EURASIAN JOURNAL OF TECHNOLOGY AND INNOVATION
Innovative Academy Research Support CenterIF= 7.7 www.in-academy.uz

DEVELOPMENT OF MODELS FOR MANAGING AN ECOLOGICAL INFORMATION PROTECTION SYSTEM BASED ON PETRI NETS

Inomjon Yarashov[1,2]
Mirabbos Akbarov[1]
[1]Diplomat university
[2]The University of World Economy and Diplomacy
e-mail: iyarashov@uwed.uz
https://doi.org/10.5281/zenodo.15400883

ARTICLE INFO

Received: 08th May 2025
Accepted: 12th May 2025
Online: 13rd May 2025

KEYWORDS

Protection system, complex environmental information system, hypergraph, Petri nets, system model, environmental information processing model.

ABSTRACT

This research focuses on modeling a security infrastructure composed of applications that utilize Petri nets. A core challenge in ensuring the safety of such systems arises from the need to handle vast volumes of environmental data related to the condition of the protected object, potential risks, and the generation of control responses. In this work, Petri net formalism is applied to represent the information flow and decision-making processes within the protection system's components. The structural and logical configuration of the system is modeled as a hypergraph, where its elements are defined as control nodes (CNs). These nodes are categorized based on their operational roles: detection nodes (DN), access nodes (AN), monitoring nodes (MN), delay nodes (DlN), and control mechanisms such as authorization and authentication modules (AuthM, AuthzM). The study also introduces detailed models for environmental data handling within specific protection units, including the abnormal movement detector, logic controller, central management unit, and a universal regulatory module.

Introduction

The protection system (PS) represents a combination of organizational, legal, and normative approaches designed to detect, counteract, and neutralize a wide range of threats targeting the system [1–8]. Contemporary protection architectures are intricate ecosystems that process environmental data and incorporate various subsystems, such as monitoring and auditing units, access regulation and control modules, cyber threat alert mechanisms, and more. The integration of environmental information across individual components of the PS is typically implemented using Petri net–based architectures. The administration of such an environmental information framework [9–14] is highly complex, primarily due to the necessity of processing extensive data regarding system conditions, threat emergence, and corresponding security management actions.

Given the critical nature and widespread application of protection systems in safeguarding various entities — ranging from environmental information infrastructures to high-risk facilities vulnerable to information threats [15–18] (e.g., intrusion detection systems,

data leakage prevention mechanisms, identification modules) — it becomes essential to construct an environmental information architecture capable of delivering rapid and efficient defense. Addressing the challenges of security management [19–22] calls for the development of models that can process situational data, relying on the logical relationships formed among system components in response to specific threat scenarios. At present, there is a lack of mathematical models that effectively account for and optimize resource utilization during the operational phases of a protection system. To bridge this gap, this work advocates for the use of Petri net–based models to describe and simulate the dynamic interactions among system elements.

Main part

The protection system encompasses a collection of software components that incorporate the following modules:

– a security and monitoring unit;

– an access regulation component;

– modules responsible for surveillance and auditing;

– subsystems for temporal coordination and dynamic event handling;

– supporting elements such as privilege assignment and emergency defense mechanisms [3,4].

In present-day implementations, advanced protection infrastructures are frequently designed using Petri nets at varying levels of complexity [5]. Typically, an integrated security framework includes:

- A central control hub equipped with a unified security panel and identifiers, or automated security modules;
- Security controllers interfaced with Petri nets;
- APIs linked to security control units;
- Application interfaces tailored for specific operational goals;
- Connection handlers and authentication controllers, among others [6,7].

One of the core difficulties in overseeing the security of such systems lies in processing extensive incoming environmental data and generating control responses that are contextually optimal. To implement robust and responsive security administration, it is vital to develop a precise mathematical representation of component operations.

In the process of constructing such a mathematical model, it is necessary to abstract away from the physical characteristics of the protection system's hardware components. At the base level of the system's structural-logical model, these components can be generalized as control points (CP), each potentially comprising multiple protection mechanisms unified by a shared objective. These CPs are functionally categorized as: detection points (DP), access nodes (AP), surveillance points (OP), and delay modules (DeP) [8]. Additionally, system-level control is maintained by authentication and authorization entities, referred to as AcM and AzM respectively.

For the resulting environmental information processing models within the protection framework to be effective, they must accurately capture the logical interplay among subsystem components. This can best be achieved using hypergraph-based modeling techniques, which offer the following strengths:

– ability to express many-to-many associations;

– support for hierarchical modeling, where each node can itself represent a graph or a nested hypergraph as complexity increases;

– suitability for implementing optimization algorithms;

– flexibility to treat the hypergraph as a collection of subsets, thus leveraging advanced principles of graph theory [6,9].

Let's imagine the protection system as a hypergraph (Fig. 1):

$$H = \{V, E\} \quad (1)$$

where V is a set of vertices, which are separate structural and logical elements of the protection system $V = \{V_1, V_2, \dots, V_n\}$.

E - a set of edges representing information relationships between individual elements of the protection system $E = \{E_1, E_2, \dots, E_n\}$

Each edge E of graph H can be described as a subset of vertices:

$$V_j = \{1, \quad 2, \dots, D\}$$

Where, $E_1 \in V$

When representing the security system of an ecological information system as a hypergraph, each edge E of the hypergraph H corresponds to some nested scheme of inter-network processes as an element of the sub-network architecture.

A route in a hypergraph will be $H = \{V, E\}$quence of vertices and edges of the form:

$$V_1, E_1, V_2, E_2, \dots, E_n, V_n, V_i \in V, i \in [0; n],$$

where $E_i \in E, (V_i, E_i) \in I, i \in [1; n]$ (3)

Through the decomposition process, the protection system's hypergraph model can be broken down into subprocesses that reflect the operational behavior of its individual components. To formally describe these subprocesses, the use of Petri nets is proposed—an approach well-suited for representing parallel execution, asynchronous interactions, and hierarchical organization within the system's architecture [6,10–13].

At the preliminary modeling phase, specific Petri net–based models were designed to capture how environmental data is processed by key protection system components, namely: the threat analyzer, the authentication controller, the central coordination manager, and the universal regulation module.

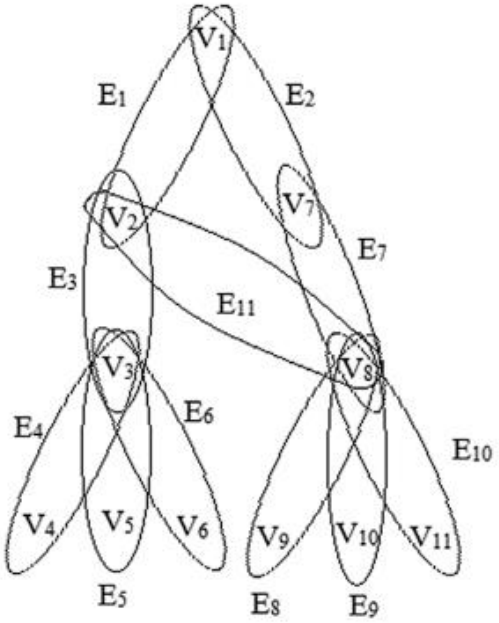

Fig 1. Fragment of the hypergraph of the security system[23]

The Petri net representing the operational logic of the analyzer (see Fig. 2) outlines its ability to autonomously initiate notification procedures and transmit actuation data to the control panel, aligned with the session layer of the ISO/OSI reference model. The analyzer is assumed to be functioning under standard conditions.

The necessary preconditions for establishing a valid connection within the system include the following:

1. Network integration of the analyzer;
2. Successful authorization by the security management unit;
3. The analyzer entering a standby operational state;
4. Detection of a malicious actor (hacker);
5. Dispatch of a corresponding notification alert;
6. Resumption of normal analyzer activity;
7. Disconnection or removal of the analyzer from the security infrastructure.

Relevant events occurring within the network context are categorized as:

1. Exchange of initialization packets;
2. Reception of threat-related data (hacker information);
3. Issuance of a system notice;
4. Dissemination of warning messages;
5. Restoration of the analyzer's active status;
6. Termination of the communication link.

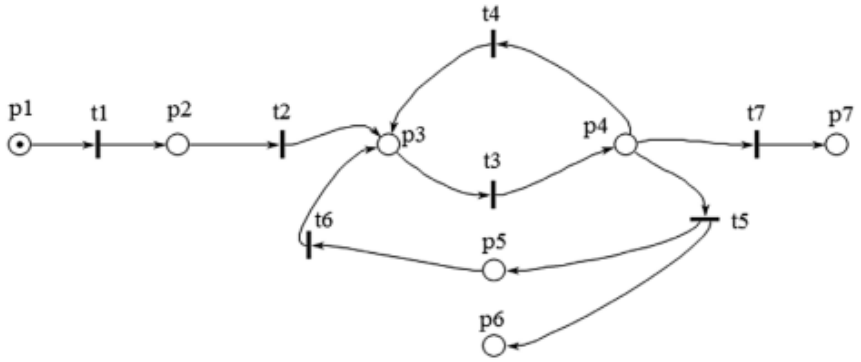

Fig 2. Description of the analyzer operation process in the form of a Petri net[24]

In the corresponding Petri net, place p1 indicates that the analyzer is activated and operational.

Place p2 represents the phase during which the analyzer processes the session initialization packet.

A token in p3 reflects the active state of the analyzer—specifically, that it is currently receiving data, whereas place p4 corresponds to the phase of data processing.

Place p5 denotes the alert or alarm mode, signifying the detection of a potential threat.

Place p6 serves as an aggregation node where incoming data is accumulated; this position is associated with the system's security control panel, reflecting a state of centralized information collection.

Lastly, place p7 indicates the termination of analyzer activity, marking the end of the analysis process.

The transitions (denoted by t) within this net describe the following processes:

t1: The analyzer receives an initialization packet from the authentication and authorization module;

t2: The response packet is returned;

t3: The analyzer transmits collected data to the authentication and authorization module, which subsequently relays it via the main manager to the central control panel;

t4: No intrusions are detected, and the analyzer transitions back to its normal operational state;

t5: An intruder (hacker) is identified, and the relevant information is forwarded to the control unit;

t6: The notification process ends, and the analyzer resumes its standard operations;

t7: The analyzer receives a packet indicating session termination and proceeds to shut down.

Additionally, extended input (I) and output (O) functions are specified, allowing for a detailed formalization of token flow across the Petri net.

$C = (P, T, I, O)$,

$P = \{p1, p2, p3, p4, p5, p6, p7\}$,

$T = \{t1, t2, t3, t4, t5, t6, t7\}$,

$I(p1) = \{0\}\ I(p2) = \{t1\}$

$I(p3) = \{t2, t6, t4\}\ I(p4) = \{t5\}$

$I(p5) = \{t5\}\ I(p6) = \{t5\}\ I(p7) = \{t7\}$

$O(p1) = \{t1\}\, O(p2) = \{t2\}\, O(p3) = \{t3\}$
$O(p4) = \{t4, t5, t7\}\, O(p5) = \{t6\}\, O(p6) = \{0\}$
$O(p7) = \{0\}$
$I(t1) = \{p1\}\, I(t2) = \{p2\}\, I(t3) = \{p3\}$
$I(t4) = \{p4\}\, I(t5) = \{p4\}$
$O(t1) = \{p2\}\, O(t2) = \{p3\}$
$O(t3) = \{p4\}\, O(t4) = \{p3\}$
$O(t5) = \{p5, p6\}\, O(t6) = \{p3\}\, O(t7) = \{p7\}$

Let's analyze the Petri net on the basis of matrix equations. An alternative to the definition of the Petri net in the form (P, T, I, O) is the definition of two matrices D^+ *and* D^-, where $D = D^+ - D^-$- is a composite matrix of changes. Each matrix has m rows (one per transition) and n columns (one per position). $D^-[j, i] = (pj, I(tj))$, and $D^+[j, i] = (pj, O(tj))$ are defined. D^- defines transition inputs, D^+ - outputs.

$$D^- = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$D^+ = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} -1 & +1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & +1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & +1 & 0 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & +1 & +1 & 0 \\ 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & +1 \end{bmatrix}$$

In the initial marking μ = (1,0,0,0,0,0) the transition t1 is allowed and leads to the marking μ`, where

$$\mu` = (1,0,0,0,0,0) + (0,1,0,0,0,0) \cdot$$

$$\cdot \begin{bmatrix} -1 & +1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & +1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & +1 & 0 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & +1 & +1 & 0 \\ 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & +1 \end{bmatrix} =$$

$$= (0,0,0,0,1,1)$$

Next (Fig. 3) is a fragment of the reachability tree of the imitation model of the analyzer

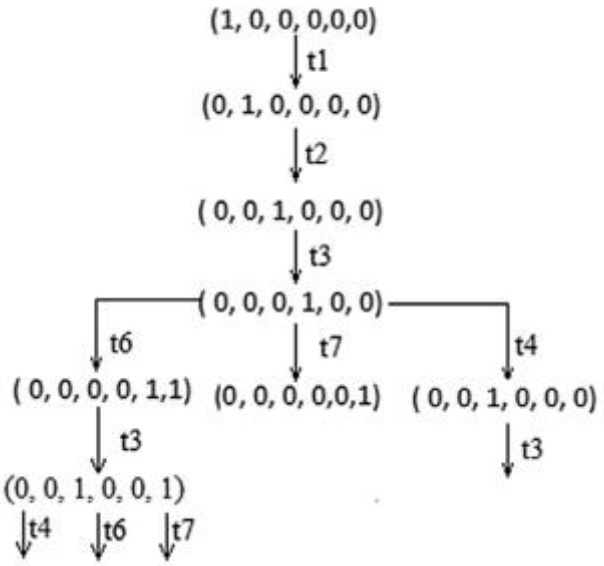

Fig 3. Fragment of the reachability tree of the imitation model of the analyzer[25]

The triggering of transition t6 leads to the accumulation of tokens in place p6, which reflects the ongoing process of collecting data on the analyzer's operational cycle. This part of the Petri net model represents the analyzer's behavior under standard operating conditions, where a potential hacker alert may be generated. Over time, the analyzer aggregates information concerning the number of operational cycles it completes.

Subsequently (as shown in Fig. 4), a Petri net model is introduced for describing the behavior of the Authentication and Authorization Manager (AAM) at the session layer of the OSI/ISO model.

The existence of this model presumes the following system-level conditions:

A functioning Security Control Unit (SCU) is present;

One or more analyzers are connected according to defined system structures;

The presence of a potential hacker threat is acknowledged;

The Authentication and Authorization Manager is correctly connected and integrated within the protection system;

All components are developed by the same entity (ensuring program-level compatibility).

Relevant system events include:

The connection of the AAM to the protection system;

The establishment and maintenance of a connection session;

The reception of operational information from the analyzer;

Notification of a disrupted or lost connection.

These preconditions and events form the basis for simulating the operation of the authentication and authorization manager. Notably, the possibility of malfunctions in the Security Control Unit is not excluded, and all interactions with the SCU are carried out through the default security manager module.

The Petri net positions (places) are defined as follows:

p1: The AAM is powered off but physically connected to the system;

p2: The AAM is in a standby state, awaiting session establishment packets;

p3: The AAM has generated a notification packet indicating the absence of a connection with the analyzer and forwards this to the SCU;

p4: The AAM attempts to establish (initialize) a session with the SCU;

p5: The AAM has successfully established a connection session with both the analyzer and the SCU;

p6: The AAM confirms that the analyzer is in standard operation mode and is ready to receive further packets;

p7: The AAM awaits an acknowledgment packet, and this place also represents the expiration of the timeout for waiting;

p8: The AAM remains in a waiting state for response packets from either the analyzer or the SCU.

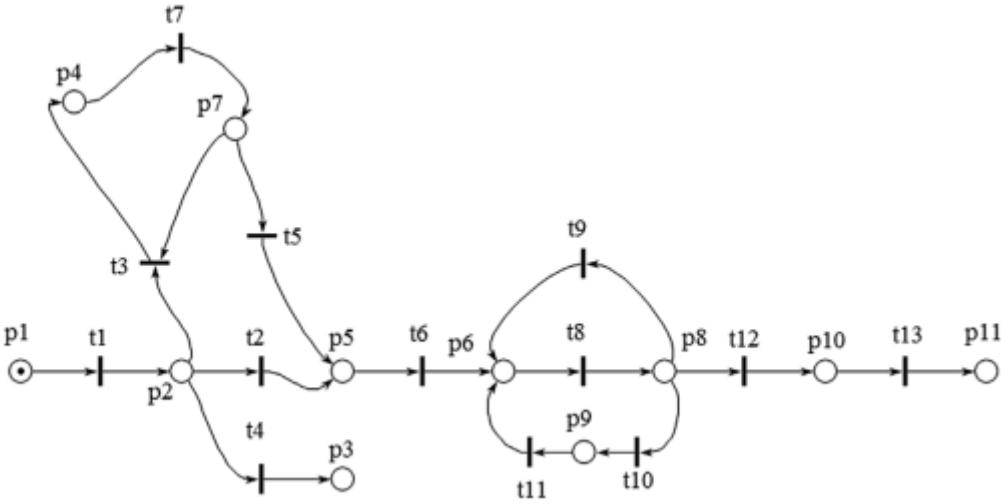

Fig. 4. Description of the process of operation of the authentication manager in the form of a Petri net[24]

Position p9 represents the state in which the security block constructs a data packet containing information about the operation of the analyzer(s) and prepares it for transmission to the security control block.

Position p10 indicates that the authentication and authorization manager is currently processing a response packet received from the security control unit.

Position p11 implies that the authentication and authorization manager is now disabled or shut down.

The transitions in this Petri net are defined as follows:

t1: Sending a packet to initiate a session between the analyzer and the security control unit, including the initialization of the main dispatcher;

t2: Receiving packets indicating successful session establishment with both the security control unit and the analyzers;

t3: Receiving a session establishment confirmation from the analyzer, but not from the security control unit—thus leading to an incomplete initialization of the authentication and authorization manager;

t4: Receiving a session confirmation packet from the security control unit but not from the analyzer;

t5: Confirmed successful establishment of a session with the security control unit;

t6: The authentication and authorization manager begins its primary operational phase;

t7: Sending a retry request to establish a session with the security control unit;

t8: Maintaining the connection session by sending a keep-alive (empty) packet to the analyzer and receiving a similar packet from the main manager;

t9: Receiving an empty packet from the analyzer and responding with its own to the manager;

t10: Receiving a data packet from the analyzer containing information on its current operational state;

t11: Transmitting operational data from the authentication and authorization manager to the security control unit and resuming normal operation;

t12: Receiving a session termination notification from the security control unit;

t13: Sending a session termination message to the analyzer from the authentication and authorization manager.

The input (I) and output (O) functions of the network are formally defined, and network analysis using matrix equations (incidence matrix, marking vector, transition vector) is carried out in a similar manner as was done for the analyzer model.

Petri net models were constructed for the main security manager and the universal security control unit. Their corresponding process diagrams are presented in Figures 5 and 6, respectively.

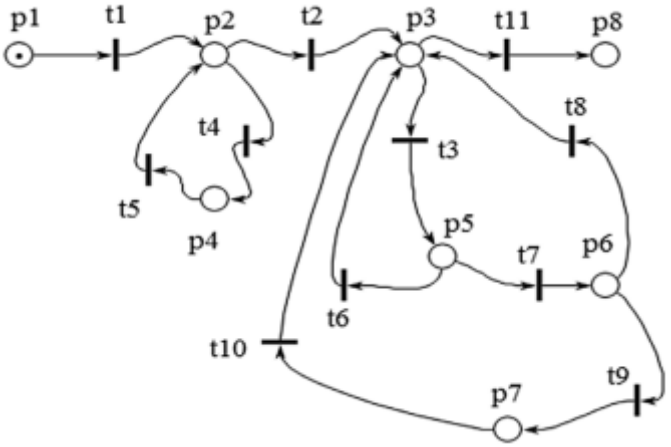

Fig 5. Description of the process of operation of the main security manager in the form of a Petri net[24]

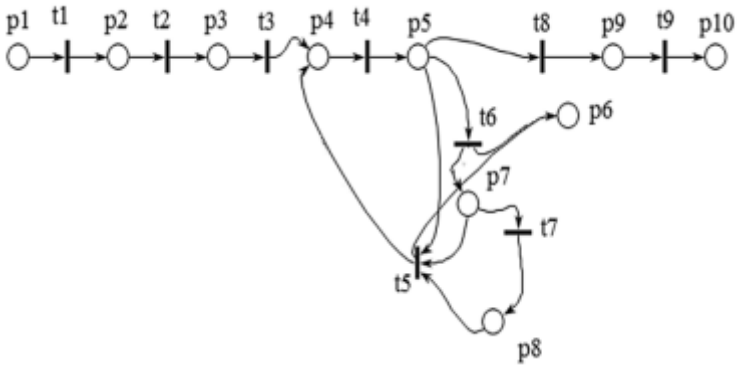

Fig 8. Description of the process of operation of the universal security control unit in the form of a Petri net[24]

Conclusion

One of the key advantages of Petri nets lies in their mathematically rigorous formalism, which enables the comprehensive analysis of models using modern information technologies. This feature makes Petri nets particularly suitable for application in complex environmental information systems within the context of safety and security management. Accordingly, models of information processing have been developed to reflect the functioning of critical elements of the protection system, such as the action analyzer, authentication and authorization manager, general manager, and universal control unit. In future work, the development of a generalized hypergraph model is planned to capture the interactions among these protection system components, taking into account potential security threats. The resulting Petri net-based models support automated analysis and facilitate multi-level system descriptions, enabling seamless transitions between various levels of abstraction by selectively enabling or disabling transitions. These models may serve as a foundation for the implementation of intelligent systems for the protection of environmental information. The integration of Petri net models into the security framework is expected to enhance the quality of security management. Specifically, such models allow for more precise coordination of system components within an attacked zone, by recognizing and accounting for

interdependencies among elements and identifying priority security tasks based on the evolving system state.

References:

1. A. Kabulov, I. Kalandarov and I. Yarashov, "Problems Of Algorithmization Of Control Of Complex Systems Based On Functioning Tables In Dynamic Control Systems," 2021 International Conference on Information Science and Communications Technologies (ICISCT), 2021, pp. 1-4, doi: 10.1109/ICISCT52966.2021.9670017.
2. A. Kabulov and I. Yarashov, "Mathematical model of Information Processing in the Ecological Monitoring Information System," 2021 International Conference on Information Science and Communications Technologies (ICISCT), 2021, pp. 1-4, doi: 10.1109/ICISCT52966.2021.9670192.
3. A. Kabulov, I. Yarashov and A. Otakhonov, "Algorithmic Analysis of the System Based on the Functioning Table and Information Security," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2022, pp. 1-5, doi: 10.1109/IEMTRONICS55184.2022.9795746.
4. A. Kabulov, I. Saymanov, I. Yarashov and F. Muxammadiev, "Algorithmic method of security of the Internet of Things based on steganographic coding," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2021, pp. 1-5, doi: 10.1109/IEMTRONICS52119.2021.9422588.
5. I. Yarashov, "Algorithmic Formalization Of User Access To The Ecological Monitoring Information System," 2021 International Conference on Information Science and Communications Technologies (ICISCT), 2021, pp. 1-3, doi: 10.1109/ICISCT52966.2021.9670023.
6. A. Kabulov, I. Normatov, I. Kalandarov and I. Yarashov, "Development of An Algorithmic Model And Methods For Managing Production Systems Based On Algebra Over Functioning Tables," 2021 International Conference on Information Science and Communications Technologies (ICISCT), 2021, pp. 1-4, doi: 10.1109/ICISCT52966.2021.9670307.
7. A. Kabulov, I. Saymanov, I. Yarashov and A. Karimov, "Using Algorithmic Modeling to Control User Access Based on Functioning Table," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2022, pp. 1-5, doi: 10.1109/IEMTRONICS55184.2022.9795850.
8. Kabulov A. V., Yarashov I. K., Jo'Rayev M. T. Computer viruses and virus protection problems //Science and Education. – 2020. – T. 1. – №. 9. – C. 179-184.
9. Kabulov A. et al. Algorithmic method of security of the Internet of Things based on steganographic coding. 2021 IEEE International IOT //Electronics and Mechatronics Conference, IEMTRONICS. – 2021.
10. Madrahimova D., Yarashov I. Limited in solving problems of computational mathematics the use of elements //Science and Education. – 2020. – T. 1. – №. 6. – C. 7-14.
11. Kabulov A., Yarashov I., Vasiyeva D. SECURITY THREATS AND CHALLENGES IN IOT TECHNOLOGIES //Science and Education. – 2021. – T. 2. – №. 1. – C. 170-178.
12. Kabulov A., Muhammadiyev F., Yarashov I. ANALYSIS OF INFORMATION SYSTEM THREATS //Science and Education. – 2020. – T. 1. – №. 8. – C. 86-91.

13. Gaynazarov S. M. et al. ALGORITHM OF MOBILE APPLICATION FOR MEDICINE SEARCH //Science and Education. – 2020. – Т. 1. – №. 8. – С. 600-605.
14. Кабулов А. В. Шерзод Туйлибоевич Болтаев, and Гулдофарид Муроджоновна Хабибжонова.«АЛГОРИТМИЧЕСКИЕ АВТОМАТНЫЕ МОДЕЛИ И МЕТОДЫ СОЗДАНИЯ РАСПРЕДЕЛЕННЫХ МИКРОПРОЦЕССОРНЫХ СИСТЕМ УПРАВЛЕНИЯ И ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ.» //WORLD SCIENCE: PROBLEMS AND INNOVATIONS. – 2019.
15. Yarashov I., Normatov I., Mamatov A. THE STRUCTURE OF THE ECOLOGICAL INFORMATION PROCESSING DATABASE AND ITS ORGANIZATION //International Conference on Multidimensional Research and Innovative Technological Analyses. – 2022. – С. 114-117.
16. Кабулов А. В. и др. АЛГОРИТМИЧЕСКИЕ АВТОМАТНЫЕ МОДЕЛИ И МЕТОДЫ СОЗДАНИЯ РАСПРЕДЕЛЕННЫХ МИКРОПРОЦЕССОРНЫХ СИСТЕМ УПРАВЛЕНИЯ И ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ //WORLD SCIENCE: PROBLEMS AND INNOVATIONS: сборник статей XXIX. – 2019. – С. 40.
17. Yarashov I., Normatov I., Mamatov A. ECOLOGICAL INFORMATION PROCESSING TECHNOLOGIES AND INFORMATION SECURITY //International Conference on Multidimensional Research and Innovative Technological Analyses. – 2022. – С. 73-76.
18. Kabulov A., Yarashov I., Mirzataev S. DEVELOPMENT OF THE IMPLEMENTATION OF IOT MONITORING SYSTEM BASED ON NODE-RED TECHNOLOGY //Karakalpak Scientific Journal. – 2022. – Т. 5. – №. 2. – С. 55-64.
19. Бабаджанов А. Ф. и др. АЛГОРИТМИЧЕСКИЙ АНАЛИЗ СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ НА ОСНОВЕ ТАБЛИЦ ФУНКЦИОНИРОВАНИЯ //International Journal of Contemporary Scientific and Technical Research. – 2022. – С. 216-219.
20. I. Yarashov, "Development of a reliable method for grouping users in user access control based on a Functioning table," 2022 International Conference on Information Science and Communications Technologies (ICISCT), 2022, pp. 1-5.
21. I. Normatov, I. Yarashov, A. Otakhonov and B. Ergashev, “Construction of reliable well distribution functions based on the principle of invariance for convenient user access control,” 2022 International Conference on Information Science and Communications Technologies (ICISCT), 2022, pp. 1-5.
22. S. Toshmatov, I. Yarashov, A. Otakhonov and A. Ismatillayev, “Designing an algorithmic formalization of threat actions based on a Functioning table,” 2022 International Conference on Information Science and Communications Technologies (ICISCT), 2022, pp. 1-5.
23. Sukhoverkhov A.S. Metodicheskiy podkhod k modelirovaniyu funktsionirovaniya sredstv zashchity informatsii na osnove primeneniya apparata teorii setey Petri-Markova, Telekommunikatsii. 2012. No 8, pp. 41-48.
24. Мальков, М. В., and С. Н. Малыгина. "Сети Петри и моделирование." Труды Кольского научного центра РАН 3 (2010): 35-40.
25. Wang, Shouguang, et al. "A reduced reachability tree for a class of unbounded Petri nets." IEEE/CAA Journal of Automatica Sinica 2.4 (2015): 345-352.