Semantic Time-Based Access Control: A Model for Patients' Data Security in a Cloud Environment

Idowu Mayowa Opakunle, *Moradeyo Oluwatomilola Motunrayo, Olaniyan Abolade Shekinah, Ojoawo Akinwale Olusola, Olawale John Adegboyega
Department of Computer Science, Adeseun Ogundoyin Polytechnic, Eruwa. Oyo State. Nigeria
*Corresponding Author tokenny2003@yahoo.com

Abstract*— Semantic technologies, which were first created for knowledge representation and reasoning on the web, are gaining traction in the business world as a viable solution to complicated information management problems. With the rapid growth of cloud computing, access control policies have emerged as a critical issue in cloud computing security. Since it involves the authorization of a user to access multiple cloud services, access control is critical in the cloud computing environment. Research has shown that several access models such as Discretionary Access Control (DAC), Obligatory Access Control (OAC), Mandatory Access Control (MAC)and Role Based Access Control (RBAC), have been adopted by most cloud systems depending on their demands but have yielded modest results in terms of how cloud resources may be used because, cloud computing is a specialized area. A Semantic Time-Based Access Control model (STBAC) is proposed as a solution to this problem, which addresses the unique security needs of the cloud environment in terms of access to cloud data resources and user access control by applying ontology to cloud resources and generating a time-sensitive key for access to such resources. In this work, a Semantic Time – Based Access Control Model was developed and applied to secure patients' medical records in a cloud environment.*

Keywords*— Semantic, Access control, cloud computing, cloud security.*

I. Introduction

Computers are now a part of our lives. They play major roles in the way we live, interact with people and the way we work. It is now almost impossible to get any work done without the use of a computing device. It is therefore no news that we are reliant on the services of the computer which emphasizes their usefulness in our everyday lives. These days, the development of cloud computing has now made it possible to network heterogeneous computing devices to perform a task that would otherwise be impossible to complete with an isolated computing device (Hayes, 2013).

Cloud computing is a combination of hardware, networks, storage, services, and interfaces that may be combined to provide computing as a service. (Mell and Grance, 2011) This service primarily entails delivering software, infrastructure, and storage through the internet as individual components or as a whole platform depending on customer demand. Advances in cloud computing technologies means that more devices and services are now available in the cloud (Liu, 2005, Wang, 2012). The availability of these devices and services also means that security is now an issue as more attempts are now made to access these devices and services without following proper protocols for interaction.The use of rights to regulate access to sensitive resources is one method of addressing this problem. Access management, in particular, directs access to mutual resources based on the user credentials, resource type, and resource owner privacy preferences (Liu, 2015). The great majority of access control systems, such as Discretionary Access Control (DAC), Obligatory Access Control (OAC), Mandatory Access Control (MAC) Attribute Based Access Control (ABAC) and Role Based Access Control (RBAC) rely on a priority and manually managed rules over resources.

Due to its open environment and minimal user-side control, information privacy and security is one of the most pressing concerns for Cloud.

Access control policies have become an important issue in the security field of cloud computing. Access control is of vital importance in cloud computing environment, since it is concerned with allowing a user to access various cloud resources. As a solution to this challenge, a Semantic Time-Based Access Control (STBAC) model is proposed to handle the unique security needs of the cloud environment in terms of access to cloud data resource and user access control by applying ontology to cloud resource and the generation of a time sensitive key for access to such resources.

II. Literature Review

Over the years, researchers have contributed different solutions to the problem of access control in cloud computing. These literatures are discussed in this section.

Dong et al. (2014) suggested a privacy-preserving cloud data access policy with semantic security. The Cipher-text Policy Attribute-Based Encryption (CP-ABE) scheme was integrated with the Identity-Based Encryption (IBE) scheme in this concept. Each data file is characterized by a set of relevant qualities, each of which was associated with a public-private key pair. Each attribute offers a distinct key to each user since the user's secret key was computed as a combination of the public key and the attribute's secret key. Decryption of a cipher text was only feasible if the user had the relevant qualities to satisfy the cipher text, ensuring privacy.

Liu et. al., (2014) introduced the TimePRE technique for the cloud service provider to execute the re-encryption when a user is revoked. Each data had an attribute-based access structure and a time of access. A user was recognized by a collection of traits as well as a set of time periods that are eligible. This specified how long the user's access right was valid. This system permits a user's access permission to expire after a preset length of time. The difficulty with this technique was that the user cannot be removed from the system at any point because the time was set in advance.

Shared Authority Protocol-Based Authentication (SAPA), a privacy-preserving cloud access policy proposed by Liu and Xiong (2015), was a shared authority based authentication scheme. An attribute-based technique was used to ensure that the cloud user only has access to his own data, and a proxy re-encryption mechanism was used to allow data sharing among multiple users. By using an anonymous access request matching mechanism that gave shared access authority, the model addresses numerous security and privacy aspects such as authentication, data anonymity, user privacy, and forward security, among others but it was only a theoretical model for authentication and authorization but not tested in the real cloud environment.

Cloud access control authentication system using dynamic accelerometers data was proposed by Zhu et al., (2017). The work unveiled a cloud access control system that combines tailored movements with the capacity to sense acceleration of Wireless Identification and Sensing Platform (WISP) tags. Users were only permitted to pass through the access control system if they accurately performed user-defined motions. Authorized users can create their own authentication movements, which not only simplifies everyday use but also improves the cloud access control system's security.

Li et al., (2018) suggested a Power Cloud Access Control (PCAC) technique that enhanced the standard CP-ABE access control paradigm. Because CP-ABE consumes so much time, they argued that PCAC encrypts the symmetric key rather than the raw data. PCACclaims combines the access tree with the Linear Secret-Sharing Scheme (LSSS) to accomplish automated access structure development and efficient operation. In addition, a zero-knowledge verification action audit step was created to protect against fraudulent users.

Thilakarathne, Navod. (2019). Investigated Improved Hierarchical Role Based Access Control Model for Cloud. The work examined existing cloud access control models and their variations, as well as their benefits and drawbacks, in order to discover more related research paths for designing a better access control model for cloud data storage.

Zang et al., (2019). Worked on an Access Control in Cloud IaaS. The work examined the OpenStack, Amazon Web Services (AWS), and Microsoft Azure Cloud IaaS systems' access control mechanisms. New approaches to boost productivity by facilitating secure information and resource exchange across renter were also highlighted in the work.

Sukmana et al., (2019). Proposed a Unified Cloud Access Control Model for Cloud Storage Broke (CSB). A unified cloud access control architecture for centralized and automated cloud resource and access control management across many CSPs by abstracting CSP services was developed. The work provided CSB stakeholders with role-based access control to cloud resources by assigning necessary rights and an access control list for cloud resources and CSB stakeholders, respectively, based on the privilege separation concept and the least privilege principle.

Thilakarathne and Wickramaaarachchi., (2020) proposed an Improved hierarchical role based access control model for cloud computing. They used AES and RSA cryptographic algorithms to implement the proposed cryptographic schema and used public and private cloud to enforce their access control security and reliability.

Garg et al., (2021). Proposed a Dynamic Access Control Solution for Cross-Tenancy in a Cloud Environment. A dynamic system that uses cross-tenancy in access control models in an interoperable environment was developed, allowing it to effectively meet the dynamic nature of the cloud access control environment.

Almutairi et al., (2021) also carried out a Survey of Centralized and Decentralized Access Control Models in Cloud Computing. Existing cloud computing access control techniques based on centralized and decentralized access control models were examined and assessed. Extensive comparisons of each model’s benefits and drawbacks were also carried out in their work. Access control's difficulties and potential research directions was also considered and analyzed.

Ra et al., (2021) suggested a Federated Framework for Fine-Grained Cloud Access Control for Intelligent Big Data Analytic by Service Providers. It uses data-owner-driven privacy-aware cloud data acquisition framework for intelligent big data analytics for service providers and users.They came up with three basic ideas. The first is a novel global identity provider model for federated outsourcing cloud fine-grained access control. The second is a new ambiguous data acquisition mechanism integrated with P-FIPS from a cloud to a service provider, and the third is the Decentralized Audit and Ordering (DAO) Chain mechanism, which ensures the correctness of obtained data to the service provider while also ensuring the owners that their data is only used for the approved purpose.

Kumar, in his work developed a Key Enforced Access Control and Performance Analysis of DES and RSA Cryptography in Cloud Computing. The management of keys in cloud environments was reviewed due to the expanded exposure to various insider and outsider threat agents. The encryption and decryption time for DES, 2DES, RSA & Modified RSA (MDRSA) algorithms were also analyzed and evaluated.

III. METHODOLOGY

A Role-Based Access Control (RBAC) was proposed in this work. Users, roles, and permissions are the three essential aspects of the RBAC concept. The user is the person who has to interact with the system object, roles define the user's level of significance and permissions describe what a user can and cannot see. Individual users were connected with roles, and roles are related with permissions. According to RBAC permission is a pair of objects and operations. Arole is utilized to link people and permissions together. The core item of the system is a user's role, which serves as a connection between the user and a set of permissions that apply to that user. In this concept, a user is a person, within an organization, a role is a function or title that a user has that is connected with power and responsibility. Permission is the organization's or use agent's consent of a certain operation to be done on one or more objects. This is shown in figure 1.

The data that the doctor and the expert will have access to is the patient record in the scenario in figure 1. The doctor may seek a second opinion from a field specialist (an expert), but the issue emerges when the expert requires access to this information. The question is how the expert would have access to the confidential patient information. Is it as a doctor (role), in which case the expert would have complete access to all information as a consulting doctor. The above scenario represents the limitation of the role-based approach in solving access control problem in cloud computing.

3.1 Proposed Semantic and Time Based Access Control (STBAC)

The proposed system is a combination of a semantic analysis of the role based system with the inclusion of a time based key generation to limit access to only authorized users, this will ensure safe communication in the cloud system. The proposed framework is shown in figure 2 and 3

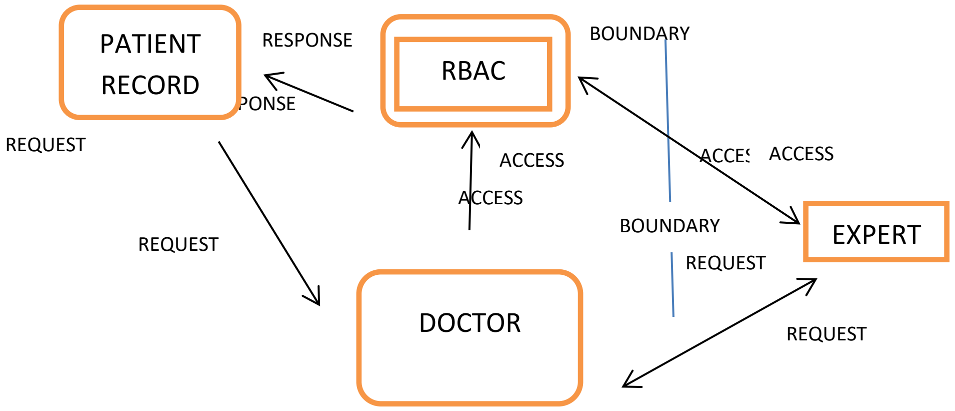


Figure 1: An Example of a Role Based Access Control System

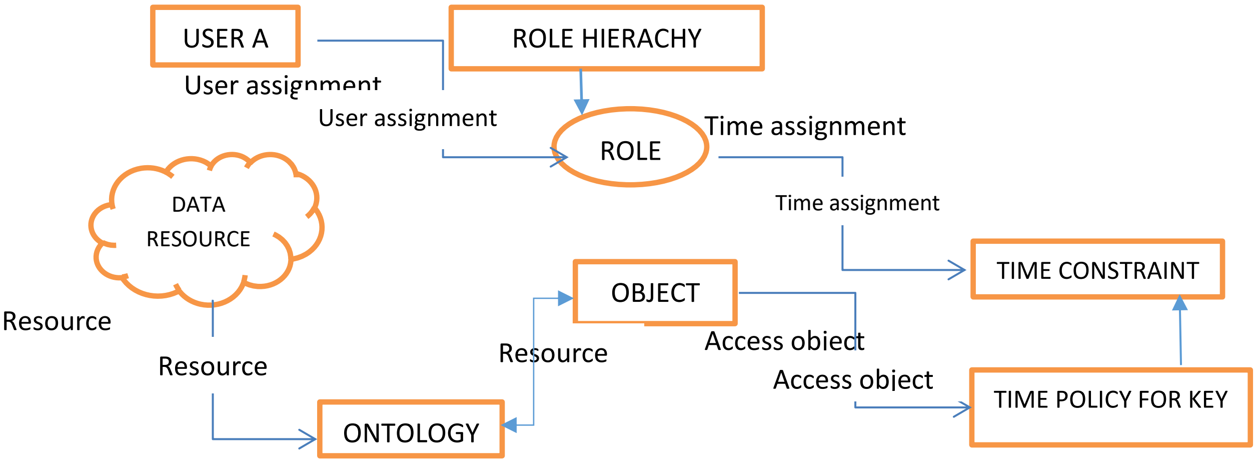


Figure 2: Framework of the Proposed Solution

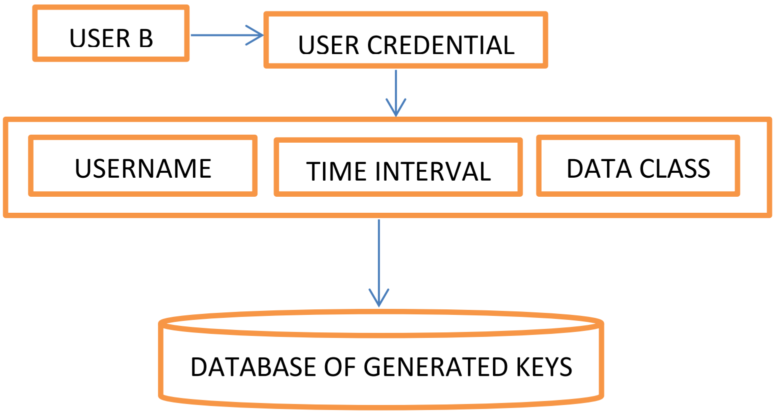


Figure 3 Framework of the Proposed Solution

To tackle the problem of the role based access control model, the role-based system will be converted to semantic propositions instead of making "role" the main connection between users and permissions, the whole process will be given a new approach by creating roles, objects and actions. RBAC involves additional effort from the host organizations in deciding which roles or users from remote organizations should have access to which object.

3.2 Semantic Part ofthe Framework

Ontology is a key component of semantic access control; it establishes a shared vocabulary for persons who need to communicate information in a domain. Ontology is useful in constructing authorization policies throughout the whole cloud computing environment by allowing individuals to have a common knowledge of the structure of information.

Figure 3 shows the framework as proposed by this work. Action is a class that represents an action that a user is allowed to perform in the system. The key word here is "allowed". Resource is a defined class, representing the authorization objects i.e., the objects the user have been cleared to access. We can identify all the objects that have been treated like a resource in the domain ontology. The domain ontology allows us to properly define these objects with little reliance on the role of the user. For example, triple *(r, o, read) onto*, where:
r = role
o = object, and
Read = the action the user wants to perform
Indicates that users assign to role *r* can read information form object *0* defined in ontology '*onto*'.

The subjects (role) and objects possess a set of attributes and access to resources is based on the specification of those set of attributes. Just Like most of the other access control systems proposed by other authors, access control based on semantic ontology system makes its decisions on four domains: *Subject*, *Object*, *Action* and *Attributes*. By modeling the access control domains using ontologies, the semantic nature of the system provides a set of ontologies: *Subjects-Ontology (SO),Objects-Ontology (00), Actions-Ontology (AO),* and *Attributes-Ontology (AtO).*Jarvanmardi et al first proposed this model and for all intent and purposes, this model will be retained for the modeling of the semantic part of the proposed system.

Actions depend on the type of the actions that the subjects intend to execute on an object. Each action type is a concept in the ontology and actions are individual concepts as defined in the action ontology *AO*. The role in subject ontology can be used or manipulated as a subject attribute, depending on the attributes provided by the ontology; a user is assigned to a certain role policy set. Other attributes can also be associated with the subject in order to achieve a fine-grained access control. If a subject is assigned to a role, it cannot access the resources directly but through the *object* ontology specified for that resource. Meanwhile, the roles are organized in a hierarchy. If a role *r1* inherits from role *r2* in the hierarchy, a user with *r2* has all the access rights of *r1*.

3.3 Semantic Policy Language

Subjects, objects, actions, and attributes (as previously stated) can be employed as the fundamental semantic components in an ontology-based semantic access control policy language, and these variables can be further expanded with syntactic elements such as *purposes, conditions, rights,* and *priority*. The majority of policies are stated in the form of regulations. Each policy in semantic access control must be linked to domain knowledge. As a result, domain knowledge is required to derive semantic components for the semantic access control policy.

By modeling the access control domains using ontologies, the semantic part of the proposed system aims at considering semantic relationships with ontology to perform or make decision about an access request. XML-based rule structure is proposed to support semantic policy description model based on semantic access control rules. This was proposed by sun et al. an example will be used to explain the basic elements of semantic access rules. In the example the policy is "*Any worker that has been in the university system greater than or equal to three (3) years can read a staff payment record, but it cannot be modified*". The above example is described by XML-based rule structure below.

XML-based semantic rule structures

```
<rule
        <Target>
                <Subject name = "Anyperson" ontologyRef=
        "SO"/>
                <AttributeVariable name = "worker" type =
                "subject" ontologyRef= "AtO"><Object
                name = "Anyperson" ontologyRef=
                "00"/><AttributeVariable name =
                "paymentRecord" type "object"
                ontologyRef= "AtO">
                <Action name = "read" ontologyRef=
                "AO"/>
        </Target>
        <Right type = "no modification"/>
        <Purpose type = "work" <Condition type = "Equals"
reference = "work more than three years"/>
</rule>
```

3.4 Time-Based Key Generation Part ofthe Framework

The time-based key generation model of the proposed framework discuses the encryption and decryption parts of the key generation process for the case discussed above. For example, if the consulting doctor decides to get a second opinion on a patient's case, the expert who the doctor is to consult would be granted access to the system but there would still be a problem of what the expert is allowed to view by the system and for how many periods of time would the expert be allowed access to the system. The assumption is that the patient record is resided on the cloud with doctors of the hospital having access to patient records and can also grant access to other experts to access the patient record.The semantic part of the framework takes care of the '*what*' question but does not solve the '*how*' part of the question.

The *how* part is what this paper would solve with the inclusion of a time-based key generation model. The model will be leveraging the work of Zhang et al.(2011)

The Semantic Role-based structure discussed above provides a way of integrating access control in the time-based generated key distribution phase in a manner that facilitates generating access keys for different data nodes where each data node in the ontology represents subjects. Suppose that there are a total of *n* doctors and *n* patient in the above example. Consider the key tree in figure 4, the partially ordered set *C,i*, where the vertices are *C = C0, CP1 , : : : , CPm , C1, C2, : : : , CN ,* where each vertex is a security class and diagramed with a node in the hierarchical tree. If *Ci Cj,* we say security class *Ci* is subordinate to security class *Cj* (or security class *Cj* is superordinate to security class *Ci*). *Ci Cj* means that *Ci Cj* and *Ci* ¤ *Cj*. If *Ci Cj*, and there is no *Cx* such that *Ci CxCj*, we say *Ci* is immediate subordinate to *Cj* (or *Cj* is immediate superordinate to *Ci*), which is denoted by *Ci, Cj* and labeled by a directed edge between the two security classes in the tree. To encrypt key data in each security class, the encryptor first establishes encryption key tree as shown in Figure 4 Without loss of generality, let *C0* be the root of the tree. Correspondingly, *K0* is the system root key used to encrypt the data in class *C0*. The nodes in the second level are called patient nodes, which is a token of the patient. All the patient data of the patient are nested under the matching patient node. Therefore, the corresponding key is called patient master key *KPj (j = 1, 2, : : : , m),* which can be distributed to patient *j*, so that the patient can derive keys of lower level when given access right. For example, the key *KP1* is related to node *P1*, which is also the root node. That means the sub-tree rooted with KPj (1 j m) is isomorphic to the tree of key data of patient *j*. Similarly, the keys in lower levels are encryption keys used to encrypt data in homologous classes in the tree.

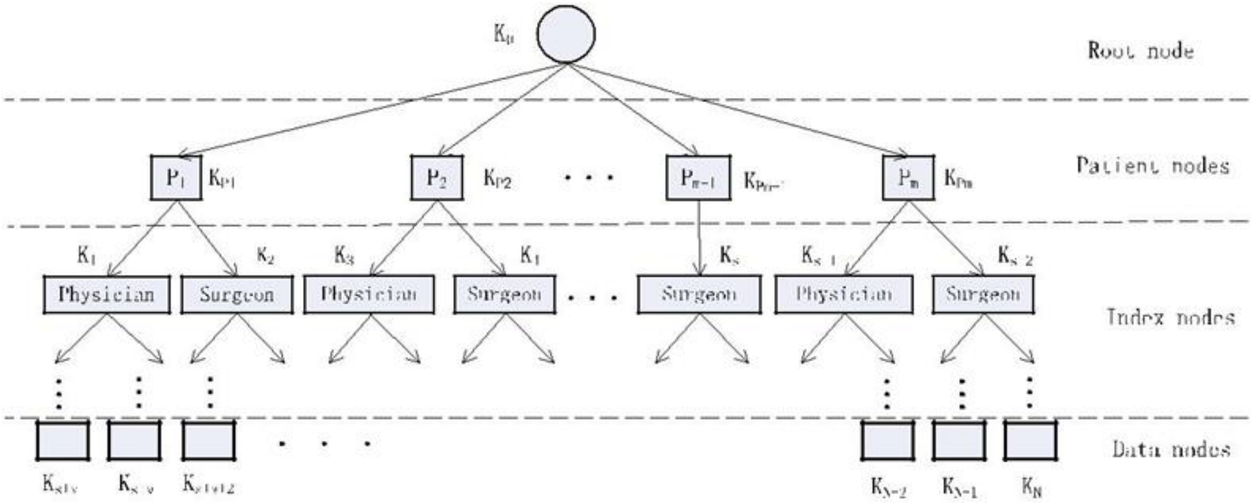


Figure 4: Encryption Key Structures (Zhang et al, 2011)

Using the Role Based Time Bound Access Control (RBTAC) model by Zhang et al in 2011 with modifications to the key generation algorithm, a key that is particular to a doctor was generated which is attached to a particular patient class. What this means is that the doctor that wants to consult an expert in order to get a second opinion on a patient would need to generate a key that will be encrypted by integrating an identifying characteristic for the doctor as assigned by the system and the class of patient under that doctor to generate a key that will be given to the expert to login to the system. An extra modification is also included as part of the generated key would be used to determine if the key has been used and if the key is still valid with respect to time of generation hence the time sensitivity of the key generation process.The above key generation system solves the '*how*' part of the question and hence the reason for the proposed framework.

To generalize this solution, the doctor represents a user, '*A*', of a particular cloud resource (owner); the patient class represents the class of resources owned by that user, '*A*' and the expert represents the user, '*B*', to which user '*A*' wants to safely communicate and share cloud resources with. To achieve safe communication in a cloud environment, there is a need to have a semantic-time based access control system. The user of any cloud resource will be classified as a subject based on some hierarchical rules of ontology and the second party would be required to have a time sensitive key generated by the owner of such resource.

The generated key would now contain the necessary information to login to the cloud and access particular resource which is defined by the ontology from which the key was generated.

In the key generation process, the first procedure is used to determine the class of the data resource that the generated key can access, it determines this by traversing the data like a tree structure with each node representing a data class, once the data class are determined, the second procedure can then take the returned node as an input to generate the key. The second procedure also uses the user credential of the owner of the data and timestamp of the system to generate the key. Then, the key that is generated would contain the user credential, the timestamp during generation and the data class that can be accessed with that key. Since the key is time sensitive, it expires after a certain time, like 24 hours, and the user would need to generate another key for user *B* should there be a need to communicate and share data again between the two parties involved.

IV. Experiment

The simulator is designed using the Microsoft C# programming language, the Microsoft visual studio integrated development environment and the MySQL database management system running on an Apache server. The simulator tries to create an environment in which a cloud access situation is mimicked. This is done to make it possible to explain different aspect of the algorithm. The simulator is simplified as much as possible so as not to lose sight of the main goal of this work which is to show that the proposed algorithm is both secure.

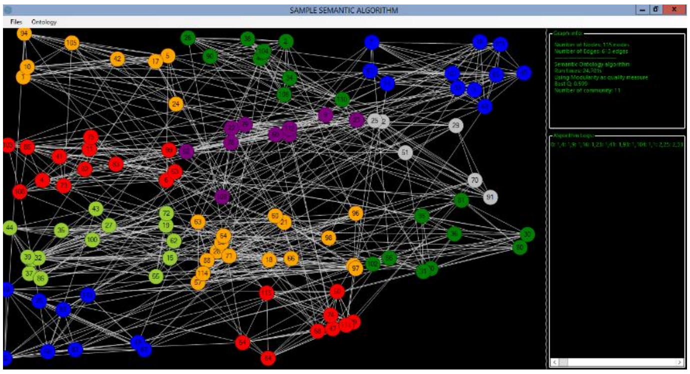
The simulator also considers some simulating criteria so as to properly leverage the proposed model, these criteria are:

- *Utilization/Efficiency:*The access to data is stripped down to just the dataset provided by the program so as to ensure fast access.
- *Throughput:* To reduce the amount of time it will take to first connect to the internet to access data, the simulator connects to the local host instead of an actual cloud environment. The host server is the Apache server which is used to serve the MySQL database.

The data used to simulate the cloud resource owned by a user is a set of graph data that makes for easy classification using some set of semantic ontology algorithm. The data is stored in a text file and is accessed by the simulator during the application of the ontology algorithm. This idea was borrowed from graph theory and the formulation of communities of interest. The idea is applied to the forming of data classes owned by a particular user using some set of ontology.

Each node serves as a data object as discussed in the methodology. A collection of data object forms a data class. The idea of having a data class which is comprised of related data object makes it easy to map each data class to a particular ontology definition. This idea as discussed earlier is firmly rooted in graph theory, and hence is a proven idea that has been applied to different problem domain. One of such problem domain is the formation of community of interest in a social network.

Figure 5 shows a possible data classification based on the ontology used in the simulation. It shows the forming of data classes by the application of a given ontology. The data set used by the simulator has eleven (11) data class which means that a user *A* has applied a set of ontology on his/her data and have successfully divided his/her data into eleven (11) data classes, hence, a user *B* can only have access to whatever data class user *A* wishes to make public and the remaining data class, which he does not wish to make public, remains private. This part of the simulation represents the first part of the work based on the Semantic Access Control Scheme (SACS) model developed by Sun et alin 2012. The ontology model offers a fine grained access to the data resource in the cloud, while the second part of the model ensures security and time based access to such data resource.

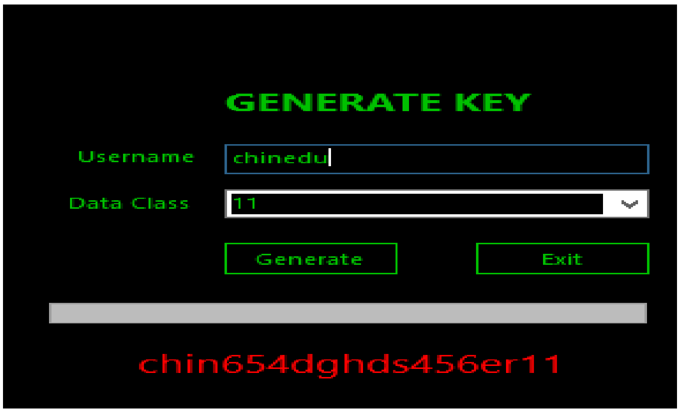


Graph info:
Number of Nodes: 115 nodes
Number of Edges: 613 edges
Semantic Ontology algorithm
Run times: 24.701s
Using Modularity as quality measure
Best Q: 0.509
Number of community: 11

Algorithm Logs:
0: 1,4: 1,9: 1,16: 1,23: 1,41: 1,93: 1,104: 1,1: 2,25: 2,33

Figure 5: Ontology Application on the Dataset

Figure 6 shows the key generation module. This part of the simulation shows the second part of the propoed model.


GENERATE KEY

Username: chinedu

Data Class: 11

chin654dghds456er11

Figure 6: The Key Generation Module

This part of the model involves the generation of a unique key which is both time and data class dependent. The key was generated using three (3) components, which are:

- A part of the username,
- The data class the key is supposed to access, and
- The time interval in which the key is valid.

A sample of such a key can be seen displayed in red in figure 6. It can be seen that the username for the user generating the key is ‘*chinedu*’ and the data class the key is supposed to access is ‘*11*’. By default, the key is supposed to last for twenty four hours (24hr) after which it will become invalid and if a user *B* still wants to access same data, user *A* would be required to generate a new key for user *B*. The time for the key to remain valid can, of cause, be changed depending on the need of the application and the environment of use.

V. Result and Discussion

This section focuses on the discussion of security and privacy analyses of the proposed model. The discussion of the security of the proposed model contains the security against possible attacks as discussed by Zhang et al (2011). basically, successfully accessing and using of cloud resource data must have both valid system identity credential and access credential. In the discussion that follows, it was assumed that a user *A* already is registered and is a valid user of a cloud resource and wishes to share such resource with a user *B*. The first scenario as put forward by Zhang et al (2011) is an adversary without system identity credential, and access credential attempts to access data. Such data may be requested from DB with a forged system identity credential and a forged access credential (or an access credential from other user) as shown in figure 7

Security Analysis

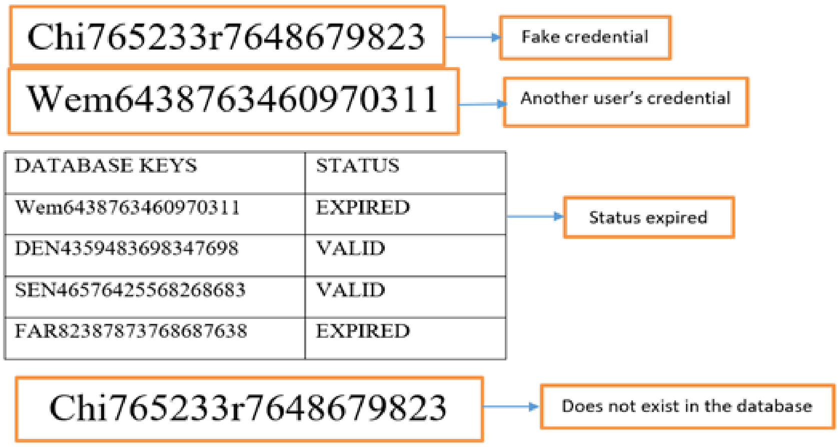
The security of the proposed model contains the security against possible attacks as discussed by Zhang et. al., 2011. Basically, succesful accessing and using of cloud resource data must have both valid system identity credential and access credential. In this further discussion, we assume that a user A is already registered and is a valid user of a cloud resource and wishes to share such resource with user B.

Attacks from the outside

Obviously, the adversary will fail to pass the verification, when DB verifies the signature of system identity credential with the provided key which is assumed to be generated by a user A. In addition, the adversary cannot use the cloud data without valid system identity credential and access credential even if he/she obtains an access credential from other users. This is because, a key is tied to a particular data class and can only access such data class within a time limit.

Attack on Unauthorized Class

The attack on unauthorized class is a malicious user *C* of the cloud system that may attempt to access data of an unauthorized data class which has never been accessed by the user in consecutive time intervals. Assume the user has legally obtained access credential from user *A* i.e., user *C* is known to user A and has been giving a key by user *A* but tries to access data of a different data class other than the one for which the generated key is meant for.



DATABASE KEYS	STATUS
Wem6438763460970311	EXPIRED
DEN4359483698347698	VALID
SEN46576425568268683	VALID
FAR82387873768687638	EXPIRED

Figure 7: Diagram Showing Key Generation Samples

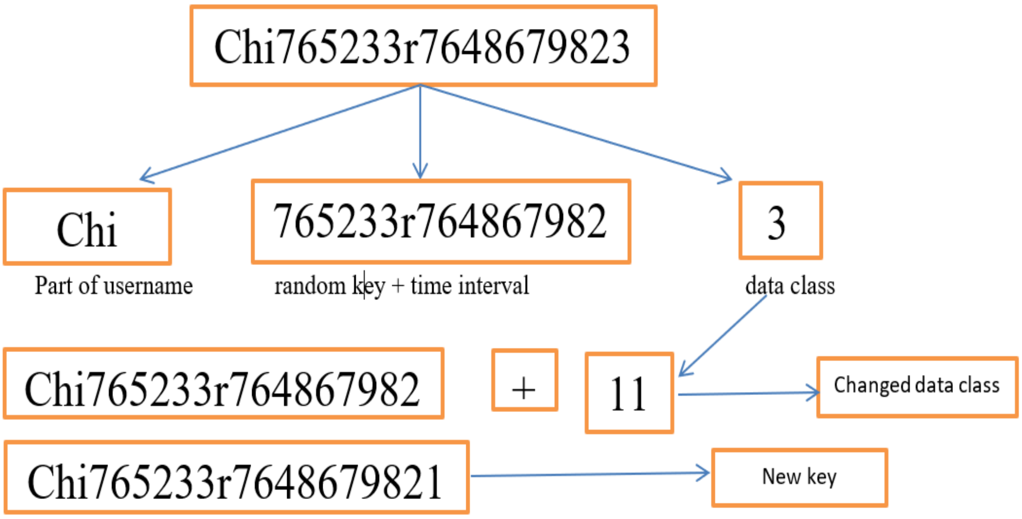


Figure 8: Attacks from the Outside

Attack on unauthorized class.

The user will not be able to access such data as each key is tied to a particular data class and upon verification for access to such data class access will be denied because the key will not be verified and will be considered invalid for access to such data class.

Attack in Unauthorized Time Interval

The third type of attacks is a malicious user *C* of the cloud system that may attempt to access data of a particular data class in unauthorized time interval. Assume that a user A generates a particular key for access to a particular data class and for a time interval of twenty four hours (24hr) and a user C who was given such key tries to access such data after the time validity period of 24hr.

The user will not be verified as the system will require that the user be required to get another key from user A to have access to such data as the time validity period of that key has expired.

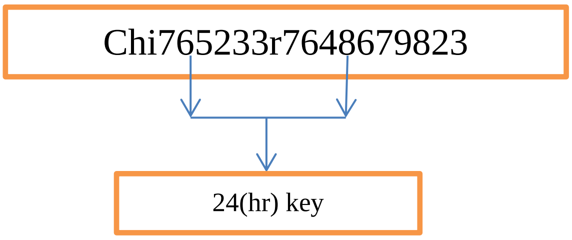


Figure 9: Diagram Showing Attack in Unauthorized Time Interval

Collusion Attack

Some existing time-bound key management schemes are proved insecure against collusion attack, such as Tzeng's scheme is insecure against Yi and Ye's attack, Chien's scheme is insecure against Yi's attack, and Bertino's scheme is insecure against Sun's attack. These collusion attacks can be described in my proposed solution like this: one or two users collude with a user *C* to derive certain access credential and access key of an unauthorized data class, and then try to request cloud resource data of such data class with the forged access credential from DB. Suppose also that user *C*is able to derive the access key from conspirators.

collision attack

Similar to the aforementioned attacks, this conspiracy or collusion will fail because firstly, a valid access is required from user *A* to gain access and that key is time bound and hence only valid for a period not more than that set by user *A*. Secondly, a key is generated based on a particular data class and any collusion of any kind will render a valid key invalid.

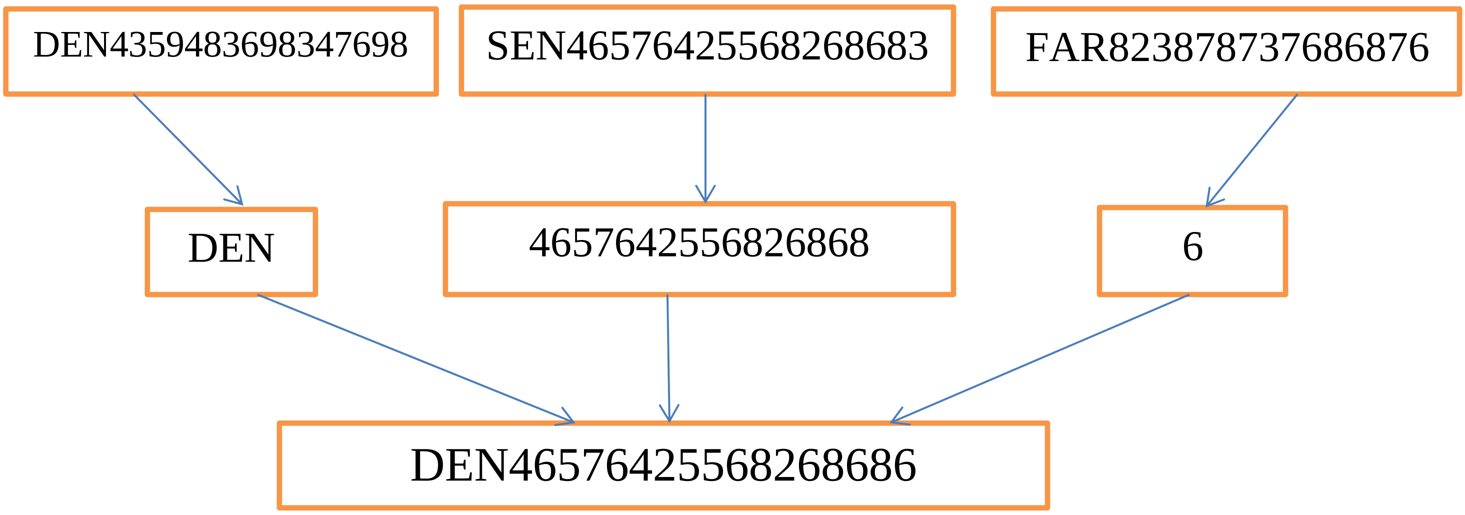


Figure 10: Diagram Showing Collision Attack

Privacy Analysis

The proposed model is capable of providing better privacy for a cloud based on the following aspects:

- When a user accesses cloud data, the proposed model not only narrows the access scope of cloud data by dividing cloud data into different data classes but also limits the access time to the period determined by the owner of such data. The user consequently is only given the right to access cloud data of authorized classes in authorized time interval. Accordingly, the privacy of users can be protected from both space and time dimension.
- All the data nodes are securely stored in remote DB, so that both users and DB are unaware of any intermediate data node and internal structure of the cloud data. It greatly reduces the probability that a deliberate user deduces additional information about the patient from intermediate nodes or the relationship of partial ordered nodes. For example, a user has accessed all blood test results of a patient under different disease category nodes. Then, he may infer other disease information about the patient if he has knowledge of those disease category nodes.
- Unlike most existing key management schemes, the proposed key management scheme does not publicly disclose any relationship value between partial ordered classes, even the relationship between owners and data classes. Thus, it avoids the situation where a particular user could infer other data from access to particular data.

VI. Conclusion

In this work, the semantic time based approach as a secure foundation to achieve access control in cloud computing environment has being proposed. The model is scalable, applicable to different environment and covers other access control models like the semantic based model proposed by Sun et al and the time based model proposed by Zhang et al(2011). The proposed model extends both semantic and time based access control models by considering the semantics of objects and associates permission with concepts instead of objects and generating a time constraint key for the semantic object. Ontology was used for cloud computing environment with highly heterogeneous and structured vocabularies to achieve secure communication between parties in the cloud environment. Considering the limitations of traditional access control method in the cloud computing, this paper introduces semantic web technologies to the distributed role-based access control method and an ontology-based semantic access control in cloud data access control. In the methodology, some syntax elements, such as subjects, objects based on attributes and action were used; and more elements purposes, conditions, rights and priority were added in the proposed model. This approach can easily solve the problem of access in heterogeneous, distributed and large environments and ideal for doing cross organizational work in cloud computing environment. We also implemented a simulator to implement the proposed methodology.

It is important to notice that our proposed semantic time based access control model can be applied to many different fields, especially sensitive information system such as government or military systems, banking systems, and e-commerce systems. The aforementioned description and experiments shows that this model can provide more stringent mandatory access control from both spatial and temporal dimensions and data confidentiality for this kind of system.

Reference

1. Al-Dahhan, R. R., Shi, Q., Lee, G. M., and Kifayat, K. (2018). Access Privilege Elevation and Revocation in Collusion-Resistant Cloud Access Control. In *2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)* (pp. 209-214). IEEE.
2. Almutairi, S., Alghanmi, N., andMonowar, M. M. (2021). Survey of centralized and decentralized access control models in cloud computing. *International Journal of Advanced Computer Science and Applications*, *12*(2).
3. Anderson, J. G. (2000). Security of the distributed electronic patient record: a case-based approach to identifying policy issues. *International Journal of Medical Informatics*, *60*(2), 111-118.
4. Byun, J. W., Bertino, E., and Li, N. (2005). Purpose based access control of complex data for privacy protection. In *Proceedings of the tenth ACM symposium on Access control models and technologies* (pp. 102-110).
5. Cheng, V. S., and Hung, P. C. (2006). Health insurance portability and accountability act (HIPPA) compliant access control model for web services. *International Journal of Healthcare Information Systems and Informatics (IJHISI)*, *1*(1), 22-39.
6. Cirio, L., Cruz, I. F., andTamassia, R. (2007). A role and attribute based access control system using semantic web technologies. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"* (pp. 1256-1266). Springer, Berlin, Heidelberg.
7. Damiani, E., Capitani di Vimercati, S. D., Fugazza, C., andSamarati, P. (2004). Extending policy languages to the semantic web. In *International Conference on Web Engineering* (pp. 330-343). Springer, Berlin, Heidelberg.
8. Dong, X., Yu, J., Luo, Y., Chen, Y., Xue, G., and Li, M. (2014). Achieving an effective, scalable and privacy-preserving data sharing service in cloud computing. *Computers & security*, *42*, 151-164.
9. Garg, S., Mehrotra, D., and Bhartiya, S. (2021). Dynamic Access Control Solution for Cross-Tenancy in a Cloud Environment. In *Security Issues and Privacy Threats in Smart Ubiquitous Computing* (pp. 111-129). Springer, Singapore.
10. Ghaffar, Z., Ahmed, S., Mahmood, K., Islam, S. H., Hassan, M. M., andFortino, G. (2020). An improved authentication scheme for remote data access and sharing over cloud storage in cyber-physical-social-systems. *IEEE Access*, *8*, 47144-47160.
11. Hayes, J. F. (2013). *Modeling and analysis of computer communications networks*. Springer Science & Business Media.
12. Hu, L., Ying, S., Jia, X., and Zhao, K. (2009). Towards an approach of semantic access control for cloud computing. In *IEEE International Conference on Cloud Computing* (pp. 145-156). Springer, Berlin, Heidelberg.
13. Huang, X., Tao, Q., Qin, B., and Liu, Z. (2015). Multi-authority attribute based encryption scheme with revocation. In *2015 24th International Conference on Computer Communication and Networks (ICCCN)* (pp. 1-5). IEEE.
14. Javanmardi, S., Amini, M., Jalili, R., andGanjiSaffar, Y. (2006). SBAC: A semantic based access control model. In *11th Nordic Workshop on Secure IT-systems (NordSec'06), Linkping, Sweden* (Vol. 22).
15. Kagal, L., Finin, T., and Joshi, A. (2003). A policy based approach to security for the semantic web. In *International semantic web conference* (pp. 402-418). Springer, Berlin, Heidelberg.
16. Kumar, Y. K., andShafi, R. M. Key-Enforced Access Control and Performance Analysis of DES and RSA Cryptography in Cloud Computing.
17. Liu, H., Ning, H., Xiong, Q., and Yang, L. T. (2014). Shared authority based privacy-preserving authentication protocol in cloud computing. *IEEE Transactions on parallel and distributed systems*, *26*(1), 241-251.
18. Li, Y., Zhang, P., and Wang, B. (2018). An improved ciphertext-policy attribute-based encryption scheme in power cloud access control. *Applied Sciences*, *8*(10), 1836.
19. Liu, Q., Wang, G., and Wu, J. (2014). Time-based proxy re-encryption scheme for secure data sharing in a cloud environment. *Information sciences*, *258*, 355-370.
20. Mell, P., &Grance, T. (2011). The National Institute of Standards and Technology's (NIST) definition of cloud computing.
21. Noy, N. F., and McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology.
22. Pan, C. C., Mitra, P., and Liu, P. (2006). Semantic access control for information interoperation. In *Proceedings of the eleventh ACM symposium on Access control models and technologies* (pp. 237-246).
23. Park, J., Sandhu, R., andSchifalacqua, J. (2000). Security architectures for controlled digital information dissemination. In *Proceedings 16th Annual Computer Security Applications Conference (ACSAC'00)* (pp. 224-233). IEEE.
24. Priebe T., Dobmeier W. and Kamprath N. (2007) "Supporting Attribute-based Access Control in Authorization and Authentication Infrastructures with Ontologies",*Journal of Software*, 2(1), pp. 27-38.
25. Ra, G., Kim, D., Seo, D., and Lee, I. (2021). A federated framework for fine-grained cloud access control for intelligent big data analytic by service providers. *IEEE Access*, *9*, 47084-47095.
26. Rath, A., Hristoskova, A., and Klein, S. (2021). PFilter: Privacy-Aware and Secure Data Filtering at the Edge for Distributed Edge Analytics. In *IFIP International Conference on Artificial Intelligence Applications and Innovations* (pp. 305-310). Springer, Cham.
27. Sukmana, M. I., Torkura, K. A., Graupner, H., Cheng, F. and Meinel, C. (2019). Unified cloud access control model for cloud storage broker. In *2019 International Conference on Information Networking (ICOIN)* (pp. 60-65). IEEE.
28. Sun, L., and Wang, H. (2010). A purpose based usage access control model. *International Journal of Computer and Information Engineering*, *4*(1), 44-51..
29. Sun, L., and Wang, H. (2011). Access control and authorization for protecting disseminative information in E-learning workflow. *Concurrency and Computation: Practice and Experience*, *23*(16), 2034-2042.

30. Sun, L., Wang, H., Yong, J., and Wu, G. (2012). Semantic access control for cloud computing based on e-Healthcare. In *Proceedings of the 2012 IEEE 16th international conference on computer supported cooperative work in design (CSCWD)* (pp. 512-518). IEEE.
31. Thilakarathne, N., and Wickramaaarachchi, D. (2019). Improved hierarchical role based access control model for cloud. *International Journal of Computer Science Networks*, *8*(5), 2277-5420.
32. Thilakarathne, N. N., and Wickramaaarachchi, D. (2020). Improved hierarchical role based access control model for cloud computing. *arXiv preprint arXiv:2011.07764*..
33. Wang, H., Zhang, Y., and Cao, J. (2008). Access control management for ubiquitous computing. *Future Generation Computer Systems*, *24*(8), 870-878.
34. Wang, H., Zhang, Y., and Cao, J. (2008). Effective collaboration with information sharing in virtual universities. *IEEE Transactions on Knowledge and Data Engineering*, *21*(6), 840-853.
35. Wang, H., Cao, J., and Zhang, Y. (2020). *Access Control Management in Cloud Environments* (pp. 3-297). Springer.
36. Zhu, B., Zhao, J., Li, D., Wang, H., Bai, R., Li, Y., and Wu, H. (2018). Cloud access control authentication system using dynamic accelerometers data. *Concurrency and Computation: Practice and Experience*, *30*(20), e4474.
37. Zhang, R., Liu, J., Han, Z., and Liu, L. (2011). RBTBAC: Secure access and management of EHR data. In *International Conference on Information Society (i-Society 2011)* (pp. 494-499). IEEE.