Survey on Access Control Mechanisms in Cloud Computing

Gözde Karataş[1] and Akhan Akbulut[2]

[1]*Department of Mathematics and Computer Science, Istanbul Kültür University, Istanbul, Turkey*
[2]*Department of Computer Engineering, Istanbul Kültür University, Istanbul, Turkey*
E-mail: g.karatas@iku.edu.tr; a.akbulut@iku.edu.tr

Received 15 February 2018; Accepted 12 April 2018;
Publication 26 May 2018

Abstract

The benefits that Internet-based applications and services have given to the end user with today's cloud computing technology are very remarkable. The distributed services instantly scaled over the Internet provided by cloud computing can be achieved by using some mechanisms in the background. It is a critical task for end users to control access to resources because lack of control often leads to security risks. In addition, this may cause systems to fail. This paper describes seven different access control mechanisms used in cloud computing platforms for different purposes. Besides, the advantages and disadvantages of various models developed from previous service-based architectures and used for cloud computing are detailed and classified. During the assessments, NIST's metrics were taken as a reference, and in the study, 109 articles from the past decade were examined. We also compared our research with the existing survey papers.

Keywords: Cloud Computing, Information Policy Making, Access Control, Security, NIST Metrics.

Journal of Cyber Security and Mobility, Vol. 7_3, 1–36. *River Publishers*
doi: 10.13052/jcsm2245-1439.731

1 Introduction

Access control systems have various attributes in order to perform access control principles. The largest advantage of these systems is having user features, which are functions that are appropriate for the procedures performed, methods, and administration characteristics rather than individually performing the procedures in different fields. Thus, the priorities of the systems can be easily determined [1, 2]. In addition, the procedures of performing or denying the determined priorities according to a set of rules can be performed. These rules are set in access control policies. These policies also need to be in accord with the priorities. When evaluating prioritization of access control, five different entities must be considered [3, 4].

Access controls provide the following benefits:

- Allows the user to system security.
- It allows the user or system authentication.
- It makes network security monitoring.
- Ensures that the above items are processed at the same time.

For the access control mechanisms to be implemented, the policies are determined by the right and decent control rules and the prioritization, which are basic components of these policies, are needed.

In this article, the approaches that have been used for the access control mechanism are classified into categories. The second section presents the classification according to the operation techniques [3–5]. In addition, the access control mechanisms that have been used in older generation service-based architectures and adapted to current conditions, the approaches that are designed according to the dynamic structure owned by cloud computing are also discussed. National Institute of Standards and Technology (NIST) standards were used as the basis for the comparisons between the systems, and the details are discussed in the third section.

The contribution of this paper is threefold:

- We make a comparative explanation regarding the access control mechanisms present in cloud computing recorded in 109 research papers published in the past decade.
- Unlike other survey studies, each access control mechanism is evaluated by using NIST metrics.
- A comparison of existing survey studies about access control is provided.

2 Preliminaries

In this section, we discuss cloud computing, its components, and features before getting into detail regarding access control mechanisms.

2.1 Cloud Computing

Cloud computing is being presented as the sixth generation service based model [1, 3]. Since this model has low equipment-software cost, improved performance, fast updating, extendable data storage, high data security, easy adaptation; this technology hosts flexible and scalable services. In other words, high resources can be reached instantaneously; can regulate resource consumption at the level of need. On this occasion, when there is a situation where high resource use is required, instant demand can be met [6]. It also has inner mechanisms that effect the service quality of the services that are presented on the cloud. Sub-elements such as security manager, service manager, provision manager and access control manager perform different roles and tasks and perfect operation process of the cloud. Every subsystem is responsible for its own field of duty and each one has dynamically different ways of working. In Cloud Computing, security is still a risky issue. Even if your resources are not attacked, you can be impacted by attacks on other resources in the cloud infrastructure [6]. Real life cloud computing examples are; Office 365 is an infrastructure that runs on MS Office cloud, which is branded in Microsoft's office software, with the OpenStack, a cloud computing service running on standard hardware has become available.

2.2 Cloud Computing Essential Characteristics

The essential characteristics help providers to determine if it is really cloud computing or not [3, 7]. These characteristics are as follows:

On-Demand Self-Service: All services are managed by the end-users. Therefore, the cloud does not support by the service provider.

Broad Network Access: The cloud has very wide access capability because service can be accessed over a network with different devices.

Resource Pooling: The use of resources is dynamic and there are many resources available, such as storage disk capacity.

Rapid elasticity: Services are limitless and can be scaled as customer demand increases.

Measured Service: The principle of "pay as much as you use" is adopted.

2.3 Cloud Computing Models

Cloud computing provides services according to three basic models [1, 7, 8]:

Software as a Service(SaaS): This model provides web-based enterprise or end-user-oriented software as a service with current versions. Additionally, the customer cannot control or manage the cloud infrastructure. Examples of SaaS model are Facebook, Office 365, Google Gmail.

Platform as a Service(Paas): This model provides the environment for the development of new applications. Similar to SaaS, the customer cannot control the primary cloud infrastructure. Examples of PaaS model are Microsoft Azure, Amazon Web Services (AWS), 3tera.

Infrastructure as a Service(IaaS): This model provides infrastructure environment. It is a tool that offers standardized data storage and other computing capabilities over the network. Examples of IaaS Amazon EC2, Rackspace, Google Compute Engine.

2.4 Cloud Computing Service Models

Cloud computing provides developers the ability to focus on what is important. As cloud computing has grown, different models and strategies have developed for user needs. Each type of these services provides different levels of control [1, 7, 8]. The following are the types of strategies to use the cloud computing service models:

Private Cloud: Cloud providers cannot be hosted from the outside for security reasons. It is mainly executed by the organization itself or by another third party company. Since the Private Cloud is a personally constructed building, all the information is at the user's disposal. For example, an international company can maintain the information processing needs of its branches in all world countries on a single cloud.

Public Cloud: Cloud services are available for public use for everyone and are generally owned by large companies. These services are free or priced by usage. Generally, public cloud providers like Microsoft, Google, etc., process their own infrastructure resources and access is only provided via the Internet.

Community Cloud: The cloud computing infrastructure is shared by specific organizations. Community members have access to applications and data.

Hybrid Cloud: Combination of several or all of the cloud service models. The purpose is to combine cloud services to create a well-managed and automated computing environment. In this type of cloud the owner can move data and applications between private and public clouds.

2.5 Cloud Computing Attacks

Since the use of cloud computing increased and companies are moving towards cloud computing, attacks are also increasing. The attacks on cloud computing are listed as follows [9]:

Denial of Service Attacks: Attacker overloads the cloud system with service requests.

Cloud Malware-Injection Attack: Attacker tries to inject malicious service into the cloud system.

Side Channel Attacks: Attacker attempts to compromise the cloud system by placing a malicious virtual machine.

Authentication Attacks: Attacker targets the mechanisms that are used to secure the authentication process.

Man in the Middle Cryptographic Attacks: This attack is performed when an attacker places himself between two users and it is based on the weakness of the Address Resolution Protocol, which is used to communicate the MAC addresses of the OSI layer at level 2. The attacker enters the network between the gateway and the computer designated as the victim and constantly broadcasts the ARP attack as if the receiver were itself. As a result of this broadcast, the victim uses the attacker's computer as the computer gateway. Attacker can read, delete or change all messages that come and go between users.

2.6 Cloud Computing Limitations and Disadvantages

Cloud computing has many benefits for users and enterprises like reducing costs, increasing flexibility, and providing easy handling. However, in addition to the advantages of cloud computing, it also provides the following risks [7]:

Security and privacy: The ease in providing and accessing cloud services can give malicious users the ability to detect and exploit vulnerabilities. In multi-tenant cloud architecture, an attacker might try to break into the other users.

Downtime: Cloud computing makes applications dependent on the Internet connection. When network service is down, the application is down as well. If Internet service is slow or inconsistent, cloud computing may not be suitable for an application.

Platform dependencies: Hosting and integrating current cloud applications on another platform may unveil some issues.

Limited control and flexibility: Cloud computing users have limited control over the functions and usage of their infrastructure.

3 NIST Metrics

NIST, founded by the United States of America, is the institute that determines the scaling methods and standards that are valid worldwide. Access control systems must be evaluated with these standards before being selected. These metrics provide a better view of control features to determine whether the model is appropriate for the system or not. The following points are the criteria that we used to examine and evaluate the controls of the subsystems in the third part [10]:

1. *Auditing:* Shows whether or not the access control carries characteristics related to the system journal (log).
2. *Privileges/Capabilities Discover:* Manages the administration members and answers the question of substantiation of the operations that are performed for these members by creating groups.
3. *Ease of Privilege Assignments:* It is related to the easiness of the operations that are carried out in the system.
4. *Syntactic And Semantic Support For Specifying AC Rules:* Provides the answer to the question of whether semantic or syntactic comparisons can be carried out while settling upon important decisions.
5. *Management Complexity:* The principles that are determined for every model need to be decided according to the access privileges.
6. *Delegation of Administrative Capabilities:* The access control can grant the authorized people the right to delegate their authorities to other people.
7. *Configuration Flexibility:* Deals with whether the system has substantial scalability/flexibility characteristics for a good performance.
8. *Horizontal Scope of Control:* It checks whether the system is in accord with other platforms-mechanisms.
9. *The Vertical Scope of Control:* Deals with whether the system is in accord with different applications-operating systems.
10. *Policy Combination, Composition, And Constraint:* Deals with the ability of the access system to combine rules and contents of different policies with certain limitations.
11. *Bypass:* Examines the possibility to skip the critical situations by the rules that are set for the access control of the system.
12. *Least Privilege:* Shows which user can obtain the requested result.
13. *Separation of Duty:* Demands the restriction of access to the system against the case of any security leak.

14. *Safety:* Determines if the access control has the security controls that will check the security leaks in the given permissions.
15. *Policy Conflict:* Determines if there may be conflicts in the decisions regarding the access permissions that will be given according to demands. A procedure needs to be implemented that creates solutions to these conflicts immediately.
16. *Operational/Situational Awareness:* Deals with the situational awareness of the system where the system decides to give access by evaluating the principle that would be applied.
17. *Enforcement Mechanism:* Implements the operations in the system when the access permissions are verified according to the permission requested.
18. *Expression (policy/model) properties:* Examines the compatibility of the system with some programming languages that are given.
19. *Adaptability:* Deals with how well the system is adaptable to the changing principles and situations as time progresses.
20. *Policy Import and Export:* Gives information regarding if the system has appropriate members that allow the system to operate on the policies.
21. *OS compatibility:* Deals with the compatibility of the system with other operating systems other than the one that it is based on.
22. *Policy Source Management:* Regulates or maintains the resources in the Access Control Policies.
23. *User Interfaces and API:* Determines if the system has a user interface such as GUI or API for the controls of the accesses or operations.
24. *Verification and Compliance Function Support:* Deals with operations the system carries out to verify or test the access rules.

4 Access Controls and Evaluation

This section explains the details about different access control mechanisms for Cloud Computing such as; Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), Discretionary Access Control (DAC), Mandatory Access Control (MAC), Fine-Grained Access Control (FGAC), Hierarchical Attribute-Based Access Control (HABE) and Attribute-Based Encryption Fine Grained Access Control (ABE FGAC).

Role Based Access Control

Role Based Access Control (RBAC) is an access control mechanism that surfaced in the 1970s. According to RBAC, the users are assigned to different roles, and the necessary permissions, limitations and authorizations

are performed because of these roles [11–14]. The general structure of the RBAC is the following: with role searches permission assignment (PA), the privileges in the system of every role are determined and the administration permissions are assigned to these roles based on the user assignment (UA), permissions and limitations of the user from the role the user [15–19]. If we were to examine the case from the perspectives of PA and UA; the user can be assigned to more than one role, and a role can have more than one user. RBAC structure is as shown in the Figure 1. A hierarchical structure is built with the role designations. Thus, a user coming to the system is assigned to the roles according to status, the requested access field, and the requested administration privileges [20, 21]. Besides, similar to how it happens in other systems, rather than making the permissions and limitations work all the time, these are called into action if the user needs them via the role. Furthermore, the users have the right to change their roles, they may want to be assigned to a role of higher rank or may want the implementation of limitations to their rights and limit their roles [14, 15, 22]. RBAC, which provides many conveniences to the user regarding its usage, also has many beneficial features. When a new role needs to be defined, this role can be obtained from the other roles by examining them or the users can be assigned to roles and these roles assign according to the priorities of the system, thus granting more flexible control and administration rights [23–26]. One of the other most important features of RBAC is that it does not leave the decision of an operation that is really important to the system to only one user; instead, it decides according to the requests from many users. Another advantage is that it has the appropriate flexibility to combine the large and complex systems [27–30].

Because of the role assignments that are used in RBAC, the users can be assigned from one role to another much easier and the given permissions can be linked. Additionally, the providing or receiving of the certain permissions can be carried out much faster if necessary. For this reason, RBAC is used

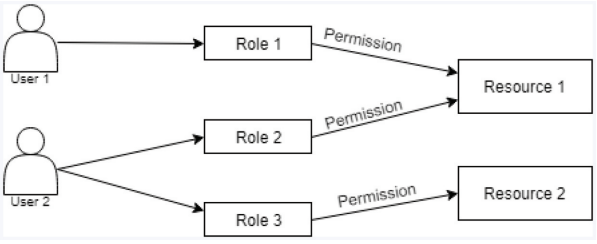

Figure 1 RBAC Structure.

more in the fields in which business and marketing applications are chosen to be used, and applying the privacy-aware access control model to the systems that are developed with RBAC is very easy. The role usage is for the benefit of the businesses, and as a result, a safer and consistent way is adapted in developing the private security policies [31–34].

The administration power is not user or group based, but it is role based [35, 36].

Review: As the system priorities are substantiated with PA in RBAC, it has an intermediate-level feature called *Privileges/Capabilities Discover.* Since the permissions and limitations are determined according to the roles, the *Ease of Privilege Assignment* is in high level. As it does not make critical decisions according to certain users and roles and as it has flexible control and administration features, the *Syntactic and Semantic Support For Specifying AC Rules* is in intermediate level. As it is accepted as a secure system, the transfer of the priorities to other people is not possible. This means the low-level *Delegation of Administrative Capabilities* is high in *Vertical Scale* because of Horizontal Scale. Since it does not have a feature to combine different policies, the *Policy Combination, Composition, And Constraint* is at low levels. As all the controls are substantiated with roles and priority transfers are not allowed, it is an access control that can be accepted as secure. Additionally, as it prompts the permissions and the limitations according to needs, it has a high level of *Operational/Situational Awareness.* As it does not find the system changes odd, it has a high level of *Adaptability.* As it develops the roles genetically from each other, it has an intermediate level of *Policy Import and Export.* As it has user interfaces such as API, it has an intermediate level of *User Interfaces and API.* The access rights are determined according to roles and the access to rights since the roles are not altered according to the working status of the system. Additionally, as it employs the necessary operations when the permissions related to the roles are verified, it has an intermediate level of *Least Privilege, Separation of Duty,* and *Enforcement Mechanism.* Since the access rights to this control are distributed via the roles, determining the access rights according to certain principles is difficult; thus, it has a low level of *Management Complexity.* Furthermore, there are not any procedures to consider any permission conflicts other than the roles; thus, the system conflicts, the *Policy Conflict*, is at high levels. As RBAC has a certain scalability and flexibility, it can be made compatible with API with certain arrangements, and this means that it has a high level of *Horizontal Scope* and *Configuration Flexibility.*

Attribute Based Access Control

Unlike RBAC, the user controls (access permissions) are ensured with attributes and not roles. Thus, in the usage of Attribute-Based Access Control (ABAC), the user attributes play an important role. These attributes can access the general characteristics of the user such as age, height, personal characteristics, and can be altered again according to this information [37–40]. The determination of the attributes is performed according to the topics. As the attributes are related to the information entered, they can be grouped into the following three categories: subject attributes (such as personal information), Resource attributes (information about the outcomes), and environment attributes (information about the environment) [41]. In this manner, the complexities that are rising in the growing and complicated systems are solved. Similar to RBAC, ABAC calls the permissions and limitations to action when they are needed. Additionally, logical comparisons, and controls are important features [37, 42, 43]. The set of user attributes will be preserved separately as shown in Figure 2.

Two models have been developed for ABAC. The first one is the model that defines the policies – the policy model-, and the other one is the architecture model that provides the web service access control [44–47]. The attributes in these models are determined by the experts and they are dynamically altered constantly. In this manner, complying with the changing access control decisions become easier.

The most convenient systems that can use ABAC is the Open and Distributed systems. It substantiates the reaching of these systems to the flexible and trustable access control technology much faster [20]. However,

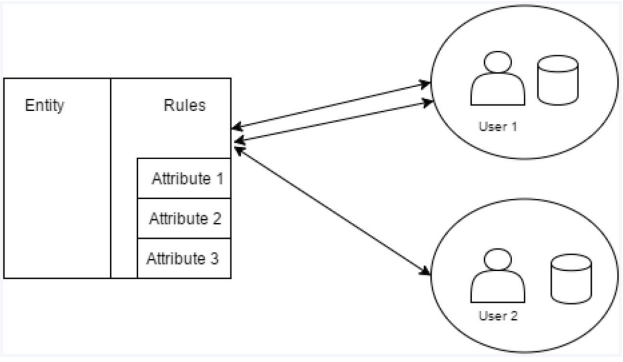

Figure 2 ABAC Structure.

this flexibility increases the occurrences of policy conflicts and makes the maintenance and administration of the policies difficult [48–51]. After a time period, the developers decided that ABAC is not efficient in system control because the access permissions given according to the attributes of the users became insufficient and unproductive. As a result, the Attribute and Role Based Access Control (ARBAC) model was implemented as the combination of ABAC and RBAC. According to the ARBAC, the system consists of four important elements and the privacy policy operations are performed via these elements [52–54]. Unfortunately, there have been no experimental studies on ABAC [55, 56].

Review: As all the operations are performed according to the attributes, the *Ease of Privilege Assignments* is at high levels. As it gives great importance to logical comparisons, it has a high level of the *Syntactic and Semantic Support For Specifying AC Rules.* It does not allow authority assignments, which RBAC does, and this means a low level of *Delegation of Administrative Capabilities.* As the attribute designation depends on the environment as well as a couple of features, different operating systems can affect the application, which means an intermediate level of *Vertical Scope* and *OS compatibility*. Two different models for two different fields were developed in order to not need different policy or model combinations. Thus, it has a low-level *Policy Combination, Composition, And Constraint.* As it is a trustable and flexible system, the *Safety* is at high levels. Additionally, as it has flexibility and the attributes are altered dynamically, it has a high level of *Adaptability.* As a result of the disadvantage of the flexibility, unlike it is advantage, it has a low level of *Policy Source Management.* As it has user interfaces such as API, it has an intermediate level of *User Interfaces and API.* As the access rights are granted according to the attributes similar to those in RBAC, the *Least Privilege* is in intermediate levels. As the access rights are determined according to attributes and the access rights are presented according to certain principles, the *Separation of Duty and Management Complexity* is at high levels. The verification processes are performed according to attributes and because it has access rights. As a result, the system does not allow the operation and function to be included according to the outcome. This means an intermediate level of *Enforcement Mechanism.* As the control does not have any procedures that can be employed in the cases of certain security leaks and access conflicts, the *Policy Conflict* is at high levels. As it is an easy control, it can be adapted to other mechanisms by introducing new arrangements with adding new parameters, therefore, it has a high level of *Horizontal Scope.*

The biggest deficiency of the control is the fact that the rights, decisions, and administrative processes are performed according to attributes and they do not have a certain flexibility. Therefore, it has a low level of *Configuration Flexibility.*

Discretionary Access Control

Discretionary Access Control (DAC) is an access control that was developed by Graham and Dennig, and it provides the basis of the security systems, which the DAC structure is as shown in the Figure 3. It is highly preferred in areas where computer security is important [11]. The unit that assigns the authorities is the person who is the owner of the objects called Owner; thus, he also determines the security principles. The owner gives the necessary authorization to individuals in the system, introduces limitations, and limits their access to the system according to his own will [12, 51]. Although the most important feature of this system is the fact that it has a high level of security, it cannot distinguish between the subjects and object domains and has security leaks [57–60]. This access control is based on the user or group control. While assigning permissions and limitations to users, instead of working on a single user, it chooses to define groups that are controlled by the owner and introduces certain permissions and limitations to them. Of course, this situation can cause serious security leaks in the case that the owner is not trustworthy [4, 61, 62]. The most important feature that can cause problems with the system is that

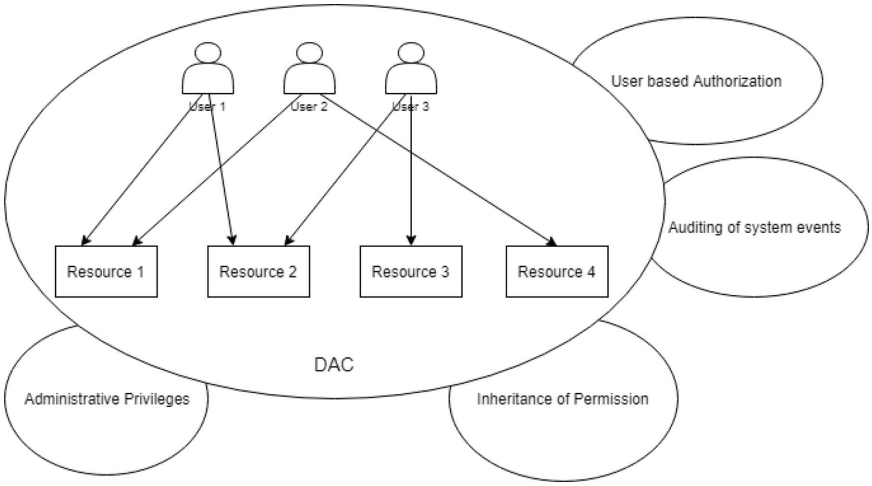

Figure 3 DAC Structure.

the owner can transfer his authority to someone else. Another problem is the information leaks; in other words, as the control policies are determined according to desires without checking the information of the users by an owner, it does not distinguish between object or subject domains. This prevents it from being able to carry out logical comparisons [61, 63]. Another important feature of the control is that it has a really high flexibility when compared to other controls [60, 62].

Review: As stated in the description, DAC carries out the assignments by mostly creating groups. This means a high level of the *Privileges/Capabilities Discover.* Also, as DAC is a system that is owner based, it is easy to understand system and can be operated very easily; thus, it has a high level of *Ease of Privilege Assignments.* However, as being owner based can hinder the processes of logical comparisons, it has a low level of the *Syntactic and Semantic Support For Specifying AC Rules.* As it allows the owner to transfer his authorities to someone else, it has a high level of *Delegation of Administrative Capabilities.* As it is one of the best controls in the matter of security, it is highly chosen to be used in operating systems. It has an intermediate level of *Vertical Scope* and *OS Compatibility.* As it has a certain level of flexibility but carries out the assignments group based, there can be problems in different policy combinations. This means an intermediate level of *Policy Combination, Composition, And Constraint.* In addition, as the access rights in the control are determined by an owner, a low level of *Least Privilege* is existent. Despite the fact that it is one of the highest ranks of the controls regarding its security, the access rights in the systems are controlled by the person who determines the principles in the system. Thus, it has an intermediate level of *Separation of Duty*, *Management Complexity*, and *Safety*. As DAC has the ability to determine which operations to perform in certain situations, it has an intermediate level of *Operational/Situational Awareness.* The user interfaces are practical and easy to use. It is also compatible with easy to understand interfaces such as API and GUI; DAC has an intermediate level of *User Interfaces and API*. It also is faithful to the decisions of the system owner very much; thus, it does not feel the need of any verification control. This means a low level of *Verification and Compliance Function Support*.

As it includes some operations that in a way that will not cause security threats and on some scale when the permissions are verified, the *Enforcement Mechanism* is in intermediate level. It does not have the procedures that are employed in the cases of permission conflict or encounter an unwanted situation; thus, it has a high level of *Policy Conflict*. As its flexibility and

scalability are high, it adapts itself to other mechanisms easily. It has a high level of *Horizontal Scope*. As mentioned in the description, one of the most important features that distinguish this control from other ones is the fact that it has a high flexibility. This means a high level of *Configuration Flexibility*.

Mandatory Access Control

Mandatory Access Control (MAC) surfaced when the experts decided that DAC is not efficient in terms of security since it is not being able to control every piece of information, which the Working model of the MAC scheme is shown in Figure 4. The decisions in MAC are not made by an owner but a central system [11, 12]. In this way, more powerful processes are performed in security models [57, 64, 65]. The wholeness and privacy of the system are the most important features after security. Thus, it can increase the security to the highest level in a whole system. Its flexibility is really low [63]. The reason for that is the system security, however, even this does not ensure the absolute privacy. It is used in government and military system improvements as a result of these features [66–69]. Also SElinux has a MAC mechanism that enables the security protocol to reduce the level of control over objects [70].

While MAC performs operations, it does not consider the relationship it has with the users into consideration, and for the security grant that ensures that all the users in the operating system perform the user assignments are considering this fact. It allows even the system administrative to perform operations according to the introduced policies by implementing limitations [63, 70, 71]. The most important feature that values this system above DAC is the fact that MAC distinguishes between the subject and object

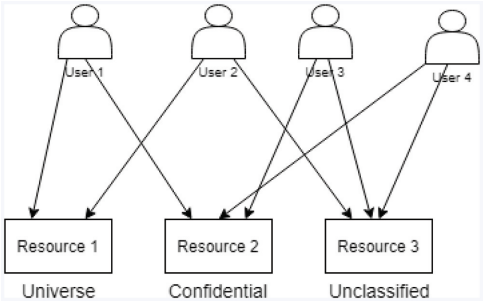

Figure 4 MAC Structure.

domains and allows the usage of permissions and limitations accordingly. Thus, it is much better in terms of logical comparisons when compared with DAC [72–74].

Review: As the permissions and limitations are substantiated according to the user, it has a low level of *Privileges/Capabilities Discover*. Although the access and usage possibilities are not that good, MAC is easy to understand and use; thus, it has an intermediate level of *Ease of Privilege Assignments* [72–74]. As MAC can do certain distinctions, it has an intermediate level of *Syntactic And Semantic Support For Specifying AC Rules*. As the assignments are carried out by a central system, no user can transfer their rights to another user, which contributes to a low level of *Delegation of Administrative Capabilities*. Every user has a unique policy it means a special policy for each user is prepared in the system and the combination process of these policies is not allowed. Thus, it has a low level of *Policy Combination, Composition, And Constraint*. The most important feature of the system is its high security; thus, a high level of *Safety* is present. MAC is aware of the operations that are performed but does not make its decisions user based, which means Allows the system to make decisions on the basis of the permissions defined for the user thus, an intermediate level of *Operational/Situational Awareness*. Additionally, the decisions are made by a central system and cause the creation of strict rules, which creates a hard adaptation process in time. MAC has a high level of *OS Compatibility* because it does not have any compatibility issues. As it has a powerful user interface and control mechanism, it has an intermediate *User Interfaces* and *API* and *Verification* and *Compliance Function Support*. Unlike DAC, the decisions are made by the central system; thus, the obtaining of the results is healthier and the partner that is given the access rights can be assigned according to the decision. Thus, MAC has an intermediate level of *Least Privilege* and *Management Complexity*. As the control researches, grants rights, and manages the system more in detail, there are few security leaks and the *Separation of Duty* is increased. Similar to DAC, MAC allows the operations to be included with certain scaling and this means an intermediate level of *Enforcement Mechanism*. Although it tries to close the security gaps with high-security measures, as it does not have a procedure to be implemented in the cases of certain access conflicts, its *Policy Conflict* is in the intermediate level. Unlike DAC, MACs largest deficiency is it does not have flexibility and it cooperates with other systems in strict rules [63]. Although it has an intermediate *Horizontal Scope* as a result of the operations that will be done via arrangements and implementing parameters, its *Configuration Flexibility* is at low levels.

Fine Grained Access Control

In various access controls, the logic behind Fine Grained Access Control (FGAC) was used to introduce developments, which the FGAC structure is as shown in Figure 5. The advantage of this mechanism compared to other ones is that it has the flexibility to determine the access rights and in which mode they can work for each user. This became the reason for the system policies to be more understandable. Another advantage of FGAC is that the users have the right to arrange their own policies and determine who can reach their own information [70, 75–81]. Until FGAC was introduced, many access control mechanisms were using an all-or-nothing approach. However, this approach is insufficient in meeting the situations that are important for most users. Thus, FGAC helped many systems develop a new approach. Over and above, because FGAC is not seen as a sufficient access control mechanism, different hybrid models were developed with the aim of making this control more productive. [75, 76, 82–84]. However, this control system also has disadvantages. One of the most important problems is that it does not create problems in the systems that work on their own. Additionally, FGAC has problems in identifying and distinguishing between the operations produced when cooperating with complex structures. Another important problem is that when the operations should be updated according to the changing environment needs, identifying the policies in a semantic manner and formalizing them, applying the policies in a secure manner, and analyzing the policies in a coherent manner becomes much harder [85–88].

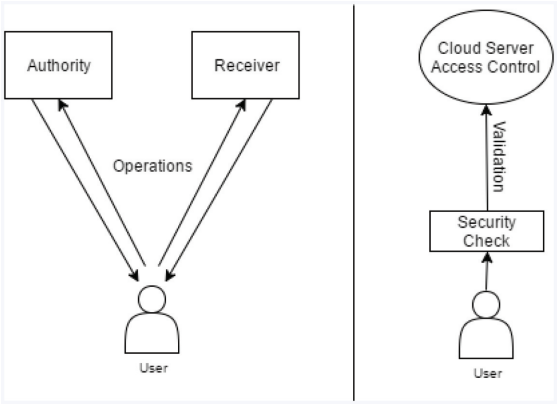

Figure 5 FGAC Structure.

Review: As the operations are performed in user-based fashion, FGAC has a low level of *Privileges/Capabilities Discover*. As the system users are acting with thinking, the usage of it is really easy. The assignments of the users can even be substantiated by another user, which means an intermediate level of *Ease of Privilege Assignments*. *Syntactic And Semantic Support For Specifying AC Rules* can be made optional. It can show changes according to the assignments the user makes. The users can transfer their authorities because they can also make policy defining and assignments. Thus, it has an intermediate level of *Delegation of Administrative Capabilities* and *Policy Combination, Composition, And Constraint.* As it has the best access control that shows the best compatibility with other systems, it has an intermediate *Vertical Scope*. It is really successful regarding the security just as DAC; thus, the FGAC *Safety* is high level as long as the user is not with bad intentions. As FGAC has problems identifying the operations, it has a low level of *Operational/Situational Awareness*. Furthermore, because it has problems adapting to changing environment conditions, it has a low level of *Adaptability*. As mentioned in the description, it does not have any problems working with a single system whereas it creates many problems when working together with complex systems. One of these problems means that it has a low level of *Policy Import and Export.* One of the most important elements that were considered while developing this access control is whether FGAC is compatible with every system. Thus, it has an intermediate level of *OS Compatibility*. The user interface is practical and it thus has an intermediate level of *User Interfaces* and *API*. FGAC creates and defines different access structures for each user, which means a high level of minimum *Least Privilege* and *Management Complexity*. Furthermore, because the control decides the functions that activate for every situation and operation, the security leaks are not allowed in any way; thus, it has a high level of *Separation of Duty*. The operations, functions, and procedures in the systems are high in mathematical terms and dense. This means that it has a low level of *Enforcement Mechanism*. Although functions were developed for every situation, there is not a system to come into effect in the cases of access conflicts; thus, it has a high level of *Policy Conflict*. Furthermore, as the control does not want the complex system to be working, it implemented some restrictions, which means a low level of *Horizontal Scope* and *Configuration Flexibility*.

Hierarchical Attribute Based Access Control

Hierarchical Attribute Based Access Control (HABE) surfaced as a result of its usage of the features of FGAC and Ciphertext-Policy Attribute-Based

Encryption (CP-ABE) mechanisms. The system is operated in a hierarchical structure, and according to this structure, the system is made up of a root master (RM) and multi-layer domain masters that consist of the set of users, and users have the set of attributes as shown in Figure 6. Furthermore, it aims the administration of the data with the same domain with certain linking processes of the data that are in the system. The system affects the assignments mechanisms in hierarchical fields and causes the ABE to increase. This rising substantiates by allowing the admin or the manager to assign their function keys to other admins (in relation with the KP-ABE and CP-ABE that is existent in ABE) [21, 89]. HABE is the best control mechanism that provides the best scalability and flexibility. However, this was not enough for the system developers and the hybrid HASBE model was developed [90, 91]. The most important feature of this access control is that it gives the information of which user, in a secure way, belongs to which class. While this feature is embraced positively by a group of people, it is seen as an element that ignores the system security and privacy [92–94].

Review: As the system consists of root master and domains, it has an intermediate level of *Privileges/Capabilities Discover*. It has an intermediate level of *Ease of Privilege Assignments* because it ensures that the user descriptions and assignments are easily made. As the system operates in a hierarchical way, it needs logical comparisons, thus, it has a high level of *Syntactic And Semantic Support For Specifying AC Rules*. It does not allow users to share their authorities with each other; thus, it has a low level of *Delegation of Administrative Capabilities*. Its Vertical Scope feature

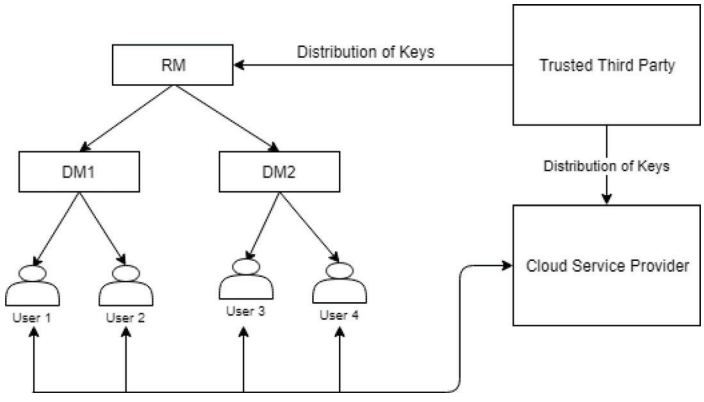

Figure 6 HABE Structure.

is low in addition to its *Horizontal Scope*. The fact that it was made with CP-ABE increases the security of the system, which means a high level of *Safety*. Furthermore, because the features of access controls such as FGAC and HIBE was used, it can create new problems in the alteration processes; thus, it has an intermediate level of *Adaptability*. Fact that it is powerful in terms of its flexibility provides the ability to be able to operate on different operating systems. Thus, an intermediate level of *OS Compatibility* is existent. HABE has a hierarchical structure; thus, the ID assignments of the users are performed according to this hierarchy, and the operations are carried out by the domain masters. Thus, the access restrictions are substantiated in powerful fashion, and in this manner, the distribution of objectives and processes are performed in the best possible way. Thus, it has a high level of *Least Privilege, Separation of Duty* and *Management Complexity*. Its *Enforcement Mechanism* is at low levels because the operations are performed similar to the manner in FGAC. It does not have any procedures to deal with conflicts, which means a high level of *Policy Conflict*. As the hierarchical structure in the control is valid for the information entries that will be implemented in the system, it results in the limitation of *Horizontal Scope* and the system does not have scalability or flexibility features, which means a low level of *Configuration Flexibility*.

Attribute Based Encryption Fine Grained Access Control

As a result of the deficiencies of FGAC, a more secure access control was needed and a hybrid model was created. It is used for Attribute Based Encryption (ABE) processes and allows the user to perform encrypting-decrypting processes. It also performs the operations on the data via a private key and a cipher text [87, 95–98]. It is separated into two categories as KP-ABE and CP-ABE [21, 99–102]. In the former category, the decryption key used for decrypting is made of many access structures and the operations are performed accordingly; however, in the latter category, the decryption key consists of many attributes [98, 103–107]. Figure 7 shows a combination of both fine-grained access control and attribute based encryptions.

Review: As the operations have a low level of *Privileges/Capabilities Discover*, The assignments of the users can be substantiated by another user, and this means an intermediate level of *Ease of Privilege Assignments*. The *Syntactic And Semantic Support For Specifying AC Rules* can be made to be optional. It can also show changes according to the assignments the user makes. The users can transfer their authorities. Thus, it has an intermediate level of the *Delegation of Administrative Capabilities*. As it has the best

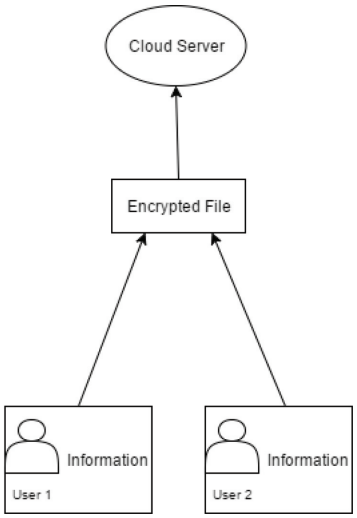

Figure 7 ABE FGAC Structure.

access control that shows the best compatibility with other systems, it has an intermediate *Vertical Scope*. It is really successful regarding the security; thus, the *Safety* is in high levels. As it has problems identifying the operations, it has a low level of *Operational/ Situational Awareness*. It also has problems adapting to changing environment conditions, which means it has a low level of *Adaptability*. It can work with a singled system but it creates many problems while working together with complex systems. One of these problems means that it has a low level of *Policy Import and Export*. It has been compatible with every system. Thus, it has an intermediate level of *OS Compatibility*. The user interface is practical; thus it has an intermediate level of *User Interfaces and API*. ABE FGAC creates and defines different access structures for each user and decides the functions for every situation and operation, and the security leaks are not allowed in any manner. This means a high level of *Least Privilege* and *Separation of Duty* and a high level of *Management Complexity*. The operations, functions, and procedures this system allows to be included in the systems are really high in mathematical terms and very dense. Consequently, it has a low level of *Enforcement Mechanism*. There has not been a system to come into effect in the cases of access conflicts; thus, it has a high level of *Policy Conflict*. Furthermore, as the control does not want the complex system to work, it implemented some restrictions, which means a low level of *Horizontal Scope* and *Configuration Flexibility.*

5 Analysis

Security is one of the most important issues in Cloud Computing, and access control mechanisms play a primer role in enabling identification, authorization, authentication, access approval, and audit services on resources.

The main result of our study is that our research is detailed enough for comprehending the essentials of controls. In this paper, we have analyzed different access control models like RBAC, ABAC, DAC, MAC, FGAC, HABE and ABE-FGAC with their characteristics, advantages, disadvantages and their compliance with NIST standards. The existing solutions are Role Based Access Control-RBAC and Discretionary Access Control-DAC. Still existing solutions are not sufficient to trust the cloud. RBAC is the basic access control model that the users are assigned to different roles and the necessary permissions, limitations and authorizations are performed because of these roles. DAC provides the basis for the security systems and also it has a really high flexibility when compared to other controls.

Since there are some several access control survey studies in the literature we aimed to show the differences of our research with others. A brief tabular comparison has been provided below in Table 1 on the basis of following criteria: investigated access control approaches, supporting main text with visual abstracts, listing advantages & disadvantages, queried databases, number of reviewed articles, and analysis variety whether the NIST metrics were used or not.

Masood and Shibli [52] analyzed cloud-based access control systems and evaluated those using NIST defined access control evaluation criteria with the perspective of 7 of 24 NIST metrics. Charanya and Aramudhan [21] analyzed features of cloud bases access controls and found their security issues. Subashini and Kavitha [3] developed the methodology of policy-based file access using ABE with cipher text scheme. Punithasurya and Jeba Priya [4] discussed various types of access control mechanisms. Langaliya and Aluvalu [108] discussed traditional access control models and their advantages-disadvantages. Msahli, Chen and Serhrouchni [12] define the profile as the combination of all possible authorization, role, and other access parameters in Cloud system. Yang, Liu, Jia, and Shen [109] focus on how to securely share video contents to a certain group of people in cloud-based multimedia systems, and propose a cryptographic approach, a provably secure time domain attribute-based access control (TAAC) scheme, to secure the cloud-based video content sharing for that they analyzed ABAC and ABE.

Table 1 Comparison of survey studies

Studies	Access Control Approaches	Graphical Definitions	Advantages Disadvantages	Databases	NIST Metrics	Number of Reviewed Articles
Masood et al. (2012) [52]	*RBAC, TRBAC, ABE FGAC, FGAC, HABE, CBAC,* **ARBAC**	√	√	*IEEE, ACM, Springer Science Direct*	√*	*11*
Charanya et al. (2016) [21]	*HIBE, HASBE, IBAC, RBAC, ABE*	*X*	*X*	*IEEE, SPRINGER*	*X*	*13*
Subashini et al. (2011) [3]	*MAC, DAC, RBAC*	*X*	*X*	*IEEE, citeseer, sersc*	*X*	*9*
Punithasurya et al. (2012) [4]	*DAC, MAC, RBAC, ABAC, dRBAC, coRBAC*	√	√	*IEEE, Springer, SemanticScholar, chinacloud, ACM, sersc*	*X*	*20*
Langaliya et al. (2015) [108]	*DAC, MAC, RBAC, ABAC, ABE*	√	√	*Springer, citeseer, ijaiem, ACM, Science Direct, IEEE*	*X*	*13*
Msahli et al. (2014) [12]	*MAC, DAC, UCON*	*X*	*X*	*academia, Springer, Science Direct, IEEE, ACM, usenix*	*X*	*30*
Yang et al. (2016) [109]	*ABAC, ABE*	*X*	*X*	*IEEE, Springer, Science Direct, ACM, academia*	*X*	*37*
Our Study	*RBAC, ABAC, DAC, MAC, FGAC, HABE, ABE-FGAC*	√	√	*IEEE, ACM, Springer, Science Direct, Wiley*	√	*109*

*7 of 24 NIST metrics were applied.

This analysis shows that all the existing access control systems have different appearances of cloud authorization. They either target a specific scenario or provide a solution for problems of access control system for the cloud. None of these systems on the cloud perform all requirements of cloud platform or user. Research in the field of authorization in the cloud environment should be reliable and scalable. Also, research focuses on the extensible framework for the cloud environment that includes different access control models. It was found that many controls seem to lack flexibility and scalability during the review, which Table 2 shows the controls. As Table 2 shows the most useful control is DAC, depending on when the studies are examined, the proposed new models are mostly RBAC, DAC, and MAC combinations. Figure 8 also shows the publication distribution of invested 109 papers by years.

Table 2 Metric numbers and access control degrees

	RBAC	ABAC	DAC	FGAC	MAC	HABE	ABE-FGAC
1	X	X	X	X	X	X	X
2	M	Y	H	L	L	L	L
3	H	H	H	M	M	M	M
4	M	H	L	M	L	H	M
5	L	H	M	H	M	H	H
6	H	L	H	L	L	M	L
7	H	L	H	L	L	L	L
8	H	H	H	L	L	L	L
9	H	M	M	M	M	L	M
10	L	M	L	M	L	L	M
11	Y	Y	M	Y	Y	Y	Y
12	M	M	L	H	M	H	H
13	M	H	M	H	H	H	H
14	M	H	H	M	H	H	M
15	H	H	H	H	M	H	H
16	H	M	M	Y	M	Y	Y
17	M	M	M	L	M	M	L
18	X	X	X	X	X	X	X
19	H	M	M	L	M	M	L
20	*	Y	Y	M	Y	M	M
21	Y	M	M	Y	H	M	Y
22	Y	L	Y	L	Y	Y	L
23	M	L	M	M	M	Y	M
24	L	H	L	M	M	M	M

(Low: L, Medium: M, High: H, *optional, Not applicable: X, Not Mentioned: Y)

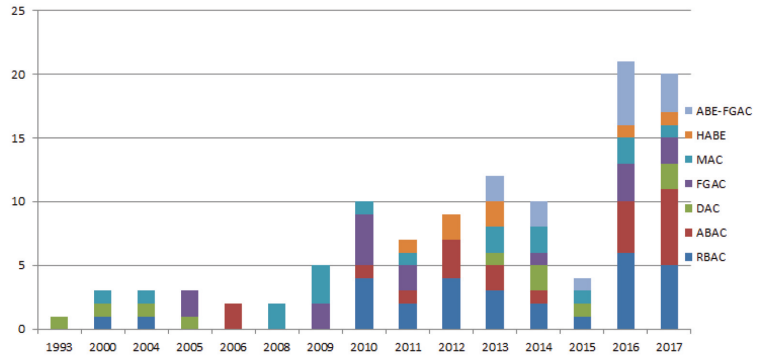

Figure 8 Number of Publications regarding Control Mechanisms per year.

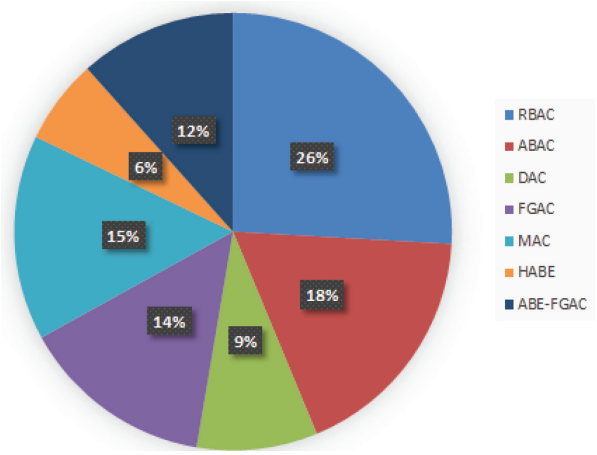

Figure 9 Percent Distributions of Control Mechanism Publications.

Figure 9 shows the percentage of distributions on different controls, and it is seen that the number of studies related to RBAC and ABAC increases every year.

6 Conclusion

In this paper, we analyzed 109 different articles that examine the access control mechanisms used in cloud computing according to the NIST standards and compared and classified processes of these articles. During our research, we employed 24 different class distinctions of NIST definitions

by using the unique advantages and disadvantages of every model. As the researches were performed with our own means and with the resources our system provided, examinations of some access control mechanisms could not be performed regarding their compliance with the standards. The Audit, Bypass and Expression properties principles of the descriptions could not be examined. The research in the literature shows that the most used model is RBAC. Similarly, the most used models while creating hybrid models are FGAC and RBAC. Furthermore, although MAC and DAC are the most trustworthy models amongst the systems that were examined, they were not preferred on their own because of the flexibility of MAC and the low security of DAC relative to MAC.

References

[1] Almubaddel, M., and Elmogy, A. M. (2016). Cloud computing antecedents, challenges, and directions. In *Proceedings of the International Conference on Internet of things and Cloud Computing*, p. 16.

[2] Chen, W. N., and Zhang, J. (2012). A set-based discrete PSO for cloud workflow scheduling with user-defined QoS constraints. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC),* (pp. 773–778).

[3] Subashini, S., and Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. *Journal of network and computer applications*, 34(1), 1–11.

[4] Punithasurya, K., and Jeba Priya, S. (2012). Analysis of different access control mechanism in cloud. *International Journal of Applied Information Systems (IJAIS), Foundation of Computer Science FCS*, 4(2).

[5] Ahmadi, M., Chizari, M., Eslami, M., Golkar, M. J., and Vali, M. (2015). Access control and user authentication concerns in cloud computing environments. In *1st International Conference on Telematics and Future Generation Networks (TAFGEN),* (pp. 39–43).

[6] Alpaslan, G., and Kalıpsız, O. Bulut Bilişim Teknolojisinin Yazılım Performans Testlerinde Kullanımı.

[7] Gajbhiye, A., and Shrivastva, K. M. P. (2014). Cloud computing: Need, enabling technology, architecture, advantages and challenges. In *Confluence The Next Generation Information Technology Summit (Confluence), 2014 5th International Conference-* (pp. 1–7). IEEE.

[8] Timmermans, J., Stahl, B. C., Ikonen, V., and Bozdag, E. (2010). The ethics of cloud computing: A conceptual review. In *IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom),* 2010 (pp. 614–620).

[9] Shikha Singh, Binay Kumar Pandey, and Ratnesh Srivastava (2014). Cloud computing attacks: a discussion with solutions. *Open Journal of Mobile Computing and Cloud Computing*, 1(1), 1–10.

[10] Hu, V. C., and Kent, K. A. (2012). *Guidelines for access control system evaluation metrics*. US Department of Commerce, National Institute of Standards and Technology.

[11] Khan, M. F. F., and Sakamura, K. (2015). Fine-grained access control to medical records in digital healthcare enterprises. In *International Symposium on*, *Networks, Computers and Communications (ISNCC), 2015* (pp. 1–6). IEEE.

[12] Msahli, M., Chen, X., and Serhrouchni, A. (2014). Towards a fine-grained access control for cloud. In *IEEE 11th International Conference on*, *e-Business Engineering (ICEBE), 2014* (pp. 286–291).

[13] Li, W., Wan, H., Ren, X., and Li, S. (2012). A refined RBAC model for cloud computing. In *11th International Conference on Computer and Information Science (ICIS), 2012 IEEE/ACIS* (pp. 43–48).

[14] Kuhn, D. R., Coyne, E. J., and Weil, T. R. (2010). Adding attributes to role-based access control. *Computer*, 43(6), 79–81.

[15] Wu, T. K., Lin, Y. W., and Lin, I. C. (2012). A cloud-user access control mechanism based on data masking. In *Sixth International Conference on Genetic and Evolutionary Computing (ICGEC), 2012* (pp. 165–168). IEEE.

[16] Fu, Y., Liu, Y., Liu, D., Lou, F., and Yan, K. (2016). An environment-based RBAC model for internal network. In *Computer Communication and the Internet (ICCCI), 2016 IEEE International Conference on* (pp. 91–94). IEEE.

[17] Elliott, A., and Knight, S. (2016). Start Here: Engineering Scalable Access Control Systems. In *Proceedings of the 21st ACM on Symposium on Access Control Models and Technologies* (pp. 113–124). ACM.

[18] Hurtuk, J., Baláž, A., and Ádàm, N. (2016). Security sandbox based on RBAC model. In IEEE 11th International Symposium on *Applied Computational Intelligence and Informatics (SACI)*, 2016 (pp. 75–80).

[19] Pandey, S., Dwivedi, A., Pant, J., and Lohani, M. (2016). Security enforcement using TRBAC in cloud computing. In *International Conference on Computing, Communication and Automation (ICCCA), 2016* (pp. 1232–1238).

[20] Chatterjee, S., Gupta, A. K., Mahor, V. K., and Sarmah, T. (2014). An efficient fine grained access control scheme based on attributes for enterprise class applications. In *International Conference on Signal Propagation and Computer Technology (ICSPCT), 2014* (pp. 273–278).

[21] Charanya, R., and Aramudhan, M. (2016). Survey on access control issues in cloud computing. In *International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS)*, (pp. 13–4). IEEE.

[22] Sirisha, A., and Kumari, G. G. (2010). API access control in cloud using the role based access control model. In *Trendz in Information Sciences & Computing (TISC), 2010* (pp. 1353–137).

[23] Zhou, L., Varadharajan, V., and Hitchens, M. (2013). Achieving secure role-based access control on encrypted data in cloud storage. *IEEE transactions on information forensics and security, 8*(12), 1947–1960.

[24] Strembeck, M., and Mendling, J. (2011). Modeling process-related RBAC models with extended UML activity models. *Information and Software Technology, 53*(5), 456–483.

[25] Chen, S. T., Xu, J. F., Hang, Y. X., and Li, J. W. (2016). Role-based access control for memory security on Network-on-Chips. In *13th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), 2016* (pp. 1422–1424). IEEE.

[26] Yaira K Rivera Sánchez, Steven A Demurjian, and Mohammed S Baihan. Achieving rbac on restful apis for mobile apps using fhir.

[27] Gunti, N., Sun, W., and Niamat, M. (2011). I-rbac: Isolation enabled role-based access control. In *Ninth Annual International Conference on Privacy, Security and Trust (PST), 2011* (pp. 79–86).

[28] Chen, H. C., and Violetta, M. A. (2013). A cognitive RBAC model with handover functions in small heterogeneous networks. *Mathematical and Computer Modelling, 58*(5-6), 1267–1288.

[29] Saenko, I., and Kotenko, I. (2017). Administrating role-based access control by genetic algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion* (pp. 1463–1470).

[30] Sergeev, A., and Matulevicius, R. (2017). An Approach to Capture Role-Based Access Control Models from Spring Web Applications.

In *Enterprise Distributed Object Computing Conference (EDOC), 2017 IEEE 21st International* (pp. 159–164).

[31] YAN, D. F., Yuan, T. I. A. N., HUANG, J. L., and YANG, F. C. (2013). Privacy-aware RBAC model for web services composition. *The Journal of China Universities of Posts and Telecommunications, 20*, 30–34.

[32] Chuanfan, L. (2010). Research on role-based access control policy of e-government. In *International Conference on E-Business and E-Government (ICEE), 2010* (pp. 714–716). IEEE.

[33] Kwon, J., and Moon, C. J. (2007). Visual modeling and formal specification of constraints of RBAC using semantic web technology. *Knowledge-Based Systems, 20*(4), 350–356.

[34] Rui-Feng Zhu, Jie Ning, and Pei Yu (2012). Application of role-based access control in information system. In *International Conference on Wavelet Active Media Technology and Information Processing (ICWAMTIP),* (pp. 426–428). IEEE.

[35] Habib, M. A., Ahmad, M., Mahmood, N., and Ashraf, R. (2017). An evaluation of role based access control towards easier management compared to tight security. In *Proceedings of the International Conference on Future Networks and Distributed Systems*, p. 44. ACM.

[36] Mitra, B., Sural, S., Vaidya, J., and Atluri, V. (2017). Migrating from RBAC to temporal RBAC. IET Information Security, 11(5), 294–300.

[37] Jin, P., and Fang-Chun, Y. (2006). Description logic modeling of temporal attribute-based access control. In *First International Conference on Communications and Electronics, 2006. ICCE'06.* (pp. 414–418).

[38] Ed-Daibouni, M., Lebbat, A., Tallal, S., & Medromi, H. (2016). A formal specification approach of privacy-aware attribute based access control (pa-abac) model for cloud computing. In *International Conference on Systems of Collaboration (SysCo),* (pp. 1–5).

[39] Tawosi, V. (2016). A light weight dynamic attribute based access control module integrated with business rules. In *IEEE 10th International Conference on Application of Information and Communication Technologies (AICT),* (pp. 1–5).

[40] Pussewalage, H. S. G., and Oleshchuk, V. A. (2016). An attribute based access control scheme for secure sharing of electronic health records. In *IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom),* (pp. 1–6).

[41] Shen, H. B., and Hong, F. (2006). An attribute-based access control model for web services. In *Seventh International Conference on Parallel and Distributed Computing, Applications and Technologies, 2006. PDCAT'06*. (pp. 74–79).

[42] Hirra Anwar and Muhammad Awais Shibli (2012). Attribute based access control in dspace. In *7th International Conference on Computing and Convergence Technology (ICCCT),* pp. 571–576. IEEE.

[43] Sabbari, M., and Alipour, H. S. (2011). Improving attribute based access control model for web services. In *Information and Communication Technologies (WICT), 2011 World Congress on* (pp. 1223–1228). IEEE.

[44] Dan, N., Hua-Ji, S., Yuan, C., and Jia-Hu, G. (2012). Attribute based access control (ABAC)-based cross-domain access control in service-oriented architecture (SOA). In *International Conference on Computer Science & Service System (CSSS),* (pp. 1405–1408).

[45] Bhatt, S., Patwa, F., and Sandhu, R. (2016). An attribute-based access control extension for openstack and its enforcement utilizing the policy machine. In *IEEE 2nd International Conference on Collaboration and Internet Computing (CIC),* (pp. 37–45).

[46] Bhatt, S., Patwa, F., and Sandhu, R. (2017). ABAC with group attributes and attribute hierarchies utilizing the policy machine. In *Proceedings of the 2nd ACM Workshop on Attribute-Based Access Control* (pp. 17–28).

[47] Heitor Henrique de Paula Moraes Costa, Aletéia Patrícia Favacho de Araújo, Joaõ José Costa Gondim, Maristela Terto de Holanda, and Maria Emília Machado Telles Walter. Attribute based access control in federated clouds: A case study in bionformatics. In *12th Iberian Conference on Information Systems and Technologies (CISTI),* (pp. 1–7).

[48] Ed Coyne and Timothy R Weil (2013). ABAC and RBAC: scalable, flexible and auditable access management. *IT Professional*, 15(3):0014–16.

[49] Carlos E Rubio-Medrano, Clinton D'Souza, and Gail-Joon Ahn (2013). Supporting secure collaborations with attribute-based access control. In *9th International Conference Conference on Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom),* pp. 525–530.

[50] Biswas, P., Sandhu, R., and Krishnan, R. (2017). Attribute transformation for attribute-based access control. In *Proceedings of the 2nd ACM Workshop on Attribute-Based Access Control*, pp. 1–8.

[51] Servos, D., and Osborn, S. L. (2017). Current research and open problems in attribute-based access control. *ACM Computing Surveys (CSUR), 49(4)*, 65.

[52] Rahat Masood, Muhammad Awais Shibli, et al (2012). Comparative analysis of access control systems on cloud. In *13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel & Distributed Computing (SNPD),* pp. 41–46.

[53] Obrsta, L., McCandlessb, D., and Ferrella, D. (2012). Fast semantic attribute-role-based access control (ARBAC) in a collaborative environment. In *8th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, (pp. 703–710).

[54] Wei, Y., Shi, C., and Shao, W. (2010). An attribute and role based access control model for service-oriented environment. In *Chinese Control and Decision Conference (CCDC),* (pp. 4451–4455).

[55] Talukdar, T., Batra, G., Vaidya, J., Atluri, V., and Sural, S. (2017). Efficient Bottom-Up Mining of Attribute Based Access Control Policies. In *IEEE 3rd International Conference on Collaboration and Internet Computing (CIC)*, (pp. 339–348). IEEE.

[56] Eugene Sanzi, Steven A Demurjian, and Jac Billings. Integrating trust profiles, trust negotiation, and attribute based access control.

[57] Auxilia, M., and Raja, K. (2012). A semantic-based access control for ensuring data security in cloud computing. In *International Conference on Radar, Communication and Computing (ICRCC),* (pp. 171–175).

[58] Zhang, K. J., and Jin, W. (2004). Putting role-based discretionary access control into practice. In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics,* (pp. 2691–2696).

[59] Osborn, S., Sandhu, R., and Munawer, Q. (2000). Configuring role-based access control to enforce mandatory and discretionary access control policies. *ACM Transactions on Information and System Security (TISSEC)*, 3(2), 85–106.

[60] Li, N., and Tripunitara, M. V. (2005). On safety in discretionary access control. In *Security and Privacy, IEEE Symposium on* (pp. 96–109). IEEE.

[61] Zamite, J., Domingos, D., Silva, M. J., and Santos, C. (2013). Group-based discretionary access control for epidemiological resources. *Procedia Technology*, 9, 1149–1158.

[62] Thomas, R. K., and Sandhu, R. S. (1993). Discretionary access control in object-oriented databases: Issues and research directions. In *Proc. 16th National Computer Security Conference* (pp. 63–74).

[63] Fan, Y., Han, Z., Liu, J., and Zhao, Y. (2009). A mandatory access control model with enhanced flexibility. In *International Conference on Multimedia Information Networking and Security, MINES'09*. (Vol. 1, pp. 120–124). IEEE.

[64] Zou, D., Shi, L., and Jin, H. (2009). DVM-MAC: a mandatory access control system in distributed virtual computing environment. In *15th International Conference on Parallel and Distributed Systems (ICPADS),* (pp. 556–563). IEEE.

[65] Briffaut, J., Lalande, J. F., and Smari, W. W. (2008). Team-based MAC policy over security-Enhanced Linux. In *Second International Conference on Emerging Security Information, Systems and Technologies, SECURWARE'08*. (pp. 41–46).

[66] Zhu, H., Lü, K., and Jin, R. (2009). A practical mandatory access control model for xml databases. *Information Sciences*, 179(8):1116–1133.

[67] Jiang, Y., Lin, C., Yin, H., and Tan, Z. (2004). Security analysis of mandatory access control model. In *IEEE International Conference on Systems, Man and Cybernetics,* (Vol. 6, pp. 5013–5018).

[68] Kerr, L., and Alves-Foss, J. (2016). Combining Mandatory and Attribute-Based Access Control. In *49th Hawaii International Conference on System Sciences (HICSS),* (pp. 2616–2623). IEEE.

[69] Wang, R., Azab, A. M., Enck, W., Li, N., Ning, P., Chen, X., and Cheng, Y. (2017). SPOKE: Scalable Knowledge Collection and Attack Surface Analysis of Access Control Policy for Security Enhanced Android. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security* (pp. 6126–624). ACM.

[70] Blanc, M., and Lalande, J. F. (2013). Improving mandatory access control for HPC clusters. *Future Generation Computer Systems,* 29(3), 876–885.

[71] Lei, Z., Hongli, Z., Lihua, Y., and Xiajiong, S. (2011). A mandatory access control model based on concept lattice. In *International Conference on Network Computing and Information Security (NCIS),* (Vol. 1, pp. 8–12).

[72] Ray, I., and Kumar, M. (2006). Towards a location-based mandatory access control model. *Computers & Security,* 25(1), 36–44.

[73] Shan, Z. (2009). Compatible and Usable Mandatory Access Control for Good-enough OS Security. In *Second International Symposium on Electronic Commerce and Security,*. ISECS'09. (Vol. 1, pp. 246–250).

[74] Taubmann, B., Rakotondravony, N., and Reiser, H. P. (2016). Cloudphylactor: Harnessing mandatory access control for virtual machine introspection in cloud data centers. In *Trustcom/BigDataSE/I SPA, IEEE* (pp. 957–964).

[75] Sujansky, W. V., Faus, S. A., Stone, E., and Brennan, P. F. (2010). A method to implement fine-grained access control for personal health records through standard relational database queries. *Journal of biomedical informatics, 43*(5), S46–S50.

[76] Ruj, S., Nayak, A., and Stojmenovic, I. (2011). Distributed fine-grained access control in wireless sensor networks. In *Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International* (pp. 352–362). IEEE.

[77] Ma, F., Gao, Y., Yan, M., Xu, F., and Liu, D. (2010). The fine-grained security access control of spatial data. In *18th International Conference on Geoinformatics,* (pp. 1–4).

[78] Li, J., Zhao, G., Chen, X., Xie, D., Rong, C., Li, W., and Tang, Y. (2010). Fine-grained data access control systems with user accountability in cloud computing. In *IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom),* (pp. 89–96). IEEE.

[79] Mazzoleni, P., Crispo, B., Sivasubramanian, S., and Bertino, E. (2005). Efficient integration of fine-grained access control in large-scale grid services. In *IEEE International Conference on Services Computing,* (Vol. 1, pp. 77–84). IEEE.

[80] Lai, Y. Y., and Qian, Q. (2015). H Base fine grained access control with extended permissions and inheritable roles. In *16th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD),* (pp. 1–5).

[81] Ulusoy, H., Kantarcioglu, M., Pattuk, E., and Hamlen, K. (2014). Vigiles: Fine-grained access control for mapreduce systems. In *IEEE International Congress on Big Data (BigData Congress),* (pp. 40–47).

[82] Shi, J., Zhu, H., Fu, G., and Jiang, T. (2009). On the soundness property for sql queries of fine-grained access control in dbmss. In *Eighth*

IEEE/ACIS International Conference on Computer and Information Science, ICIS 2009. (pp. 469–474). IEEE.

[83] Yang, T., Shen, P., Tian, X., and Chen, C. (2017). A Fine-Grained Access Control Scheme for Big Data Based on Classification Attributes. In *IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW),* (pp. 238–245).

[84] Pooryousef, S., and Amini, M. (2016). Fine-grained access control for hybrid mobile applications in Android using restricted paths. In *13th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC),* (pp. 85–90).

[85] Moore, N. (2011). Computational complexity of the problem of tree generation under fine-grained access control policies. *Information and Computation, 209*(3), 548–567.

[86] Yu, S., Wang, C., Ren, K., and Lou, W. (2010). Achieving secure, scalable, and fine-grained data access control in cloud computing. In *Infocom, 2010 proceedings IEEE* (pp. 1–9).

[87] Baseri, Y., Hafid, A., and Cherkaoui, S. (2016). K-anonymous location-based fine-grained access control for mobile cloud. In *13th IEEE Annual Consumer Communications & Networking Conference (CCNC), 2016* (pp. 720–725). IEEE.

[88] Santanu Chatterjee, Sandip Roy, Ashok Kumar Das, Samiran Chattopadhyay, Neeraj Kumar, Goutham Reddy Alavalapati, Kisung Park, and YoungHo Park (2017). On the design of fine grained access control with user authentication scheme for telecare medicine information systems.

[89] Xie, Y., Wen, H., Wu, B., Jiang, Y., and Meng, J. (2015). A modified hierarchical attribute-based encryption access control method for mobile cloud computing. *IEEE Transactions on Cloud Computing*.

[90] Wang, G., Liu, Q., Wu, J., and Guo, M. (2011). Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers. *computers & security, 30*(5), 320–331.

[91] Das, A. K., Massand, A., and Patil, S. (2013). A novel proxy signature scheme based on user hierarchical access control policy. *Journal of King Saud University-Computer and Information Sciences, 25*(2), 219–228.

[92] Wan, Z., Liu, J. E., and Deng, R. H. (2012). HASBE: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing. *IEEE transactions on information forensics and security, 7*(2), 743–754.

[93] Liu, X., Xia, Y., Jiang, S., Xia, F., and Wang, Y. (2013). Hierarchical attribute-based access control with authentication for outsourced data in cloud computing. In *12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom),* (pp. 477–484).

[94] Asim, M., Ignatenko, T., Petkovic, M., Trivellato, D., and Zannone, N. (2012). Enforcing access control in virtual organizations using hierarchical attribute-based encryption. In *Seventh International Conference on Availability, Reliability and Security (ARES),* (pp. 212–217).

[95] Mamatha, B., and Haritha, A. Secure attributes based mechanism through access cipher policies in outsourced cloud data.

[96] Xia, Z., Zhang, L., and Liu, D. (2016). Attribute-based access control scheme with efficient revocation in cloud computing. *China Communications, 13*(7), 92–99.

[97] Luo, E., Liu, Q., and Wang, G. (2016). Hierarchical multi-authority and attribute-based encryption friend discovery scheme in mobile social networks. *IEEE Communications Letters, 20*(9), 1772–1775.

[98] Zhou, K., and Ren, J. (2016). Secure fine-grained access control of mobile user data through untrusted cloud. In *25th International Conference on Computer Communication and Networks (ICCCN),* (pp. 1–9). IEEE.

[99] Cao, Z., Lang, B., and Wang, J. (2016). An Efficient and Fine-Grained Access Control Scheme for Multidimensional Data Aggregation in Smart Grid. In *Trustcom/BigDataSE/I? SPA, 2016 IEEE* (pp. 362–369). IEEE.

[100] Yang, K., Han, Q., Li, H., Zheng, K., Su, Z., and Shen, X. (2017). An efficient and fine-grained big data access control scheme with privacy-preserving policy. *IEEE Internet of Things Journal, 4*(2), 563–571.

[101] Wei Li, Wei Ni, Dongxi Liu, Ren Ping Liu, and Shoushan Luo. Fine-grained access control for personal health records in cloud computing.

[102] Niu, X. (2017). Fine-grained Access Control Scheme Based on Cloud Storage. In *2017 International Conference on Computer Network, Electronic and Automation (ICCNEA)* (pp. 512–515). IEEE.

[103] N. Pandeeswari, P. Ganesh Kumar, and PC Rubini. A serial based encryption for enhanced access control in cloud computing.

[104] Chatterjee, S., Gupta, A. K., and Sudhakar, G. V. (2015). An efficient dynamic fine grained access control scheme for secure data access in cloud networks. In *IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT),* (pp. 1–8).

[105] Ximeng Liu, Hui Zhu, Jianfeng Ma, Jun Ma, and Siqi Ma (2014). Key-policy weighted attribute based encryption for fine-grained access control. In *IEEE International Conference on*, *Communications Workshops (ICC), 2014* (pp. 694–699).

[106] Wang, Q., Zhu, Y., and Luo, X. (2014). Multi-user searchable encryption with fine-grained access control without key sharing. In *3rd International Conference on Advanced Computer Science Applications and Technologies (ACSAT),* (pp. 145–150).

[107] Tamizharasi, G. S., Balamurugan, B., and Manjula, R. (2016). Attribute based encryption with fine-grained access provision in cloud computing. In *Proceedings of the International Conference on Informatics and Analytics* (p. 88). ACM.

[108] Langaliya, C., and Aluvalu, R. (2015). Enhancing cloud security through access control models: A survey. *International Journal of Computer Applications, 112*(7).

[109] Yang, K., Liu, Z., Jia, X., and Shen, X. S. (2016). Time-domain attribute-based access control for cloud-based video content sharing: A cryptographic approach. *IEEE Transactions on Multimedia, 18*(5), 940–950.

Biographies

Gözde Karataş received her undergraduate degree from Mathematics and Computer Science Department of Istanbul Kültür University in 2009 and her graduate degree from Computer Engineering Department of Istanbul Kültür University in 2013. In 2015 she completed her master thesis on NoSql Database Testing in Istanbul Kültür University. During her master studies, she worked on distributed databases. Also she has been working at the Department of Mathematics and Computer Science in Istanbul Kültür University as Research Assistant.

Akhan Akbulut is an assistant professor of Computer Engineering at Istanbul Kültür University. He received his undergraduate and graduate degrees from Computer Engineering Department of Istanbul Kültür University in 2001 and 2008, respectively. In 2013 he completed his Ph.D. thesis on "Extending Wireless Sensor Networks to Internet using Cloud Computing" in Istanbul University. His current research interests focus on design and performance optimization of software-intensive systems, Internet architectures, and broadening participation in computing education and research. Dr. Akbulut is also interested in innovation in distributed systems, especially via the use of Cloud Computing Technology.