An Implementation of Multi-Prime RSA Algorithm in Data Cloud using Cloud SQL

Naveen Kumar R,
Research Scholar, Department of Computer Science,
S.V.University, Tirupathi

Narayana Galla
Research Scholar, Department of Computer Science,
S.V.University, Tirupathi

Almuhteb Sulaiman
Research Scholar, Department of Computer Science,
S.V.University, Tirupathi

Prof . PadmavathammaMokkala*
*HOD, Department of Computer Science, S.V.University,
Tirupathi

Abstract: **Aim of this journal is to protect the data stored on cloud by using security algorithm. Cloud computing model advances many web applications because of its elasticity nature. This type of computing reduces operating cost and increases the efficiency of computing while data fetching. Even though efficiency, Data increased, still there is security threat for the data that is stored in third party area especially in Internet while data mining. Due to data security issue with cloud computing many business organization have fear in storing their data in Cloud. So the most challenging task of the business organization is to provide high security for their data since the data are sensible related to their business. To ensure the security of data while data mining, we proposed a method of providing security by implementing Multi-Prime RSA algorithm using cloud SQL to the data mining that will be stored in the third party area.**

Keyword: RSA, Multi-Prim RSA algorithm, Cloud computing, Cloud SQL, Data storage security, Security , Data mining

INTRODUCTION:

In the modern distributed era different servicesoffered in the Internet as a traditional hosting system. Butin the traditional hosting system storage and usage arefixed. But the current trend in business requires dynamismin compute and data storage. This leads to thedevelopment of cloud model.Cloud computing [1,2,3] proposes new model for computingand related issues like compute, storage, software. It provides development environment, allocation andreallocation of resources when needed, storage andnetworking facility virtually. It satisfies the on-demandneeds of the user. It facilitates the sharable resources "as-a-service" model. For the organization, the cloud offersdata centers to move their data globally. It eliminates theresponsibility of local nodes for maintaining their data and also cloud supports customizable resources on the web.Cloud Service[4] Providers maintains computing resourcesand data automatically via software.Data security is an important aspect of quality of service (Cong et al., 2009). As a result, security must beimposed on data by using encryption strategies to achievesecured data storage and access. Because of opaqueness nature of cloud, it is still having security issues. The cloudinfrastructure even more reliable and powerful then personal computing, but wide range of internal, externalthreats for data stored on the cloud. Since the data are notstored in client area, implementing security measurescannot be applied directly. In this work, we implementRSA[5,6] algorithm before storing the sensitive data in cloud.When the authorized user request the data for usage thendata decrypted and provided to the user.

Whenever new features introduced then automatically reflected in the browser by refreshing it. Additional functionalities released in small sized chunks, this leads to reduce the change management hurdles. In order to satisfy the customer needs from anywhere the information posted by the customer is not maintained in a single site or computer, rather maintained in number of trusted nodes. To get high reliability and availability the data processed by the customer is stored and updated in multiple machines. If anyone node gets failed, the other one provides the service.

Google cloud SQL is very easy to use and not requiring any other software. Google cloud SQL concern, MySQL instance used and are similar to MYSQL. It is having all features and facilities provided by MYSQL.

In this Journal, we propose a way of implementingMutliPrimeRSA algorithm with cloud SQL to guaranty the datastorage security in cloud. This approach can be eitherimplemented by the party who stores his data or by theservice provider

Security challenges in cloud: Though Cloud offers sophisticated storage and access environment; it is not 100% reliable; the challenge exists in ensuring the authorized access. Because third parties make the decision regarding our data, security is a big concern. So cloud must ensure that the data accessed is by the trusted users. Cloud computing uses multi-domain environments and each of

which having different requirements for security. Authentication and identity management can help the users to authenticate and getting services based on their credentials (Bertino*et al*., 2009; Ko*et al*., 2009). Key issue about identity management in cloud is different kinds of protocols and its interoperability. This multidomain issue complicates protection measures (Bruening and Treacy, 2009). Fine-grained access control needed for cloud because of its heterogeneity of services and multi-domain access requirements. Dynamic, context based and attributes based requirements needed by the cloud. It must also ensure that the implemented polices is managed easily. Different providers offer various services and variety of security approaches used. So a mechanism needed to ensure that dynamic interoperability among providers. But current literature has individual domain polices that are verified during integration. Thus trusted framework has to be developed to establish trust. Whenever the organization wanted to move business data to cloud, they might have greatest fear about security of their private information. The protection against unauthorized access is the major issue for private data. So the cloud providers must give assurance to their customers regarding higher transparency for all operations and privacy assurance. Because users may work with different places like office, home, public places and try to access the data, they should be able to use their identity in terms of digital signature and transfer data. Also privacy preserving standards have to verify the identity related attributes. Cloud storage concern the user does not have control over data until he has been gain access. To provide control over data in the cloud data-centric security is needed. Before accessing the data it should satisfy the policy rules already defined. So cloud should enforce this scheme by using cryptographic approaches.

METHODOLOGYS

Mutli-Prime RSA algorithm: We use the Multi-Prime RSA algorithm as a basis to provide data-centric security for shared data:

- Randomly chosen distinct primes p1, . . . , pr.
- Calculate n =$\Pi^{r}_{i=1}P_i$.
- Calculate Ø(n) = $\Pi^{r}_{i=1}(P_i-1)$
- Select e such that e is relatively prime to Ø (n) and less than Ø (n).
- Calculate d such that de congruent modulo 1 (modØ (n)) and d<Ø (n).
- Public key = {e, n}
- Private key = {d, n}
- Cipher message c = (msg^e)mod n
- Plain text p = c^dmod n

Implement Mutli-Prime RSA algorithm in cloud SQL:

The following are the procedure to create Database, Tables in Cloud SQL and to implement RSA algorithm:

Step 1: Click instance name to see the properties associated with it

Step 2: Select "SQL Prompt" tab. All databases automatically loaded

Step 3: Create database for the application by using "create database…" query and create necessary tables

Step 4: Insert records to the tables by using "Insert into …" Query

Step 5: Create user interface for the application

Step 6: Write Java code to implement Multi Prime RSA algorithm in cloud and debug the application in cloud environment.

Step 7: Store the data in an encrypted format. Display the content in decrypted format while accessing

RESULTS AND DISCUSSION

We have created and tested the application using Java and JSP in Eclipse.

Step 1: Database instance created in google cloud named as "SVU-College".

Step 2: "ResearchStudent" table created in SVU-College databaseand it has all necessary fields about theResearchStudent.

Step 3: An application "sampleResearchApp" wascreated in google app engine

Step 4: User interface designed to manipulate the supplierdetails. From the home page choose ResearchStudnetlink,then it displays ResearchStudentsentry form to enter the details.

Step5: By clicking the "submit" button the entereddetails and encrypted by the MutliPrime RSA algorithm

Step 6: During retrieval of data, it is decrypted after checking the generated private key with existing Public key

Step 7: Using the interface, decrypted data displayed

CONCLUSION

In this paper, we have implemented Mutli-Prime RSA algorithm in cloud Application Engine using cloud SQL. From the results we obtained it is proved that Multi-Prime RSA gives more protection for the data compare to RSA, which is stored in Cloud. Only authorized user canretrieve the encrypted data and decrypt it. Even if anyone happens to read the data accidentally, the original meaning of the data will not be understood. Also we argued that the importance of security and privacy of data stored and retrieved in the cloud. We utilize Mutli-Prime RSA algorithm on Cloud Application Engine to provide efficient and secured data storage scheme.

REFERENCES

[1] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing", Journal of Network and Computer Applications, Vol. 34, pp. 1-11 **(2011)**.

[2] S. Ramgovind, M. M. Eloff and E. Smith, "The Management of Security in Cloud Computing", Information Security for South Africa (ISSA), pp. 1-7 **(2010)**.

[3] A. F. Mohammad and H. Mcheick, "Cloud Services Testing: An Understanding", Procedia Computer Science, Vol. 5, pp. 513-520 **(2011)**.

[4] A. F. Mohammad and H. Mcheick, "Cloud Services Testing: An Understanding", Procedia Computer Science, Vol. 5, pp. 513-520 **(2011)**.

[5] C. H. Lin and C. C. Chang, "A Server-Aided Computation Protocol for RSA Enciphering Algorithm", Intern. J. Computer Math., Vol. 53, pp. 149-155 **(1993)**.

[6] R. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signature and Public Key Cryptosystems", Communications of the ACM, Vol. 21, No. 2, pp. 120-126 **(1978)**.