Trust Monitoring in a Cyber-Physical System for Security Analysis Based on Distributed Computing

Elena Basan[1(✉)], Maria Lapina[2], Alexander Lesnikov[1], Anatoly Basyuk[1], and Anton Mogilny[1]

[1] Southern Federal University, Taganrog, Russian Federation
ebasan@sfedu.ru
[2] North Caucasus Federal University, Stavropol, Russian Federation

Abstract. Cyber-physical systems are widely used. Nevertheless, security issues are quite acute for them. First of all, because the system must work constantly without downtime and failures. The Cyber-Physical System (CPS) must quickly transfer the parameters to the monitoring system, but if the system is not flexible enough, fast and optimal, then collisions and additional loads on the CPS may occur. This study proposes a system for monitoring and detecting anomalies for CPS based on the principles of trust, which allows you to verify the correctness of the system and detect possible anomalies. In our study, we focus on traffic analysis and analysis of the CPU operation, since these parameters are the most critical in the operation of the CPS itself. The technique is based on computationally simple algorithms and allows to analyze the basic parameters that are typical for most CPS. These factors make it highly scalable and applicable to various types of CPS, despite the fragmentation and a large number of architectures. A distributed application architecture was developed for monitoring and analyzing trust in the CPS. The calculation results show the possibility of detecting the consequences of the influences of denial-of-service attacks or CPS. In this case, three basic parameters are sufficient for detection. Thus, one of the features of the system is reflexivity in detecting anomalies, that is, we force devices to independently analyze their behavior and make a decision about the presence of anomalies.

Keywords: Trust · Reflection · Anomaly Detection · Attacks · Denial of Service · Monitoring

1 Introduction

A Cyber-Physical System (CPS) provides a tight link between the cyber and physical domains by embedding cyber processes (e.g., communications, computing, or control) into physical devices. Intrusion detection systems are designed to detect anomalous behavior or unexpected activities in networks by automatically analyzing their behavior based on a given hypothesis and/or policies that are governed by the network's security

A. Alikhanov et al. (Eds.): APAMCS 2022, LNNS 702, pp. 430–440, 2023.
https://doi.org/10.1007/978-3-031-34127-4_42

rules [1]. The system monitors system configuration, data files, and/or network transmissions to check for an attack. Thus, this system is an important first step in preventing any covert/overt actions aimed at exploiting security vulnerabilities to crash or hijack the system. Such misuse can be defined as any undesirable action that could cause any harm in terms of performance or security of the entire group. Attacks exploit vulnerabilities in CPS that can result from network misconfiguration, implementation errors, design and/or protocol failures [2].

The issues of CPS safety monitoring are discussed in many works of the authors. To ensure the uninterrupted operation of cyber-physical systems, decision-making systems are used based on information received by the information security management system from the monitoring system. In this regard, we single out the monitoring system as an important step in the operation of the CPS information security management system. Modern information security management systems (ISMS) are devoted to a large number of research works, from architectural solutions [3–8] to the search for methods for solving security problems [9, 10]. The authors of the paper [11] highlight the problem of choosing the most appropriate set of methods for solving security problems for a particular CPS configuration. To solve this problem, the authors present a method for managing an adaptive information security monitoring system. The method consists in solving the problem of multiobjective discrete optimization under Pareto optimality conditions when the available data, methods, or external requirements change. An experimental study was carried out on the example of intrusion detection in a smart home system. As a result, the information security monitoring system acquires the property of adaptability to changing tasks and available data. As the number and complexity of cyberattacks have increased, machine learning (ML) has been actively used to detect cyberattacks and malicious activity. Cyber-Physical Systems (CPS) combined calculations with physical procedures. An embedded computer and network monitor and control physical processes, usually with feedback. Normally physical procedures affect computations, and ML approaches have been vulnerable to data poisoning attacks. Improving network security and achieving the reliability of network schemes defined by ML have been critical issues in the growth of attacks and the size of the CPS. In the paper [12] authors develop a new stochastic fractal search algorithm with a deep learning based intrusion detection system (SFSA-DLIDS) for the CPS cloud environment. The presented SFSA-DLIDS approach primarily implements a minimum-maximum data normalization approach to transform input data into a compatible format. Monitoring systems are necessary for the analysis and control of the CPS behavior. CPS s are associated with real-time constraints and physical phenomena that are usually not taken into account in typical information systems. In the paper presented by the authors of paper [13], the CPS-MT system is shown, aimed at creating a universal tool for monitoring CPS in real time from a security point of view.

Thus, the problem of security monitoring in CPS is quite relevant. The authors propose a large number of different solutions. At the same time, there is no specifics on which parameters are analyzed and whether they are universal. In addition, the concept of collecting information about parameters has not been fully developed. The CPS must quickly transfer the parameters to the monitoring system, but if the system is not flexible enough, fast and optimal, then collisions and additional loads on the CPS may occur. This study proposes a system for monitoring and detecting anomalies for CPS based

on the principles of trust, which allows you to verify the correctness of the system and detect possible anomalies. In our study, we focus on traffic analysis and analysis of the CPU operation, since these parameters are the most critical in the operation of the CPS itself.

2 Materials and Methods

2.1 Basic Concept of a Cyber-Physical System based on State Analysis

These layers are important for the features of the interaction of the system components and understanding which components interact directly and which through intermediaries. This understanding is important when modeling attack vectors on a system. As a rule, an attacker acts through communication channels if he does not have direct access to the system. Physical layer components may not have network interfaces but be connected to other components through Low-Level Management components.

$$CPS = \{HLC\} \cup \{NLC\} \cup \{LLC\} \cap \{PH\}, \tag{1}$$

where $\{HLC\}$ - set high-level management components, $\{NLC\}$- set of network layer components, $\{LLC\}$- set of Low-Level Management components, $\{PH\}$- set of Physical layer components.

Moreover, the components of the set of high-level, network and low-level management do not intersect, but the components of the low-level management and the physical layer can intersect, since their properties intersect. This will be proven below.

Physical layer components include sets of sensors $S = \{s_0, ..., s_n\}$, detectors $D = \{d_0, ..., d_j\}$, actuators $A = \{a_0, ..., a_m\}$, power supplies $PS = \{ps_0, ..., ps_i\}$ and other additional devices that ensure the functioning of the CPS. In this case, the number of elements of the set may differ. Thus, the set $\{PH\}$ has 4 subsets, and none of these subsets is a subset of the other. The characteristics of the system can be:

- indications of various sensors (cyber-physical parameters),
- state of cyber-physical objects.

$\{PH\}$ is characterized by a set of cyber-physical parameters $CP = \{cp_0, ..., cp_i\}$ that can be obtained from sensors. This set depends on the sets {S}, {D}, {A}, {PS}. In particular, the more sensors, transmitters, etc. are installed, the more data can be obtained from them, and the more cyber-physical parameters can be processed.

$$|CP_i| = |CPS_i| + |CPD_i| + |CPA_i| + |CPPS_i| \tag{2}$$

where $|CP_i|$ is the number of all elements in the finite set of the given CPS_i, CPS_i is the number of all cyber-physical parameters received from the sensor system, CPD_i is the number of all cyber-physical parameters received from sensors, CPA_i is the number of all cyber-physical parameters received from actuators, $CPPS_i$ is from the accumulator and other peripheral devices.

In the case of a cyber-physical system, a microcontroller can be used to control sensors, sensors, and actuators. Accordingly, the data from the sensors comes to a higher

level from the microcontroller. A microcontroller belongs to a set of microcontrollers $MC = \{mc_0, ..., mc_i\}$ that are a subset of the LLC (low-level control), so the following is true:

$$mc_{fc,i} \in \{LLC\} \tag{3}$$

Since the microcontroller essentially includes a set of sensors that are connected to it and from which it receives information that it transmits further, and it can also transmit information to actuators, it can be said that the sets of low-level control and physical level objects can intersect and an element such as a microcontroller belongs to both sets:

$$LLC \cap PH = \left\{mc_{fc,i} | mc_{fc,i} \in LLC, mc_{fc,i} \in PH\right\} \tag{4}$$

Accordingly, we will assume that the controller is also characterized by a set of cyber-physical parameters that it gives to a higher level. Moreover, it can receive cyber-physical parameters from several elements of the PH set at once. Thus, a vector of parameters is formed, which may include a set of cyber-physical parameters of the controller. Thus, the evaluation of trust in a cyber-physical system is reduced to an assessment of trust in the quality of changes in cyber-physical parameters.

2.2 Trust-Based Verification Method of CPS Operation

To determine the degree of confidence in the current state of the CPS, we define a set of states. The trusted state is such a state when the change in the CPS parameters does not exceed the allowable values and corresponds to the expected values, which are trustworthy and allow for the smooth operation of the CPS.

An untrusted state is a state when the change in the parameters of the CPS exceeds the allowable confidence intervals, or does not reach the minimum values, which leads to failures in the operation of the CPS. In this study, we have focused only on active malicious activities and attacks that can damage the integrity and availability of the system. Let's define a set of metrics that are used to assess the state of the CPS (Table 1).

Table 1. A set of metrics for monitoring the state of the system

Metric	Formula	Note
Reliability of the functions performed in the current state		
1. Confidence limits		
$CP_i^{\min}$ lower limit of confidence interval	$CP_{i,st}^{\min} = CP_{i,st} - \sigma_i^{st}$ $CP_{i,s_{n-1}}^{\min} = CP_{S_n-1} - \sigma_i$ $CP_{i,current}^{\min} = \overline{CP_i} - t \cdot \sigma_{omi}^{current}/\sqrt{n}$	$CP_{i,st}^{\min}$-minimum value in the presence of targets, CP_{S_n-1}- the value of the cyber-physical parameter from the previous state, $t * \sigma/\sqrt{n}$ - estimation accuracy, t – argument of the Laplace function, where $(t) = \frac{\alpha}{2}$, α - given reliability, σ_i- allowable deviation, $CP_{i,current}^{\min}$- the minimum value based on the collected parameters for the previous time intervals

(*continued*)

Table 1. (*continued*)

Metric	Formula	Note
$CP_i^{\max}$- upper limit of the confidence interval	$CP_{i,st}^{\max} = CP_{i,st} - \sigma_i^{st}$, $CP_{i,s_{n-1}}^{\max} = CP_{S_n-1} + \sigma_i$, $CP_{i,current}^{\max} = \overline{CP_i} + t \cdot \sigma_{omi}^{current}/\sqrt{n}$	$CP_{i,st}^{\max}$- the maximum value in the presence of target, $CP_{i,current}^{\max}$- the maximum value based on collected data for previous time intervals
2. Estimation of the probability of going beyond the confidence interval		
Cumulative function for the Poisson distribution -f_{pois}	$f_{pois}(CP_i\|\overline{CP_i}) = \sum_{j=1}^{CP_i^n} \frac{\overline{CP_i} CP_i e^{-\overline{CP_i}}}{CP_i!}$, $f_{pois,\min}(CP_i\|CP_{i,\min}) = \sum_{j=1}^{CP_i^n} \frac{CP_{i,\min}^{CP_i} e^{-CP_{i,\min}}}{CP_i!}$ $f_{pois,\max}(CP_i\|CP_{i,\max}) = \sum_{j=1}^{CP_i^n} \frac{CP_{i,\max}^{CP_i} e^{-CP_{i,\max}}}{CP_i!}$	The cumulative Poisson probability is related to the probability that the random Poisson frequency is greater than a given limit and less than a given upper limit $CP_{i,\max}$- upper redistribution of the value of the cyber-physical parameter, $CP_{i,\min}$- the lower limit of the cyber-physical parameter
The average value of the cyber-physical parameter in the range of the sliding window	$\overline{CP_i} = \frac{1}{n} \sum_{j=1}^{n} P_i \Delta w_{ij}$	n is the sample size, P_i is the values of the sample parameters, Δw is the sliding window for a given time interval of values, equal to n

Metric 1. Boundary of the confidence interval.

Boundary of the confidence interval in the presence of targets. Target indicators are those values of cyber-physical parameters that the cyber-physical system must achieve. In the case when the CPS operates in an autonomous mode, such indicators can be taken from the technical certificate, as well as during an expert assessment of normal indicators.

Boundary based on knowledge of CPS. The value of the parameter can be used to define the limit of the indicator, which is obtained in the idle state, when the system is not performing active actions, but is already enabled. Values for determining the boundary of the confidence interval can be taken from the previous state of the system operation.

Confidence interval bound based on previous values from the sample. If there is no input information about the normal performance of the system, as well as information about the reference behavior of the system, then it is possible to calculate the boundaries of the interval dynamically. To do this, it is necessary to build a confidence interval based on the previous behavior of the system. In this case, the confidence limits of the interval will be regulated by the standard deviation. In any mode of functioning of a cyber-physical system, which is included in the set of normal states of the system, the change in cyber-physical parameters should occur smoothly. Even if the growth of the function is observed, in order for it not to have a critical impact on the system, it must be smooth.

Metric 2. Reliability of performed functions in the current state.

This metric allows you to determine how much the current parameters of the cyber-physical system correspond to the given boundaries. That is, it is necessary to evaluate whether the running process goes beyond the allowable interval. Such an assessment is possible only for those parameters for which the normal or preset values are reliably known. The reliability of the functions performed is understood as the degree of confidence (or the probability that) in the i-th function or action of the process, which is described by a change in cyber-physical parameters such that their current change does not go beyond the confidence interval. Thus, this metric allows you to control that the process being executed does not go beyond the confidence limits.

To determine whether the system can be trusted, it is necessary to determine whether the current technological process of the CPS goes beyond the permissible range. To do this, the boundaries of the confidence interval are calculated and the degree of exceeding or reaching them is estimated. Thus, if values close to 1 are observed, there is a correspondingly high probability that the system does not correspond to the given boundaries.

3 Results

3.1 Software Module for Monitoring and Analyzing Trust in CPS

Figure 2 shows the general architecture for data collection, monitoring and determining the state of the CPS. One of the main modules of the project. Engaged in obtaining and normalizing all monitored cyber-physical parameters of the device according to certain algorithms. Since the module and its functionality is very extensive, it is divided into several subroutines: Subroutine for logging server errors; Subroutine for logging network sniffer errors; A subroutine for logging errors in monitoring the cyber-physical parameters of the device; Subroutine for logging errors during normalization and saving parameters; Subroutine for reading the configuration file. Designed to read the configuration file and initialize critical parameters for the module to work; TCP server initialization routine; Network sniffer initialization routine. Designed to track and monitor the state of the network; Subprogram for initialization of the module for monitoring the cyber-physical parameters of the device. It is designed to obtain the cyber-physical parameters of the device, for example: CPU load, CPU temperature, RAM load; Subroutine for initializing the normalization module and saving the received parameters. It is intended for constructing series of parameter values according to the given settings, normalizing the obtained series and providing normalized information for analysis on the corresponding module.

There are a collection of basic parameters that may be relevant for most CPSs: CPU load, CPU temperature, network traffic load (number of packets for each protocol), RAM load.

Transferring current device settings to monitoring website. The data for connection is obtained by the parameter collection module from the configuration file. Further, when accessing the API of the monitoring site, the current parameters of the device are transferred. The analysis module is designed to analyze normalized series in order to obtain confidence values. Recording the received rows of information in the database on the

device (low-level control). The monitoring module implemented using web technologies is located at the high management and receives data from the low management layer. Carrying out calculations at a low control level, on the one hand, loads the microprocessor or microcontroller, but it allows the low-level device to independently detect an anomaly and make trust decisions in a distributed network. Between the analysis module and the monitoring module, data is transmitted over the network, which is a vulnerability. If an attack is carried out on a communication channel, then the data will be lost and the system will not respond in time. In the proposed architecture, the monitoring module is only informative. All decisions are made by a distributed system consisting of small computers.

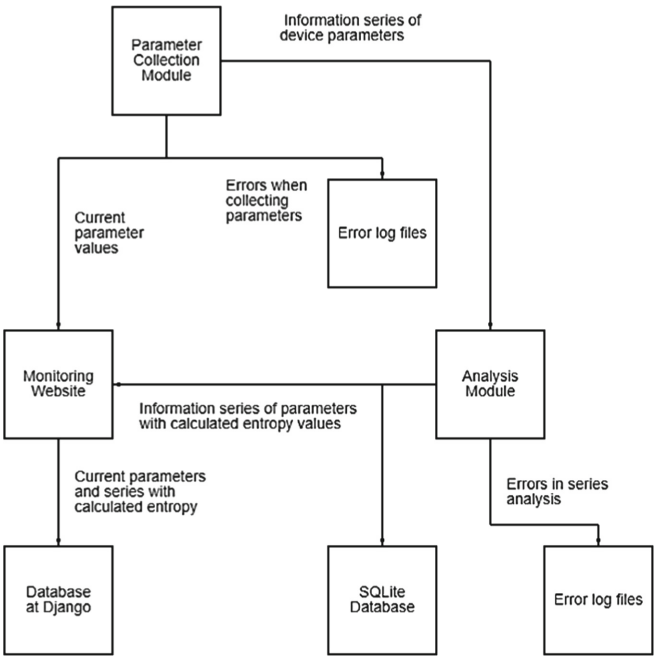


Fig. 2. Overall System Architecture

3.2 Results of Calculating the Probability of Values Going Beyond the Confidence Interval

An experimental study was carried out for the CPS model, which was built following the example of a full-scale model of an automated plant [14]. At the same time, 4 types of malicious impact were carried out: low-intensity TCP flood attack, medium-intensity TCP flood attack, high-intensity UDP flood attack and high-intensity ICMP flood attack [15–17]. The intensity of the attack was regulated by the speed and number of packets sent. Data collection and analysis modules are deployed on three low-level control devices: the device responsible for the human machine interface (HMI), the

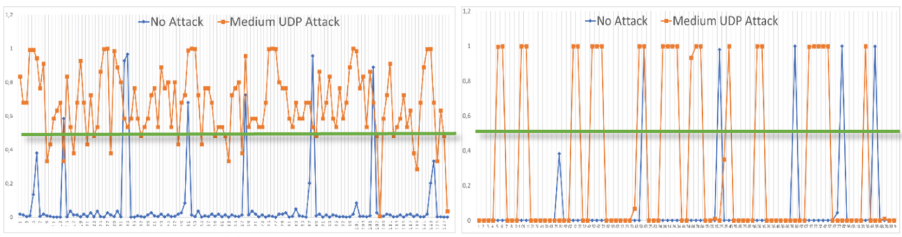
device with the Supervisory Control And Data Acquisition (SCADA) system installed, the device responsible for the control of the Programmable Logic Controller (PLC). The parameters of each of the devices change a little differently due to the fact that they perform different functions. Consider the results of calculations for the SCADA device, which are shown in Fig. 3. The SCADA system passively collects data. In this case, the operator can connect to the module through the web interface for monitoring. As can be seen from the figures, each attack affects each parameter of the SCADA system. With the exception of the RAM load indicator, this parameter did not change during the experimental study, so the graph for it is not shown. Next, consider the impact of attacks on the PLC system. The calculation results are shown in Fig. 4. Figures 3 and 4 show that even for a normal situation, there are single peaks for processor load and transmitted traffic. This is due to the fact that since the system controls automated production, according to the algorithm, control commands are transmitted automatically at certain periods.

Thus, single excesses can occur, and such a situation will be considered normal. The main condition is that the sequence of peaks does not exceed three. In the case of an attack, we see entire sequences of exceeding values.

4 Discussion

For the PLC, such peaks are observed more often, because it directly controls microcontrollers. In addition, the peaks are systemic and regular, which indicates the monotonicity of the process. Tracking the frequency of peaks can later become an indication of normal operation and one of the signs of normal behavior.

In addition, you can link CPU peaks to traffic load. It is with this that the resulting excess loads are associated. During the TCP flood attack, all three parameters were affected. This is because the protocol itself is quite resource-intensive and the process of maintaining the set values most affects the device. For the PLC, during the implementation of the TCP flood attack, unacceptable consequences were observed when the automated production did not work correctly, and the work could stop. At the same time, the node stopped transmitting data, it could only write them to the internal database. At the same time, with each attack, an excess of CPU indicators and a level of traffic



(a) (b)

Fig. 4. The result of calculating the probability of values going beyond the confidence interval (a) for the CPU utilization level (b) for the network traffic utilization level, where the parameter excess level is shown vertically, and the time interval is horizontal. The green horizontal line indicates the limit value.

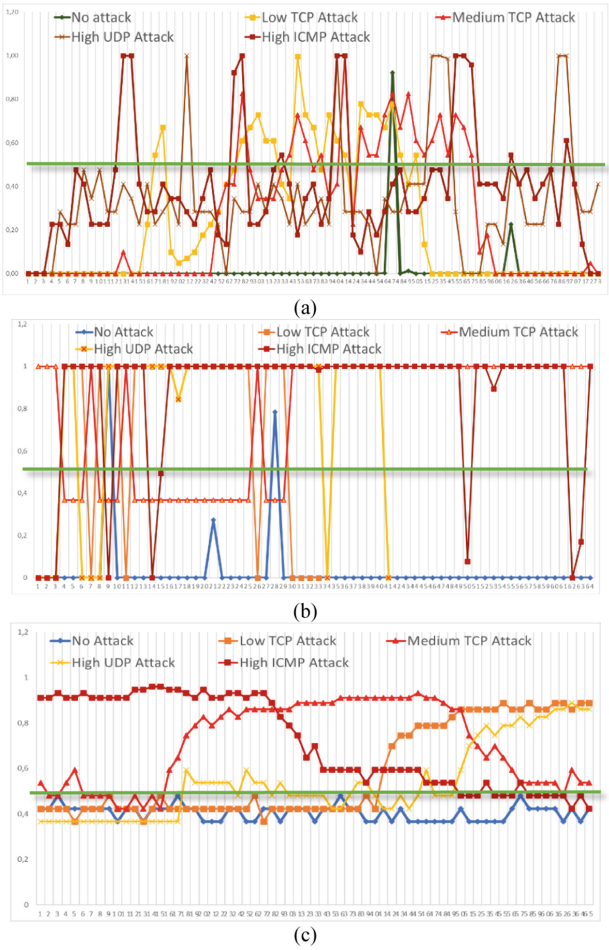


Fig. 3. The result of calculating the probability of values going beyond the confidence interval (a) for the CPU utilization level (b) for the network traffic utilization level (c) for the CPU temperature, where the parameter excess level is shown vertically, and the time interval is horizontal. The green horizontal line indicates the limit value.

congestion were observed. At the same time, the monitoring system could not receive correct data, so the graphs are not shown. However, by maintaining an internal database, incident detection becomes possible.

At the same time, during the UDP attack, the PLC flood reacted only to the CPU load and network traffic, other parameters did not change significantly. It should also be noted that the graphs show the total number of traffic as a summary parameter. When analyzing packets according to the protocols for the UDP flood attack, you can immediately see a significant excess of UDP packets compared to the normal state of operation. Thus, the detection becomes more accurate. In addition, if you track received and sent packets

separately, then the overshoot rate will increase for received packets compared to sent ones. These provisions will be explored in the future.

The ICMP flood attack of high intensity and the TCP attack of medium intensity had the greatest impact on the parameters of the SCADA system. This is due to the fact that SCADA does not exchange UDP traffic and does not have open ports that can be influenced. In this case, the PLC exchanges UDP traffic with microcontrollers and therefore the UDP flood attack has a significant impact. In one case or another, each of these attacks is detected by the analysis system successfully.

5 Conclusion

Thus, in this paper, the concept of CPS was considered from the point of view of functioning and the possibility of analyzing states. The relationship between the processes and levels of the CPS with the change in cyber-physical parameters is proved. Based on this evidence, a method for analyzing changes in cyber-physical parameters has been developed as a basis for detecting malfunctions in the system. The technique is based on computationally simple algorithms and allows to analyze the basic parameters that are typical for most CPS. These factors make it highly scalable and applicable to various types of CPS, despite the fragmentation and a large number of architectures. A distributed application architecture was developed for monitoring and analyzing trust in the CPS. The architecture, due to the distribution of calculations, allows continuous analysis of trust in the system locally on the control devices of low-level control. The calculation results show the possibility of detecting the consequences of the influences of denial-of-service attacks or CPS. In this case, three basic parameters are sufficient for detection. Thus, one of the features of the system is reflexivity in detecting anomalies, that is, we force devices to independently analyze their behavior and make a decision about the presence of anomalies.

Acknowledgments. The research was supported by the Council for Grants of the President of the Russian Federation at the expense of the scholarship of the President of the Russian Federation for young scientists and graduate students (Competition SP-2022) No. SP-858.2022.5 on the topic "Technology for ensuring cybersecurity of automated systems from active information attacks based on the principle of reflection".

References

1. Choi, S., Woo, J., Kim, J., Lee, J.Y.: Digital twin-based integrated monitoring system: korean application cases. Sensors **22**, 5450 (2022). https://doi.org/10.3390/s22145450
2. Yang, B., Xin, L., Long, Z.: An improved residual-based detection method for stealthy anomalies on mobile robots. Machines **10**, 446 (2022). https://doi.org/10.3390/machines10060446
3. Kotenko, I.V.: Primenenie tekhnologii upravleniya informaciej i sobytiyami bezopasnosti dlya zashchity informacii v kriticheski vazhnyh infrastrukturah. Trudy SPIIRAN Vyp **1**, 2–7 (2012)

4. Lavrova, D.S., Zaitseva, E.A., Zegzhda, D.P.: Approach to presenting network infrastructure of cyberphysical systems to minimize the cyberattack neutralization time. Autom. Control. Comput. Sci. **53**(5), 387–392 (2019). https://doi.org/10.3103/S0146411619050067
5. Stevens, M.: Security Information and Event Management (SIEM). In Proceedings of the NEbraska CERT Conference, Omaha, NE, USA, 9–11 August 2005. http://www.certconf.org/presentations/2005/files/WC4.pdf
6. Knapp, E.D., Langill, J.T.: Chapter 12-Security Monitoring of Industrial Control Systems. In: Eric, D., Knapp, J.T. (eds.) Industrial Network Security, 2nd ed., pp. 351–386. Syngress, New York (2015)
7. Lavrova, D.S.: Podhod k razrabotke SIEM-sistemy dlya Interneta veshchej. Probl. Inf. Bezopasnosti. Komp'yuternye Sist. **2**, 51–59 (2016)
8. Siddiqui, S., Khan, M.S., Ferens, K., Kinsner, W.: Fractal based cognitive neural network to detect obfuscated and indistinguishable internet threats. In: Proceedings of the 2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI*CC), Oxford, UK, 26–28 July 2017; pp. 297–308 (2017)
9. Wang, C., Wang, D., Xu, G., He, D.: Efficient privacy-preserving user authentication scheme with forward secrecy for industry 4.0. Sci. China Inf. Sci. **65**(1), 1–15 (2021). https://doi.org/10.1007/s11432-020-2975-6
10. Jiang, Y., Yin, S., Kaynak, O.: Data-driven monitoring and safety control of industrial cyber-physical systems: basics and beyond. IEEE Access **6**, 47374–47384 (2018)
11. Poltavtseva, M., Shelupanov, A., Bragin, D., Zegzhda, D., Alexandrova, E.: Key concepts of systemological approach to CPS adaptive information security monitoring. Symmetry **13**, 2425 (2021). https://doi.org/10.3390/sym13122425
12. Duhayyim, M.A., et al.: Evolutionary-based deep stacked autoencoder for intrusion detection in a cloud-based cyber-physical system. Appl. Sci. **12**, 6875 (2022). https://doi.org/10.3390/app12146875/
13. Thakur, S., Chakraborty, A., De, R., Kumar, N., Sarkar, R.: Intrusion detection in cyber-physical systems using a generic and domain specific deep autoencoder model. Comput. Electr. Eng. **91**, 107044 (2021)
14. Sauer, F., Niedermaier, M., Kiebling, S., et al.: LICSTER – a low-cost ICS security testbed for education and research. In: 6th International Symposium for ICS & SCADA Cyber Security Research (2019). https://doi.org/10.14236/ewic/icscsr19.1
15. Gamec, J., Basan, E., Basan, A., Nekrasov, A., Fidge, C., Sushkin, N.: An adaptive protection system for sensor networks based on analysis of neighboring nodes. Sensors **21**, 6116 (2021). https://doi.org/10.3390/s21186116
16. Basan, E., Basan, A., Nekrasov, A.: Method for detecting abnormal activity in a group of mobile robots. Sensors **19**, 4007 (2019). https://doi.org/10.3390/s19184007/
17. Basan, E., Basan, A., Makarevich, O.: Detection of anomalies in the robotic system based on the calculation of kullback-leibler divergence. In: 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2019, pp. 337-340 (2019). https://doi.org/10.1109/CyberC.2019.00064