Risk-based Auto-Delegation for Probabilistic Availability

Leanid Krautsevic, Fabio Martinelli, Charles Morisset, Yautsiukhin Artsiom

Risk-based Auto-Delegation for Probabilistic Availability *

Leanid Krautsevich[1], Fabio Martinelli[2], Charles Morisset[2], and Artsiom Yautsiukhin[2]

[1] University of Pisa, Department of Computer Science
Largo B. Pontecorvo 3, 56127 Pisa, Italy
krautsev@di.unipi.it
[2] IIT-CNR, Security Group
Via Giuseppe Moruzzi 1, 56124 Pisa, Italy,
firstname.lastname@iit.cnr.it

Abstract. Dynamic and evolving systems might require flexible access control mechanisms, in order to make sure that the unavailability of some users does not prevent the system to be functional, in particular for emergency-prone environments, such as healthcare, natural disaster response teams, or military systems. The auto-delegation mechanism, which combines the strengths of delegation systems and "break-the-glass" policies, was recently introduced to handle such situations, by stating that the most qualified available user for a resource can access this resource.
In this work we extend this mechanism by considering availability as a quantitative measure, such that each user is associated with a probability of availability. The decision to allow or deny an access is based on the utility of each outcome and on a risk strategy. We describe a generic framework allowing a system designer to define these different concepts. We also illustrate our framework with two specific use cases inspired from healthcare systems and resource management systems.

Keywords: Auto-delegation, Access control, Risk, Availability

1 Introduction

The main function of an access control system is to intercept any access request made by an active entity (subject) to a passive entity (object) in order to decide if an access should be granted or denied. An access control system usually consists of two parts: an access control policy and a reference monitor. The access control policy explicitly defines which access requests should be allowed and which should be denied. The reference monitor is a piece of software responsible for intercepting access requests and matching them against the policy.

* work partially supported by EU FP7-ICT project NESSoS (Network of Excellence on Engineering Secure Future Internet Software Services and Systems) under the grant agreement n. 256980 and by EU-funded project "CONNECT".

The problem of defining a powerful-enough access control system has been identified long time ago [21] and since then several well-known policies and systems have been proposed. Examples of access control models are the discretionary model [21, 16], used, for instance, in operating systems; the Bell-LaPadula model [22], used in a military environment; the Chinese Wall model [2], used in the consulting world; and more recently-proposed the Role-Based (RBAC) model [13], used in databases and business information systems.

A common feature among these models is that they have the main focus on subjects: each policy describes which objects a subject can access. Usually, such policy defines a decision about the access for a subject independently from the other subjects. For instance, in an health-care system, a nurse cannot access a medical record regardless of the fact that a physician can or cannot access this medical record. Although this independence property makes sense in the general case, in some situations, subjects which have access to a critical object may be unavailable. In such situations the object cannot be accessed by anyone and this limitation of the access control system leads to a potentially life-threatening situation. An example could be a patient record required for curing a person who is having an heart attack or essential military intelligence report when it is unknown if the responsible officer is alive or not.

Hence, in some situations, there is a clear need to provide the access control mechanism with the possibility of granting an access that was not originally allowed. Two main approaches exist in the literature to address this need: the enforcement of delegations, and "break-the-glass" policies. A delegation mechanism [4] allows a user to delegate some of her permissions to another user. For example, a doctor delegates her right to access records of her patients to her assistant when she is not available. The main drawback of delegations is that they need to be activated beforehand, and they are not suitable in case of unexpected unavailability. Moreover, the delegatee may be also unavailable at the time of need. On the other hand, "break-the-glass" policies [1, 29, 3] grant access to any subject in case of emergency, usually enforcing auditing and logging mechanisms. Thus, a poorly qualified subject may get access to a critical object.

Crampton and Morisset introduced an auto-delegation mechanism [10], which tries to combine the advantages of the delegation and "break-the-glass" approaches, while limiting the drawbacks. In a system enforcing the auto-delegation mechanism, a user is associated with a level of qualification for each object. Similar to usual access control system behaviour, an access control policy in such system allows the most qualified available user for an object to access this object at any time. On the other hand, if this most qualified subject is unavailable, another, a bit less qualified user is allowed to access this object. Moreover, as for the delegation mechanism and in contrast to "break-the-glass" policies, an access is not authorised if there is a more qualified available subject.

However, Crampton and Morisset [10] define the notion of availability as a boolean notion: a user is either available or she is not. In practice, the availability of a user can depend on many parameters, such as her localisation, her level of commitment for other projects, etc. Thus, availability can be only estimated with

some uncertainty, due to the *staleness* or *freshness* of the security attributes, as identified in [18].

1.1 Contributions

In the current work we assume a level of uncertainty for availability of users to be a quantitative value (probability of availability), and we propose a quantitative approach to the problem of auto-delegation. In other words, the proposed approach is capable of deciding whether an access should be granted to a user, according to the probability that a more qualified user is available. The main contribution of this paper is therefore the definition of such a policy together with the corresponding framework for the decision-making process. In this work, we provide a general mathematical model for a wide-range of situations and describe a couple of scenario-specific examples with exact equations.

This work is structured as follows. In Section 2 we describe basics of the auto-delegation mechanism and provide a high level example for finding availability probabilities required for our method. Section 3 contains the main contribution of this paper - the general approach for making decisions for auto-delegation mechanism under uncertainty. Concrete examples of application of our approach in a healthcare and in a resource-management environment are shown in Section 4. Section 5 is devoted to possible extensions of our model. Finally, we present the related work (Section 6) and we summarise results of our work (Section 7).

2 Background

2.1 Auto-delegation mechanism

We recall in this section the auto-delegation mechanism introduced in [10]. We write $\mathcal{S} = \{s_1, s_2, \cdots, s_n\}$ for the set of subjects, $\mathcal{O} = \{o_1, o_2, \cdots, o_m\}$ for the set of objects. An *access* is a pair (s, o), meaning that the subject s accesses the object o. Access modes are voluntarily not considered, for the sake of the exposition. However, this can be easily done by considering the set of permissions *à la* RBAC, where a permission is pair (object, access mode), instead of the set of objects. The availability of the subjects is considered to be always decidable, and, therefore, the authors introduce a set $Av(\mathcal{S}) \subseteq \mathcal{S}$, such that a subject is available if, and only if, it belongs to $Av(\mathcal{S})$.

Each object $o \in \mathcal{O}$ is associated with a qualification hierarchy $(Q(o), \leqslant_o)$, and each subject is associated with a qualification through a function $\lambda_o : \mathcal{S} \to Q(o)$, such that $\lambda_o(s)$ denotes the qualification level of s, with respect to o. Given two subjects s_1 and s_2, $\lambda_o(s_1) \leqslant_o \lambda_o(s_2)$ means that s_2 is more qualified than s_1 to access o. Note that the relation $\leqslant_o$ is a partial-order, and therefore two qualifications might not be comparable.

Finally, an authorization function $Auth_{adm}$ is given, such that given $\leqslant_o$, $Av(\mathcal{S})$, and an access request (s, o), $Auth_{adm}(\leqslant_o, Av(\mathcal{S}), (s, o))$ returns allow

if (s, o) is authorized according to the auto-delegation mechanism, and deny otherwise. More precisely,

$$Auth_{adm}(\leqslant_o, Av(\mathcal{S}), (s, o)) = \begin{cases} \mathsf{deny} & \text{if there exists } s' \in Av(\mathcal{S}) \text{ such that } \lambda_o(s) \leq_o \lambda_o(s'), \\ \mathsf{allow} & \text{otherwise.} \end{cases}$$

In other words, a request by s to access o is allowed if s is one of the most qualified of the available subjects (and denied otherwise).

The auto-delegation mechanism can be either used as a standalone policy, for instance in the context of resource management, or as a combination with another policy. In the latter case, the auto-delegation mechanism is consulted only if the "normal" policy denies the access. We focus on this usage here, and we refer to [10] for a more detailed presentation.

2.2 Uncertain Availability

Usually availability of a subject is considered as a zero-one value, i.e., a subject is either available or not. But sometimes we do not have complete information about the availability of a subject and can only speak about it with some degree of certainty. For example, a doctor may be in a hospital but working on a different floor of the building. We cannot know precisely at each moment of time if the doctor is available or not for a prompt response in case of an emergency. Another example could be a person who could be currently out of her office and the only information available about her whereabouts is that 10 minutes ago the person was in some other office. Hence, it is uncertain if the person will be able to answer an urgent e-mail in the next twenty minutes. We model uncertain availability of a person as the possibility of a subject to response to the request by a certain moment of time in the future. Then we decide if another subject with certain availability but lower qualification can get an access to an object (e.g., patient medical record or a e-mail account) while the availability of more qualified subject is uncertain.

Uncertainty is usually expressed with probability. In our case, we need the probability that a person is available. Such probability could be found in very different ways. The probability could be simply assigned by an analyst, could be derived out of statistics, could be computed, etc. An example of how the probability is computed can be found in the work of Krautsevich et. al.,[19, 18], though, in general, the way of acquiring the probability does not affect the following discussion.

Krautsevich *et al.*, [19, 18] model position and movement of a subject with a Markov chain (assuming that Markovian property holds[3]). States (nodes) of the Markov chain represent possible spatial positions of the subject. Edges of the Markov chain represent possible transitions between the states. Transition

[3] In fact, the experimentation results presented in the paper prove that Markovian property does hold for such example.

probabilities, taken from the analysis of historical data, are assigned to every edge. Knowing the position of a subject at some point of time in the past the probability that the subject is in a specific location (available) at the current moment of time is computed[4].

Both discrete-time and continuous-time Markov chains may be used for the assessment of the probability. A discrete-time Markov chain is a suitable model when the number of position changes during considered time intervals is known. A continuous-time Markov chain is an appropriate model when the number of changes is not known and only the considered interval is available.

The authors also provide a decision making technique in case of uncertain position of a subject. In order to decide to grant access or to deny it a decision matrix is used with possible benefits and risks as utility. Thus, the decision is made comparing possible benefits and losses of granting access or denying it.

In the sequel, we assume that the probability of availability is known for each subject, regardless of the way of acquiring it. This assumption relies on the fact the behavior of subjects follows a predictable pattern, that can be modeled. For instance, if we do not model the possibility for a user to be hit by lightning, and if this event occurs, then we might not be able to assess that the user is not available. In other words, the accuracy of the probability of availability depends on the accuracy of the model. In the current work, we use this probability to develop a model for making a rational decision about *delegating* the right to access an object. We also use a decision matrix for the decision-making process (see Section 3.2).

3 Auto-delegation under uncertainty

We present in this section a very abstract and general model of access control under uncertainty. We reduce an access control mechanism as a binary decision mechanism: given an access request, the access control mechanism can either allow it (positive decision) or deny it (negative decision). This decision is based on the information present in the current state of the system. When there is no uncertainty on the information, then the decision making is straight-forward.

However, when there is some uncertainty over the information present in the state, the decision process is more complex, as it is possible to make some errors. For instance, consider a simple policy when an access a is allowed if, and only if a parameter x is *true*. If we know with certainty the value of x, then the decision making simply consists in checking this value, and allowing or denying the access a accordingly. On the contrary, if there is an uncertainty over the value of x, then we can have four different decisions.

1. A *true-positive* is an access correctly allowed. For instance, we allow the access a and the value of x is *true*.

[4] In fact, the proposed approach computes the probability of a subject to appear in a forbidden state (be unavailable) during the considered interval. But, in order to find the probability to be in a particular state we need only slightly change the model (i.e., remove absorbing states).

2. A *true-negative* is an access correctly denied. For instance, we denied the access a and the value of x is $false$.
3. A *false-positive* is an access wrongly allowed. For instance, we allow the access a and the value of x is $false$.
4. A *false-negative* is an access wrongly denied. For instance, we deny the access a and the value of x is $true$.

In order to make a decision, the mechanism must evaluate the impact of each decision, and the corresponding probabilities. In general, specifying the actual impact of a decision is a hard task, and can depend on the request and the environment. Moreover, the impact of a false-positive might be not comparable with the impact of a false-negative. Indeed, intuitively, a false-positive can damage the system by leaking some information to non-authorised users, or allowing them to modify the system in an undesired way, while a false-negative will prevent a user to perform an access that she actually needs in order to fulfil a task, thus potentially leading to the failure of this task.

For instance, considering extreme cases, a system denying every access is clearly secure, but is of very limited interest, and does not provide any *gain*. Conversely, a system allowing every access has a high *gain*, but at the same time leads to high *damage*. The objective here is then to strike the right balance between being too strict and being too lax.

We now introduce more precisely notions of "utility", "gain", and "damage", and then we present our model to make decisions under uncertainty.

3.1 Utility, gain and damage

The shift from a binary viewpoint, where a state is either "good" or "bad", to a more quantitative approach, where a state is associated with a "level of goodness" and/or a "level of badness", has been undergoing in the last decade. The Computer Research Association stated it as a grand challenge to develop an accurate risk analysis for cyber-security [9], and some measures need to be defined. There have been several approaches taking advantages of such measures [8, 7, 6, 26] or trying to calculate them using market algorithms [25].

Clearly, defining a risk measure for access control is complex [11], particularly due to the lack of a precise notion of utility (gain and/or damage), that is, a clear way to define the impact of granting or denying an access, on the contrary of, say, defining the impact of increasing the temperature of a water-heater or purchasing shares on a stock-market. Moreover, when human lives are potentially at stake, it is hard to put a number on the possible impact. However, some approximations can be done.

For instance, in an healthcare system, a situation where a patient is having a heart attack and where a medical record is unavailable might lead to a wrong or unadapted course of treatment. Such a situation might lead the patient to sue the hospital, and the financial impact on the hospital can be estimated from past similar cases. Also giving access to the medical record to unauthorised employees might lead to the divulgation of confidential information, and the

patient again might sue the hospital for breach of confidentiality. In this case, gain and damages can be evaluated as financial impact over the hospital budget.

As recently stated by Kephart [17], the access control community must first establish how to measure the utility of access control decisions in order to meaningfully apply risk-based strategies. However, the objective of the work presented here is not to define what is utility, but to provide the tools to take advantage of it once it can be more precisely defined for concrete systems. Hence, we assume that there exist two general utility functions μ_g and μ_d, for the utility of granting and denying an access, respectively. For instance, these functions can be defined by a gain function γ and a damage function δ. Such functions can dynamically evolve, and can return complex values, such as probabilistic costs. Note that in general, the gain from allowing an access can be different from the damage of denying this access, for instance when penalties for denying accesses are involved. We provide examples of such functions in Section 4.

We assume that the domain of these utility functions is a total order, in other words, it is always possible to compare two or more different utility values and to pick the "best" value. Moreover, the qualification hierarchy for each object needs to be consistent with the utility: intuitively, if a subject s_1 is more qualified than a subject s_2 to access an object o, then the utility of the access (s_1, o) is better than the utility of (s_2, o). In practice, a good way to define the qualification hierarchy and the qualification assignment is simply to base them on the utility functions.

3.2 Mathematical Model

The question we try to answer here is when a subject s_i asks to access an object o whether the monitor should grant this access or not. In order to make such a decision, we need to determine the following points:

- the probability that a more qualified subject is available,
- the utility of granting and denying the access to s_i if such a subject exists,
- and the utility of granting and denying the access to s_i otherwise.

For the sake of exposition, we order subjects from the most qualified to the least qualified, *i.e.* s_1 is the most qualified subject for o and s_n is the least qualified. Thus, if qualifications of subjects relate as $\lambda_o(s_i) \leq_o \lambda_o(s_j)$ indexes relate as $i > j$. We also denote by p_j the probability that subject s_j is available.

Given the subject s_i, Table 1 describes, for a subject s_j, $i \leq j$, (first column), the probability p_a^j that s_j is the most qualified subject available (second column), and the utility in this case of granting or denying the access to s_i over o (third and fourth columns).

It is easy to see that when s_i is the most qualified available subject, granting the access is a true-positive while denying it leads to a false-negative. Conversely, when there exists a subject s_j available, with $j > i$, then granting the access is a false-positive while denying it creates a true-negative. We define four utility functions C^{TP}, C^{FN}, C^{FP}, and C^{TN} for these respective outcomes, such that,

Highest available	Probability	Utility of grant(s_i, o)	Utility of deny(s_i, o)
s_i	$(1-p_1)\cdots(1-p_{i-1})$	$\mu_g(s_i, s_i)$	$\mu_d(s_i, s_i)$
s_{i-1}	$p_{i-1}\cdot(1-p_1)\cdots(1-p_{i-2})$	$\mu_g(s_i, s_{i-1})$	$\mu_d(s_i, s_{i-1})$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
s_2	$p_2\cdot(1-p_1)$	$\mu_g(s_i, s_2)$	$\mu_d(s_i, s_2)$
s_1	p_1	$\mu_g(s_i, s_1)$	$\mu_d(s_i, s_1)$

Table 1. Alternatives of granting or denying the access to the object o

according to decision theory [15], the access should be granted only if Equation 1 holds.

$$C^{TP} + C^{FP} > C^{TN} + C^{FN} \tag{1}$$

In order to calculate these utility functions, we first write p_a^j for the probability of the subject j to be the most qualified subject available. Note that since s_i is asking for the access, we automatically assume that this subject is available, and it follows that the sum of p_a^j, for $j \geq i$, is equal to 1. More formally, we define $p_a^j = p_j \cdot \prod_{k>j}(1-p_k)$

Finally, Equations 2, 3, 5, and 4 give the definition of the utility functions. For the sake of simplicity, we do not consider here any special utility or reward for the accuracy of the decision process, only the expected impact of the access, or the absence of access. However, in some situations, these utility functions might also include a mechanism to reward the fact to make good decisions, and/or to punish the fact to make bad ones.

$$C^{TP} = p_a^i \cdot \mu_g(s_i, s_i) \tag{2}$$

$$C^{FP} = \sum_{j>i} p_a^j \cdot \mu_g(s_i, s_j) \tag{3}$$

$$C^{FN} = p_a^i \cdot \mu_d(s_i, s_i) \tag{4}$$

$$C^{TN} = \sum_{j>i} p_a^j \cdot \mu_d(s_i, s_j) \tag{5}$$

The Equation 6 corresponds to the equation 1, after simplification, such that, if this equation holds, then the access is granted, otherwise the access is denied. We clearly focus here on the notion of expected utility, i.e., we average the utility of granting and the utility of denying, and simply pick the best option. More complex notions of risk, such as risk aggregation or worst-case scenario, could be easily included, and will be considered in future works.

$$\sum_{j\geq i} p_a^j \cdot \mu_g(s_i, s_j) > \sum_{j\geq i} p_a^j \cdot \mu_d(s_i, s_j) \tag{6}$$

The definition of the μ functions is context-dependent: for instance, granting an access over an object to a subject might prevent another, more qualified subject to access this object, in which case the maximum obtainable gain is the one from the less qualified subject. In other situations, there might be no

damage associated with granting an access, thus reducing the access control mechanism to a traditional optimization problem. We define in the next section two examples of the μ functions, one in the healthcare environment, and the other one for resource management.

4 Examples

The mathematical model presented in the previous section only requires a system administrator to define the functions μ_g and μ_d for her specific environment. We consider here an healthcare system and a resource management system.

4.1 Auto-Delegation for Healthcare

We consider an emergency in an hospital (*e.g.* a patient has a heart attack) when the personnel of the hospital must respond immediately. We suppose the following hierarchy of the personnel: $\lambda_o(intern) <_o \lambda_o(attending\ physician) <_o \lambda_o(senior\ attending\ physician) <_o \lambda_o(chief\ of\ medicine)$. A subject who is trying to tackle the situation needs to access the *medical record* of the patient for the selection of a proper treatment. Thus, an object o is the medical record of a patient. In this setting, the access to a resource is not exclusive: a more qualified subject can still access the resource even if a less qualified subject has already got the access. Indeed, a senior attending physician still can access the medical record of a patient during an emergency even if the record has been accessed by an intern.

The utility functions μ_g and μ_d are computed from the gain function γ and the damage function δ. Given an object o, we write γ_k for the gain obtained by the subject s_k, and δ_k for the damage caused by the same subject. We suppose that the more qualified is the subject accessing the medical record, the better the treatment the patient receives. Similarly, the more qualified is the subject accessing the resource, the lesser is the damage to the privacy of a patient.

$$\mu_g(s_i, s_j) = \begin{cases} \gamma_i - \delta_i & \text{if } s_i = s_j \\ \gamma_j - (\delta_i + \delta_j) & \text{otherwise} \end{cases}$$

$$\mu_d(s_i, s_j) = \begin{cases} -\delta_0 & \text{if } s_i = s_j \\ \gamma_j - \delta_j & \text{otherwise} \end{cases}$$

Considering the utility $\mu_g(s_i, s_i)$, we assume that the subject s_i executes her duties, however some damage δ_i is possible, e.g., the subject will corrupt the privacy of a patient. The cost $\mu_g(s_i, s_j)$ if $i \neq j$ corresponds to the case when the subject s_i accesses the resource, while one or several more qualified specialists are available, for example when an intern accesses the record while a senior attending physician is available. In this case, we assume that giving the access to the less qualified subject does not bring any gain, and therefore the gain is only the one of the most qualified available subject. However, we sum the damages caused by both subjects that access the object.

The utility $\mu_d(s_i, s_i) = -\delta_0$ stands for the damage occurring when the object is not accessed at all. The utility $\mu_d(s_i, s_j)$ if $i \neq j$ assesses the case when the access is denied to s_i and the subject s_j is available. In this case we face the gain and the damage from s_j. Table 2 illustrates the utility functions C^{TP}, C^{FP}, C^{TN} and C^{FN} with the previous definitions for the functions μ_g and μ_d.

C^{TP}	C^{FP}	C^{TN}	C^{FN}
$p_a^i \cdot (\gamma_i - \delta_i)$	$\sum_{j>i} p_a^j \cdot (\gamma_j - \delta_i - \delta_j)$	$\sum_{j>i} p_a^j \cdot (\gamma_j - \delta_j)$	$p_a^i \cdot \delta_0$

Table 2. Utility functions C for the healthcare example

We are now in position to instantiate Equation 6 with the previous definitions of μ_g and μ_d, which is given by Equation 7, and simplified in Equation 8.

$$p_a^i \cdot (\gamma_i - \delta_i) + \sum_{j>i} p_a^j \cdot (\gamma_j - (\delta_i + \delta_j)) > \sum_{j>i} p_a^j \cdot (\gamma_j - \delta_j) - p_a^i \cdot \delta_0 \quad (7)$$

$$p_a^i \cdot \gamma_i - \delta_i > -p_a^i \cdot \delta_0 \quad (8)$$

Equation 8 illustrates two strengths of our approach: if it is certain that s_i is not the most qualified available subject, then $p_a^i = 0$ and the access is not granted, regardless of the value of δ_i and δ_0. On the other hand, if δ_0 is important enough (*e.g.* a threat of patient death), then even if the probability that s_i is the most qualified available subject is very small, then the access can potentially be granted, thus working as a "break-the-glass" policy.

4.2 Auto-delegation for resource management

We illustrate here that our framework can also be used to address a typical resource management problem. We consider a channel with limited bandwidth which can be accessed by premium and regular users. A premium user pays γ^p while a regular user pays $\gamma^r < \gamma^p$, therefore, the premium user has a priority for using the channel. The resource owner delegates the right to access the resource to a regular user when the resource is not occupied by a premium user, *i.e.* the premium user is unavailable. The qualification relation is $\lambda_o(\textit{regular user}) <_o \lambda_o(\textit{premium user})$. An object o is the channel and two users cannot access the channel simultaneously. Suppose that a regular user is asking for the channel and the availability of premium users is uncertain. The resource owner needs to decide what is more profitable: giving the access to the regular user or considering that a premium user is actually available and will require the channel.

$$\mu_g(s_i, s_j) = \gamma^r \qquad \mu_d(s_i, s_j) = \begin{cases} 0 & \text{if } s_i = s_j \\ \gamma^p & \text{otherwise} \end{cases}$$

Note, that since we only have two categories of users, s_j in the definition of μ_d automatically stands for a premium user. For the sake of simplicity, we consider that there is no special penalty for granting an access to a regular user while a premium user is available. There is only an implicit loss of gain, since the system could have gained more by giving the access to the premium user. The Table 3 illustrates the utility functions C^{TP}, C^{FP}, C^{TN} and C^{FN} with the previous definitions for the functions μ_g and μ_d.

C^{TP}	C^{FP}	C^{TN}	C^{FN}
$p_a^i \cdot \gamma_r$	$\sum_{j>i} p_a^j \cdot \gamma_r$	$\sum_{j>i} p_a^j \cdot \gamma_p$	0

Table 3. Utility functions C for resource management

Finally, we can instantiate Equation 6 with the corresponding values, and we find, after simplification, that the access to a regular user is granted if $\gamma^r > (1-p_a^i) \cdot \gamma^p$. Clearly, this is not a surprising result, as the inequality corresponds to the intuitive decision process: the regular user can access the resource only if the expected gain is greater than the expected gain of giving the access to a premium user weighted by the probability that such a user is available. Clearly, this example illustrate that resource management can also be done within our framework.

5 Extensions

We propose here several straight-forward extensions to the model presented in the previous section.

5.1 Auto-Delegation for a Dynamic System

In this extension we are going to consider the case when parameters of the systems may change over time. Parameters of the system impact the utility functions, thus re-evaluation of access decision should be done each time when parameters change. Usage control model (UCON) is based on changing attributes and allows controlling how a subject uses an object even after access is granted. We are going to propose an auto-delegation model for UCON, which should help to control the access to objects in case of dynamically changing utility. The model described in Section 3.2 assumes that the utilities for a given access is constant. However, in many situations, the utility of an access can vary over a period of time.

First, if we assume that we know the estimated time of availability of a subject, then the model can take it into account. As an example, consider the situation where a nurse needs to access a medical record for which the official

physician is currently performing a surgery, and therefore unavailable for a period of n minutes. The mechanism must weight the outcomes between granting to the nurse the access now, which we denote t_0, with the potential incurred damage, thus getting the utility at t_0, or wait for n minutes and let the official physician access the medical record, and in this case getting the utility at $t_0 + n$.

Second, the probability that the subject will be available may change in time. For instance, if we have a tracing system in the hospital, we can re-evaluate the availability of more qualified subject each minute, and, on the basis of this information, grant or deny access for a less qualified subject.

5.2 Forcing Availability

In some cases, a subject unavailable in a given system can be forced to be available, at a given cost. A simple example is a physician unavailable because on leave at home, and who can be called back to the hospital, however at the cost of paying her extra hours. Another example is the unavailability of a subject due to some conflicts.

For instance, consider a subject s currently accessing a resource o_1 and being the more qualified subject for another resource o_2, although not accessing it. The policy forbids simultaneous accesses to both o_1 and o_2. If another subject s' asks to access o_2, then the mechanism has to consider two options: either granting the access to s' or releasing the access of s to o_1 so that s can be available for o_2. However, releasing the access over o_1 might incur some extra costs.

In other words, the decision mechanism might have the choice between granting the access, denying it, or forcing a more qualified subject to be available, and can choose the option with the best utility.

6 Related Work

Clearly, our approach is related to risk-based access control, as we try to make an access control decision under uncertainty. Some authors use risk as a static parameter which simply helps to assign correct privileges taking into account possible losses [23, 14, 28]. For example, Skalka *et al.* [28] discussed an approach for risk evaluation of authorisations, the formal approach is used to assess and combine the risks of assertions that are used in the authorisation decision. Other authors use risk as a dynamic value which depends on the current value of possible losses and benefits as well as on the probability of abusing privileges by a concrete subject [30, 12, 24, 8]. These approaches do not cover the delegation problem, but introduce useful concepts for defining the *cost* functions.

Ni *et al.* [27] considered risk-based access control system (RAC) which assumes that the access to a resource can be granted to a risky subject if mitigation actions (post-obligations) will be applied in the future. The authors proposed an approach for the risk estimation under incomplete and imprecise data using fuzzy inferences. Similarly, Chen and Crampton propose a mitigation strategy [5]

in the context of risk-based access control for RBAC models. The use of mitigations can provide the access control monitor with a wider range of possible decisions: denying a request, allowing it with some mitigations, or allowing it with no condition. Each of these decision comes with a different impact, and thus permits a finer-grained risk strategy.

Krautsevich *et al.* [20, 19, 18] applied risk analysis for usage control model. In [20] the authors considered the selection of the less risky data processor in service-oriented architecture. The authors indicated how risk can change after granting the access to a data processor and how the data processor can reduce its risk level to provide better service. In [19, 18] authors described the approaches for risk-aware usage decision making under uncertainties caused by freshness of the policy attributes. The approaches exploits discrete-time and continuous-time Markov chains.

7 Conclusion

In this paper we have presented an access control mechanism for making delegation decisions when availability of subjects is uncertain. Our approach fits within the recent trend of tackling access and usage control issues from a risk-based viewpoint. Thus, this approach enables a finer-grained access control, but is not conservative with respect to more "traditional" approaches. Indeed, our mechanism might grant an access that should be denied (false-positive). On the contrary, a conservative approach could require to grant an access only if the system can be sure that this is the right decision. However, such a system would also deny accesses that should be allowed (false-negative), thus jeopardising the functional interest of the system. Moreover, such behaviour of the system, to some extent, encourages subjects to override the security mechanism in order to perform an access they believe they should be authorised to.

We have illustrated our framework with two simple, yet interesting examples. The first example presents the usage of the auto-delegation within an healthcare system, and the intuitive definition of the utility functions leads to the desired behaviour: as long as it is certain that a more qualified subject is available, no delegation will be given, which is consistent with the idea that an emergency situation should not automatically imply that unnecessary accesses can be granted. Hence, when there is no uncertainty, our approach is equivalent to enforcing directly the plain auto-delegation mechanism. However, if there is some uncertainty about the availability of a more qualified subjects, it is possible to grant the access to a less qualified subject, especially if the damage of not granting the access is huge. Thus, following the idea that some resources must remain accessible at all time. The second example illustrates that we can easily consider resource management problems within our framework.

The main drawback of our approach, and of most risk-based access control mechanisms, is the lack of precise utility, gain and/or damage measures for real-world applications. Concrete impact studies need to be conducted by practitioners in order to evaluate the consequences of allowing and denying accesses,

beyond the usual classification between "good" and "bad" accesses. However, this lack of measures can be also understood by the relative lack of concrete risk-based tools, which creates few incentive for practitioners to perform such studies. We believe that with the growing interest of the research community for risk-based security solutions, more useful measures will be found.

Finally, the framework we have proposed is quite fertile, and paves the way for several interesting extensions, such as the introduction of time in the calculation of the utility of an access and the widening of the set of decisions to be considered. Indeed, in practice, gain and damages are time-dependent, and it might increase the accuracy of our approach if we consider whether it is more interesting for the system to grant an access now to a subject, or to wait later for another, more qualified subject to be available. Concerning the widening of the decisions set, the use of mitigations and obligations policies may lower the damage of a less qualified subject, but also lower its gain, thus requiring the monitor to use a complex risk strategy.

References

1. C.A. Ardagna, S. Capitani Di Vimercati, T. Grandison, S. Jajodia, and P. Samarati. Regulating exceptions in healthcare using policy spaces. In *Proceeedings of the 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security*, pages 254–267, Berlin, Heidelberg, 2008. Springer-Verlag.
2. D. F. C. Brewer and M. J. Nash. The Chinese Wall Security Policy. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 329–339, May 1989.
3. A. D. Brucker, H. Petritsch, and A. Schaad. Delegation assistance. *Policies for Distributed Systems and Networks, IEEE International Workshop on*, 0:84–91, 2009.
4. A. Chander, J.C. Mitchell, and D. Dean. A state-transition model of trust management and access control. In *Proceedings of the 14th IEEE Computer Security Foundations Workshop*, pages 27–43. IEEE Computer Society Press, 2001.
5. L. Chen and J. Crampton. Risk-aware role-based access control. In *Proceedings of 7th International Workshop on Security and Trust Management*, 2011. To appear.
6. P.-C. Cheng and P.A. Karger. Risk modulating factors in risk-based access control for information in a manet. Technical Report RC24494, IBM T.J. Watson, February 2008.
7. P.-C. Cheng and P. Rohatgi. IT security as risk management: A research perspective. Technical Report RC24529, IBM T.J. Watson, April 2008.
8. P.-C. Cheng, P. Rohatgi, C. Keser, P. A. Karger, G. M. Wagner, and A. S. Reninger. Fuzzy multi-level security: An experiment on quantified risk-adaptive access control. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 222–230, 2007.
9. Computing Research Association. Four grand challenges in trustworthy computing, November 2003.
10. J. Crampton and C. Morisset. An auto-delegation mechanism for access control systems. In *Proceedings of 6th International Workshop on Security and Trust Management*, 2010. To appear.
11. G. Cybenko. Why johnny can't evaluate security risk. *IEEE Security and Privacy*, 4:5, January 2006.

12. N. N. Diep, L. X. Hung, Y. Zhung, S. Lee, Y.-K. Lee, and H. Lee. Enforcing access control using risk assessment. In *Proceedings of the Fourth European Conference on Universal Multiservice Networks*, pages 419–424, Washington, DC, USA, 2007.
13. D. F. Ferraiolo and D. R. Kuhn. Role-based access control. In *Proceedings of the 15th National Computer Security Conference*, pages 554–563, 1992.
14. Y. Han, Y. Hori, and K. Sakurai. Security policy pre-evaluation towards risk analysis. In *Proceedings of the 2008 International Conference on Information Security and Assurance*, pages 415–420, Washington, DC, USA, 2008. IEEE.
15. S. O. Hanson. Decision theory: A brief introduction, August 1994.
16. M. A. Harrison, W. L. Ruzzo, and J. D. Ullman. Protection in operating systems. *Communications of the ACM*, 19(8):461–471, 1976.
17. J. Kephart. The utility of utility: Policies for self-managing systems. In *Proceedings of Policies for Distributed Systems and Networks*, 2011. To appear.
18. L. Krautsevich, A. Lazouski, F. Martinelli, and A. Yautsiukhin. Influence of attribute freshness on decision making in usage control. In *Proceedings of the 6th International Workshop on Security and Trust Management*. Springer, 2010.
19. L. Krautsevich, A. Lazouski, F. Martinelli, and A. Yautsiukhin. Risk-aware usage decision making in highly dynamic systems. In *Proceedings of The Fifth International Conference on Internet Monitoring and Protection*. IEEE, 2010.
20. L. Krautsevich, A. Lazouski, F. Martinelli, and A. Yautsiukhin. Risk-based usage control for service oriented architecture. In *Proceedings of the 18th Euromicro International Conference on Parallel, Distributed and Network-Based Computing*. IEEE, 2010.
21. B. Lampson. Protection. In *Proceedings of the 5th Annual Princeton Conference on Information Sciences and Systems*, pages 437–443, Princeton University, 1971.
22. L.J. LaPadula and D.E. Bell. Secure Computer Systems: A Mathematical Model. *Journal of Computer Security*, 4:239–263, 1996.
23. Y. Li, H. Sun, Z. Chen, J. Ren, and H. Luo. Using trust and risk in access control for grid environment. In *Proceedings of the 2008 International Conference on Security Technology*, pages 13–16, Washington, DC, USA, 2008. IEEE.
24. R. W. McGraw. Risk-adaptable access control (RAdAC). available via `http://csrc.nist.gov/news_events/privilege-management-workshop/radac-Paper0001.pdf` on 16/08/09, 2007.
25. I. Molloy, P.-C. Cheng, and P. Rohatgi. Trading in risk: Using markets to improve access control. In *Proceedings of the 15th ACM New Security Paradigms Workshop, Lake TAhoe, CA, USA, September, 2008*, New York, NY, USA, 2008. ACM.
26. I. Molloy, L. Dickens, C. Morisset, P.-C. Cheng, J. Lobo, and A. Russo. Risk-based access control decisions under uncertainty. Technical Report RC25121, IBM T.J. Watson, September 2011.
27. Q. Ni, E. Bertino, and J. Lobo. Risk-based access control systems built on fuzzy inferences. In *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, pages 250–260, New York, NY, USA, 2010. ACM.
28. C. Skalka, X. S. Wang, and P. Chapin. Risk management for distributed authorization. *J. Comput. Secur.*, 15(4):447–489, 2007.
29. J. Wainer, P. Barthelmess, and A. Kumar. W-RBAC - a workflow security model incorporating controlled overriding of constraints. *International Journal of Cooperative Information Systems*, 12:455–485, 2003.
30. L. Zhang, A. Brodsky, and S. Jajodia. Toward information sharing: Benefit and risk access control (BARAC). In *Proceedings of the 7th IEEE International Workshop on Policies for Distributed Systems and Networks*, pages 45–53, Washington, DC, USA, 2006.