Cyber Attack Detection and Trust Management Toolkit for Defence-Related Microgrids

Medentzidis Charalampos-Rafail, Kotsiopoulos Thanasis, Vellikis Vasileios, Ioannidis Dimosthenis, Tzovaras Dimitrios, Sarigiannidis Panagiotis

Cyber attack detection and trust management toolkit for defence-related microgrids

Medentzidis Charalampos-Rafail[1], Kotsiopoulos Thanasis[1,2], Vellikis Vasileios[1], Ioannidis Dimosthenis[1], Tzovaras Dimitrios[1], and Sarigiannidis Panagiotis[2]

[1]Information Technologies Institute, Centre for Research & Technology Hellas, 57001 Thermi, Greece
[2]Dept. of Electrical and Computer Engineering, University of Western Macedonia, Karamanli & Ligeris Street, 50100 Kozani, Greece

Abstract. The rise of microgrids in defence applications, as a greener, more economical and efficient source of energy and the consequential softwarization of networks, has led to the emerge of various cyber-threats. The danger of cyber-attacks in defence microgrid facilities cannot be neglected nor undermined, due to the severe consequences that they can cause. To this end, this paper presents a cyberattack detection and cyber attack severity calculation toolkit, with the aim to provide an end-to-end solution to the cyberattack detection in defense IoT/microgrid systems. Concretely, in this paper are presented and evaluated the SPEAR Visual Analytics AI Engine and the SPEAR Grid Trusted Module (GTM) of the SPEAR H2020 project. The aim of the Visual Analytics AI Engine is to detect malicious action that intend to harm the microgrid and to assist the security engineer of an infrastructure to easily detect abnormalities and submit security events accordingly, while the GTM is responsible to calculate the severity of each security event and to assigns trust values to the affected assets of the system. The accurate detection of cyber-attacks and the efficient reputation management, are assessed with data from a real smart home infrastructure with an installed nanogrid, after applying a 3-stage attack against the MODBUS/TCP protocol used by some of the core nanogrid devices.

Keywords: Microgrids · Artificial Intelligence · Fuzzy Logic · cyber-attack detection

1 Introduction

The usage of renewable energy resources instead of the traditional coal-based energy generation systems is gaining more and more popularity in defence applications among other domains. There are numerous reasons for the adoption of renewable energy in this domain, such as the extremely dangerous and costly usage and transportation of fossil fuel in remote regions of military operations that could jeopardize the security of military personnel [15]. Also, the severe environmental footprint of coal usage as the basic means of energy production

is another reason. Finally, relocatable temporary camps usually depend on oversized generators to meet peak loads, even if these loads are very rare, which is both costly and inefficient [10].

Thus, it is understandable that more efficient and secure ways of energy production are needed. One of the basic means of green energy production towards that purpose, is based on microgrid technology. A microgrid is a small-scale local energy cyber-physical system with a controllable group of interconnected loads and distributed energy resources (DERs) that can operate independently both when connected and isolated from the main grid [21], which is one of the main reasons of its usefulness in military applications. In such cyber-physical systems, the physical part is strongly influenced by the integrity of the cyber part [12].

On the one hand, the monitoring and control of flexible assets within a microgrid by means of information and communication technologies enhances the resilience of the system, but on the other hand also incurs the risk of security breaches. The more complex the microgrid is, the more is the reliance on distributed, active control of the network, increasing the potential impact of an attack. In order to minimize that risk, AI technology can be leveraged to train algorithms that can capture patterns of data flows in such systems, that indicate malfunctions or specific cyber-attacks. Secure and Private Smart Grid (SPEAR) H2020 project [1], offers a holistic solution for the protection of Smart Grid infrastructures, including microgrids. The technology developed within SPEAR has been tested and evaluated in a Smart Home infrastructure with an installed nanogrid among other systems. To this end, the contribution of this paper can be summarized in the following points:

- A deep-learning methodology for anomaly detection, combining unsupervised learning to capture the 'normal' operation patterns and an ensemble hard voting to classify a sample as normal/anomaly.
- Visualization of the detected anomalies through the visual analytics dashboard, with the possibility to submit a security event.
- A fuzzy-logic methodology for severity quantification of a security event and consequentially the calculation and assignment of a representative trust value to the affected microgrid asset.
- The evaluation of the proposed solution is implemented in a real nanogrid testbed of a Smart Home.

2 Related Works

Many studies have been investigated regarding the security of microgrid systems in terms of attack detection, node trust assessment or both. In [9] a methodology is proposed based on time-frequency information using the parametric time-frequency logic (PTFL). This technique does not require the modeling knowledge of the microgrid, but the anomalous electricity measurements, called traces are required for anomaly detection related to false data injection (FDI) and denial of service attacks (DoS), as well as physical faults of a microgrid. The same

anomalies are detected by a framework that does not require system knowledge, by monitoring the outputs of inverters/converters against operational bounds, using metric temporal logic (MLT)[8].

An FDI attack detection and mitigation mechanism for the distributed secondary control of AC microgrids is proposed by the authors of [21]. Kullbback-Liebler (KL) divergence is applied to measure the difference between the Gaussian distributions of actual and expected measurements. Trust for individual DERs is represented by the entropy of a DER's own and its neighbour's trustworthiness information on a communication graph. Another deception attack (replay, spoofing, FDI, stealth) detection approach, this time for DC microgrids, is presented in [23], where an analytical consistency-based anomaly detection mechanism is utilized, which manipulates primal and dual variables associated with the proposed distributed algorithm.

In [20] an online monitoring system models the state of the Cyber Physical System (CPS), as a function of its relationships between constituent components, using a combination of model-based and data-driven strategies. The state estimation is done by using the KASE (Kalman Autoregressive State Estimation with Latent Factors and Exogenous Inputs) Invariant Algorithm. The system is periodically retrained using historical data while also updating the CPS state estimation using new data instances. The illiad system has also a front-end section and warns the user in case of anomalies.

The authors in [18] proposed a framework for distributed frequency control and intrusion detection in isolated microgrids. By casting it as a consensus optimization problem, the partial primal-dual algorithm is adopted. For the intrusion problem, two types of malicious network attacks are studied. As a mitigation mechanism, model-based anomaly detection and localization strategies are developed by exploring dual variable-related metrics.

In [16] an intelligent anomaly detection method based on prediction intervals (PIs) is introduced to distinguish malicious attacks with different severities during a secured operation. The proposed anomaly detection method is constructed based on the lower and upper bound estimation method (LUBE) and a modified symbiotic organisms search (SOS) algorithm to provide optimal feasible PIs over the smart meter readings.

The rest of the paper is organized as follows: Section 3 describes the system architecture by presenting its constituent components and the methodologies supporting their functionality. Section 4 discusses the evaluation by presenting the experiments that where conducted to assess the anomaly detection capabilities and the the trust management efficiency. Finally, section 5 includes the conclusions of this paper and the future steps regarding the foreseen extensions of the toolkit.

3 System Architecture

Our toolkit consists of two main components. The Anomaly Detection Module and the Grid Trusted Module (GTM) engine. The Anomaly Detection Module

is part of the SPEAR Visual Analytics component and is responsible to provide visual-based anomaly detection techniques to the security engineer, allowing also to submit security events in the format defined by [2]. GTM engine is responsible to correlate the security events with the impact that a cyberattack can deliver to each asset of the microgrid. Both of the modules are part of the SPEAR SIEM [13].

3.1 Anomaly Detection Module

The purpose of the Anomaly Detection Module is to identify anomalies in the network packets. The network packets are captured via the SPEAR sensor [13] using the wireshark tool [5]. The features used from the wireshark filtering in order to perform the anomaly detection are: packet length, tcp window size, modbus tcp length, modbus tcp prot. id, modbus tcp trans. id, modbus tcp unit id, modbus function code, modbus reference number (8 in total). A brief description of each feature can be found in [6]. The data preprocessing includes the selection of the best features and the resolve of NaN values. The best features are selected by calculating the correlation matrix, we select high correlated features i.e. correlation threshold 0.75. There is a problem arising with NaN values as protocols are changing during time, this problem is resolved by setting 1e-10 to NaN values.

AI Engine An Ensemble hard voting method has been implemented as the core of the AI engine. Ensemble learning fuses the outcome of multiple models and provides prediction with increased accuracy [22]. The ensemble method consists of a seq2seqLSTM autoencoder and three classifiers. The Random Forest classifier, the Logistic Regression classifier and the Gaussian Naive Bayes classifier of the scikit-learn package [4].

First of all, for each feature of the network packets, a seq2seqLSTM autoencoder is created. Figure 1 depicts the seq2seqLSTM autoencoder architecture. The input data is converted to sequences using the sliding window technique. The target data is the input data but offset by one in the future, a process called teacher forcing [17]. The network packet fields from normal conditions are used for training the seq2seq autoencoder, to learn the patterns. The data records used during the prediction phase are unlabeled.

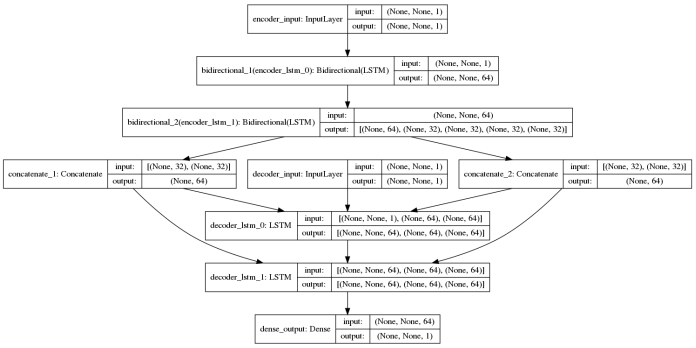

Fig. 1. Seq2SeqLSTM Model

Afterwards, the mean absolute error is calculated between the predicted values and the unlabeled data which was not used for training. The mean absolute error is used as an anomaly score. This technique is common and can be found also in [11, 25]. The seq2seqLSTM autoencoder is built using the Tensorflow library [7] with the Adam optimizer and the mean absolute error as loss function.The anomaly scores for each data source and the ground truth label are then passed as input to the Voting Ensemble Classifier. Each classifier estimates the probability for each sample to belong either to class normal or class anomalous. After each prediction, the output of the hard voting Ensemble method is the label for each sample representing the normal or anomalous category. Figure 2 illustrates the architecture of the Ensemble method.

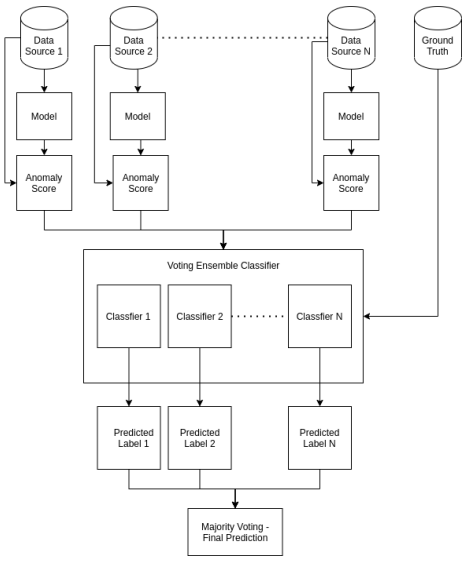

Fig. 2. Architecture of Anomaly Detection Module

3.2 GTM Engine

The goal of GTM is to quantify the severity of the various security events and calculate a reputation value for each asset of the microgrid. This kind of quantification intends to reflect on the one hand the impact of the detected anomaly for each asset and on the other, how trustworthy, safe and secure each asset is. To this end, GTM communicates with the Message Bus to receive the various security events produced by the Anomaly Detection module of the Visual Analytics component, by the BDAC component and by the SPEAR SIEM BASIS. GTM is built using Python and the Fuzzy Logic library called scikit-fuzzy [3]. The utilized fuzzy logic systems described below are based on the Mamdani fuzzy inference approach [19]. The defuzzification of each crisp value is implemented with the centroid method.

Figure 3 depicts the GTM engine architecture. GTM is a backend component and SPEAR VIDS is utilised for the visualization of its outcomes and its configuration, defining a specific threshold value for each asset. If an asset's reputation value or the first derivative of the reputation value drops below the particular thresholds, then a GTM alert is generated for the specific asset(s).All the security events received from the Message Bus, are undertaken by the GTM Fuzzy Logic Core and the GTM Fuzzy Reputation Reduction System, to calculate a reputation value for each asset. These reputation values are sent to the VIDS to visualize them. Finally, the reputation values of GTM are stored into the GTM database as historical data.

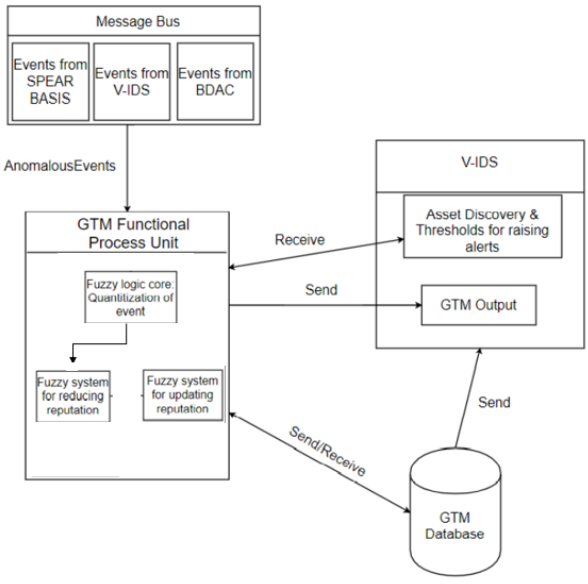

Fig. 3. Architecture of the GTM engine

The core of the GTM engine is the GTM Functional Process Unit, which consists of three elements: (a) the Fuzzy Logic Core, (b) the Fuzzy Reputation Reduction System and (c) the Fuzzy Reputation Recovery System.

The Fuzzy Logic Core quantifies the incoming anomalous incidents using Fuzzy Logic and by taking into consideration first of all the subcategory of the event (Cyber Attack or Anomaly) and the following security event fields: asset value, the event risk, the priority and the reliability. These specific fields are defined by the SPEAR OSSIM [13] which is an extension of [2]. The Fuzzy Logic Core utilises the fuzzy theory to map the value of each aforementioned variable into a quantified value without specifying rules in a strict manner. Table 1 illustrates indicative fuzzy logic rules used by the Fuzzy Logic Core. The rules are derived by forming the fuzzy universe and they are in total 161. The fuzzy universe is unique and mandatory for each variable used to calculate the quantified value of the security event.

Table 1. Indicative Rules of Fuzzy Logic Core

No	Input	Output
Rule 1	asset value: high AND priority: high AND event risk: high AND subcategory: Attack AND reliability: high	quant. value: low
Rule 2	asset value: low AND priority: low AND event risk: low AND subcategory: anomaly AND reliability: low	quant. value: high
Rule 3	asset value: high AND priority: medium AND event risk: high AND subcategory: Attack AND reliability: medium	quant. value: low
Rule 16	asset value: low AND priority: low AND event risk: low AND subcategory: Attack AND reliability: medium	quant. value: medium
Rule 20	asset value: high AND priority: medium AND event risk: low AND subcategory: Attack AND reliability: medium	quant. value: medium

The Fuzzy Reputation Reduction System operates to produce the reputation value of any asset related to the corresponding security event. The reputation value of each asset is computed, taking into account the time difference between the previous reputation reduction value and the current security event as well as the quantified value of the Fuzzy Logic Core. The reputation reduction is applied in this way, since an asset that receives malicious events occasionally and not continuously should not have the same reputation as a node that receives malicious events simultaneously [14]. Table 2 includes the 9 fuzzy logic rules used by the Fuzzy Reputation Reduction System.

Table 2. Rules of Fuzzy Reputation Reduction System

No	Input	Output
Rule 1	time: low AND quant. value: low	reput. value: low
Rule 2	time: low AND quant. value: medium	reput. value: low
Rule 3	time: low AND quant. value: high	reput. value: medium
Rule 4	time: medium AND quant. value: low	reput. value: low
Rule 5	time: medium AND quant. value: medium	reput. value: medium
Rule 6	time: medium AND quant. value: high	reput. value: high
Rule 7	time: high AND quant. value: low	reput. value: medium
Rule 8	time: high AND quant. value: medium	reput. value: high
Rule 9	time: high AND quant. value: high	reput. value: high

Last but not least, the Fuzzy Reputation Recovery System undertakes to increase the reputation value based on the time difference between the last reduction of an asset's reputation value and the current time. The Fuzzy System for reputation recovery works in parallel with the Fuzzy Logic Core and the Fuzzy System for reputation reduction. A time interval threshold is also applied in order to start calculating reputation update for each asset. The threshold is configurable by the user and is based on his/her desire. The functionality of the Fuzzy Reputation Recovery System is also based on fuzzy rules. Table 3 visualizes the Reputation Recovery System fuzzy rules.

Table 3. Rules of Fuzzy Reputation Update System

No	Input	Output
Rule 1	time: low AND quant. value: low	reput. value: medium
Rule 2	time: low AND quant. value: medium	reput. value: medium
Rule 3	time: low AND quant. value: high	reput. value: high
Rule 4	time: medium AND quant. value: low	reput. value: medium
Rule 5	time: medium AND quant. value: medium	reput. value: medium
Rule 6	time: medium AND quant. value: high	reput. value: high
Rule 7	time: high AND quant. value: low	reput. value: medium
Rule 8	time: high AND quant. value: medium	reput. value: high
Rule 9	time: high AND quant. value: high	reput. value: high

4 Evaluation

The evaluation of the toolkit that was described in the previous section, was performed in the Smart Home test bed of CERTH/ITI, where a nanogrid is deployed as described in [24]. The PV inverter and the battery energy storage system inverters/chargers of the nanogrid, support monitoring and control through MODBUS/TCP protocol. An attack to their communication channels could have severe consequences, as it could not only lead to the nanogrid system

corruption, but also to possibly irreversible equipment damage. Due to the importance of preserving the security of those channels, the evaluation was based on a 3-stage cyber-attack against the MODBUS/TCP protocol. The attack was performed against a deployed production honeypot imitating the PV inverter of the Smart Home, so that no real equipment would be endangered. The 3-stage cyber-attack consists of the following steps:

- Uid Brute Force : As a first step a scan of the supported IDs of all the MODBUS clients, is performed.
- Function Enumeration : The second step of the attack enumerates the supported function codes of the target MODBUS device.
- WriteAllRegisters: As a final step, a DoS attempt is performed by arbitrarily writing values in the registers of the target MODBUS device.

After performing the attacks, the toolkit was evaluated by creating a test set with both normal packet fields and packet fields from network traffic related to the aforementioned attack. The classification results of our methodology illustrated in Table 4, showed quite satisfactory performance, indicating the appropriateness of our methodology. The method was able to achieve 83% Accuracy, 80% Precision, 95% Recall and 87% F1 Score.

Table 4. Confusion Matrix of the test set results

		Actual Class	
		Anomalous	Normal
Predicted Class	Anomalous	312	15
	Normal	79	153

The dashboard of the visual analytics tool, assists the security engineer to identify security events by highlighting the detected anomalies in corresponding time series plots, where each time step corresponds to a different packet, as can be seen in Figure . After further investigating the related data, the security engineer can come up with a conclusion about the nature of the security event and submits it by using the form in Figure 4.

The GTM engine was evaluated both in terms of reputation reduction behaviour and scalability. As more security events are produced for a microgrid's asset, the GTM engine reduces the reputation more rapidly as can be seen in Figure 5 a. Specifically, 5 consecutive events were generated and after the second event, the GTM starts to decrease the reputation of the asset with higher rate.

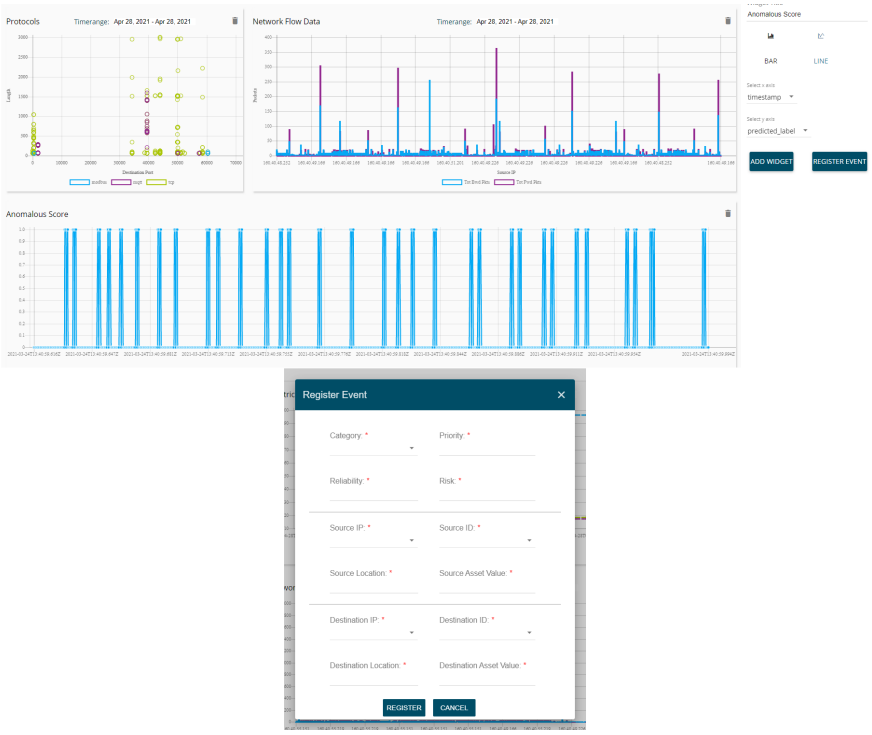

Fig. 4. a. Visualization of the anomaly detection outcome, b.Security event submission form

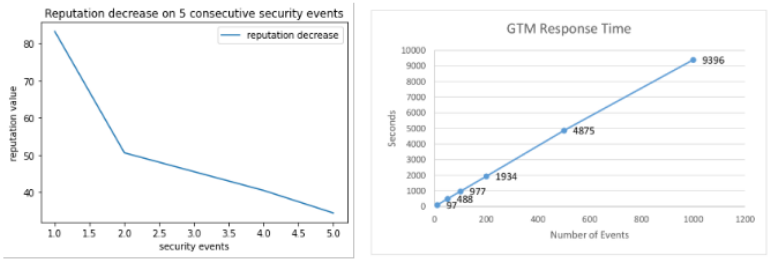

Fig. 5. a.Attacked asset reputation reduction after 5 consecutive events b. GTM response time

In order to assess the scalability of GTM, we submitted several artificial security events to get the response time, which is depicted in Figure 5 b. As can easily be observed from the figure, the response time follows a linear course and

is not affected by the increasing number of submitted security events (10, 50, 100, 200 ,500, 1000 respectively).

5 Conclusions

In this paper an anomaly detection and visualization system tool with trust management functionality was presented and evaluated in a real infrastructure. The AI algorithms used in the Anomaly Detection Module and the three main elements of GTM engine's core, were listed and described. Afterwards the evaluation of the toolkit and the 3-stage cyber attack were documented. The results for the Anomaly Detection Module are very satisfying and the accuracy of the detection system is very high. In further detail,the anomaly detection toolkit achieved 83% Accuracy, 80% Precision, 95% Recall and 87% F1 Score. The GTM engine response time follows a linear course and is not affected by the increasing number of submitted security events. Future steps of the project include the expansion of this toolkit with an addition of a fully developed self-learning feature, making it a fully functional Decision Support System, able to recognize and categorize the nanogrid's anomalies.

6 Acknowledgement

The aforementioned work effort in this paper is conducted under the framework of the SPEAR project, a Horizon 2020 program, funded by the European Union under the grant agreement No. 787011.

References

1. Home page - spear project. https://www.spear2020.eu/, (Accessed on 03/26/2021)
2. Ossim: The open source siem — alienvault. https://cybersecurity.att.com/products/ossim, (Accessed on 03/04/2021)
3. Scikit-fuzzy — skfuzzy v0.2 docs. https://pythonhosted.org/scikit-fuzzy/overview.html, (Accessed on 03/24/2021)
4. scikit-learn: machine learning in python — scikit-learn 0.24.1 documentation. https://scikit-learn.org/stable/, (Accessed on 03/29/2021)
5. wireshark -google. https://www.wireshark.com, (Accessed on 03/29/2021)
6. Wireshark · display filter reference: Modbus. https://www.wireshark.org/docs/dfref/m/modbus.html, (Accessed on 03/29/2021)
7. Abadi, M., Agarwal, A., Barham, P., et al.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015), https://www.tensorflow.org/, software available from tensorflow.org
8. Beg, O.A., Yadav, A.P., Johnson, T.T., Davoudi, A.: Formal online resiliency monitoring in microgrids. In: 2020 Resilience Week (RWS). pp. 99–105 (2020)
9. Beg, O.A., Nguyen, L.V., Johnson, T.T., Davoudi, A.: Cyber-physical anomaly detection in microgrids using time-frequency logic formalism. IEEE Access **9**, 20012–20021 (2021)

10. Berardi, U., Tomassoni, E., Khaled, K.: A smart hybrid energy system grid for energy efficiency in remote areas for the army. Energies **13**(9), 2279 (2020)
11. Borghesi, A., Bartolini, A., Lombardi, M., Milano, M., Benini, L.: Anomaly detection using autoencoders in high performance computing systems. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 9428–9433 (2019)
12. Canaan, B., Colicchio, B., Ould Abdeslam, D.: Microgrid cyber-security: Review and challenges toward resilience. Applied Sciences **10**(16), 5649 (2020)
13. Grammatikis, P.R., Sarigiannidis, P., Iturbe, E., Rios, E., Sarigiannidis, A., Nikolis, O., Ioannidis, D., Machamint, V., Tzifas, M., Giannakoulias, A., et al.: Secure and private smart grid: The spear architecture. In: 2020 6th IEEE Conference on Network Softwarization (NetSoft). pp. 450–456. IEEE (2020)
14. Hadjichristofi, G., Varveris, G.: Visualizing and aggregating behavior for trust evaluation. In: 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD). pp. 1–6. IEEE (2019)
15. Kashem, S.B.A., De Souza, S., Iqbal, A., Ahmed, J.: Microgrid in military applications. In: 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018). pp. 1–5. IEEE (2018)
16. Kavousi-Fard, A., Su, W., Jin, T.: A machine-learning-based cyber attack detection model for wireless sensor networks in microgrids. IEEE Transactions on Industrial Informatics **17**(1), 650–658 (2020)
17. Lamb, A., Goyal, A., Zhang, Y., Zhang, S., Courville, A., Bengio, Y.: Professor forcing: A new algorithm for training recurrent networks. arXiv preprint arXiv:1610.09038 (2016)
18. Lu, L.Y., Liu, H.J., Zhu, H., Chu, C.C.: Intrusion detection in distributed frequency control of isolated microgrids. IEEE Transactions on Smart Grid **10**(6), 6502–6515 (2019)
19. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. International journal of man-machine studies **7**(1), 1–13 (1975)
20. Muralidhar, N., Wang, C., Self, N., Momtazpour, M., Nakayama, K., Sharma, R., Ramakrishnan, N.: illiad: Intelligent invariant and anomaly detection in cyber-physical systems. ACM Transactions on Intelligent Systems and Technology (TIST) **9**(3), 1–20 (2018)
21. Mustafa, A., Poudel, B., Bidram, A., Modares, H.: Detection and mitigation of data manipulation attacks in ac microgrids. IEEE Transactions on Smart Grid **11**(3), 2588–2603 (2019)
22. Opitz, D., Maclin, R.: Popular ensemble methods: An empirical study. Journal of artificial intelligence research **11**, 169–198 (1999)
23. Shi, D., Lin, P., Wang, Y., Chu, C.C., Xu, Y., Wang, P.: Deception attack detection of isolated dc microgrids under consensus-based distributed voltage control architecture. IEEE Journal on Emerging and Selected Topics in Circuits and Systems (2021)
24. Tsolakis, A.C., Bintoudi, A.D., Zyglakis, L., Zikos, S., Timplalexis, C., Bezas, N., Kitsikoudis, K., Ioannidis, D., Tzovaras, D.: Design and real-life deployment of a smart nanogrid: A greek case study. In: 2020 IEEE International Conference on Power and Energy (PECon). pp. 321–326 (2020)
25. Zimmerer, D., Kohl, S.A., Petersen, J., Isensee, F., Maier-Hein, K.H.: Context-encoding variational autoencoder for unsupervised anomaly detection. arXiv preprint arXiv:1812.05941 (2018)