

Batch wise prognosticated resource allocation through cloud computing

M. Mohith Riteesh

Department of computer science and Engineering

KoneruLakshmaiah Education Foundation

Vaddeswarm, Guntur District, Andhra Pradesh, India

N. Prudvi Krishna

Department of computer science and Engineering

KoneruLakshmaiah Education Foundation

Vaddeswarm, Guntur District, Andhra Pradesh, India

Y.S.R. Harshith

Department of computer science and Engineering

KoneruLakshmaiah Education Foundation

Vaddeswarm, Guntur District, Andhra Pradesh, India

Dr. M. S. Mekala

Department of computer science and Engineering

Vaddeswarm, Guntur District, Andhra Pradesh, India

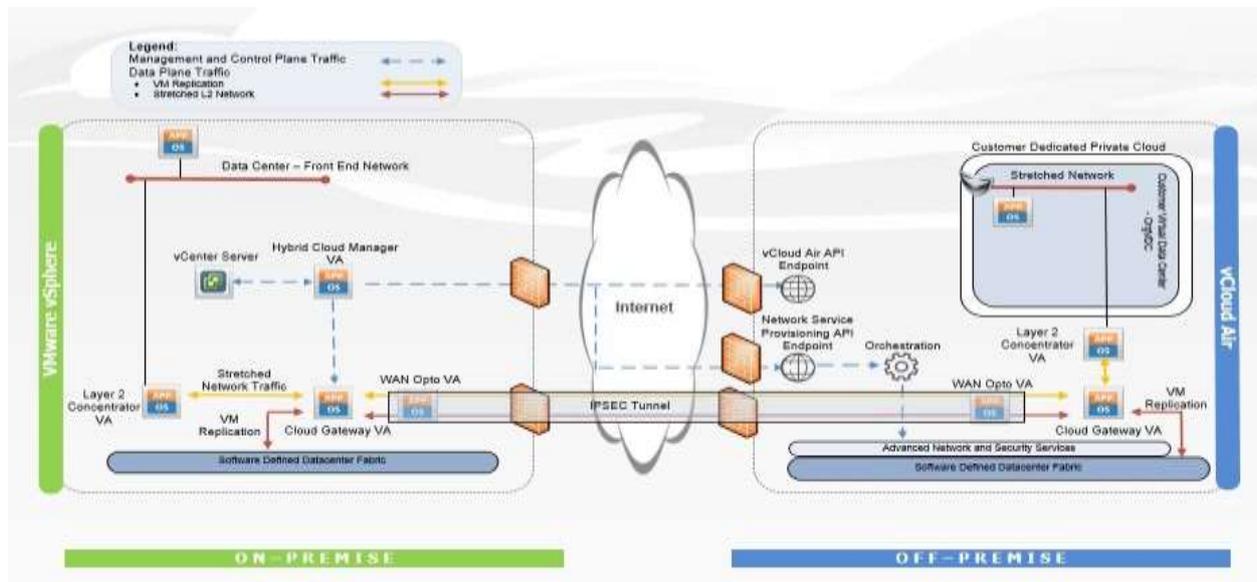
Abstract— Information get to control is a viable method to guarantee the asset designation dependent on information security in the cloud. In any case, because of information re-appropriating and untrusted cloud servers and asset designation, the information get to control turns into a difficult issue in distributed storage frameworks. Existing access control plans are not, at this point relevant to distributed storage frameworks, since they either produce numerous encoded duplicates of similar information or require a completely confided in cloud server. Figure content Policy Attribute-based Encryption (CP-ABE) is a promising method for get to control of scrambled information. It requires a believed authority deals with all the characteristics and disseminates enters in the framework. In distributed storage frameworks, there are numerous specialists coincide and every authority can give qualities autonomously. In any case, existing CP-ABE plans can't be straightforwardly applied to the entrance control for multi-authority distributed storage frameworks, because of the wastefulness of decoding and renouncement.

Keywords: Encryption, Security Service Algorithm, Distributed Storage frameworks.

I. Introduction

Distributed computing is the utilization of processing assets (equipment and programming) that are conveyed as a help over a system (ordinarily the Internet). The name originates from the basic utilization of a cloud-formed image as a reflection for the mind-boggling foundation it contains in framework outlines. Distributed computing endows remote

administrations with a client's information, programming and calculation. Distributed computing comprises of equipment and programming assets made accessible on the Internet as oversight outsider administrations. These administrations ordinarily give access to cutting edge programming applications and top of the line systems of server PCs.



The objective of distributed computing is to apply customary supercomputing, or elite figuring power, ordinarily utilized by military and research offices, to perform many trillions of calculations for each second, in customer situated applications, for example, monetary portfolios, to convey customized data, to give information stockpiling or to control enormous, vivid PC games. The distributed computing utilizes systems of enormous gatherings of servers normally running ease customer PC innovation with particular associations with spread information preparing errands across them. This mutual IT foundation contains huge pools of frameworks that are connected together. Frequently, virtualization strategies are utilized to expand the intensity of distributed computing.

Qualities AND SERVICES MODELS:

The striking qualities of distributed computing dependent on the definitions gave by the National Institute of Standards and Terminology (NIST) are sketched out beneath:

ON-DEMAND SELF-SERVICE:

A buyer can singularly arrangement figuring capacities, for example, server time and system stockpiling, varying consequently without requiring human communication with each specialist organizations.

Wide NETWORK ACCESS:

Abilities are accessible over the system and gotten to through standard components that advance use by heterogeneous slim or thick customer stages (e.g., cell phones, PCs, and PDAs).

Asset POOLING:

There is a feeling of area autonomy in that the client for the most part has no control or information over the specific area of the gave assets yet might have the option to determine area at a more elevated level of deliberation (e.g., nation, state, or server farm). Instances of assets incorporate capacity, preparing, memory, arrange data transfer capacity, and virtual machines.

Quick ELASTICITY:

Abilities can be quickly and flexibly provisioned, at times consequently, to rapidly scale out and quickly discharged to rapidly scale in. To the customer, the abilities accessible for provisioning frequently have all the earmarks of being boundless and can be bought in any amount whenever.

Estimated SERVICE:

Cloud frameworks consequently control and enhance asset use by utilizing a metering capacity at some degree of reflection suitable to the kind of administration (e.g., capacity, preparing, data transfer capacity, and dynamic client accounts). Asset utilization can be overseen, controlled, and detailed giving straightforwardness to both the supplier and purchaser of the used assistance.

Administrations MODELS: Distributed computing contains three distinctive help models, to be specific Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). The three assistance models or layer are finished by an end client layer that typifies the end client point of view on cloud administrations. The model is appeared in figure underneath. On the off chance that a cloud client gets to administrations on the foundation layer, for example, she can run her own applications on the assets of a cloud framework and stay answerable for the help, support, and security of these applications herself.

Advantages OF CLOUD COMPUTING:**Accomplish ECONOMIES OF SCALE:**

Increment volume yield or efficiency with less individuals. Your expense per unit, venture or item plunges.

Lessen SPENDING ON TECHNOLOGY INFRASTRUCTURE:

Keep up simple access to your data with negligible forthright spending. Pay more only as costs arise (week after week, quarterly or yearly), in view of interest.

GLOBALIZE YOUR WORKFORCE ON THE CHEAP:

Individuals worldwide can get to the cloud, if they have an Internet association.

Diminish CAPITAL COSTS:

There's no compelling reason to spend enormous cash on equipment, programming or permitting expenses.

IMPROVE ACCESSIBILITY:

You approach whenever, anyplace, making your life so a lot simpler!

Screen PROJECTS MORE EFFECTIVELY:

Remain inside financial plan and in front of consummation process durations.

LESS PERSONNEL TRAINING IS NEEDED:

It takes less individuals to accomplish more work on a cloud, with an insignificant expectation to absorb information on equipment and programming issues.

Limit LICENSING NEW SOFTWARE:

Extend and develop without the need to purchase costly programming licenses or projects.

IMPROVE FLEXIBILITY:

You can alter course without genuine "individuals" or "budgetary" issues in question.

Focal points:

PRICE: Pay for just the assets utilized.

SECURITY: Cloud examples are detached in the system from different occurrences for improved security.

PERFORMANCE: Instances can be included immediately for improved execution. Customers approach the absolute assets of the Cloud's center equipment.

SCALABILITY: Auto-send cloud examples when required.

UPTIME: Uses various servers for greatest redundancies. If there should arise an occurrence of server disappointment, cases can be consequently made on another server.

CONTROL: Able to login from any area. Server preview and a product library lets you convey custom examples.

TRAFFIC: Deals with spike in rush hour gridlock with snappy sending of extra examples to deal with the heap.

EXISTINGSYSTEM: Due to information re-appropriating and untrusted cloud servers furthermore, asset designation and asset distribution and asset portion, the information get to control turns into a difficult issue in distributed storage systems Existing access control plans are not, at this point material to distributed storage frameworks, since they either produce different encoded duplicates of similar information or require a completely confided in cloud server.

Detriments: Be that as it may, distributed storage administration isolates the jobs of the information proprietor from the information specialist co-op, and the information proprietor doesn't cooperate with the client legitimately for giving information get to support, which makes the information get to control a difficult issue in distributed storage frameworks. Since the cloud server can't be completely trusted by information proprietors, customary server-based access control techniques are not, at this point material to distributed storage frameworks.

II. PROPOSED SYSTEM

we first build another multi-authority CPABE conspire with proficient unscrambling and structure a productive trait disavowal strategy for it. At that point, we apply them to plan a compelling access control plot for multi-authority frameworks.

The fundamental commitments of this work can be summed up as follows.

We propose DAC-MACS (Data Access Control for Multi-Authority Cloud Storage), a successful and secure information get to control conspire for multi-authority distributed storage frameworks, which is provably secure in the irregular prophet model and has preferable execution over existing plans.

We develop another multi-authority CP-ABE plot with effective unscrambling. In particular, we redistribute the fundamental calculation of the unscrambling by utilizing a token based decoding strategy.

We likewise plan a proficient prompt quality denial technique for multi-authority CP-ABE conspire that accomplishes both forward security and in reverse security. It is productive as in it causes less correspondence cost and calculation cost of the repudiation.

III. LITERATURE REVIEW.

MULTI-AUTHORITY ATTRIBUTE BASED ENCRYPTION:

Creators: M. Pursue, A personality-based encryption conspire, every client is distinguished by a one of a kind character string. A property-based encryption plot (ABE), conversely, is a plan wherein every client is recognized by a lot of traits, and some capacity of those credits is utilized to decide unscrambling capacity for each ciphertext. Sanai and Waters presented a solitary position quality encryption plan and left open the topic of whether a plan could be developed in which different specialists were permitted to disseminate properties [SW05]. We answer this inquiry in the certifiable. Our plan permits any polynomial number of autonomous specialists to screen properties and disseminate mystery keys. An encryption can pick, for every power, a number dk and a lot of traits; he would then be able to encode a message with the end goal that a client can possibly unscramble in the event that he has in any event dk of the given qualities from every position k . Our plan can endure a subjective number of degenerate specialists. We additionally tell the best way to apply our strategies to accomplish a multiauthority form of the huge universe fine grained get to control ABE introduced by Gopal et al.

IMPROVING PRIVACY AND SECURITY IN MULTI-AUTHORITY ATTRIBUTE-BASED ENCRYPTION:

AUTHORS: M. Pursue and S.S.M. Chow Attribute based encryption (ABE) [13] decides decoding capacity dependent on a client's traits. In a multi-authority ABE plot, numerous trait specialists screen various arrangements of characteristics and issue relating unscrambling keys to clients and encryptions can necessitate that a client acquire keys

for suitable qualities from every authority before decoding a message. Pursue [5] gave a multi-authority ABE plot utilizing the ideas of a confided in focal power (CA) and worldwide identifiers (GID).

Nonetheless, the CA in that development has the ability to decode each ciphertext, which appears to be some way or another conflicting to the first objective of dispersing power over numerous possibly untrusted specialists. In addition, in that development, the utilization of a predictable GID permitted the specialists to consolidate their data to manufacture a full profile with the entirety of a client's characteristics, which superfluously bargains the security of the client. In this paper, we propose an answer which expels the confided in focal position, and ensures the clients' security by keeping the specialists from pooling their data on specific clients, in this way making ABE increasingly usable by and by

DECENTRALIZING ATTRIBUTE-BASED ENCRYPTION:

Creators: A.B. Lewko and B. Waters We propose a Multi-Authority Attribute-Based Encryption (ABE) framework. In our framework, any gathering can turn into a power and there is no prerequisite for any worldwide coordination other than the making of an underlying arrangement of normal reference parameters. A gathering can essentially go about as an ABE authority by making an open key and giving private keys to various clients that mirror their characteristics. A client can scramble information regarding any Boolean recipe over properties gave from any picked set of specialists. At long last, our framework doesn't require any focal position.

In developing our framework, our biggest specialized obstacle is to make it intrigue safe. Earlier Attribute-Based Encryption frameworks accomplished agreement obstruction when the ABE framework authority "tied" together various parts (speaking to various traits) of a client's private key by randomizing the key. Nonetheless, in our framework every segment will originate from a possibly extraordinary power, where we accept no coordination between such specialists. We make new procedures to integrate key parts and forestall intrigue assaults between clients with various worldwide identifiers.

We demonstrate our framework secure utilizing the ongoing double framework encryption system where the security verification works by first changing over the test ciphertext and private keys to a semi-useful structure and afterward contending security. We follow an ongoing variation of the double framework verification method due to Lewko and Waters and fabricate our framework utilizing bilinear gatherings of Composite request. We demonstrate security under comparative static suspicions to the LW paper in the irregular prophet model.

Property BASED DATA SHARING WITH ATTRIBUTE REVOCATION:

Creators: S. Yu, C. Wang, K. Ren, and W. Lou Ciphertext-Policy Attribute Based Encryption (CP-ABE) is a promising cryptographic crude for fine-grained get to control of shared information. In CP-ABE, every client is related with a lot of properties and information are scrambled with get to structures on traits. A client can unscramble a ciphertext if and just if his qualities fulfil the ciphertext get to structure. Adjacent to this essential property, down to earth applications generally have different necessities. In this paper we centre around a significant issue of trait denial which is awkward for CP-ABE plans. Specifically, we settle this difficult issue by considering progressively handy situations where semi-trustable on-line intermediary servers are accessible. When contrasted with existing plans, our proposed arrangement empowers the power to renounce client traits with negligible exertion. We accomplish this by interestingly incorporating the strategy of intermediary re-encryption with CP-ABE, and empower the power to appoint the vast majority of difficult errands to intermediary servers. Formal examination shows that our proposed plot is provably secure against picked ciphertext assaults. What's more, we show that our strategy can likewise be material to the Key-Policy Attribute Based Encryption (KP-ABE) partner.

Versatile AND SECURE SHARING OF PERSONAL HEALTH RECORDS IN CLOUD COMPUTING USING ATTRIBUTE-BASED ENCRYPTION:

Creators: M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou Personal wellbeing record (PHR) is a rising patient-driven model of wellbeing data trade, which is regularly

redistributed to be put away at an outsider, for example, cloud suppliers. Be that as it may, there have been wide security worries as close to home wellbeing data could be presented to those outsider servers and to unapproved parties. To guarantee the patients' power over access to their own PHRs, it is a promising technique to scramble the PHRs before re-appropriating. However, issues, for example, dangers of protection presentation, versatility in key administration,

adaptable access and productive client renouncement, have remained the most significant difficulties toward accomplishing fine-grained, cryptographically authorized information get to control.

In this paper, we propose a novel patient-driven system and a set-up of instruments for information get to control to PHRs put away in semi-confided in servers. To accomplish fine-grained and versatile information get to control for PHRs, we influence trait-based encryption (ABE) strategies to scramble every patient's PHR document. Unique in relation to past works in secure information re-appropriating, we centre around the numerous information proprietor situation, and gap the clients in the PHR framework into different security spaces that significantly lessens the key administration multifaceted nature for proprietors and clients. A high level of patient security is ensured all the while by abusing multi-authority ABE. Our plan likewise empowers dynamic change of access strategies or document characteristics, underpins effective on-request client/quality repudiation and break-glass access under crisis situations. Broad logical and exploratory outcomes are introduced which show the security, versatility and productivity of our proposed plot.

IV. SOFTWARE REQUIREMENT SPECIFICATIONS:

Framework REQRIMENTS&ANALYSIS:

EXISTING SYSTEM: This new worldview of information facilitating and information get to administrations acquaints an incredible test with information get to control. Since the cloud server can't be completely trusted by information proprietors, they can no longer depend on servers to do get to control. Figure content Policy Attribute-based Encryption (CP-ABE) is viewed as one of the most suitable technologies for information get to control in distributed storage frameworks, since it gives the information proprietor more straightforward control on get to approaches. In CP-ABE plot, there is a position that is answerable for property the executives and key dispersion.

Weaknesses OF EXISTING SYSTEM: Purse's multi-authority CP-ABE convention permits the focal position to unscramble all the figure writings, since it holds the ace key of the framework. .

PROPOSED SYSTEM: In this paper, we initially propose a revocable multiauthority CP-ABE plot, where an effective and secure repudiation technique is proposed to take care of the property denial issue in the framework. Our quality disavowal strategy is effective as in it brings about less correspondence cost and calculation cost, and is secure as in it can accomplish both in reverse security (The denied client can't unscramble any new ciphertext that requires the renounced ascribe to decrypt)and forward security (The recently joined client can likewise decode the recently distributed ciphertexts¹, in the event that it has sufficient. Attributes).Our plan doesn't require the server to be completely trusted, in light of the fact that the key update is authorized by each property authority not the server. Regardless of whether the server isn't semi trusted in certain situations, our plan can at present assurance the regressive security. At that point, we apply our proposed revocable multi-authority CP-ABE conspire as the hidden procedures to develop the expressive and secure information get to control plot for multi-authority distributed storage frameworks.

Points of interest OF PROPOSED SYSTEM: We change the structure of the plan and make it progressively functional to distributed storage frameworks, in which information proprietors are not engaged with the key age. We extraordinarily improve the effectiveness of the property renouncement strategy. We likewise profoundly improve the expressiveness of our entrance control conspire, where we evacuate the constraint that each property can just show up all things considered once in a ciphertext.

SYSTEM CONFIGURATION:

HARDWARE SYSTEM CONFIGURATION:

Processor :	Intel(R) Core i5-5200U CPU @2.20GHz 2.20 GHz,
Installed Memory (RAM)	: 8.00GB
System Type	: 64-bit operating System, x64-based Processor
Hard Disk -	1TB
Key	Board Logitech Wireless Bluetooth Multimedia Keyboard
Mouse	Optical Mouse / Pointing Device
Monitor -	LCD

S/W SYSTEM CONFIGURATION:

Operating System :	Windows 10
Application Server :	Tomcat 8.5
Front End :	HTML, Java, Jsp
Scripts :	JavaScript.
Server side Script	Java Server Pages.
Database :	Mysql 5.5
Database Connectivity	JDBC.

V. SOURCE CODE: -

Admin.jsp

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<title>CLOUD COMPUTING BASED SECURED APPLICATIONS FOR PLATFORMS BUILDING DEVELOPMENT</title>
```

```
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
```

```
<link href="style.css" rel="stylesheet" type="text/css" />
```

```
<!--<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/arial.js"></script>
<script type="text/javascript" src="js/cuf_run.js"></script>-->
<script>
    function validate(){
        var aid=document.name.adminid.value;
        var pass=document.name.password.value;
        if(aid==0){
alert("Enter Adminid");
document.name.adminid.focus();
            return false;
        }
        if(pass==0){
alert("Enter password");
document.name.password.focus();
            return false;
        }
    }
</script>
</head>
<body>
<%
    if(request.getParameter("msg")!=null){
out.println("<script>alert('incorrect password')</script>");
    }
    if(request.getParameter("msgg")!=null){
out.println("<script>alert('username not exist')</script>");
    } %>
<div class="main">
<div class="header">
```

```
<div class="header_resize">

<div class="logo">

<h1><a href="index.html">CLOUD COMPUTING BASED SECURED APPLICATIONS FOR PLATFORMS BUILDING
DEVELOPMENT</a></h1>

</div>

<div class="menu_nav">

<ul>

<li><a href="index.html">Home</a></li>

<li><a href="admin.jsp">Admin</a></li>

<!--<li><a href="about.html">About Us</a></li-->

<li><a href="user.jsp">User</a></li>

<li><a href="ca.jsp">CA</a></li>

</ul>

</div>

<div class="clr"></div>

<div style="position: absolute;left:210px;top: 150px">

<h2>ADMIN LOGIN PAGE</h2>

<form action="admin_verify.jsp"method="get"name="name" onsubmit="return validate()">

<strong> ADMIN ID:</strong><br>

<input type="text" name="adminid"placeholder="enter admin id"></input><br>

<strong>PASSWORD:</strong><br>

<input type="password" name="password" placeholder="enter password"></input><br></br>

<input type="submit" value="SignIn"></input>

</form>

</div>

<div class="clr"></div>

</div>

<div class="content">

<div class="content_resize">
```

```
<div class="mainbar">
</div>
<div class="sidebar">
<div class="gadget">
<div class="clr"></div>
</div></div>
<div class="clr"></div>
</div>
</div>
<div class="fbg">
<div class="fbg_resize">
<div class="clr"></div>
</div>
</div>
<div class="footer">
</div>
</div>
</body>
</html>
```

Download.jsp

```
<% @page import="java.sql.Blob"%>
<% @page import="java.io.FileOutputStream"%>
<% @page import="java.io.InputStream"%>
<% @page import="java.sql.ResultSet"%>
<% @page import="java.sql.DriverManager"%>
<% @page import="java.sql.Statement"%>
<% @page import="java.sql.Connection"%>
<% @page import="java.sql.Connection"%>
<%
try{
```

```
// Class.forName("org.sqlite.JDBC");

// Connection con=DriverManager.getConnection("jdbc:sqlite:/home/ibn/Desktop/Nadanapathy/NetBeansProjects/multi-
// authority_cloud_storage/multiauthority");

//
Class.forName("com.mysql.jdbc.Driver");

    Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/multiauthority","root","root");

//

    Statement st=con.createStatement();

ResultSet rt=st.executeQuery("select * from upload_files");

    if(rt.next()){

        String user=rt.getString("userid");

byte[] bt=new byte[256];

        Blob blob=rt.getBlob("file");

InputStreamisr=blob.getBinaryStream();

FileOutputStreamfos=new FileOutputStream(user+".txt");

        while(isr.read(bt)>0){

fos.write(bt);

        }

fos.close();

    }

out.println("<script>alert('file stored')</script>");

}

catch(Exception e){

out.println(e);

}

%>

Activate.jsp
```

```
<% @page import="java.sql.ResultSet"%>
<% @page import="java.util.Random"%>
<% @page import="pack.MailUtil"%>
<% @page import="java.sql.Statement"%>
<% @page import="java.sql.DriverManager"%>
<% @page import="java.sql.Connection"%>
<%
String id=request.getQueryString();Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/multiauthority","root","root");
//
System.out.println("Database connected");
Statement st=con.createStatement();
Random r=new Random();
int ran=r.nextInt();
String mes=ran+"";
int i=st.executeUpdate("update user set status='yes',product_id='"+mes+"' where iduser='"+id+"'");
ResultSet rt=st.executeQuery("select userid from user where iduser='"+id+"'");
    if(rt.next()){
String[] to=new String[]{rt.getString("userid")};
    new MailUtil().sendMail(to, to, "id",mes );
    }
    if(i!=0){
response.sendRedirect("users.jsp");
    }
else{
out.println("error" )%>
Connect.jsp<% @page import="sun.net.ftp.FtpClient"%>
<%
FtpClient client=new FtpClient();
```

```
client.connect("ftp.drivhq.com");  
  
client.login("root", "root");  
  
    is=new FileInputStream("");  
  
client.storeFile("store", is);  
  
is.close();  
  
client.logout();  
  
client.disconnect();
```

```
%>
```

```
KeySend.jsp
```

```
<% @page import="java.sql.ResultSet"%>
```

```
<% @page import="java.sql.Statement"%>
```

```
<% @page import="java.sql.DriverManager"%>
```

```
<% @page import="java.sql.Connection"%>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<title>CLOUD COMPUTING BASED SECURED APPLICATIONS FOR PLATFORMS BUILDING  
DEVELOPMENT</title>
```

```
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
```

```
<link href="style.css" rel="stylesheet" type="text/css" />
```

```
<!--<script type="text/javascript" src="js/cufon-yui.js"></script>
```

```
<script type="text/javascript" src="js/arial.js"></script>
```

```
<script type="text/javascript" src="js/cuf_run.js"></script>-->
```

```
<script type="text/javascript">
```

```
window.history.forward();
```

```
    function noBack()
```

```
window.history.forward();
```

```
    } function validate()
```

```
{
}
</script>
<style>
table,td,tr{
    border-style: solid;
    border-collapse: collapse;
}
</style>
</head>
<body onLoad="noBack();" onpageshow="if (event.persisted) noBack();" onUnload="">

<%
    String user,file,key,mail_status;
try{
    if(request.getParameter("msg")!=null){
out.println("<script>alert('key sent')</script>");
    }
    if(request.getParameter("msgg")!=null){
out.println("<script>alert('failed')</script>");
    }
//    Class.forName("org.sqlite.JDBC");
//    Connection con=DriverManager.getConnection("jdbc:sqlite:/home/ibn/Desktop/Nadanapathy/NetBeansProjects/multi-
//        authority_cloud_storage/multiauthority");
//
Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/multiauthority","root","root");
    Statement st=con.createStatement();
    Statement st1=con.createStatement();
ResultSet rt=st.executeQuery("select * from upload_file ");
%>
```

```
<div class="main">
<div class="header">
<div class="header_resize">
<div class="logo">
<h1>CLOUD COMPUTING BASED SECURED APPLICATIONS FOR PLATFORMS BUILDING
DEVELOPMENT</h1>
</div>
<div class="menu_nav" >
<ul>
<li><a href="aa_page.jsp" style="font-size: 20px">back</a></li>
<li><a href="index.html" style="font-size: 20px">Logout</a></li>
</ul>
</div>
<div class="clr"></div>
<div style="position: absolute;left:100px;top: 100px">
<h2>KEY SENDING</h2>
<form action="sendkey.jsp"method="post"name="name">
<table>
<!--<tr><td> CHOOSE FILE </td><td><input type="file" name="file"</td></tr>
<tr style="height: 15px"></tr>
-->
<tr style="background: scroll;font-size: 18px;background-color: #323a3f;">
<tdalign="center" >OWNER</td><td></td>
<tdalign="center">FILE</td><td></td>
<tdalign="center">KEY</td><td></td>
<tdalign="center">STATUS</td><td></td>
</tr>
<% while(rt.next()){
```

```
        user=rt.getString("userid");

        file=rt.getString("filename");

        key=rt.getString("skey");

mail_status=rt.getString("mail_status");

        %>

<tr>

<td ><%=user%></td><td></td>

<tdalign="center"><%=file%></td><td></td>

<tdalign="center" ><%=key%></td><td></td>

<tdalign="center" ><%=mail_status%></td><td></td></tr>

<%

}

con.close();

}

catch(Exception e){

out.println(e);

}

%>

</table>

<br></br>

<div style="position: absolute;left: 200px;">

<input type="submit" value="send" style="width: 100px;background-color: yellowgreen"></input>

</div>

</form>

</div>

<div class="clr"></div>

</div>

</div>

<div class="content">
```

<div class="content_resize">

<div class="mainbar">

</div>

<div class="sidebar">

<div class="gadget">

<div class="clr"></div>

</div>

</div>

<div class="clr"></div>

</div>

</div>

<div class="fbg">

<div class="fbg_resize">

<div class="clr"></div>

</div>

</div>

<div class="footer">

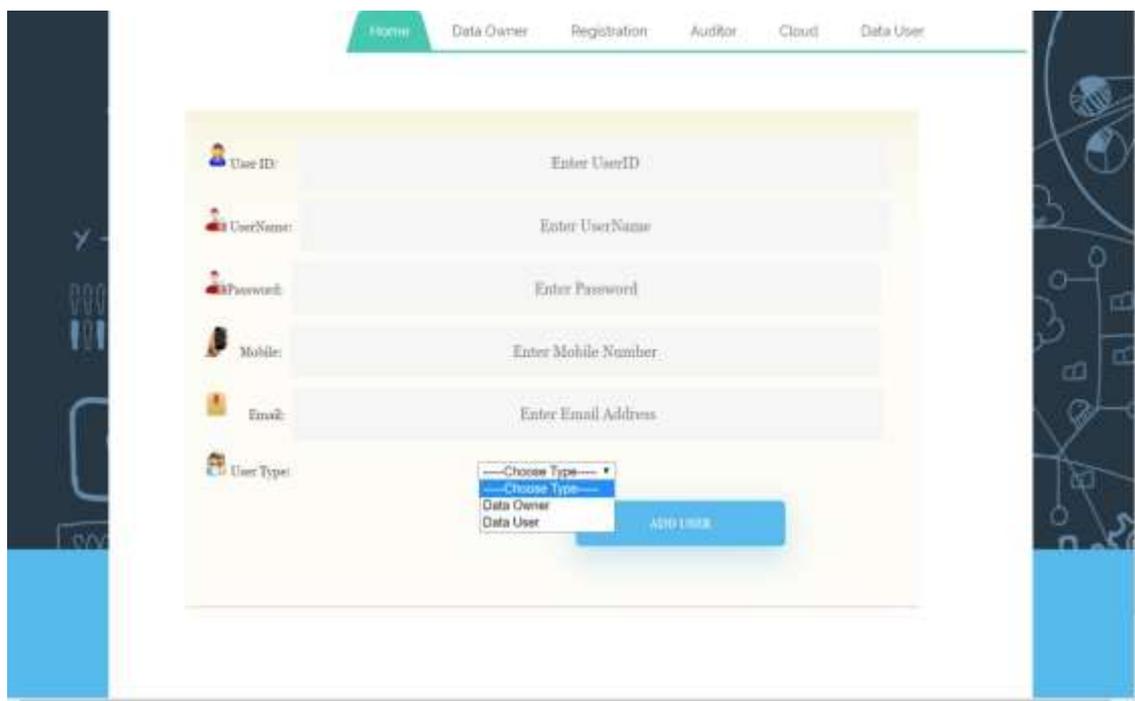
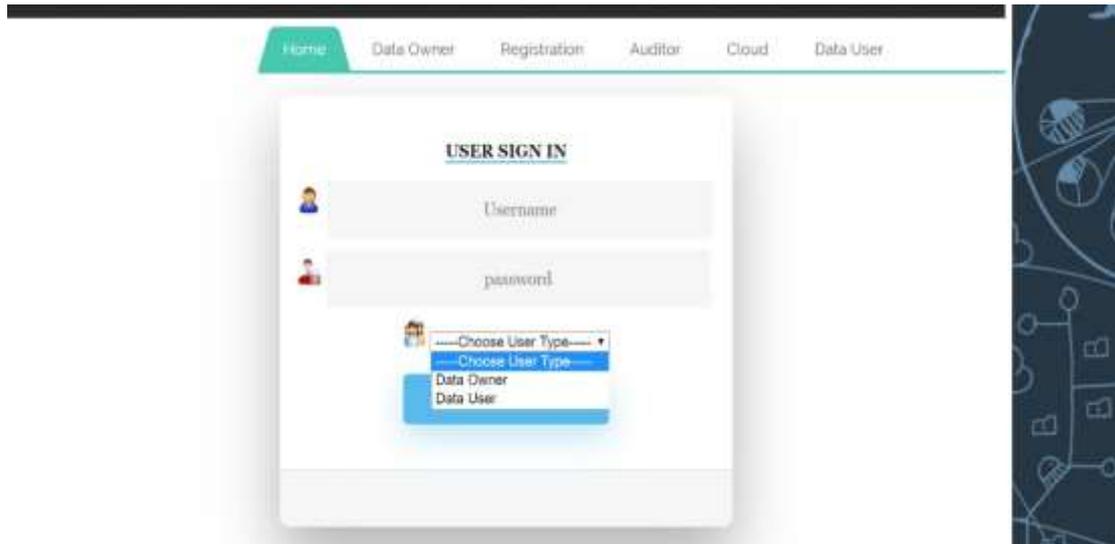
</div>

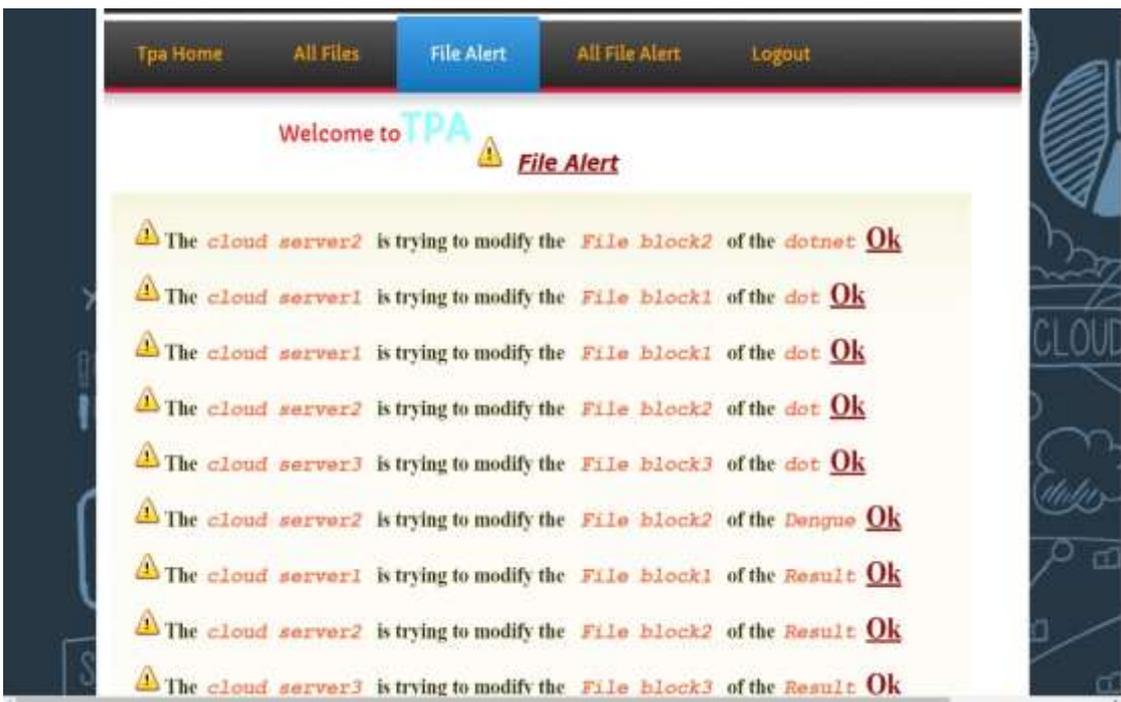
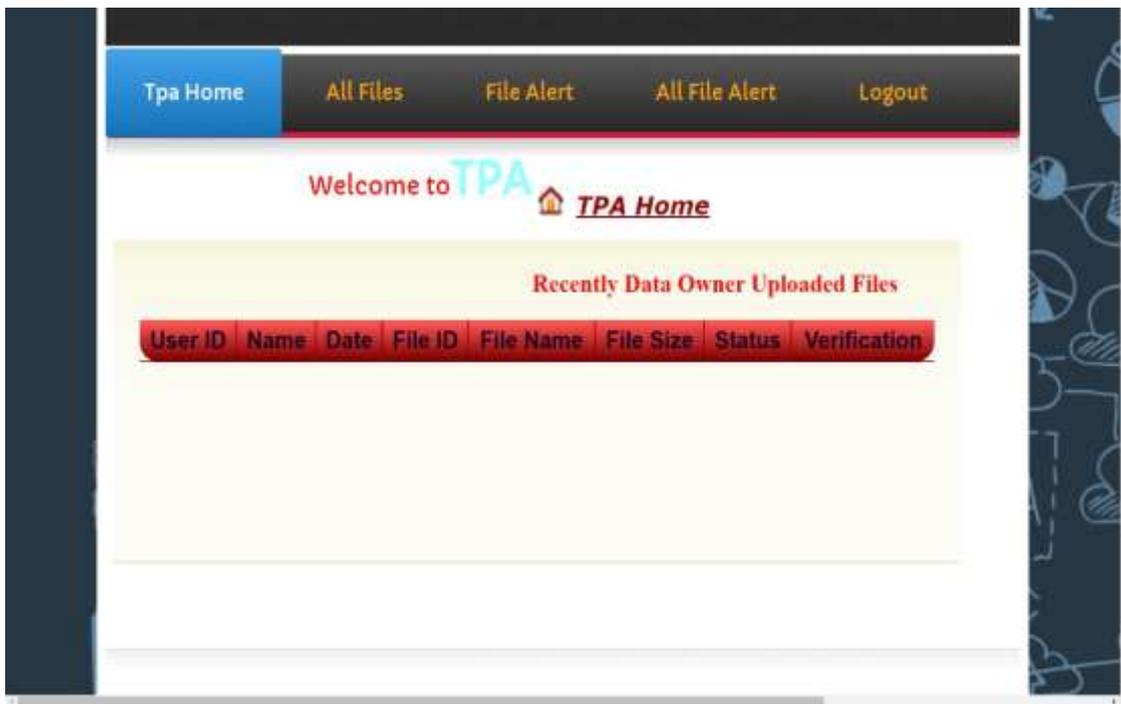
</div>

</body>

</html>

VI. Results





Welcome to **TPA**  **TPA ALL ALERTS**

User ID	Name	File ID	File Name	File Size	File Block	Trying To Modify
12	Ganeshkumar	11	jsp	4.279296875 KB	File block3	cloud server3
13	Balaji	10	aspx	0.935546875 KB	File block1	cloud server1
14	Kalaikumar	12	.net	0.9052734375 KB	File block1	cloud server1
13	Balaji	9	java	5.1552734375 KB	File block3	cloud server3
15	sunny	14	dotnet	0.287109375 KB	File block1	cloud server1
15	sunny	14	dotnet	0.287109375 KB	File block2	cloud server2
15	sunny	14	dotnet	0.287109375 KB	File block3	cloud server3
16	kumar	15	dot	0.287109375 KB	File block1	cloud server1
16	kumar	15	dot	0.287109375 KB	File block1	cloud server1
16	kumar	15	dot	0.287109375 KB	File block2	cloud server2
16	kumar	15	dot	0.287109375 KB	File block3	cloud server3
16	kumar	86	Dengue	0.5400390625 KB	File block2	cloud server2
16	kumar	87	HINIES	0.4697265625 KB	File block1	cloud server1
16	kumar	87	HINIES	0.4697265625 KB	File block2	cloud server2
16	kumar	20	rt	0.0283203125 KB	File block3	cloud server3
16	kumar	98	Result	0.873046875 KB	File block1	cloud server1
16	kumar	98	Result	0.873046875 KB	File block2	cloud server2
16	kumar	98	Result	0.873046875 KB	File block3	cloud server3
16	kumar	99	ww	0.873046875 KB	File block1	cloud server1
16	kumar	99	ww	0.873046875 KB	File block2	cloud server2
16	kumar	99	ww	0.873046875 KB	File block3	cloud server3
16	kumar	99	ww	0.873046875 KB	File block1	cloud server1

User Home File Request Download Files Logout

Welcome to **sunny**  **New File Request**

File ID :

User ID :

Access Rights :

```
mysql> select * from user;
```

uid	name	userid	pass	mobile	email	date	utype
15	sunny	sunny	sunny	9666563655	vadde.seetha@gmail.com	21/02/2018	User
16	kumar	kumar	kumar	9666563655	kumar@gmail.com	22/02/2018	Owner
17	raman	raman	raman	9666365256	vadde.seetha@gmail.com	15/09/2019	Owner
18	aditya	aditya	aditya	9888458233	ram.javaapps@gmail.com	15/09/2019	User
19	vamsee	1010	vamsee	9133181444	vamseekazes@gmail.com	23/11/2019	Owner
20	kiran	kiran	kiran	9133181444	kiranmayi@gmail.com	11/03/2020	Owner

```
mysql>
```



VII. CONCLUSION:

In this paper, we proposed a revocable multi-authority CPABE plot that can bolster productive property renouncement. At that point, we built a compelling information get to control scheme for multi-authority distributed storage frameworks. We likewise demonstrated that our plan was provable secure in the random prophet model. The revocable multi-authority CPABE is a promising method, which can be applied in any remote capacity frameworks and online interpersonal organizations and so forth.

VIII. REFERENCES:

- Wang, L., Zhou, G., Xu, Y., and Liu, M. (2012). An upgraded Pareto-based fake honey bee state calculation for the multi-objective adaptable employment shop booking. *The International Journal of Advanced Manufacturing Technology*, 60(9-12), 1111-1123.
- Babu, K. R., and Samuel, P. (2016). Upgraded honey bee state calculation for productive burden adjusting and planning for cloud. In *Innovations in bio-inspired computing and applications* (pp. 67-78). Springer, Cham.
- Kaur, An., and Kaur, B. (2016, October). Burden adjusting in errands utilizing bumble bee conduct calculation in distributed computing. In *2016 fifth International Conference on Wireless Networks and Embedded Systems (WECON)* (pp. 1-5). IEEE.
- Hashem, W., Nashaat, H., and Rizk, R. (2017). Bumble bee based burden adjusting in distributed computing. *KSI Transactions on Internet and Information Systems*, 11(12).
- Karaboga, D., and Basturk, B. (2007). An incredible and proficient calculation for numerical capacity improvement: counterfeit honey bee settlement (ABC) calculation. *Diary of worldwide streamlining*, 39(3), 459-471.
- Gupta, T., Handa, S. S., and Panda, S. A Survey on Honey Bee Foraging Behavior and Its Improved Load Balancing Technique.

AUTHOR PROFILE



M. Mohith Riteesh is a student of the Computer Science and Engineering Department at the KoneruLakshmaiah Education Foundation situated at Vaddeswarm, Guntur District.



N. Prudvi Krishna is a student of the Computer Science and Engineering Department at the KoneruLakshmaiah Education Foundation situated at Vaddeswaram, Guntur District.



Y.S.R. Harshit is a student of the Computer Science and Engineering Department at the KoneruLakshmaiah Education Foundation situated at Vaddeswaram, Guntur District.



Dr.M.S.Mekala working as Associate Professor in Computer Science and Engineering Department at the KoneruLakshmaiah Education Foundation situated at Vaddeswaram, Guntur District.