International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.03, March 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

EFFICIENT SHORT-TERM KEYWORD SEARCH AND DE-DUPLICATION APPROACH FOR UN-TRUSTED CLOUD ENVIRONMENT

S.AKILA, A.DEIVANAI, M.MANJU LA, Mrs B.KALAISELVI AP/CSE

Computer Science and Engineering, Mahendra Engineering College for Women (India)

Computer Science and Engineering, Mahendra Engineering College for Women (India)

Computer Science and Engineering, Mahendra Engineering College For Women (India)

Computer Science and Engineering (Ap), Mahendra Engineering College for Women (India)

ABSTRACT

Temporary keyword search on confidential data in a cloud environment is the main focus of this research. The cloud providers are not fully trusted. So, it is necessary to outsource data in the encrypted form. In the "attribute-based keyword search (ABKS) schemes", the authorized users can generate some search tokens and send them to the cloud for running the search operation. These search tokens can be used to extract all the cipher texts which are produced at any time and contain the corresponding keyword. Since this may lead to some information leakage, it is more secure to propose a scheme in which the search tokens can only extract the cipher texts generated in a specified time interval. To this end, in this paper, we introduce a new cryptographic primitive called" key-policy attribute-based temporary keyword search (KPABTKS) "which provide this property. In addition to providing secure keyword search, duplication of data is also avoided. Therefore our system achieves efficient memory usage and attains effective and secure data retrieval. Furthermore, we show that the complexity of the encryption algorithm is linear with respect to the number of the involved attributes. Performance evaluation shows our scheme's practicality

INTRODUCTION

Today, cloud computing plays an important role in our daily life, because it provides efficient, reliable and scalable resources for data storage and computational activities at a very low price. However, the direct access of the cloud to the sensitive information of its users threatens their privacy. A trivial solution to address this problem is encrypting data before outsourcing it to the cloud. However, searching on the encrypted data is very difficult. Public key encryption with keyword search (PEKS) is a cryptographic primitive which was first introduced to facilitate

searching on the encrypted data. In PEKS, each data owner who knows the public key of the intended data user generates a searchable cipher text by means of his/her public key, and outsources it to the cloud. Then, the data user extracts a search token related to an arbitrary keyword by using his/her secret key, and issues it to the cloud. The cloud service provider (CSP) runs the search operation by using the received search token on behalf of the data user to find the relevant results to the intended keywords .The notion of attribute-based keyword search (ABKS) to allow a data owner to control the access of data users for searching on his/her outsourced encrypted data is introduced. They used attribute-based encryption (ABE) to construct a searchable cryptographic primitive in the multi-sender/multi receiver model. In their work, the legitimate data users can enlist the cloud to run the search operation on behalf of them without requiring any interaction with the data owner. In a secure ABKS scheme, a data owner cannot obtain any information about the keywords which the data users intend to look for. However, in all of the PEKS and ABKS schemes, once the cloud receives a valid search token related to a certain keyword, the cloud can investigate the keyword's presence in the past and any future cipher text. So, if the adversary realizes the corresponding keyword of the target search token, then she will be able to get some information about the next documents which will be outsourced to the cloud. Therefore, it will be more secure to limit the time period in which the search token can be used.

The notion of public key encryption with temporary keyword search (PETKS) which restricts the validation of the token to a certain time period. They applied anonymous identity-based encryption in their generic scheme. In addition, proposed another public key searchable encryption in the context of temporary keyword search. Despite the good features of their schemes, these schemes do not provide the facility for data owners to enforce their intended access policy. In this paper, we propose a novel notion of Key-Policy Attribute-Based Temporary Keyword Search (KP-ABTKS). In KP-ABTKS schemes, the data owner generates a searchable cipher text related to a keyword and the time of encrypting according to an intended access control policy, and outsources it to the cloud. After that, each authorized data user selects an arbitrary time interval and generates a search token for the intended keyword to find the cipher text.

Then, he/she sends the generated token to the cloud to run the search operation. By receiving the token, the cloud looks for the documents contain the intended keyword. The search result on a cipher text is positive, if (i) the data user's attributes satisfies the access control policy, (ii) the time interval of the search token encompasses the time of encrypting, and (iii) the search token and the cipher text are related to the same keyword. To show that the proposed notion can be realized, we also propose a concrete instantiation for this new cryptographic primitive based on bilinear map.

Cloud computing greatly facilitates data providers who want to outsource their data to the cloud without disclosing their sensitive data to external parties and would like users with certain credentials to be able to access the data. This requires data to be stored in encrypted forms with access control policies such that no one except users with attributes (or credentials) of specific forms can decrypt the encrypted data.

An encryption technique that meets this requirement is called attribute-based encryption (ABE), where a user's private key is associated with an attribute set, a message is encrypted under an access policy (or access structure) over a set of attributes, and a user can decrypt a cipher text with his/her private key if his/her set of attributes satisfies the access policy associated with this cipher text. However, the standard ABE system fails to achieve secure de duplication, which is a technique to save storage space and network bandwidth by eliminating redundant copies of the encrypted data stored in the cloud.

On the other hand, to the best of our knowledge, existing constructions for secure de duplication are not built on attribute-based encryption. Nevertheless, since ABE and secure de duplication have been widely applied in cloud computing, it would be desirable to design a cloud storage system ossessing both properties.

EXISTING SYSTEM:

The data user extracts a search token related to an arbitrary keyword by using his/her secret key, and issues it to the cloud. The cloud service provider (CSP) runs the search operation by using the received search token on behalf of the data user to find the relevant results to the intended keywords. The PEKS and ABKS schemes, once the cloud receives a valid search token related to a certain keyword, the cloud can investigate the keyword's presence in the past and any future cipher text. So, if the adversary realizes the corresponding keyword of the target search token, then she will be able to get some information about the next documents which will be outsourced to the cloud.

DISADVANTAGES:

- Security in data storing and retrieving is less.
- Possible for adversaries to guess sequence of encrypted keyword.
- Maintaining storage is difficult.

PROPOSED SYSTEM

A novel notion of Key-Policy Attribute-Based Temporary Keyword Search (KP-ABTKS) is proposed. In KP-ABTKS schemes, the data owner generates a searchable cipher-text related to a keyword and the time of encrypting according to an intended access control policy, and outsources it to the cloud. After that, each authorized data user selects an arbitrary time interval and generates a search token for the intended keyword to find the cipher-text. Then, he/she sends the generated token to the cloud to run the search operation. By receiving the token, the cloud looks for the documents contain the intended keyword. The search result on a cipher-text is positive, if (i) the data user's attributes satisfies the access control policy, (ii) the time interval of the search token encompasses the time of encrypting, and (iii) the search token and the cipher-text are related to the same keyword. To show that the proposed notion can be realized, we also propose a concrete instantiation for this new cryptographic primitive based on bilinear map. In addition, while uploading data from users to cloud duplication is verified before updating it. Hence this achieves better result compared to existing system.

International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.03, March 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

ADVANTAGES

- Achieves better result in securing data from cloud.
- Attains maximum accuracy.
- Efficient storage utilization
- .Security level is high

SYSTEM REQUIREMENTS.

HARDWARE REQUIREMENTS (Minimum requirement):

The section of hardware configuration is an important task related to the software development insufficient random access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

- System : Pentium Dual Core.
- Hard Disk : 120 GB.
- Monitor : 15'' LED
- Input Devices : Keyboard, Mouse
- Ram : 1GB.

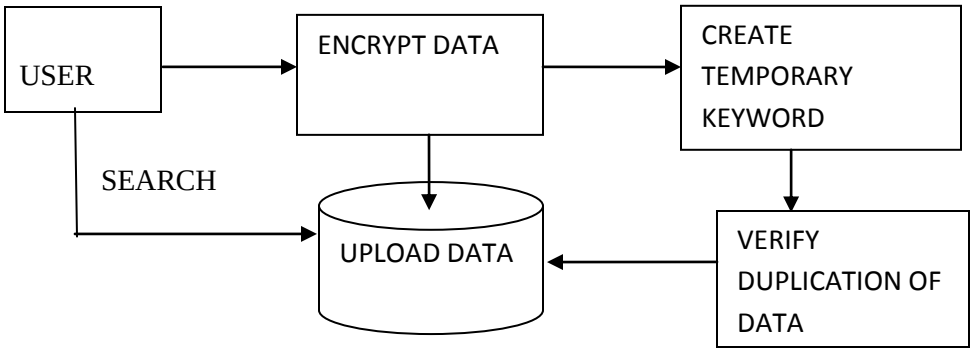
SOFTWARE REQUIREMENTS:

A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system. This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. It allow the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

- Operating system : Windows 7.
- Coding Language : JAVA/J2EE
- Tool : Net beans 7.2.1
- Database : MYSQL

International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.03, March 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

SYSTEM ARCHITECTURE



CONCLUSION

Securing cloud storage is an important problem in cloud computing. We addressed this issue and introduced the notion of key-policy attribute-based temporary keyword search (KPABTKS). According to this notion, each data user can generate a search token which is valid only for a limited time interval. We proposed the first concrete construction for this new cryptographic primitive based on bilinear map. We formally showed that our scheme is provably secure in the random oracle model. The complexity of encryption algorithm of our proposal is linear with respect to the number of the involved attributes. In addition, the number of required pairing in the search algorithms is independent of the number of the intended time units specified in the search token and it is linear with respect to the number of attributes. Similarly duplication of data is avoided during uploading to cloud. Hence this achieves maximum secure data retrieval from cloud and efficient storage maintenance.

REFERENCE

Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren, "Toward efficient multi keyword fuzzy search over encrypted outsourced data with accuracy improvement," IEEE Transactions on Information Forensics and Security, vol. 11, no. 12, pp. 2706–2716, 2016.

[10] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 340– 352, 2016.