Hindawi
Security and Communication Networks
Volume 2018, Article ID 6476315, 11 pages
https://doi.org/10.1155/2018/6476315

Research Article

A Feasible Fuzzy-Extended Attribute-Based Access Control Technique

Yang Xu,[1] **Wuqiang Gao**,[1] **Quanrun Zeng**,[1] **Guojun Wang**,[2] **Ju Ren**,[1] **and Yaoxue Zhang**[1]

[1] *School of Information Science and Engineering, Central South University, Changsha 410083, China*
[2] *School of Computer Science and Educational Software, Guangzhou University, Guangzhou 510006, China*

Correspondence should be addressed to Yang Xu; xuyangcsu@csu.edu.cn

Received 29 December 2017; Revised 19 April 2018; Accepted 29 April 2018; Published 5 June 2018

Academic Editor: Debasis Giri

Attribute-based access control (ABAC) is a maturing authorization technique with outstanding expressiveness and scalability, which shows its overwhelmingly competitive advantage, especially in complicated dynamic environments. Unfortunately, the absence of a flexible exceptional approval mechanism in ABAC impairs the resource usability and business time efficiency in current practice, which could limit its growth. In this paper, we propose a feasible fuzzy-extended ABAC (FBAC) technique to improve the flexibility in urgent exceptional authorizations and thereby improving the resource usability and business timeliness. We use the fuzzy assessment mechanism to evaluate the policy-matching degrees of the requests that do not comply with policies, so that the system can make special approval decisions accordingly to achieve unattended exceptional authorizations. We also designed an auxiliary credit mechanism accompanied by periodic credit adjustment auditing to regulate expediential authorizations for mitigating risks. Theoretical analyses and experimental evaluations show that the FBAC approach enhances resource immediacy and usability with controllable risk.

1. Introduction

The burgeoning communication and computing technologies such as the 5G mobile Internet [1] and network computing [2–5] have substantially enhanced the availability and usability of resources to end users. Consequently, new evolutions including the popularity of telecommuting [6] and the general acceptance of "bring your own device" [7] have inadvertently driven the emergence of more complex and diverse resource access and usage scenarios. However, the developments in access control technologies have somewhat lagged behind. The typical role-based access control (RBAC) [8] model and older paradigms such as mandatory access control (MAC) [9] and discretionary access control (DAC) [10] are insufficient to support dynamic, distributed, and unpredictable access scenarios, because of their inherent limitations in flexibility, scalability, adaptability, and control granularity. More effective solutions that consider additional relevant parameters (e.g., subject states, object states, and contextual information) have also been explored, among which the attribute-based access control (ABAC) is the most promising approach for the new era. It has successfully transitioned from purely academic studies [11–20] to the practical application phase [21–24]. By enforcing attribute-formed policies on access requests, this adjustive, expressive, and highly extensible authorization model has an overwhelmingly competitive advantage, especially in dynamic and complicated environments.

Unfortunately, the ABAC ineluctably encounters practical problems during the use in current dynamic and complex scenarios spawned by the latest communication and computing techniques. Due to the rigid policy-based access control enforcement and the inability to automatically and efficiently handle exceptional access requests, some urgent requests which may not fully comply with the original ABAC policies would not be authorized in time due to the requirements of inefficient human involved approval processes, which impacts the resource availability and thereby affects the business timeliness and even leads to irreversible unfortunate consequences. There is a particular negative example that a world's top chip manufacturer once restricted its private cloud services only accessible by on-site staffs within the working hours for security purpose. Nevertheless, the staffs were easily frustrated in policy matching due to not only human factors but also technical reasons (The mobile positioning can be unsteady or outdated due to the functional defects or optimization reasons. Besides, the time limit obstructs lots of workflows in practice.). In absence of a flexible and efficient exceptional request handling mechanism, consequently, the working efficiency was severely affected as staffs could not get expected services in time when inefficient administrator involvements were often required for handling exceptional requests. Undoubtedly, the problem can be even worse in some time-sensitive cases, such as the sudden and urgent needs for classified information in stock or futures markets, the remote patient privacy data requirements in emergency surgeries, and the interorganizational confidential information requests in critical intelligence analyses.

Obviously, a more flexible and efficient exceptional access authorization method is badly needed by the stock ABAC paradigm to guarantee the business timeliness, especially for emergency situations, so as to make the ABAC more feasible, flexible, and adaptive for fitting current dynamic, distributed, unexpectable, and complicated situations.

In a sense, access control can be regarded as risk control. Therefore, the concept of risk and the opposite concept of trust have naturally been introduced as an effective and flexible assistive tool for the authorization decision-making process. For instance, the risk assessment method has already been integrated into classical models like RBAC and multiple levels of security (MLS) [25, 26]. By estimating the risk of the certain request based on the specific involving information and comparing the risk with some preset acceptance criteria of risk, these risk-oriented enhanced models have achieved flexible and efficient unattended authorizations for urgent requests which do not comply with the basic access rules in original models. More recently, risk and trust evaluation schemes are increasingly viable in access control when taking more parameters (e.g., environment states) into account, which yields more expressive and flexible solutions [27–31]. Because of these encouraging attempts, we are reasonably confident that the ABAC paradigm will benefit from risk evaluation schemes as well, especially the more flexible and efficient decision-making ability to deal with exceptional urgent access requests in dynamic and complex access environments. In this context, fuzzy logic [32], as one of the most recognized math tools for assessment that reasons probability from vague knowledge, is a viable option to determine the semantic matching degree of access requests and ABAC policies.

Focusing on the situations described above, in this article, we propose a feasible ABAC-based access control paradigm named fuzzy-extended ABAC (FBAC) to improve the flexibility and time efficiency when tackling low-risk exceptional authorizations for the emergency cases. We use the fuzzy assessment mechanism to evaluate the policy-matching degrees of requests failed to meet policies and then make authorization decisions according to both the denial threshold and the credit available to the requesters, to achieve unattended temporary authorization for the exceptional urgent access requests which are initiated by reputable users (reflected by credit values) but slightly violate the predefined ABAC policies. Furthermore, we designed an auxiliary credit system to impose restrictions on special authorizations and perform periodic credit adjustment auditing, to reduce the potential for abuse of expediential approvals. In addition, we describe a detailed case study to help readers understand the FBAC better and finally demonstrate our improvements from the perspectives of usability, security, and performance theoretically and experimentally.

The major contributions of our work are summarized as follows.

(1) We introduce the matching-degree-based fuzzy evaluation method into the original ABAC paradigm, which enables more efficient and flexible unattended approval for exceptional urgent authorization cases, to increase the resource usability and thereby the business timeliness.

(2) We keep the risk of special authorization abuse under control by not only using the configurable threshold to intercept high-risk requests directly but also by building a credit system combined with periodic credit adjustment audit mechanism.

(3) We analyzed the FBAC model theoretically for its usability, risk, and complexity and then implemented a prototype system to evaluate its effectiveness and efficiency by experiments, to demonstrate our enhancements in usability and immediacy, as well as the acceptance of security risks.

The remainder of this article is organized as follows. We introduce some articles related to our work in Section 2. In Section 3, we review several basic concepts of fuzzy logic. In Section 4, we propose our fuzzy-extended ABAC (FBAC) paradigm and detail it in the case study. Section 5 gives a brief discussion of FBAC's usability, risk, and complexity. Then in Section 6, we evaluate our prototype and analyze the experimental results. The last section summarizes this paper and describes possible improvements.

2. Related Work

Access control is an indispensable security technology for preventing sensitive resources from illegal access. A variety of access control models have been studied over the years, and different ones are designed for addressing discrete challenges focusing on confidentiality, integrity, scalability, manageability, etc. Some typical patterns like DAC [10], MAC [9], and RBAC [8] have emerged. Nonetheless, these classical models above are not expressive enough to take into account the effects of other additional factors (e.g., time of the day or user IP). As a result, they are gradually unable to meet the new requirements of geographical, temporal, and context-aware information systems.

Breaking the limitation of the subject-object pattern, more revealing access control paradigms are well studied.

One inspiring endeavor is bringing in risk factor to strike balance between system security and usability. The concept of "fuzzy" has been introduced to the RBAC for achieving better flexibility in handling exceptional requests [25]. The fuzzy RBAC carried out the more relaxed assignments of user-role and role-permission compared with the original RBAC model. And the assignment degrees were subjectively assigned to represent the accompanying uncertainties and risks of corresponding assignments. Then the access control enforcement was based on the risks of requests reflected by the overall assignment degrees. However, this conceptual solution did not provide a practical and detailed calculation method of assignment degrees. Cheng et al. [26] proposed the fuzzy MLS, a risk self-adjusting access control technique, which can quantify the potential risks associated with the exceptional access and thereby optimize the risk-benefit trade-off. In this model, the risk of the request was quantitatively assessed according to both the value of the object and the empirical illegal disclosure probability determined by the MLS tags (security level, etc.) of the involving subject and object and then made the access decision by comparing the risk with a preset risk scale and asking the user to provide corresponding risk tokens assigned by the administrator. Meanwhile, trust mechanism, closely connected to the concept of risk, has also been ushered in. Dimmock et al. expanded the existing access control framework and combined the trust-based assessment with reasoning to form a dynamic model that can manage risk more intelligently [27]. Liu combined the dynamic hierarchical fuzzy system with trust evaluation, then introduced a fuzzy multiattribute trust access control scheme for cloud manufacturing system [28]. Mahalle et al. [29] developed a trust-extended fuzzy authorization scheme and put forward the concept of trust rating for identity management. Context awareness is a significant precondition for accurately perceiving and properly handling risks. Feng et al. [30] integrated user behaviors and operating environment to propose a scalable trust-based and context-aware access control technique for large-scale, widely distributed networks. Taking into account both factors of trust and environmental perception, Bhatti et al. [31] constructed a trust-enhanced, environment-sensitive authorization model for network traffic based on X-GTRBAC (XML-based generalized temporal RBAC) framework.

As cross-organizational, multisectoral cooperations become integral parts of current business processes, to overcome the drawbacks of the mainstream access control models while unifying their advantages, there has been considerable interest in a more general model, namely ABAC [11, 12], which is considered as “next generation” authorization model for its dynamic, context-aware, and fine-grained features, defines a multidimensional access control paradigm where access requests are accepted or rejected based on all kinds of assigned attributes, including subject attributes (e.g., age, department, job title), action attributes (e.g., read, write, append), object attributes (e.g., owner, size, classification), and contextual attributes (e.g., time, location), and a set of policies. ABAC empowers more precise access control, facilitating the generation of expressive and flexible policies through the combination of a wide range of factors.

Determined attempts have been made not only by standards organizations [11] but also by many IT giants such as IBM and Cisco [21, 22], which contributes much to the development and widespread deployment of ABAC technique. Meanwhile, the academic community has also invested significant effort in this research area [13]. Li et al. [14] conducted in-depth discussions on the inherent logical relations and system architecture of ABAC. Jin [15] has formalized the ABAC scheme and achieved the simulation of other classical models. Sookhak et al. [16] carried out an exhaustive survey on ABAC techniques befitting cloud and distributed environment. Based on the authorization requirements of grid systems, Bo et al. [17] developed an efficient multipolicy ABAC technique suitable for grid computing based on the third-party authorization framework.

Regardless the benefits of ABAC, its rigid policy-enforcement mechanism as well as the guideless policy-configuration process may somehow lead to the reduction of resource usability and then the time efficiency of business. Demchenko and Ngo [18] mitigated this problem by proposing a specific ABAC solution for the cloud tenants which enables hierarchical delegations to support the efficient collaborations among tenants. Although this approach contributes to yield a more flexible ABAC paradigm, it is not a general solution which can only fit for limited scenarios. In a more intrinsic view, it reflects the fact that ABAC is thoughtless in how to efficiently deal with exceptional access requests.

Considering all these challenges and even more complex and urgent application scenarios, in our previous conference paper [19], we put forward a rough fuzzy ABAC framework conceptually aiming to achieve flexible special authorizations for exceptional urgent requests with low risks. However, it did not consider the effects of benign users' unintentional misoperations and ignored the differences in importance among attributes. Besides, its credit management mechanism is not reasonable enough while the experimental evaluation and analysis are not included. This research is inclined to make up for the past deficiencies so as to achieve an innovative approach with the auxiliary exceptional requests handling functionality, for enhancing the resource usability and thereby business timeliness in highly dynamic and unexpectable environments.

3. Preliminary

This section goes through some necessary concepts of the fuzzy theory [32].

Fuzzy Set. Fuzzy set is an extension of sets whose elements have degrees of membership. A fuzzy set can be defined as a pair (U, μ) in which U is the universe set of elements and μ is the membership function that mapping elements to corresponding membership degree, as follows:

$$x \in U \longrightarrow \mu(x) \in [0,1]. \tag{1}$$

Fuzzy Logic. The fuzzy logic is one type of multivalue logic which is based on fuzzy set theory. In fuzzy logic, the true/false value is replaced with membership values, which are real numbers between 0 and 1. A possible definition of operations in fuzzy logic is based on max/min function [33] in which the AND operator means taking the minimum value among membership values, while the OR operator means taking the maximum.

Table 1: The major notations and definitions.

Notations	Definitions
q_i	the ith request.
p_j	the jth clause in policy set.
$a_{j,k}$	the kth attribute involved in the clause p_j.
$w_{j,k}$	the weight of $a_{j,k}$ in the p_j.
$\xi_{j,k}(q_i)$	The fuzzy membership function for calculating the membership degree of the q_i to the constraint range of attribute $a_{j,k}$.
$\nu_j(q_i)$	The fuzzy membership function for calculating the membership degree of the q_i to the clause p_j.
$\mu(q_i)$	The fuzzy membership function for calculating the membership degree of the q_i to the policies.
$cost(q_i)$	The credit cost of special approval for the q_i
H	The rejection threshold (a rational number in $(0,1)$).
c_{max}	The credit-line (a rational number in $(0,1)$).
c_x	The credit value of the subject x (a rational number in $(0,c_{max})$).
r	The credit recover ratio (a rational number in $(0,1)$).

4. FBAC

In this section, we define several necessary notations at the beginning. Then, we introduce the architecture of FBAC briefly and describe its workflow step by step. Further, we demonstrate its essential components in detail. And finally, we study a detailed case to help readers understand the FBAC better.

For convenience, we only adopt granting policies (Although the policies in ABAC can be granting or denying ones, they are mutually transformable.) in this paper and employ a refusal precedence principle for the decision-making process; i.e., a granted decision would be made when the request meets at least one clause in the policy set.

```
**Input:** $q_i$, $c_x$
**Output:** $Decision \in \{\textbf{granted}, \textbf{denied}\}$
**if** match any policy **then**
    **return granted**
**end if**
$\mu(q_i) \leftarrow \max_{i=1}^{n}(\nu_i(q_i))$
$cost(q_i) \leftarrow 1 - \mu(q_i)$
**if** $\mu(q_i) < H$ **or** $c_x < cost(q_i)$ **then**
    **return denied**
**end if**
$c_x \leftarrow (c_x - cost(q_i))$
**return granted**
```

Algorithm 1: The FBAC Decision-Making Procedure.

4.1. FBAC Model. The FBAC model wraps the standard ABAC as a preliminary screening module and integrates additional decision support components for improving the resource usability, thereby gaining better business timeliness.

Notations. Throughout this paper, we use the notations in Table 1 for simplified description purpose.

Architecture and Workflow. As seen in Figure 1, the FBAC is built upon the standard ABAC model with additional fuzzy evaluation component and credit component. The first component is developed to support unattended special authorizations, while the second is a security remedial measure. These additional components are independent to standard ABAC which contributes to the effortless integration.

When a request is reached, the FBAC firstly collects the states of related attributes of that request, including the attributes of subject, object, context, and action (Steps 1-2). After applying the policies, if this request is not granted by the standard ABAC process, it will be delivered to our fuzzy evaluation component for a further decision based on the membership degree calculation and the rejection threshold filter (Step 3). The credit component will check the available credits of the requester and denies the request if the requester is unable to afford the credit cost for approving this exceptional request (Step 4). If the corresponding subject has sufficient credits to pay the credit cost, the credit component will issue a prompt to ask the requester to confirm the credit consumption (Step 5). Once confirmed by the requester, the request will be granted and logged, at the expense of corresponding credit consumption. Note that part of the consumed credit will be restored after audit if the subject is not malicious. Otherwise, this request will be denied (Step 6). The final decision is delivered to the enforcement facility which will mediate the corresponding access to the object accordingly (Step 7). The major decision-making process is illustrated in Algorithm 1.

Apart from the major decision-making process, there is an audit process which will router the recorded exceptional access authorizations to administrators for review periodically. And then the credit audit system will restore a part of the users' credit according to the auditing results (Step a).

Fuzzy Evaluation Component. When a request q_i is rejected by the standard ABAC module because it can not exactly match any policy, the FBAC system will turn to fuzzy evaluation component for further judgments. This component will evaluate the matching degree of the q_i to policies through membership degree calculation. Specifically, for the

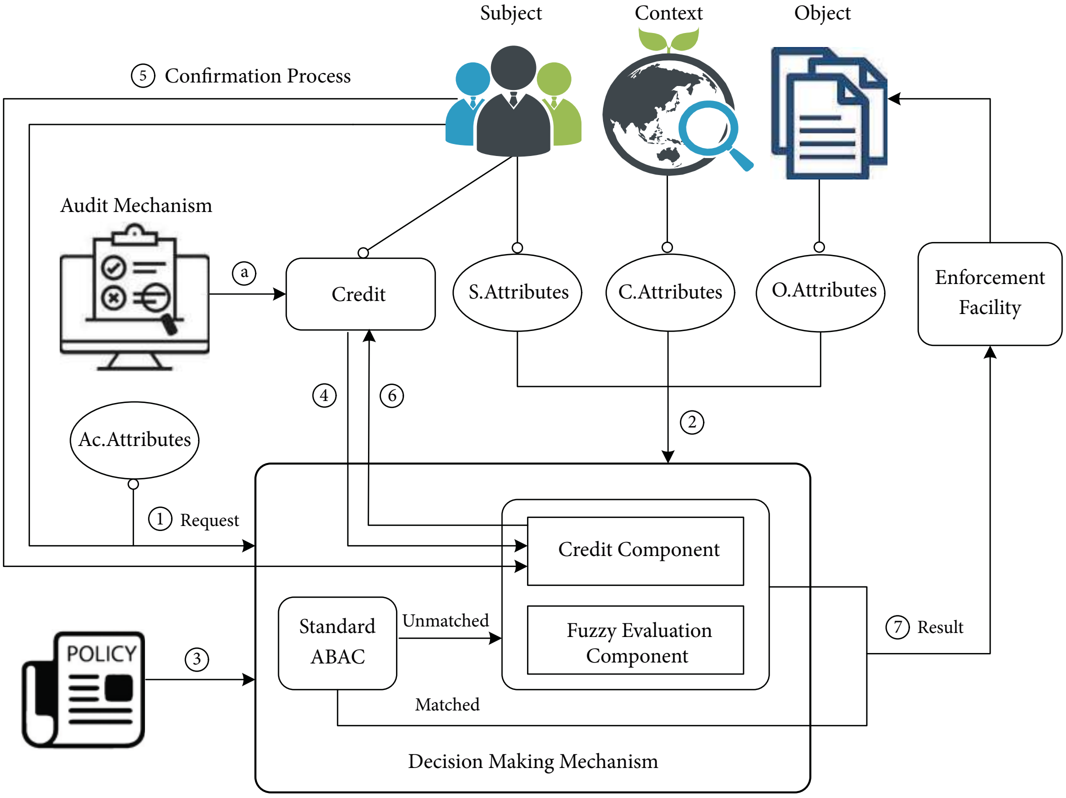


Figure 1: Architecture and workflow of FBAC.

jth clause in the policy set, this component will calculate the membership degree of the request q_i to that clause as follows:

$$\nu_j(q_i) = \frac{\sum_{k=1}^{n} w_{j,k}\xi_{j,k}(q_i)}{\sum_{k=1}^{n} w_{j,k}}. \tag{2}$$

In formula (2), $\xi_{j,k}(q_i)$ is the membership subfunction that maps q_i to a certain membership degree according to the matching degree of q_i to the constraint range of the kth attribute in the jth clause. The design of $\xi_{j,k}$ is closely related to the meaning of the corresponding attribute and policy clause and also depends on administrators subjectively. There exist several primary guidelines for determining the membership subfunction [34]. And the most commonly recommended function templates include the trapezoid subordinate function, the trigonometric membership function, the step function, etc. In this paper, we select the trapezoid subordinate function and the step function for different policy clauses respectively (cf. Section 4.2). The FBAC gives the administrators greater freedom to determine the attributes which should be fuzzy processed based on practical administrative needs. In general, the continuous attributes can be fuzzy processed, while the discrete ones (e.g., users names) should be fully matched for obtaining final authorizations. Additionally, if the discrete attributes can be somehow transformed into continuous ones based on partial ordered relations, they can also be fuzzy processed similarly, e.g., converting the discrete and hierarchical job titles to continuous level numbers. $w_{j,k}$ is the weight of the corresponding attribute. Introducing weight factor enables administrators to adjust the influence of each attribute in the policies, so as to provide more flexible and expressive manageability.

Since there usually exist more than one clause in the policy set, the holistic matching degree is synthesized with maximum synthesis rules [33], as shown in the following formula:

$$\mu(q_i) = \max_{j=1}^{n} \nu_j(q_i). \tag{3}$$

After obtaining the matching degree $\mu(q_i)$, the FBAC will compare $\mu(q_i)$ with the rejection threshold H. If $\mu(q_i) < H$, the request q_i will be denied by FBAC. Otherwise, the credit component will be invoked for supporting further judgments.

Credit Component and Audit Mechanism. The fuzzy evaluation component provides users with extra access opportunities without manual reviews. However, in spite of the benefits in the resource usability and business timeliness, this fuzzy evaluation module poses potential threats such as abuse issues unintentionally. Therefore, we build a credit component combined with periodic credit adjustment auditing mechanism as the countermeasure to mitigate the risk of abuse.

Our credit component maintains a credit value c_{x_*} ($c_{x_*} \in [0, c_{max}]$, where $c_{max} \in (0, 1)$ is the preset credit line) for each subject x_*. When the FBAC is initialized, every c_{x_*} will be set as c_{max} without discrimination. During the use, the credit component will be invoked to provide further decision support for the request q_i if its matching degree $\mu(q_i)$ exceeds the rejection threshold H. We define $cost(q_i) = 1 - \mu(q_i)$ as the special approval cost for the request q_i with the matching degree $\mu(q_i)$, because the $cost(q_i)$ can reflect the gap between the states of the q_i and the precise requirements of policies. Thus, the credit component will compare the credit c_x of the requester x with the corresponding special approval cost $cost(q_i)$. If $c_x < cost(q_i)$, then a denial suggestion will be issued for the q_i as the requester does not have enough credit to afford the cost. Otherwise, the FBAC will ask the requester for confirmation to consume that $cost(q_i)$ and enforce the requester to comment reasons for the unusual request. This additional prompt scheme is quite useful to avoid user misuse and is also helpful for future audits. Then if the requester x replies in the affirmative to that credit consumption prompt, the FBAC will grant the request q_i by charging the requester corresponding fee, i.e., deducting $cost(q_i)$ from c_x. In fact, for individuals, the FBAC would degrade to standard ABAC when they max out their credits.

Furthermore, for achieving better credit management and thereby controlling credit abuse risks, a periodic manual audit mechanism is also integrated into the FBAC model. During an audit, the unusual authorization records will be reviewed by the system administrators according to all the relevant information in the system including corresponding explanatory comments typed by requesters in the confirmation process. Based on auditing results, the audit routine will restore credits for the users who pass checks successfully, while disables such recovery for the suspects unless proved innocent (More tougher punishments can be given when the suspect is finally proven guilty.), to ensure the credit system works well, thereby providing enough flexibility with controllable abuse risks.

Note that the credit recovery strategy depends on the administrator. For instance, our approach gives the proportional credit back (r in 100%) of the margin between the credit line c_{max} and the current credit value c_x (i.e., $c_{max} - c_x$) after each audit process. This is because we hold a conservative opinion that the special approval is a compromise for improving business timeliness, which should not be encouraged in routine work. Therefore, the formula for calculating new credit value c_x' is as follows:

$$c_x' = r\left(c_{max} - c_x\right) + c_x, \quad \text{where } r \in (0, 100\%]. \tag{4}$$

4.2. Case Study. This subsection provides a case study of FBAC to help people understand how it works in detail.

Assuming there exists an FBAC system with the threshold $H = 0.8$, $c_{max} = 0.3$, $r = 0.5$ and two clauses in the policy set as follows:

$$policy: \begin{cases} \text{(1) IF } (location = (112.54153E \pm 0.00001, 28.95117N \pm 0.00001)) \\ \quad \text{and } (job\ title \text{ is } manager) \text{ THEN granted} \\ \text{(2) IF } (location = (112.54153E \pm 0.00001, 28.95117N \pm 0.00001)) \\ \quad \text{and } (time \in [8:00, 18:00]) \text{ and } (job\ title \text{ is } staff) \text{ THEN granted} \end{cases} \tag{5}$$

We can see that there are 3 types of attributes involved in the policy set: *time* is the timestamp of the request, *location* denotes the requester's location (given in latitude and longitude), and *job title* denotes the *subject*'s job position. Then we define the membership functions as follows:

$$\begin{aligned} \mu\left(q_i\right) &= \max\left(\nu_1\left(q_i\right), \nu_2\left(q_i\right)\right) \\ \nu_1\left(q_i\right) &= \frac{\sum_{j=1}^{2} w_{1,j}\xi_{1,j}\left(q_i\right)}{\sum_{j=1}^{2} w_{1,j}} \\ \nu_2\left(q_i\right) &= \frac{\sum_{j=1}^{3} w_{2,j}\xi_{2,j}\left(q_i\right)}{\sum_{j=1}^{3} w_{2,j}} \end{aligned} \tag{6}$$

In this case, we set all the attributes in the same policy to the same weight, as shown below:

$$\begin{aligned} \nu_1\left(q_i\right) &= \frac{\sum_{j=1}^{2} \xi_{1,j}\left(q_i\right)}{2} \\ \nu_2\left(q_i\right) &= \frac{\sum_{j=1}^{3} \xi_{2,j}\left(q_i\right)}{3} \end{aligned} \tag{7}$$

In order to describe $\xi_{i,j}$, we firstly predefine a function $distance(x, y)$ to describe the distance between x and y in meters. Then, we give the definitions of $\xi_{i,j}$ as follows:

$$\begin{aligned} \xi_{1,1}\left(q_i\right) &= \max\left(1 - \frac{distance\left(location, office\right)}{100}, 0\right) \\ \xi_{1,2}\left(q_i\right) &= \begin{cases} 1 & job\ title \text{ is } manager \\ 0 & \text{otherwise} \end{cases} \\ \xi_{2,1}\left(q_i\right) &= \xi_{1,1}\left(q_i\right) \\ \xi_{2,2}\left(q_i\right) &= \begin{cases} 2 \cdot time - 16, & time \in (7.5, 8] \\ 1, & time \in (8, 18] \\ 37 - 2 \cdot time, & time \in (18, 18.5] \\ 0, & \text{otherwise} \end{cases} \\ \xi_{2,3}\left(q_i\right) &= \begin{cases} 1, & job\ title \text{ is } staff \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{8}$$

Then we assume that a subject S initiates a request q_1 as follows:

$$q_1 = \left\{ \begin{array}{c} time = 18:35 \\ job\ title = manager \\ location = (112.54180E, 28.95117N) \end{array} \right\} \quad (9)$$

When request q_1 is initiated, the FBAC attempts to match q_1 with policies but fails. Then it turns to the fuzzy evaluation process. As the credit cost of the q_1 is $cost(q_1) \approx 1 - 0.85 = 0.15$, then 0.15 is going to be consumed from c_S for making q_1 be granted. The system will ask subject S for the consumption confirmation in order to make sure whether S is willing to consume required credits to continue. Suppose that S chooses to spend his credits, then q_1 is granted, and c_S is decreased to 0.15.

Next, when S try to initiate another request q_2 later as follows:

$$q_2 = \left\{ \begin{array}{c} time = 23:03 \\ job\ title = manager \\ location = (112.54187E, 28.95117N) \end{array} \right\}, \quad (10)$$

in the same way, we get that $cost(q_2) \approx 0.19$. Since $c_S = 0.15$ after the request q_1, S can not afford the cost of the q_2, so q_2 will be rejected directly.

In addition, if S passes the audit with his credit value $c_S = 0.15$, then c_S will be restored to 0.225 according to expression (4).

5. Discussion

In this section, we will briefly analyze the effect on usability and security of FBAC, followed by complexity analyses.

Usability and Security. To describe the enhancive effect on the overall resource usability of FBAC, we chose the granted rate, which is defined as the rate of the granted requests to total requests per unit time, as a reflection of usability.

Let U denote the usability and R denote the granted rate; then we get the following expression in which R_{normal} and $R_{special}$ denote the granted rates of requests matching or not matching policies, respectively, while notation "$\propto$" denotes the relationship of positive correlation.

$$U \propto \left(R = R_{normal} + R_{special} \right) \quad (11)$$

Since FBAC shares the same R_{normal} with its elder sibling ABAC obviously, the FBAC obtains extra usability improvement ΔU which is positively correlated with $R_{special}$ when compared with ordinary ABAC, namely,

$$\Delta U \propto R_{special} \quad (12)$$

Naturally, the configurable threshold H is closely associated with the usability. For any request q_* failed in policies matching with overall matching degree $\mu(q_*)$, we suppose that $\mu(q_*) = x$ obeys a probability density distribution $f(x)$ while the probability of available credit of requester $c_* \geq \mu(q_*)$ obeys another probability density distribution $h(x)$, then we can deduce the following relational expression:

$$R_{special} \propto \int_{H}^{Max} h(x) f(x)\, dx \quad (13)$$

Since $h(x)$ and $f(x)$ are commonsensically positive, we find an inverse correlation between the incremental usability ΔU and the threshold H in expression (13); that is, a lower H leads to more approvals on requests. Apparently, the FBAC would deteriorate to standard ABAC if H tends to the upper bound, i.e., the value 1 in our case.

Not surprisingly, the usability improvement also comes with security risks. As the FBAC may authorize exceptional access requests which do not fully comply with the current policies in some cases, this feature can be abused by indiscreet users or even be exploited by malicious users for accessing extra resources and thereby bringing additional risks to the system. Here, the deviation between the overall matching degree of the exceptional request (i.e., $\mu(q_*)$) and the closest matching policy (the standard normalization value "1") is used as the risk indicator of each exceptional authorization.

Correspondingly, the FBAC has effective countermeasures to mitigate the risks induced by the fuzzy assessment mechanism to the acceptable level. Firstly, as a general and indiscriminate defense, the reject threshold is used to screen out high-risk requests deviating far from current policies, i.e., any request q_* with overall matching degree $\mu(q_*)$ lower than the threshold H would be declined directly, because the FBAC is aiming at improving the flexibility and efficiency of exceptional authorizations rather than invalids the security policies. Thus, the security risk of each exceptional authorization is limited within the controllable range $1 - H$. Secondly, the credit mechanism is used as the individualized constraint against the abuse attacks on the FBAC. As for each requester, each exceptional authorization definitely comes with corresponding credit cost which is determined by the risk of that request q_* (i.e., $cost(q_*) = 1 - \mu(q_*)$). In other words, a request q_* will be declined if the corresponding requester x_* does not have enough credit to afford the credit cost $cost(q_*)$ of the exceptional request, i.e., $c_{x_*} < cost(q_*)$. Therefore, the immoderate and even malicious exceptional access behaviors are mitigated due to the limitation of credit. According to the analysis above, then the maximum security risk of one exceptional authorization associated with a requester x_* is further limited within Minimum($1 - H$, c_{x_*}). Meanwhile, within each audit cycle, the total security risk which can be caused by the exceptional authorizations related to each single requester x_* is limited below his credit value c_{x_*} (the value at the beginning of the audit cycle). In addition, for each subject x_*, the credit consumption has the additive restrictive effect on future requests because only a portion of the already consumed credits could be restored according to credit recovery mechanism. Briefly, the more credits the requester used in one audit cycle, the less total amount he will have in the future, which further reduces the abuse risks of the exceptional authorizations. Finally, the FBAC integrates a periodic manual audit mechanism as

Table 2: The parameter configuration.

Case	C_{max}	r	H	Time weight	Location weight
1	0.80	0.50	0.80	0.50	0.50
2	0.80	0.50	0.85	0.50	0.50
3	0.80	0.50	0.90	0.50	0.50
4	0.80	0.50	0.80	0.40	0.60
5	0.80	0.50	0.80	0.20	0.80

the post-security mechanism to review all the exceptional authorizations. As for the suspects, their credit restorations would be suspended until proven innocent. As a result, they would lose the privileges to obtain instant approvals for their exceptional requests as their credits will keep reducing and can not get replenished. Therefore, the entire risk which can be caused by the exceptional authorizations granted for a single suspect identified during the audits is limited within the credit line c_{max}.

Summarily, the FBAC broadens the granting bounds to a certain extent for all the requests with the help of fuzzy evaluation mechanism and limits the special approval rate of each individual requester with the help of credit and audit mechanism, thereby achieving better timely usability than standard ABAC with the controllable sacrifice of security.

Complexity. The complexity of access control is related to the number of concurrent requests, policies, and attributes contained in each policy. The more the attributes are involved in a policy, the higher the computational complexity of this policy will be. Generally, as the granularity of access control becomes finer, the complexity of policy increases and the time cost of decision-making process also grows slightly and tends to flatten out.

Assuming there are m policies and n attributes, the number of requests that occur at the same time in the system is k, the computational complexity of a basic matching process is $O(1)$ in original ABAC model. In the worst case, each policy and attribute needs a matching calculation, and thus the complexity of a single decision is $O(mn)$. Because complexity is proportional to the number of requests made simultaneously, the total computational complexity of the whole system is $O(kmn)$.

Correspondingly, the computational complexity of both a basic matching process and credit evaluation process in our FBAC model is also $O(1)$; that is to say, the complexity of a single decision is still $O(mn)$; thus the total computational complexity remains at $O(kmn)$.

Compared with the standard ABAC model, our FBAC model has two additional processes, the credit-based judgment and the fuzzy assessment, which is a little complex than the simple yes/no decision. And the overhead of both parts can be considered of the same order of magnitude as the former. This explains why both models (i.e., ABAC and FBAC) have the same computational complexity. It also shows that the impact of FBAC in terms of performance is within an acceptable range.

6. Experimental Evaluation

We developed an FBAC prototype to evaluate its availability, security, and performance through several experiments.

6.1. Test Scenarios. By modifying the ABAC source codes of Deter Project [35], we implemented a prototype of FBAC and deployed it to 5 virtual servers on a single physical machine (64-bit CentOS 7, 4vCPUs (i5-7500 3.4GHz), 16GB RAM, 1TB Storage, supported by OpenStack (Pike v3.12.0)) for experiments.

In our FBAC systems, we firstly configured the following policy set and set the audit time interval to one week uniformly.

$$policy : \begin{cases} \text{IF } (location = (112.54153E \pm 0.0001, 28.95117N \pm 0.0001)) \\ \text{and } (time \in [8:00, 18:00]) \text{ THEN granted} \end{cases} \tag{14}$$

And then we conducted four experiments with respective FBAC configuration parameters shown in Table 2. And in each experiment, we simulated 500 users to initiate requests to FBAC servers. These users follows Poisson distribution in time and move around according to Random Way Point (RWP) [36] model to fit the mobile features. The simulation system will randomly regenerate the destination and the moving speed for each user every 30 minutes. Additionally, we also introduced small noises (±10m) randomly to users' location coordinate data for simulating the fluctuations in the real positioning system. These users were set as "benign" or "malicious" separately with several different user behavioral patterns correspondingly to generate requesting data. Furthermore, we set that benign users will abort their requests randomly in responding to credit misuse prompts whereas malicious users will not, according to the knowledge that benign users are more compliance with rules.

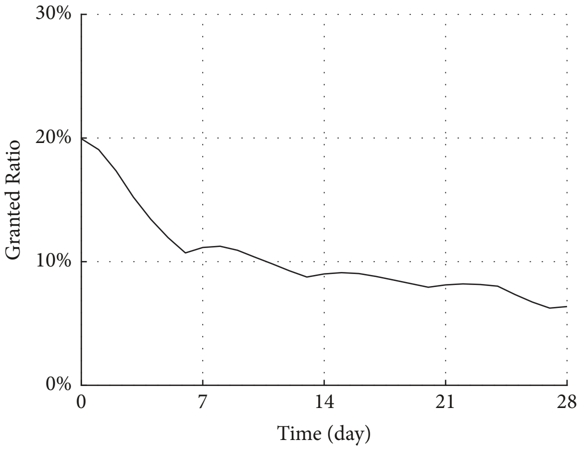


Figure 2: The average granted ratio of requests.

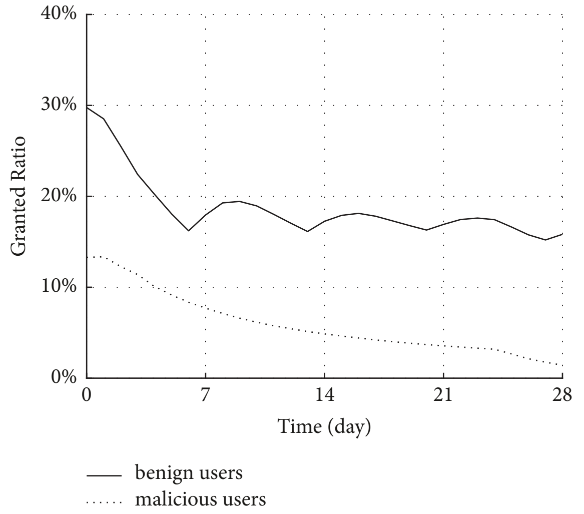


Figure 3: $R_{special}$ of benign and malicious users.

Note that in the fourth and fifth cases, we forced all the users to obey the time restriction to articulate the effect of attribute weights.

The experiments last for four weeks and each audit period is 5 days long. All the access histories are recorded in access logs for further analyses.

6.2. Analysis

Usability. As the granted ratio of requests which fail to meet policies (denoted by $R_{special}$) reflects the extra improvement on immediate resource usability, we count up such average granted rate based on the Case 1, as shown in Figure 2. We can learn that the average granting rate of exceptional requests is maintained in a positive range during the experiment, which illustrates the usability increment of FBAC compared with ABAC through the employment of fuzzy evaluation method.

Security. Again, based on Case 1, we evaluated the resistance of FBAC against security risks. Figure 3 shows the granted ratios of both benign and malicious user respectively. It is clear that $R_{special}$ of benign users is limited to a certain upper bound by the threshold, particularly, below 35% in Case 1, while that of malicious users is even far lower throughout the test duration. Furthermore, it also illustrates that such rates of both benign and malicious users are further constrained by credit mechanism. With the consumption and partial recovery of credits controlled by credit and audit mechanism, $R_{special}$ of benign users reveals a hysteretic declined trend within each audit cycle and will fluctuate along with audit cycles during the testing period. When it comes to malicious users, this ratio is decreasing continuously over audit cycles and is gradually converging to 0.

Such results demonstrate that the threshold provides a general and coarse-grained restriction on requests while credit system supplies additive restrictive effect on the requests in each audit cycle. In addition, the audit mechanism is effective in limiting $R_{special}$ of users with malicious or abnormal behaviors as their credits will be used up easily and can hardly be restored because of the audit mechanism. Therefore, the FBAC is sufficient to defend against abuse attacks.

Table 3: The time cost of the decision-making process.

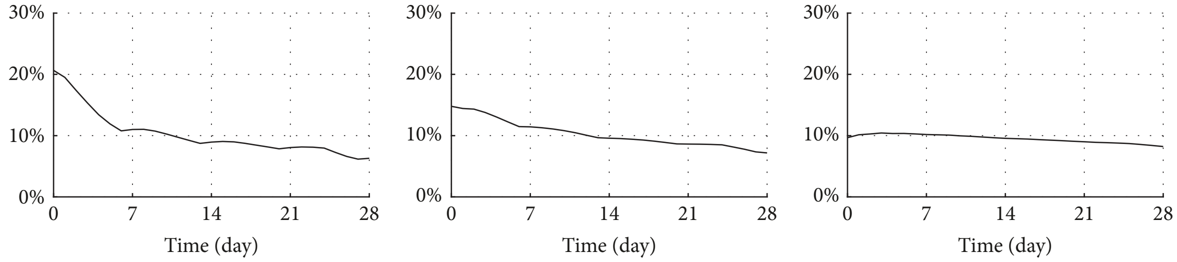
Model	Average time (ms)	Best time (ms)	Worse time (ms)
FBAC	0.033	0.019	0.245
ABAC	0.017	0.002	0.081

Parameter Effects. We have tuned two major regulative parameters in FBAC to explore their potential influence.

(1) Threshold. To study the impact of the reject threshold, we increased the threshold H by 0.05 in Case 1, Case 2 and Case 3 gradually. Unsurprisingly, Figure 4 illustrates that $R_{special}$ in FBAC is closely related to the threshold H; i.e., the higher H is, the lower the granted rate will be. Besides, although a low H may accelerate the credit consumption, which in turn affects the granted rate due to the rejection cases caused by credit insufficiency, this side effect is unable to impact the main trend on a macroscale.

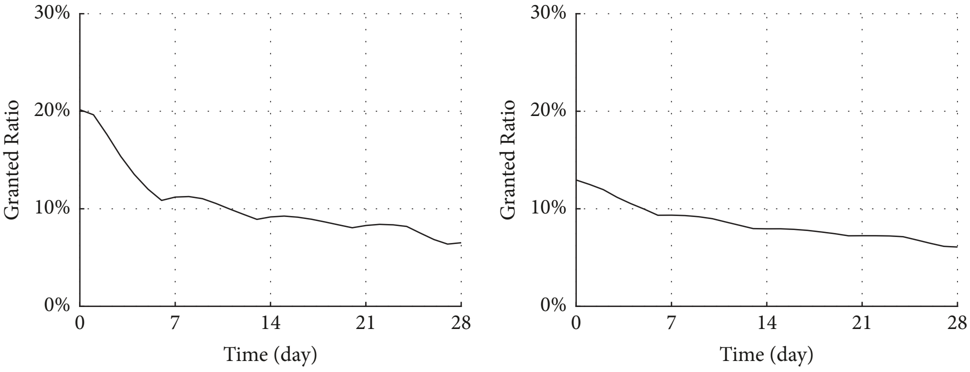
(2) Attribute Weight. When it comes to the attribute weight, s Cases 4 and 5 were selected for comparison as they set the time variable to fixed value by obeying the time restriction and share the same C_{max} and H parameters. As seen in Figure 5, the bigger weight coefficient for the location attribute in Case 5 leads to a lower granted rate when compared with that of Case 4. This shows that the weight mechanism can effectively adjust the overall impact of each attribute on the decision-making process.

Performance. We evaluated the time cost of decision-making processes of both FBAC and ABAC to measure the performance. According to the results in Table 3, although FBAC wraps ABAC and adds additional mechanisms for making authorization decisions, it only incurs quite light and



(a) $H = 0.80$ (b) $H = 0.85$ (c) $H = 0.90$

Figure 4: $R_{special}$ under different thresholds.



(a) $w_t = 0.4$, $w_l = 0.6$ (b) $w_t = 0.2$, $w_l = 0.8$

Figure 5: $R_{special}$ under different attribute weights.

acceptable overhead in average compared with ABAC, which is almost imperceptible to requesters.

7. Conclusion

In this paper, a feasible FBAC technique is proposed that improves upon the standard ABAC paradigm with good flexibility and time efficiency in dealing with exceptional urgent requests which do not comfort to policies in the dynamic and unpredictable environment. Beyond ABAC, we use a fuzzy evaluation method to do unattended special authorizations for exceptional requests that failed in policy matching. We also use credit and corresponding audit mechanisms to limit the abuse risk of special approvals. A tangible example is given to explain the working details, which indicates the suitability of FBAC in mobile and dynamic scenarios. In addition, the theoretical analyses and experimental evaluations show that the FBAC paradigm reinforces the system in favor of time efficiency and usability with the controllable expense of security.

In future work, we would like to further refine the authorization decision-making scheme with the support of the latest deep learning techniques (e.g., neural network) to discover benign and riskful access patterns based on the access behavior mining for helping the FBAC better distinguish between benign and malicious requests, thereby enabling more intelligent and accurate handling for exceptional access cases. Moreover, we also believe that deploying the FBAC system in China's current Xiangya medical big data system would have more practical and exploratory meanings.

Disclosure

This work was presented in part at the SpaCCS 2017, Guangzhou, China, 12–15 December 2017.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 61702562 and 61472451, the Mobile Health Ministry of Education-China Mobile Joint Laboratory, the Hunan Provincial Innovation Foundation for Postgraduate under Grant CX2015B047, the China Scholarship Council Foundation under Grant 201506370106, the Guangdong Provincial Natural Science Foundation under Grant 2017A030308006, and the Joint Research Project between Tencent and Central South University.

References

[1] G. Fettweis and S. Alamouti, “5G: personal mobile internet beyond what cellular did to telephony,” *IEEE Communications Magazine*, vol. 52, no. 2, pp. 140–145, 2014.

[2] Y. Zhang, K. Guo, J. Ren, J. Wang, and J. Chen, “Transparent computing: A promising network computing paradigm,” *Computing in Science Engineering*, vol. 19, no. 1, p. 20, 2017.

[3] Y. Zhang, J. Ren, J. Liu, C. Xu, H. Guo, and Y. Liu, “A survey on emerging computing paradigms for big data,” *Journal of Electronics*, vol. 26, no. 1, pp. 1–12, 2017.

[4] J. He, Y. Zhang, J. Lu, M. Wu, and F. Huang, “Block-Stream as a Service: A More Secure, Nimble, and Dynamically Balanced Cloud Service Model for Ambient Computing,” *IEEE Network*, vol. 32, no. 1, pp. 126–132, 2018.

[5] T. Peng, Q. Liu, and G. Wang, “A multilevel access control scheme for data security in transparent computing,” *Computing in Science & Engineering*, vol. 19, no. 1, Article ID 7802524, pp. 46–53, 2017.

[6] I. Hardill and A. Green, “Remote working - Altering the spatial contours of work and home in the new economy,” *New Technology, Work and Employment*, vol. 18, no. 3, pp. 212–222, 2003.

[7] A. M. French, C. Guo, and J. P. Shim, “Current status, issues, and future of bring your own device (BYOD),” *CAIS*, vol. 35, pp. 1–10, 2014.

[8] D. F. Ferraiolo, R. S. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, “Proposed NIST standard for role-based access control,” *ACM Transactions on Information and System Security*, vol. 4, no. 3, pp. 224–274, 2001.

[9] S. Upadhyaya, “Mandatory access control,” in *Encyclopedia of Cryptography and Security*, pp. 756–758, Springer, 2011.

[10] L. Liu and M. Tamer Özsu, “Discretionary access control,” in *Encyclopedia of Database Systems*, pp. 864–866, Springer, 2009.

[11] V. C. Hu, D. Ferraiolo, R. Kuhn et al., “Guide to Attribute Based Access Control (ABAC) Definition and Considerations,” National Institute of Standards and Technology NIST SP 800-162, 2014.

[12] V. C. Hu, D. R. Kuhn, and D. F. Ferraiolo, “Attribute-based access control,” *The Computer Journal*, vol. 48, no. 2, Article ID 7042715, pp. 85–88, 2015.

[13] D. Servos and S. L. Osborn, “Current research and open problems in attribute-based access control,” *ACM Computing Surveys*, vol. 49, no. 4, article no. 65, 2017.

[14] X. Li, D. Feng, Z. Chen, and Z. Fang, “Model for attribute based access control,” *Journal on Communications*, vol. 29, no. 4, pp. 90–98, 2008.

[15] X. Jin, *Attribute-based access control models and implementation in cloud infrastructure as a service [Ph.D. thesis]*, The University of Texas at, San Antonio, 2014, Ph.D. dissertation.

[16] M. Sookhak, F. R. Yu, M. K. Khan, Y. Xiang, and R. Buyya, “Attribute-based data access control in mobile cloud computing: Taxonomy and open issues,” *Future Generation Computer Systems*, vol. 72, pp. 273–287, 2017.

[17] B. Lang, I. Foster, F. Siebenlist, R. Ananthakrishnan, and T. Freeman, “A flexible attribute based access control method for grid computing,” *Journal of Grid Computing*, vol. 7, no. 2, pp. 169–180, 2009.

[18] C. Ngo, Y. Demchenko, and C. De Laat, “Multi-tenant attribute-based access control for cloud infrastructure services,” *Journal of Information Security and Applications*, vol. 27-28, pp. 65–84, 2016.

[19] Y. Xu, W. Gao, Q. Zeng, G. Wang, J. Ren, and Y. Zhang, “FABAC: A flexible fuzzy attribute-based access control mechanism,” in *Proceedings of the Proc. 10th International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*, pp. 332–343, Springer, 2017, pp. 332-343.

[20] X. Liu, Q. Liu, T. Peng, and J. Wu, “Dynamic access policy in cloud-based personal health record (PHR) systems,” *Information Sciences*, vol. 379, pp. 62–81, 2017.

[21] IBM Corporation, https://www.ibm.com/support/knowledgecenter/en/SSNGTE_7.0.0/com.ibm.tspm.doc_7.0/install/concept/AttributeBasedAccessControl.htm.

[22] “Cisco Systems, Inc,” https://www.cisco.com/c/en/us/td/docs/security/asa/asa97/asdm77/firewall/asdm-77-firewall-config/virtual-access-vm-attributes.pdf.

[23] Axiomatics, https://www.axiomatics.com/.

[24] Jericho Systems Corporation, https://www.jerichosystems.com/technology/glossaryterms/attribute_based_access_control.html.

[25] C. Martnez-Garca, G. Navarro-Arribas, and J. Borrell, *Fuzzy role-based access control*, vol. 111 of *Information Processing Letters*, Elsevier, 2011, pp. 483-487.

[26] 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 (SP ’07)*, pp. 222–230, IEEE, Berkeley, Calif, USA, May 2007.

[27] N. Dimmock, A. Belokosztolszki, D. Eyers, J. Bacon, and K. Moody, “Using trust and risk in role-based access control policies,” in *Proceedings of the Proceedings on the Ninth ACM Symposium on Access Control Models and Technologies, SACMAT 2004*, pp. 156–162, usa, June 2004.

[28] Y. Li, *The research of access control mechanism based on attiubute and trust evaluation , Masters thesis [Master, thesis]*, Southwest Jiaotong University, 2016.

[29] P. N. Mahalle, P. A. Thakre, N. R. Prasad, and R. Prasad, “A fuzzy approach to trust based access control in internet of things,” in *Proceedings of the Proc. 3rd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace Electronic Systems (VITAE*, pp. 1–5, 2013.

[30] F. Feng, C. Lin, D. Peng, and J. Li, “A trust and context based access control model for distributed systems,” in *Proceedings of the Proc. 10th IEEE International Conference on High Performance Computing and Communications*, pp. 629–634, 2008.

[31] R. Bhatti, E. Bertino, and A. Ghafoor, “A trust-based context-aware access control model for web-services,” *Distributed and Parallel Databases*, vol. 18, no. 1, pp. 83–105, 2005.

[32] F. J. Pelletier, “Metamathematics of fuzzy logics by Petr Hajek,” *Bulletin of Symbolic Logic*, vol. 6, no. 3, pp. 342–346, 2000.

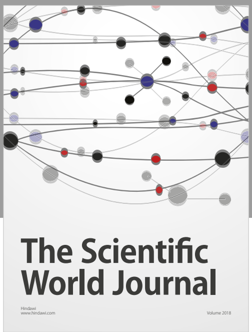
[33] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller,” *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, 1975.

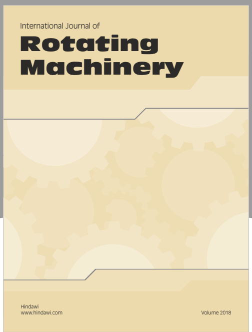
[34] J. Dombi, “Membership function as an evaluation,” *Fuzzy Sets and Systems*, vol. 35, no. 1, pp. 1–21, 1990.

[35] DETER Project, https://abac.deterlab.net/.

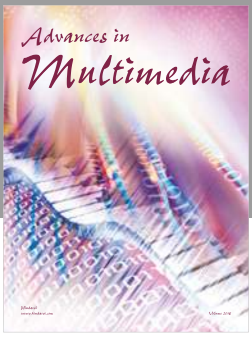
[36] D. Johnson and D. Maltz, “Dynamic source routing in Ad Hoc wireless networks,” in *The Kluwer International Series in Engineering and Computer Science*, vol. 353, pp. 153–181, 1996.

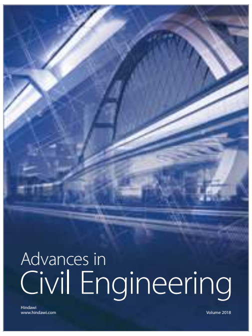
Journal of
Engineering

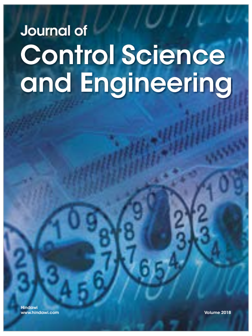

The Scientific
World Journal


International Journal of
Rotating
Machinery

Journal of
Sensors

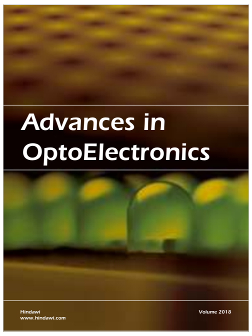

Advances in
Multimedia


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

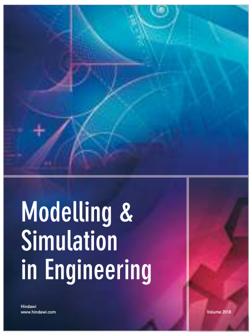
Journal of
Robotics

Journal of
Electrical and Computer
Engineering


Advances in
OptoElectronics

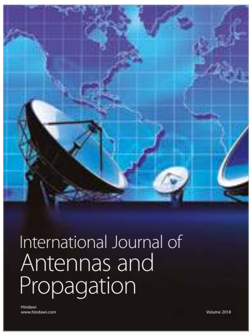
VLSI Design

International Journal of
Navigation and
Observation

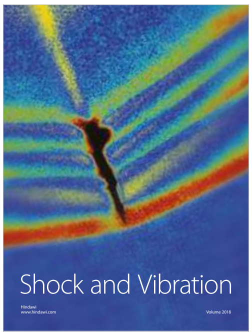

Modelling &
Simulation
in Engineering

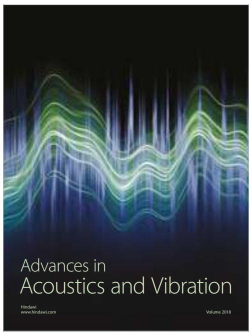
International Journal of
Aerospace
Engineering

International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration