Optimization of centralization of network OS security resources based on dynamic transfer of control between network nodes

Anatoliy Sachenko[1,2,†], Tomas Sochor[3,†], Bohdan Savenko[4,†], Yuriy Stetsyuk[4,*,†] and Vasil Stetsyuk[4,†]

[1] *West Ukrainian National University, Lvivska Street 11 46009 Ternopil, Ukraine*
[2] *Casimir Pulaski Radom University, Malczewskiego St. 29 26-600 Radom, Poland*
[3] *Prigo University College European Research University Vítězslava Nezvala 801/1 Havířov 736 01 Czech Republic European Union*
[4] *Khmelnitsky National University, Instytutska street 11 29016 Khmelnitsky, Ukraine*

Abstract

The current state of development of OS security subsystems is analyzed. Attention is paid to the principles of building centralized security systems for network OSes based on dynamic transfer of control between network nodes. This provides increased resistance to leaks of confidential information as a result of the destructive effects of malicious software and computer attacks. A description of the process of dynamic transfer of control between computer network nodes is presented, and mechanisms for forming centralized security resource bases are considered. The issue of optimizing security resources that are subject to centralized control by the current network control node when using dynamic transfer of control between computer network nodes is also considered. Strategies for forming global privilege bases, security policies, and network connections during each cycle of control transfer from the current computer network node to the next are presented. Several series of experiments were conducted with a network of virtual machines running the FreeBSD 13.1 operating system, the results of which confirmed theoretical calculations and mathematical modeling. A comparative analysis of the effectiveness of full centralization of security resources and their partial centralization with dynamic control transfer between computer network nodes was performed. The advantages of the proposed approach were manifested in a reduction in the time of control transfer between network nodes by 37 percent, a reduction in the attack surface due to the minimization of points of influence of malicious software on the security system.

Keywords

operating system, computer networks, centralized security systems, security strategies, centralization of security resources

1. Introduction

The sphere of information technologies has become a defining path of development of mankind, transforming not only the ways of its communication, but also the foundations of the world economy in all its directions. Our world and its vital processes have become critically dependent on the stable functioning of complex electronic systems and various software that processes, transmits and protects information. It is already clear that this direction of development is not a temporary phenomenon, but an objective stage of progress that has already determined the trajectory of the future, which has no turning back. We, as a society, have finally entered the information era, in which the dominant value is not material resources, but information.

Technological achievements of mankind are faced with new large-scale challenges, threats to information security, which today have become a constant background of digital life. Now success in any field, be it science, economics, or security is determined by the level of access to reliable information, the speed of its processing and the ability to protect it from external interference.

AISSLE-2025: The International Workshop on Applied Intelligent Security Systems in Law Enforcement, October, 30–31, 2025, Vinnytsia, Ukraine.
[1,*] Corresponding author.
[†] These authors contributed equally.
as@wunu.edu.ua (A. Sachenko); tomas.sochor@osu.cz (T. Sochor); savenko_bohdan@ukr.net (B. Savenko); yuriy.stetsuk@khmnu.edu.ua (Y. Stetsyuk); swmua@khmnu.edu.ua (V. Stetsyuk)
0000-0002-0907-3682 (A. Sachenko); 0000-0002-1704-1883 (T. Sochor); 0000-0001-5647-9979 (B. Savenko); 0000-0003-0312-2276 (Y. Stetsyuk); 0000-0001-9880-2666 (V. Stetsyuk)

The relevance of the task of ensuring information protection and increasing the resistance of information systems to leaks of confidential information, despite the large-scale efforts of the planetary economy, does not decrease. The analysis of the state of information systems protection shows that this task remains extremely relevant today. Despite the fact that this problem is in the constant focus of attention of the scientific community and the large number of methods and means of protecting information systems proposed by it, it cannot be considered solved.

This article is aimed at solving the problems of OS resistance to leaks of confidential information processed in computer systems. Its purpose is to develop models of the process of random dynamic transfer of control between network nodes, their study in the direction of optimizing the security resources of the network OS, which are subject to centralized management, which in turn will allow reducing the OS's time spent on transferring control between nodes and, thereby, ensuring the continuity and stability of network management, which ultimately will lead to increasing the OS's resistance to information leaks.

2. Analysis of known solutions

OSs have gone through several technological stages in their development. Together with them, their security systems have also overcome this path, until they have really reached a level that can guarantee the protection of information processed in computer systems under their control. And all this in conditions of constant, ever-growing influence of malicious software, which in turn is constantly being improved. Developing under its continuous pressure, OS security systems today represent powerful control subsystems that include a large range of highly effective protective mechanisms. The architecture of OS security systems, especially network ones, has gone from decentralized to complexly organized centralized, covering all the resources of a modern computer system with its control. Thus, in [1] a model is presented that describes the interaction of protective mechanisms within the framework of a centralized security system for the OS of a network node, which allows building a security architecture of a network OS that minimizes the problem of leakage of confidential information during attacks on the system's RAM. The search for a balanced architecture of the OS security system that effectively ensures the stability of the OS is considered in [2].

Many works are devoted to methods for improving the operation of network incident response systems (IDS\IPS) and approaches that can be applied to increase the level of security in networks in terms of counteracting the leakage of confidential information [3, 4]. In [5], a new IDS system based on a combination of a multilayer perceptron network (MLP) is proposed. For software-defined networks (SDN), a mechanism based on the entropy values of the source and destination IP addresses of flows observed by the network controller is proposed [6]. In [7], an innovative approach is presented that provides proactive protection against destructive DDoS attacks. It is based on the use of an echo state network (ESN), specially designed for SDN. To detect DDoS traffic, in [8] a model is proposed that calculates its threshold value for applications using the network. In real time, using a machine learning (ML) model, it is determined whether this traffic is DDoS traffic. A new FLBC-IDS machine learning technique that combines horizontal federated learning (HFL), Hyperledger blockchain, and EfficientNet for intrusion detection is proposed in [9]. In [10], means for improving the efficiency of IDS systems that analyze and cluster network traffic are proposed. A new metric for evaluating IDS systems that takes into account the delay in detecting cyberattacks is presented in [11].

Analysis of the performance of IDS based on machine learning has shown its dependence on the implementation of functions, and the spatial and temporal correlation of network data attributes complicates the manual design of functions. The proposed IDS [12] uses an optimized one-dimensional convolutional neural network block and sufficient memory to automatically extract spatial and temporal features from the input data. In addition, a knowledge transfer method is used to transfer features, which allows detecting zero-day attacks.

Also, much attention is paid to strategies and mechanisms for protecting network OSs in terms of countering malicious software. A self-adaptive system based on SVM to ensure network resilience to botnet cyberattacks called BotGRABBER is considered in [13, 14]. Another self-adaptive system in which resilience is ensured by adaptive network reconfiguration is presented in [15]. In [16], a new method for

detecting DDoS botnets is proposed based on the analysis of their network characteristics, and in [17] on the use of artificial immune system algorithms. A new step is the proposed distributed DDoS protection scheme based on shared agents [18], which detects and prevents DDoS attacks within Internet service providers (ISPs). The distributed security systems are reinforced by the genetic algorithm proposed in [19, 20], based on the selection and variations of search parameters. Another approach to ensuring network security is data storage based on the architecture of a semi-network OS [21]. Integrated management systems based on threat and risk models are proposed [22]. The idea of creating a secure OS with controlled complexity is promising [23]. In [24], another research direction was initiated, based on serverless security, which has a high potential for reliable protective measures.

An interesting solution is to supplement the OS security system with a memory isolation mechanism that does not allow bypassing it by virtualizing the memory management unit [25]. Also, works [26, 27] are devoted to countering memory attacks (MCA), which change the contents of some memory areas in order to disrupt the normal operation of computing systems, causing a leak of confidential data or disruptions in current processes. In [28], attacks on modern heterogeneous embedded computing platforms FPGA-SoC, which contain the most advanced memory and peripheral isolation mechanisms, are investigated. In [29], a security model is proposed that provides a higher level of protection compared to such existing approaches as recurrent neural networks and the support vector method. It is promising for an active security control strategy for cyber-physical systems.

OS security on various hardware platforms is a fundamental goal of current evaluation. Identifying and evaluating OS security factors are essential components in OS design. In [30], a formal verification method for the RIOT OS crypto module is proposed for software analysis of Frama-C code, in order to ensure its security aspects. In [31, 32, 33], the causes of vulnerabilities are investigated, in particular for IoT, IIoT, SCADA and Android systems for embedded systems, and the implementation of effective malware detection strategies is proposed. The work [34] focuses on the problem of Kernel Address Space Allocation Randomization (KASLR) violation on modern mobile devices without using cache memory, which is related to the KASLR violation problem on ARM processors. In the search for OS architectures resistant to information leaks, in [35] a tool is proposed for research and modeling of a prototype system with a microkernel architecture that provides high reliability and scalability.

An interesting authentication system is proposed in [36]. It is based on a protocol that includes the Linux security module for user-based NAC. It requires neither user accounts nor a secure user space; it loads signed rules and keys for the user from a USB security key, securely authenticates the user, and controls network permissions directly from the Linux kernel.

In [37], the issue of minimizing the provision of access privileges to resources is raised as a way to increase resilience to information leaks. In particular, attackers can use vulnerabilities in file systems and disk drivers to leak or manipulate the contents of program files. The key problem is that the cancellation of the transfer of OS services privileges from the kernel level to the user level does not solve the problem. It is proposed to use a special Mirage mechanism that deprives them of the right to access the contents of files, while preserving the functions of manipulating them. In [38], the problem of the Windows Embedded OS, which uses a security policy based on discretionary access control (DAC), which is vulnerable to external hacker attacks, is raised. It is proposed to improve security by prohibiting the execution of files from outside the white list.

It is noted that traditional OS security systems based on static protection methods do not always cope with increasingly complex attacks [39], where attackers gain an advantage by using even one vulnerability to compromise the system, adapting to traditional protection measures. In order to overcome the problem, the implementation of proactive and adaptive security systems is proposed [40, 41]. Another direction for ensuring information protection in many computer systems is the change of the control center. A method for determining the next centralization option without user intervention is proposed in [42].

The integration of artificial intelligence into computer networks is one of the advanced technologies. It is assumed that the concept of using artificial intelligence will solve the problems faced by computer networks, especially with regard to the leveling of information leakage threats [43]. The fundamental idea of using artificial intelligence is to move to a proactive OS security model that includes intelligent threat detection, in-depth behavior analysis, and dynamic countermeasures against attackers [44], where

traditional methods, especially rule-based approaches, have encountered significant difficulties in protecting confidential data from constantly changing threats, especially in conditions of increasing data volumes.

At the same time, it is noted [45] that artificial intelligence and machine learning methods are increasingly used in cyberattacks. This means that, as in many other areas of our lives, there will be no absolute victory, the struggle will continue and, as is customary, the more competent and experienced will tend to win.

The task of countering the leakage of confidential information in network OSes remains incompletely solved. This includes incomplete centralization of access control to resources, countering multi-vector threats, and the lack of a universal approach to coordinating policies from different formats and nodes, which leads to gaps in protection.

3. Formulation of the problem

The implementation of dynamic transfer of control between computer network nodes provides a number of technical and organizational advantages, namely, improved flexibility of network management, increased resistance to failures and counteraction to malicious software. However, obtaining maximum efficiency of this method is closely related to many factors, the most important of which is the optimal structuring and transfer of critical security resources to the new control node. The formation of an optimal set of global security resources that must be transferred to the new control node at the time of a change in the control center is a key aspect of the successful implementation of dynamic control of a computer network. Solving this problem will avoid unnecessary duplication, reduce the volume of data to be transferred and, accordingly, reduce the time for their synchronization.

This set should include only those security resources of the network OS that directly affect the integrity, availability and manageability of the computer network. As a rule, its minimum content includes a global privilege database, security policies together with access policies to confidential data, a connection database for managing network traffic. All other security resources that are mainly of local importance (log files, local device drivers, etc.) should be left under the control of the corresponding network nodes, which will ensure a reduction in the costs of transferring control. The formation of an excessively large set of global resources will lead to an increase in the transfer time and, as a result, the risk of delays in control, the emergence of data synchronization problems between nodes. Therefore, optimizing the volume of security resources subject to centralization is a critically necessary condition for building an effective partially centralized OS security management model. This work is devoted to an attempt to scientifically solve this problem.

Thus, the main task of this study is to solve the problem of ensuring maximum efficiency of dynamic transfer of control between network nodes by optimizing security resources subject to centralized control. This, in turn, should reduce the time spent on transferring control.

4. The main part

Transfer of control from one network node to another, randomly selected, is a rather complex and resource-intensive process. One of the issues to be solved in this case is determining the volume of transfer of security resources, or the degree of their centralization. The main resource is the global privilege base. But to ensure stable and secure functioning of a computer network with dynamic random transfer of control between its nodes, continuity of network control, it is necessary to transfer a number of critical system data to the new control node, which ensure the integrity of security policies, fault tolerance during the transfer of control.

In addition to the global privilege base, this includes security policies, including sensitive information labeling policies, network policies (Firewall operation, ACL), shared resource access policies, procedures for transferring control between nodes, synchronization with privacy databases (e.g., DLP), local event logs (audit.log), hardware configuration parameters (USB, ports, local drivers), local session caches (e.g., memory pages without sensitive data), background security services (local antivirus scanner, self-

monitoring scripts), lists of authorized local processes, low-level drivers and the OS kernel of the network node, and others.

To reduce the overhead of random dynamic transfer of control between computer network nodes without compromising security and ensuring continuity of control, it is proposed to leave part of the security resources that are not critical for global decision-making; do not affect the consistency of the entire security system; have a low probability of conflict or attack during local control, under the control of local nodes, and centralize those of them that affect access, interaction between nodes, security policies and transfer of control.

The process of dynamic transfer of control between network nodes. To describe the process of dynamic transfer of control between computer network nodes with the separation of centralized and local resource management, notations for key aspects are required. The set of network nodes is denoted as:

$N = \{N_1, N_2, \ldots, N_n\}$, where $N_1, N_2, \ldots, N_n$ are the computer network nodes.

The set of node security resources is defined as R_{BR}:

$R_{BR} = \{R_1,\ R_2,\ \ldots,\ R_k\ \}$, where $R_1,\ R_2,\ \ldots,\ R_k$ are the security resources (privilege databases, security policies, connection database, etc.).

We define resources subject to centralized management as its subset $R_{centr} \in R_{BR}$. Then the subset of resources that can remain under local control can be defined as R_{local}, as one that is not intersecting with a subset R_{centr} . Let's define this as $R_{local} = R_{BR} / R_{centr}$, while their union forms a set R_{BR}. To describe the process of dynamic control transfer between computer network nodes, we will also introduce the necessary notation. The current control node at time t is defined as $CSMM_t$. At the same time $CSMM_t$ belongs to a set of network nodes N: $CSMM_t\ \epsilon\ N$. Then the probability of transferring control from the current network node $CSMM_t$ to the next $CSMM_{t+1}$ that will take control of the network at time $t+1$ can be defined as:

$$P\left(CSMM_{t+1} = N_j \middle| CSMM_t = N_i\right) \quad (1)$$

where $CSMM_t$ and $CSMM_{t+1}$ are the control nodes of the computer network at the points in time and, respectively, from the set of network nodes N; N_i is the current control node at time t; N_j is the index of the next control node from the set N at time $t+1$, with the control transfer subject to constraint $j \neq i$ (Figure 1, position 1).

Based on the above, we can conclude that there is a uniform distribution of the probability of becoming the network control node in the process of dynamic control transfer for each active network node, i.e. each node has the same chance of becoming the central network control node. This can be represented by the formula:

$$P\left(CSMM_{t+1} = N_j\right) = \frac{1}{n-1},\ \forall\ j \neq i \quad (2)$$

where N_j is the index of the next control node $CSMM_{t+1}$ from the set N at time $t+1$; n – the total number of active network nodes (currently working) in the set N;. $n-1$ takes into account that the current control node cannot become the network control node again ($j \neq i$).

The chosen law of distribution of probability of a network node becoming the central control node of the network allows to guarantee randomness and uniformity of control transfer, allows to avoid overload of the network node, increases resistance to attacks due to the difficulty for the attacker to predict who will be the next control node, supports the idea of partial centralization - no node remains a permanent center. Thus, it can be described as a Markov process, since the next state of the network is determined only by its current state with a uniform probability of dynamic control transfer between nodes. The transfer of control between network nodes itself actually means that the control of the network is transferred from the central control module $CSMM_t$ of the centralized security service of the OS of the current node to the same control module $CSMM_{t+1}$ of the next active network node. The transfer process

is initiated by the current control node $CSMM_t$ by sending the control token $T_{Governance}$ to a randomly selected active network node (Figure 1, position 1). After receiving the control token (Figure 1, position 2), the new $CSMM_{t+1}$ control node sends a request (Figure 1, position 3) to all active network nodes to transfer security resources that require centralized management: a global privilege base - there must be one consistent copy for all nodes, sensitive data labeling policies, they affect all access logic, network policies is require a single security strategy, a control transfer schedule between nodes, its violation can cause collisions in network management, security incident handling policies, also require centralized response and analysis.

Formation of global databases for centralized network management. In practice, this means that such OS resources as /etc/security/policies.conf, /etc/firewall.rules, global_privileges.db are the minimum possible set of them that require centralized management. Such delimitation of management allows you to reduce traffic, accelerate the transfer of control between nodes, and at the same time maintain high resistance to leaks of confidential information. However, it is necessary to take into account the nuances of the mechanisms for forming centralized copies of these resources for the new $CSMM_{t+1}$ control node. The logic of ensuring the maximum level of resistance to leaks of confidential information dictates the methods of their implementation.

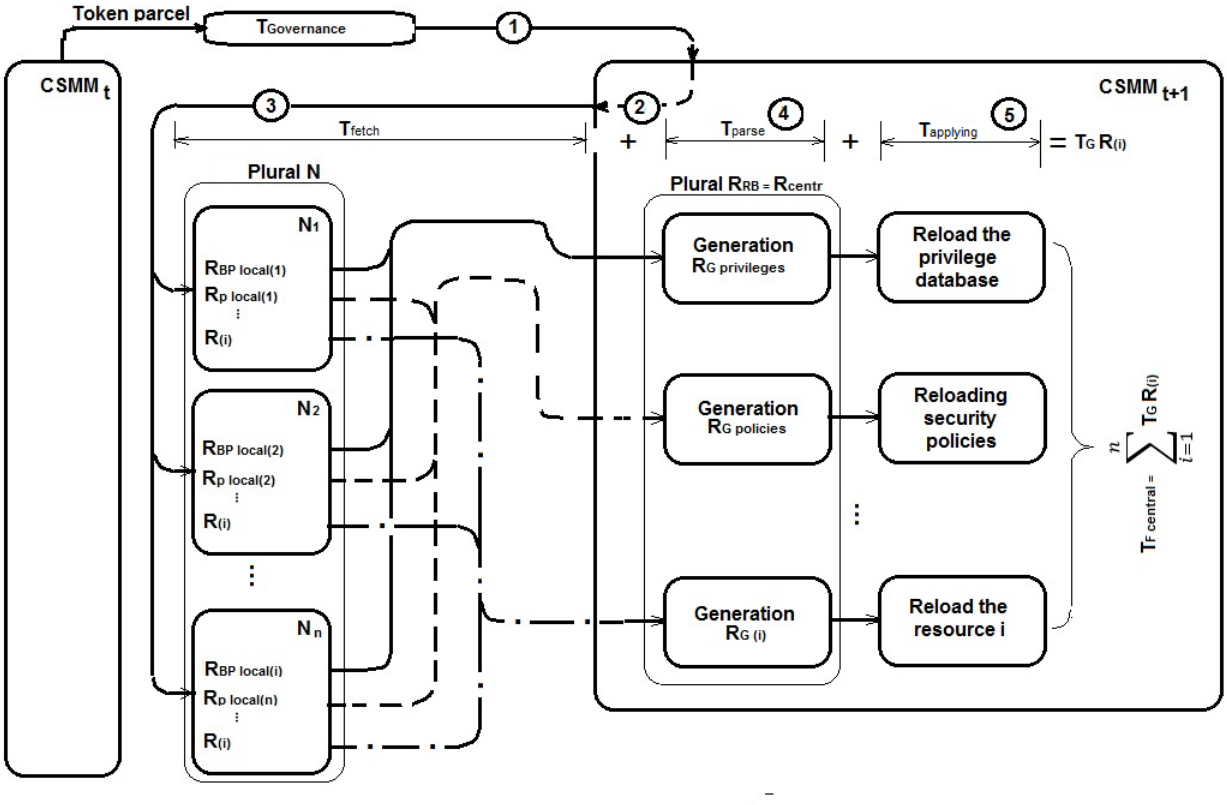

Figure 1: Model of the process of dynamic transfer of control of network nodes with complete centralization of security resources of the network OS security system.

Thus, the current centralized privilege database global_privileges.db for the next control node $CSMM_{t+1}$ is obtained by combining the local privilege databases of active network nodes (Figure 1, position 4). To represent this mechanism, we introduce the following notations: $R_{BP\,local}$ is the privilege database of a local node. Then the global privilege database $R_{G\,privileges}$ can be represented as follows:

$$R_{G\,privileges} = U_{i \in A}\, R_{BP\,local}(i) \qquad (3)$$

where $R_{BP\,local}(i)$ is the local privilege base of the i-th network node, where i belongs to the set A of network nodes active at the time of control transfer, which in turn is a subset of the set of network nodes N.

In the process of implementing management actions by the current $CSMM_t$ module, all changes in the global privilege base $R_{G\ privileges}$ are transactionally replicated on all relevant local bases of network nodes, which allows, in the event of an incident with the current management node, to obtain a new up-to-date global privilege base for the new $CSMM_{t+1}$ management module, which increases the OS's resistance to information leaks.

Let us consider the formation of a global security policy database global_policies.conf which is another centralized resource that is updated at each cycle of control transfer between nodes. For convenience of description, we will introduce the following notations: $R_{P_i(x)}$ – security policy x i-th network node as a security resource; $R_{P\ local}$ – security policy of some local node. It can be defined as the union of all security policies of a given local node of a computer network:

$$R_{P\,local} = U_{i=1}^{n}\{R_{P_i(x)}\} \tag{4}$$

where $R_{P_i(x)}$ is the security policy of x i-th node; n is the number of network nodes.

The next step is to obtain the global security policy base of the computer network. It should be noted that when implementing the function of combining all local security policy bases, it is possible to use several mechanisms that implement different access rules, lists of permitted actions, etc., and from which it is necessary to choose the most optimal one that will ensure maximum system resistance to information leakage. This may be a consensus approach, which is based on including in the global $R_{G\ policies}$ security policy base all security policies present in the local security policy bases of active nodes. Another option for the mechanism for forming a centralized policy base involves its implementation as an intersection of security policies of all local bases of network nodes. It is also possible to use a mechanism that, when forming a global $R_{G\ policies}$ policy base, takes into account the level of trust in nodes - a weighted union (for example, taking into account their reliability, incident history, etc.), giving priority to including security policies from local bases of nodes with a higher level of trust.

Since we are talking about the formation of centralized databases of security resources of the network OS security system, the most expedient and justified seems to be the use of a mechanism based on the intersection of security policies of local databases of network nodes, since it is quite simple to implement and at the same time provides a regime of the strictest rules for applying security policies from the central control node, which in turn guarantees increased OS resistance to leaks of confidential information processed in the system under its control.

Then the global $R_{G\ policies}$ security policy database can be represented as follows:

$$R_{G\,policies} = \cap_{i\in A}\{R_{p\ local}(i)\} \tag{5}$$

where $R_{p\ local}(i)$ is the local security policy base of the i-th node of the computer network, where i belongs to the set of active nodes of the A network at the moment of control transfer.

To form the global connection base $R_{G\ firewall}$, we will use the same mechanism as when forming the global security policy base $R_{G\ policies}$, since connection management also, due to its importance, requires the strictest control rules.

Then $R_{G\ firewall}$ will be defined as follows:

$$R_{G\,firewall} = \cap_{i\in A}\{R_{FW\,local}(i)\} \tag{6}$$

where $R_{FW\ local}(i)$ is the local connection base of the i-th node of the computer network, where i belongs to the set of active nodes of the A network at the moment of control transfer.

Justification of the effectiveness of partial centralization of security resources. To do this, we will perform a comparative calculation of the costs of dynamic transfer of control between computer network nodes for situations with complete centralization of all security resources of the network OS security system and centralization of only privilege databases, security policies, and network connections.

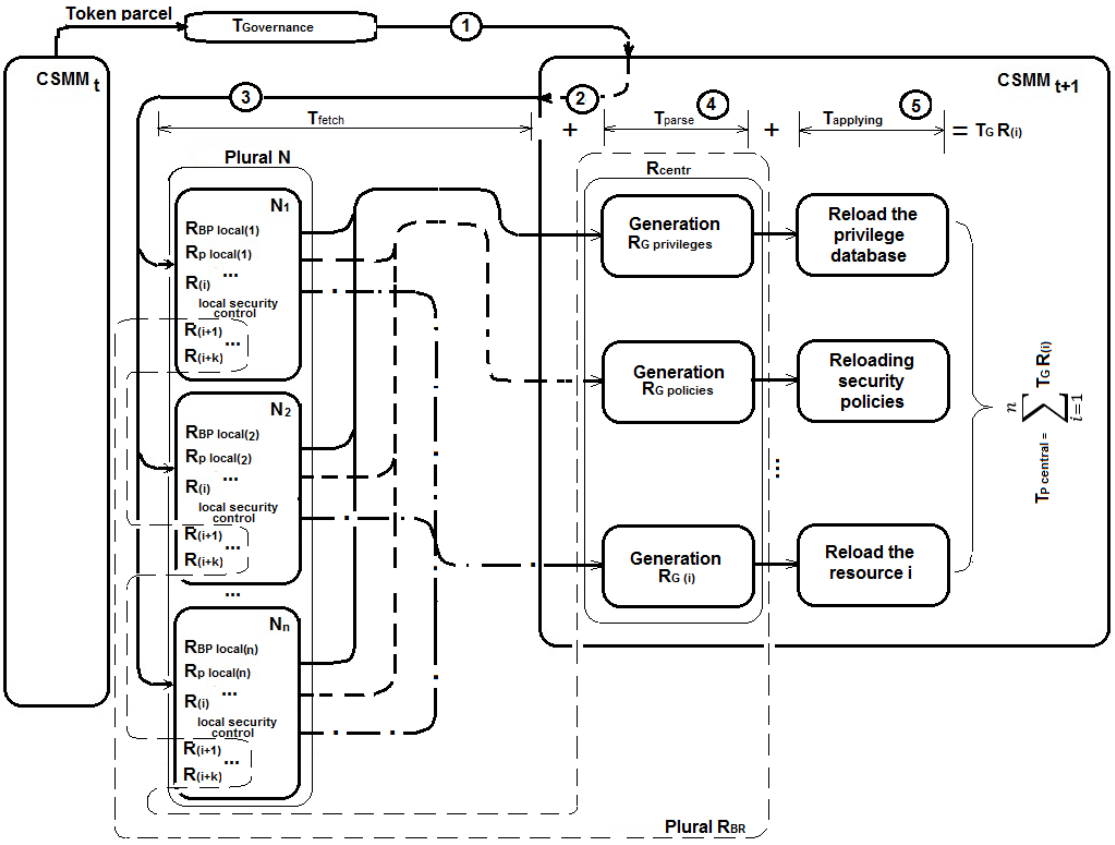

Figure 2: Model of the process of dynamic transfer of control of network nodes with partial centralization of security resources of the network OS security system.

To mathematically represent the total costs of transferring control between network nodes, we will use the previously introduced notations: R_{BR} – set of network security resources; $R_{centr} \in R_{BR}$ – subset of resources subject to centralized control for a partial centralization situation, and we will also introduce the notation of comparative criteria: T_{fetch} – time to poll all active network nodes and receive copies of the resource R_i from them to form its current global database; T_{parse} – time that the new control node spends on parsing, unifying or confirming the compliance of the received data (checking the integrity of the database, checking the consistency of security policies, building a generalized global version of the resource); $T_{aplying}$ – time to restart a new global version of the resource on the new control node (activating it in services or OS subsystems, transmitting confirmation to other network nodes if necessary).

Then the total update time of the global version of some R_i security resource of the network OS can be represented as the sum of all time periods of its formation:

$$T_G(R_i) = T_{fetch}(R_i) + T_{parse}(R_i) + T_{applying}(R_i) \quad (7)$$

where T_{fetch} is the time to request local copies of the R_i resource from all active network nodes; T_{parse} is the time to form a global version of the R_i resource; $T_{applying}$ is the time to restart a new version of the global R_i resource.

The implementation of formula (7) is illustrated in (Figure 1, position 3, 4, 5) and (Figure 2, position 3, 4, 5), which present the processes of preparing global security resource bases for the next $CSMM_{t+1}$ control node of the OS security system with full (Figure 1) and partial (Figure 2) resource centralization.

Now, for a situation with complete centralization of all security resources of the network OS, the time of transfer of control to the new control node $CSMM_{t+1}$, $T_{F\ central}$ can be represented as the sum of all time periods required to form all global versions of all security resources of the OS according to the scheme (Figure 1). Taking into account the above and (formula 7), it will take the form:

$$T_{F\ central} = \sum_{i=1}^{n} \left(T_{fetch}\left(R_i\right) + T_{parse}\left(R_i\right) + T_{applying}\left(R_i\right) \right) \quad (8)$$

where T_{fetch} is the time to request local copies of the R_i resource from all active network nodes; T_{parse} is the time to form a global version of the R_i resource; $T_{applying}$ is the time to restart a new version of the global R_i resource; n is the total number of OS security resources in the set R_{BP}.

The time of transfer of control to the new control node $CSMM_{t+1}$ with partial centralization of security resources $T_{P\ central}$ can be calculated using formula (8) with the only difference that the total number of OS security resources n will correspond to the number of elements of the set R_{centr}, which is a subset of the set R_{BP}. This feature of the construction of the model of dynamic control transfer is shown in (Figure 2). With such a scheme of operation, part of the security resources remain in the local control of the network node, respectively, they do not require the collection and centralization of data about them when transferring control to the new central node $CSMM_{t+1}$, which significantly reduces the time spent on the transfer of control itself. To minimize them, it is important that the number of elements of the set R_{centr} be as small as possible, but on the other hand such that the required level of OS resistance to leakage of confidential information is ensured.

Then the ratio of the time of transfer of control to a new control node with full centralization of security resources $T_{F\ central}$ and the time of transfer of control with partial centralization of security resources $T_{P\ central}$ greater than one will indicate a better efficiency of the OS security system with partial centralization:

$$E_{central} = \frac{T_{F\ central}}{T_{P\ central}} > 1 \quad (9)$$

To represent the value of $E_{P\ central}$ efficiency in percent, we present formula (9) in the following form:

$$E_{P\ central} = \frac{T_{F\ central} - T_{P\ central}}{T_{F\ central}} \times 100\,\% \quad (10)$$

where $T_{F\ central}$ is the time of transfer of control to a new control node with full centralization of security resources; $T_{P\ central}$ is the time of transfer of control with partial centralization of security resources.

A method for assessing the effectiveness of partial centralization of security resources during dynamic transfer of control between network nodes in comparison with their full centralization has been developed for use in the study of network OS security systems.

5. Experiments

Experimental environment setup. In order to confirm the effectiveness of the network OS security system with partial centralization of security resources during dynamic transfer of control between network nodes, several series of experiments were conducted. For this purpose, a test environment was deployed based on the use of a virtual computer network that includes virtual nodes VM01 - VM04, which served as target machines during the experiments (Figure 3). All virtual machines operate under the control of the Virtual Box hypervisor. With its help, virtual machines are combined into a separate local virtual network necessary for conducting experiments. FreeBSD 13.1 is installed as a network OS on each virtual machine.

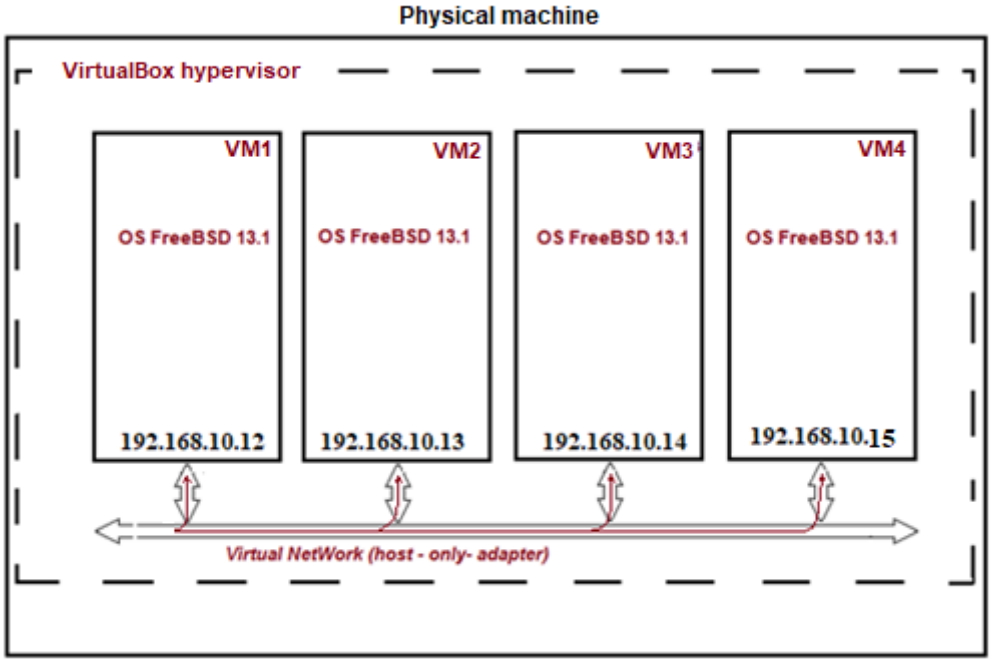

Figure 3: Scheme of an experimental setup for determining the effectiveness of a system with partial centralization of network OS security resources.

Calculation of the efficiency of partial centralization of resources. As a result of experiments with a virtual network with simulation of dynamic control transfer between its nodes, the data given in Table 1 were obtained.

Table 1.
Averaged experimental data.

Security resource name	$T_{applying}$, sec	$T_{applying}$, sec	$T_{applying}$, sec	$T_G(R_i)$, sec	Resource centralization
Privilege base	1.03	2.27	2.01	5.31	Yes
Security policies	5.22	1.59	2.73	9.54	Yes
Network connections	3.44	1.28	3.49	8.21	Yes
Audit Logs	3.432	1.144	1.562	6.14	No
Access rights	0.6943	0.88	1.2314	2.81	No
Antivirus settings	2.5545	1.2183	1.3755	5.15	No

Substituting the data into formula (8) to calculate the control transfer time between computer network nodes for a situation with complete centralization of security resources (taking into account $R_i \in R_{BP}$), we obtain:

$$T_{F\,central} = \sum_{i=1}^{n} \left(T_{fetch}(R_i) + T_{parse}(R_i) + T_{applying}(R_i) \right) = (5.31 + 9.54 + 8.21 + 6.14 + 2.81 + 5.15) = 37.16\,sec$$

As part of the next step, we calculate the time for control transfer between network nodes for a situation with partial centralization of resources (formula (8) taking into account $R_i \in R_{central}$) and obtain:

$$T_{P\,central}=\sum_{i=1}^{n}\left(T_{fetch}\left(R_i\right)+T_{parse}\left(R_i\right)+T_{applying}\left(R_i\right)\right)=\left(5.31+9.54+8.21\right)=23.06\ \ sec$$

Using formula (10), we calculate the efficiency of partial centralization of security resources during dynamic transfer of control between network nodes compared to complete centralization of security resources:

$$E_{P\ central}=\frac{T_{F\ central}-T_{P\ central}}{T_{F\ central}}\times 100\,\%=\frac{37.16-23.06}{37.16}\times 100\,\%=37.9\,\%$$

Calculations based on the results of the experiments confirmed the effectiveness of the method for optimizing the centralization of network OS security resources during random dynamic transfer of control between computer network nodes (Figure 4).

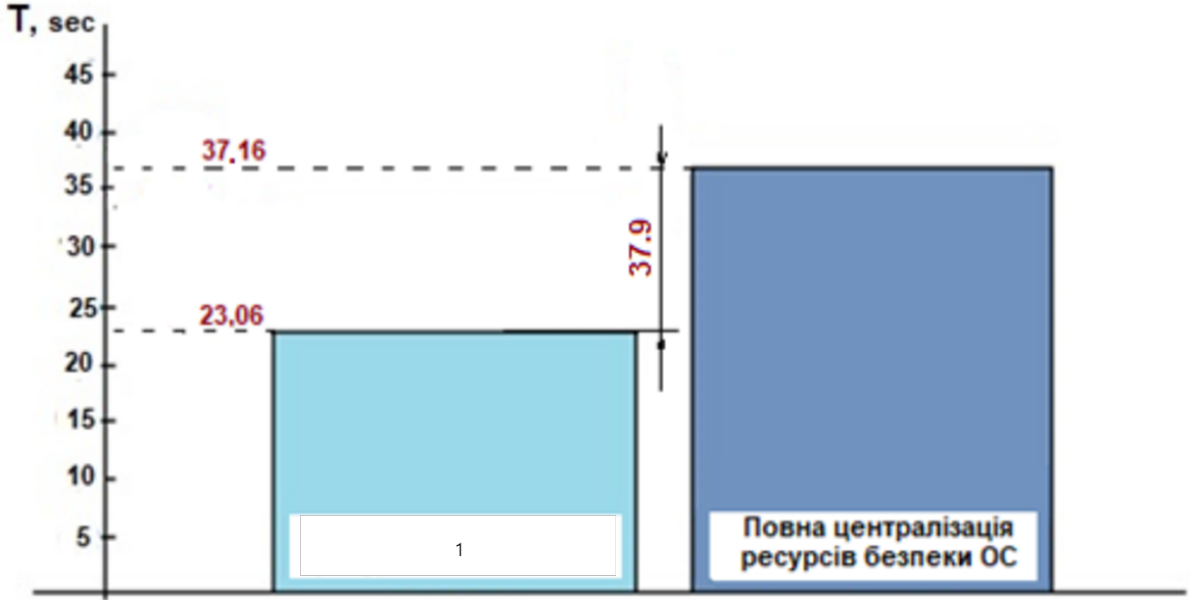

Figure 4: Graph of the effectiveness of partial centralization of security resources with dynamic control transfer between network nodes: 1 is Partial centralization of OS security resources; 2 is Complete centralization of OS security resources.

The reduction in the time for transferring control between computer network nodes with partial centralization of security resources was, as can be seen from the calculations, ≈ 38%, which was achieved by excluding secondary security resources (log files, local device drivers, etc.) from centralized replication, reducing the amount of data for synchronization, and reducing the overhead of data verification, normalization, and logging.

6. Conclusions

The result of the proposed approach is the confirmation of a significant reduction in time (Figure 4) during dynamic random transfer of control between network nodes, which, in turn, reduces the probability of losing control of the network or reducing its security during the transition period. In addition, this approach allows for better scaling of the system, since an increase in the number of nodes does not proportionally affect the amount of data transferred to the new control node, due to the optimization of the set of critical security resources.

Another advantage of optimizing the set of global security resources is the reduction of the attack surface, since the number of points through which an attacker could affect the security system during the transfer of control is reduced. This is especially important in conditions where attacks can be coordinated and aimed at the moments of maximum vulnerability of the system, namely at the moments of changing the control node.

Thus, the formation of an optimal set of global OS security resources for partially centralized network management based on dynamic control transfer between its nodes is not only an engineering task, but also a scientific tool for increasing the efficiency, reliability and security of the entire computer network. This approach to building centralized OS security systems allows maintaining a balance between the flexibility of dynamic management and the necessary rigidity of control over its critical aspects.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] Y. Stetsyuk, M. Stetsyuk, V. Kyrylo, V. Paiuk, M. Kvassay, A model of a centralized security system, as an information technology for the synthesis of an OS architecture protected against the leakage of confidential information, 1st International Workshop on Advanced Applied Information Technologies (AdvAIT-2024) volume 3899, Khmelnytskyi Ukraine and Zilina Slovakia, 2024, pp. 224-233.

[2] Y. Stetsyuk, M.Stetsyuk, B. Savenko, A. Kwiecien, L. Kopania, Method of random dynamic control transfer between network nodes for a partially centralized OS security system, IntelITSIS'2025: 6th International Workshop on Intelligent Information Technologies and Systems of Information Security, volume 3963, Khmelnytskyi Ukraine, 2025, pp. 264-283.

[3] S. Leventopoulos, K. Pipyros, D. Gritzalis, Retaliating against cyber-attacks: a decision-taking framework for policy-makers and enforcers of international and cybersecurity law, Int. Cybersecur. Law Rev, 5 (2024) 237–262. doi:10.1365/s43439-024-00113-5.

[4] N. D. Viet, D.D. Quan, Proposing, A New Approach for Detecting Malware Based on the Event Analysis Technique. International Journal of Innovative Technology and Exploring Engineering 8 (2023) 21 - 27 doi:10.35940/ijitee.h9651.0712823.

[5] B. Hajimirzaei, N.J. Navimipour, Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm. ICT Express, 5(1) (2019) 56–59. doi:10.1016/j.icte.2018.01.014.

[6] J. Dalou, B. Al-Duwairi, M. Al-Jarrah, M. Adaptive entropy-based detection and mitigation of DDOS attack in software defined networks. International Journal of Computing, (2020) 19(3), 399-410. https://doi.org/10.47839/ijc.19.3.1889.

[7] S. Singaravelan, P. Velayutha Perumal, R. Arun, V. Selvakumar, D. Murugan, Deep Learning-Based Echo State Neural Network for Cyber Threat Detection in IoT-Driven IICS Networks. International Journal of Computing, (2024), 23(2), 205-210.https://doi.org/10.47839/ijc.23.2.3538.

[8] R.S., Kanavalli, A., Gupta, A., Pattanaik, S. Agarwal, . Real-time DDoS Detection and Mitigation in Software Defined Networks using Machine Learning Techniques. International Journal of Computing, (2022), 21(3), 353-359. https://doi.org/10.47839/ijc.21.3.2691.

[9] A.J. Govindaram, FLBC-IDS: a federated learning and blockchain-based intrusion detection system for secure IoT environments, Multimed Tools Appl 84 (2025) 17229–17251. doi:10.1007/s11042-024-19777-6.

[10] D. Liao, R. Zhou, H. Li, GE-IDS: an intrusion detection system based on grayscale and entropy, Peer-to-Peer Netw. Appl. 15 (2022) 1521–1534. doi:10.1007/s12083-022-01300-z.

[11] M. Lopez-Vizcaino, F.J. Novoa, D. Fernandez, F. Cacheda, Time Aware F-Score for Cybersecurity Early Detection Evaluation, BASEL: Mdpi volume 14(2) (2024) 574. doi:10.3390/app14020574

[12] H. Lu, Y. Zhao, Y. Song, Y. Yang, G. He, H. Yu, Y. Ren, A transfer learning-based intrusion detection system for zero-day attack in communication-based train control system, Cluster Comput 5 (2024) 8477–8492. doi:10.1007/s10586-024-04376-9.

[13] S. Lysenko, K. Bobrovnikova, O. Savenko, A. Kryshchuk, BotGRABBER: SVM-Based Self-Adaptive System for the Network Resilience Against the Botnets' Cyberattacks, Communications in Computer and Information Science . volume 1039 (2019) 127 – 143. ISSN: 1865-0929.

[14] S. Lysenko, O. Savenko, K. Bobrovnikova, A. Kryshchuk, B. Savenko, Information technology for botnets detection based on their behaviour in the corporate area network, Communications in Computer and Information Science, volume 718 (2017) 166–181. ISSN: 1865–0929.
[15] S. Lysenko, O. Savenko, K. Bobrovnikova, A. Kryshchuk, Self-adaptive system for the corporate area network resilience in the presence of botnet cyberattacks, Communications in Computer and Information Science 860 (2018) 385-401.
[16] O. Savenko, K. Bobrovnikova, S. Lysenko, DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering, CEUR-WS volume 2104 (2018) 688-695.
[17] S. Lysenko, K. Bobrovnikova O. Savenko, A Botnet Detection Approach Based on The Clonal Selection Algorithm, 9th International Conference on Dependable Systems, Services and Technologies (DeSSerT-2018), Kyiv, Ukraine, May 24-27 2018, pp. 424-428.
[18] K. Singh, K. Singh Dhindsa, B. Bhushan, Performance analysis of agent based distributed defense mechanisms against DDOS attacks. International Journal of Computing, (2018), 17(1), 15-24. https://doi.org/10.47839/ijc.17.1.94.
[19] P. Bykovyy, V. Kochan, A. Sachenko and G. Markowsky, "Genetic Algorithm Implementation for Perimeter Security Systems CAD," 2007 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Dortmund, Germany, 2007, pp. 634-638, doi: 10.1109/IDAACS.2007.4488498
[20] P. Bykovyy, Y. Pigovsky, V. Kochan, A. Sachenko, G. Markowsky and S. Aksoy, "Genetic algorithm implementation for distributed security systems optimization," 2008 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, Istanbul, Turkey, 2008, pp. 120-124, doi: 10.1109/CIMSA.2008.4595845.
[21] Y.S. Zhang, Analysis of OSPU security effect and data assembly verification under semi-network OS architecture, Int. J. Inf. Secur 22 (2023), pp 1497–1509. doi: 10.1007/s10207-023-00702-1
[22] Y.T.Y. Azura, M.A. Azad, Y. Ahmed, An integrated cyber security risk management framework for online banking systems. J BANK FINANC TECHNOL (2025), doi: 10.1007/s42786-025-00056-3.
[23] P.C. Pathak, M. Nadeem, S.A. Ansar Security assessment of operating system by using decision making algorithms, Int. j. inf. tecnol. 17 (2025) 3609–3618. doi: 10.1007/s41870-023-01706-9
[24] P. Escaleira, V.A. Cunha, J.P. Barraca, A systematic review on security mechanisms for serverless computing, Cluster Comput 28 (2025) 465. doi: 10.1007/s10586-025-05371-4
[25] Q. Zhou, X. Jia, J. Chen, Q. Huang, H. Du, LightArmor: A Lightweight Trusted Operating System Isolation Approach for Mobile Systems. In: Pitropakis, N., Katsikas, S., Furnell, S., Markantonakis, K. (eds) ICT Systems Security and Privacy Protection. SEC 2024. IFIP Advances in Information and Communication Technology, vol 710. Springer, Cham., July 2024 pp 206–220 doi: 10.1007/978-3-031-65175-5_15
[26] C. Ma, N. Xi, D. Lu, CToMP: a cycle-task-oriented memory protection scheme for unmanned systems, Sci. China Inf. Sci. 67 (2024) 162305. doi:10.1007/s11432-023-3865-0.
[27] I. Herrera Montano, J. J. García Aranda, J. Ramos Diaz, Survey of Techniques on Data Leakage Protection and Methods to address the Insider threat, Cluster Comput 25 (2022) 4289–4302. doi:10.1007/s10586-022-03668-2.
[28] M. Gross, N. Jacob, A. Zankl, Breaking TrustZone memory isolation and secure boot through malicious hardware on a modern FPGA-SoC, J Cryptogr Eng 12 (2022) 181–196. doi:10.1007/s13389-021-00273-8.
[29] R. Meganathan, R. Anand, Security establishment using deep convolutional network model in cyber-physical systems, Multimed Tools Appl 83 (2024) 76201–76221. doi: 10.1007/s11042-024-18535-y
[30] N. Rai, J. Grover, Analysis of crypto module in RIOT OS using Frama-C, J Supercomput 80 (2024) 18521–18543. doi: 10.1007/s11227-024-06171-0.
[31] C.S.Yadav, S. Gupta, A Review on Malware Analysis for IoT and Android System, SN COMPUT. SCI. 4 (2023) 118. doi:10.1007/s42979-022-01543-w.
[32] N. Sharma, P. G. Shambharkar, Multi-layered security architecture for IoMT systems: integrating dynamic key management, decentralized storage, and dependable intrusion detection framework, Int. J. Mach. Learn. & Cyber (2025). doi:10.1007/s13042-025-02628-7

[33] A. Kumar, L. Kavisankar, S. Venkatesan, IoT device security audit tools: a comprehensive analysis and a layered architecture approach for addressing expanded security requirements, Int. J. Inf. Secur. 24 (2025) 13. doi: 10.1007/s10207-024-00930-z
[34] M. Seddigh, M. Esfahani, S. Bhattacharya, Breaking KASLR on mobile devices without any use of cache memory (extended version), J Cryptogr Eng 14 (2024) 281–294. doi: 10.1007/s13389-023-00344-y.
[35] Z. Qian, R. Xia, G. Sun, X. Xing, K. Xia, A measurable refinement method of design and verification for micro-kernel operating systems in communication network, Digital communications and networks, 10 (2023), volume 9(5), pp.1070-1079. doi: 10.1016/j.dcan.2022.03.024.
[36] S.T. Cheng, G.J. Horng, C.W. Hsu, Per-user network access control kernel module with secure multifactor authentication, J Supercomput 80 (2024) 970–1008. doi:10.1007/s11227-023-05480-0.
[37] W.T. Li, Z.X. Wang, J.Y. Gu, Harmonizing Security and Performance in Microkernel File Servers, J. Comput. Sci. Technol. 40 (2025) 464–481. doi:10.1007/s11390-025-3762-3.
[38] C. Cho, Y. Seong, Y. Won, Mandatory Access Control Method for Windows Embedded OS Security, Electronics (Basel), (2021), volume10(20), 2478. doi: 10.3390/electronics10202478
[39] S. Islam, N. Basheer, S. Papastergiou, Intelligent dynamic cybersecurity risk management framework with explainability and interpretability of AI models for enhancing security and resilience of digital infrastructure, J Reliable Intell Environ 11 (2025) 12. doi:10.1007/s40860-025-00253-3.
[40] P. Derasari, G. Venkataramani, Autonomous Hardware-Based Proactive Defenses with Deep Reinforcement Learning, J Hardw Syst Secur (2025). doi:10.1007/s41635-025-00163-z.
[41] W.Wu, H.Fouzi, B. Benamar, Deep learning-based stacked models for cyber-attack detection in industrial internet of things, Neural Comput & Applic (2025) 32. doi: 10.1007/s00521-025-11418-9
[42] A. Kashtalian, S. Lysenko, T. Kysil, A. Sachenko, O. Savenko, B. Savenko Method and Rules for Determining the Next Centralization Option in Multicomputer System Architecture, International Journal of Computing 24(1) (2025) 35-51. https://doi.org/10.47839/ijc.24.1.3875
[43] H. Meziane, N. Ouerdi, A survey on performance evaluation of artificial intelligence algorithms for improving IoT security systems, Sci Rep 13 (2023) 21255. doi:10.1038/s41598-023-46640-9.
[44] D. Ajish, The significance of artificial intelligence in zero trust technologies: a comprehensive review, Journal of Electrical Systems and Inf Technol 11 (2024), 30. doi:10.1186/s43067-024-00155-z.
[45] L. Fritsch, A. Jaber, A. Yazidi, An Overview ofArtificial Intelligence Used in Malware. In: E. Zouganeli, A. Yazidi, G. Mello, P. Lind (eds), Nordic Artificial Intelligence Research and Development, NAIS 2022, Communications in Computer and Information Science, volume 1650, Oslo Norway, 2022, pp. 41-51. doi:10.1007/978-3-031-17030-0_4.