L. Fritsch et al. (Eds.): Open Identity Summit 2017,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 153

A meta-heuristic for access control test data creation in access control model testing

Matthias Winterstetter[1], Sebastian Kurowski[2]

Abstract: User to Document Access data is in most cases protected and as such difficult to acquire for research purposes. This work seeks to circumvent this problem by creating research data on the basis of reference processes through the evolutionary Algorithm. Data created through this method, while not as accurate as real data, still has it's foundation in reality through the reference process and can as such be used as a replacement.

Keywords: evolutionary algorithm, access control, meta-heuristic, test data,

1 Introduction

Access Control mechanisms are an important countermeasure, to be able to control and mitigate risks associated with subjects accessing objects within an organization. The basic model of access control therefore involves subjects, that want to access an object and a reference monitor, which allows or prohibits the access based upon an access control model which is administered by the organization [Sa96]. Especially, the access control model has proven to have an impact on efforts and complexity, involved in administration [KW16], [OL10]. Additionally, different access control models have shown to not be applicable to different scenarios. For instance, [KW16] found that the widely used [OL10] role based access control (RBAC) models [Sa96] may not be applicable to access rights enforcement for documents in engineering, and [KB16] applied attribute based access control [YT05] models to an eHealth domain, rather than RBAC.

Selection and application of an access control model to a scenario, requires extensive knowledge of this scenario. At least, researchers and security engineers must know which subject is accessing which object. Usually, gathering this information is the most extensive part of designing the access control model. Naturally, multiple approaches have aimed at automating this process, e.g. as in the topic of role mining for RBAC models (see [Fr09], [Va07], [Ku03], [Mo09]). However, using these approaches, the respective access control models are limited to the applied case study, and comparative research is hardly achievable due to lack of data. Therefore, this contribution aims at providing an approach to provide access control test data that approximate the real scenario as close as possible, by estimating which subjects maybe accessing which

1 University of Stuttgart, IAT, Nobelstr. 12, Stuttgart, 70569, Matthias.Winterstetter@iat.fraunhofer.de
2 University of Stuttgart, IAT, Nobelstr. 12, Stuttgart, 70569, Sebastian.Kurowski@iat.fraunhofer.de

objects based on reference processes and standards.

Data indicating, which subjects access which objects cannot trivially be acquired in most cases, due to legal and operational constraints. For instance, national legislation may prohibit access logging as it enables performance monitoring. However, the largest constraint is due to the size and advantage of logging this information. Besides having very precise access control data, this information provides very little benefit, while requiring additional resources for logging, and large amounts of data storage. We have therefore in many cases found, that it is unlikely to retrieve this data from the organization, especially when it comes to more fine-grained applications of access control such as access control for document usages. However reference process, e.g. for automotive engineering, such as the ISO-Standard 10303-242 for the STEP data exchange format provide an abstract, agreed on reference process [IS14] of a scenario (in this case automotive engineering). Such reference processes may indicate which document types are used jointly, in which process parts.

This contribution builds upon this finding, and aims at providing an approach for generating data that indicates which subjects may access which objects in a scenario. The contribution uses an evolutionary algorithm (see Section 2.2), to provide a meta heuristic that builds bipartite subject-object graphs out of a graph that is obtained from the respective reference process (see Section 2.3). This contribution begins by introducing the basic concepts of the algorithm in Section 2.1.

2 Test Data Creation Approach

To create viable test data, it must be as similar to a real-world scenario as possible. We aim at achieving this realism, by using a reference process as foundation. We assume that joint usage of objects within a process, indicates that these objects are used together by at least one user.

2.1 From communication graphs to bipartite graphs

In order to display the joint usage of objects within the reference process with the resulting test data, we build a communication graph, as shown in [KW16] (See Fig. 1). In a communication graph, each node represents an object. Two nodes are connected, if the respective objects are jointly used within a process. The desired test data however, requires a bipartite graph which consists of subjects and objects (See Fig. 1). Therefore, the fitness function of the algorithm (see Section 2.2), constantly compares the difference between the joint usage of documents indicated in the communication graph, and the generated bipartite graph, in order to retrieve a result which matches the used reference process as close as possible.

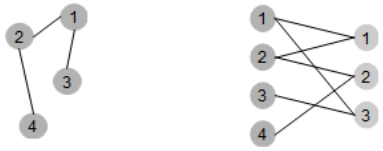

Fig. 1: Communication graph (left) and Bipartite graph (right)

2.2 Selection of the evolutionary algorithm

To generate test data, or in other words obtain a bipartite graph that follows the restrictions provided by the reference process that they are based on, a method is required to optimize the initial state of a bipartite graph, that can be generated randomly, in regard to the given restrictions. This method is a meta-heuristic.

Since there are many different kinds of meta-heuristics, the first problem to solve was choosing an appropriate meta-heuristic for the given task. For this sake, multiple meta-heuristic were considered, among which the following four could potentially be used for optimizing the bipartite graph.

The Cuckoo Search Algorithm which was used in [YD13] for designing architecture blueprints with multiple attributes. Parallel Simulated Annealing which was used in [Łu08] to solve the graph colouring problem. The Evolutionary Algorithm which was used in [Wi09] to solve scheduling Problems. Lastly the ASIA Algorithm which was used in [DN15] to optimize timetables for schools.

The final selection was depended on the similarity between the problems which would not only simplify the implementation but also decrease the potential problems with applying an algorithm on a different data set. It also increases worth of the solution, the more similar the structures of the problems are, since the same solution was already applied successfully on a similar problem.

The selected meta-heuristic was the evolutionary algorithm for it's similar tree-structure like data set. The similarities between the problems make it possible to use similar ideas for the mutation and recombination phase that can be applied to optimize the relations between the nodes of a solution.

2.3 Fitness Function of the Evolutionary Algorithm

Based on the evolutionary algorithm, an algorithm for the creation of test data, in form of bipartite graphs, can be defined. For this step, it is vital to determine a fitness function that allows the algorithm to judge how well each created solution upholds the provided restrictions. To that end we define the following two graphs.

Graph $K = (V,E)$ (1)

Graph *K* represents the communication graph, which consists of the set of nodes *V* and the set of edges *E* who's Elements connect the Elements of *V*.

$$\text{Graph B} = (U,O,C) \tag{2}$$

Graph *B* represents the test data, which consists of two sets of the nodes *U* and *O*, which stand for the Users and the Objects respectively, as well as the set of edges C whose elements connect one element from the set U with one element from the set O with each other. Furthermore, since the communication graph is the foundation for the created bipartite graphs, the cardinality |O| of graph B and |V| of graph K are equal and stand for the same set of objects. The following definition and equations are used to calculate the fitness value for each solution.

$$\forall\, o_j, o_i \in O \; and \; v_j, v_i \in V, 1 \le i \le |O|, 1 \le j \le |O| : \; (o_i, o_j, v_i, v_j) \tag{3}$$

The in (3) defined range of values will be used as indexes for the functions (4) and (5). With these indexes, the neighbourhood function *Ng* in (4) is used to determine if two objects which are connected in graph *K* are used together by at least one user in graph *B* or conversely, if two objects are not connected in graph *K* and not used together in graph *B*. If these conditions are fulfilled the fitness will increase.

$$(o_i, o_j, v_i, v_j) = \begin{pmatrix} 1, \text{if } N_g(o_i) \cap N_g(o_j) = \varnothing \text{ and } v_j \notin N_g(v_i) \\ 1, \text{if } N_g(o_i) \cap N_g(o_j) \neq \varnothing \text{ and } v_j \in N_g(v_i) \\ 0, \text{else} \end{pmatrix} \tag{4}$$

The in function in (5) will repeat this process for every possible combination of objects.

$$\text{Fitness} F = \sum_{i=1}^{|O|-1} \sum_{j=i+1}^{|O|} (o_i, o_j, v_i, v_j) \tag{5}$$

The resulting value F represents the fitness of the graph. If the value is equal to *(|V|-1)!* than graph B upholds all restrictions provided by the reference process.

2.4 Algorithm for Access Control Test Data Generation

The evolutionary algorithm itself can be distributed into three different phases, which are selection, recombination and mutation. In selection, the created individual solutions are being evaluated according to the given fitness function for the algorithm. Followed by that a part of the solution set is removed.

$$\text{Survivability} = ((\text{F} - \text{GenMin})/(\text{GenMax}/\text{GenMin})) * 100 \tag{6}$$

To that end, the probability of survival for each solution can be calculated through (6) where *GenMin* and *GenMax* represent the highest and lowest fitness value for the current

generation respectively. In other words, the higher the relative fitness the higher the chance to "survive". The recombination phase seeks to refill the solution set after the selection phase. To achieve this, two solutions are selected randomly but under consideration of their fitness value, with priority given to better solutions. From these solutions, a new solution will be created by randomly copying individual document nodes together with their connected edges, from both parents, until the new solution has all the required document nodes. As a result, the new solution consists of document nodes and their relations from both parents. This process will be repeated until the solution set is refilled.

The new solutions will undergo slight changes or mutations in the following mutation phase. These changes randomly create or remove a relationship between a user and a document node in the solution. How often these changes should happen for each solution should be determined on a case by case basis.

As shown in Fig. 2, these phases will be repeated in the order they were described until the given requirements are met. The relationships between the nodes of the Initial solution set can be created randomly using the number of objects in the reference process.

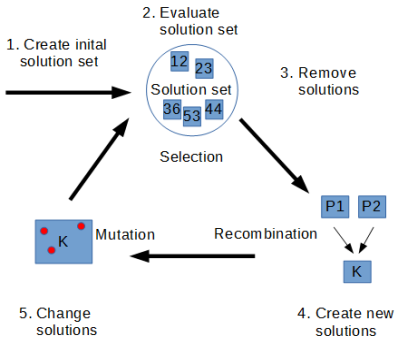

Fig. 2: Essential steps for the implemented evolutionary algorithm

3 Conclusion and Future Work

This work introduces a method that enables the creation of test data, in form of user-document relations/requirements, for the use-case that was presented in the Introduction. To achieve this goal, the algorithm estimates the required access control through an evolutionary algorithm and a foundation in form of the restrictions provided by a reference process. Future research will include an evaluation of this method, e.g. by using different reference processes. The test data created with this method will be further used to identify formal statements on access control requirements. The overall goal of the research, is to formalize access control requirements, enabling the selection of the appropriate access control model beyond an educated guess.

4 Bibliography

[DN15] Das, D.; Natarajan, S.: A Structured Timing Itinerary Using an Augmented Swarm Intelligent Algorithm. IEEE, 1221-1228, 2015.

[Fr09] Frank, M. et al.: A probabilistic approach to hybrid role mining. In: Proceedings of the 16th ACM conf. on Comp. and comm. security. ACM, pp 101-111, 2009

[IS14] ISO: Industrial automation systems and integration - Product data representation and exchange - Part 242: Application protocol: Managed model-based 3D engineering. Geneva. 2014.

[KB16] Kuhlisch, R.; Bittings, S.: Aligning ABAC Policies with Information Security Policies using Controlled Vocabulary. In Open Identity Summit 2016, 181-191, 2016

[Ku03] Kuhlmann, M.; Shohat, D.; Schimpf, G.: Role mining-revealing buisness roles for security administration using data mining technology. In: Proc. of the eighth ACM symposium on Acc. contr. mod. and tech. ACM, 179-186, 2003

[KW16] Kurowski, S.; Wehrenber, I.: Enterprise Rights Management Integration in Engineering. In Prostep iVip Recomm., Darmstadt, 2016.

[Łu08] Łukasik, S.; Kokosiński, Z; Świętoń, G.: Parallel Processing and Applied Mathematics Parallel, Simulated Annealing Algorithm for Graph Coloring Problem, Springer Berlin Heidelberg, Berlin, 2008.

[Mo09] Molloy, I. et al.: Evaluating Role Mining Algorithms. In: Proceedings of the 14th ACM Symp. on Acc. Contr. Mod. and Techn. ACM, New York 95-104, 2009.

[OL10] O'Connor, A.C.; Loomis, R.J.: 2010 Economic Analysis of Role-Based Access Control, Gaithersburg, 2010

[Sa96] Sandhu, R.S. et al.: Role Based Access Control Models.In : IEEE Comp. 29/1996,38-47, 1996

[SP94] Sandhu, R.S.; Samarati, P.: Comm. Mag. Access control: principle and practice. 32/1994, 40-48, 1994

[Va07] Vaidya, J.; Atluri, V; Guo, Q.: The role mining problem: finding a minimal descriptive set of roles. In: Proc. of the 12th ACM symp. on Acc. contr. mod. and techn. ACM,175-184,2007.

[Wi09] Wilfried, J. et al.: Schnelles Scheduling mit Hilfe eines hybriden Evolutionären AlgorithmusFast Scheduling by Means of a Hybrid Evolutionary Algorithm. at Automatisierungstechnik 57/2009, 01/2009

[YD13] Yang, X.; Deb, S.: Multiobjective cuckoo search for design optimization. Comp.& Op. Res. 40/2013, 1616-1624, 6/2013.

[YT05] Yuan, E.; Tong, J.: ICWS'05. Attribute based access control (ABAC) for Web services. In: Proc. of IEEE Int. Conf. on Web Serv., 2005