World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:14, No:12, 2020

Survey of Access Controls in Cloud Computing

Monirah Alkathiry, Hanan Aljarwan

Abstract**—**Cloud computing is one of the most significant technologies that the world deals with, in different sectors with different purposes and capabilities. The cloud faces various challenges in securing data from unauthorized access or modification. Consequently, security risks and levels have greatly increased. Therefore, cloud service providers (CSPs) and users need secure mechanisms that ensure that data are kept secret and safe from any disclosures or exploits. For this reason, CSPs need a number of techniques and technologies to manage and secure access to the cloud services to achieve security goals, such as confidentiality, integrity, identity access management (IAM), etc. Therefore, this paper will review and explore various access controls implemented in a cloud environment that achieve different security purposes. The methodology followed in this survey was conducting an assessment, evaluation, and comparison between those access controls mechanisms and technologies based on different factors, such as the security goals it achieves, usability, and cost-effectiveness. This assessment resulted in the fact that the technology used in an access control affects the security goals it achieves as well as there is no one access control method that achieves all security goals. Consequently, such a comparison would help decision-makers to choose properly the access controls that meet their requirements.

Keywords**—**Access controls, cloud computing, confidentiality, identity and access management.

I. Introduction

CLOUD Computing (CC) produces on-demand services over the Internet. The CC has several computing resources such as network, storage, application, servers, etc. NIST defines the cloud as "a model for enabling ubiquitous, 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" [1].

CC has three different service models, which are: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Service as a Service (SaaS). In the IaaS model, users will be provided with the whole required infrastructure for CC, such as Amazon EC2. As for the PaaS model, a CSP will give the user the needed platform to develop the desired applications, such as Microsoft Azure. The last model, which is SaaS, the user will get the software; however, it will be provided over the internet, such as Dropbox [2].

As for the deployment models, CC services can be provisioned in four different manners: public, private, hybrid and community. In the public cloud, the users will be provided with an access to the cloud services only over the internet; i.e. it is accessible publicly whereas in the private model, it is the opposite; the users will have the infrastructure to get the cloud services inside their network, which means less cost in the long term and better security in the long term as the users will have the complete control over the service. In the community cloud, this model is somewhat similar to the private; however, this service is shared by a number of organizations that have similar requirements and objectives. That last deployment model is hybrid cloud, in which users use two or more different deployment models based on their requirements [2].

The world has been witnessing rapid changes and developments in technologies, leading to an explosion of data, which can be seen in big data technology and businesses over the net; as a result, a considerable percentage of users and organizations have chosen to outsource their data to cloud services. On the other hand, there has been an increase in data leakages and breaches incidents and attacks targeting data, especially data stored on the cloud. Therefore, the world needs appropriate mechanisms and techniques to secure and manage access to data stored on the cloud, in particular. Thus, several access controls must be implemented.

Access control is the mechanism that specifies who (subject) and how to access the resources (objects) on a system and decides whether or not a user is able to get resources from that system. There are different ways to implement access controls; it depends on what an organization wants to protect and the type of the organization. Moreover, dealing with commercial data is different from healthcare data or government data.

This paper is organized as follows; in Section II, we review some of cloud security issues. In Section III, we provide some of background knowledge. In Sections IV and V, we discuss some of main cloud access control solutions in CC and summarize our assessment. Finally, we conclude this paper in Section VI.

II. CC Security Issues

CC provides the world with various services. It stores a large amount of data and information on the cloud. So, when the user needs to retrieve the data from the cloud, the cloud should focus concerns on different issues that can be exploited by vulnerabilities during data transmission, which might result in the exposure of confidential data and the leakage of information to unauthorized parties, which would result in enormous havoc. The security issues that the cloud faces can be summarized as data security, integrity, authentication, and confidentiality. Given that, it is an important concern to build and implement secure access controls that achieve security goals which guarantee access by authorized users only.

Monirah Alkathiry and Hanan Aljarwan are with the Imam Mohammed Ibn Saud Islamic University, College of Computer and Information Sciences, Saudi Arabia (e-mail: malkathiry@sm.imamu.edu.sa, haljarwan @sm.imamu.edu.sa).

A. Integrity

Integrity guarantees that data cannot be altered or modified improperly by unauthorized users.

B. Availability

Availability guarantees that data are available for legitimate users when they need it.

C. Confidentiality

Confidentiality guarantees that data can be seen only by authorized parties. The need for the concealment of secret data arises due to building secure control communication on the cloud that guarantees data stored on the cloud can only be accessed by authorized parties and prevents unauthorized parties. The important question associated with confidentiality is how to ensure the secrecy of the data. There are a variety of techniques that can be used to achieve the goal of confidentiality using cryptographic methods (encryption and Blockchain).

III. Background

- Authentication is a process in which an identity of an entity, such as a program or a person, is verified. This verification can be done by software, for example [7].
- Authorization is a process of allowing or denying access to a certain resource based on several factors related to the authenticated user. In other words, it is the process that specifies who and what can be performed on the resources [7].
- XACML stands for eXtensible Access Control Markup Language [3]. It is an attribute-based access control. It is also considered to be a policy-based access control
- RSA stands for Rivest–Shamir–Adleman. It is one of the most robust public-key cryptographic algorithms used to generate tow keys. The first key is a public key that is employed to encrypt data, and the second key is a private key used to decrypt data [4].
- Advanced encryption standard (AES) is a modern encryption standard which considered to be secure against all known attacks. It has a 128 bits block length and 128/192/256-bit keys [4].
- Blockchain technology is blocks connected to a chain that holds the information; each block has three main attributes: the data, the hash value of its data, and the previous block's hash value although the Blockchain employs a Proof-Of-Work (POW) procedure to prevent tampering with blocks. The notion of POW work is to shorten the creation of a new block, which will raise the security of Blockchain. Finally, the blocks on the Blockchain are distributed by a peer-to-peer network (P2P), so when any new user gains access to the Blockchain, it takes a full copy of the blocks.
- Shamir secret sharing scheme is one of the cryptographic methods used on the private key. The benefits of using it are the decentralized risk and flexibility. It splits the private key into several parts (n) stored on SHARD and can be constructed from (t) parts, so it has to assemble and reconstruct the key [5].
- Ethereum is an open-source decentralization platform for the applications [6].
- CP-ABE is an attribute-based encryption scheme consisting of authority, owner, user. The authority is responsible for issuing the users secret keys, the owner designs the access policy that is used to encrypt the data and the user is the one who accesses the encrypted data when its attributes satisfy the access policy of the file.

IV. Cloud Access Control Mechanisms

A. Identity and Access Management (IAM)

In CC environment, IAM is considered to be one of the major security concerns. This could be a result of the fact that IAM is handled by the CSPs, who, in most cases, do not satisfy the requirements of the users [7].

Data storage and processing are the responsibility of the customer (organization) or a third-party; however, CSPs are the one responsible for securing the whole infrastructure including the data; meaning, the customer has to be certain that the credentials he uses to be authenticated are safe and secure. Therefore, using IAM systems became crucial as they perform several security functions, such as authentication and authorization. In other words, IAM systems make sure that only verified users with the appropriate access rights can access the cloud services and information [7].

Given the special and crucial nature of CC, strong IAM mechanisms are needed because failing to do so could lead to undesirable consequences, such as data leakage. Such mechanisms mainly include authentication and authorization mechanisms [7].

1) Authentication Mechanisms

In general, there are three categories for authentication techniques [8]:

- *Something you know:* username and password, PIN, Implicit Password Authentication System (IPAS)
- *Something you have:* such as hard tokens and smart cards.
- *Something you are:* biometric technologies (e.g. fingerprint recognition)

IAM systems have various authentication techniques depending on the nature of the resource to be accessed. Therefore, there are physical and digital authentication mechanisms.

- Physical: Physical authentication techniques are usually used to protect the access to cloud service facilities, such as cloud data centers (CDCs). A CDC is a central location that has the networks, servers and applications in one place in order to make it easier for the users to access the resources and information. Examples of physical authentication mechanisms used to secure such centers are biometric access controls, such as fingerprint recognition, facial recognition and retina or iris recognition, and access cards [7].
- Digital:

a) Credentials and Secure Shell keys: The use of credentials

provides a proof that a user is authenticated and entitled to access the resources. One of the famous methods of access credentials management in the CC environment is the use of Active Directory (AD) and Lightweight Directory Access Protocol (LDAP). Their servers can be managed by the organization itself or by a third-party. With these techniques, the organization should always make sure to properly manage the accounts (e.g. add or remove), when an employee joins, resigns or leaves the organization [7]. As for the Secure Shell Keys (SSH) techniques, it uses a challenge-response authentication or public key cryptography to identify the SSH server without the need to insert a password over a network, which is significantly helpful to protect against password brute force attacks and interception attacks.

b) Multifactor authentication: In addition to the traditional credentials, another security factor (extra layer of security) is used. Examples of such second factors include One Time Password (OTP), Patterns and Captcha. OTP is mostly used with the online financial transactions. The server uses a certain algorithm to generate the OTP, which is sent to either the user's phone number or email. There is another way to generate the OTP, which is the use of hardware or software tokens. These tokens give the user an OTP to be used only once in a certain amount of time [7]. Patterns can also be used as a second factor. It comes in different forms. For example, web applications use matching image selection whereas mobile applications use dotted patterns. Additionally, security questions are considered to be one of the patterns techniques in which a user chooses and answers a number of security questions from a predefined list of questions, which he has to answer correctly whenever he needs to be authenticated [7].

c) Chip and Personal Identification Number (PIN): This authentication technique is primarily used with financial transactions. This technique uses encryption, specifically asymmetric encryption, where public and private keys are used to encrypt and decrypt the data. The chip has a microprocessor which stores user's data and the security key. The security key is used to encrypt and sign the communication between the authentication server and the client. The authentication server stores the security keys to verify the signature as well as decrypt the communication. As for the PIN, it is used to perform the authentication of the client so it can access user's data and keys stored on the chip [7].

2) Single Sign-On (SSO):

Often, a user can subscribe to enormous cloud services. However, the regular authentication mechanisms may not be suitable to be used with them for a number of reasons, such as a user must memorize many login credentials as well as it might allow for software piracy. Therefore, a central login mechanism was needed, which is known as Single Sign-On (SSO). Such a technique can save the users' time; meaning, they will not need to login for every service for which they subscribed. It is also better when it comes to security matters; better auditing and monitoring. There are different technologies that can achieve SSO: Security Assertion Markup Language (SAML), Open Authentication (OAuth) and OpenID [7].

a) Security Assertion Markup Language (SAML): SAML is an open standard for authentication, which can be used for communication between two participants, which are identity provider and resource provider (application). This technique uses tokens for requests and responses. SAML makes sure that user authentication to the application is achieved securely by first encrypting and encoding the data transmitted between the two parties as well as the fact that SAML tokens not having any information regarding user's credentials [7]. SAML can be used to achieve SSO for web browsers and provides interoperability. This can be accomplished by performing some specifications on the web browser SSO profile, which include service provider, identity provider and user's role [7]. The way this mechanism works is whenever a user wants to access the application, the application communicates with the identity provider for authentication assertions, according to which the application decides whether to allow or deny user's access [7].

b) OpenIDs: OpenID is also an open standard authentication protocol. It can be used to authenticate a user to multiple websites, which are called relying parties (RP). To achieve this, identity providers are also needed. OpenID can be used as a mechanism for SSO as it allows a user to have one pair of credentials to login to several websites and applications. The way this works is that the identity provider creates a list of the users' credentials, which users can use to create their accounts. These users can use these accounts to log into any application that supports OpenID [7]. Additionally, OpenID Connect (OIDC) is the most recent version of OpenID protocol, which is built on OAuth. This version has a number of great advantages; for example, it offers the feature of communication encryption and signing between the communicating parties. This protocol can be used with regular applications as well as mobile ones [7].

c) Open Authentication (OAuth): Open Authentication is an open standard for authentication, too, which could be one-way authentication or mutual authentication. It facilitates user authentication to multiple websites and services without these websites having to share users' passwords. These websites are called relying parties (RP). The authentication process requires the use of so-called OAuth token provider, with which RPs need to register. Once registered, these websites will be able to acquire user's identifiers and secrets [7]. The way this works is that whenever a user wants to login to a website (RP), the authentication is achieved by redirecting the user to the identity provider. After a successful user authentication, several pieces of information will be shared by the identity provider, which include OAuth access token,

client secret, client ID as well as a refresh token. Such information will be obtained by the RP to be used. For example, the RP will use the refresh token to renew the access token when it expires [7]. The latest version of OAuth is OAuth 2.0. However, this version does not provide digital signature for the access token [7].

3) Authorization Mechanisms

There are various authorization mechanisms as follows:

a) Mandatory access control (MAC): MAC is considered to be one of the traditional mechanisms of authorization. In this mechanism, the operating system or security kernel are responsible for assigning the access permissions. Meaning, the data owners do not have the ability to give or deny the access permissions to the resources. So, these access permissions are set by the system manager and enforced by the OS and security kernel [7]. In this technique, objects are assigned a classification, such as secret, top secret and confidential, and subjects (clients) are assigned classification and clearance as well. So, whenever a subject wants to access a certain resource, the OS or security kernel views the subject's credentials and clearance information in order for it to decide to either allow or deny the access [7]. MAC has both advantages and disadvantages. MAC offers a better level of security; however, MAC does not have the desirable flexibility for processing the access permissions as well as it requires constant tracking and updating of all classification and clearance labels [7].

b) Discretionary access control (DAC): In this mechanism, the data owner is the one responsible for assigning access permissions to the subjects. The authorization process is done during the authentication process; meaning, when the username and the password are validated [7]. This technique is called discretionary because the control over the access rights is the data owner's responsibility. DAC, just like any other authorization mechanism, has its advantages and disadvantages. Compared to MAC, DAC is more flexible; however, it offers a lower level of security [7].

c) Entitlement/Task based access control: This mechanism means a subject has a specific and different access permission for every single task, process and action. As mentioned earlier, this model has its advantages and disadvantages. This method is significantly capable of dealing with complicated access conditions to either allow or deny the access. Nevertheless, it might cause some inconvenience as one subject (user) must submit extremely many requests that need to be approved. Additionally, it requires proper maintenance of the entitlement sets [7]. This model is able to implement some of the hierarchal access control mechanisms, such as attribute-based access control and role-based access control [7].

d) Role based access control (RBAC): In this model, access permissions are granted according to user's roles and privileges. Moreover, other factors are considered: user's roles, role permissions and role-role relationship [7]. There are two categories of the roles: application/ technical role and organizational/business role.

- *An application/technical role:* It is a mixture of various tasks or entitlement-based access permissions that are related to one certain application.
- *Organizational/business role:* It is created according to the job functions and permissions that were assigned to an employee. An organization role is a group of several application/technical roles.

To assign permissions, RBAC uses three rules: role assignment, role authorization and permission authorization. This type of access controls is suitable to organizations that have a great number of employees. It offers them security on the administration level. The advantage of this model is that it offers a better security when it comes to granting access permissions. However, it requires constant update as roles assigned to a user usually change every once in a while [7].

e) Attribute based access control (ABAC): This mechanism uses policies to control access permissions. These policies are created using various attributes like resource attributes, subject attributes, object attributes and environment attributes. They are used to determine a user's access rights. In this technique, a user's privileges and roles are predefined [7]. ABAC has several advantages, such as a flexible implementation and an enhanced compliance; it also solves a number of authorization-related issues [7].

f) Trust based CC access control model: This model is proposed by a number of researches as an improvement to the traditional role based trusted access control model. It uses several factors and elements to make the decision to whether or not grant access to users, which include user credibility, trust threshold value and trust grade. Also, this model takes into consideration the dynamic nature of cloud, so, even the permission granting process is dynamic [9]. This model first starts with a user authentication; if authenticated, the user is a legal user; if not, he is an illegal user. After successful authentication, this model produces the user credibility value, which corresponds to a trust grade, to be used in another authentication phase, in which this trust grade will be compared to the trust threshold value. The result of this comparison will determine user access permission; if it is larger than the threshold, then he will be assigned a proper role along with access permissions. Nevertheless, these granted permissions are not permanent or constant; there is a component called trust center that does real time monitoring to examine and analyze user's behavior and then update user's credibility value, which could cause a role re-assignment. This is what makes the permission granting process dynamic [9]. Moreover, access permissions vary according to the trust grade. In other words, there are four modes of authorization that correspond to user's credibility value: zero-level permission, level-1 permission, level-2 permission and level-3 permission. Zero-level permission is the lowest

and level-3 permission is the highest; meaning, level-3 permission has top permissions to access the resource whereas zero-level permission does not [9]. The main advantage, as stated by the researchers, is that this model is able to stop malicious attacks from happening from the beginning. Additionally, the experiment they conducted demonstrated a high capability of resistance against malicious attacks [9].

B. Blockchain

This field of study deals with a concept of how to enhance security when the user or organization preferred to use the cloud service. There are several techniques used to guarantee the security goal that is applied on the cloud, such as using a trusted third party to control the decryption process when the user authenticates and ensures that data must be encrypted before it is uploaded on the cloud server. However, in the case in which the third party or the cloud server is malicious, exposing user's data should be prevented. So, the main focus of those experiments is to design a method using Blockchain technology to prevent the third party or the cloud service from obtaining the private key that is used to decrypt the data stored on the cloud, which causes a data leakage using the decentralization concepts [10].

Blockchain-Based Access Control (BACC) [10] proposed the decentralized network on the smart contract to distribute the access control using the Shamir secret sharing scheme. The system model consists of the data owner, data user, the cloud storage server, Blockchain which consists of four smart contracts (Auditing Contract (AUD-Contract), Access Control Policy Contract (ACP-Contract), Look Up Contract (LKP-Contract), and Contract Look Up (CLU-Contract)) and the data user.

The system works as follows: When the data owner appends new information to the cloud server, s/he needs first to encrypt the data using (AES-128) then upload the encrypted data to the cloud. After that, the owner needs to divide the decryption key using the Shamir secret sharing scheme into n pieces then distributed this onto the node master. Moreover, the owner needs to specify the legitimate users by assigning access rights to them using ACP-Contract; besides that, the owner should keep track of all actions that are performed to access the data on the Blockchain, so the data owner deploys AUD-Contract which stores required auditing information sent by cloud provider via a transaction process. In the case in which the user requests to download the data from the cloud storage server, the cloud server should first confirm the access rights form ACL that stores a list of legitimate users and access rights granted in a table that is stored on the ACP-Contract. After the user is granted access, the cloud sends the encrypted data to her/him. When the user receives the encrypted data, s/he needs to get the decryption keys, so, the user sends the transaction to the LKP-Contract address of the (t) master nodes; as a result, the user receives the decryption key parts then s/he reconstructs the decryption key and decrypts the data. Furthermore, the idea of [11] introduces a cryptographic method using the blockchain technology together with CP-ABE to secure and control access to cloud storage. For the Blockchain, they choose an Ethereum environment, which is a decentralized application; as a result, there will be no trusted third party. For controlling the access, the researchers [11] used period time attribute and set access structure as an access tree, which has leaf nodes and non-leaf nodes; the former represents the attribute, and the latter represents the threshold. Even so, this experiment grants authenticity through a smart contract, which is a program stored on the Blockchain, and it operated, and the result executed in the Blockchain system container.

The system components consist of four entities—the first entity is the cloud server, which stores the encrypted data. The second is data owner who is responsible for creating a public and master key, a smart contract, encrypting the data, uploading the encrypted data file to the cloud server and policies that control access to its data. Third İS data user who requests access to the encrypted data, and the last component is the Ethereum Blockchain, where the smart contracts are deployed on Ethereum.

The system model works when the data owner uploads data to the cloud server. In details, data owner uses the AES encryption algorithm to encrypt the data (file and its ID) using a symmetric key (CK) and hash256 function to hash the file ID. The resulted hash is sent to the smart contract to be stored into Ethereum. Then the owner creates the package, which contains the contract address, file ID and encrypted data, and then uploads it to the cloud server. Likewise, s/he encrypts the PK and CK and the access tree and saves the result into the smart contract.

The decryption process works when data user sends a data access request to the data owner. Hence, the owner will initialize the access period attribute then create a common key to share it with the user using the Diffie Hellman exchange key. As a result, the owner will encrypt the private key (SK) that is generated by the master key using the AES algorithm then transmit it into the smart contract. After that, the data user will download the encrypted package from the cloud then decrypt the data using the CK that s/he receives via the smart contract within the encrypted data and SK` if access period time is valid. So, s/he will decrypt the SK` using the common key then decrypt the encrypted data to gain the CK that is used to decrypt the encrypted package.

C. RSA

Data stored on the cloud must be on a ciphered format to ensure the confidentiality security goal. To achieve that, the data owner needs to employ a cryptographic technique on the data before s/he uploads the data into the cloud server. Moreover, building a secure access technique ensures that no malicious entities gain access to the cloud's data.

Research has been conducted in which the researchers use the three main concepts that will ensure the aim of confidentiality, and secure data (files and images) sharing access [12] through the RSA encryption algorithm with CP-ABE Ciphertext Policy attribute encryption schema and time-based access control. The proposed solution utilized two

attributes category in the CP-ABE. The first attribute they have chosen is the continent, and the second one is the gender of the user (female or male) because ABE has two weaknesses which are non-efficiency and non-existence of attribute revocation mechanism. Users may share the same attribute, but the private and public keys are associated with exactly one user. So, they used the time attribute to give the user a time access duration, which is employed to combine the revocation feature. The system worked as follows: the data have to be encrypted before uploading it to the cloud using the RSA accompanied by the CP-ABE then time control is added; this will generate the password and the encrypted data. Furthermore, the decryption function works only for legitimate users who have a matching key, attributes, and allowed time so they will be able to download, then decrypt the data from the cloud.

Another research [13] demonstrates two phases: the first is sub-parameters of the system and the second is the complete scheme. The first phase consists of two stages that deal with data security protection using encryption and access control. In the encryption stage, they used the RSA algorithm to gain high confidentiality and deliver privacy. However, they applied some development on the RSA on the way the keys work. To demonstrate that, the first key is used to execute operation by the third party on the encrypted data, which the researchers named as a valuation key (VK), and the second key known only by the data owner is private key (SK), which is utilized to encrypt and decrypt data.

The access control stage applies the RBAC module that represents the role and privilege to merge it with XACML.

The complete schema phase works when the data owner generates the keys. The owner encrypts data using the SK then stores the encrypted data on the cloud; after that, it sends the SK and VK to the third party. When the user requests access to the data, first s/he sends a request to the owner and access manager. Then the access manager checks the user permission through several steps and returns the access policy. So, all permissions will be checked for subject and process; after that, if everything goes right, the third party will send the encrypted data concatenated with SK to the user to decrypt data to gain access to it.

D. Time-Based Access

The researchers [14] employed the Time Released Encryption (TRE) and CP-ABE concepts then merged them to gather and develop the time and attribute factors schema (TAFC). The system has four attributes: the data owner (DO), the data user (DU), CSP, and the central authority (CA).

TABLE I
DIFFERENCES BETWEEN ACCESS CONTROLS THAT ACHIEVE CONFIDENTIALITY

	Access Control Techniques	Cryptography Technique	Encryption and Decryption Method	Third-Party	Security
			Blockchain		
BACC [10]	ACL at ACP-Contract.	Blockchain	AES-128 Symmetric key. Shamir secret sharing scheme.	No third party Decentralization	➢ ACL can be modified only by data owner via ACP-Contract. ➢ Authentication. ➢ Authorization. ➢ Integrity using AUD-Contract to control auditing. ➢ Confidentiality using AES-128 Symmetric key.
Blockchain-Based access control [11]	Access tree. CP-ABE.	Blockchain AES key exchange: Diffie Hellman	AES Symmetric key	No third party Decentralization	➢ Authorization using period time attribute specifies access time to the document. ➢ Authorization Access tree. ➢ Shared key throw DH. ➢ Confidentiality using AES Symmetric key.
			RSA		
Attribute and time factors combined CP-ABE and RSA based access control scheme for public cloud [12]	Time-based access control. CP-ABE.	RSA.	RSA Matching key, attributes, and allowed time.	Yes -Authority center	➢ Authorization using Time-based access control. ➢ Authorization using Attribute Access tree. ➢ Confidentiality using AES Symmetric key.
A secure CC system by using encryption and access control model [13]	Role-based access control (RBAC) and XACML.	Enhanced RSA.	Private key	Yes	➢ Authorization using (RBAC and XACML). ➢ Confidentiality using RSA
			Time		
TAFC: Public Cloud [14]	TRE. CP-ABE.	Symmetric key	Symmetric key and matching attributes, and allowed time	Center Authority	➢ Authorization using Time released access control. ➢ Authorization using Attribute-set. ➢ Confidentiality using Symmetric algorithm.
TRBAC and SCT [15]	TRBAC.	SCT.	SCT	n/a	➢ Authorization using task and role. ➢ Confidentiality using SCT.

CA is responsible for generating and publishing the public parameter and secret keys to the DO and the DU and managing the system security and protection. After the DO received the public parameter of the system, s/he can encrypt the data using a symmetric key algorithm. Before encrypting, the data owner needs to combine the data with access policy attribute, and time points are released, then encrypts the file and uploads it to the cloud. In addition to that, the DO creates

the TAFC access structure that contains the leaf node represent access policy attribute, time trap-doors that represent time point, and non-leaf nodes represent threshold gate relations, for example (AND, OR) that are stored on the cloud. CA will generate the time token using a trusted time agent then release it to the cloud to be assigned to the TAFC access structure. In case of DU need to gain access to the data, her/his access policy attributes, and current access time matches the access structured tree, s/he be able to decrypt the data.

Different research [15] constitutes a relatively new area that has emerged from Task Role-Based Access Control module (TRBAC) and Substitution Clock and its' Time (SCT). They utilized the TRBAC with the SCT to secure big data stored on the cloud. The model encrypts the data using the clock timing SCT and then provides access according to the user task and role. The TRBAC has four sets. First, the user set is any subject that uses the system, and, second, the session set, which activates the role of the user. Also, the role set, which depends on the user's position in the organization. Finally, the task set, connected to the object, with it, is permission, then the task assigning to the user according to its role that given to her/him.

V. Discussion

We performed an assessment on those different access control mechanisms in terms of various aspects, depending on the nature of the access control itself. Table I demonstrates the results of this assessment.

VI. Conclusion

This paper reviews some of the available mechanisms that the cloud environment can use to manage and control access to the cloud services. Also, we demonstrated the differences between all of the access control mechanisms covered in the paper, which would, hopefully, help the decision-makers to choose the appropriate solution that satisfies their needs.

References

[1] G. A. Osorio, C. S. Del Real, C. A. F. Valdez, M. C. Miranda, and A. H. Garay, "Effect of inclusion of cactus pear cladodes in diets for growing-finishing lambs in central Mexico," *Acta Hortic.*, vol. 728, pp. 269–274, 2006.

[2] J. Surbiryala and C. Rong, "Cloud Computing: History and Overview," *2019 IEEE Cloud Summit*, pp. 1–7, 2020, doi: 10.1109/cloudsummit47114.2019.00007.

[3] A. Anderson *et al.*, "extensible access control markup language (xacml) version 1.0," *OASIS*, no. January, pp. 1–154, 2003, Accessed: 11-Apr-2020. (Online). Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html#_Toc325047066.

[4] S. Douglas R and P. Maura B, *Cryptography Theory and Practice*. 2019.

[5] A. Shamir, "How to Share a Secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979, doi: 10.1145/359168.359176.

[6] Ethereum, "What is Ethereum? | Ethereum.org," 2020. https://ethereum.org/what-is-ethereum/ (accessed Apr. 06, 2020).

[7] I. Indu, P. M. R. Anand, and V. Bhaskar, "Identity and access management in cloud environment: Mechanisms and challenges," *Eng. Sci. Technol. an Int. J.*, vol. 21, no. 4, pp. 574–588, 2018, doi: 10.1016/j.jestch.2018.05.010.

[8] T. S. Lokhande and P. R. R. Shelke, "A Review on Cloud Computing Security Using Authentication Techniques," vol. 4, no. 6, pp. 2015–2018, 2017.

[9] L. Huang, Z. Xiong, G. Wang, and C. Ye, "A trust-based cloud computing access control model," *Int. J. Knowledge-Based Intell. Eng. Syst.*, vol. 20, no. 4, pp. 197–203, 2016, doi: 10.3233/KES-160345.

[10] N. Sohrabi, X. Yi, Z. Tari, and I. Khalil, "BACC: Blockchain-Based Access Control for Cloud Data," *ACM Int. Conf. Proceeding Ser.*, 2020, doi: 10.1145/3373017.3373027.

[11] S. Wang, X. Wang, and Y. Zhang, "A Secure Cloud Storage Framework with Access Control Based on Blockchain," *IEEE Access*, vol. 7, pp. 112713–112725, 2019, doi: 10.1109/access.2019.2929205.

[12] P. Radhakrishnan, "Attribute and Time Factors Combined CP-ABE and RSA based Access Control Scheme for Public Cloud," *Int. J. Inf. Syst. Comput. Sci.*, vol. 8, no. 2, pp. 124–127, 2019, doi: 10.30534/ijiscs/2019/29822019.

[13] G. S. Mahmood, D. J. Huang, and B. A. Jaleel, "A secure cloud computing system by using encryption and access control model," *J. Inf. Process. Syst.*, vol. 15, no. 3, pp. 1–12, 2019, doi: 10.3745/JIPS.03.0117.

[14] J. Hong *et al.*, "TAFC: Time and Attribute Factors Combined Access Control for Time-Sensitive Data in Public Cloud," *IEEE Trans. Serv. Comput.*, vol. 13, no. 1, pp. 158–171, 2020, doi: 10.1109/TSC.2017.2682090.

[15] "Data Encryption using SCT and access control using TRBAC in Cloud Computing for Big Data."