affect the players' payoffs, analyses specific cases where nodes belong to more than one clusters, and discusses the situation where a Local Intrusion Detection System (LIDS) is compromised.

In this paper, a game model between a potential attacker, and the IDSs used in a WSN is proposed. The potential attacker is an internal user of the system, who acts normally most of the times, but occasionally attacks the system, by compromising a node of the WSN.

Following Osborne's and Rubinstein's dimensions (*Osborne & Rubinstein, 1994*), the *player*, the *plan of actions*, and the *information*, upon which three divisions of game theoretic models are based, a *non-cooperative* game theoretic model is constructed, in

extensive form that allows each player to think about his plan of actions whenever he plays, formulating the sequential moves of interactions, with *imperfect information*.

The under construction game is named *GoWiSeN* as an abbreviation of *'Game of Wireless Sensor Networks'*. According to prescriptive game theory, theoretical examination of the constructed game, allows us to determine how players should play it, and to recommend strategies. As a result, it is possible to give advice that helps players to make better decisions.

The main contributions of the article are summarized in the following:

- A three player non-cooperative game is modeled, to study the interactions between an insider and the IDSs used in a WSN (Section The Game).
- The solution of the game by locating Nash Equilibria in mixed strategies is provided (Section Solution of the Game).
- Evaluation of the model is conducted using a dataset from realistic network environment that includes normal and botnet traffic (Section Experimental Evaluations).
- Discussion on special cases of nodes that belong to overlapping clusters is given (Section Special Cases).
- The situation where a LIDS is compromised is examined (Section Special Cases).

The paper has been organized in nine sections. In the Related Work section, review related works and discussion against the proposed work is provided. The architecture of the proposed model is illustrated in The Architecture of the Model section. The game that models the interactions between the different parties of this architecture is constructed and solved in The Game and Solution of the Game sections. A complete case study is presented in the A Case Study section and two different scenarios are described to explain the implementation and functioning of the game model. In the Experimental Evaluations simulated results on a realistic dataset are presented. In the Special Case section, two special cases are discussed with regards to how they can be modeled and analyzed. Finally, in the Conclusion and Future Work section, our research work is summarized by evaluating the model and its operation, and suggesting future directions.

RELATED WORK

The problem of detecting attackers in WSNs has been addressed by various approaches. The ultimate aim is the discrimination between compromised nodes and normal ones. A malicious node should be isolated as soon as identified, because it has no trust value and prohibits the normal network operation. Among the large number of constraints inherited in WSNs, the most important are; the low computational capability, and the limited recourses regarding memory and energy. In addition to these, security issues enhance, in many ways, their vulnerabilities and make them easy targets.

A number of detection engines proposed as a solution to this problem are presented in *Sharma & Athavale (2019)*, through the discussion of their corresponding research works that incorporate different tools, technologies and methods. Probabilistic models, game

theoretic approaches, K-means clustering, artificial immune systems, distributed anomaly detection techniques, genetic algorithms and Random Neural Networks (RNN) are a sample of these solution approaches. The authors give valuable conclusions for the importance of detection in WSNs and the possible architectures that might be incorporated in constructing dedicated IDSs for these type of networks.

Game Theory acts as a set of tools to model interactive situations. Camerer describes it as the answers to mathematical questions regarding what players with ranging rationality will do in the future (*Camerer, 2003*). Game theoretic approaches consequently have gained favor in many research works the last few years (*Naseer et al., 2021*), many related to cybersecurity and specifically to intrusion detection. *Kiennert et al. (2018)* have investigated different solutions that might improve the efficiency of intrusion detection systems with the use of game theory.

Another survey on game theoretic approaches used in WSNs concentrates on three main problems; sensor's energy efficiency, network security, and pursuit-evasion games (*Machado & Tekinay, 2008*). The majority of the works described tackle the sensor's energy efficiency problem, though network security and pursuit-evasion games are very significant too. Examining the problem of security, two types of threats are mentioned, the external attacker, and the malicious nodes within the sensor network. Another interesting survey on game theoretic approaches for WSNs can be found in *Shi et al. (2012)*.

The problem of compromised nodes has been addressed by *Zhang, Yu & Ning (2006)*. They propose an application-independent framework for identifying compromised sensor nodes, and they develop alert reasoning algorithms for this identification. Their technical approach uses an observer model.

LIDSs have been defined in an adapted architecture for an intrusion detection system for manets in *Albers et al. (2002)*. In another work *Ma et al. (2006)*, take advantages of LIDSs to have a tradeoff among the security of WSN and communication overhead.

To address security problems mentioned above, *Subba, Biswas & Karmakar (2018)* propose in, a game theoretic multi layered intrusion detection framework for WSNs. The proposed framework uses specification rules and a lightweight anomaly detection module to identify malicious sensor nodes. The framework models the interactions between the IDS and a node as a two player non-cooperative Bayesian game, which guides the IDS how to choose proper strategies based on the Bayesian Nash Equilibrium. The game is supported by two mechanisms that strengthen cooperation, the Shapley Value and the Vickery–Clark–Grooves (VCG) mechanism.

Han et al. (2019) propose in a game theoretic solution approach to solve two problems in WSNs; the limited resources and the efficiency in detecting malicious nodes. A non-cooperative, complete-information, static game model is constructed and solved using the Nash equilibrium approach, to isolate the optimal defense strategies that balances the system's detection efficiency and energy consumption. In addition, an autoregressive model is built as a prediction model to locate the attacker's target node.

In *Wang et al. (2016)*, focus on the area of Intrusion Detection to solve security problems in Cyber-Physical Embedded Systems (CPESs), admitting that existing security

mechanisms cannot directly be applied to Embedded Sensor Networks (ESNs). They propose a new attack-defense game model to detect malicious nodes and choose to play the game repeatedly. The IDS takes into consideration two important parameters; error detection and missing detection. The proposed model achieves to reduce energy consumption and increase the detection rate, and thus enhance ESNs security using this active defense approach.

The problem of IoT heterogeneous devices connected to untrusted networks is discussed in *Sedjelmaci, Senouci & Taleb (2017)*. The authors expose the advantages gained by the combination of the anomaly and the misuse detection techniques in a single detection engine, which reduces the high false positive and false negative alarms respectively. But, this simultaneous activation in low-resource IoT devices could generate a high-energy consumption. To overcome this drawback, *Sedjelmaci, Senouci & Taleb (2017)* propose a game theoretic approach that activates anomaly detection technique only when a new attack's signature become evident. The anomaly detection technique models the normal behavior of a node through a learning algorithm, and when a new attack pattern appears, it models it following a set of rules. Moreover, a reputation model based on game theory supplements the aim of reducing false rates.

Bai et al. (2019) construct a game to model the attack and defense interactions between nodes and an IDS in WSNs. The aim is to determine the optimal defensive strategies an IDS should select in order to reduce energy consumption and improve detection efficiency. The scalability of the system is another problem addressed by the incorporation of the agent technology, which also leads to system fault tolerance enhancement.

The incorporation of trust management models in WSNs as an alternative security mechanism has been suggested in different research works (*Han et al., 2014*). They aim at detecting malicious attacks, secure routing, secure data aggregation, secure localization, or secure node selection. The common point in most of them is trust. Some research works that employ a trust model to detect malicious nodes in WSNs are discussed in the next paragraphs.

Wu et al. (2015) start by pointing out two WSN features, the openness feature imposed by the wireless connectivity and the inherent self-organization feature in WSNs, and both reveal a significant concern regarding the trust evaluation of network nodes. By accepting the difficulty in recognizing a node's behavior and making a decision over it, they propose a new trust model based on fuzzy theory and revised evidence theory, to detect nodes that have anomaly behavior. It is a trust-based anomaly detection model that aims at verifying the normal operation of the network by identifying malicious nodes in it (*Wu et al., 2015*).

A hybrid IDS for WSNs that combines the anomaly and the misuse detection technique is presented in *Singh, Singh & Singh (2017)*. The proposed system is based on fuzzy rule sets along with the Multilayer Perceptron Neural Network and addresses three types of attacks, the Sybil attack, the wormhole attack, and the hello flood attack, by adopting specific algorithms for each of them. Experimental results show the effectiveness of the

presented IDS in detecting malicious nodes in a WSN, with high true positive rate and low false positive rate.

A trust evaluation model based on an entropy weight assignment method is proposed in *Yin & Li (2019)* for the detection of malicious nodes in WSNs. The work addresses attack types related to packet dropping or packet modification by incorporating trust indicators and estimates trust values that reflect adjacent nodes behavioral information. The use of an entropy weight method improves the evaluation procedure and the trust concluding results, which lead to the detection of malicious nodes.

Another method to detect insiders in WSN is a trust-based mechanism presented in *Meng, Li & Kwok (2013)*. A Bayesian model is deployed in a detection mechanism to isolate malicious nodes and permit the benign ones to continue the normal network operation. This trust model allows the establishment of trust values in an hierarchical structure, which reduces the network traffic and facilitates detection. The authors conclude that the proposed method shows experimentally that a Bayesian trust model is a suitable solution for the detection of malicious nodes in WSNs comparing to their corresponding work in wired networks.

An alternative approach to build a detection model, adequate for WSN, with a novel nonparametric Bayesian method is proposed in *Alhakami et al. (2019)*. This method works with no need to specify parameters such as the number of clusters to detect both known and unknown attacks. A Bayesian-based MCMC inference for infinite bounded generalized Gaussian mixture models is used to learn the event patterns. Experimental evaluation results show the efficiency of the proposed method to detect a few types of attacks.

Another trust model based on clustering is proposed in *Jaint et al. (2018)*. A weighted trust method is examined in a WSN that consists of a base station, a number of sensor nodes divided into some clusters, a node of each cluster selected to be the head of it, and a forward node that transmits aggregated data to the base station. The method aims at malicious node detection. Different scenarios regarding the clustering structure show that the detection time is less, the accuracy and the scalability are better, and the propagation time is less, when clustering includes multiple cluster heads with non-overlapping grid comparing to single cluster head without grid.

A detailed comparison of different trust models and the proposed GoWiSeN with respect to (1) method, (2) objective, (3) strengths, and (4) weaknesses is provided in Table 1.

The proposed work focuses on the special case of internal attackers, and formulates a game model with three players, one assumed as a potential attacker (insider), an IDS monitoring a node of a WSN, named Local Intrusion Detection System (LIDS), and another one, located at the base station that cooperates with many LIDSs, named Global Intrusion Detection System (GIDS). The games played are between different LIDSs, the GIDS and an insider, as imposed by the model architecture in multiplayer game constructions. To the best of our knowledge, this is the only game-theoretic approach for intrusion detection in WSNs that constructs and solves multiplayer games for insiders.

Table 1 Comparison of trust models.

Strengths and Weaknesses

Name	Trust model	Method	Objective	Strengths	Weaknesses
A game of wireless sensor networks (GoWiSeN)	A multiplayer game model, game theory	Game theoretic approach over IDS decisions.	Insider attacks in WSNs	Construction and solution of a three-player game with insiders. Game outcomes determined from a real network dataset. Examination of a compromised LIDS scenario.	Preferences reflect one type of attackers only.
Trust-based anomaly detection model (*Wu et al., 2015*)	Fuzzy trust model, Fuzzy Theory and Revised Evidence Theory	Trust-based anomaly detection technique with a weighting algorithm.	Detection of nodes that have anomaly behavior.	Energy efficient anomaly detection scheme towards practical application.	The evaluation node only accepts the packets from one-hop neighbors.
Advanced hybrid intrusion detection system (AHIDS) model (*Singh, Singh & Singh, 2017*)	Fuzzy trust model, Fuzzy Theory.	A hybrid IDS for WNS, Fuzzy rule sets along with the Multilayer Perceptron NN.	Detection of malicious nodes in WSNs.	Specific algorithms for each attack. Effective in detecting malicious nodes in WSNs.	Limited attack types. Extension to address more requires new specific algorithms.
Trust evaluation model (*Yin & Li, 2019*)	Entropy-based model.	Based on an entropy weight assignment method.	Detection of malicious nodes in WSNs.	The entropy-based weight assignment improves the objectivity of trust evaluation and obtains fast convergence rate.	Limited attack types. It addresses packet dropping and packet modification attacks.
Bayesian model (*Meng, Li & Kwok, 2013*)	Bayesian trust model.	Bayesian trust-based mechanism.	Detection of insiders in WSNs.	Trust values are established in an hierarchical structure to reduce network traffic and facilitate detection. More suitable for WSNs.	The work is developed at an early stage. The WSN is very limited.
Non-parametric Bayesian model (*Alhakami et al., 2019*)	Bayesian model.	Bayesian-based MCMC inference.	Detection of known and unknown atatcks in WSNs.	This method works with no need to specify parameters such as the number of clusters.	It addresses general network security detection problems rather than specific to WSNs.
Clustering trust model (*Jaint et al., 2018*)	Clustering trust model.	Cluster based weighted trust evaluation method.	Detection of malicious nodes in clustered WSNs.	Detection and propagation time are less, and the accuracy and the scalability are better with multiple cluster heads with non-overlapping grid.	Limited experimentation.

Prior work has shown that, in the special case of insiders, Intrusion Detection can efficiently be employed in a game theoretic framework, to address significant security problems in WSNs (*Kantzavelou, Tzikopoulos & Katsikas, 2013*). Several other recent works study and take into account these preliminary results (*Li et al., 2017*; *Ranaweera et al., 2019*; *Sánchez, Parra & Medina, 2019*), and employ similar conceptual approaches. *Li et al. (2017)* recently designed an intrusion sensitivity-based trust management model for WSNs to defend against insider attacks. In their proposed work, each IDS evaluates the trustworthiness of others and automatically assigns the values of intrusion sensitivity with the use of machine learning techniques. The cooperation between different IDSs resembles the combined working of LIDSs and the GIDS, which was proposed in *Kantzavelou, Tzikopoulos & Katsikas (2013)*. *Ranaweera et al. (2019)* also give an extension

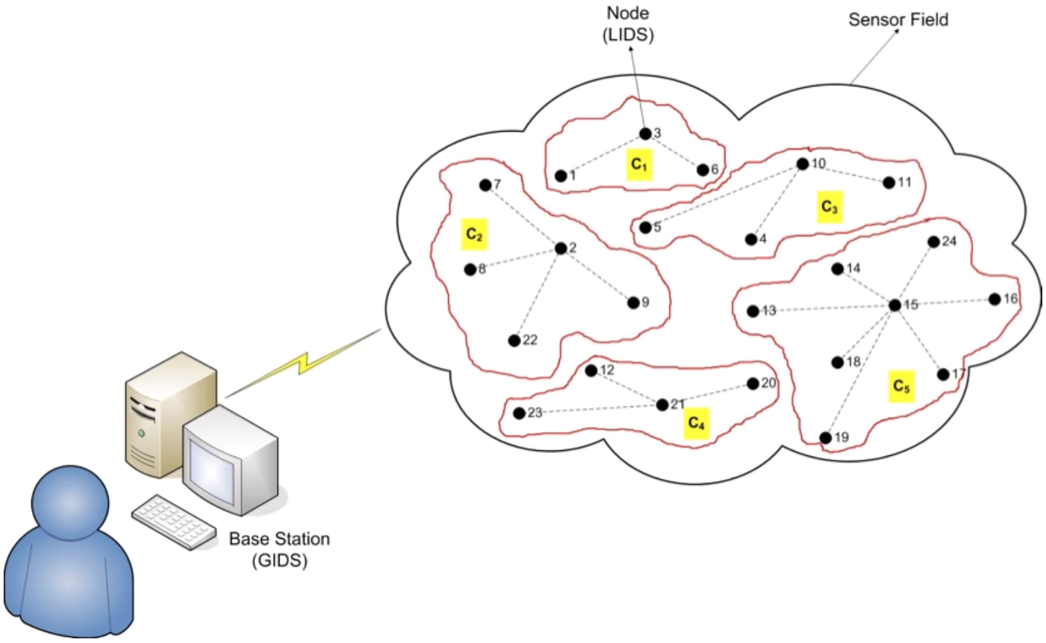

Figure 1 **The architecture of the model.** Full-size DOI: 10.7717/peerj-cs.791/fig-1

by applying traffic flow theory in order to identify anomalous data in vehicular networks, and provide reliable and consistent predictions against incorrect decisions. Finally, in *Sánchez, Parra & Medina (2019)*, a game theoretic model for intrusion detection in WI-FI networks is presented.

THE ARCHITECTURE OF THE MODEL

We have assumed a WSN with a large number of nodes, densely scattered in an area. This network consists of a set C of n clusters, $C = \{C_1, C_2, \ldots, C_n\}$. Each cluster covers a number of nodes, quantified as m_i for the i_{th} cluster, that communicate one another and each with a base station. Thus, the set of nodes for the i_{th} cluster is $N_i = \{N_{i1}, N_{i2}, \ldots, N_{im}\}$. Although the problem of positioning base stations in a sensor network is already under consideration (*Bogdanov, Maneva & Riesenfeld, 2004*), we have chosen the use of a single base station, for simplicity reasons.

In this network, there is a Global IDS, the GIDS, established on a separate machine, at the base station. This GIDS receives signs of suspicious or trusting activities from the total number of Local IDSs (LIDSs), E, as calculated in (1), each one installed on a node of the network. The set of the m_i LIDSs included in the ith cluster of the network is $L_i = \{L_{i1}, L_{i2}, \ldots, L_{im}\}$, sketched in Fig. 1.

$$E = \sum_{i=1}^{n} (m_i) \tag{1}$$

The notation adopted in this paper is summarized in Table 2.

A LIDS is a light version of a classical IDS adopted to run on a node of a sensor network. Its efficiency is absolutely affected and limited by the local memory, the battery power and

Table 2 Notation used in this paper.

Notations	
C	Set of n clusters
C_i	The i_{th} cluster
m_i	The number of nodes in the i_{th} cluster
N_i	The set of nodes for the i_{th} cluster
N_{im}	The m_{th} node of the i_{th} cluster
E	The total number of LIDSs in a WSN
L_i	The total number of m_i LIDSs in the i_{th} cluster
L_{im}	The m_{th} LIDS in the i_{th} cluster

all related characteristics inherent in a node of a sensor network. All these constraints prevent the direct employment of ordinary operating systems and applications on sensor nodes (*Wittenburg & Schiller, 2006*). Therefore, alternative OSs have been developed to support the special requirements of these devices, like the most common open source TinyOS (*Hill et al., 2000*), created at UC Berkeley. Likewise, special software design patterns have been introduced to be employed in the TinyOS (*Gay, Levis & Culler, 2007*). Moreover, specific applications implemented for WSNs use the C standard programming language. Such a decision has been implied as this is the actual language used for embedded systems that work under constraints in memory and power, and limited computational capabilities (*Wittenburg & Schiller, 2006*).

Figure 1 depicts a network of this architecture, with a sensor field of five clusters, and a ranging number of nodes included in each of them. Every node of a cluster communicates with all others in the same cluster and connects with the base station. The GIDS has been installed on the base station and twenty four LIDSs have been set up to work on the corresponding nodes.

THE GAME

Suppose a user is using the sensor network from a node of a cluster. He is a legitimate user and he has specific rights granted from the system, in accordance with its security policy. But the user is a potential attacker, who is acting normally and intrusively, depending on what goals he wants to achieve. When he is acting normally, the node looks as if it were a regular node of the network, misleading that it works properly.

Consider a case where this user requests resources from the network, breaching the security policy of the system. In other words, the user tries to get from the system resources but he has no right to do so. There are two possible reasons that this might happen; either the user has the right to get this resource but his request exceeds some specified limits, or the user tries to get resources but he is not authorized. In both situations, the user exploits the fact that he is a user of the system, and as so, he has some rights to use it. So, he is authorized for certain actions. Therefore, under these circumstances, it is a challenging task to draw a line that separates the user between normal and attacker.

A game with three players is constructed to model what described in Section The Architecture of the Model. The number of players has been decided and imposed by the architecture of the model (Fig. 1). In an extended version of the proposed game there might be four players or more that could trigger distributed attacks against a WSN. But in non-cooperative Game Theory, the more players play the game the more difficult it is to solve (*Daskalakis, 2008*; *Daskalakis, Goldberg & Papadimitriou, 2009*; *Roughgarden, 2010*). Daskalakis and Papadimitriou prove in *Daskalakis & Papadimitriou (2005)* that computing a Nash equilibrium in a three player game is a problem that belongs in the PPAD-complete class, a class defined to address the problem of finding a NE in polynomial time. Furthermore, they prove in *Daskalakis, Goldberg & Papadimitriou (2005)* that finding a Nash equilibrium in a 4-player game is also PPAD-complete.

In order to define the game, the following formal elements should be specified:

- **The list of *players*.** There are three players, the Potential Attacker (PA), the Local Intrusion Detection System (LIDS), and the Global Intrusion Detection System (GIDS).
- **Their possible *actions*.** PA's strategy set includes two action types, the *normal action* type, and the *attacking action* type. LIDS's actions are two, the *suggestion for suspicious* and the *suggestion for trusting*. The GIDS's actions are also two, the *exclude* action and the *admit* action.
- **What the players know when they act.** When the game starts, no player has enough information, and thus they act under great uncertainty. PA might know that LIDSs and a GIDS monitor the WSN. The LIDSs and the GIDS keep history traces on how the game is being played, valuable information for next rounds.
- **The *outcomes* of the players' actions.** The total number of the possible outcomes of the game is eight, which derives from all possible combinations between the three players' actions ($2 * 2 * 2$).
- **The players *preferences* over these outcomes.** To quantify the outcomes of the game, first there is a need to specify preferences over outcomes. The von Neumann-Morgenstern utility function is used. A player prefers a strategy over another, because he gains more or he loses less. Following Binmore's method (*Binmore, 2007*), numbers are assigned to reflect these preferences, and all players' utility functions are constructed. Next, 0 is set to the least preferred strategy and 1 to the most preferred strategy. Using rational numbers, a value is assigned to every strategy, according to corresponding rankings. Then, the values free of fractions are obtained, after multiplying with their least common factor. The results are the payoffs of the game.

The first player is a user of the system who is a Potential Attacker (PA) under certain circumstances, namely a node of the sensor network. It is assumed that the PA is anyone excpet the system administrator, because in this situation, the user could be fully authorized and no distinction can be easily made between *normal action* and *attacking activity*. The second player is the Local IDS (LIDS), hosted by the node responsible for making suggestions for *suspicious* or *trusting* activities, triggered by the PA. Finally, the third player is the Global IDS (GIDS) that resides in the base station of the network, takes into account the LIDS suggestions, examines the history of the corresponding node's activities, and decides whether to *exclude* or to *admit* what the user has requested.

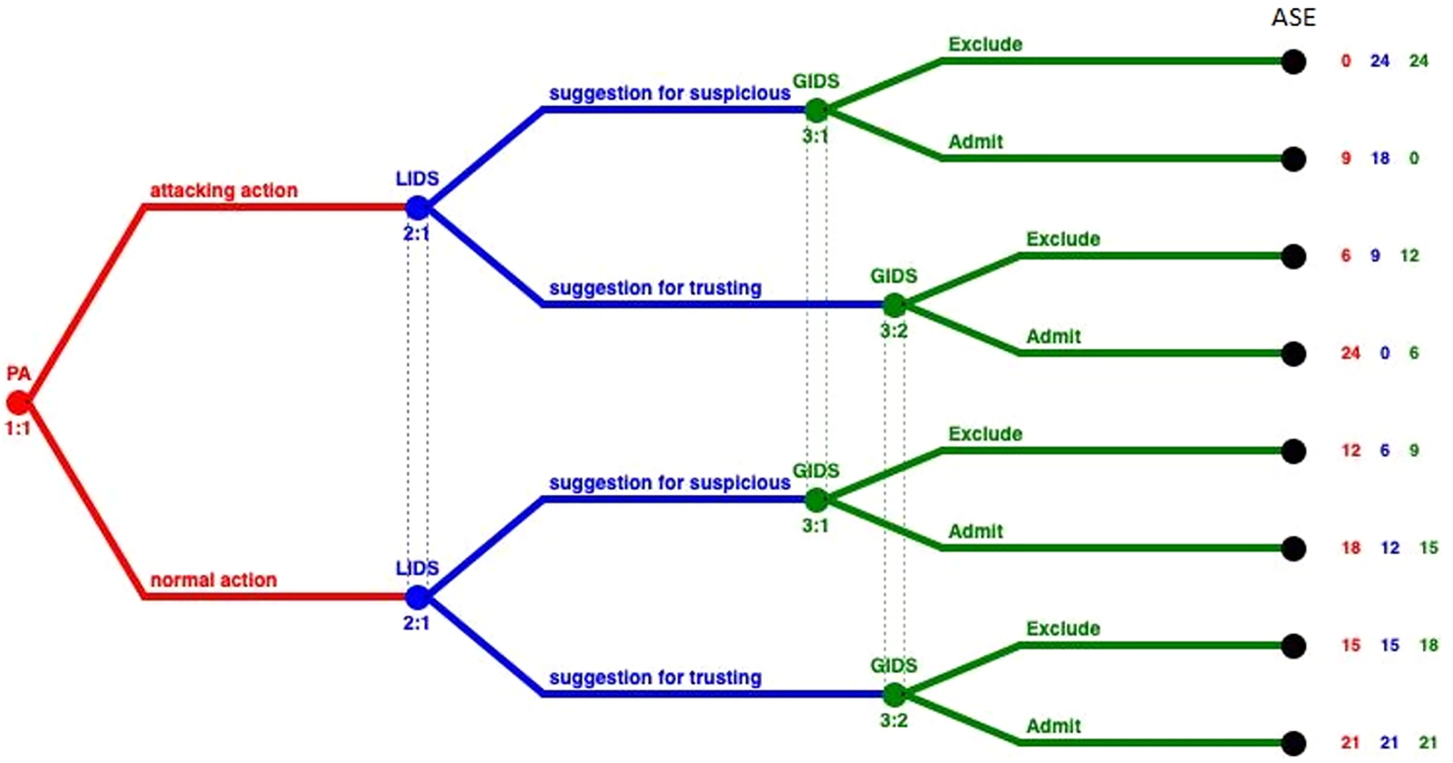

Figure 2 **GoWiSeN in an extensive form game.** Full-size DOI: 10.7717/peerj-cs.791/fig-2

Summarizing, player PA has the two strategies *normal action* and *attacking action*, player LIDS has the two strategies *suggestion for trusting* and *suggestion for suspicious*, and player GIDS has the two strategies *admit* and *exclude* a user's action as normal or intrusive respectively.

There are two main forms for the construction of a game that represents a problem; the normal and the extensive forms. In the normal form games players act simultaneously, whereas in extensive form games players act sequentially, the one after the other. We have chosen the extensive form for the construction of the proposed game model to represent appropriately the interactions that take place between an internal attacker, a LIDS and the GIDS (*Osborne, 2004*).

Figure 2 represents the extensive form of the GoWiSeN. Extensive form games are portrayed by trees. Player *PA* moves first at the initial node (the root) of the game, denoted by a red circle. The player's name is displayed above the node. Below the node, the default labeling is the information set's number. It is a unique identifier of the information set, in the form *player number*:*information set number* (*e.g.* 1:1 means the first move of the first player, *i.e.* the first move of player *PA*). Von Neumann defined information sets to model the progressive learning of which decisions will actually be made (*Binmore, 2007*).

Similarly, player *LIDS*'s moves start at blue circles, above which *LIDS* is written, and below, a corresponding pair, labeling its information set's number (2:1 means that the second player, player *LIDS*, moves for the first time).

It is assumed that players *LIDS* and *GIDS* are not totally certain that player *PA* has chosen one of the actions included in his action set. This is consistent with the hypothesis that there is no detection engine with 100% detection rate. Thus, player *LIDS*'s (or *GIDS*'s) sub-trees belong to the same information set, connected with a dotted line to indicate this. In short, the dotted line connects player *LIDS*'s (or *GIDS's*) nodes to indicate the Local or the Global IDS accuracy respectively, and thus, the degree of uncertainty whether player *LIDS* (or *GIDS*) has chosen a (*suggestion for suspicious*) or (*suggestion for trusting*) action (or an (*Exclude*) or (*Admit*) action).

Looking at the ends of the branches, 8 outcomes are identified. The number of outcomes derives from all the possible combinations between the PA's actions and the Local and Global IDSs' actions (2^3). There are three capital letters at the end of the upper branch of the tree only and above the node as an example, that denotes player *PA*'s, player *LIDS*'s, and player *GIDS*'s choices, respectively. Specifically, (*ASE*) means that player *PA* has chosen an (*attacking*) action, player *LIDS* followed with a *suggestion for suspicious* action, and player *GIDS* chose afterwards to *Exclude PA*'s action.

Finally, the tuple of three numbers next to each end node is the players' payoffs, that is to say, the outcome a player receives when a certain action has been chosen, represented as a number. The outcomes of the game are quantified, by first specifying preferences over outcomes, and then by using the von Neumann-Morgenstern utility function (*Binmore, 2007*). The red number belongs to player *PA*, the blue number belongs to player *LIDS*, and the green number belongs to player *GIDS*.

The players play the game repeatedly an infinite number of times. The reason is that, the user is not a random attacker, but an internal user of the system, who spends a long time every day in front of it. We assume him as a traitor rather than a masquerader. In repeated games, the actions are called strategies to distinguish them from the actions in the stage game.

SOLUTION OF THE GAME

In noncooperative game theory, the NE is the most commonly used solution concept. A Nash equilibrium of a game is a set of players decisions that results in an outcome, such that, no player has any reason to deviate from his choices, given that all the players do the same. John Nash proved that every noncooperative game has at least one Nash equilibrium (NE) (*Nash, 1950*; *Holt & Roth, 2004*). When no NE exists in pure strategies, then there is at least one in mixed strategies. In games with more than one NE, the problem of multiple NE and which one to choose appears (*Osborne, 2004*).

We proceed to solve the game as a three player game. A three player game obviously is more complicated than the two player games with a more complex tree. For each round of the game, the tree has three different levels. This increases the size of the tree, which makes the solution of the game difficult to be located.

In a three player game, where each player has a limited number of strategies, a matrix can be used. The matrix should have three dimensions with the third dimension devoted to the third player strategies. In practice, this is easily accomplished with an 'add pages' technique (*Dixit & Skeath, 1999*). The first page of the matrix depicts the payoffs of the first

Table 3 1st page of the matrix-Attacking action strategy of PA.

PA chooses an attacking action

		Global IDS	
		Exclude	**Admit**
Local IDS	Suggestion for suspicious	0, 24, 24	9, 18, 0
	Suggestion for trusting	6, 9, 12	24, 0, 6

Table 4 The 2nd page of the matrix-normal action strategy of PA.

PA chooses a *normal action*

		Global IDS	
		Exclude	**Admit**
Local IDS	Suggestion for suspicious	12, 6, 9	18, 12, 15
	Suggestion for trusting	15, 15, 18	21, 21, 21

strategy of the third player. The second page of the matrix depicts the payoffs of the second strategy of the third player, etc. Tables 3 and 4 depict the two pages of the matrix that describes the GoWiSeN game, which correspond to the two strategies PA might choose, the *attacking action* or the *normal action* respectively. Each page has two rows for player LIDS' strategies (*suggestion for suspicious* and *suggestion for trusting*) and two columns for player GIDS' strategies (*Exclude* and *Admit*). The two pages follow one another to represent the same instance. There is a pair of three numbers in each cell, which correspond to the payoffs of the PA, the LIDS, and the GIDS respectively.

Removing dominated strategies

We first solve the GoWiSeN game by applying the domination criterion, which says that a rational player should not use a dominated strategy. *Binmore (2007)* expresses the domination criterion by assuming two strategies s_1 and s_2 of a player I and three strategies t_1, t_2, and t_3 of a player II. Then we decide that for player I, strategy s_2 strongly dominates strategy s_1 when

$$\pi_1(s_2, t) > \pi_1(s_1, t) \tag{2}$$

for all three values of player II's strategy t. Moreover, if the relation between two strategies is $\geq$, then the one strategy weakly dominates the other.

In our game we express in algebraic terms the above criterion to check if it holds. First, we consider that player *PA* chooses an *attacking action* (Table 3). In this case, given that player *GIDS* chooses *Exclude*, player *LIDS* will choose strategy *suggestion for suspicious*, which dominates *suggestion for trust*, because 24 > 9. Similarly, given that player *GIDS* chooses *Admit*, player *LIDS* will choose again strategy *suggestion for suspicious*, which dominates *suggestion for trust* too, because 18 > 0.

Using this domination argument, we remove strategy *suggestion for trust* from the payoff matrix of Table 3 and the matrix changes to the following (Table 5):

Table 5 The 1st page of the matrix-attacking action strategy of PA, altered by removing strategy *suggestion for trust*.

PA chooses an attacking action			
		Global IDS	
		Exclude	Admit
Local IDS	Suggestion for suspicious	0, 24, 24	9, 18, 0

Table 6 The 1st page of the matrix-attacking action strategy of PA, altered by removing strategy *Admit*.

PA chooses an attacking action		
		Global IDS
		Exclude
Local IDS	Suggestion for suspicious	0, 24, 24

Reversing the above reasoning, we consider that player *LIDS* chooses *suggestion for suspicious*. Then, player *GIDS* will choose strategy *Exclude*, which dominates *Admit*, because 24 > 0. Similarly, given that player *LIDS* chooses *suggestion for trusting*, player *GIDS* will choose again strategy *Exclude*, which dominates *Admit* too, because 12 > 6.

Thus how we reduce the payoff matrix again by removing strategy *Admit* which is dominated by strategy *Exclude*. The payoff matrix now has the following form (Table 6):

Consequently, the above deletions lead to the conclusion that player *GIDS* will prefer to choose strategy *Exclude* regardless what player *LIDS* chooses. Another conclusion is that both players, the *LIDS* and the *GIDS*, have dominant strategies, which is indispensable precondition for three player games to have equilibrium. Therefore, in this subgame there is a unique equilibrium, the (*attacking action*, *suggestion for suspicious*, *Exclude*) = (0, 24, 24).

Finally, we examine the above equilibrium, if it is a Nash equilibrium or not. A Nash equilibrium must hold that no players have interest to leave the equilibrium and select another strategy. In this three player game, we check whether players *LIDS* and *GIDS* would choose other strategies than those located at the equilibrium. If player *LIDS* chooses strategy *suggestion for suspicious*, then player *GIDS* will choose strategy *Exclude* as the most beneficial (24 > 0). Conversely, if player *GIDS* chooses strategy *Exclude*, then player *LIDS* will choose strategy *suggestion for suspicious* as the most beneficial (24 > 9). Obviously, no one between players *LIDS* and *GIDS* has any interest to leave the equilibrium (*attacking action*, *suggestion for suspicious*, *Exclude*). Therefore, this equilibrium is a Nash equilibrium.

Then, we check if the domination criterion expressed in Eq. (2) holds, when considering that player *PA* chooses a *normal action* (Table 4). Following the same reasoning, we conclude that player *GIDS* will prefer to choose strategy *Admit* regardless what player *LIDS* chooses. Another conclusion is that both players again, the *LIDS* and the *GIDS*, have dominant strategies, which is indispensable precondition for three player games to have

equilibrium. As a result, in this subgame there is a unique equilibrium where all players receive 21 each. This equilibrium is very beneficial for player *PA*, because he gets his highest payoff of the game (payoff 21). The equilibrium is the (*normal action*, *suggestion for trusting*, *Admit*) = (21, 21, 21).

We now examine the last equilibrium, if it is a Nash equilibrium or not, in a similar way as we did for the first page of the matrix of the game. We check whether players *LIDS* and *GIDS* would choose other strategies than those located at the equilibrium. If player *LIDS* chooses strategy *suggestion for trusting*, then player *GIDS* will choose strategy *Admit* as the most beneficial (21 > 18). Conversely, if player *GIDS* chooses strategy *Admit*, then player *LIDS* will choose strategy *suggestion for trusting* as the most beneficial (21 > 12). Obviously, no one between players *LIDS* and *GIDS* has any interest to leave the equilibrium (*normal action*, *suggestion for trusting*, *Admit*). Therefore, this equilibrium is also a Nash equilibrium.

Summarizing, the subgames equilibria located in the GoWiSeN game are two Nash equilibria; the (*attacking action*, *suggestion for suspicious*, *Exclude*) = (0, 24, 24) and the (*normal action*, *suggestion for trusting*, *Admit*) = (21, 21, 21). Decoding these findings we conclude that both equilibria are absolutely desirable for our model architecture described in Section The Architecture of the Model. In a case of the first equilibrium, although player *PA* attacks the network, both IDSs, the Local and the Global detect the attack and react properly. In a case of the second equilibrium, player *PA* behaves as a normal node of the network, the LIDS trusts its activity, and the GIDS permits this normal activity to be continued.

Solving with gambit

Computing a Nash equilibrium is a fundamental problem in Algorithmic Game Theory (*Nisan et al., 2007*). Following Daskalakis and Papadimitriou proofs in *Daskalakis & Papadimitriou (2005)*, computing Nash equilibria in a three player game is a problem that belongs in the PPAD-complete class, a class defined to address the problem of finding a NE in polynomial time. The complexity of computing a NE was consequently addressed in *Daskalakis, Goldberg & Papadimitriou (2009)* and *Roughgarden (2010)*. The Gambit tool provides algorithms to compute NE in non-cooperative finite games (*The Gambit Project, 2019*).

In Removing Dominated Strategies, we located two Nash equilibria in the pure strategies of the GoWiSeN game. In order to locate also equilibria in mixed strategies, we use the Gambit tool (*McKelvey, McLennan & Turocy, 2007*). The Gambit tool runs different algorithms. The GoWiSeN is a three player extensive form game. The 'Compute equilibria of a game using polynomial systems of equations' algorithm was selected as the most suitable, following Gambit' s documentation (*The Gambit Project, 2019*). The computational complexity of the PPAD class is thoroughly discussed in *Nisan et al. (2007)*.

Solving the GoWiSeN game in extensive form with the Gambit tool we get one Nash equilibrium. This unique Nash equilibrium in mixed strategies indicates the existence also

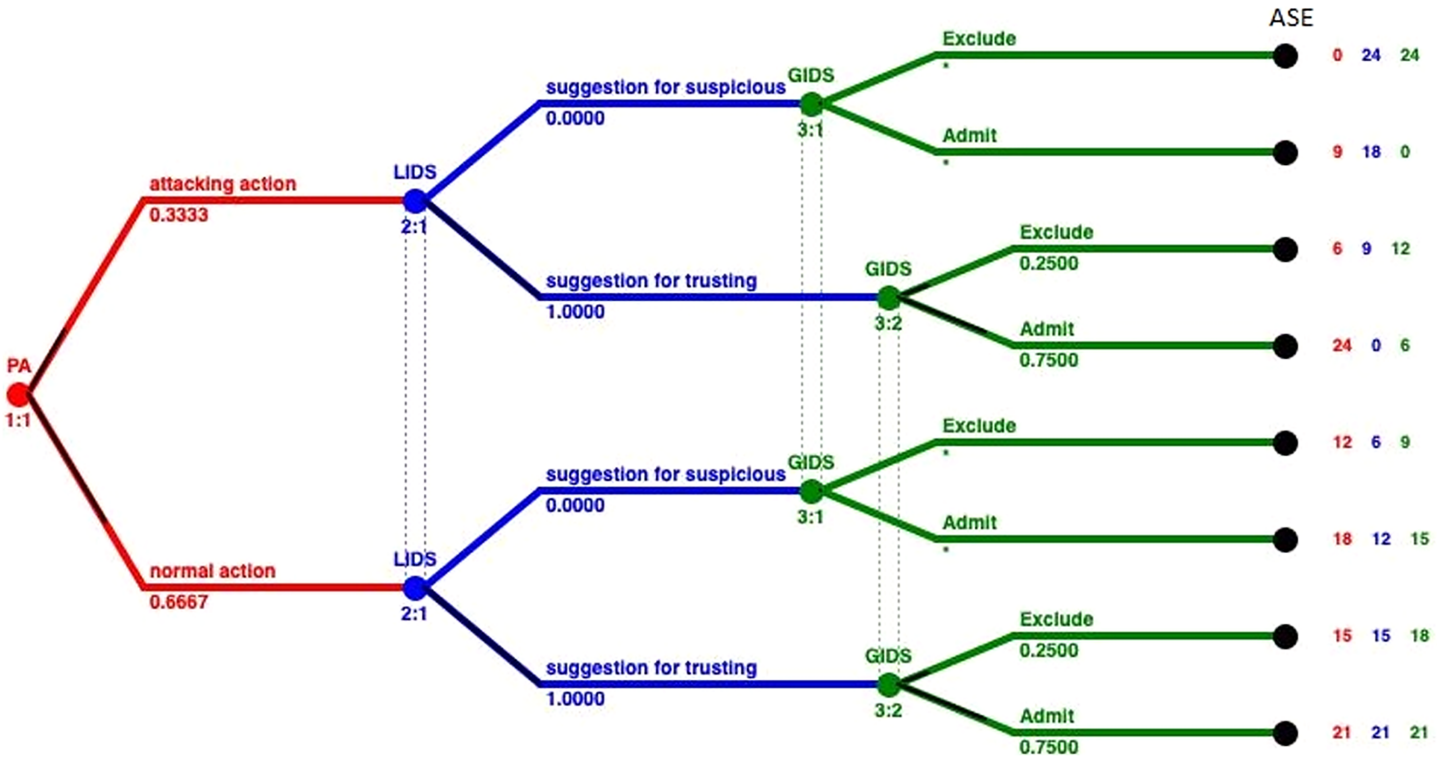

Figure 3 **Mixed strategies of the GoWiSeN game located by the Gambit tool.** Full-size DOI: 10.7717/peerj-cs.791/fig-3

of a unique equilibrium in behavioral strategies. In Fig. 3 the solution of the GoWiSeN game reveals the mixed strategies.

Numbers below each branch (zeroes, ones, or others) indicate the solution of the game. Zero means that an action will not be chosen (probability 0), whereas one means that an action will be selected with certainty (probability 1). Likewise, 0.25 for instance is the probability the related action to be chosen. Assuming that the GoWiSeN game starts with player *PA* to have 1/3 probability to be an attacker and 2/3 probability to be a normal user, the Gambit tool shows that regardless an *attacking action* or a *normal action* has been chosen, player *LIDS* will definitely choose *suggestion for trusting* with probability 1. This is because the probability player *PA* to be a normal user is much higher than to be an attacker.

As for player *GIDS*, it excludes with probability 1 an *attacking action*, for which player *LIDS* gives with zero probability a *suggestion for suspicious*. This happens because payoffs have been calculated upon players' preferences and thus they reflect their tendencies and first choices, which might totally conflict other players' beliefs. In the same information set that matches the case of a normal user (3:1), player *GIDS* chooses *Exclude* with probability 1 when player *LIDS* has suggested suspicious activity. Next, in the second information set that matches the case of an attacker (3:2), player *GIDS* chooses *Exclude* with probability 0.25 or *Admit* with probability 0.75 regardless whether player *LIDS* has chosen *suggestion for suspicious* or *suggestion for trusting*.

A CASE STUDY

We have chosen a case study to confirm the functionality of the proposed game applied in fire fighting. WSNs can significantly assist the work of fire extinguishing, when they are working securely. Fire fighting is one of the most dangerous jobs, often with human victims. The risks associated with it are connected with several factors, as for example, the incomplete information about the exact location and the extent of a fire. The use of WSNs might reduce the number of risks associated with the firemen, and assist the quick and effective fire extinguishing. Finally, they might give additional information to the experts who investigate the cause of a fire, especially in cases where the fire has been caused intentionally.

Wireless sensor networks for fire fighting

WSNs can be distinguished between data collection and event detection networks (*Dutta, 2004*). In those applications where the aim is the data collection, the sensors might be necessary to collect information every short periods at predefined time intervals of the day. As a consequence, for the rest of the time, the sensor node remains idle, so it saves power. However, in those cases where a WSN would be used for event detection, as it is the fire detection case, the sensor nodes should be alert, consuming continuously their power.

WSNs applied in fire fighting have special requirements that can be summarized as in the following (*Sha, Shi & Watkins, 2006*):

- False alarms must be kept to a minimum, because they consume time and resources of the fire brigade and thus might lead to unavailability of services in real instances.
- The WSN should be secure so that malicious activities must be deterred, because they might cause false alarms and send false information.
- As a fire might spread out quickly, the initial node that detects the event must send the data as soon as possible. If it fails to do so, then some alerts might not be set and valuable information might be lost. Moreover, this initial node must awake other adjacent nodes before destruction.
- The fire brigade must be connected with the WSN in order to exchange information.
- The network must be able to reroute its packages in cases of partial destruction, *i.e.*, when specific nodes are destroyed, to ensure the uninterruptible operation of the network, Therefore, a feature that allows the automatic adjustment of the routing table is required.
- The data transfer rate should be significantly high in order to keep information valuable and accurate.
- The fire brigade should know the exact positions of the sensor nodes.
- There must be a visualized demonstration of the location and spread of the fire as well as of the temperatures inside the building.
- The sensor nodes must be properly protected against high temperatures, to ensure their functionality and their ability to work accurately.

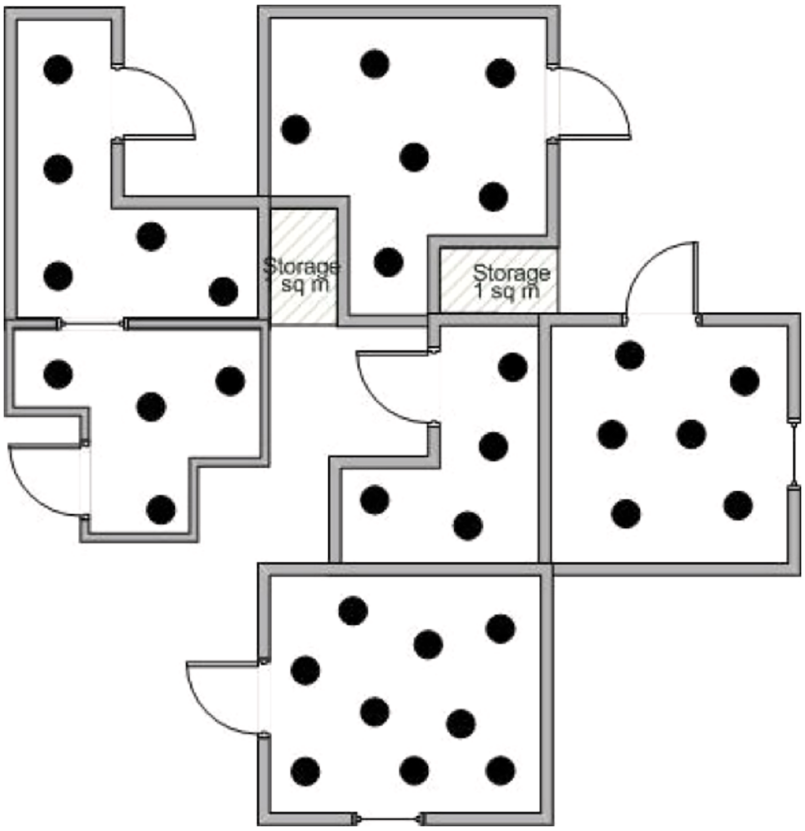

Figure 4 **A division of the area covered by the WSN.** Full-size DOI: 10.7717/peerj-cs.791/fig-4

Among the described requirements, the proposed game mainly aims at fulfilling the first and the second one, by detecting intrusive activities and so reducing the number of false alarms raising by malicious nodes.

The installation of a wireless sensor network in a building

Consider a WSN installed throughout the rooms of a company's headquarters in a flat of a skyscraper in NY. The whole area extents to 2,500 square meters, and numerous small rooms are shaped as offices, using special light separators. Figure 4 presents only a division of the area occupied by the company. There are small rooms, nodes are scattered densely in each room, and a base station is cited somewhere in a safe place of the network's deployment region. Different types of sensor nodes have been used: temperature sensors for detection and tracking, smoke detectors for detection, infrared detectors for tracking and smoke and movement detectors too.

The WSN is always connected with the fire brigade *via* the internet, so that, it could send an emergency signal in case of fire detection. As a result, when fire is detected, the director of the fire brigade will be able, *via* this connection, to send to firemen information related to the initial fire location and to the progress and direction of the fire. Moreover, when the firemen arrive at the fire location, they could connect to the WSN to receive the latest information about the fire spread. Detailed sketches that depict the

building infrastructure and the node locations of the WSN will be helpful to the fire brigade.

Upon arrival, the firemen could use a portable computer in the place of the base station computer. Thus, as long as the nodes of the network are properly working, even if the base station has been destructed, the WSN will continue its functioning without problems.

Implementing the GoWiSeN model

The safety of a WSN applied for fire fighting might be threatened by malicious unauthorized persons who try to gain access to the network. Next, such an intruder might intentionally cause a false alarm, or even alter information collected from the sensors of the network, affecting the validity and integrity of the information being transmitted to the base station.

The GoWiSeN model has been designed to protect a WSN of similar types of attacks. For this purpose, a Local Intrusion Detection System will be installed over each sensor node, named Local IDS. This LIDS is a light version of an IDS, with minimum power requirements. Furthermore, the base station will host a complete IDS package, named Global IDS (GIDS).

The GIDS keeps history records for all the events that have been identified as illegal activities of the network nodes. Consequently, a list is maintained for each node, rating from 0 to 5 the number of times the corresponding node has acted illegally. In particular, the value of 0 indicates a node that operates normally and has never been detected for malicious activities, whereas, the value of five denotes that the node has already been detected 5 times for attacking actions.

In addition, when a node communicates with other adjacent nodes, because it has a LIDS installed, it could examine their activities and preliminary conclude whether a node acts legitimately or not. Subsequently, it would send a signal of *suggestion for suspicious* or a signal of *suggestion for trusting* to the GIDS, depending on its conclusion. Finally, the GIDS on its turn, will decide whether it should *admit* or *exclude* the node's activity.

A fire event is detected when some sensor nodes of the network detect temperatures higher than 40 °C for a longer period than 10 s. In case of fire, the LIDS will send a signal of *suggestion for trusting* to the GIDS, if the adjacent nodes report the same information. Under certain circumstances with high temperatures due to physical reasons, such as the phenomenon of heatwave during summer, the network manager should take care of the limits adjustment.

The GIDS should then examine three different parameters, in order to decide whether to admit or to exclude an activity suggested by a node. First, it should consider the LIDS's suggestion either for trusting or for suspicious. Next, it should take into account the list that corresponds to the node reported the problem. As a final point, it might use the detection engine integrated in it, which uses classical detection techniques, to verify the initial findings and deduce a definite conclusion.

Two different scenarios demonstrate the network operation and functioning in the following subsections. Scenario A shows in detail the steps for fighting a real fire event,

while scenario B presents the proposed model functioning, when an attacking activity, originated by a compromised node, takes place.

Scenario A

1. Suppose node N_{21} is a temperature detection sensor, belongs to cluster C_2 and operates normally.
2. A fire event takes place somewhere inside the area occupied by the company's headquarters. The fire point is next to node N_{21}.
3. Node N_{21} detects temperature higher than 40 °C for a longer period than 10 s, and therefore, an alarm is automatically triggered.
4. The fire event detection is reported to the adjacent nodes.
5. The adjacent nodes also detect high temperatures according to the predefined thresholds, evaluate the received information, and they trigger alarms as well.
6. Node N_{21} sends a signal of fire event detection to the base station.
7. The adjacent nodes' LIDSs send signals of *suggestion for trusting* to the base station.
8. The GIDS puts under consideration all the received information, examines the corresponding list with the node ranking, takes into account the adjacent nodes suggestions, and concludes to *admit* the reported event as real.
9. The base station sends all the collected data to the center of the fire brigade, the fire brigade estimates the current situation and counteracts against the fire.
10. When the operation has been completed, the fire brigade sends information to the GIDS regarding the accuracy of the initial information reported the problem.

Scenario B

1. Suppose node N_{21} is a temperature detection sensor, belongs to cluster C_2 and operates normally.
2. A skilled intruder gains access to the WSN, by compromising node N_{21}. His intentions include continuous attacks every short time periods. His aim is to set out of order a considerable part of the network, each time he completes an attack.
3. The compromised node N_{21} initiates an attacking activity. The attack is based on the flooding attack technique and will be carried out at the transportation layer of the network. The aim of the flooding attack is to exhaust the resources of the adjacent nodes, so that, a part of the network will be set out of use.
4. Each LIDS, that resides on an adjacent node, examines the data captured from this activity. Based on this data, the LIDS detects illegal activity, because the number of connection requests, from the compromised node N_{21}, is unusually great. Therefore, all the LIDSs of the adjacent nodes will send signals of *suggestion for suspicious* to the base station, considering this activity as an attacking activity.
5. The GIDS puts under consideration all the received information, examines the corresponding list with the node ranking, takes into account the adjacent nodes

suggestions, and activates the anomaly detection technique module to handle the unexpected operation. The module detects that the number of requests for connections is over the upper threshold, and the GIDS concludes to *exclude* the requested connections as malicious activities, preventing the network flooding.

6. The GIDS updates the list assigned to the compromised node, by increasing its ranking by 1. Then it checks if the number of times the node has been acted illegally is already 5, and if this is true, the GIDS adds the compromised node N_{21} into a blacklist, for all the routing tables of the network.

Applying the GoWiSeN model

Applying the GoWiSeN model should reveal the way the game described and solved in sections The Game and Solution of the Game would be played, in order to protect the WSN established for fire fighting.

In Scenario A, player *PA* is a normal user of the system, the user of node N_{21}, which works normally. A fire event that takes place very close to this node, triggers an alarm by this node, which also sends information to the adjacent nodes. In GoWiSeN, player *PA* has chosen *normal action*. Afterwards, the adjacent nodes detect the event and evaluate the information received from node N_{21}. Since they conclude that the information is valid, they recommend suggestion for trusting to the base station. This means that the main LIDS as well the adjacent LIDS will play *suggestion for trusting*, also because this strategy has the highest payoff regardless what the GIDS will choose.

The base station evaluates all the information collected by the related LIDS and estimates what the real situation is. Considering that the ranking of node N_{21} is low (its value is 0) and taking into account what the adjacent nodes recommend, player *GIDS* chooses *Admit*. This strategy also gives the best payoff to player *GIDS*, no matter what player *LIDS* plays. It is the Nash equilibrium located in Section Removing Dominated Strategies, (*normal action*, *suggestion for trusting*, *Admit*) = (21, 21, 21).

In Scenario B, player *PA* is an internal attacker of the system, the user of node N_{21}, which is compromised. We assume that node N_{21} might work normally most of the times, but sometimes it is used by the insider to attack periodically the system. The insider aims at causing unavailability over a significant part of the network each time he accomplishes an attack. In such a case, the compromised node attacks the system. This means that player *PA* chooses *attacking action*. Any adjacent node and its *LIDS* that accepts the attack evaluates this move. The LIDS detects unusual behavior, because the compromised node suddenly starts sending a large number of connection requests. Player *LIDS* will choose *suggestion for suspicious* and will send the relevant information to the base station. Nevertheless, this strategy has the highest payoff.

In the base station, the GIDS examines the event and chooses *Exclude* preventing its completion. This decision is based also upon an anomaly detection technique which detects the unusual behavior of the aforementioned node, the history and the list with the rankings and updates the corresponding ranking for the compromised node N_{21} by adding 1. If the node ranking will be 5, then it becomes blacklisted for all the routing tables of

Table 7 Attack types in the Bot-IoT dataset.

Category	Attack type	Flow count	Training	Test
BENIGN	BENIGN	9,543	7,634	1,909
Information gathering	Service scanning	1,463,364	117,069	29,267
	OS Fingerprinting	358,275	28,662	7,166
DDoS attack	DDoS TCP	19,547,603	1,563,808	390,952
	DDoS UDP	18,965,106	1,517,208	379,302
	DDoS HTTP	19,771	1,582	395
DoS attack	DoS TCP	12,315,997	985,280	246,320
	DoS UDP	20,659,491	1,652,759	413,190
	DoS HTTP	29,706	2,376	594
Information theft	Keylogging	1,469	1,175	294
	Data theft	118	94	24
Total	/	73,370,443	5,877,647	1,469,413

Table 8 Simulation parameters.

Number of Players	3
Number of Nodes [Bot-IoT dataset]	8
Mean Simulation Time [Training & Testing]	1,220 [KMeans], 760 [Naive Bayes]
Actions	Alarm-Exclude\Admit

the network. In this case, player *GIDS* will play *Exclude* regardless what player *LIDS* chooses. It is the Nash equilibrium located in Section Removing Dominated Strategies, (*attacking action*, *suggestion for suspicious*, *Exclude*) = (0, 24, 24).

EXPERIMENTAL EVALUATIONS

After completing the theoretical analysis as presented in the previous sections, we decided to conduct simulations on realistic datasets using IDSs with different capabilities.

Dataset

We use a recent dataset that was collected from a realistic network environment, namely the Bot-IoT dataset (*Koroniotis & Moustafa, 2021*) for the experiments. Table 7 summarizes the statistics of attacks in Training and Testing datasets. Both datasets satisfies the eleven indispensable characteristics of a valid IDS dataset, namely Anonymity, Attack Diversity, Complete Capture, Complete Interaction, Complete Network Configuration, Available Protocols, Complete Traffic, Feature Set, Metadata, Heterogeneity, and Labelling (*Gharib et al., 2016*). Table 8 gives a synopsis of the main simulation parameters that affect the experimental evaluations.

The BoT-IoT dataset contains more than 72,000,000 records devised on 74 files, each row having 46 features. We use the version proposed by *Koroniotis et al. (2019)*, which is a version of training and testing with 5% of the entire dataset.

Table 9 Confusion matrix.

		Predicted class	
		Negative class	Positive
Actual class	Negative Class	True negative (TN)	False positive (FP)
	Positive Class	False negative (FN)	True positive (TP)

Table 10 Evaluation of different IDSs.

Method	True positive (TP)	False negative (FN)	True negative (TN)	False positive (FP)
Naive Bayes	0.999972	0.000028	0.975	0.025
Kmeans	0.55	0.45	0.916	0.084

Clustering and classification techniques

Assuming that the architecture described in Fig. 1 consists of IDSs of different capabilities due to power, storage and computational constraints, we decided to evaluate the performance of different IDSs on Bot-IoT dataset.

IDS performance is evaluated based on its capability of classifying network traffic into a correct type. Table 9, also known as confusion matrix, shows all the possible cases of classification.

We decided to use an IDS with poor performance to play the role of the LIDS and a classifier with very good performance as the GIDS. This is due to the fact that each simple node can play the role of LIDS and we cannot expect it to have high computation or energy capabilities and this we need to have a simple method running on it. On the same time GIDS is a specific node that can have additional capabilities. For LIDS we have used Kmeans clustering and for the GIDS the Naive Bayes method. The overall performance of both methods is presented in Table 10.

The aforementioned values correspond to the different arcs in the tree that is presented in Fig. 3. The pairing of each value of the IDS and the arcs on the tree is represented in Fig. 5. Based on this pairing, we changed the payoff of each player for every strategy, by assigning the payoff proportion, which corresponds to the probability a strategy profile being selected, as shown in Table 11. These probabilities derive from the detection rates provided by the dataset, and complies with the game theoretic solution approach followed to locate NE in mixed strategies (*Bhuse, Gupta & Al-Fuqaha, 2007*). Based on these calculations, the payoffs that incorporate the performance of the IDSs are inserted into the model, as shown in Fig. 5.

We solve again the model with the Gambit tool (*McKelvey, McLennan & Turocy, 2007*) and we get one Nash equilibrium. This unique Nash equilibrium in mixed strategies indicates the existence also of a unique equilibrium in behavioral strategies. In Fig. 6 the solution of the GoWiSeN game reveals the mixed strategies.

Numbers below each branch (Zeros, ones, or others) indicate the solution of the game. Zero means that an action will not be chosen (probability 0), whereas one means that an action will be selected with certainty (probability 1). Likewise, 0.20 for instance is the

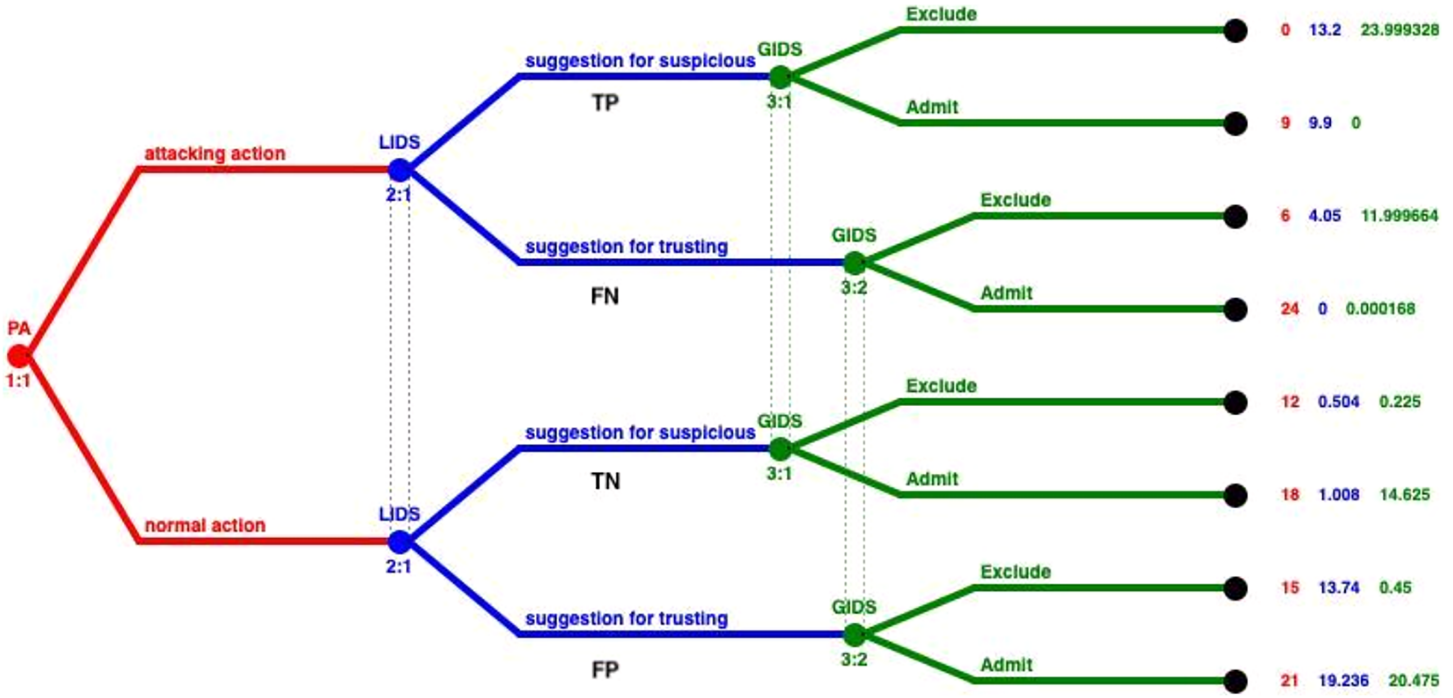

Figure 5 **Integrating performance of IDSs in mixed strategies.** Full-size DOI: 10.7717/peerj-cs.791/fig-5

Table 11 **Recalculated rewards for each player.**

State	LIDS performance	GIDS performance	LIDS initial reward	GIDS initial reward	LIDS final reward	GIDS final reward
TP–TP	0.55	0.999972	24	24	13.2 5	23.999328
TP–FN	0.55	0.000028	18	0	9.9	0
FN–TP	0.45	0.999972	9	12	4.05	11.999664
FN–FN	0.45	0.000028	0	6	0	0.000168
FP–FP	0.084	0.025	6	9	0.504	0.225
FP–TN	0.084	0.975	12	15	1.008	14.625
TN–FP	0.916	0.025	15	18	13.74	0.45
TN–TN	0.916	0.975	21	21	19.236	20.475

probability the related action to be chosen. By incorporating the evaluation metrics of the IDSs, we see that the probabilities in each branch has changed, representing a more realistic scenario, without deterministic decisions that dominated the previous model. We can observe that GIDS takes a on/off decision following the decision of the LIDS. LIDS on the other hand has probability 0.2 to fire an alarm and 0.8 to accept the bahavior of the node as normal, regardless of the action of the PA.

SPECIAL CASES

Except from the general case where one PA is performing in a malicious way and the LIDS is detecting this, there are some special cases that need to be modeled in a different way. These two special cases are presented in the following subsections.

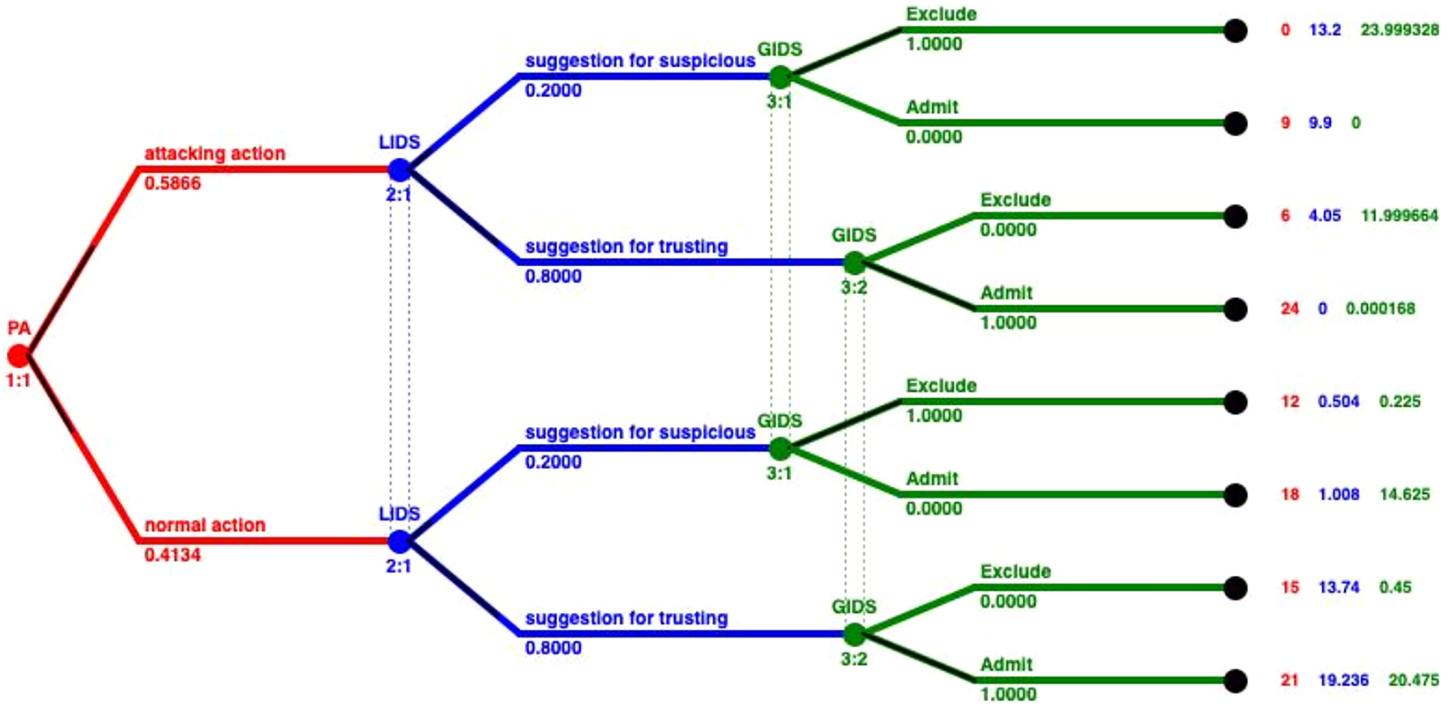

Figure 6 **Mixed strategies with evaluation metrics of IDSs.** Full-size DOI: 10.7717/peerj-cs.791/fig-6

Overlapping clusters

In WSNs clustering is an efficient approach used to achieve optimal performance of the network. In traditional clustering methods disjoint clusters are created, using some specific criteria that include distance, energy, delay etc. On the other hand, many scholars have highlighted the significant advantages of creating overlapping clusters when applying intercluster routing, node localization, and time synchronization protocols (*Youssef, Youssef & Younis, 2009*; *Maglaras & Katsaros, 2011*; *Abbasi & Younis, 2007*).

In this case, several PAs belong to two clusters and thus are monitored by two LIDSs as represented in Fig. 7. In order to model this situation we need to construct different games. In Fig. 8, we can see the representation of a game that consists of the PA, that belongs to clusters A and B, and LIDSa of cluster A (left part of the figure). On the right part of the figure, in case these two IDSs perform the same, thus having the same TP, TN, FP, TN, these two short trees can be merged creating the same outcome as the normal case. In the occasion that these IDSs have different performance due to running different detection algorithms, or due to other parameters that affect the observation capabilities (distance, interference or even trust levels) these two trees cannot be merged, and the outcomes of the two LIDSs must be combined using smart techniques (*Maglaras, Jiang & Cruz, 2016*).

LIDS under attack

During the analysis and evaluation of the system we have assumed that the LIDSs are immune to cyber attacks. This assumption is not very realistic since any node can play the

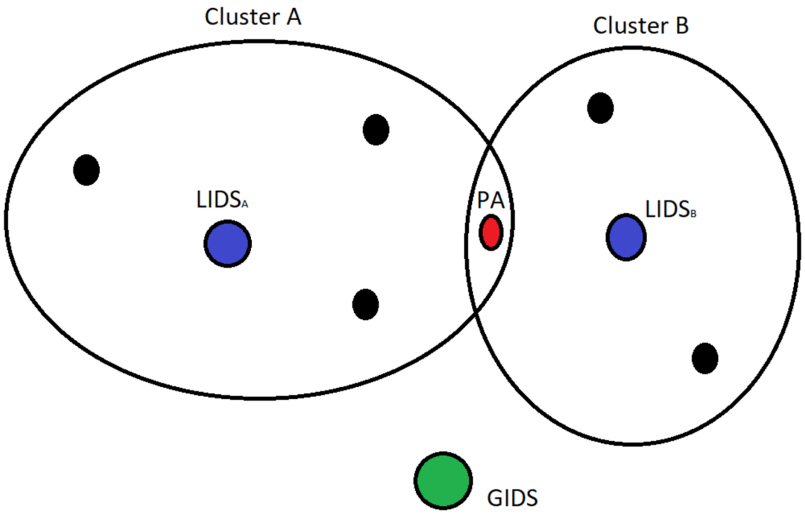

Figure 7 Special case of a PA that belongs to two clusters.
Full-size DOI: 10.7717/peerj-cs.791/fig-7

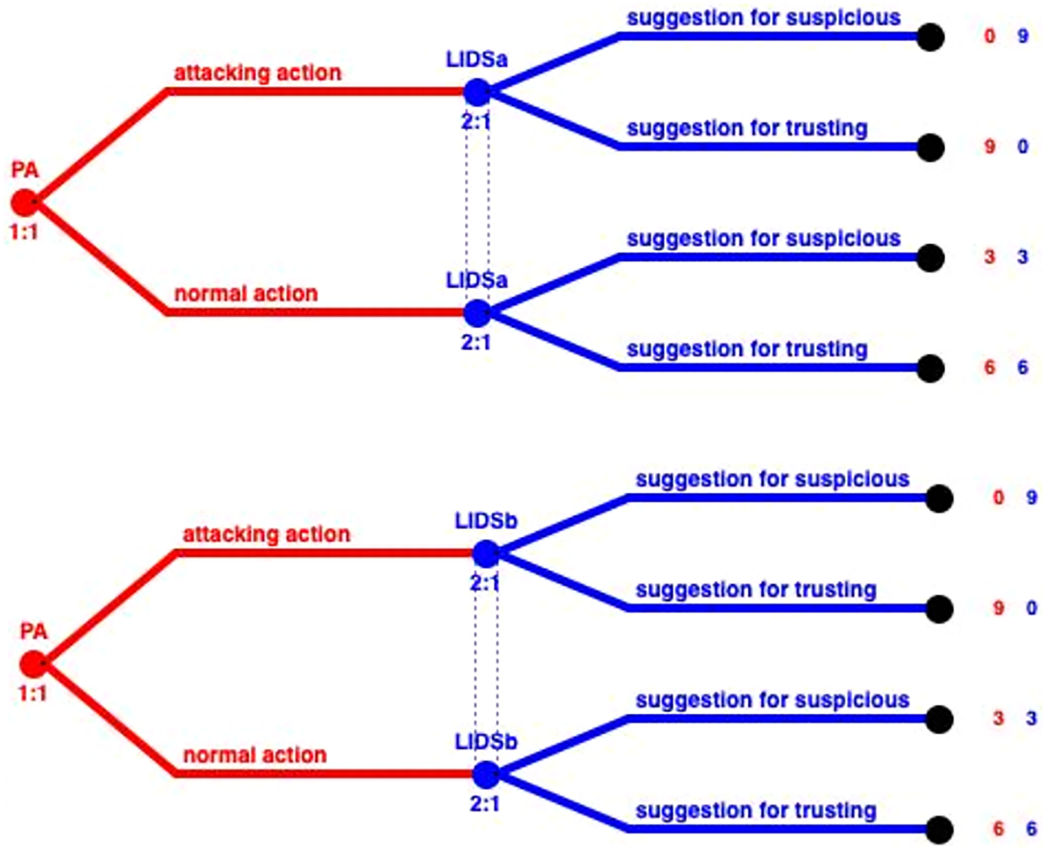

Figure 8 Subtrees of LIDSa and LIDSb.
Full-size DOI: 10.7717/peerj-cs.791/fig-8

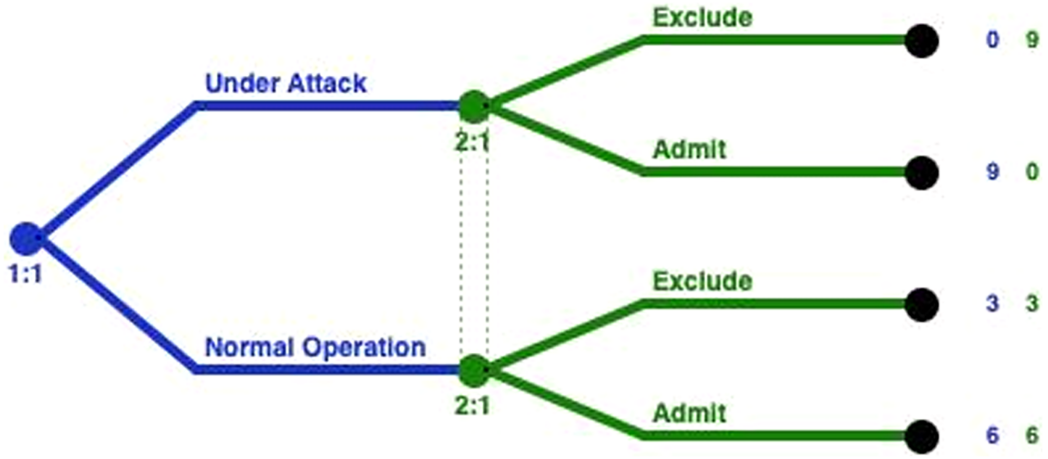

Figure 9 **LIDS under attack model.** Full-size DOI: 10.7717/peerj-cs.791/fig-9

role of a LIDS. In the specific case where a LIDS is infected, the GIDS can detect the attack and report back to the system, in order to ignore its alarms and also appoint another node as LIDS for the cluster. The reappointment of the LIDS can be done similar to a reclustering procedure, where nodes, through voting, elect the node that will play this role for their cluster (*Behera et al., 2019*).

For this special case, the model that can be constructed consists of only two players, the LIDS (that plays the role of the PA) and the GIDS (that plays the role of the LIDS). Such a model is represented in Fig. 9 and can be solved in a similar way as the general case, but with fewer players and less states.

CONCLUSION AND FUTURE WORK

The incorporation of a game theoretic approach in the area of Intrusion Detection to confront internal attackers in WSNs was selected. To address the problem of detecting compromised nodes, a three player non-cooperative game was modeled, between Local IDSs located on different sensor nodes (LIDSs), a Global IDS (GIDS) and an insider. Possible interactions between an insider and Intrusion Detection Systems used in a WSN were examined, and possible suggesting strategies were studied through the solution of the game, by locating Nash Equilibria in mixed strategies. The evaluation of the model using a realistic dataset from a network environment was decided, to challenge its ability to discriminate between normal and malicious nodes. In special cases, nodes that belong to overlapping clusters were considered and the corresponding model was studied. Finally, the situation where a LIDS is compromised was examined and different solution approaches to this problem too were discussed. The results show how the game should be played, what the players choose to play and under which circumstances. An ultimate goal is the identification of a node as compromised and its exclusion of the network to prevent further damage.

The integration of the evaluation metrics of LIDSs and GIDS through the use of a realistic dataset further improves the efficiency of the proposed model. The use of different datasets, combination of more machine learning techniques and the use of smart ensemble

methods can be considered as future research paths. Regarding a future extension of the theoretical game model, subgame perfect equilibria will be considered to improve collaboration between LIDSs and GIDS as well as different preference rankings that correspond to miscellaneous types of insiders.

ACRONYMS

CPES	Cyber-Physical Embedded System
DDoS	Distributed Denial of Service
ESN	Embedded Sensor Network
FN	False Negative
FP	False Positive
GoWiSeN	Game of Wireless Sensor Networks
GIDS	Global Intrusion Detection System
IDS	Intrusion Detection System
IoT	Internet of Things
LIDS	Local Intrusion Detection System
NE	Nash Equilibrium
NY	New York
OS	Operating System
PA	Potential Attacker
TN	True Negative
TP	True Positive
VCG	Vickery-Clark-Grooves
WSN	Wireless Sensor Networks

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

The authors received no funding for this work.

Competing Interests

Leandros Maglaras is an Academic Editor for PeerJ Computer Science.

Author Contributions

- Ioanna Kantzavelou conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Leandros Maglaras conceived and designed the experiments, performed the experiments, performed the computation work, prepared figures and/or tables, and approved the final draft.
- Panagiotis F Tzikopoulos analyzed the data, prepared figures and/or tables, and approved the final draft.

- Sokratis Katsikas conceived and designed the experiments, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The data is available at figshare: Maglaras, Leandros (2021): Datasets. figshare. Dataset. https://doi.org/10.6084/m9.figshare.16627693.v1.

REFERENCES

Abbasi AA, Younis M. 2007. A survey on clustering algorithms for wireless sensor networks. *Computer Communications* **30(14–15)**:2826–2841 DOI 10.1016/j.comcom.2007.05.024.

Abdalzaher M, Seddik K, Elsabrouty M, Muta O, Furukawa H, Abdel-Rahman A. 2016. Game theory meets wireless sensor networks security requirements and threats mitigation: a survey. *Sensors* **16(7)**:1003 DOI 10.3390/s16071003.

Agarwal A, Sharma P, Alshehri M, Mohamed AA, Alfarraj O. 2021. Classification model for accuracy and intrusion detection using machine learning approach. *PeerJ Computer Science* **7(3)**: e437 DOI 10.7717/peerj-cs.437.

Albers P, Camp O, Percher JM, Jouga B, Mé L, Puttini R. 2002. Security in ad hoc networks: a general intrusion detection architecture enhancing trust based approaches. In: *1st International Workshop on Wireless Information System (WIS 2002) Proceedings*.

Alcaraz C. 2019. *Security and privacy trends in the industrial internet of things*. Berlin: Springer.

Alhakami W, ALharbi R, Alroobaea A, Bourouis S, Bouguila N. 2019. Network anomaly intrusion detection using a nonparametric bayesian approach and feature selection. *IEEE Access* **7**:52181–52190 DOI 10.1109/ACCESS.2019.2912115.

Bai F, Liu XY, Zhang YL, Lang DP. 2019. Research on game model of wireless sensor network intrusion detection. In: *Proceedings of the 2019 International Conference on Embedded Wireless Systems and Networks, EWSN '19*. USA: Junction Publishing, 373–378.

Behera TM, Mohapatra SK, Samal UC, Khan MS, Daneshmand M, Gandomi AH. 2019. Residual energy-based cluster-head selection in wsns for IoT application. *IEEE Internet of Things Journal* **6(3)**:5132–5139 DOI 10.1109/JIOT.2019.2897119.

Bhuse V, Gupta A. 2006. Anomaly intrusion detection in wireless sensor networks. *Journal of High Speed Networks* **15**:33–51.

Bhuse V, Gupta A, Al-Fuqaha A. 2007. Detection of masquerade attacks on wireless sensor networks. In: *IEEE International Conference on Communications 2007 (ICC '07) Proceedings*. 1142–1147.

Binmore K. 2007. *Playing for real-a text on game theory*. Oxford: Oxford University Press.

Bogdanov A, Maneva E, Riesenfeld S. 2004. Power-aware base station positioning for sensor networks. In: *23rd Conference of the IEEE Communications Society (IEEE INFOCOM '04)*. Piscataway: IEEE, 575–585.

Camerer CF. 2003. *Behavioral game theory*. Princeton: Princeton University Press.

Daskalakis C. 2008. The complexity of nash equilibria. PhD thesis, University of California, Berkeley.

Daskalakis C, Goldberg P, Papadimitriou C. 2005. The complexity of computing a nash equilibrium. *Electronic Colloquium on Computational Complexity* TR05–TR115. *Available at https://people.csail.mit.edu/costis/simplified.pdf*.

Daskalakis C, Goldberg P, Papadimitriou C. 2009. The complexity of computing a nash equilibrium. *Communications of the ACM* **52(2)**:89–97 DOI 10.1145/1461928.1461951.

Daskalakis C, Papadimitriou C. 2005. Three-player games are hard. *Electronic Colloquium on Computational Complexity* TR05–TR139.

Dixit A, Skeath S. 1999. *Games of strategy*. New York: W. W. Norton & Company, Inc.

Dua S, Du X. 2016. *Data mining and machine learning in cybersecurity*. Boca Raton: CRC Press.

Dutta KP. 2004. On random event detection with wireless sensor networks. M.Sc. Thesis, Columbus, OH, USA.

Elhag S, Fernández A, Bawakid A, Alshomrani S, Herrera F. 2015. On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems. *Expert Systems with Applications* **42(1)**:193–202 DOI 10.1016/j.eswa.2014.08.002.

Gay D, Levis P, Culler D. 2007. Software design patterns for tinyos. *ACM Transactions on Embedded Computing Systems* **6(4)**:1–39 DOI 10.1145/1274858.1274860.

Gharib A, Sharafaldin I, Lashkari AH, Ghorbani AA. 2016. An evaluation framework for intrusion detection dataset. In: *2016 International Conference on Information Science and Security (ICISS), 2016 International Conference*. Piscataway: IEEE, 1–6.

Han S, Chang E, Gao L, Dillon T. 2005. Taxonomy of attacks on wireless sensor networks. In: *First European Conference on Computer Network Defence (EC2ND 2005)*. Andrew Blyth ed., University of Glamorgan, 97–105.

Han G, Jiang J, Shu L, Niu J, Chao H-C. 2014. Management and applications of trust in wireless sensor networks: a survey. *Journal of Computer and System Sciences* **80(3)**:602–617 Special Issue on Wireless Network Intrusion DOI 10.1016/j.jcss.2013.06.014.

Han L, Zhou M, Jia W, Dalil Z, Xu X. 2019. Intrusion detection model of wireless sensor networks based on game theory and an autoregressive model. *Information Sciences* **476(3)**:491–504 DOI 10.1016/j.ins.2018.06.017.

Hill J, Szewczyk R, Woo A, Hollar S, Culler DE, Pister KSJ. 2000. System architecture directions for networked sensors. In: *9th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. 93–104.

Holt CA, Roth AE. 2004. The nash equilibrium: a perspective. *Proceedings of the National Academy of Sciences (PNAS) of the United States of America* **101(12)**:3999–4002 DOI 10.1073/pnas.0308738101.

Jaint B, Singh V, Tanwar LK, Indu S, Pandey N. 2018. An efficient weighted trust method for malicious node detection in clustered wireless sensor networks. In: *2018 2nd IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*. Piscataway: IEEE, 1183–1187.

Kantzavelou I, Tzikopoulos PF, Katsikas SK. 2013. Detecting intrusive activities from insiders in a wireless sensor network using game theory. In: *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 'TM13*. New York, NY, USA: Association for Computing Machinery.

Karlof C, Wagner D. 2003. Secure routing in wireless sensor networks: attacks and countermeasures. *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols* **1(2–3)**:293–315 DOI 10.1016/S1570-8705(03)00008-8.

Kiennert C, Ismail Z, Debar H, Leneutre J. 2018. A survey on game-theoretic approaches for intrusion detection and response optimization. *ACM Computing Surveys* **51(5)**:1–31 DOI 10.1145/3232848.

Koroniotis N, Moustafa N. 2021. Bot-IoT dataset. *Available at https://research.unsw.edu.au/projects/bot-iot-dataset* (accessed 2 June 2021).

Koroniotis N, Moustafa N, Sitnikova E, Turnbull B. 2019. Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: bot-IoT dataset. *Future Generation Computer Systems* **100(7)**:779–796 DOI 10.1016/j.future.2019.05.041.

Li W, Meng W, Kwok L-F, IP HHS. 2017. Enhancing collaborative intrusion detection networks against insider attacks using supervised intrusion sensitivity-based trust management model. *Journal of Network and Computer Applications* **77(2)**:135–145 DOI 10.1016/j.jnca.2016.09.014.

Ma J, Zhang S, Zhong Y, Tong X. 2006. SAID: A self-adaptive intrusion detection system in wireless sensor networks. In: *The 7th International Workshop on Information Security Applications (WISA'06)*. 60–73.

Machado R, Tekinay S. 2008. A survey of game-theoretic approaches in wireless sensor networks. *Computer Networks* **52(16)**:3047–3061 DOI 10.1016/j.gaceta.2008.07.003.

Maglaras LA, Jiang J, Cruz TJ. 2016. Combining ensemble methods and social network metrics for improving accuracy of ocsvm on intrusion detection in scada systems. *Journal of Information Security and Applications* **30(10)**:15–26 DOI 10.1016/j.jisa.2016.04.002.

Maglaras LA, Katsaros D. 2011. Layered backpressure scheduling for delay reduction in ad hoc networks. In: *2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*. Piscataway: IEEE, 1–9.

Manzo M, Roosta T, Sastry S. 2006. Time synchronization attacks in sensor networks. In: *The 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '05)*. ACM, 107–116.

McKelvey RD, McLennan AM, Turocy TL. 2007. Gambit: software tools for game theory. *Available at http://www.gambit-project.org* (accessed 8 June 2021).

Meng Y, Li W, Kwok L. 2013. Evaluation of detecting malicious nodes using bayesian model in wireless intrusion detection. In: Lopez Javier, Huang Xinyi, Sandhu Ravi, eds. *Network and System Security*. Berlin, Heidelberg: Springer Berlin Heidelberg, 40–53.

Naseer S, Minhas Q, Saleem K, Siddiqui GF, Bhatti N, Mahmood H. 2021. A game theoretic power control and spectrum sharing approach using cost dominance in cognitive radio networks. *PeerJ Computer Science* **7(4)**:e617 DOI 10.7717/peerj-cs.617.

Nash JF. 1950. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences of the United States of America* **36(1)**:48–49 DOI 10.1073/pnas.36.1.48.

Nawir M, Amir A, Yaakob N, Lynn OB. 2016. Internet of things (IoT): taxonomy of security attacks. In: *2016 3rd International Conference on Electronic Design (ICED)*. IEEE, 321–326.

Nisan N, Roughgarden T, Tardos E, Vazirani V. 2007. *Algorithmic game theory*. Cambridge: Cambridge University Press.

Osborne MJ. 2004. *An introduction to game theory*. New York: Oxford University Press.

Osborne MJ, Rubinstein A. 1994. *A course in game theory*. Cambridge: The MIT Press.

Pirretti M, Zhu S, Narayanan V, McDaniel P, Kandemir M, Brooks R. 2005. The sleep deprivation attack in sensor networks: Analysis and methods of defense. In: *Innovations and Commercial Applications of Distributed Sensor Networks Symposium (ICA DSN 05)*.

Ranaweera M, Seneviratne A, Rey D, Saberi M, Dixit VV. 2019. Anomalous data detection in vehicular networks using traffic flow theory. In: *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*. Piscataway: IEEE, 1–5.

Roughgarden T. 2010. Computing equilibria: a computational complexity perspective. *Economic Theory* **42(1)**:193–236 DOI 10.1007/s00199-009-0448-y.

Sedjelmaci H, Senouci SM, Taleb T. 2017. An accurate security game for low-resource iot devices. *IEEE Transactions on Vehicular Technology* **66(10)**:9381–9393 DOI 10.1109/TVT.2017.2701551.

Sha K, Shi W, Watkins O. 2006. Using wireless sensor networks for fire rescue applications: requirements and challenges. In: *2006 IEEE International Conference on Electro/information Technology*. Piscataway: IEEE, 239–244.

Sharma R, Athavale VA. 2019. Survey of intrusion detection techniques and architectures in wireless sensor networks. *International Journal of Advanced Networking and Applications* **10(4)**:3925–3937 DOI 10.35444/IJANA.2019.10044.

Shi H-Y, Wang W-L, Kwok N-M, Chen S-Y. 2012. Game theory for wireless sensor networks: a survey. *Sensors* **12(7)**:9055–9097 DOI 10.3390/s120709055.

Singh R, Singh J, Singh R. 2017. Fuzzy based advanced hybrid intrusion detection system to detect malicious nodes in wireless sensor networks. *Wireless Communications and Mobile Computing* **2017(2)**:1–14 DOI 10.1155/2017/3548607.

Stankovic J. 2008. Wireless sensor networks. *IEEE Computer* **41(10)**:92–95 DOI 10.1109/MC.2008.441.

Subba B, Biswas S, Karmakar S. 2018. A game theory based multi layered intrusion detection framework for wireless sensor networks. *International Journal of Wireless Information Networks* **25(4)**:399–421 DOI 10.1007/s10776-018-0403-6.

Sánchez JFM, Parra OJS, Medina J. 2019. Design and implementation of an intrusion prevention system for wi-fi networks 802.11 ac. In: Renault É, Boumerdassi S, Bouzefrane S, eds. *Mobile, Secure, and Programmable Networking*. Cham: Springer International Publishing, 32–41.

The Gambit Project. 2019. Gambit documentation. Release 16.0.1. *Available at https://gambitproject.readthedocs.io/_/downloads/en/stable/pdf/* (accessed 8 June 2021).

Wang K, Du M, Yang D, Zhu C, Shen J, Zhang Y. 2016. Game theory-based active defense for intrusion detection in cyber-physical embedded systems. *ACM Transactions on Embedded Computing Systems* **16(1)**:1–21 DOI 10.1145/2886100.

Wittenburg G, Schiller J. 2006. Running real-world software on simulated wireless sensor nodes. In: *The ACM Workshop on Real-World Wireless Sensor Networks (REALWSN '06)*. 7–11.

Wu R, Deng X, Lu R, Shen X. 2015. Trust-based anomaly detection in emerging sensor networks. *International Journal of Distributed Sensor Networks* **2015(7)**:1–14 DOI 10.1155/2015/363569.

Yin X, Li S. 2019. Trust evaluation model with entropy-based weight assignment for malicious node's detection in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking* **2019(1)**:602 DOI 10.1186/s13638-019-1524-z.

Youssef M, Youssef A, Younis M. 2009. Overlapping multihop clustering for wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems* **20(12)**:1844–1856 DOI 10.1109/TPDS.2009.32.

Yu B, Xiao B. 2006. Detecting selective forwarding attacks in wireless sensor networks. In: *20th International Parallel and Distributed Processing Symposium (IPDPS 2006)*.

Zhang Q, Yu T, Ning P. 2006. A framework for identifying compromised nodes in sensor networks. In: *International Conference on Security & Privacy in Communication Networks 2006 (Securecom & Workshops 2006)*. 1–10.