DOI: 10.18721/JCSTCS.13304
УДК 681.518

APPLICATION OF CLASSIFIER SEQUENCES IN THE TASK OF STATE ANALYSIS OF INTERNET OF THINGS DEVICES

M.E. Sukhoparov[1], I.S. Lebedev[1], A.V. Garanin[2]

[1] St. Petersburg Institute for Informatics and Automation of RAS,
St. Petersburg, Russian Federation;

[2] New technologies LLC,
St. Petersburg, Russian Federation

Development of the industrial Internet concept dictates the need for identification and improvement of approaches, models, and methods for analyzing the state of the Internet of Things. Implementation of modern industrial, social, and household systems is impossible without the use of artificial intelligence methods in the machine-to-machine communication of individual elements, automatic data collection, analysis, and storage. The paper presents an approach to identifying the state of devices based on the application of classification technology, which implements compositions of independently trained algorithms processing time series, reflecting the functioning of elements during the implementation of processes. The application of the proposed solution allows parallel processing of information received from the device, which enables scaling. The developed approach was tested on time series sequences, obtained experimentally in different operating conditions, and processed by a sequence of classifiers. The paper presents the results of the probability estimate of erroneously classified states. The main advantages of the proposed solution are relatively small requirements to computational resources, simplicity of implementation, and the ability to scale by adding new classification algorithms.

Keywords: state analysis, Internet of Things, discriminant analysis, state monitoring, classification algorithm, Bayesian classifier, decision trees.

Citation: Sukhoparov M.E., Lebedev I.S., Garanin A.V. Application of classifier sequences in the task of state analysis of Internet of Things devices. Computing, Telecommunications and Control, 2020, Vol. 13, No. 3, Pp. 44–54. DOI: 10.18721/JCSTCS.13304

ПРИМЕНЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТЕЙ КЛАССИФИКАТОРОВ В ЗАДАЧЕ АНАЛИЗА СОСТОЯНИЯ УСТРОЙСТВ ИНТЕРНЕТА ВЕЩЕЙ

М.Е. Сухопаров[1], И.С. Лебедев[1], А.В. Гаранин[2]

[1] Санкт-Петербургский институт информатики и автоматизации РАН,
Санкт-Петербург, Российская Федерация;

[2] ООО "Новые технологии",
Санкт-Петербург, Российская Федерация

Развитие концепции промышленного Интернета обусловливает необходимость поиска и совершенствования подходов, моделей и методов анализа состояния элементов Интернета Вещей. Реализация современных индустриальных, промышленных, со-

циальных и бытовых систем невозможна без применения методов искусственного интеллекта межмашинного обмена отдельных элементов, автоматического сбора, анализа, хранения данных. В статье представлен подход к идентификации состояния устройств, основанный на использовании технологии классификации, реализующей композиции независимо обученных алгоритмов, обрабатывающих временные ряды, отражающих функционирование элементов во время выполнения процессов. Применение предлагаемого решения позволяет осуществлять параллельную обработку поступающей от устройства информации, что дает возможность масштабирования. Разработанный подход протестирован на последовательностях временных рядов, полученных экспериментальным путем в различных условиях функционирования, обработанных последовательностью классификаторов. Приведены результаты оценки вероятности ошибочно классифицированных состояний. Основными достоинствами предложенного решения являются относительно небольшие требования к вычислительным ресурсам, простота реализации, возможности по масштабированию путем добавления новых классифицирующих алгоритмов.

Ключевые слова: анализ состояния, Интернет Вещей, дискриминантный анализ, мониторинг состояния, алгоритм классификации, байесовский классификатор, деревья решений.

Ссылка при цитировании: Сухопаров М.Е., Лебедев И.С., Гаранин А.В. Применение последовательностей классификаторов в задаче анализа состояния устройств Интернета Вещей // Computing, Telecommunications and Control. 2020. Vol. 13. No. 3. Pp. 44–54. DOI: 10.18721/JCSTCS.13304

Introduction

Rapid development of the Internet of Things (IoT) concept based on wireless technologies, cloud computing, and distributed systems is a fundamental trend for information and cyber-physical systems. Implementation of modern industrial, social, and household systems is impossible without the use of artificial intelligence methods in the machine-to-machine communication of individual elements, automatic data collection, analysis, and storage. The common usage of various sensing elements and networks is focused on the solution of a large number of industrial and household tasks as well as social needs with minimum human participation. It brings undeniable advantages, on one hand, and determines the necessity to solve problems of state analysis and reveal functioning anomalies connected with failures, malfunctions, and incorrect process execution, on the other hand.

Interaction of IoT components with each other through the external environment predetermines the need to create various condition monitoring systems, which provide secure machine-to-machine communication, network access, data transfer, routing, intelligent data processing, etc. At the same time, it is necessary to take into account the dynamic development of information systems, assuming simultaneous use of both “old” and “new” devices of various developers, a large number of exchange protocols, and data processing in conditions of constant addition of new segments.

Production processes of typification and unification of separate computing elements, lack of proper physical security of the IoT elements that results from being outside the controlled zone of devices, the possibility of situations related to firmware upgrades, software updates, information collection, and access to standard devices allow the use of reverse engineering methods to improve condition monitoring in various operating modes.

Thus, the task of the condition monitoring system development for IoT devices appears where one of the directions presents the development of algorithms, models, methods of processing and analysis of the side channels data containing information about the ongoing process, as is proven in papers [5–8].

Execution of instructions and predefined sequences of actions is put in correspondence with the permissible values of functioning parameters that are registered through various channels. Detectable values form time series on the basis of which, with the use of methods of machine learning and statistical analysis, templates are determined, and normal and abnormal states are calculated.

Available approaches

During the operation of IoT devices, collisions may arise both at the level of the information system and a stand-alone device. For example, the introduction of software and hardware firmware upgrades, containing errors in the manufacture of household devices, such as routers, printers, and webcams, was associated with a number of situations that resulted in the loading of channels. Such loading limited the processes of receiving and transmitting information, as demonstrated in works [10–12].

To prevent such incidents, improvement and adaptation of models and methods of condition monitoring aimed at evaluating functionality and performance take place. These are based on the principles of statistical analysis, cause-effect analysis, transitions, and formation of precedent and event models, as described in papers [5–9].

Models based on statistics accumulate information about the functioning parameters in different modes and states, and later, with the purpose of the abnormal situation detection with the help of methods of neural networks, Markov models, machine learning, and others, tuples of features are processed, according to monograph [13] and paper [14].

For the detection of internal failures and malfunctions of functioning devices, monitoring programs are used that monitor execution of code segments and destabilizing situations such as buffer overflow, as demonstrated in works [8, 12–16].

Another direction is the processing of side channels, where the state is analyzed using time series of parameters reflecting changes in CPU utilization, internal memory usage, and intensity of message exchange, according to monograph [17] and papers [18–20].

The variety of IoT elements, a large number of objects, interaction protocols, data processing technologies, heterogeneity of formats, constantly changing architecture, and configuration changes may lead to various failures and malfunctions affecting the functioning parameters. Analysis of values of side channels (for example, electromagnetic and acoustic radiations, voltage, and power consumption) during the execution of various operations and instructions by the device makes it possible to implement external, relatively independent, not consuming computing resources of the IoT devices for the condition monitoring system [21–23].

Research objective

The development of IoT devices, software, and hardware is carried out using standard microchips and standard libraries of different manufacturers and developers. Methods of rapid development of software and hardware parts, which make it possible to use ready-made components of different manufacturers, lead to the devices starting to represent a “black box”.

IoT devices do not have vast computational resources and have a limited set of executable instructions, which permits consideration and identification of a relatively small number of states and their transitions.

During operation, the processes of IoT devices run in the dynamics, while many parameters are changed simultaneously.

The state of the external environment $u(t)$, caused by the receipt of control instructions to the device, reception and transmission of messages, and the element functioning, determined by the internal situations of data processing and implementation of computational algorithms, characterized by the transient characteristics $h(t)$, makes it possible to consider the device as a dynamic system. There are q inputs and d outputs, according to manual [11]; a control instruction and the values of the environment variables are supplied to the input. Then signals $S(t)$ (for example, indicating the resource load) appear at the output.

These signals are recorded by different sensors. The values of signals received through external channels contain the values of the noise component $\nu(t)$, determined by the properties of the measuring device, characteristics of the received signal, etc.

The state model of the IoT device is determined by the ratio presented in paper [12]:

$$\sum_{i=1}^{q}\sum_{j=1}^{d}\int_0^t u_i(t)h_{ij}(t-\tau)d\tau = \sum_{j=1}^{d}\int_0^t f(s_j(t-\tau),\nu_j(t-\tau))d\tau, \quad (1)$$

where q is the number of source channels; h are transient responses of the i-th channel for the j-th channel, registering sensor values received through the channel; f is the function of measured values.

At discrete instants of time of device operation $t_0, t_1, ..., t_n$, registration of vectors of numerical sequences takes place. Values $X(t)$ reflect data received from the sensors, containing a mixture of wanted signal $S(t)$ and the noise expressed by the parameter $\nu(t)$:

$$X(t) = \mathrm{F}\left[S(t), v(t)\right], \quad (2)$$

where vector X represents the result of mixed, mutually independent signals $S(t)$ with distortion of the noise component $\nu(t)$. Vector X represents a time series of values received from recording devices.

Vectors $X_1, X_2, ..., X_n$ reflect the process behavior in multidimensional coordinate space and define a set of states Z. The states are separated by a set of classes C, where the subsets are divided into dangerous C_1 and safe C_2 states.

Thus, there is a labeled finite training set:

$$X = \left\{(x_{11},\ ...,\ x_{n1}),\ (x_{12},\ ...,\ x_{n2}),\ ...,\ (x_{1m},\ ...,\ x_{nm})\right\}. \quad (3)$$

It is necessary to build a classification algorithm a of an input vector X_i for the representation $Z \to C$.

Proposed approach

The labeled training set contains values of time series from recording devices in predefined states and modes of operation. The known states $\{z_1, ..., z_l\} \in Z$ are defined only for the objects of the observed sequences $\{(x_{11}, ..., x_{n1}),(x_{12}, ..., x_{n2}), ..., (x_{1m}, ..., x_{nm})\}$.

From the investigated IoT device, a random vector function $X(t) = f(S(t), v(t))$ is observed in the interval $t_0 \le t \le T$, where time series $x_i = X(t_i)$ is registered at discrete instants of time $t_0, t_1, ..., t_k$.

A set of state classes $C = \{c_0, c_1, ..., c_n\}$ has been determined, in one of which at the discrete instant of time t_j the system can be located.

There are k classifiers a_i, $I = 1, ..., k$ trained independently of each other; X – a variety of feature sets; $a_i(x_i) \to c_j \in C$ – response of the i-th classifier; $\{P_i(c_j \mid x_i)\}_{j=0}^{n}{}_{k}$ – a posteriori probability for the i-th classifier after training; $w_i = \frac{1}{k}$ – weighting factors; $a(x) = \arg\max_{j=0,...,n} \sum_{i=0}^{k} w_i P_i(c_j \mid x_i)$ – the general classifier.

Such models can be trained independently of each other, which makes it possible to parallelize processes. The proposed approach to state identification distinguishes itself by the use of the classification technology, which implements compositions of independently trained algorithms. These algorithms process time series and reflect the functioning of the device during the process execution. It makes it possible to determine the device state without increasing the volume of the stored information.

Experiment

The analysis of the above approach was carried out based on the experiment, during which the state determined by the data processing algorithm of the computational node was identified. Time series reflecting

computational resource load recorded by the monitoring program were used as input data. The schematic course of the experiment is presented in Fig. 1.

Various algorithms were run on the computing device. Only background processes were functioning in the state Z_1. In the second case, node C acted as a transit node, transmitting the incoming information without processing (state Z_2). In the third situation (state Z_3), besides the processes of receiving and transmitting, processes of searching for predetermined information were additionally carried out (Fig. 2–5).

In the course of the experiment, the classification algorithms a_j of the input vector X_i for the representation $Z \to C$ were considered. Operating classifiers $\{a_1, a_2, ..., a_k\} \in k = 4$ (Naive Bayes classifier, decision trees, discriminant mining, and k-nearest neighbor algorithm), trained independently of each other, produced sequences of results $Z = \{(z_{a_1}^{c_0}, z_{a_1}^{c_1}, ..., z_{a_1}^{c_n}), (z_{a_2}^{c_0}, z_{a_2}^{c_1}, ..., z_{a_2}^{c_n}), ..., (z_{a_j}^{c_0}, z_{a_j}^{c_1}, ..., z_{a_j}^{c_n}), ..., (z_{a_k}^{c_0}, z_{a_k}^{c_1}, ..., z_{a_k}^{c_n})\}$.

The resulting class c_i of the state z_i predicted by each model is determined by averaging the values of the calculated probabilities:

$$a_{c_i} = \frac{1}{K}\sum_{k=1}^{K} w_k a_k(x_i).$$

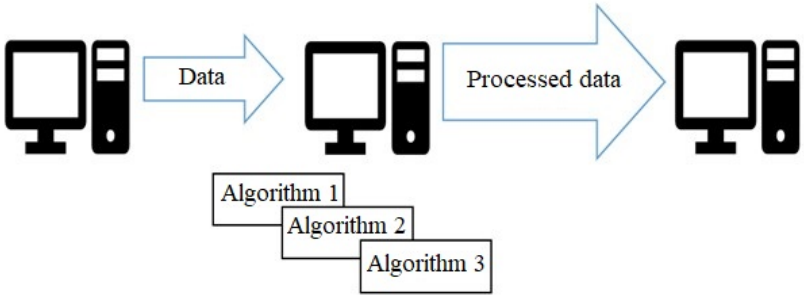

Fig. 1. Schematic course of the experiment

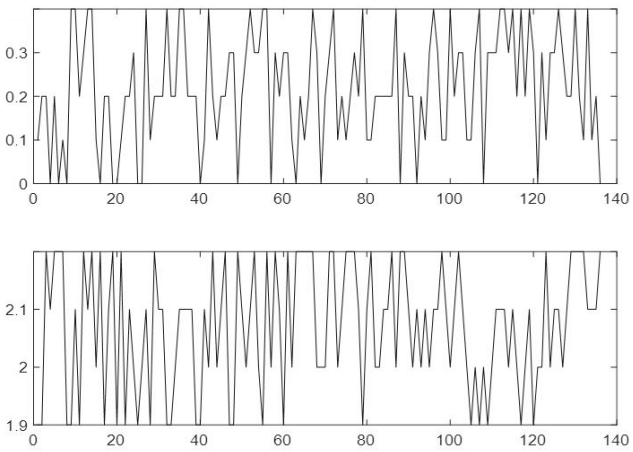

Fig. 2. Example of a sample of resource loading, expressed in percent (from top to bottom – network and processor, respectively), from time samples (time reports from 0 to 140) for the state Z_1

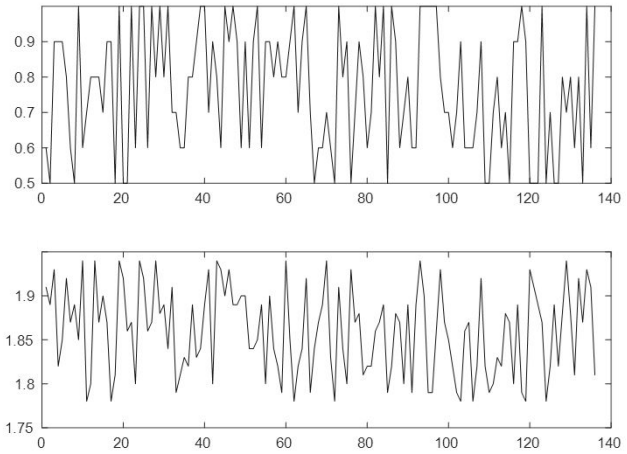

Fig. 3. Example of a sample of resource loading, expressed in percent (from top to bottom – network and processor, respectively), from time samples (time reports from 0 to 140) for the state Z_2

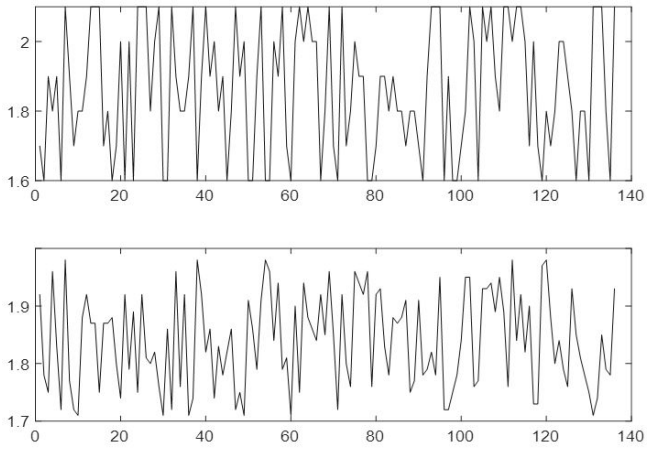

Fig. 4. Example of a sample of resource loading, expressed in percent (from top to bottom – network and processor, respectively), from time samples (time reports from 0 to 140) for the state Z_3

Table 1 presents the probabilities of erroneous classification obtained as the result of applying several "weak" classifiers trained in advance on the labeled sample of classifiers a_{c_i}: the Naive Bayes classifier, decision trees, discriminant mining, and *k*-nearest neighbor algorithm.

Table 1

Probability of erroneously classified sample values

Classifiers	Cluster 1	Cluster 2	Cluster 3	Total for sample
Naive Bayes classifier	0.18	0.02	0.02	0.07
Discriminant mining	0.16	0.02	0.02	0.07
Decision tree	0.08	0.04	0.04	0.05
k-nearest neighbor algorithm	0.2	0.08	0.06	0.11

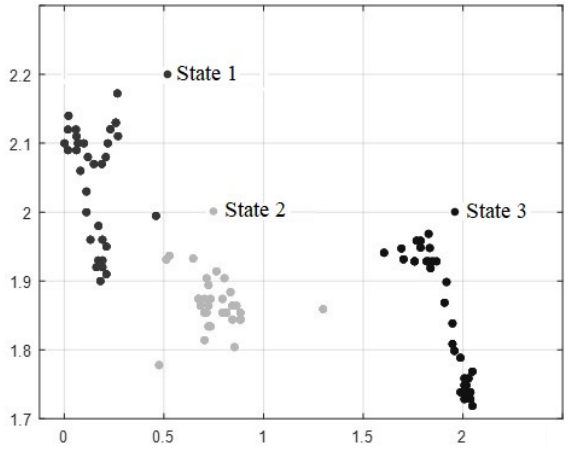

Fig. 5. Results of states on two-dimensional coordinate axes

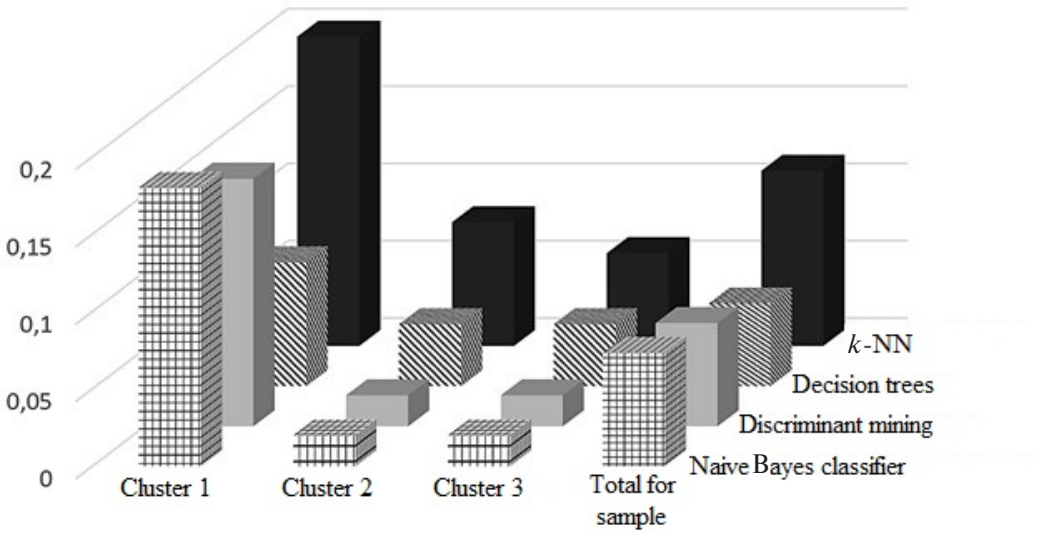

Fig. 6. Probabilities of erroneous classification

Fig. 6 presents a visualization of the probability estimate of erroneous classification.

The overall accuracy of the approach on the obtained experimental data for the case of full classification amounted to 0.93. At the same time, it should be noted that the data were not pre-processed or cleaned from noise, and the sampling rate of the obtained values was relatively low.

Thus, the proposed approach allows us to determine the class of the current state. The presented solution can be used as a theoretical basis for the integration of machine learning methods in the state analysis of the information security of IoT devices.

Conclusions

Analysis of a large number of different dynamically changing indicators in order to determine the states of IoT devices represents a time-consuming process that requires automation.

The heterogeneous characteristics of sequences received from recording devices in different modes of operation are unbalanced, and they have “emissions” that cannot always be separately identified by

different classifiers in a correct manner. Application of a sequence of different classifiers has an impact on the results of the method and makes it possible to avoid detailed analysis of possible hidden patterns, deregulation, and correlation of sequences.

The proposed approach is focused on using several classifiers, which produce a response independently from each other and average the error by "collective voting".

The use of classifiers in a parallel mode of processing of incoming sequences allows to reduce the processing time when determining the current state class.

The main limitation of the proposed approach is the necessity to select synchronized time series from recording devices, and in case of averaging – the lengths of the considered intervals.

The main advantages of the proposed approach are relatively small requirements to computational resources, simplicity of its implementation, and the possibility of scaling by adding new classifiers.

REFERENCES

1. **Farwell J.P., Rohozinski R.** Stuxnet and the Future of Cyber War. *Survival*, 2011, Vol. 53, No. 9, Pp. 23–40. DOI: 10.1080/00396338.2011.555586

2. **Yeung D.Y., Ding Y.** Host-based intrusion detection using dynamic and static behavioral models. *Pattern Recognition*, 2003, Vol. 36, Pp. 229–243. DOI: 10.1016/S0031-3203(02)00026-2

3. **Igure V., Laughter S., Williams R.** Security issues in SCADA networks. *Computers & Security*, 2006, Vol. 25, No. 7, Pp. 498–506. DOI: 10.1016/j.cose.2006.03.001

4. **Semenov V.V., Lebedev I.S., Sukhoparov M.Ye.** Identifikatsiya sostoyaniya informatsionnoy bezopasnosti bespilotnykh transportnykh sredstv s ispolzovaniyem iskusstvennykh neyronnykh setey [State identification of information security of unmanned vehicles using artificial neural networks]. *Metody i tekhnicheskiye sredstva obespecheniya bezopasnosti informatsii: Materialy XXVIII nauchno-tekhnicheskoy konferentsii* [*Proc. of the 28th Scientific and Technical Conf. on Methods and Information Security Engineering*], 2019, No. 28, Pp. 46–47. (rus)

5. **Zikratov I.A., Zikratova T.V., Lebedev I.S.** Doveritelnaya model informatsionnoy bezopasnosti multiagentnykh robototekhnicheskikh sistem s detsentralizovannym upravleniyem [Trusted model of information security of multi-agent robotics systems with decentralized control]. *Nauchno-Tekhnicheskiy Vestnik Informatsionnykh Tekhnologiy, Mekhaniki i Optiki* [*Scientific and Technical Journal of Information Technologies. Mechanics and Optics*], 2014, No. 2 (90), Pp. 47–52. (rus)

6. **Gao D., Reiter M., Song D.** Beyond output voting: Detecting compromised replicas using HMM-based behavioral distance. *IEEE Transactions on Dependable and Secure Computing*. 2009, Vol. 6, No. 2, Pp. 96–110. DOI: 10.1109/TDSC.2008.39

7. **Devesh M., Kant A.K., Suchit Y.R., Tanuja P., Kumar S.N.** Fruition of CPS and IoT in context of Industry 4.0. *Intelligent Communication, Control and Devices. Advances in Intelligent Systems and Computing*, 2020, Vol. 989, Pp. 367–375.

8. **Bevir M.K., O'Sullivan V.T., Wyatt D.G.** Computation of electromagnetic flowmeter characteristics from magnetic field data. *Journal of Physics D Applied Physics*, 1981, Vol. 14, No. 3, Pp. 373–388. DOI: 10.1088/0022-3727/14/3/007

9. **Semenov V.V., Lebedev I.S., Sukhoparov M.E., Salakhutdinova K.I.** Application of an autonomous object behavior model to classify the cybersecurity state. *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, 2019, Pp. 104–112. DOI: 10.1007/978-3-030-30859-9_9

10. **Semenov V.V., Lebedev I.S., Sukhoparov M.Ye.** Podkhod k klassifikatsii sostoyaniya informatsionnoy bezopasnosti elementov kiberfizicheskikh sistem s ispolzovaniyem pobochnogo elektromagnitnogo izlucheniya [Approach to classification of the information security state of elements of cyber physical systems by applying side electromagnetic radiation]. *Nauchno-Tekhnicheskiy Vestnik Informatsionnykh Tekhnologiy, Mekhaniki i Optiki* [*Scientific and Technical Journal of Information Technologies, Mechanics and Optics*], 2018, No. 1, Pp. 98–105. (rus)

11. **Soshnikova L.A, Tamashevich V.N., Ushbe G., Shefer M.** *Mnogomernyj statisticheskij analiz v ekonomike* [*Multivariate statistical analysis in economics*]. Moscow: UNITI – Dana Publ., 1999, 598 p. (rus)

12. **Sukhoparov M.Ye., Semenov V.V., Salakhutdinova K.I., Lebedev I.S.** Vyyavleniye anomalnogo funktsionirovaniya ustroystv «Industrii 4.0» na osnove povedencheskikh patternov [Detection of abnormal functioning of Industry 4.0 devices based on behavioral patterns]. *Problemy Informatsionnoy Bezopasnosti. Kompyuternyye Sistemy* [*Information Security Problems. Computer Systems*], 2020, No. 1 (41), Pp. 96–102. (rus)

13. **Bendat D., Pirsol A.** *Primeneniye korrelyatsionnogo i spektralnogo analiza* [*Application of correlation and spectral analysis*]. Moscow: Mir Publ., 1983. 312 p. (rus)

14. **Zasov V.A., Tarabardin M.A., Nikonorov Ye.N.** Algoritmy i ustroystva dlya identifikatsii vkhodnykh signalov v zadachakh kontrolya i diagnostiki dinamicheskikh obyektov [Algorithms and devices for identification of input signals in tasks of control and diagnostics of dynamic objects]. *Vestnik Samarskogo Gosudarstvennogo Aerokosmicheskogo Universiteta* [*Journal of Samara University. Aerospace and Mechanical Engineering*], 2009, No. 2, Pp. 115–123. (rus)

15. **Lockhart D.J.** Expression monitoring by hybridization to high-density oligonucleotide arrays. *Natural Biotechnol*, 1996, Vol. 14, Pp. 1675–1680. DOI: 10.1038/nbt1296-1675

16. **Golub T.R.** Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 1999, Vol. 286, Pp. 531–537. DOI: 10.1126/science.286.5439.531

17. **Anderberg M.R.** *Cluster analysis for applications*. New York: Academic Press, 1976. 376 p.

18. **Dembele D., Kastner P.** Fuzzy C-means method for clustering microarray data. *Bioinformatics*, 2003, Vol. 19, No. 8, Pp. 973–980. DOI: 10.1093/bioinformatics/btg119

19. **Rousseeuw J.P.** Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 1987, Vol. 20, Pp. 53–65. DOI: 10.1016/0377-0427(87)90125-7

20. **Whitfield M.L.** Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Molecular Biology of the Cell*, 2002, Vol, 13, No. 6, Pp. 1977–2000. DOI: 10.1091/MBC.02-02-0030

21. **Wyglinski A.M., Huang X., Padir T., Lai L., Eisenbarth T.R., Venkatasubramanian K.** Security of autonomous systems employing embedded computing and sensors. *IEEE Micro*, 2013, Vol. 33 (1), Art. No. 6504448, Pp. 80–86. DOI: 10.1109/MM.2013.18

22. **Nikolaevskiy I., Lukyanenko A., Polishchuk T., Polishchuk V.M., Gurtov A.V.** isBF: scalable in-packet bloom filter based multicast. *Computer Communications*, 2015, Vol. 70, Pp. 79–85. DOI: 10.1145/2480362.2480484

23. **Gupta H., Sural S., Atluri V., Vaidya J.** A side-channel attack on smartphones: deciphering key taps using built-in microphones. *Journal of Computer Security*, 2018, Vol. 26 (2), Pp. 255–281. DOI: 10.3233/JCS-17975

Received 28.08.2020.

СПИСОК ЛИТЕРАТУРЫ

1. **Farwell J.P., Rohozinski R.** Stuxnet and the Future of Cyber War // Survival. 2011. Vol. 53. No. 9. Pp. 23–40. DOI: 10.1080/00396338.2011.555586

2. **Yeung D.Y., Ding Y.** Host-based intrusion detection using dynamic and static behavioral models // Pattern Recognition. 2003. Vol. 36. Pp. 229–243. DOI: 10.1016/S0031-3203(02)00026-2

3. **Igure V., Laughter S., Williams R.** Security issues in SCADA networks // Computers & Security. 2006. Vol. 25. No. 7. Pp. 498–506. DOI: 10.1016/j.cose.2006.03.001

4. **Семенов В.В., Лебедев И.С., Сухопаров М.Е.** Идентификация состояния информационной безопасности беспилотных транспортных средств с использованием искусственных нейронных сетей // Методы и технические средства обеспечения безопасности информации: Матер. XXVIII науч.-техн. конф. 2019. № 28. С. 46–47

5. **Зикратов И.А., Зикратова Т.В., Лебедев И.С.** Доверительная модель информационной безопасности мультиагентных робототехнических систем с децентрализованным управлением // Научно-технический вестник информационных технологий, механики и оптики. 2014. № 2 (90). С. 47–52.

6. **Gao D., Reiter M., Song D.** Beyond output voting: Detecting compromised replicas using HMM-based behavioral distance // IEEE Transactions on Dependable and Secure Computing. 2009. Vol. 6. No. 2. Pp. 96–110. DOI: 10.1109/TDSC.2008.39

7. **Devesh M., Kant A.K., Suchit Y.R., Tanuja P., Kumar S.N.** Fruition of CPS and IoT in context of Industry 4.0. // Intelligent Communication, Control and Devices. Advances in Intelligent Systems and Computing. 2020. Vol. 989. Pp. 367–375.

8. **Bevir M.K., O'Sullivan V.T., Wyatt D.G.** Computation of electromagnetic flowmeter characteristics from magnetic field data // J. of Physics D Applied Physics. 1981. Vol. 14. No. 3. Pp. 373–388. DOI: 10.1088/0022-3727/14/3/007

9. **Semenov V.V., Lebedev I.S., Sukhoparov M.E., Salakhutdinova K.I.** Application of an autonomous object behavior model to classify the cybersecurity state // Internet of Things, Smart Spaces, and Next Generation Networks and Systems. 2019. Pp. 104–112. DOI: 10.1007/978-3-030-30859-9_9

10. **Семенов В.В., Лебедев И.С., Сухопаров М.Е.** Подход к классификации состояния информационной безопасности элементов киберфизических систем с использованием побочного электромагнитного излучения // Научно-технический вестник информационных технологий, механики и оптики. 2018. № 1. С. 98–105.

11. **Сошникова Л.А., Тамашевич В.Н., Усбе Г., Шефер М.** Многомерный статистический анализ в экономике. М.: ЮНИТИ – Дана, 1999. 598 с.

12. **Сухопаров М.Е., Семенов В.В., Салахутдинова К.И., Лебедев И.С.** Выявление аномального функционирования устройств «Индустрии 4.0» на основе поведенческих паттернов // Проблемы информационной безопасности. Компьютерные системы. 2020. № 1 (41). С. 96–102.

13. **Бендат Д., Пирсол А.** Применение корреляционного и спектрального анализа. М.: Мир, 1983. 312 с.

14. **Засов В.А., Тарабардин М.А., Никоноров Е.Н.** Алгоритмы и устройства для идентификации входных сигналов в задачах контроля и диагностики динамических объектов // Вестник Самарского государственного аэрокосмического университета. 2009. № 2. С. 115–123.

15. **Lockhart D.J.** Expression monitoring by hybridization to high-density oligonucleotide arrays // Natural Biotechnol. 1996. Vol. 14. Pp. 1675–1680. DOI: 10.1038/nbt1296-1675

16. **Golub T.R.** Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring // Science. 1999. Vol. 286. Pp. 531–537. DOI: 10.1126/science.286.5439.531

17. **Anderberg M.R.** Cluster analysis for applications. New York: Academic Press, 1976. 376 p.

18. **Dembele D., Kastner P.** Fuzzy C-means method for clustering microarray data // Bioinformatics. 2003. Vol. 19. No. 8. Pp. 973–980. DOI: 10.1093/bioinformatics/btg119

19. **Rousseeuw J.P.** Silhouettes: A graphical aid to the interpretation and validation of cluster analysis // J. of Computational and Applied Mathematics. 1987. Vol. 20. Pp. 53–65. DOI: 10.1016/0377-0427(87)90125-7

20. **Whitfield M.L.** Identification of genes periodically expressed in the human cell cycle and their expression in tumors // Molecular Biology of the Cell. 2002. Vol. 13. No. 6. Pp. 1977–2000. DOI: 10.1091/MBC.02-02-0030

21. **Wyglinski A.M., Huang X., Padir T., Lai L., Eisenbarth T.R., Venkatasubramanian K.** Security of autonomous systems employing embedded computing and sensors // IEEE Micro. 2013. Vol. 33 (1). Art. No. 6504448. Pp. 80–86. DOI: 10.1109/MM.2013.18

22. **Nikolaevskiy I., Lukyanenko A., Polishchuk T., Polishchuk V.M., Gurtov A.V.** isBF: scalable in-packet bloom filter based multicast // Computer Communications. 2015. Vol. 70. Pp. 79–85. DOI: 10.1145/2480362.2480484

23. **Gupta H., Sural S., Atluri V., Vaidya J.** A side-channel attack on smartphones: deciphering key taps using built-in microphones // J. of Computer Security. 2018. Vol. 26 (2). Pp. 255–281. DOI: 10.3233/JCS-17975

Статья поступила в редакцию 28.08.2020.

THE AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Sukhoparov Mikhail E.
Сухопаров Михаил Евгеньевич
E-mail: sukhoparovm@gmail.com

Lebedev Ilya S.
Лебедев Илья Сергеевич
E-mail: isl_box@mail.ru

Garanin Anton V.
Гаранин Антон Владимирович
E-mail: anton.v.garanin@yandex.ru