DIDEROT: An Intrusion Detection and Prevention System for DNP3-based SCADA Systems

Panagiotis Radoglou-Grammatikis
Department of Electrical and Computer Engineering, University of Western Macedonia
Kozani, Greece
pradoglou@uowm.gr

Panagiotis Sarigiannidis
Department of Electrical and Computer Engineering, University of Western Macedonia
Kozani, Greece
psarigiannidis@uowm.gr

George Efstathopoulos
0infinity Limited
London, UK
george@0infinity.net

Paris-Alexandros Karypidis
SIDROCO HOLDINGS Ltd
Nicosia, Cyprus
pkarypidis@sidroco.com

Antonios Sarigiannidis
SIDROCO HOLDINGS Ltd
Nicosia, Cyprus
asarigia@sidroco.com

ABSTRACT

In this paper, an Intrusion Detection and Prevention System (IDPS) for the Distributed Network Protocol 3 (DNP3) Supervisory Control and Data Acquisition (SCADA) systems is presented. The proposed IDPS is called DIDEROT (Dnp3 Intrusion DetEction pReventiOn sysTem) and relies on both supervised Machine Learning (ML) and unsupervised/outlier ML detection models capable of discriminating whether a DNP3 network flow is related to a particular DNP3 cyberattack or anomaly. First, the supervised ML detection model is applied, trying to identify whether a DNP3 network flow is related to a specific DNP3 cyberattack. If the corresponding network flow is detected as normal, then the unsupervised/outlier ML anomaly detection model is activated, seeking to recognise the presence of a possible anomaly. Based on the DIDEROT detection results, the Software Defined Networking (SDN) technology is adopted in order to mitigate timely the corresponding DNP3 cyberattacks and anomalies. The performance of DIDEROT is demonstrated using real data originating from a substation environment.

CCS CONCEPTS

• **Security and privacy** → *Intrusion detection systems*; • **Computing methodologies** → *Neural networks.*

KEYWORDS

Anomaly Detection, Autonencoder, Intrusion Detection, Machine Learning, SCADA, SDN, Smart Grid

ACM Reference Format:
Panagiotis Radoglou-Grammatikis, Panagiotis Sarigiannidis, George Efstathopoulos, Paris-Alexandros Karypidis, and Antonios Sarigiannidis. 2020. DIDEROT: An Intrusion Detection and Prevention System for DNP3-based SCADA Systems. In *The 15th International Conference on Availability, Reliability and Security (ARES 2020), August 25–28, 2020, Virtual Event, Ireland.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3407023.3409314

ARES 2020, August 25–28, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-8833-7/20/08…$15.00
https://doi.org/10.1145/3407023.3409314

1 INTRODUCTION

In the era of hyper-connected digital economies, the smart technologies play a vital role in the operation of the electrical grid, transforming it into a new, decentralised model with multiple benefits, such as distributed generation, pervasive control, remote monitoring and self-healing. According to S.Tan et al. [39], the Smart Grid (SG) is going to constitute the greatest paradigm of the Internet of Things (IoT). However, although this new reality introduces significant advantages, it also raises crucial security and privacy risks due to the heterogeneous characteristics of the involved legacy and smart energy systems. In particular, SG includes legacy systems, such as Supervisory Control and Data Acquisition System (SCADA) that rely on protocols designed without having security in mind. On the other side, the rapid progression of IoT makes harder the security and information management of the various entities [13]. Therefore, SG is exposed against a plethora of cyberattacks and malware, including Denial of Service (DoS), data privacy breaches and ransomware. It is noteworthy that due to the interconnected and independent nature of the Critical Infrastructure (CIs), a failure in SG can affect significantly other CIs, generating cascading effects and endangering the general population and economy across national borders [5]. A characteristic example was the cyberattack against the Ukrainian substation, resulting in the power outage for more than 225,000 people [33]. More recent cases are the Dragonfly 2.0 Advance Persistent Threat (APT) campaign against multiple energy companies [32] and the cyberattack against the US electrical grid in 2019 [22].

Both academia and industry have identified countermeasures in order to address the cybersecurity issues of SG. Specifically, IEC 62351 [17] defines a set of security and privacy guidelines for Industrial Control Systems (ICS) based on existing technologies. Moreover, Intrusion Detection and Prevention Systems (IDPS) can handle a lot of information, thus recognising possible intrusions.

Panagiotis Radoglou-Grammatikis, Panagiotis Sarigiannidis, George Efstathopoulos, Paris-Alexandros Karypidis, and Antonios Sarigiannidis. 2020. DIDEROT: an intrusion detection and prevention system for DNP3-based SCADA systems. In Proceedings of the 15th International Conference on Availability, Reliability and Security (ARES '20). Association for Computing Machinery, New York, NY, USA, Article 115, 1–8. DOI:https://doi.org/10.1145/3407023.3409314

Undoubtedly, these solutions are valuable, eliminating cybersecurity incidents. However, the rapid evolution of cyberattacks and malware requires the continuous adoption of the appropriate measures. The guidelines of IEC 62351 limit to specific security controls related to industrial protocols and cannot be quickly adopted by SCADA, especially when they operate since safety issues can occur, such as brownouts or even blackouts. On the other side, the current IDPS tools cannot recognise potential cyberattacks and anomalies against the industrial application layer protocols. Therefore, the challenge of ensuring smart, safe, sustainable and efficient SG becomes major as the road ahead for SG is a difficult one, filled with significant and far-reaching challenges to sustainability.

In this paper, we present a Machine Learning (ML)-based IDPS capable of detecting and preventing cyberattacks and anomalies against the Distributed Network Protocol (DNP3) protocol. The proposed IDPS is called DIDEROT (Dnp3 Intrusion DetEction pReventiOn sysTem) and relies on network flow statistics, including two detection layers: a) intrusion detection and b) anomaly detection that work supplementarily. The first layer (i.e., intrusion detection) relies on supervised ML detection methods and is responsible for recognising particular DNP3 cyberattacks, including a) injection, b) flooding, c) DNP3 reconnaissance, d) replay attacks and e) masquerading. The second layer (i.e., anomaly detection) is activated when the first layer classifies a network flow as normal. Thus, the second layer undertakes to identify whether a DNP3 anomaly takes place either due to a security violation or a possible electricity disturbance. Moreover, it is noteworthy that DIDEROT takes full advantage of the Software Defined Networking (SDN) [35] technology in order to mitigate the DNP3 cyberattacks/anomalies [10]. In particular, based on the detection results, DIDEROT informs appropriately the SDN controller in order to drop the corresponding malicious/anomalous network flows. Therefore, the contributions of this paper are summarised in the following sentences:

- **Detecting DNP3 cyberattacks**: DIDEROT can detect a plethora of DNP3 cyberattacks by using supervised ML detection methods and particularly a decision tree classifier.
- **Detecting DNP3 anomalies**: DIDEROT can recognise DNP3 anomalies that take place either due to a security violation or an electricity disturbance. To this end, both unsupervised/outlier and semisupervised/novelty detection methods were investigated. Moreover, an autoencoder Deep Neural Network (DNN) was developed for this purpose called DIDEROT Autoencoder.
- **Evaluating a plethora of ML methods in recognising DNP3 cyberattacks and anomalies**: Various ML methods were assessed, using real DNP3 network traffic data originating from a substation environment.
- **Mitigating DNP3 cyberattacks/anomalies**: Based on the DIDEROT detection results, DIDEROT takes full advantage of SDN in order to mitigate timely the DNP3 cyberattacks and anomalies.

The rest of this paper is organised as follows. Section 2 discusses relevant works. Section 3 provides a background related to a) the DNP3 SCADA systems, b) IDPS systems and c) ML methods used for detecting intrusions. Next, section 4 analyses the DIDEROT architecture. Finally, section 6 is devoted to the DIDEROT evaluation, while section 7 concludes this paper.

2 RELATED WORK

This section is devoted to describing previous similar works. Each paragraph focuses on a separate case, analysing the proposed Intrusion Detection System (IDS) architecture, the detection mechanisms, their efficiency and the relevant cyberattacks. Finally, based on this concise analysis, the limitations and shortcomings of the existing intrusion detection methods are summarised.

Multiple survey papers have examined thoroughly the various IDS devoted to protecting SG. Some of them are listed below [15, 20, 33, 34]. In our previous work in [33], we have examined in detail 37 IDS cases specially designed for a) the entire SG ecosystem, b) AMI, c) SCADA, d) substations and e) synchrophasors. In particular, after providing the necessary background in the SG elements, communications, the typical IDS architecture and the detection methods, the appropriate requirements of an IDS for SG were identified. Next, a detailed analysis of 37 IDS cases follows, describing their architecture, the detection methods and their evaluation results. Based on this study, the limitations of the existing IDS for SG are identified, and directions for future research efforts are given.

More specifically, in [25], S. Lee et al. present a method called LDA-based Automatic Rule Generation (LARGen), which can identify and generate signatures used by signature-based IDS. Based on the authors, a signature is considered as a set of hexadecimal or American Standard Code for Information Interchange (ASCII) characters that identify a malicious pattern. If a network packet includes this pattern in the header or in the payload, then it is detected successfully by a signature-based IDS containing the specific signature. The functionality of LARGen relies on the Latent Dirichlet Allocation algorithm [19]. In particular, LARGen first exports signature strings from network flows; then categorises the network flows based on the exported signature strings and finally, through LDA identifies key content strings indicating the malicious traffic. The architecture of LARGen consists of three main modules, namely: a) Pre-processing: Construction of Documents, b) LDA-based Topic Modeling and c) Attack Signature Generation. The applicability and efficiency of LARGen were evaluated, utilising a plethora of experiments, including the following cyberthreats: a) trojan horses, b) overflow attacks, c) DoS, d) injection, e) insecure methods, f) arbitrary code execution and g) others. Based on the experimental results, LARGen presents 1.6% False Positives (FP).

In [9], N. Dutt et al. introduce an anomaly-based IDS imitating the operation of the Human Immune System (HIS). The architecture of the proposed IDS is composed of two intrusion detection mechanisms: a) Statistical Modeling based Anomaly Detection (SMAD) and b) Adaptive Immune-based Anomaly Detection (AIAD). In particular, SMAD imitates the behaviour of the HIS innate immune system, while AIDS works like the adaptive immune system, considering both T-cells and B-cells. The effectiveness of SMAD and AIAD was validated, using real-time network traffic data as well as traces from the UNSW-NB15 [30] intrusion detection dataset. The True Positive Rate (TPR) of SMAD reaches 97%, while the TPR of AIAD approaches 99%.

In our previous work [11], we developed an anomaly-based IDS system for SG, utilising real operational data (i.e., raw electricity measurements) originating from a power plant in Greece. The architecture of the proposed IDS is composed of 6 modules, namely, a) Data Collection Module, b) Pre-processing Module, c) Feature Selection Module, d) Model Training, e) Anomaly Detection Module and f) Response Module. Several ML methods were evaluated under different time window values, including Principal Component Analysis (PCA), OneClass Support Vector Machine (SVM), Isolation Forest, Angle-Based Outlier Detection (ABOD), Stochastic Outlier Selection (SOS) and autoencoder. Based on the evaluation results, the overall detection improvement due to the proposed complex data representation method is validated.

O. Igbe et al. in [18] introduce and IDS devoted to the DNP3 communications. The architecture of the proposed IDS consists of three primary blocks: a) Packet Capture block, b) Pre-processing block and c) Deterministic Dendritic Cell Algorithm (dDCA) [14] Signal Processing block. In particular, the first block undertakes to monitor and capture the DNP3 network packets; the following one executes the necessary pre-processing actions while the last is responsible for the detection process. A DNP3 intrusion detection dataset was created for the development and evaluation of the proposed IDS, including DNP3 packet modification and injection attacks, DNP3 cold restart attacks, Distributed Denial of Service (DDoS), DNP3 Disable unsolicited attacks and Man in The Middle (MiTM). The performance of the proposed IDS is demonstrated using Receiver Operating Characteristic (ROC) curves.

In [22], S. Kwon et al. focuses on the DNP3 and IEC 61850 protocols. The proposed Network Based Intrusion Detection Systems (NIDS) relies on a Bidirectional Recurrent Neural network (BRNN) [37], which analyses both the header and payload of DNP3 packets. Thus, BRNN can detect three cyberattacks, including Cyber-Physical (CPS) malware behaviour, disabling reassembly attacks and False Data Injection (FDI) [26, 27] attacks. Based on the authors, the first cyberattack includes five subcategories: a) reconnaissance attacks, b) DoS Operate, c) DoS Cold Restart, d) DoS Warm Restart and e) File Transfer. The proposed IDS was applied and validated in an IEEE 1815.1-based Korean substation. However, the relevant cyberattack data was produced by emulating the corresponding cyberattacks through the Triangle MicroworksâĂŹ Distributed Test Manager (DTM) [29] and Ostinato [31, 38], which is an opensource network traffic generator. A numerical analysis demonstrates the efficiency of the proposed NIDS.

X. C. Yin et al. in [40] present also an IDS for the DNP3 protocol. First, they studied the vulnerabilities of DNP3. Next, they executed relative DNP3 cyberattacks in order to collect necessary data and finally, they provide a DNP3 IDS, which adopts ML classification and visualisation processes. Their IDS consists of three main modules, namely a) Data Input System, b) Data Analysis System and c) Classification and Detection System. The Data Input System is responsible for collecting the appropriate data needed for ML. Multiple data were used, including the cyberattacks' outcomes performed by the authors as well as traces from various malware, including a) Triton [7], b) Industroyer, c) BlackEnergy [21], d) Stuxnet [12, 23], e) Duqu [3], f) Flame [3] and g) Gauss [3]. The normal data was produced from experiments simulating normal DNP3 operations as well as from publicly available datasets. Subsequently, the Data Analysis System pre-processes this data, and finally, the Classification and Detection System applies an ML classifier, thus discriminating the benign and abnormal states. The efficiency of the proposed IDS is demonstrated based on a visualisation diagram depicting which points are classified as benign or malicious.

Admittedly, the previous works provide significant insights and useful methodologies. In particular, each detection category is characterised by the corresponding advantages and disadvantages. In particular, the signature-based IDS are characterised by high TPR; however, they are unable to recognise zero-day attacks or unknown anomalies. On the other side, the anomaly-based IDS adopting statistical analysis and ML solutions can identify unknown threats, but yield high False Positive Rate (FPR). Finally, the specification-based IDS are characterised by low FPR and can detect unknown anomalies; nevertheless, the corresponding specification rules should be updated accordingly based on the operation of the SG environment.

3 BACKGROUND

This section introduces some necessary background information about a) the DNP3 SCADA systems, b) IDPS and c) intrusion detection processes relying on ML methods. In particular, the typical architectures of the SCADA and IDPS systems are described, while functional characteristics of the DNP3 protocols and the ML methods used for detecting intrusions/anomalies are detailed.

3.1 DNP3 SCADA Systems

A SCADA system [32] is a dominant component of a CI, which monitors, automates and controls the operations of the industrial equipment. Therefore, it constitutes a crucial target for cybercriminals. In particular, a SCADA system consists of five elements: a) measurement instruments, b) logic controllers, c) a Master Terminal Unit (MTU), d) a communication interface and e) a Human Machine Interface (HMI). The measurement instruments refer to sensors and actuators that monitor the physical environment and collect relevant measurements, such as temperature, voltage and current. A logic controller is usually a Programmable Logic Controller (PLC) or Remote Terminal Unit (RTU), responsible for collecting the measurements by the previous measurement instruments, identifying operational abnormalities and activating/deactivating or configuring other devices. The MTU is a host or a server through which the system operator has the ability to control and configure the logic controllers usually via a friendly user interface called HMI. Finally, the interaction between MTU and the logic controllers is achieved through a communication interface, which relies on industrial protocols, such as DNP3.

DNP3 [32] is a reliable protocol adopted in CIs, mainly in the US. In a SCADA system, DNP3 is used to exchange messages between a master (i.e., MTU) and outstation or differently slave (i.e., PLC or RTU). It can be applied with several network topologies, such as a) point-to-point, where one master and an outstation interact with each other, b) multiple-drop, where there are one or more masters and multiple outstations and c) the hierarchical topology, where a device can play both roles of a master and an outstation. DNP3 is coming with three layers, namely, a) link layer, b) transport layer and c) application layer. The link-layer provides addressing services, link control, error checking, data fragmentation and multiplexing.

The DNP3 transport layer is similar to the transport layer of the Open Systems Interconnection (OSI) model and is represented by only one byte used for fragmenting DNP3 packets. Finally, the application layer defines a set of data messages used for managing and controlling the SCADA applications. Apart from the serial line DNP3 application, DNP3 can operate upon Transmission Control Protocol/Internet Protocol (TCP/IP) where all DNP3 layers are incorporated into the application layer of TCP/IP.

Based on N. Rodofile et al. [36], in this paper, we investigate five DNP3 cyberattacks: a) injection, b) flooding, c) DNP3 reconnaissance, d) replay and e) masquerading. The first cyberattack refers to a malicious insider that injects malicious DNP3 packets. The second attack denotes a DNP3 DoS, where the cyberattacker floods the target with multiple DNP3 packets. Next, a DNP3 reconnaissance refers to particular DNP3 packets that are sent to the target system in order to identify whether it uses the DNP3 protocol or not. The replay attack captures legitimate DNP3 packets and re-transmit them after a specific delay. Finally, the masquerading attack impersonates the DNP3 behaviour of the legitimate asset.

3.2 IDPS Systems

Based on the Request For Comments (RFC) document 2828 [33], intrusion detection is defined as the process aiming to audit and investigate security events in order to recognise a possible security policy violation. In 1980, the IDS term was introduced as a hardware or software system capable of automating the intrusion detection process. In particular, in 1980, J. Anderson [2] highlighted the significance of the log files during an intrusion detection procedure. Another remarkable case is the paper [6] of D. Denning, who defined a theoretical IDS model based on abstract feature patterns. According to D. Denning, if a system cannot operate based on its specifications, then it has been probably affected by a threat.

A typical IDS architecture consists of three main elements: a) Agents, b) Analysis Engine and c) Response Module [33]. Agents undertake to monitor the examined infrastructure, thus collecting and sometimes pre-processing the necessary data for the detection process. Based on the position of Agents, two IDS types are distinguished: a) Host-based IDS (HIDS) and NIDS. In the first case, an Agent can monitor only a specific host, thereby detecting anomalies only related to this host. On the other side, the Agent is placed in a specific network point, where the entire network traffic can be captured, thus detecting anomalies in the overall network. The Analysis Engine is the core component of an IDS, which receives the information of the various Agents and implements the intrusion detection process. Finally, the Response Module receives the outcome of the Analysis Engine and notifies the responsible user/operator. Sometimes, the Response Module can perform some automate mitigation strategies, such as the activation of specific firewall rules in order to mitigate and prevent similar intrusions; thus, the IDPS term is used in this case.

The detection process implemented by the Analysis Engine can be classified into three main categories: a) signature-based detection, b) anomaly-based detection and c) specification-based detection [33]. The first category defines specific rules called signatures that reflect malicious patterns. If the characteristics of the monitoring data match with those of the signatures, then a possible security violation takes place. On the other side, anomaly-based detection applies statistical analysis and Artificial Intelligence (AI) methods. Finally, the specification-based detection defines a set of rules called now specifications that define the normal operation of the monitored system/infrastructure. If the characteristics of the monitored data do not agree with those of the specifications, then a security violation is carried out.

3.3 Intrusion Detection based on ML Methods

In this subsection, a brief overview of ML-based intrusion detection processes is provided. More detailed information is available in recent survey papers in [1, 4, 16]. Despite the existence of various ML methods, all of them follow the subsequent phases.

- **Preprocessing Phase**: This phase processes appropriately the input data so that it will be in accordance with the corresponding ML model. Usually, data-preprocessing methods are applied, such as min-max scaling, normalisation, standardisation, robust scaler and max abs scaler.
- **Training Phase**: The ML model is trained with normal or/and abnormal data pre-processed data called features. There are multiple ML methods for detecting anomalies. They can be divided into three main categories, namely, a) supervised detection methods, b) unsupervised/outlier detection methods and c) semi-supervised/novelty detection methods. The first category uses labelled data, such as "Normal" or "Anomaly" or labels indicating the cyberattack type. Characteristic examples of this method are: SVM, neural networks and decision trees. The second category relies mainly on clustering techniques and unlabeled datasets, assuming that the majority of the instances are normal; however, the unlabelled datasets can comprise some outliers. Characteristic examples of this case are: ABOD, Isolation Forest, Local Outlier Factor (LOF), SOS and k-means. Finally, the semi-supervised/novelty detection methods use training data that does not include outliers. Therefore, the ML model aims to identify whether a new observation is an outlier or not. In this case, the outlier is named novelty. One class deep neural networks and One Class-SVM compose examples of this category.
- **Prediction Phase**: After the training phase, the ML model can be deployed in order to predict unknown data after the execution of the same pre-processing tasks of the first phase.

Regarding the performance of the aforementioned ML methods, particular evaluation metrics can be used, including Accuracy, TRP, FPR and the F1 score. The following equations define these metrics. True Positives (TP) indicates the number of the correct classifications that recognised successfully the cyberattacks or the anomalous behaviours. True Negatives (TN) denotes the number of the correct classifications that recognised the normal activities. FP denotes the number of the mistaken classifications that detected the normal activities as cyberattacks or anomalies. Finally, False Negatives (FN) defines the number of the wrong classifications that classified the cyberattacks or the anomalous behaviours as normal activities.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

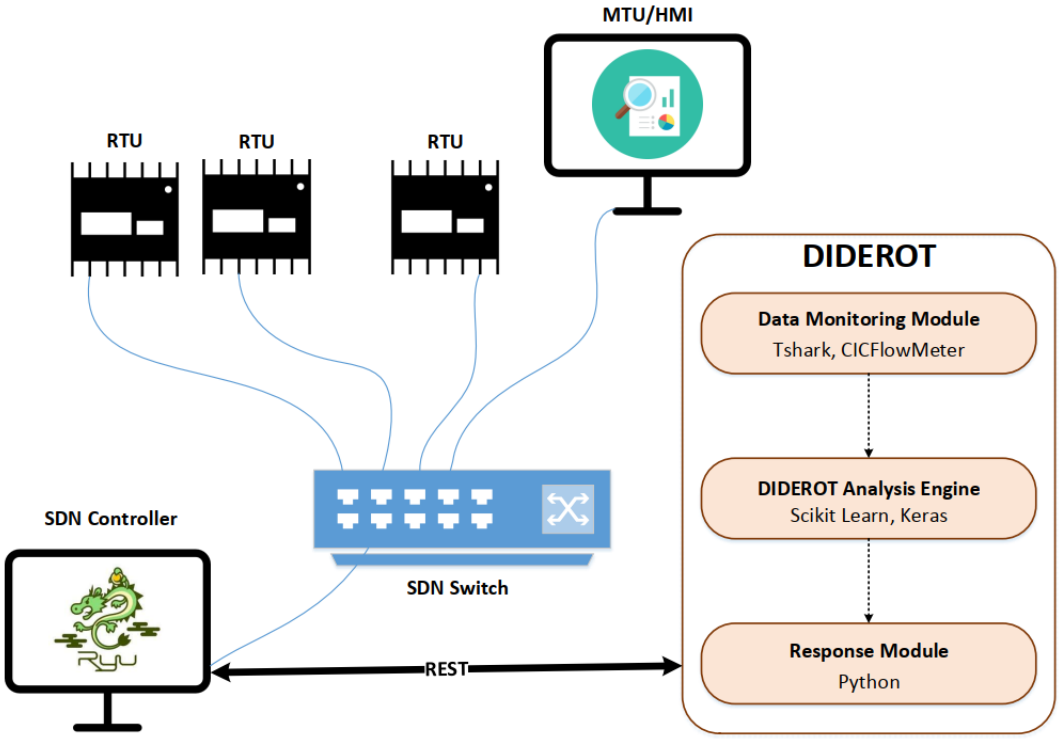

Figure 1: DIDEROT Architecture

$$FPR = \frac{FP}{FP + TN} \tag{2}$$

$$TPR = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = \frac{2 \times Precision \times TPR}{Precision + TPR} \text{ where } Precision = \frac{TP}{TP + FP} \tag{4}$$

4 DIDEROT ARCHITECTURE

Fig. 1 illustrates the architecture of DIDEROT. Based on the typical IDS architecture [33], DIDEROT consists of three modules, namely, a) Data Monitoring Module, b) DIDEROT Analysis Engine and c) Response Module. Each of them is detailed in the following subsections. The Data Monitoring Module feeds the DIDEROT Analysis Engine with the necessary data to detect DNP3 cyberattacks/anomalies. The DIDEROT Analysis Engine is responsible for the detection process, including two detection layers: a) intrusion detection and b) anomaly detection that operate complementarily. Finally, based on the outcome of the DIDEROT Analysis Engine, the Response Module generates security events and activates the SDN controller in order to mitigate timely the respective DNP3 cyberattacks/anomalies.

4.1 Data Monitoring Module

Based on the typical IDS infrastructure described in section 3.2, the Data Monitoring Module operates as an agent, which monitors the network traffic and feeds the DIDEROT Analysis Engine with network flows statistics. In particular, Tshark is used to capture the network traffic data, while CICFlowMeter [8, 24] extracts bidirectional network flow statistics. Based on the placement of the Data Monitoring Module, DIDEROT can operate either as HIDS or NIDS. Moreover, before the transmission of the network flow statistics to the DIDEROT Analysis Engine, the Data Monitoring Module pre-processes them, using the min-max scaler defined by Equation 5.

$$z = \frac{x - min(x)}{max(x) - min(x)} \tag{5}$$

4.2 DIDEROT Analysis Engine

The DIDEROT Analysis Engine is the core component of DIDEROT, which receives the network flow statistics and performs the detection process. It consists of two detection layers, namely a) intrusion detection and b) anomaly detection that operate supplementarily. The first layer adopts multiclass supervised ML detection methods, thus detecting particular DNP3 cyberattacks as described in subsection 3.1: a) Injection, b) Flooding, c) DNP3 Reconnaissance, d) Replay and e) Masquerading. On the other side, the second layer is activated only when the first layer classifies a network flow as normal. It considers both unsupervised/outlier and semi-supervised/novelty ML detection methods. Moreover, for the purpose of the second layer, the DIDEROT Autoencoder was developed. Its analysis is carried out in the following section. Finally, the efficiency of both layers in terms of Accuracy, TPR, FPR and the F1 score is analysed in section 6.

4.3 Response Module

Following the typical IDS architecture, the Response Module receives the outcome of the DIDEROT Analysis Engine and generates security events based on the format of those produced by the AlienVault OpenSource Security Information and Event Management (OSSIM) [41]. Then, the Response Module informs appropriately the SDN controller (i.e., Ryu) [28] in order to drop the malicious/anomalous DNP3 flows. In particular, the Response Module uses the Representational State Transfer (REST) Application Programming Interface (API) of Ryu in order to identify which malicious/anomalous DNP3 network flows will be dropped, by setting empty the action field. The DNP3 network flows are defined via the source and destination IP addresses as well as the source and destination ports.

5 DIDEROT AUTOENCODER

The DIDEROT Autoencoder is a DNN devoted to identifying DNP3 anomalies. As illustrated in Fig. 2, it is composed of six fully connected layers and maps input data $x \in X = R^n$ to an output $x\text{fi} \in X$. In particular, it consists of an encoder $f: X \implies Z$ and a decoder $g: Z \implies X$ which together result in the output $x\text{fi} = g(f(x))$. The low-dimensional latent representation of x is obtained from the encoder and is defined as $z = f(x) \in Z = R^m (m << n)$. As a result of this dimensionality reduction, the DIDEROT Autoencoder avoids to become an identity function, and the training process aims to minimise the reconstruction error $L(x, x\text{fi})$, which is typically the Euclidean distance in space X. Since the proposed autoencoder is trained, anomalies are detected by measuring the reconstruction error $L(x, x\text{fi})$ and comparing it with a threshold T, classifying all operational data samples y with $L(y, g(f(y))) > T$ as anomalies. The selected threshold T is estimated heuristically based on the reconstruction error L of all normal training data samples. In practice the threshold T in order to be more robust is selected to be a large percentile of the reconstruction error $T = p0.9(L(x, x\text{fi})|x \in X)$ or if a validation dataset is available, it is selected to maximise the performance for the validation data. It is noteworthy that the training dataset should only consist of normal observations, and therefore it is expected to be reconstructed well.

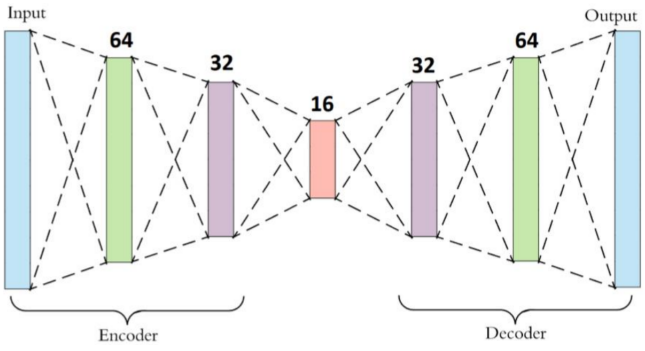

Figure 2: DIDEROT Autoencoder

6 EVALUATION ANALYSIS

This section focuses on the DIDEROT evaluation. First, the evaluation environment is presented. Then, the dataset used to train and test the DIDEROT Analysis Engine is described. Finally, the evaluation results in terms of Accuracy, TPR, FPR and the F1 Score are provided.

6.1 Evaluation Environment

The data used for the DIDEROT evaluation process originates from an emulated substation environment equipped with real RTUs that control the operation of the other Intelligent Electronic Devices (IEDs). A centralised server acts as MTU, including an HMI through which the operator can interact with the various RTUS, using the DNP3 protocol. All assets are connected to an SDN switch. Through SPAN, the entire network traffic generated in the substation environment is destined into MTU, which also hosts DIDEROT. Consequently, the Data Monitoring Module can receive the overall network traffic and extract the respective network flow statistics, thereby feeding the DIDEROT Analysis Engine. Finally, the SDN controller (i.e., Ryu) is located in a different host.

6.2 Dataset

The first detection layer of the DIDEROT Analysis Engine (i.e., intrusion detection) relies on supervised ML detection methods; thus, a labelled dataset composed of both normal and malicious network flow statistics is necessary. The normal records were collected via the normal network traffic generated in the substation environment. On the other side, the dataset created by N. Rodofile et al. [36] was used for the malicious records. The malicious records reflect the aforementioned DNP3 attacks: a) injection, b) flooding, c) DNP3 reconnaissance, d) replay and e) masquerading. The overall dataset was balanced in order to include the same number of normal and malicious records. Moreover, among the malicious records, each DNP3 cyberattack involves the same number of records. On the other side, the second detection layer of the DIDEROT Analysis Engine considers both unsupervised and semi-supervised ML detection methods. Therefore, the training data can include only normal records originating from the substation environment. However, the testing dataset should include both normal and abnormal records. Therefore, again the dataset created by N. Rodofile et al. [36] was used. As in the case of the first detection layer, the testing dataset was balanced, including equal normal and anomalous records.

CICFlowMeter [8, 24] was used to generate the network flow statistics. In particular, both detection layers are fed by the following network flow features.

- **Flow Duration**: Defines the network flow duration in seconds.
- **TotLen Fwd Pkts**: Denotes the entire size of packets to the forward direction.
- **Fwd Pkt Len Mean**: Identifies the average size of the packets to the forward direction.
- **Fwd Pkt Len Mean**: Specifies the average size of the network packets to the forward direction.
- **Bwd Pkt Len Std**: Indicates the standard deviation value of the packets to the backward direction.

- **Flow IAT Std**: Defines the standard deviation time between two packets sent to the forward direction.
- **Bwd Pkts/s**: Denotes the number of the packets transmitted to the backward direction per second.
- **Subflow Bwd Pkts**: Specifies the average number of packets in a subflow to the backward direction.
- **Init Bwd Win Bytes**: Implies the number of bytes sent in an initial window to the backward direction.
- **Active Mean**: Denotes the average time of a network flow, which remained active before becoming idle.

6.3 Evaluation Results

Table 1 summarises the evaluation results of the first detection layer (i.e., intrusion detection). Several supervised ML detection methods were evaluated, including Quadratic Discriminant Analysis, AdaBoost, Multi-Layer Perceptron (MLP), Random forest, SVM (Linear kernel), SVM (RBF kernel), Naive Bayes, Decision Tree, Linear Discriminant Analysis (LDA) and Logistic Regression. Considering all evaluation metrics, the Decision Tree classifier presents the best performance. Fig. 3 illustrates the confusion matrix of the Decision Tree classifier.

Table 1: Evaluation Results of the First DIDEROT Detection Layer - Intrusion Detection

ML Method	Accuracy	TPR	FPR	F1
Quadratic Discriminant Analysis	0.722	0.166	0.166	0.166
AdaBoost	0.798	0.396	0.120	0.396
MLP	0.911	0.733	0.053	0.733
Random Forest	0.931	0.793	0.041	0.793
SVM Linear	0.893	0.680	0.063	0.680
SVM RBF	0.864	0.592	0.081	0.592
Naive Bayes	0.910	0.731	0.053	0.731
Decision Tree	**0.997**	**0.991**	**0.001**	**0.991**
LDA	0.896	0.688	0.062	0.688
Logistic Regression	0.907	0.722	0.055	0.722

Accordingly, Table 2 summarises the evaluation results of the second detection layer (i.e., anomaly detection) of DIDEROT. Many unsupervised/outlier and semisupervised/novelty ML detection methods were evaluated and compared with each other, including a) MCD, b) LOF, PCA, Isolation Forest and DIDEROT Autoencoder. DIDEROT Autoencoder presents the best performance in terms of all evaluation metrics. Fig. 4 shows the confusion matrix of DIDEROT Autoencoder.

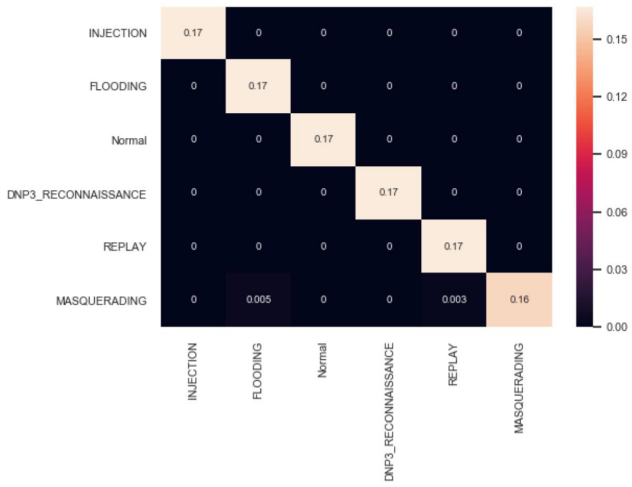

Figure 3: Confusion Matrix of the First DIDEROT Detection Layer (Decision Tree)

Table 2: Evaluation Results of the Second DIDEROT Detection Layer - Anomaly Detection

ML Method	Accuracy	TPR	FPR	F1
MCD	0.946	1	0.107	0.949
LOF	0.942	1	0.114	0.945
PCA	0.5	0	0	0
Isolation Forest	0.950	1	0.098	0.953
DIDEROT Autoencoder	**0.951**	**1**	**0.097**	**0.953**

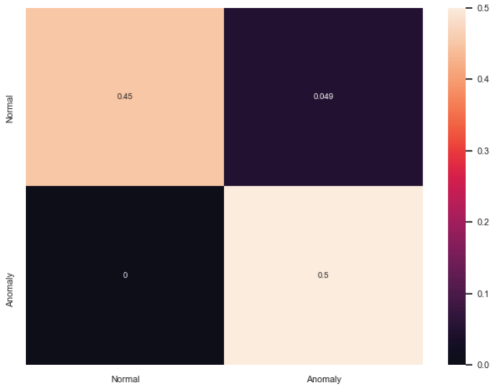

Figure 4: Confusion Matrix of the Second DIDEROT Detection Layer (DIDEROT Autoencoder)

7 CONCLUSIONS

The rapid progression of the Information and Communication Technology (ICT) converts the conventional electrical grid into a new paradigm called SG, offering multiple services in both energy consumers and utility companies, such as the two-way communication, pervasive control, self-healing and in general better use of the existing resources. Despite the fact that this new technological leap leads the energy world into a new, digital reality, it raises serious cybersecurity risks.

In this paper, we provide a DNP3 IDPS called DIDEROT. DIDEROT relies on two detection layers that operate complementarily. The first detection layer relies on a decision tree classifier responsible for recognising specific DNP3 cyberattacks, while the second detection layer uses an autoencoder DNN capable of detecting DNP3

anomalies either due to a potential security violation or an electricity disturbance. The efficiency of DIDEROT is demonstrated using real DNP3 network traffic data originating from an emulated substation environment.

Our future plans related to this work include the creation of appropriate association rules that will combine the two detection layers of DIDEROT in order to identify new DNP3 intrusions/anomalies. Moreover, intrusion detection mechanisms related to other industrial application layer protocols will be investigated, including IEC 61850, IEC 60870-5-104, Profinet and EtherCAT.

ACKNOWLEDGMENTS

The research leading to these results has received funding from the European UnionâĂŹs Horizon 2020 research and innovation programme under grant agreement No 833955.

REFERENCES

[1] Mohammed Ali Al-Garadi, Amr Mohamed, Abdulla Al-Ali, Xiaojiang Du, Ihsan Ali, and Mohsen Guizani. 2020. A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE Communications Surveys & Tutorials* (2020), 1–42.

[2] James P. Anderson. 1980. *Computer security threat monitoring and surveillance.* Technical Report. Box 42 Fort Washington, Pa. 19034 215 646-4706.

[3] Boldizsár Bencsáth, Gábor Pék, Levente Buttyán, and Mark Felegyhazi. 2012. The cousins of stuxnet: Duqu, flame, and gauss. *Future Internet* 4, 4 (2012), 971–1003.

[4] Anna L Buczak and Erhan Guven. 2015. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications surveys & tutorials* 18, 2 (2015), 1153–1176.

[5] Jui-Sheng Chou and Citra S Ongkowijoyo. 2019. Hybrid decision-making method for assessing interdependency and priority of critical infrastructure. *International Journal of Disaster Risk Reduction* 39 (2019), 101134.

[6] Dorothy E Denning. 1987. An intrusion-detection model. *IEEE Transactions on software engineering* SE-13, 2 (1987), 222–232.

[7] Alessandro Di Pinto, Younes Dragoni, and Andrea Carcano. 2018. TRITON: The first ICS cyber attack on safety instrument systems. In *Proc. Black Hat USA*. Black Hat USA, Las Vegas, USA, 1–26.

[8] Gerard Draper-Gil, Arash Habibi Lashkari, Mohammad Saiful Islam Mamun, and Ali A Ghorbani. 2016. Characterization of encrypted and vpn traffic using time-related. In *Proceedings of the 2nd international conference on information systems security and privacy (ICISSP)*. Science and Technology Publications, Italy, 407–414.

[9] Inadyuti Dutt, Samarjeet Borah, and Indra Kanta Maitra. 2020. Immune System Based Intrusion Detection System (IS-IDS): A Proposed. *IEEE Access* 8 (2020), 34929–34941.

[10] Samuel East, Jonathan Butts, Mauricio Papa, and Sujeet Shenoi. 2009. A Taxonomy of Attacks on the DNP3 Protocol. In *International Conference on Critical Infrastructure Protection*. Springer, Springer Berlin Heidelberg, Berlin, Heidelberg, 67–81.

[11] G. Efstathopoulos, P. R. Grammatikis, P. Sarigiannidis, V. Argyriou, A. Sarigiannidis, K. Stamatakis, M. K. Angelopoulos, and S. K. Athanasopoulos. 2019. Operational Data Based Intrusion Detection System for Smart Grid. In *2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*. IEEE, Limassol, Cyprus, 1–6.

[12] Nicolas Falliere, Liam O Murchu, and Eric Chien. 2011. W32. stuxnet dossier. *White paper, Symantec Corp., Security Response* 5, 6 (2011), 29.

[13] Panagiotis I Radoglou Grammatikis, Panagiotis G Sarigiannidis, and Ioannis D Moscholios. 2019. Securing the Internet of Things: Challenges, threats and solutions. *Internet of Things* 5 (2019), 41–70.

[14] Feng Gu, Julie Greensmith, and Uwe Aickelin. 2013. Theoretical formulation and analysis of the deterministic dendritic cell algorithm. *Biosystems* 111, 2 (2013), 127–135.

[15] Yan Hu, An Yang, Hong Li, Yuyan Sun, and Limin Sun. 2018. A survey of intrusion detection on industrial control systems. *International Journal of Distributed Sensor Networks* 14, 8 (2018), 1550147718794615.

[16] Fatima Hussain, Rasheed Hussain, Syed Ali Hassan, and Ekram Hossain. 2020. Machine learning in IoT security: current solutions and future challenges. *IEEE Communications Surveys & Tutorials* (2020), 1–23.

[17] SM Suhail Hussain, Taha Selim Ustun, and Akhtar Kalam. 2019. A review of IEC 62351 security mechanisms for IEC 61850 message exchanges. *IEEE Transactions on Industrial Informatics* 16, 9 (2019), 5643–5654.

[18] Obinna Igbe, Ihab Darwish, and Tarek Saadawi. 2017. Deterministic dendritic cell algorithm application to smart grid cyber-attack detection. In *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*. IEEE, New York, NY, USA, 199–204.

[19] Hamed Jelodar, Yongli Wang, Chi Yuan, Xia Feng, Xiahui Jiang, Yanchao Li, and Liang Zhao. 2019. Latent Dirichlet Allocation (LDA) and Topic modeling: models, applications, a survey. *Multimedia Tools and Applications* 78, 11 (2019), 15169–15211.

[20] Julius Jow, Yang Xiao, and Wenlin Han. 2017. A survey of intrusion detection systems in smart grid. *International Journal of Sensor Networks* 23, 3 (2017), 170–186.

[21] Rafiullah Khan, Peter Maynard, Kieran McLaughlin, David Laverty, and Sakir Sezer. 2016. Threat analysis of blackenergy malware for synchrophasor based real-time control and monitoring in smart grid. In *4th International Symposium for ICS & SCADA Cyber Security Research 2016 4*. ScienceOpen, UK, 53–63.

[22] Sungmoon Kwon, Hyunguk Yoo, and Taeshik Shon. 2020. IEEE 1815.1-Based Power System Security With Bidirectional RNN-Based Network Anomalous Attack Detection for Cyber-Physical System. *IEEE Access* 8 (2020), 77572–77586.

[23] Ralph Langner. 2011. Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy* 9, 3 (2011), 49–51.

[24] Arash Habibi Lashkari, Gerard Draper-Gil, Mohammad Saiful Islam Mamun, and Ali A Ghorbani. 2017. Characterization of Tor Traffic using Time based Features.. In *ICISSP*. Science and Technology Publications, Porto, Portugal, 253–262.

[25] Suchul Lee, Sungho Kim, Sungil Lee, Jaehyuk Choi, Hanjun Yoon, Dohoon Lee, and Jun-Rak Lee. 2016. LARGen: automatic signature generation for Malwares using latent Dirichlet allocation. *IEEE Transactions on Dependable and Secure Computing* 15, 5 (2016), 771–783.

[26] Gaoqi Liang, Steven R Weller, Junhua Zhao, Fengji Luo, and Zhao Yang Dong. 2016. The 2015 ukraine blackout: Implications for false data injection attacks. *IEEE Transactions on Power Systems* 32, 4 (2016), 3317–3318.

[27] Yao Liu, Peng Ning, and Michael K Reiter. 2011. False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security (TISSEC)* 14, 1 (2011), 1–33.

[28] Pedro Manso, José Moura, and Carlos Serrão. 2019. SDN-based intrusion detection system for early detection and mitigation of DDoS attacks. *Information* 10, 3 (2019), 106.

[29] Triangle MicroWorks. 2017. Distributed test manager (dtm).

[30] Nour Moustafa and Jill Slay. 2016. The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Information Security Journal: A Global Perspective* 25, 1-3 (2016), 18–31.

[31] Bharat Rahuldhev Patil, Minal Moharir, Pratik Kumar Mohanty, G Shobha, and S Sajeev. 2017. Ostinato-A Powerful Traffic Generator. In *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*. IEEE, Bangalore, India, 1–5.

[32] Dimitrios Pliatsios, Panagiotis Sarigiannidis, Thomas Lagkas, and Antonios G Sarigiannidis. 2020. A Survey on SCADA Systems: Secure Protocols, Incidents, Threats and Tactics. *IEEE Communications Surveys & Tutorials* (2020), 1–36.

[33] Panagiotis I Radoglou-Grammatikis and Panagiotis G Sarigiannidis. 2019. Securing the smart grid: A comprehensive compilation of intrusion detection and prevention systems. *IEEE Access* 7 (2019), 46595–46620.

[34] Slavica V Boštjančič Rakas, Mirjana D Stojanović, and Jasna D Marković-Petrović. 2020. A Review of Research Work on Network-Based SCADA Intrusion Detection Systems. *IEEE Access* 8 (2020), 93083–93108.

[35] Mubashir Husain Rehmani, Alan Davy, Brendan Jennings, and Chadi Assi. 2019. Software defined networks-based smart grid communication: A comprehensive survey. *IEEE Communications Surveys & Tutorials* 21, 3 (2019), 2637–2670.

[36] Nicholas R Rodofile, Kenneth Radke, and Ernest Foo. 2017. Framework for SCADA cyber-attack dataset creation. In *Proceedings of the Australasian Computer Science Week Multiconference*. Association for Computing Machinery, New York, NY, USA, 1–10.

[37] Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing* 45, 11 (1997), 2673–2681.

[38] Shalvi Srivastava, Sweta Anmulwar, AM Sapkal, Tarun Batra, Anil Kumar Gupta, and Vinodh Kumar. 2014. Comparative study of various traffic generator tools. In *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*. IEEE, Chandigarh, India, 1–6.

[39] Song Tan, Debraj De, Wen-Zhan Song, Junjie Yang, and Sajal K Das. 2017. Survey of security advances in smart grid: A data driven approach. *IEEE Communications Surveys & Tutorials* 19, 1 (2017), 397–422.

[40] Xiao Chun Yin, Zeng Guang Liu, Lewis Nkenyereye, and Bruce Ndibanje. 2019. Toward an Applied Cyber Security Solution in IoT-Based Smart Grids: An Intrusion Detection System Approach. *Sensors* 19, 22 (2019), 4952.

[41] Susana GonzÃąlez Zarzosa. 2017. *D2.1 In-depth analysis of SIEMs extensibility.* Technical Report 1. DiSIEM Project. 1–148 pages.