Project Phase-2 Report

On

Implementing CACDES:

Centralized **A**ccess **C**ontrol in a **D**istributed **E**nvironment **S**ystem

Submitted in partial fulfilment of the requirement for the Degree of Bachelor of Technology in Information Technology

at

DIT University, Dehradun

By

Akshat Joshi

1401051080

Under the guidance of

Mrs Garima Verma, Assistant Professor
Department of Information technology

DIT UNIVERSITY, DEHRADUN

(State Private University through State Legislature Act No. 10 of 2013 of Uttarakhand and approved by UGC)

Mussoorie Diversion Road, Dehradun, Uttarakhand - 248009, India.

2017-2018

CANDIDATE DECLARATION

I hereby certify that the work, which is being presented in the project report, entitled **Implementing CACDES**, in partial fulfilment of the requirement for the award of the Degree of **Bachelor of Technology** and submitted to the institution is an authentic record of my/our own work carried out during the period *March'17-April'17* under the supervision of **Mrs Garima Verma.**

Date: Signature of the Candidate

This is to certify that the above statement made by the candidate is correct to the best of my /our knowledge.

Date: Signature of the Supervisor

CERTIFICATE

This is to certify that the project report entitled "Implementing CACDES" being submitted by **"Akshat Joshi"** in partial fulfillment for the award of the Degree of Bachelor of Technology in Information Technology to the DIT University is a record of bona fide work carried out by them under my guidance and supervision.

The results embodied in this project report have not been submitted to any other University or Institute for award of any Degree or Diploma.

Signature of Guide

Mrs. Garima Verma
Asst. Professor
Department of IT
DIT University

Signature of HOD

Dr. Rama Sushil
Head of Department
Department of IT
DIT University

ACKNOWLEDGEMENT

It is my privilege to express my sincerest regards to my project coordinator, **Mrs Garima Verma,** for her valuable inputs, able guidance, encouragement, whole-hearted cooperation and constructive criticism throughout the duration of my project.

I deeply express my sincere thanks to our Head of Department **Dr Rama Sushil** for encouraging and allowing me to present the project “Implementing CACDES “at our department premises for the partial fulfilment of the requirements leading to the award of B-Tech degree.

I take this opportunity to thank all my faculties who have directly or indirectly helped in my project. I pay my respects and love to my parents and all other family members and friends for their love and encouragement throughout my career.

Akshat Joshi

ABSTRACT

The idea behind this project is based on the research paper titled “***Implementation of highly efficient Authentication and Transaction Security***” to create and establish a distributed environment that can allow users to access data servers with a safe and secure mechanism. Through the implementation of this project we can communicate between two or more systems, transfer files & replicate data etc. within same network through a centralized access server that authenticates each and every client that wants access the services provided by the data servers. The connections are established using Java Socket programming and secured via JCE (Java Cryptography Extension).Whole system is centrally controlled via Access server.

TABLE OF CONTENTS

LIST OF FIGURES

HAPTER 1- INTRODUCTION

Prior to the emergence of low-cost desktop computer power, computing was generally centralized to one computer. Although such centres still exist, distribution networking applications and data operate more efficiently over a mix of desktop workstations, local area network servers, regional servers, Web servers, and other servers. Enterprises that have grown in scale over the years and those that are continuing to grow are finding it extremely challenging to manage their distributed network in the traditional client/server computing model. My project is to implements a simple and secure peer-to-peer distributed network implemented through java socket programming. There is also a file sharing system using a software library designed and implemented for the use of applications on the system. Understanding distributed systems requires knowledge of a number of areas including system architecture, networking, transaction processing, and security, among others.

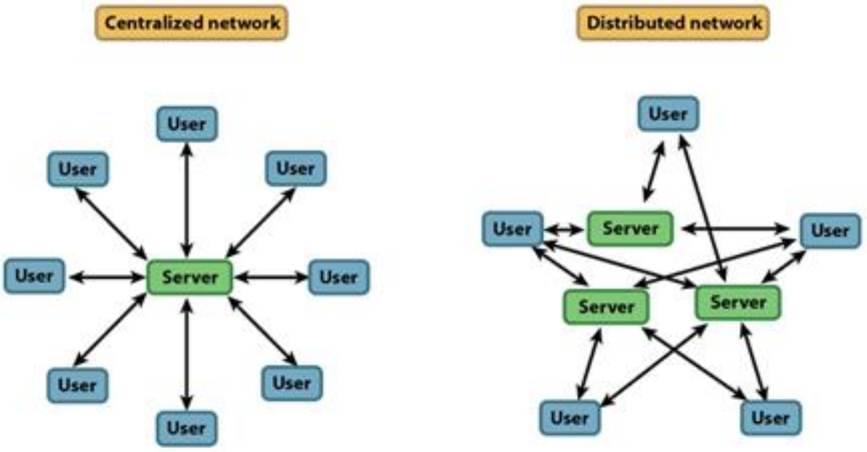


Figure-1 Centralized System vs. Distributed System

1.1 Distributed Systems

A distributed system is one in which both data and transaction processing is divided between one or more computers connected by a network, each computer playing a specific role in the system. A Distributed system is a collection of autonomous computers linked by a network that appear to the users of the system as a single computer.

1.2 Access Server

Access Server is used to describe computer software and hardware that can authorize or prohibit a client to access data servers. It is used for key management and authentication of clients.

1.3 Data Server

Data server is the phrase used to describe computer software and hardware (a database platform) that delivers database services. Also called a database server it also performs tasks such as data analysis, storage, data manipulation, archiving, and other tasks using client/server architecture.

1.4 Sockets

A **socket** is one endpoint of a two-way communication link between two programs running on the network. A **socket** is bound to a port number so that the TCP layer can identify the application that data is destined to be sent to. An endpoint is a combination of an IP address and a port number.

1.5 Socket Programming

Java Socket programming is used for communication between the applications running on different JRE. Java Socket programming can be connection-oriented or connection-less. Socket and Server Socket classes are used for connection-oriented socket programming and Datagram Socket and Datagram Packet classes are used for connection-less socket programming. The client in socket programming must know two information:

1. IP Address of Server, and
2. Port number.

Socket class

A socket is simply an endpoint for communications between the machines. The Socket class can be used to create a socket.

Important methods:

Method	Description
1) public InputStream getInputStream()	returns the InputStream attached with this socket.
2) public OutputStream getOutputStream()	returns the OutputStream attached with this socket.
3) public synchronized void close()	closes this socket

ServerSocket class

The ServerSocket class can be used to create a server socket. This object is used to establish communication with the clients.

Important methods:

Method	Description
1) public Socket accept()	returns the socket and establish a connection between server and client.
2)public synchronized void close()	closes the server socket.

Features of the Distributed Network

Performance

- The collection of processors can provide higher performance.

Incremental Growth

- If required more processing, we can add the systems to the distributed systems.

Reliability

- If one machine gets faulty, other machines can still survive in the network.

Speed

- One machine can have more computing power than a mainframe.

Open System

- One machine in the network acts as an open system so that the resource of that system can be accessed over the whole network.

Economic

- Better price than the other mainframe.

Sharing data or resources

- Shared data is essential to many applications such as banking, reservation systems etc., other resources (expensive printers) can also be shared.

Communication

- Facilitates human-to-human communication.

Data Replication

- Data can be replicated at multiple terminals to ensure its safety.

Scalability

- The System can be horizontally and vertically scaled as per requirements.

Data Flow Diagram

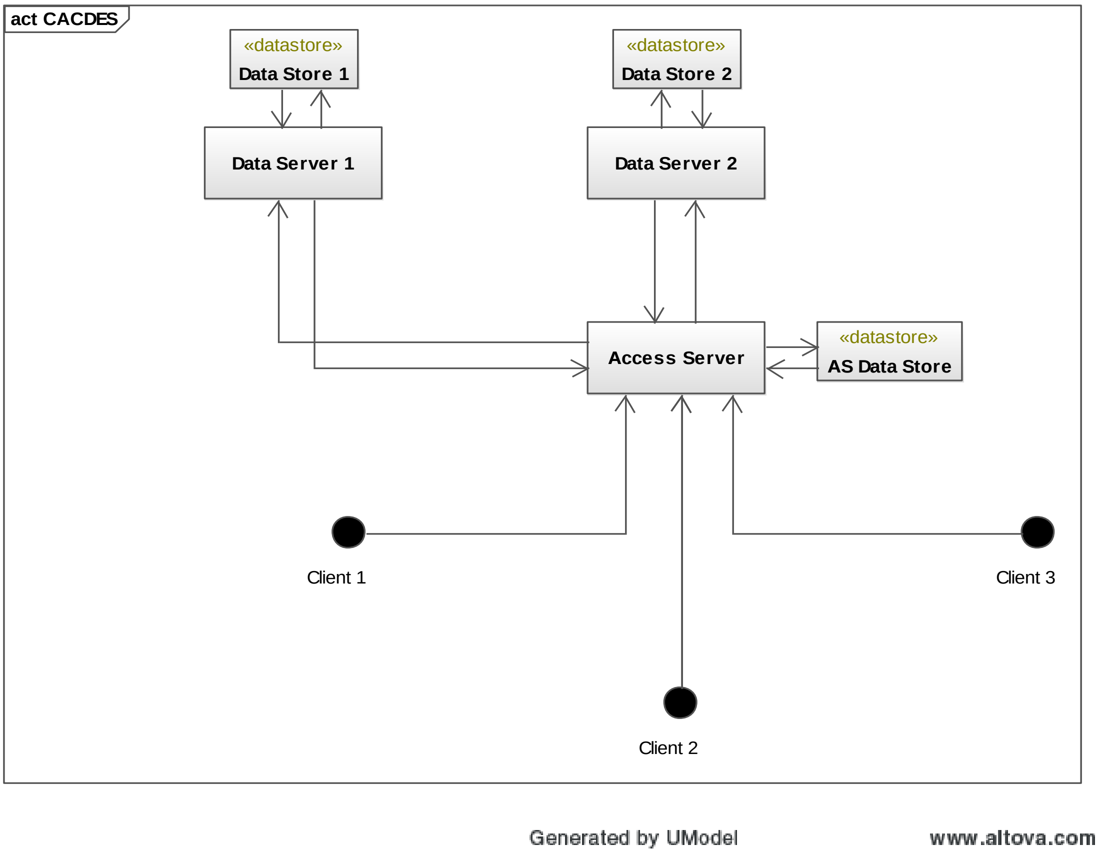


Figure-2 CACDES DFD

CHAPTER 2 PROBLEM STATEMENT & LITERATURE REVIEW

2.1 PROBLEM STATEMENT

Distributed Computing is a highly efficient mode of computing in which security is a major concern due the remote locations of different terminals. Through this project I am proposing a stable and secure distributed system in which client is allowed data server's services only if the client is authenticated by access server.

2.2 LITERATURE REVIEW

There are various studies are done in the area of distributed systems. Because of the systems are available in different locations it is very much required to have highly secure system to access the contents from the data servers.

G. Verma, R. Arora (2011) Kerberos is a network authentication protocol & is designed to provide strong authentication for client/server applications by using secret-key cryptography. Our research was aimed at enhancing the security of transactions over a network. In this paper, we used Kerberos Encryption Technique for authentication and transaction security in the network. Further, we created an Authentication Server that used to derive a 64 bit key from user's password. This password was of arbitrary length. The generated key then was used by authentication server, to encrypt ticket granting ticket + session key. The key generated by authentication server was then used by the client at the time of transaction through the transaction server to validate an authentic transaction. However, there was an issue of cross-validation of the ticket by the transaction server for which we included a database and encryption of all the text sent by any client to the transaction server.

Dr. S. Santhosh Baboo, K. Gokulraj (2010), Authentication is one of the essential security features in network communication. Authentication process ascertains the legitimacy of the communicating partners in communication. The authors introduced a new authentication scheme based on dynamicity which is relatively a different approach to ensure and enhance the smart card based remote authentication and security. This method discusses about the authentication for smart card based network systems. This method introduces a dynamic authentication scheme which includes number of factors, among them the password, password index, and date of modification are important factors which decides the dynamicity.

K. Aruna et. al (2010), The aim of this paper is to establish a collaborative trust enhanced security model for distributed system in which a node either local or remote is trustworthy. They have also proposed a solution with trust policies as authorization semantics. Kerberos, a network authentication protocol is also used to ensure the security aspect when a client requests for certain services. In the proposed solution, they have also considered the issue of performance bottlenecks.

Steve Mallard(2010), He has defined various authentication method In order to protect the assets on your network. Like username and password, Biometric systems, Kerberos etc.

Dr.Mohammad N. Abdullah & May T. Abdul-Hadi (2009) they try to establish a secure communication between the clients and mobile-bank application server in which they can use their mobile phone to securely access their bank accounts, make and receive payments, and check their balances.

Hongjun liu et. al (2008), This paper has discussed potential server bottleneck problem when the Kerberos model is applied in large-scale networks because the model uses centralized management. They have proposed an authentication model based on Kerberos. This tries to overcome the potential server bottleneck problem and can balance the load automatically.

CHAPTER 3-METHODOLOGY AND IMPLENTATION DETAILS

3.1 Methodology

The methodology used to develop this project is Java TCP Socket programming and database management system. The project is divided in following modules –

- Analysis
- Design
- Implementation and Testing

We will work on this project in three phases-

<u>Phase 1</u>: This phase will be covered in 6th semester. Phase 1 is further divided into following phases:-

Sr. No	Phases	Time Duration
1.	Software Requirement Specification	1 week
2.	Designing User Interface	1 week
3.	25% Coding	2 weeks

<u>Phase 2</u>: This phase will be covered in 7th semester. Phase 2 is further divided into following phases:-

Sr. No	Phases	Time Duration
1.	50% Coding	2 weeks
2.	Testing	1 week

<u>Phase3</u>: This phase will also be covered in 8th semester. Phase 2 is further divided into following phases:-

Sr. No	Phases	Time Duration
1.	25% Coding	2 weeks
2.	Implementation	1 week
3.	Testing	1 week

3.2 Implementation Details

3.2.1 Hardware Requirements

Client Side

- ❖ Processor :Pentium III or higher
- ❖ 1GB RAM
- ❖ 250 Gb HDD or more
- ❖ Monitor
- ❖ Keyboard
- ❖ Mouse
- ❖ Internet Connection

Server Side

- ❖ Processor :Intel Core i3 or higher
- ❖ 4 GB RAM
- ❖ 1Tb HDD or more
- ❖ Monitor
- ❖ Keyboard
- ❖ Mouse
- ❖ Internet Connection

3.2.2 Software Requirements

- J2SDK
- Netbeans
- MySql
- Apache Tomcat Server
- Microsoft word 2000

3.3 Languages and Backend

3.3.1 Language

Java

Java is a set of computer software and specifications developed by Sun Microsystems, later acquired by Oracle Corporation that provides a system for developing application software and deploying it in a cross-platform computing environment. Java is used in a wide variety of computing platforms from embedded devices and mobile phones to enterprise servers and supercomputers. While less common, Java applets run in secure, sandboxed environments to provide many features of native applications and can be embedded in HTML pages. The Java platform is a suite of programs that facilitate developing and running programs written in the Java programming language. A Java platform will include an execution engine (called a virtual machine), a compiler and a set of libraries; there may also be additional servers and alternative libraries that depend on the requirements. Java is not specific to any processor or operating system as Java platforms have been implemented for a wide variety of hardware and operating systems with a view to Java programs running identically on all of them.

Figure-3 Java Logo

Apache Tomcat Server

Apache Tomcat, often referred to as **Tomcat Server**, is an open-source Java Servlet Container developed by the Apache Software Foundation (ASF). Tomcat implements several Java EE specifications including Java Servlet, JavaServer Pages (JSP), Java EL, and WebSocket, and provides a "pure Java" HTTP web server environment in which Java code can run.

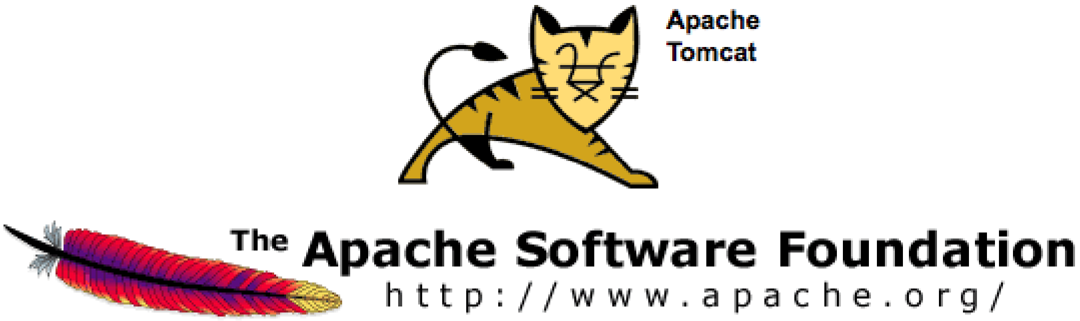


Figure-4 Tomcat Logo

Tomcat is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation, released under the Apache License 2.0 license, and is open-source software.

MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founders Michael Wideness's daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL_AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL and Perl/PHP/Python". Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

Figure-5 MySql Logo

3.4 Functional Requirements

1. Client Login and Registration – allows non-registered user to create his/her account and existing user can sign in to the system.
2. OTP Generation – A unique one-time password is generated by access server on the request from client.
3. Client Verification – The generated OTP is used for client verification.

3.5 Non Functional Requirements

1. Security
2. Durability
3. Reliability
4. Scalability

CHAPTER 4- MODULES OF CACDES

4.1 Client Control Module

This module is for registration of client on AWT based GUI. This module has three features namely register, login (if already registered) and forgot password.

1. **Register** - First time mandatory registration for new clients. Requires name, username, password and email address.
2. **Login -** User enters his/her credentials which are stored in a database.
3. **Forgot Password -** In the case when client forgets his/her login password, the client can request for a new password through email verification.

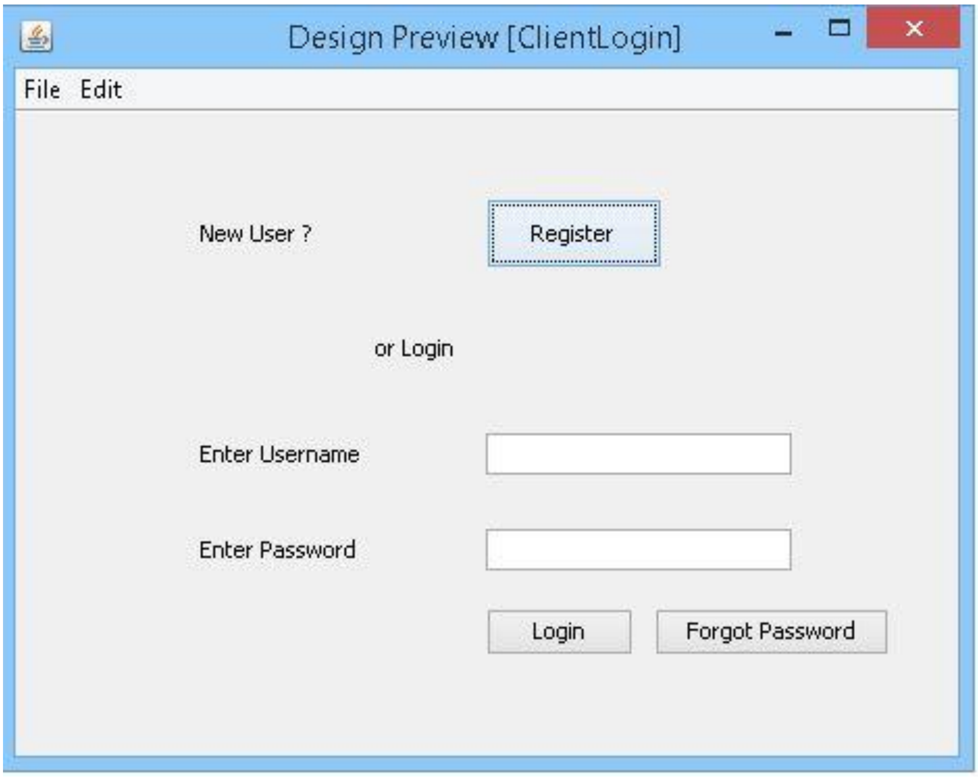


Figure-6 Client Login

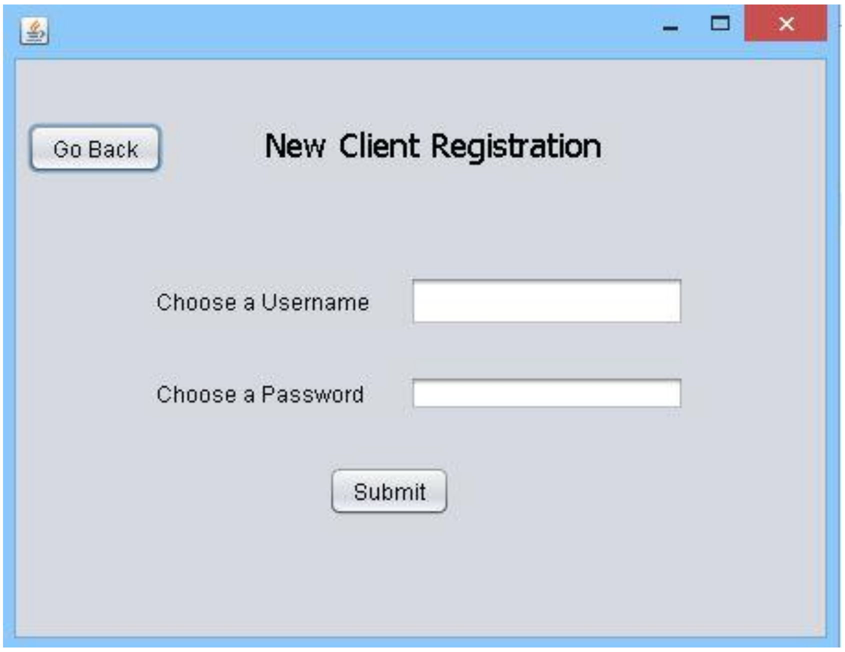
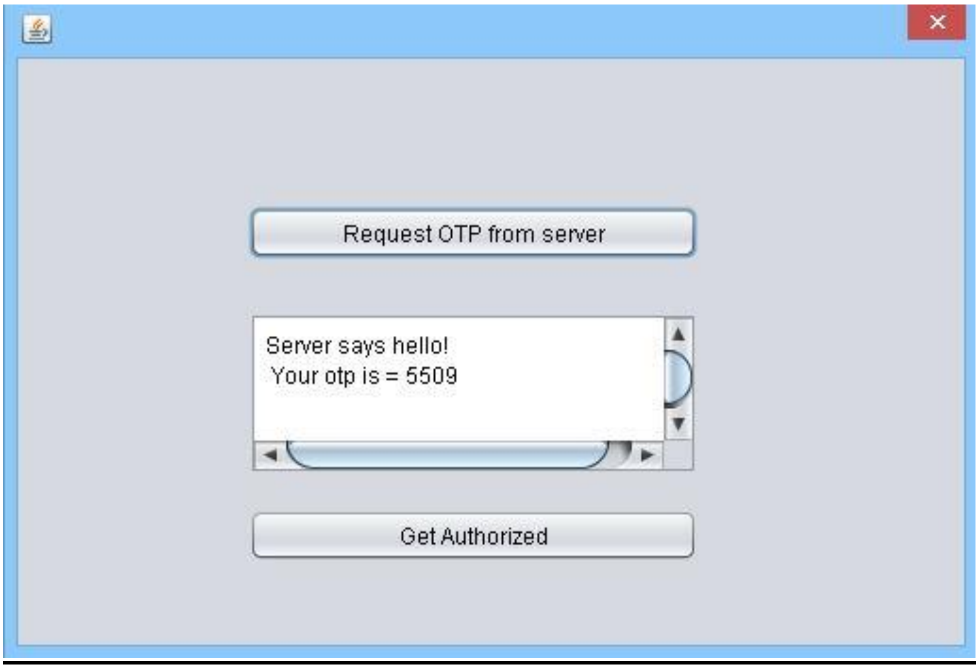


Figure-7 Client Registration

4.2 Access Control Module

This module is the handling of access server. After starting the access server the client details will be submitted to the server and user verification will be done through the server. If the verification is successful a unique code called as OTP is generated and sent back to client, otherwise Access is denied.

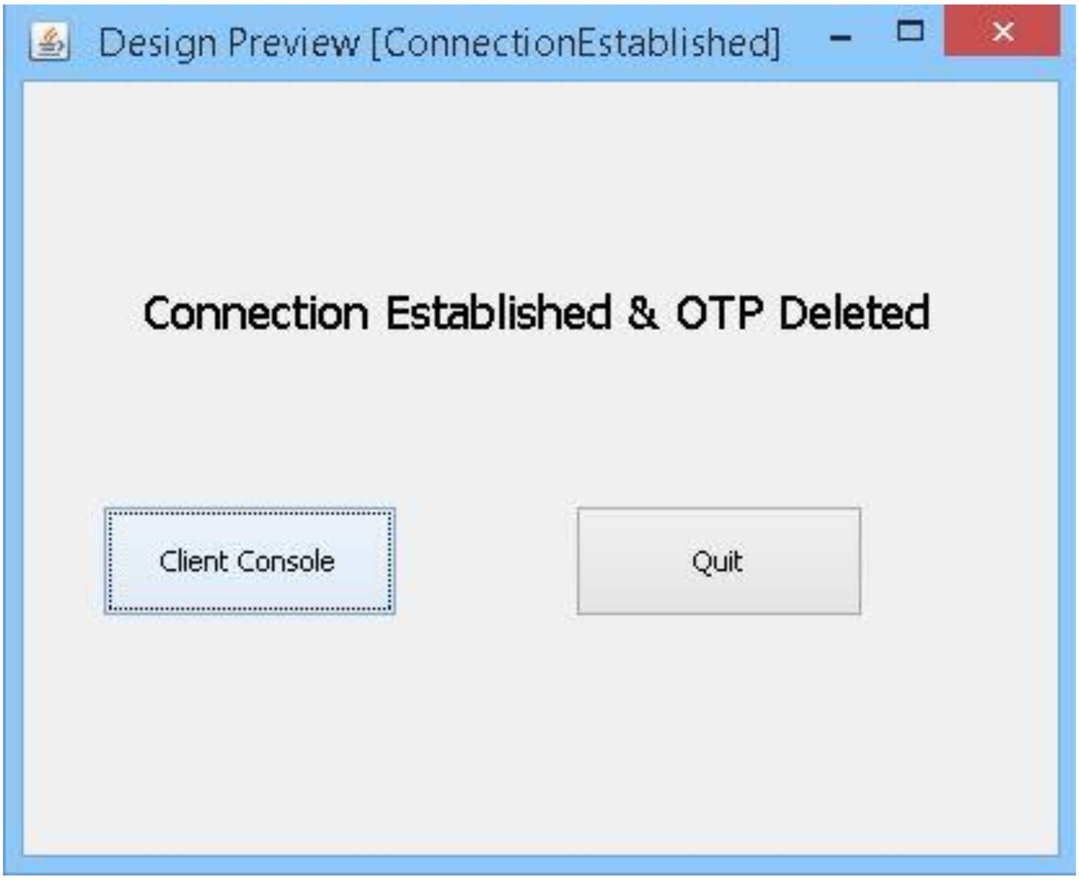


```
Server says hello!
 Your otp is = 5509
```

Figure-8 OTP Generation

4.3 Client Verification Module

The client is verified with help of his unique id and the otp generated. If the client is verified the client verification table at the backend is dropped. This ensures that the onetime password cannot be used again. And to regain access a new request has to be sent to the access server.



Design Preview [ConnectionEstablished]

Connection Established & OTP Deleted

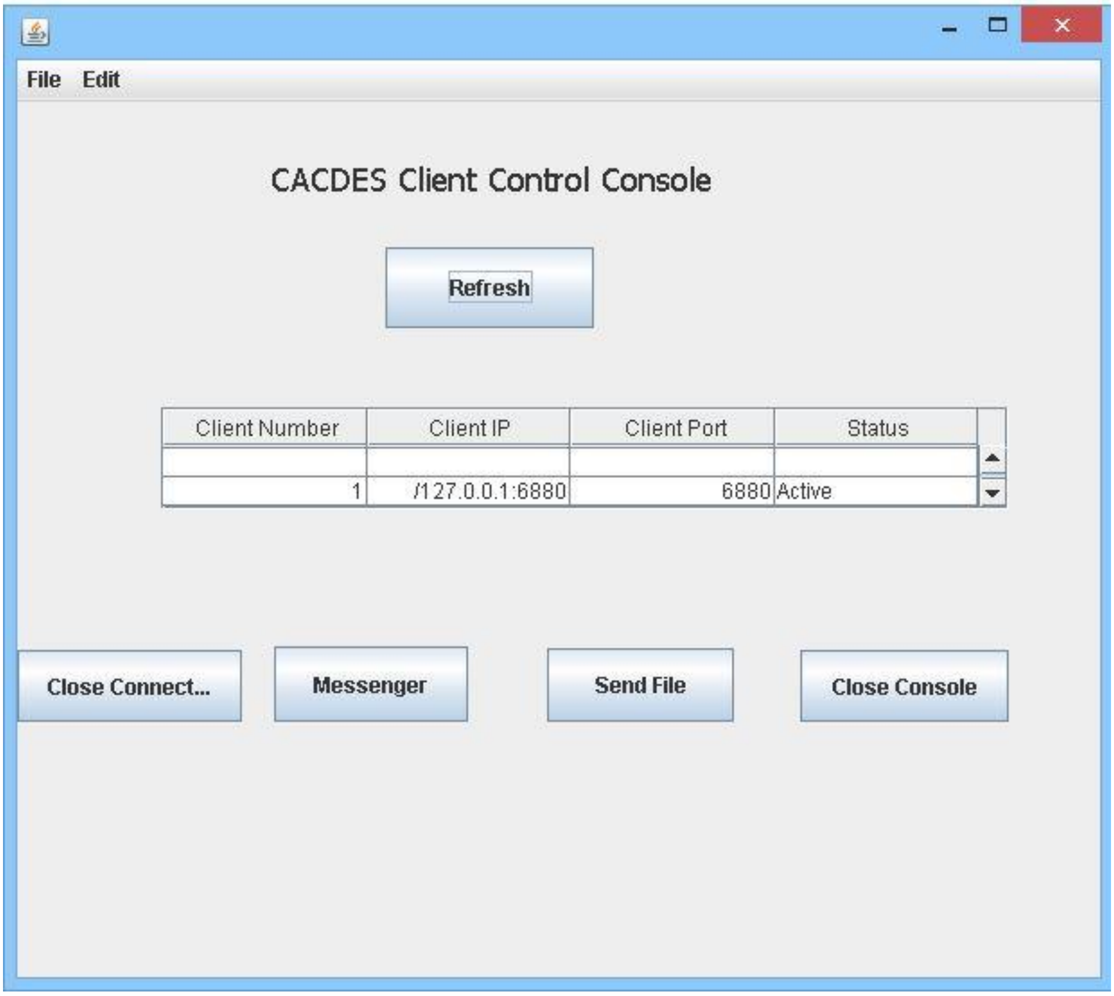
Figure-9 Client Verification

4.4 Client Details Console Module

The client control console is a graphical user interface for the client, so that he can interact with servers and request for services he require.

The Client control module provides the following functionalities:

- Display client’s connection information.
- Provides a messaging interace for the client.
- Provide the functionality to transfer files to servers database.



Client Number	Client IP	Client Port	Status
1	/127.0.0.1:6880	6880	Active

Figure-10 Client Details Console

CHAPTER 5- CONCLUSION AND FUTURE WORK

5.1 Conclusions

sIn the project we have purposed an access controlled distributed system application that can securely transfer data between systems that are connected in a local area network. The Communication that is established through this application is only authorised if the user and / system are registered on administrator server. The data transferred between client and data server is securely encrypted using JCE. In the phase-1 we have covered client registration module and access server module. The authenticated client can only logged in to the system. The verification of client is done by the access server and if the client is authenticated then OTP i.e. one time password will be generated by the access server.

5.2 Scope for Future Work

In the future we will continue with phase-3 in 8th semester. In which we will provide more functionalities and enhance security.

REFERENCES

- G. Verma, R. Arora, "*Implementation of highly efficient Authentication and Transaction Security*", International Journal of Computer Application, Vol-21, No-3, May 2011, pp. 43-49 ISSN-0975-8887. (Indexed in Ebsco, Cabell's Directory, Google scholar, Proquest Database. Impact factor: 0.75).
- Elliotte Rusty Harold, Java Network Programming, 4th Edition, O'Reilly, 2014
- Herbert Schildt, Java : The Complete Reference, 9th Edition, McGraw-Hill,2014
- Avi Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 6th Edition, McGraw-Hill, 2009
- George Coulouris, Distributed Systems Concepts & Design, 4th edition, Addison Wesley ,2005
- www.stackoverflow.com
- www.github.com
- www.w3schools.com
- www.javatpoint.com
- www.oracle.com