GRD Journal for Engineering | Volume 1 | Issue 5 | April 2016
ISSN: 2455-5703

An Efficient and Dynamic Search on Encrypted Data

Sowjanya C M
Assistant Professor
Department of Information Science & Engineering
JVIT, Bidadi

Sushmitha
Student
Department of Information Science & Engineering
JVIT, Bidadi

Ramya N
Student
Department of Information Science & Engineering
JVIT, Bidadi

Supreeth
Student
Department of Information Science & Engineering
JVIT, Bidadi

Sagar Koirala
Student
Department of Information Science & Engineering
JVIT, Bidadi

Abstract

A secure outsourcing of computation to an untrusted (cloud) service provider is becoming more and more important. To protect data privacy, the sensitive data should be encrypted by data owner before outsourcing, which makes the search sequence useless. In this paper, we present an efficient, secure and parallel dynamic search scheme on encrypted cloud data which concurrently supports dynamic update operation like deletion and insertion of documents. The most important shortcoming that is addressed by OTPs is that, in contrast to static passwords, they are not vulnerable to replay attacks. The AES algorithm is use to encrypt and decrypt the message with efficient multi-keyword Boolean search. We show that our approach is highly efficient and ready to be deployed in the real world cloud storage systems.

Keywords- Multi-keyword Boolean search, OTP, AES algorithm, dynamic update, encrypted cloud data

I. INTRODUCTION

Cloud computing is one way of computing. Here the computing resources are shared by many users. The benefits of can be extended from individual users to organizations. The data storage in cloud is one among them. The advantages of cloud computing are that the use will be able to access applications and data on the cloud from anywhere in the globe, making the cloud appear a single point of access. Many cloud platforms like Google drive, cloud, SkyDrive, amazon S3 Drop box and Microsoft Azure provide storage services. Security and privacy concerns have been the major challenges in cloud computing. The hardware and software security mechanisms like firewalls etc. Have been used by cloud provider. These solutions are not sufficient to protect data in cloud from unauthorized users because of low degree of transparency. However, some problems may be caused in this circumstance since the cloud service provider possesses full control of the outsourced data. Unauthorized operation on the outsourced data may exist on account of curiosity or profit. To protect the privacy of sensitive information, sensitive data (e.g., emails, photo albums, personal health records, financial records, etc.) should be encrypted by the data owner before outsourcing, which makes the efficient plaintext keyword search sequence useless.

Cloud providers and customers must share the responsibility for security and privacy in cloud computing environments, but sharing levels will differ for different delivery models, which in turn affect cloud extensibility:

- In SaaS, providers typically enable services with a large number of integrated features, resulting in less extensibility for customers. Providers are more responsible for the security and privacy of application services, more so in public than private clouds where the client organization might have stringent security requirements and provide the needed enforcement services. Private clouds could also demand more extensibility to accommodate customized requirements.
- In PaaS, the goal is to enable developers to build their own applications on top of the platforms provided. Thus, customers are primarily responsible for protecting the applications they build and run on the platforms. Providers are then responsible for isolating the customers’ applications and workspaces from one another.
- IaaS is the most extensible delivery model and provides few, if any, application-like features. It’s expected that the consumers secure the operating systems, applications, and content. The cloud provider still must provide some basic, low-level data protection capabilities.

II. Related work

The encryption on data is an effective way to protect the confidentiality of data in cloud. But when it comes to searching, efficiency gets low. In literature many research works are not efficient in searching specially for complex queries. This inefficiency may lead to leakage of valuable information to unauthorized peoples.

A. *Security Challenges for the Public Cloud*

Author: K. Ren, C.Wang, Q.Wang et al.,
Abstract: Cloud computing represents today's most exciting computing paradigm shift in information technology. However, security and privacy are perceived as primary obstacles to its wide adoption. Here, the author's outline several critical securities challenges and motivates further investigation of security solutions for a trustworthy public cloud environment.

B. *Cryptographic Cloud Storage*

Author: Seny Kamara, Kristin Lauter
Abstract: We consider the problem of building a secure cloud storage service on top of a public cloud infrastructure where the service provider is not completely trusted by the customer. We describe, at a high level, x We survey the benefits such architecture would provide to both customers and service providers and give an overview of recent advances in cryptography motivated specifically by cloud storage.

C. *A Fully Homomorphic Encryption Scheme*

Author: C. Gentry
Abstract: We propose the first fully homomorphic encryption scheme, solving an old open problem. Fully homomorphic encryption has numerous applications. For example, it enables encrypted search engine queries—i.e., a search engine can give you a succinct encrypted answer to your (Boolean) query without even knowing what your query was. It also enables searching on encrypted data; you can store your encrypted data on a remote server, and later have the server retrieve only files that (when decrypted) satisfy some Boolean constraint, even though the server cannot decrypt the files on its own. More broadly, it improves the efficiency of secure multiparty computation.

III. Proposed system

A. *One Time Password*

Dynamic password (namely, One-Time-Password) technology is a sequence password system and is the only password system proved non-decrypted in theory. Its basic idea is to add uncertain factor in authentication so that users need to provide different messages for authentication each time. By this way, the applications themselves can obtain higher security guarantee than those use static password technologies. When login request from user is received, server system generates a one-time password and sends it through a SMS to a GSM cell phone registered for that specified user. The one-time password has a default timeout. In the second phase of the authentication, a request is sent with the user id and a hash of the one-time password. If both the one- time and user specified password is valid then the user will be authenticated.
Two-way one-time authentication works as follows:

1) Step 1. User send a login request server with its ID and Pin (Static password)
2) Step 2. If ID and PIN match with the ID and PIN stored in database, server generate a onetime password (OTP) and send it through SMS or email to the user.
3) Step 3. Server request user for OTP.
4) Step 4. User enters OTP and if it matches then user is authenticated.

B. *System Architecture*

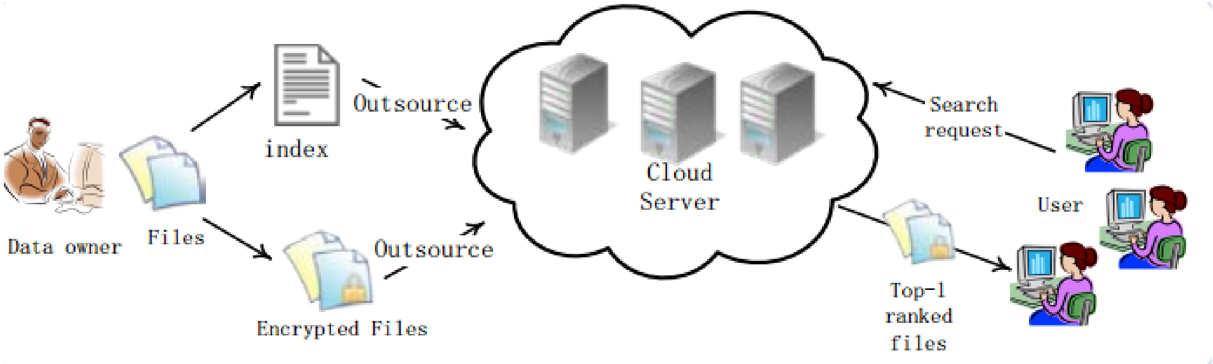

Fig. 1: Architecture of dynamic search on encrypted data

1) Modules

1) Data User Module
2) Data Owner Module
3) File Upload Module
   Encryption
4) Rank Search Module
5) File Download Module
   Decryption
6) View Uploaded and Downloaded File.

1) Data User Module

This module includes the user registration login details.

2) Data Owner Module

This module helps the owner to register them details and also include login details

3) File Upload Module

This module helps the owner to upload his file with encryption using algorithm. This ensures the files to be protected from unauthorized user.

4) Rank Search Module

This module ensures the user to search the file that is searched frequently using rank search.

5) File Download Module

This module allows the user to download the file using his secret key to decrypt the downloaded data.

6) View Uploaded and Downloaded File

This module allows the Owner to view the uploaded files and downloaded files

C. AES Algorithm

AES is an iterative and a symmetric key block cipher that uses three keys strengths of 128, 192 and 256 bits. The AES encryption and AES decryption occurs in blocks of 128 bits. The maximum block size can be 256 bits however the key size has no theoretical maximum. Unlike the public key ciphers, the AES cryptography uses the same key to encrypt and decrypt data. The user simply need to select AES encrypt or AES decrypt and the encrypt or will do the rest. It is one of the perfect cryptography algorithms to protect personal data. The encrypt AES tool converts the input plain text to cipher text in a number of repetitions based on the encryption key. The AES decrypt method uses the same process to transform the cipher text back to the original plain text using the same encryption key. It is one of the strongest encryption methods that are hard to break. The hash is used to protect the encryption key against brute force attack. It is being used to secure online information, financial transactions by banks, e-commerce sites and other financial institutions.

IV. Conclusion

In this paper, a multi-keyword ranked search scheme on encrypted cloud data is proposed, which meanwhile supports parallel search on encrypted data. We use one-time password technique. Taking security and privacy into consideration, we employ AES algorithm to encrypt the index and the queried vector, so that we can obtain the accurate ranked results and protect the confidence of the data well. The experimental effect is remarkable.

Acknowledgement

I would like to express my thanks to my Hod GSS Rao and my guide Asst Prof. Sowjanya. C.M. highly for their appreciable support and encouragement also for guidance is a force behind the completion of this paper. We are grateful for all the suggestions and hints provided by them. Our acknowledgment of gratitude to all who supported to make it possible.

References

[1] Security challenges for the public cloud Author: K. Ren, C.Wang, Q.Wang et al.,
[2] Cryptographic cloud storage Author: Seny Kamara, Kristin Lauter
[3] A fully homomorphic encryption scheme Author: C. Gentry