International Journal of Engineering Research and General Science Volume 2, Issue 6, October-November, 2014
ISSN 2091-2730

Analysis of Access Control Techniques: R3 and RBAC

Priyanka Jairath[*], Rajneesh Talwar[#]

M.Tech Student [*](Deptt. Of Computer Science), Principal CGC- COE [#] Punjab Technical University

prpjairath1@gmail.com [*] , rtmtechguidance@gmail.com [#] and +91 9780608772[*]

Abstract— Cloud computing could be a new approach of computing that leverages the economical pooling of on-demand, self managed, virtual infrastructure. Multicloud designs deploying and evolving our application to include new clouds. This paper provides a survey regarding however multi cloud design will scale back the protection risks and by exploitation multiple distinct clouds at an equivalent time helps in disaster recovery, geo-presence, and redundancy. Though respectable progress has been created, a lot of analysis has to be done to deal with the multi-faceted security issues that exist among cloud computing.

Keywords— Cloud Computing, Access Control, R3, RBAC, Algorithm, Comparison.

INTRODUCTION

Cloud computing is that the evolution of associate existing IT infrastructure that has a long-dreamed vision of computing as a utility. The emergence of cloud technologies over last many years had important impacts on several aspects of IT business. In step with the survey conducted regarding cloud computing, most of medium and tiny firms use cloud computing services attributable to varied reasons that embrace reduction of price in infrastructure and quick access to their application. Cloud computing has been represented in terms of its delivery and preparation models. though cloud computing emerges from existing technologies, its computing (delivery and deployment) models and characteristics raise new security challenges attributable to some incompatibility problems with existing security solutions.

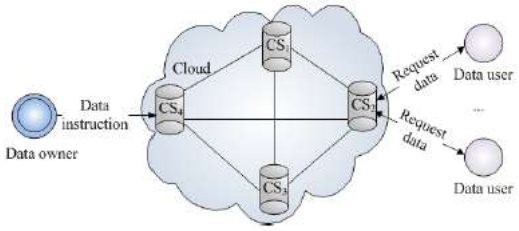

Figure 1: A cloud Environment

A multi-cloud strategy can even improve overall enterprise performance by avoiding "vendor lock-in" and mistreatment completely different infrastructures to fulfil the requirements of various partners and customers. A multi-cloud approach offers not solely the hardware, software system and infrastructure redundancy necessary to optimize fault tolerance, however it can even steer traffic from completely different client bases or partners through the quickest doable components of the network. Some clouds are higher suited than others
for a specific task. For instance, an explicit cloud would possibly handle massive numbers of requests per unit time requiring little knowledge transfers on the typical, however a distinct cloud would possibly perform higher for smaller numbers of requests per unit time involving massive knowledge transfers on the typical.

NIST defines 3 main service models for cloud computing:

1. Software package as a Service (SaaS) – The cloud supplier provides the cloud shopper with the aptitude to deploy associate degree application on a cloud infrastructure [1].
2. Platform as a Service (SaaS) – The cloud supplier provides the cloud shopper with the aptitude to develop and deploy applications on a cloud infrastructure victimization tools, runtimes, and services supported by the CSP [1].
3. Infrastructure as a Service (SaaS) – The cloud supplier provides the cloud shopper with basically a virtual machine. The cloud shopper has the power to provision process, storage, networks, etc., and to deploy and run discretional software package supported by the software system pass by the virtual machine [1].

NIST conjointly defines four readying models for cloud computing: public, private, hybrid, and community clouds. Confer with the agency definition of cloud computing for his or her descriptions [1].

One of the foremost appealing factors of cloud computing is its pay-as-you-go model of computing as a resource. This revolutionary model of computing has allowed businesses and organizations in would like of computing power to buy as several resources as they have while not having to place forth an oversized capital investment within the IT infrastructure. Different blessings of cloud computing are unit large measurability and accrued flexibility for a comparatively constant value. As an example, a cloud user will provision a thousand hours of procedure power on one cloud instance for a similar value as one hour of procedure power on a thousand cloud instances [2].

Despite the various blessings of cloud computing, several massive enterprises area unit hesitant to adopt cloud computing to interchange their existing IT systems. Within the Cloud Computing Services Survey done by IDC IT cluster in 2009, over eighty seven of these surveyed cited security because the best issue preventing adoption of the cloud [3]. For adoption of cloud computing to become a lot of widespread, it's vital that the safety issue with cloud computing be analyzed and addressed, and projected solutions be enforced in existing cloud offerings.

The organization of the remainder of this paper is as follows. The second section discusses the framework with that it will be able to address the safety problems in cloud computing, and therefore the third section elaborates on every of the sections in my framework. Finally, the fourth section of this paper discusses conclusions and future work to be worn out the realm of cloud computing security.

Role based mostly Access management

In role-based access management (RBAC) model, roles are mapped to access permissions and users are mapped to applicable roles. As an example, users are assigned membership to the roles supported their responsibilities and qualifications within the organisation. Permissions are assigned to qualified roles rather than **individual** users. Moreover, in RBAC, a job will inherit permissions from alternative roles; therefore there's a data structure of roles. In RBE theme, the owner of the info encrypts the info in such how that solely the users with applicable roles as fixed by a RBAC policy will rewrite and look at the info. The role grants permissions to users United Nations agency qualify the role and might conjointly revoke the permissions from existing users of the role. The cloud supplier (who stores knowledge/ the info/the information}) won't be able to see the content of the data if the provider isn't given the suitable role. RBE theme is in a position to upset role hierarchies, whereby roles inherit permissions type alternative roles. A user is in a position to affix a job once the owner has encrypted the info for that role. The user will be able to access that information from then on, and therefore the owner ought not to re-encrypt the info. A user is revoked at any time during which case; the revoked user won't have access to any future encrypted information for this role. With our new RBE theme, revocation of a user from a job doesn't have an effect on alternative users and roles within the system.

Security Analysis: We've got shown that our theme is semantically secure beneath the overall Decisional Diffie- dramatist Exponent assumption (GDDHE) introduced in [15] by process a particular GDDHE drawback.

I. BASIC R3

In the basic R3 theme, we have a tendency to think about ideal conditions, wherever the information owner and every one of the cloud servers within the cloud share a synchronised clock, and there aren't any transmissions and queuing delays once corporal punishment scan and write commands.

A. Intuition

The data owner can initial generate a shared secret key to the CSP. Then, when the information owner encrypts every file with the acceptable attribute structure and time slice, the information owner uploads the come in the cloud. The CSP can replicate the file to numerous cloud servers. Every cloud server can have a replica of the shared secret key.

Let us assume that a cloud server stores Associate in Nursing encrypted file F with A and TSi. Once a user queries that cloud server, the cloud server initial uses its own clock to work out this time slice. Presumptuous that this time slice is TSi+k, the cloud server can mechanically re-encrypt F with TSi+k, without receiving any command from the information owner. Throughout the method, the cloud server cannot gain the contents of the cipertext and also the new cryptography keys. Solely users with keys satisfying A and TSi+k are going to be ready to rewrite F.

B. Protocol Description

We divide the outline of the fundamental R3 theme into 3 components: information owner formatting, information user scan information and Data owner write information. We'll deem the subsequent functions.

1) Setup() → (PK;MK; s) : At TS0, the information owner publishes the system public key PK, keeps the system

Algorithm 1: Basic R3 (synchronized clock with no delays)

while Receive a write command W (F; seqnum) at TSi

do
Commit the write command so as at the tip of TSi
while Receive a scan command R(F) at TSi do
Re-encrypt file with TSi
master key MK secret, and sends the shared secret key s to the cloud.[3]

2) GenKey(PK;MK; s; PKAlice;A; T) → (SKAlice;) : Once {the information/the info/the information} owner desires to grant data user Alice attributes A with valid fundamental measure T , the data owner generates SKAlice and mistreatment the system public key, the system passkey, the shared secret key, Alice's public key, Alice's attributes and eligible time.

3) Encrypt(PK;A; s; TSt; F) → (CtA) : At TSt, the information owner encrypts file F with access structure A, and produces ciphertext CtA mistreatment the system public key, access structure, the system secret key, time slice, and plaintext file.

4) Decrypt(PK;CtA ; SKAlice; 1_j_ni) → F : At TSt, user U, United Nations agency possesses version t attribute secret keys on all attributes in CCi, recovers F mistreatment the system public key, the user identity secret key, and also the user attribute secret keys.

5) REncrypt(CtA ; s; TSt+k) → Ct+k
A : once the cloud server needs to come back back a data user with the file at TSt+k, it updates the ciphertext from CtA to Ct+kA victimisation the shared secret key.

1) Information owner initialization: the info owner runs the Setup operate to initiate the system. Once the info owner needs to upload file F to the cloud server, it initial defines associate degree access management A for F, and so determines this time slice TSi. Finally, it runs the write operate with A and TSi to output the ciphertext. once {the information|the info|the information} owner needs to grant a collection of attributes in a very amount of your time to data user Alice, it runs the GenKey operate with attributes and effective times to come up with keys for Alice.

2) Information user scan information: once data user Alice needs to access file F at TSi, she sends a scan command R(F) to the cloud server, wherever F is that the file name. On receiving the scan command R(F), the cloud server runs the REncrypt operate to re-encrypt the file with TSi. On receiving the ciphertext, Alice runs the decode operate victimisation keys satisfying A and TSi to recover F.

3) Information owner write data: once the info owner needs to put in writing file F at TSi, it'll send a write command to the cloud server within the kind of: W (F; seqnum), wherever seqnum is that the order of the write command. This seqnum is critical for ordering once the info owner problems multiple write commands that got to happen in just once slice. On receiving the write command, the cloud server can commit it at the top of TSi. Formula one shows the actions of the cloud server.

Algorithm 2: Extended R3 (asynchronized clock with delays)

while Receive a write command W(F; ti+1; seqnum) do
if Current time is before ti+1 + nine then
Build Window i for file F
Commit the write command in Window i at ti+1 + nine
else
Reject the write command
Inform the info owner to send write command earlier
while Receive a scan request R(F; TSi) do
if Current time is later than ti+1 + prosecutor then
Re-encrypt the move into Window i with TSi
else
Hold on the scan command till ti+1 + prosecutor.[3]

II. ROLE BASED ACCESS CONTROL

In role-based access control (RBAC) model, roles are mapped to access permissions and users are mapped to appropriate roles. For instance, users are assigned membership to the roles based on their responsibilities and qualifications in the organization. Permissions are assigned to qualified roles instead of individual users. Moreover, in RBAC, a role can inherit permissions from other

roles; hence there is a hierarchical structure of roles. Since being first formalized in 1990's, RBAC has been widely used in many systems to provide users with flexible access control management, as it allows access control to be managed at a level that corresponds closely to the organization's policy and structure.

In traditional access control systems, enforcement is carried out by trusted parties which are usually the service providers. In a public cloud, as data can be stored in distributed data centres, there may not be a single central authority which controls all the data centres. 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 data in such a way that only users who are allowed to access the data as specified by the access policies will be able to do so. We refer to this approach as a policy based encrypted data access. The authorized users who satisfy the access policies will be able to decrypt the data using their private key, and no one else will be able to reveal 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.

The main review contributions of this paper are

(i) A new role-based encryption (RBE) scheme with efficient user revocation that combines RBAC policies with encryption to secure large scale data storage in a public cloud,

(ii) A secure RBAC based hybrid cloud storage architecture which allows an organization to store data securely in a public cloud, while maintaining the sensitive information related to the organization's structure in a private cloud,

(iii) A practical implementation of the proposed RBE scheme and description of its architecture and

(iv) Analysis of results demonstrating efficient performance characteristics such as efficient encryption and decryption operations on the client side as well as superior characteristics of the proposed RBE scheme such as constant size cipher text and decryption key as well as efficient user revocation. Given these characteristics, the proposed RBE system has the potential to be a suitable candidate for developing practical commercial cloud data storage systems.

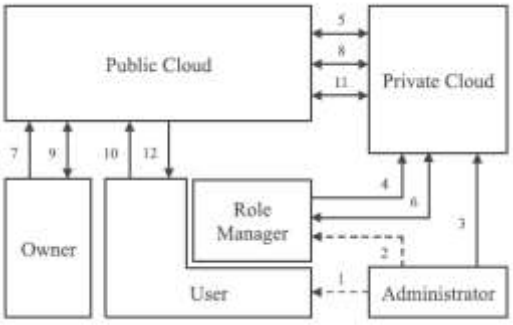

Figure 2: RBE Architecture

In the RBE scheme has the following four types of entities. SA is a system administrator that has the authority to generate the keys for users and roles, and to define the role hierarchy. RM is a role manager2 who manages the user membership of a role. Owners are the parties who want to store their data securely in the cloud. Users are the parties who want to access and decrypt the stored data in the cloud. Cloud is the place where data is stored and it provides interfaces so all the other entities can interact with it.

We can define the following algorithms for RBE scheme:

Setup (λ) takes as input the security parameter λ and outputs a master secret key mk and a system public key pk. mk is kept secret by the SA while pk is made public to all users of the system. **Extract (**mk**,** ID**)** is executed by the SA to generate the key associated with the identity ID. If ID is the identity of a user, the generated key is returned to the user as the decryption key. If ID is the identity of a role, the generated key is returned to the RM as the secret key of the role, and an empty user list *RUL* which will list all the users who are the members of that role is also returned to the RM.

ManageRole (mk**,** ID_R**,** PR_R**)** is executed by the SA to manage a role with the identity ID_R in the role hierarchy.
PR_R is the set of roles which will be the ancestor roles of the role. This operation publishes a set of public parameters pub_R to cloud.

AddUser (pk**,** sk$_R$**,** *RUL*$_R$**,** ID$_U$ **)** is executed by the role manager RM of a role *R* to grant the role membership to the user ID$_U$, which results in the role public parameters pub$_R$ and role user list *RUL*$_R$, being updated in cloud.

RevokeUser (pk, sk$_R$**,** *RUL*$_R$**,** ID$_U$ **)** is executed by a role manager RM of a role *R* to revoke the role membership from a user ID$_U$, which also results in the role public parameters pub$_R$ and role user list *RUL*$_R$, being updated in cloud.

Encrypt (pk**,** pub$_R$**)** is executed by the owner of a message *M*. This algorithm takes as input the system public key pk, the role public parameters pub $_R$, and outputs a tuple_*C, K*_, where *C* will b a part of the ciphertext, and $K \in K$ is the key that will be used to encrypt the message *M*.(Note the ciphertext consists of C and the encrypted M).

We assume that the system uses a secure encryption scheme *Enc*, which takes *K* as the key space, to encrypt messages. The ciphertext of the message *M* will be in the form of _*C, Enc*$_K$*(M)*_ which can only be decrypted by the users who are the members of the role *R*. When this operation finishes, a ciphertext is output and uploaded to cloud by the owner.

Decrypt (pk**,** pub$_R$**,** dk**,** *C***)** is executed by a user who is a member of the role *R*. This algorithm takes as input the system public key pk, the role public parameters pub$_R$, the user decryption key dk, the part *C* from the ciphertext downloaded from cloud, and outputs the message encryption key $K \in K$. The key *K* can then be used to decrypt the ciphertext part *Enc*$_K$*(M)* and obtain the message *M* [4].

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of CGC Landran College, for partial work reported in the paper.

CONCLUSION

In this paper, we analyzed and study the R3 scheme, a new method for managing access control based on the cloud server's internal clock. Our technique does not rely on the cloud to reliably propagate re-encryption commands to all servers to ensure access control correctness. Secondly, we studied a new RBAC scheme that achieves efficient user revocation and have more privacy and security of Information stored in cloud. But all data remains in public cloud.

III. FUTURE SCOPE

We review the two techniques namely R3 and RBAC of cloud based architecture. The future holds of this cloud storage architecture which allows an organization to store data securely in a public cloud, while maintaining the sensitive information related to the organization's structure in a private cloud. In future, we can construct an effective data access control scheme for multi-authority cloud storage systems. Where we can prove that existing scheme can be made more secure in the random oracle model. The new scheme should be a promising technique, which can be applied in any remote storage systems and online social networks etc.

REFERENCES

[1] P. Samarati and S. D. C. di Vimercati, "Data protection in outsourcing scenarios: Issues and directions," in *Proc. ASIACCS*, Apr. 2010, pp. 1–14.

[2] Lan Zhou, Vijay Varadharajan, and Michael Hitchens, " Achieving Secure Role-Based Access Control on Encrypted Data in Cloud Storage", IEEE transactions on information forensics and security, vol. 8, no. 12, December 2013,pp. 1947-1960.

[3] Qin Liu*yz*, Chiu C. Tanz, Jie Wuz, and Guojun Wang*y*, "Reliable Re-encryption in Unreliable Clouds", conference/journal, pp. 1-5.

[4] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in *ASIACRYPT* (Lecture Notes in Computer Science), vol. 4833. New York, NY, USA: Springer-Verlag, 2007, pp. 200–215