FINE-GRAINED TWO-FACTOR ACCESS CONTROL FOR WEB-BASED CLOUD COMPUTING SERVICES

A Project Report submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Nitin Kundu(121810306003),
Shiva Shankar Varaprasad(121810306021)
K. Rohit Sai(121810306033)
K. Prithvi Raj(121810306046)
K Balaji(121810306062)

Under the esteemed guidance of

Prof. Srinivas Prasad

PhD

DEPARTMENT OFCOMPUTER SCIENCE & ENGINEERING

GITAM

(Deemed to be University)

VISAKHAPATNAM

OCTOBER 2021

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM INSTITUTE OF TECHNOLOGY

GITAM

(Deemed to be University)

DECLARATION

We hereby declare that the project report "Fine-grained two-factor authentication (2FA) access control system for web-based cloud computing services" is an original work completed in the Department of Computer Science and Engineering, GITAM Institute of Technology, GITAM (Deemed to be University) and submitted as part of the prerequisites for the award of a bachelor's degree B.Tech. in Computer Science and Engineering. No other institution or university has accepted the work for the purpose of awarding a degree or diploma.

Registration No(s).	Name(s)	Signature(s)
121810306003	Nitin Kundu	
121810306021	Shiva Shankar Varaprasad	
121810306033	K. Rohith Sai	
121810306046	K. Prithvi Raj	
121810306062	K Balaji	

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM INSTITUTE OF TECHNOLOGY

GITAM

(Deemed to be University)

CERTIFICATE

This is to certify that the project report entitled **"FINE GRAINED TWO FACTOR ACCESS CONTROL FOR WEB BASED CLOUD COMPUTING SERVICES"** is a bonafide record of work carried out by **Nitin Kundu(121810306003) , Prithvi Raj(121810306046), Rohith Sai K(121810306033), Siva Sankar Vara Prasad(121810306021), K. Balaji (121810306062)** submitted in partial fulfillment of requirement for the award of degree of Bachelors of Technology in Computer Science and Engineering.

Project Guide

SRINIVAS PRASAD

PROFESSOR

Head of the Department

R-Sireesha

Dr. R.SIREESHA

PROFESSOR

Head of the Dept.
Department of Computer Science & Engineering
GITAM (Institute of Technology)
Gandhi Institute of Technology and Management (GITAM)
(Deemed to be University)
Visakhapatnam-530 045

TABLE OF CONTENTS

ABSTRACT

A new fine-grained two-factor authentication (2FA) access control system for web-based cloud computing services is introduced in this research. An attribute-based access control mechanism is implemented in our proposed 2FA access control system, which requires a user secret key as well as a small security gadget Because a person can't access the system until they have both, the approach can improve system security, particularly in situations where multiple users share the same computer for web-based cloud services. Furthermore, attribute-based control in the system allows the cloud server to limit access to users who share the same set of traits while maintaining personal privacy, i.e., the cloud server only knows that the user meets certain criteria.

INTRODUCTION

CLOUD computing is regarded as a prospective com-puting paradigm in which resource is supplied as service over the Internet. It has met the increasing needs of computing resources and storage resources for some enterprises due to its advantages of economy, scalability, and accessibility. Recently, several cloud stor-age services such as Microsoft Azure and Google App Engine were built and can supply users with scalable and dynamic storage. With the increasing of sensitive data outsourced to cloud, cloud storage services are facing many challenges including data security and data access control. To solve those problems, attribute-based encryption (ABE) schemes [1-3] have been applied to cloud storage services. Sahai and Waters [1] first proposed ABE scheme named fuzzy identity-based encryption which is derived from identity-based encryption (IBE) [4]. As a new proposed cryptographic primitive, ABE scheme not only has the advantage of IBE scheme, but also provides the character-istic of "one-to-m any" encryption. Presently, ABE mainly includes two categories called ciphertext -policy ABE (CP-ABE) [2] and key-policy ABE (KP-ABE) [3]. In CP-ABE, ciphertexts are associated with access policies and user's private keys are associated with attribute sets. A user can decrypt the ciphertext if his attributes satisfy the access policy embedded in the ciphertext. It is contrary in KP-ABE. CP-ABE is more suitable for the outsourcing data architecture than KP-ABE because the access policy is defined by the data owners. In this article, we pr esent an efficient CP-ABE with user revocation ability.

Problem Identification & Objectives

EXISTING SYSTEM:

Mediated cryptography was first developed as a means of allowing public keys to be revoked immediately. The main principle behind mediated cryptography is that each transaction is handled by an online mediator. Because it controls security capabilities, this online mediator is referred to as a SEM (SEcurity Mediator). If the SEM refuses to comply, no transactions using the public key will be available.
Key-insulated security was designed to store long-term keys in a physically secure but computationally constrained device. Users save short-term secret keys on a powerful but insecure device that performs cryptographic computations. Short-term secrets are renewed at regular intervals by interaction between the user and the base, but the public key remains unchanged.

DISADVANTAGES OF EXISTING SYSTEM:

Every time period, all users of a key-insulated cryptosystem must update their keys. The security device is required for the key updating process.
The signing or decryption algorithm no longer requires the device within the same time period once the key has been updated.
Traditional account/password authentication does not protect your privacy. Privacy, on the other hand, is widely accepted as an important characteristic to address in cloud computing systems.
It is normal for multiple persons to share a computer. Hackers may find it simple to install spyware and learn the login password from the web browser.
The attacker takes on the role of the cloud server and attempts to identify the person with whom it is conversing.

PROPOSED SYSTEM

Using a lightweight security device, we propose a fine-grained two-factor access control protocol for web-based cloud computing services in this work. The following are the characteristics of the device: (1) It can perform several lightweight algorithms, such as hashing and exponentiation; and (2) it is tamper resistant, in the sense that no one is supposed to be able to get into it and obtain the secret information stored inside.
In this research, we use a lightweight security device to propose a fine-grained two-factor access control mechanism for web-based cloud computing services. The following are the characteristics of the device. It can do some simple algorithms, such as hashing and exponentiation, and it is tamper-resistant, meaning that no one should be able to access the secret data stored inside.
With our protocol provides a 2FA security. First the user secret key (which is usually stored inside the computer) is required. In addition, the security device should be also connected to the computer (e.g. through USB) in order to authenticate the user for accessing the cloud. The user can be granted access only if he has both items.
Furthermore, the user cannot use his secret key with another device belonging to others for the access. Our protocol supports fine-grained attribute-based access which provides a great flexibility for the system to set different access policies according to different scenarios. At the same time, the privacy of the user is also preserved. The cloud system only knows that the user possesses some required attribute, but not the real identity of the user. To show the practicality of our system, we simulate the prototype of the protocol.

ADVANTAGES OF PROPOSED SYSTEM:

Our protocol allows fine-grained attribute-based access, which gives the system a lot of flexibility in terms of creating alternative access policies for different scenarios. At the very same time, the user's privacy is safeguarded. The cloud system just know that the user has a certain attribute, but not his or her true identity.
We simulate the protocol prototype to demonstrate the viability of our approach.

Tamper-resistance. Once the security device is initialised, the content contained inside it is neither accessible nor editable. Furthermore, it will always adhere to the algorithm's specifications.
It has the ability to evaluate hash functions. It can also generate random numbers and compute the exponentiations of a cyclic group defined over a range of values.

SYSTEM ARCHITECTURE:

SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

System : Pentium Dual Core.

Hard Disk : 120 GB.

Monitor : 15'' LED

Input Devices : Keyboard, Mouse

Ram : 1GB

SOFTWARE REQUIREMENTS:

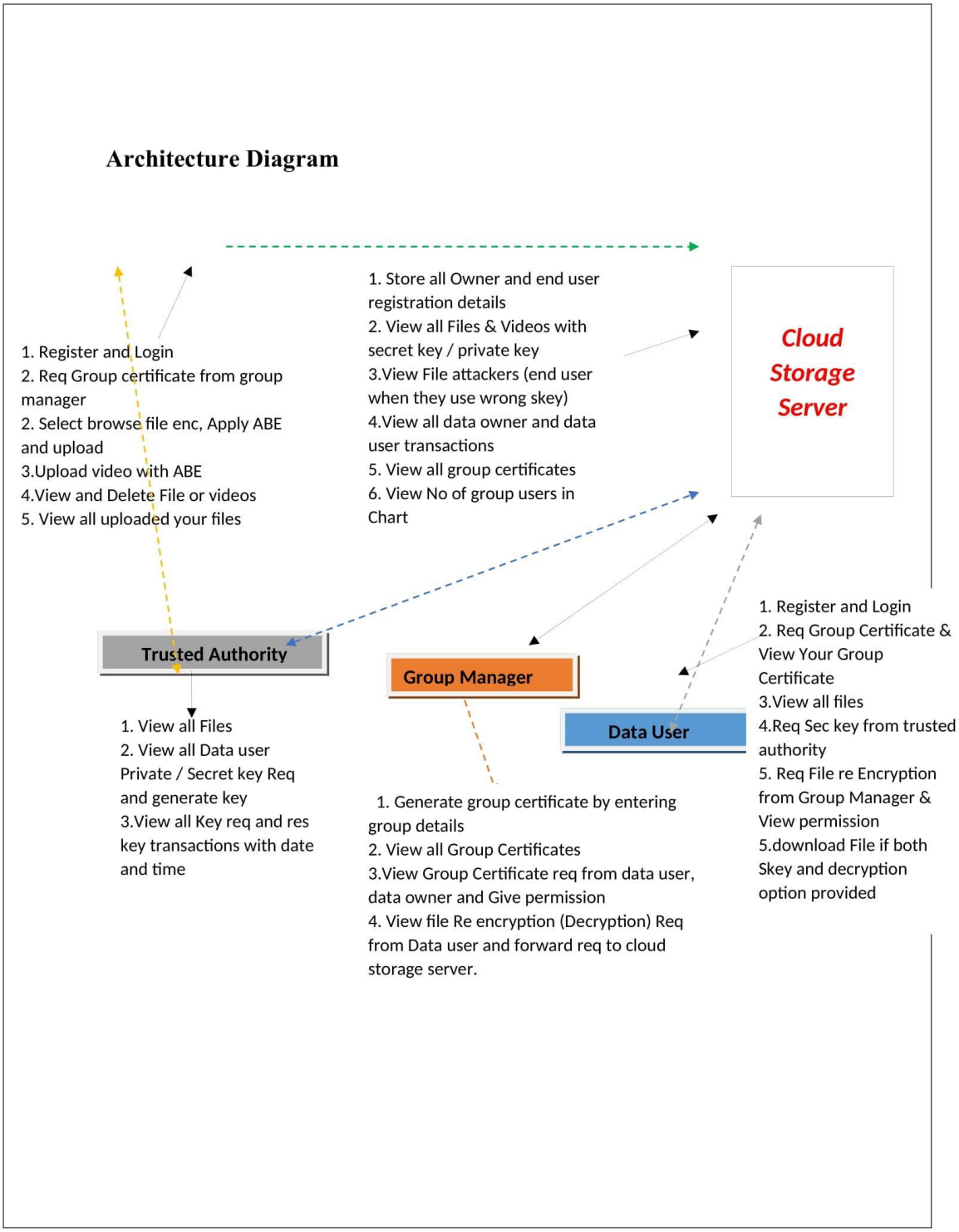
Operating system : Windows 7.

Coding Language : JAVA/J2EE

Tool : Netbeans 7.2.1

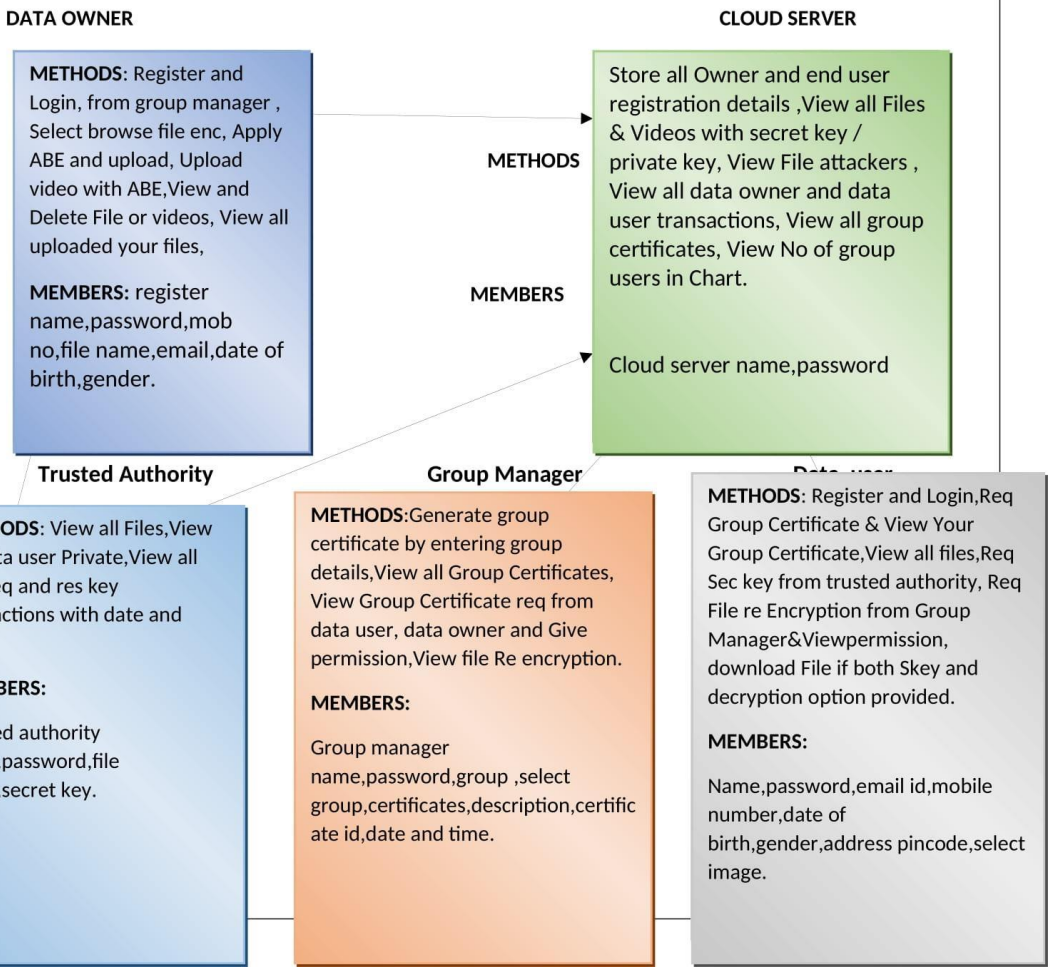
Database : MYSQL

Architecture Diagram



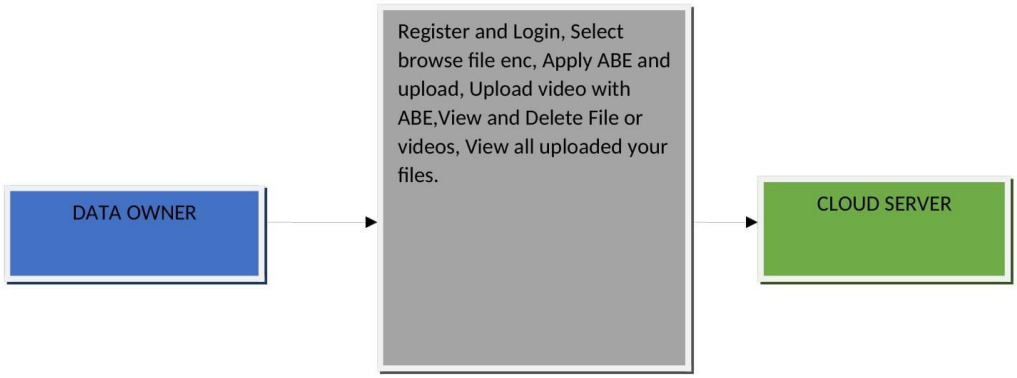
Acronym	Description
TA	trusted authority
KG-CSP	key generation cloud server provider
D-CSP	decryption cloud server provider
S-CSP	storage cloud server provider
DO	data owner
DU	data user

Class Diagram

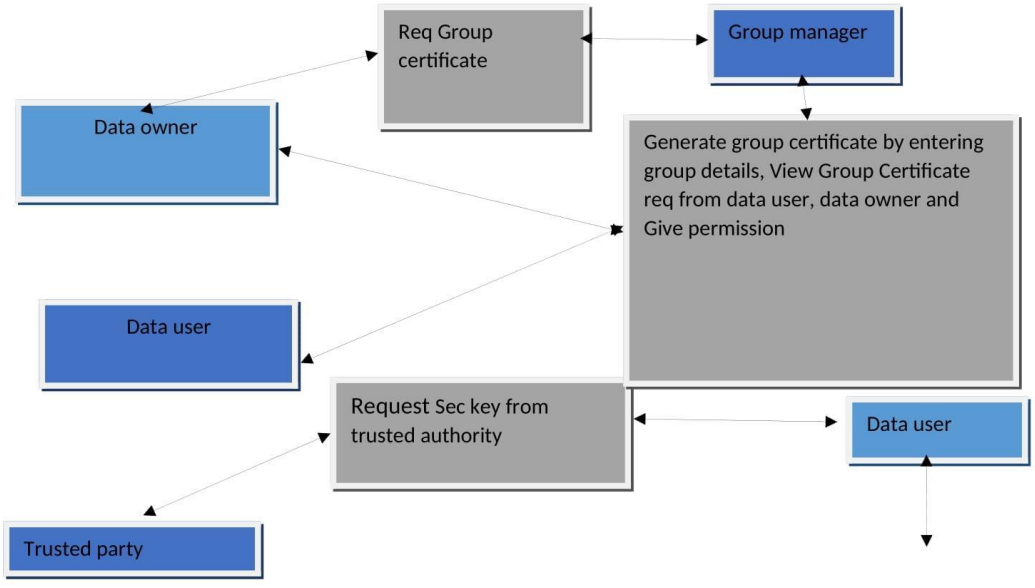


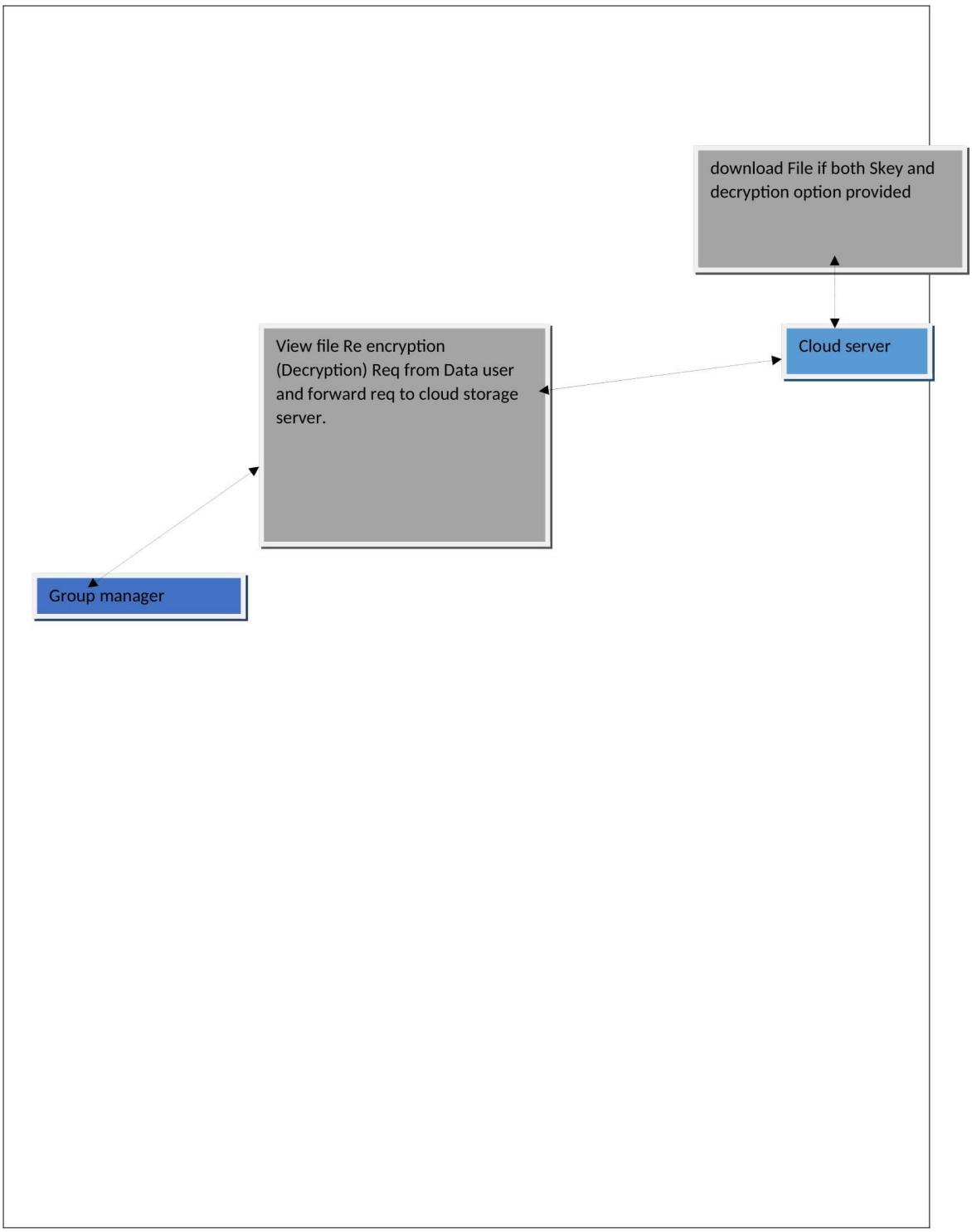
DATAFLOW

Level -0

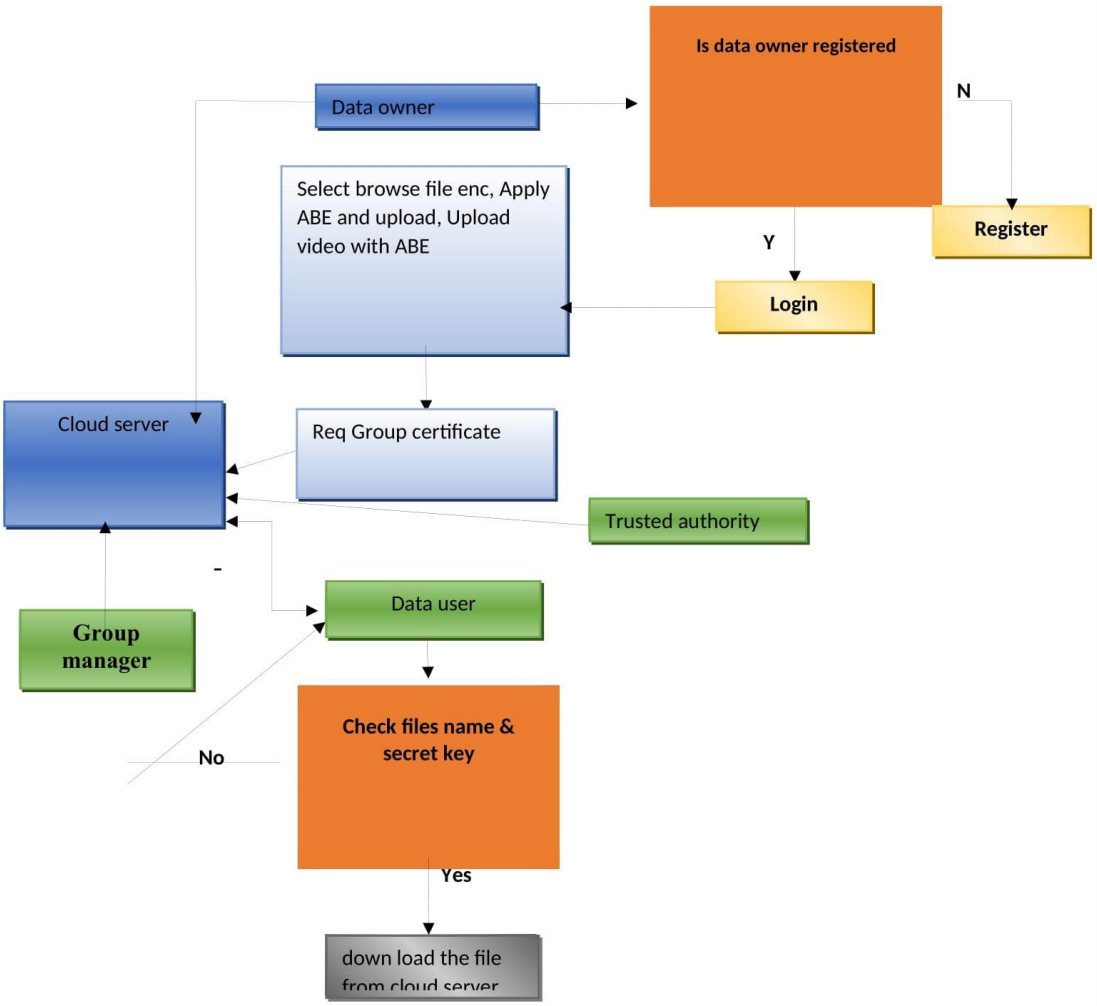


LEVEL 1

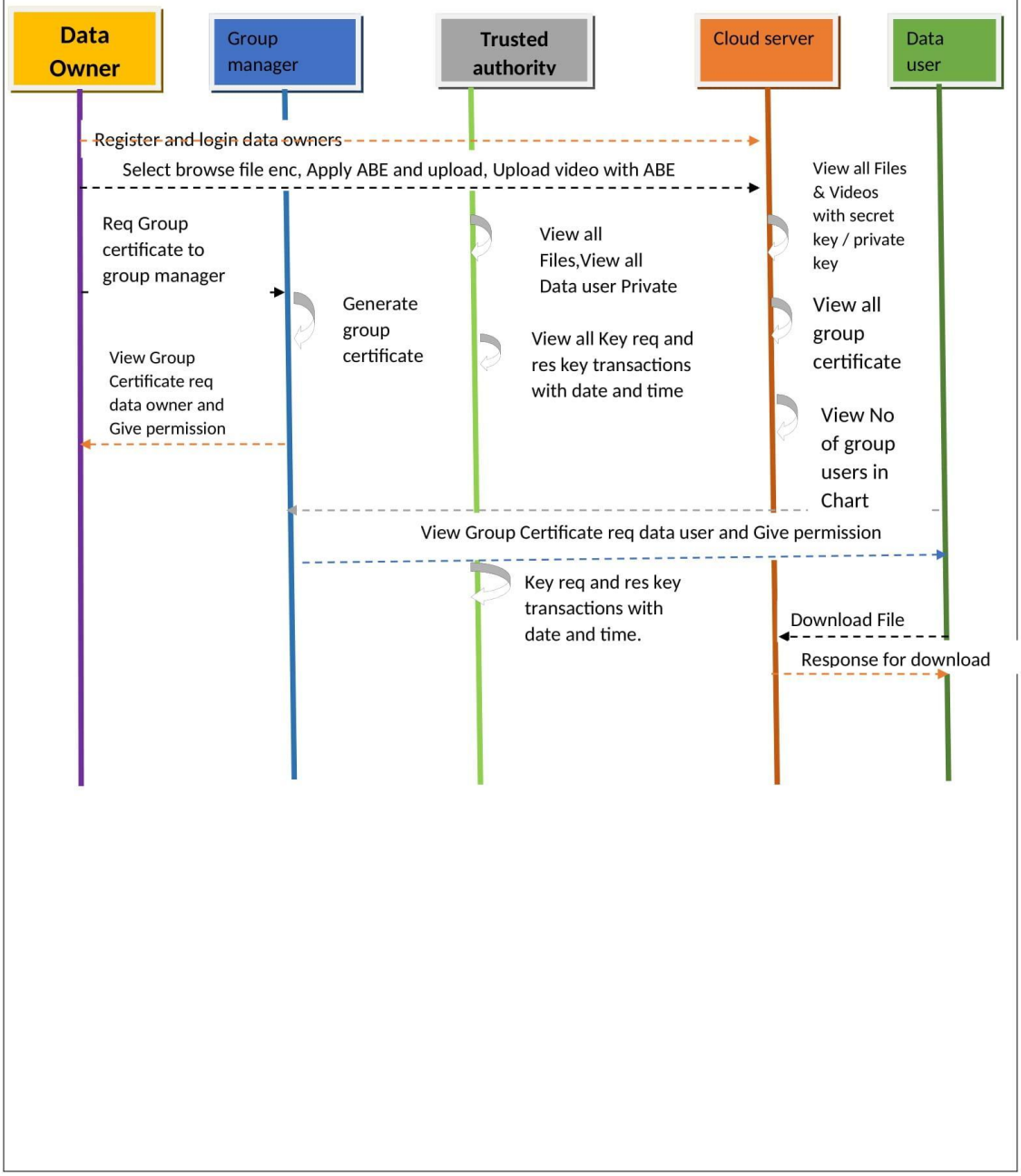



download File if both Skey and decryption option provided
Cloud server
View file Re encryption (Decryption) Req from Data user and forward req to cloud storage server.
Group manager

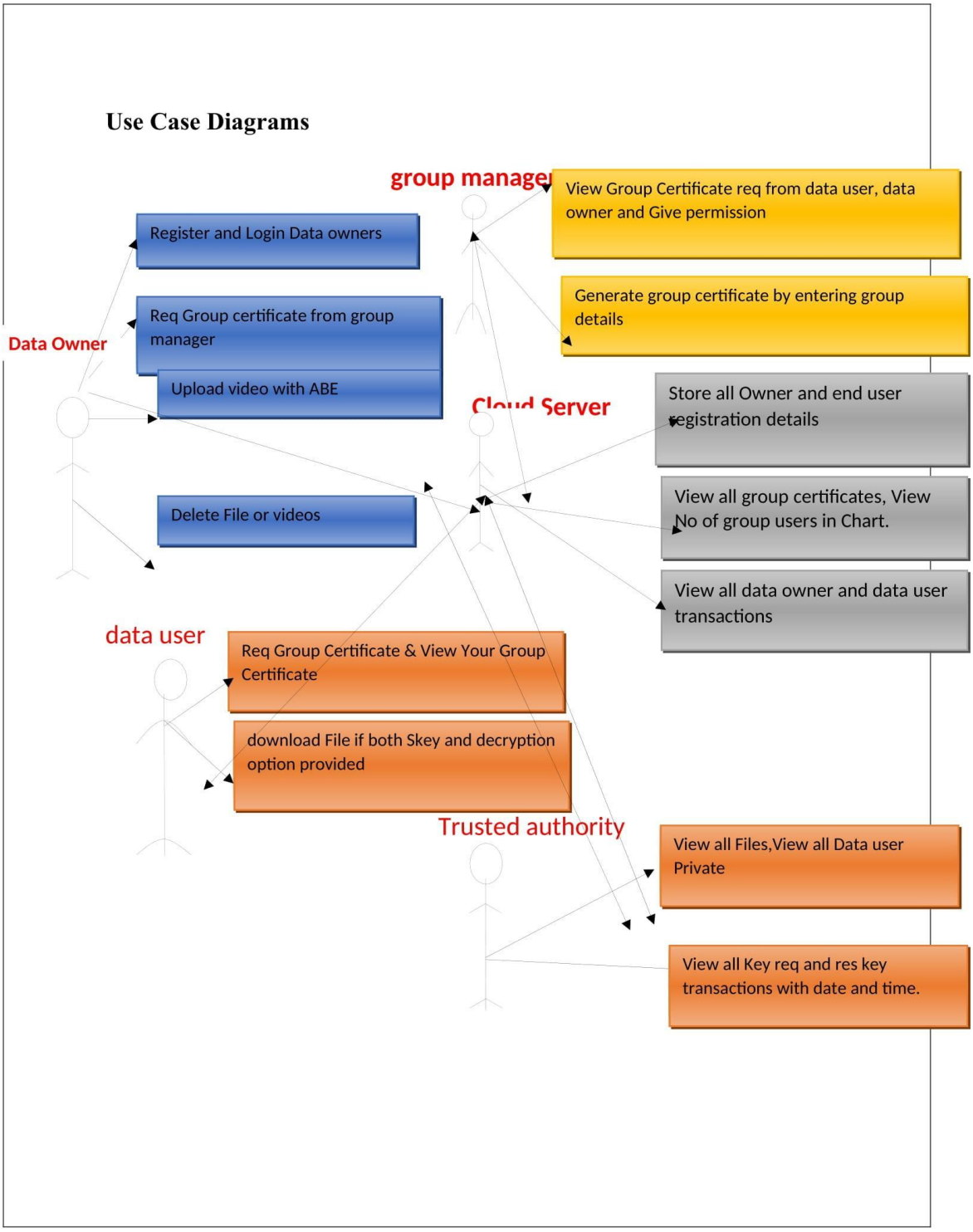
FLOWCHART


Is data owner registered
Data owner
N
Register
Select browse file enc, Apply ABE and upload, Upload video with ABE
Y
Login
Cloud server
Req Group certificate
Trusted authority
Data user
Group manager
Check files name & secret key
No
Yes
down load the file from cloud server

Sequence Diagram



Use Case Diagrams



OVERVIEW OF TECHNOLOGIES

J2EE Software Environment

Client Server

Overview of the Client/Server:

Client Server is one of the many topics in the realm of computers that has generated more heat than light, as well as more hype than reality. With its specialised conferences and periodicals, this technology has gained a certain critical mass of attention. Client Servers are the primary future market for major computer companies such as IBM and DEC. According to a poll conducted by DBMS magazine, 76 percent of its readers were interested in the client-server solution. The client-server development tools market grew from $200 million in 1992 to over $1.2 billion in 1996.

Client-server implementations are difficult, yet the underlying notion is straightforward and effective. A client is a programme that runs on local resources but can access a remote server's database and connect to its services. MIDDLEWARE is the term for the software that mediates the client-server relationship.

The typical client is a PC or a Workstation that is connected through a network to a more powerful PC, Workstation, Midrange, or Main Frames server that can handle several client requests. However, depending on the settings, the server could also behave as a client. In order to complete the original client request, a server may need to connect to another server.

The primary client server concept is that the client, as a user, is essentially shielded from the physical location and formats of the data their programme requires. A client input from or report can transparently access and manipulate both local and remote databases on one or more servers with the right middleware. The client server also enables multi-vendor database access with heterogeneous table joins, which is a nice feature.

What is the difference between a client and a server?

Client server and file server systems are the two most common systems. It's critical to understand the difference between client servers and file server systems. Both enable data access across a shared network, but the comparison ends there! The file server is nothing more than a remote disc drive that LAN programmes can access file by file. The client server provides complete relational database capabilities, including SQL-Access, record modification,

Insert, and Delete, as well as full relational integrity backup and restore performance for high volume transactions. The client server middleware provides a flexible interface between the client and the server, allowing the client and server to determine who performs what, when, and to whom.

Why is Client Server used?

Client-server computing evolved to address a challenge that has existed since the dawn of computers: how to best distribute computing, data creation, and data storage resources to achieve efficient, cost-effective departmental and enterprise-wide data processing. During the mainframe era, options were restricted. Both the CPU and the DATA were placed in a single unit (cards, tapes, drums and later disks). Initially, access to these resources was limited to batch The corporation was administered by a powerful central information service department. The rest of the corporation's responsibility was confined to requesting new or more frequent reports, as well as supplying handwritten forms from which the central data banks were constructed and updated. As a result, the early client-server solutions are best described as "SLAVE-MASTER."
The picture has changed as a result of time-sharing. Subject to access rights, a remote terminal could examine and potentially edit central data. Online users may construct adhoc queries and produce local reports without adding to the MIS applications development backlog as the central data banks matured into complex relational databases with non-programmer query languages. Remote access was provided via dumb terminals, while the client server remained under the control of the SlaveMaster.

User Interface Design (UID) is a term that refers to the design of the
To achieve the Distributed Concept, the entire user interface is expected to be designed in a browser-specific environment with a touch of Intranet-Based Architecture.
The browser-specific components are created using HTML standards, and the dynamism of the system is achieved by focusing on Java Server Pages constructs.
Tiers of communication and database connectivity

The communication architecture is created by focusing on Servlet and Enterprise Java Bean Standards. The JavaDataBaseConnectivity class is used to establish database connectivity.
The standards of three-tire design are given special attention in order to maintain higher cohesiveness and limited coupling for operational effectiveness.
Characteristics of the Language Used
For my project, I chose Java as the programming language.
Discussing Java
The language was originally known as "oak," but in 1995 it was renamed "Java."
The fundamental reason for developing this language was the requirement for a platform-independent (i.e., architecture-neutral) programming language that could be used to generate software for usage in a variety of consumer electronic devices.
Finally, Java is to Internet programming where C was to system programming.

The language was originally known as "oak," but in 1995 it was renamed "Java."

The fundamental reason for developing this language was the requirement for a platform-independent (i.e., architecture-neutral) programming language that could be used to generate software for usage in a variety of consumer electronic devices.

Java's Importance on the Internet
On the Internet, Java has had a significant impact. This is due to the fact that Java increases the number of items that can freely move around in cyberspace. Two types of objects are transmitted between the Server and the Personal Computer in a network. Passive information and dynamic active programmes are the two types. In the realms of security and probability, dynamic, self-executing systems cause major issues. Java, on the other hand, addresses these concerns and, as a result, has ushered in an exciting new type of application known as the Applet.
Java may be used to make two different kinds of programmes.
Java's Importance on the Internet

On the Internet, Java has had a significant impact. This is due to the fact that Java increases the number of items that can freely move around in cyberspace. Two types of objects are transmitted between the Server and the Personal Computer in a network. Passive information and dynamic active programmes are the two types. In the realms of security and probability, dynamic, self-executing systems cause major issues. Java, on the other hand, addresses these concerns and, as a result, has ushered in an exciting new type of application known as the Applet.
Java may be used to make two different kinds of programmes.

Features Of Java

Java Security Features

You run the chance of contracting a virus every time you download a "regular" programme. Prior to Java, most users did not routinely download executable applications, and those that did did so screened them for viruses before running them. The majority of people are still concerned about viruses invading their computers. In addition, there is another form of hazardous programme that must be avoided. Credit card numbers, bank account balances, and passwords are all examples of private information that can be gathered by this type of application. Both of these concerns are addressed by Java, which acts as a "firewall" between a network application and your machine.
You can download Java applets without risk of virus infection or malicious intent if you use a Java-compatible Web browser.

Portability

Some method of generating portable executable code is required for programmes to be dynamically downloaded to all the various sorts of platforms linked to the Internet.
As you'll see, the same technology that aids in security also aids in mobility. Java's answer to these two issues is, in fact, both elegant and efficient.

The code of bytes

The fact that the Java compiler produces Byte code is the key to Java's ability to tackle security and portability issues. Byte code is a set of highly efficient instructions meant to be executed by the Java Virtual Machine, which is the Java

run-time system (JVM). That is, the JVM is a byte code interpreter in its most basic form.
When a Java programme is converted to byte code, it becomes considerably easier to run it in a number of settings. The reason for this is that once a system's run-time package is installed, any Java programme can execute on it.
Despite the fact that Java was created for interpretation, there is nothing in the language that stops byte code from being converted to native code on the fly. Sun has just finished its byte code Just In Time (JIT) compiler. When the JIT compiler is included in the JVM, it compiles byte code into executable code in real time, on-demand. An full Java application cannot be compiled into executable code at the same time. because Java runs a variety of run-time tests that can only be performed at run time During execution, the JIT compiles code as needed.

Virtual Machine, Java (JVM)

There's also the Java virtual machine, which is separate from the language. The Java virtual machine is a crucial component of the Java platform. The virtual machine can run in a web browser or on a computer's operating system. A piece of Java code is verified once it has been put onto a machine. A class loader is run as part of the loading process, and it performs byte code verification to ensure that the code created by the compiler will not destroy the machine on which it is loaded. At the end of the compilation process, byte code verification is performed.

Picture showing the development process of JAVA Program

A diagram depicting the JAVA programme development process.
Java is a computer language that generates and executes byte codes. The first box shows that the Java source code is contained in a Java file that is processed using the javac Java compiler. The byte code is stored in a file called a. class file, which is generated by the Java compiler. The class file is subsequently loaded into the execution environment, which is the Java virtual machine, which

interprets and executes the byte code, either over the network or locally on your workstation.

Architecture in Java

The Java architecture provides a development environment that is portable, robust, and high-performing. Java achieves portability by producing byte codes for the Java Virtual Machine, which is then interpreted by the run-time environment on each platform. Java is a dynamic system that can load code from a machine in the same room or across the globe when it's needed.

Coding compilation

The Java compiler generates machine code (also known as byte code) for a fictitious machine called Java Virtual Machine when you compile the code (JVM). The byte code is designed to be executed by the JVM. The JVM was developed to address the issue of portability. The code is written and compiled for a single machine, but it is interpreted by all machines. Java Virtual Machine is the name of this machine.

Object-Oriented

Object-Oriented Java was not created with the intention of being source-code compatible with any other programming language. This gave the Java team the opportunity to start from scratch. One result was a simple, useful, and pragmatic approach to things. Simple types, such as integers, are preserved as high-performance non-objects in Java's object paradigm, which is simple and straightforward to modify.

Robust

The multi-platform environment of the Web places extraordinary demands on a program, because the program must execute reliably in a variety of systems. The ability to create robust programs was given a high priority in the design of Java. Java is strictly typed language; it checks your code at compile time and run time.

Java virtually eliminates the problems of memory management and deallocation, which is completely automatic. In a well-written Java program, all run time errors can –and should –be managed by your program.

JAVASCRIPT

JAVASCRIPT

Netscape Communication Corporation created JavaScript, a script-based computer language. JavaScript was initially known as Live Script before being renamed JavaScript to reflect its connection to Java. The development of both client and server components of Web-based applications is supported by JavaScript. It can be used to develop programmes that are executed by a Web browser within the context of a Web page on the client side. It can be used to develop Web server applications that process data submitted by a Web browser and then update the browser's display accordingly on the server side.

Despite the fact that JavaScript allows both client and server Web programming, we recommend it for client-side programming because it is supported by the majority of browsers.. JavaScript is almost as easy to learn as HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags

<SCRIPTS>..</SCRIPT>.

<SCRIPT LANGUAGE = "JavaScript">

JavaScript statements

</SCRIPT>

Here are some examples of what we can do with JavaScript:

Validate and calculate the contents of a form.

To the Browser's status line, add scrolling or changing messages.

Animate or rotate images so that they change when the mouse is moved over them.

Detect the browser being used and show different content for each one.

Installed plug-ins are detected, and the user is notified if a plug-in is necessary.

With JavaScript, we can do a lot more, including constructing complete applications.

THE DIFFERENCE BETWEEN JAVASCRIPT AND JAVA

The languages JavaScript and Java are diametrically opposed. The following are a handful of the most notable differences:

Java applets are often shown in a box within a web document; however, JavaScript can alter any portion of the online content.
While JavaScript is ideal for small apps and adding interactive elements to Web pages, Java can handle extremely sophisticated programmes.
There are numerous more distinctions, but the most crucial is that JavaScript and Java are two distinct languages. They are both beneficial for distinct purposes; in fact, they can be combined to maximise their benefits.

ADVANTAGES

JavaScript is a server-side and client-side programming language.
It has a greater degree of flexibility than VBScript.
Because all browsers support JavaScript, it is the default scripting language on the client side.

Hyper Text Markup Language

Hypertext Markup Language (HTML), one of the World Wide Web's (WWW) languages, enables users to create Web pages with text, pictures, and links to other Web pages (Hyperlinks).
HTML is not a programming language, but rather a hypertext version of ISO Standard 8879, SGML (Standard Generalized Markup Language), designed for the Web. Instead of reading content in a tight linear structure, Hypertext allows us to hop from one point to another with ease. We can sort the material according to our interests and preferences. A markup language is nothing more than a collection of components separated by special characters that specify how text or other items included within the elements should be presented. Hyperlinks are highlighted or bolded works that take you to other papers or sections of the same document.

HTML can be used to display any sort of document on a host computer that is located in a different geographical location. It's a flexible language that may be used on any platform or on a desktop computer.
Tags (special codes) are used in HTML to make a document more appealing. The case of HTML tags is unimportant. Graphics, typefaces, varied sizes, colour, and other elements can improve the document's display. Anything that isn't a tag is considered part of the document.

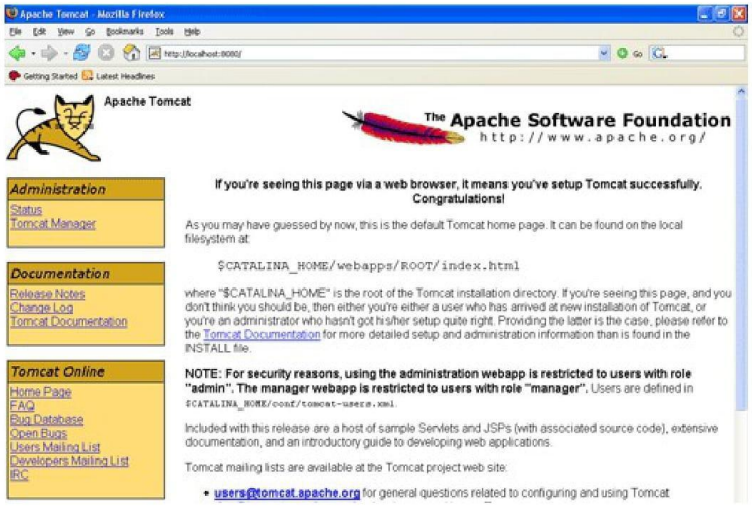
ADVANTAGES

- A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
- HTML is platform independent.
- HTML tags are not case-sensitive.

Tomcat 6.0 web server

The Apache Group created Tomcat, an open source web server.The Java Servlet and Java Server Pages technologies' official Reference Implementation. uses Apache Tomcat as its servlet container. Sun created the Java Servlet and Java Server Pages specifications as part of the Java Community Process. Web

servers, such as Apache Tomcat, only handle web components, but an application server supports both web and business components (BEAs Weblogic, is one of the popular application server). Install any web server, such as JRun, Tomcat, or others, to run your jsp/servlet-based web application.



Conclusion

In this article, we provided a formal definition and security model for CP-ABE with user revocation. We also construct a concrete CP-ABE scheme which is CPA secure based on DCDH assumption. To resist collusion attack, we embed a certificate into the user's private key. So that malicious u s-ers and the revoked users do not have the ability to gener-ate a valid private key through combining their private keys. Additionally, we outsource operations with high computation cost to E-CSP and D-CSP to reduce the user's computation burdens. Through applying the technique of outsource, computation cost for local devices is much lower and relatively fixed. The results of our experiment show that our scheme is efficient for resource constrained devic-es.

References

[1] A. Sahai and B. Waters, “Fuzzy Identity-Based Encryption,” EUROCRYPT ’05, LNCS, vol. 3494, pp. 457-473, 2005.

[2] J. Bethencourt, A. Sahai and B. Waters, “Ciphertext-Policy Attribute- Based Encryption,” Proc. IEEE Symposium on Security and Privacy, IEEE Transactions on Services Computing,Volume:PP,Issue:99,Date of Current Version:22.January.2016pp. 321-334, May 2007, doi: 10.1109/SP.2007.11.

[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-Based En-cryption for Fine-Grained Access Control of Encrypted Data,” Proc. 13th ACM Conference on Computer and Communications Security (CCS ’06), pp. 89-98, 2006, doi:10.1145/1180405.1180418.