Web Application Critical Resources Protection

Bogdan Korniyenko
Department of Information Systems and Technologies
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kyiv, Ukraine
bogdanko@gmx.net

Lesya Ladieva
Department of Hardware and Software Automation
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kyiv, Ukraine
lrynus@yahoo.com

Liliya Galata
Department of Computerized Information Security Systems
National Aviation University
Kyiv, Ukraine
galataliliya@gmail.com

Olesya Yakovenko
Department of Computerized Information Security Systems
National Aviation University
Kyiv, Ukraine
yak-olesya@ukr.net

Andrii Nesteruk
Department of Information Systems and Technologies
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kyiv, Ukraine
aonesterukr@gmail.com

Viktoriia Ivannikova
Department of Air Transportation Management
National Aviation University
Kyiv, Ukraine
victoriia.ivannikova@gmail.com

2021 IEEE 3rd International Conference on Advanced Trends in Information Theory (ATIT) | 978-1-6654-3847-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/ATIT54053.2021.9678541

***Abstract* - Developed web application protection system by using modern technologies NET Framework, ASP. NET Core, EF, SSMS, Swagger. The system is resistant to changes and outside interference, able to prevent unauthorized access. The main types of vulnerabilities in web applications are considered. The most popular ready-made services for the implementation of the appropriate protection are described. The white list model of developing secure web applications and the main steps of the model implementation is defined. Implemented a white list model for a web application by using a system of roles and access. The server part of the web application has been developed, which includes the built-in functionality of the basic methods of hacking prevention. Impact of SQL injection through project architecture is not possible. A method for accessing private user information has been developed by using the Rijndael encryption algorithm.**

Keywords - web application; security; protection system; threats; white list; model

I. INTRODUCTION

Protection of web resources remains one of the most important trends of information security. Annually, the number of web resources increases, as does the amount of confidential information that is localized on remote access servers (especially by using cloud technologies). As a result, not only the number of attacks on web resources is growing, but also the economic consequences of such attacks.

There isn't unique approved methodology in Ukraine for a set of protection mechanisms against unauthorized access. That's why the level of computer systems protection does not correspond to the existing threats. In this regard, it is proposed to develop a system that will be protected from a set of possible options for intrusion, interference or unauthorized access to the information [1-6].

The purpose of the article is to develop the web application protection system by using the modern programming technologies and to develop a database that will be resistant to changes and third-party interference, able to prevent unauthorized access to the web application.

II. THREATS AND CRITICAL RESOURCES OF THE WEB APPLICATION PROTECTION METHODS OVERVIEW

Today, web vulnerabilities do outweigh any other information security issues. Most external attacks on corporate information systems target the vulnerabilities of web applications.

Every year the statistics of attacks changing, so in 2019 the most popular was "Cross-site scripting", in 2018 - "Information leakage", and today the most popular attack is "Insufficient transport layer protection" - obtaining data during transmission [7-9].

Based on the mentioned above (Fig. 1), we can conclude that there's enough to properly check the input data for the further protection against the most popular attack types. It is also recommended to use the encrypted protocol HTTRS and build a resource application on one of the known software frameworks (Framework), which has built-in mechanisms for verification, encryption and validation of input data.

№	Attack type	Web resources vulnerability, %	Reaction
1	Insufficient transport layer protection	70 %	Using the https protocol
2	Information leakage	56 %	Software testing of the resource, check server-side messages, error notification monitoring
3	Cross-site scripting	47 %	Purification and validation of input data
4	Brute force	29 %	Use of highly complex passwords, server configuration for the analysis of incoming requests
5	Content spoofing	26 %	Refrain from using frames, do not pass absolute or local files paths in the parameters
6	Cross-site request	24 %	Checking input data from forms
7	URL redirector abuse	16 %	Validation of input data
8	Predictable resource location	15 %	Access control to server files

Fig. 1. Classification of attack types

Web resource protection technologies:

WAF (Web Application Firewall) - the solution allows you to prevent many types of attacks. These include SQL and PHP injections, cross-site scripting (XSS), password

selection, exploiting zero-day vulnerabilities in web applications, application-level DDoS attacks, and resource-intensive page attacks.

The essence of Web Application Firewall is to protect the company's WEB resources that are exposed. WAF is a highly specialized device and it actively controls only HTTP / HTTPS protocols.

Open VAS is a full-featured vulnerability scanner. Its capabilities include non-authenticated testing, authenticated testing, various high-quality and low-level Internet and industrial protocols, performance tuning for large-scale scanning, and a powerful internal programming language for implementing any type of vulnerability test [10-12].

CVSS 3.0 system analysis. The Common Vulnerability Scoring System (CVSS) is an open basis for communicating the characteristics and severity of software vulnerabilities. CVSS consists of three metric groups: Base, Temporal and Environmental. Baseline metrics give a score in the range of 0 to 10, which can then be changed by evaluating the time metric and the environment. The CVSS score is also presented as a vector series, a concise text representation of the values used to display the scores.

Comparative analysis of existing methodologies for information security testing

The Open Source Security Testing Methodology Manual (OSSTMM) is a fairly formalized and well-structured document for network testing.

The advantage of this document: it describes terms the techniques of checking the security of a computer system in general and their brief description; links to software products that must be used for testing are provided; references to other regulations and methodologies [13-16].

The disadvantages of the methodology are: this document was developed in 2008; at the moment it does not correspond to the current state of development of information technologies and unauthorized interference methods into computer networks.

OWASP methodology (Open Web Application Security Project) Testing Guide. OWASP (Open Web Application Security Project) - an international open community that focuses on improving software security.

The advantage of this document: OWASP Command staff provides all the necessary information for each stage of the life cycle of secure software development; is the most popular and complete collection of web application security testing tools available on the Internet.

PTES methodology - Penetration Testing Execution Standard - TechnicalGuidelines. The standard designed to combine both business requirements and security capabilities, and scaling penetration tests.

Advantage of this methodology: technical guide that contains detailed technical information about tools and commands for each stage of penetration testing.

ISSAF methodology - Information System Security Assessment Framework. The ISSAF methodology allows to simulate the requirements for internal security measures, and aims to assess the security of computer networks, systems and applications.

BSI Methodology - Study A Penetration Testing Model. Developed by the German division of the Federal Office for Information Security. The document describes the correct strength tests of the system.

Advantages of this methodology: the methodology is quite detailed and tries to anticipate all aspects of strength tests, technical, organizational and legal; the annexes contain the software description that can be used to test the objects described in the method [17-20].

Conclusion: After analyzing the problem of the web applications creating, we can state that this problem is relevant and extensive and has common features with the general concept of creating the secure applications. This problem partly depends on the mechanisms of the used web frameworks protection where is the web application, the used database, security measures from the server where the web application is executed and most of all on the developer’s professional skills.

III. WHITE LIST METHOD FOR SECURE WEB APPLICATIONS

The white list concept is well known and used for a long time in many areas before the advent of information and digital technologies [21-24].

The "white list" is the practice of identifying entities that are granted certain privileges, services, mobility, access or recognition. Entities on the list will be accepted, approved and/or recognized. The "white list" is the opposite of black list, the practice of person’s identification who is not recognized.

A. Description of the white list model for developing secure web applications

In theory, the white list model web application development will allow developers to increase security by detecting unpredictable and dangerous web application behavior, terminating it, or redirecting it to a safe one.

A mismatch between expected and actual web application behavior can be an indicator of an attack to the web application. This research focuses on OWASP 4 and 7 vulnerabilities: dangerous direct object references. Lack of access control functions. You must implement a security mechanism for your web application by anticipating allowed operations. Any application sequence that was not in the development specification will be rejected by the logic application and redirected.

This article defines, creates, and implements a list of allowed web application interactions. These rules will control the HTTP requests and responses processed by the web application. The definition of white list items should be developed during the design stage.

Ideally, the white list should be created dynamically during web application development by using behavioral monitoring tools and program execution. But since our main goal is to determine whether the security of the web application will be improved, we will use a static white list, which we will define at the design stage.

B. Formal definition of the white list model for developing secure web applications

Define the white list as a set of four sets {C, D, W, S}, where:

C is the set of elements {u, c_1, c_2,…,c_n}, where c_1, c_2,…,c_n are components within the system and u is a component outside the system.

D is the set of elements {d_1, d_2,…,d_n}, where d_1, d_2,…,d_n is the state of the components.

Each cell dimensional matrix | C | X | C | contains a unique subset of x, {x: x ⊆ D}. If the estimate of states X returns 1, then the ordered pair of the state transition from initial $c_{o,}$ to final c_d is added to the set W.

W is the set of ordered pairs {($c_{o,}$ c_d): $c_{o,}$ c_d is C} each the pair represents the transition from the initial component c_0 to the final c_d

S dimension matrix | C | X | C | $S_{co,\ cd} = c_{(safe)}$ defines safe components { c_s : c_s is C } where c_d cannot follow c_0.

Equation (1) - the transition from one component to another is controlled by the transition function where the transition from initial to final occurs then and only then if ($c_{o,}$ c_d) is W, otherwise the transition function is called through (c_o, $S_{co,\ cd}$).

$$T(c_0, c_d) = \begin{cases} c_d, \ if\ (c_0, c_d) \in W, else \\ T(c_0, S_{c_0 c_d}) \end{cases} \quad (1)$$

Several white list operations are defined. Transactions are divided into two categories according to when they can be applied. The following operations will be used during development:

• Create ($c_{o,}$ c_d) in set W: add ordered pair to set

• Remove ($c_{o,}$ c_d) from set W: remove ordered pair from set

• Enter {d_x} to set D: add states d_x to set D.

• Remove {d_x} from set D: remove states d_x from set D.

• Add {d_x} to subset x of set W $c_{o,}$ c_d.

• Remove {d_x} from subset x of set W $c_{o,}$ c_d.

• Enter {c_s} in cell matrix $S_{co,cd}$.

• Update {c_s} in the matrix cell.

The white list operations that are allowed when performed:

• Calculation of T ($c_{o,}$ c_d)

• Variation c_o à c_d: c_d can follow to c_o if this statement is false, then all states belonging to the subset x y $W_{co,\ cd}$, the c_d will return the truth. Otherwise, the transition to the safe state c_s.

The set of all components C and relations W can be represented as a binary matrix, where 1 means allowed state transition and 0 means not allowed. Each cell of the matrix will take the value either 0 or 1 according to the values of the subset of states. The matrix S will contain safe state transitions in case the state transition from c_o to c_d is not allowed.

C. Steps of model implementation

The first step is to identify the allowed behavior of the web application by creating a chart that will show how the program works. The diagram should show all allowed interactions between application components. You have to investigate each operation and identify a subset of state transitions (c_o, c_d) and place them in the appropriate cells of the matrix W. You also need to identify safe components and transitions in case of return of the erroneous value of the state transition permission check. Safe components c_s should be placed in cells that correspond (c_o, c_d) in the matrix W to the matrix S. So at this stage of development we have subsets of transition states that are written to the matrix W and take values 1 or 0 depending on the permissibility of state transitions. For simplicity, we call such a representation as M, where | C | X | C | = M where $M_{co,cd} = 1$ if (c_o, c_d) is W and $M_{co,cd} = 0$ if (c_o, c_d) does not belong to W. The white list model can be changed or supplemented depending on the course of development. It should also be noted that the states in the set D should be simple and not complex.

Let's define the following steps to build the white list model:

• Create a chart that shows the assigned valid behavior of the web application;

• Set subsets of states to allow and assign permitted transitions and application relationships; {{1}}

• Place each subset of the transition of states co, cd corresponding to the cell of the matrix | C | x | C |;

• Identify safe components c_s and place them in the cells corresponding to (c_o,c_d) in the matrix W to matrix S;

• Assign a value of 1 or 0 for each subset of state transitions depending on the correspondence of the white list;

• $M_{co,cd} = 1$ if (c_o,c_d) is W and $M_{co,cd}= 0$ if (c_o,c_d) does not belong;

• Properly adjust the development process to implement the white list methodology into the existing development process at any stage.

D. The white list method implementation for web applications

Let's have the web application that requires user authentication to be able to use it. Let the application allow 3 authentication attempts. If the user fails 3 attempts, the application will block the access. If the user successfully passes the authentication, the application will redirect the user to his personal page. The user will be able to edit his/her profile or to connect with other users. The user can also log out of his/her profile at any time.

Green arrows allowed the white list operations. Red is not allowed (Fig. 2).

The application model of this example consists of 5 components. C = {u, c_1, c_2, c_3, c_4, c_5}. U is a component outside the system and is included in C for completeness.

For the global set of states D, suppose they mean the following states:

d1: the user is anonymous.

d2: user is authorized.

d3: session time is valid.

d4: previous representation.

d5: subsequence representation

d6: authentication attempts <3.

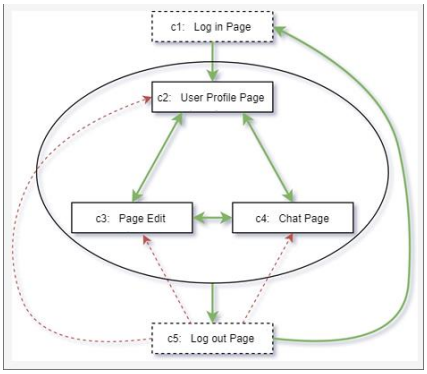

Fig. 2. Example chart showing the behavior of the program

	u	c1	c1	c3	c4	c5
u	-	-	-	-	-	-
c1	-	-	x = {d_2, d_3, d_4 = login view, d_5 = user portal view, d_6}	-	-	-
c2	-	-	-	-	-	-
c3	-	-	-	-	-	-
c4	-	-	-	-	x = {d_2, d_3, d_4 = edit profile view aбo user portal view, d_6}	-
c5	-	-	-	-	-	-

Fig. 3. The W matrix of ordered pairs of components' transition states

The white list contains a subset of D in each cell of the matrix. For example, the white list below shows a subset of the allowed state transitions from c_1, c_2, and another subset that results in an unauthorized transition from c_5 to c_4. For the allowed transition from c_1, c_2, the subset of states: x = {d_2, d_3, d_4 = login view, d_5 = user portal view, d_6}. All states in x must return the truth. For unauthorized transitions from c_5 to c_4, the subset of states x = {d_2, d_3, d_4 = edit profile view or user portal view, d_6}.

Obviously, the transition from state c_5 to c_4 is not allowed, because the first state in subset d_2 cannot be reached if the user has logged out. Let's fill our transitions into the matrix of the state transitions (Fig. 3).

Let the set of ordered pairs of the state transitions be as follows

W = {(u; u) ; (u; c1) ; (c1; c1) ; (c1; c2) ; (c2; c2) ; (c2; c3) ; (c2; c4) ; (c2; c5) ; (c3; c2) ; (c3; c3) ; (c3; c4) ; (c3; c5) ; (c4; c2) ; (c4; c3) ; (c4; c4) ; (c4; c5) ; (c5; c1)}

Then we can present the matrix W in binary form (Fig. 4):

	u	c1	c2	c3	c4	c5
u	1*	1	0	0	0	0
c1	0	1	1	0	0	0
c2	0	0	1	1	1	1
c3	0	0	1	1	1	1
c4	0	0	1	1	1	1
c5	0	1	0	0	0	0

Fig. 4. Binary matrix of components states transition W

The next step is to fill the matrix S with safe components to redirect the transition to the safe state if it is not allowed (Fig. 5).

	u	c1	c2	c3	c4	c5
u	NA	c1	c1	c1	c1	c1
c1	c5	c1	c1	c1	c1	c1
c2	c5	c5	c5	c1	c1	c5
c3	c5	c5	c5	c5	c5	c5
c4	c5	c5	c5	c5	c5	c5
c5	c5	c5	c5	c5	c5	c5

Fig. 5. A matrix for the web pages transition to safe states

Thus, the concept of typical application of the white list in IT is analyzed and also the white list model and the implementation technique of this model for secure web applications is developed, as well as the example of use. It can be concluded that any concept of protection or delimitation of the access can be adapted to any process, namely the process of developing secure web applications. According to the analysis of the created example the model is working and successfully provides safety of the designed application.

IV. WEB APPLICATION CRITICAL RESOURCES PROTECTION

The project used ASP.NET - a technology for creating web applications and web services, namely ASP.NET Core, this framework is a complete census, which combines previously separate ASP.NET MVC and ASP.NET Web API into a single programming model.

A. Information system design

Every large project uses patterns to improve the code structure and its support in the future. When develop an application by using ASP.NET Core technology, the pattern was chosen that divided the system into three interrelated parts: data model, user interface and control module. It is used to separate the data (model) from the user interface so that changes to the user interface have minimal impact on the data, and changes in the data model can be made without changes to the user interface.

The main features of the program (Fig. 6):

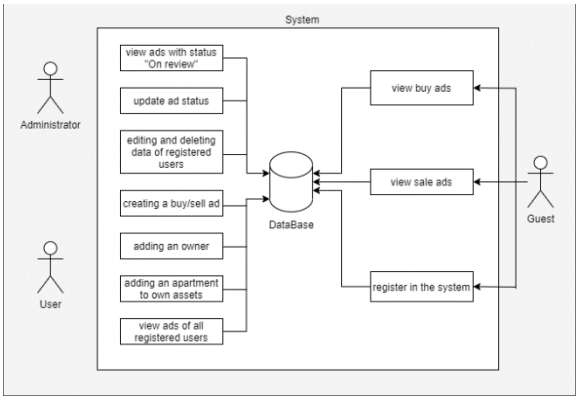

Fig. 6. Chart of precedents

To develop the mentioned above system, was decided to use the object-oriented programming language C # and use

the following technologies such as: .NET Framework, ASP.NET Core, EF, SSMS, Swagger.

During the analysis of the received task the question of deployment of the server arose. IIS Express is provided for this purpose. Since the main task was to develop Api, the project uses the Swagger library to display functionality in the browser.

The SQL language is selected to build database requests. Today it is the most commonly used solution because it is time-tested. An SSMS database control system has been selected for access and database management.

We use six controllers (top-level classes): Account Controller, Apartment Owners Controller, Apartments Controller, Rent Announcements Controller, Sale Announcements Controller, Announcements Controller.

These classes respond to user requests. They are all at the presentation level. To further request processing, they go to the trough of business logic, where they turn to the interfaces of their services, which in their turn call to the services themselves. The following is a link to a specific repository interface, which directs to the specific repository.

Then all repositories send requests to the database. This structure is implemented to provide more flexibility and modularity.

The first thing to note is that we will have two databases. We need one as the working system and the second one - for authentication and authorization.

B. *Project structure*

It was decided to use three-level architecture to develop the system.

The presentation layer is the part of the system that is managed and used by the consumer. At this level there is a user interface together with the developed system for receiving data from it. Presentation layer is implemented in the project EstateEgancy.API.

Business layer (level of business logic) - contains a set of components that are responsible for processing the data obtained from the presentation level, implements all the necessary logic of the application, all calculations, interacts with database and transmits the results of processing to the presentation level. The business layer is implemented in the EstateEgancy.BLL project.

Data Access layer - stores models that describe entities, as well as specific classes for working with various data access technologies, such as a data context class Entity Framework. There are also repositories through which business logic interacts with the database. Data Access layer is implemented in the project EstateEgancy.DAL.

It should be noted that the extreme levels cannot interact with each other; therefore the presentation level cannot directly access the database. This is only possible through the use of business logic.

While the Data Access layer is developed a "repository" and a "unit of work" are also implemented, as well as certain data objects (DTOs). A DTO is an object that determines how data will be sent over a network.

Each user has his/her own profile that contains the user's private information that needs protection.

Using the symmetric Rijndael block encryption algorithm (Advanced Encryption Standard), developed a method of accessing private user information.

The method of accessing the user's private information has been developed by using the symmetric block encryption algorithm Rijndael (Advanced Encryption Standard). In order to get an undistorted representation of the data, the user must enter his/her key. Thus, only a user with one correct key can access this data. In the case of a key loss scenario, the user must contact the administrator to generate and obtain the new key.

Implemented SQL injection prevention, used queries that are parameterized and supported by most web programming languages. }} For example, for an application developed by using PHP and MySQL, it looks like:

```
$stmt = $pdo ->prepare ('SELECT * FROM table WHERE
column = :value');$stmt ->execute (array ('value' =>
$parameter));
```

C. *Test example of the program*

Figure 7 shows how a user can see the site. There are three roles in the system: administrator, user and guest. Everyone has different levels of access.

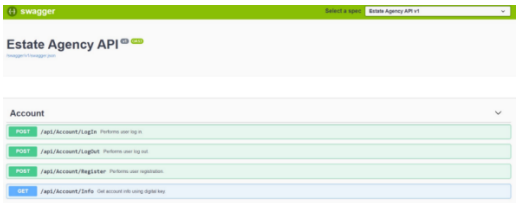

Fig. 7. Initial view of the site

This system is designed for comfortable use. Therefore, it is very important that each user does not afraid that someone may change his data or do something harmful. If everything is entered correctly and such user exists, the program will issue a code of 200. However, it can return a code of 400 in case of incorrect data or invalid model.

Figure 8 shows an example of obtaining the personal user data via the Get api / Account / AccountInfo route. Valid data will only be returned if the correct key is transmitted. Otherwise, the data will be decrypted incorrectly and returned in a distorted form.

Fig. 8. Receiving personal user data

In general, this project looks like a developed server part of the web application, which includes built-in functionality to prevent basic hacking methods.

V. CONCLUSION

Web applications protection from hackers depends on the technologies and components used to create web applications as well as the possible vulnerabilities of those components. There are different classifications of

vulnerabilities. Each attack has its own characteristics due to vulnerabilities, but the cause of vulnerabilities are errors in the developing, implementation and application of web application components, hence the need to search for vulnerabilities and respond to information about their location.

The main types of vulnerabilities in web applications are considered. It also describes the most popular ready-made services for the implementation of appropriate protection.

The result is a software product, a web application for renting and selling real estate with a built-in security system. The architecture of the project prevents the impact of SQL injection. The white list model is implemented by using a role-access system. The method for accessing user's private information has been developed by using the Rijndael encryption algorithm.

REFERENCES

[1] Howard MD. LeBlanc. Writing Secure Code 2nd ed., Redmond, Washington: Microsoft Press, 2003.

[2] Rasmi M, Jantan A. Attack Intention Analysis Model for Network Forensics. Software Engineering and Computer Systems; 2011, pp. 403-411.

[3] Ben Arfa Rabai L, Jouini M, Ben Aissa A, Mili A. A cybersecurity model in cloud computing environments. Journal of King Saud University – Computer and Information Sciences; 1: 2012, pp. 63-75.

[4] Ben Arfa Rabai L, Jouini M, Ben Aissa A, Mili A. An economic model of security threats for cloud computing systems. International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec); 2012, pp. 100-105.

[5] R. L. Church, M.P. Scaparra. Protecting critical assets: The R-interdiction median problem with fortification. Geographical Analysis, 39, 2006, pp. 129–146.

[6] R. L. Church, M.P. Scaparra. Analysis of facility systems' reliability when subject to attack or a natural disaster. In Critical infrastructure: Reliability and vulnerability: Advances in spatial science, Edited by: Murray, A. T. and Grubesic, T. H. Berlin: Springer, 2007, pp. 221–241.

[7] A. J. Holmgren, E. Jenelius, J. Westin. Evaluating strategies for defending electric power networks against antagonistic attacks. IEEE Transactions on Power Systems, 22(1), 2007, pp. 76–84.

[8] E. Jenelius, J. Westin, J. Holmgren. Critical infrastructure protection under imperfect attacker perception. International Journal of Critical Infrastructure Protection, 3(1), 2010, pp. 16–26.

[9] U. Jüttner, H. Peck, M. Christopher. Supply chain risk management: Outlining an agenda for future research. International Journal of Logistics: Research and Applications, 6(4), 2003, pp. 197–210.

[10] Galata, L., Korniyenko, B. Research of the Training Ground for the Protection of Critical Information Resources by iRisk Method. Mechanisms and Machine Science, 70, 2020, pp. 227-237. DOI: 10.1007/978-3-030-13321-4_21

[11] Korniyenko, B., Galata, L., Ladieva, L. Mathematical model of threats resistance in the critical information resources protection system. CEUR Workshop Proceedings, 2577, 2019, pp. 281-291.

[12] Korniyenko, B., Galata, L., Ladieva, L. Research of Information Protection System of Corporate Network Based on GNS3. 2019 IEEE International Conference on Advanced Trends in Information Theory, ATIT 2019 - Proceedings, № 9030472, 2019, pp. 244-248. DOI: 10.1109/ATIT49449.2019.9030472

[13] Korniyenko, B., Ladieva, L., Galata, L. Control system for the production of mineral fertilizers in a granulator with a fluidized bed. ATIT 2020 - Proceedings: 2020 2nd IEEE International Conference on Advanced Trends in Information Theory, № 9349344, 2020, pp. 307-310. DOI: 10.1109/ATIT50783.2020.9349344

[14] Korniyenko, B., Galata, L. Implementation of the information resources protection based on the CentOS operating system. 2019 IEEE 2nd Ukraine Conference on Electrical and Computer Engineering, UKRCON 2019 - Proceedings, № 8879981, 2019, pp. 1007-1011. DOI: 10.1109/UKRCON.2019.8879981

[15] Kornienko, Y.M., Liubeka, A.M., Sachok, R.V., Korniyenko, B.Y. Modeling of heat exchangement in fluidized bed with mechanical liquid distribution. ARPN Journal of Engineering and Applied Sciences, 14 (12), 2019, pp. 2203-2210.

[16] Babak V.P., Babak S.V., Myslovych M.V., Zaporozhets A.O., Zvaritch V.M. Methods and models for information data analysis. Studies in Systems, Decision and Control, 281, 2021, pp. 23-70. https://doi.org/10.1007/978-3-030-44443-3_2

[17] Babak, V., Shchepetov, V., Nedaiborshch, S. Wear resistance of nanocomposite coatings with dry lubricant under vacuum. Scientific Bulletin of National Mining University Issue 1, 2016, pp. 47-52.

[18] Shymkovych, V., Telenyk, S. & Kravets, P. Hardware implementation of radial-basis neural networks with Gaussian activation functions on FPGA. Neural Computing and Applications 33, 2021, pp. 9467–9479. https://doi.org/10.1007/s00521-021-05706-3

[19] Kravets P., Shymkovych V. Hardware Implementation Neural Network Controller on FPGA for Stability Ball on the Platform. In: Hu Z., Petoukhov S., Dychka I., He M. (eds) Advances in Computer Science for Engineering and Education II. ICCSEEA 2019. Advances in Intelligent Systems and Computing, vol 938, 2020. Springer, Cham. https://doi.org/10.1007/978-3-030-16621-2_23

[20] Yudin, O., Ziubina, R., Buchyk, S., Matviichuk-Yudina, O., Suprun, O., Ivannikova, V. Development of methods for identification of informationcontrolling signals of unmanned aircraft complex operator, Eastern-European Journal of Enterprise Technologies, vol. 2, no. 9-104, 2020, pp. 56-64.

[21] Savchenko, V., Laptiev, O., Kolos, O., Lisnevskyi, R., Ivannikova, V., Ablazov, I. Hidden Transmitter Localization Accuracy Model Based on Multi-Position Range Measurement, Proceeding of the 2nd IEEE International Conference on Advanced Trends in Information Theory, 2020, pp. 246-249.

[22] Yudin, O., Hahanova, A., Parkhomenko, M., Shmakov, V., Shaigas, O. Method of encoding binary structures of stationary component of video stream: Proceeding of the International Conference on Advanced Trends in Information Theory (ATIT 2020), Kyiv, Ukraine, 25.11.2020-27.11.2020, pp. 68-71.

[23] Kornienko, Y.M., Sachok, R., Tsepkalo, O.V. Modelling of multifactor processes while obtaining multilayer humic-mineral solid composites. Chemistry, 20 (3), 2011, E19-E26.

[24] Kornienko, Ya.N., Podmogilnyi, N.V., Silvestrov, A.N., Khotyachuk, R.F. Current control of product granulometric composition in apparatus with fluidized layer. Journal of Automation and Information Sciences, 31 (12), 1999, pp. 97-106.