SERBAC framework on Encryption of cloud data using Raspberry PI

Sunitha B S
Associate Professor, ISE Department, EPCET
PhD Research Scholar, VTU
Bangalore, INDIA

Dr. Anirban Basu
Professor, CSE Department,
APSCE
Bangalore, INDIA

Abstract— **This Cloud computing and its services are quickly rising, as a solution, there is a growing movement to use the data storage in cloud. This has brought up the imperative issue of security of how to minimize and avoid accessing unauthroizing access of data in the Cloud. One of well known understood access controlled model is the Securely Coupled and Extendable Role Based Access Control (SERBAC) [1], which gives versatile controls and administrations by having multiple mappings, rules and roles to users and standards to advantages on information objects. This paper proposes a Secure encryption based on role (SEBR)plan which joins the cryptographic methods with SERBAC. Proposed SEBR scheme permits SERBAC policies need be authorized for encrypted data to be available in public cloud. The proposed work represents a protected SEBR-based hybrid cloud architecture that permits an enterprises to store information safely in a public cloud, while keeping up the sensitive data identified with the organization 's structure in a private cloud. We report the proposed SERB-based design and discourse with the performance results. We prove that users just need to hold a one key for decryption, and framework operations are systematic regardless of the complex role hierarchy and membership of user the organization productive paying little mind to the miscellaneous role of the chain of command and client participation in framework. AES encryption/decryption calculation is used for key based encryption. We have utilized trusted based support key methodology for executing Key based encryption/decoding calculation on an ARM. This methodology made the framework more secure and reliable**

Keywords— Cloud computing, Encryption/Decryption, AES Algorithm

I. INTRODUCTION

There has been an exponential increase in the recent times to store data in the cloud with an improved in the quantity of digital data, such as users' personal information to larger organization need to database backup or store files. Data storage of cloud can be especially attractive for end users with dynamic demand for storage , requiring an cost effective storage tier. With outsourcing consumers' data to cloud storage, Cloud service providers can concentrate to a greater extent on the invention of social functions to improve the user experience of their services without worrying about resources to store the growing amount of data. Cloud can also provide on-demand resources for storage, which can help service providers to decrease their maintenance costs. Furthermore, cloud storage can offer a flexible and convenient means for users to access their information from anywhere on any device. In Securely Coupled and Extendable role-based access control (SERBAC) framework, each user experiences a role along with a request, authentication takes place. The session authenticator service provider verifies the tokens and evaluates the functions based on dual encryption mechanism for the user thus providing the efficiency. The protocols used in SERBAC based on the data integrity preservation and the dual encryption mechanism based on efficient access control system SERBAC. On traditional access control schemes, enforcement is held out by trusted parties which are usually the service providers. In a public cloud, as data can be stored in distributed data centers, there may not be a single central authority which controls all the data centers. Furthermore the administrators of the cloud provider themselves would be able to access the data if it is stored in plain format. To protect the privacy of the data, data owners employ cryptographic techniques to encrypt the information in such a manner that only users who are permitted to access the information as specified by the access policies will be able to do. We refer to this method as a policy based encrypted data access. The authorized users who fulfill the access policies will be able to decrypt the information using their private key, and no one else will be able to expose the data content. Therefore, the problem of managing access to data stored in the cloud is transformed into the problem of management of keys which in turn is determined by the access policies. Consequently, the issue of overseeing access to information kept in the cloud is changed into the issue of administration of keys which is found by the access policies. In this work, we uncover the origination of a SERBAC based cloud storage framework where the access control policies are connected by a most recent secure Encryption based Role (SEBR) that we proposed in the paper. This SEBR plan forces SERBAC approaches on encryption information held in the cloud with a productive user revocation using a broadcast encryption strategy examined in [3]. In our SEBR plan, the holder of the information decrypts the information in such a way, to the point that only the user with proper rules determined by a SERBAC strategy can decrypts and in addition view the information. The rule used is granting permissions to end users who meet the role and can also revoking the permissions from the existing users of the purpose. The cloud provider will not be able to view the content of the data, if the cloud provider is not given the appropriate authority . Proposed SEBR scheme is able to manage with hierarchies in roles, whereby roles derive permissions from other regions. Cloud user is permitted to connect to a role after the owner has encrypted the information for that purpose. The user will have the ability to get to that information from that point on, and the owner does not require re-encrypt the information. A user can be repealed whenever, in which occasion, the revoked user won't access any future encrypted information for this determination. With our proposed SEBR scheme, role based revocation does not influence other cloud users and various users in the enterprise. In summation, decryption computation partly outsourced in the outline of cloud, in which needed public parameters. By utilizing access, SEBR scheme proposes an efficient customer side decryption.

II. RELATED WORK

There exist many hierarchy access control scheme [2], [6] ,[10] Which have been constructed based on hierarchical key management (HKM) schemes and approaches using HKM schemes to enforce RBAC policies for data storage are discussed in [1], [9],[6]. But this scheme has disadvantages that when the user’s access permission is revoked, all the keys known to this user as well as all the public values related to these keys need to be changed. In the traditional control access system, enforcement is carried out by trusted parties which are usually service provider. As we know in public cloud data can be distributed at different data centre. Furthermore when owner of data upload any data to cloud the service provider itself was able to access that particular document. This raised to security issue of the document. To protect the data, data owner uses the cryptographic encryption scheme to encrypt the data in such a way that user who has decryption key was able to decrypt the data and see the original content of the data. But this scheme leads to the problem of management of keys. To overcome the drawback of above system; there is Role Based Access Control (RBAC) model which can be used to protect data which is stored in the cloud. Although cryptographic RBAC scheme have been developed recently to secure data outsourcing, but these scheme assumes the existence of trusted administrator managing all the users and roles, which is not realistic in large-scale system. In this paper ,we proposed Secure extensible Role Based Encryption (SERBAC) scheme [4] which can be used efficiently with RBAC scheme to provide security to data which is stored in the cloud storage. However the revocation of user in this scheme require the update of the all the role related parameter.

III. OVERVIEW OF THE WORK

The proposed SEBR scheme, we are securing cloud data storage architecture using a hybrid cloud infrastructure. This hybrid cloud architecture comprises of private cloud and public cloud, where the private cloud is used to keep only the organization’s sensitive structure information such as the role hierarchy and user membership information, and the public cloud is used to store the actual data that is in the encrypted format. Raspberry Pi is used for client Signup. A Role manager will assign the roles for client based on cloud based SERBAC. The roles assigned to client, each client had been allotted Role 0 ,who is not able to access other client details. The admin had a Role 1, the admin can check only encrypted form data of clients. Characterize one technique to force access control approaches is to change the access control issue into a key administration issue. In the literature review, there exist various access control plans[1] which have been built based on hierarchical key administration (HKM) conspires, and approaches utilizing HKM plans to uphold RBAC strategies for data memory are discussed in [1]. In any case, these outcomes have a few limitations. For instance, if there is an huge number of data owners proprietors and users included, the overhead required into setup key foundation can be high for sure. Moreover, when a users access permit is repudiated, all the keys referred to this user as effortlessly as all the public values related to these keys need to change, which makes these plans unrealistic. Elective methodology for the administration of keys is Hierarchical ID-based Encryption (HIBE, for example, [1].However, in a HIBE plan, the length of the identity turns out to be longer with the development hierarchy . In summation, the identity of a user must be a subset of its ancestor node so that its ancestor node can infer this current node's private key for unraveling. Along these lines, this node can't be determined as a relative node of another role in the hierarchy order tree unless the character of the other part is additionally the superset of this node identity. We presented a secure role based encryption plan (SEBR) in [1]. Nonetheless, the client revocation in this plan requires the redesign of the considerable number of roles related parameters. Another plan was proposed in [1]. In this system, the size of encrypted text increments straightly with the quantity of all the predecessor roles. Also, the administration of the user membership for every individual role requires the usage of the framework secret keys. The plan proposed in this paper conquers these confinements, and every role can utilize its own secret keys to get membership of user without the need to know the framework secret keys. Also, the framework recommended in this paper gives proficient client revocation. Other than SERBAC, there are additionally different access control models, for example, Attribute Based Access Control (ABAC). In ABAC, access is permitted in view of attributes of the user. Frameworks characterize mix of qualities as the access plocies and users need to build up that they have these attributes so as to obtain access. In 2006, the primary attribute based encryption (ABE) framework was proposed in [5] taking into account the work in [1], and some other ABE plans have been proposed a short time later. In these frameworks, information is encrypted to an association of properties, and users private keys connected with these characteristics can decry pt the data. These works have offered an option access to secure the data in distributed environment utilizing an alternate access control system, in [1]; we have exhibited that an ABE plan can be used to implement RBAC policies. However, in that method, the user key size is not invariant, and the revocation of a user will end-result in a key update of all users having same role. [1]

Various approaches to preserve privacy of data in a cloud environment include using direct encryption and proxy re-encryption. In these cryptographic methods, data are appropriated to be encrypted instantly to the users with whom the owners share the data. This is practically identical to the access control strategies in Discretionary Access Control (DAC) model. Subsequently they are ordinarily utilized as a part of associations where the DAC model is received.

A. Architecutre

In this part, we show in figure 1 the secure cloud storage architecture. It is a hybrid cloud architecture comprising a private cloud, which is employed to store sensitive role hierarchies of the organization and user memberships, and a public cloud storing the encrypted data and public parameters associated with the SEBR system. The users who wish to access the encrypted information and the data owners who wish to encrypt their data only interact with the public cloud. The role hierarchy and user to role mappings related to the organization are maintained in the private cloud, which is completely accessible to the administrator of the scheme. The administrator defines the role hierarchy and the role managers who handle the user membership relations.

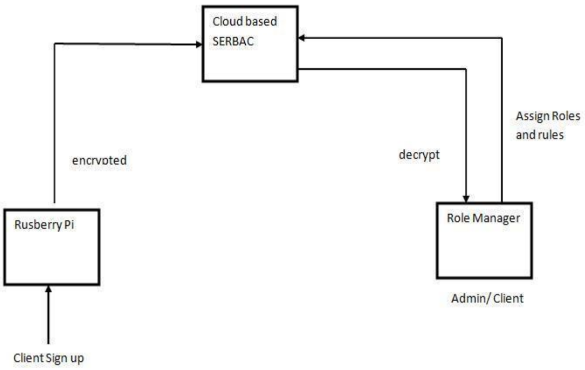

Fig 1: System Architecture

Public cloud is a third party cloud service provider which resides outside the infrastructure of the organizations, and organizations outsource their encrypted data of users to the public cloud. Since the public cloud is untrusted, data stored in the public cloud could be accessed by unauthorized entries, such as employees such as administrator and users from other organizations who are also using services from the same public cloud. Therefore only public cipher text will be stored in the public cloud. An untrusted public cloud will not allow user’s permissions to access data stored in the cloud . Such behaviors will end-result in the users not able to access the information stored in the cloud, but will not yield in policy violation of SERBAC. These behaviors can be observe, as a user can notice the failure immediately after one establish communication with the cloud.

The users wish to download certain data from the public cloud. Administrator authenticate each user in the role-based system; once authentication successful , the user is granted a session key which is linked to the user's identity. Users are not implied in any procedure related to organizational structure updates, including user membership updates and changes in the role hierarchy. So they are not allowed to communicate directly with the private cloud.

Relationship between roles and users are managed by role manager . Each role having its own role parameters which defines the membership of user. Roles parameters are accumulated in the cloud. When user role membership need to update, the role manager will compute role parameters and update them in the cloud. None of users are influenced by this operation, so role administrators don't have to communicate with users, and they just need to communicate with the private cloud. Prior to an user is incorporated into a role, the role manager should confirm the user keeping in mind for role user is qualified . We don't consider the verification components in this paper; we expect that such instruments exist and role managers will allow part membership just to suitably qualified users in the framework.

The administrator is the certificate authority of the organization. The administrator generates the system parameters and issues all the necessary credentials. In summation, the administrator manages the role hierarchy structure of the system. To put a role in the organization’s hierarchy structure, the administrator computes the parameters for that role. These parameters represent the position of the role in the role hierarchy, and are stored in the private cloud. When the role hierarchy changes, the administrator updates these parameters for the roles that have been changed in the private cloud.

Owners who have the data and need to store the encrypted data in the public cloud for different clients to access; Owners can indicate who can get to the data regarding policy based on role. In the RBAC model, they are the entities who deal with the relationship amongst authorizations and roles. A owner can be a client inside the enterprise or an outside, The entity who needs to transmit information to clients in the framework. In this architecture , we consider a owner to be an intelligently isolate segment despite the fact that a client can be a owner and the other way around. Owners just interface with public cloud, and no confidential information are required for these communications. They don't need to hold any parameters in the RBE plan, and they request to get all the required parameters from users in cloud when they perform their encryption operations.

B. System Operations

We describe the framework operations of our proposed design utilizing the strides appeared as a part of figure 1. Simulate the framework utilizes a strong encryption plan Enc to encode messages utilizing the key created as a part of the Encryption algorithm.

Extracting:

This technique is executed by the administrator to include a client or a part in the association. Step1 depicts to the collaboration to create an decryption key for a client, and step 2 depicts to the communication to produce a role confidential key for role. The admin the secret key dk for the client or sk for the role, and sends the key to the client or part by means of a secret channel.

Manage-Role:

The method of making out a role in the hierarchy of role of importance structure is in like manner performed by the admin. Steps 3 depicts to the cooperation's in the administration of a role. The admin characterizes the legacy relationship of the role, and upgrades the position of a character in the role. This is performed in step 3 where the admin processes and transfers the accompanying tuple to the private cloud.

Add Users/ revoke Users:

These two operations are performed by role administrators to the client role membership, and the associations for these operations are depicted. While including or disavowing users, a role manager sends the pair ,IDU is the client personality and τ shows the kind of operation - add or revoke) to the private cloud. In step 5, the private cloud advances this solicitation to public cloud and the general population cloud registers and returns to the private cloud.

Encryption

The role manager performs these steps and upgrade the user role membership, and the associations for these operations are communicated. While adding or revoking clients, a part administrator sends the pair ,IDU and τ demonstrates the kind of operation to the private cloud. In step 5, the private cloud communicates to public cloud and the public cloud computes and returns to the private cloud.

Decryption

At the point when user U needs to regard the information M that has been already encrypted and available in public cloud , the client first request the encrypted content of M from public. Since the role parameters utilized as a part of decryption are forwarded to private cloud, public cloud needs to require these parameters from the private cloud. Proposed Architecture Block Diagram Description as shown in figure 1.

The AES algorithm is used to encrypt or decrypt the file will be implemented on ARM embedded within Raspberry PI. ARM processor encrypts and decrypts the file uploaded by the user. While encrypting the files a special key will be generated. This generated key is linked up with the enciphered file. Whenever user wants to download the content, authorization and authentication is done by front end application. After Successful authentication arm matches the user provided key with the one linked with the enciphered file. When the match is perfect then only the file is decrypted and user can download the file. Encryption, Decryption, Key generation and Key association with file, Key matching will be done in arm processor.

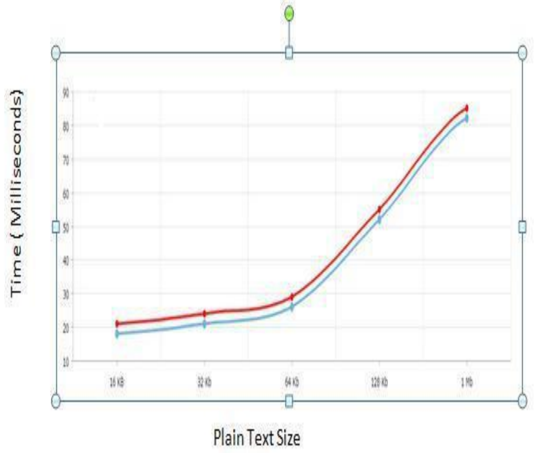

Figure 2: Response for variation in file size

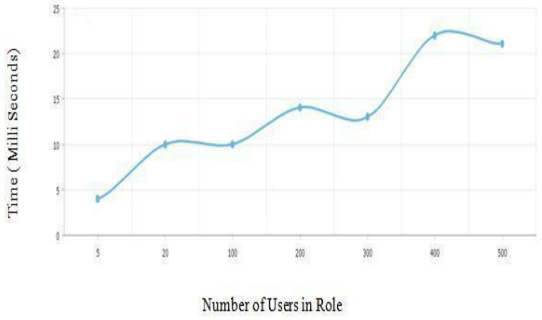

Figure 2: Response for variation in number of users in role

IV. CONCLUSION

Let us first look at the ciphertext size. From the description of the SEBR plan, we discover that the encrypted characters don't contain user related information, we compare encrypted text size when the target role has 10 ancestor roles separately. The figure 2 above demonstrates the ciphertext sizes when the sizes of are 16KB, 32KB and 1MB individually. To start with, we see that the distinctions in size between the plain-text and cipher text are consistent. Besides, the ciphertext size continues as before when the quantity of ancestor role changes. We infer that the ciphertext size is straightly corresponding to the measure of the plaintext. The decoding key size is another vital variable in the cloud system. The portable decryption key can be use by various users from storage services of cloud.

The figure 3 above demonstrates the ideal opportunity for encrypt and decode various files of various sizes on the user side. In this analysis, we created 5 roles, 10 roles and until 500 roles, users in every role which is appeared in figure 3. According to measurements, the encryption time was started from the fourth measurement when a owner chooses files on the Client web application once picks the document to be encrypted, to the time when the upload has been done and the owner gets the cloud's response showing that the successful completion of transaction. The decryption time was measured from the fourth measurement when a client begins getting the cipher text from the cloud till the time the plaintext is saved to a file on the local drive. We have assembled information for the time take for encryption and decryption of the file data that we uploaded and download from enterprise. The figure 2 indicates size of records shifted from 1KB to 1MB.

REFERENCES

[1] Securely Coupled and Extendable Role Based Access Control (SERBAC) in International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 9 (2016) pp 6334-6341

[2] H. R. Hassen, A. Bouabdallh, H. Bettahar, and Y. Chllal, ―Key management for content ‖ Access control in hierarchies,‖

Netw. vol. 51, no 11, pp. 3197 – 3219, 2007.

[3] Public Cloudfrrr Y. Zhu, D. Ma, C. Hu, and D. Huang, "How to use attribute-based encryption to implement role-based access control in the cloud," in Proc. Int. Workshop Sec. Cloud Comput., 2013, pp. 33–40.

[4] Comput., Y. Zhu, D. Ma, C. Hu, and D. Huang, "How to use

attribute-based encryption to implement role-based access

control in the cloud," in Proc. Int. Workshop Sec. Cloud

2013, pp. 33–40.I.S.

[5] Jacobs and C.P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[6] K. Elissa, "Title of paper if known,"

R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press. M .J. Atallh, K. B. Frikken, and M.Blanton, ―Dynamic and efficient keymanagment‖For access control in hierarchy,

[7] Conference paper "Efficient implementation of Advanced Encryption Standard (AES) for ARM based platforms" IEEE

arch 2012.17 March 2012.erence paper 17 March 2012.

[8] http://www.avanade.com/Documents/Research%20and%20Insights/Global_Survey_Slide_Graphics_Has_Cloud_Matured.pdf

[9] S. D. C. Di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati,‖Over Encryption: Mangment of access control evolution on outsourced data,‖ in proc. VLDB, Sep. 2007, pp. 123-134

[10] Cryptographic solution to problem of access control in HierarchyTrans,‖ ACM Trans. Comput ol. 1. No. 3, pp. 239-248, 1983