ARTICLE INFORMATION

Article title

A Dataset to Train Intrusion Detection Systems based on Machine Learning Models for Electrical Substations

Authors

Esteban Damián Gutiérrez Mlot* (a)

Jose Saldana (a)

Ricardo J. Rodríguez (b)

Igor Kotsiuba (c)

Carlos H. Gañan (d)

Affiliations

(a) CIRCE Technology Center, Zaragoza, Spain
(b) Aragón Institute for Engineering Research, University of Zaragoza, Zaragoza, Spain
(c) Durham University, UK
(d) Delft University of Technology, Delft, the Netherlands

Corresponding author's email address and Twitter handle

esguti@protonmail.com

Keywords

cybersecurity, critical infrastructure, testbed, IEC61850, IEC60870-5-104, IEC104

Abstract

The growing integration of Information and Communication Technology into Operational Technology environments in electrical substations exposes them to new cybersecurity threats. This paper presents a comprehensive dataset of substation traffic, aimed at improving the training and benchmarking of Intrusion Detection Systems (IDS) installed in these facilities that are based on machine learning techniques. The dataset includes raw network captures and flows from real substations, filtered and anonymized to ensure privacy. It covers the main protocols and standards used in substation environments: IEC61850, IEC104, NTP, and PTP. Additionally, the dataset includes traces obtained during several cyberattacks, which were simulated in a controlled laboratory environment, providing a rich resource for developing and testing machine learning models for cybersecurity applications in substations. A set of complementary tools for dataset creation and preprocessing are also included to

standardize the methodology, ensuring consistency and reproducibility. In summary, the dataset addresses the critical need for high-quality, targeted data for tuning IDS at electrical substations and contributes to the advancement of secure and reliable power distribution networks.

SPECIFICATIONS TABLE

Subject	Artificial Intelligence
Specific subject area	This work focuses on using machine learning to enhance intrusion detection systems for cybersecurity in electrical substations.
Type of data	Network captures: Raw and Processed
Data collection	Data was collected from two real substations in Ukraine and Spain by capturing network traffic using embedded software and tcpdump over a seven-day period. Additionally, cyberattack traces were generated in a controlled lab environment using testbeds simulating attacks such as Denial of Service, packet flooding, fuzzing, and replay. The data was filtered, anonymized, and processed to extract relevant features using scripts, ensuring privacy and consistency for machine learning model training and testing.
Data source location	Data was obtained from: - Real electrical substation located in Iltsi (Ukraine) - Real electrical substation located in Granada (Spain) - Laboratory testbeds located in Zaragoza (Spain). The data is available on Zenodo: https://doi.org/10.5281/zenodo.13898982
Data accessibility	Repository name: Dataset to Train Intrusion Detection Systems based on Machine Learning Models for Electrical Substations Data identification number: 10.5281/zenodo.13898982 Direct URL to data: https://doi.org/10.5281/zenodo.13898982 The data is accompanied by a code repository for processing: https://github.com/esguti/cybersecurity-datasets/
Related research article	

VALUE OF THE DATA

- **Training and Benchmarking ML Models**: Researchers can use the dataset to train machine learning models for tasks such as intrusion and anomaly detection in substation environments. Given the scarcity of publicly available datasets based on real substation traffic [1] [2], this dataset fills a critical gap, providing realistic data that faithfully reflects actual operating conditions. It enables the benchmarking of multiple models, allowing researchers to evaluate and compare their accuracy, reliability, and robustness under the same conditions. This helps develop more effective machine learning algorithms, improving the overall security and resilience of substation systems against cyber threats.
- **Feature Engineering and Algorithm Development**: The dataset provides raw PCAP files (network captures), allowing researchers to perform custom preprocessing and feature extraction. This flexibility supports the development of new algorithms designed to detect specific threats or improve existing detection methods.
- **Standardize the process of files**: The dataset is accompanied by a set of scripts specifically designed to standardize the processing of the files in the dataset. These scripts are available in the repository [3]. This standardization is essential given the notable absence of a documented methodology for processing such files in the existing literature.
- **Extending to Other Critical Infrastructure**: While the dataset primarily focuses on electrical substations, it can be adapted for research in other critical infrastructure scenarios, such as water treatment plants or transportation systems, helping to generalize solutions across sectors.
- **Collaborative Studies and Comparative Analysis**: Researchers can use the dataset to conduct collaborative studies, compare results, and validate findings with other datasets, fostering innovation and improving overall cybersecurity practices.

BACKGROUND

Substations play a fundamental role in the electrical grid. They are responsible for converting electrical voltage to levels suitable for transmission and distribution, manage system protection and interconnection to keep the network grid stable and secure, and support fault isolation and maintenance through sophisticated switching operations. The digitalization of substations, through standards such as IEC61850 [4] and IEC60870-5-104 [5] (also known as IEC104), is essential for communication and automation in electrical substations, but introduces new security problems [3] [4].

Substations are typically organized into three levels: *Station*, *Bay*, and *Process*, connected by the *Station* and *Process bus* (see Figure 1). Each level is explained in more detail below.

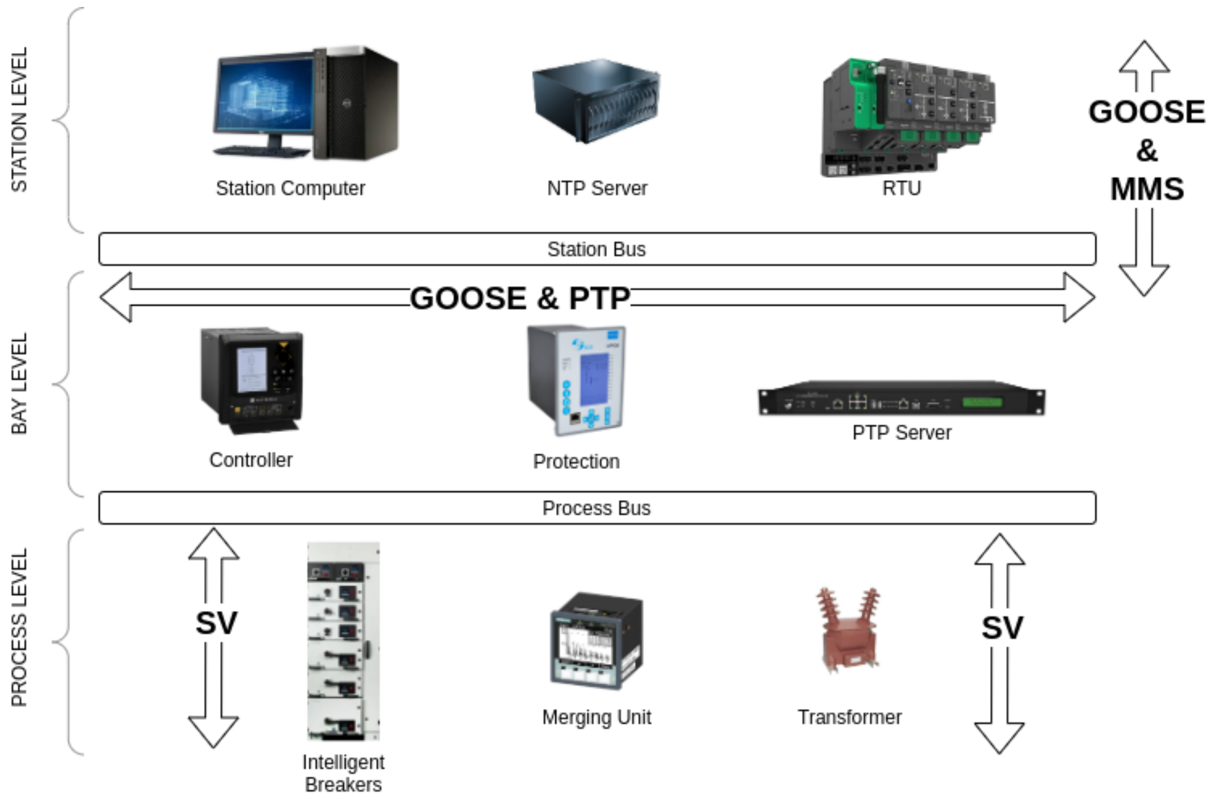


Figure 1 Substation architecture diagram

The **Station Level** is responsible for monitoring, controlling, and communicating with external systems such as control centers and other substations. Typical protocols used at this level are IEC104, Network Time Protocol (NTP), and Precision Time Protocol (PTP). This level typically includes: a Supervisory Control and Data Acquisition (SCADA) system for real-time monitoring and control of the entire substation through a Remote Terminal Unit (RTU); a Human-Machine Interface (HMI) that allows operators to interact with the substation control systems, providing graphical displays of operations and controls; other servers and workstations that host software applications for data processing, visualization, and control; time synchronization servers; and a router to connect to the control center.

The **Bay Level** is responsible for the control and protection of individual sections (or “bays”) of the substation, i.e., transformers, feeders, and busbars. It executes control commands and protection algorithms, and includes the following components: Intelligent Electronic Devices (IEDs), responsible for controlling specific bays; protection relays capable of detecting faults and initiating corresponding protective actions (e.g., tripping a circuit breaker); and control panels and a local HMI, for operation and control of bay equipment.

The **Process Level** directly interacts with the physical electrical equipment. It performs real-time data acquisition from sensors and actuators and sends control commands to the primary equipment (e.g., transformers and circuit breakers). It may include multiple merging units, which digitize the electrical signal and share these measurements via the Sampled Values protocol (defined by IEC61850).

Substation Communication Protocols: IEC61850 and IEC104

IEC61850 is a comprehensive standard designed to modernize substation automation, emphasizing interoperability and open system architectures. It enables seamless integration between devices from different manufacturers and supports real-time communication and data modeling within substations. This standard uses an object-oriented approach to represent each device as a collection of logical nodes, facilitating efficient performance even in complex and large-scale environments. It also includes the definition of several network protocols. In particular: *Manufacturing Message Specification* (MMS), which is used for client-server communication between IEDs and control systems, allowing the exchange of data, control commands, and status information in real time via TCP/IP; *Generic Object Oriented Substation Event* (GOOSE), which is designed to support real-time protection and automation functions and has very strict delay constraints (3 milliseconds in some cases), so it is sent directly over Ethernet. Finally, *Sampled Values* (SV) is used to transmit digitized analog data, such as current and voltage measurements, from merging units to protective relays and other IEDs. Like GOOSE, it is sent over Ethernet.

IEC104 extends the IEC60870-5 standard to include network access via Ethernet, focusing on remote control and monitoring of substations. It is especially useful for telecontrol tasks, using the standard TCP/IP stack to leverage existing network infrastructures.

DATA DESCRIPTION

The core of the dataset consists of network traffic captures and flow files. The content of each file is self-described in its name, which is composed of:

- **file type**: it can be *captured61850* or *captured104*, depending on whether it contains IEC61850 or IEC104 protocol captures;
- **attack**: it can have no attacks (*attackfree*) or a specific attack name (see **Error! Reference source not found.**);
- **function**: optionally, if there are additional details about the captured functionality (*normalfault*) or specific protocol capture (PTP); and
- **file extension**: it can be PCAP (network capture) or CSV (flow file).

Additionally, two file types have been added: one containing all the features found in the CSV files (*headers_[iec104|iec61850]_all.txt*) and another with a selection of relevant features (*headers_[iec104|iec61850].txt*) used in the example described in the section "Illustrative Example". All these files can be found in [8] and are released under the CC BY-NC-SA 4.0 license [9].

Attack	IEC104	IEC61850
DoS	✓	
Packet flooding	✓	✓
Fuzzing	✓	✓
Packet starvation	✓	
NTP DoS	✓	
PTP attack		✓
Port scanning	✓	
PitM	✓	
Replay		✓

Table 1 Attacks included in the testbed traces.

The dataset is accompanied by a set of scripts specifically designed to standardize the processing of dataset files, available in our software repository [3] under the GNU/GPLv3 license [10]. The scripts are organized into two folders:

- **ids**: contains the Python scripts for running the machine learning algorithms to test the datasets.
- **tools**: tools to process the dataset files.

EXPERIMENTAL DESIGN, MATERIALS AND METHODS

The dataset provides operational data collected from two substations. The data obtained from the first substation includes frames corresponding to the IEC104 and NTP protocol. The second substation provided data using IEC61850 standard and PTP. We will call this data “real substation traces” (see section “Real Substation Traces”). In addition, the dataset also contains attack traces. To obtain them, a testbed with specific hardware has been implemented in our laboratory. We will call them “testbed traces” (see section “Testbed Traces”).

Real Substation Traces

These traces were obtained in two real substations. Specifically, the IEC104 data belongs to a facility located in Iltsi (Ukraine) and operated by JSC (“Prykarpattyaoblenergo”) within regional power distribution networks with a capacity of 110/35/10 kV, while the IEC61850 data belongs to a substation placed in Granada (Spain), which houses two 30 MVA transformers operating at 66/20 kV and contains two 20 kV bars with a total of 14 output lines (7 per busbar), supplying electricity to several municipalities. For confidentiality reasons, we cannot disclose internal schematics of the substations.

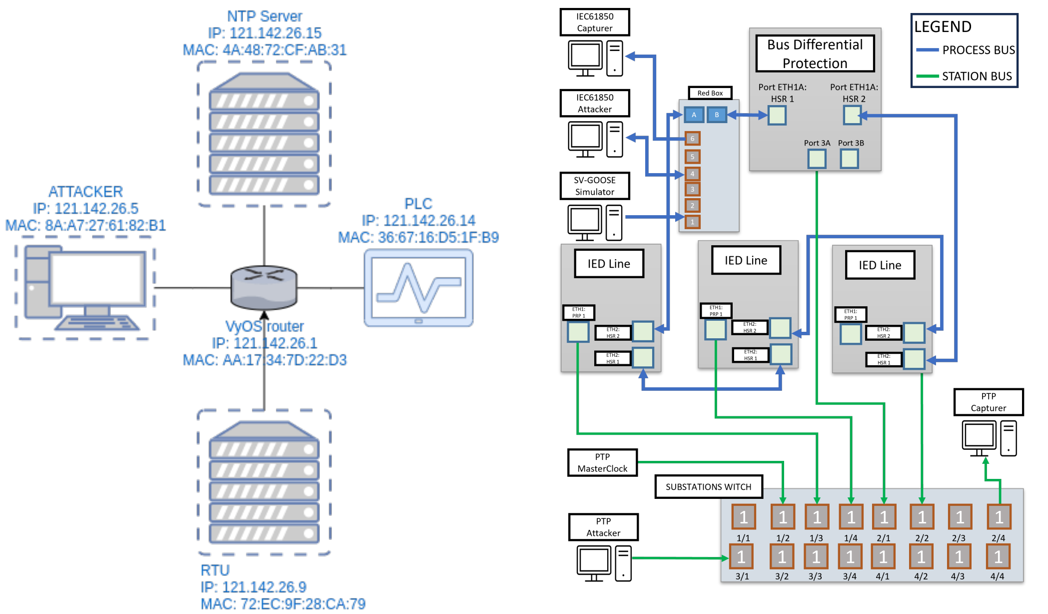
The IEC104 and IEC61850 data captures correspond to a seven-day period, spanning 24 hours each day, within the internal network of the Iltsi (for IEC104) and Granada (for IEC61850) substations. The traffic was filtered to include only IEC104, IEC61850, PTP and NTP protocols. The files were anonymized, and in the case of IEC104, also processed to obtain a listing of the TCP connections. The resulting files are called *flows* and are stored in CSV files.

Testbed Traces

To obtain attack traces, it was necessary to perform attack simulations in a controlled laboratory environment, since conducting these tests in real substations is infeasible due to the critical nature of the infrastructure. In this sense, laboratory simulators provide a safe and controlled environment to test and analyze the effects of various cyberattack scenarios, avoiding any real-world consequences. The attack traces have been obtained using two specifically prepared test environments: the IEC104 and IEC61850 testbeds.

The IEC104 testbed (detailed in Figure 2a) consists of five virtual machines: two of them simulate specific industrial devices (specifically, an RTU and a Programmable Logic Controller or PLC), while the remaining ones correspond to the networking infrastructure: an NTP server and a VyOS [7] router, and finally, a machine controlled by the attacker. All components are connected to the same local network.

The IEC61850 testbed (in Figure 2b) consists of two virtual machines (one controlled by the attacker and a GOOSE/SV simulator), two embedded devices (a GOOSE/SV capturer and a PTP capturer), and four IEDs. These devices are interconnected through two different networks. The first one is dedicated to the transmission of power grid control packets, including GOOSE, SV, and MMS protocols, while the second one carries PTP messages for time synchronization purposes. The IEDs protect the substation equipment against overcurrent faults. They monitor SV frames, which carry samples of electrical signals, for anomalies indicative of failure. Initially, the system operates for about 3000 milliseconds without faults, followed by a “line to ground” fault (known as an AG fault) which triggers the protection mechanism and opens the line. This scenario is then repeated under the condition of a cyberattack to observe the impact on the protection process.



(a) Substation model for IEC104 testbed (b) Substation model for IEC61850 testbed

Figure 2 Testbeds used to generate attack traces.

Error! Reference source not found. summarizes the attacks included in this dataset, specifying the testbed where they were generated. Each of them is stored in a separate file for easy labeling.

DoS refers to a DoS attack against the PLC (IEC104 testbed), where numerous TCP SYN packets are sent skipping the subsequent SYN+ACK response. The **packet flooding** attack in the IEC61850 dataset floods the Bus Differential Protection (BDP) with packets, thereby inducing a fault within the substation electrical network and disrupting the flow of electricity. In the IEC104 dataset, it floods the RTU with messages from the PLC. In the **fuzzing** attack, random commands are sent to cause failures in the RTU (IEC104 dataset) or the BDP (IEC61850 dataset). During the **packet starvation** attack, the RTU is overwhelmed with connections until it stops responding. Similarly, **NTP DoS** also involves attacking the NTP server to disrupt the operation of the service. In the **PTP attack**, a new time source is introduced into the network, which disrupts the master clock and messes up the time settings. The **Port scanning** attack involves reconnaissance attack on the PLC, RTU, NTP server, and VyOS router (IEC104 dataset). In the **PitM** attack (IEC104 dataset), ARP poisoning is conducted to isolate and drop traffic between the RTU and the PLC. Finally, the **Replay** attack tricks an IED into failing based on a repeated (*replayed*) packet, leading to operational issues such as opening an electrical circuit breaker at an unexpected time.

Preprocessing

The PCAP files available in the dataset are appropriately filtered and anonymized to prevent the disclosure of sensitive information such as topology or equipment models, which could be used to attack the critical infrastructure used for the creation of the dataset. This process is followed by a feature extraction process, during which CSV files are generated.

Filtering was performed using *tshark* [8]. Due to issues with handling large files, we first split the files into 10GB chunks, which were then merged after preprocessing. Splitting and filtering were performed using the *filter_and_split.sh* script, and subsequent merging was performed using the *merge_pcap.sh* script. Both scripts are available in our software repository [3]. After this, the anonymization process is performed using the script *anonymize.sh*, which is based on *Sanicap* [9].

The final stage in the preprocessing process is feature extraction. Below, we provide an illustrative example of feature selection and extraction. Additionally, our dataset provides the original PCAP files to allow users to perform their custom feature processing.

The IEC104 protocol operates on top of the transport layer (specifically, over TCP/IP protocol), unlike the IEC61850 protocol that operates on top of the link layer. This disparity requires the use of distinct features for training algorithms. To extract TCP/IP flows relevant to IEC104, we have used the *CICFlowMeter* [10] tool. Additionally, *tshark* was used to extract crucial features from IEC61850 frames. Our dataset provides scripts for feature extraction in each protocol: *generatecsv_iec104.sh* and *generatecsv_iec61850.sh*. A final step in the feature extraction process is labeling: an additional column, called “Label”, is appended to each CSV file and stores the attack type, or lack thereof, which is derived from the file name.

Illustrative Example

An example of usage is provided in the Python script *pycaret_ids.py*, created to facilitate the execution and comparison of various machine learning algorithms, specifically those used for classification tasks.

In particular, this script leverages the PyCaret [11] library, an open-source tool that simplifies and automates the process of developing machine learning models.

The script reads all the CSV files from the dataset, using the “Label” column to categorize the data, removes invalid values, and runs several classification models to compare them. Finally, it stores the model with the best results found for future predictions.

We have employed a variety of machine learning models for our analysis, covering multiple algorithmic categories: *Linear Models* (Logistic Regression and Ridge Classifier), *Nearest Neighbors* (K Neighbors Classifier*), Support Vector Machines* (Linear Support Vector Machine), *Decision Trees and Ensembles* (Decision Tree Classifier, Random Forest Classifier, Extra Trees Classifier, Gradient Boosting Classifier, Light Gradient Boosting Machine and Extreme Gradient Boosting), *Naive Bayes* (Naive Bayes Classifier), *Discriminant Analysis* (Linear Discriminant Analysis and Quadratic Discriminant Analysis) and *Dummy Classifier* (just for benchmarking). This selection allowed us to explore a wide range of approaches to identify the most effective model for each anomaly detection task.

The Area Under the ROC Curve (AUC) is often recommended for comparing models [12], particularly with imbalanced datasets, as it provides a balanced view of performance across all thresholds. F1-Score (F1) is also very valuable in such scenarios, as it balances the importance of Precision (Prec.) and Recall. Furthermore, the Matthews’s Correlation Coefficient (MCC) is beneficial for a comprehensive evaluation of classifiers, considering all aspects of the confusion matrix. Using these three metrics, we can conclude that the Linear Discriminant Analysis model performs better than the rest of the models. The table also shows the Accuracy, the Cohen’s kappa coefficient (κ), and the Training Time (in seconds; TT).

We ran this script on subsets of our dataset to show how it facilitates model comparison. We have employed zscore normalization and StratifiedKFold validation, with a 70% partition for the training data. These experiments were run on a machine with two Intel Xeon Gold @2.20GHz and 128GB of RAM. For IEC104, all available traces have been used to detect the attacks described in **Error! Reference source not found.** (multiclass classification). For IEC61850, a single attack (binary classification) has been carried out to illustrate another type of classification. More details and additional examples can be found in [8].

Error! Reference source not found. provides the results for the IEC104 data. The results indicate that classifier models such as Extra Trees and Random Forest achieve an excellent balance between predictive performance and training time, positioning them as the most suitable for real-world applications in this context. In particular, the Extra Trees classifier exhibited the highest accuracy (0.8217) and competitive results in AUC (0.8297), with a moderate training time of 2.620 seconds. Similarly, Random Forest performed well in both AUC (0.9127) and F1-score (0.8059), while maintaining a relatively short training time (1.989 s), making it a strong candidate for practical deployment.

Likewise, Table 3 illustrates the detection of fuzzy attacks on the IEC61850 dataset. LightGBM and Extreme Gradient Boosting offer the best predictive performance, although they incur higher computational costs. Linear Discriminant Analysis offers a solid balance between performance and efficiency, making it a good choice in situations where fast training is essential. Models such as Ridge

Classifier and SVM underperform, while simple models such as Naive Bayes and K-Neighbors are also viable alternatives in this context.

Model	Accuracy	AUC	Recall	Prec.	F1	κ	MCC	TT (s)
Dummy Classifier	**0.8592**	0.5000	**0.8592**	0.7383	0.7942	0.0000	0.0000	**0.4640**
Ridge Classifier	0.8586	0.0000	0.8586	0.7879	0.8148	0.1714	0.2154	0.6540
Logistic Regression	0.8584	0.9454	0.8584	0.7978	0.8217	0.2253	0.2572	7.2600
SVM - Linear Kernel	0.8566	0.0000	0.8566	0.8222	0.8345	0.3263	0.3390	2.1980
Linear Discriminant Analysis	0.8566	0.9286	0.8566	0.8532	**0.8546**	**0.4264**	**0.4266**	1.4800
Gradient Boosting Classifier	0.8551	**0.9506**	0.8551	0.7979	0.8217	0.2339	0.2588	76.4960
Light Gradient Boosting Machine	0.8482	0.9370	0.8482	0.7934	0.8170	0.2207	0.2394	1400.
Extreme Gradient Boosting	0.8419	0.9484	0.8419	0.7943	0.8167	0.2394	0.2494	4.0510
Naive Bayes	0.8409	0.8314	0.8409	0.8198	0.8126	0.2668	0.2809	0.6700
K Neighbors Classifier	0.8292	0.8785	0.8292	0.7920	0.8094	0.2147	0.2200	7.7830
Extra Trees Classifier	0.8247	0.8297	0.8247	0.7730	0.7964	0.1377	0.1458	2.6200
Decision Tree Classifier	0.8245	0.8238	0.8245	0.7682	0.7941	0.1267	0.1351	0.7070
Random Forest Classifier	0.8245	0.9127	0.8245	0.7888	0.8059	0.2090	0.2128	1.9890
Quadratic Discriminant Analysis	0.6505	0.8668	0.6505	**0.8770**	0.7329	0.1895	0.2299	1.1370

Table 2 Comparison of different machine learning models evaluating IEC104 on our dataset. The best results for each metric have been highlighted in bold with an orange background.

Model	Accuracy	AUC	Recall	Prec.	F1	κ	MCC	TT (s)
Dummy Classifier	0.8768	0.5000	0.8768	0.7688	0.8192	0.0000	0.0000	6.9830
Ridge Classifier	0.8766	0.0000	0.8766	0.8540	0.8515	0.2334	0.2521	**6.3390**
Logistic Regression	0.8768	0.7111	0.8768	0.8618	0.8670	0.3442	0.3522	8.0220
SVM - Linear Kernel	0.8767	0.0000	0.8767	0.8078	0.8307	0.0857	0.0866	6.9760
Linear Discriminant Analysis	**0.8768**	0.7152	**0.8768**	**0.8768**	**0.8768**	**0.4297**	**0.4297**	7.5940
Gradient Boosting Classifier	0.8765	0.7424	0.8765	0.8470	0.8517	0.2247	0.2554	80.1890
Light Gradient Boosting Machine	0.8764	**0.7435**	0.8764	0.8430	0.8458	0.1822	0.2201	186.9080
Extreme Gradient Boosting	0.8761	0.7427	0.8761	0.8512	0.8577	0.2709	0.2904	13.3200
Naive Bayes	0.8761	0.7134	0.8761	0.8765	0.8763	0.4281	0.4282	7.0930
K Neighbors Classifier	0.8742	0.6968	0.8742	0.8470	0.8539	0.2473	0.2685	130.1370
Extra Trees Classifier	0.8758	0.7412	0.8758	0.8509	0.8576	0.2708	0.2898	68.6430
Decision Tree Classifier	0.8757	0.7411	0.8757	0.8509	0.8576	0.2708	0.2898	8.4140
Random Forest Classifier	0.8758	0.7414	0.8758	0.8506	0.8572	0.2680	0.2876	103.6080
Quadratic Discriminant Analysis	0.8761	0.7130	0.8761	0.8764	0.8763	0.4280	0.4280	7.4910

Table 3 Comparison of different machine learning models evaluating IEC61850 on our dataset. The best results for each metric have been highlighted in bold with an orange background.

LIMITATIONS

None

ETHICS STATEMENT

The authors have read and follow the ethical requirements for publication in Data in Brief and confirming that the current work does not involve human subjects, animal experiments, or any data collected from social media platforms.

CRediT AUTHOR STATEMENT

Esteban Gutiérrez: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, **Jose Saldana**: Supervision, Writing - Review & Editing **Ricardo J. Rodríguez**: Supervision, Writing - Review & Editing **Igor Kotsiuba**: Writing - Review & Editing **Carlos H. Gañan**: Writing - Review & Editing.

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to Volodymyr Shcherbiak from JSC ("Prykarpattiaoblenergo"), for granting us access to Iltsi substation and for his invaluable support throughout this research, and CUERVA Energy for their support capturing data from the Granada substation.

Funding

The research of E. D. Gutiérrez and J. Saldana has been supported by the European Union's Horizon Europe Energy Research and Innovation programme eFORT, Grant Agreement No 101075665. The research of R. J. Rodríguez was supported in part by TED2021-131115A-I00 (MIMFA), funded by MCIN/AEI/10.13039/501100011033, by the Recovery, Transformation and Resilience Plan funds, financed by the European Union (Next Generation), by the Spanish Ministry of Universities, by the Spanish National Cybersecurity Institute (INCIBE) under "*Proyecto Estratégico CIBERSEGURIDAD EINA UNIZAR*", and by the University, Industry and Innovation Department of the Aragonese Government under "*Programa de Proyectos Estratégicos de Grupos de Investigación*" (DisCo research group, ref. T21-23R). The research of C. H. Gañán by the RAPID project (Grant No. CS.007) financed by the Dutch Research Council (NWO).

DECLARATION OF COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] S. A. Gutierrez, J. F. Botero, N. G. Gomez, L. A. Fletscher y A. Leal, «Next-Generation Power Substation Communication Networks: IEC 61850 Meets Programmable Networks,» *IEEE Power and Energy Magazine,* vol. 21, p. 58–67, September 2023.

[2] S. E. Quincozes, C. Albuquerque, D. Passos y D. Mosse, «A Survey on Intrusion Detection and Prevention Systems in Digital Substations,» *Computer Networks,* vol. 184, p. 107679, 2021.

[3] E. D. Gutierrez Mlot, «cybersecurity-datasets,» 2024. [En línea]. Available: https://github.com/esguti/cybersecurity-datasets/. [Último acceso: 11 11 2024].

[4] IEC, «IEC 61850,» 2013. [En línea]. Available: https://iec61850.dvl.iec.ch/. [Último acceso: 12 05 2024].

[5] IEC, «IEC 60870-5-104,» 2004. [En línea]. Available: https://webstore.iec.ch/publication/25035. [Último acceso: 12 05 2024].

[6] A. Akbarzadeh, L. Erdodi, S. H. Houmb, T. G. Soltvedt y H. K. Muggerud, «Attacking IEC 61850 Substations by Targeting the PTP Protocol,» *Electronics,* vol. 12, p. 2596, 2023.

[7] A. Baiocco y S. D. Wolthusen, «Indirect Synchronisation Vulnerabilities in the IEC 60870-5-104 Standard,» de *2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Sarajevo, 2018.

[8] E. D. Gutierrez Mlot, «Dataset to Train Intrusion Detection Systems based on Machine Learning Models for Electrical Substations,» 2024. [En línea]. Available: https://doi.org/10.5281/zenodo.13898982. [Último acceso: 11 11 2024].

[9] Creative Commons, «Attribution-NonCommercial-ShareAlike 4.0 International,» 2024. [En línea]. Available: https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode. [Último acceso: 2024 11 12].

[10] Free Software Foundation, «GNU General Public License,» 2007. [En línea]. Available: https://www.gnu.org/licenses/gpl-3.0-standalone.html. [Último acceso: 12 11 2024].

[11] VyOS, «VyOS - Open source router and firewall platform,» 2013. [En línea]. Available: https://vyos.io/. [Último acceso: 05 07 2024].

[12] Wireshark, «Wireshark,» 1998. [En línea]. Available: https://www.wireshark.org/. [Último acceso: 18 06 2024].

[13] thepacketgeek, «Sanicap,» 18 06 2014. [En línea]. Available: https://github.com/thepacketgeek/sanicap.

[14] C. I. for Cybersecurity, «CICFlowMeter - Applications,» 19 06 2018. [En línea]. Available: https://www.unb.ca/cic/research/applications.html.

[15] M. Ali, «PyCaret: An open source, low-code machine learning library in Python,» 2020. [En línea]. Available: https://www.pycaret.org/. [Último acceso: 15 07 2024].

[16] J. Huang y C. X. Ling, «Using AUC and accuracy in evaluating learning algorithms,» *IEEE Transactions on Knowledge and Data Engineering,* vol. 17, p. 299–310, March 2005.