Research Article/Araştırma Makalesi

GAZİ
JOURNAL OF ENGINEERING SCIENCES

Machine Learning-Based DDoS Attack Detection on SDN-Based SCADA Systems

Esra Söğüt[*a], Adem Tekerek[b], O. Ayhan Erdem[c]

Submitted: 03.08.2023 Revised: 13.09.2023 Accepted: 28.09.2023 doi:10.30855/gmbd.0705090

ABSTRACT

Supervisory Control and Data Acquisition (SCADA) systems monitor and control critical infrastructure processes. SCADA systems do not have adequate detection and defense mechanisms against developing cyber attacks and contain many security vulnerabilities. Using SCADA systems in critical infrastructures of national and international importance means new targets for malicious attackers. In addition, using SCADA systems with new technologies brings new perspectives to the security world. When technologies such as SDN are integrated with SCADA systems, it brings advantages to the system regarding manageability and programmability. However, security problems also occur against attacks such as DDoS. For these reasons, it is imperative to ensure the cyber security of SCADA systems. In this study, the case of Software Defined Network (SDN)-based SCADA systems exposed to DDoS attacks are discussed. Logistic Regression, K-Nearest Neighbors, Random Forest, and Support Vector Machine (SVM) classification algorithms have been used for attack detection. A ready-made dataset has been studied, and accordingly, the model that makes the most accurate determination has been proposed in our study. The results show that the proposed SVM classifier model (97.2% accuracy rate) effectively detects DDoS attacks against SDN-based SCADA systems.

Keywords: SCADA, SDN, DDoS, Machine Learning, Modbus

[a,*] Gazi University,
Faculty Of Technology,
Department of Computer Engineering,
06560 – Ankara, Türkiye
Orcid: 0000-0002-0051-2271
e-mail: esrasogut@gazi.edu.tr

[b] Gazi University,
Faculty Of Technology,
Department of Computer Engineering
06560 – Ankara, Türkiye
Orcid: 0000-0002-0880-7955
e-mail: atekerek@gazi.edu.tr

[c] Gazi University,
Faculty Of Technology,
Department of Computer Engineering
06560 – Ankara, Türkiye
Orcid: 0000-0001-7761-1078
e-mail: ayerdem@gazi.edu.tr

[*]Corresponding author:
esrasogut@gazi.edu.tr

SDN Tabanlı SCADA Sistemlerinde Makine Öğrenmesi Tabanlı DDoS Saldırı Tespiti

ÖZ

Kritik altyapılardaki süreçleri izlemek ve denetlemek için Denetleyici Kontrol ve Veri Toplama (SCADA) sistemleri kullanılmaktadır. SCADA sistemleri gelişen siber saldırılar karşısında yeterli tespit ve savunma mekanizmalarına sahip değildir ve birçok güvenlik açıklığı barındırmaktadır. Ulusal ve uluslararası öneme sahip kritik altyapılarda SCADA sistemlerinin kullanılması kötü niyetli saldırganlar için yeni hedefler anlamına gelmektedir. Ayrıca SCADA sistemlerinin yeni teknolojilerle birlikte kullanılması güvenlik dünyasına yeni bakış açıları kazandırmaktadır. Software Defined Network (SDN) gibi teknolojiler SCADA sistemleriyle bütünleştirildiğinde, sisteme yönetilebilirlik ve programlanabilirlik konularında avantajlar kazandırmaktadır. Bunun yanı sıra DDoS gibi saldırılara karşı güvenlik sorunları da barındırmaktadır. Bu sebeplerden dolayı SCADA sistemlerinin siber güvenliğinin sağlanması zorunlu hale gelmiştir. Bu çalışmada SDN tabanlı SCADA sistemlerinin DDoS saldırılarına maruz kalması durumu ele alınmıştır. Saldırı tespitinin yapılması için Logistic Regression, K-Nearest Neighbors, Random Forest ve Support Vector Machine (SVM) sınıflandırma algoritmaları kullanılmıştır. Hazır bir veriseti üzerinde çalışılmış ve buna göre en doğru tespiti gerçekleştiren model çalışmamızda önerilmiştir. Sonuçlar önerilen SVM sınıflandırıcı modelinin (%97.2 oranında doğruluk), SDN tabanlı SCADA sistemlerine yönelik DDoS saldırılarını etkili bir şekilde tespit ettiğini göstermiştir.

Anahtar Kelimeler: SCADA, SDN, DDoS, Makine Öğrenmesi, Modbus

To cite this article: E. Söğüt, A. Tekerek and O. A. Erdem, "Machine Learning-Based DDoS Attack Detection on SDN-Based SCADA Systems," *Gazi Journal of Engineering Sciences,* vol. 9, no. 3, pp. 596-611, 2023. doi:10.30855/gmbd.0705090

1. Introduction

The Supervisory Control and Data Acquisition (SCADA) systems collect real-time data from terminal units, such as input-output units and sensors, and store them in a central unit. The SCADA system evaluates the collected data according to the criteria, creates warning messages, and informs the system's operator. Control points and data flow are monitored with the interface of the SCADA system. SCADA systems serve in critical production, distribution, and utilization infrastructures like water, gas, electricity, and oil. Operations performed in strategic infrastructures are monitored and controlled using SCADA systems throughout the process. The data collected from the terminal units of the SCADA system are used to make predictions about the system's operation. In addition, the control mechanism of the SCADA system provides a fast response to the faults in the terminal units [1].

The SCADA system, a computer-based structure that allows management machines to spread over a wide area from a single center, ensures easy data control and high efficiency in enterprises or industrial facilities. SCADA provides system administrators with detailed reports on the system's operation [2]. Uninterrupted operations in critical infrastructures depend on the robust functioning of SCADA systems. Any disruption in SCADA systems will adversely affect other connected systems, beneficiaries, and institutions. Cyber attacks that develop day by day also target critical infrastructures. For example, cyber attacks targeting the electricity distribution infrastructure can cause cities to face power outages all day and remain dark.

Another known example is the Stuxnet attack. By remotely interfering with the operation of the centrifuges at the nuclear power plant, the attackers secretly disrupted the system and managed to damage the facility physically [3]. Experience shows that these problems can occur at any moment and cause severe financial losses. Therefore, taking every step necessary to ensure cyber security in SCADA systems against cyber attacks is vital.

Production and distribution infrastructures administered with SCADA systems have failed to keep up with the developing technology. Since SCADA systems' primary aim is to manage critical infrastructures efficiently, cyber security in the SCADA systems is of secondary importance. Therefore, SCADA systems are vulnerable to cyber-attacks that are steadily becoming more sophisticated. In addition to the difficulties in adding new elements to this inflexible system, replacing old ones with new ones, and providing security, their closed-system designs make these systems more vulnerable to attacks due to the widespread use of the internet. Using the remote control feature is another insecure practice in the system. Integrating new technologies such as Smart Grid, Internet of Things, 5G, cloud computing, blockchain, and Software Defined Network (SDN) with SCADA systems brings security problems along with many advantages [4], [5].

SDN technology, which offers a dynamic, flexible, and programmable architecture, can eliminate or minimize these problems experienced in the traditional structure of SCADA systems. SDN-based SCADA system obtained by combining SDN technology and SCADA system offers solutions to the complications encountered in regular networks. For example, developing information and communication technologies generated new requirements in accessibility, dynamic management, high bandwidth, and high connectivity. SDN-based SCADA system provides solutions to these manageability, complexity, and quality of service requirements [6].

Besides advantages, SDN technology has disadvantages, such as cyber-attacks specific to SDN architecture. The most threatening type of attack on SDN-based systems is Distributed Denial-of-Service (DDoS) attacks [7]. DDoS attacks that can occur in SDN-based SCADA systems with security vulnerabilities can cause devastating results. Slowdown, downtime, or dysfunction of vital infrastructure processes cause national or international problems. When a critical infrastructure using the SCADA system undergoes DDoS attacks that might affect all or some of the infrastructure, this situation might lead to dangerous consequences in cities, such as untreated drinking water, electricity cuts, or signal failures on high-speed trains. For this reason, it is imperative to maintain business continuity uninterrupted by providing cyber security in SCADA systems.

The current study proposed a model to control and detect DDoS attacks on SDN-based SCADA systems. In the study, four different machine learning algorithms were used to test the reliability of the proposed model. The evaluation was made according to the success metrics such as precision, recall, f1-score, and accuracy. According to the analysis results of the algorithms, Confusion Matrix and ROC Curve were obtained and interpreted. The evaluation results revealed that the model with the highest performance detected DDoS

attacks. Furthermore, the comparisons made with similar literature studies confirmed the success of the proposed model. It has been shown that using complex, hybrid, or advanced algorithms is unnecessary, and successful results are obtained with machine learning methods.

The second part of the study explicitly presented similar studies in the literature and their differences from the proposed model. The third chapter explained the concepts of SCADA, SDN, and DDoS. At the same time, the fourth chapter covered the materials and methods used in the study and outlined the experimental results. The fifth section is the Conclusion part.

2. Literature Review

In this section, studies from the literature concerning the detection of DDoS attacks targeting SCADA systems, communication protocols, and SDN-based SCADA systems have been summarized.

Alhaidari et al. developed a frame against DDoS attacks in the SCADA system using the KDDCup'99 dataset. They employed J48, Naive Bayes (NB), and Random Forest (RF) algorithms to determine the attack pattern and got the best classification success rate with the RF algorithm [8].

Skripcak and Tanuska designed and simulated a prototype for a real-time and online information generation component that can operate in SCADA systems. They utilized a Passive-Aggressive Classifier algorithm with an Online Machine Learning algorithm on the process alarm forecasting scenario, which is considered a binary classification problem [9].

Beaver et al. utilized machine learning methods to detect command and data injection attacks in critical gas pipeline infrastructures. They used a benign and malicious command traffic dataset to identify attacks and made analyses using NB, RF, OneR, J48, Nearest-Neighbor (NNge), and Support Vector Machine (SVM) learning algorithms [10].

Hink et al. used machine learning techniques to detect cyberattacks against the power system and activate the operator. Their study employed NB, RF, OneR, NNge, SVM, JRipper, and Adaboost algorithms and their suggested technique, "Adaboost+JRipper" [11].

Benisha and Ratna proposed a new method for detecting and classifying malicious data in a water storage system using a SCADA system. They utilized the Enhanced Cuckoo Search Optimization algorithm in optimum classification feature selection and the Genetic Machine Learning based Neural Network algorithm in classification [12].

Perez et al. tried to detect network attacks against SCADA systems using machine learning techniques. They used a real dataset from gas pipeline systems and employed SVM and RF methods to implement various Intrusion Detection System classifiers [13].

In their study, Söğüt and Erdem used a dataset from gas pipeline control systems focused on attack detection. This dataset contained data on Command Injection, Reconnaissance, and DoS against the Modbus protocol. They utilized Decision Stump, Hoeffding Tree, Random Tree (RT), and REP Tree algorithms in the study [1].

Wan et al. proposed the Event-Based Hidden Markov Model (HMM) as an anomaly detection approach for communication protocols in SDN-based control systems. They generated data through the simulation environment and operated the Profinet protocol in the study. Furthermore, they evaluated the proposed approach by comparing the performances of Event-Based HMM, BP Neural Network, and NB methods [14].

In their study, da Silva et al. detected cyber attacks on the electrical network components that used the OpenFlow protocol. They proposed an approach based on One-Class Classification algorithms and SDN. For this purpose, they used the One-Class SVM and Support Vector Data Description algorithms [15].

Radoglou-Grammatikis et al. presented an Intrusion Detection and Prevention System with SDN technology for SCADA systems using Distributed Network Protocol 3 (DNP3). The study processing actual data from a transformer station used Minimum Covariance Determinant, Local Outlier Factor, Principal Component Analysis, Isolation Forest, and DIDEROT Auto-encoder methods for the proposed system [16].

Choubineh et al., who detected cyber attacks against a gas pipeline, used Hoeffding Tree, NB, RT, Bayes Net, and OneR machine learning algorithms in their study. They utilized different methods to increase the proposed algorithm's performance and efficiency and compared the results [17].

Wang et al. performed attack detection for the SCADA system using datasets of gas pipeline and power transmission systems using the Modbus protocol. In their study, where they simulated attack scenarios such as Short-circuit fault, line maintenance, command injection, relay setting change, data injection, and DOS, they tested NNge, RF, NB, Adaboost, SVM, Decision Tree, oneR, J48, JRip, and AdaboostJRip, as well as their proposed model, "Stacked Deep Learning" for attack detection [18].

Basnet et al. proposed a new deep learning-based ransomware detection framework in a SCADA-controlled electric vehicle charging station. Therefore, they created ransomware-driven DDoS attacks and ransomware-driven false data injection attacks in the simulation environment. They used Deep neural networks, 1D Convolution Neural Network (CNN), and Long Short-Term Memory (LSTM) algorithms for attack detection [19].

Rajesh et al. created a real-time SCADA network traffic dataset to detect attacks against Industrial Process Control Systems. They utilized Chi-square, ANOVA, and LASSO with SVMS-MOTE metrics to organize the feature values. Then, they applied RF, SVM, K-Nearest Neighbors (KNN), and NB machine learning algorithms for attack detection [20].

Polat et al. detected DDoS attacks on SDN-based SCADA systems. They obtained the dataset by establishing the experimental environment with Modbus TCP communication. Hybrid LSTM, Gated Recurrent Units, and SVM methods were used to detect attacks [7].

Some studies in the literature use their own or ready-made datasets. These studies; used machine learning, deep learning, or hybrid approaches to detect attacks against SCADA systems. The reviewed studies frequently discussed NB, RF, SVM, and LSTM algorithms.

In addition to previous research, SDN-based SCADA systems were discussed in this study, and machine learning methods were used to detect attacks against these systems. A DDoS attack detection system model is proposed. The proposed model highlighted the use of different machine learning methods to eliminate the deficiencies of previous studies and contribute to the field. Besides, a dataset adopting a real-time SDN infrastructure–tested in similar literature studies–was used to perform attack detection. The current study proposed a valuable model for attack detection using LR, KNN, RF, and SVM methods.

3. SCADA, SDN, and Security Relationship

This section includes information about SCADA and SDN technologies, the services offered by these technologies, and cyber security vulnerabilities.

3.1. SCADA systems

SCADA is a system that transmits the information received from the terminal units in the work environment to the central unit, sends commands from the center to the peripheral elements, manages communication, and monitors all operations. While SCADA systems serve in many areas, such as airlines, railways, space systems, power plants, and critical infrastructures of different scopes, Human Machine Interface (HMI) monitors and manages these systems. SCADA systems consist of Master Terminal Units (MTU), Remote Terminal Units (RTU), and end devices. Figure 1 shows SCADA system components.

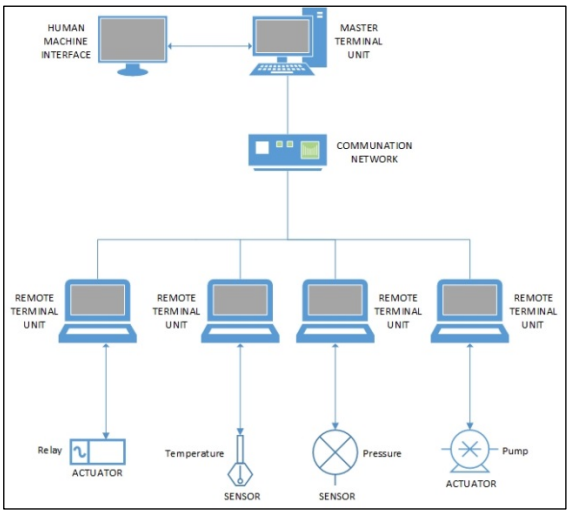

Figure 1. SCADA system components

As seen in Figure 1, end devices, such as sensors or actuators in the fields, send data to the RTU. RTU transmits the data received to MTU. After performing the controls and actions, MTU transfers the decision and the necessary command to the RTU. RTU conveys the relevant commands to terminals, and specified operations thus become complete. Besides these, SCADA systems have a database for event information records and an HMI for the user. These features make it easy to conduct retrospective case studies, data analysis, and process tracking [1].

SCADA systems are designed independently from the external network. Therefore, it works in a closed circuit, has no cyber security mechanisms planned, and has security vulnerabilities [21]. The increased number of sensors used in devices or terminal units in the SCADA system and higher data transmission between the elements raise the system's complexity. Using the Internet in SCADA has enabled the installation of many new technologies. System users can use default or weak-featured passwords, share passwords with others, and prefer remote access to the system. Security solutions such as attack detection and prevention systems or antivirus software cannot ultimately provide security for SCADA systems. Therefore, security vulnerabilities in SCADA systems allow attackers to infiltrate and damage the system. These security vulnerabilities can manipulate SCADA system components with different attack types, such as DOS, DDoS, data modifying, and packet injection [18]. Abnormalities or malfunctions of system components can stop the SCADA and even damage the functioning of other associated systems.

Insecure communication protocols also compromise system security by causing vulnerabilities. Modbus TCP/IP protocol provides no encrypted transmission, authentication check, or authorization; it has many security vulnerabilities. However, it is the most preferred communication protocol because it is an open-source and easy-to-use application [22]. Due to security holes, the Modbus TCP/IP protocol is vulnerable to attacks such as DoS, DDoS, and MITM [23]. DDoS attacks might manipulate the lack of authorization vulnerability of the Modbus TCP/IP protocol and cause the Master device to send messages to the RTUs. This process can consume the resources of the RTU and render them unusable, as in the Alabama Browns Ferry Nuclear Power Plant that was closed down because of DDoS attacks [22].

3.2. Software-Defined Network (SDN)

With the inability to meet the new needs in the IT world with the traditional network approach, new technologies such as SDN have emerged. SDN technology has brought a different perspective to network management by providing opportunities for innovations, such as adding new units to the network, increasing the variety of devices, and using diverse technologies simultaneously. SDN can produce easy and fast solutions to problems and facilitates network management.

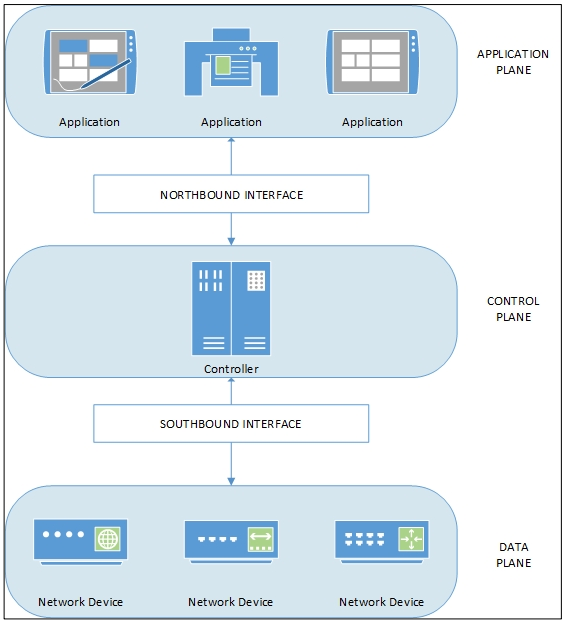

Figure 2. SDN architecture

As seen in Figure 2, there are three planes in the SDN architecture: control, data, and application. The data plane, programmed and managed by the control plane, consists of various communication devices such as routers. Devices on this plane cannot perform high-level network operations. The application plane assists the control plane in security, routing, and network configuration. Applications are programmed with the software located in this plane. In the control plane between the two planes, there are SDN controllers, which are the brains of the network. The controllers communicate with the devices in the data plane using the Southbound interface and the services and applications running on the application plane using the Northbound interface. The control plane manages communication devices [24].

3.3. Distributed Denial-of-Service (DDoS) attacks

DDoS attacks aim to prevent the systems from responding to their users. Network resources in the attacked systems struggle to resist the invasion that massively damages the network infrastructure and renders the services temporarily or permanently inoperable. These attacks occur in two modes: First, numerous requests are forwarded to the target system via infected zombie computers. Second, network security vulnerabilities are detected by infiltrating the target system, and the system is rendered inoperable. DDoS attacks can cause financial, reputation, time losses, and information thefts because of system malfunctions.

SCADA systems are the most exposed when it comes to DDoS attacks. These attacks cause heavy traffic by sending too many requests to the MTU or terminals in the SCADA system. Thus, the target machine cannot respond even to actual requests [25]. A SCADA system that cannot react to user requests can also affect connected systems–machines or terminals–and terminate data exchange.

4. Material and Method

This section includes information about the dataset, methods, and experimental results. In addition, the comparison of the results obtained with the literature is also included in this section.

4.1. Dataset

The current study utilized the dataset produced by Polat et al. in their DDoS attack detection experiments conducted in an SDN-based SCADA network environment [7]. The experimental studies covered four scenarios: TCP flood attack, UDP flood attack, ICMP flood attack, and normal (no attack) scenario. Table 1 shows the network packet (samples) numbers of these scenarios. The dataset had 89 attributes, benign (normal) network traffic data (1188 rows), and DDoS attacks (TCP, UDP, and ICMP flooding) traffic data (a total of 3012 rows). Since the number of samples of these four classes in the dataset is close to each other, the dataset is balanced.

Table 1. The number of packages in the scenarios

Attacks	Number of Packages
TCP flooding UDP flooding ICMP flooding	3012
Normal Request	1188
Total	4200

4.2. Methods

To detect attacks, the study utilized Logistic Regression, K-Nearest Neighbors, Random Forest, and Support Vector Machine–machine learning classification algorithms.

4.2.1. Logistic Regression (LR)

LR is a linear classifier algorithm that finds a hyperplane in the feature space. It separates the obtained observation results according to their classes and converts the output of a linear function using a logistic sigmoid function. Thus, "probability values" that can match a particular class are calculated [26].

It is a successful method of classifying categorical dependent variables using independent variables. LR is used in many areas with nonlinear classification problems, especially in market research, finance, and engineering. The dependent variable is usually coded as "1" and "0" in binary logistic regression models. If the observed result is successful or has a positive meaning, it is coded as "1". In the opposite case, it is coded as "0". Unlike traditional regression models, the error term is hidden in logistic models. In the traditional regression model, there is no e, as in y=b+ax+e, but an error value ei plays a role in the background [27].

4.2.2. K-Nearest Neighbors (KNN)

The KNN algorithm, a preference for classification and prediction problems, represents an easy-to-use supervised machine learning algorithm. The algorithm determines the k nearest neighbors to an unknown taken for classification. By looking at the classes these neighbors belong to, the unknown to be classified is assigned to one or more classes at the closest distance. While determining the data class, it finds the nearest neighbors and calculates the distances [28]. This research found the k value for classification to be 10.

In order to apply the nearest neighbor algorithm method, it is necessary to determine the distance measurement method. For this, one of the "Euclidean Distance" or "Cosine Similarity" measures is usually used [29].

$$X = (X1, X2, \dots, Xn) X = (X1, X2, \dots, Xn) \quad (1)$$

$$Y = (Y1, Y2, \dots, Yn) Y = (Y1, Y2, \dots, Yn) \quad (2)$$

The examples in the classes are shown with Eq.1 and the data to be classified is shown with Eq.2.

$$D(X, Y) = \sqrt{\sum_{i=1}^{n} (xi - yi)^2} \quad (3)$$

Accordingly, the Euclidean Distance between the two vectors is calculated by Eq.3.

4.2.3. Random Forest (RF)

RF can be defined as a collection of tree-structured classifiers. Using the best split among all variables, RF splits the best among a randomly selected subset of predictors at a node instead of dividing all. It creates a new training dataset by modifying the original and then grows a tree using random feature selections. Users can create as many trees as they want. The RF algorithm has been used extensively in different applications [30].

Instead of one classifier, it generates multiple classifiers and then classifies the new data with the votes taken from their predictions. To start this algorithm, two parameters must be defined by the user. These parameters

are the number of variables (m) used at each node and the number of trees to be developed (N) to determine the best split. The number of m variables, taken equal to the square root of the total number of M variables, generally gives the closest result to the optimum result. RF generates trees using the Classification and Regression Tree (CART) algorithm. At each node, branches are created according to the criteria of the CART algorithm (e.g., GINI index) [31]. The GINI index measures class homogeneity and can be expressed by Eq.4 below.

$$\sum\sum_{j\neq i}\left(\frac{f(Ci,T)}{|T|}\right)\left(\frac{f(Ci,T)}{|T|}\right) \quad (4)$$

In Eq.4, T is the training dataset, Ci is the class of a randomly selected pixel, and f(Ci,T)/|T| shows the probability that the selected sample belongs to class Ci. As the GINI index increases, class heterogeneity increases, while class homogeneity increases in the opposite case. When all N trees are produced, the class of candidate pixel is determined based on the prediction results obtained from N trees [32].

4.2.4. Support Vector Machine (SVM)

SVM is a supervised machine learning model that uses regression and classification analysis. It identifies and analyzes patterns. It separates data into two or more dimensions using a line, plane, or hyperplane. For this, it determines the appropriate decision function [33]. The dataset of the two classes can be represented by Eq.5.

$$(x1, y1), (x2, y2), \ldots, (xi, yi)\ i = 1, 2, \ldots, M \quad (5)$$

In Eq.5, x is the input vector with properties belonging to classes of size N. Y indicates the class labels corresponding to them, equal to +1 or -1 [34].

4.3. Experimental results

The current study performed experiments using the dataset to detect DDoS attacks in SDN-based SCADA systems. In the experiments study, four different algorithms were employed. These algorithms are LR, KNN, RF, and SVM. Detailed information about the algorithms is given in the material and methods section. The comparative results obtained from the experiments are presented in this section below.

There are four different classes in the dataset. These three contain different types of DDoS attacks, and the fourth class is normal. For experiments, the dataset is divided for train and testing, 80% of the dataset is arranged for training and 20% for testing. Accordingly, 2410 rows from the attack classes and 950 from the normal class were used for the train. The remaining 602 rows of data from the attacks and 238 rows from the normal class were used for testing. In order to create a highly accurate attack detection system, it is necessary to prepare and train a suitable model [35].

Table 2 presents the experiment results for attack detections performed by the LR classifier algorithm.

Table 2. LR classification results

	Precision	Recall	F1-Score	Support
Normal	0.888	0.974	0.929	155
ICMP Flooding	1.000	0.901	0.948	212
TCP Flooding	0.976	1.000	0.988	239
UDP Flooding	1.000	1.000	1.000	238
Accuracy			0.970	844
Macro Avg	0.966	0.969	0.966	844
Weighted Avg	0.973	0.970	0.970	844

Table 2 shows the analysis results of the LR classifier algorithm by the precision, recall, f1-score, support, and accuracy evaluation metrics. According to the performance results of the model on the f1-score metric, this model obtained the following correct prediction rates: Class-1 (Normal): 0.929, Class-2 (ICMP): 0.948, Class-3 (TCP): 0.988, and Class-4 (UDP): 1.00. The accuracy value of the model was 0.970. While the model's prediction in Class-1 was weak, it was superb in Class-4.

The study utilized a confusion matrix to understand the results of the classification model created, compare the actual values and estimation results, and evaluate the errors (Figure 3). In addition, the results for the

Receiver Operating Characteristic (ROC) Curve value are presented in Figure 4. The ROC Curve furnishes a means for the comprehensive evaluation of a model across the entire spectrum of score thresholds yielded by a classifier [36].

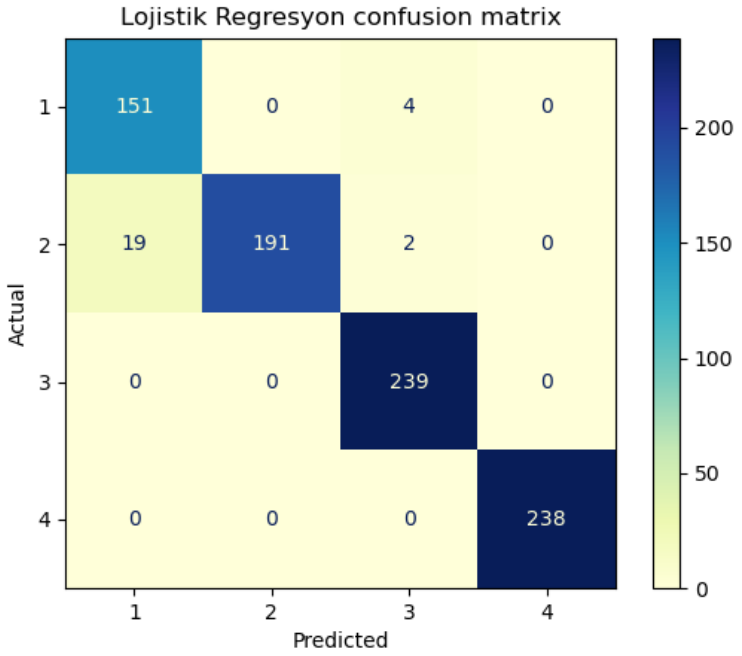

Figure 3. LR confusion matrix results

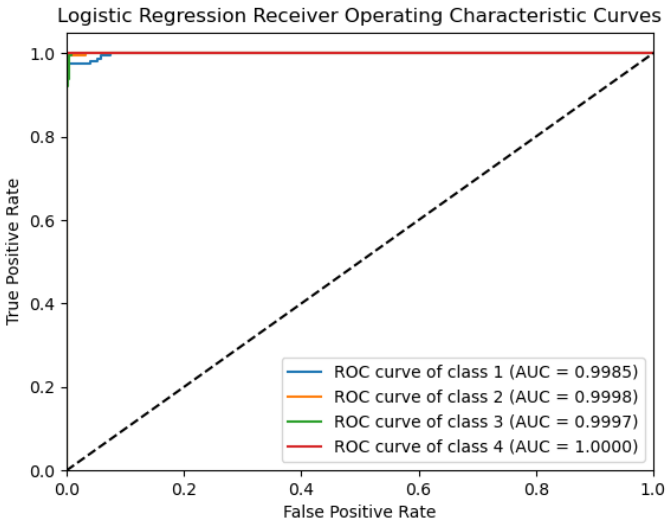

Figure 4. LR ROC Curve

According to the confusion matrix results, the LR classifier algorithm classified Class-1 with 151 true and four false predictions, Class-2 with 191 true and 21 false, Class-3 with 239 true and zero false, and Class-4 with 238 true and zero false. According to the results for ROC Curve, 99.85% for the 1st Class, 99.98% for the 2nd Class, 99.97% for the 3rd Class, and 100% for the 4th Class were obtained.

Table 3 presents the analysis results of the attack detections performed by the KNN classifier algorithm.

Table 3. KNN classification results

	Precision	Recall	F1-Score	Support
Normal	0.985	0.839	0.906	155
ICMP Flooding	0.898	1.000	0.946	212
TCP Flooding	0.987	0.983	0 985	239
UDP Flooding	1.000	1.000	1.000	238
Accuracy			0.966	844
Macro Avg	0.968	0.955	0.959	844
Weighted Avg	0.968	0.966	0.965	844

The KNN classifier algorithm's performance predictions based on the f1-score were as follows: Class-1 (Normal): 0.906, Class-2 (ICMP): 0.946, Class-3 (TCP): 0.985, and Class-4 (UDP): 1.00. The accuracy value of the model was 0.966.

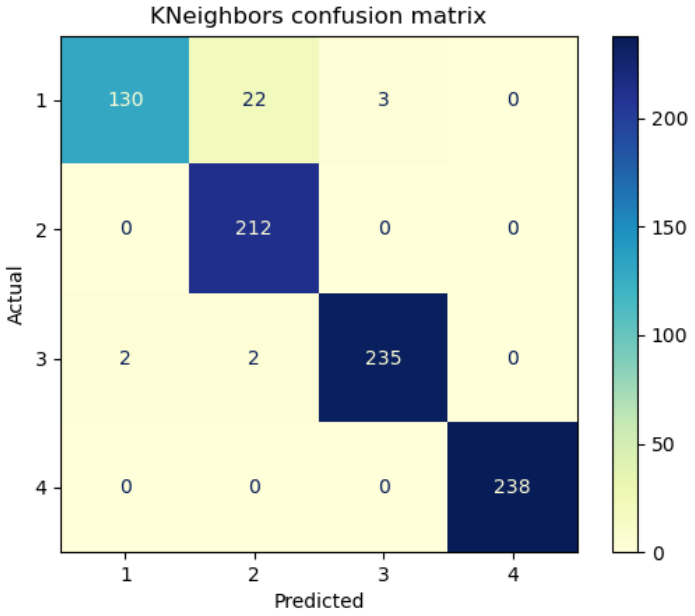

Figure 5. KNN confusion matrix results

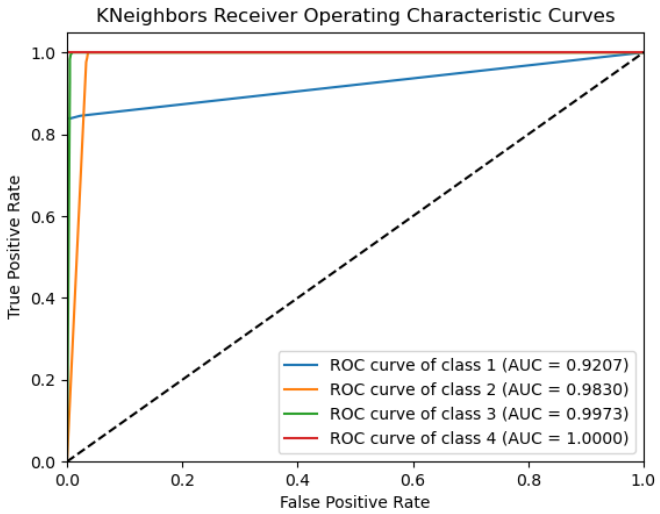

Figure 6. KNN ROC Curve

According to the confusion matrix results (Figure 5), the KNN classifier model classified Class-1 with 130 true and 25 false predictions, Class-2 with 212 true and zero false, Class-3 with 235 true and four false, and Class-4 with 238 true and zero false. According to the results obtained for ROC Curve (Figure 6), 92.07 % results were obtained for the 1st Class, 98.30% for the 2nd Class, 99.73% for the 3rd Class, and 100% for the 4th Class.

Table 4 presents the analysis results of the attack detections performed by the RF classifier algorithm.

Table 4. RF classification results

	Precision	Recall	F1-Score	Support
Normal	1.000	0.535	0.697	155
ICMP Flooding	0. 691	1.000	0.817	212
TCP Flooding	0.944	0.854	0.897	239
UDP Flooding	1.000	1.000	1.000	238
Accuracy			0.873	844
Macro Avg	0.909	0.847	0.853	844
Weighted Avg	0.907	0.873	0.869	844

Table 4 shows the analysis results of the RF classifier model. According to the performance results of the model on the f1-score metric, this model obtained the following correct prediction rates: Class-1 (Normal): 0.697, Class-2 (ICMP): 0.817, Class-3 (TCP): 0.897, and Class 4 (UDP): 1.00. The accuracy value of the model was 0.873.

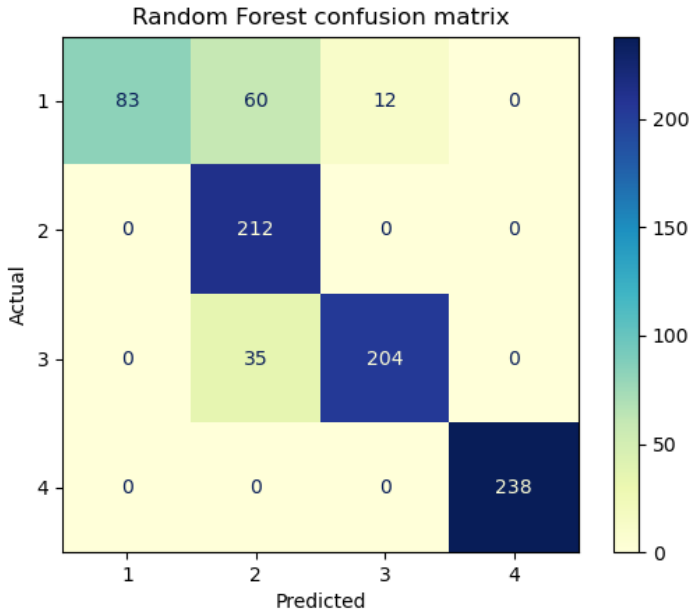

Figure 7. RF confusion matrix results

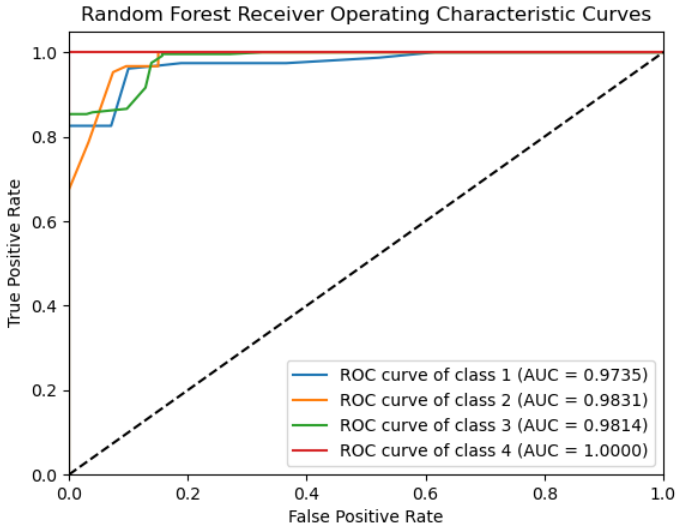

Figure 8. RF ROC Curve

According to the confusion matrix results (Figure 7), the RF classifier model classified Class-1 with 83 true

and 72 false predictions, Class-2 with 212 true and zero false, Class-3 with 204 true and 35 false, and Class-4 with 238 true and zero false. According to the results obtained for the ROC Curve (Figure 8), 97.35% for the 1st Class, 98.31% for the 2nd Class, 98.14% for the 3rd Class, and 100% for the 4th Class were obtained.

Table 5 presents the analysis results of the attack detections performed by the SVM classifier algorithm.

Table 5. SVM classification results

	Precision	Recall	F1-Score	Support
Normal	0.884	0.981	0.930	155
ICMP Flooding	1.000	0.896	0.945	212
TCP Flooding	0.980	1.000	0.990	239
UDP Flooding	1.000	1.000	1.000	238
Accuracy			0.972	844
Macro Avg	0.968	0.969	0.968	844
Weighted Avg	0.974	0.971	0.971	844

According to the performance results of the SVM algorithm on the f1-score metric, this model obtained the following correct prediction rates: Class-1 (Normal): 0.930, Class-2 (ICMP): 0.945, Class-3 (TCP): 0.990, and Class-4 (UDP): 1.00. The accuracy value of the model was 0.972. In general, the model predicted all classes successfully.

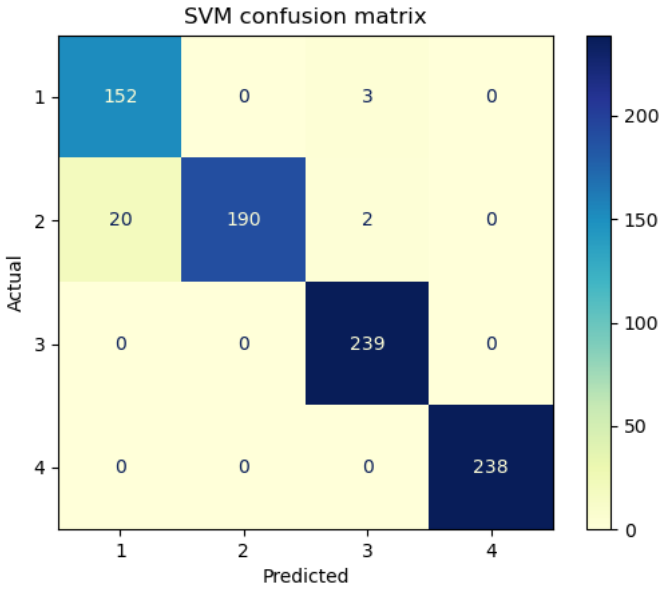

Figure 9. SVM confusion matrix results

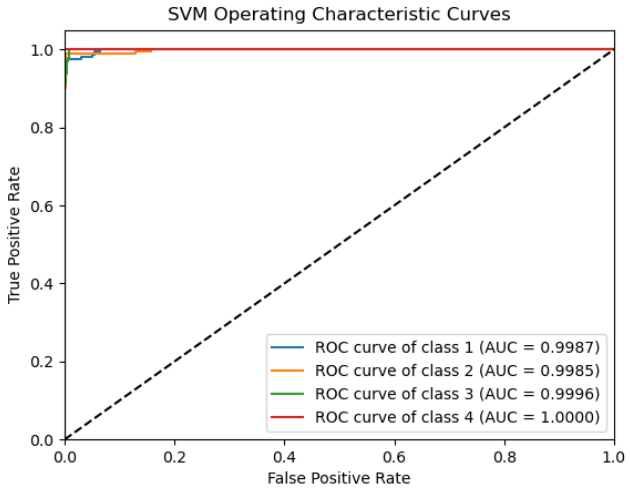

Figure 10. SVM ROC Curve

As seen in Figure 9, the SVM classifier model classified Class-1 with 152 true and 3 false predictions, Class-2 with 190 true and 22 false, Class-3 with 239 true and zero false, and Class-4 with 238 true and zero false. According to the results for the ROC Curve given in Figure 10, 99.87% for the 1st Class, 99.85% for the 2nd Class, 99.96% for the 3rd Class, and 100% for the 4th Class were obtained.

Four different classification models were evaluated, and the results were analyzed. Table 6 shows all the analysis results comparatively.

Table 6. Comparison of classification algorithms

	LR	KNN	RF	SVM
Accuracy	0.970	0.966	0.873	0.972
Precision	0.973	0.968	0.907	0.974
Recall	0.970	0.966	0.873	0.971

As seen in Table 6, the models are compared according to accuracy, precision, and recall metrics. Accuracy is a widely used metric to measure the success of a classification model. The comparison of the classification models showed that the RF model had a worse performance than other models. The LR and KNN classification models were close in performance. The results revealed that the SVM classification model showed the best performance.

The results obtained with the proposed model were compared with previous studies in the literature. Table 7 shows the comparison results. The comparisons conducted according to the sensitivity and accuracy values showed that the proposed model gave relatively better results.

Table 7. Comparison with other studies in the literature

Study	Dataset	Algorithms	Best Performance	Accuracy (%)	Precision (%)
[1]	T. H. Morris et al.	Decision Stump, Hoeffding Tree, RT, REP Tree	RT	84.00	-
[7]	Their dataset	LSTM, Gated Recurrent Units, SVM	Hibrit	97.62	-
[12]	Gamesa Wind Turbines	RF, ECSO-GML	ECSO-GML	97.60	98.10
[14]	Their dataset	Event-Based HMM, BP Neural Network, NB	Event-Based HMM	91.08	-
[16]	N. Rodofile et al.	Minimum Covariance Determinant, Local Outlier Factor, Principal Component Analysis, Isolation Forest, DIDEROT Autoencoder	DIDEROT Autoencoder	95.10	-
[18]	Mississippi State University's SCADA Lab	NNge, RF, NB, Adaboost, SVM, Decision Tree, oneR, J48, JRip, AdaboostJRip, Stacked Deep Learning method	Stacked Deep Learning	97.38	98.59
Proposed Study	The dataset of reference [7]	LR, KNN, RF, SVM	SVM	97.2	97.4

5. Conclusions

The safe continuation of activities in businesses, industrial facilities, or institutions, especially critical infrastructures, depends on the correct functioning of SCADA systems. For this purpose, the cyber security of the system in the structures using the SCADA system has been the primary research topic. Security issues have gained a different perspective by integrating a new technology such as SDN into the SCADA system. This study focuses on detecting three different DDoS attacks on SDN-based SCADA systems.

In addition, the normal class is also included in the dataset so that the system can detect non-attack situations. The dataset used in the study was prepared and made available by [7]. This dataset combines SCADA and SDN technologies, is obtained in a virtual environment, and produces network flow data of DDoS attacks against this system. Due to these features, it differs from the datasets in the literature. The results of the models created with four different machine learning methods on the dataset were examined in detail.

Our study reached the highest accuracy value of 97.2% with the model created with the SVM algorithm. According to other studies examined in the literature, it has been concluded that a complex model is optional to create an attack detection model that can be integrated into SCADA and SDN systems. When the studies are examined based on the accuracy value, our study produced a model with an average of 7.14% higher performance than those with lower success. Again, considering the accuracy values, there is only an average

of 0.33% loss of success compared to the complex structured studies that achieved higher success than ours. In particular, a less complex structure was presented compared to the study, in which the same dataset was used, and a slight difference of 0.42% was obtained in success.

Considering the distributions of attack types, the model we obtained in our study detects TCP and UDP attacks with 100% success. Of the attacks, only ICMP was detected as Normal at a rate of 9.43%, and TCP was detected at a rate of 0.94%. This shows that ICMP and Normal data flow are similar. In the estimation of normal network data, misclassification as TCP was performed with a low rate of 1.94%.

In systems where SCADA and SDN technologies are used together, using simple models that can be adapted efficiently and will not tire the system to provide cyber security may be advantageous. For this purpose, models have been produced for the security of the SDN-based SCADA system with fast machine learning algorithms frequently used in the literature. Fast machine learning algorithms are preferred instead of complex models such as deep learning and hybrid approaches, and the differences are discussed.

In the future, we plan to embed the model we developed into SDN-based SCADA systems. It is aimed to diversify the types of attacks that can be made on these systems and to run different algorithms for attack detection. We aim to contribute more to this field by working on the cyber security of SCADA and SDN-based SCADA systems.

Conflict of Interest Statement

The authors declare that there is no conflict of interest.

References

[1] E. Söğüt and O. A. Erdem, "Endüstriyel Kontrol Sistemlerine (SCADA) Yönelik Siber Terör Saldırı Analizi", *Journal of Polytechnic,* vol. 23, no. 2, pp. 557-566, June 2020. doi:10.2339/politeknik.562570

[2] D. Upadhyay and S. Sampalli, "SCADA (Supervisory Control and Data Acquisition) systems: Vulnerability assessment and security recommendations", *Computers & Security*, vol. 89, 101666, February 2020. doi:10.1016/j.cose.2019.101666

[3] J. P. Farwell and R. Rohozinski, "Stuxnet and the Future of Cyber War", *Survival*, vol. 53, no.1, pp. 23–40, February-March 2011. doi:10.1080/00396338.2011.555586

[4] M. A. Ferrag, M. Babaghayou and M. A. Yazici, "Cyber security for fog-based smart grid SCADA systems: Solutions and challenges", *Journal of Information Security and Applications*, vol. 52, 102500, June 2020. doi:10.1016/j.jisa.2020.102500

[5] P. L. S. Jayalaxmi, R. Saha, G. Kumar and T. H. Kim, "Machine and deep learning amalgamation for feature extraction in Industrial Internet-of-Things", *Computers and Electrical Engineering*, vol. 97, 107610, January 2022. doi:10.1016/j.compeleceng.2021.107610

[6] H. Polat and O. Polat, "An Intellıgent Software Defined Networking Controller Component To Detect And Mitigate Denial Of Service Attacks", *Journal of Information and Communication Technology*, vol. 20, no. 1, pp. 57–81, January 2021. doi:10.32890/JICT.20.1.2021.6288

[7] H. Polat, M. Türkoğlu, O. Polat and A. Şengür, "A novel approach for accurate detection of the DDoS attacks in SDN-based SCADA systems based on deep recurrent neural networks", *Expert Systems with Applications*, vol. 197, 116748, July 2022. doi:10.1016/j.eswa.2022.116748

[8] F. A. Alhaidari and E. M. Al-Dahasi, "New approach to determine DDoS attack patterns on SCADA system using machine learning", in *the 2019 International Conference on Computer and Information Sciences, ICCIS 2019, Sakaka, Saudi Arabia, May 2019.* pp. 1-6. doi:10.1109/ICCISCI.2019.8716432

[9] T. Skripcak and P. Tanuska, "Utilisation of On-line Machine Learning for SCADA System Alarms Forecasting", *in the 2013 Science and Information Conference, London, UK, October 2013.* pp. 477-484.

[10] , J. M. Beaver, R. C. Borges-Hink and M. A. Buckner, "An evaluation of machine learning methods to detect malicious SCADA communications", in *the12th International Conference on Machine Learning and Applications, ICMLA 2013, 2, Miami, FL, USA, December 2013.* pp. 54–59. doi:10.1109/ICMLA.2013.105

[11] R. C. Borges Hink, J. M. Beaver, M. A. Buckner, T. Morris, U. Adhikari and S. Pan, "Machine learning for power system disturbance and cyber-attack discrimination", *in the 7th International Symposium on Resilient Control Systems, ISRCS 2014, Denver, CO, USA, August 2014.* pp. 1-8. doi:10.1109/ISRCS.2014.6900095

[12] R. B. Benisha and S. Raja Ratna, "Detection of Intrusion using Enhanced Machine Learning Model in SCADA Wireless Network", *International Journal of Future Generation Communication and Networking*, vol. 13, no. 1, pp. 85–98, 2020.

[13] R. Lopez Perez, F. Adamsky, R. Soua and T. Engel, "Machine Learning for Reliable Network Attack Detection in SCADA Systems", in *the 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications and 12th IEEE International Conference on Big Data Science and Engineering, Trustcom/BigDataSE 2018, New York, NY, USA, September 2018*. pp. 633–638. doi:10.1109/TrustCom/BigDataSE.2018.00094

[14] M. Wan, J. Yao, Y. Jing and X. Jin, "Event-based anomaly detection for non-public industrial communication protocols in SDN-based control systems", *Computers, Materials and Continua*, vol. 55, no. 3, pp. 447–463, 2018. doi:10.3970/cmc.2018.02195

[15] E. G. da Silva, A. S. da Silva, J. A. Wickboldt, P. Smith, L. Z. Granville and A. Schaeffer-Filho, "A One-Class NIDS for SDN-Based SCADA Systems", in the *International Computer Software and Applications Conference, 1, Atlanta, GA, USA, June 2016*. pp. 303–312. doi:10.1109/COMPSAC.2016.32

[16] P. Radoglou-Grammatikis, P. Sarigiannidis, G. Efstathopoulos, P. A. Karypidis and A. Sarigiannidis, "DIDEROT: An intrusion detection and prevention system for DNP3-based SCADA systems", in the *15th International Conference on Availability, Reliability and Security, 115, New York, NY, USA, August 2020*. pp. 1-8. doi:10.1145/3407023.3409314

[17] A. Choubineh, D. A. Wood and Z. Choubineh, "Applying separately cost-sensitive learning and Fisher's discriminant analysis to address the class imbalance problem: A case study involving a virtual gas pipeline SCADA system", *International Journal of Critical Infrastructure Protection*, vol. 29, 100357, 2020. doi:10.1016/j.ijcip.2020.100357

[18] W. Wang, F. Harrou, B. Bouyeddou, S. M. Senouci and Y. Sun, "A stacked deep learning approach to cyber-attacks detection in industrial systems: application to power system and gas pipeline systems", *Cluster Computing*, vol. 25, no. 1, pp. 561–578, 2022. doi:10.1007/s10586-021-03426-w

[19] M. Basnet, S. Poudyal, M. H. Ali and D. Dasgupta, "Ransomware detection using deep learning in the SCADA system of electric vehicle charging station", in the *2021 IEEE PES Innovative Smart Grid Technologies Conference - Latin America, ISGT Latin America, Lima, Peru, September 2021*. doi:10.1109/ISGTLATINAMERICA52371.2021.9543031

[20] L. Rajesh and P. Satyanarayana, "Evaluation of Machine Learning Algorithms for Detection of Malicious Traffic in SCADA Network", *Journal of Electrical Engineering and Technology*, vol. 17, no. 2, pp. 913–928, 2022. doi:10.1007/s42835-021-00931-1

[21] S. East, J. Butts, M. Papa and S. Shenoi, "A taxonomy of attacks on the DNP3 protocol", *IFIP Advances in Information and Communication Technology*, vol. 311, pp. 67–81. March 2009. doi:10.1007/978-3-642-04798-5_5

[22] P. Kamal, A. Abuhussein and S. Shiva, "Identifying and Scoring Vulnerability in SCADA Environments", in the *Future Technologies Conference, Vancouver, Canada*, *November 2017*. pp. 845-857.

[23] B. Chen, N. Pattanaik, A. Goulart, K. L. Butler-Purry and D. Kundur, "Implementing attacks for modbus/TCP protocol in a real-time cyber physical system test bed", in *the CQR 2015: 2015 IEEE International Workshop Technical Committee on Communications Quality and Reliability*, *Charleston, SC, USA, May 2015*. pp. 1-6. doi:10.1109/CQR.2015.7129084

[24] M. Kuerban, Y. Tian, Q. Yang, Y. Jia, B. Huebert and D. Poss, "FlowSec: DOS attack mitigation strategy on SDN controller", in *the 2016 IEEE International Conference on Networking Architecture and Storage, Long Beach, CA, USA, August 2016*. pp. 1-2. doi:10.1109/NAS.2016.7549402

[25] S. Shitharth, and D. P. Winston, "A Comparative Analysis between Two Countermeasure Techniques to Detect DDoS with Sniffers in a SCADA Network", *Procedia Technology*, vol. 21, pp. 179–186, 2015. doi:10.1016/J.PROTCY.2015.10.086

[26] A. Desai, Y. Guo, S. Sheng, S. Sheng, C. Phillips and L. Williams, "Prognosis of Wind Turbine Gearbox Bearing Failures using SCADA and Modeled Data", *Annual Conference of the PHM Society*, vol. 12, no. 1, pp. 1-10, 2020. doi:10.36001/phmconf.2020.v12i1.1292

[27] W. H. Greene, *Econometric Analysis,* Statistical Papers, Springer, vol. 52, iss. 4, 2011, pp. 983-984. doi:10.1007/s00362-010-0315-8

[28] A. Gumaei, M. M. Hassan, S. Huda, Md. R. Hassan, D. Camacho, J. Del Ser and G. Fortino, "A robust cyberattack detection approach using optimal features of SCADA power systems in smart grids", *Applied Soft Computing*, vol. 96, 106658, 2020. doi:10.1016/j.asoc.2020.106658

[29] G. Silahtaroğlu, *Kavram ve Algoritmalarıyla Veri Madenciliği*. Papatya Yayınları, İstanbul, 2008.

[30] J. Zhang, M. Zulkernine and A. Haque, "Random-forests-based network intrusion detection systems", *IEEE Transactıons On Systems, Man, and Cybernetics-Part C: Applications And Reviews*, vol. 38, no. 5, pp. 649–659, 2008. doi:10.1109/TSMCC.2008.923876.

[31] L. Breiman, "Random Forests", *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001. doi:10.1023/A:1010933404324

[32] J. D. Watts, S. L. Powell, R. L. Lawrence and T. Hilker, "Improved Classification of Conservation Tillage Adoption Using High Temporal and Synthetic Satellite Imagery", *Remote Sensing of Environment*, vol. 115, no. 1, pp. 66–75, 2011. doi:10.1016/j.rse.2010.08.005

[33] Z. Qi., Y. Tian and Y. Shi, "Robust Twin Support Vector Machine for Pattern Classification", *Pattern Recognition*, vol. 46, no. 1, pp. 305-316, 2013. doi:10.1016/j.patcog.2012.06.019

[34] H. Adaminejad, I. Shayegani, M. Ohammadi and E. Farjah, "An Algorithm for Power Quality Events Core Vector Machine-Based Classification", *The Modares Journal of Electrical Engineering*, vol. 12, no. 4, pp. 50-59, 2013.

[35] E. Duman and O. A. Erdem, "Anomaly Detection in Videos Using Optical Flow and Convolutional Autoencoder," *IEEE Access*, vol. 7, pp. 183914-183923, 2019. doi:10.1109/ACCESS.2019.2960654

[36] S. Oyucu, H. Sever, and H. Polat, "Otomatik Konuşma Tanımaya Genel Bakış, Yaklaşımlar ve Zorluklar: Türkçe Konuşma Tanımanın Gelecekteki Yolu", *Gazi University Journal of Science Part C: Design and Technology*, vol. 7, no. 4, pp. 834-854, 2019. doi:10.29109/gujsc.562111