International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 2 Issue: 3 08-Apr-2014,ISSN_NO: 2321-3337

# DATA SECURITY AS A SERVICE IN CLOUD

*Jency Anand*[1],*Prof.V.Vijayaganth*[2],*Dr.J.George Chellin Chandran,*[3]

PG Scholar, Dept of PG CSE, CSI College of Engg, Ketti, India[1].

Asst.Professor, Dept.of PG CSE, CSI College of Engg, Ketti, India[2]

Principal, CSI College of Engg, Ketti, India[3]

***ABSTRACT-*** *CLOUD computing presents a replacement thanks to supplement this consumption and delivery model for IT services supported the net, by providing for dynamically scalable and sometimes virtualized resources as a service over the net. The JAR file includes a group of straightforward access management rules specifying whether or not and the way the cloud servers and probably different knowledge stakeholder's square measure approved to access the content itself. Once the authentication succeeds, the service supplier is allowed to access the info capsulate within the JAR. Reckoning on the configuration settings outlined at the time of creation, the JAR can offer usage management related to work, or can offer solely work practicality. As for the work, every time there\'s associate access to the info, the JAR can mechanically generate a log record. In this paper to provide the data security in cloud and to view the log records .log records are using to find out the unauthorized person.*

**Keywords- Secure production, Jar creation, log Record generation, Push and Pull Mode.**

## 1. INTRODUCTION

Organizations use the Cloud during a sort of completely different service models (SaaS, PaaS, IaaS) and preparation models (Private, Public, and Hybrid). In most cases, the supplier should make sure that their infrastructure is secure which their clients' knowledge and applications square measure protected whereas the client should make sure that the supplier has taken the right security measures to safeguard their data.

For IT services are supported by the internet, by providing for dynamically scalable and sometimes virtualized resources as a service over the internet. Knowledge handling will be outsourced by the direct Cloud Service Provider (CSP) to different entities within the cloud and theses entities can even delegate the tasks to others, and so on. Second, entities are allowed hitching and leaving the cloud in a very versatile manner. As a result, knowledge handling within the cloud goes through a dynamic ranked service chain that doesn't exist in standard environments. The Cloud information irresponsible framework projected during this work conducts machine-controlled work and distributed auditing of relevant access performed by any entity, administered at any purpose of time at any cloud service supplier. It has 2 major components: logger and log harmonizer. The JAR file includes a group of straightforward access management rules specifying whether or not and the way the cloud servers and probably different knowledge stakeholder's square measure approved to access the content itself. Once the authentication succeeds, the service suppliers are allowed to access the info capsulate within the JAR. Reckoning on the configuration settings outlined at the time of creation, the JAR can offer usage management related to work, or can offer solely work practicality. As for the work, every time there is associate access to the info, the JAR can mechanically generate a log record. JAR's which allows it to monitor the loss of any logs

from any of the JARs. Moreover, if a JAR is not able to contact its central point, any access to its enclosed data will be denied.

This paper briefly discusses the application of Cloud Computing as a computing paradigm to Information Support Systems (ISS) and how it can serve as a future technology for such systems. In this section, we have a tendency to first review connected works addressing the privacy and security problems within the cloud. Then, we briefly discuss works that adopt similar techniques as our approach however serves for various functions.

## 2. SYSTEM PROCESS

In a cloud setting, the unit of access management is often a sharable piece of user data—for example, a document in an exceedingly cooperative editor. Ideally, the system offers some analogous confinement of that information, proscribing its visibility solely to licensed users and applications whereas permitting broad latitude for what operations are done thereon. This can make writing secure systems easier for programmers as a result of confinement builds it harder for buggy code to leak information or for compromised code to grant unauthorized access to information. A computer program may realize different (Fig 1) ways in which to exhilarate information, like using an aspect channel or covert channel, however the priority here is to support benign developers, whereas creating all applications and their actions on users' sensitive information a lot of simply auditable to catch improper usage.

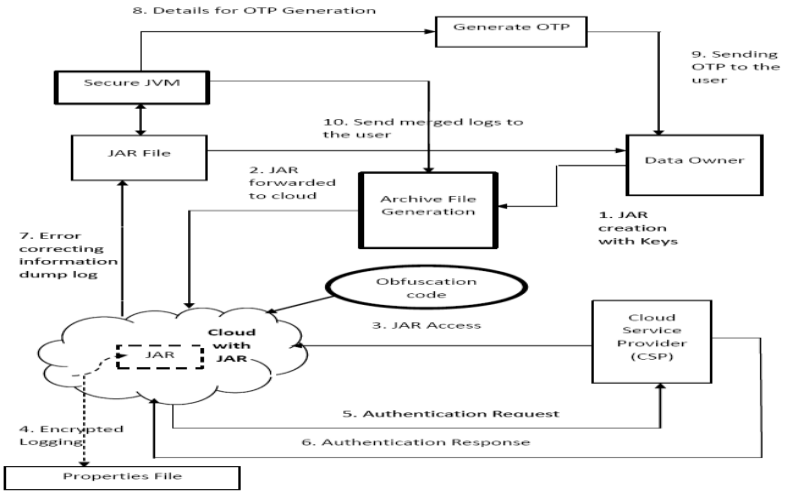


*Fig.1 System Architecture*

Application developers don't have to be compelled to reinvent the wheel application code is freelance of ACL enforcement.

Third-party auditing and standards compliance are easier; and the verifiable platform extends to virtualized environments designed a top it. Finally, the value of examining the platform is amortized across all its users, which suggests vital economies of scale for a large-scale platform supplier.

### 2.1 Cloud computing:

The issue of cloud computing has been obtaining an immense coverage in recent years. Rather than maintaining a technical infrastructure so as to control, calculate or no matter else you are doing along with your information, the log in through the internet and know on the cloud provider's systems instead. There is no single cloud model, then beneath the umbrella, that would possibly an option for a wholesale transfer of services, place one a part of the organization onto the cloud, select one cloud service or perhaps simply however a cloud back-up.

### 2.1 Applications:

The original motivation for identity-based secret writing is to assist the preparation of a public key infrastructure. During this section, we have a tendency to show many alternative unrelated applications.

### 2.3 Keys Generation:

Another application of elliptic curves in cryptography has recently emerged within the type of a replacement system for doing Identity-Based cryptography. Identity-Based cryptography could be a public key cryptography theme wherever any string may be a user's public key, including, for instance, the user's email address or name. The advantage of ID based mostly cryptography is that no certificate is required to bind names to public keys. The sender will use the receiver's ID as its public key, and ought not to get and verify a certificate on the recipient's public key beforehand. Once associate degree encrypted communication has been received, a user will contact a central CA to get the key comparable to its public key.

### 2.4 Bits Comparison:

At the 163-bit ECC/1024-bit RSA security level, associate degree elliptic curve involution for general curves over discretionary prime fields is roughly five to fifteen times as quick as associate degree RSA non-public key operation, betting on the platform and optimizations. At the 256-bit ECC/3072-bit RSA security level the quantitative relation has already increased to between twenty and sixty, betting on optimizations. To secure a 256-bit AES key, ECC-521 will be expected to get on average four hundred times quicker than fifteen, 360-bit RSA.

## 3. PROBLEM STATEMENT

A user signed to a cloud service sometimes must send his/her information further as associated access management policies (if any) to the service supplier. Once the information is received by the cloud service supplier, the service supplier can have granted access rights, like scan, write, and copy, on the information. Victimization standard access management mechanisms, once the access rights are granted, the information is going to be absolutely out there at the service supplier. The work ought to be suburbanized so as to adapt to the dynamic nature of the cloud. A lot of specifically, log files ought to be tightly finite with the corresponding information (Fig 2) being controlled, and need tokenism infrastructural support from any server.

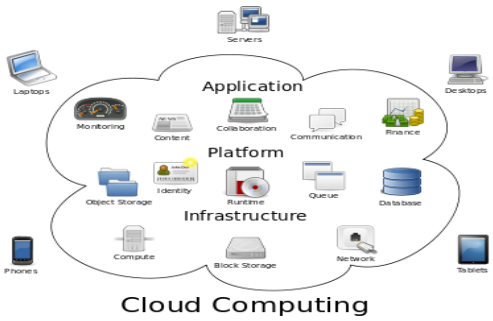


*Fig.2 Cloud computing*

Log files ought to be sent back to their information house owners sporadically to tell them of this usage of their information. A lot of significantly, log files ought to be recoverable anytime by their information house owners once required regardless the situation wherever the files are kept.

## 4. DATA PROTECTION

Currently, users should believe totally on legal agreements and implicit economic and reputational damage as a proxy for application trait. As an alternate, a cloud platform may facilitate bring home the bacon a sturdy technical resolution by creating it simple for developers to write down reparable applications that defend user information within the cloud, thereby providing constant economies of scale for security and privacy as for computation and storage. Sanctionative freelance verification each of the platform's operation and therefore the runtime state of applications on that, therefore users will gain confidence that their information is being handled properly.

Much as associate OS provides isolation between methods however permits substantial freedom within a process, cloud platforms may provide transparently verifiable partitions for applications that cipher on information units, whereas still permitting broad machine latitude among those partitions. Information protections as a service enforces fine-grained access management policies on information units through application confinement and knowledge flow checking. It employs science protections at rest and offers strong work and auditing to produce answerableness.

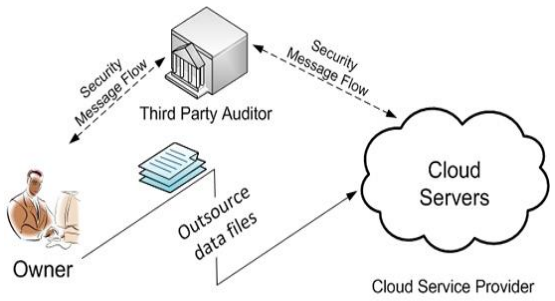


*Fig: 3 Data sharing in cloud*

### 4.1 Models Function:

There are unit 2 major parts of the United States intelligence agency, the primary being the lumberjack, and therefore the second being the log harmonizer. The lumberjack is that the element that is powerfully let alone the user's knowledge, so it's downloaded once the info area unit accessed, and is derived whenever the info area unit derived. It handles a selected instance or copy of the user's knowledge and is chargeable for work access thereto instance or copy. The log harmonizer forms the central element that permits the user access to the log files.

### 4.2 JAR Generation:

The JAR file contains a collection of access management rules specifying whether or not and the way the cloud servers and presumably different information interested party (users, companies) square measure licensed to access the content itself. Looking on the configuration settings outlined at the time of creation, the JAR can give usage management related to work, or can give solely work practicality.

### 4.3 Logger Creation:

We leverage the programmable capability of JARs to conduct machine-driven work. A lumberman element could be a nested Java JAR file that stores a user's knowledge things and corresponding log files. The most responsibility of the outer JAR is to handle authentication of entities that need to access the info hold on within the JAR file. In our context, the info homeowners might not apprehend the precise CSPs that are aiming to handle the info. Hence, authentication is mere in step with the servers', instead of the server's computer address or identity. The information owner will specify the permissions in user-centric terms as critical

the standard code-centric security offered by Java, exploitation Java Authentication and Authorization Services. Moreover, the outer JAR is additionally to blame of choosing the right inner JAR in step with the identity of the entity UN agency requests the info.

**4.4 Push mode:**
In this mode, the logs area unit sporadically pushed to the info owner (or auditor) by the harmonizer. The push action are going to be triggered by either variety of the subsequent 2 events: one is that the time elapses for a particular amount in line with the temporal timer inserted as a part of the JAR file; the opposite is that the JAR file exceeds the scale stipulated by the content owner at the time of creation. Once the logs area unit sent to the info owner, the log files are going to be drop, alongside the log files, the error correcting info for those logs is additionally dropped. This push mode is that the basic mode which might be adopted by each the Pure Log and therefore the Access Log, notwithstanding whether or not there's an invitation from the info owner for the log files. This mode serves 2 essential functions within the work design, it ensures that the scale of the log files doesn't explode and it permits timely detection and correction of any loss or injury to the log files.

**4.5 Pull mode:**
This mode permits auditors to retrieve the logs. Any time once they need to visualize the recent access to their own knowledge. The pull message consists merely of associate degree FTP pull command, which might be problems from the instruction. For naive users, a wizard comprising a batch file will be simply engineered. The request are going to be sent to the harmonizer, and also the user are going to be abreast of the data's locations and acquire associate degree integrated copy of the authentic and sealed log file.

**4.6 Cloud Data Attacks:**
Because user registrations should be etch to forestall unauthorized users from directive calls to themselves or elsewhere, our system uses digest authentication. This suggests that the system can always verify a shared secret between the server and therefore the shopper via challenge-response before permitting access. Our analysis relies on a semi honest oppose model by assumptive that a user doesn't unleash his master keys to unauthorized parties, whereas the aggressor might try and learn further data from the log files.

**4.6.1 Copying Attack:** The foremost intuitive attack is that the aggressor copies entire JAR files. The aggressor might assume that doing therefore permits accessing the info within the JAR file while not being noticed by the info owner.
**4.6.2 Disassembling Attack:** Another potential attack is to break up the JAR file of the lumberman then arrange to extract helpful data out of it or spoil the log records in it.
**4.6.3 Man-in-the-Middle Attack:** An aggressor might intercept messages throughout the authentication of a service supplier with the certificate authority, and reply the messages so as to masquerade as a legitimate service supplier.
**4.6.4 Compromised JVM Attack:** An aggressor might try and compromise the JVM. To quickly find and proper these problems, a way to integrate oblivious hashing to ensure the correctness of the JRE and the way to correct the JRE before execution, just in case any error is detected.

**5. ALGORITHMS**
The maximum size at that logs square measure pushed out could be a parameter which may be simply designed whereas making the lumberman part. The pull strategy is most required

once the information owner suspects some misuse of his data; the pull mode permits him to observe the usage of his content forthwith.

**5.1 The AES Algorithm:**

AES may be a block cipher that encrypts a 128-bit block (plaintext) to a 128-bit block (cipher text), or decrypts a 128-bit block (cipher text) to a 128-bit block (plaintext). AES uses a key (cipher key) whose length will be 128, 192, or 256 bits. Hereafter encryption/decryption with a cipher key of 128, 192, or 256 bits is denoted AES- 128, AES192, AES-256, severally.

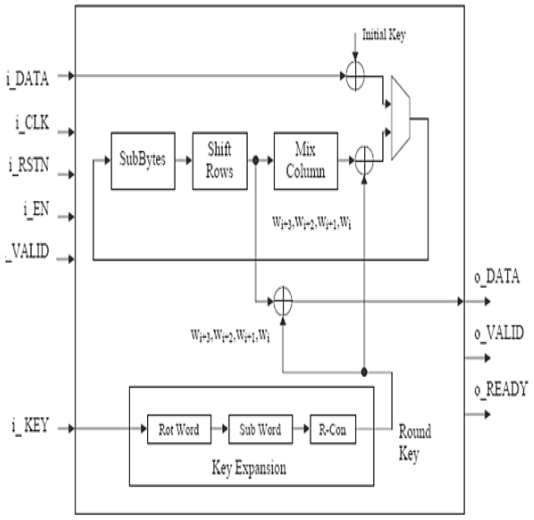


*Fig.4 AES encryption block diagram*

**6. RELATED WORK**

**6.1 Data Accountability:**

Data Accountability knowledge answerability describes the authorization necessities for one knowledge usage policy. We tend to introduce weak knowledge answerability, that describes that a given usage policy might are obtained properly. Weak knowledge answerability expresses that associate degree agent should give an authorization proof which all delegated responsibilities should even be accounted for, i.e. for any received policies accustomed derive the policy, there's knowledge answerability for causing of that policy at the causing agent. The key distinction in our implementations is that the authors still admit centralized information to take care of the access records, whereas the things being protected are command as separate files.

**6.2 Identity-Based Encryption:**

In a very typical setting, associate degree IBE theme involves a sure third party, the Personal Key Generator (PKG). The PKG generates the theme public parameters and a master personal key. For the asking of users, the PKG derives from the key the personal coding key associated to a public identity by running associate degree extraction formula. A lot of formally, associate degree IBE theme is outlined as follows.

**6.2.1 Identity-Based coding scheme**: Associate degree identity-based encryption theme is specific by a quadruple of algorithms$(Setup, EX, E, D)$.

**Setup:** Given a security parameter of the *Setup*formula generates the general public parameters of the theme and a master personal key;
**Extract:** Given a key *mk* and a public identity *id*€{0, 1}*, *EX(mk, id)* computes the corresponding coding key *sk*;
**Encrypt:** Given a public identity *id* and a message *m*, *E(id,m)* computes a ciphertext *c* cherish the coding of *m* below *id*;
**Decrypt:** Given a personal coding key *sk* and ciphertext *c*, *D(sk, c)* returns either the plaintext cherish the coding of c, if it's a legitimate ciphertext, or a distinguished worth otherwise.
In case of Access Log, the on top of formula is changed by adding a further check once step half dozen. Precisely, the Access Log checks whether or not the CSP accessing the log satisfies all the conditions per the policies bearing on it. If the conditions are satisfied, access is granted; otherwise, access is denied. No matter the access management outcome, the tried accesses to the info within the JAR file are logged. Our auditing mechanism has 2 main sanctifications. First, it guarantees a high level of accessibility of the logs. Second, the employment of the harmonizer minimizes the quantity of employment for human users in browsing long log files sent by totally different copies of JAR files. For a more robust understanding of the auditing mechanism, we tend to gift the subsequent example.

## VIII. CONCLUSION AND FUTURE WORK

Cloud computing permits extremely scalable services to be simply consumed over the internet on associate degree as-needed basis. Entrusting log management to the cloud appears to be a sustainable cost saving measure for securely upholding the log records over comprehensive period of time. Thus, the reliability of the log files and that of the logging progression is safeguarded at all times. In addition, the confidentiality and privacy of log records are secured. In this paper,
The direction of future work focused on pull mode in which the user may misuse the data after being retrieved like illegal distribution. These difficulties can be overcome by some ways of embedding data with some copyright information or serial numbers.

## 9. REFERENCES

[1]. P. Ammann and S. Jajodia, “Distributed Timestamp Generation in Planar Lattice Networks,” ACM Trans. Computer Systems, vol. 11, pp. 205-225, Aug. 1993.
[2]. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable Data Possession at Untrusted Stores,” Proc. ACM Conf. Computer and Comm. Security, pp. 598- 609, 2007.
[3]. E. Barka and A. Lakas, “Integrating Usage Control with SIP-Based Communications,” J. Computer Systems, Networks, and Comm.,vol. 2008, pp. 1-8, 2008.
[4]. D. Boneh and M.K. Franklin, “Identity-Based Encryption from the Weil Pairing,” Proc. Int’l Cryptology Conf. Advances in Cryptology, pp. 213-229, 2001.
[5]. R. Bose and J. Frew, “Lineage Retrieval for Scientific Data Processing: A Survey,” ACM Computing Surveys, vol. 37, pp. 1- 28, Mar. 2005.
[6]. M. Bellare and B. S. Yee, “Forward integrity for secure audit logs,” Dept. Comput. Sci., Univ. California, San Diego, Tech. Rep., Nov. 1997.
[7]. BalaBit IT Security (2011, Sep.).*Syslog-ng—Multiplatform Syslog Server and Logging Daemon* [Online]. Available: http://www.balabit.com/network-security/syslog-ng
[8]. J. Kelsey, J. Callas, and A. Clemm, *Signed Syslog Messages*, Request for Comment RFC 5848, Internet Engineering Task Force, Network Working Group, May 2010.

[9]. D. Ma and G. Tsudik, "A new approach to secure logging," *ACM Trans.Storage*, vol. 5, no. 1, pp. 2:1–2:21, Mar. 2009.
[10]. U. Flegel, "Pseudonymizingunix log file," in *Proc. Int. Conf. Infrastruture Security*, LNCS 2437. Oct. 2002, pp. 162–179.
[11]. C. Eckert and A. Pircher, "Internet anonymity: Problems and solutions," in*Proc. 16th IFIP TC-11 Int. Conf. Inform. Security*, 2001, pp. 35–50 .
[12]. M. Rose, *The Blocks Extensible Exchange Protocol Core*, Request for Comment RFC 3080, Internet Engineering Task Force, Network Working Group, Mar. 2001.
[13]. B. Schneier and J. Kelsey, "Security audit logs to support computer forensics," *ACM Trans. Inform. Syst. Security*, vol. 2, no. 2, pp. 159–176, May 1999.
[14]. J. E. Holt, "Logcrypt: Forward security and public verification for secure audit logs," in *Proc. 4th Australasian Inform. Security Workshop*, 2006,pp. 203–211.
[15]. R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second generation onion router," in *Proc. 12th Ann. USENIX Security Symp.*,Aug. 2004, pp. 21–21.
[16]. The Tor Project, Inc. (2011, Sep.) *Tor: Anonymity Online* [Online]. Available: http://www.torproject.org
[17]. D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inform. Theory*, vol. 29, no. 2, pp. 198–208, Mar. 1983.
[18]. A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11,pp. 612–613, Nov. 1979.
[19]. G. R. Blakley, "Safeguarding cryptographic keys," in *Proc. Nat. Comput.Conf.*, Jun. 1979, p. 313.