# OSTIS Ecosystem Security Problems

Vasili Khoroshavin, Vladimir Zakharov
*Belarusian State University of*
*Informatics and Radioelectronics*
Minsk, Belarus
Email: vasya.khoroshavin@gmail.com, zakharov@bsuir.by

***Abstract*—In this article the threats and vulnerabilities relevant for ostis-systems are examined. The differentiation of access to the knowledge bases of ostis-systems, the implementation of mechanisms of configuration of a personal ostis-assistant and the safety of agents' source code are defined as the main directions for ensuring security of ostis-systems. Options for implementing the according security mechanisms in those directions are proposed.**

***Keywords*—security, knowledge base, ostis-system, OSTIS Ecosystem, knowledge processing.**

## I. Introduction

In the era of digitalization and the rapid growth of information technologies, the safety issues of information systems become more and more relevant and significant. The next-generation intelligent computer systems, operating with global knowledge bases and capable of autonomous decision-making, open new horizons for various spheres of human activity. From the financial sector and health care to space research — the potential for the use of such systems is unlimited. Even the cybersecurity area is not an exception [1], [2], [3], [4]. However, it is worth noting that safety requirements in different areas are not the same, which is confirmed by existing standards in the field of information security (for example, an evaluation assurance level [5]). Consequently, with the expansion of areas of the use of intelligent computer systems, the criticality of the implementation of potential threats of security increases as well, which makes the task of protecting them especially relevant and complex.

Modern intelligent systems, in contrast to traditional computer systems and neural networks, are a combination of global knowledge bases and problem solvers built using multi-agent architecture, which puts special requirements for ensuring security. The problem is not only in protecting data from unauthorized access, but also in ensuring the protection of the decision-making process itself. The aspects associated with the addition of a new level of processing — knowledge processing — must also be considered. This entails the necessity to take into account the new corresponding vulnerabilities and threats. Also, one of the features of the next-generation intelligent computer systems that should be noted is their interoperability [6], which entails the need to unite them into collectives to jointly solve problems, whereby the OSTIS Ecosystem is formed [6].

Currently the most attention is paid to machine learning methods. Despite they show good quality in several cases vulnerabilities that might be exploited are created as well [7]. So this article is focused on semantic aspects of AI and in this work as its main goal was set to consider threats and vulnerabilities that are relevant for ostis-systems and ostis-communities, as well as propose ways for solving certain security problems at the level of both individual ostis-systems and their collectives.

## II. Intelligent systems security vulnerabilities

Based on the ostis-systems architecture there are three main origins of vulnerabilities: devices running the system; the knowledge base; communication channels and communicative processes; incorrect actions of users.

Since nowadays there is no hardware implementation of semantic computers and only software implementation is used, the vulnerabilities related to devices aren't considered and are beyond the scope of this paper. It is also worth noting that in contradiction to the traditional systems two aspects of protection in the knowledge bases can be distinguished: data and knowledge. This means that such classic problem as access to fragments of knowledge bases is not the only one that must be solved. The correctness of their contents and the influence of the process of solving certain problems on the emergence of new data in the knowledge base, the presence of which in the public domain is unacceptable, must also be taken into account.

It is easy to notice that the only new issues are those related to knowledge processing. For this reason, when compiling a vulnerabilities hierarchy, CWE (Common Weakness Enumeration), namely the CWE-1000 view [8], was taken as basis.

Taking into account the aforementioned necessity to consider the features of knowledge processing, an analysis of the specifics of working with them was carried out. According to [9], knowledge has the following traits: connectivity, complex structure, (internal) interpretability, activity and presence of semantic metrics. The activity of knowledge is the source of activity in the knowledge-driven system [10]. So one of the main challenges is handling non-factors of knowledge. In practice, in almost all cases, knowledge possesses

certain non-factors. According to the classification given in [11], it is possible to decompose the set of non-factors $N$ into two types: $N_1$ and $N_2$ ($N = N_1 \cup N_2$). The first one contains those, which might be received from experts automatically. These are fuzziness, uncertainty, inaccuracy and under-determination. non-factors of the second type, respectively, include those that arise in other ways (incompleteness, inconsistency, incorrectness, non-monotonicity). The second-type factors are characterized by their dynamic nature. For example, incorrectness of knowledge might easily flow into inconsistency, incompleteness can cause non-monotonicity and so on. The "cause of occurrence" dependency of second-type factors on the other factors might be described with the relation $C$ ($xCy$ means that $x$ is the cause of $y$ occurrence).

$$C \subset \{< x, y > |(x \in N) \wedge (y \in N_2) \wedge (x \neq y)\}$$

Notice that some of the mentioned factors can be detected automatically and their negative influence might be limited [12], [13]. Partly the solution of this problem depends on the knowledge engineers who must design knowledge bases taking into account the following requirements:

- origins of knowledge must be easily detected;
- uncertainty, inaccuracy and under-determination rates must be explicitly saved.

This thorough design and appropriate processing mechanisms will drastically increase knowledge bases security.

The main non-factors are the following:

1) Uncertainty is the factor preceding incompleteness. It is determined by the fact that this or that knowledge is set by a certain degree of confidence, which can have a complex nature.
2) Inaccuracy is a factor associated with the impossibility of accurately obtaining a particular value. For example, due to the error of measurement devices [14].
3) Under-determination is a factor that, unlike inaccuracy, is associated with the possibility of clarifying the value, but this is not necessary within the framework of a specific task [14].
4) Inconsistency is a state in which knowledge base contains fragments contradicting to each other.
5) Incompleteness is the absence of the elements in the knowledge base. The criteria for completeness in the knowledge base is defined by a set of formal statements about completeness [15].

So there are the following ostis-systems vulnerabilities related to the knowledge processing:

1) processing of uncertain knowledge as certain knowledge;
2) improper usage of inaccurate or under-determined knowledge;
3) certainty check mechanism failure (or its absence);
4) completeness check mechanism failure (or its absence);
5) inconsistency search mechanism failure (or its absence);
6) reliance on knowledge from untrusted sources;
7) knowledge verification violation during logical inference.

Top-level of formalized ontology of threats and vulnerabilities contains 11 basic classes of vulnerabilities. Their subclasses are more accurate vulnerabilities and provide the corresponding descriptions, so they might be used in practical evaluation of the systems.

***ostis-system vulnerability***
- ⊃ *improper access control*
- ⊃ *improper interaction between multiple correctly-behaving entities*
- ⊃ *improper control of a resource through its lifetime*
- ⊃ *incorrect calculation*
- ⊃ *insufficient control flow management*
- ⊃ *protection mechanism failure*
- ⊃ *incorrect comparison*
- ⊃ *improper check or handling of exceptional conditions*
- ⊃ *improper neutralization*
- ⊃ *improper adherence to coding standards*
- ⊃ *improper processing of knowledge non-factors*

According to [6] the OSTIS Ecosystem is a collective of interacting:

1) ostis-systems;
2) users of these ostis-systems (both end users and developers);
3) computer systems that are not ostis-systems, but considered by them as additional information resources or services.

Since systems that are not ostis-systems can also be actors in the Ecosystem, in the case of consideration of the entire Ecosystem, the CWE-1000 view with additions related to knowledge processing introduced into it will be completely inherited.

## III. Security threats for the intelligent systems

Taking the goals of security assurance in the traditional systems given in [16] the following goals of security assurance in the next-generation intelligent systems might be defined:

- ensuring the safety of semantic compatibility of information;
- protection of the reliability and integrity of information;
- ensuring the availability of information at different levels of the intelligent system;
- minimizing damage from events that pose a threat to information security.

Given the specifics of the operation of individual ostis-systems and ostis-communities that exist within the OSTIS Ecosystem [6] and knowledge-driven systems in general [17], the following threats can be distinguished.

***threat in the ostis-system***

- ⊃ *violation of the confidentiality of information*
  - ≔ [unauthorized reading of information]
- ⊃ *violation of the integrity of information*
  - ≔ [unauthorized or erroneous changes, distortion or destruction of information, as well as unauthorized impact on technical and software tools for processing information]
- ⊃ *violation of accessibility*
  - ≔ [blocking access to the system, its individual components, functions or information, as well as the impossibility of timely obtaining information (unacceptable delays in obtaining information)]
- ⊃ *violation of semantic compatibility*
  - ≔ [violation of the commonality of concepts and commonality of basic knowledge]
- ⊃ *destruction of the semantics of knowledge bases*
  - ≔ [semantic virus]
  - ≔ [replacement or removal of nodes and connections between them in the knowledge base]
- ⊃ *excessive volume of incoming information*
- ⊃ *violation of non-repudiation*
  - ≔ [the issuance of unauthorized actions for legitimate, as well as concealment or substitution of information about the actions of the subjects]
- ⊃ *violation of accountability*
  - ≔ [unauthorized or erroneous change, distortion or destruction of information about the performance of actions by the subject]
- ⊃ *violation of authenticity*
  - ≔ [performing actions in the system on behalf of another person or issuing unreliable resources (including data) for genuine]
- ⊃ *violation of reliability*
  - ≔ [deliberate or unintentional provision and use of erroneous (incorrect) or irrelevant (at a particular moment in time) information, as well as the implementation of procedures in violation of the order (protocol)]

Thus, one can distinguish several main areas for the implementation of protection measures in the OSTIS Ecosystem and its components:

- providing mechanisms for access differentiation to knowledge bases;
- organization of safe mechanisms of communication and interaction of ostis-systems within the ostis-communities;
- implementation of mechanisms for protecting the interaction process of a personal ostis assistant and user;
- implementation of the mechanisms of automated verification of the source code of agents of ostis-systems.

## IV. Potential solutions

### A. Differentiation of access to knowledge base

When solving the problem of differentiation of access to ostis-system knowledge base there are two main subproblems: access differentiation to the entire knowledge base and access differentiation to specific fragments of the knowledge base.

The main difficulty arising in the design of the ontology of the subject domain of users is that, firstly, it is necessary to guarantee quick access to access rights for a particular sc-element, and, secondly, minimize the number of additionally created sc-elements in the knowledge base.

The formation of structures in the knowledge base for which a special access policy will be configured entails the creation of additional sc-connectors. Those will not be isomorphic to the search template for access rights for a particular sc-element. For example, on the Fig. 1 those would be edges *edge 1*, *edge 2* and *edge 3*.

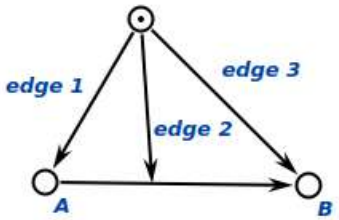


Figure 1: Example of sc-connectors that are not isomorphic to the search template for access rights for an sc-element.

Taking this into account it is necessary to implement a special mechanism of access rights detection for them. In order to do so the special bit in the sc-address of an element will be used. This bit set to 1 will indicate that the sc-element belongs to the class of sc-elements that can be edited only with administrator rights.

The current implementation of the ostis-platform uses linked lists for storing elements incident with a specific sc-element, so in order to decrease access time to the sc-node denoting the fragment of the knowledge base all those access edges must be placed in the beginning of the linked list. To represent and process access rights, the following concepts in the knowledge base are defined: ***user***, ***group of users***, ***user action class within sc-structure****, ***knowledge base reading***, ***knowledge base editing***, ***interpretation of an scp-program***.

The proposed scheme of access rights processing is described by the attribute based access control (ABAC)

model. Users and groups of users act as subjects of impact in ostis-systems. As a result of the analysis of the classes of operations on sc-memory, we define 3 main classes of actions which will build the decomposition of the set of operations. These are **reading** (as a result of search in knowledge base), **writing** and **execution** of a program, represented in sc-code. An example of a fragment of the knowledge base, recorded in the SCg language, with the access rights specified for it is presented in the Fig. 2. The construction with the access right for *user 2* to edit all knowledge base is presented in the Fig. 3. The construction with the access right for *group 2* to edit a fragment of the knowledge base is presented in the Fig. 4.

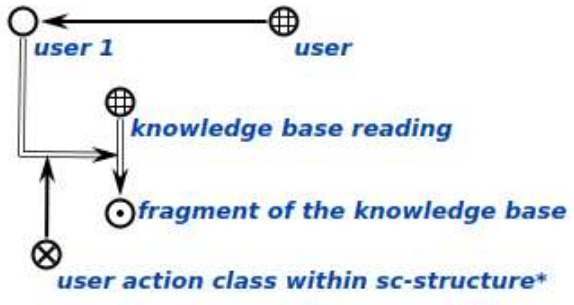


Figure 2: Formal representation of a user's right to edit the fragment of the knowledge base.

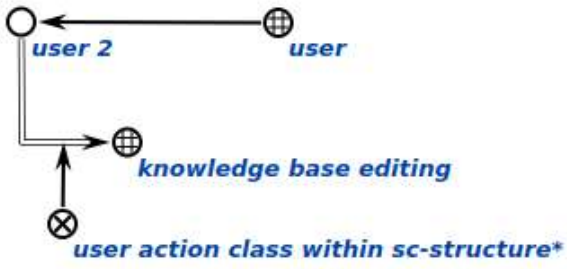


Figure 3: Formal representation of a user's right to edit all knowledge base.

*B. Options for solving security problems of the OSTIS Ecosystem*

One of the fundamental advantages of the ostis-systems is their high level of interoperability, which eases their ability to form collectives (ostis-communities) for cooperative solving of problems, as well as formation of the digital ecosystem, referred to as the OSTIS Ecosystem, on the basis of these communities [6]. So provision of secure functioning of the entire OSTIS Ecosystem is one of the priority directions for research.

Among the existing communication protocols, special attention should be paid to the Matrix protocol. It is a set

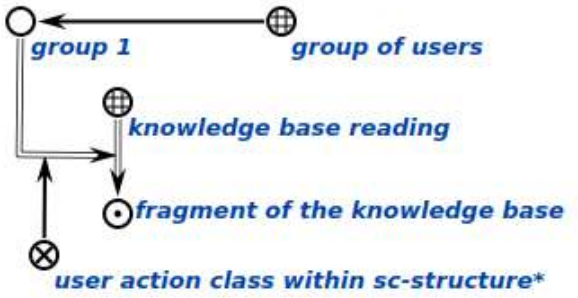


Figure 4: Formal representation of a group of users' right to edit the fragment of the knowledge base.

of open APIs for decentralised communication, suitable for securely publishing, persisting and subscribing to data over a global open federation of servers with no single point of control [18]. For communication in Matrix protocol virtual "rooms" are used. The local copies of their descriptions are stored on homeservers and are automatically synchronized between each other. Fig. 5 shows a schematic example of a room.

Nodes *@alice:alice.com*, *@bob:bob.com* and *@charlie:charlie.com* represent the clients of the end users and *matrix.alice.com*, *matrix.bob.com* и *matrix.charlie.com* represent the homeservers.

Based on the specification of this protocol, the following structural elements of the Ecosystem can be distinguished: ***ostis-community***, ***room***, ***storage***, ***homeserver***. The Fig. 6 shows a fragment of the description of the OSTIS Ecosystem.

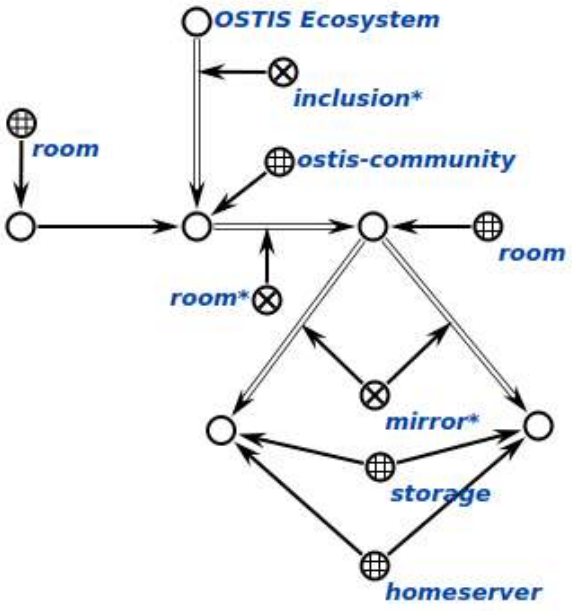


Figure 6: A fragment of the OSTIS Ecosystem structure.

Every community in the Ecosystem will create a

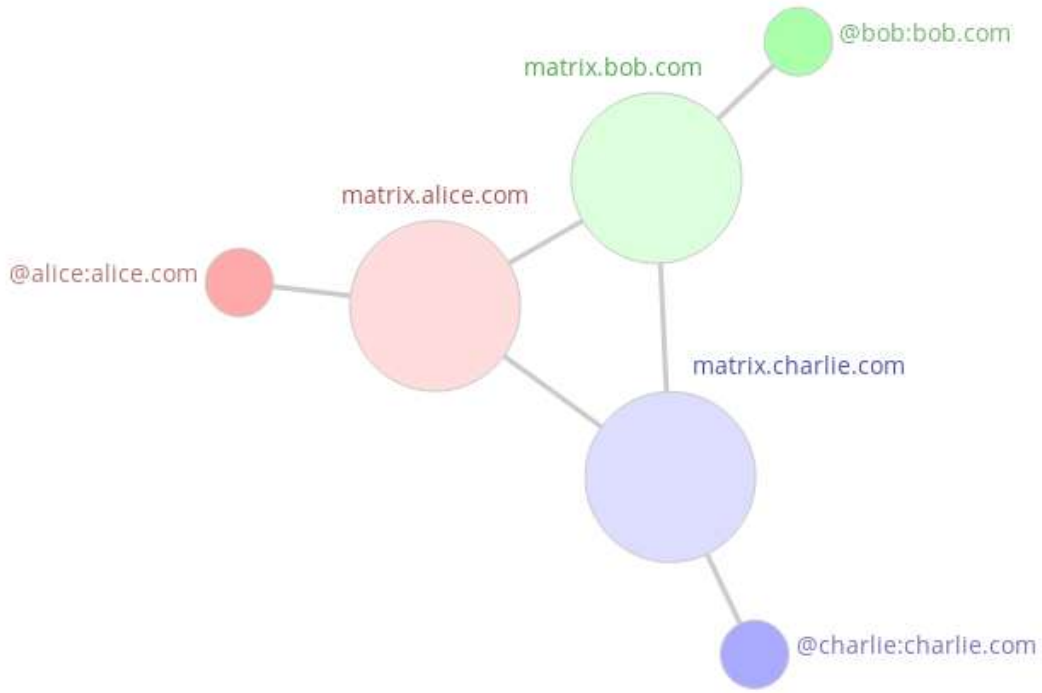


Figure 5: Structural scheme of a room in the Matrix protocol.

corresponding room. Any ostis-community can be a member of any number of rooms. Resources distributed by communities will be kept on a specialized servers. These are referred as storages and also will be used as homeservers. The given protocol might be used with enabled end-to-end encryption in order to implement private ostis-communities. This structure might be used to create the hierarchy of ostis-systems collectives in OSTIS Ecosystem by simultaneous use of the corporate ostis-systems as clients and homeservers.

It is also important to take into account the necessity of verification of the sources of fragments of knowledge bases (including agents) transmitted over the network within the Ecosystem. This task can be solved by usage of the existing digital signature protocols (for example, OpenPGP [19]).

### *C. Security of the personal ostis-assistant*

A personal ostis-assistant is an ostis-system that provides comprehensive adaptive maintenance of a particular user on all issues related to his interaction with any other ostis-systems, as well as representing the interests of this user through the entire global network of ostis-systems [6]. Combination of a (human) user of the ostis-system and its corresponding personal assistant is a minimal ostis-community, where personal assistant takes a role of a ***corporate ostis-system**** of this community as shown on the Fig. 7.

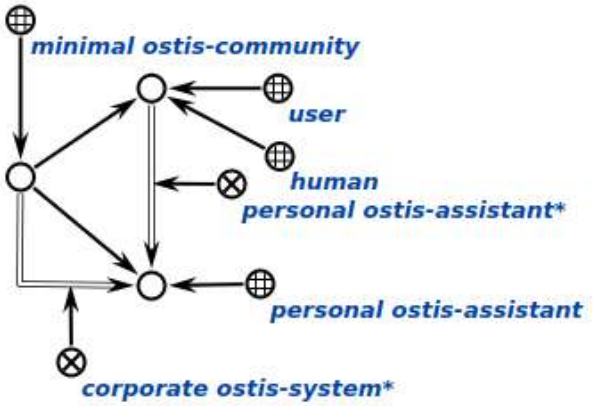


Figure 7: Example of minimal ostis-community members and their corresponding roles.

One of the main problems of ensuring the security of the personal ostis-assistant is the complex of the facts that the user is the main source of threats in the system (regardless of whether he is purposefully creating threats or not) and that the personal ostis-assistant is his

representative, i.e. its main purpose is to be the user's bridge to the Ecosystem. It follows that when solving this problem, the assistant should pay special attention to the class of tasks that it is designed to solve, and limit the user from changing its main functions, while preserving its extensibility. To do this, it is necessary to create a separate fragment of the knowledge base of the personal ostis-assistant and limit it for direct access of the user for edit. Accordingly, the assistant must also have a certain set of agents which will be responsible for the verification of the user data and compute the corresponding degrees of trust. The problems of accounting for non-factors that arise in the knowledge base in the process of interacting with the user are extremely acute here. Given the non-triviality of the solution of the problem of automatic processing of all non-factors, the decision to develop methods of their detection and memorization is an acceptable one at the initial stage. In the future, when accessing this knowledge, the personal ostis-assistant can try to eliminate some of those non-factors through dialog with the user by providing the knowledge from the Ecosystem and its personal knowledge base as a source of reliable knowledge.

It is also worth paying attention that the personal ostis-assistant is responsible for the safety of the user himself. In this case, the main task lying on it is to preserve the confidentiality of the personal data of the user. On the one hand, this problem is solved by the user determining the data that he does not want to provide to third parties, and the assistant, accordingly, must save these preferences and follow them. On the other hand, at a certain stage of the development of the Ecosystem, it may be possible to automate the interaction of a personal ostis-assistant with third party systems that may require user data. In this case it is necessary at the level of communities requiring certain personal data to provide their personal data processing policy for open access. So these policies might be automatically processed by personal ostis-assistants. Depending on the preferences of the user through a special configuration of the assistant, you can set the rules that he can automatically accept or reject, and in extreme cases show the relevant parts of the policy to the user and explicitly request his decision.

### *D. Agents' code security*

Since the main goal of existence of every system is solving certain tasks, we can say that the most large-scale source of threats are executable programs and their source code. In the multi-agent systems these are agents. The transmission of agents between ostis-systems for their storage and execution is one of the provided methods for the collective solution of problems. In turn, this requires providing ways to confirm the security of the received agent. The task of providing developers with tools that perform an automated verification of the code for errors and check the security of the developed agent is also no less important.

When receiving an executable file in traditional systems, accomplishment of security checks is not a trivial task due to very low level of machine instructions and manipulated data, i.e. the security degree of the executable code strictly depends on its interpretation. Currently this problem is solved via usage of the large-scale databases containing already known viruses and their signatures in the executable code. SCP being the native ostis-systems programming language uses relatively high-level code for machine instructions and manipulated data. So this allows to analyze the executable code developed for ostis-platforms with the same or even lower complexity as the source code written for traditional systems. Comparison of a "'Hello World" program written in SCP and machine code is shown on Fig. 8.

Note that due to the possibility of complete scanning of actions performed by the agent in the process of its work, one can completely analyze the tasks that it is designed to solve even without its execution. To evaluate the assurance level for agents, it is proposed to use an algorithm based on checking the number of attempts to perform prohibited actions for the user who initiated the execution of the agent. Thus, by using the mechanisms of an abstract interpretation [20], it is possible to implement not only a system that determines an agents' safety at the level of attempts to execute prohibited actions, but also analyzes possible bugs in the code during the development. Accordingly, each agent in the system will be assigned the appropriate assurance level. For example, if some agent gets in to the system from the outside, then the lowest assurance level must be set for it by default. Obviously, in the process of solving of specific problems some intermediate data might be produced and saved in the knowledge base so it is crucial to set up the appropriate access rights for the results of agents' work as well. The mentioned abstract interpretation algorithm in joint with more complex one as given in the [21].

With the growth of the security measures the attacks' complexity will grow as well. One can suppose that they will have more delayed nature and try to lurk behind actions that are harmless on their own. In order to detect those, the mechanisms based on analysis of events and states of the system might be used as given in the [22] in combination with semantic logging of events in the system [23].

## V. Conclusion

The article examined the main threats and vulnerabilities relevant for individual ostis-systems and OSTIS Ecosystem in general. The differentiation of access to the knowledge bases of ostis-systems, the implementation of mechanisms of configuration of a personal ostis-assistant and the safety of the agents' source code were defined as the main directions for ensuring security.

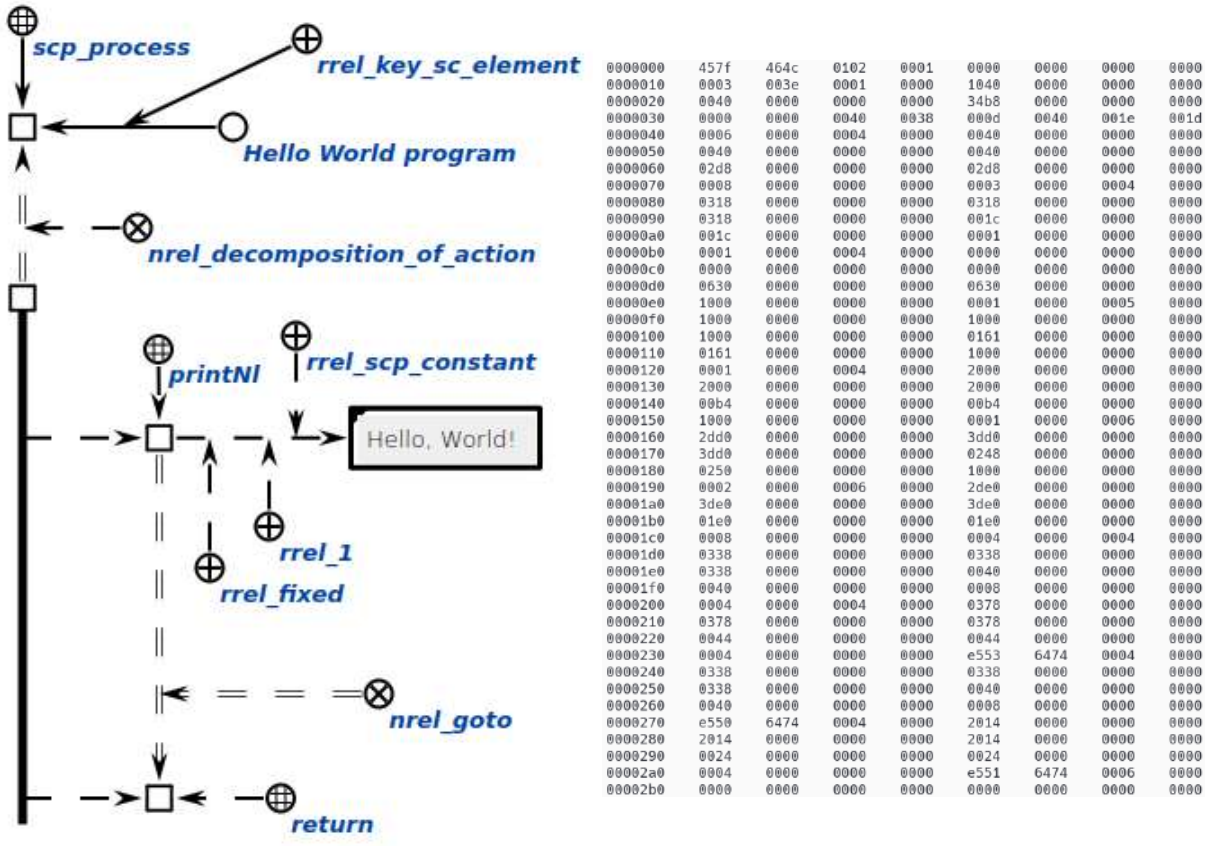


Figure 8: Example of a “Hello World” program written in SCP (left) and machine code (right).

An option was proposed to implement a mechanism for differentiation of access to the knowledge bases of ostis-systems based on the ABAC model. The work examined an example of the architecture of the OSTIS Ecosystem based on the Matrix protocol, as well as ideas for the implementation of safety measures of a personal ostis-assistant and for the agents’ source code.

## References


[1] S. Isoboev, D. Vezarko, and A. Chechelnitsky, “Intellektual’naya sistema monitoringa bezopasnosti seti besprovodnoi svyazi na osnove mashinnogo obucheniya [wireless communication network security intelligent monitoring system based on machine learning],” *Ekonomika i kachestvo sistem svyazi [Economics and quality of communication systems]*, vol. 1, no. 23, pp. 44–48, 2022.

[2] V. Chastikova and A. Mitiugov, “Metodika postroeniya sistemy analiza intsidentov informatsionnoi bezopasnosti na osnove neiroimmunnogo podkhoda [methodology for building a system for analyzing information security incidents based on the neuroimmune approach],” *Elektronnyi Setevoi Politematicheskii Zhurnal «Nauchnye Trudy Kubgtu» [Electronic Network Polythematic Journal «Scientific Works of the KubSTU»]*, no. 1, pp. 98–105, 2022.

[3] A. Skrypnikov *et al.*, “Reshenie zadach informatsionnoi bezopasnosti s ispol’zovaniem iskusstvennogo intellekta [solving information security problems using artificial intelligence],” *Sovremennye naukoemkie tekhnologii [Modern high technologies]*, no. 6, pp. 277–281, 2021.

[4] E. Sozinova, “Primenenie ekspertnykh sistem dlya analiza i otsenki informatsionnoi bezopasnosti [application of expert systems for analyzing and assessing the information security],” *Molodoi uchenyi [Young scientist]*, vol. 10, no. 33, pp. 64–66, 2011.

[5] *Information security, cybersecurity and privacy protection — Evaluation criteria for IT security — Part 1: Introduction and general model*, ISO/IEC Std. 15 408-1, 2022.

[6] V. Golenkov, Ed., *Tehnologija kompleksnoj podderzhki zhiznennogo cikla semanticheski sovmestimyh intellektual’nyh komp’juternyh sistem novogo pokolenija [Technology of complex life cycle support of semantically compatible intelligent computer systems of new generation]*. Bestprint, 2023.

[7] D. Abdurakhman, “Iskusstvennyi intellekt i mashinnoe obuchenie v kiberbezopasnosti [artificial intelligence and machine learning in cybersecurity],” *Sovremennye problemy lingvistiki i metodiki prepodavaniya russkogo yazyka v vuze i shkole [Modern problems of linguistics and methodology of teaching Russian language at university and school]*, no. 34, pp. 916–921, 2022.

[8] Cwe view: Research concepts. Available at: https://cwe.mitre.org/data/definitions/1000.html (accessed 2024, Feb).

[9] V. Ivashenko, *Modeli resheniya zadach v intellektual’nykh sistemakh. V 2 ch. Ch. 1 : Formal’nye modeli obrabotki informatsii i parallel’nye modeli resheniya zadach : ucheb.-metod. posobie [Models for solving problems in intelligent systems. In 2 parts, Part 1: Formal models of information processing and parallel models for solving problems: a tutorial]*. BGUIR, 2020.

[10] V. Druzhinin and D. Ushakov, Eds., *Kognitivnaya psikhologiya. Uchebnik dlya vuzov [Cognitive psychology. Textbook for universities]*. PER SE, 2002.

[11] R. Dushkin, *Metody polucheniya, predstavleniya i obrabotki znanii s NE-faktorami [Methods of gaining, representation and processing of knowledge with NON-factors]*, 2011.

[12] A. Gribkov, “Formirovanie dostovernogo znaniya: nakhozhdenie znanii i vyyavlenie defektov [the formation of reliable knowledge: Finding knowledge and identifying defects],” *Vestnik Leningradskogo gosudarstvennogo universiteta imeni A. S. Pushkina [A.S. Pushkin Leningrad State University Journal]*, pp. 74–90, 2023.
[13] A. Dementev, “Metriki semanticheskikh dannykh [metrics for semantic data],” *Molodoi uchenyi [Young scientist]*, vol. 24, no. 419, pp. 48–51, 2022.
[14] A. Narinyani, “Ne-faktory: Netochnost’ i nedoopredelennost’ - razlichie i vzaimosvyaz’ (doformal’noe issledovanie) [non-factors: Inaccuracy and under-determination - distinction and relationship (pre-formal study)],” *Mezhdunarodnyi Seminar DIALOG’99 Tarusa [International Seminar DIALOG’99 Tarusa]*, 1999.
[15] I. Davydenko, “Semantic models, method and tools of knowledge bases coordinated development based on reusable components,” *Otkrytye semanticheskie tehnologii proektirovanija intellektual’nyh sistem [Open semantic technologies for intelligent systems]*, pp. 99–118, 2018.
[16] A. Ostroukh, *Intellektual’nye sistemy: monografiya [Intelligent systems: monograph]*. Nauchno-innovatsionnyi tsentr, 2020.
[17] A. Baranovich, “Semanticheskie aspekty informatsionnoi bezopasnosti: kontsentratsiya znanii [semantic aspects of information security: concentration of knowledge],” *Istoriya i arkhivy [History and Archives]*, vol. 13, no. 75, pp. 38–58, 2011.
[18] Matrix specification. Available at: https://spec.matrix.org/v1.9/ (accessed 2024, Mar).
[19] Openpgp. Available at: https://www.openpgp.org/ (accessed 2024, Mar).
[20] P. Cousot and R. Cousot, “Abstract interpretation: Past, present and future,” *CSL-LICS ’14: Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pp. 1–10, 2014.
[21] V. Khoang and A. Tuzovskii, “Resheniya osnovnykh zadach v razrabotke programmy podderzhki bezopasnosti raboty s semanticheskimi bazami dannykh [solutions to the main problems in the development of a program to support the security of work with semantic databases],” *Doklady Tomskogo gosudarstvennogo universiteta sistem upravleniya i radioelektroniki [Proceedings of Tomsk State University of Control Systems and Radioelectronics]*, vol. 2, no. 28, pp. 121–125, 2013.
[22] A. Lajevardi and M. Amini, “Big knowledge-based semantic correlation for detecting slow and low-level advanced persistent threats,” *Journal of Big Data*, vol. 8, no. 148, 2021.
[23] V. Ivashenko, N. Zotov, and M. Orlov, “Semantic logging of repeating events in a forward branching time model,” *Pattern Recognition and Information Processing (PRIP’2021): Proceedings of the 15th International Conference*, p. 149–152, 2021.

# ПРОБЛЕМЫ БЕЗОПАСНОСТИ ЭКОСИСТЕМЫ OSTIS

Хорошавин В. Д., Захаров В. В.

В данной работе рассматриваются угрозы и уязвимости, актуальные для ostis-систем. Разграничение доступа к базам знаний остис-систем, реализация механизмов настройки персонального остис-ассистента и безопасность исходного кода агентов определены как основные направлениями обеспечения безопасности остис-систем. Предложены варианты реализации соответствующих механизмов безопасности по этим направлениям.

Received 13.03.2024