International Journal of Contemporary Research in Computer Science and Technology (IJCRCST) e-ISSN: 2395-5325
Volume3, Issue 7 (July '2017)

A CRYPTOGRAPHIC APPROACH USING TAAC SCHEME FOR CONTENT SHARING IN CLOUD ENVIRONMENT

G. Nithya,
Research Scholar,
Department of Computer Science,
Rajah Serfoji Govermment College,
Thanjavur,Tamilnadu,India.

Dr. K. Mohan Kumar,
Research Guide & Head,
Department of Computer Science,
Rajah Serfoji Govermment College,
Thanjavur,Tamilnadu,India.

Abstract: With the ever-increasing demands on multimedia applications, cloud computing, due to its economical but powerful resources, is becoming a natural platform to process, store, and share multimedia contents. However, the employment of cloud computing also brings new security and privacy issues as few public cloud servers can be fully trusted by users. In this paper, they focus on how to securely share video contents to a certain group of people during a particular time period in cloud-based multimedia systems, and propose a cryptographic approach, a provably secure time-domain attribute-based access control (TAAC) scheme, to secure the cloud-based video content sharing. Specifically, they first propose a provably secure time-domain attribute-based encryption scheme by embedding the time into both the cipher texts and the keys, such that only users who hold sufficient attributes in a specific time slot can decrypt the video contents. They also propose an efficient attribute updating method to achieve the dynamic change of users' attributes, including granting new attributes, revoking previous attributes, and regranting previously revoked attributes. They further discuss on how to control those video contents that can be commonly accessed in multiple time slots and how to make special queries on video contents generated in previous time slots. The security analysis and performance evaluation show that TAAC is provably secure in generic group model and efficient in practice.

Keywords: Cloud Computing, Security, Access Control, Cryptographic

I.INTRODUCTION

Research in cloud computing is receiving a lot of attention from both academic and industrial worlds. In cloud computing, users can outsource their computation and storage to servers (also called clouds) using Internet.. Security and privacy are thus very important issues in cloud computing. The cloud can hold the user accountable for the data it outsources, and likewise, the cloud itself accountable for the services it provides. Apart from the technical solutions to ensure security and privacy, there is also a need for law enforcement [1].

As such, it is a crucial demand of customers to have a strong evidence that the cloud servers still possess their data and it is not being tampered with or partially deleted over time.PDP is a technique for validating data integrity over remote servers. should be guaranteed. One dangerous attack is rootkit[2], which is a kind of malware, can enable further privileged access to a computer. Meanwhile, it could hide the existence of itself and other malicious processes by making some changes and modifications to the system. Therefore, detection of rootkit is not as easy as that of other malwares. The kernel rootkits modify the constant resources or dynamic resources of kernel, which compromises the integrity of the OS kernel [3].

Although many researches aim at detecting kernel rootkits, detecting rootkits in a cloud environment should better be transparent to cloud users, without modification to applications and guest OSs. Meanwhile, enforced security methods usually inevitably introduce additional overhead, resulting in performance degradation. This issue becomes particularly prominent on the low-level security when they build a cloud infrastructure. It is still an open issue to improve the security of clouds while keeping the performance, and make a tradeoff between efficiency and security [4].

In a virtualized system, a hypervisor, software layer creating and managing virtual machines (VMs), is responsible for isolating any illegal accesses across virtual machine boundaries. However, there have been concerns over vulnerabilities in hypervisors. Although several recent studies improve the security of hypervisors, the hypervisors, which have been increasing their code sizes for better performance and more features, have become too complex to simply verify their secure execution[5]. If hypervisors cannot be trusted, even a trustworthy cloud provider cannot guarantee the protection of a virtual machine from a malicious co-tenant. To support the secure execution of guest virtual machines, and to guarantee the privacy of user information, memory protection across virtual machines is a critical component. A hypervisor maps the memory of guest Vms to the real memory with guest-physical to machine address translation. Although several different techniques are used to support efficient address translation to the machine address space, all these techniques rely on hypervisors to prevent any illegal memory access across virtual machines. A successful attack on the hypervisor can completely expose the memory contents of guest virtual machines [6].

A hypervisor determines a set of memory pages to be allocated for a VM, and maintains a mapping table from guest-physical to machine address (nested page table) for each VM. With decoupling, the role of a hypervisor is limited to the memory resource allocation to utilize the physical memory efficiently. The hardware processor is responsible for updating the page mapping and setting the pointer to the nested page table to schedule a VM on a core.[7] The currently available hardware-assisted virtualization in commercial processors can lead to a significantly improved memory protection under an untrusted hypervisor, as long as the hardware is securely protected in the server room of the cloud provider. Based on the threat model, this paper proposes a practical design for the hardware-based VM isolation, called hardware-assisted secure virtual machine (H-SVM) architecture, minimizing the changes from the currently available architectural supports for virtualization.[8]

II.PROBLEM STATEMENT

In our literature review done the following problems are formulated in the cloud computing, related to access control, file security and privacy preserving [9, 10].

i. Distributed access control of data stored in cloud so that only authorized users with valid attributes can access them.
ii. Evoked users cannot access data after they have been revoked.
iii. The proposed scheme is resilient to replay attacks. A writer whose attributes and keys have been revoked cannot write back stale information.
iv. The protocol supports multiple read and writes on the data stored in the cloud.
The costs are comparable to the existing centralized approaches, and the expensive operations are mostly done by the cloud.

The above problems are solved in our Proposed Methodology

According to existing scheme a user can create a file and store it securely in the cloud. This scheme use the two protocols ABE and ABS. There are multiple KDCs, which can be scattered. A creator on presenting the token to one or more KDCs receives keys for encryption/decryption and signing. The access policy decides who can access the data stored in the cloud. Write proceeds in the same way as file creation. By designating the verification process to the cloud, it relieves the individual users from time consuming verifications.

The following are the disadvantages in the Existing system

- Lack in security.
- De-duplication may occur in data sharing.
- Classical access policy may cause lack in flexibility.
- Need to maintain the access policy by separate admin.
- Need to manage key from the trusted party.
- Malicious admin may take over data.
- Implementation cost is high.

III.PROPOSED METHODOLOGY

The proposed method of **Distributed Encrypted Key Authentication and Access Control (DEKAAC)** will able to solve the above mentioned problems. In this proposed method using self adaptive access control and delegation framework. This will provide access control for the client from viewing others data users attribute key to encrypt and decrypt data. So, the data key is generated for every data the user uploaded to the server. In case of authentication and access control scheme the key for data is stored in one server and the data stored in cloud so any admin can't view clients data. For providing access control and reply attacks the user control mechanism is implemented.

Algorithm for authentication server

The following algorithm is for server Authentication:

procedure AUTHENTICATION
get user_name
get user_password
get app_id
get client_ip
result **select** db_password **for** user_name **and** app_id
if result = 1
then
if db_password = user_password
then
select access_key **for** user_name **and** app_id
return access_key
endif
else
invalidate user_name
endif
else
return false
end procedure

DEKAAC Architecture

The following Figure1 is the Architecture for the proposed methodology

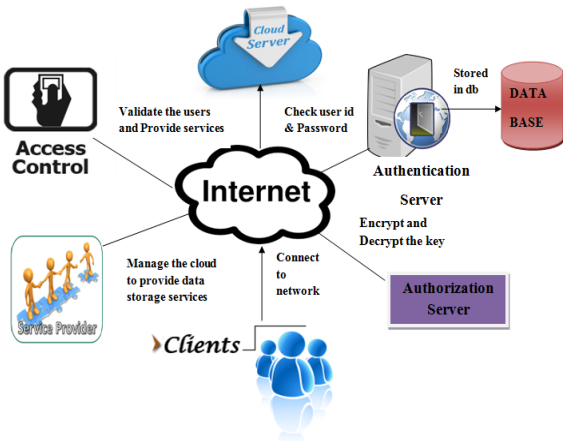

Figure1: DEKAAC Architecture

The Architecture diagram described in the following modules

Modules

The Following Modules are in this new approach:

- Cloud Server.
- Authentication Server.
- Authorization Server.
- Access Controller.
- Service Provider.

IV.RESULT AND DISCUSSION

In this Analysis the following comparisons are made between existing and the New Methodology:

- Cloud Load balancing

- Authentication Time
- Encryption
- Upload performance

Cloud load balancing

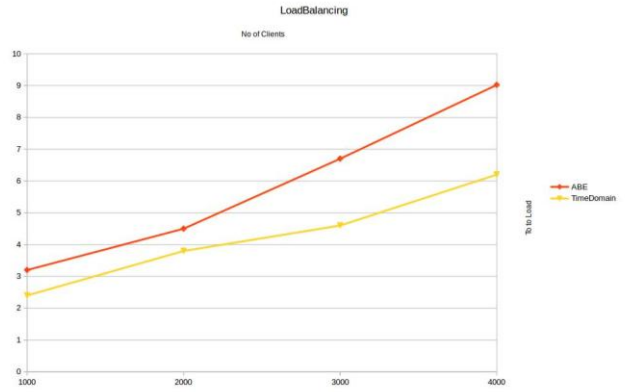

Figure 2: Graph comparison of load balancing

It is a type of load balancing and not to be confused with Domain Name System (DNS) load balancing. While DNS load balancing uses software or hardware to perform the function, cloud load balancing uses services offered by various computer network companies.

Authentication Time

The scheme prevents replay attacks and supports creation, modification, and reading data stored in the cloud. The communication, computation, and storage overheads are comparable to centralized approaches. When the time limit of the file expired, the file will be automatically revoked and cannot be accessible to anyone in future. Manual Revocation also supported. Policy based file renewal is proposed. The Renewal can be done by providing the new key to the existing file, will remains the file until the new time limit reaches.

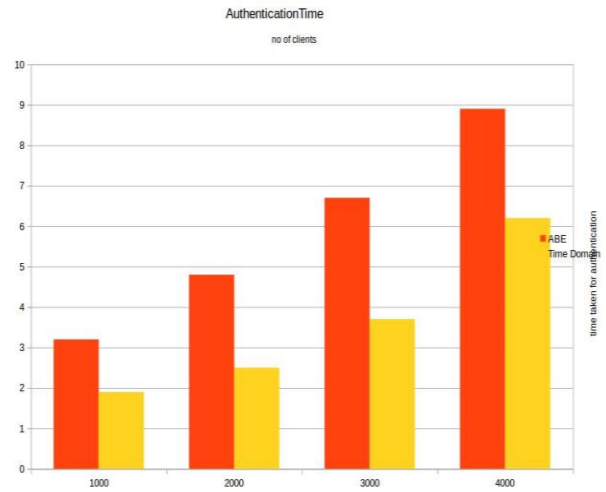

Figure 3: Time taken for authentication

First the client was authenticated with the username and password, which is provided by the user. Then the user was asked to answer two security levels with his/her choice. Each security levels consist of 5 user selectable questions. The user may choose any one question from two security levels. The private key for encrypt the file was generated with the combination of username, password and the answers for the security level questions. After generating the private key the client will request to the key manager for the public key . If the policy matches with the file name then same public key will be generated. Otherwise new public key will be generated. With the public key and private key the file will be encrypted and uploaded into the cloud. If a user wants to download the file he/she would be authenticated. If the authentication succeeded, the file will be downloaded to the user. Still the user cant able to read the file contents.

We should request the public key to the key manager. According to the authentication, the key manager will produce the public key to the user. Then the user may decrypt the file using the login credentials given by the user and the public key provided by the key manager.

IV.CONCLUSION

In this paper, they introduced the (DEKAAC) Distributed Encrypted Key Authentication and Access Control scheme used for the client from viewing others data users attribute key to encrypt and decrypt data. So, the data key is generated for every data the user uploaded to the server. In case of authentication and access control scheme the key for data is stored in one server and the data stored in cloud so any admin can't view clients' data. For providing access control and reply attacks the user control mechanism is implemented. Finally, they implemented the proposed scheme, and conducted comprehensive performance analysis and evaluation, which showed its efficiency and advantages over existing schemes. In future, they will work towards designing a more expressive scheme, which can be proved to have full security under the standard model, with better performance in confidentiality of user access privilege and user secret key accountability can be achieved.

V.FUTURE ENHANCEMENT

By implementing these three algorithm we provide authenticity security and data integrity to the data in file triple encryption .The time complexity is high because it is a one by one process but in future this time complexity could be reduced. In future new selection algorithm.

VI.REFERENCES

[1] Shahzad, Farrukh. "State-of-the-art survey on cloud computing security Challenges, approaches and solutions." Procedia Computer Science 37 (2014): 357-362.

[2] Zhang, Qi, Lu Cheng, and Raouf Boutaba. "Cloud computing: state-of the-art and research challenges." Journal of internet services and applications 1.1 (2010): 7-18.

[3] Takabi, Hassan, James BD Joshi, and Gail-Joon Ahn. "Security and privacy challenges in cloud computing environments." IEEE Security & Privacy 6 (2010): 24-31.

[4] Mell, Peter, and Tim Grance. "The NIST definition of cloud computing." (2011).

[5] Pham, Quan, et al. "On a taxonomy of delegation." computers & security 29.5 (2010): 565-579.
[6] Qureshi, Nauman A., Ivan J. Jureta, and Anna Perini. "Towards a requirements modeling language for self-adaptive systems." Requirements Engineering: Foundation for Software Quality. Springer Berlin Heidelberg, 2012. 263-279.
[7] Qureshi, Nauman A., Ivan J. Jureta, and Anna Perini. "Requirements engineering for self-adaptive systems: Core ontology and problem statement." Advanced Information Systems Engineering. Springer Berlin Heidelberg, 2011.
[8] Brun, Yuriy, et al. "Engineering self-adaptive systems through feedback loops." Software engineering for self-adaptive systems. Springer Berlin Heidelberg, 2009. 48-70.
[9] Younis, Younis A., Kashif Kifayat, and Madjid Merabti. "An access control model for cloud computing." Journal of Information Security and Applications 19.1 (2014): 45-60.
[10] Bussard, Laurent, Anna Nano, and Ulrich Pinsdorf. "Delegation of access rights in multi-domain service compositions." Identity in the Information Society 2.2 (2009): 137-154.