



ORIGINAL RESEARCH

Oversampling and undersampling for intrusion detection system in the supervisory control and data acquisition IEC 60870-5-104

M. Agus Syamsul Arifin^{1,2}  | Deris Stiawan³  | Bhakti Yudho Suprpto¹ |
Susanto Susanto² | Tasmu Salim^{1,4} | Mohd Yazid Idris^{5,6} | Rahmat Budiarto⁷

¹Departement of Computer systems engineering, Faculty of Engineering, Universitas Sriwijaya, Palembang, Indonesia

²Departement of Informatics Engineering, Faculty of Engineering, Universitas Bina Insan, Lubuklinggau, Indonesia

³Departement of Computer System, Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia

⁴Departement of Computer System, Faculty of Computer Science, Universitas Indo Global Mandiri, Palembang, Indonesia

⁵Departement of Software Engineering, Faculty of Computing, Universiti Teknologi Malaysia, Johor, Malaysia

⁶Media and Game Centre of Excellence (MaGICX), Institute of Human Centred Engineering (iHumEn), Universiti Teknologi Malaysia, Johor Bahru, Johor, Malaysia

⁷Departement of Computer Science, College of Computing and Information, Al-Baha University, Albahah City, Albahah, Saudi Arabia

Correspondence

Deris Stiawan, Departement of Computer System, Faculty of Computer Science, Universitas Sriwijaya, Palembang 30119, Indonesia.
Email: deris@unsri.ac.id

Abstract

Supervisory control and data acquisition systems are critical in Industry 4.0 for controlling and monitoring industrial processes. However, these systems are vulnerable to various attacks, and therefore, intelligent and robust intrusion detection systems as security tools are necessary for ensuring security. Machine learning-based intrusion detection systems require datasets with balanced class distribution, but in practice, imbalanced class distribution is unavoidable. A dataset created by running a supervisory control and data acquisition IEC 60870-5-104 (IEC 104) protocol on a testbed network is presented. The dataset includes normal and attacks traffic data such as port scan, brute force, and Denial of service attacks. Various types of Denial of service attacks are generated to create a robust and specific dataset for training the intrusion detection system model. Three popular techniques for handling class imbalance, that is, random over-sampling, random under-sampling, and synthetic minority oversampling, are implemented to select the best dataset for the experiment. Gradient boosting, decision tree, and random forest algorithms are used as classifiers for the intrusion detection system models. Experimental results indicate that the intrusion detection system model using decision tree and random forest classifiers using random under-sampling achieved the highest accuracy of 99.05%. The intrusion detection system model's performance is verified using various metrics such as recall, precision, F1-Score, receiver operating characteristics curves, and area under the curve. Additionally, 10-fold cross-validation shows no indication of overfitting in the created intrusion detection system model.

KEYWORDS

computer network security, power distribution control, power system security

1 | INTRODUCTION

Supervisory control and data acquisition (SCADA) systems have come to be a fundamental factor in Industry 4.0 in monitoring and controlling industrial processes due to their capability in industrial automation processes that allows visualisation of plant production processes and translates into better decisions based on relevant information. At the same time, security becomes a crucial challenge in SCADA systems/networks as the systems are vulnerable to various threats and

attacks. Thus, SCADA systems require intelligent and robust intrusion detection systems (IDSs) as security tools.

The development of machine learning-based IDS requires ideal datasets in terms of size and balance distribution of classes to achieve high accuracy [1]. Classification in machine learning works as a training system using labelled datasets to identify new unseen or unknown patterns of attacks in the dataset. Thus, the imbalanced distribution of classes in the dataset becomes the main challenge [2] and an important consideration [3]. Various traditional machine learning methods assume that the target class

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs](https://creativecommons.org/licenses/by-nc-nd/4.0/) License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2024 The Authors. *IET Cyber-Physical Systems: Theory & Applications* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

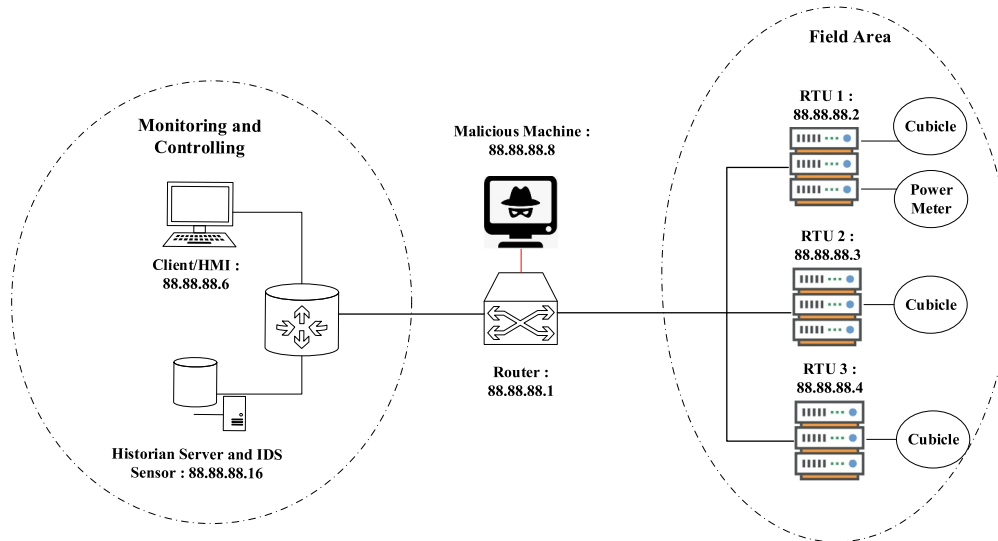


FIGURE 1 The SCADA IEC 60870-5-104 testbed topology.

TABLE 1 The instructions used in the normal scenario.

Instruction	Description	CoT
C_SC_NA_1	Single command	6
C_IC_NA_1	Setpoint command, normalised value	6
M_ME_NA_1	Measured value, normalised value	3
M_SP_NA_1	Single point information	3

has the same number of distributions as other classes in the dataset [3]. Unfortunately, in practice, class imbalance in the dataset is unavoidable, that is, the number of target classes is very small compared to the normal class.

This paper creates a dataset by running a SCADA testbed network running the SCADA IEC 60870-5-104 (IEC 104) protocol. The captured traffic data consists of the Normal traffic data as the majority of data, and the attack traffic data includes port scan, brute force, and Denial of service (DoS). Different types of DoS attacks, that is, internet control message protocol (ICMP) flood, Syn flood, Xmas and IEC 104 flood are generated to produce a more specific and robust dataset to be used for training the best IDS model. The IEC 104 flood is a flood attack by sending massive application service data unit (ASDU) data packets to drop communication between remote terminal unit (RTU) and human-machine interface (HMI).

Then, three popular techniques for handling the class imbalance in the dataset, that is, random over-sampling technique (ROS), random under-sampling technique (RUS) and synthetic minority oversampling technique (SMOTE) are implemented, and the best dataset will be selected for the experiment. The gradient boosting (GrB), decision tree (DT), and random forest (RF) algorithms are used as the machine learning-based classifier for the proposed IDS models. Then, we measure the performance of the IDS models in terms of accuracy, precision, recall, F-measure (F1-Score), receiver

operating characteristics (ROC) curves and area under the curve (AUC). Finally, the 10-fold cross-validation technique is used to validate that the model is not affected by overfitting and trust the accuracy results of the resulting IDS model. This paper poses two main contributions as follows.

- Creation of the SCADA network traffic dataset on the IEC 60870-5-104 protocol by using a testbed with physical devices to simulate real-world conditions within a SCADA network system.
- Comparing optimal techniques for addressing dataset imbalances to develop a robust IDS model for SCADA network systems.

This paper is structured as follows. The related work is provided in Section 2. The methodology of this research is described in Section 3. The experimental results along with the analysis are discussed in Sections 4, and Section 5 provides the conclusion of this research.

2 | RELATED WORK

The machine learning-based IDS models are facing the problem of high false alarms and low detection rates. To improve the performance of IDS on imbalanced datasets, oversampling [4] and undersampling [5] techniques are commonly applied.

The ROS is a technique of taking random samples by replacing the minority class, thus increasing the amount of data in the dataset [5]; this technique is the data-centric type [6]. The RUS technique is a random sampling technique to balance the desired class by eliminating instances of the majority class [7, 8]. While SMOTE is a technique of oversampling to make each class in the dataset balanced for each class by synthesising new samples from the minority class and re-sampling the minority class [9, 10]. SMOTE is an oversampling technique derived from ROS [11].

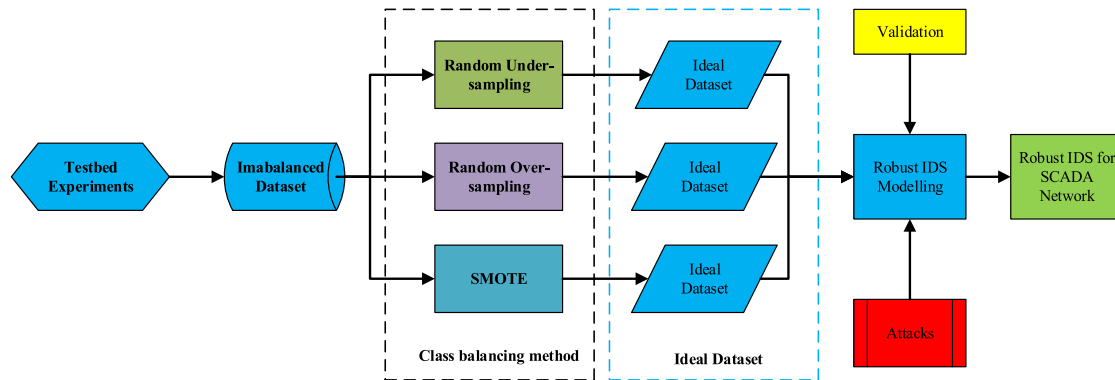


FIGURE 2 The proposed method to find the best model of robust IDS for the SCADA network.

```

7/26-10:04:44.986083 [**] [1:9000002:1] Ack and RST detected, potential DoS/Scan [**]
Classification: Potentially Bad Traffic [Priority: 2] (TCP) 88.88.88.3:8767 -> 88.88.88.8:50965
7/26-2022-10:04:44.986083 [**] [1:9000002:1] Ack and RST detected, potential DoS/Scan [**]
Classification: Potentially Bad Traffic [Priority: 2] (TCP) 88.88.88.3:8767 -> 88.88.88.8:50965
Frame 55029: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
Encapsulation type: Ethernet (1)
Arrival Time: Jul 26, 2022 10:04:44.986083000 WIB
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1658884684.986083000 seconds
[Time delta from previous captured frame: 0.000035000 seconds]
[Time delta from previous displayed frame: 0.000035000 seconds]
[Time since reference or first frame: 3833.849098000 seconds]
Frame Number: 55029
Frame Length: 60 bytes (480 bits)
Capture Length: 60 bytes (480 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ethertype:ip:tcp]
[Coloring Rule Name: TCP RST]
[Coloring Rule String: tcp.flags.reset eq 1]
Ethernet II, Src: Espress1_06:a2:f8 (7c:9e:bd:06:a2:f8), Dst: PcsCompu_c5:30:a1 (08:00:27:c5:30:a1)
Transmission Control Protocol, Src Port: 15688, Dst Port: 50965, Seq: 1, Ack: 1, Len: 0
Source Port: 15688
Destination Port: 50965
[Stream Index: 2501]
[TCP Segment Len: 0]
Sequence number: 1 (relative sequence number)
Sequence number (raw): 0
[Next sequence number: 1 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
Acknowledgment number (raw): 818586226
6191 -> -> Header Length: 20 bytes (5)
* Flags: 0x014 (RST, ACK)
Window size value: 5744
[Calculated window size: 5744]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x6110 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
* [SEQ/ACK analysis]
* [Timestamps]
6 Jul 26, 2022 10:04:44.986083000 WIB 14 88.88.88.3 88.88.88.8 15688 50965 60

```

FIGURE 3 Correlation alert of Snort and Suricata with data extraction for port scan.

The research by Gupta et al. [6] used ROS and long short-term memory (LSTM) to improve IDS performance in detecting malicious activities in network traffic with reduced false alarm rates. The ROS technique may cause overfitting in the resulted IDS model [12, 13]; thus, various validation methods are required to prove that the implementation of the ROS algorithm to overcome the imbalanced dataset problem does not cause overfitting problems in the resulted IDS model. The RUS was used in Ref. [14] to get a better class balance for a network intrusion detection system (NIDS) in detecting attacks using a wavelet neural network model.

The research conducted by Qaddoura et al. [15] used SMOTE to overcome the problem of class imbalance in building IDS for security on the Internet of Things (IoT) network. The use of SMOTE was implemented by Al and Dener [12] to reduce the effect of imbalanced data on the performance of IDS built-in big data network environments. Enchanted RF and SMOTE are used by T. Wu et al. [16] to improve IDS performance in detecting malicious activity attacks on various data sources in computer networks that decrease the false alarm rate while increasing the accuracy.

Validation for the performance results of each created IDS model needs to be done to trust the accuracy of the model,

since the oversampling and undersampling process on the dataset and the training data risks increase the possibility of overfitting the created IDS model.

3 | METHODOLOGY

3.1 | Scenario and testbed network topology

This study discusses the malicious activity on the SCADA system running the IEC 60870-5-104 (IEC 104) protocol. The protocol is being chosen because it is widely used in the power plant industry to monitor and control distribution lines [17, 18]. The IEC 104 protocol has become popular in the power plant industry because it supports automation generation control (ACG) [19], and it is a Transmission Control Protocol/Internet Protocol (TCP/IP)-based modification of the IEC 101 standard for power system monitoring and telecontrol [20].

Referring to Wang and Foo [21], to produce a realistic dataset from a SCADA system testbed, four elements must be included, that is, input, controller, output and network. Therefore, a SCADA EIC104 system testbed was set up accordingly. The testbed uses the standard instructions of the

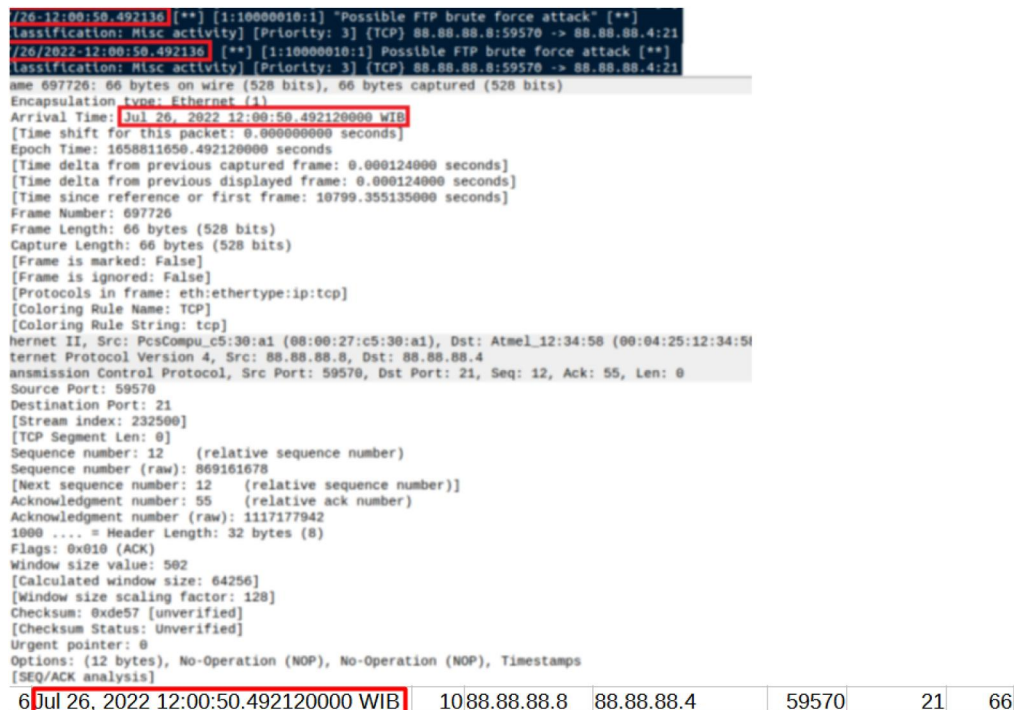


FIGURE 4 Correlation alert Snort and Suricata with data extraction for brute force attack.

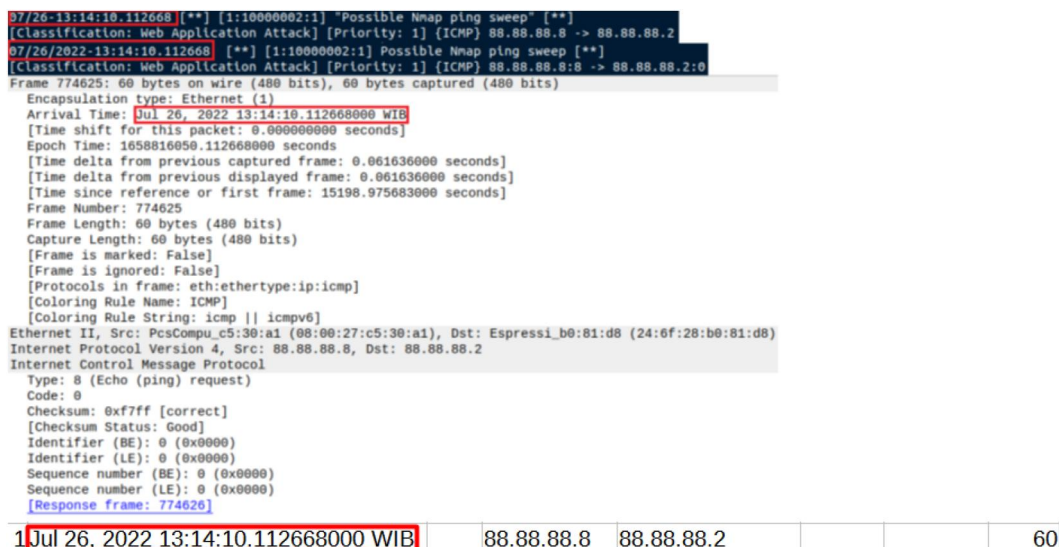


FIGURE 5 Correlation alert Snort and Suricata with data extraction for ICMP flood.

SCADA EIC104 system as the input; the path switching control process carried out on RTU 1 based on HMI commands as the controller; the output that produces outputs based on the input process and the controller; then, the network will describe the data traffic conditions that exist when communication occurs.

The testbed consists of physical devices including HMIs, RTUs and sensors connected to open network devices using the IEC 104 protocol that supports the TCP/IP protocol, as illustrated in Figure 1. The input in the SCADA system is a

device that initialises input commands, such as HMI and sensors. For the experiment's purpose, a sensor as a physical input device is used to read the current, voltage and frequency of the electricity network, while the HMI functions to send commands to open and close the network. The controller is a device used to read inputs and generate outputs based on certain commands. An example of a controller is an RTU, and this paper uses three RTUs with the following details; one RTU has a sensor to read information from the power grid as well as execute commands from the HMI to

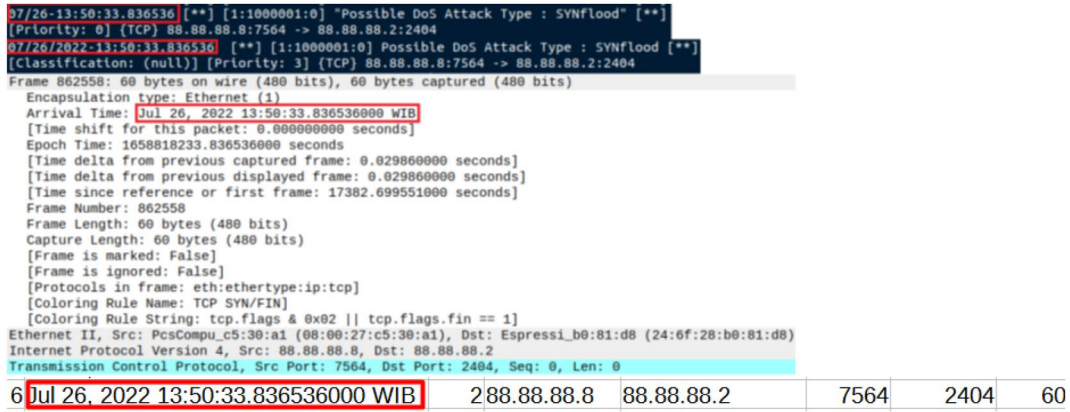


FIGURE 6 Correlation alert Snort and Suricata with data extraction for syn flood.

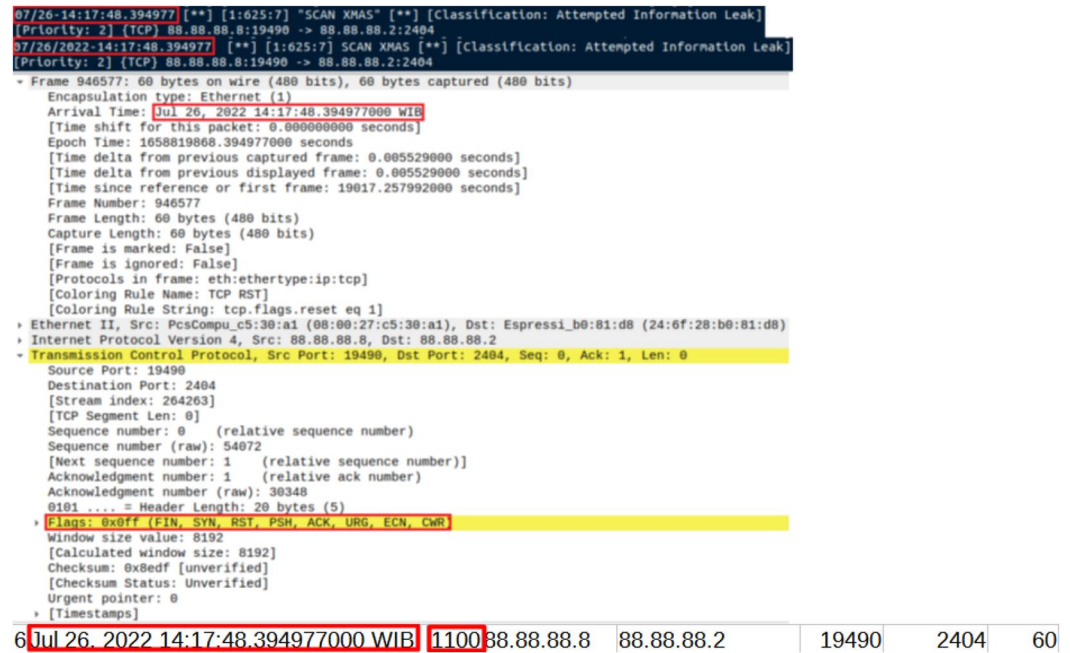


FIGURE 7 Correlation alert Snort and Suricata with data extraction for Xmas.

open and close the circuit, and two RTUs function to read information from the power grid. The output of the testbed is the result of input commands processed by the controller to produce the desired output. The result of the combination of input, controller and output is a SCADA IEC 104 system that affects the realistic level of the created dataset. In addition, the testbed considers the real conditions of the SCADA system of the distribution system section of the power plant industry, and the devices used in the testbed are also physical devices so they are very close to real conditions. When the input, controller, output and network processes are taking place, attacks scenarios are performed to see the effects that occur as well as to produce a realistic dataset based on real conditions.

Two scenarios are conducted: normal and attack scenarios. In a normal scenario, the HMI device monitors the condition of the electricity network and can send commands to the RTU

to open or close the circuit. Table 1 shows the instructions used in the normal scenario with a valid cause of transmission (CoT). The attack scenario starts by port scan activity, followed by performing a brute force attack on RTU 3 to gain access to secure shell (SSH) and file transfer protocol (FTP) services, and finally completed by performing DoS attacks. The DoS attacks involve ICMP flood, syn flood, Xmas and the IEC 104 traffic flood. The IEC 104 flood is performed by flooding the RTU with ASDU packets that are not recognised by the RTU with invalid CoT values.

Having done creating the dataset, some experiments are carried out to evaluate the created dataset, that is, experiment on attacks detection using GrB, DT, and RF algorithms along with their cross-validation; experiment on improving the imbalance data in the dataset using ROS, RUS, and SMOTE; and experiment to validate the attack traffic features using Wireshark, Snort and Suricata software.

TABLE 3 The comparison precision of the IDS model.

Classifier	Method	Precision (%)						
		Normal	Port scan	Brute force	ICMP flood	Syn flood	Xmas	IEC 104 flood
Gradient boosting	Imbalance	91	88	88	100	90	98	100
	RUS	91	93	95	100	89	90	100
	ROS	93	93	95	100	86	96	100
	SMOTE	93	93	95	100	85	96	100
Decision tree	Imbalance	93	93	91	100	93	99	100
	RUS	100	99	98	100	100	97	100
	ROS	94	98	98	100	93	96	100
	SMOTE	92	97	98	100	93	96	100
Random forest	Imbalance	95	93	95	100	90	99	100
	RUS	100	98	99	100	100	97	100
	ROS	95	98	98	100	92	96	100
	SMOTE	94	97	98	100	91	96	100

Note: Bold values indicate the highest values of each model used.

TABLE 4 The comparison recall of the IDS model.

Classifier	Method	Recall (%)						
		Normal	Port scan	Brute force	ICMP flood	Syn flood	Xmas	IEC104 flood
Gradient boosting	Imbalance	94	92	72	100	84	96	100
	RUS	91	93	94	100	90	98	100
	ROS	84	91	95	100	92	100	100
	SMOTE	84	90	94	100	92	100	100
Decision tree	Imbalance	96	97	92	100	87	96	100
	RUS	99	97	98	100	99	100	100
	ROS	92	95	98	100	93	100	100
	SMOTE	92	94	98	100	92	100	100
Random forest	Imbalance	94	97	88	100	90	97	100
	RUS	99	97	98	100	99	100	100
	ROS	90	95	98	100	95	100	100
	SMOTE	90	94	98	100	94	100	100

Note: Bold values indicate the highest values of each model used.

systems [26]. ROC and AUC curves are used to measure the true positive rate (TPR) and false positive rate (FPR) of the model. Figure 2 shows the steps used to find the best method to create a robust IDS model for the SCADA network.

4 | EXPERIMENTAL RESULT AND ANALYSIS

4.1 | Feature extraction result

The recognition of attack patterns and malicious activities is carried out by the observation of results from running the

Wireshark and validation using Snort and Suricata. For common attacks on traditional computer networks, default rules from Snort and Suricata are used to detect port scan, brute force, ICMP flood, syn flood and Xmas. For the IEC 104 flood, manual observations were made. The timestamp in the warning messages from Snort and Suricata is in accordance with the information captured by Wireshark.

Port scanning is the first activity performed by an attacker to see what services are open on the victim's device so that they can perform more specific attacks. In port scanning, the attacker sends fake commands to see the ports and services on the victim's device [27]. The attacker will send TCP packets with syn flags to check what ports are open, and if the port is

TABLE 5 The comparison F1-Score of the IDS model.

Classifier	Method	F1-Score (%)						
		Normal	Port scan	Brute force	ICMP flood	Syn flood	Xmas	IEC 104 flood
Gradient boosting	Imbalance	93	90	79	100	87	97	100
	RUS	89	93	95	100	89	98	100
	ROS	88	92	95	100	89	98	100
	SMOTE	88	90	94	100	87	97	100
Decision tree	Imbalance	95	95	91	100	90	98	100
	RUS	99	98	98	100	100	98	100
	ROS	93	96	98	100	93	100	100
	SMOTE	92	96	98	100	93	98	100
Random forest	Imbalance	94	95	91	100	90	98	100
	RUS	99	98	98	100	100	98	100
	ROS	93	96	98	100	93	98	100
	SMOTE	96	96	98	100	93	98	100

Note: Bold values indicate the highest values of each model used.

open, then the victim device will reply with syn-ack and rst-ack flags for closed ports [28]. Figure 3 shows the results of port scan alert detection using Snort and Suricata along with the correlation with the extraction feature data with a closed port state because it has the rst-ack flags.

The brute force attack scenario is carried out by attacking RTU 3 by targeting the SSH protocol service on port 22 and FTP on port 21. The brute force attack is carried out by entering many username and password combinations to take over the victim's system [29, 30]. Brute force attacks were carried out using the Hydra and Medusa penetration test tools on Kali Linux. Figure 4 shows the correlation between alert from snort and Suricata and extraction results. Figure 4 shows a brute force attack directed at the FTP service with port 21.

ICMP flood is an attack by utilising ICMP echo-request packets to flood the victim's device which causes the target to be inaccessible through legitimate devices [31]. In this scenario, the attacker device will flood the RTU with ICMP packets so that the RTU's communication with the HMI is disrupted. Figure 5 shows the correlation between the Snort and Suricata alert and the dataset extraction results for the ICMP flood attack. The timestamp in the warning messages from Snort and Suricata follows the information captured by Wireshark for the ICMP flood attack.

Syn flood attacks take advantage of massively sent syn-flag packets to overwhelm the victim's network and cause a DoS for legitimate communications [32]. On RTU devices in SCADA systems, this attack will be very fatal because RTU devices have limited computing resources. Figure 6 shows the correlation of the extracted data with the Snort and Suricata alert for syn flood attacks. The matched timestamp indicates that the syn flood attack is indeed present in the dataset.

Xmas is a port scan technique that sends TCP header packets containing Urgent, Push, and Final flags [33]. In other attack techniques, Xmas can be utilised to flood a communication line using TCP packets containing all active flags [34]. In

this research, we use the Xmas attack to flood the communication line between RTU and HMI by sending TCP packets with all flags active. The timestamp and TCP header containing all active flags indicate that the conformity of the Xmas attack is indeed present in the dataset. Figure 7 shows the correlation between Snort and Suricata alerts and data extraction results.

For the IEC 104 flood attack, a modified ASDU packet is sent using a CoT value of 42, a reqco3 value of 40 and an object address (OA) using address 104. Figure 8 shows the detection results of Snort and Suricata with the results of data extraction. Timestamp, CoT value, reqco3 value and OA showing IEC 104 flood attack conformance contained in the dataset.

After the normalisation process, the total amount of data from all classes is 1,048,574. Figure 9 shows the number of classes on the dataset.

4.2 | IDS model performance

We compare the performance of the created IDS models using three classification algorithms namely GrB, DT and RF. The performance of the IDS model is compared using datasets that have not been over-sampled or under-sampled and those that have been over-sampled. Table 2 presents a comparison of the accuracy of the created IDS models. By using the RUS technique, the resulting accuracy increased by 5.36%, while ROS and SMOTE techniques increased the accuracy by 2.92%.

To validate the performance in efficiency, the selected features were evaluated using precision, recall and F1-score [35]. Table 3 presents the value of precision.

Overall, the RF algorithm has better precision results than GrB and RF. The RUS technique in general in this study increases the precision with better values than the ROS and SMOTE techniques.

For the majority class, the results of precision in the DT and RF algorithms with the oversampling method get better

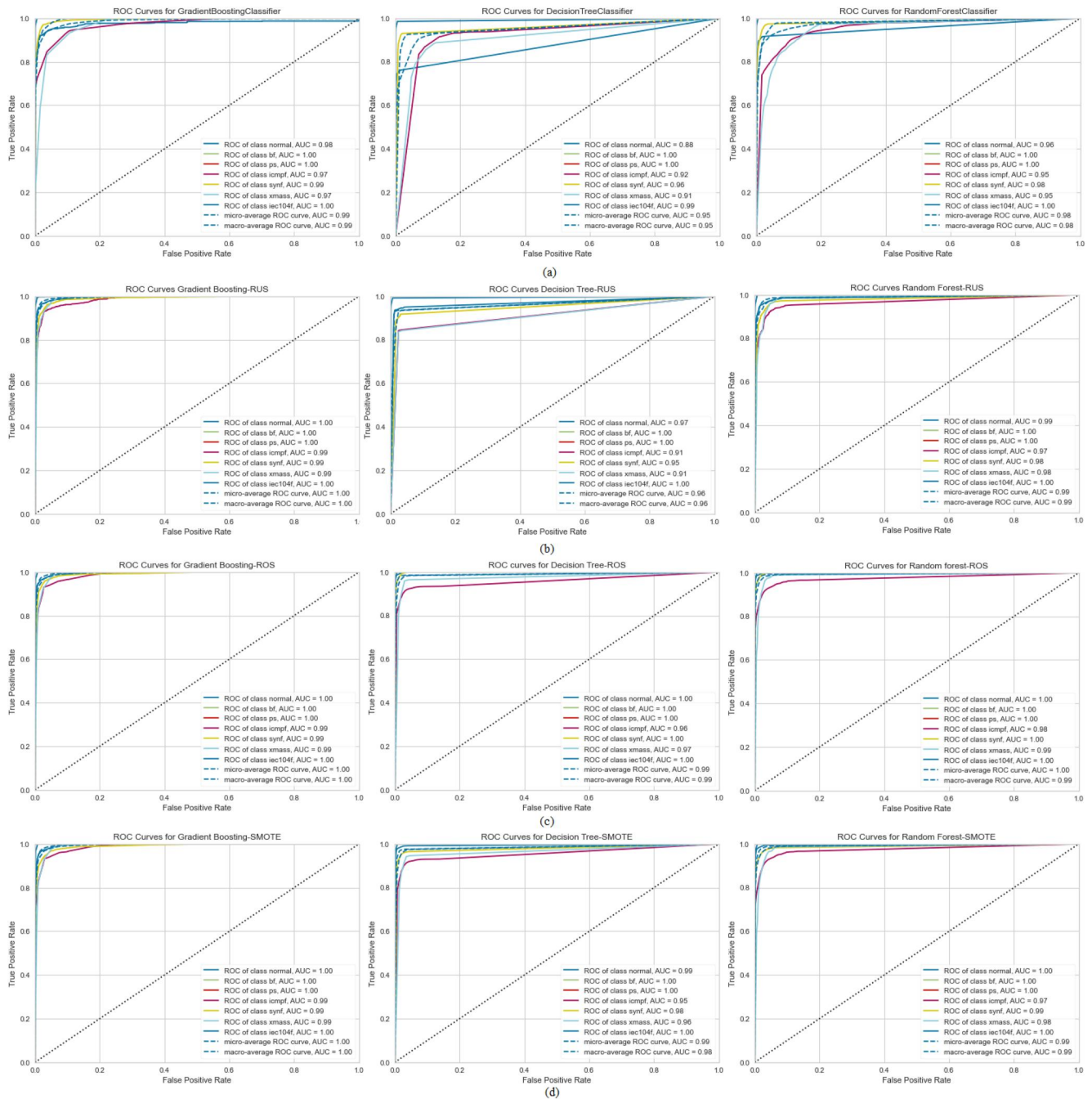


FIGURE 10 ROC curves and AUC values of each trained IDS model.

values than the undersampling method, but for the GrB algorithm, the oversampling method gets better results than the undersampling method. For the minority class, each oversampling and undersampling technique increases the precision with the highest increase when using the RUS. The technique used to balance the classes in the dataset can increase the precision value of each IDS model. Table 4 shows the recall measurement results of the created IDS models.

Recalls on IDS models with the DT algorithm and RF with the RUS technique have the best results for each class. For all machine learning algorithms, the use of the undersampling

technique can increase the Recall of the IDS model. Table 5 shows the F1-Score results of the IDS model.

The class balancing technique on the dataset can improve the F1-score results; the highest improvement is obtained with the RUS technique with DT and RF algorithms.

The ROC curve is used to measure the TPR and FPR of the model, and the AUC value obtained from the ROC curve reflects the accuracy of a classifier model in classification [6, 36]. Figure 10 shows the results of the ROC curves and AUC values of each trained IDS model. (a) are ROC curves and AUC Values for models trained with imbalance dataset, (b) are ROC curves and AUC Values for models trained with balanced

TABLE 6 Cross-validation result.

Classifier	Method	Cross validation (%)	Error (%)
Gradient boosting	Imbalance	91	0
	RUS	94	0.1
	ROS	95	0
	SMOTE	94	0
Decision tree	Imbalance	88	0
	RUS	93	0.1
	ROS	96	0
	SMOTE	95	0
Random forest	Imbalance	89	0
	RUS	93	0.1
	ROS	96	0
	SMOTE	95	0

dataset using ROS, (c) are ROC curves and AUC Values for models trained with balanced dataset using RUS, (d) are ROC curves and AUC Values for models trained with balanced dataset using SMOTE.

The ROC-AUC results show that the GrB algorithm has the best performance with high TPR and low FPR. The ROC curves and AUC values show that the RUS, ROS and SMOTE techniques can improve the performance of the IDS model in detecting attacks on the dataset.

4.3 | IDS model validation

Table 6 presents the validation results using cross-validation and standard deviation on the created IDS model.

From the validation results using cross-validation with 10-folds, no indication of overfitting was found.

5 | CONCLUSION

The SCADA EIC104 system testbed, which is close to real conditions, is required to produce reliable datasets. The created dataset in this paper contains normal traffic and various attacks traffic data and has good quality of data because the traffic data are captured from a proper SCADA EIC 104. Thus, the dataset can be used for training machine learning-based IDS models for SCADA EIC 104 systems.

The created dataset overall consists of a balanced distribution between the normal class and the attacks class; however, if we focus on the individual attack classes, the distribution between the normal class and each individual attack class is considered imbalanced. The experimental results show that the imbalance of classes in the dataset affects the performance of the IDS model in detecting attacks. The use of oversampling and undersampling techniques improves the performance of the IDS models on the created dataset. Thus,

the dataset is suitable for experiments that require both balanced and imbalanced datasets.

From the experimental results, the RF algorithm provides the best accuracy of 93% for an imbalanced dataset. The RUS technique in the DT and RF algorithms provides the highest accuracy of 99.05%. RF also provides the highest accuracy of 96.61% when using the ROS technique. DT and RF with SMOTE provide an accuracy of 96.61%. Using the RUS technique in the RF algorithm increases accuracy by 5.36%, while the ROS technique and SMOTE increase by 2.92%. In the DT algorithm, the RUS technique can increase accuracy by 5.46% while the ROS technique is 2.06% and SMOTE increases by 3.02%. In the GrB algorithm, the RUS technique can increase accuracy by 3.68%, ROS by 3.32% and SMOTE by 3.05%.

The ROC curves and AUC values showed that the created IDS model using GrB has a high TPR and low FPR in detecting attacks. The ROC curves and AUC values also represent the overall performance improvement of the IDS model in detecting attacks when using ROS, RUS and SMOTE on the dataset.

Cross-validation results showed that there is no overfitting on the IDS models. The best cross-validation results are obtained in the IDS model with ROS on the DT and RF which has an average accuracy of 96% with 0% error. Overall, the experimental results showed that the ROS, RUS and SMOTE can improve the detection performance of the IDS models with any of the classifiers: GrB, DT and RF.

AUTHOR CONTRIBUTIONS

M. Agus Syamsul Arifin: Writing—review and editing. **Deris Stiawan:** Writing—review and editing. **Bhakti Yudho Suprpto:** Writing—review and editing. **Susanto Susanto:** Writing—review and editing. **Tasmi Salim:** Writing—review and editing. **Mohd Yazid Idris:** Writing—review and editing. **Rahmat Budiarto:** Writing—review and editing.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

Data available on request due to privacy/ethical restrictions.

ORCID

M. Agus Syamsul Arifin  <https://orcid.org/0000-0002-6568-5173>

Deris Stiawan  <https://orcid.org/0000-0002-9302-1868>

REFERENCES

1. Shaikh, S., et al.: Towards improved classification accuracy on highly imbalanced text dataset using deep neural language models. *Appl. Sci.* 11(2), 1–20 (2021). <https://doi.org/10.3390/app11020869>
2. Puri, A., Gupta, M.K.: Comparative analysis of resampling techniques under noisy imbalanced datasets. In: *IEEE Int. Conf. Issues Challenges Intell. Comput. Tech. ICICT 2019* (2019). <https://doi.org/10.1109/ICICT46931.2019.8977650>
3. Mohammed, R., Rawashdeh, J., Abdullah, M.: Machine learning with oversampling and undersampling techniques: overview study and experimental results. In: *2020 11th Int. Conf. Inf. Commun. Syst. ICICS 2020*, pp. 243–248 (2020). <https://doi.org/10.1109/ICICS49469.2020.239556>

4. Wang, J.H., Septian, T.W.: Combining oversampling with recurrent neural networks for intrusion detection. In: Database Systems for Advanced Applications, vol. 12680, pp. 305–320. LNCS (2021). https://doi.org/10.1007/978-3-030-73216-5_21
5. Bagui, S., Li, K.: Resampling imbalanced data for network intrusion detection datasets. *J. Big Data* 8, 6 (2021). <https://doi.org/10.1186/s40537-020-00390-x>
6. Gupta, N., Jindal, V., Bedi, P.: LIO-IDS: handling class imbalance using LSTM and improved one-vs-one technique in intrusion detection system. *Comput. Netw.* 192(March), 1–19 (2021). <https://doi.org/10.1016/j.comnet.2021.108076>
7. Zuech, R., Hancock, J., Khoshgoftaar, T.M.: Detecting web attacks using random undersampling and ensemble learners. *J. Big Data* 8(1), 75 (2021). <https://doi.org/10.1186/s40537-021-00460-8>
8. Divekar, A., et al.: Benchmarking datasets for anomaly-based network intrusion detection: KDD CUP 99 alternatives. In: Proc. 2018 IEEE 3rd Int. Conf. Comput. Commun. Secur. ICCS 2018, pp. 1–8 (2018). <https://doi.org/10.1109/CCCS.2018.8586840>
9. Tan, X., et al.: Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm. *Sensors* 19(1), 203 (2019). <https://doi.org/10.3390/s19010203>
10. Seo, J.H., Kim, Y.H.: Machine-learning approach to optimize smote ratio in class imbalance dataset for intrusion detection. *Comput. Intell. Neurosci.* 2018, 1–11 (2018). <https://doi.org/10.1155/2018/9704672>
11. Jin, F., et al.: Intrusion detection on internet of vehicles via combining log-ratio oversampling, outlier detection and metric learning. *Inf. Sci.* 579, 814–831 (2021). <https://doi.org/10.1016/j.ins.2021.08.010>
12. Al, S., Dener, M.: STL-HDL: a new hybrid network intrusion detection system for imbalanced dataset on big data environment. *Comput. Secur.* 110, 102435 (2021). <https://doi.org/10.1016/j.cose.2021.102435>
13. Zhang, H., et al.: An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. *Comput. Netw.* 177(April), 107315 (2020). <https://doi.org/10.1016/j.comnet.2020.107315>
14. Hamid, Y., Shah, F.A., Sugumaran, M.: Wavelet neural network model for network intrusion detection system. *Int. J. Inf. Technol.* 11(2), 251–263 (2018). <https://doi.org/10.1007/s41870-018-0225-x>
16. Wu, T., et al.: Intrusion detection system combined enhanced random forest with SMOTE algorithm. *EURASIP J. Adv. Signal Process.* 2022(1), (2022). <https://doi.org/10.1186/s13634-022-00871-6>
15. Qaddoura, R., et al.: A multi-stage classification approach for IoT intrusion detection based on clustering with oversampling. *Appl. Sci.* 11(7), 3022 (2021). <https://doi.org/10.3390/app11073022>
17. Grigoriou, E., Lagkas, T.: Protecting IEC 60870-5-104 ICS / SCADA systems with honeypots. In: IEEE International Conference on Cyber Security and Resilience (CSR), pp. 345–350 (2022). <https://doi.org/10.1109/CSR54599.2022.9850329>
18. Lin, C.-Y., Nadjm-Tehrani, S.: Protocol study and anomaly detection for server-driven traffic in SCADA networks. *Int. J. Crit. Infrastruct. Prot.* 42(June 2022), 100612 (2023). <https://doi.org/10.1016/j.ijcip.2023.100612>
19. Mai, K., et al.: IEC 60870-5-104 network characterization of a large-scale operational power grid. In: Proc. - 2019 IEEE Symp. Secur. Priv. Work. SPW 2019, pp. 236–241 (2019). <https://doi.org/10.1109/SPW.2019.00051>
20. Baiocco, A., Wolthusen, S.D.: Causality re-ordering attacks on the IEC 60870-5-104 protocol. *IEEE Power Energy Soc. Gen. Meet. 2018(August)*, 1–5 (2018). <https://doi.org/10.1109/PESGM.2018.8586010>
21. Wang, X., Foo, E.: Assessing industrial control system attack datasets for intrusion detection. In: 2018 3rd Int. Conf. Secur. Smart Cities, Ind. Control Syst. Commun. SSIC 2018 - Proc, pp. 1–8 (2018). <https://doi.org/10.1109/SSIC.2018.8556706>
22. Shafique, H., et al.: Machine learning empowered efficient intrusion detection framework. *VFAST Trans. Softw. Eng.* 10(2), 27–35 (2022). <https://doi.org/10.21015/vtse.v10i2.1017>
23. Artur, M.: Review the performance of the Bernoulli Naïve Bayes classifier in intrusion detection systems using recursive feature elimination with cross-validated selection of the best number of features. *Proc. Comput. Sci.* 190(2019), 564–570 (2021). <https://doi.org/10.1016/j.procs.2021.06.066>
24. Aamir, M., Zaidi, S.M.A.: DDoS attack detection with feature engineering and machine learning: the framework and performance evaluation. *Int. J. Inf. Secur.* 18(6), 761–785 (2019). <https://doi.org/10.1007/s10207-019-00434-1>
25. Gumaei, A., et al.: A robust cyberattack detection approach using optimal features of SCADA power systems in smart grids. *Appl. Soft Comput. J.* 96(November 2020), 1–17 (2020). <https://doi.org/10.1016/j.asoc.2020.106658>
26. Wang, Z., et al.: A lightweight approach for network intrusion detection in industrial cyber-physical systems based on knowledge distillation and deep metric learning. *Expert Syst. Appl.* 206(May 2022), 1–17 (2022). <https://doi.org/10.1016/j.eswa.2022.117671>
27. Kumar, M.S., et al.: Artificial intelligence managed network defense system against port scanning outbreaks. In: Proc. - Int. Conf. Vis. Towar. Emerg. Trends Commun. Networking, ViTECoN 2019, pp. 1–5 (2019). <https://doi.org/10.1109/ViTECoN.2019.8899380>
28. Hartpence, B., Kwasinski, A.: Combating TCP port scan attacks using sequential neural networks. In: International Conference on Computing, Networking and Communications, ICNC 2020, pp. 256–260 (2020). <https://doi.org/10.1109/ICNC47757.2020.9049730>
29. Hancock, J., Khoshgoftaar, T.M., Leevy, J.L.: Detecting SSH and FTP brute force attacks in big data. In: Proceedings - 20th IEEE International Conference on Machine Learning and Applications, ICMLA 2021, pp. 760–765 (2021). <https://doi.org/10.1109/ICMLA52953.2021.00126>
30. Raikar, M.M., Meena, S.M.: SSH brute force attack mitigation in Internet of Things (IoT) network: an edge device security measure. In: ICSCC 2021 - International Conference on Secure Cyber Computing and Communications, pp. 72–77 (2021). <https://doi.org/10.1109/ICSCC51823.2021.9478131>
31. Ali Shah, S.Q., Zeeshan Khan, F., Ahmad, M.: The impact and mitigation of ICMP based economic denial of sustainability attack in cloud computing environment using software defined network. *Comput. Netw.* 187 (January), 107825 (2021) <https://doi.org/10.1016/j.comnet.2021.107825>
32. Hussain, K., et al.: SYN flood attack detection based on Bayes estimator (SFADBE) for MANET. In: 2019 International Conference on Computer and Information Sciences, ICCIS 2019, pp. 1–4 (2019) <https://doi.org/10.1109/ICCISci.2019.8716416>
33. Banu, R., et al.: Monosek - a network packet processing system for analysis detection of TCP Xmas attack using pattern analysis. In: 2019 International Conference on Intelligent Computing and Control Systems, ICCS 2019, no. Iccics, pp. 952–956 (2019). <https://doi.org/10.1109/ICCS45141.2019.9065325>
34. Abbas, S.G., et al.: Generic signature development for IoT Botnet families. *Forensic Sci. Int. Digit. Investig.* 38, 301224 (2021). <https://doi.org/10.1016/j.fsidi.2021.301224>
35. Upadhyay, D., et al.: Intrusion detection in SCADA based power grids: recursive feature elimination model with majority vote ensemble algorithm. *IEEE Trans. Netw. Sci. Eng.* 8(3), 2559–2574 (2021). <https://doi.org/10.1109/TNSE.2021.3099371>
36. Priyadarsini, P.I.: ABC-BSRF: Artificial Bee Colony and Borderline-SMOTE RF Algorithm for Intrusion Detection System on Data Imbalanced Problem, vol. 56, pp. 15–29. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-8767-2_2

How to cite this article: Arifin, M.A.S., et al.: Oversampling and undersampling for intrusion detection system in the supervisory control and data acquisition IEC 60870-5-104. *IET Cyber-Phys. Syst., Theory Appl.* 9(3), 282–292 (2024). <https://doi.org/10.1049/cps2.12085>