ISSN: 0975-766X
CODEN: IJPTFI
Research Article

Available Online through
www.ijptonline.com

DEBTDS - A DOUBLE ENCRYPTION BASED ON TOKENS FOR DATA SECURITY IN CLOUD COMPUTING

[1]Rashmi Ranganathan, [2]Kumar Ashwin Hubert, [3]R. Manjula
[1,2,3]VIT University, Vellore.
Email: rashmi05rash@gmail.com

Received on: 18.10.2016 *Accepted on: 11.11.2016*

Abstract

Many organizations are beginning to be increasingly dependent on the cloud for their data operations and management. Cloud services are utilized for both data storage and processing. As more organizations move their business to the cloud, ensuring security of data during transfer becomes more important. A large data space and a multitude of services provided by the cloud ensures consistency across the platform. Increased security and reliability in transmission will increase client confidence and adoption of cloud services. In this article, a double encryption method based on tokens is proposed to strengthen security in cloud data transfer between Cloud Client (CC) and Cloud Service Provider (CSP).

Keywords: Cloud Service; Encryption; Security algorithm; SHA algorithm.

1. Introduction

Massive amount of data is often used by large Organizations and Enterprises to run their business. Cloud Computing offers a platform and infrastructure to perform complex services for the customers depending on their application. Cloud services not only provides Space for data storage but also provides software, infrastructure, virtual hardware and related services. The Characteristics of Cloud Service Provider (CSP) include Reliability, instant service with 99.5% uptime, Cost Efficient, and Bandwidth Allocation limit. The top 10 Cloud Computing Companies according to businessinsider.com include Amazon (AWS) Rackspace, Microsoft Azure, Google, RedHat, Citrix, Salesforce, Linode, VMware, and Verizon. The comparison of various CSP is presented in [1]. Figure 1 represents the cloud usage by public as of 2015. The services offered by CSP are categorized into three storeys in a Stack as Software-as-a-Service (SaaS), Platform-as-a-service (PaaS) and Infrastructure-as-a-service (IaaS) from tip to underside. SaaS are intended for end-client to deliver service over the web. PaaS is a set of tools intended to make computation and deployment of applications faster and effectively. IaaS is the Hardware and software that controls storage, network, server and OS [https://blog.rackspace.com/top-10-common-uses-for-the-cloud-for-2012].

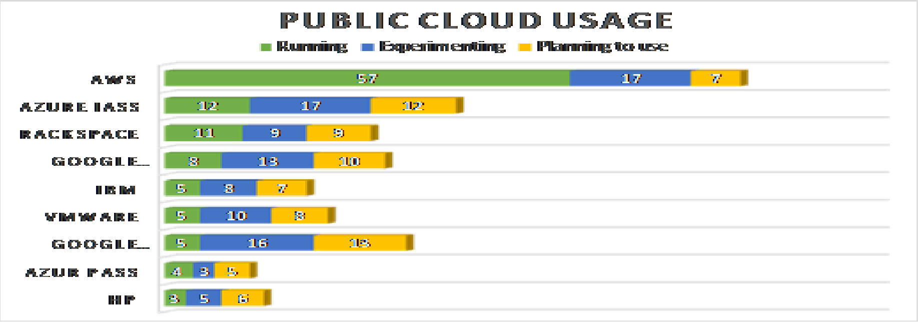

Figure 1: Public Cloud Usage.

Table 1: Cloud Service Models.

CATEGORIES	USERS	SERVICE PROVIDER
SaaS	Client (End-User)	Applications
PaaS	Application Owner	Application Code
IaaS	Middleware and Hardware support	Cloud Storage

Even though CSP offers a cost-efficiency and flexibility to Clients, it has numerous security risk to differentiate the data of one client from others to achieve privacy and integrity demands. In the paper [2], a literature review was conducted on methods taken in cloud computing to achieve efficiency. It has been concluded that 45% of the techniques are based on Encryption out of which 71% of them are accurate and 67% of them validates experimentally. This proves that researchers and students are interested in the subdivision of security risk under cloud computing. The top challenges of implementing a successful cloud comprise Security (64%), followed by Administration/Support and Control/Vendor Lock-In at 58% and 40% respectively (not exact). Additional obstacles include Performance, Provider Reputations, Consumption Pricing, Speed to Activate, Portability. The above statistics are shown in Figure 2. The Identical theory is supported by [3] which states that “security is considered to be a critical barrier for cloud computing “[4].

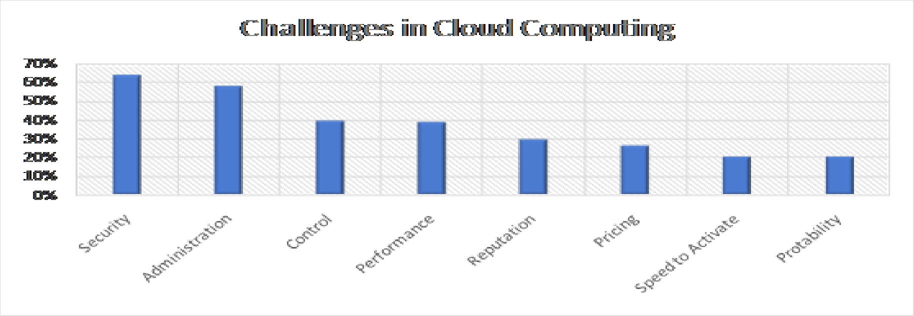

Figure 2: Challenges in Cloud Computing.

The prominent security risk happens malicious code/data send to the cloud ruins the existing data in the cloud which exploits data or calculation and steal data or forcing to result in false outcomes; personnel of the cloud service provider could leak data by misusing their capabilities; and data leakage or manipulated computation could occur due to the vulnerabilities in the shared resources [5].

Confidentiality of data, Integrity (data untampered) and availability (Server uptime) are the significant requirements of cloud clients [6]. Various cryptographic methods are proposed which can perform computations on encrypted data [7], or secure and verify data stored in cloud [8].

Similarly, subjective computation on personal information can be accomplished with entirely homomorphic encryption [9], in collaboration with garbled circuits [10] for verifiability [11].

2. Related Works

2.1 Issues related to Cloud Data Security

- Data Confidentiality: Data Confidentiality is all about ensuring that information is not disclosed or provided to unauthorized person or parties in any manner.
- Data Integrity: Data Integrity ensures that information stored in the cloud is a proper representation of the information intended and the data has not been modified in any way by an unauthorized person.
- Data Availability: Data availability refers to ensuring that information or data is available to authorized parties or persons when needed.
- Data Theft: Data can be stolen from the cloud server by a malicious or unauthorized user.
- Data Location: User’s data is hosted in some country and user probably doesn’t know exactly where the data is. The privacy rules and regulation of that country may pose a security risk.
- Data Security on Cloud Vendor Level: Cloud Provider or Vendor has to make sure that the server will be secure from all external threats and attacks. A cloud is considered as good only when a good security is provided by the vendor to its customers.
- Data Security on Cloud Client Level: Although the cloud service provider has improved it security algorithm to give a good security protection to the client, it is the responsibility of cloud client to protect his/her data and make sure that its own action shouldn’t lead to any loss of data or tampering of data for client who are using the identical cloud.

2.2 Literature work on Security Algorithms

Security Algorithms are developed to minimize the risk in cloud computing. Algorithms such are RSA, MD5 and SHA are used worldwide to encrypt and decrypt data for authentication of the user as discussed in [12, 13]. In paper [14] the authors have used RSA Algorithm in data security which captures the intruder from getting the original data from CSP. But these algorithms are time consuming as the entire data is encrypted. The following are existing

- A twin cloud model has proposed the idea of creating an intermediate cloud (Trusted Cloud) which encrypts the data which needs to be stored and later sends to the untrusted public cloud. The cloud user will communicate with a server cloud through the Trusted Cloud (which could be either a private cloud or cloud built from many secure hardware components). [15]
- Cross-platform integration model uses a security key as well as a secure communication over the internet. Sha hash algorithm has been used for Data encryption in the above model. [16].
- Ruj et al [17] have proposed a model where the cloud stores encrypted data and uses Distributed Access Control in Clouds (DACC) algorithm by employing attribute-based encryption and the keys being distributed by KDCs (Key Distribution Centres).
- Hierarchical identity-based encryption (HIBE) is considered to be flexible, scalable and has fine-grained access control when compared to other encryption schemes like Attribute Based Encryption (ABE), Key-Policy Attribute Based Encryption (KP-ABE), Cipher-Policy ABE (CP-ABE) [18]
- Encryption of data using public-key cryptography algorithms such as RSA where it involves usage of two keys – a private key and a public key or using cryptographic hash function algorithm such as MD5 which uses a 128-bit hash value. [12].

Hence, a new approach is introduced in [19] where a new TOKEN_ID is generated for a specific Cloud service making it more reliable and worthy. An Enhancement of this idea along with Encryption is used in this paper to provide a high security along with less computation time.

2.3 SHA Algorithm

Hash function acts as essential modules in many information security applications such as key derivation, generation, authentication of digital signatures, random bit generation and password security. Important characteristics of Information Security Application includes collision resistance and preimage resistance which are key features in Authentication. Some of the longstanding hash functions include MD5, SHA-0, and SHA-1. But due to the successful attacks of these hash functions, NIST perceived a need for an alternate and dissimilar cryptographic hash function entitled as SHA-3. SHA-3 belongs to the family of sponge functions which uses permutation as the building block. The SHA-3 family comprises six functions out of which four are cryptographic hash functions, named SAH3-224, SHA3-256, SHA3-384, and SHA3-512 and two are extendable-output functions (XOFs), named SHAKE128 and SHAKE256. The Input and output of a hash function are termed as message and digest respectively. The message length may vary but the length of the digest (or hash value) is fixed [http://csrc.nist.gov/publications/drafts/fips-202/fips_202_draft.pdf] [http://opencores.org/usercontent,doc,1359445372].

3. Proposed Work

3.1 Inspiration

New Algorithms are introduced as a result of high security risk in Cloud Computing. Though existing security algorithms are adequate for Authentication purpose, they lose their consistency in large data space. Protecting such huge quantity of data/ services in cloud demands additional security. The existing approach in [19] provides security by generating an automatic TOKEN_ID from CSP for each service, establishing more consistency in accessing the service from the cloud. Nevertheless, neither the TOKEN_ID is encrypted nor the user has a unique identification leading to vulnerability of cloud data. This paper proposes an enhanced procedure to conquer the problems of the above paper by introducing the best encryption algorithm and user identification along with TOKEN_ID. A combination of user security key and service key (TOKEN_ID) is encrypted and sent along with Timestamp to provide more secure and reliable process. The Encryption is performed in both Client side (CC) and server side (CSP) to dodge intruders throughout the process of authentication. The Procedure is explained below in detail.

3.2 Nomenclature

Table 2: Abbreviation and Nomenclature.

CSP	Cloud Service Provider
CC	Cloud Client
NCC	New Cloud Client
NAC	New Account Creation
MEM_CON	Membership Confirmed
SECURITY_TOKEN	Security Token
ENCRYPT_SECURITY_TOKEN	Encrypted Security Token

NEW_CS	New Cloud Service
SERVICE_TOKEN	Service Token
ENCRYPT_ SERVICE_TOKEN	Encrypted Service Token
T	Time
REQ	Request
DATA_ACCESS	Data Access
ENCRYPT_EMBEDDED_TOKEN	Encrypt Embedded Token
DB	Database

3.3 Proposed Algorithm

```
Step 1. [NEW CLIENT REGISTRATION FOR CLOUD CLIENT]
        IF (NCC SENDS REQ: = NAC)
                THEN New Account Created & Client Registered.
        If (MEMBER: = CONFIRMED)
        THEN SECURITY_TOKEN is auto-generated by CSP & ENCRYPT_SECURITY_TOKEN send to CC.
Step 2. [NEW SERVICE REQUEST & GENERATION OF SERVICE TOKEN]
        When CC SEND REQ: = NEW_CS,
        IF (MEMBER: = CONFIRMED)
                THEN UNIQUE SERVICE_TOKEN is auto-generated on that T for SPECIFIC CLOUD SERVICE by CSP and ENCRYPT_ SERVICE_TOKEN is send to CC.
Step 3. [REQUEST FOR DATA ACCESS]
        IF (CC SEND REQ: = DATA_ACCESS)
                THEN send REQ, T and ENCRYPT_EMBEDDED_TOKEN by combining and encrypting ENCRYPT_SECURITY_TOKEN, ENCRYPT_SERVICE_TOKEN to CSP.
Step 4. [VALIDATION OF REQUEST AT CSP]
        Now the same encryption procedure is followed by CSP with TOKENs in DB and verified with the ENCRYPT_EMBEDDED_TOKEN.
        IF (ENCRYPT_EMBEDDED_TOKEN: = VALID & T: = VALID)
        {
                Authenticated Client ();
        }
        ELSE
        {
                Fake Client (Intruder);
        }
Step 5. [AUTHENTICATION PROCESS]
        IF (AUTHENTICATION SUCCESSFUL)
                THEN SECURE CHANNEL is established for DATA TRANSFER
        ELSE
                 INTRUDER Present and REPEAT FROM STEP 2.
```

3.4 Encryption Method

The SECURITY_TOKEN and SERVICE_TOKEN are autogenerated by the CSP for a specific user and specific service respectively. The Encrypted form of SECURITY_TOKEN and SERVICE_TOKEN using SHA encryption algorithm are termed as ENCRPT_SECURITY_TOKEN and ENCRPT_SERVICE_TOKEN respectively. The ENCRPT_SECURITY_TOKEN is sent to CC on successful account registration whereas the ENCRPT_SERVICE_TOKEN is forwarded to CC during Cloud Service request. The CC uses a standard amalgamation of these token to form an EMBEDDED_TOKEN. A pictorial diagram of forming the EMBEDDED_TOKEN is represented in Figure 3. The combination can be designed as one of the following ways:

a) EMBEDDED_TOKEN formed by joining both ENCRPT_SECURITY_TOKEN and ENCRPT_SERVICE_TOKEN adjacently.

   Eg. ENCRPT_SECURITY_TOKEN = abcdef
   ENCRPT_SERVICE_TOKEN = uvwxyz
   EMBEDDED_TOKEN = abcdefuvwxyz

b) EMBEDDED_TOKEN containing alternate characters of ENCRPT_SECURITY_TOKEN and ENCRPT_SERVICE_TOKEN.

   Eg. ENCRPT_SECURITY_TOKEN = abcdef
   ENCRPT_SERVICE_TOKEN = uvwxyz

EMBEDDED_TOKEN = aubvcwdxeyfz

c) Combination of tokens in multiples of 2’s or 3’s or prime numbers, etc.

The EMBEDDED_TOKEN is further encrypted at cloud Client and directed back to CSP. The ultimate data sent from CC comprises ENCRPT_EMBEDDED_TOKEN and Timestamp. The same procedure of encryption is followed at CSP and the ENCRPT_EMBEDDED_TOKEN from CC is compared with the one at CSP.

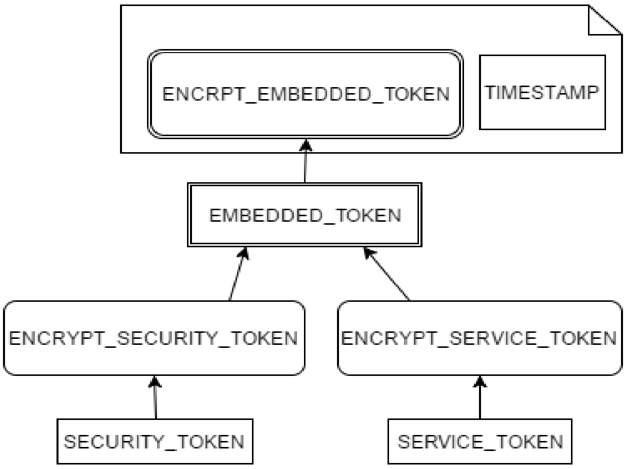

Figure 3: Encryption Technique.

3.5 Working Explanation

Initially, a NEW CLIENT sends a request for New Account Creation (NAC) to the cloud service provider (CSP). The CSP creates a new account and the client is registered. Once the membership is confirmed, CSP generates an SECURITY_TOKEN and stores in DATABASE (DB). In addition, it sends the encrypted SECURITY_TOKEN (ENCRYPT_SECURITY_TOKEN) to CLOUD CLIENT (CC) for further communication.

In the former step, when CC logs in and a sends request to access a new cloud service (NEW_CS), the CSP verifies client’s membership and after confirmation, it generates a unique SERVICE_TOKEN at that time (T) for the specific cloud service and stores in DB.

CSP sends the encrypted SERVICE_TOKEN (ENCRYPT_SERVICE_TOKEN) to CC for further use. In the following step, CC sends a request for DATA_ACCESS to CSP, at the same time T along with ENCRYPT_ EMBEDDED_TOKEN which is generated by encrypting the combination of ENCRYPT_SECURITY_TOKEN and ENCRYPT_SERVICE_TOKEN for validation. Then in the third step, the CSP retrieves the SECURITY_TOKEN, SERVICE_TOKEN from the database (DB) for the particular user and encrypts them to form ENCRYPT_SECURITY_TOKEN and ENCRYPT_ SERVICE_TOKEN respectively, and then encrypts the combination of them to validate it with the received ENCRYPT_EMBEDDED_TOKEN. If received ENCRYPT_EMBEDDED_TOKEN and generated ENCRYPT_EMBEDDED_TOKEN at CSP matches, then data transfer fails because INTRUDER or fake client may try to steal the data due to the insecurity of channel. The client sends a request again and then repeats the steps mentioned earlier.

3.6 Methodology Diagram

3.6.1 Flowchart (Proposed Flowchart)

A flowchart represents step by step working procedure of a process. The detailed explanation of performing the encryption at CC and CSP are shown in Figure 4. The flow diagram starts with a new client registers to access the data from the cloud / existing client logs in to request a new service from cloud to access the data. The procedure flows through encryption process at CSP and CC until the Client is validated. Once the client is validated, the data is transferred successfully.

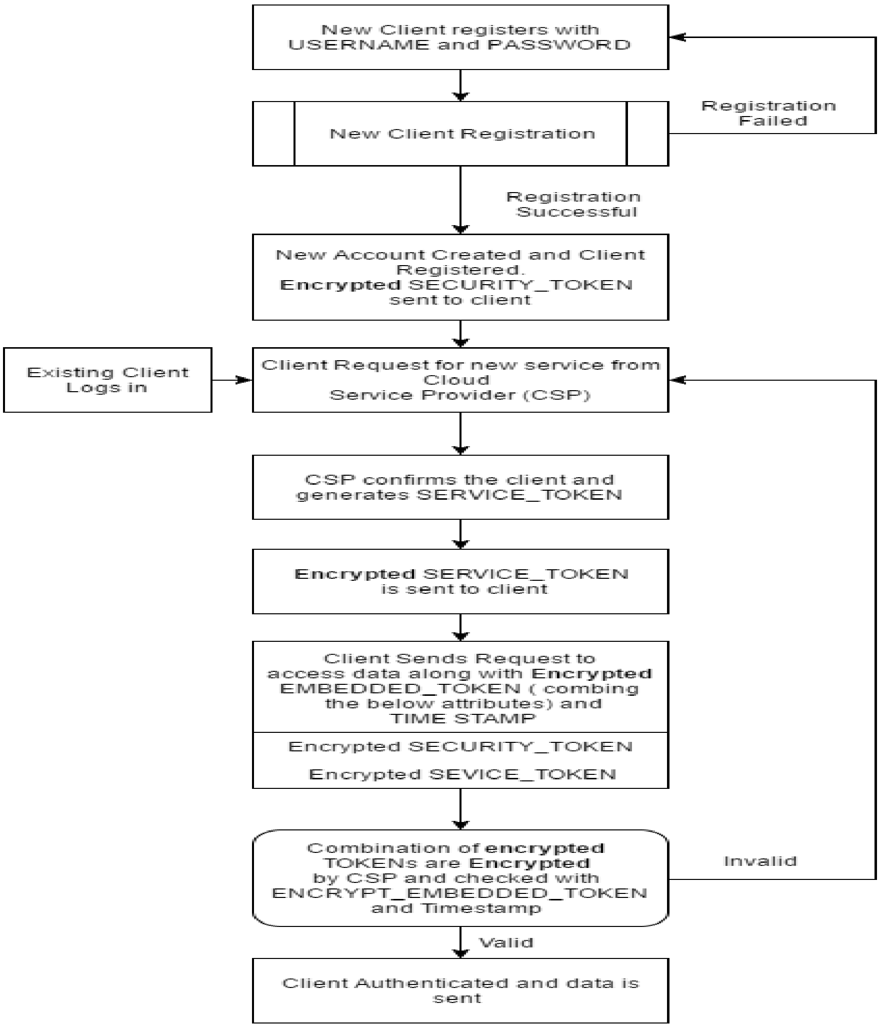
New Client registers with USERNAME and PASSWORD
New Client Registration
Registration Failed
Registration Successful
New Account Created and Client Registered. Encrypted SECURITY_TOKEN sent to client
Existing Client Logs in
Client Request for new service from Cloud Service Provider (CSP)
CSP confirms the client and generates SERVICE_TOKEN
Encrypted SERVICE_TOKEN is sent to client
Client Sends Request to access data along with Encrypted EMBEDDED_TOKEN (combing the below attributes) and TIME STAMP
Encrypted SECURITY_TOKEN
Encrypted SEVICE_TOKEN
Combination of encrypted TOKENs are Encrypted by CSP and checked with ENCRYPT_EMBEDDED_TOKEN and Timestamp
Invalid
Valid
Client Authenticated and data is sent

Figure 4: DEBTDS Flow chart.

3.6.2 Sequence Diagram

A sequence diagram symbolizes the process with respect to the Object’s point of view. Each function originates at an object and ends at another. Here, Cloud Client (CC), Cloud Service Provider (CSP) and database (DB) are objects of the data security procedure. The sequence diagram for the proposed model is represented in Figure 5.

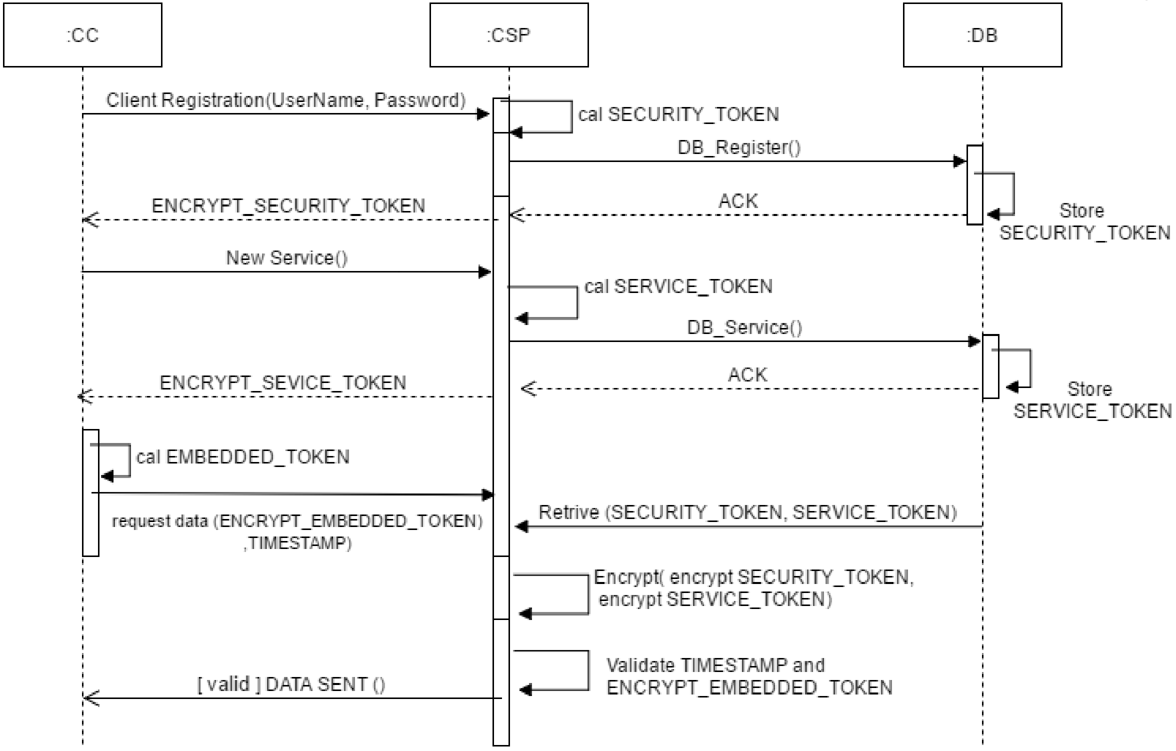
:CC
:CSP
:DB
Client Registration(UserName, Password)
cal SECURITY_TOKEN
DB_Register()
ENCRYPT_SECURITY_TOKEN
ACK
Store SECURITY_TOKEN
New Service()
cal SERVICE_TOKEN
DB_Service()
ENCRYPT_SEVICE_TOKEN
ACK
Store SERVICE_TOKEN
cal EMBEDDED_TOKEN
request data (ENCRYPT_EMBEDDED_TOKEN) ,TIMESTAMP)
Retrive (SECURITY_TOKEN, SERVICE_TOKEN)
Encrypt(encrypt SECURITY_TOKEN, encrypt SERVICE_TOKEN)
Validate TIMESTAMP and ENCRYPT_EMBEDDED_TOKEN
[valid] DATA SENT ()

Figure 5: DEBTDS Sequence diagram.

4. Conclusion

This paper proposes an enhanced approach to provide secure and reliable transfer of data/service between Cloud Client and Cloud Service Provider. The complexity of the algorithm is proportional to the encryption method and computation of EMBEDDED_TOKEN. Furthermore, double encryption procedure makes the algorithm robust compared to TBDS algorithm. The Combination technique to compute EMBEDDED_TOKEN plays a vital role in the complexity of the algorithm. When compared to advanced approaches, this algorithm has high computation speed as it involves only encryption (not decryption). Thus, taking into consideration the above aspects, we assuredly propose the procedure to be secure, reliable and efficient.

The algorithm Proposed in this paper can be implemented using web development Languages to demonstrate it’s more reliable, secure than existing security algorithms. Future work includes conducting a literature review on various security algorithm in detail to scale the reliability. Usage of Twin cloud can be supplementary to this algorithm to provide additional security and diminish latency. HTTPS (Hypertext Transfer Protocol Secure) and SSH (Secure Shell) can be adopted in web application along with this algorithm to enhanced data security and integrity.

References

1. Anthony Bisong and Syed M. Rahman “An Overview of the Security Concerns in Enterprise Cloud Computing”, International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.1, January 2011, pp. 30-45.

2. Aized Amin Soofi, M. Irfan Khan, Fazal-e-Amin “A Review on Data Security in Cloud Computing”, International Journal of Computer Applications (0975 – 8887) Volume 94 – No 5, May 2014, pp.12-20.
3. Khorshed, T.M., Ali, A.B.M.S. and Wasimi, S.A. (2012).” A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing”. Future Generation Computer Systems, pp. 833–851.
4. Monjur Ahmed and Mohammad Ashraf Hossain “Cloud Computing And Security Issues In The Cloud”, International Journal of Network Security & Its Applications (IJNSA), Vol.6, No.1, January 2014, pp.25-36.
5. Ahmad-Reza Sadeghi, Thomas Schneider, and Marcel Winandy “Token-Based Cloud Computing Secure Outsourcing of Data and Arbitrary Computations with Lower Latency”, A. Acquisti, S.W. Smith, and A.-R. Sadeghi (Eds.): TRUST 2010, LNCS 6101, pp. 417–429, 2010.
6. Sven Bugiel, Stefan Nurnberger, Ahmad-Reza Sadeghi, Thomas Schneider “Twin Clouds: An Architecture for Secure Cloud Computing”, B. de Decker et al. (Eds.): CMS 2011, LNCS 7025, pp. 32–44, 2011.
7. Atallah, M.J., Pantazopoulos, K.N., Rice, J.R., Spafford, E.H” Secure outsourcing of scientific computations”, Advances in Computers, Vol 54, pp. 216–272 (2001).
8. Kamara, S., Lauter, K. “Cryptographic cloud storage”, Workshop on Real-Life Cryptographic Protocols and Standardization (RLCPS 2010) - co-located with Financial Cryptography, January 2010, LNCS. Springer, Heidelberg.
9. Gentry, C. “Fully homomorphic encryption using ideal lattices”, ACM Symposium on Theory of Computing (STOC 2009), pp. 169–178. ACM, New York 2009.
10. Yao, A.C. “How to generate and exchange secrets”, IEEE Symposium on Foundations of Computer Science (FOCS 1986), pp. 162–167. IEEE, Los Alamitos (1986).
11. Gennaro, R., Gentry, C., Parno, B.”Non-interactive verifiable computing: Outsourcing computation to untrusted workers”, Cryptology ePrint Archive, Report 2009/547 (2009).
12. M. Vijayapriya “Security Algorithm in Cloud Computing: Overview”, International Journal of Computer Science & Engineering Technology (IJCSET) Vol. 4 No. 09 Sep 2013, pp. 1209- 1211.
13. K.S.Suresh, K.V.Prasad “Security Issues and Security Algorithms in Cloud Computing”, International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE) Volume 2, Issue 10, October 2012, pp. 110-114.

14. Parsi Kalpana, Sudha Singaraju “Data Security in Cloud Computing using RSA Algorithm”, International Journal of Research in Computer and Communication technology (IJRCCT) Vol 1, Issue 4, September 2012, pp.143-146.
15. Sven Bugiel, Stefan Nurnberger, Ahmad-Reza Sadeghi, and Thomas Schneider, “Twin Clouds: Secure Cloud Computing with Low Latency”, B. de Decker et al. (Eds.): CMS 2011, LNCS 7025, pp. 32–44, 2011.
16. Y. Ghebghoub, S. Oukid, and O. Boussaid , “A Survey on Security Issues and the Existing Solutions in Cloud Computing”, International Journal of Computer and Electrical Engineering, Vol. 5, No. 6, December 2013, pp. 587-590.
17. S. Ruj, A. Nayak, and V. Stojmenovic, “DACC: Distributed Access Control in Clouds”, International Joint Conference of IEEE TrustCom-11/IEEE ICESS-11/FCST-11, pp. 91-98, 2011.
18. N. Antony and A. A. R. Melvin, “A Survey on Encryption Schemes in the Clouds for Access Control,” International Journal of Computer Science and Management Research, Vol. 1, Issue 5, pp. 1135-1139, December 2012.
19. R. K. Seth, Rimmy Chuchra, Simran “TBDS- A New Data Security Algorithm in Cloud Computing”, International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 5 (3), 2014, pp.2703-2706.