Algorithms for detecting network attacks in an enterprise industrial network based on data mining algorithms

A M Vulfin, V I Vasilyev, S N Kuharev, E V Homutov and A D Kirillova
Ufa State Aviation Technical University, 12, K. Marks st., Ufa, 450008, Russian Federation

E-mail: vulfin.alexey@gmail.com

Abstract. The article proposes the structure of a network traffic analysis system using machine learning models. The analysis of the specialized dataset WUSTL-IIOT-2018 was carried out, the main stages of data preprocessing, construction and testing of classifiers for detecting network attacks were performed. The possibility of embedding models as modules of specialized network equipment is proposed, which makes it possible to increase the efficiency of the analysis of network traffic, including specialized industrial protocols.

1. Introduction

One of the key directions of ensuring energy security is to improve the state of protection of critical infrastructure facilities of fuel and energy complexes. This is facilitated by an increase in the number of vulnerabilities found in the equipment of industrial control systems (ICS) in the fuel and energy complex. According to the Claroty report, their number has grown by almost 25% in 2020 compared to 2019.

Thus, the securing the perimeter and entry points to industrial systems becomes a critical task. For industrial equipment and edge systems, industrial network entry points, it is necessary to provide deep traffic analysis to detect network attacks with support for industrial protocol analysis. Improvement of network infrastructure protection means is aimed at developing tools for intelligent monitoring of network traffic and the state of objects and nodes of the industrial network.

Considering the above, we can conclude that it is relevant and necessary to improve systems for detecting network attacks based on the use of artificial intelligence methods, as a key element in ensuring the cybersecurity of critical infrastructure of the fuel and energy complex in the concept of the digital economy development.

The goal of the work is to increase the security of the industrial network of ICS based on algorithms for intelligent analysis of network traffic. To achieve this goal, the tasks of developing algorithms for intelligent analysis of network traffic and evaluating the effectiveness of the proposed solution based on field data have been set and solved.

2. Analysis of the features of detecting network attacks in industrial systems based on machine learning methods

In connection with the spread of the practice of organizing remote access to the industrial network segment through public networks, the risk of implementing cyber-attacks on SCADA systems...
increases. Intrusion Detection System (IDS) is an effective security monitoring tool for industrial and hybrid networks, analyzing traffic and the system to identify malicious actions of an attacker [1].

Machine learning (ML) and artificial intelligence methods are widely used to create effective intelligent IDS, which have proven their effectiveness in detecting anomalous events in the network traffic flow [2]. IDs created on the basis of ML methods, for training ML-models, use publicly labeled network traffic databases marked by attack types and operating modes (NSL-KDD, CICIDS-2017, UNSW-NB15, BOT-IOT and others), assembled using semi-natural stands simulating corporate and industrial networks. To ensure the ability to detect new network attacks implemented by constantly evolving attacker tools, it is necessary to periodically update training sets with the implementation of new attack scenarios and fixing the parameters of their implementation for further training of ML-models in the IDS core.

In [3], a stand is described, built using industrial equipment, for researching ML algorithms for detecting network attacks. During the implementation of complex attacks according to various scenarios, network traffic is collected that corresponds to the normal operation of the system and anomalous states - network attacks. A feature of this dataset is the emphasis on the use of industrial protocols, primarily the Modbus protocol in the Modbus-over-TCP version. The work [4] emphasizes the special role of Modbus as one of the most popular protocols in industrial networks. After conducting reconnaissance and gaining a foothold in the industrial network, an attacker can modify control commands or sensor readings, which can lead to serious cyber-physical consequences.

A fragment of the basic architecture of the ICS stands of the water supply system that controls the water storage tank (figure 1) includes two water level sensors (LS₁ and LS₂), a programmable logic controller (PLC), water pumps (WP₁ and WP₂), a valve (V) that controls tank level, HMI for real-time monitoring and control of the water storage system, History Logs server for storing system logs and events, and a router with integrated firewall functions. The main network control protocol is Modbus-over-TCP.

![Figure 1. Fragment of the basic architecture of ICS.](image)

Network attacks on SCADA systems can be divided into three categories: reconnaissance, command injection, and denial of service (DoS).

Considered [3] reconnaissance network attacks in which the network is scanned by an attacker to identify possible vulnerabilities, the exploitation of which will allow him to gain a foothold in the industrial network segment. Some of the attacks, the implementation of which significantly increases the number of transmitted packets, are confidently detected using standard signature methods. But most of the attacks using exploits practically do not change the basic characteristics of the traffic, which makes it very difficult to select signatures to detect them. Application of ML methods makes it possible to reveal the features of anomalous traffic and build an appropriate detector.
The main reconnaissance attacks, for which the traffic was captured and the subsequent highlighting of significant features:

- Port Scanner – identification of common SCADA protocols;
- Address Scan Attack – scanning network addresses and determining the address of the Modbus server;
- Device Identification Attack (passive and aggressive modes) – formation of a list of identifiers of Modbus slave devices;
- Exploit Injection – determination of the state of individual actuators.

3. The structure of the network traffic analysis system based on ML algorithms

The proposed network traffic analysis system based on ML algorithms is presented in figure 2.

![Figure 2. The structure of the network traffic analysis system.](image)

Let's consider the main elements of the proposed system. The history log server (1) of network sessions and the firewall (2) are a source of data for further training of the IDS ML core when implementing new attack scenarios.

The preprocessing and feature generation module allows to select the essential parameters of a network session for the subsequent construction of a training sample of the model. Two-way interaction with the security information and event management / security operations center (SIEM / SOC) is used to enrich data on network sessions and related information security events. The process of marking (enriching) records of network sessions is controlled by a network security specialist (3) of the current segment.

Labeled data allows to create a database of examples for training ML-models. The bank of ML-models (5) is replenished with tested models. The process of selecting model hyperparameters is coordinated by a knowledge engineer (4). The prepared models are ready for embedding in the form of software modules into the corresponding network equipment (routers, managed switches and firewalls) or for use as part of a network IDS.

Next, consider the process of preparing data and training ML models on labeled data.
4. Computational experiment

4.1. Preliminary data analysis
The main features that identify network sessions in the bench system are presented in table 1. The total number of examples of network sessions classified as normal is 6634581. The number of examples classified in the class of attacks is 403402.

At the first stage, each feature is normalized – reduced to zero mean and unit standard deviation. In view of a sufficiently large number of examples in the analyzed set, visualization using the PCA does not provide an unambiguous assessment of the separability of objects of two classes. Next, apply t-Distributed Stochastic Neighbor Embedding (t-SNE) [5; 6] to reduce the dimension of the feature space and visualize the distribution of examples by class (figure 3).

Table 1. Features selected for creating a dataset and describing network sessions.

| Feature | Description                                      |
|---------|--------------------------------------------------|
| Sport   | Source port number                               |
| TotPkts | Total number of packages in a transaction        |
| TotBytes| Total transaction bytes                          |
| SrcPks  | Number of source / destination packages          |
| DstPks  | Number of destination / source packets           |
| SrcBytes| Source / trailing transaction bytes              |
| Target  | Network session tag (normal operation / attack)  |

![Figure 3. Feature space visualization using t-SNE. 0 – “normal operation”, 1 – “attack”.

The use of t-SNE makes it possible to assess the separability of anomalous network sessions from the main dataset and to make an assumption about the possibility of constructing an effective classifier.

4.2. Assessing the significance of features
The dataset is divided into training and test sets in a ratio of 70% and 30%. The training set contains 4644485 examples of normal traffic and 282103 records corresponding to the attacks being implemented. The test set contains 1990096 examples of normal traffic and 121299 entries corresponding to attacks.

A classifier based on a decision tree [7; 8] is constructed on the training set using a cross-validation algorithm with division into 10 groups, and its parameters are estimated. The algorithm for maximizing the Gini uncertainty criterion is applied. To assess the significance of features, all features
are selected for which the importance indicator (specified by the ML model with a teacher) exceeds the specified threshold value $\tilde{\theta} = 0.1667$.

Sorted in descending order, the assessment of the significance of the features is shown in Fig. 4. The classifier based on the decision tree allows to make a conclusion about the strong redundancy of the collected features and the possibility of separating examples by two or even one feature. A classifier based on a committee of random decision trees [9] also allows analyzing the significance of features.

Sorted in descending order, the evaluation of the significance of the features is shown in figure 4.

![Figure 4. Assessing the significance of features using a decision tree classifier and a random tree committee classifier.](image)

### 4.3. Building final classifiers

For a classifier based on a committee of random trees, hyperparameter selection is performed using a grid search. The criterion for comparing the models is the F1-score, weighted by the coefficient that determines the proportion of examples in the classes of normal and abnormal work to level the imbalance, and accuracy (table 2). A total of 1200 combinations of classifier parameters were analyzed.

**Table 2. Selection of classifier hyperparameters.**

| Parameter                  | Variants          | Selected value |
|----------------------------|-------------------|----------------|
| number of trees on the committee | 5, 10, 25, 50, 100 | 10             |
| limitation on the number of objects in leaves | 3, 5, 10, 20 | 3              |
| number of features to split | 3, 5, 6           | 6              |

The confusion matrix on the test samples for the classifier with fitted hyperparameters is shown in table 3.

**Table 3.** The confusion matrix on the test sample for the classifier with fitted hyperparameters.

| Actual class | Predicted class | normal operation | attack |
|--------------|-----------------|------------------|--------|
| normal operation | 1990094       | 2                |
| attack       | 0               | 121299           |

Classification quality metrics on the test sample: Accuracy = 0.9999, Precision = 0.9999, Recall = 1.0, F1 = 0.9999.
5. Results
Let us consider the effectiveness of using various classifiers, the parameters of which are presented in table 4, to solve the problem of detecting network attacks. Classifier testing was performed with 5-pass cross validation.

| Model                          | Accuracy | Precision | Recall | F1      | Training time, s. |
|--------------------------------|----------|-----------|--------|---------|------------------|
| RandomForestClassifier (RF)    | 1.000    | 1.000     | 1.000  | 0.919   | 111.36           |
| LogisticRegression (LR)        | 0.999    | 0.983     | 0.995  | 0.918   | 175.21           |
| MLPClassifier (MLP)            | 0.999    | 0.986     | 0.995  | 0.919   | 2459.23          |

6. Discussion
The trained RF and MLP ML-models are supposed to be used in the form of embedded software modules of the corresponding network equipment. With the help of the translator [10] from the Python language, header files and implementation files in the C language were created with the unloading of the coefficients of the trained models as static parameters. Further compilation using the GCC [11] cross-compiler made it possible to build executable modules for the ARM platform of the NXP LX2160A [12] family of specialized processors.

The efficiency of the obtained solutions in assessing the quality of network attack detection on the original dataset is comparable for the tested models. The most promising classifier for use in specialized signal processors of network equipment is a classifier based on a committee of random trees, since it provides good quality detection of network attacks and does not require significant computing resources when starting a model with coefficients selected during the training process.

7. Conclusion
The paper presents the structure of a network traffic analysis system using machine learning methods. An exploratory analysis of the WUSTL-IIOT-2018 dataset was carried out, and the significance of the features was assessed. The classification quality for several analyzed models was assessed using a weighted measure value $F1 = 0.91-0.95$ under various test scenarios. The possibility of embedding the obtained models as modules of network equipment to increase the efficiency of the analysis of network traffic of industrial systems is considered.

Acknowledgments
The reported study was funded by RFBR according to the research project No. 19-07-00972 A

References
[1] Priya V et al. 2021 Robust attack detection approach for IIoT using ensemble classifier. *Computers, Materials & Continua* **66**(3) 1-14
[2] Arshad J et al. 2020 A review of performance, energy and privacy of intrusion detection systems for IoT. *Electronics* **9**(4) 1-24
[3] Teixeira M A et al. 2018 SCADA system testbed for cybersecurity research using machine learning approach. *Future Internet* **10**(8) 76
[4] Miciolino E E, Bernieri G, Pascucci F and Setola R 2015 Communications network analysis in a SCADA system testbed under cyber-attacks. *Proceedings of the 23rd Telecommunications Forum, Belgrade, Serbia, 24-26 November*
[5] Van der Maaten L and Hinton G 2008 Visualizing data using t-SNE. *Journal of machine learning research* **9**(11) 2579-2605
[6] Arora S, Hu W and Kothe P K 2018 An analysis of the t-sne algorithm for data visualization. *Conference on Learning Theory, PMLR* 1455-1462
[7] Kohavi R and Quinlan J R 2002 Data mining tasks and methods: Classification: decision-tree
discovery. *Handbook of data mining and knowledge discovery* 267-276

[8] Kotsiantis S B 2013 Decision trees: a recent overview. *Artificial Intelligence Review* **39**(4) 261-283

[9] Denisko D and Hoffman M M 2018 Classification and interaction in random forests. *Proceedings of the National Academy of Sciences*. **115**(8) 1690-1692

[10] Lê N C et al. 2020 A Machine Learning Approach for Real Time Android Malware Detection. *2020 RIVF International Conference on Computing and Communication Technologies (RIVF), IEEE* 1-6

[11] Fursin G et al. 2011 Milepost gcc: Machine learning enabled self-tuning compiler. *International journal of parallel programming* **39**(3) 296-327

[12] Egorov V B 2021 Evolution of network processors. *Systems and Means of Informatics* **31**(1) 111-121