A Risk Awareness Approach for Monitoring the Compliance of RBAC-based Policies

Faouzi Jaidi and Faten Labbene Ayachi
Digital Security Research Unit (DSRU), Higher School of Communication of Tunis (Sup'Com), Tunis, Tunisia

Keywords: RBAC, Databases Security, Policy Compliance, Risk Awareness, Quantified Risk.

Abstract: The considerable increase of the risk associated to inner threats has motivated researches in risk assessment for access control systems. Two main approaches were adapted: (i) a risk mitigation approach via features such as constraints, and (ii) a risk quantification approach that manages access based on a quantified risk. Evaluating the risk associated to the evolutions of an access control policy is an important theme that allows monitoring the conformity of the policy in terms of risk. Unfortunately, no work has been defined in this context. We propose in this paper, a quantified risk-assessment approach for monitoring the compliance of concrete RBAC-based policies. We formalize the proposal and illustrate its application via a case of study.

1 INTRODUCTION

Incorporating risk awareness in access control systems has received considerable attention in literature to face the huge increase of the risk related to inner threats. Several works had been well defined to (i) mitigate the risk of access requests, or to (ii) quantify that risk in order to deny risky accesses.

On the other hand, inner threats are one of the most dangerous threats that access control system face today. More, DataBase Management Systems (DBMSs) function as firewalls to control access to data, but unlike firewalls the access control policy is managed in the same place and way as the data it protects and consequently, it is highly exposed to corruption attempts. To face this problem, we defined a system that offers a global vision on the process of developing trusted access control policies (Jaidi and Ayachi, 2014), (Jaidi and Ayachi, 2015c). It provides a solution for monitoring the compliance of the policy and defines mechanisms for detecting anomalies (Jaidi and Ayachi, 2015d) that may corrupt the policy. Evaluating the risk associated to the detected anomalies is a very motivating and promising theme. Hence, the main contributions of this paper are structured as follows:

1. We define a risk assessment approach that aims to measure the distance of evolution, in terms of risk, between two instances of a security policy.
2. We focus when defining our approach on how to help the *security architect* to quantify that risk.
3. We define the necessary algorithms and formulas to compute the risk of the RBAC components.
4. We propose a formal representation of the main features of the defined approach.

The remainder of the paper is structured as follows. Section 2 introduces the risk assessment approach. Section 3 illustrates its relevance via a case of study. Section 4 discusses related works. Finally, Section 5 concludes the paper.

2 THE RISK ASSESSMENT APPROACH

This section introduces our approach for risk assessment of concrete RBAC-based policies.

2.1 Risk Aware Components

To define the RBAC components that need to be risk aware, we rely on the definitions of the non-compliance anomalies that may corrupt the policy (Jaidi and Ayachi, 2015a), (Jaidi and Ayachi, 2015b). Figure 1 shows the risk aware components when the policy evolves from instance 1 (the initial specification) to instance 2 (its current implementation referred by the suffix "***_IMP***").

Jaidi F. and Labbene Ayachi F..
A Risk Awareness Approach for Monitoring the Compliance of RBAC-based Policies.
DOI: 10.5220/0005577304540459
In *Proceedings of the 12th International Conference on Security and Cryptography* (SECRYPT-2015), pages 454-459
ISBN: 978-989-758-117-5

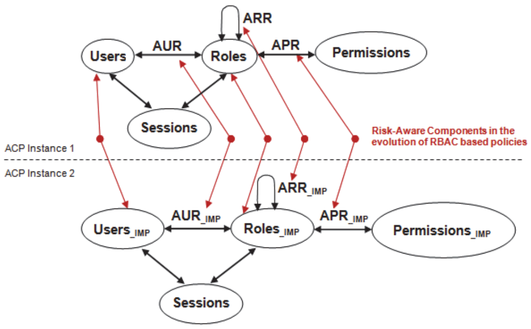


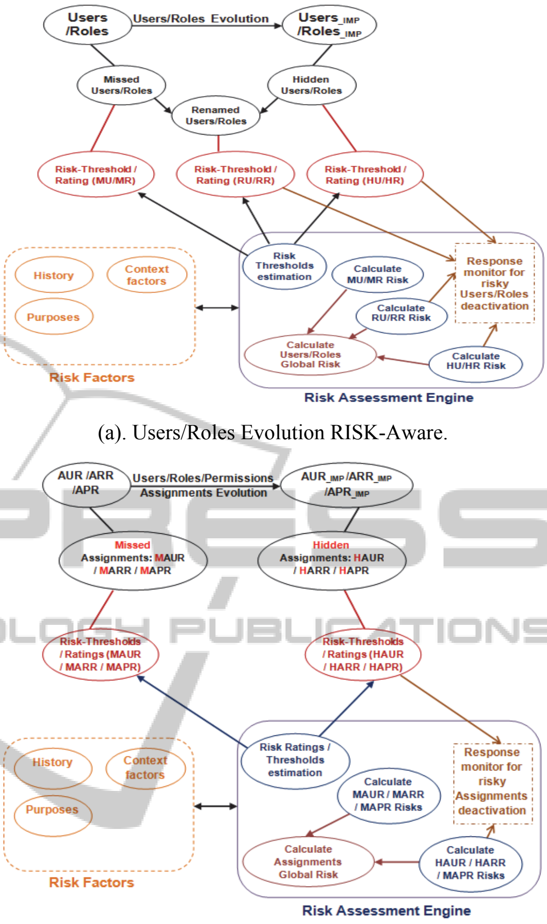
Figure 1: The risk-aware components.

- **Users:** three possible anomalies are associated to the set of users: (i) ***Hidden Users*** created and assigned rights bypassing what was planned; (ii) ***Missed Users*** removed or non implemented; and (iii) ***Renamed Users*** to avoid an audit or a system investigation. So, the ***Users*** component should be risk-aware to assess the risk of those anomalies.
- **Roles:** similarly, (i) a subset of ***Hidden Roles*** can be created; (ii) a subset of ***Missed Roles*** may be absent or removed; and (iii) a subset of ***Renamed Roles*** may be renamed. To assess the risk of those anomalies, this component should be risk-aware.
- **Users-Roles Assignments (AUR):** as the same, we identify ***Hidden AUR*** and ***Missed AUR*** the subsets of users-roles assignments illegally granted (resp. revoked/not implemented). So, the ***AUR*** component should be risk aware.
- **Roles-Roles Assignments (ARR):** similarly, we note ***Hidden ARR*** and ***Missed ARR*** the subsets of roles-roles assignments illegally assigned (resp. removed/not implemented). So, the ***ARR*** component should be risk-aware.
- **Permissions-Roles Assignments (APR):** idem, we identify ***Hidden APR*** and ***Missed APR*** the subsets of permissions-roles assignments illegally granted (resp. revoked/not implemented). Hence, the ***APR*** component should be risk-aware.

2.2 Modeling of the Approach

2.2.1 Risk Rating / Threshold Assessment

The risk associated to the evolution of the policy components is computed as a metric where a higher value is more risky than the lower one. The risk assessment engine, defined in figure 2, is able to estimate and re-estimate a risk threshold or a risk rating for each component based on the predefined risk factors such as history, contextual events, etc.


Users /Roles
Users/Roles Evolution
Users_IMP /Roles_IMP
Missed Users/Roles
Renamed Users/Roles
Hidden Users/Roles
Risk-Threshold / Rating (MU/MR)
Risk-Threshold / Rating (RU/RR)
Risk-Threshold / Rating (HU/HR)
History
Context factors
Purposes
Risk Factors
Risk Thresholds estimation
Calculate MU/MR Risk
Calculate RU/RR Risk
Calculate Users/Roles Global Risk
Calculate HU/HR Risk
Response monitor for risky Users/Roles deactivation
Risk Assessment Engine

(a). Users/Roles Evolution RISK-Aware.

AUR /ARR /APR
Users/Roles/Permissions Assignments Evolution
AUR_IMP /ARR_IMP /APR_IMP
Missed Assignments: MAUR / MARR / MAPR
Hidden Assignments: HAUR / HARR / HAPR
Risk-Thresholds / Ratings (MAUR / MARR / MAPR)
Risk-Thresholds / Ratings (HAUR / HARR / HAPR)
History
Context factors
Purposes
Risk Factors
Risk Ratings / Thresholds estimation
Calculate MAUR / MARR / MAPR Risks
Calculate Assignments Global Risk
Calculate HAUR / HARR / MAPR Risks
Response monitor for risky Assignments deactivation
Risk Assessment Engine

(b). Assignments Evolution RISK-Aware.

Legend		
HU: Hidden Users	**HR**: Hidden Roles	**RU**: Renamed Users
MU: Missed Users	**MR**: Missed Roles	**RR**: Renamed Roles
HAUR: Hidden AUR	**HARR**: Hidden ARR	**HAPR**: Hidden APR
MAUR: Missed AUR	**MARR**: Missed ARR	**MAPR**: Missed APR

Figure 2: The risk assessment approach.

Table 1 defines an initial risk rating that will be updated based on the evolution of the risk factors. The choice of five rates is not compulsory and may vary depending on the *security architect* viewpoint.

Table 1: The initial risk rating.

Risk Rating	Percentage	Description
Extremely High	≥ 80%	The component evolution associates an extremely high risk
High	≥ 60% and <80%	The component evolution associates a high risk
Moderate	≥ 40% and <60%	The component evolution associates a medium risk
Low	≥ 20% and <40%	The component evolution associates a low risk
Minor	≥ 0% and <20%	The component evolution associates a minor risk

For each rate we associate a minimum (*Rate_MinPerc*) and a maximum (*Rate_MaxPerc*) percentages to limit its borders. Below, we propose an algorithm to re-estimate the risk rating:

```
for all Rate ∈ RATING do
    If Level(Rate)== MinLevel then
        Rate_MinPerc ← 0%;
    else
        Rate_MinPerc ← $\frac{(Level(Rate)-LevelStep)*100}{MaxLevel+(\alpha*CL+\beta*H+\gamma*P+\varepsilon*TR+\theta*AR)}\%$;
    end if
    If Level(Rate) == MaxLevel then
        Rate_MaxPerc ← 100%;
    else
        Rate_MaxPerc ← $\frac{Level(Rate)*100}{MaxLevel+(\alpha*CL+\beta*H+\gamma*P+\varepsilon*TR+\theta*AR)}\%$;
    end if
end
```

Level (Rate) computes the level of each *Rate*; ***LevelStep*** is the step of levels; ***MaxLevel*** is the highest level*;* ***CL*** is the *Criticality Level* of the system; ***H*** is the *History* risk factor; ***P*** is the *Purposes* risk factor; ***TR*** quantifies the probability of risk in an average of time. For instance, accesses are more risky in time out of service than in the time of service; ***AR*** quantifies the probability of risk relative to access types. For instance, accesses are more risky from outside the office than from the inside; $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\varepsilon}$ and $\boldsymbol{\theta}$ are coefficients defined by the *security architect* that quantify the risk factors.

The idea of using a dynamic rating is important that allows decreasing the rating borders in critical situations and increasing them in normal situations.

2.2.2 Permissions, Roles and Users Risk Assessment

Different nuanced approaches defined in literature allow calculating permissions, roles and users risks. To simplify this task, we consider the following formulas, while in reality it is more complicated:

Formula (1) computes a permission risk, where *R(Pi)* denotes the risk of the permission *Pi*, *Pr(k)* denotes the probability of occurrence of a particular malicious usage k; *k= {1, . . . , m}*; and *C(k)* is the cost associated to the malicious usage *k*:

$$\mathrm{R(Pi)} = \sum_{k=1}^{m} \mathrm{Pr}(k) * \mathrm{C}(k) \qquad (1)$$

Hence, the risk of a role *Rj* is computed by formula (2) as the sum of the risk values of all permissions assigned to it, where *APR(Rj)* is the set all permissions assigned to *Rj*.

$$\mathrm{R(Rj)} = \sum_{i=0}^{n} \mathrm{R(P}i) \mid \mathrm{P}i \in \mathrm{APR(Rj)} \qquad (2)$$

The risk of the user *Ui* is evaluated by formula (3) to the sum of the risk values of the roles assigned to it defined by the set *AUR(Ui)*.

$$\mathrm{R(Ui)} = \sum_{j=0}^{n} \mathrm{R(R}j) \mid \mathrm{R}j \in \mathrm{AUR(Ui)} \qquad (3)$$

2.2.3 Roles-evolution Risk Assessment

The risk assessment engine computes the global risk associated to the evolution of the set of ***Roles*** as the sum of the risk values associated to ***Hidden***, ***Renamed*** and ***Missed Roles***. We focus, when computing the risk of those anomalies, on their impacts on the system. So, we evaluate this risk as the risk introduced by the set of *anomalous roles* regarding the set of *maintained roles*. Maintained roles are specified roles which are preserved in the implementation, computed as ***(Roles$_{IMP}$ ∩ Roles)***.

The risk of hidden roles is evaluated in formula (4) as the sum of the risk values of all hidden roles divided by the sum of the risk values of all maintained roles. We multiply it by 100 to obtain a percentage used to classify that risk.

$$\mathrm{R}(\textbf{\textit{Hidden Roles}}) = \frac{\sum_{j=0}^{n} \mathrm{R(R}j) \mid \mathrm{R}j \in \textbf{\textit{Hidden Roles}}}{\sum_{l=0}^{m} \mathrm{R(R}l) \mid \mathrm{R}l \in (\textbf{\textit{Roles\_IMP}} \cap \textbf{\textit{Roles}})} * 100\% \qquad (4)$$

Similarly, in formula (5) (i.e. formula (6)), the risk of the renamed roles (i.e. missed roles) is computed as the sum of the risk values of all renamed roles (i.e. missed roles) divided by the sum of the risk values of maintained roles.

$$\mathrm{R}(\textbf{\textit{Renamed Roles}}) = \frac{\sum_{j=0}^{n} \mathrm{R(R}j) \mid \mathrm{R}j \in \textbf{\textit{Renamed Roles}}}{\sum_{l=0}^{m} \mathrm{R(R}l) \mid \mathrm{R}l \in (\textbf{\textit{Roles\_IMP}} \cap \textbf{\textit{Roles}})} * 100\ \% \qquad (5)$$

$$\mathrm{R}(\textbf{\textit{Missed Roles}}) = \frac{\sum_{j=0}^{n} \mathrm{R(R}j) \mid \mathrm{R}j \in \textbf{\textit{Missed Roles}}}{\sum_{l=0}^{m} \mathrm{R(R}l) \mid \mathrm{R}l \in (\textbf{\textit{Roles\_IMP}} \cap \textbf{\textit{Roles}})} * 100\ \% \qquad (6)$$

Thus, the **global risk** associated to the evolution of the set of ***Roles*** is computed by formula (7).

$$\mathrm{R}(\textbf{\textit{Global Roles}}) = \mathrm{R}(\textbf{\textit{Hidden Roles}}) + \mathrm{R}(\textbf{\textit{Missed Roles}}) + \mathrm{R}(\textbf{\textit{Renamed Roles}}) \qquad (7)$$

2.2.4 Users-evolution Risk Assessment

The global risk value associated to the evolution of the set of Users is the sum of the risk values associated to ***Hidden***, ***Renamed*** and ***Missed Users***. Similarly, we evaluate this risk as the risk introduced by the set of *anomalous users* regarding the set of *maintained users*. Maintained users are specified and implemented users defined as ***(Users$_{IMP}$ ∩ Users)***.

The risk of the set hidden users is evaluated in formula (8) as the sum of the risk values of all hidden users divided by the sum of the risk values of all maintained users.

$$\mathrm{R}(\boldsymbol{Hidden\ Users}) = \frac{\sum_{i=0}^{n} \mathrm{R}(\mathrm{U}i)\ \mid \mathrm{U}i\ \in\ \boldsymbol{Hidden\ Users}}{\sum_{l=0}^{m} \mathrm{R}(\mathrm{U}l) \mid \mathrm{U}l \in (\boldsymbol{Users\_IMP} \cap \boldsymbol{Users})} * 100\ \% \quad (8)$$

Similarly, in the formula (9) (i.e. formula (10)), the risk of renamed users (i.e. missed users) is computed by dividing the sum of the risk values of all renamed users (i.e. missed users) by the sum of the risk values of all maintained users.

$$\mathrm{R}(\boldsymbol{Renamed\ Users}) = \frac{\sum_{i=0}^{n} \mathrm{R}(\mathrm{U}i)\ \mid \mathrm{U}i\ \in\ \boldsymbol{Renamed\ Users}}{\sum_{l=0}^{m} \mathrm{R}(\mathrm{U}l)\ \mid \mathrm{U}l\ \in (\boldsymbol{Users\_IMP} \cap \boldsymbol{Users})} * 100\% \quad (9)$$

$$\mathrm{R}(\boldsymbol{Renamed\ Users}) = \frac{\sum_{i=0}^{n} \mathrm{R}(\mathrm{U}i)\ \mid \mathrm{U}i\ \in\ \boldsymbol{Renamed\ Users}}{\sum_{l=0}^{m} \mathrm{R}(\mathrm{U}l)\ \mid \mathrm{U}l\ \in (\boldsymbol{Users\_IMP} \cap \boldsymbol{Users})} * 100\% \quad (10)$$

2.2.5 Assignments-evolution Risk Assessment

The risk related to the evolution of assignments relations is computed as the sum of the risk of *Hidden AUR* (**HAUR**); *ARR* (**HARR**); *APR* (**HAPR**) and *Missed AUR* (**MAUR**); *ARR* (**MARR**); *APR* (**MAPR**). We evaluate the risk of *Hidden/Missed Assignment* as the risk introduced by the set of defined/removed assignments regarding the set of *maintained assignments*. Maintained assignments are computed as ***(AUR_IMP* ∩ *AUR)*** or ***(ARR_IMP* ∩ *ARR)*** or ***(APR_IMP* ∩ *APR)***.

Formula (11) computes the risk value of the users-roles assignment relation *AUR(k)* that attributes the role *Rj* to the user *Ui*.

$$\mathrm{R}(\boldsymbol{AUR(k)}) = \frac{\mathrm{R(Rj)}}{\mathrm{R(Ui)}} \quad (11)$$

The risk associated to the set of **HAUR** is computed according to the formula (12) as the sum of the risk values of all *hidden users-roles assignments* divided by the sum of the risk values of maintained *users-roles assignments*.

$$\mathrm{R}(\boldsymbol{HAUR}) = \frac{\sum_{k=0}^{n} \mathrm{R}(\mathrm{AUR}(k))\ \mid \mathrm{AUR}(k)\ \in\ \boldsymbol{HAUR}}{\sum_{h=0}^{m} \mathrm{R}(\mathrm{AUR}(h))\ \mid \mathrm{AUR}(h) \in (\boldsymbol{AUR\_IMP} \cap\ \boldsymbol{AUR})} * 100\ \% \quad (12)$$

Similarly, the risk value of the role-role assignment relation *ARR(k)* that attributes the role *Rj* to the role *Ri* is evaluated by formula (13).

$$\mathrm{R}(\boldsymbol{ARR(k)}) = \frac{\mathrm{R(Rj)}}{\mathrm{R(Ri)}} \quad (13)$$

The risk associated to the set of **HARR** is computed according to the formula (14) as the sum of the risk values of all *hidden roles-roles assignments* divided by the sum of the risk values of maintained *roles-roles assignments*.

$$\mathrm{R}(\boldsymbol{HARR}) = \frac{\sum_{k=0}^{n} \mathrm{R}(\mathrm{ARR}(k))\ \mid \mathrm{ARR}(k)\ \in\ \boldsymbol{HARR}}{\sum_{h=0}^{m} \mathrm{R}(\mathrm{ARR}(h))\ \mid \mathrm{ARR}(h) \in (\boldsymbol{ARR\_IMP} \cap\ \boldsymbol{ARR})} * 100\% \quad (14)$$

Idem, formula (15) evaluates the risk value of the permission-role assignment relation *APR(k)* that attributes the permission *Pj* to the role *Ri*.

$$\mathrm{R(APR(k))} = \frac{\mathrm{R(Pj)}}{\mathrm{R(Ri)}} \quad (15)$$

Thus, formula (16) computes the risk associated to the set of **HAPR** as the sum of the risk values of all *hidden permissions-roles assignments* divided by the sum of the risk values of all *maintained permissions-roles assignments*.

$$\mathrm{R}(\boldsymbol{HAPR}) = \frac{\sum_{k=0}^{n} \mathrm{R}(\mathrm{APR}(k))\ \mid \mathrm{APR}(k)\ \in\ \boldsymbol{HAPR}}{\sum_{h=0}^{m} \mathrm{R}(\mathrm{APR}(h))\ \mid \mathrm{APR}(h) \in (\boldsymbol{APR\_IMP}\ \cap\ \boldsymbol{APR})} * 100\ \% \quad (16)$$

The risk associated to the set of **MAUR** is computed according to formula (17) as the sum of the risk values of all *missed users-roles assignments* divided by the sum of the risk values of *maintained users-roles assignments*.

$$\mathrm{R}(\boldsymbol{MAUR}) = \frac{\sum_{k=0}^{n} \mathrm{R}(\mathrm{AUR}(k))\ \mid \mathrm{AUR}(k)\ \in\ \boldsymbol{MAUR}}{\sum_{h=0}^{m} \mathrm{R}(\mathrm{AUR}(h)) \mid \mathrm{AUR}(h) \in (\boldsymbol{AUR\_IMP}\ \cap\ \boldsymbol{AUR})} * 100\ \ \% \quad (17)$$

The risk associated to the set of **MARR** is computed according to the formula (18) as the sum of the risk values of all *missed roles-roles assignments* divided by the sum of the risk values of *maintained roles-roles assignments*.

$$\mathrm{R}(\boldsymbol{MARR}) = \frac{\sum_{k=0}^{n} \mathrm{R}(\mathrm{ARR}(k))\ \mid \mathrm{ARR}(k)\ \in\ \boldsymbol{MARR}}{\sum_{h=0}^{m} \mathrm{R}(\mathrm{ARR}(h)) \mid \mathrm{ARR}(h) \in\ (\boldsymbol{ARR\_IMP}\ \cap\ \boldsymbol{ARR})} * 100\% \quad (18)$$

Formula (19) computes the risk associated to the set of **MAPR** as the sum of the risk values of all *missed permissions-roles assignments* divided by the sum of the risk values of *maintained permissions-roles assignments*.

$$\mathrm{R}(\boldsymbol{MAPR}) = \frac{\sum_{k=0}^{n} \mathrm{R}(\mathrm{APR}(k))\ \mid \mathrm{APR}(k)\ \in\ \boldsymbol{MAPR}}{\sum_{h=0}^{m} \mathrm{R}(\mathrm{APR}(h)) \mid \mathrm{APR}(h) \in\ (\boldsymbol{APR\_IMP}\ \cap\ \boldsymbol{APR})} * 100\ \% \quad (19)$$

2.2.6 The Response Monitor

The risk assessment engine defines a response monitor in order to automatically deactivate risky components. Risky components are identified according to the defined risk thresholds and rating. The monitor classifies the risk associated to each risk-aware component and reacts by deactivating the components based on a threshold fixed by the *security architect*. For example, if the threshold is *high risk*, the monitor deactivates all hidden and renamed users/roles if they associate *high* risk values or more. Idem, it revokes risky hidden assignments.

In order to automatically deactivate risky hidden and renamed users/roles and revoke risky hidden assignments, the monitor should be able to connect with administrative privileges to the database and execute administrative SQL statements.

3 CASE OF STUDY

We illustrate the application of our proposal via the sample discussed in (Jaidi and Ayachi, 2014) that describes a small part of a medical information system. Its functional part contains three elements: patients, doctors and medical records. Each medical record belongs to exactly one patient. Its content stores confidential data whose integrity must be preserved. Confidential data are managed only by doctors responsible for the correspondent patients. The security part of the system defines five users: two nurses Alice and Bob, two doctors Charlie and David, and Paul as a secretary. Doctors and nurses are part of the medical staff.

Like mentioned in (Jaidi and Ayachi, 2014), the formal verification and validation framework has identified the following anomalies: **Hidden Users** = {Martin, Marie}; **Missed Users** = {Bob}; **Renamed Users** =∅; **Hidden Roles** = {MedicalStudent}; **Missed Roles** =∅; **Renamed Roles** =∅; **Hidden ARR** = {(Secretary|-> MedicalStaff)}; **Missed ARR** =∅; **Hidden AUR** = {(Martin|->{MedicalStudent}), (Paul|-> {Nurse}), (Marie|-> {Secretary})}; **Missed AUR** = {(Bob|-> {Nurse}}; **Hidden APR** = {(MedicalStudent |-> (MedicalRecord|-> {modify}))}; **Missed APR** =∅.

To simplify the assessment of the risk associated to the detected anomalies, we adopt this hypothesis: R(MedicalRecord |-> modify) =8; R(MedicalRecord |-> create) = 1; R(MedicalRecord |-> read) = 1; R(MedicalRecord_Validate |-> readop) = 1; R(Patient |-> create) = 1; and R(Patient |-> read) = 1.

Applying formula (2), we compute the roles risk as follows: R(Doctor)= 10; R(MedicalStudent)= 8 R(Nurse)= 2; R(Secretary)= 2; R(MedicalStaff)= 1;.

We compute the users risk by applying formula (3): R(Alice)= 2; R(Martin)= 8; R(Charlie)= 10; R(David)= 10; R(Bob)= 2; R(Marie)= 2; R(Paul)= 4.

The risk values of the identified anomalies are computed according to the defined formulas and classified relative to the initial risk rating as follows: {R(Hidden Users)= 38.46%; ***(Low risk)***}; {R(Missed Users)= 7.69%; ***(Minor risk)***}; {R(Renamed Users)= 0%; ***(Minor risk)***}; {R(Hidden Roles)= 53.33%; ***(Medium risk)***}; {R(Missed Roles) = 0%; ***(Minor risk)***}; {R(Renamed Roles)= 0%; ***(Minor risk)***}; {R(Hidden ARR)= 83.33%; ***(Extremely High risk)***}; {R(Missed ARR)= 0%; ***(Minor risk)***}; {R(Hidden AUR)= 71.42%; ***(High risk)***}; {R(Missed AUR)= 28.57%; ***(Low risk)***}; {R(Hidden APR)= 32.25%; ***(Low risk)***}; {R(Missed APR)= 0%; ***(Minor risk)***}.

4 RELATED WORKS

Several approaches incorporated trust in RBAC systems where users are assigned to roles based on trustworthiness (Chakraborty and Ray, 2006), (Feng et al., 2008). Several authors had focused on constraints-based risk mitigation approaches (Simon and Zurko, 1997), (Jaeger, 1999) in RBAC systems while authors in (Chen and Crampton, 2011) propose a mitigation strategy based on risk thresholds and obligation pairs. As for the risk quantification approaches, authors in (Cheng et al., 2007) propose a model to quantify risk for access control and provide an example for multilevel information sharing. In (Ni et al., 2010), authors propose a model for estimating risk and induce fuzziness in the access control decision. Authors in (Molloy et al., 2012) propose a risk-benefit approach for avoiding communication overhead in distributed access control. In (Bijon et al., 2012), authors propose a quantified risk-aware RBAC sessions framework. In (Ma et al., 2010), authors calculate the risk associated to a user when activating a role based on the level of confidence assigned to the role and the clearance level of the user. In (Nissanke and Khayat, 2004), authors propose a risk based security analysis of permissions in RBAC. Authors in (Aziz et al., 2006) propose a model for reconfiguring RBAC policies using risk semantics. In (Baracaldo and Joshi, 2012), authors propose a model based on risk and trust evaluation in RBAC systems in order to react to inner threats. Authors in (Ma, 2012), (Bijon et al., 2013) propose formal approaches to react based on quantified risk in RBAC systems.

The main goal of the cited works and approaches is: (i) to enhance trustworthiness relationships in RBAC systems; or (ii) to define mitigation strategies based on constraints; or (iii) to manage accesses based on a quantified risk. According to our knowledge, no work has been defined to assess the risk associated to the evolution of the components of RBAC policies. To fill this gap, our proposal aims to quantify the risk associated to the evolution of the policy components. This evaluation is associated to the detected anomalies of non-compliance that may characterize the states evolution of RBAC policies.

5 CONCLUSION

This paper proposes a formal risk-awareness approach for qualifying the states evolution of RBAC-based policies in terms of risk. The proposal is a dynamic quantified approach that computes the risk values and the corresponding risk rating and thresholds. It incorporates also an automatic response monitor to quickly react face risky non compliance anomalies. This allows monitoring the compliance of RBAC policies based on risk metrics. Ongoing works address mainly the refinement of the formalization of the proposal as well as its finer integration in the verification and validation system.

REFERENCES

Aziz, B., Foley, S. N., Herbert, J., Swart, G., 2006. Reconfiguring role based access control policies using risk semantics. In *Journal of High Speed Networks*.

Baracaldo, N., Joshi, J., 2012. A trust-and-risk aware rbac framework: tackling insider threat. In: *SACMAT 2012*, pp. 167–176, ACM, New York.

Bijon, K. Z., Krishnan, R., Sandhu, R., 2013. A framework for risk-aware role based access control. *In Communications and Network Security*, pp. 462–469.

Bijon, K. Z., Krishnan, R., Sandhu, R., 2012. Risk-aware RBAC sessions. *In Information Systems Security*, pp. 59–74, Springer.

Chakraborty, S., Ray, I., 2006. Trustbac: integrating trust relationships into the rbac model for access control in open systems. In Proc. of *the 11th ACM symposium on Access control models and technologies, SACMAT '06*, pp. 49-58, USA.

Chen, L., Crampton, J., 2011. Risk-aware role-based access control. In Proc. of *the 7th International Workshop on Security and Trust Management*.

Cheng, P.-C., Rohatgi, P., Keser, C., Karger, P.A., Wagner, G.M., Reninger, A.S, 2007. Fuzzy multi-level security: an experiment on quantified risk-adaptive access control. In *Security and Privacy*, pp. 222 –230.

Feng, F., Lin, C., Peng, D., Li, J., 2008. A trust and context based access control model for distributed systems. In Proc. of *the 10th IEEE International Conference on High Performance Computing and Communications, HPCC '08*, pp. 629-634, USA.

Jaeger, T., 1999. On the increasing importance of constraints. In *fourth ACM workshop on Role-based access control*, pp. 33–42.

Jaidi, F., Labbene Ayachi, F., 2014. An approach to formally validate and verify the compliance of low level access control policies. The *13th International Symposium on Pervasive Systems, Algorithms, and Networks (I-SPAN 2014)*.

Jaidi, F., Labbene Ayachi, F., 2015. A formal system for detecting anomalies of non-conformity in concrete RBAC-based policies. *International Conference on Computer Information Systems WCCAIS-2015- ICCIS*.

Jaidi, F., Labbene Ayachi, F., 2015. The problem of integrity in RBAC-based policies within relational databases: synthesis and problem study. *The 9th International Conference on Ubiquitous Information Management and Communication ACM IMCOM*.

Jaidi, F., Labbene Ayachi, F., 2015. To summarize the problem of non-conformity in concrete RBAC-based policies: synthesis, system proposal and future directives. *In NNGT International Journal of Information Security*, vol. 2, pp. 1-12.

Jaidi, F., Labbene Ayachi, F., 2015. A formal approach based on verification and validation techniques for enhancing the integrity of concrete role based access control policies. In *8th International Conference on Computational Intelligence in Security for Information Systems*, CISIS 2015.

Ma, J., 2012. A formal approach for risk assessment in RBAC systems. *Journal of Universal Computer Science*, vol. 18, pp. 2432-2451.

Ma, J., Adi, K., Mejri, M., Logrippo, L., 2010. Risk analysis in access control systems. In *Eighth Annual International Conference on Privacy Security and Trust (PST)*, pp. 160-166.

Molloy, I., Dickens, L., Morisset, C., Cheng, P.-C., Lobo, J., Russo, A., 2012. Risk-based security decisions under uncertainty. *CODASPY '12*.

Ni, Q., Bertino, E., Lobo, J., 2010. Risk-based access control systems built on fuzzy inferences. *ASIACCS'10*, pp. 250-260, USA.

Nissanke, N., Khayat, E. J., 2004. Risk based security analysis of permissions in rbac. In Proc. of *the 2nd International Workshop on Security in Information Systems*, pp. 332-341, INSTICC Press.

Simon, T. R., Zurko, M. E., 1997. Separation of duty in role based environments. In *Computer Security Foundations Workshop*, pp. 183–194.