Journal of
Cybersecurity and Privacy

MDPI

Review

Distributed Authentication and Authorization Models in Cloud Computing Systems: A Literature Review

Abdulghafour Mohammad

The Department of Informatics, University West, 46132 Trollhattan, Sweden; abdulghafour.mohammad@hv.se

Abstract: As the functionality and services provided by cloud computing increase, control access to these services becomes more complex, and more security breaches are generated. This is mainly based on the emergence of new requirements and constraints in the open, dynamic, heterogeneous, and distributed cloud environment. Despite the importance of identifying these requirements for designing and evaluating access control models, the available studies do not provide a rigorous review of these requirements and the mechanisms that fulfill them. The purpose of this study was to conduct a literature review of the published articles that have dealt with cloud access control requirements and techniques. This paper allowed us to answer the following two research questions: What cloud access control security requirements have been presented in the published literature? What access control mechanisms are proposed to fulfill them? This review yielded 21 requirements and nine mechanisms, reported by 20 manuscripts. The identified requirements in this review will help researchers, academics and practitioners assess the effectiveness of cloud access control models and identify gaps that are not addressed in the proposed solutions. In addition, this review showed the current cloud access control mechanisms used to meet these requirements such as access control based on trust, risk, multi-tenant, and attribute encryption.

Keywords: access control models; access control requirements; role-based access control; attribute-based encryption model; multitenancy model; cloud computing

check for updates

Citation: Mohammad, A. Distributed Authentication and Authorization Models in Cloud Computing Systems: A Literature Review. *J. Cybersecur. Priv.* **2022**, *2*, 107–123. https://doi.org/10.3390/jcp2010008

Academic Editor: Danda B. Rawat

Received: 5 January 2022
Accepted: 25 February 2022
Published: 4 March 2022

1. Introduction

Cloud computing is a prominent paradigm that provides cost-effective, on-demand services such as Software as a Service (SAAS), Platform as a Service (PAAS), and Infrastructure as a Service (IAAS). Despite these advantages, the cloud computing paradigm still confronts several challenges, including data security and privacy, cyber-attacks, and cloud service abuse.

Among all cloud computing security techniques, access control (AC) plays an important role in preserving the integrity of the information by limiting alteration to resources only to legitimate users. In addition, access control protects confidentiality by ensuring that the data are disclosed only to customers who are authorized to access it. In addition, access control protects resources from unauthorized users launching denial-of-service attacks [1]. The data model that forms the basis of the implementation of access control is the "Access Control Model" (ACM) that defines the relationships between permissions, operations, objects, and subjects [2]. According to the definition of the National Institute for Standards and Technology (NIST), cloud computing is " a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [3]. In this definition, the cloud computing paradigm is specified as services working in distributed, open, dynamic, heterogeneous environments. As the functionality and services provided by cloud computing increase, access control to these services becomes more complex, and more security breaches are generated [4]. This is primarily due to the advent of new requirements such as multi-tenant hosting, the dynamic change of the previously unknown users, the heterogeneity of users, security policies, resources, rules, and domains [2,5–19]. These requirements were not considered in full or partly in the design of the majority of traditional access control models such as in the mandatory access control (MAC), discretionary access control (DAC), and role-based access control (RBAC). Therefore, the identification of these new requirements is integral for designing an effective access control model for cloud environments, and they must be considered to specify access control for cloud-based services [6] Despite the importance of ACMs for data and privacy protection, there are relatively few studies that defined ACM requirements in a cloud environment, compared to a large number of proposed ACMs. This, therefore, leads to the design of ineffective ACMs. In this sense, the purpose of this paper is to present an integrative review of the research related to the requirements of ACMs in the cloud environment and to identify critical gaps that are not met by conventional access control models.

This article is structured as follows: Section 2 presents the process of selecting articles, followed by Section 3 with a description of the results obtained through the analysis of selected articles. These results present the access control requirements and mechanisms in the cloud environment. In addition, the findings are discussed in Section 4, and the conclusions are presented in Section 5.

2. Method

This study set out to answer the following research questions:

1. What cloud access control security requirements have been presented in the published literature?
2. What access control mechanisms are proposed to fulfill them?

We used these two research questions to determine the content and structure of the review, to design strategies, to locate and select primary studies, to critically evaluate studies, and to analyze their results.

2.1. Search Method

A review of the literature concerning cloud access control model requirements and mechanisms was conducted using IEEE, Web of Science, Science Direct, and Springer, with no restrictions placed on country or publication date. Search terms included the following: (access control OR authentication, authorization) AND (model OR mechanisms OR techniques) AND (requirements OR evaluation criteria OR Assessment criteria) AND (cloud). Relevant papers were also located by reviewing the references of previously found papers (backward search) and by finding newer publications that contained the cited article (forward search).

2.2. Criteria

The selected article had to meet four basic criteria: The paper was written in English, the article focused on access control requirements in the cloud environment, the study focused on cloud access control techniques, and it was published in peer-reviewed journals and conferences. Studies were excluded if they did not focus specifically on the cloud computing environment, were not focused on requirements, and did not focus on access control mechanisms.

2.3. Information Extraction

Information concerning requirements and mechanisms was extracted from the articles. Information gathered for each study primarily concerned the presence and description of a requirement, the solution used to fulfill this requirement, findings, and limitations. The author also took note of the researchers' conclusions, counted many characteristics among the findings and conclusions of the articles, and summarized the requirements, approaches, and conclusions from different articles.

3. Results

3.1. Cloud Access Control Requirements Included in the Review

The review comprised a total of 21 requirements identified by 20 articles. At each level of the selection process, the number of studies that were identified, screened, included, or excluded is shown in Figure 1.

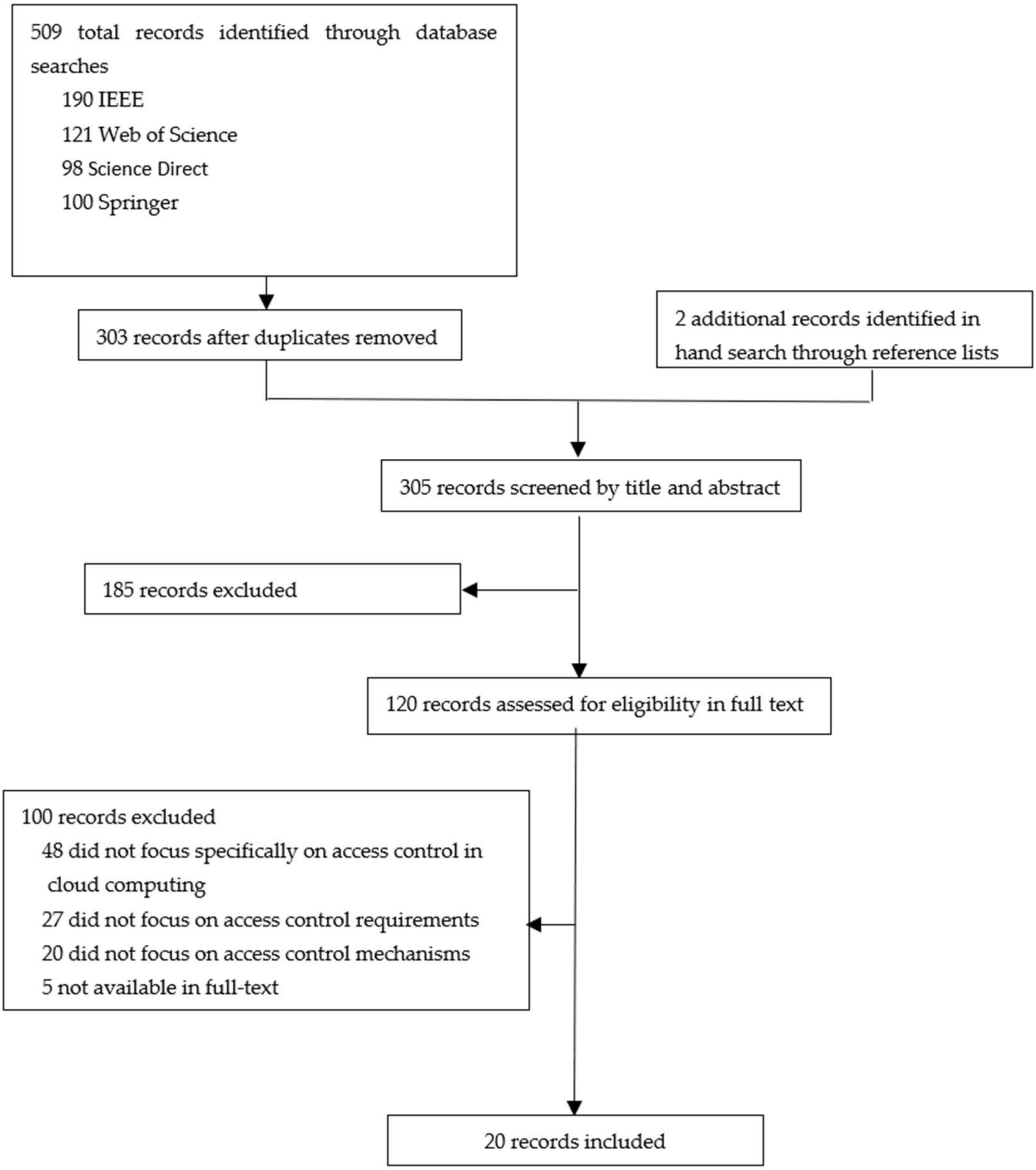
509 total records identified through database searches
190 IEEE
121 Web of Science
98 Science Direct
100 Springer
303 records after duplicates removed
2 additional records identified in hand search through reference lists
305 records screened by title and abstract
185 records excluded
120 records assessed for eligibility in full text
100 records excluded
48 did not focus specifically on access control in cloud computing
27 did not focus on access control requirements
20 did not focus on access control mechanisms
5 not available in full-text
20 records included

Figure 1. Flowchart of the review process.

Table 1 describes 21 requirements contained in the published papers. Six requirements (R1–R5, R12) were identified as general requirements for ACM in the context of a web service environment. Three requirements (R13–R15) represent the security principles. These principles provide a general approach to ACM assessment, whether the proposed model has been implemented in a cloud computing environment or other types of computing.

Table 1. Requirements included in the review.

Nr	Requirement	Description	Research
R1	The privacy	The identities and attributes of cloud users must be unknown to the system at the time of the request	[9,19–21]
R2	Users' heterogeneity	An AC system should be able to effectively deal with the authorizations and authentication of many users with different attributes	[9,15,21,22]
R3	Resource's heterogeneity	AC should be able to support access to many resources of any kind	[22–24]
R4	Contextual information	All constraints should be taken into account in the decision-making process of AC	[20,25,26]
R5	Fine-grained AC	AC must have the ability to specify fine-grained policy which represents different access scenarios	[20,22,23,25]
R6	The tenant can control users	Cloud ACM should provide the tenant with the opportunity to control its users and support policy specifications and apply them properly	[13,20,23]
R7	AC for network access	AC allows heterogeneous client devices to have access to a wide range of networks and through conventional protocols.	[20,25,27]
R8	Resource pooling	AC must consider the methods of implementing resource pooling while ensuring the isolation of shared resources	[21,25]
R9	Rapid elasticity	AC must have the ability to quickly assess the security of new VMs and determine whether the newly added VMs are qualified to carry out a given activity	[21,25–27]
R10	Protection of metering data	AC must consider the protection of metering data	[20,21]
R11	Data Sharing	To make data sharing easier, concepts such as federated identity trust and AC attributes must be considered. Regardless of the service model, consumers are allowed to be accountable for the security of their data on the cloud and who has the right to access it	[20–22,28]
R12	Policy description	AC languages are compatible with all access control techniques and processes.	[9,21,25]
R13	Least privilege principle	Every subject in the cloud should be assigned the basic permissions necessary to fulfill her responsibilities	[20,26,28]
R14	Separation of duties	SoD should be considered in the design of the cloud access control model	[13,20,25]
R15	Delegation	It is essential to support the delegation in the access control policy	[21,25,29]
R16	Interoperability	AC support interoperability, integration, or migration from one service provider to another	[26,27]
R17	Scalability	The maintenance, operation, and administration costs should not increase as the number of applications and customers rises	[20,21,26]
R18	Flexibility	The access rules and constraints in cloud AC should be applied without affecting the flexibility of the system.	[25,26,30]
R19	Confidentiality	When access control sets up a delegation in the cloud, the system should protect data confidentiality and user privacy	[20–22,31]
R20	Functionality	AC implements different types of SoD without affecting the functionality of the system	[15,25,26,31–33]
R21	Auditing	In AC systems, the audit is responsible for keeping track of the present state of the system, recording any failure to make a decision, and reporting any attempts to circumvent the access policy or change privileges	[20,21,34]

Six requirements (R6–R11) have defined the security requirements of cloud computing. However, these requirements have been defined based on a specific perspective of cloud models, implementation, or the type of service provided. The other requirements (R16–R22) represent the control characteristics for determining the ability of ACM to work in an open and dynamic environment. All requirements were defined to evaluate the proposed ACMs or to measure the effectiveness of current ACMs in the cloud environment.

Most of the research focused on defining the general requirements of access control models, and only a few works [7–9] have defined security requirements in the context of cloud computing. However, these requirements have been defined based on a specific perspective of cloud models, implementation, or type of service provided.

To build a more secure and effective cloud authentication and authorization mechanism, the designer of the cloud access control model must consider the different security requirements and from different perspectives [10]. The following requirements were treated in the published articles.

3.1.1. Privacy

The user's privacy in the cloud should be preserved such that the user's location and identity cannot be tracked as the user moves around the cloud [10]. Furthermore, cloud cryptography-based solutions are used to protect data without any knowledge about user identity or attributes [11]. As a result, the identities and attributes of cloud users were unknown to the system at the time of the request. Therefore, controlling access to cloud services is essential, while maintaining user privacy [12].

3.1.2. Resources Heterogeneity

As the size of the cloud increases, the number of heterogeneous resources from various domains increases. Service providers often offer their different types of resources, such as interfaces, applications, APIs, and infrastructures, resulting in a heterogeneous cloud environment [5]. As a result, the number of threats increases proportionally. Moreover, in an open environment such as a cloud, some objects and resources are unknown. Therefore, security management issues are becoming extremely difficult. Thus, access control should be able to support access to many resources of any kind [6].

3.1.3. Users Heterogeneity

Users of cloud environments have different attributes. They access cloud services at any time and from any location [13]. Furthermore, their roles might change frequently in their companies that consume cloud services. Therefore, managing them against many protected resources becomes exceptionally difficult, and an access control system should be able to effectively deal with the authorizations and authentication of these types of users [14].

3.1.4. Contextual Information

Contextual information has a crucial impact in making the final access control decision, especially in a dynamic environment such as the cloud. Context information may include content constraints such as in and out parameters, dynamic relationship between subject and object, and the dynamic conditions of the environment such as time, location, and platform [13]. Therefore, to provide efficient access to resources, all these constraints should be taken into account in the decision-making process of access control. In addition, this process must be automated according to the dynamics and diversity of these conditions. This makes it easier to manage the security of a huge number of heterogeneous objects and resources and saves administrative work [13].

3.1.5. Fine-Grained AC

Traditional access control models that are still used in cloud computing fail to represent different access situations and scenarios because these models only have coarse-grained security policy such as in OpenStack [7]. This gap can be exploited by attackers to gain access to protected resources. Thus, access control models in the open, dynamic, and heterogeneous environments such as cloud computing environments should have the ability to specify fine-grained policy, which represents different access scenarios such as in AWS [6].

3.1.6. Tenants Should Have Full Control over Their Users

Cloud architecture often does not allow organizations that adopt cloud services to specify their access control policy to organize the user access to protected objects [15]. In OpenStack, adding users to tenants and the assignment of roles are only performed by the administrator of the service provider. In AWS, tenants can manage their users. Furthermore, the service provider often does not provide a means to support policies enforcement [16]. Therefore, the expected cloud access control model should provide the tenant with the opportunity to control its users and support policy specifications and apply them properly [6].

3.1.7. Access to a Broad Network

Cloud services are accessible over a wide range of networks and through conventional protocols, allowing heterogeneous client devices (e.g., mobile phones, tablets, laptops, workstations) to access them [17]. This raises security concerns about network access. For example, denial of service (DoS) attacks can be launched against a cloud system, making its resources unavailable to legitimate users. Thus, AC for network access should be managed [6].

3.1.8. Resource Pooling

The computing resources of a cloud system (e.g., network bandwidth, storage, memory, processing) are pooled to provide multiple users via a multi-tenant model (where a single application instance and its support resources serve multiple users) a variety of virtual and physical resources that are dynamically allocated and reallocated in response to customer demands [10]. Information may be disclosed if the resource assigned to a user can be accessed by another co-located user. There is also a concept of location independence in that the user has almost no knowledge of the precise location of the resources given. Location may be specified at a higher degree of abstraction (e.g., data center, country state) that leads to security problems. Therefore, the methods of implementing resource pooling while ensuring the isolation of shared resources should be considered in the AC design [10].

3.1.9. Rapid Elasticity

Cloud services can be elastically provided and released (automatically, in some cases) to rapidly scale inward and outward commensurate with demands. For the consumer, provisioning resources typically appear to be limitless and can be used at any time and in any amount, and are compatible with the addition of new virtual machines (VMs) with specified computing resources [10]. However, the ability to quickly assess the security of new VMs and determine whether the newly added VMs are qualified to carry out a given activity is a challenge for AC design [10].

3.1.10. Measured Service

For controlling and optimizing resource consumption automatically, cloud systems use an appropriate measuring capacity for the type of service (e.g., processing, storage, bandwidth, active end-user account) [10]. Resource usage is tracked, managed, and reported to give transparency to both the provider and the user of the service. To maintain resource usage, cloud consumers should be allowed to review but not to modify their metering data since this could lead to the falsification of payments required for cloud services [10]. Thus, it is reasonable for AC to consider the protection of metering data.

3.1.11. Data Sharing

Sharing information between different organizations is not a trivial task, as a cloud system needs to meet the same organization's security requirements to achieve this [17]. To make data sharing easier, concepts such as federated identity trust and AC attributes must be considered, and building that trust is paramount [8,18,19]. Regardless of the service model, consumers are allowed to be accountable for the security of their data on the cloud and who has the right to access it [19,20]. As a result, data are never controlled by cloud service providers and always remain in the hands of cloud users (the exception to this is log data, but consideration should still be given to how privacy and security are affected by such data). Although a cloud service provider may become the custodian of a customer's data, it should not have access to it [35]. Cloud administrators may be able to display consumer data if they are not protected. In this situation, the customer data should be recorded and marked as accessible by the service provider (based on the provider's access privileges to the data), and the consumer should be informed immediately [10].

3.1.12. Policy Description

Access control languages today are built for specific applications or architecture. As a result, they are incompatible with all access control techniques and processes. In addition, logic operators and Boolean logic are effective ways for the developer of access control policies to deal with complex policy semantics. As a result, not all access control languages provide logical standards in the programming logic [14].

3.1.13. The Least Privilege Principle

In cloud computing, the design of any ACM should be supported, such that every subject in the cloud should be assigned the basic permissions necessary to fulfill her responsibilities [21]. Service providers in the cloud often manage access to data and services based on service or system perspectives without considering the least privilege principle [18]. This causes several threats to organizations that adopted these services. For example, users in both AWS and OpenStack are always associated with full permissions instead of only the permissions needed for their tasks. Once users have access to a resource, they might misuse their permission to configure or delete the protected resources to which they do not require access. Another threat is malicious insider attacks that cause more damage than systems that use the least privilege principle [14]. Therefore, achieving this principle is an essential requirement in the access control model to protect resources efficiently against misusing and malicious insider attacks.

3.1.14. Principle of Separation of Duties (SoD)

This preserves privacy and avoids conflicts, abuses, frauds, and errors such that SoD should be considered in the design of the cloud access control model. SoD divides permissions of a crucial task between different roles [22].

3.1.15. Delegation (D)

In order to protect sensitive collaboration data between service providers and their users, it is essential to support the delegation in the access control policy [23].

3.1.16. Interoperability

Every service provider in cloud computing has its specialty and capacity to supply services sought by its customers. As a result, different service providers frequently collaborate by donating their resources based on the needs of the customers. However, the variety of access control rules and interfaces can lead to poor interoperability, preventing any integration or migration from one service provider to another [24].

3.1.17. Scalability

In terms of the number of users, policy evaluation, and application points, any cloud access control solution should be scalable. Scalability must also take into account maintenance, operation, and administration costs, and they should not increase as the number of applications and customers rises [25].

3.1.18. Flexibility

It is important to maintain security without compromising the flexibility of the system, for example. Inadequate application of the least privilege principle could lead to unnecessary restrictions that affect the flexibility of the system such that the user could still perform her tasks when enforcing the least privilege principle [26]. Therefore, the access rules and constraints in cloud access control should be applied without affecting the flexibility of the system.

3.1.19. Confidentiality

When access control sets up a delegation in the cloud, the system should protect data confidentiality and user privacy, for example, when most access controls are delegated to the service provider [25].

3.1.20. Functionality

The cloud access control module must implement different types of SoD without affecting the functionality of the system, for example, when a valid user in specific cases needs to access two roles at the same time [10].

3.1.21. Auditing

Auditing is a crucial element in protecting cloud computing and the access control systems that accompany it. In access control systems, the audit is responsible for keeping track of the present state of the system, recording any failure to make a decision, either by authorizing or denying access, and reporting any attempts to circumvent the access policy or change privileges. It must also track and record the capabilities assigned to subjects, as well as any changes made to objects, such as renaming, copying, and deleting [25].

3.2. Cloud Access Control Mechanisms Included in the Review

Figure 1 shows the number of articles identified, screened, and included or excluded at each phase of article selection. A total of nine mechanisms reported by 20 articles were included in the review. These mechanisms are described in Table 2. From 1960 up to the writing of this paper, several access controls models have been proposed. This section reviews the predominant access control models used in the cloud computing paradigm against cloud computing environment requirements.

3.2.1. Discretionary Access Control (DAC)

In DAC, each user is uniquely distinguished by his/her identity and, on the basis of this identity and authorization policies, DAC can make access control decisions. The authorization policies in DAC clearly specify for each user of the authorized access modes (read/write/execute) to access each object in the system. Thus, DAC determines who can access what. Each object has access control lists that determine who can access the object in the corresponding access mode. The owner of an object may choose to provide access permissions to other users and modify the access control list [27]. Such functionality makes DAC flexible and easy to use by allowing users to grant permissions, modifying and customizing their access policies individually. However, this feature may be exploited by a third party by abusing the owner's permission and inserting malware since there is no constraint on the use of the information [28]. In addition, DAC will incur significant management costs in open and distributed environments such as the cloud. Therefore, it is generally only used by legacy systems [29].

Table 2. Access control mechanisms included in the review.

Mechanism	Description	Research
Discretionary access control (DAC)	Each user is uniquely distinguished by his/her identity and, on the basis of this identity and authorization policies, DAC can make access control decisions	[27–29]
Mandatory access control (MAC)	The owner does not make access decisions; instead, a central authority makes the policy decision	[6,28]
Role-based access control (RBAC)	Access control decisions are made based on the user's roles and responsibilities, that are trying to access services or protected resources	[22,26,35–37]
Attribute-based access control (ABAC)	The attributes of the involved entities are the basis to make the access control decision.	[13,23,37]
Identity-Based Encryption (IBE)	A master public key and private key are used for encryption and decryption, the data owner can encrypt the data using the receiver's public key, and the receiver can decrypt the data using his private key	[20,24]
Attribute-based encryption access control (ABEAC)	The ABEAC generates encryption keys and a ciphertext based on the user's attributes	[38,39]
Trust-based Access Control Model (TBAC)	The roles are assigned to users based on their trust value.	[19,25,38]
Risk-based Access control models (RiskBAC)	Statistical methods are used to determine and calculate factors contributing to risk in a user's inquiry	[21,25]
Multi-Tenant Access Control (MTAC)	The access decision depends on the attributes of tenants, networks, storage, and other resources in the cloud	[15]

3.2.2. Mandatory Access Control (MAC)

In MAC, the owner does not make access decisions; instead, a central authority does the same as in DAC. Thus, it provides a high level of security and a low level of flexibility, since the subject has not had to control object permissions, and the users have not had absolute privacy [38]. It is therefore used in government and military systems. The security of a cloud system IaaS has a strong relationship with virtualization (hypervisor). As a result, new solutions are being adopted to impose access control on virtual machines and hypervisors [6], where MAC is being used to isolate different virtual machines and to enforce AC on both hypervisors and virtual machines. This will protect the hypervisor from illegal access and various types of attacks, such as a denial of service attack (DOS) on a virtual machine (VM) and hijacking attacks on a hypervisor [6].

3.2.3. Role-Based Access Control (RBAC)

The central purpose of RBAC proposed by [30] is to deal with security management complexity in large organizations by replacing the subjects in the ACLs model with roles and assigning separately each subject to a role. This means that access control decisions are made based on the user's roles and responsibilities that are trying to access services or protected resources. RBAC uses security assurance principles such as least privilege and static and dynamic separation of duties.

This model has a centralized administration by using a central protected base managed by a single authority, which comprises all the security policies for the organization. When a user is assigned a role, that user instantly has all the rights and privileges of that role. Although this approach simplifies the management of subjects and objects in the system and reduces the cost, complexity, and errors, it cannot be scaled for many anonymous users, especially in a coalition environment where organizations that collaborate in the cloud have no secure centralized authority [30]. In RBAC, subjects are assigned roles and permissions to objects by the administrator, which is cumbersome and expensive because of the large number of users and resources in the cloud. The traditional operation–object form is used to represent the authorization in RBAC. However, this paradigm makes security management worse with many objects, different operations, and object hierarchy [31]. A common drawback of RBAC is that it is a coarse-grained access control model that is incapable of dynamically adapting permissions by environment information, such as time and location. However, such factors are used partially in RBAC extensions [37].

Most of the existing cloud platforms do not support assigning users to roles. Instead, each user in the cloud is independent of others and is assigned a root privilege to access resources [32]. In IaaS platforms such as Amazon's EC2, Eucalyptus provides limited support for the RBAC mechanism. Windows Azure, which is a PaaS platform, does not support user-role assignment but assigns instead applications that users deployed within Azure to roles [32].

3.2.4. Attribute-Based Access Control (ABAC)

The use of ABAC to overcome the constraints of traditional access control models has triggered a lot of interest recently. The basic object of ABAC is to use the attributes of involved entities as a basis to make the access control decision [24]. To represent the user's access profile, ABAC uses the combination of the attributes associated with the subject's "S", such as the identity, role, and department; the attributes associated with the resource "R" such as owner, type, and created time [24]; and the attributes of the environment "E" in which authorization process occurs as a time, location, or other technical and operational settings. Thus, ABAC provides a fine-grained expression of the authorization policies rules.

$$\text{RULE X: f (ATTR(s), ATTR(r), ATTR(e))} \rightarrow \text{can\_access(s,r,e)}$$

Such a feature is vital in dynamic and open environments such as the cloud and the use of different attributes in ABAC instead of one "role" as in RBAC, which is expected to satisfy the flexibility needs of the cloud paradigm. AWS uses this characteristic and provides such a type of authorization based on context information such as location, time, and address which is represented in ABAC authorization policies [13]. Many researchers have designed cloud computing access control models by merging the flexible administration feature of RBAC and the dynamic features of ABAC. RBAC was extended by including attributes of the subject, object, and the cloud context environment in a dynamic authorization process and used the roles in RBAC [26,33]. However, this model sacrificed the performance of the dynamic access decision process in the cloud, which depends on verifying a list of access policies in the system against the cloud environment context. In [34], an access control model for cloud computing was proposed using the roles and attributes of subject and object. This provides more specific fine-grained rules in the access policy. However, the frequent and dynamic changing in attributes requires a new calculation to assign the permission and take the access decision, which may be exploited by attackers to destroy the data.

3.2.5. Identity-Based Encryption (IBE)

IBE is a public-key crypto technique, where a master public key and private key are used for encryption and decryption operations; the data owner can encrypt the data using the receiver's public key, and the receiver can decrypt the data using his private key [23]. However, there is a lack of confidence in IBE related to insecure online exchanging of the keys. Besides the problem of flexibility and scalability, other algorithms are used to exchange securely the keys such as the Diffie Hellman algorithm [35,36]. To avoid the problem of flexibility and scalability, attribute-based encryption is used.

Using IBE to control access to data is not enough in cloud computing, but there is a need for a technique for revoking users from the system when needed, e.g., there is an expiration of authority for some users or unauthorized disclosure of secret keys. Here, users lose the required privilege to access their shared data. Therefore, while transforming data to servers, users should have a right to control access to these data [19].

3.2.6. Attribute-Based Encryption Access Control (ABEAC)

In cloud computing to preserve data privacy and make more protection against threats, either symmetric or asymmetric encryption technology can be used [7]. However, these authorization mechanisms are not scalable and flexible, because of using shared secret keys or public keys by the data owner to publish his data to the cloud. To make the access control policy more representative and adaptable for adding new users in the cloud, [24] proposed a flexible and scalable attribute-based encryption algorithm based on an identity-based encryption algorithm (IBE). The flexibility in this model comes from the public key encryption mechanism in ABE where user attributes are used instead of user identity in IBE. The encryption algorithm in ABE generates encryption keys and a ciphertext based on the user's attributes. Scalability in ABE is enabled because the ciphertext can be decrypted based on the receiver's attributes, which must satisfy the encryption policy. This leads to implementing "one to many" encryptions such that encrypted data can be decrypted by users who match specific requirements [24]. Therefore, data owners do not need to do anything when a new user joins the cloud system [7].

ABE has two approaches. The ciphertext-policy ABE (CP-ABE) and the key-policy ABE (KP-ABE). In CP-ABE, a user's attribute is associated with the private key, and a ciphertext determines an access policy. The plaintext can only be recovered if the user's attribute matches the specified access policy of the ciphertext. In KP-ABE, an access policy is defined with a secret key, and ciphertext is generated based on an attribute list. If the user's attribute complies with the access policy tied with the secret key, he/she can decrypt the ciphertext [7]. Conversely, CP-ABE is better than KP-ABE because the data owner decides the access policy.

3.2.7. Trust-Based Access Control Model (TBAC)

Using roles and attributes in designing cloud access control models produces two challenges related to the waste of resources and the dynamic changing in attributes which needs more calculations that may be exploited by the attackers [35]. To address these issues, researchers proposed the trust concept in the access control model. Ref. [20] introduced an access control model by a combination of trusted mechanisms and roles. The roles are assigned to users based on their trust value. However, attackers can gain access to the system by building up trust values. Ref. [20], introduced an access control model based on trust. This model assigned permission by gathering the different trust properties of users. However, this mechanism becomes more complicated in the cloud computing environment, where the users have many trust characteristics.

Ref. [20] used the user behavior trust to assign permissions and roles to the cloud's users. However, there are two steps to authorize the user, that is, activating the role and authorizing the role. However, the access control decision process is not flexible. A fine-grained cloud access control model was proposed to enforce access based on the user trust values and roles [37]. Once a user accesses a cloud service, he/she will obtain an initial trust value, and then roles and permissions are assigned to his/her based on the initial trust value.

3.2.8. Risk-Based Access Control Models (RiskBAC)

Uncertainty in cloud access control models is one aspect of risk generated by a non-dynamic access control system policy, or by a non-flexible access control decision support mechanism. Therefore, several ACMs were proposed to mitigate these risk factors in access control. A novel approach to managing the user risk in RiskBAC was proposed by [38,39]. In this model, statistical ways are used to determine and calculate factors contributing to risk in a user's inquiry. However, this model does not study the risk of object and permissions, which are the basic elements in the RiskBAC model. To include more risk factors in the access decision-making process, a dynamic mechanism was proposed to make an access control decision based on an assessment of the risk related to the environment, object, and user [40]. However, implementing this model lacks a mechanism to measure and collect the required information to assess the risk.

To enhance the access control decision, more risk factors are included in the decision-making process. For instance, in the Fuzzy Multi-Level Security model [38], the risk of negative effects is expected in case of unauthorized revelation of information is considered in access decision. However, extensive work is required to calculate the risk of unauthorized detection. In the same direction, new fuzzy risk values were added in the access decision process in the cloud [41]. These factors are the sensitivity of the information, historical risk information, and the importance of the procedure. However, this model lacks an obvious mechanism to measure the risk value. To make a more accurate risk assessment, ref. [42] included context as a factor to assess the risk; however, it is not clear how the risk value should be calculated when the environment's state changes. However, it is not appropriate to use the trust value of this relation from the history of the previous user. Ref. [21] presented an access control framework. This model contained a component to assess the risk provided by the context data. Although it is a more organized model, this framework suffers from accuracy because it does not consider all risk factors in the access decision process.

3.2.9. A Multi-Tenant Access Control (MTAC)

Multi-tenancy is a cloud architecture, which is applied to SaaS, PaaS, and IaaS. When numerous tenants share the same resources in the cloud, multi-tenancy access control becomes critical. Each tenant applies its configuration based on the hardware or software needed, with no effect on the privacy and security of other users [15]. In the multi-tenant architecture, tenants refer to the customers using the software/hardware resources [18]. The reserving of the privacy and security of the customer's data is essential in a multi-tenancy cloud environment, especially to avoid a tenant from accessing other's resources. Thus, multiple tenants can share a single application without breaching the security of the others.

Several multi-tenancy access control models for cloud computing have been proposed. Some of them exploited the security features in the traditional access control models to support a multi-tenancy environment. To manage and separate tenants in a cloud environment [42], RBAC was extended by using users' roles and an identity management scheme. However, to manage resource sharing in a multi-tenant environment, an extended RBAC and task access control model was proposed [43]. An ABAC model was introduced for cloud IaaS where the access decision depends on the attributes of tenants, networks, storage, and other resources in the cloud [25,44–46].

To facilitate the management of access control in federated multi-cloud applications, a new mechanism was proposed in [25,27,45–51]. This approach used the Representational State Transfer (REST) services interfaces for managing access to different XACML component functionality. To cope with a wide range of access control policies, a cloud access control model was proposed for heterogeneous multi-tenancy architectures [18]. The proposed model used the concepts that existed in RBAC, TRBAC, and ABAC.

Table 3 summarizes the results of access control requirements and techniques included in the current review. The requirements are used as assessment criteria for access control models in cloud environments.

Table 3. Access control mechanisms and requirements included in the review, N: no, Y: yes, N/A: not applicable.

Requirements	Cloud Access Control Models								
	MAC	**DAC**	**RBAC**	**ABAC**	**ABEAC**	**IBEAC**	**TBAC**	**RISKBAC**	**MTAC**
R1	N	Y	Y	Y	Y	Y	Y	Y	Y
R2	N	N	N	Y	Y	N	Y	Y	Y
R3	N	N	N	Y	Y	N	Y	Y	Y
R4	N	N	N	Y	Y	N	Y	Y	Y
R5	N	N	N	N	Y	N	N	N	N
R6	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Y
R7	N	N	N	N	N	N	N	N	Y
R8	N	Y	N	N	N	N	Y	Y	Y
R9	N/A	N/A	Y	N/A	Y	Y	Y	Y	Y
R10	N/A	N/A	N	N	N/A	Y	Y	N/A	Y
R11	N	N	N	N	Y	Y	Y	Y	Y
R12	N	N	N	Y	Y	N	N	Y	Y
R13	N	N	Y	Y	Y	Y	Y	Y	Y
R14	N	N	Y	Y	Y	Y	Y	Y	Y
R15	N	Y	Y	N	N	N	N	N	N
R16	N	N	N	Y	Y	Y	Y	Y	Y
R17	N	Y	Y	Y	Y	Y	Y	Y	Y
R18	N	N	N	Y	Y	N	N	N	N
R19	N	N	Y	Y	Y	Y	Y	Y	Y
R20	N	N	N	Y	Y	N	Y	Y	Y
R21	N	N	Y	Y	Y	N	Y	Y	Y

4. Discussion

This paper's goal was to review the requirements and mechanisms used by researchers designing ACMs for the cloud computing environment. These requirements are considered as evaluation criteria for the assessment of the proposed ACMs and as a mechanism for identifying gaps in their design.

Researchers and designers of security solutions for the cloud environment should be mindful of requirements regarding the dynamic and complex nature of cloud computing. Cloud services, for example, can be elastically provisioned and released (in certain circumstances automatically) in response to demands. Provisioning resources appear to be limitless to the user and can be used in any amount at any time and are compatible with the addition of new virtual machines (VMs) with specified computing resources [10]. The ability to quickly assess the security of new VMs and to determine whether newly added VMs are qualified to carry out a particular activity (R9) is a challenge to the design of ACMs [10]. Despite the importance of this requirement, only three manuscripts considered it in the design of ACMs.

This review indicated that some articles did not take into account cloud architecture requirements in designing a cloud access control model. For example, the multi-tenancy cloud architecture requirement (R6) is only considered in five articles. The importance of this cloud architecture requirement is represented when several tenants share the same resources. In addition, this requirement is directly related to the privacy and security of the customer's data requirement (R1), which is essential to prevent a tenant from accessing the resources of others in a multi-tenancy cloud environment. Therefore, multiple tenants can share a single application without compromising the security of the others. Another requirement related to the cloud architecture is data sharing and resource isolation (R11). To facilitate data sharing, this review showed that the trust and risk concepts were used. Risk, for example, was used as a mechanism to mitigate uncertainty in cloud access control models generated by a non-dynamic access control system policy, or by a non-flexible access control decision support mechanism.

The current review showed that some researchers have not evaluated their MCAs in accordance with all the essential requirements of the cloud environment. Some researchers, for example, evaluated privacy, while others focused on data protection without taking into account the elastic nature of the cloud (R9) or the protection of cloud metering data (R10). There are several possible reasons why an ACM has not been assessed according to all criteria. Such reasons are a lack of time and resources and a focus of researchers on solving one of the ACM issues.

This review showed that the evaluation criteria used by the majority of researchers appeared unclear or were not stated at all. Nine articles, for example, did not contain clear evaluation criteria or referred to them as a task to include them in future work. This may limit the efficiency of the proposed ACMs. Although the importance of the ACM evaluation process for obtaining feedback to improve the design of the ACM and measure its capacity and deficiencies, this process was not clearly stated in most manuscripts. Researchers in this area should therefore be encouraged to publish more detailed information on the evaluation process used to evaluate their proposed ACMs.

A strength of the current review is its innovation. This is the first study of its kind to review the requirements of ACMs in the cloud computing environment. In addition, the search for relevant studies occurred across a number of sources and databases. However, only published studies were included in this review, resulting in a publication bias. Other biases are also thought to exist. These biases occurred throughout the selection and evaluation of the papers, as well as during the synthesis and analysis of the data. Therefore, there is a possibility of subjectivity in the interpretation of research, which could influence the final conclusion.

5. Challenges and Future Directions

On the basis of the results in the result section and the discussion in the Discussion section of the literature review section, the author has identified some important challenges, as follows:

(1) Some of the fundamental and important requirements of a cloud access control model have not been taken into account in access control mechanisms designed for the cloud environment, according to the author's best knowledge. This may limit the effectiveness of the proposed ACMs.

(2) This review has not discovered any studies that define a clear and systematic access assessment process to evaluate proposed cloud access control models to the best of the author's knowledge.

In order to improve the capacity of access control mechanisms in a cloud environment, further research is needed on the above-mentioned aspects. Most of the current research on access control focused mainly on improving the design of CAMs for cloud computing. Additionally, it intended to address the security issues such as privacy and resource sharing. However, the above topics are not significantly addressed in the past literature, to the best of the authors' knowledge. There were not enough studies found to identify requirements and the valuation process for ACMs in the cloud environment. Therefore, it may be worthwhile to carry out further research, considering these aspects. The limitations mentioned earlier in the discussion section are also areas worth investigating.

6. Conclusions

The role of access control techniques in cloud computing is critical and has expanded in recent years. The key contribution of this review is to provide a clear picture that summarizes what has already been written about requirements and mechanisms of access control. The review identified the most important and relevant studies in the field, providing details on the topics that have promoted more academic attention and detailing access control requirements in a cloud computing context. The methodology chosen to answer research questions was a literature review.

Concerning Question 1, i.e., "What cloud access control security requirements have been presented in the published literature?" The following requirements are identified: maintaining users' privacy, supporting users and resources heterogeneity, dynamic decision-making process based on contextual information, ability to specify fine-grained policy. whether tenants should have full control over their users, access to a broad network, implementing resource pooling while ensuring the isolation of shared resources, ability to quickly assess the security of the new VMs, the least privilege access, separation of duties, tenant delegation, and the protection of metering data.

Concerning Question 2, i.e., "What access control mechanisms are proposed to fulfill them?" The following mechanisms are used: trust, risk, multi-tenant, and attribute encryption-based access control.

To conclude, cloud access control requirements have been considered partly in the design of most traditional or proposed access control models. Thus, the creation of a clear evaluation standard for the assessment of the access control solution can help to create more effective access control solutions. Therefore, further research focusing on access control validation is needed.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dhanalakshmi, B.K.; Srikantaiah, K.C.; Venugopal, K.R. Carry Forward and Access Control for Unused Resources in Multi Sharing System of Hybrid Cloud. *Future Gener. Comput. Syst.* **2020**, *110*, 282–290. [CrossRef]
2. Suresha, K.; Vijayakarthick, P.; Dhanasekaran, S.; Murugan, B.S. Threshold Secret Sharing and Multi-Authority Based Data Access Control in Cloud Computing. *Mater. Today Proc.* **2021**, in press. [CrossRef]
3. Mell, P.; Grance, T. *The NIST Definition of Cloud Computing*; National Institute of Standards & Technology: Gaithersburg, MD, USA, 2011. [CrossRef]
4. Sangeetha, M.; Vijayakarthik, P.; Dhanasekaran, S.; Murugan, B. Fine grained access control using H-KCABE in cloud storage. *Mater. Today Proc.* **2020**, *37*, 2735–2737. [CrossRef]
5. Liang, W.; Xie, S.; Cai, J.; Wang, C.; Hong, Y.; Kui, X. Novel Private Data Access Control Scheme Suitable for Mobile Edge Computing. *China Commun.* **2021**, *18*, 92–103. [CrossRef]
6. Ahuja, R.; Mohanty, S.K. A Scalable Attribute-Based Access Control Scheme with Flexible Delegation Cum Sharing of Access Privileges for Cloud Storage. *IEEE Trans. Cloud Comput.* **2020**, *8*, 32–44. [CrossRef]
7. Hu, V.C.; Iorga, M.; Bao, W.; Li, A.; Li, Q.; Gouglidis, A. *General Access Control Guidance for Cloud Systems*; NIST Special Publication: Gaithersburg, MD, USA, 2020.
8. Deng, H.; Qin, Z.; Wu, Q.; Guan, Z.; Deng, R.H.; Wang, Y.; Zhou, Y. Identity-Based Encryption Transformation for Flexible Sharing of Encrypted Data in Public Cloud. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 3168–3180. [CrossRef]
9. Rizwan Ghori, M.; Ali Ahmed, A. Review of Access Control Mechanisms in Cloud Computing. In *Journal of Physics: Conference Series*; Institute of Physics Publishing: Johor, Malaysia, 2018; Volume 1049. [CrossRef]
10. Charanya, R.; Aramudhan, M. Survey on Access Control Issues in Cloud Computing. In Proceedings of the 2016 International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS), Pudukkottai, India, 24–26 February 2016. [CrossRef]
11. Sun, P.J. Security and Privacy Protection in Cloud Computing: Discussions and Challenges. *J. Netw. Comput. Appl.* **2020**, *160*, 102642. [CrossRef]
12. Huang, L.; Xiong, Z.; Wang, G. A Trust-Role Access Control Model Facing Cloud Computing. In Proceedings of the 2016 35th Chinese Control Conference (CCC), Chengdu, China, 27–29 July 2016; IEEE Computer Society; pp. 5239–5242. [CrossRef]
13. Qi, S.; Zheng, Y. Crypt-DAC: Cryptographically Enforced Dynamic Access Control in the Cloud. *IEEE Trans. Dependable Secur. Comput.* **2021**, *18*, 765–779. [CrossRef]
14. Wang, H.; He, D.; Han, J. VOD-ADAC: Anonymous Distributed Fine-Grained Access Control Protocol with Verifiable Outsourced Decryption in Public Cloud. *IEEE Trans. Serv. Comput.* **2020**, *13*, 572–583. [CrossRef]
15. Zhang, Y.; Deng, R.H.; Xu, S.; Sun, J.; Li, Q.; Zheng, D. Attribute-Based Encryption for Cloud Computing Access Control: A Survey. *ACM Comput. Surv.* **2020**, *53*, 1–41. [CrossRef]

16. Gupta, M.; Awaysheh, F.M.; Benson, J.; Alazab, M.; Patwa, F.; Sandhu, R. An Attribute-Based Access Control for Cloud-Enabled Industrial Smart Vehicles. *IEEE Trans. Ind. Inform.* **2021**, *17*, 4288–4297. [CrossRef]
17. Abu Jabal, A.; Davari, M.; Bertino, E.; Makaya, C.; Calo, S.; Verma, D.; Williams, C. ProFact: A Provenance-Based Analytics Framework for Access Control Policies. *IEEE Trans. Serv. Comput.* **2019**, *14*, 1914–1928. [CrossRef]
18. Albulayhi, K.; Abuhussein, A.; Alsubaei, F.; Sheldon, F.T. Fine-Grained Access Control in the Era of Cloud Computing: An Analytical Review. In Proceedings of the 2020 10th Annual Computing and Communication Workshop and Conference, CCWC, Las Vegas, NV, USA, 6–8 January 2020; pp. 748–755. [CrossRef]
19. Challagidad, P.S.; Birje, M.N. Efficient Multi-Authority Access Control Using Attribute-Based Encryption in Cloud Storage. *Procedia Comput. Sci.* **2020**, *167*, 840–849. [CrossRef]
20. Institute of Electrical and Electronics Engineers. *IEEE Standard for Identity-Based Cryptographic Techniques Using Pairings*; 2013. [CrossRef]
21. Shaikh, R.A.; Adi, K.; Logrippo, L. Dynamic Risk-Based Decision Methods for Access Control Systems. *Comput. Secur.* **2012**, *31*, 447–464. [CrossRef]
22. Sandhu, R.S.; Coyne, E.J.; Feinstein, H.L.; Youman, C.E. Computer Role-Based Access Control Models. *Computer* **1996**, *29*, 38–47. [CrossRef]
23. Shin, D.; Akkan, H.; Claycomb, W.; Kim, K. Toward Role-Based Provisioning and Access Control for Infrastructure as a Service (IaaS). *J. Internet Serv. Appl.* **2011**, *2*, 243–255. [CrossRef]
24. Teng, W.; Yang, G.; Xiang, Y.; Zhang, T.; Wang, D. Attribute-Based Access Control with Constant-Size Ciphertext in Cloud Computing. *IEEE Trans. Cloud Comput.* **2017**, *5*, 617–627. [CrossRef]
25. Houssein, R.; Younis, Y.A. Deploying Risk Access Models in a Cloud Environment: Possibilities and Challenges. In Proceedings of the 2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering, MI-STA, Tripoli, Libya, 25–27 May 2021; pp. 234–238. [CrossRef]
26. Cai, F.; He, J.; Ali Zardari, Z.; Han, S. Distributed Management of Permission for Access Control Model. *J. Intell. Fuzzy Syst.* **2019**, *38*, 1539–1548. [CrossRef]
27. Slawik, M.; Blanchet, C.; Demchenko, Y.; Turkmen, F.; Ilyushkin, A.; de Laat, C.; Loomis, C. CYCLONE: The Multi-Cloud Middleware Stack for Application Deployment and Management. In Proceedings of the 2017 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), Hong Kong, China, 11–14 December 2017; pp. 347–352. [CrossRef]
28. Hasebe, K.; Mabuchi, M. Capability-Role-Based Delegation in Workflow Systems. In Proceedings of the 2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, Hong Kong, China, 11–13 December 2010; EUC 2010. pp. 711–717. [CrossRef]
29. Keromytis, A.D.; Smith, J.M. Requirements for Scalable Access Control and Security Management Architectures. *ACM Trans. Internet Technol.* **2007**, *7*, 8-es. [CrossRef]
30. Hu, V.C.; Kuhn, R.; Yaga, D. *Verification and Test Methods for Access Control Policiesmodels*; NIST Special Publication: Gaithersburg, MD, USA, 2017. [CrossRef]
31. Kuang, T.P.; Ibrahim, H.; Sidi, F.; Udzir, N.I.; Alwan, A.A. An Effective Naming Heterogeneity Resolution for XACML Policy Evaluation in a Distributed Environment. *Symmetry* **2021**, *13*, 2394. [CrossRef]
32. Patil, V.; Mei, A.; Mancini, L.V. Addressing Interoperability Issues in Access Control Models. In Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security, Singapore, 20–22 March 2007; ASIACCS '07. pp. 389–391. [CrossRef]
33. Hu, V.C.; Scarfone, K. *Guidelines for Access Control System Evaluation Metrics*; NIST Special Publication: Gaithersburg, MD, USA, 2012. [CrossRef]
34. Calista Bebe, P.; Akila, D. Bloom Hash Probabilistic Data Structure and Benaloh Cryptosystem for Secured Data Storage and Access Control in Cloud. *Mater. Today Proc.* 2021, in press. [CrossRef]
35. Wei, J.; Liu, W.; Hu, X. Secure Data Sharing in Cloud Computing Using Revocable-Storage Identity-Based Encryption. *IEEE Trans. Cloud Comput.* **2018**, *6*, 1136–1148. [CrossRef]
36. He, Y.; Dong, G.; Liu, D.; Peng, H.; Chen, Y. Access Control Scheme Supporting Attribute Revocation in Cloud Computing. In Proceedings of the 2021 International Conference on Networking and Network Applications (NaNA), Lijiang City, China, 29 October–1 November 2021; pp. 379–384. [CrossRef]
37. Karataş, G.; Akbulut, A. Survey on Access Control Mechanisms in Cloud Computing. *J. Cyber Secur. Mobil.* **2018**, *7*, 1–36. [CrossRef]
38. Chunge, L.; Mingji, M.; Bingxu, L.; Shuxin, C. Design, and Implementation of Trust-Based Access Control Model for Cloud Computing. In Proceedings of the IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 12–14 March 2021; pp. 1934–1938. [CrossRef]
39. Chakraborty, S.; Ray, I. TrustBAC-Integrating Trust Relationships into the RBAC Model for Access Control in Open Systems. In Proceedings of the eleventh ACM symposium on Access control models and technologies, Lake Tahoe, CA, USA, 7–9 June 2006; pp. 49–58.
40. Huang, L.; Xiong, Z.; Wang, G.; Ye, C. A Trust-Based Cloud Computing Access Control Model. *Int. J. Knowl.-Based Intell. Eng. Syst.* **2016**, *20*, 197–203. [CrossRef]

41. Institute of Electrical and Electronics Engineers. Madras Section; Institute of Electrical and Electronics Engineers. In Proceedings of the 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 15–16 March 2019.
42. Celikel, E.; Kantarcioglu, M.; Thuraisingham, B.; Bertino, E. A Risk Management Approach to RBAC. *Risk Decis. Anal.* **2009**, *1*, 21–33. [CrossRef]
43. Yang, S.J.; Lai, P.C.; Lin, J. Design Role-Based Multi-Tenancy Access Control Scheme for Cloud Services. In Proceedings of the 2013 International Symposium on Biometrics and Security Technologies, Chengdu, China, 2–5 July 2013; ISBAST 2013. pp. 273–279. [CrossRef]
44. Hu, V.C.; Ferraiolo, D.F.; Kuhn, D.R. *Attribute Considerations for Access Control Systems*; NIST Special Publication: Gaithersburg, MD, USA, 2019. [CrossRef]
45. Li, J.; Bai, Y.; Zaman, N. A Fuzzy Modeling Approach for Risk-Based Access Control in EHealth Cloud. In Proceedings of the 2013 12th IEEE International Conference on Trust, Security, and Privacy in Computing and Communications, Melbourne, VIC, Australia, 16–18 July 2013; TrustCom 2013. pp. 17–23. [CrossRef]
46. Diep, N.N.; Hung, L.X.; Zhung, Y.; Lee, S.; Lee, Y.K.; Lee, H. Enforcing Access Control Using Risk Assessment. In Proceedings of the Fourth European Conference on Universal Multiservice Networks (ECUMN'07), Toulouse, France, 14–16 February 2007; pp. 419–424. [CrossRef]
47. Ngoc Diep, N.; Lee, S.; Lee, H. Contextual Risk-Based Access Control. *Secur. Manag.* **2007**, *2007*, 406–412.
48. Tang, B.; Li, Q.; Sandhu, R. A Multi-Tenant RBAC Model for Collaborative Cloud Services. In Proceedings of the 2013 11th Annual Conference on Privacy, Security and Trust, Tarragona, Spain, 10–12 July 2013; PST 2013. pp. 229–238. [CrossRef]
49. Madani, M.A.; Erradi, M.; Benkaouz, Y. A Collaborative Task Role-Based Access Control Model. J. *Inf. Assur. Secur.* **2016**, *11*, 348–358.
50. Jin, X.; Krishnan, R.; Sandhu, R. Role and Attribute-Based Collaborative Administration of Intra-Tenant Cloud IaaS. In Proceedings of the 10th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, Miami, FL, USA, 22–25 October 2014; pp. 261–274. [CrossRef]
51. Nazerian, F.; Motameni, H.; Nematzadeh, H. Secure Access Control in Multidomain Environments and Formal Analysis of Model Specifications. *Turk. J. Electr. Eng. Comput. Sci.* **2018**, *26*, 2525–2540. [CrossRef]