Attacks Scenarios in a Correlated Anomalies Context: Case of Medical System Database Application

Pierrette Annie Evina[1], Faouzi Jaidi[1,2], Faten Labbene Ayachi[1] and Adel Bouhoula[3]
[1]*University of Carthage, Higher School of Communication of Tunis (Sup'Com), LR18TIC01 Digital Security Research Lab, Tunis, Tunisia*
[2]*University of Carthage, National School of Engineers of Carthage, Tunis, Tunisia*
[3]*Arabian Gulf University Department of Next-Generation Computing, College of Graduate Studies, Kingdom of Bahrain*

Keywords: Anomaly Detection, Vulnerability Mask, Access Control, Databases Security.

Abstract: In Information Systems (IS) and specifically in databases, both internal and external attacks require a lot of attention. Due to inadequate manipulations in these systems, the access control policy (ACP) which is designed to control and protect resources from non-authorized users, may be subject to diverse alterations in its expression with significant anomalies. In the present paper, we study and establish basic scenarios that are encountered in such circumstances. We discuss other advanced scenarios based on correlation cases between basic ones. We mainly consider three basic concepts: Hidden User, Corrupted User and ACP vulnerability. Our contribution consists in the definition of a vulnerability mask, which makes it possible to calculate all the critical objects and to classify malicious users. This allows fine and reliable configuration of the risk management systems and the audit system as well as an objective and optimized analysis of log files and audit data. We present the architecture of our approach for the detection of anomalies in a correlated risk management context. Our contribution specifically considers groups of anomalies for which occurrences are linked both temporally and spatially.

1 INTRODUCTION

Databases (DB), as central nodes in Information Systems (IS), are particularly concerned by security measures. So, several procedures and actions can be taken or initiated for that purpose and the access control policy makes it possible to filter or control users in systems and installations.

According to *"insider threats report of 2019"*, a significant majority of organizations (60%) have experienced one or more insider attacks within the last 12 months. The term "*Insider Threat*" is often associated to employees with malicious intentions to directly harm the company through theft or sabotage. In truth, negligent employees or contractors can unintentionally pose an equally high risk of security breaches and leaks by accident. But insider threat solutions should detect and address all insider threats, regardless of the underlying motivation or cause (Insider threats report, 2019). So, by monitoring the database activity, intrusions and malicious users are regularly detected leading to a better an adequate risks assessment.

Regarding the implementation of the access control policy, we handle this problem for a relational database system. We scrutinize the access control policy and we address the attacks scenarios that appear when the access control policy evolves over time. Using log files, we develop a message filtering mechanism to retrieve the necessary messages. We consider different intrusion scenarios in order to establish a comparison between them. From this comparison, an assessment of the risk factor of each object in the system follows according to the intrusion scenarios in which they are involved.

Our contribution highlights an attack scenario construction that takes into consideration the correlations, as described in (Evina et al., 2018), between anomalies that occur in an access control policy during its evolution. It consists in defining a vulnerability mask which makes it possible to calculate all the critical objects and to classify malicious users for a fine and reliable configuration of the risk management systems and the audit system. We propose an approach which goes deeply in the expression of a security policy, considers its

Evina, P., Jaidi, F., Ayachi, F. and Bouhoula, A.
Attacks Scenarios in a Correlated Anomalies Context: Case of Medical System Database Application.
DOI: 10.5220/0010475303480355
In *Proceedings of the 16th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2021)*, pages 348-355
ISBN: 978-989-758-508-1

evolution and formally identifies the different classes of anomalies in the expression of the policy because we believe that the cohabitation of several anomalies can initiate more complex attack scenarios. The architecture of the anomalies detection approach in a correlated risk management context is also presented in the present paper.

The remainder of this paper is structured as follow: in section 2, we present the state-of-the-art. In section 3, we present our detection approach for the specific case of inconsistency anomalies and partial implementation anomalies. In section 4, we discuss our solution and present some perspectives. In section 5 we conclude and give an overview of the work in progress.

2 THE STATE-OF-THE- ART

The existing solutions in the insider threat field can be categorized according to the strategy for threat detection into signature-based solutions, rule-based solutions and user behavior analytics. The signature-based technique concerns the misuse detection. It has a predefined repository that contains the set of patterns that describe the different misuse scenarios. This technique fails to account for unknown threats. The rule-based technique relies on a set of rules for detecting intrusion scenarios. The user behavior analytics is a technique which studies the user behavior in order to detect potential threats. These techniques differ from each another by used algorithms in each approach. Anyway, various works exist in each particular field.

For intrusion detection (ID) in relational database management system (RDBMS), the proposed approach in (Senthil et al., 2013) defines an ID mechanism that consists of two main elements tailored for RDBMS: an anomaly detection system (ADS) and an anomaly response system (ARS). In the ADS, the construction of database access profiles of role and users and the use of such profiles for the AD tasks are concerned. Alongside their paper, the authors describe the response component of their intrusion detection system for a DBMS that response to an anomalous user request.

Considering malicious insiders, authors in (Khan et al., 2018) take a sequence of queries rather than one SQL query in isolation and a model behavior to detect malicious RDBMS accesses using frequent and rare item sets mining. They consider their approach as an alternative to the conventional anomaly-based detection approach because auditing log for data mining needs are not anomalies free and can already contain possible anomalies. They extend their approach with the conventional anomaly-based detection approach in order to detect the mimicry attacks or frequent attacks query pattern.

In (Ramachandran et al., 2018), authors propose a novel method of anomaly detection in "*role-administrated relational database*". They produce a mechanism for finding the anomalies in RBAC policies by using machine learning technique such as classification using a support vector machine (SVM) classifier. The detection is made through three phases: the profile creation; the training phase; and the intrusion detection phase.

In (Sallam et al., 2016), authors propose to detect anomalies in user access by learning profiles of normal access patterns in different database management systems. Database exfiltration attempt from insiders is particularly concerned. They make a classification of detected anomalies by using a naive Bayesian and the multi-labeling methods. The related architecture is presented in the paper. An internal representation of the queries is also presented followed by the description of the use of classification and clustering to detect anomalies.

In "Detection of Temporal Insider Threats to Relational Database", Sallam et al. propose techniques for detecting anomalous accesses in relational databases, that are able to track users actions across time. In order to detect correlated ones that collectively flag anomalies, they deal with queries that retrieve amounts of data larger than normal (Sallam et al., 2017).

Although anomalies detection is an effective technique for flagging early signs of insider attacks, modern techniques for the detection of anomalies in databases are not able to detect several sophisticated data updates and aggregation of data by insider that exceeds his or her need to perform job functions (Sallam et al., 2019). In their paper, the authors propose an anomaly detection technique designed to detect data aggregation and attempt to track data updates. Their technique captures the normal data access rates from past logs of user activity during a training phase (Sallam et al., 2019), then they build profiles for DB tables and tuples. This technique operates in two phases: training and detection.

Authors in (Grushka-Cohen, 2019) present Data Activity Monitoring Systems (DAMS) that are commonly used by organizations to protect the organizational data, knowledge and intellectual properties. A DAMS has two roles: monitoring (documenting activities) and alerting anomalous activities. Generally, such systems are just using sample of activity due to the high amount of data.

They redefine the sampling problem as a special case of multi-armed bandits (MAB) problem. Their algorithm explores randomly and uses expert knowledge to analyze the effect of diversity on coverage and downstream event detection tasks using simulated datasets. DAMS are used to help implementing security policies and detecting attacks and data abuse. The authors suggest the incorporation of the concept of diversity into logging policies. They use MABs as a strategy for policy setting based on sampling for decision-making, to set data collection policy in their anomaly detection system. In this work, authors suggest the viewing of the diverse problems for DAMS sampling strategies as a MAB problem, where the risk of the transactions logged is used as the reward function.

Studies on attack scenarios are usually completed and reinforced by the assessment of the risk of occurrence of anomalies in access control policies. Several authors have looked into that issue.

The authors in (Atlam et al., 2020) present traditional Access Control (ACL, DAC, MAC, and RBAC are the common and popular approaches or examples of traditional access control) and Dynamic Access Control (Risk-based access control, trust-based access control, and combination of risk with trust are common examples of dynamic access control). They recall that dynamic access control adaptive to unpredicted situations and conditions that policies could not expect. Resolving risks and threats in real time, especially when handling a previously unidentified threat is important. So, they provide a systematic review and examination of the state-of-the-art of the risk-based access control models.

A synthetic study of risk-based access control approaches was also presented in (Evina et al., 2020). In order to compare the different approaches encountered in the literature, the specificities of each approach have been highlighted through some defined criteria.

The authors of (Cao et al., 2020) propose an effective access control framework and risk assessment approach for policy enforcement to assess user's behaviors. The user's risk value is calculated based on their historical behavior and the current access request.

The authors of (Costante et al., 2013) for example, have developed a machine learning system that automatically acquires knowledge related to normal user behavior during database manipulation. Their system compares the user's SQL requests exchanged with the database server and also evaluates the sensitivity of the data manipulated in order to avoid data leaks in the database. In (Darwish, 2016), the author proposes to detect anomalies using the correlation between queries in DBMS transactions with log records.

In the context of IoT, authors in (Cramer et al., 2018) propose an approach to detect anomalous behavior of devices by analyzing event data. In fact, data is analyzed to detect automatically unusual behavior patterns. The paper is concerned with transaction caused by devices with servers. They develop a general purpose analysis template for the detection of anomalies through feature generation, data aggregation and data analysis.

Literature provides very little work that deals with technical problems related to an unauthorized modification in the expression of ACP. Our approach deals with the formal identification of different classes of anomalies in the expression of the policy because we believe that the cohabitation of several anomalies can initiate more advanced attack scenarios. Our intention is to define and characterize a risk management system based on the investigation of anomalies taken individually and the investigation of possible correlations between certain anomalies.

3 THE DETECTION APPROACH

3.1 Principle and Anomalies

As previously explained, an ACP is subject to delinquency or misconduct by malicious users, when in use.

So, let (U, P, R, AUR, APR, ARR) be a state of the ACP denoted ACP_0 assumed to be a reference state because it is valid at an instant T_0 and such that:

- U: all users with authentication parameters.
- R: all valid roles.
- P: the set of valid permissions.
- AUR: all valid assignments of roles to users.
- ARR: the valid role hierarchy.
- APR: all valid assignments of permissions to roles.

Over time, the access control policy has undergone various changes.

Let ACP' = (U', P', R ', AUR', APR', ARR') be a state of the ACP at a time T' assumed to be the current state and such that:

- U': all defined users.
- R': all roles.
- P': all permissions.
- AUR': all roles assignments to users.

• ARR': all roles assignments to roles.

• APR': the set of permissions assignments to roles.

The observed deviations in the expression of ACP' relative to ACP_0 are referred to as nonconformity anomalies and are of four types: inconsistency anomalies, redundancy anomalies, contradiction anomalies and partial implementation anomalies (Jaidi et al., 2019). We are interested in the last two types of anomalies because the value of the risk factor that these anomalies generate is from *MODERATE* to *HIGH*.

As for inconsistency anomalies, one of the following conditions is verified:

- New users, permissions and / or roles not defined in ACP_0 are defined in ACP'.

- New assignments (assignments of permissions to roles and / or roles to users) not valid in ACP_0 are defined in ACP'.

- The role hierarchy in ACP' covers the valid role hierarchy in ACP_0.

The formal calculation of the gap between the two policies in this case makes it possible to highlight the following components:

• Set of hidden users noted HU so that:

HU = U' - U.

• Set of hidden permissions noted HP so that:

HP = P' - P.

• Set of hidden roles noted HR and so that:

HR = R' - R.

• Set of hidden assignments of permissions to roles noted HAPR and so that:

HAPR = APR'- APR.

• Set of hidden role assignments to roles denoted HARR and so that:

HARR = ARR' - ARR.

• Set of hidden role assignments to users denoted HAUR and so that:

HAUR = ARU' - ARU.

The inference system which makes it possible to calculate each of these set respectively is correct and complete. The HAUR set can be developed to distinguish the following assignments:

- Assignment of hidden roles to valid users.
- Assignment of valid roles to hidden users.
- Assignment of hidden roles to hidden users.

We therefore introduce the additional and invalid pseudo-politics (HU, HP, HR, HAPR, HARR, HAUR).

Otherwise, partial implementation anomalies occur whenever we notice the absence of one or more components of the ACP' compared to the components of the same category present in ACP_0. The following conditions are verified:

- Users previously defined in the ACP_0 (the specification) but not referenced in ACP '.

- Roles initially identified in ACP_0 but not defined in the implemented policy.

- Missing segments in the role hierarchy.

- Missing assignments of valid permissions to valid roles.

- Missing assignments of valid roles to valid users.

The analysis of the difference calculated between ACP_0 and ACP' makes it possible to highlight the following new components:

• Set of missing users. This set is denoted MU such that:

MU = U - U'

• Set of missing roles noted MR such as:

MR = R - R'

• Set of missing assignments of permissions to roles noted MAPR and such as:

MAPR = APR - APR'

The pseudo-policy (MU, MP, MR, MAPR, MARR, MAUR) is normally valid but not deployed in ACP".

Table 1 summarizes the different cases of anomalies in the policy.

Table 1: Current implemented policy.

Valid ACP			
Initial and complete ACP	Pseudo-Policy ACP" (with missed elements)	Invalid additional Pseudo-Policy ACP' (not valid)	ACP' (current or implemented)
U	MU	HU	U ∪ HU - MU
P	MP	HP	P ∪ HP - MP
R	MR	HR	R ∪ HR - MR
APR	MAPR	HAPR	APR ∪ APR - APR
ARR	MARR	HARR	ARR ∪ HARR - MARR
ARU	MARU	HARU	ARU ∪ HARU - MARU

3.2 Masks and Vulnerabilities

When a permission is mentioned in the access control policy, it defines an action authorized by the security administrator on a given object of the database which is then said to be visible object.

Let (U, P, R, AUR, APR, ARR) be the reference access control policy and let perm ϵ P. perm is a valid permission and corresponds to an authorization

to perform an action a on an object o of the database. We denote it by perm = (a, o).

Now, we consider (HU, HP, HR, HAUR, HAPR, HARR) the hidden and invalid access control policy section and perm' ϵ HP.

We analyze the pair (user, permission), also noted (user, (action, object)), when it is related to a hidden permission. At this stage we talk about a vulnerability because the illegal granting of privileges to users is a security breach and an uncontrolled opening to the outside.

We identify, in table 2, the following masks to discuss possible vulnerabilities.

Table 2: Identification of masks.

Class	***Vulnerability Mask***	***Designation***
(1)	**(u, (ă, o))**	Authorized user **u** who is granted an illegal action **ă** on a visible object **o**.
(2)	**(u, (..., õ))**	Authorized user **u** granted unauthorized permission **(..., õ)**.
(3)	**(ũ, (ă, o))**	Unauthorized user **ũ** to whom an illegal action **ă** has been assigned on a visible object **o**.
(4)	**(ũ, (..., õ))**	Unauthorized user **ũ** with an unauthorized permission **(..., õ)**.
(5)	**(ũ, (a, o))**	Unauthorized user **ũ** who is assigned a valid access permission **(a, o)**.

The security mechanisms deployed in the database servers record only the actions performed by users of the database objects. Assuming that these vulnerabilities have been exploited by users, it is appropriate to investigate the log files. The identified masks make our task easier because they allow us to calculate the filters for browsing these files.

Vulnerability masks highlight two types of users to watch out for:

• Users known to the access control system but considered corrupt or "*insiders*" be-cause they are capable of performing illegal actions on the database.

• Unauthorized users or "*intruders*", initially not specified in the reference ACP and are therefore usurpers.

These two sets of users are calculated by the following algorithm (***Algorithm 1***), where leaf (r) is a function that returns all the leaves of the root tree r.

Algorithm 1: Insiders and Intruders Computing.

Step 1 :

Calculate the set of roles that can be reached from hidden permissions

FirstLevelRoles =

$\{r \in R \cup HR \,/\, \exists\ perm' \in HP \wedge (perm', r) \in APR \cup HAPR\}$

Step 2 :

For each element in FirstLevelRoles find all the terminal roles (leaves) associated with it by transitivity.

LeafRoles = $\{r \in R \cup HR \,/\, r \in leaf\ (r) \wedge r \in FirstLevelRoles\}$

Step 3:

Calculate the two sets

Insiders = $\{u \in U \,/\, \exists (u, r) \in ARR \cup HARR \wedge r \in LeafRoles\}$

Intruders = $\{u \in HU \,/\, \exists (u, r) \in ARU \cup HARU \wedge r \in LeafRoles\}$

For simplificaion reasons, we consider the following sets represented in figure 1, that materialize the five categories of users.

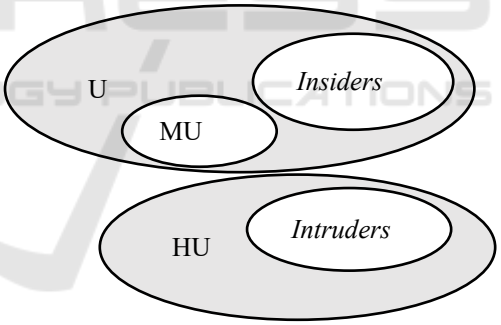

Figure 1: Users categories.

Wecan now easly check that:

• Insiders ⊂ U

• Intruders ⊂ HU.

• In addition to being hidden users, thieves have illegal permissions on the database.

• Class (5) highlights hidden users which belongs to the (HU – Intruders) set.

3.3 Algorithms

Taking into account the coexistence of anomalies as well as the correlation between these anomalies, we defined "compositional abnormalities".

Compositional anomalies are the reason of our scientific contribution in our research activities. They refer to any anomaly, resulting from the association of one or more basic non-compliance anomalies.

Portions of detection codes are given below.

```
if(clinique.webapp.core.exceptions.
AnomalyInterceptor(userAction)    !=
null){
       anomaly                          =
clinique.webapp.core.exceptions.Ano
malyInterceptor(userAction).get();

if(anomaly.getRevokedUsers().contai
ns(user)  AND  anomaly.getRoles()==
null){
        if(tabUsers.contains(user)
AND ! tabUsersImp.contains(user)){
          anomaly.setName("MISSED
USER ANOMALY");
                }
       else
if(!tabUsers.contains(user)      AND
tabUsersImp.contains(user)){
         anomaly.setName("HIDDEN
USER ANOMALY");
               }
    }
```

The first one shows the procedure of basics anomalies detection, and is expressed as follows for some basic anomalies.

The following code shows the procedure of detection of compositional anomalies.

We implemented our approach and highlighted the relevance of our solution based on a review of obtained results, from a real world context of a medical system application called "Santé Plus App" used by the medical center "Clinique Santé Plus" for managing employees, patients, services, etc.

Figure 2, as an example, is a screenshot showing the identification of a hidden user. Much information is retrieved such as the number of such anomaly, its name, its type, its description, the date of occurrence, the intrusion scenario, etc. In the next section, we illustrate how we evaluate the risk associated to such anomaly and even the correlated risk when another anomaly appears concomitantly.
Our attacks scenarios module is part of our Correlated Risk Management System deployed in our "Santé Plus App".

```
if(clinique.webapp.core.exceptions.A
nomalyInterceptor(userAction)      !=
null){
         initialize(anomaly) ;
    if(anomaly.getAnomalies().len
gth() >= 2){
    anomaly.setType(   'Compositional
anomaly) ;
    }
    Var tabBasicAnomalies :[1…] list
of basic anomalies;
    Var   tabTransactionAnomalies   :
[1…] list of transaction anomalies ;
     Var i,j :entier ;
    i ← 1 ;j←1 ;
    do{
    tabBasicAnomalies[i]            ←
Basic_Anomaly_detection.run();
    i← i+1;
    }while(Basic_anomaly_detection.r
un()==null) ;
    do{
    tabTransactionAnomalies[j]      ←
detection_anomaly_trans.algo.run();
    j← j+1;
    }while(detection_anomaly_trans.a
lgo.run()==null) ;
    anomaly                         ←
clinique.webapp.core.AbstractFacade.
compose(tabBasicAnomalies,
tabTransactionAnomalies) ;
    anomaly.setName("Compositional
anomaly ");
    }
```

4 THE RISK ASSESSMENT APPROACH

In our risk assessment approach, we consider the residual risk which is defined here as the difference between the risk of a compositional anomaly and the resultant risk of anomalies when occurring alone.

We think that the mentioned residual risk is not insignificant when it is mastered and therefore, it makes it possible to minimize the impact of anomalies on the integrity of information system resources. Consequently, we assess the overall risk of anomalies during the evolution of the expression of the ACP by taking into account on the one hand the risks of appearance of the various anomalies taken one by one and on the other hand the residual risk resulting from the cohabitation of anomalies.

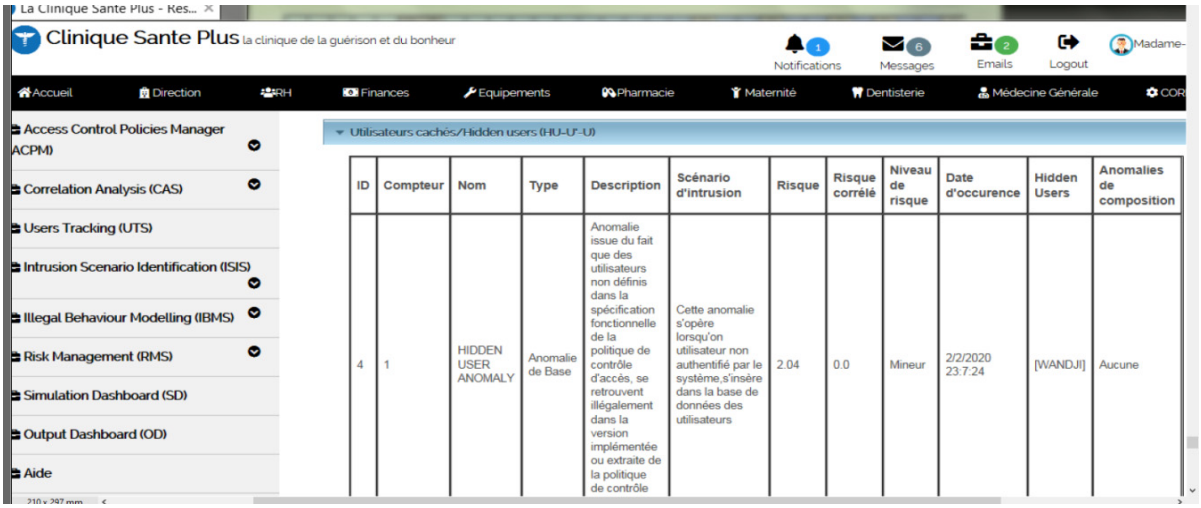

Figure 2: Identification and risk assessment of hidden users.

In figure 2 above, the risk of a particular anomaly has been evaluated. Particularly here, is the value of the hidden user anomaly.

The risk associated with the presence of anomalies in the access control policy at a given time is given by the formula (1) where Ri (An) represents the risk associated with each Anomaly and Ri (Att) represents the risk of an attribute defined in the ACP and preserved in the ACP'.

$$R = \frac{\sum_{i=1}^{N} Ri(An)}{\sum_{i=1}^{N} Ri(Att_{PCA \cap PCA'})} \times 100 \quad (1)$$

The correlation risk r, computed according to formula (2), is such that:

$$r = \left| R - \frac{\sum_{i=1}^{N} R_i}{N} \right| \quad (2)$$

Where:

- R is the value of the risk associated with the presence of anomalies in the access control policy at a given time
- Ri corresponds to the value of risk associated with each Anomaly
- N represents the number of defections in ACP.
- | | is an operator denoting the absolute value. |x| denoting the absolute value of the quantity x.

5 DISCUSSIONS AND PERSPECTIVES

The study allows us to highlight critical objects and thus determine the incident users with a given authorization. Thus, we can evaluate the security impact associated while assigning permission to a given user. To do so, a defined vulnerability mask undoubtedly allows us to prevent incidents or events. Thanks to the vulnerability mask, the exploration of log-files is a little easier and the detection of anomalies is technically less difficult.

From a risk management perspective, vulnerability removal involves disabling roles associated with invalid permissions and updating the access control policy by removing invalid assignments. It will also be a matter of closing the gates by adjusting the audit parameters to monitor and record the future activity of corrupt users (insiders) and control the actions on critical database objects.

The identified masks make our task easier because they allow us to calculate the filters for browsing log-files.

6 CONCLUSIONS

Throughout this work, we are interested in attack scenarios that appear in access control policies during their evolution. Indeed, an ACP which evolves in time can have its expression that does not match with the expression initially known or

implemented. This fact reveals a certain number of anomalies known as non-conformity ones. We developed an intrusion detection approach that takes into account the correlation that may exist between these anomalies, with our "correlated threats management system (CORMSYS)". This is a novelty as many works exist in the field of anomaly or threat detection which do not especially explore that aspect of correlation. In our future work, we intend to manage the risk related to such anomalies, according to identified users' behavior, by developing the RMS subsystem which is briefly described here.

REFERENCES

Atlam, H. F., Azad M. A., Alassafi M. O., Alshdadi A. A., Alenezi, A. (2020). "Risk-Based Access Control Model: A Systematic Literature Review", in Future Internet 2020.

Cao Y., Huang Z., Yu Y., Ke C., Wang Z. (2020). A topology and risk-aware access control framework for cyber-physical space, Frontiers of Computer Science.

Costante, E., Vavilis, S., Etalle, S., Petkovic M., Zannone, N. (2013). Database Anomalous Activities: Detection and Quantification, SECRYPT 2013: 603-608.

Cramer, I., Govindarajan, P., Martin, M., Savinov, A. (2018). Detecting Anomalies in Device Event Data in IoT. In the 3rd International Conference on Internet of Things, Big Data and Security.

Darwish, S. M. (2016). Machine learning approach to detect intruders in database based on hexplet data structure. Journal of Electrical Systems and Information Technology.

Evina, P., Labbene Ayachi, F., Jaidi, F., Bouhoula, A. (2018). Anomalies Correlation for Risk-Aware Access Control Enhancement. In: Proceedings of the 13th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE).

Evina P. A, Ayachi F. L., Jaidi F. and Bouhoula A. (2020) Enforcing Risk-Awareness in Access Control Systems: Synthesis, Discussion and Guidelines, IWCMC 2020.

Grushka-cohen, H., Biller, O., Sofer, O., Rokach, L., Shapira, B. (2019). Diversifying Database Activity Monitoring with Bandits. Computer Science, Mathematics Published in ArXiv.

Insider threats report, (2019).

Jaidi, F., Ayachi, F. L., Bouhoula, A. (2017). A comprehensive Formal Solution for Access Control Policies Management: Defect Detection, Analysis and Risk Management. *In: proceedings of the 8th International Symposium on Symbolic Computation in Software Science*, Gammarth, Tunisia.

Khan, M., OrSullivan, B., Foley, S. (2018). Towards Modelling Insiders Behaviour as Rare Behaviour to Detect Malicious RDBMS Access. In: IEEE International Conference on Big Data (Big Data). DOI: 10.1109/BigData.2018.8622047.

Ramachandran, R., Nidhin, R., Shogil, P. (2018) Anomaly Detection in Role Administrated Relational Databases-A Novel Method. In: Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI).

Sallam, A., Xiao, Q., Fadolalkarim, D., Bertino, E. (2016). Anomaly Detection Techniques for Database Protection against Insider Threats in the *17th International Conference on Information Reuse and Integration*.

Sallam, A., Bertino, E. (2017). Detection of Temporal Insider Threats to Relational Databases. In: 2017 IEEE 3rd International Conference on Collaboration and Internet Computing (CIC).

Sallam, A., Bertino, E. (2019). Result-Based Detection of Insider Threats to Relational Databases. CODASPY '19: Proceedings of the Ninth ACM Conference on Data and Application Security and Privacy Pages 133–143.

Senthil, P., Pothumani, S. (2013). Anomaly detection in RDBMS. In: International Journal of Advanced Research in Computer Science and Software Engineering.