International Journal on Applications in Engineering and Technology
Volume 8 : Issue 2 : May 2022, pp 149 –156 www.aetsjournal.com ISSN (Online) : 2455 - 0523

SECURE PASSWORD-PROTECTED ENCRYPTION KEY FOR DEDUPLICATED CLOUD STORAGE SYSTEMS

MADHAVARAJ SS[1], ARUN KUMAR NS[2], SRINIVASAN K[3]

[1,2,] *Undergraduate student, Department of Computer Science and Engineering,*

Paavai College of Engineering

[3] *Assistant Professor, Department of Computer Science and Engineering,*

Paavai College of Engineering

Abstract: - We present SPADE, an encrypted data deduplication system that is resistant to compromised key servers and relieves users of the key management problem, in this work. We present a proactivation system for servers-aided message-locked encryption (MLE) that replaces key servers with newly engaged ones on a periodic basis, renewing security protection and preserving encrypted data deduplication. To defend against dictionary guessing attacks, we describe a server-assisted password-hardening protocol. We also propose a password-based multilayer encryption mechanism and a password-based authentication mechanism based on the protocol, which we incorporate into SPADE to allow users to access their data using only their passwords. Comprehensive analysis and experimental evaluations establish SPADE's provable security and great efficiency.

***Key Words*:** Message-locked encryption, brute-force attacks, password-hardening protocol, password-based layered encryption, dictionary guessing attacks, password-based authentication

I. INTRODUCTION

Steganography is a form of encrypted communication that allows for private and secure

Because of the rapid growth of cloud computing, cloud storage has gained widespread acceptance among individuals and businesses for its benefits of universal access, low pricing, and on-demand service. To alleviate their computing burden, users can outsource difficult computations to the cloud [40], [41]. Users can also outsource their large-scale data to the cloud to relieve the strain on their local storage. As a result of this development, it is critical to ensure the quality of data storage services for users and the cloud. On the one hand, due to the unavoidable operating faults or software/hardware problems in the cloud, the outsourced data may be destroyed or lost. As a result, developing cloud storage auditing, which allows users to check the integrity of cloud storage, is crucial.

Because a secret "seed" is contained in the convergent key, the cloud cannot deduce or derive the convergent key from the content of the file. Unfortunately, using brute-force dictionary attacks, the key server can infer or extract the content from the file's hash value sent from the user.

As a result, this technique will not be able to completely avoid brute-force dictionary attacks. Furthermore, all users who want to upload a file to the cloud must create a file index and submit it to the cloud for duplicate detection. The cloud can check whether a file uploaded by a user is duplicated or not using the file index. If the cloud has saved the file index, subsequent users will no longer need to upload data to the cloud. The hash value of the file is usually used as the file index in most deduplication techniques. Because malicious cloud or other parties may guess or derive the content of a file using brute-force dictionary attacks, data privacy will be compromised.

1.1 CLOUD STORAGE AUDITING

A cloud auditor is a third party who can conduct an independent review of cloud service controls with the goal of expressing an opinion. Audits are conducted to ensure that standards are being followed by reviewing objective evidence. Because the term cloud is a metaphor for the Internet in cloud computing, the term is defined as a type of Internet-based computing in which various services are delivered to an organization's computers and devices via the Internet. Cloud computing holds a lot of promise for IT applications; however, there are still some issues that need to be addressed for individuals and businesses to store data and deploy applications in the Cloud computing environment. One of the most major impediments to its adoption is data security.

1.2 DEDUPLICATION

Data 'chunks' (also known as 'byte patterns') are distinct, contiguous blocks of data that must be compared throughout the deduplication process. During the analysis process, these chunks are discovered and saved, and they are compared to other chunks in the data. When a match is found, the superfluous chunk is replaced by a short reference to the stored chunk. The amount of data that must be saved or communicated can be considerably decreased because the same byte pattern may occur dozens, hundreds, or even thousands of times (the match frequency is depending on the chunk size). Data compression algorithms such as LZ77 and LZ78 are not the same as deduplication. While compression algorithms identify redundant data within individual files and more efficiently encode it, the goal of deduplication is to examine large volumes of data and identify large sections of data that are identical, such as entire files or large sections of files, and replace them with a shared copy.

1.3 STRONG PRIVACY PROTECTION

Cloud computing provides enterprises with the opportunity to simply connect to the cloud and use the available resources on a Pay-Per-Use basis, avoiding the company's capital investment on supplemental on-premise infrastructure resources. It quickly scales up and down rendering to meet business needs. Cloud client, services, application platform, storage, and infrastructure-measured services are all part of it. Thus, cloud computing is a highly automated utility-based paradigm shift that consists of an efficient and optimised framework that includes virtual desktops, servers, and allocates services for computer networks over the internet, as well as software applications and platforms for easy and agile data management deployment.

1.3 DATA SECURITY Throughout its lifecycle, data security refers to the process of securing data from illegal access and data corruption. Data encryption, hashing, tokenization, and key management are all data security strategies that safeguard data across all applications and platforms.

1.4 CLOUD STORAGE Cloud storage is a computer data storage strategy in which digital data is kept in logical pools that are referred to as "the cloud." A hosting business often owns and manages the physical environment, which consists of several servers (occasionally in different countries). These cloud storage providers are in charge of keeping the data safe and accessible, as well as the physical environment secure and operational. To store user, organisation, or application data, people and businesses buy or lease storage capacity from suppliers.

II SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

These approaches can be divided into four categories: basic ciphertext access control, hierarchical access control, completely homomorphic encryption access control, and attribute-based encryption access control (ABE). All of these suggestions are intended for use in a non-mobile cloud context.

• Tysowski et al. looked at a specific cloud computing environment in which data is accessed by resource-constrained mobile devices and proposed novel ABE modifications that shifted the higher computational overhead of cryptographic operations to the cloud provider while lowering the total communication cost for the mobile user.

2.2 DISADVANTAGE

• Many data owners are concerned about the privacy of personal sensitive data.

• The CSP's cutting-edge privilege management/access control techniques are either insufficient or inconvenient to use.

• They are unable to address all of the needs of data owners.

• They use up a lot of storage and processing power, which isn't available on mobile devices.

• Current solutions do not adequately address the issue of user privilege changes. A large revocation cost could come from such an operation. This also does not apply to mobile devices. Clearly, there is no adequate answer to the problem of secure data sharing in the mobile cloud.

1.5 PROPOSED SYSTEM

• We focus on a novel aspect of being resistant to brute-force dictionary assaults on cloud storage audits in this project.

• We present a deduplication-based cloud storage auditing method with robust privacy protection, in which the privacy of a user's le is not exposed to the cloud or other parties when the user's le is predictable or comes from a small space.

• We suggest a new method for generating the le index for duplicate check, as well as a new strategy for generating the key for le encryption.

• In addition, to produce data authenticators, validate cloud data integrity, and retrieve the le from the cloud, the user simply needs to do lightweight computing.

•The suggested approach delivers desirable security and efficiency, as demonstrated by the security proof and performance evaluation.

• We suggest a new method for generating the le index for duplicate check, as well as a new strategy for generating the key for le encryption.

• In addition, to produce data authenticators, validate cloud data integrity, and retrieve the le from the cloud, the user simply needs to do lightweight computing.

2.4 ADVANTAGE

• Experiments suggest that LDSS can significantly minimise client overhead while introducing only a minor additional cost on the server side. Implementing a realistic data sharing security method for mobile devices requires such an approach.

•The results also reveal that LDSS performs better over ciphertext than existing ABE-based access control systems**.**

• Multiple revocation operations are consolidated into a single operation, lowering overall overhead.

• When compared to data files, the storage overhead required for access control in LDSS is quite tiny.

1.6 SYSTEM ARCHITECTURE

A system architecture is a conceptual model that defines a system's structure, behaviour, and other views. A system architecture can be made up of system components and subsystems that will collaborate to construct the overall system.

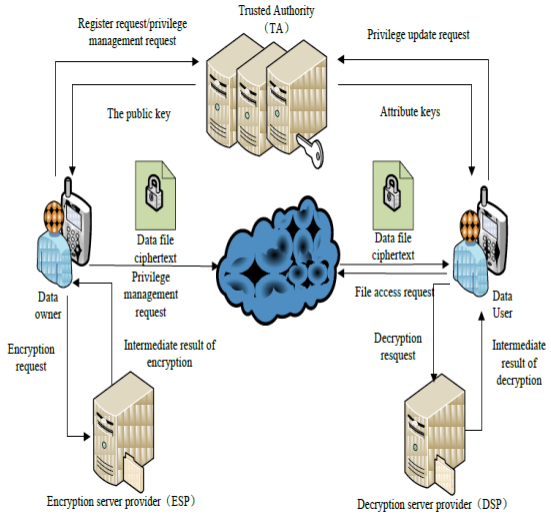


Fig 2.4 System Architecture

III MODULE IMPLEMENTATION

3.1 Admin Module

The admin is the principal person in this module who has the ability to monitor all of the activities that are processed by the cloud user. Upload, download, and update are some of the features available.

3.2 Data User Module

• The data user in this module is the individual who wants to safely store data in the cloud.

• He will keep the data on the cloud server, which he will then download whenever he needs access to it.

• He has the ability to upload, download, and update files.

1.7 PRIVATE CLOUD MODULE

• At the time of registration, he will grant or refuse authorization to the cloud user in this module.

• He also has the capability of granting access privileges to people who request that their uploaded files be uploaded, downloaded, or updated.

1.8 DEDUPLICATION

• Data will not be duplicated in this module for storage in the cloud data base.

ISSN (Online) : 2455 - 0523

• If a file that has previously been stored on the server by a user such as X' cannot be stored by any other user, including X'.

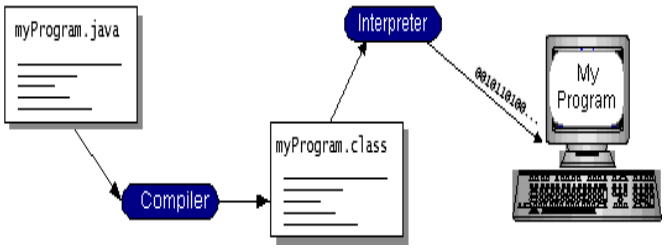
1.9 PRIVACY PRESERVING MODULE

• We may give privacy protection for the uploaded data in the cloud server by encrypting the data stored on the cloud server with this module.

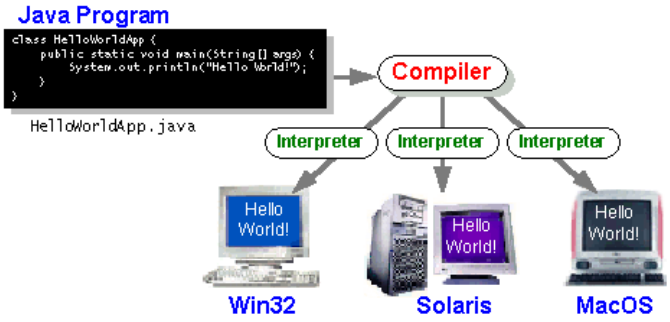
IV SOFTWARE ENVIRONMENT

4.1 The Java Programming Language

Java byte codes are platform-independent codes that are interpreted by the Java platform's interpreter. Each Java byte code instruction is parsed and executed by the interpreter on the computer. Compilation takes place only once; interpretation takes place each time the programme is run. This is demonstrated in the diagram below.



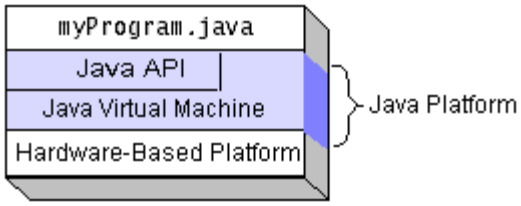
Java byte codes can be thought of as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter is an implementation of the Java Virtual Machine, whether it's a programming tool or a Web browser that can run applets. Java byte codes make it feasible to "write once, run anywhere." Any platform with a Java compiler can compile your application into byte codes. The byte codes can then be executed on any Java VM implementation. That means that a Java programme created in the Java programming language can operate on Windows 2000, a Solaris workstation, or an iMac as long as the computer has a Java virtual machine.



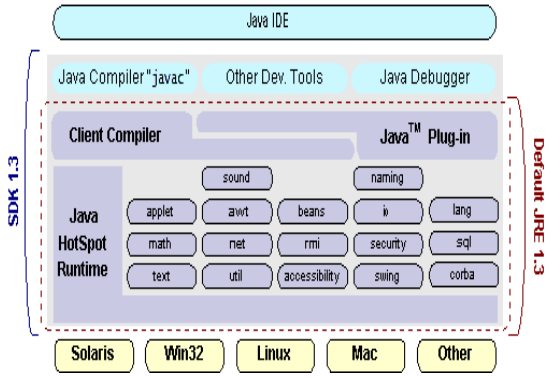
4.2 The Java Platform

The hardware or software environment in which a programme runs is referred to as a platform. Some of the most popular platforms, such as Windows 2000, Linux, Solaris, and MacOS, have already been mentioned. The majority of platforms are made up of a combination of operating system and hardware. In contrast to most other platforms, Java is a software-only platform that works on top of other hardware-based platforms.

The Java API is a comprehensive collection of ready-to-use software components that include graphical user interface (GUI) widgets and other essential features. The Java API is organised into packages, which are collections of related classes and interfaces.



APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, voice, animation, and more are available on the Java platform. The contents of the Java 2 SDK are depicted in the diagram below.



• Get started quickly: Despite being a strong object-oriented language, Java is simple to learn, especially for programmers who are already familiar with C or C++.

• Write fewer lines of code: A programme written in the Java programming language can be four times smaller than a programme written in C++, according to programme metrics (class counts, method counts, and so on).

4.3 ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for database systems providers and application developers. Programmers had to utilise proprietary languages for each database they wished to connect to before ODBC became the de facto standard for Windows programmes to interface with database systems. From a coding standpoint, ODBC has rendered the database system essentially unimportant, which is exactly what it should be. When business demands change, application developers have far more essential things to worry about than the syntax required to transfer their programme from one database to another.

Windows 95 does not install the ODBC system files on your computer. Rather, they are installed when a separate database application, such as SQL Server Client or Visual Basic 4.0, is installed. A file called ODBCINST.DLL is used when the ODBC icon is installed in Control Panel. A stand-alone tool called ODBCADM.EXE can also be used to manage your ODBC data sources. This programme comes in two versions: 16-bit and 32-bit, each with its own list of ODBC data sources.

4.4 JDBC

Sun Microsystems created Java Database Connectivity, or JDBC, in order to create an independent database standard API for Java. JDBC is a general SQL database access technology that provides a uniform interface to a wide range of relational database management systems (RDBMSs). The usage of "plug-in" database connectivity modules, or drivers, allows for a consistent interface. If a database vendor wants JDBC support, the driver must be provided for each platform that the database and Java operate on.

4.5 JDBC Goals

Few software products are created without a set of objectives in mind. JDBC is one of the APIs that inspired the development of the API due to its numerous aims. These objectives, together with early reviewer feedback, have resulted in the JDBC class library being a robust basis for developing Java database applications. The objectives specified for JDBC are crucial.

4.6 SQL Level API

The designers believed that the most important task was to create a Java SQL interface. Although it is not the lowest database interface level imaginable, it is low enough to allow for the creation of higher-level tools and APIs. It is, on the other hand, at a high enough level for application programmers to securely use it. In order to achieve this goal, future tool manufacturers will be able to "create" JDBC code and hide many of JDBC's complexity from the end user.

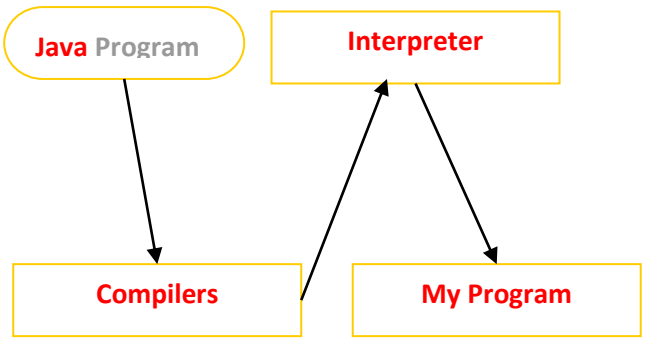
4.6 SQL Conformance

As you migrate from one database vendor to the next, the SQL syntax changes. JDBC will allow any query statement to be given through it to the underlying database driver in order to accommodate a wide range of vendors. This enables

ISSN (Online) : 2455 - 0523

the connectivity module to handle non-standard capabilities in a user-friendly manner.

Compilation takes place only once; interpretation takes place each time the programme is run. The diagram shows how this works.

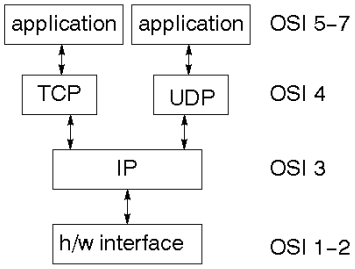


Java byte codes make it feasible to "write once, run anywhere." On my platform, which contains a Java compiler, you can compile your Java application into byte codes. The byte codes can then be run on any Java VM implementation. The same Java programme, for example, can run on Windows NT, Solaris, and Macintosh.

NETWORKING

TCP/IP stack

The TCP/IP stack is shorter than the OSI one:



IP datagram's

The IP layer provides a delivery method that is connectionless and unreliable. It examines each datagram separately from the others. Any datagram association must be provided by the higher layers. A checksum is provided by the IP layer, which includes its own header. The source and destination addresses are included in the header. The IP layer is in charge of Internet routing. It's also in charge of disassembling big datagrams for transmission and reassembling them at the other end.

UDP

UDP is likewise unstable and connectionless. It adds a checksum for the datagram's contents as well as port numbers to IP. These are utilised to create a client/server model, which will be discussed later.

TCP

Above IP, TCP provides logic to provide a dependable connection-oriented protocol. It creates a virtual circuit through which two processes can communicate.

Internet addresses

You must be able to locate a service in order to use it. For machines to be located on the Internet, an address scheme is used. The address is a 32-bit integer that is used to determine the IP address. This encodes a network ID as well as additional addressing. According to the size of the network address, the network ID is divided into several groups.

Network address

The network address in Class A is 8 bits long, leaving 24 bits for other addressing. Class B employs a 16-bit network addressing scheme. Class C uses a 24-bit network addressing scheme, while class D use all 32 bits.

Port addresses

A port identifies a service that exists on a host. This is an integer of 16 bits. To send a message to a server, type it into the port for that service on the server's host. This isn't the same as location transparency! Several of these ports are "famous."

Sockets

A socket is a data structure that the system keeps track of in order to handle network connections. The call socket is used to create a socket. It returns an integer that functions similarly to a file descriptor. In fact, this handle can be utilized with the Read File and Write File Methods in Windows.

```
#include <sys/types.h>

#include <sys/socket.h>

int socket(int family, int type, int protocol);
```

For IP communications, the "family" will be AF INET, the protocol will be zero, and the type will depend on whether TCP or UDP is used. When two processes want to communicate via a network, they each build a socket. These look like the ends of a pipe, but the pipe itself does not yet exist.

V CONCLUSION

We investigate how to address the problem of user privacy leaking in cloud storage audits with deduplication when brute-force dictionary assaults are launched in this project. We create a deduplication-based cloud storage auditing technique that provides robust privacy protection. The user's privacy can be properly protected from the cloud and other parties under the proposed approach. The user reduces the amount of time it takes to generate data authenticators and check data integrity. The proposed scheme is secure, according to the security proof. Experiments are also used to give thorough comparisons between our proposed method and other existing schemes. The proposed technique achieves improved storage efficiency and is more efficient, according to experimental data.

REFERENCES

[1]. The Gnu Multiple Precision Arithmetic Library (GMP). Accessed: Oct. 2019. [Online]. Available: http://gmplib.org/

[2]. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, ``Provable data possession at untrusted stores," in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), 2007, pp. 598609.

[3]. M. Bellare, S. Keelveedhi, and T. Ristenpart, ``Message-locked encryption and secure deduplication," in Proc. Annu. Int. Conf. Theory Appl. Crypto- graph. Techn. Berlin Germany: Springer, 2013, pp. 296312.

[4]. H. Cui, R. H. Deng, Y. Li, and G. Wu, ``Attribute-based storage supporting secure deduplication of encrypted data in cloud," IEEE Trans. Big Data, vol. 5, no. 3, pp. 330342, Sep. 2019.

[5]. R. Ding, H. Zhong, J. Ma, X. Liu, and J. Ning, ``Lightweight privacy- preserving identity-based veritable IoT-based health storage system," IEEE Internet Things J., vol. 6, no. 5, pp. 83938405, Oct. 2019.

[6]. J. R. Douceur, A. Adya, W. J. Bolosky, P. Simon, and M. Theimer, ``Reclaiming space from duplicate files in a server less distributed file system," in Proc. 22nd Int. Conf. Distrib. Comput. Syst., Jul. 2002, pp. 617624.

[7]. Y. Fan, X. Lin, G. Tan, Y. Zhang, W. Dong, and J. Lei, ``One secure data integrity verification scheme for cloud storage," Future Gener. Comput. Syst., vol. 96, pp. 376385, Jul. 2019.

[8]. J. Gantz and D. Reinsel. (2012). The Digital Universe Decade Are You Ready (2010). [Online]. Available: http://www. emc.com/collateral/analyst-reports/idcdigital-universe-are-you-ready.pdf

[9]. [9] X. Ge, J. Yu, H. Zhang, C. Hu, Z. Li, Z. Qin, and R. Hao, ``Towards achieving keyword search over dynamic encrypted cloud data with symmetric-key based verification," IEEE Trans. Dependable Secure Com- put., to be published.

[10]. W. Guo, H. Zhang, S. Qin, F. Gao, Z. Jin, W. Li, and Q. Wen, ``Out- sourced dynamic provable data possession with batch update for secure cloud storage," Future Gener. Comput. Syst., vol. 95, pp. 309322, Jun. 2019.

[11]. S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, ``Proofs of own- ership in remote storage systems," in Proc. 18th ACM Conf. Comput. Commun. Secur. (CCS), New York, NY, USA, 2011, pp. 491500.

[12]. D. Harnik, B. Pinkas, and A. Shulman-Peleg, ``Side channels in cloud services: Deduplication in cloud storage," IEEE Secur. Privacy Mag., vol. 8, no. 6, pp. 4047, Nov. 2010.

[13]. K. He, J. Chen, Q. Yuan, S. Ji, D. He, and R. Du, ``Dynamic group- oriented provable data possession in the cloud," IEEE Trans. Dependable Secure Comput., to be published.

[14]. H. Hou, J. Yu, and R. Hao, ``Cloud storage auditing with deduplication supporting different security levels according to data popularity," J. Netw. Comput. Appl., vol. 134, pp. 2639, May 2019.

[15]. A. Juels and B. S. Kaliski, ``Pors: Proofs of retrievability for large files," in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), 2007, pp. 584597.

BIOGRAPHIES (Optional not mandatory)

MADHAVARAJ SS is an Undergraduate student, department of computer science and engineering in paavai college of engineering.

ARUN KUMAR NS is an Undergraduate student, department of computer science and engineering in paavai college of engineering.

SRINIVASAN K is an Assistant Professor, Department of Computer Science and Engineering, Paavai College of Engineering.