An Anomaly-Based Intrusion Detection System for the Smart Grid Based on CART Decision Tree

Panagiotis I. Radoglou-Grammatikis, *Student Member, IEEE,* and Panagiotis G. Sarigiannidis, *Member, IEEE*

***Abstract*—The Smart Grid (SG) paradigm constitutes the new technological evolution of the traditional electrical grid, providing remote monitoring and controlling capabilities among all its operations through computing services. These new capabilities offer a lot of benefits, such as better energy management, increased reliability and security, as well as more economical pricing. However, despite these advantages, it introduces significant security challenges, as the computing systems and the corresponding communications are characterized by several cybersecurity threats. An efficient solution against cyberattacks is the Intrusion Detection Systems (IDS). These systems usually operate as a second line of defence and have the ability to detect or even prevent cyberattacks in near real-time. In this paper, we present a new IDS for the Advanced Metering Infrastructure (AMI) utilizing machine learning capabilities based on a decision tree. Decision trees have been used for multiple classification problems like the distinguishment between the normal and malicious activities. The experimental evaluation demonstrates the efficiency of the proposed IDS, as the Accuracy and the True Positive Rate of our IDS reach 0.996 and 0.993 respectively.**

***Index Terms*—Advanced Metering Infrastructure, Intrusion Detection System, Security, Smart Grid**

I. Introduction

The Smart Grid (SG) paradigm constitutes the new generation of the conventional electrical grid, where its operations are monitored and controlled through Information and Communication Technology (ICT) services. According to [1], SG will probably form the largest application of the Internet of Things (IoT), which will be called Enernet. In particular, SG provides the communication architecture which enhances the energy generation, transmission and distribution processes, providing multiple benefits both for energy consumers and utility companies. On the one hand, energy consumers can monitor the energy consumption in real time resulting in more economical pricing. On the other hand, through remote control operations, utility companies can activate self-healing and self-maintenance mechanisms providing this way more reliability and security.

Although SG provides multiple benefits, it also induces significant security challenges. SG combines a set of heterogeneous technologies, such as smart meters, Supervisory Control and Data Acquisition (SCADA) systems, electric vehicles, automation substations and synchrophasor systems. Each of these technologies is characterized by various security threats that can cause devastating consequences such as power outage and brownout [2]. For instance, in 2015, utilizing spear-phishing techniques, a Russian hacker group attacked the Ukrainian electrical grid, resulting in a power outage which affected more than 225000 consumers [3], [4]. In the SG paradigm, the cyberattacks usually target the availability and integrity of computing systems, firstly, while targeting the confidentiality secondly. For instance, various kinds of Denial of Service (DoS) and Distributed DoS (DDoS) attacks can compromise the availability of systems. On the other hand, False Data Injection (FDI) attacks can jeopardize the integrity of information. Finally, Man-in-The-Middle (MiTM) attacks can threaten the confidentiality of systems.

An effective countermeasure against the previous security threats is Intrusion Detection Systems (IDS). The rapid evolution of computing systems resulted in the need to create intelligent mechanisms, such as IDS that can detect or even prevent security threats in near real-time. A significant advantage of these systems is that they can detect zero-day attacks, using Machine Learning (ML) techniques. The existence of these systems in SG is necessary, as in this environment possible cyberattacks can generate disastrous consequences. Therefore, in this paper, we provide an IDS for the Advanced Metering Infrastructure (AMI) which utilizes a decision tree in order to detect possible cyberattacks. Specifically, we present an anomaly-based IDS for SG, utilizing an up to date dataset which was created in 2017 and includes multiple types of cyberattacks. Our IDS aims at predicting abnormal behavior patterns in the network traffic, which was sent and received by a specific component of AMI, the data collector. The detection process is based on a decision tree which can efficiently recognize normal and abnormal behaviors. The ACC and True Positive Rate (TPR) demonstrate the efficiency of our IDS, as they are calculated at 0.996 and 0.993 respectively.

The rest of this paper is organized as follows: Section II presents related IDS systems for AMI. Section III introduces a brief overview of SG, focusing on AMI. In section IV, we provide background for IDS systems. Similarly, section V provides background for decision trees. Section VI analyzes the proposed IDS. Section VIII evaluates our IDS and finally, section VIII concludes this paper.

II. Related Work

Many authors have examined the use of IDS in the SG paradigm. This section describes briefly these efforts.

In [5], the authors presented a signature-based IDS which can detect active power limitation attacks. The proposed IDS is based on a stateful analysis plugin which is integrated into the Suricata IDS. This plugin is composed of three

P. I. Radoglou-Grammatikis and P. G. Sarigiannidis are with the Department of Informatics and Telecommunications Engineering, University of Western Macedonia, Kozani 50100, Greece e-mail: {pradoglou,psarigiannidis}@uowm.gr

P. I. Radoglou-Grammatikis and P. G. Sarigiannidis, "An anomaly-based intrusion detection system for the smart grid based on cart decision tree,"in2018 Global Information Infrastructure and Networking Symposium(GIIS), Oct 2018, pp. 1–5.

The published version is available at IEEE Xplore, October 2018, pp. 1-5, DOI: 10.1109/GIIS.2018.8635743

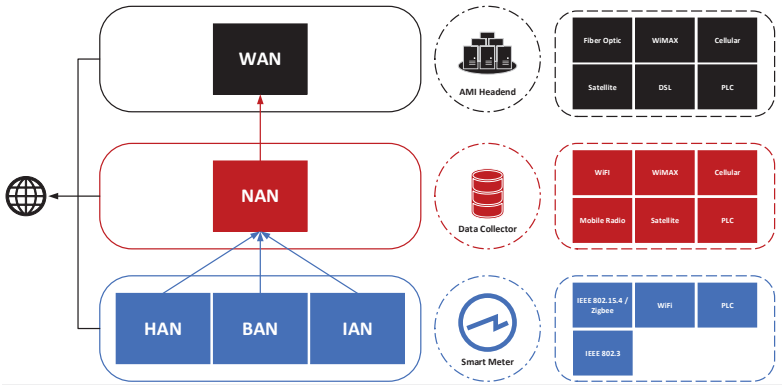

Fig. 1: Communication architecture of AMI.

modules: a) protocol decoder, b) rule analysis engine and c) state manager. The evaluation of the proposed IDS was made through two attack examples. However, the authors do not provide numerical analysis regarding the efficiency of their IDS.

In [6], the authors introduced an anomaly-based IDS for the AMI protection, assessing multiple ML algorithms. The introduced IDS consists of individual IDSs that monitor and control each component of AMI. In order to deploy and test ML algorithms, they utilized the KDD CUP 1999 and NSL KDD datasets.

In [7], A. Patel et al. also developed an anomaly-based IDS for SG, where operation is based on a Support Vector Machine (SVM), an Ontology Knowledge Base (OKB) and a fuzzy analyzer. In order to deploy and test SVM, they utilize the KDD 1999 dataset and their own experiments. Their experimental analysis demonstrates the efficiency of the proposed IDS, as the Area Under Curve (AUC) metric approaches 0.994.

R. Vijayanand et al. [8] proposed an IDS for AMI. Based on multiple SVMs, the proposed IDS can identify various security threats. In particular, they deployed three SVMs with different kernel functions. For the training and testing processes, they used the ADFA-LD dataset. They argue that their system approaches 99% Accuracy (ACC).

In [9], R. Berthier et al. presented a specification-based IDS for the AMI communications that utilize the ANSI C12.22 protocol. The proposed IDS is based on specification rules that are organized into three classes: a) device rules, b) network rules and c) application rules. They claim that the True Negative Rate (TNP) and TPR approach 99.57% and 100% respectively.

In [10] R. Mitchell and R. Chen introduced a distributed IDS for the protection of AMI. In particular, the proposed IDS consists of multiple IDSs that apply a predefined set of specification rules. For each component of AMI, the appropriate specification rules have been determined. The authors claim that the False Positive Rate (FPR) and TPR reach 0.2% and 100% respectively.

Each of the previous cases presents the corresponding advantages and disadvantages. Specifically, the specification-based IDS [5] possibly presents high ACC, but it cannot detect unknown types of cyberattacks. The anomaly-based IDSs [6]–[8] are able to detect zero-day cyberattacks, but the most of them have been deployed using outdated datasets that present material weaknesses [11]. Finally, the specification-based IDSs [9], [10] present high ACC and are able to detect zero-day attacks, but in a dynamic environment, such as SG, the corresponding specification rules have to be updated continuously.

III. Smart Grid Background

The SG paradigm can be defined as the interconnection between the traditional electrical grid with ICT services, allowing the two-way communication between energy consumers and utility companies as well as the remote control of all operations from the generation process toward the distribution process. SG combines multiple and heterogeneous technological entities such as smart meters, SCADA devices, automation substations, electrical vehicles and microgrids. The most crucial part of SG is AMI which enables the interaction between energy consumers and utility companies. Fig. 1 illustrates an architectural model of AMI, by presenting the most critical entities and the communication among them. In particular, AMI consists of three components that belong to different communication areas. Smart meters constitute the first component of AMI and are responsible for monitoring and recording the electricity consumption and other statistics either of Home Area Networks (HAN) or Business Are Networks (BAN) or Industry Area Networks (IAN). The second component of AMI is the data collector which is deployed in a Neighbour Area Network (NAN) and undertakes to aggregate the information received from smart meters. The last component is the AMI headend which correspondingly collects the data received from multiple data collectors.

IV. IDS Background

An IDS system aims at detecting or even preventing possible security threats timely. A typical architecture of IDS consists of three components: a) agents, b) analysis engine and c) response module. More specifically, it can incorporate one or

more agents that are responsible for monitoring and capturing the network activities of one or more computing systems. The analysis engine component undertakes to detect possible cyberattacks. Finally, the response module informs the security or system administrator about potential security violations. The analysis engine component can integrate various mechanisms to detect cyberattacks. These mechanisms can be classified into three categories: a) signature-based, b) anomaly-based and c) specification-based. The first category is based on the matching of the activities that are detected by the agents with a predefined set of cyberattack patterns called signatures. The second category aims at identifying abnormal behaviors patterns by comparing features of normal and abnormal data. Usually, this category adopts methods from ML such as, decision trees, Artificial Neural Networks (ANN) and clustering algorithms. Moreover, this category is characterized by the capability to detect zero-days cyberattacks. Finally, the third category is based on matching of the network activities with a predefined set of normal behavior features called specifications.

V. Decision Trees Background

The anomaly-based IDS systems adopt specific algorithms from the Artificial Intelligence (AI) field in order to identify unknown types of intrusions that are not included in cyberattacks signatures. In particular, the techniques used are part of the field of ML, which explores the study and deployment of mechanisms that aim at predicting unknown situations. In this section, we provide an overview of decision trees, which constitute a popular supervised ML mechanism for classification problems.

A. Classification Problem Overview

ML includes various categories, such as supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. These categories can solve various problems like classification, regression and clustering. The classification problem refers to the learning process of a target function capable of matching unknown instances to a predetermined set of categories. Usually, supervised learning techniques are used for this problem. In particular, the goal of the learning process is to generate a model, which will be able to predict the categories of an unknown set of instances (testing set) based on specific features. For this process, it is necessary a labelled set of instances (training set) that has to be representative of each class. Therefore, the learning process aims to train a mechanism with a training set in order to classify unknown instances into predefined categories with high ACC. This process is called training process and the provided mechanism is called classifier. When the training process is completed, the classifier can be used for testing processes.

B. Decision Trees Overview

A decision tree is an efficient mechanism for classification and regression processes. In particular, it consists of multiple nodes that can be characterized either as internal or leaves. The internal nodes possess outgoing edges, while aiming at dividing the entire instance space into smaller sub-spaces that will be as homogeneous as possible concerning the corresponding classes. In more detail, they divide the entire instance space based on the values of specific features of the training set. On the other hand, leaves do not include outgoing edges and represent a class of the classification problem. Also, it is possible a leaf can carry a probability vector, which indicates the probability for each class. Hence, a directed tree is formed through which the classification of the various instances is possible, following the paths of the tree. Specifically, each path of the tree can be interpreted as a logical rule.

Many algorithms can generate a decision tree automatically, utilizing a dataset. Some of them are Iterative Dichotomiser 3 (ID3), Classification And Regression Trees (CART), J48, C4.5, C5.0 Chi-square Automatic Interaction Detector (CHAID) and Quick, Unbiased, Efficient, Statistical Tree (QUEST). Based on the selected features and the use of a discrete function, these algorithms check the splitting of the training set recursively, attempting to reduce the generalization error or other evaluation measures, such as the number of nodes. In most cases, the internal nodes use a discrete function, which pursues to split the instance spaces based on the value of a single feature. This means that each algorithm focuses on finding those features that divide the instances spaces with the most effective way. There are various criteria that can be used for this purpose, such as the Information Gain (IG) and the Gini Index (GI). In this paper, we use IG, which is calculated based on the following equations.

$$I(S,A) = \frac{|S_1|}{|S|}E(S_1) + \frac{|S_2|}{|S|}E(S_2) + ... + \frac{|S_j|}{|S|}E(S_j) = \sum_{k=1}^{k=j}\frac{|S_k|}{|S|}E(S_k) \quad (1)$$

S denotes the entire instance space, which is divided into smaller sub-spaces based on the values of a specific feature A. S_k indicates a smaller sub-space, which attempts to identify a specific class. Accordingly, $|S|$ and $|S_k|$ signify the number of all sub-spaces and the number of S_k sub-spaces respectively. Finally, $E(S_k)$ denotes the entropy of S_k, which is calculated through the Equation 2. Therefore, on the basis of the above, $I(S,A)$ refers to the information resulting from the splitting of S.

$$E(S_k) = -\sum_{i=1}^{m} p_i \log_2(p_i) \quad (2)$$

m denotes the set of classes and p_i indicates the probability of the i class in the sub-space S_k.

$$IG(S,A) = E(S) - I(S,A) \leq \delta \quad (3)$$

The splitting of the entire instance space is recursively made until there is no substantial gain from additional separations.

In other words, the splitting process is recursively made until a stopping criterion is met. The Equation 3 defines IG, where $E(S)$ and $I(S, A)$ are the entropy of the entire instance space and the information resulting from the splitting of S respectively. Finally, δ denotes the stopping criterion.

C. CART Classifier

In this paper, we utilize the CART algorithm. An advantage of this algorithm is that it can be used for both classification and regression processes. Its main characteristic is that each internal node possesses two outgoing edges, thus forming a binary tree. For the splitting process, it applies the Cost Complexity Pruning method and can also use IG, GI, as well as twoing criteria. We utilized an optimal version of the CART algorithm, working with the scikit-learn library.

VI. Proposed IDS System For SG

In this section, we analyze the methodology of the proposed IDS system. The proposed IDS focuses on the network flows that are sent and received by the data collector device of the AMI architecture. We consider that this device is the most critical component of AMI, as it constitutes the intermediate point of the connection between the energy consumers and the utility companies. Consequently, it receives data both of smart meters and AMI headend. Therefore, we believe the security of the data collector device is crucial for the overall protection of SG. Based on captured network flows, the proposed anomaly-based IDS can classify them either as normal behavior or a possible cyberattack. The architecture of our IDS consists of four modules: a) Network Monitoring Module, b) Network Flows Extraction Module, c) Analysis Engine Module and d) Response Module. The following subsections analyze further these modules.

A. Network Monitoring Module

Network Monitoring Module undertakes to monitor and capture the Transmission Control Protocol/Internet Protocol (TCP/IP) traffic which is exchanged between the data collector and the other devices. This process can be executed continuously or periodically. To this end, we utilize the Scapy library, which possesses the ability to decode and manipulate a wide range of network protocols.

B. Network Flow Extraction Module

Network Flow Extraction Module receives the network traffic from the previous module and extracts the corresponding bidirectional network flows. Network flows can be classified either as unidirectional or bidirectional. The first category concerns only the network traffic, which originates from the source address to the destination address, while the second category concerns the total network traffic exchanged between two endpoints. The term network flows will be used from now on in this paper for referring to bidirectional network flows. Therefore, in the case of the TCP/IP stack, a network flow is defined based on the following features: a) source IP address, b) destination IP address, c) source network port and d) destination network port. Furthermore, a network flow can include statistic information, such as the number of the exchanged network packets, the number of the exchanged bytes and time information. Based on this information, various cyberattacks can be detected. Our implementation extracts and processes the following features:

- **Flow duration**: This feature denotes the duration of the network flow in microseconds.
- **Bwd Packet Length Min**: In the backward direction, this feature indicates the minimum length of packets.
- **Bwd Packet Length Std**: In the backward direction, this feature implies the standard deviation of the length of packets.
- **Subflow Fwd Bytes**: In the forward direction, this feature denotes the average number of bytes in a sub-flow.
- **Init Win bytes forward**: In the forward direction, this feature signifies the total number of bytes sent in the initial window.

The selection of these features was based on [11], in which the authors evaluate a plethora of statistic features utilizing the Random Forest Regression algorithm and the CICIDS2017 dataset [11].

C. Analysis Engine Module

Analysis Engine Module constitutes the core of the proposed IDS and is responsible for detecting possible cyberattacks. In particular, this module receives the selected features of network flows from the previous module, while utilizing a decision tree; thus it is able to classify network flows either as normal behavior or sa possible cyberattack. The generation of the particular decision tree was based on the CART algorithm and the CICIDS2017 dataset [11]. The specific dataset was created in 2017 and includes, among others, the aforementioned features of the network flows. Furthermore, it comprises network flows that correspond to DoS/DDoS attacks, brute force attacks, botnets, infiltration attacks, web attacks and port scanning attacks. In conclusion, based on the previous features and by utilizing the labeled network flows of the CICIDS2017 dataset [11], we deployed and trained the CART algorithm in order to generate an efficient decision tree, which is able to detect all the attacks mentioned above with high efficiency. The deployment and the testing process of the CART algorithm were implemented through the scikit-learn library. The performance of the generated decision tree is analyzed in the next section.

D. Response Module

Response Module executes the last processes of the proposed IDS. In particular, it informs the system or security administrator about possible cyberattacks.

VII. Evaluation Analysis

In order to evaluate the proposed IDS and in particular, the capability of the generated decision tree to detect possible cyberattacks with high ACC, we adopt the following performance metrics: a) True Positive (TP) which indicates the

number of cyberattacks that were recognized as cyberattacks, b) True Negative (TN), which denotes the number of normal-behavior activities that were classified as normal behavior, c) False Positive (FP), which expresses the number of normal activities that were detected as cyberattacks and d) False Negative (FN), which implies the number of cyberattacks that were identified as normal behavior. Therefore, based on these terms, the Equations 4 and 5 define the metrics that calculate the performance of our IDS. ACC is defined as the ratio of the total predictions that were correct. On the other hand, TPR measures the proportion of actual cyberattacks that were classified correctly. Emphasis is given to TPR than ACC, since the proposed IDS is mainly focused on detecting all possible cyberattacks.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$TPR = \frac{TP}{TP + FN} \tag{5}$$

As mentioned before, we deployed and trained the CART algorithm with the CICIDS2017 dataset in order to generate an efficient decision tree. For the training and testing processes, we utilized 75% and 25% of the dataset. Moreover, we tested all the possible combinations of the five features as mentioned before, i.e., 31 combinations. Consequently, we deployed and tested 31 different decision trees. Among these decision trees, the best performance based on the previous equations, is presented by applying all the aforementioned features. Table I depicts the confusion matrix of the most efficient decision tree, presenting the values of TP, TN, FP and FN. Based on these values, ACC and TPR are calculated at 0.9966 and 0.9930 respectively.

TABLE I: Confusion Matrix

	Actual Cyberattack	**Actual Normal Behavior**
Predicted Cyberattack	TP = 138735	FP = 1390
Predicted Normal Behaviors	FN = 965	TN = 566596

VIII. Conclusions

The cybersecurity of the SG paradigm and especially of AMI is crucial, since possible security violations can cause disastrous consequences. Anomaly-based IDS systems provide an efficient solution against these attacks, by providing the system and security administrators with appropriate means to detect or even prevent potential threats automatically. The presence of such systems in the protection of AMI is necessary as they can timely detect cyberattacks or even zero-day attacks.

In this paper, we developed an IDS which aims at protecting the data collector device of AMI. The proposed IDS utilizes a decision tree which can detect various cyberattacks such as brute force attacks, DoS/DDoS, web attacks, infiltration attacks, port scanning and botnets. In particular, it monitors the captured network flows and classifies them either as normal behavior or as possible cyberattack. The evaluation results demonstrate the efficiency of the decision tree, as ACC and TPR are calculated at 0.9966 and 0.9930 respectively.

In our future work, we intend to implement a distributed anomaly-based IDS system, which will monitor and control the network activity of all components of AMI. More specifically, it will consist of individual IDS agents that will monitor the network activities of smart meters, data collectors and the AMI headends. The IDS agents will communicate with a central server, which will be responsible for the detection and visualization processes. Furthermore, the proposed IDS will be able to detect the type of cyberattacks.

IX. Acknowledgement

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No. 787011 (SPEAR).

References

[1] S. Tan, D. De, W. Song, J. Yang, and S. K. Das, "Survey of security advances in smart grid: A data driven approach," *IEEE Communications Surveys Tutorials*, vol. 19, no. 1, pp. 397–422, Firstquarter 2017.

[2] Z. E. Mrabet, N. Kaabouch, H. E. Ghazi, and H. E. Ghazi, "Cyber-security in smart grid: Survey and challenges," *Computers & Electrical Engineering*, vol. 67, pp. 469 – 482, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0045790617313423

[3] N. Falliere, L. O. Murchu, and E. Chien, "W32. stuxnet dossier," *White paper, Symantec Corp., Security Response*, vol. 5, no. 6, p. 29, 2011.

[4] A. Hansen, J. Staggs, and S. Shenoi, "Security analysis of an advanced metering infrastructure," *International Journal of Critical Infrastructure Protection*, vol. 18, pp. 3 – 19, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1874548217300495

[5] B. K. K. M. S. Sezer, "Towards a stateful analysis framework for smart grid network intrusion detection," in *Proceedings of the 4th International Symposium for ICS & SCADA Cyber Security Research*, 2016, p. 124.

[6] M. A. Faisal, Z. Aung, J. R. Williams, and A. Sanchez, "Data-stream-based intrusion detection system for advanced metering infrastructure in smart grid: A feasibility study," *IEEE Systems Journal*, vol. 9, no. 1, pp. 31–44, March 2015.

[7] A. Patel, H. Alhussian, J. M. Pedersen, B. Bounabat, J. C. Jnior, and S. Katsikas, "A nifty collaborative intrusion detection and prevention architecture for smart grid ecosystems," *Computers & Security*, vol. 64, pp. 92 – 109, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404816300748

[8] R. Vijayanand, D. Devaraj, and B. Kannapiran, "Support vector machine based intrusion detection system with reduced input features for advanced metering infrastructure of smart grid," in *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Jan 2017, pp. 1–7.

[9] R. Berthier, W. H. Sanders, and H. Khurana, "Intrusion detection for advanced metering infrastructures: Requirements and architectural directions," in *2010 First IEEE International Conference on Smart Grid Communications*, Oct 2010, pp. 350–355.

[10] R. Mitchell and I. Chen, "Behavior-rule based intrusion detection systems for safety critical smart grid applications," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1254–1263, Sept 2013.

[11] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." in *ICISSP*, 2018, pp. 108–116.