ISSN 2347 - 3983
Volume 8. No. 4, April 2020
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter12842020.pdf
https://doi.org/10.30534/ijeter/2020/12842020

Development of an Algorithm for Implementing Mandatory and Role-Based Access Control

Mir-khusan Kadirov[1], Zoxidjon Tulyaganov[2], Nodirakhon Tojikhujaeva[3], Nazimakhan Karimova[4]

[1]Department Information Technologies, Tashkent State Technical University named after Islam Karimov, Tashkent, Uzbekistan, mirxusan.qodirov@tdtu.uz
[2]Department Information Technologies, Tashkent State Technical University named after Islam Karimov, Tashkent, Uzbekistan, zohidtulyaganov7@gmail.com
[3]Department Information Technologies, Tashkent State Technical University named after Islam Karimov, Tashkent, Uzbekistan, nodira.tojixujaeva_9@tdtu.uz
[4]Department Information Technologies, Tashkent State Technical University named after Islam Karimov, Tashkent, Uzbekistan, nozimaxon.karimova_6@tdtu.uz

ABSTRACT

The article discusses the algorithm for combining role and mandatory security policies. The security policy was simulated, including role-based and mandatory access control. Based on the model, a method has been developed for creating a complex security policy. An approach to combine mandatory security policies of two computer systems with different value grids is proposed. The result of combining these two approaches can be presented both as a concept based on security labels and as a hierarchy of roles. The protection of data integrity, to which subjects' access is granted in the information system, is achieved due to the fact that the information system is presented as part of the formal security model of logical mandatory and role-based access and information flow management and integrity control.

Key words: algorithm, object, subject, role, access control, access rights, access matrix, security label, MAC, RBAC, security policy, privacy levels, graph theory, lattice graphs, integrity, information flows.

1. INTRODUCTION

Nowadays, in the world, a huge attention is paid to the development and improvement of information protection systems on information and communication systems. At the current level of development of information and communication systems, problems of information protection in computer systems and networks, which is one of the most vital mechanisms for ensuring effective information security, are becoming especially relevant [1].
The threats of confidentiality, integrity and accessibility of information are becoming especially urgent [2]. In the context of the widespread use of computer technology for organizing business documents, storing and transmitting confidential information, the task of ensuring computer security comes to the fore. Statistics of facts of unauthorized access to information shows that most modern information systems are quite vulnerable from a security point of view.
The solution to the problem of protecting information from unauthorized access in any information system is based on the implementation of control and delimitation of access rights of subjects to protected resources, primarily to file objects, since they are designed specifically for storing processed data. The main type of information threats, for the protection of which an entire technology is created at each enterprise, is unauthorized access of attackers to data. Attackers plan criminal actions in advance, which can be carried out by direct access to devices or by remote attack using specially designed programs to steal information [3,4].
The development of a methodology for building a security policy for a structured enterprise based on models of role and mandatory access control is relevant. Many systems have role-based and mandatory security policies [5, 6].

2.SECURITY POLICY IMPLEMENTATION METHOD WITH ROLE AND MANDATE ACCESS CONTROL

As a rule, in accordance with the requirements of the security policy and data storage. Today, the use of database management systems has become common practice.
As part of the mandatory separation of access to multiple permissions, there should be no requirements for objects of a computer system. The decision on access is made by comparing the level of confidentiality and the level of trust. The roll, endowed with authorized access, allows the use of many permitted system operations. The decision to allow access is made based on a role comparable to the subject.
As part of the mandatory separation of access to multiple permissions, there should be no requirements for objects of a computer system. The decision on access is made by comparing the level of confidentiality and the level of trust. The roll, endowed with authorized access, allows the use of many permitted system operations. The decision to allow access is made based on a role comparable to the subject.
Attempts to provide combined access control services are built into a number of database management systems. For example, in the widespread Oracle DBMS [7], already in

version 7, the corresponding Trusted Oracle7 tool was developed, which allows the administrator to enter security labels in addition to roles. The main requirement for mandatory access control in this application was the dominance of the user label over the line label. Starting with the version of the DBMS Oracle 8, this product is called Oracle Label Security.

2.1 Formalization of role security policy.

The RBAC model allows you to differentiate the access of subjects in the system relative to the tasks they perform individually and at the same time provides tools for differentiating access to equivalent objects. Moreover, the user's access rights in the system are not permanent and may vary depending on the role with which the user is authorized.
RBAC security policy is based on the permission or prohibition of actions in the system as a whole without reference to individual objects of the system [8]. A privilege is a unit of access to system information. We assume that system information is representable using many objects$O$. A role is a named set of privileges, that is, a set of allowed types of access to system objects. The set of all possible types of access to system objects is denoted by A.
By privilege we mean a pair (L, m), where L is a system object $(L \in O)$, m is a nonempty set of access types $(m \in A)$.
A role is a named set of privileges, which in the future will be represented as a pair $(rname, rpset)$, where $rname$ is a unique identifier, $rpset$ is a set of privileges.
If the role r is defined, then its name is $r.rname$, and the set of privileges is $r.rpset$.
The main elements of the basic RBAC model are:
$U -$ many users;
$R -$ many roles;
$P -$ many access rights to computer system objects;
$S -$ many user sessions;
We also define a function that plays an important role in the administration of systems with role-based access control: $\Psi: R \to 2^{|P|}$. This mapping shows the privileges of the specified role. In fact, $\Psi(r) = r.rpset$. The role concept provides access to system information.
The role path $p(r_i, r_j)$ between the two roles r_i and r_j will be called the chain $r_i \to^* r_j$.
For a given relation on the set of roles, we can associate a directed graph in which an arc (r_1, r_2) exists if and only if the role r_1 is authorized for the role r_2. Obviously, the role path $p(r_i, r_j)$ is isomorphic to the oriented path in this digraph from the vertex r_i to the vertex r_j.
Trivial is a role path consisting of one role, that is, a path of zero length from a vertex to itself. We say that the role r_i dominates the role r_j, and the role r_j submits to the role r_i if there is a role path $p(r_i, r_j)$. Or in a graph statement: the vertex r_i dominates the vertex r_j, and the vertex $r_j -$ obeys the vertex r_i if there exists an oriented path $p(r_i, r_j)$.
It is easy to prove that the relation of dominance of one role over another defines a partial order relation on the set of roles R.

2.2 Formalization of mandate security policy.

Credential security policies are based on the concepts of the level of information security and the level of trust in the user. There are various approaches to determine the level of secrecy of information. The most general approach is based on a grid of values.
A lattice is a partially ordered set in which each two-element subset has both an exact upper (sup) and an exact lower (inf) face belonging to this set.
For A, B an element $C = sup(A, B)$ is called the exact or least upper bound if:

1. $A \le C, B \le C$.
2. $\forall D: D \le A, D \le B \Rightarrow C \le D$.

For A, B an element $E = inf(A, B)$ is called the exact or largest lower bound if:

1. $E \le A, E \le B$.
2. $\forall D: D \le A, D \le B \Rightarrow D \le E$.

Each object and subject of the system is associated with a "Security label", which is an element of the lattice. When a subject requests access to an object, security labels are compared. Access is allowed if the security label of the subject dominates the security label of the object, in other cases access is denied.
Due to the fact that the basic concepts of role-based access are formulated in terms of graph theory, we introduce similar concepts for mandatory access control.

A lattice graph is a directed graph whose vertices form a lattice. Moreover, the order relation is determined by the dominance relation on the set of graph vertices: if $\exists p(r_1, r_2)$ then $(r_1 \ge r_2) -$ The smallest upper bound sup грань (r_1, r_2) is defined as the nearest vertex that dominates r_1 and r_2. The largest lower bound inf (r_1, r_2) is defined as the nearest vertex subordinate to the vertices r_1 and r_2.
Let us more formally define the concepts of the smallest upper and largest lower faces in the context of a directed graph:
$r = sup(r_1, r_2) \Leftrightarrow$

1. $\exists p(r, r\_1) \& p(r, r\_2)$,that is, r is the upper bound.

If a$\exists p(r', r_1) \& p(r', r_2)$, then $\exists p(r', r)$, that is, r is minimal among all the upper faces.
$r = inf(r_1, r_2) \Leftrightarrow$

$\exists p(r\_1, r) \& p(r\_2, r)$, that is, r is the bottom face.

If $\exists p(r_1, r') \& p(r_2, r')$, then $\exists p(r, r')$, that is, r is maximal among all the lower bounds.
A lattice graph isomorphic to a given lattice is not unique. Indeed, for example, the graphs in Figure 1. are isomorphic to the same lattice (M, P), where $M = \{a, b, c, d\}$ is the set of lattice nodes, $P = \{(a, b), (a, c), (a, d), (b, d), (c, d)\}$ be the partial order relation defined on M. We call the lattice graphs G_1 and G_2 equivalent to $(G_1 \sim G_2)$, if they are isomorphic to the same lattice.

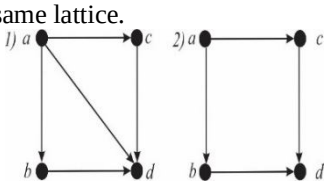

Figure 1: Equivalent lattice graphs.

A source in a network without oriented cycles dominates any vertex, and a sink obeys any vertex of this graph. Let $r -$ be an arbitrary vertex of the network without oriented cycles,

$s-$ is the source,$t-$ is the sink. We will build an oriented path, starting from the vertex r and adding one arc at each step. Therefore, the network has at least one oriented path $p\ (r,t)$. But then t obeys the vertex r.

The existence of the oriented path $p\ (s,r)$ is proved similarly, but it's construction is carried out in the direction opposite to the orientation of the arcs.

In a network without oriented cycles, for any pair of vertices, there are at least one upper face and one lower face.

The second reason for the "lattice" of the graph is easily illustrated by an example: graph (1) in Figure 2 - a network without oriented cycles, but it is not a lattice graph. Indeed, the vertices a and b have two incomparable lower bounds c and d, and another lower bound t is certainly smaller than c and d; the vertices c and d have two incomparable upper faces a and b, another upper face s is certainly larger than a and b.

But the requirement that there are no subgraphs in the network that have more than one sink or source, like the subgraph generated by the set of vertices $\{a,b,c,d\}$ (Figure 2 (1)), is not a sufficient lattice condition.

Indeed, adding the vertex e (Figure 2 (2)) makes the considered network a lattice graph: $\mathrm{inf(a,b)} = \mathrm{e}$ (c and d are still incomparable, but $(\mathrm{e} \geq \mathrm{c})\&(e \geq d)$) and $\mathrm{sup(c,d)} = \mathrm{e}$ ($a$and b are still incomparable, $but\ (e \geq a)\ \&\ (e \geq b)$).

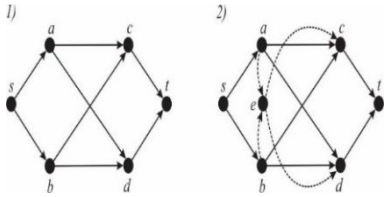

Figure 2: A network that is not a lattice graph (1) and a network that is a lattice graph (2)

If the role hierarchy graph is lattice, or it can be expanded to a lattice using a valid transformation, then the role security policy allows for a consistent combination with a mandatory security policy.

If necessary, we expand the graph of the hierarchy of roles to the lattice graph and denote it by GM. The mandatory security policy is defined by the lattice L. Then we can take the Cartesian product of the lattice built on the vertices of the lattice graph GM and the lattice L. Such a Cartesian product, denoted by $GM \times L$, is a lattice [9, 10, 11].

As an illustration of the proposed method, we combine the role-based security policy shown in figure 3. and the mandatory security policy built on a linear set of three elements [12, 13].

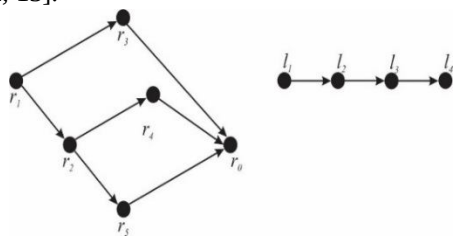

Figure 3:$\boldsymbol{R}$ role security policy and $\boldsymbol{L}$ value grid

A role policy is defined by six roles, one of which (r_0) is "empty", that is, it does not have any privileges and is subordinate to any other role. Let graph G be a network, and removing the drain turns it into a oriented tree. Then G is a lattice one. Let s be the source, t be the sink, and $\mathrm{G\{t\}} = \mathrm{T}-$ the tree obtained from the original graph by removing the sink. If $R-$ is the set of vertices of the graph G, then $R \setminus \{t\}$ is the set of vertices of the tree T and s is the root of the tree. Therefore, the graph shown in Figure 3. is a lattice.

Let the mandatory security policy be defined by the lattice L, the elements of which are the nodes l_1, l_2, l_3, l_4 and the order relation is set in such a way that $l_1 \geq l_2 \geq l_3 \geq l_4$.

According to the role-based, security policy allows a consistent combination with a mandatory security policy, a consistent combination of defined security policies is possible. To do this, it is necessary to construct the lattice $R \times L$, which is the Cartesian product of the lattices R and L, where R is the lattice defined by the lattice graph shown in Figure 2.

The elements of the $R \times L$ lattice are pairs (r_i, l_j), for $i = 0,\dots,4$ and$j = 1,\dots,5$. Moreover, the order relation is defined as follows: $(r_i, l_j) \geq (r_k, l_m)$ if $r_i \geq r_k$ and $l_j \geq l_m$. Note that the nodes r_2 and r_3, r_4 and r_5, r_3 and r_4, r_3 and r_5 are pairwise incomparable. The lattice graph isomorphic to the $R \times L$ lattice is shown in Figure 4.

On the resulting $R \times L$ lattice, you can set a mandatory security policy. In turn, a role-based security policy can be built on the resulting oriented graph.

And so, the result of combining these two approaches can be presented both as a concept based on security labels, and as a hierarchy of roles.

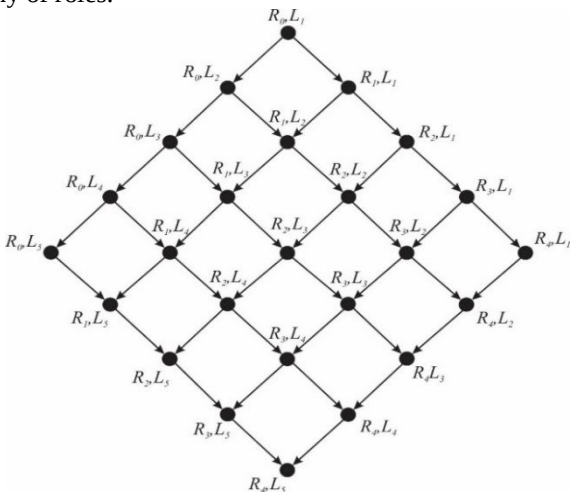

Figure 4: Combining role and credential security policies

3.IMPLEMENTATION OF MANDATORY AND ROLE ACCESS MANAGEMENT

Protection of data integrity, which is granted access to subjects in the information system, is achieved due to the fact that the information system is presented in the framework of the formal security model of logical mandatory and role-based access and information flow management and integrity control:

- each role is assigned an integrity level that does not exceed the integrity levels of the roles to which this role is subordinate in the hierarchy;
- each role is assigned access rights to ownership or record to the entity only when the integrity level of the entity is not higher than the integrity level of the role;
- the subject is granted access to the role only when the level of integrity of the role does not exceed the current level of integrity of the subject.

As a security state of a protected information system, a full set of access entities is considered, including subjects, objects, containers and roles, and their security parameters, the composition and impact of which on security is determined by the type and version of the operating environment of the protected information system, including such entities and security settings:
- accounts of trusted and untrusted users;
- elements of the file system, including disks, directories, files, links;
- processes, threads, daemons, drivers, devices, services, synchronization objects;
- lists of privileges and access rights of roles to entities, labels of shared containers;
- access level labels, confidentiality and integrity, CCR tags of the access method inside containers; hierarchies of entities, including roles and subjects.

Formally, when implementing the method, in general, the information system $\sum(G^*, OP)$ is represented by the set of all its states - G^* and the set of state transformation rules – OP [6]. Moreover, each state of the information system $\sum(G^*, OP)$ is represented by a tuple (PA, A, F) and includes the following elements in its description:

$E = O \cup C -$ is the set of entities, where O is the set of objects (for example, files), C is the set of containers (for example directories) and $O \cup C = \emptyset$;

$S \subseteq E -$ many entities operating on behalf of user accounts;

$R_r = \{read_r, write_r, execute_r, own_r\} -$ many types of access rights, while $read_r -$ is the right to read, $write_r -$ is the right to write, $execute_r -$ is the right to execute, $own_r -$ is the right to own;

$R_a = \{read_a, write_a, own_a\} -$ many types of access, with $read_a -$ read access, $write_a -$ write access, $own_a -$ ownership access;

$Rf = \{write_m, write_t\} -$ many types of information flows (from memory and time, respectively);

$P \subseteq (E \cup R) \times R_r -$ many access rights to entities and roles;

$A \subseteq S \times (E \cup R) \times R_a -$ multiple access of entities to entities and roles;

$F \subseteq (E \cup R) \times (E \cup R) \times R_f -$ many information flows;

$PA: R \to 2^P -$ a function of access rights to entities and roles of roles, and for each access right $p \in P$ there is a role $r \in P$ such that the condition $p \in PA(r)$ is satisfied;

$(LC, \leq) -$ multi-level security lattice of confidentiality levels (as a rule, the Cartesian product of a linear scale of data confidentiality levels and the set of all subsets of a finite set of non-hierarchical data categories);

$a: E \cup R \to LC -$ a function that sets the level of confidentiality for each entity or role;

$a: S \to LC -$ a function that sets for each subject its current access level;

$(LI, \leq) -$ linear scale of two levels of data integrity, where $LI = \{d_{low}, d_{high}\}$;

$m: E \cup R \to LI -$ a function that sets the integrity level for each entity or role;

$n: S \to LI -$ a function that sets for each subject its current level of integrity;

In each state of the information system $\sum(G^*, OP)$ ensure that the following conditions are met:
- each role has access rights $execute_r, -$ to all roles: for every two roles $r, r' \in R$ the condition $(r, execute_r) \in PA(r')$is fulfilled;
- the confidentiality level of the entity or role that is part of the container entity or role, respectively, does not exceed its confidentiality level: for entities or roles $e, e' \in E \cup R$, if $e \leq e'$, then $a(e) \leq a(e')$;
- the integrity level of an entity or role that is part of a container entity or role, respectively, does not exceed its integrity level: for entities or roles $e, e' \in E \cup R$, if $e \leq e'$, then $a(e) \leq a(e')$;
- a role can contain access rights to own or write to entities or roles with no higher integrity level than it: for the role $r \in R$ and the entity or role $e \in E \cup R$, if $(e, \alpha\_r \in PA(r)$, then $m(e) \leq m(r)$, where $\alpha_r \in \{own_r, write_r\}$.

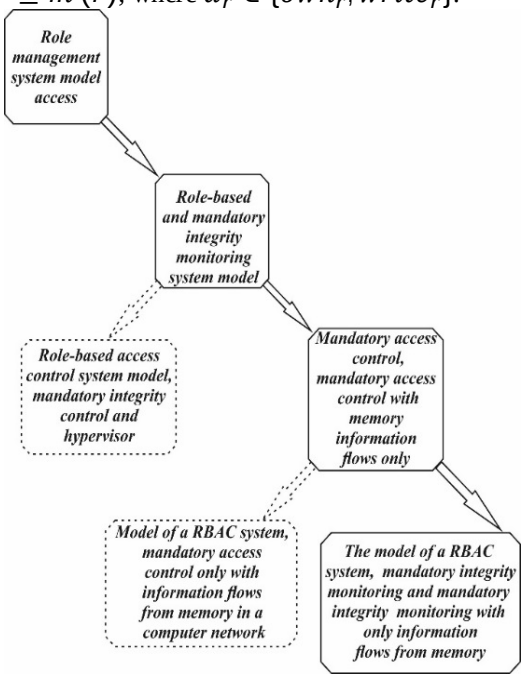

Figure 5: Hierarchical representation of the mandatory and role access model and its possible extensions

In the hierarchical description of mandatory access control based on the role-based access control model, the following levels are currently set (figure 5):

- the first level is a RBAC system model;
- the second level is a model of a RBAC system and mandatory integrity control;

- the third level - a model of a RBAC system, mandatory integrity control and mandatory access control only with information flows from memory;

- the fourth level is a model of a RBAC system, mandatory integrity control and mandatory access control with information flows in memory and in time.

This approach allows to complicate the wording of the definitions and statements of the model gradually as you include elements corresponding to the next level under consideration. Each lower level of the model represents an abstract system, the elements of which are independent of new elements belonging to a higher level, which, in turn, inherits and, if necessary, corrects or supplements the elements of the lower level.

With such a hierarchical description, the hypervisor model for the OS (figure 5) is considered as an alternative (additional) third level (model of the role-based access control system, mandatory integrity monitoring and hypervisor), since it can be assumed that the hypervisor for the OS must ensure the correct functioning of its mandatory integrity control, and mandatory access control in the OS should not be implemented by means of a hypervisor [14, 15]. Similarly, the role-based model of access control in a computer network should be considered alternative to the fourth level of representation of mandatory access control based on the role-based access control model, since mandatory integrity control and mandatory access control with memory information flows are essential in this model.

If the role-based access control model meets the requirements of strictly mandatory access control, then in it for any objects $o, o' \in O$ such that $c(o) > c(o')$, an information flow from o to o' is impossible.

For lattices of confidentiality levels $(L, \leq) = \{LS, HS\}$ and integrity levels $(L, \leq) = \{LI, HI\}$, Figures 6-7 show role hierarchies for two possible combinations of role hierarchies of liberal or strict access control with liberal or strict mandatory integrity control.

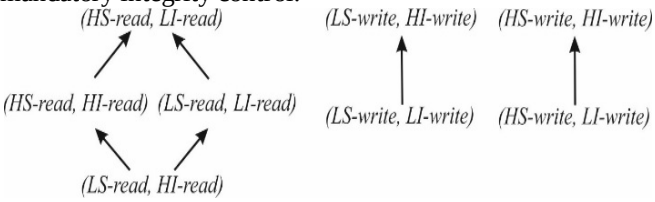

Figure 6: Strict Mandatory Access Control and Liberal Mandatory Integrity Control

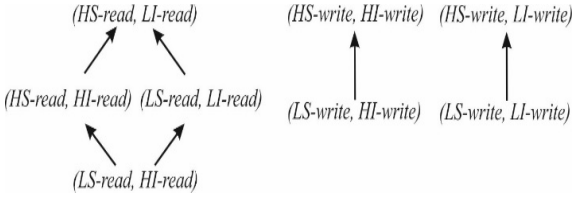

Figure 7: Liberal Mandatory Access Control and Strict Mandatory Integrity Control

The dependencies of the conditions and results of applying de jure and de facto rules for transforming the states of the presented model will change significantly. The diagram of these dependences is shown in Fig. 8, in which solid lines show dependencies that arise when applying de jure rules (with the exception of information flows), and dashed lines show dependencies that arise when applying de facto rules or as a result of obtaining information flows when applying de jure rules.

Since the mandate role model uses the constraint mechanism, consider the following statement.

Let $G_0 -$ be the initial state of the system $\sum(G^*, OP, G_0)$, in which the functions $(i_u, i_e, i_r, i_s)_0$ satisfy the level of integrity of the role does not exceed the levels of integrity of roles with which it is subordinate in the hierarchy of roles, as well as the current level of integrity of the subject-session should not exceed the integrity level of the user account on whose behalf it operates, and the current level of the subject-session to which it is subordinate in the hierarchy. The role integrity level cannot be higher than the integrity level of the user account that can be authorized on it, and the current integrity level of the subject-session, in the many current roles of which it is included. Then in any state G_N of any trajectory $G_0 \vdash_{op1} G_1 \vdash_{op2} \dots \vdash_{opN} G_N$, where $op_1, \dots, op_N -$ are the rules of state transformation and $N \geq 0$, the functions UA_N, PA_N and $roles_N$ satisfy the relevant restrictions $Constrraint_U, Constrraint_P$ and $Constrraint_S$.

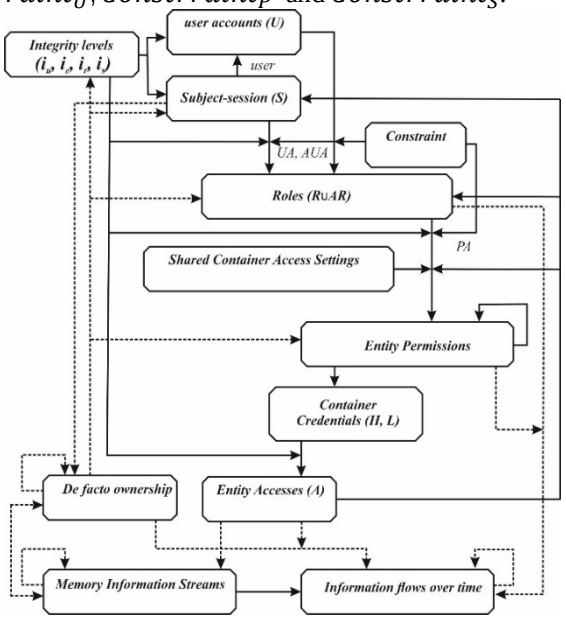

Figure 8"Scheme of dependence of conditions and results of applying the rules for transforming states of the mandatory role model

Thus, the proposed algorithm for implementing mandatory and role-based access control makes it possible to check the security status of a secure information system after trusted entities have completed their tasks of changing the system's functioning parameters and can increase the security of the information system taking into account the formation of access control rules.

4. EVALUATION OF THE PERFORMANCE OF THE DEVELOPED SOFTWARE PRODUCT

Based on the algorithm, the "MK Universal" software package was developed. The developed software package based on role and mandatory access control allows to increase the security of the information system, taking into account the formation of access control rules, and also allows to implement flexible access control rules that change dynamically during the functioning of the computer system. The hierarchical structure of the developed software package is presented in Figure 10.

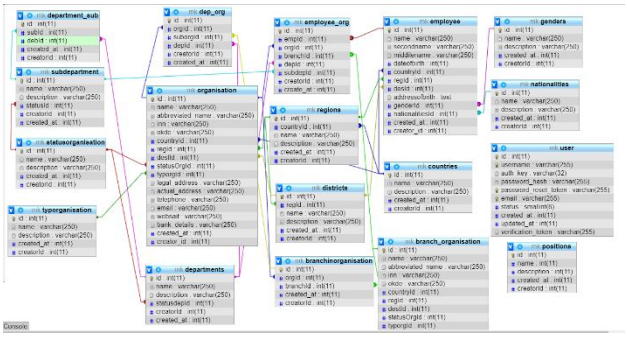

Figure 10: General structure of the software package

From the above diagrams it can be seen that, if we neglect the error, the time for collecting information about files grows almost linearly depending on the number of files. Also, it can be seen that the construction time of the adjacency matrix for the access graph has an exponential dependence on the number of files. Thus, the experimental data confirm the validity of the theoretical performance assessment.

Based on the data obtained, we will make a comparative analysis of information security tools (IST) against unauthorized access. We offer the following method for conducting a comparative analysis of the considered IST from unauthorized access:

1. A table of factors (criteria) is compiled, according to which an objective comparison of the analyzed IST from unauthorized access can be made;
2. Each factor is assigned a coefficient that expresses the weight or significance of this factor in the framework of the comparison;
3. The table of factors is filled with values for each of the analyzed IST from unauthorized access;
4. By summing the values of factors, taking into account their coefficients, the "efficiency" of each IST from unauthorized access is determined, while positive factors are taken with a plus sign, and negative factors with a minus sign;
5. By comparing the "efficiencies", the IST from the unauthorized access is selected, to a greater extent satisfying the modern requirements of consumers.

Thus, a comparative analysis of n IST from unauthorized access is reduced to solving the optimization problem (1).

$$\sum_{i=1}^{s} F_{ik}^{+} \cdot P_i^{+} - \sum_{j=1}^{t} F_{jk}^{-} \cdot P_j^{+} \to \max \qquad (1)$$

where $k \in [1, n], F_{ik}^{+}, i = \overline{1, s}$ – are the values of positive factors for the k –th IST from unauthorized access $F_i^{+} \in (0,1]$ – is the weight of the i-th positive factor, $F_i^{-} \in (0,1] j$ – is the weight of the jth negative factor.

If a factor value can be represented literals "+" and "-", then the replacement is normalized values by numbers 1 and 0, respectively.

5 RESULT ANALYSIS

Table 1 shows the values of the comparison criteria developed by studying documentation and analyzing sources [16] for the analyzed software and hardware-software IST from unauthorized access.

Table 1 shows the values of the comparison criteria developed by studying documentation and analyzing sources [16] for the analyzed software and hardware-software IST from unauthorized access.

Table 1: Comparative analysis of protection against unauthorized access to information

№	Factor weight	IST from UA / Faktor	MK- Universal	Secret Net	Kripton Zamok	Straj NT
1	1	Implementation of the Mandatory Model of Access Control	+	+	-	+
2	0.9	Implementation of OS "trusted boot" mechanisms	+	-	+	-
3	0.7	Providing integrity control of file system objects	+	-	-	+
4	0.1	Providing control over printing documents	-	+	-	+
5	0.2	Guaranteed destruction of deleted information	+	+	-	+
6	1	Event Registration	+	+	+	+
7	0.9	Windows OS Support	+	+	+	+
8	0.7	The ability to protect PCs networked	+	+	+	+
The value of the objective function $\sum_{i=1}^{s} F_{ik}^{+} \cdot P_i^{+} - \sum_{j=1}^{t} F_{jk}^{-} \cdot P_j^{-}$			5,9	4,4	4,1	5,6

The experiment carried out showed that the developed software package increased the security of the automated system by 13%.

6. CONCLUSION

To sum up, we suggest the following outcomes regarding proposal paper:

- the security policy was simulated, including role-based and mandatory access control.
- based on the model, a methodology has been developed for creating a comprehensive security policy;
- an approach to combine mandatory security policies of two computer systems with different value gratings is proposed. When combining the two approaches to the solution, the result is presented in the form of a concept based

on security labels and a hierarchy of roles;

- in the information system $\sum(G^*, OP)$, the rules for transforming states were implemented: de jure and de facto. The same rules are used to administer the parameters of the access control mechanism in the system;
- an algorithm for combining role and mandatory security policies has been developed;
- it was revealed that the integration of several different models provides an opportunity to reduce the vulnerability of the network related to obtaining unauthorized access and ensure the security of the information system;
- the practical value is to obtain a methodology for building security policies for enterprises and computer systems to prevent information leakage.

REFERENCES

1. Sherzod, G.,Dilmurod, A.,Nodira, M.,Husniya, A. **Construction of schemes, models and algorithm for detection network attacks in computer networks**, *International Journal of Innovative Technology and Exploring Engineering*, Volume 8, Issue 12, 2019, pp. 2234–2241. https://doi.org/10.35940/ijitee.L2481.1081219
2. Mirpulatovich K. M., Zakirovna T. N., Ismoilovna K. G. **Classification of Modern Security Monitoring Systems in Computer Systems and Networks**, *International Journal of Advanced Research in Science, Engineering and Technology*, Vol. 5, Issue 9, India 2018, p. 6764–6769.
3. Rajaboevich G. S., Mirpulatovich K. M., Yakubdjanovich T. Z. The **Methodology of the Ways for Increasing the Efficiency of Intrusion Detection Systems** //*International Journal of Engineering Innovations and Research*, Vol. 5, Issue 5, India 2016, P. 296–301.
4. Sagatov M., Irgasheva D., Mirhusan K. **Construction Hardware Protection Infocommunication Systems from Network Attacks** //*Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE).*, 2015. – P. 271.
5. Rivera Sanchez Y. K., Demurjian S. A., Baihan M. S. **A Service-Based RBAC & MAC Approach Incorporated into the Fast Healthcare Interoperable Resources (FHIR) standard** //*Submitted to special issue on 2017IEEE Mobile Cloud Conference submissions, Elsevier journal of Digital Communications and Networks*. – 2017.
6. Devyanin P.N. **Security models for computer systems. Control of access and information flows**. Textbook for higher schools. 2nd ed. M.: Goryatchaya liniya – Telecom, 2013. P 338.
7. Knox D. et al. **Applied Oracle Security: Developing Secure Database and Middleware Environments**. – *McGraw-Hill, Inc.*, 2009.
8. Sun W., Su H., Liu H. **Role-Engineering Optimization with Cardinality Constraints and User-Oriented Mutually Exclusive Constraints** //*Information*. – 2019. – Т. 10. – №. 11. – P. 342. https://doi.org/10.3390/info10110342
9. Mac Lane S. **Categories for the working mathematician**. – *Springer Science & Business Media*, 2013. – Т. 5.
10. Grätzer G., Wehrung F. (ed.). **Lattice theory: special topics and applications**. – *Springer International Publishing*, 2016.
11. Garg V. K. **Introduction to lattice theory with computer science applications**. – New Jersey: Wiley, 2015. https://doi.org/10.1002/9781119069706
12. Belim S.V., Rakitsky Yu.S. **Association of Mandate Security Policies** // *Mathematical Structures and Modeling*. - 2010. - No. 1 (21).
13. Belim S.V., Bogachenko N.F., Rakitsky Yu.S. **Combining role and credential security policies** // *Problems of information processing and protection*. - 2010. P. 117-132.
14. Kadirov M.M.**Approach to assessing the security of information from unauthorized access**. *International Journal of Advanced Research in Science, Engineering and Technology*. Vol. 6, Issue 12, India 2019, p. 12182-12187.
15. Devyanin P. N. **Implementation of a non-degenerate lattice of integrity levels within the framework of the hierarchical representation of the MROSL DP model** // *Applied Discrete Mathematics. Application.* - 2017. - No. 10.
16. Sandhu R. S., Samarati P. **Access control: principle and practice** //*IEEE communications magazine*. – 1994. – Т. 32. – №. 9. – С. 40-48. https://doi.org/10.1109/35.312842