Genetic Algorithms for Role Mining in Critical Infrastructure Data Spaces

Igor Saenko
ITMO University
49, Kronverkskiy prospekt, St.Petersburg, Russia
ibsaen@comsec.spb.ru

Igor Kotenko[1,2]
[1] Saint-Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences
14-th Liniya, 39, St.Petersburg, 199178, Russia
[2] ITMO University
49, Kronverkskiy prospekt, St.Petersburg, Russia
ivkote@comsec.spb.ru

ABSTRACT[1]

In the paper, a Role Mining problem, which is the cornerstone for creating Role-Based Access Control (RBAC) systems, is transferred to the domain of data spaces. RBAC is the most widespread model of access control in different multi-user information systems, including critical infrastructures. The data spaces is the perspective concept of creating information storage systems, which transforms the concept of databases, integrating in one system the information resources from other systems, and allows us to control their security on a centralized basis. The paper considers a mathematical statement of the RBAC design problem for data spaces and offers the approaches to its solving based on genetic algorithms. The proposed approaches consider requirements of compliance with role-based security policies in case of combining all users' sets and all permissions' sets in the data space. The paper considers main decisions on creation and enhancement of genetic algorithms which implementation increases their operational speed. The experimental assessment of the offered genetic algorithms shows their high performance.

CCS CONCEPTS

• **Security and privacy → Access control;** *Computing methodologies*; Heuristic function construction

KEYWORDS

RBAC, role mining, access control, genetic algorithm, data space

ACM Reference format:
I. Saenko and I. Kotenko, 2018. Genetic Algorithms for Role Mining in Critical Infrastructure Data Spaces. In *GECCO '18: Genetic and Evolutionary Computation Conference Companion Proceedings, 8 pages.*
DOI: 10.1145/3205651.3208283

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07…$15.00
https://doi.org/10.1145/3205651.3208283

1 INTRODUCTION

Nowadays, one of the perspective directions on creating storage systems in critical infrastructures is implementation of data spaces [1]. The data spaces assume that different information systems, which are a part of a critical infrastructure, are integrated together in the common space. At the same time, the user's sets and sets of information resources come together in the conditions of preserving the access control policies realized in the original local information systems. As a result, any user of the data space can get access to any information resource of the data space [2].

Role-Based Access Control (RBAC) is an access control model which is the most widespread in different multi-user information systems. The RBAC model has many varieties and is well supported by different developers of information security tools [3]. The main idea of the RBAC model consists in changeover of the "users – permissions" mapping to the sequentially connected mappings "users – roles" and "roles – permissions". This idea was offered in [4]. The RBAC model found wide popularity, because of its application simplifies the design of access control schemes and reduces the administrator's costs on their reconfiguration.

The problem of searching the "users – roles" and "roles – permissions" mappings, meeting the requirements of the access control policy, received the name Role Mining Problem (RMP), and is a difficult problem from the Data Mining area [5]. Many works offer different statements, methods and algorithms of its solving [6, 7]. Considering that this problem is NP-full, in our previous works [8-10] the genetic algorithms, which were intended both for original design, and for reconfiguration of RBAC schemes [11], were suggested.

The essential feature of solving RMP in the data spaces is that in this case it is necessary to consider the following additional restrictions. First, the same user can be a user in different local information systems integrated in the data space. Secondly, the same information resource can be available to users in different local information systems. Thirdly, the common RBAC scheme of the data space shall integrate the RBAC schemes of local systems. At the same time, the requirements of access control policies, set in local RBAC schemes, shall not be violated in the common RBAC scheme.

As a result, the Role Mining problem in the data space becomes a new problem, which has two directions of their solving. The first direction is related to combining local access control policies to a common policy and its use as initial data for

solving the problem of the RBAC scheme design by the known methods, including genetic algorithms. The second direction is related to combining local RBAC schemes in the initial common RBAC scheme. In this case local RBAC schemes can be considered as additional elements of initial data in case of solving the problem of reconfiguring the RBAC scheme. The first direction is selected when the Role Mining problem for the data space is solved in off-line mode, and the second one – for on-line mode.

Comparative assessment of results of the problem solution for both directions is *one of main goals* of the paper. *The other goals* are as follows:

(1) Mathematical statement of the problem of the common RBAC scheme formation in the data space and development of methods of its solving;

(2) Development and enhancement of genetic algorithms to solve the problem in the statement offered;

(3) Implementation of the test bed and the experimental assessment of offered genetic algorithms.

The main *theoretical contribution* of the paper consists in the following.

First, the mathematical background for the common RBAC scheme formation in the data space, which describes the problem statement and possible methods of its solving, are offered.

Secondly, the advanced genetic algorithm is suggested to solve the problem by the method related to serial reconfiguration of the initial version of the common RBAC scheme. The proposed enhancements are related to adaptation of the fitness function of the genetic algorithm to the restrictions imposed from the data space.

Thirdly, the developed test bed considers conditions of combining local RBAC schemes that allows us to increase the reliability of estimates received during modeling.

Further structure of the paper is as follows. Section 2 provides the overview of related works. Section 3 considers the mathematical foundations of the problem. Section 4 discusses the issues of development of the genetic algorithms. Section 5 presents the testbed and the experimental results. Conclusions and future research directions are outlined in section 6.

2 RELATED WORK

The papers [5, 7] show that different versions of the RMP statements (Basic RMP, Edge RMP et al.) are the NP-full problems. These versions differ from each other in the criteria for assessment of the access control scheme.

Various approaches were proposed to solve different versions of RMP. In [12-13] simple heuristic algorithms for solving RMP in the Basic option are suggested. These solutions are based on combinatorial algorithms. The papers [14-15] offer probability models to reduce the complexity of the solution. However, in case of this approach the high accuracy of the problem solution is not guaranteed. An approach based on clustering model for Boolean data is offered in [16]. This paper shows that it can be applied only for separate versions of the RMP. The cluster based approach was considered in [17]. However, a lack of this work is the need of accounting the additional parameters characterizing business processes and needs of users. The cost-driven approach is proposed in [18], where the criterion considering administration expenses is used. We will use this criterion to create the fitness function in one of the offered genetic algorithms.

There are works in which the issues of creating the role-based access control models in the data spaces are considered. In [19-20] the integrated RBAC model for the adaptive flow system is offered. The distributed RBAC model was suggested in [21]. The paper [22] shows that in case of storing the heterogeneous data in a data space it is necessary to make simple requests with associations of similar resources. We implement this principle in our paper. In [23] an approach to integration of secure information by web-based framework is considered. This and other works considered above show that the topics of access control in data spaces is rather novel. At the same time, there are no works where the issues connected to Role Mining in data spaces are considered.

There are works in which the possibility to apply the genetic algorithms for the access control and computer security is investigated. For example, in [24] authors apply genetic algorithms for access control of web services. However, these problems are less difficult than RMP. In [25] the architecture of intrusion detection system, where the genetic algorithm uses both temporal and spatial information of the generated rule set, is considered. However, in this system the genetic algorithm is standard. In [26] the genetic algorithm is used for solving the multi-objective optimization problem. In this paper the network architecture is searched, and the parameters of Medium Access Control protocol that achieve the Pareto-optima are calculated. However, in this task the variables have scalar values; therefore, this task is less difficult than RMP. In [27], the genetic algorithm for selecting a minimal number of optimally positioned monitors to capture network traffic is applied, and one chromosome with binary elements is used.

The review of related works shows that, despite existence of individual methods and algorithms of solving RMP, including genetic algorithms, in known works the issues of the uniform RBAC scheme formation and applying genetic algorithms for these purposes are not considered.

3 MATHEMATICAL BACKGROUND

Let the data space integrates S local information systems. In each s-th local system (s = 1, …, S) the access control scheme $RBAC_s$ exists which is set as follows (designations are taken from [5, 12]):

$$RBAC_s = <U_s, PRMS_s, ROLES_s, UA_s, PA_s>, \quad (1)$$

where $U_S = \{u_{si}\}, i = 1,...,m_S, m_S = |U_S|$ is a set of users in s-th local system; $PRMS_S = \{p_{sj}\}, j = 1,...,n_S, n_S = |PRMS_S|$ is a set of permissions; $ROLES_S = \{r_{sl}\}, l = 1,...,k_S, k_S = |ROLES_S|$ is a set of roles; UA_s is a binary mapping between sets U_s and $ROLES_s$; PA_s is a binary mapping between sets $PRMS_s$ and $ROLES_s$.

Let us represent the mapping UA_s between the set of users U_s and the set of roles $ROLES_s$ as the $m_S \times k_S$ Boolean matrix $\mathbf{X_s}$, in which 1 in cell $\{il\}$ indicates the assignment of roles l to user i.

Let us represent the mapping PA_s between the set of roles $ROLES_s$ and the set of permissions $PRMS_s$ as the $k_S \times n_S$ Boolean matrix $\mathbf{Y_s}$.

Existence of mappings UA_s and PA_s between the sets of users, roles and permissions allows us to say that there is a mapping UPA_s between the set of users U_s and the set of permissions $PRMS_s$ which can be represented by the Boolean matrix $\mathbf{A}_s$.

The matrix $\mathbf{A}_s$ defines the given initially mapping which is defined by a security policy of the s-th local system and which needs to be executed by means of the access control scheme $RBAC_s$.

The essence of forming the access control scheme $RBAC_s$ consists in search of such matrixes $\mathbf{X}_s$ and $\mathbf{Y}_s$ in case of which the following equality would be fair:

$$\mathbf{X}_S \otimes \mathbf{Y}_S = \mathbf{A}_S , \quad (2)$$

where the symbol $\otimes$ denotes Boolean matrix production.

The matrix equation (2) has very large number of solutions. Therefore, when finding $\mathbf{X}_s$ and $\mathbf{Y}s$, it is necessary to consider additional criteria. The most known criteria are: (1) minimum quantity of roles; (2) minimum summary quantity of unit elements in matrixes $\mathbf{X}_s$ and $\mathbf{Y}_s$.

Finding solutions to the equation (2) with taking into account these and/or some other criteria is the NP-full problem of Boolean Matrix Factorization [28] known as Role Mining Problem (RMP) [5, 7]. RMP version when we account the first criterion is named as the Basic RMP, and the RMP version when we account the second criterion is named as the Edge RMP.

Formally, the first criterion can be represented as:

$$\left|ROLES_S\right| \Rightarrow \min . \quad (3)$$

The second criterion is as follows:

$$\left|UA_S\right| + \left|PA_S\right| \Rightarrow \min . \quad (4)$$

Further in our paper we will use the criterion of the Basic RMP. In case of the criterion of the Edge RMP or other criteria of RMP, the further reasoning is similar.

The common access control scheme *RBAC* in the data space by the analogy with expression (1) is set as follows:

$$RBAC = <U, PRMS, ROLES, UA, PA>, \quad (5)$$

where $U = \bigcup U_S$ is a common set of users of the data space; $PRMS = \bigcup PRMS_S$ is a common set of permissions; *ROLES* is a common set of roles; *UA* is a binary mapping between the sets *U* and *ROLES*; *PA* is a binary mapping between the sets *PRMS* and *ROLES*. The mappings *UA* and *PA* define the mapping *UPA* between the sets *U* and *PRMS*.

By the analogy with the mappings in local RBAC schemes, we will represent the mappings *UA*, *PA*, and *UPA* by Boolean matrixes **X**, **Y** and **A**, respectively.

Then the problem definition on formation of the common access control scheme RBAC will be as follows.

Initial data are:

(1) $\{RBAC_s\}$, $s = 1, \ldots, S$ – a set of local access control schemes;

(2) $\mathbf{A} = \bigcup \mathbf{A}_S$ – a common access control policy in the data space combining local policies.

It is needed to find the following matrices **X** and **Y**:

$$\mathbf{X} \otimes \mathbf{Y} = \mathbf{A} , \quad (6)$$

$$\left|ROLES\right| \Rightarrow \min . \quad (7)$$

At the same time, the following restrictions are considered:

(1) The combination of sets of users and the combination of sets of permissions

$$U = \bigcup U_S , \; PRMS = \bigcup PRMS_S ; \quad (8)$$

(2) The fairness of expression (2) in relation to each $RBAC_s$.

Let us explain this problem statement graphically on the example of combining two local information systems in the data space. Let one system have the access control scheme $RBAC_1$ shown in Fig. 1-a, and the second one – the scheme $RBAC_2$ shown in Fig. 1-b.

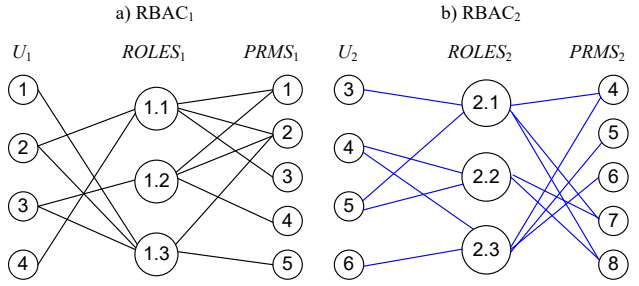

Figure 1: Access control schemes $RBAC_1$ and $RBAC_2$.

For convenience of further discussion it is necessary to consider that in the schemes $RBAC_1$ and $RBAC_2$ the cardinalities of the appropriate sets of users, powers and roles coincide. At the same time, the users u_3 and u_4 are included into the both sets U_1 and U_2, and the set $U = U_1 \cup U_2$ consists of six elements. Similarly, the permissions p_4 and p_5 are included into the both sets $PRMS_1$ and $PRMS_2$, and the set $PRMS = PRMS_1 \cup PRMS_2$ consists of eight elements. Sets of roles $ROLES_1$ and $ROLES_2$ have no common elements. Each of these sets consists of three elements.

For search of the matrixes **X** and **Y** according to (6), it is necessary to know the matrix **A**. The process of creating **A** is shown in Fig. 2. This matrix is created as $\mathbf{A} = \mathbf{A}_1 \cup \mathbf{A}_2$. The matrixes $\mathbf{A}_1$ and $\mathbf{A}_2$ are received according to (2) from the matrixes $\mathbf{X}_1$, $\mathbf{Y}_1$, $\mathbf{X}_2$, and $\mathbf{Y}_2$ which represent different mappings which are available in Fig. 1.

Knowing the matrix **A**, the solution of the problem of formation of the uniform access control scheme *RBAC* in the data space can be found by the following *two methods*:

(1) Solving the problem of the RBAC design according to criteria (6) and (7);

(2) Solving the problem of the RBAC reconfiguration, considering that initial values of matrixes $(\mathbf{X}_0, \mathbf{Y}_0)$, are received by joining $(\mathbf{X}_1, \mathbf{Y}_1)$ and $(\mathbf{X}_2, \mathbf{Y}_2)$.

The order of creating $(\mathbf{X}_0, \mathbf{Y}_0)$ is shown in Fig. 3.

Blue color shows the interconnections between sets of users, roles and permissions in the scheme $RBAC_2$ and their place in the scheme *RBAC*.

In case of *the first method* of solving the problem, the known methods, for example, genetic algorithms are used. The result contains the matrixes **X** and **Y** corresponding to criteria (6) and (7). This method in case of large dimensionality of the problem, that is characteristic for the data space, requires a big time. During solving the problem, the work of access control system is stopped. Therefore, it is necessary to apply this method in off-line mode.

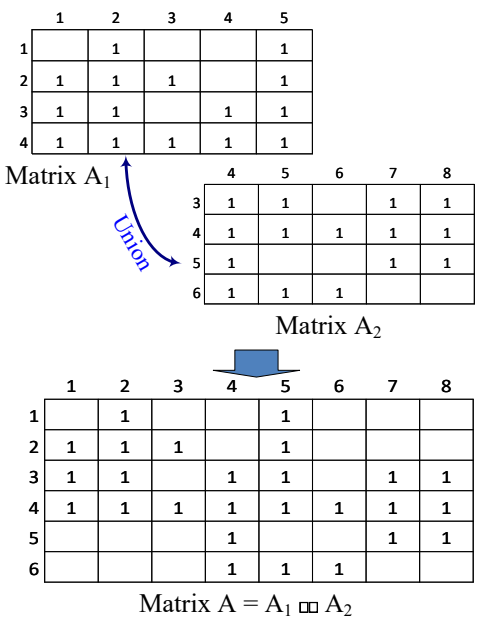

Figure 2: Creating the matrix A.

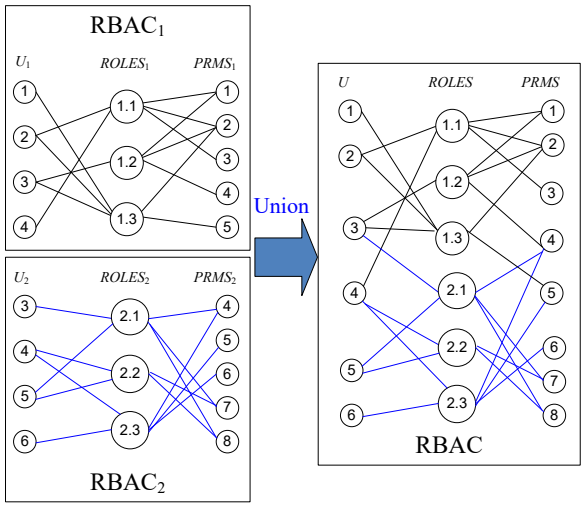

Figure 3: Creating the common access control scheme *RBAC*.

In case of *the second method* of solving the problem, a serial transition from the scheme $RBAC_0$, set by means of (**X**0, **Y**0), to the new scheme RBAC′, set by means of (**X′**, **Y′**), with the quantity of roles smaller on 1, is carried out.

At the same time, such solution is selected, that provides the minimum number of changes that should be made upon transition from ($\mathbf{X}_0$, $\mathbf{Y}_0$) to (**X′**, **Y′**). As a result, the second method does not require stopping the operation of the access control system. It can be executed in on-line mode.

The conditions of the transition from ($\mathbf{X}_0$, $\mathbf{Y}_0$) to (**X′**, **Y′**) are registered as follows:

(1) Each pair of matrixes ($\mathbf{X}_0$, $\mathbf{Y}_0$) and (**X′**, **Y′**) satisfies to expression (6).

(2) If to designate the quantity of roles in the scheme ($\mathbf{X}_0$, $\mathbf{Y}_0$) as $|ROLES_0|$ and the quantity of roles in the scheme (**X′**, **Y′**) as $|ROLES'|$ then the following equality shall take place:

$$|ROLES_0| - |ROLES'| = 1. \quad (9)$$

(3) The condition of the minimum changes in the RBAC scheme upon transition to the scheme (**X′**, **Y′**) is set by the following criterion:

$$\|\mathbf{X}_0 \oplus \mathbf{X}'\|_1 + \|\mathbf{Y}_0 \oplus \mathbf{Y}'\|_1 \to \min, \quad (10)$$

where the symbol $\oplus$ denotes "exclusive OR" and the operation $\|\mathbf{B}\|_1$ is the L_1-norm [7] that defines the number of nonzero elements in the matrix **B**.

The example of transition from the scheme ($\mathbf{X}_0$, $\mathbf{Y}_0$) to the scheme (**X′**, **Y′**) is given in Fig. 4.

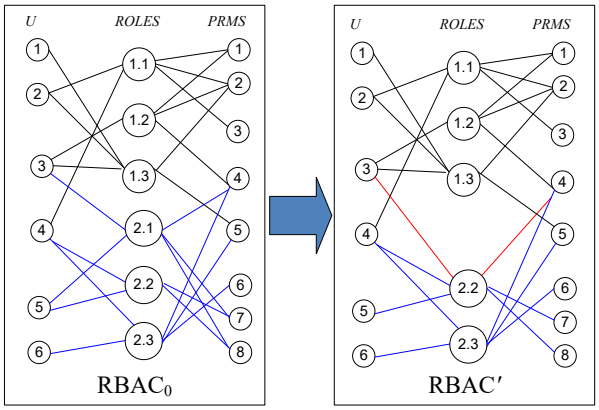

Figure 4: Transition to new scheme *RBAC′*.

Fig. 4 shows that in the new scheme *RBAC′* the role $r_{2.1}$ is removed from the set of roles *ROLES*. However, to conform to requirements of the expression (6), the links (u_3, $r_{2.2}$) and ($r_{2.2}$, p_4), which are highlighted in red color, are added to the scheme.

As a result, the condition (9) is satisfied, and the number of the changes, made in the scheme accordingly (10), is equal to seven (two added links and five links deleted together with role $r_{2.1}$). Therefore, owing to the small number of necessary changes, the transition to the scheme *RBAC′* does not require stopping the operation of the access control system and can be executed in on-line mode.

Now let us consider how genetic algorithms for solving the problem according to the first and second methods are created. For simplicity of further references to these algorithms, let us give to them the following designations: *GA-M1* is the genetic algorithm implementing the first method of solving the problem, *GA-M2* is the genetic algorithm implementing the second method of solving the problem.

4 GENETIC ALGORITHMS

4.1 Algorithm GA-M1

4.1.1 Structure of the GA-M1.

As the first method of creating the RBAC scheme for the data space consists in the use of genetic algorithms for searching the

solution of the Boolean Matrix Factorization problem written in the form of (6)-(7), the algorithm GA-M1 will correspond to the genetic algorithms considered in our earlier works devoted to their application for solving the RBAC design problem, for example, [29]. Therefore, we confine ourselves to a brief description of the algorithm GA-M1.

The algorithm GA-M1 has classical structure [25] and contains the following units:

(1) generation of initial population;

(2) execution of the current iteration containing the crossover, mutation and selection operations;

(3) algorithm termination.

The features of GA-M1 operation distinguishing it from the classical decisions are as follows:

(1) structures of chromosomes;

(2) fitness function;

(3) execution of the crossover, mutation and selection.

4.1.2 Chromosome structure.

For creating chromosomes in the algorithm GA-M1 the following enhancements are used:

(1) multi-chromosomal approach,

(2) use of matrix columns as genes,

(3) internal coding of genes in the form of numerical values.

In accordance with multi-chromosomal approach, each individual in the population of the algorithm has not one, but two chromosomes. One of these chromosomes Chr_X encodes the matrix **X**, and the second chromosome Chr_Y – the matrix **Y**. Such approach allows us to get rid of shortcomings, which will take place if the individuals have only one chromosome consisting of elements of both matrixes **X** and **Y**. These shortcomings are the low convergence of the algorithm and complexity of implementation of the crossover operation for the chromosomes having different length. Besides, among the new individuals who are turning out as a result of the crossover there is a very high percent of "spoilage" when new individuals are not suitable for RBAC in the physical sense and are excluded from further evolution. In case of the multi-chromosomal approach the percent of "spoilage" will be small.

Use of matrix columns as genes has two essential advantages.

First, it allows reducing and aligning the length of chromosomes in comparison with the case when as genes of chromosomes the 0-1 elements of Boolean matrixes are used. The length of the chromosomes Chr_X and Chr_Y will be identical and equal to the quantity of roles in the current RBAC scheme.

Secondly, it allows reducing the percent of "spoilage" in the crossover output if the individuals having different quantity of roles are used in the crossover.

Internal coding of genes in the form of numerical value is accepted for the purpose of simplification of program implementation of the algorithm. In computer memory the genes are stored not as columns, but in the form of decimal numbers. For example, if the column of the chromosome Chr_X is a column $(010100)^T$, then the decimal number 20 corresponds to it. Internal representation of genes in a decimal form is required for their storage. External representation in a form of matrix columns is required for computation of fitness functions and execution of the crossover and mutation operations.

Representation of genes of the chromosomes Chr_X and Chr_Y in internal and external formats for an example of the RBAC scheme shown in Fig. 3 is given in Fig. 5.

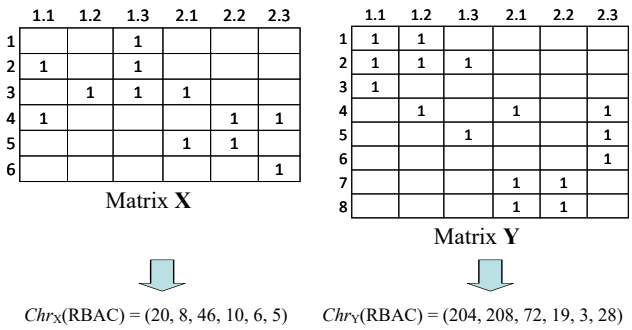

Figure 5: Representation of chromosomes Chr_X and Chr_Y.

4.1.3. Fitness function.

The following expression is used as the fitness function:

$$F_{GA-M1} = w_1 k + w_2 \sum_{i=1}^{n} \sum_{j=1}^{m} \left| a_{ij} - \sum_{l=1}^{k} x_{il} y_{lj} \right|, \qquad (11)$$

where k is a quality of roles in the RBAC scheme, which is reflected by the current individual of the algorithm; $\{a_{ij}\}$ are the elements of the given matrix **A**; $\{x_{il}\}$ are the elements of the matrix **X**; $\{y_{lj}\}$ are the elements of the matrix **Y**; w_1 and w_2 are the weight coefficients. Between the weighting coefficients the ratio $w_1 << w_2$ is set. This ensures that during the algorithm GA-M1 firstly the solutions meeting the condition (6) are searched, and then the solutions with smaller k values are generated.

4.1.4. Features of the crossover, mutation and selection.

The individuals from the current population for the crossover are selected according to the given probability of the crossover W_{cross} which is a parameter of the genetic algorithm. All individuals of the population with equal probability are exposed to the choice. It allows achieving the maximum separation of the directions of search for solutions and, in our judgment, promotes the increase in convergence of the algorithm.

In case of the crossover each of the chromosomes Chr_X and Chr_Y of the selected individuals-parents will be exposed to the division into two parts from an accidentally selected point of the crossover and to further exchange of these parts. All possible options of exchange are considered. As a result, four individuals-descendants will be on the output of the crossover, they are added to the current population.

If two individuals-parents with different quantity of roles are exposed to the crossover, then in this case the necessary quantity of empty roles is added to the RBAC scheme with smaller quantity of roles. The columns corresponding to empty roles are located in the end of the chromosomes Chr_X and Chr_Y. As a result, in case of crossing such individuals the individuals-descendants making a physical sense always turn out. The percent of "spoilage" is minimal.

The individuals for the mutation are selected according to the probability W_{mut} which is a parameter of the genetic algorithm. Two selection procedures are performed. The first procedure selects genes from the chromosomes Chr_X and Chr_Y with the probability W_{mut1}. These genes will be the subject to changes. The second procedure with the probability W_{mut2} selects the elements from the columns which values then are inverted. The probabilities W_{mut1} and W_{mut2} are the algorithm parameters as well.

The new individuals received as a result of the crossover or the mutation get the value of the fitness function calculated according to (11) and are added to the current population. However, before this procedure, the check of new chromosomes on uniqueness concerning the individuals already available in population is made. If this check was unsuccessful, then the new individual is excluded from further reviewing. The size of the population is constant for all iterations and is equal to N_{pop} that is the algorithm parameter. The individuals who are exceeding the quantity N_{pop} and having the greatest value of $F_{GA\text{-}M1}$ are deleted from an the algorithm.

4.1.5. Initial population and termination.

Initial population has the size N_{pop} and is generated in a random way in case of the fixed values of n and m. At the same time, the quantity of roles k changes accidentally in the range [1; n]. The genetic algorithm was terminated in case of achieving the given maximum number of iterations. This quantity depended on the dimensionality of the problem.

4.2 Алгоритм GA-M2

The algorithm GA-M2 provides finding the scheme *RBAC'* (as shown in Fig. 4) in which the quantity of roles is one less in comparison with the initial scheme. Therefore, the difference of the algorithm GA-M2 from the algorithm GA-M1 consists in the following two aspects: (1) function fitness; (2) all individuals in the population have the identical length of chromosomes Chr_X and Chr_Y which is equal to the quantity of roles in the initial scheme $RBAC_0$. Therefore, for the algorithm GA-M2 we will consider only the fitness function $F_{GA\text{-}M2}$. It is as follows:

$$F_{GA-M2} = w_1 F_1 + w_2 (k - k' + 1) + w_3 F_3 , \quad (12)$$

$$F_1 = \sum_{i=1}^{m} \sum_{l=1}^{K} \left| x0_{il} \oplus x'_{il} \right| + \sum_{j=1}^{n} \sum_{l=1}^{K} \left| y0_{jl} \oplus y'_{jl} \right| , \quad (13)$$

$$F_3 = \sum_{i=1}^{n} \sum_{j=1}^{m} \left| a_{ij} - \sum_{l=1}^{k} x'_{il} y'_{lj} \right| , \quad (14)$$

where $\{x0_{il}\}$ and $\{y0_{jl}\}$ are elements of the matrices $\mathbf{X}_0$ and $\mathbf{Y}_0$, respectively; $\{x'_{il}\}$ and $\{y'_{jl}\}$ are elements of the matrices **X'** and **Y'**, respectively; k' is a quality of roles in the scheme *RBAC'*; w_1, w_2 and w_3 are weight coefficients, between which the ratios $w_1 << w_2$ and $w_2 << w_3$ are set. This ensures that during the algorithm GA-M2 firstly the solutions that meet the condition (6) will be generated, then – the condition (9), and at last – the condition (10).

5 TESTBED AND EXPERIMENTS

5.1 Testbed

The testbed was developed for assessment of the developed genetic algorithms GA-M1 and GA-M2. C# was used for implementation. The structure of the testbed is given in Fig. 6.

The testbed consists of the following modules:

(1) **Initiator** in which the initial data and parameters of the genetic algorithms are entered. One of the elements of the initial data is S – a number of local systems in the data space.

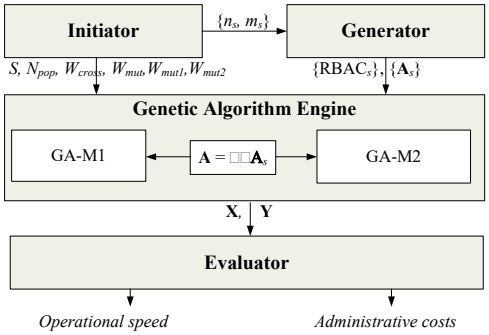

Figure 6: Test bed structure.

(2) **Generator** that generates local schemes $\{RBAC_s\}$, based on the values $\{n_s, m_s\}$ transferred from **Initiator**, and creates the local access control policies $\{\mathbf{A}_s\}$ for them.

(3) **Genetic Algorithm Engine** which implements the algorithms GA-M1 and GA-M2 allowing us to find the matrixes **X** and **Y** and the common access control scheme *RBAC* for the data space. In this module, the matrix **A** defining the common access control policy in the data space is created.

(4) **Evaluator** which assesses the developed algorithms on indices of operational speed and administrative costs.

The testbed has the following functions: (1) generating $\{RBAC_s\}$ and $\{\mathbf{A}_s\}$ for local systems; (2) finding the solution of the problem of the uniform RBAC scheme design according to the algorithm GA-M1; (3) finding the scheme *RBAC'* according to the algorithm GA-M2; (4) assessment of the efficiency of the algorithms GA-M1 and GA-M2.

The efficiency of the genetic algorithms is defined by two indices: operational speed and administrative costs. The first index is defined by the number of the iterations of the algorithm required to solve the problem. The second index is applied only to the GA-M2 assessment. It shows the number of necessary changes which should be made to pass to the new scheme *RBAC'*. It is calculated according to the expression (5).

5.2 Experimental results

5.2.1 Evaluation of the GA-M1 algorithm.

Evaluation of the operational speed of the genetic algorithm GA-M1 was carried out taking into account the change of the following parameters: (1) the quantity of the local information systems integrated in the data space, (2) the dimension of the problem, (3) coefficients of generality of users and permissions presented at the local RBAC schemes.

The quantity of the local information systems S during the experimental assessment of the genetic algorithms had the following values: 2, 3, 4, and 5. The bigger value of S was not considered, as we believed that S = 5 is rather great value to define regularities of change of the efficiency indices for the developed genetic algorithms. The number of users ms and the number of permissions n_s were equal to the appropriate values m and n of the selected dimensions in each of dimensions.

The coefficient of users and permissions generality π shows the proportion of users or permissions, which are common, i.e. are present also at other local information systems. In experiments, this index had the following values: π = 0.1; 0.2; 0.3. Great values were not considered as, on our opinion, in this case each user

(permission) in each local system will also be present at least at one other local system.

For each combination of the problem parameters, the experiments were conducted 10 times, and then we calculated the average value. At the same time the dispersion of values in statistical selection did not exceed 10 percent.

Results of operation speed evaluation for the algorithm GA-M1 are depicted in Fig. 7. The analysis of these data allows us to draw the following conclusions.

First, in case of equal values π (π = 0.1) the algorithm GA-M1 has the largest operational speed in case of the most small value $S = 2$, and the smallest speed – in case of the greatest value $S = 5$. It is caused by the fact that the higher the value S, the greater the quantity of users (permissions) in the access control scheme RBAC. The same dependence can be seen also in case the value S does not change ($S = 5$), but the value π changes. In case of bigger value π = 0.3, the operational speed of the algorithm becomes greater and vice versa. It is caused by the fact that the higher the generality coefficient π, the greater the number of common users (permissions) and, therefore, the lesser their total quantity in the data space.

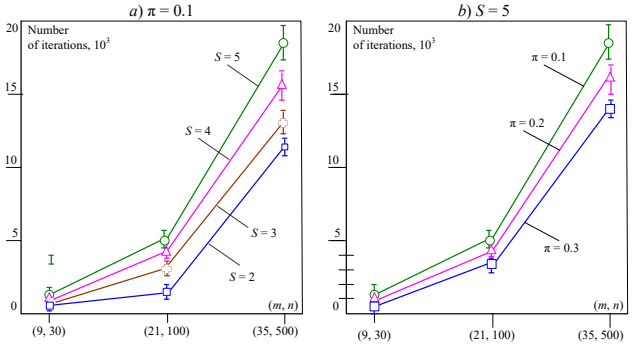

Figure 7: Results of the algorithm GA-M1 evaluation (*a* – π = 0.1, *S* is various; *b* – *S* = 5, π is various).

It is necessary to mark the nature of increase of operational speed of the algorithm in case of increase of π or reduction of S. If to determine the dimension not by couple of values (m, n), but by the value of $L = m \times n$, it is possible to note that for the algorithm GA-M1 the dependence of the number of iterations on the dimension is the almost linear. It means that the developed genetic algorithm is rather powerful method to generate the common access control scheme *RBAC* in the data space, which, as was stated above, belongs to the NP-complete class.

5.2.2 Evaluation of the GA-M2 algorithm.

The algorithm GA-M2 was evaluated on two indices: (1) the average quantity of iterations T_{av} required to find the new scheme *RBAC′*, having the quantity of roles, smaller on 1; (2) the average number of changes Δ_{av} in the scheme $RBAC_0$, needed to transit from $RBAC_0$ to *RBAC′*. Results of evaluation of efficiency of the algorithm GA-M2 are depicted in Fig. 8 and Fig. 9. The analysis of these data allows us to draw the following conclusions.

The operational speed of the algorithm GA-M2 also increases under the linear law in case of change of the problem dimension and the parameters S and π as well. It confirms the conclusion on high efficiency of the algorithm in case of solving the problem belonging to the NP-complete class.

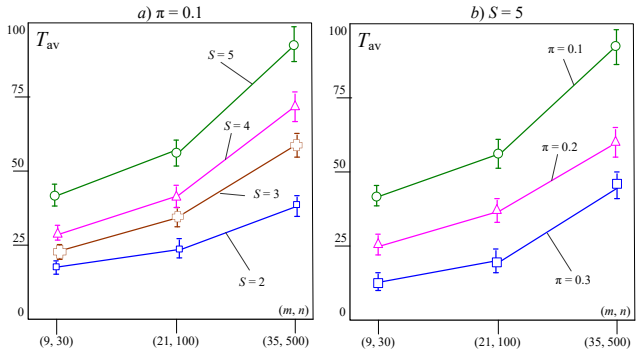

Figure 8: Results of the evaluation of the average number of iterations T_{av} for the algorithm GA-M2 (*a* – π = 0.1, *S* is various; *b* – *S* = 5, π is various).

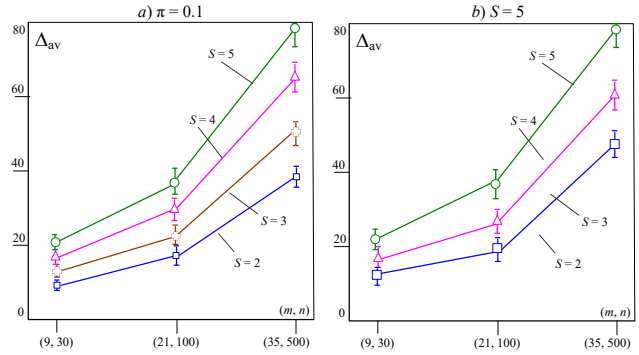

Figure 9: Results of the evaluation of the average number of changes Δ_{av} for the algorithm GA-M2 (*a* – π = 0.1, *S* is various; *b* – *S* = 5, π is various).

Comparing the number of the iterations required for the complete solution of the problem by means of the algorithm GA-M1 and finding the intermediate solution by means of the algorithm GA-M2 it is easy to note that in the second case much less iterations are required. For example, in case of $S = 5$, π = 0.1 and (m, n) = (35, 500), that is the most labor-consuming case among the considered variants, the algorithm GA-M1 requires about 20000 iterations, and the algorithm GA-M2 – no more than 100. It means that the security administrator can apply the algorithm GA-M2 in on-line mode.

Evaluating, at the same time, the number of changes needed to make with the initial access control to pass to the intermediate scheme RBAC′, it is possible to claim that this number is also not very high. Therefore, the security administrator can realize such changes in real time (in online mode). The access control system of the data space at the same time will not stop its operation.

Thus, the received experimental results on assessment of the overall efficiency of the developed genetic algorithms GA-M1 and GA-M2 showed their high operational speed and good scalability, i.e. ability to save the linear dependence of operational speed from dimensions in case of the big dimension of the problem. At the same time, the algorithm GA-M2 looks for the intermediate solution much quicker, than the algorithm GA-M1 looks for the final solution of the problem. In other words, it is necessary to use the algorithm GA-M1 off-line and the algorithm

GA-M2 – on-line. Therefore, the security administrator shall make a conscious choice of these algorithms for creating the uniform role-based access control policy in the data space and consider the received results of their experimental assessment.

6 CONCLUSIONS

The paper presents a new approach to solve the problem of cresting the uniform RBAC scheme in the data space of critical infrastructure, integrating several local information systems with local RBAC schemes. The sharpness of this problem is caused by two reasons. First, combining users and permissions of local systems in the data space significantly increases the dimension of the problem. Secondly, it is necessary to consider the restrictions peculiar to the data space. These restrictions concern the need of accounting the common users and permissions and preserving the local access control policies in the data space as well. Two types of the genetic algorithms are developed to solve the problem. The first algorithm (GA-M1) is oriented on solving the traditional RMP problem for which combining of local access control policies is an element of initial data. The second algorithm (GA-M2) is oriented on serial improving of the RBAC scheme with the minimum administrative costs. The experimental assessment of the algorithms, carried out on the testbed specially developed for these purposes, showed, firstly, their high efficiency and scalability. Secondly, it demonstrated that the algorithm SA-M1 should be used in off-line mode and the algorithm GA-M1 could be used on-line. Thereby the security administrator can make a reasonable choice of the necessary algorithm proceeding from the results of the experimental assessment.

The further research relates with transferring the offered genetic algorithms on new access control domains and investigation of an opportunity to use other varieties of the bio-inspired algorithms for solving the problems of design and reconfiguration of access control schemes. Besides, we plan to apply the developed genetic algorithms in the specific domain of critical infrastructure, for example, in smart city.

ACKNOWLEDGMENTS

This work was partially supported by grants of RFBR (projects No. 16-29-09482, 18-07-01369, 18-07-01488), by the budget (the project No. AAAA-A16-116033110102-5), and by Government of Russian Federation (Grant 08-08).

REFERENCES

[1] M.Franklin, A.Halevy, and D.Maier. 2005. From databases to dataspaces: a new abstraction for information management. *SIGMOD Rec.* 34, 4 (December 2005), 27-33.

[2] A.Halevy, M.Franklin, and D.Maier. 2006. Principles of dataspace systems. In *Proceedings of the ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems* (PODS '06). ACM, New York, NY, USA, 1-9.

[3] B.Mitra, S.Sural, J.Vaidya, and V.Atluri. 2016. A Survey of Role Mining. *ACM Comput. Surv.* 48, 4, Article 50 (February 2016), 37 pages.

[4] R.S.Sandhu, E.J.Coyne, H.L.Feinstein, and C.E.Youman. 1996. Role-Based Access Control Models. *Computer* 29, 2 (Feb. 1996), 38–47.

[5] M.Frank, J.M.Buhmann, and D.Basin. 2010. On the definition of role mining. In *Proceedings of the 15th ACM symposium on Access control models and technologies* (SACMAT '10). ACM, New York, NY, 35–44.

[6] G.Verma, V.Verma. 2012. Role and Applications of Genetic Algorithm in Data Mining. *Intern. Journal of Computer Applications*, 48, 17 (2012) 5–8.

[7] J.Vaidya, V.Atluri, and Q.Guo. 2007. The role mining problem: finding a minimal descriptive set of roles. In *Proceedings of the ACM symposium on Access control models and technologies* (SACMAT '07). ACM, New York, NY, 175-184.

[8] I.Saenko and I.Kotenko. 2012. Design and Performance Evaluation of Improved Genetic Algorithm for Role Mining Problem. In *Proceedings of the 2012 20th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP '12)*. IEEE Computer Society, Washington, DC, 269–274.

[9] I.Saenko and I.Kotenko. 2012. Design and Performance Evaluation of Improved Genetic Algorithm for Role Mining Problem. In *Proceedings of the 2012 20th Euromicro International Conference on Parallel, Distributed and Network-based Processing* (PDP '12). IEEE Computer Society, Washington, DC, 269–274.

[10] I.Kotenko and I.Saenko. 2015. Improved genetic algorithms for solving the optimisation tasks for design of access control schemes in computer networks. *Int. J. Bio-Inspired Comput.*7, 2 (May 2015), 98–110.

[11] I.Saenko and I.Kotenko. 2016. Using Genetic Algorithms for Design and Reconfiguration of RBAC Schemes. In *Proceedings of the 1st International Workshop on AI for Privacy and Security* (PrAISe '16). ACM, New York, NY, Article 4, 9 pages. DOI: https://doi.org/10.1145/2970030.2970033

[12] J.Vaidya, V.Atluri, and J.Warner. 2006. RoleMiner: mining roles using subset enumeration. In *Proceedings of the ACM conference on Computer and communications security (CCS '06)*. ACM, New York, NY, 144–153.

[13] Carlo Blundo and Stelvio Cimato. 2010. A simple role mining algorithm. In *Proceedings of the 2010 ACM Symposium on Applied Computing (SAC '10)*. ACM, New York, NY, USA, 1958–1962.

[14] C.Blundo and S.Cimato. 2010. A simple role mining algorithm. In *Proceedings of the 2010 ACM Symposium on Applied Computing (SAC '10)*. ACM, New York, NY, USA, 1958–1962.

[15] M.Frank, J.M.Buhman, and D.Basin. 2013. Role Mining with Probabilistic Models. *ACM Trans. Inf. Syst. Secur.* 15, 4, Article 15. 28 pages.

[16] M.Frank, A.P.Streich, D.Basin, and J.M.Buhmann. 2012. Multi-assignment clustering for boolean data. *J. Mach. Learn. Res.* 13, 1. 459–489.

[17] M.Kuhlmann, D.Shohat, and G.Schimpf. 2003. Role mining - revealing business roles for security administration using data mining technology. In *Proceedings of the eighth ACM symposium on Access control models and technologies (SACMAT '03)*. ACM, New York, NY, USA, 179-186.

[18] A.Colantonio, R.D.Pietro, and A.Ocello. 2008. A cost-driven approach to role engineering. In *Proceedings of the 2008 ACM symposium on Applied computing (SAC '08)*. ACM, New York, NY, 2129–2136.

[19] N.C.Narendra. 2003. Design of an Integrated Role-Based Access Control Infrastructure for Adaptive Workflow Systems. *Journal of Computing and Information Technology*, 11, 4, 2003, 293–308.

[20] S.Zafar, K.Winter, R.Colvin, and R.G.Dromey. 2006. Verification of an Integrated Role-Based Access Control Model. In *Proceedings of the 1st International Workshop - Asian Working Conference on Verified Software (AWCVS'06)*. 12 pages

[21] M.V.Tripunitara and B.Carbunar. 2009. Efficient access enforcement in distributed role-based access control (RBAC) deployments. In *Proceedings of the 14th ACM symposium on Access control models and technologies (SACMAT '09)*. ACM, New York, NY, USA, 155-164.

[22] P.Parikh, M.Kantarcioglu, V.Khadilkar, B.Thuraisingham, and L.Khan. 2012. In *Proceedings of the IEEE IRI 2012*, 659–663.

[23] H.V.Nguyen, K.Böhm, F.Becker, B.Goldman, G.Hinkel, and E.Müller. 2015. Identifying User Interests within the Data Space – a Case Study with SkyServer. In *Proceedings of the 18th International Conference on Extending Database Technology (EDBT'18)*. 641-652.

[24] N.Semmanche and S.Selka. 2008. Access control of Web services using genetic algorithms. In *Proceedings of the 2008 High Performance Computing & Simulation Conference* (HPCS'08), ECMS, Nicosia, Cyprus, 249–254

[25] N.Rai and K.Rai. Genetic Algorithm Based Intrusion Detection System. 2014. *International Journal of Computer Science and Information Technologies*, 5, 4 (2014), 4952–4957

[26] H.-S.Yang, M.Maier, M.Reisslein, and W.M.Carlyle. 2003. A Genetic Algorithm based Methodology for Optimizing Multi-Service Convergence in a Metro WDM Network. *Journal of Lightwave Technology*, 21, 5 (2003), 1114–1146.

[27] R.Mueller-Bady, R.Gad, M.Kappes, and I.Medina-Bulo. 2015. Using Genetic Algorithms for Deadline-Constrained Monitor Selection in Dynamic Computer Networks. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation* (GECCO'15), Sara Silva (Ed.). ACM, New York, NY, 867–874.

[28] V.Snasel, J.Platos and P.Kromer. 2008. On Genetic Algorithms for Boolean Matrix Factorization. In *Proceedings of the Eighth International Conference on Intelligent Systems Design and Applications* (ISDA'08), Vol. 2, IEEE Press, New York, 170–175.

[29] I.Saenko and I.Kotenko. 2017. Administrating role-based access control by genetic algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion* (GECCO'17). ACM, New York, NY, USA, 1463-1470.

[30] D.E.Goldberg. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Inc., Boston, MA, USA