

UDC 004.056.53

Doi: 10.31772/2587-6066-2020-21-4-466-477

For citation: Anashkin E. V., Zhukova M. N. Algorithmic and software of the system profiling the actions of users of the information system. *Siberian Journal of Science and Technology*. 2020, Vol. 21, No. 4, P. 466–477. Doi: 10.31772/2587-6066-2020-21-4-466-477

Для цитирования: Анашкин Е. В., Жукова М. Н. Алгоритмическое и программное обеспечение системы профилирования действий пользователей информационной системы // Сибирский журнал науки и технологий. 2020. Т. 21, № 4. С. 466–477. Doi: 10.31772/2587-6066-2020-21-4-466-477

ALGORITHMIC AND SOFTWARE OF THE SYSTEM PROFILING THE ACTIONS OF USERS OF THE INFORMATION SYSTEM

E. V. Anashkin*, M. N. Zhukova

Reshetnev Siberian State University of Science and Technology
31, Krasnoyarskii rabochii prospekt, Krasnoyarsk, 660037, Russian Federation

*E-mail: a.yegoriy@gmail.com

The paper describes the software of the system for profiling the actions of users of the information system. This profiling system is aimed at solving the problem of trust in users of information systems. The system should regulate access to protected resources by analyzing user behavior. The algorithmic component of the system is represented by a user behavior model and a general system operation algorithm. The user behavior model is based on the apparatus of Markov chains. Software implementation allows in practice to obtain the foundations of the proposed approach to work. At the development stages, the choice of software architecture is carried out. The client-server architecture was chosen as a reasonable decision. The software component of the user activity profiling system consists of five separate software modules. At the end of development, a brief testing of the components is carried out. The novelty of this work lies in the proposal of an approach that uses the profiling of user actions as an additional determining factor in managing access to objects, as a way to strengthen the basic measures "Controlling access of subjects to access objects" in the order system of FSTEC of Russia.

Keywords: user behavior analysis, access control, UBA, information security software.

АЛГОРИТМИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СИСТЕМЫ ПРОФИЛИРОВАНИЯ ДЕЙСТВИЙ ПОЛЬЗОВАТЕЛЕЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Е. В. Анашкин*, М. Н. Жукова

Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева
Российская Федерация, 660037, г. Красноярск, просп. им. газ. «Красноярский рабочий», 31

*E-mail: a.yegoriy@gmail.com

В работе приводится описание алгоритмического и программного обеспечения системы профилирования действий пользователей информационной системы. Данная система профилирования направлена на решение проблемы с доверием к пользователям информационных систем. Система должна регулировать доступ к защищаемым ресурсам путем анализа поведения пользователей. Алгоритмическая составляющая системы представлена моделью поведения пользователя и общим алгоритмом работы системы. Модель поведения пользователя строится на базе аппарата марковских цепей. Программная реализация позволяет на практике получить подтверждение работоспособности предлагаемого подхода. На этапах разработки осуществляется выбор архитектуры программного обеспечения. В качестве обоснованного решения выбрана архитектура типа «клиент – сервер». Программная составляющая системы профилирования действий пользователей состоит из пяти отдельных программных модулей. В конце разработки проводится краткое тестирование работы компонентов. Новизна данной работы заключается в предложении подхода, который использует профилирование действий пользователей как дополнительный определяющий фактор при управлении доступом к объектам, как способ усиления базовых мер «Управление доступом субъектов к объектам доступа» в системе приказов ФСТЭК России.

Ключевые слова: анализ поведения пользователей, контроль доступа, ПО СЗИ, UBA.

Introduction. The development of information security in information systems offers various options for solution to the issue of trust in system users.

In some information systems, the issue of trust in users is limited to the password authentication procedure. The system trusts a user if the user knows the password.

However, there are several cases where password authentication as a trust criterion is not enough. The first case is password theft or brute-force attack. The use of disgraced accounts is the second most popular method used by cybercriminals to conduct attacks aimed at stealing confidential data [1]. The second case is the presence of an internal attacker or insider on the information system. According to the study, 90 % of information leaks in Russia occur with an internal attacker (63.5 % in the world) [2].

Another option can be called a “communist” approach, which includes creating white and black lists: approved and prohibited programs, web resources, actions, etc. However, in the current conditions of development of information technologies, when new programs are being developed every day, new malicious resources are being created, and their tools are modified, it is problematic to create a full-fledged and comprehensive list. The same applies to business, the organizations themselves cannot always determine and fix the set of software, services and resources that they need to implement business processes today and tomorrow [3]. Even if a program is on the white list, it may have a vulnerability that could be exploited by a malefactor to circumvent restrictions of prohibitive policy. Thus, this approach is not sufficient to provide the environment for trust.

Another approach, called “ZeroTrust” [4], is based on distrust of the system components to each other, as well as the system does not trust the user. In this key, to confirm trust, it is proposed to use profiling of user actions in the system.

Profiling employee actions will make it possible to quickly identify all atypical actions that go beyond the usual behavior. This will help to identify the guilty party aiming at punishment or additional training. This will also allow detecting cases of the use of disgraced accounts, because, in order to circumvent behavioral analysis, a malefactor must accurately simulate all the actions of the employee who owned the disgraced account.

The scientific novelty of the work lies in the fact that among the scientific works that have appeared over the past 10 years, one can find a considerable number that affect user behavior for solving a different spectrum of tasks. To ensure enterprise protection, profiling of user actions occurs within the framework of access to the server [5]. In the paper [6], to protect cloud services, it is proposed to use fuzzy logic to calculate the level of trust in users based on their behavior. Another area of application for analysis of user behavior is security auditing and anomaly detection in databases [7]. In this paper a method of a one-class support vector machine is applied. Another work [8] is aimed at protecting information stored in databases. Its goal is to automate the adjustment of security policies and rules for accessing database tables. To do this, the operating rules are used, initially set by the security administrator to users, and user behavior is determined by access patterns. The following work analyzes, using Markov chains, the sequences of Unix-systems of variable length, which are entered by the user in the terminal server [9]. Also, user behavior is added to malware

detection [10]: wherefrom users download files, and wherefrom they lunch files, along with what file they lunch, that affects the calculation of the host's security level, as well as the accuracy of the detection system.

Nevertheless, despite the available variety of activities related to behavior profiling, there is a shortage of works that would fully cover user behavior in desktop operating systems for workstations. In this connection, this work proposes its own view of the system for profiling user actions in the OS of the Windows family.

The novelty of this work lies in the proposal of an approach that uses the profiling of user actions as an additional determining factor in managing access to objects, as a way to strengthen the basic measures “Controlling access of subjects to access objects” in the order system of FSTEC of Russia No. 17, 21, 31, 239 [11–14].

Description of the system profiling the actions of users. User actions profiling system (hereinafter referred to as UAPS) can have a different architecture and directivity, which depend on the protected object that is an information system (hereinafter referred to as IS). For example, for cloud ISs, UAPS is needed that supports a cloud architecture, and UAPS with an autonomous architecture is needed for an IS consisting of a single AWP.

In accordance with the official statistics for the regions of the Russian Federation for 2019, half of the employees at the enterprises are equipped with personal computers [15]. Therefore, the largest group is made up of ISs with a multiuser environment.

Tracking the actions of employees of the organization who are users of the IS, each of whom works at a separate personal computer, requires the installation of an UAPS on each of these computers. However, the computing power of these computers may either not meet the system requirements for installing a separate autonomous copy of the UAPS on them, or significantly affect the performance of the system and make it difficult to perform work duties and tasks.

To reduce system requirements and increase ergonomic parameters, the architecture of the “client-server” type was chosen as the architecture for the UAPS software. In this case, there is a separation of the functional capabilities of the UAPS into two components. The minimum required functionality is implemented in the client part. This reduces the system requirements for the personal computers of employees. Functions requiring high computing resources are moved to the server side. The server part allows performing centralized computing operations.

The UAPS architecture is shown in fig. 1

The client part consists of two mandatory components:

- agent programs;
- file system drivers.

The agent program has the following functional purpose:

- collection of system events from the Windows event service;
- interpretation of a system event into user action;

- recording user actions in the database;
- receiving control signals from the server;
- transmission of a control signal to the file system driver.

Mini-filter driver of the file system [16] is required to intercept operations with file resources and prohibit access to the resource in case of receiving a corresponding control signal from the agent program.

The server part includes the following components:

- database;
- web interface;
- the decision module.

The database stores the actions of all IS users that come from the installed agents.

The web interface serves for remote management and introduction to user profiles, their latest actions and other information.

The decision module calculates the correspondence of the current incoming user action with the profile. If the degree of compliance falls below the threshold value, the decision module sends a signal to the agent program to block access to the user.

Algorithmic support. Its own algorithmic support has been developed for the operation of the UAPS. The key element of the UAPS is the user behavior model, on the basis of which the UAPS makes decisions about the character of user actions. Markov chains are chosen as the mathematical apparatus used to construct a model of user behavior. The user behavior model consists of three Markov chains:

- “action-action” chain builds the probabilities of performing certain actions, after performing a certain action;
- “time-action” chain fixes the probabilities of performing actions at a certain time.
- “time-time” chain shows at what time the user is likely to take the next action.

To use the apparatus of Markov chains, a formula is needed that will correlate the probability of a state with the specificity of the user.

Several formulas have been proposed as options for a suitable formula:

$$H_t = H_{t-1} + \frac{e^{L_T} \cdot P_c}{e^{L_{\max}}} - \frac{L_A \cdot D}{L_T}, \quad (1)$$

$$H_t = H_{t-1} + P - \frac{L_A \cdot D}{L_{\max}}, \quad (2)$$

$$H_t = H_{t-1} + P - \frac{L_A \cdot D}{L_{\max} + L_T}, \quad (3)$$

$$H_t = H_{t-1} + \frac{L_T \cdot P}{L_{LA}} - \frac{L_A \cdot D}{L_{LT}}, \quad (4)$$

where H_t – current user specificity, H_{t-1} – specificity in the previous step, L_T – length of a chain of typical actions (number of typical actions in a row), L_A – the length of the current chain of atypical actions, L_{\max} – maximum chain length (adjustable value: length of the sequence is enough to determine the specificity), P – probability of

such an action, L_A – length of the chain of atypical actions (the number of atypical actions in a row), D – reduction ratio (for example 0,1), L_{LA} – length of the last chain of atypical actions, L_{LT} – length of the last chain of typical actions.

The range of values of specificity H_t should lie in the segment $[0; 1]$: 1 – typical, 0 – not typical. The exponents in (1) are chosen to reduce the rate of increment of specificity to 1. Formulas (1), (2), (3) and (4) satisfy the following conditions:

- The longer the chain of typical user actions is, the higher and more stable the specificity should be. Resilience is a resistance to a sharp decrease in specificity in the event of an atypical action;
- the more likely the action, the higher the specificity;
- one atypical (but not dangerous) action should not invert the specificity.

After the development of the internal model of the UAPS, a general algorithm of the UAPS was designed. A diagram of the general algorithm of the UAPS operation is shown in fig. 2.

The work of the UAPS is carried out in two stages: the training stage and the working stage.

At the training stage, the system accumulates actions that the IS user performs. During this period, such components as the agent program on the client side and the database on the server side interact with each other.

After a time period of training, the UAPS administrator must initiate the creation of a profile for the user using the web service. When a profile is created, an interaction occurs between the web service and the database.

After creating the profile, the UAPS goes into the working stage. At this stage, all new incoming user actions are sent to the decision module. The decision module calculates the correspondence of the current user action with behavior profile. If the calculated compliance coefficient is less than the threshold value, then the decision module sends a control signal about the need to block access to controlled objects of the file system. Having received a control signal, the agent program sends a command to the file system driver to block access to controlled objects of the file system.

Agent component. The development of the Agent component was carried out in the C # programming language [17] in the Visual Studio Community 2019 development environment. The project name of the developed program is Julia Agent. The class diagram of the developed program is shown in fig. 3.

The agent program is installed on the system as a service that starts when a user logs on to the system. When installing the program, a script is also executed that configures the Windows audit policy and additionally installs the Sysmon service [18]. The agent program window at startup is shown in fig. 4.

The program window provides the following information:

- the server to which the agent has successfully connected;
- the name of the account whose activity is currently being tracked;
- the number of events received from the Windows event log.

When minimized, the program is hidden in the notification window.

Further, to assess the information technology aspect, we measured the impact of the Julia Agent program on the system resources. The measurements were carried out on a system with the configuration shown in tab. 1.

If there is no user activity, the program exerts the following average network load: 42 bytes per second sent, 26 bytes received per second.

When user activity appears, the network load increases.

In this case, the total network traffic averages 1250 bytes (1.226 KB) per second.

The average load on the CPU agent process is 0.08 %.

The amount of memory used when the program is running is 6,812 KB (6.65 MB).

When interacting with the hard disk, the program reads an average of 1089 bytes (1.65 Kb) per second.

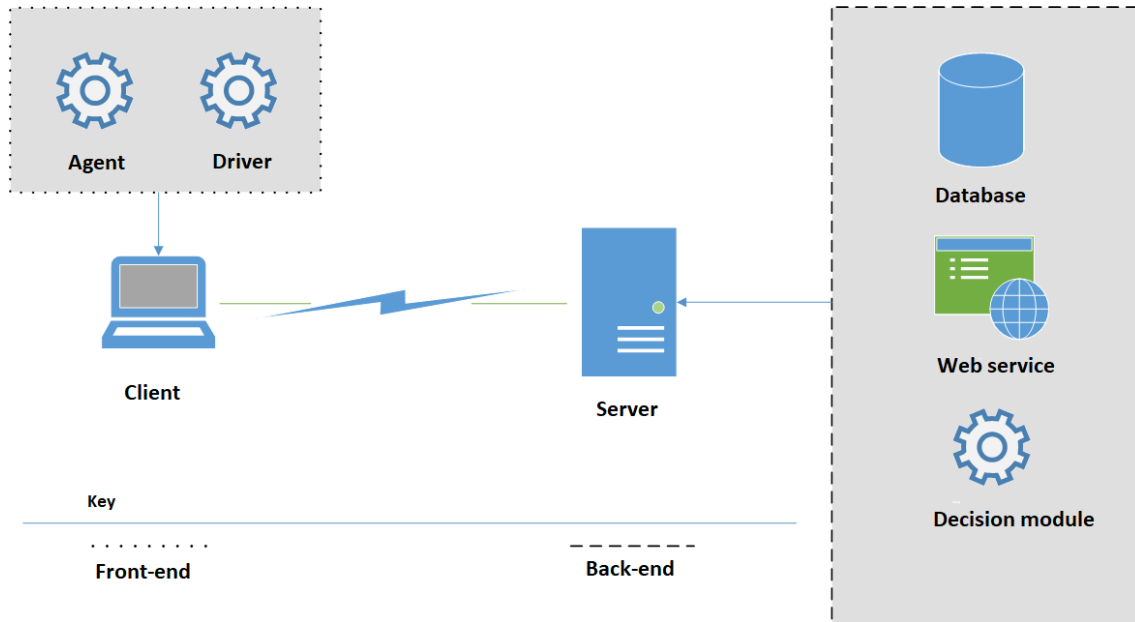


Fig. 1. Structural architecture of the user actions profiling system

Рис. 1. Структурная архитектура системы профилирования действий пользователя

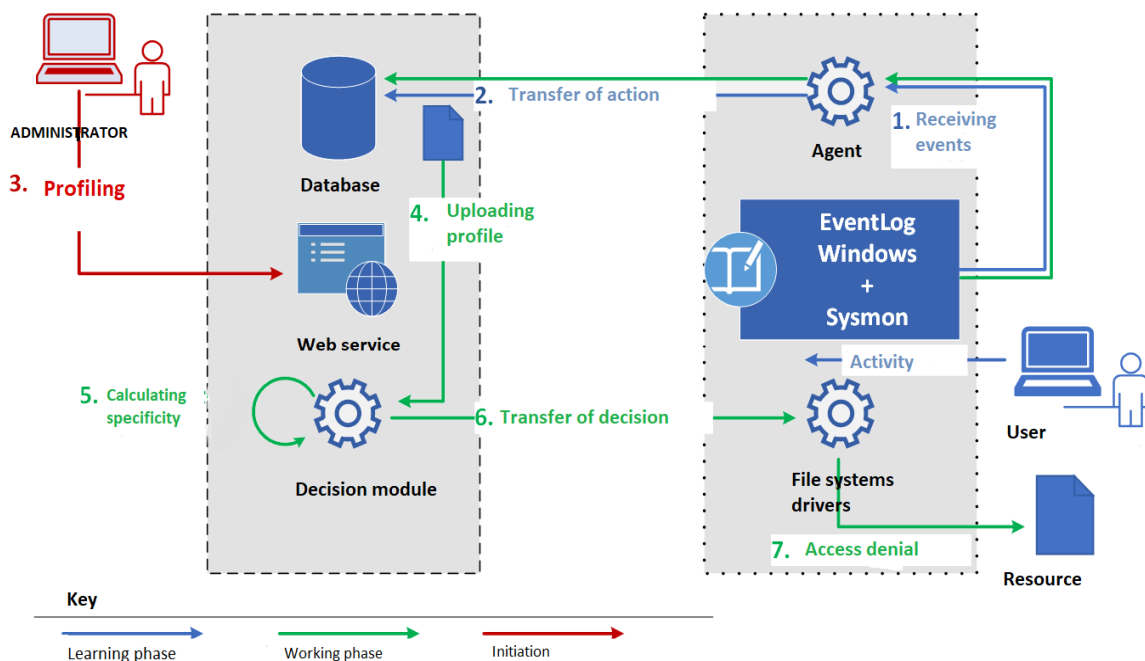


Fig. 2. General algorithm of the system for profiling user actions

Рис. 2. Общий алгоритм работы СПДП

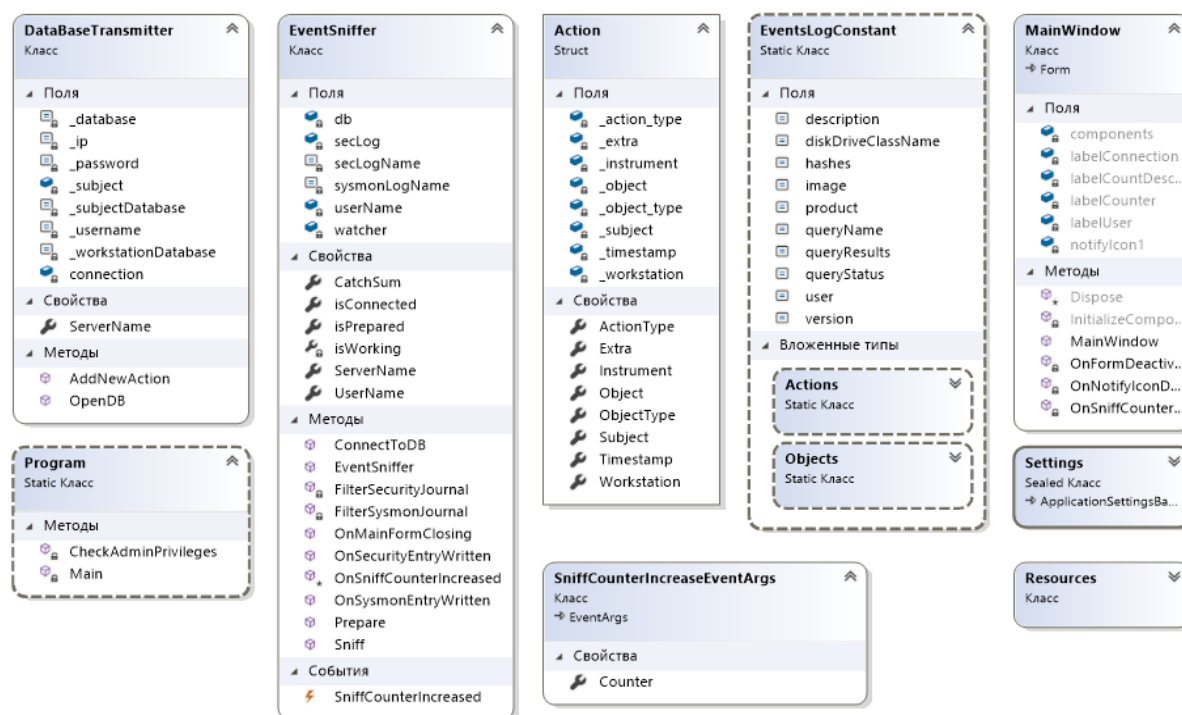


Fig. 3. Agent class diagram

Рис. 3. Диаграмма классов агента

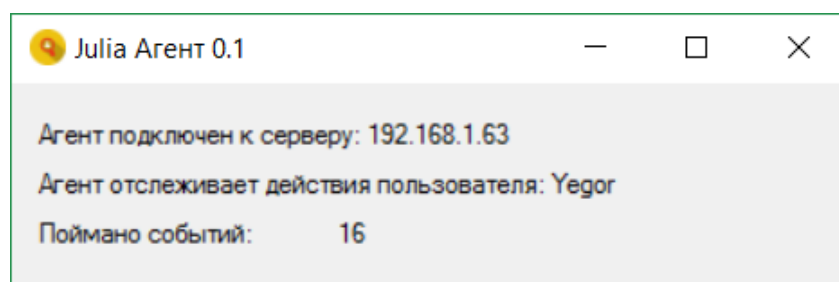


Fig. 4. The main window of the agent program

Рис. 4. Главное окно программы агента

The obtained average indicators of the use of PC resources are listed in tab. 2.

Thus, in all aspects, the program has a low consumption of system resources.

Web Service component. The development of the Web Service component was carried out in Python [19] in the PyCharm Community Edition development environment using the Django framework.

The web service designed for remote control of the UAPS by the security administrator provides the following set of capabilities:

- display of the latest actions of all users;
- displaying a list of all system users;
- formation and viewing of a user profile.

Fig. 5 shows the main page of the web service, which displays a list of recent user actions. The administrator

can select the number of actions to display using the corresponding control located above the actions table.

When you select the “Users” item (Russian: Пользователи) in the administration panel menu, a page opens containing a list of users connected to the UAPS, which is shown in fig. 6. At the time of testing the Users component, two users have been registered in the system.

After selecting a specific user, the page of this user is opened. This page allows you to:

- see for how many days the statistics of actions were collected;
- generate a user profile if the profile does not exist yet;
- open user profile, if it exists;
- view recent user actions.

Table 1

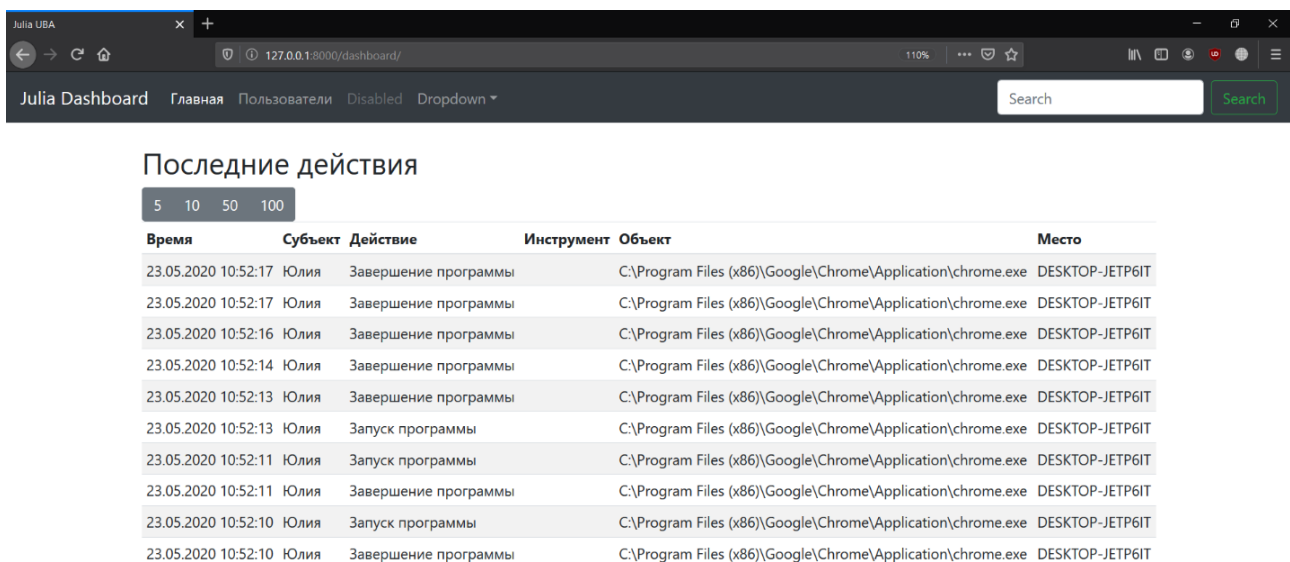
System characteristics for testing

System component	Characteristic
Central processing unit	AMD Ryzen 7 2700: 3.2 GHz 8 core
Random access memory (RAM)	16 Gbyte
Mass storage device	SSD Samsung 860 Evo 250 Gbyte HDD WD Blue 1 TByte
OS	Windows 10

Table 2

Average indicators of resource use by the “Agent” component

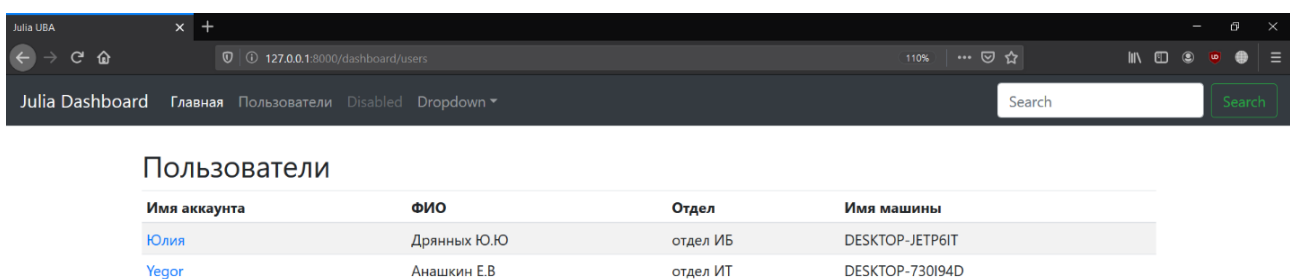
Resource indicator	Average load value
Network load idle	42 byte/s – outgoing 26 byte/s – incoming
Network load when active	1250 byte/s
CPU utilization	0.08 %
Memory occupied by RAM	6.65 Mb
Hard disk load	1089 byte/s – reading



Время	Субъект	Действие	Инструмент	Объект	Место
23.05.2020 10:52:17	Юлия	Завершение программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:17	Юлия	Завершение программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:16	Юлия	Завершение программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:14	Юлия	Завершение программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:13	Юлия	Завершение программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:13	Юлия	Запуск программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:11	Юлия	Запуск программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:11	Юлия	Завершение программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:10	Юлия	Запуск программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT
23.05.2020 10:52:10	Юлия	Завершение программы	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe		DESKTOP-JETP6IT

Fig. 5. Web service home page

Рис. 5. Главная страница веб-сервиса



Имя аккаунта	ФИО	Отдел	Имя машины
Юлия	Дрянных Ю.Ю.	отдел ИБ	DESKTOP-JETP6IT
Yegor	Анашкин Е.В.	отдел ИТ	DESKTOP-730I94D

Fig. 6. System users

Рис. 6. Пользователи системы

This page is shown in fig. 7.

After clicking the “To generate user profile” button (Russian: Сформировать профиль пользователя), the process of creating a user profile is started based on the currently available statistics. A profile consists of a visible and a hidden part. The displayed profile contains:

- a list of 10 most frequently used programs;
- a list of the 10 most frequently used network resources;
- a list of 10 most used files.

The top programs used by the user are shown in fig. 8.

The most used network resources are shown in fig. 9.

Top file resources used are shown in fig. 10.

The hidden part of the profile is a model of Markov chains written to a JSON file. The contents of the hidden part are shown in fig. 11.

Driver component. The development of the Driver component was carried out in the C programming lan-

guage [20] in the Visual Studio Community 2019 development environment.

The driver reads from a special configuration file a list of file resources to which access must be controlled. Then the driver compares any access to the system files with the list of monitored resources. If the file is included in the set of controlled resources, then the driver further checks the value of the flag variable, which serves as an indicator of trust in the user. By default, the flag is not selected, which means trusting the user and granting access to resources. The “Agent” component can change the flag by sending the appropriate control signal to the driver.

To check the operation of the file system driver, a control signal has been simulated to deny access to the “New text document.txt” file (Russian: «Новый текстовый документ.txt»). The driver successfully blocked access to the text file, as shown in fig. 12.

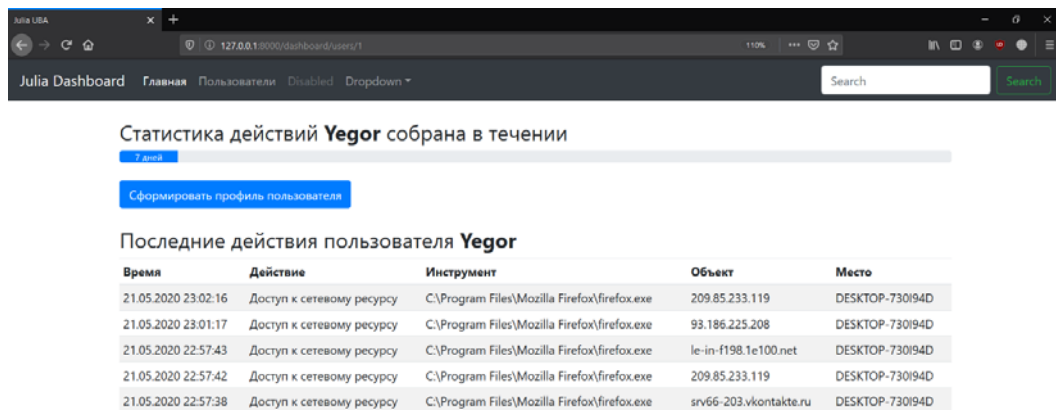


Fig. 7. User specific page

Рис. 7. Страница конкретного пользователя

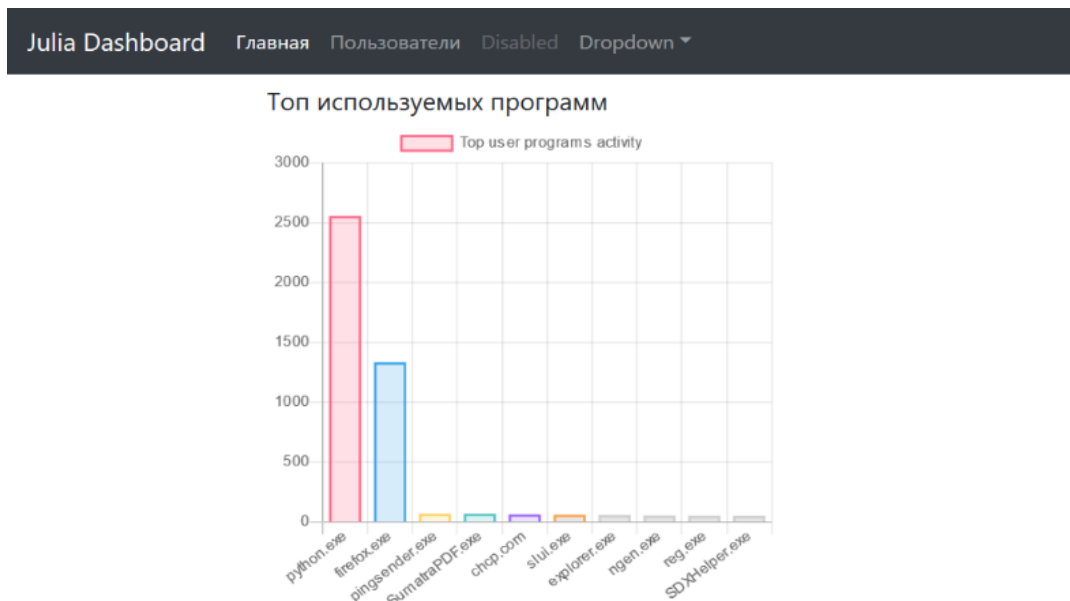


Fig. 8. Popular programs used

Рис. 8. Популярные используемые программы

Топ используемых сетевых ресурсов

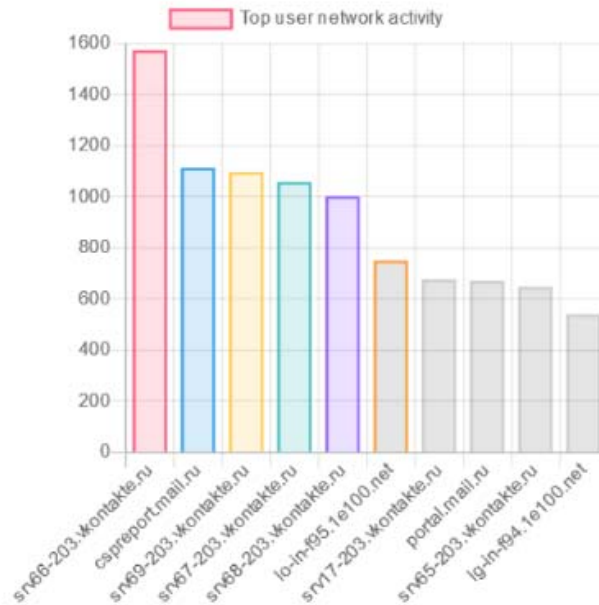


Fig. 9. Popular online resources

Рис. 9. Популярные сетевые ресурсы

Топ используемых файлов

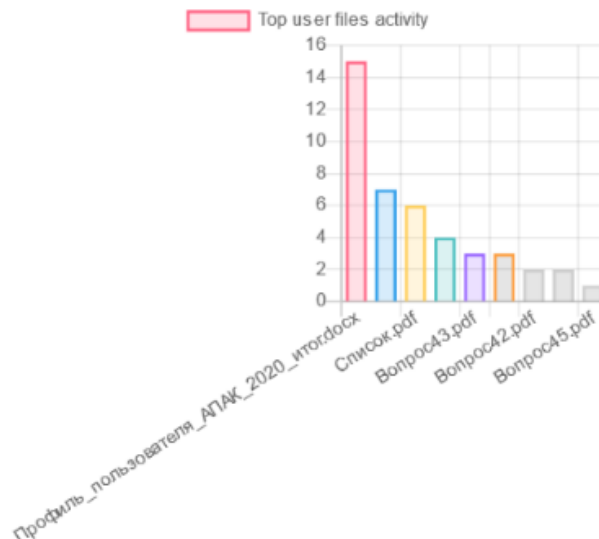


Fig. 10. Popular documents used

Рис. 10. Популярные используемые документы

Database component. A ready-made software product, PostgreSQL 12 DBMS [21] is used as the Database component.

To store data related to user activity, an uba database has been created, which has a structure in accordance with the diagram in fig. 13.

Database component implements the conceptual data model of user action, which has the following form: [TIME] [DATE] [SUBJECT] [ACTION] [OBJECT] from [LOCATION] using [TOOL].

Decision module component. The development of the Decision module component was carried out in Py-

thon in the PyCharm Community Edition development environment.

The decision module receives a notification from the database component if a new record (new user action) is added to the database. Then the decision module loads a new record from the database. From this entry, the decision module recognizes the user who performed the ac-

tion. Then the decision module loads the profile of the given user.

The user profile is a calculated Markov chain. From the Markov chains, the decisive module learns the probability of such an action by a given user.

Thereafter the specificity of the user is calculated by the formula (1).

```

1  {
2  "action_to_action": {
3      "Запуск программы C:\Program Files (x86)\ATI Technologies\ATI.ACE\Core-Static\MMLoadDrv.exe": {
4          "Запуск программы C:\Program Files\Mozilla Firefox\firefox.exe": 0.25,
5          "Создание файла C:\Users\Yegor\Downloads\лекции от\лекции от\-% Средства коллективной и индивидуальной защиты работающих.doc": 0.16666666666666666,
6          "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmp56thnj\__gen_py\__init__.py": 0.08333333333333333,
7          "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmpcybxfb\__gen_py\__init__.py": 0.16666666666666666,
8          "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmpub8y56\__gen_py\__init__.py": 0.08333333333333333,
9          "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmpyrlvec\__gen_py\__init__.py": 0.08333333333333333,
10         "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmpzldv9x\__gen_py\__init__.py": 0.08333333333333333,
11         "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmpb_dnl1\__gen_py\__init__.py": 0.08333333333333333
12     },
13     "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmpexn7q4cpsycharm-management\setuptools-40.8.0\setuptools\tests\test_archive_util.py": {
14         "Создание файла C:\Users\Yegor\AppData\Local\Temp\tmpexn7q4cpsycharm-management\setuptools-40.8.0\setuptools\tests\test_dep_util.py": 1.0
15     },
16     "Доступ к сетевому ресурсу 104.244.42.193": {
17         "Доступ к сетевому ресурсу 152.199.19.161": 0.022727272727272728,
18         "Доступ к сетевому ресурсу ec2-44-227-11-155.us-west-2.compute.amazonaws.com": 0.13636363636363635,
19         "Доступ к сетевому ресурсу DESKTOP-SB1VQM7": 0.045454545454545456,
20         "Доступ к сетевому ресурсу portal.mail.ru": 0.022727272727272728,
21         "Доступ к сетевому ресурсу srv135-185-240-87.vk.com": 0.045454545454545456,
22         "Доступ к сетевому ресурсу srv51-203.vkontakte.ru": 0.045454545454545456,
23         "Запуск программы C:\Program Files\Mozilla Firefox\firefox.exe": 0.3409090909090909,
24         "Доступ к сетевому ресурсу srv66-203.vkontakte.ru": 0.045454545454545456,
25         "Доступ к сетевому ресурсу cspreport.mail.ru": 0.06818181818181818,
26         "Доступ к сетевому ресурсу bar.love.mail.ru": 0.045454545454545456,
27         "Доступ к сетевому ресурсу 185.90.61.157": 0.045454545454545456,
28         "Доступ к сетевому ресурсу a88-221-216-200.deploy.static.akamaitechnologies.com": 0.022727272727272728,
29         "Доступ к сетевому ресурсу srv152-227.vkontakte.ru": 0.022727272727272728,
30         "Доступ к сетевому ресурсу lt-in-f19.1e100.net": 0.022727272727272728,
31         "Завершение программы C:\Program Files (x86)\Microsoft Visual Studio\Installer\resources\app\ServiceHub\Services\Microsoft.VisualStudio.Setup.Ser
32         "Доступ к сетевому ресурсу lq-in-f102.1e100.net": 0.022727272727272728,
33         "Доступ к сетевому ресурсу ec2-52-35-83-137.us-west-2.compute.amazonaws.com": 0.022727272727272728
34     },
35 }

```

Fig. 11. Snippet of hidden user profile

Рис. 11. Отрывок скрытого профиля пользователя

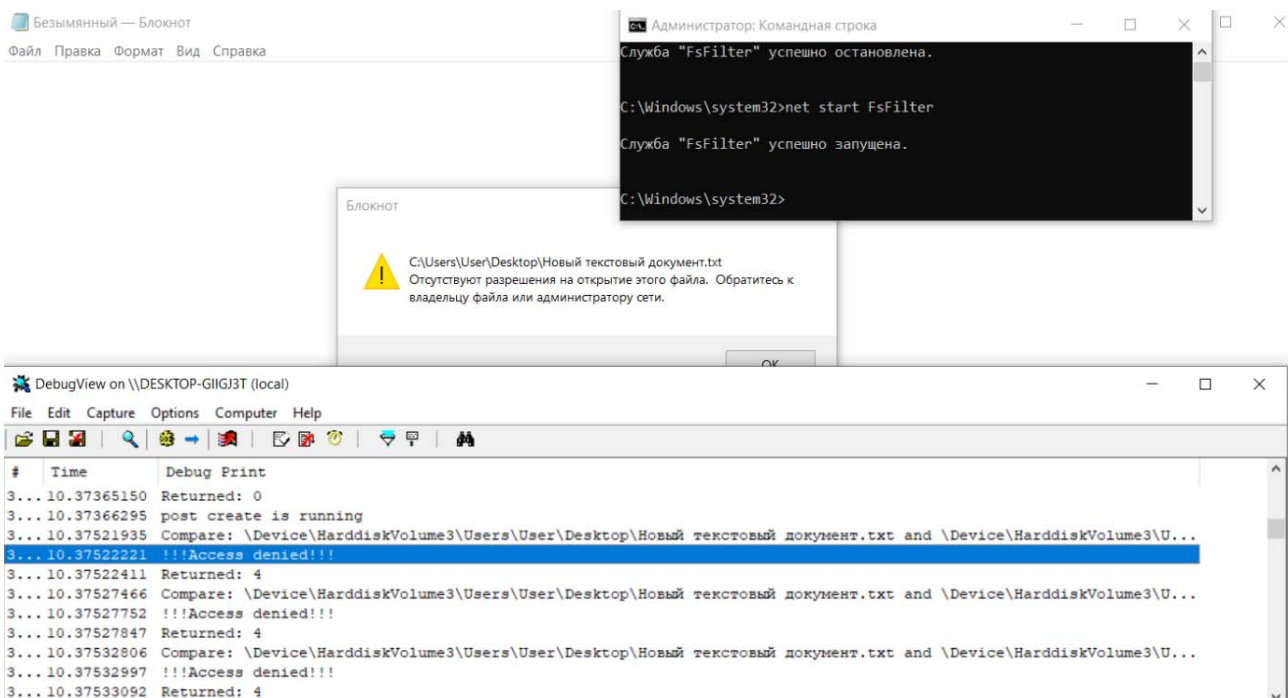


Fig. 12. The observed behavior of the file system driver confirms the correct operation of the component

Рис. 12. Наблюдаемое поведение драйвера файловой системы подтверждает корректность работы компонента

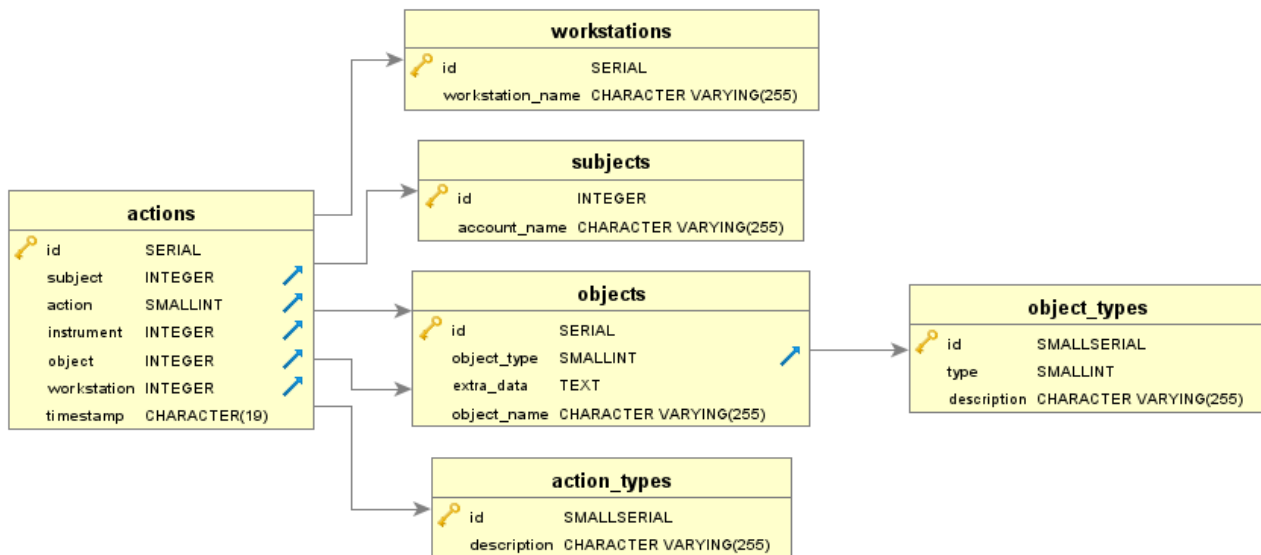


Fig. 13. Database component schema

Рис. 13. Схема компонента «База данных»

To apply formula (1), the decision module stores the following variables for each user:

- the last user action;
- the current specificity of the user;
- the length of the chain of typical actions;
- the length of the chain of atypical actions.

The parameters that are the same for all users are:

- the maximum chain length;
- the reduction factor.

These parameters in the decision module are specified as constant variables.

Conclusion. The client-server architecture was chosen as the software architecture for constructing a system for profiling user actions. The choice is based on the need to provide ergonomic parameters and statistics on the number of computerized workplaces in organizations. The system software is broken down into several main components:

- Agent;
- Driver;
- Database;
- Web Service;
- Decision module.

The software of each component has been successfully designed and developed. The development results have been verified. The software implementation of the components of the user profiling system has shown the viability of the proposed approach to access control based on user behavior analysis.

References

1. Data Breach Investigations Report. 2019, 78 p. Available at: <https://enterprise.verizon.com/resources/reports/2019-data-breach-investigations-report.pdf> (accessed 14.09.2020).
2. *Utechki dannyykh. Rossiya* [Analytical report GK Infowatch], Moscow, 2018 (In Russ). Available at: <https://www.infowatch.ru/resources/analytics/reports/russia2018> (accessed 14.09.2020).
3. Lukatskiy A. V. *Novaya kontsepsiya kiberbezopasnosti Cisco Trusted Access* [New cybersecurity concept Cisco Trusted Access]. Samara, 2019, 55 p. (In Russ). Available at: <https://www.slideshare.net/lukatsky/zero-trust-196618076> (accessed 15.09.2020).
4. Kindervag J. No More Chewy Centers: The Zero Trust Model Of Information Security, Forrester, March 23, 2016. 18 p.
5. Shashanka M., Shen M., Wang J. User and Entity Behavior Analytics for Enterprise Security. *IEEE International Conference on Big Data (Big Data)*. 2016, P. 1867–1874. Doi: 10.1109/BigData.2016.7840805.
6. Alruwaythi M., Nygard K. E. Fuzzy logic Approach Based on User behavior Trust in Cloud Security. *2019 IEEE International Conference on Electro Information Technology (EIT)*. Brookings, SD, USA, 2019. Doi: 10.1109/EIT.2019.8834173.
7. Li Y., Zhang T. Anomaly Detection of User Behavior for Database Security Audit Based on OCSVM. *3rd International Conference on Information Science and Control Engineering*. Beijing, China, 2016, P. 214–219. Doi: 10.1109/ICISCE.2016.55.
8. Ghazinour K., Ghayoumi M. An Autonomous Model to Enforce Security Policies Based on User's Behavior. *Conf. 14th International Conference on Computer and Information Science (ICIS)*, Las-Vegas, USA, June 28 – July 1 2015, 6 p. Doi: 10.1109/ICIS.2015.7166576.
9. Xi X., Shu-tao X., Xin-guang T., Qi-bin Z. Anomaly detection of user behavior based on DTMC with states of variable-length sequences. *The Journal of China Universities of Posts and Telecommunication*. Vol. 18(6), P. 106–115. Doi: 10.1016/S1005-8885(10)60128-8.
10. Yang F., Wu J., Tang S., Zhang H. Dynamic Knowledge Repository-based Security Auxiliary System of User behavior. *Conf. IEEE International Conference on Green Computing and Communications and IEEE*

Internet of Things and IEEE Cyber, Physical and Social Computing, Beijing, China, 20–23 Aug. 2013. Doi: 10.1109/GreenCom-iThings-CPSCoM.2013.390.

11. FSTEC of Russia. Acts. On approval of requirements for ensuring information security in automated production and technological process control systems at critical facilities, potentially dangerous facilities, as well as objects that pose an increased risk to human life and health and the environment : order of the FSTEC of Russia No. 31 : approved on March 14, 2014 : registered by the Ministry of justice of Russia on February 22, 2018, registration number 50118. Available at: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/868-prikaz-fstek-rossii-ot-14-marta-2014-g-n-31> (accessed 15.09.2020).

12. FSTEC of Russia. Acts. On approval of requirements for ensuring the security of significant objects of critical information infrastructure of the Russian Federation: order of the FSTEC of Russia No. 239: approved on December 25, 2017: registered by the Ministry of justice of Russia on March 26, 2018, registration number 50524. Available at: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/1592-prikaz-fstek-rossii-ot-25-dekabrya-2017-g-n-239> (accessed 15.09.2020).

13. FSTEC of Russia. Acts. On approval of requirements for the protection of information that does not constitute a state secret contained in state information systems: order of the FSTEC of Russia No. 17 : approved on February 11, 2013: registered by the Ministry of justice of Russia on May 31, 2013, registration number 28608. Available at: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/702-prikaz-fstek-rossii-ot-11-fevralya-2013-g-n-17> (accessed 16.09.2020).

14. FSTEC of Russia. Acts. On approval of the composition and content of organizational and technical measures to ensure the security of personal data during their processing in personal data information systems: order FSTEC of Russia No. 21: approved on February 18, 2013: registered by the Ministry of justice of Russia on May 14, 2013, registration number 28375. Available at: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/691-prikaz-fstek-rossii-ot-18-fevralya-2013-g-n-21> (accessed 16.09.2020).

15. Federal state statistics service. region of Russia. Socio-economic indicators-2019. Information and communication technologies. Number of personal computers per 100 employees: official website. Available at: https://gks.ru/bgd/regl/b19_14p/IssWWW.exe/Stg/d02/19-04.docx (accessed 16.09.2020).

16. Microsoft Docs. File System Minifilter Drivers: official documentation. Available at: <https://docs.microsoft.com/en-us/windows-hardware/drivers/ifs/filter-manager-concepts> (accessed 17.09.2020).

17. Shildt G. C# uchebnyy kurs [C# Training course]. St.Petersburg, Piter Publ., 2003, 20 p.

18. Microsoft Docs. Sysmon: official documentation. Available at: <https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon> (accessed 17.09.2020).

19. Python: official site. Available at: <https://docs.python.org/3/> (accessed 17.09.2020).

20. Kernigan, B.V. Yazyk Si [Language C]. Moscow, Williams Publ, 2017, 288 p.

21. PostgreSQL: The World's Most Advanced Open Source Relational Database. Available at: <https://www.postgresql.org/> (accessed 18.09.2020).

Библиографические ссылки

1. Data Breach Investigations Report. 2019. 78 p. [Электронный ресурс]. URL: <https://enterprise.verizon.com/resources/reports/2019-data-breach-investigations-report.pdf> (дата обращения: 14.09.2020).

2. Утечки данных. Россия, 2018 // ГК Infowatch [Электронный текст]. URL: <https://www.infowatch.ru/resources/analytics/reports/russia2018> (дата обращения 14.09.2020).

3. Лукацкий А. В. Новая концепция кибербезопасности Cisco Trusted Access: презентация. Самара, 2019. 55 слайдов [Электронный ресурс]. URL: <https://www.slideshare.net/lukatsky/zero-trust-196618076> (дата обращения: 15.09.2020).

4. Kindervag J. No More Chewy Centers: The Zero Trust Model Of Information Security // Forrester. March 23, 2016. 18 p.

5. Shashanka M., Shen M., Wang J. User and Entity Behavior Analytics for Enterprise Security // 2016 IEEE International Conference on Big Data (Big Data). P. 1867–1874.

6. Alruwaythi M., Nygard K. E. Fuzzy logic Approach Based on User behavior Trust in Cloud Security // 2019 IEEE International Conference on Electro Information Technology (EIT).

7. Yong Li, Tao Zhang. Anomaly Detection of User Behavior for Database Security Audit Based on OCSVM // 3rd International Conference on Information Science and Control Engineering. P. 214–219.

8. Ghazinour K., Ghayoumi M. An Autonomous Model to Enforce Security Policies Based on User's Behavior // Conf. 14th International Conference on Computer and Information Science (ICIS), Las-Vegas, June 28 – July 1 2015. 6 p.

9. Xi X., Shu-tao X., Xin-guang T., Qi-bin Z. Anomaly detection of user behavior based on DTMC with states of variable-length sequences // The Journal of China Universities of Posts and Telecommunication. Vol. 18(6). P. 106–115.

10. Yang F., Wu J., Tang S., Zhang H. Dynamic Knowledge Repository-based Security Auxiliary System of User behavior // Conf. IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, 20–23 Aug. 2013.

11. ФСТЭК России. Акты. Об утверждении требований к обеспечению защиты информации в автоматизированных системах управления производственными и технологическими процессами на критически важных объектах, потенциально опасных объектах, а также объектах, представляющих повышенную опасность для жизни и здоровья людей и для окружающей природной среды : приказ ФСТЭК России № 31 : утвержден 14 марта 2014 года : зарегистрирован Минюстом России 22 февраля 2018 г., регистрационный № 50118 [Электронный ресурс].

URL: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/868-prikaz-fstek-rossii-ot-14-marta-2014-g-n-31> (дата обращения 15.09.2020).

12. ФСТЭК России. Акты. Об утверждении требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры российской федерации : приказ ФСТЭК России № 239 : утвержден 25 декабря 2017 года: зарегистрирован Минюстом России 26 марта 2018 г., регистрационный № 50524 [Электронный ресурс]. URL: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/1592-prikaz-fstek-rossii-ot-25-dekabrya-2017-g-n-239> (дата обращения 15.09.2020).

13. ФСТЭК России. Акты. Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах: приказ ФСТЭК России № 17 : утвержден 11 февраля 2013 года: зарегистрирован Минюстом России Минюсте России 31 мая 2013 г., регистрационный № 28608 [Электронный ресурс]. URL: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/702-prikaz-fstek-rossii-ot-11-fevralya-2013-g-n-17> (дата обращения 16.09.2020).

14. ФСТЭК России. Акты. Об утверждении состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных: приказ ФСТЭК России № 21 : утвержден 18 февраля 2013 года: зарегистрирован Минюстом России Минюсте России 14 мая 2013 г., регистрационный № 28375 [Электронный ресурс]. URL: <https://fstec.ru/normotvorcheskaya/akty/53-prikazy/691-prikaz-fstek-rossii-ot-18-fevralya-2013-g-n-21> (дата обращения 16.09.2020).

15. Федеральная служба государственной статистики. Регионы России. Социально-экономические показатели – 2019. Информационные и коммуникационные технологии. Число персональных компьютеров на 100 работников: официальный сайт [Электронный ресурс]. URL: https://gks.ru/bgd/regl/b19_14p/IssWWW.exe/Stg/d02/19-04.docx (дата обращения 16.09.2020).

16. Microsoft Docs. File System Minifilter Drivers: official documentation [Электронный ресурс]. URL: <https://docs.microsoft.com/en-us/windows-hardware/drivers/ifs/filter-manager-concepts> (дата обращения 17.09.2020).

17. Шилдт Г. С# учебный курс. СПб. : Питер, 2003. 20 с.

18. Microsoft Docs. Sysmon: official documentation [Электронный ресурс]. URL: <https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon> (дата обращения 17.09.2020).

19. Python: official site [Электронный ресурс]. URL: <https://docs.python.org/3/> (дата обращения 17.09.2020).

20. Керниган Б. В., Ричи Д. М. Язык Си. М. : Вильямс, 2017. 288 с.

21. PostgreSQL: The World's Most Advanced Open Source Relational Database [Электронный ресурс]. URL: <https://www.postgresql.org/> (дата обращения 18.09.2020).

© Anashkin E. V., Zhukova M. N., 2020

Anashkin Yegor Vadimovich – PhD student, assistant lecturer, department of Information technology security; Reshetnev Siberian State University of Science and Technology, Institute of Informatics and Telecommunications. E-mail: a.yegoriy@gmail.com.

Zhukova Marina Nikolaevna – Cand. Sc., associate professor of the department of Information technology security, Reshetnev Siberian State University of Science and Technology, Institute of Informatics and Telecommunications.

Анашкин Егор Вадимович – аспирант, ассистент кафедры безопасности информационных технологий; Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева, Институт информатики и телекоммуникаций. E-mail: a.yegoriy@gmail.com.

Жукова Марина Николаевна – кандидат технических наук, доцент кафедры безопасности информационных технологий; Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева, Институт информатики и телекоммуникаций.
