SPECIMEN

Faces in the Clouds: Time-Domain Attribute based access control for cloud-based video content sharing

A PROJECT REPORT

Submitted by

MOHANASREE.P [211417104148]

MAHALAKSHMI.D [211417104139]

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE, CHENNAI 600123.

ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2021

BONAFIDE CERTIFICATE

Certified that this project report **"Faces in the Clouds: Time-Domain Attribute based access control for cloud-based video content sharing"** is the bonafide work of **"MOHANASREE.P [211417104148], MAHALAKSHI.D [211417104139]''** who carried out the project work under my supervision.

SIGNATURE

Dr. S. MURUGAVALLI, M.E., Ph.D.,
HEADOFTHEDEPARTMENT

DEPARTMENT OF CSE,
PANIMALAR ENGINEERING COLLEGE,
NAZARATHPETTAI,
POONAMALLEE,
CHENNAI-600 123.

SIGNATURE

Ms. Maheshwari,
Assistant Professor (Gr-18)

DEPARTMENT OF CSE,
PANIMALAR ENGINEERING
COLLEGE, NAZARATHPETTAI,
POONAMALLEE,
CHENNAI-600 123.

Certifiedthattheabovecandidate(s)was/wereexaminedintheA

nnaUniversityProjectViva- Voce Examination held

on.....

INTERNALEXAMINER

EXTERNALEXAMINER

ACKNOWLEDGEMENT

We express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We would like express our heartfelt and sincere thanks to our Directors **Tmt.C.VIJAYARAJESWARI**, **Dr.C.SAKTHIKUMAR, M.E., Ph.D.,** and **Tmt. SARANYASREE SAKTHIKUMAR B.E.,M.B.A.,** for providing us with the necessary facilities for completion of this project.

We also express our gratitude to our Principal **Dr.K.MANI, M.E., Ph.D.** for his timely concern and encouragement provided to us throughout the course.

We thank the HOD of CSE Department, **Dr. S.MURUGAVALLI , M.E.,Ph.D.,** for the support extended throughout the project.

We would like to thank my **Project Guide Ms. Maheshwari** and all the faculty members of the Department of CSE for their advice and suggestions for the successful completion of the project.

NAME OF THE STUDENTS
P. MOHANASREE
D. MAHALAKSHMI

ABSTRACT

Media streaming applications have recently attracted a large number of users on the Internet. With the advent of these bandwidth-intensive applications, it is economically inefficient to provide streaming distribution with guaranteed QoS relying only on central resources at a media content provider. Cloud computing offers an elastic infrastructure that media content providers (e.g., Video on Demand (VoD) providers) can use to obtain streaming resources that match the demand. Media content providers are charged for the number of resources allocated (reserved) in the cloud. Most of the existing cloud providers employ a pricing model for the reserved resources that is based on non-linear time-discount tariffs (e.g., Amazon CloudFront and Amazon EC2). Such a pricing scheme offers discount rates depending non-linearly on the period of time during which the resources are reserved in the cloud. In this case, an open problem is to decide on both the right amount of resources reserved in the cloud, and their reservation time such that the financial cost on the media content provider is minimized. We propose a simple - easy to implement - algorithm for resource reservation that maximally exploits discounted rates offered in the tariffs, while ensuring that sufficient resources are reserved in the cloud. Based on the prediction of demand for streaming capacity, our algorithm is carefully designed to reduce the risk of making wrong resource allocation decisions. The results of our numerical evaluations and simulations show that the proposed algorithm significantly reduces the monetary cost of resource allocations in the cloud as compared to other conventional schemes.

List of figures: **Pg. No.**

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1. OVERVIEW:

Cloud computing is the delivery of on-demand computing services from applications to storage and processing power typically over the internet and on a pay-as-you-go basis. Rather than owning their own computing infrastructure or data centres, companies can rent access to anything from applications to storage from a cloud service provider. One benefit of using cloud computing services is that firms can avoid the upfront cost and complexity of owning and maintaining their own IT infrastructure, and instead simply pay for what they use, when they use it. In turn, providers of cloud computing services can benefit from significant economies of scale by delivering the same services to a wide range of customers.

Several computing paradigms have remarkably changed the way of "web surfing" experience in the past two decades. Grid, cloud, fog, and edge computing represent the key pillars of this evolution. With incorporation of smart phones and high-speed communications, inter-network access has reached a new level of sophistication. Multiple cloud domains and fog services are currently engaged in providing a required set of data or information to its users or customers. Favourable aggregations of service-oriented aspects are presently acting as the basis of such interventions. Users are indebted towards data storage, analysis, visualization, computation, and persuasion as per the prescribed notions of the cloud, fog or edge vendors. Subsequently, network users are getting heavily dependent on the availability of internet connectivity to persuade for the "opted" jobs.

1.2. PROBLEM DEFINITION:

This project studies the problem of resource allocations in the cloud for media streaming applications. We have considered non-linear time-discount tariffs that a cloud provider charges for resources reserved in the cloud.

We develop a resource allocation system that can avoid overload in the system effectively while minimizing the number of servers used.

CHAPTER 2

LITERATURE SURVEY

[1] Shan-Hsiang Shen, proposed a paper titled “Efficient SVC Multicast Streaming for Video Conferencing With SDN Control”. In this paper they have proposed a novel SVC multicast streaming scheme named adaptive SDN-based SVC multicast (ASCast). Each video layer forms a multicast tree, and they formulate a linear programming problem for the tree construction. To address the problem, we design static and dynamic heuristic algorithms to build multicast trees and maximize overall video quality with limited TCAM space. Moreover, to reduce TCAM space consumption, they carefully consider multicast integer programming address assignment for video layers and forwarding rule installation. Based on our evaluation, ASCast provides a 35% higher video data rate and installs 66% fewer forwarding rules into switches than other SVC video multicast schemes.

[2] Mohammad H. Hajiesmaili, Lok To Mak, Zhi Wang, Chuan Wu, Minghua Chen, and Ahmad Khonsari, published a paper called “Cost-Effective Low-Delay Design for Multi-Party Cloud Video Conferencing”. This paper discusses a joint problem of user-to-agent assignment and transcoding-agent selection. The ultimate objective is to simultaneously minimize the cost of the service provider and the conferencing delay. The problem is combinatorial in nature which belongs to the NP-hard node assignment problems. They leverage the Markov approximation framework and devise an adaptive parallel algorithm that finds a close-to-optimal solution to our problem with a bounded performance guarantee. To evaluate the performance of our solution, they implement a prototype video conferencing system, and carry out trace-driven experiments. In a set of large-scale experiments using PlanetLab traces, our

solution decreases the operational cost by 77% and simultaneously yields lower conferencing delay compared to an existing alternative.

[3] Deming Zhai, Xianming Liu, Xiangyang Ji, published a paper titled “Joint Gaze Correction and Face Beautification for Conference Video using Dual Sparsity Prior”. This paper proposes to jointly solve the hole-filling problem and the face beautification problem (subtle modifications of facial components and contour to enhance attractiveness of the rendered face) using dual sparsity prior. Specifically, prior to the start of a video conference session, they first train two dictionaries separately offline using two large datasets: one with general face images, the other with “beautiful” human faces, which means faces with high beauty scores. During the actual conference session, they solve the hole-filling and facial components beautification problems simultaneously by seeking two code vectors—one is sparse in the first dictionary and explains the available DIBR-synthesized pixels, the other is sparse in the second dictionary and matches well with the first vector in terms of feature space distance. This ensures an acceptable level of recognizability of the conference subject, while increases proximity to “beautiful” facial features to improve attractiveness. Experimental results show naturally rendered human faces with noticeably improved attractiveness.

[4] Yuanhuan Zheng, Di Wu, Yihao Ke, Can Yang proposed a paper titled “Online Cloud Transcoding and Distribution for Crowdsourced Live Game Video Streaming”. In this paper, they address the problem of cost-effective adaptive live game video streaming from the perspective of CLGVS service providers. Their purpose is to minimize the operational cost for CLGVS service providers by making live transcoding decisions, bit-rate adaptation decisions and datacenter assignment decisions dynamically. Meanwhile, the proposed algorithm also ensures good-enough service quality for viewers. Due to the diversity of game genres, we also take game genres into account when

designing our algorithm. To achieve the above purpose, we formulate the problem into a constrained stochastic optimization problem. By leveraging the Lyapunov optimization framework, we derive the online strategy with provable performance bound. To evaluate the effectiveness of the proposed algorithm, we further conduct a series of trace-driven simulations. The experimental results demonstrate the effectiveness of the algorithm in terms of operational cost and service quality. The proposed algorithm can reduce operational cost by up to 50% while achieving goodenough viewer QoE compared with other alternatives.

[5] Xiangbo Li, Mohsen Amini Salehi, Magdy Bayoumi, proposed a paper titled "Cost-Efficient and Robust On-Demand Video Stream Transcoding Using Heterogeneous Cloud Services". In this paper they proposed an idea to transcode them in an on-demand (i.e., lazy) manner using cloud computing services. The challenge in utilizing cloud services for on-demand video transcoding, however, is to maintain a robust QoS for viewers and cost-efficiency for streaming service providers. To address this challenge, in this paper, they present the Cloud-based Video Streaming Services (CVS2) architecture. It includes a QoS-aware scheduling component that maps transcoding tasks to the Virtual Machines (VMs) by considering the affinity of the transcoding tasks with the allocated heterogeneous VMs. To maintain robustness in the presence of varying streaming requests, the architecture includes a cost-efficient VM Provisioner component. The component provides a self-configurable cluster of heterogeneous VMs. The cluster is reconfigured dynamically to maintain the maximum affinity with the arriving workload. Simulation results obtained under diverse workload conditions demonstrate that CVS2 architecture can maintain a robust QoS for viewers while reducing the incurred cost of the streaming service provider by up to 85%.

CHAPTER 3

SYSTEM ANALYSIS

3.1. EXISTING SYSTEM

Virtual machine monitors (VMMs) like Xen provide a mechanism for mapping virtual machines (VMs) to physical resources. This mapping is largely hidden from the cloud users. Users with the Amazon EC2 service, for example, do not know where their VM instances run. It is up to the cloud provider to make sure the underlying physical machines (PMs) have sufficient resources to meet their needs. VM live migration technology makes it possible to change the mapping between VMs and PMs While applications are running. The capacity of PMs can also be heterogeneous because multiple generations of hardware coexist in a data centre.

DISADVANTAGES:

A policy issue remains as how to decide the mapping adaptively so that the resource demands of VMs are met while the number of PMs used is minimized. This is Challenging When the resource needs of VM's are heterogeneous due to the diverse set of applications they run and vary with time as the workloads grow and shrink. The two main disadvantages are overload and green computing.

3.2. PROPOSED SYSTEM

We have proposed algorithms that optimally determine both the amount of reserved resources in the cloud and their reservation time - based on prediction of future demand for streaming capacity – such that the financial cost on the media content provider is minimized. The proposed algorithms exploit the time discounted rates in the tariffs, while ensuring that sufficient resources are reserved in the cloud without incurring wastage. We have evaluated the performance of our algorithms numerically and using simulations. The results show that our algorithms adjust the trade-off between resources reserved on the

cloud and resources allocated on-demand. In future work, we shall perform experimental measurements to characterize the streaming demand in the Internet and develop our own demand forecasting module. We shall also investigate the case of multiple cloud providers and consider the market competition when allocating resources in the clouds.

ADVANTAGES:

Structured Meetings with Improved Communications

Reduced Travel Time and Costs

Increased Productivity

3.3. TECHNOLOGY STACK:

3.3.1. H/W SYSTEM CONFIGURATION:

Processor	-	Pentium –III
RAM	-	4 GB
Hard Disk	-	260 GB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA

3.3.2. S/W SYSTEM CONFIGURATION:

Operating System	- Windows95/98/2000/XP
Front End	- HTML, Java, Jsp
Scripts	- JavaScript.
Server-side Script	- Java Server Pages.
Database	- My sql
Database Connectivity	- JDBC.

CHAPTER 4

SYSTEM DESIGN

4.1. ER DIAGRAM:

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties.

By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases.
ER diagrams are used to sketch out the design of a database.

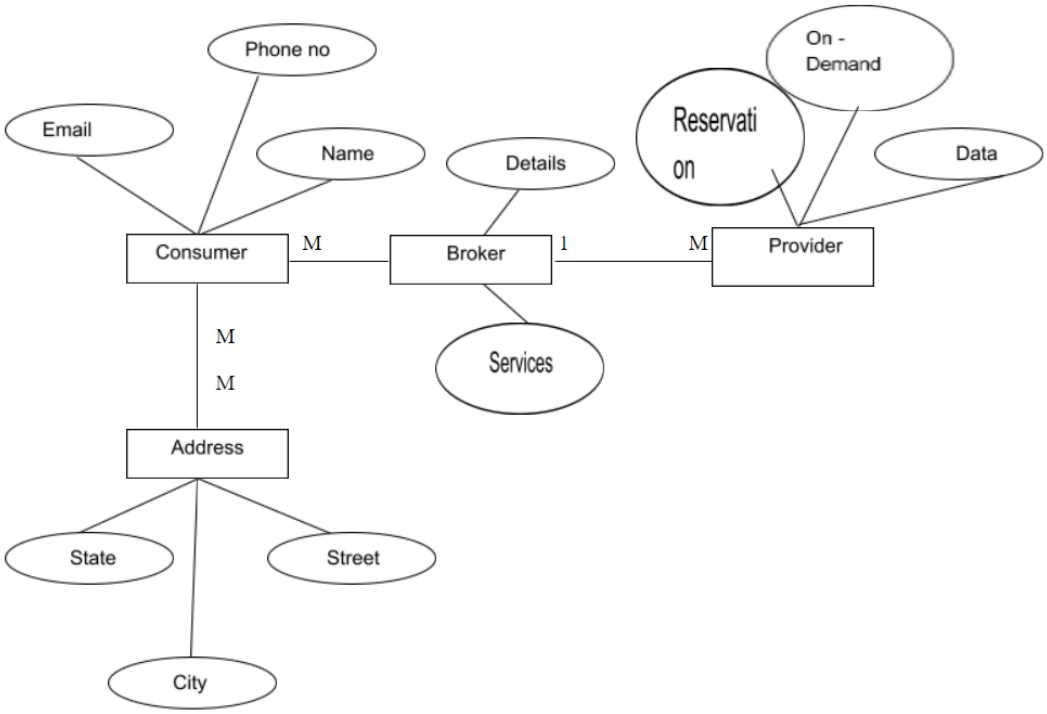

Fig 1: ER Diagram

4.2. USE CASE DIAGRAM:

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

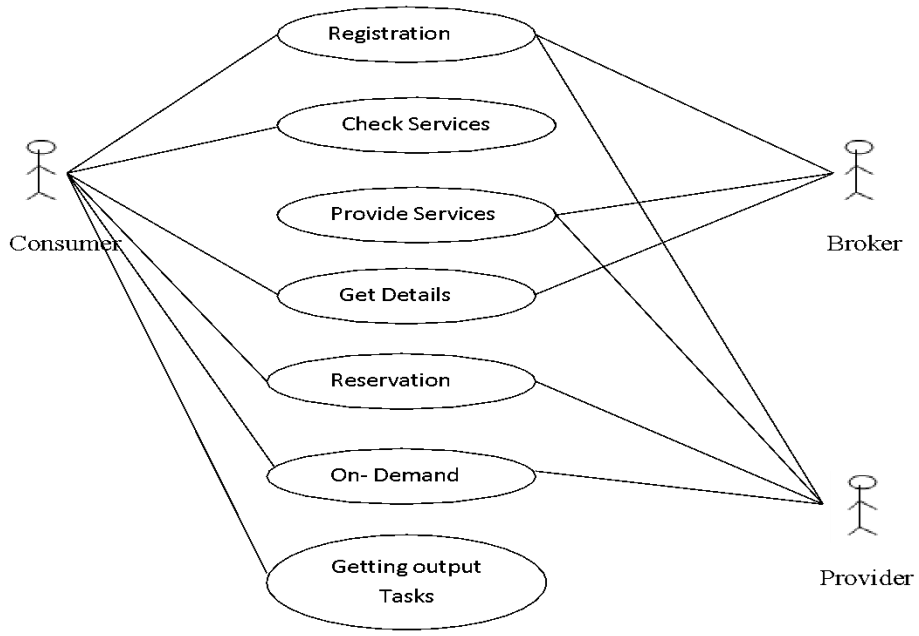

Fig 2: Use Case Diagram

4.3. UML DIAGRAM:

The Unified Modeling Language is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.

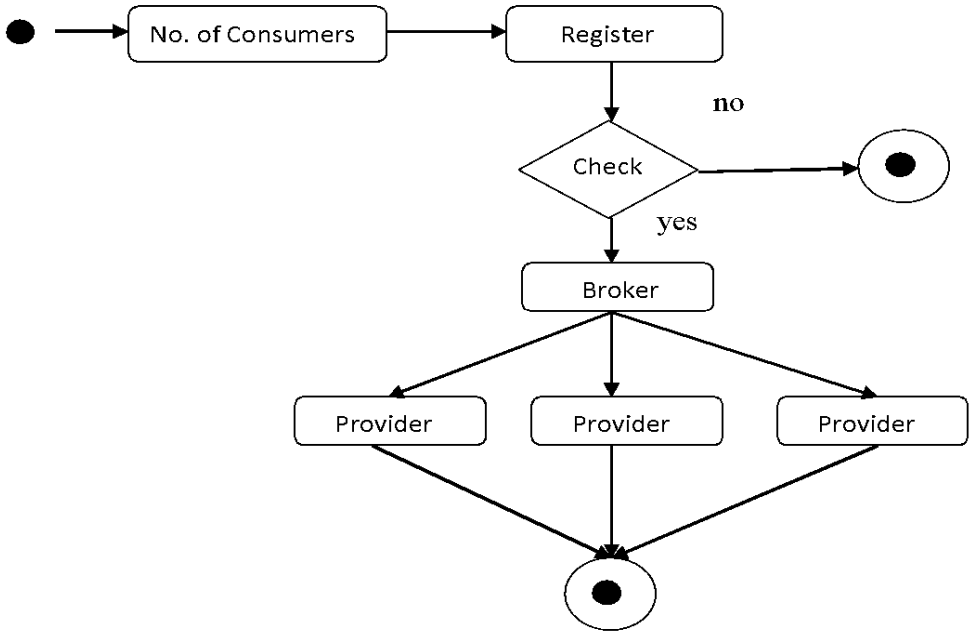

Fig 3: UML Diagram

CHAPTER 5

SYSTEM ARCHITECTURE

5.1. ARCHITECTURE OVERVIEW

We have proposed algorithms that optimally determine whether both the amount of reserved resources in the cloud and their reservation time is based on the prediction of future demand for streaming capacity and the financial cost on the media content provider is minimized. The algorithms are proposed in such a way that it makes use of the discounted time rates in the tariffs, while ensuring that sufficient resources are reserved in the cloud without incurring wastage. The evaluation of the performance of our algorithms is done numerically and also using simulations. Structured meetings with improved communications, reduced travel time and costs and increased productivity are few advantages of the proposed system.

The figure 4 shows the architecture of the proposed work. From the cloud consumer a request signal sent to the cloud broker via virtual machine repository. Then from the cloud broker the signal is passed on to the cloud providers. Now a respond to the request signal is sent back to the consumer from the cloud provider.

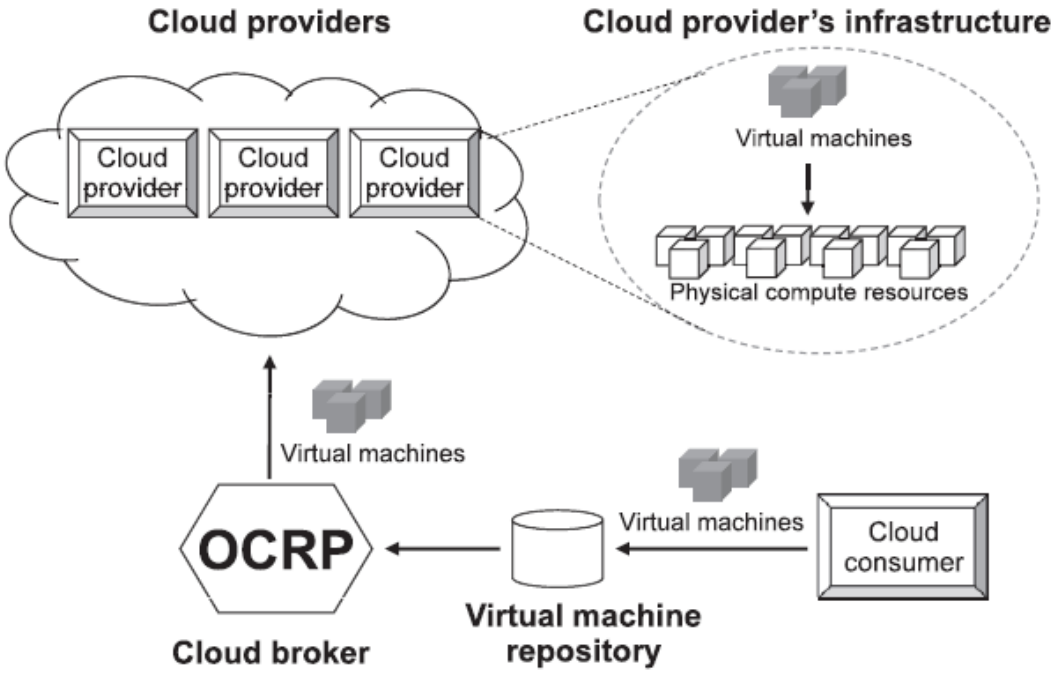

Fig 4: Architecture diagram of Proposed System

5.2. MODULES DESIGN SPECIFICATION:

5.2.1. Resource Provisioning (RP):

An optimal cloud resource provisioning is proposed by formulating a stochastic programming model. RP algorithm can provision computing resources for being used in multiple provisioning stages as well as a long-term plan, e.g., four stages in a quarter plan and twelve stages in a yearly plan. The demand and price uncertainty are considered in RP. In particular, an optimal cloud resource provisioning is proposed to minimize the total cost for provisioning resources in a certain time period. To make an optimal decision, the demand uncertainty from cloud consumer side and price uncertainty from cloud providers are taken into account to adjust the trade-off between on-demand and oversubscribed costs.

5.2.2. Overload avoidance:

An important issue when operating a load-balanced service is how to handle information that must be kept across the multiple requests in a user's session. If this information is stored locally on one backend server, then subsequent requests going to different backend servers would not be able to find it. This might be cached information that can be recomputed, in which case load-balancing a request to a different backend server just introduces a performance issue.

A variety of scheduling algorithms are used by load balancers to determine which backend server to send a request to. Simple algorithms include random choice or round robin. More sophisticated load balancers may take into account additional factors, such as a server's reported load, recent response times, up/down status (determined by a monitoring poll of some kind), number of active connections, geographic location, capabilities, or how much traffic it has recently been assigned.

5.2.3. Minimizing skewness:

Application delivery controllers have evolved from basic server load balancing functional units to fully integrate with cloud workflows and provisioning systems so that they help users enable fast roll-out of new applications to a mobilized work force, improve end-user satisfaction, and reduce the time and cost of application deployment We introduce the concept of skewness to quantify the unevenness in the utilization of multiple resources on a server.

5.2.4. VM migrations:

We aim to migrate away the VM that can reduce the server's Usage the most. In case of ties, we select the VM whose removal can reduce the skewness of the server the most. For each VM in the list, we see if we can find a destination server to accommodate it. The server must not become a hot spot after accepting this VM. Among all such servers, we select one whose skewness can be reduced the most by accepting this VM. Note that this reduction can be negative which means we select the server whose skewness increases the least. If a destination server is found, we record the migration of the VM to that server and update the predicted load of related servers. Otherwise, we move onto the next VM in the list and try to find a destination server for it. As long as we can find a destination server for any of its VMs, we consider this run of the algorithm a success and then move onto the next hot spot. Note that each run of the algorithm migrates away at most one VM from the overloaded server. This does not necessarily eliminate the hot spot, but at least reduces its temperature. If it remains a hot spot in the next decision run, the algorithm will repeat this process. It is possible to design the algorithm so that it can migrate away multiple VMs during each run. But this can add more load on the related servers during a period when they are already overloaded.

5.3. Program Design Language:

Algorithm 1 MinCSS($ST_x, UL_{x,t}$)

```
1:  X_e, X_r ← ∅
2:  N_{x,t} ← ∅
3:  u_e ← the most left empty leaf node
4:  X_e ← Path(u_e)
5:  for each u_r ∈ UL_{x,t} do
6:      add Path(u_r) to X_r
7:  X_r ← X_r \ X_e
8:  for each v_e ∈ X_e do
9:      if v_e is not a leaf node then
10:         v_lc ← left child of v_e
11:         if v_lc ∉ X_r ∪ X_e then
12:             add v_lc to N_{x,t}
13: for each v_r ∈ X_r do
14:     if v_r is not a leaf node then
15:         v_lc ← left child of v_r
16:         if v_lc ∉ X_r then
17:             add v_lc to N_{x,t}
18:         v_rc ← right child of v_r
19:         if v_rc ∉ X_r then
20:             add v_rc to N_{x,t}
21: if N_{x,t} = ∅ then
22:     add the root node x to N_{x,t}
23: Return N_{x,t}
```

Algorithm:

The CP-ABE scheme consists of four operations:

Setup, KeyGen, Encrypt and Decrypt. It is described as follows:

1) (PK;MSK) ← Setup(1). The probabilistic operation takes a security parameter as input and outputs public key PK and master secret key MSK.

2) (SK) ← KeyGen(PK;MSK; S). The operation inputs PK, MSK and a set of attributes S and creates a secret key SK.

3) (CT) ← Encrypt(PK; ck;A). The operation inputs PK, ck = {ck1; : : : ; ckk} and a hierarchical access tree A as shown in the Fig. 2. At last, it creates an integrated
ciphertext of content keys CT.

4) (cki(i ∈ [1; k])) ← Decrypt(PK;CT; SK). The algorithm inputs PK, CT which includes an integrated access structure A, SK described by a set of attributes

S. If the S matches part of A, some content keys cki(i ∈ [1; k]) can be decrypted. If it matches the whole A, all the content keys can be decrypted. Then, the corresponding files mi(i ∈ [1; k]) will be decrypted with the content keys by the symmetric decryption algorithm.

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1. CLIENT-SIDE CODING:

```
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.Date"%>
<%@page import="java.text.SimpleDateFormat"%>
<%@page import="java.text.DateFormat"%>
<!DOCTYPE html>
<html>
<head>
<script>
function validateTime()
{
  var txt1=document.getElementById("text1").value;
  var txt2=document.getElementById("text2").value;
  if(txt1==="")
  {
    alert("Error: uid cannot be blank!");
    return false;
  }
  if(txt2==="")
  {
    alert("Error: pwd cannot be blank!");
    return false;
  }
  else{return  true;}
}
</script>
<title>Free Retail Hosting Website Template | Hosting ::
w3layouts</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
<link href="css/style.css" rel="stylesheet" type="text/css"
media="all"/>
<link href='http://fonts.googleapis.com/css?family=Electrolize'
rel='stylesheet' type='text/css'>
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript">
jQuery(document).ready(function($)
{
  $(".scroll").click(function(event)
  {
    event.preventDefault();

$('html,body').animate({scrollTop:$(this.hash).offset().top},1200);
                });
          });
</script>
</head>
```

```
<body>
<div class="header">
<div class="header_top">
<div style="font-size: 25px;color: white;margin-left:
120px">Innovative Schemes for Resource Allocation
in the Cloud for Media Streaming Applications</div>
<div class="wrap">
<div class="menu">
<ul>
<li class="active"><a
href="Admin_Home.jsp">Home</a></li>
<li><a href="Admin_Login.jsp">Logout</a></li>
<div class="clear"></div>
</ul>
</div>
<div class="clear"></div>
</div>
</div>
<div class="main">
<div class="services_heading" style="margin-right: 200px">
<p>Admin Home</p>
</div>
<div class="wrap">
<div class="services">
<div style="margin-left:100px;height: 200px;">
<%Integer hitsCount =
      (Integer)application.getAttribute("hitCounter");
 if(hitsCount<=0)
 {
 response.sendRedirect("Home.jsp");
 }
 if(hitsCount<5){
 %>
 <div style="font-size: 20px;color: #6fa803;margin-left:
300px">Running From Server-1 </div>
 <div style="font-size: 20px;color: #6fa803;margin-left:
300px">Number Of Users :<%=hitsCount%></div>
<%
}
if(hitsCount>=5 && hitsCount<10)
{
%>
<div style="font-size: 20px;color: #6fa803;margin-left:
300px">Running From Server-2 </div>
<div style="font-size: 20px;color: #6fa803;margin-left:
300px">Number Of Users : <%=hitsCount%></div>
<%
}
if(hitsCount>10){
%>
<div style="font-size: 20px;color:#6fa803;margin-left:
300px">Running From Server-3 </div>
<div style="font-size: 20px;color: #6fa803;margin-left:
300px">Number Of Users : <%=hitsCount%></div>
<%
}
%></div>
```

```
</div>
</div>
</div>
<div class="copy_right">
<p>Company Name Â© All rights Reseverd | Design by <a
href="http://w3layouts.com"> W3Layouts </a></p>
</div>
</body>
</html>
```

6.2. SERVER – SIDE CODING:

6.2.1. ADMIN LOGIN:

```
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.Date"%>
<%@page import="java.text.SimpleDateFormat"%>
<%@page import="java.text.DateFormat"%>
<!DOCTYPE html>
<html>
<head>
<script>
function validateTime() {
var txt1=document.getElementById("text1").value;
var txt2=document.getElementById("text2").value;
if(txt1===""){
    alert("Error: uid cannot be blank!");
    return false;
}
if(txt2===""){
    alert("Error: pwd cannot be blank!");
    return false;
}
else{return  true;}
}
</script>
<title>Free Retail Hosting Website Template | Hosting ::
w3layouts</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
<link href="css/style.css" rel="stylesheet" type="text/css"
media="all"/>
<link href='http://fonts.googleapis.com/css?family=Electrolize'
rel='stylesheet' type='text/css'>
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript">
jQuery(document).ready(function($) {
$(".scroll").click(function(event){
                      event.preventDefault();
     $('html,body').animate({scrollTop:$(this.hash).offset().top},1
200);
                });
          });
```

```
</script>
</head>
<body>
<div class="header">
<div class="header_top">
<div style="font-size: 25px;color: white;margin-left: 120px"
>Innovative Schemes for Resource Allocation
in the Cloud for Media Streaming Applications</div>
<div class="wrap">
<div class="menu">
<ul>
<li class="active"><a href="index.jsp">Home</a></li>
<li><a href="Login.jsp">Login</a></li>
<li><a href="Register.jsp">Register</a></li>
<li><a href="Login_Provider.jsp">Provider</a></li>
<li><a href="">Contact</a></li>
<div class="clear"></div>
</ul>
</div>
<div class="clear"></div>
</div>
</div>
<div class="main">
<div class="services_heading" style="margin-right: 200px">
<p>Admin Login</p>
</div>
<div class="wrap">
<div class="services">
<div style="margin-left:350px;height: 200px;margin-top: 100px">
<%String msg=request.getParameter("msg");
if(msg!=null)
    {
%>
<blink><font size="4" color="red"><%=msg%></font></blink> <br>
<%} %><br>
<form name="form" action="login_1_1.jsp" method="post"
onsubmit="return validateTime();">
<label for="text1" style="color: white">Username: </label> <input
id="text1" type="text" name="txt1" value="" /><br /> <br />
<label for="text2" style="color: white">Password  
:</label><input id="text2" type="password" name="txt2" value="" />
<br />     <br />
          <input
type="submit" value="Login" />
</form>
</div>
</div>
</div>
</div>
<div class="copy_right">
<p>Company Name Â© All rights Reseverd | Design by  <a
href="http://w3layouts.com"> W3Layouts </a></p>
</div>
</body>
</html>
```

6.2.2. LOGIN PROVIDER:

```
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.Date"%>
<%@page import="java.text.SimpleDateFormat"%>
<%@page import="java.text.DateFormat"%>
<!DOCTYPE html>
<html>
<head>
<script>
function validateTime() {
var txt1=document.getElementById("text1").value;
var txt2=document.getElementById("text2").value;
if(txt1===""){
    alert("Error: uid cannot be blank!");
    return false;
}
if(txt2===""){
    alert("Error: pwd cannot be blank!");
    return false;
}
else{return  true;}
}
</script>
<title>Free Retail Hosting Website Template | Hosting ::
w3layouts</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
<link href="css/style.css" rel="stylesheet" type="text/css"
media="all"/>
<link href='http://fonts.googleapis.com/css?family=Electrolize'
rel='stylesheet' type='text/css'>
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript">
jQuery(document).ready(function($) {
$(".scroll").click(function(event){
                event.preventDefault();
$('html,body').animate({scrollTop:$(this.hash).offset().top},1200);
                });
          });
</script>
</head>
<body>
<div class="header">
<div class="header_top">
<div style="font-size: 25px;color: white;margin-left: 120px"
>Innovative Schemes for Resource Allocation
in the Cloud for Media Streaming Applications</div>
<div class="wrap">
<div class="menu">
<ul>
<li><a href="index.jsp">Home</a></li>
<li><a href="Login.jsp">Login</a></li>
<li><a href="Register.jsp">Register</a></li>
<li class="active"><a href="Login_Provider.jsp">Provider</a></li>
<li><a href="">Contact</a></li>
```

```
<div class="clear"></div>
</ul>
</div>
<div class="clear"></div>
</div>
</div>
<div class="main">
<div class="services_heading" style="margin-right: 200px">
<p>Provider Login</p>
</div>
<div class="wrap">
<div class="services">
<div style="margin-left:350px;height: 200px;margin-top: 100px">
<form name="form" action="login_pro.jsp" method="post"
onsubmit="return validateTime();">
<label for="text1" style="color: white">Username: </label> <input
id="text1" type="text" name="txt1" value="" /><br /> <br />
<label for="text2" style="color: white">Password  
:</label><input id="text2" type="password" name="txt2" value="" />
 <br />     <br />
          <input
type="submit" value="Login" />
</form>
</div>
</div>
</div>
</div>
<div class="copy_right">
<p>Company Name Ã© All rights Reseverd | Design by  <a
href="http://w3layouts.com"> W3Layouts </a></p>
</div>
</body>
</html>
```

6.2.3. PROVIDER HOME:

```
<%@page import="java.io.InputStream"%>
<%@page import="java.io.FileInputStream"%>
<%@page import="java.io.File"%>
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.Date"%>
<%@page import="java.text.SimpleDateFormat"%>
<%@page import="java.text.DateFormat"%>
<!DOCTYPE html>
<html>
<head>
<script>
function validateTime() {
var txt1=document.getElementById("text1").value;
var txt2=document.getElementById("text2").value;
if(txt1===""){
    alert("Error: uid cannot be blank!");
    return false;
}
if(txt2===""){
    alert("Error: pwd cannot be blank!");
```

```
return false;
}
else{return  true;}
}
</script>
<title>Free Retail Hosting Website Template | Hosting ::
w3layouts</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
<link href="css/style.css" rel="stylesheet" type="text/css"
media="all"/>
<link href='http://fonts.googleapis.com/css?family=Electrolize'
rel='stylesheet' type='text/css'>
<script type="text/javascript" src="js/jquery.min.js"></script>
          <script type="text/javascript">
                    jQuery(document).ready(function($) {
               $(".scroll").click(function(event){
                    event.preventDefault();

     $('html,body').animate({scrollTop:$(this.hash).offset().top},1
200);
               });
          });
</script>
</head>
<body>
<div class="header">
<div class="header_top">
<div style="font-size: 25px;color: white;margin-left: 120px"
>Innovative Schemes for Resource Allocation
in the Cloud for Media Streaming Applications</div>
<div class="wrap">
<div class="menu">
<ul>
<li class="active"><a href="Provider_Home.jsp">Home</a></li>
<li><a href="Provider_Play.jsp">Channel Details</a></li>
<li><a href="Provider_Upload.jsp">Upload Videos</a></li>
<li><a href="Provider_Play_1.jsp">View Stream</a></li>
<li><a href="Logout.jsp">Logout</a></li>
<div class="clear"></div>
</ul>
</div>
<div class="clear"></div>
</div>
</div>
<div class="main">
<div class="services_heading" style="margin-right: 200px">
<p>Provider Home</p>
</div>
<div class="wrap">
<div class="services">
<div style="margin-left: 150px">
<form name="f1">
<%
String st=request.getParameter("s1");
```

```
String provider=(String)session.getAttribute("user");
if(provider != null){
if(provider.equals("provider1")){
provider="access";
}
if(provider.equals("provider2")){
provider="server1";
}
if(provider.equals("provider3")){
provider="server2";
}
if(provider.equals("provider4")){
provider="spam";
}}
DateFormat df = new SimpleDateFormat("yy-MM-dd");
String formattedDate = df.format(new Date());
Class.forName("com.mysql.jdbc.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/resource","
root","admin");
Statement stmt=con.createStatement();
Statement stmt1=con.createStatement();
ResultSet rsx=stmt.executeQuery("select uid from admin where
d_provider='"+provider+"'");
%>
<div style="margin-left: 300px">
<h4><font color='white'>Select User</font></h4>
<select name="s1" onChange="document.f1.submit()">
<option>SELECT </option>
<%
  while(rsx.next())
   {
    String x=rsx.getString(1);
   %>
<option <%
   if(st !=null && st.equals(x))
   out.println(x);
      %>
    ><%=x%></option>
<%
   }
   %>
</select><br>
</div>
  <%
     if(st !=null)
       {

            Statement stmt11;

        try {
          Class.forName("com.mysql.jdbc.Driver");
         con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/resource","
root","admin");
```

```
        String qry="select * from admin where uid='"+st+"' and
date1>'"+formattedDate+"'";
        stmt = con.createStatement();
        ResultSet  rst = stmt.executeQuery(qry);
        if(rst.next())
          {
            String DB=rst.getString("d_provider");
            String tbl=rst.getString("u_tab");
           Connection conn = null;
            Statement st2=null;
           long filesizeInKB=0;
           String filename=request.getParameter("file");
           File file = new File("C:\\Documents and
Settings\\vinoth.TN\\Desktop\\"+filename);
            long filesize = file.length();
            filesizeInKB = filesize / 1024;
long tot_db_len= Integer.parseInt(rst.getString("d_size")) ;
 try {
      Class.forName("com.mysql.jdbc.Driver");
         conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/sns","root"
,"admin");
     st2=conn.createStatement();
String sqlQuer="SELECT table_schema , SUM( data_length +
index_length) / 1024' Size in KB' FROM information_schema.TABLES
where table_schema='"+DB+"' GROUP BY table_schema  ";
     ResultSet rs=st2.executeQuery(sqlQuer);
     if(rs.next()){
         long used_db_len=rs.getInt(2);
         long db_free_space=tot_db_len-used_db_len;
         long tot_tbl_len=Integer.parseInt(rst.getString("u_size"));
         Statement stt = null;
        try {

             Class.forName("com.mysql.jdbc.Driver");
         conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/sns","root"
,"admin");
         stt = conn.createStatement();
String sqlQuery="SELECT TABLE_NAME,round(((data_length +
index_length)/ 1024 ),2) 'Size in MB' FROM information_schema.TABLES
WHERE table_schema = '"+DB+"' and   TABLE_NAME='"+tbl+"' ";
         ResultSet rss=stt.executeQuery(sqlQuery);
         while(rss.next()){
                  long used_tbl_len=rss.getInt(2);
                long free=tot_tbl_len-used_tbl_len;
                  if(db_free_space>filesizeInKB){
if(free>filesizeInKB){
              String uid =request.getParameter("uid");
            String file1= request.getParameter("file1");
if(file1 != null){

FileInputStream fis = null;
File image = null;
PreparedStatement st5=null;
if(file1 != ""){
```

```
image = new File("D:\\Documents and
Settings\\vinoth\\Desktop\\"+file1);
fis = new FileInputStream(image);
        }
try{
Class.forName("com.mysql.jdbc.Driver");
  con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/access","ro
ot","admin");
String tbb=" "+rst.getString("u_tab")+" ";
st5 =con.prepareStatement("insert into "+tbb+" values (?,?)");
st5.setString(1, uid);
if(file1 != "")
    st5.setBinaryStream(2, (InputStream)fis, (int)(image.length()));
else
    st5.setBinaryStream(2, null);
int i =st5.executeUpdate();

response.sendRedirect("user_page.jsp");

 }
catch(Exception e)
        {
  out.println(e);
            }
}
%>

<font color="#669900 ">

<div style="height: 500px;width: 500px; border-style: ridge ;font-
style: italic;margin-left: 100px ">

<center>
<pre>  <b>   <div> Domain Name                :
<%=rss.getString(1)%></div></b><br></pre>
<pre>  <b>   <div> Domain Size                : <%=tot_tbl_len%>
Kb</div></b><br></pre>
<pre>  <b>   <div> Used Space                 : <%=rss.getInt(2)%>
Kb</div></b><br></pre>
<pre>  <b>   <div> Free space                 : <%=free%>
Kb</div></b><br></pre>
<pre>  <b>   <div> Total Provider Memory     : <%=
tot_db_len%>Kb</div></b><br></pre>
<pre>  <b>   <div> Available Provider Memory : <%=used_db_len%>
Kb</div></b><br></pre>
<pre>  <b>    <div> Used Provider Memory      : <%= db_free_space%>
Kb</div></b><br></pre>
</center> </div>
</font>
<%
else{
out.println("out of tbl memory");

}}else{
out.println("out of DB memory");
```

```
}
}}
 catch (Exception e) {
out.println("Error in Index JSP" + e);
}
}
}
catch (Exception ea) {
                      out.println("Error in Index JSP" + ea);
      }

 }
else{
out.println("<font color='white'>Not a Valid Profile</font>");
        }}
catch (Exception e) {
          e.printStackTrace();
        }
}
%>
</select>
</form>
</div>
</div>
</div>
</div>
<div class="copy_right">
<p>Company Name Â© All rights Reseverd | Design by  <a
href="http://w3layouts.com"> W3Layouts </a></p>
</div>
</body>
</html>
```

6.2.4. VIDEO STREAM:

```
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.Date"%>
<%@page import="java.text.SimpleDateFormat"%>
<%@page import="java.text.DateFormat"%>
<!DOCTYPE html>
<html>
<head>
<title>Free Retail Hosting Website Template | Hosting ::
w3layouts</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
<link href="css/style.css" rel="stylesheet" type="text/css"
media="all"/>
<link href='http://fonts.googleapis.com/css?family=Electrolize'
rel='stylesheet' type='text/css'>
<script type="text/javascript" src="js/jquery.min.js"></script>
           <script type="text/javascript">
                     jQuery(document).ready(function($) {
```

```
                $(".scroll").click(function(event){
                     event.preventDefault();

     $('html,body').animate({scrollTop:$(this.hash).offset().top},1
200);
                });
           });
</script>
</head>
<body>
<div class="header">
<div class="header_top">
<div style="font-size: 25px;color: white;margin-left: 120px"
>Innovative Schemes for Resource Allocation
in the Cloud for Media Streaming Applications</div>
<div class="wrap">
<div class="menu">
<ul>
<li class="active"><a href="Play_Video.jsp">Home</a></li>
<li><a href="Profile.jsp">Profile</a></li>
<li><a href="Purchase.jsp">Purchase</a></li>
<li><a href="">Mail</a></li>
li><a href="Logout.jsp">Logout</a></li>
<div class="clear"></div>
</ul>
</div>
<div class="clear"></div>
</div>
</div>
</div>
<%
String Ad_sess=(String)session.getAttribute("username");
String v_name=request.getParameter("name").trim();
String time=request.getParameter("time");
String S_Time=null;
if(Ad_sess.equals("vinoth")){

 try{ Class.forName("com.mysql.jdbc.Driver");
      Connection con=
DriverManager.getConnection("jdbc:mysql://localhost:3306/resource","
root","admin");
Statement stw =  con.createStatement();
stw.executeUpdate("update video set time='"+time+"' where
F_Name='"+v_name+"' ");
 }
catch(Exception w){
              out.println(w);
              }
}
Statement s2;
try {
Class.forName("com.mysql.jdbc.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/resource","
root","admin");
String qry1="select * FROM VIDEO where F_Name='"+v_name+"'  ";
```

```
s2 = con.createStatement();
ResultSet  r = s2.executeQuery(qry1);
if(r.next())
{
S_Time=  r.getString("time");
}}
catch(Exception r){
out.println(r);
}
%>
<div class="main">
<div class="services_heading">
<p> </p>
</div>
<div class="wrap">
<div class="services">
<%
String video=request.getParameter("name");
int Time=0;
Statement s1;
DateFormat df = new SimpleDateFormat("HH:mm:ss");
 String formattedDate = df.format(new Date());
        out.println(formattedDate);
try {
  Class.forName("com.mysql.jdbc.Driver");
         Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/resource","
root","admin");
String qry1="SELECT (minute
(TIMEDIFF('"+formattedDate+"','"+S_Time+"'))*60)+Second
(TIMEDIFF('"+formattedDate+"','"+S_Time+"')) AS 't1 - t2',
TIMEDIFF(@t2,@t1) AS 't2 - t1'";
s1 = con.createStatement();
ResultSet  rstt = s1.executeQuery(qry1);
if(rstt.next())
{
Time=rstt.getInt(1);
}}
catch(Exception r){
out.println(r);
}
 out.println(Time);

%>
<div style="">
<video width="1024" height="300"   autoplay>
<source src="<%=video%>#t=<%=Time%>" type="video/mp4">
<object data="<%=video%>#t=<%=Time%>" width="320" height="240">
<embed width="320" height="240" src="111.swf">
</object>
</video></div>
</div>
</div>
</div>
<div class="copy_right">
<p>Company Name Â© All rights Reseverd | Design by  <a
href="http://w3layouts.com"> W3Layouts </a></p>
```

```
</div>
</body>
</html>
```

6.2.5. UPLOAD INFO:

```
<%@page import="java.net.InetAddress"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<%@ page import="java.util.List" %>
<%@ page import="java.util.Iterator" %>
<%@ page import="java.io.File" %>
<%@ page
import="org.apache.commons.fileupload.servlet.ServletFileUpload"%>
<%@ page
import="org.apache.commons.fileupload.disk.DiskFileItemFactory"%>
<%@ page import="org.apache.commons.fileupload.*"%>
<%@ page
import="java.io.*,java.sql.*,java.util.*,java.text.*,java.text.Simpl
eDateFormat" %>
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<%@ page import="java.io.*" %>

<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <meta http-equiv="content-type" content="text/html;
charset=utf-8" />
        <title>ONLINE DATA ACCESS</title>
        <meta name="keywords" content="" />
        <meta name="description" content="" />
        <link href="styles.css" rel="stylesheet" type="text/css" />

        <script type="text/javascript" src="lib/jquery-
1.3.2.min.js"></script>
        <script type="text/javascript"
src="lib/jquery.tools.js"></script>
        <script type="text/javascript"
src="lib/jquery.custom.js"></script>

        <!-- Pirobox setup and styles -->
        <script type="text/javascript"
src="lib/pirobox.js"></script>
        <script type="text/javascript">
            $(document).ready(function() {
                $().piroBox({
                    my_speed: 400, //animation speed
                    bg_alpha: 0.1, //background opacity
                    slideShow : false, // true == slideshow on,
false == slideshow off
                    slideSpeed : 4, //slideshow duration in
seconds(3 to 6 Recommended)
                    close_all : '.piro_close,.piro_overlay'// add
class .piro_overlay(with comma)if you want overlay click close
piroBox

                });
            });
        </script>
```

```
        <%
            String msg = request.getParameter("username");
            if (msg != null) {
                session.invalidate();
            }
        %>
        <link href="images/style.css" rel="stylesheet"
type="text/css" />
        <!-- Pirobox setup and styles end-->

    </head>
    <body>
<div id="main">
<!-- header begins -->
<div id="header">
<div id="buttons">
<a></a><div class="but_razd"></div>
<a href="Options.jsp?username=<%=msg%>" class="but"
title="">Options</a><div class="but_razd"></div>
<a href="Uploadfiles.jsp?username=<%=msg%>" class="but"
title="">Upload Files</a><div class="but_razd"></div>
<a href="Downloadfiles.jsp?username=<%=msg%>" class="but"
title="">Files</a><div class="but_razd"></div>
<a href="Logout.jsp" class="but"  title="">Log Out</a>
</div>
</div>
<!-- header ends -->
<!-- content begins -->
<div class="gallery">
<div class="gallery_bot">
<h1>Our Photo Gallery</h1>
<div class="row">
<div class="box_img2">
<div class="g_size"><a class="pirobox_gal1" title="1st Project
Image"><img src="images/upload.png" alt="" /></a></div>
</div>
<div class="box_razd"></div>
<div class="box_img3">  <br/><br/><br/>
<center><table border="2">
<tr><td><h1>FILE UPLOAD</h1></td></tr>
<%
 try {
String saveFile="",l="",ln="",mname="",dob="",celno="";
String
sex="",voterno="",email="",bg="",pstreet="",cstreet="",parea="",care
a="",pcity="",ccity="";
String ppinno="",cpinno="",pass="",repass="";
int fileidnum=0,downloadcount=0,vc=0;
String contentType = request.getContentType();
// Create a factory for disk-based file items
DiskFileItemFactory factory = new DiskFileItemFactory();
// Set factory constraints
factory.setSizeThreshold(4012);
//factory.setRepository("c:");
// Create a new file upload handler
ServletFileUpload upload = new ServletFileUpload(factory);
```

```
// Set overall request size constraint
//upload.setSizeMax(10024);
// Parse the request
List items = null;
try {
     items = upload.parseRequest(request);
            } catch (FileUploadException e) {
    e.printStackTrace();
            }
byte[] data = null;
String fileName = null;
// Process the uploaded items
Iterator iter = items.iterator();
while (iter.hasNext()) {
FileItem item = (FileItem) iter.next();
if (item.isFormField()) {
//processFormField(item);
String name = item.getFieldName();
String value = item.getString();
if (name.equalsIgnoreCase("uid")) {
l = value;
System.out.println("fn" + l);
                    }
else  if (name.equalsIgnoreCase("cat")) {
 ln = value;
System.out.println("fn" + l);
}
else {
    System.out.println("ERROR");
     }
 } else {
      data = item.get();
      fileName = item.getName();
                }
            }
 saveFile = fileName;
String path =
request.getSession().getServletContext().getRealPath("/");
String patt=path.replace("\\build", "");
String strPath = patt+"access"+"\\"+saveFile;
File ff = new File(strPath);
FileOutputStream fileOut = new FileOutputStream(ff);
fileOut.write(data, 0, data.length);
fileOut.flush();
fileOut.close();
out.println(saveFile);
System. out.println("Thrid");
%><br><table border="2"><tr><td><b>You have successfully upload the
file:</b> 
<%out.println(saveFile);%></td></tr></table>
<%
Connection con = null;
PreparedStatement st5 = null;
FileInputStream fis;
InputStream sImage;
try {
```

```
File f = new File(strPath);
long length = f.length();
System.out.println("length " + length);
String ip=request.getRemoteAddr();
Class.forName("com.mysql.jdbc.Driver");
con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/resource","
root","admin");
String queryString = "insert into video values (?,?,?,?,?,?)";
st5 = con.prepareStatement(queryString);
fis = new FileInputStream(f);
st5.setInt(1,0);
st5.setString(2,
saveFile);
st5.setString(3, "");
st5.setString(4,"1");
st5.setString(5, ln);
st5.setString(6, (String)session.getAttribute("user"));
int s = st5.executeUpdate();
response.sendRedirect("Provider_Upload.jsp?Video Uploaded
Successfully");
}catch(Exception e)           {
       out.println(e);
                }
}
catch (Exception ex) {
           out.println(ex);
     }
%>
</br> </table>
</center></div>
</div>
<div style="height:13px"></div>
<div style="height:10px"></div>
</div>
</div>
</div>
</body>
</html>
```

CHAPTER 7

SYSTEM TESTING

7.1. UNIT TESTING:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input produces valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.2. INTEGRATION TESTING:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

CHAPTER 8

CONCLUSION

8.1. CONCLUSION & FUTURE ENHANCEMENTS:

In this project considers recreations of a multi-client, cloud assisted worldwide video conferencing framework. We accept a video switch (for example as created by the V connect venture) that can move between the cloud facilitated areas in a user transparent way. Utilizing sensible interest from genuine world information, we examined two diverse interest situations (one in view of gaming the other dependent on instruction). We thought about situations where video switch areas were chosen statically what's more, progressively. Our analyses were rushed to decide the postpone experienced by clients because of the decisions of cloud have area and courses for their video session.

CHAPTER 9

REFERENCES

[1] Shan-Hsiang Shen, “Efficient SVC Multicast Streaming for Video Conferencing with SDN Control,” IEEE Transactions on Network and Service Management, vol. 16, pp. 403 - 416, June 2019.

[2] E. Coronado, R. Riggio, J. Villalón, and A. Garrido, “Efficient real-time content distribution for multiple multicast groups in SDNbased WLANs,” IEEE Trans. Netw. Service Manag., vol. 15, no. 1, pp. 430–443, Mar. 2018.

[3] Mohammad H. Hajiesmaili, Lok To Mak, Zhi Wang, Chuan Wu, Minghua Chen, and Ahmad Khonsari, “Cost-Effective Low-Delay Design for Multiparty Cloud Video Conferencing, ” in IEEE Transactions on Multimedia, vol: 19, pp. 2760 - 2774, Dec. 2017

[4] A. Bentaleb, A. C. Begen, R. Zimmermann, and S. Harous, “SDNHAS: An SDN-enabled architecture to optimize QoE in HTTP adaptive streaming,” IEEE Trans. Multimedia, vol. 19, no. 10, pp. 2136–2151, Oct. 2017.

[5] S. Y. Shah, B. Szymanski, P. Zerfos, and C. Gibson, “Towards relevancy aware service-oriented systems in wsns,” IEEE Transactions on Service Computing, vol. 8, 2015.

[6] S. Agarwal, M. Kodialam, and T. V. Lakshman, “Traffic engineering in software defined networks,” in Proc. IEEE INFOCOM, 2013, pp. 2211–2219.

[7] I. Papapanagiotou, M. Falkner, and M. Devetsikiotis, “Optimal functionality placement for multiplay service provider architectures,” IEEE Trans. Network and Service Management, vol. 9, no. 3, pp. 359–372, 2012.

[8] D. Roberts, T. Duckworth, C. Moore, R. Wolff, , and J. O’Hare, “Comparing the end to end latency of an immersive collaborative environment and a video conference,” in Proceedings of the IEEE/ACM international Symposium on Distributed Simulation and Real Time Applications, 2009, pp. 89–94.

[9] M. Chen, G. Su, and M. Wu, “Dynamic resource allocation for robust distributed multi-point video conferencing,” IEEE Trans. on Multimedia, vol. 10, no. 5, August 2008.

[10] N. Laoutaris, G. Smaragdakis, K. Oikonomou, I. Stavrakakis, and A. Bestavros, “Distributed placement of service facilities in large-scale networks,” in IEEE INFOCOM, 2007, pp. 2144–2152.

[11] Y. Chu, S. Rao, S. Seshan, and H. Zhang, “End-to-end delay analysis of videoconferencing over packet-switched networks,” IEEE/ACM Transactions on Networking, vol. 8, no. 4, pp. 479–492, 2000.

APPENDICES

A.1. SAMPLE SCREENS:

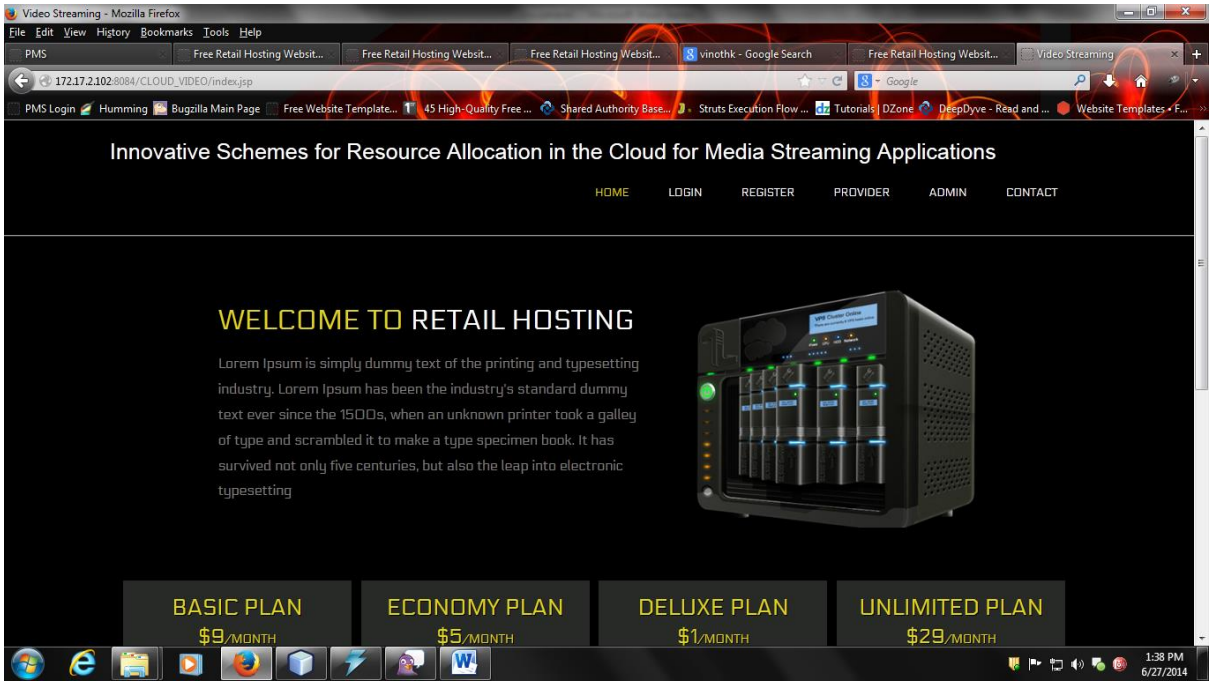

Fig 5: Home Page1

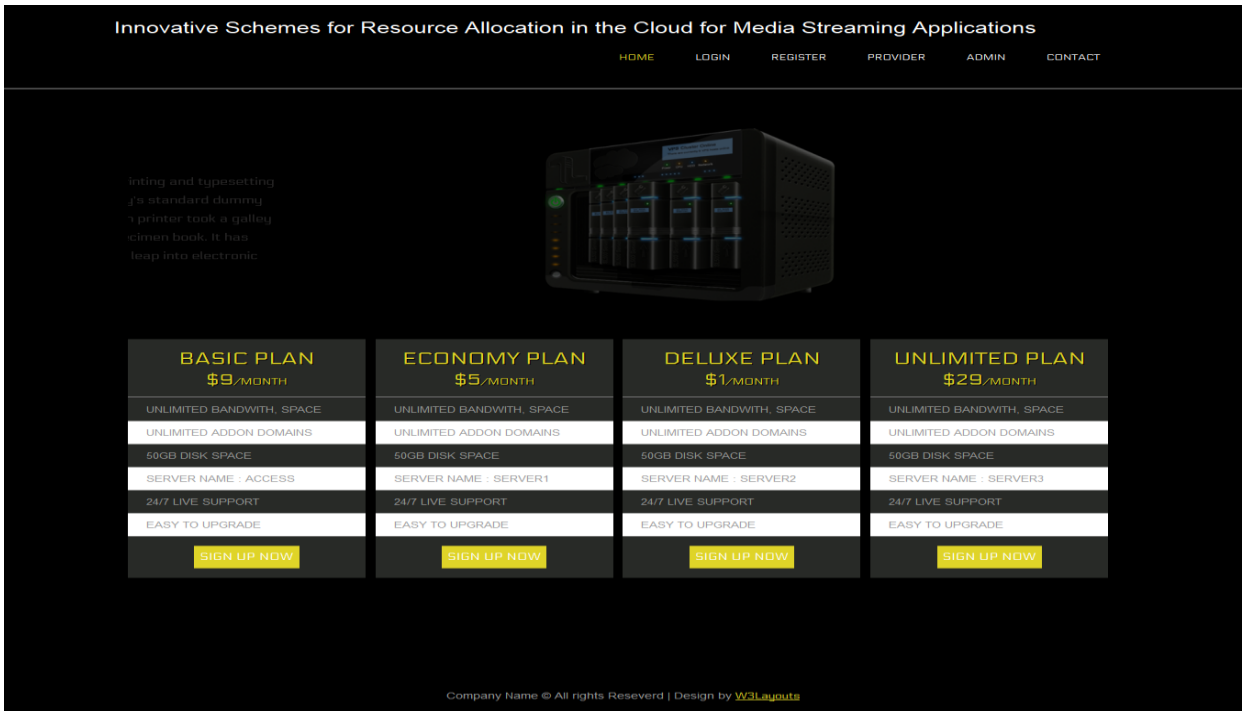

Fig 6: Home Page2

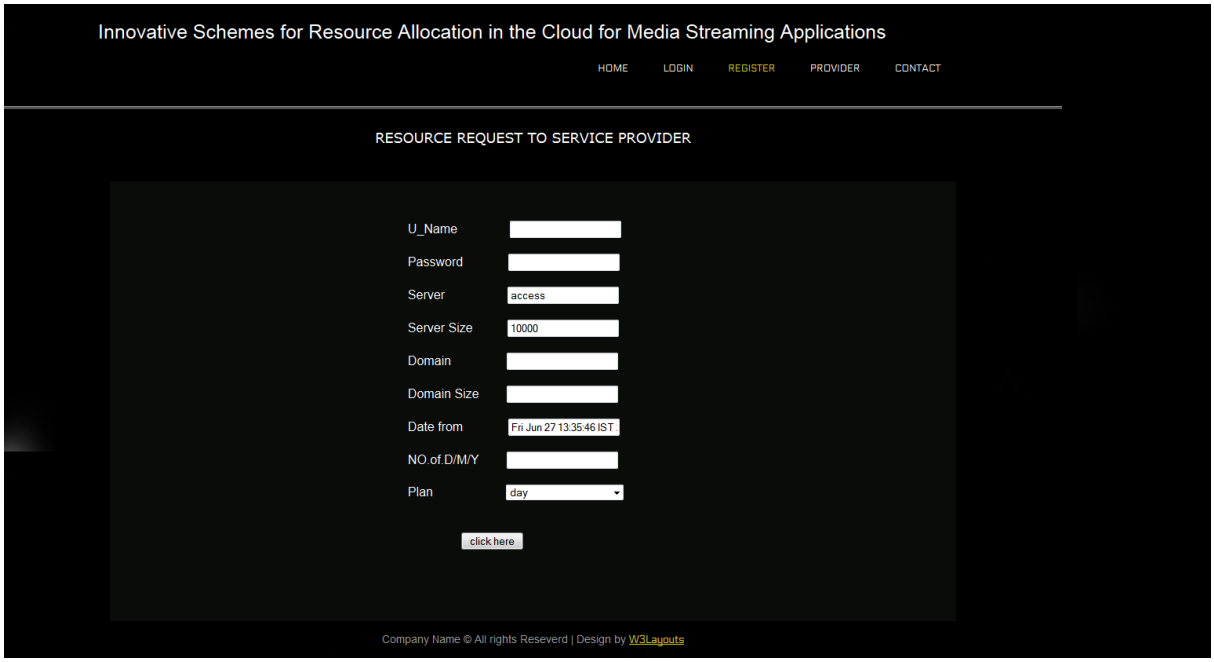

Fig 7: Request to service Provider

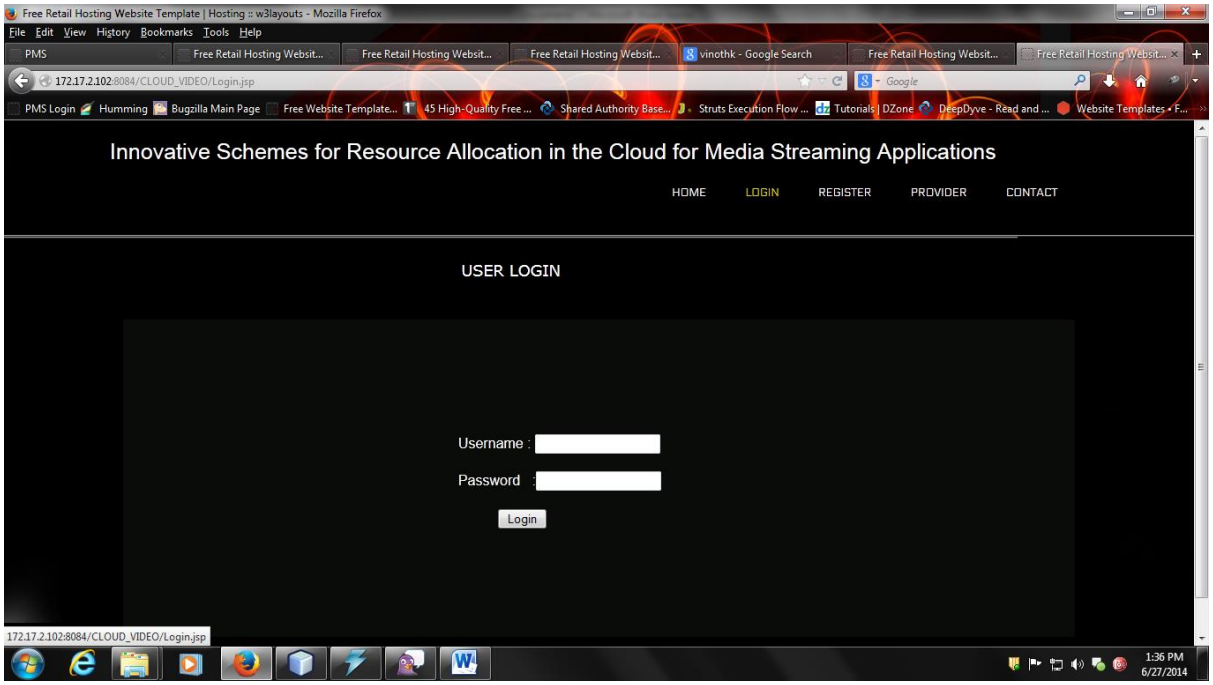

Fig 8: User Login

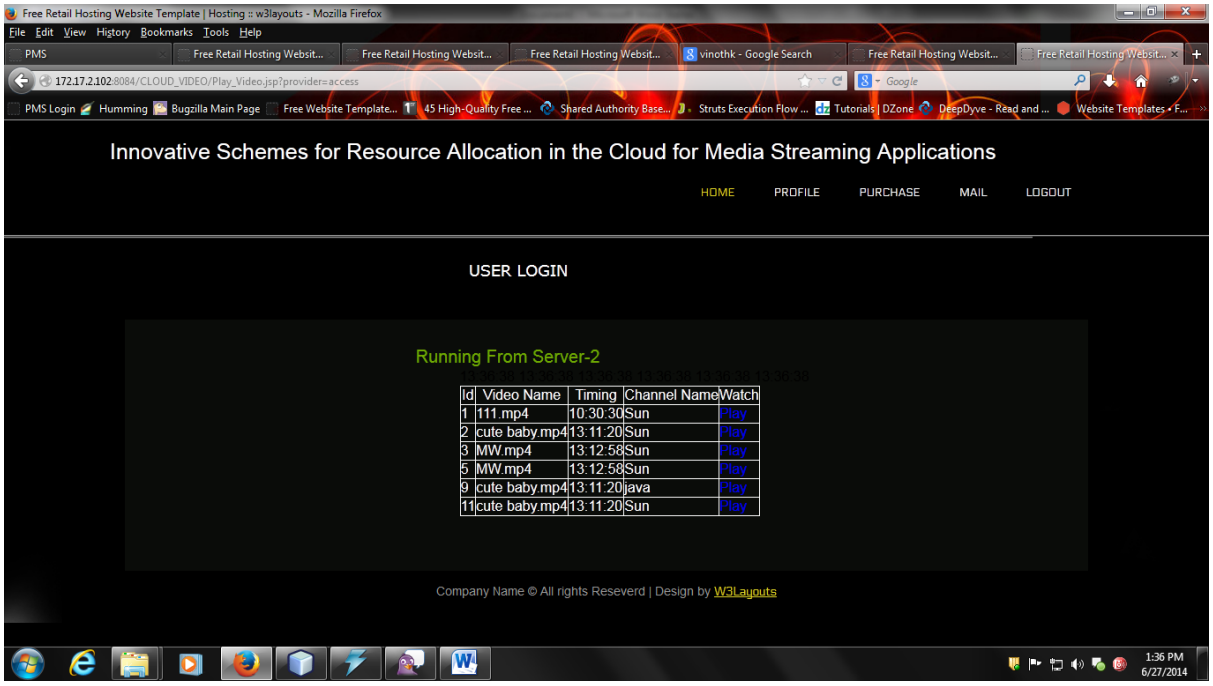

Fig 9: Videos for user

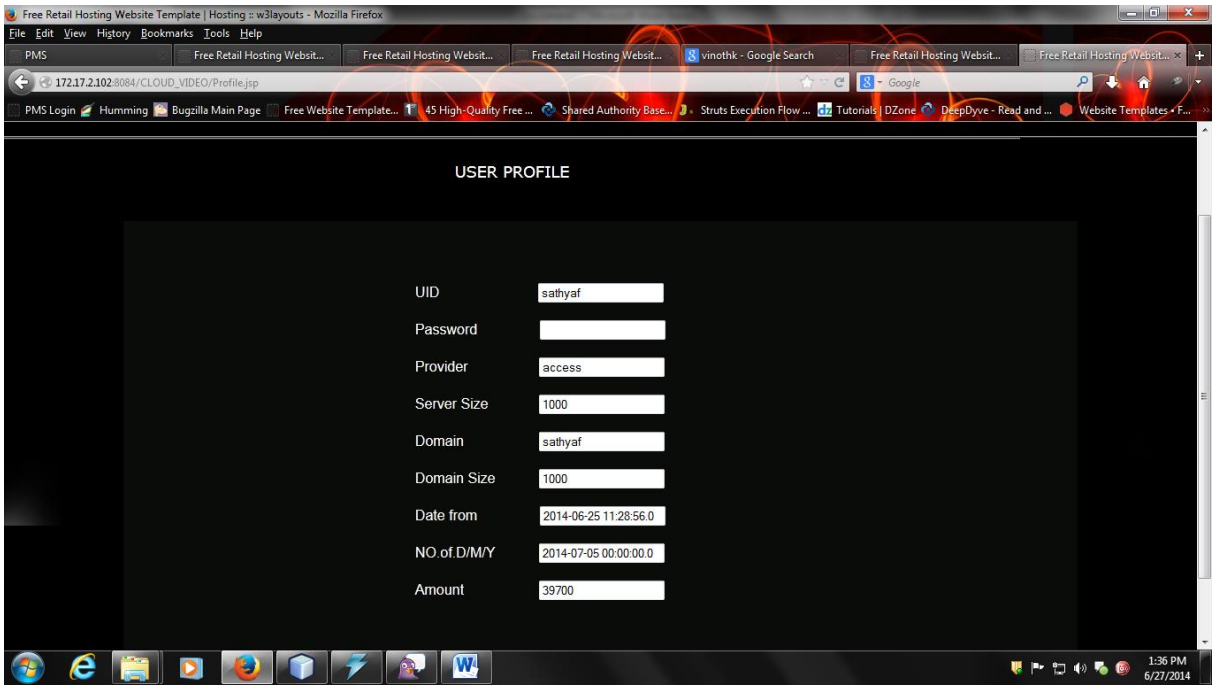

Fig 10: User Profile

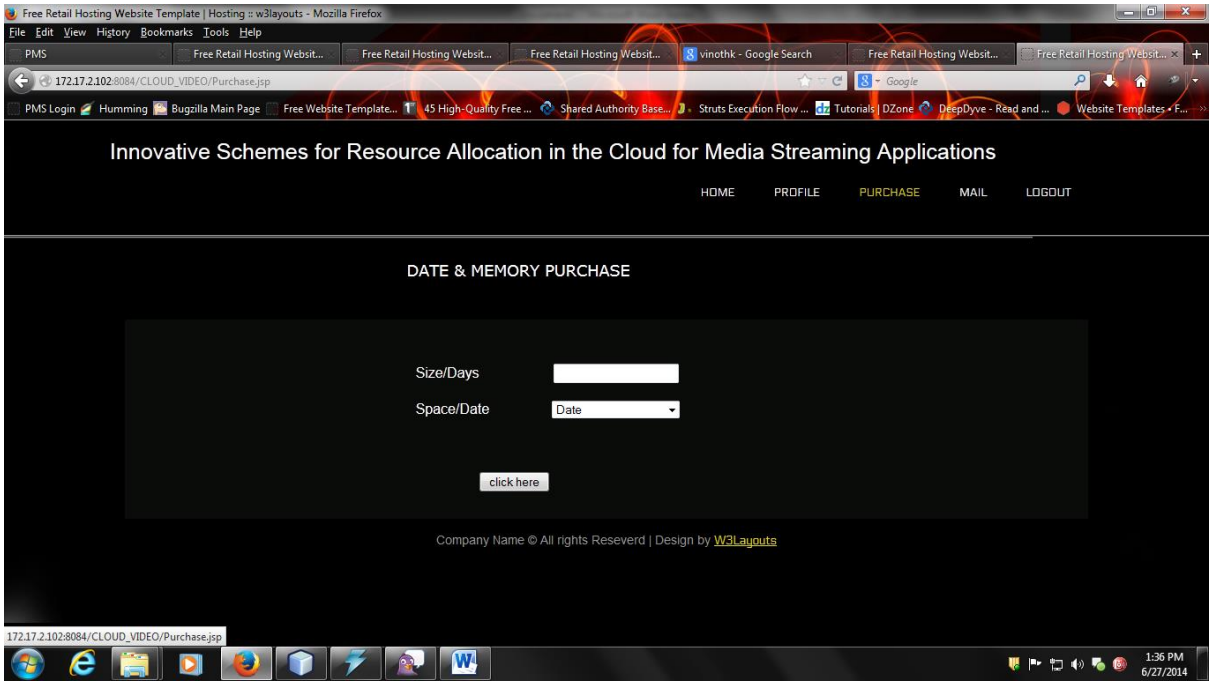

Fig 11: Data and memory Purchase

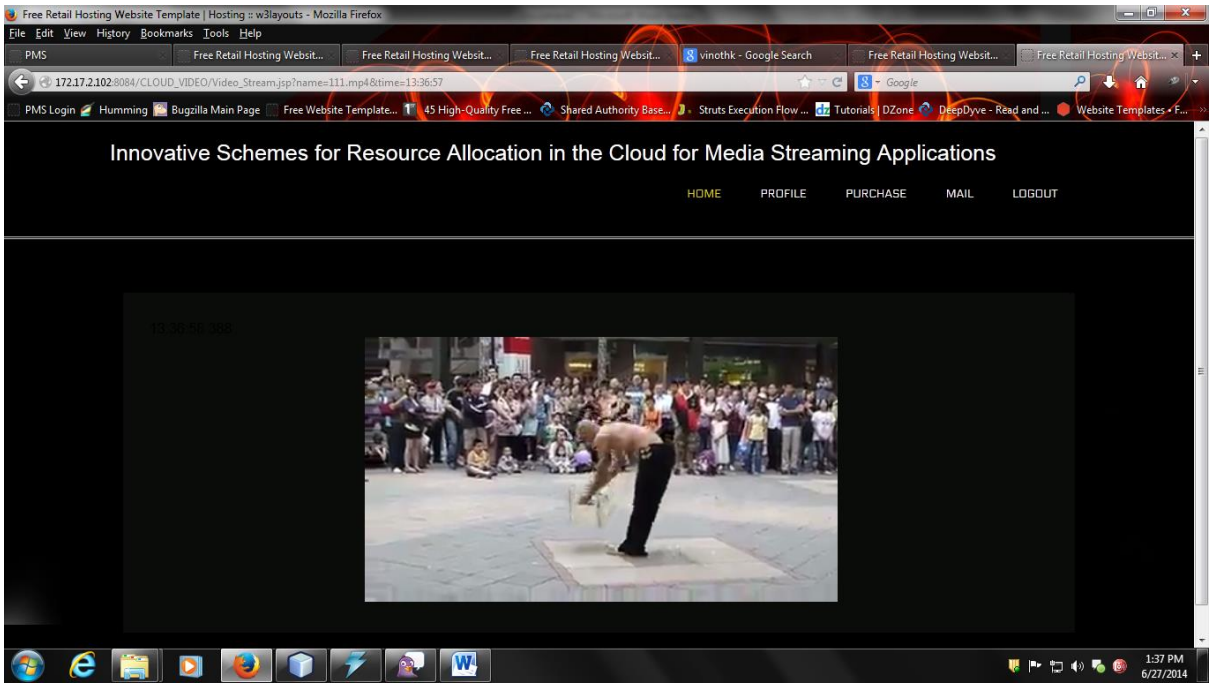

Fig 12: Video playing

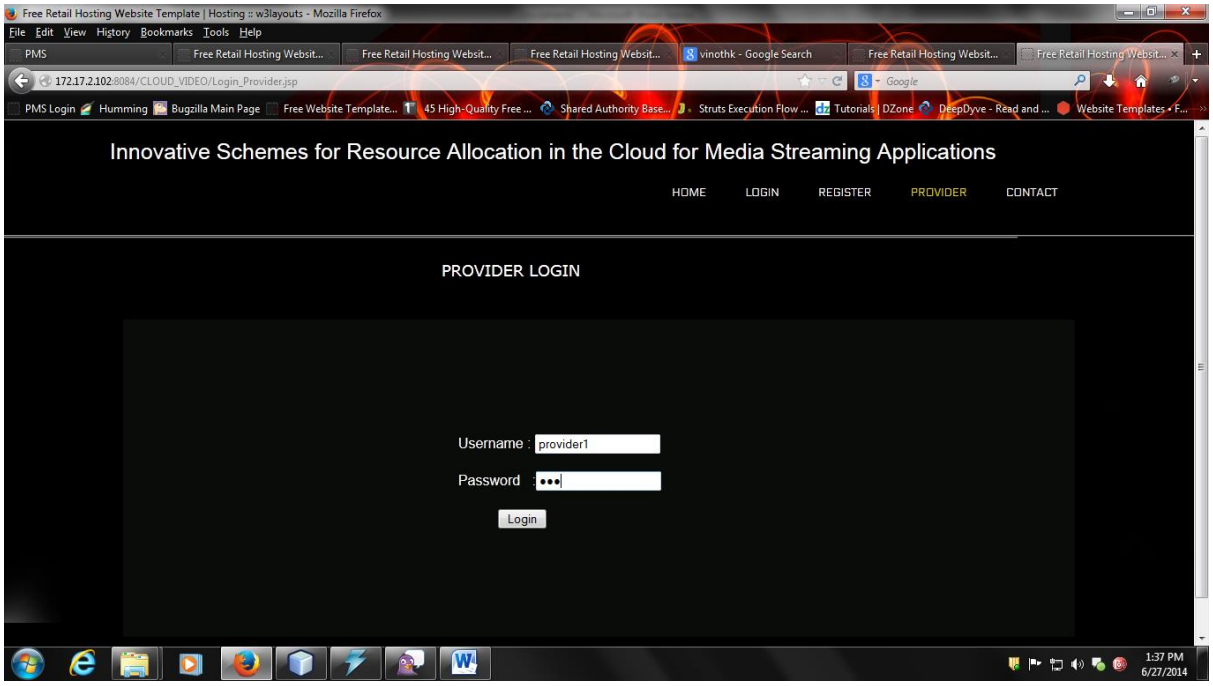

Fig 13: Provider Login

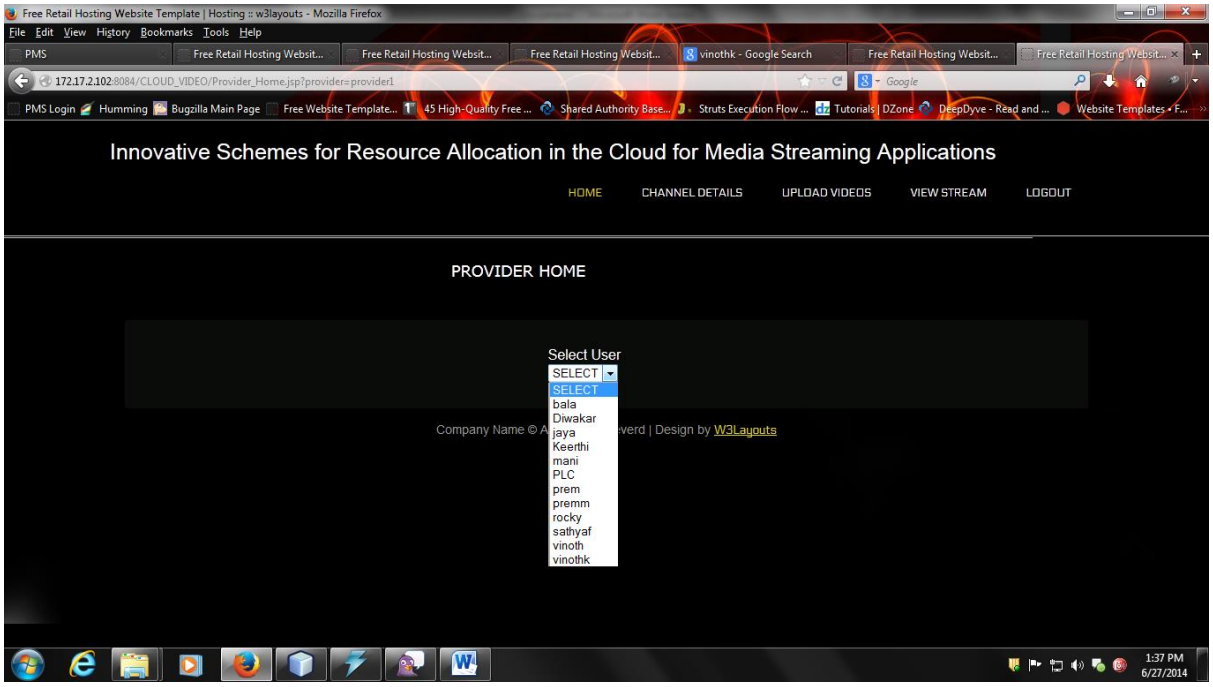

Fig 14: Provider Home

Fig 15: User details

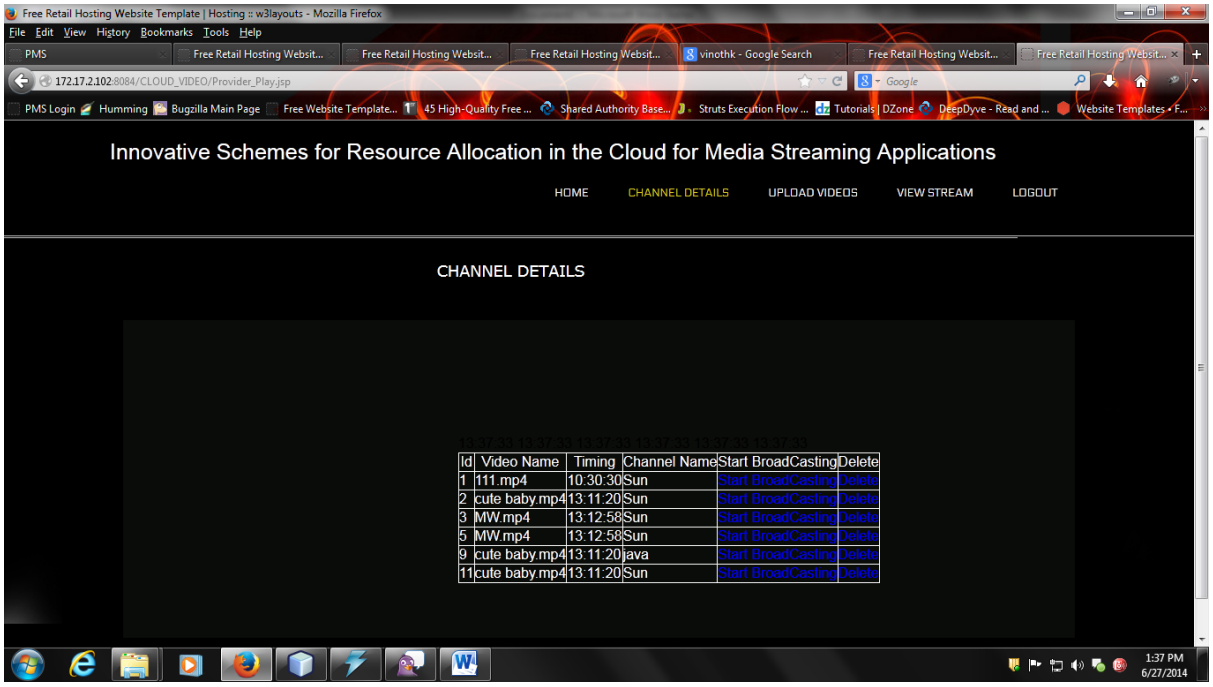

Fig 16: Channel Details

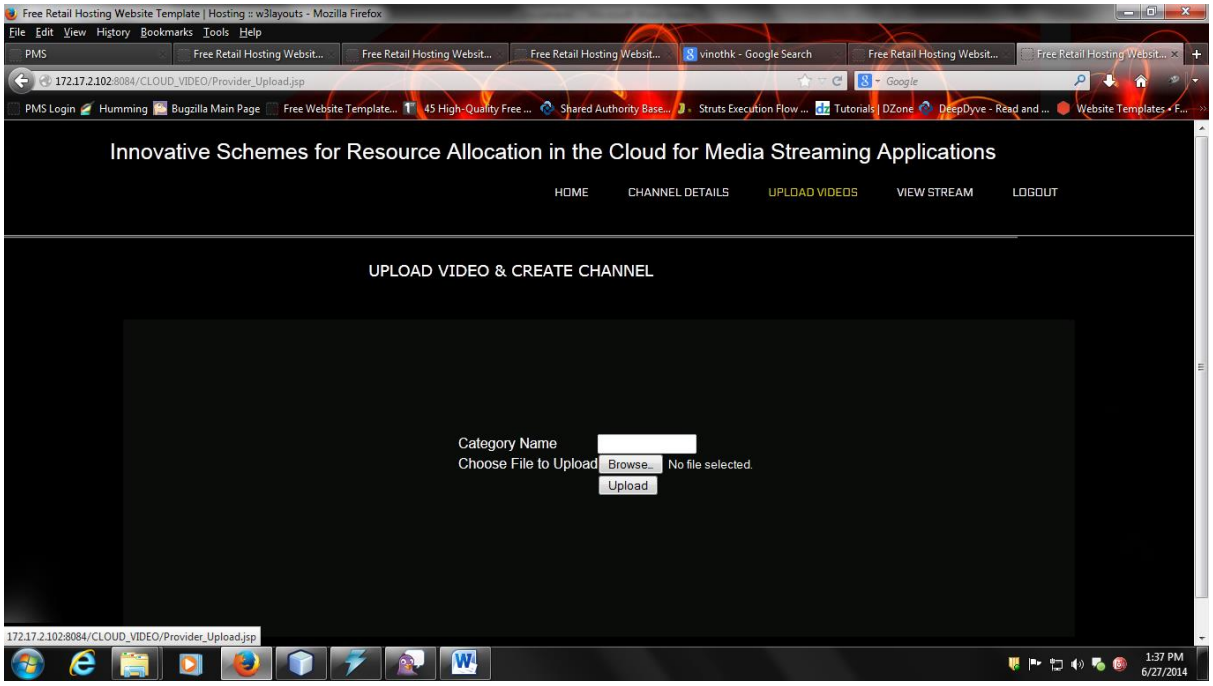

Fig 17: Create Channel

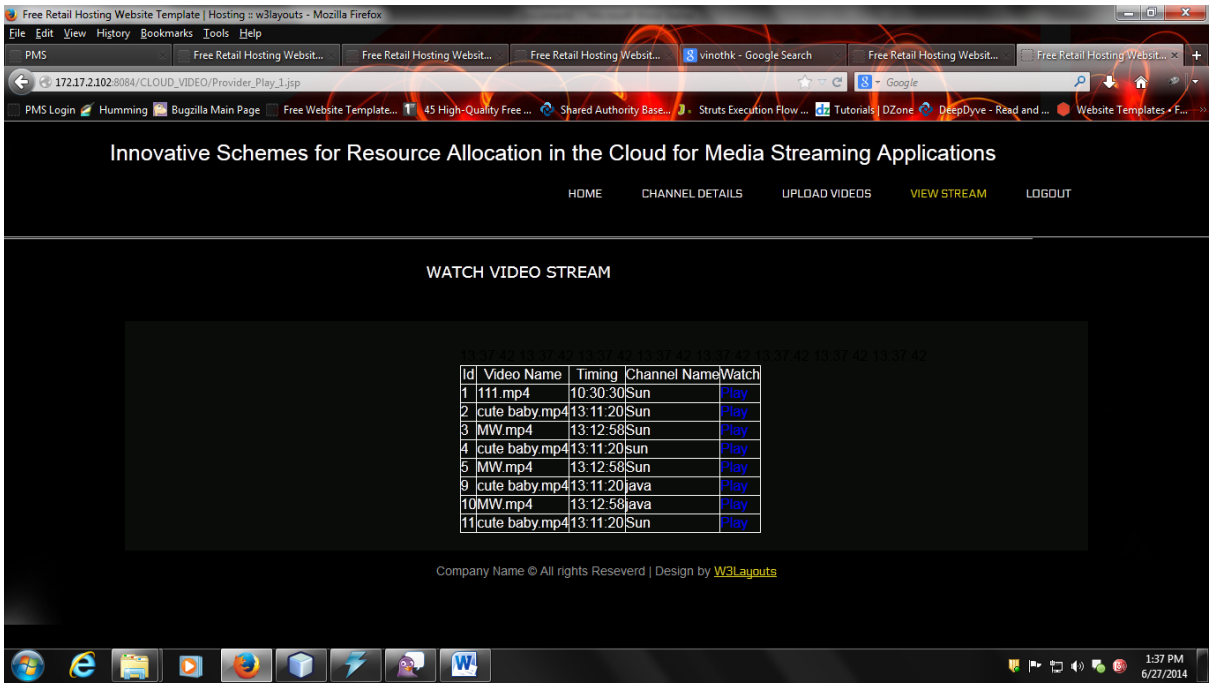

Fig 18: Video Stream

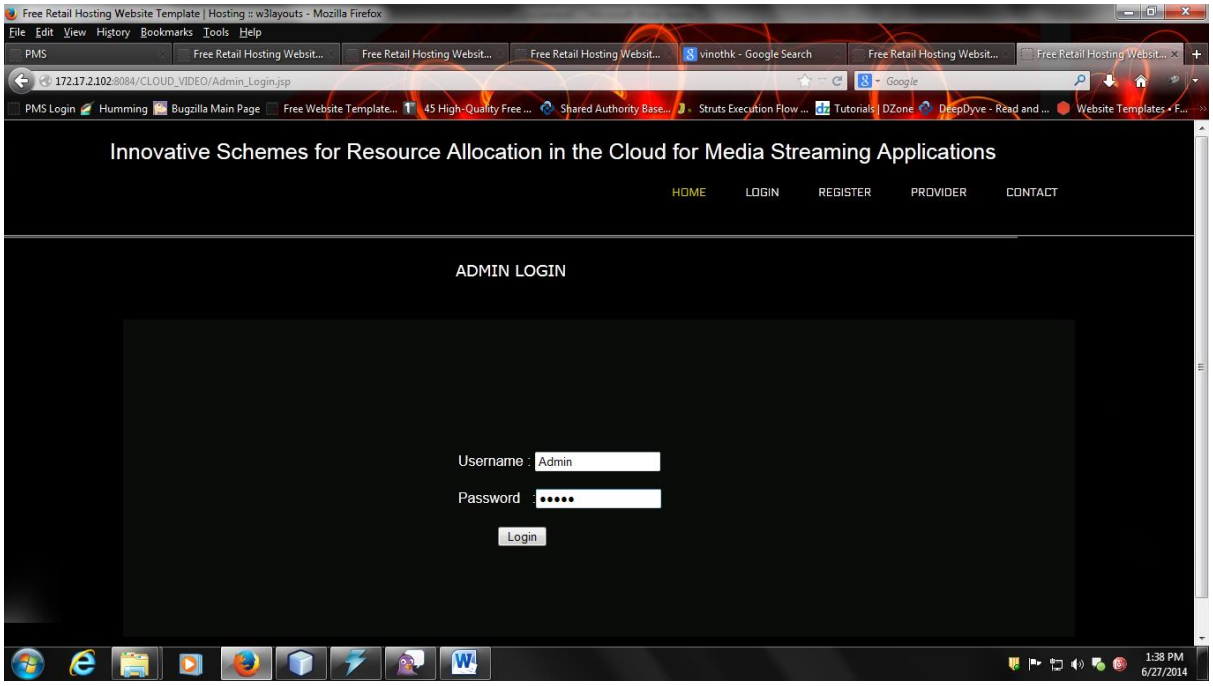

Fig 19: Admin Login

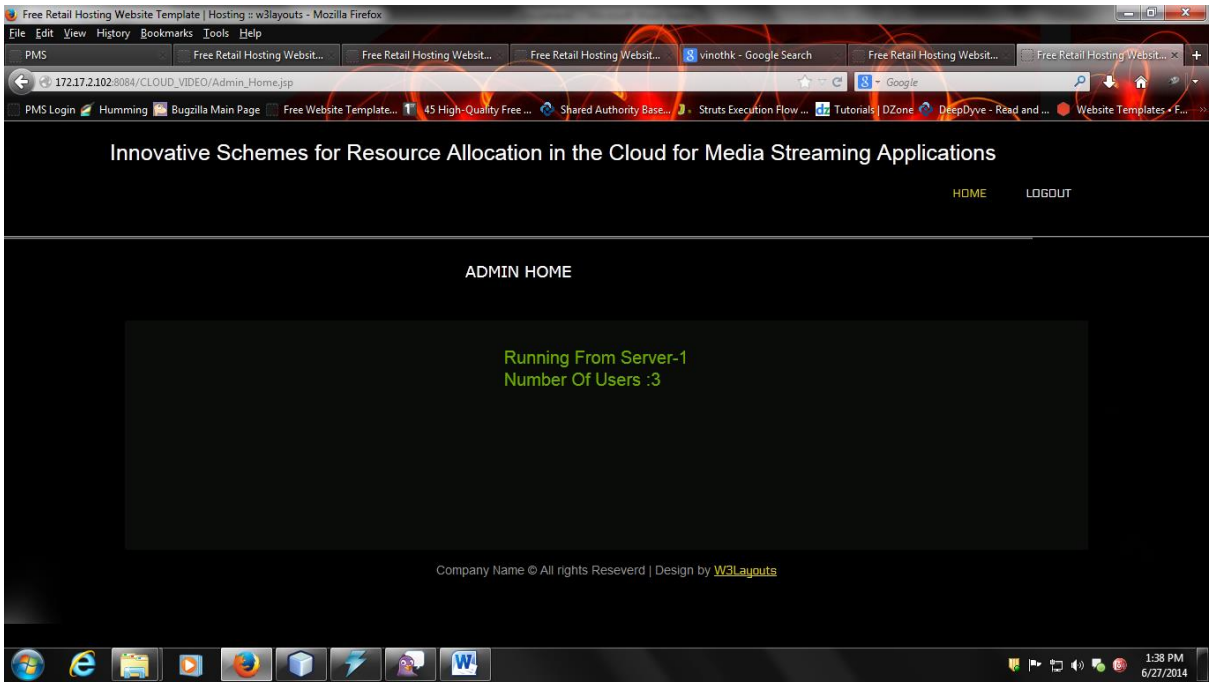

Fig 20: Details regarding the users to admin

A.2. PUBLICATION:


e-ISSN: 2395-0056 p-ISSN: 2395-0072
International Research Journal of Engineering and Technology (IRJET)
(An ISO 9001 : 2008 Certified Journal)
Is hereby awarding this certificate to
Mahalakshmi. D
In recognition the publication of the manuscript entitled
Faces in the Clouds: Time-Domain Attribute based Access Control for Cloud-based Video Content Sharing
published in our Journal Volume 8 Issue 4 April 2021
Editor in Chief
E-mail : editor@irjet.net
Impact Factor : 7.529
www.irjet.net

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

Faces in the Clouds: Time-Domain Attribute based Access Control for Cloud-based Video Content Sharing

Maheshwari. M[1], Mohana Sree. P[2], Mahalakshmi. D[3]

[1]Asst. Professor, CSE of Panimalar Engineering College, Tamil Nadu, India
[2,3]UG Scholar, CSE of Panimalar Engineering College, Tamil Nadu, India

---

Abstract - *Media streaming applications have attracted a vast number of users on the Internet recently. Nowadays, it is economically inefficient to provide streaming distribution with guaranteed QoS relying only on central resources at a media content provider with the arrival of bandwidth intensive applications. One of the reason due to which the media content providers are obtaining streaming resources that match the demand is because of the elastic infrastructure offered by Cloud Computing. The media content providers charge for the resources allocated in the cloud. These resources pricing scheme offers discount rates depending on the non-linearity during the period of time in which these resources are reserved in the cloud. In such case, an open problem is to decide that what is the right amount of resources that must be reserved in the cloud, and what should be their reservation time so that the cost invested on the media content provider can be minimized. Thus, we propose a simple - easy to implement - algorithm for resource reservation that provides maximum utilization of the discounted rates offered in the tariffs, and also ensuring that sufficient resources are reserved in the cloud. Based on the demand for the streaming capacity, we have designed our algorithm accordingly to minimize the risk of making wrong resource allocation decisions. The outcome which we got from our numerical evaluations and simulations reveals that the proposed algorithm significantly reduces the monetary cost of resource allocations in the cloud as compared to other conventional schemes.*

***Key Words*: Video Conferencing, Cloud Computing, VM Migration, Resource Reservation, Overload avoidance.**

1. INTRODUCTION

Cloud computing allows assured gain to share lots of configurable framework assets and more elevated level administrations that can be designed with insignificant the board exertion, over the Internet. The idiom "Cloud" is a tarn of huge infrastructural possessions which offers multiple diverse amenities for the end users including applications as services and the hardware along with system software, end user accesses the complete computer requirements alternatively in a consistent manner. The user does not need to know where the computer requirement is discovered and how the cloud works. Cloud providers control all resources usefully available to the customers, where constructed on service level agreement (SLA) "Computing" is concluded. The cloud computing emerged, as it supports virtualization technologies and provides opportunity for end user to deploy virtual resources at negligible cost without having any infrastructure.

Users can avail of resources over the internet on the availability of the cloud storage. Cloud computing requires large storage for multiple users. Multiple locations can share the network from the centralized resource. It is called as Enterprise cloud by single organizer and called as Public cloud by multiple organizers. It is economic to society. cloud computing operation is faster, maintenance is minimum and the manageability can be improved. When the user demand is high, cloud computing has the capability to provide the network and this is referred burst computing capability. And provides high network facility, the cost is economical, its architecture is service-oriented and also automotive. The cloud computing is using different features, such as NIST, OMG, and DMTF for interoperability. For the convenience of cloud computing NIST is used, it provides the on-demand and service for the user and also reduces the service of the management. Cloud computing also provides management of product and service in cloud, can protect the data in a cloud architecture, and can also interact with one cloud service to another. It gives a password to the user, moving workload from one location to other location, transferring of stored Data from one system to other, process management.

Multi-user video conferencing is now extremely common. These systems are often enhanced by software within the cloud. Cloud routers instantiated at appropriate locations can improve the user experience by taking advantage of overlay networks and application-layer multicast (ALM). A multi-server architecture for multi-user video chat enables flexibility in the choice of topology for transmitting video streams from one client to another. Multiple servers are used by popular systems such as Google Hangouts and Skype.

The system used by Google Hangouts and Skype for video chat is that users connect to their nearest server. Traffic from user A to user B is sent from user A to the server nearest to user A, then to the server nearest to user B and finally to user B (we refer to this routing policy as "StayOnRoute"). The amount of traffic sent to each user from

the server will increase linearly with every additional user. However, if each user connects to a different server, the amount of traffic sent between server's scales as the square of the number of servers. Perhaps for this reason, both Google Hangouts and Skype limit the number of participants in a video call to ten at the time of writing. This scaling problem would be improved if the users connected to a smaller number of servers with the capability of dynamically migrating between sites. We investigate this and other routing policies for video traffic routed via cloud servers.

2. LITERATURE SURVEY

Shan-Hsiang Shen, proposed a paper titled "Efficient SVC Multicast Streaming for Video Conferencing with SDN Control". In this paper they have proposed a novel SVC multicast streaming scheme named adaptive SDN-based SVC multicast (ASCast). The video layer formulates a linear programming problem for the tree construction because each video layer is formed by a multicast tree. To overcome the problem, they designed static and dynamic heuristic algorithms to build multicast trees and maximize overall video quality with limited TCAM space. They considered multicast integer programming address assignment for video layers and forwarding rule installation to reduce the TCAM consumption. As a result, they found that ASCast provides a 35% higher video data rate and installs 66% fewer forwarding rules into switches than other SVC video multicast schemes.

Mohammad H. Hajiesmaili, Lok To Mak, Zhi Wang, Chuan Wu, Minghua Chen, and Ahmad Khonsari proposed a paper called "Cost-Effective Low-Delay Design for Multi-Party Cloud Video Conferencing". This paper discusses a joint problem of user-to-agent assignment and transcoding-agent selection. The main objective is to simultaneously minimize the cost of the service provider and the conferencing delay. The problem discussed is combinatorial in nature which belongs to the NP-hard node assignment problems. They devise an adaptive parallel algorithm that finds a close-to-optimal solution to the problem with a bounded performance guarantee with the help of the Markov approximation framework. They implement a prototype video conferencing system, and carry out trace-driven experiments to evaluate their proposed solution performance. In a number of large-scale experiments using PlanetLab traces, the solution decreases the operational cost by 77% and simultaneously yields lower conferencing delay compared to an existing alternative.

3. RELATED WORK

A mechanism for mapping virtual machines (VMs) to physical resources is provided by Virtual Machine Monitors like Xen. The cloud users are largely not aware of this mapping. For example, users with the Amazon EC2 service, do not know where their VM instances run. The cloud provider has to make sure that the underlying physical machines (PMs) have sufficient resources to meet up to their needs. VM live migration technology makes it possible to change the mapping between VMs and PMs even when the application is still running. Since multiple generations of hardware coexist in a data center capacity of PMs can also be heterogeneous.

The main disadvantage is the policy issue which remains as how to decide the mapping adaptively so that the resource demands of VMs are met while the number of PMs used is minimized. The resources require VM's that are heterogeneous but due to the diverse set of applications they run and vary with time as the workloads grow and shrink and this becomes challenging. The other disadvantages are overload and green computing.

4. ARCHITECTURE OF THE PROPOSED WORK

We have proposed algorithms that optimally determine whether both the amount of reserved resources in the cloud and their reservation time is based on the prediction of future demand for streaming capacity and the financial cost on the media content provider is minimized. The algorithms are proposed in such a way that it makes use of the discounted time rates in the tariffs, while ensuring that sufficient resources are reserved in the cloud without incurring wastage. The evaluation of the performance of our algorithms is done numerically and also using simulations. Structured meetings with improved communications, reduced travel time and costs and increased productivity are few advantages of the proposed system.

The figure 1 shows the architecture of the proposed work. From the cloud consumer a request signal sent to the cloud broker via virtual machine repository. Then from the cloud broker the signal is passed on to the cloud providers. Now a respond to the request signal is sent back to the consumer from the cloud provider.

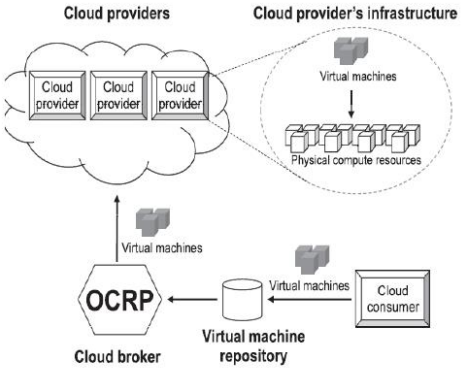

Fig -1: Architecture

5. MODULES AND ITS IMPLEMENTATION

Resource Provisioning (RP):
A cloud resource provisioning is proposed by formulating a stochastic programming model which is optimal. RP algorithm can provision computing resources for being used in multiple provisioning stages as well as a long-term plan, e.g., four stages in a quarter plan and twelve stages in a yearly plan. The demand and price uncertainty is considered in RP. In particular, an optimal cloud resource provisioning is proposed to minimize the total cost for provisioning resources in a certain time period. The uncertainty in the demand from cloud consumer side and price uncertainty from cloud providers are taken into account to adjust the trade-off between on-demand and oversubscribed costs to make optimal decision.

Overload avoidance:
It is hard to deal with data that should be kept across the multiple requests in a user session while working in a load balanced service. What's more, in the event that this data is put away locally on one of the backend servers, resulting subsequent requests which are going to various backend servers would not have the option to discover it. Load-balancing a request to a different backend server just introduces a performance issue if the shared information is stored locally in one backend turn into a cached information that can be recomputed. Load balancers, to determine which backend server to send a request to by using a variety of scheduling algorithms. Simple algorithms for this module include random choice or round robin. ore sophisticated load balancers may take into account additional factors, such as a server's reported load, recent response times, up/down status (determined by a monitoring poll of some kind), number of active connections, geographic location, capabilities, or how much traffic it has as of late been allocated.

Minimizing skewness:
Application delivery controllers have evolved from basic server load balancing functional units to fully integrate with cloud workflows and provisioning systems so that they help users enable fast roll-out of new applications to a mobilized work force, improve end-user satisfaction, and reduce the time and cost of application deployment We introduce the concept of skewness to quantify the unevenness in the utilization of multiple resources on a server.

VM migrations:
We intend to move away the VM that can diminish the server's usage the most. If there should be an occurrence of ties, we select the VM whose evacuation can lessen the skewness of the server the most. For each VM in the rundown, we check whether we can discover an objective server to oblige it. The server should not turn into a problem area in the wake of tolerating this VM. Among every single such, we select one whose skewness can be diminished the most by tolerating this VM. Note that this decrease can be negative which implies we select the server whose skewness builds the least. On the off chance that an objective server is discovered, we record the movement of the VM to that worker and update the anticipated load of related server. Else, we move onto the following VM in the rundown and attempt to discover an objective server for it. However long we can discover an objective server for any of its VMs, we consider this run of the algorithm a triumph and afterward move onto the following problem area. Note that each run of the calculation moves away all things considered one VM from the overloaded server. This does not necessarily eliminate the hot spot, however in any event decreases its temperature. In the event that it stays a hotspot in the following decision run, the algorithm will repeat this process. It is feasible to design the algorithm with the goal that it can move away different VMs during each run. However, this can include more load the connected servers during a period when they are over-loaded.

6. DEVELOPING METHODOLOGIES

The test process is started by building up an exhaustive arrangement to test the overall usefulness and unique highlights on an assortment of stage blends. Severe quality control strategies are utilized. The process confirms that the application meets the prerequisites determined in the framework necessities document and is without bug. Coming up next are the contemplations used to build up the framework from building up the testing procedures.

Unit testing

Unit testing includes the design of tests that approve that the internal program logic is working appropriately, and that program input produces substantial yields. All decision branches and internal code flow ought to be approved. It is

the testing of individual programming units of the application. It is done after the finish of an individual unit before integration. This is a primary testing, that depends on information on its development and is obtrusive. Unit tests perform fundamental tests at component level and test a specific business process, application, and/or system configuration. Unit tests guarantee that every one-of-a-kind way of a business process performs precisely to the documented determinations and contains unmistakably determined inputs and anticipated outcomes.

Functional test

Functional tests give precise exhibits that capacities tried are accessible as determined by the business and specialized prerequisites, framework documentation, and client manuals.

Functional testing is fixated on the accompanying things:

Valid Input: distinguished classes of substantial info should be acknowledged.

Invalid Input: distinguished classes of invalid information should be dismissed.

Functions: distinguished capacities should be worked out.

Output: distinguished classes of utilization yields should be worked out.

Frameworks/Procedures: interfacing frameworks or systems should be conjured.

System Test

System testing guarantees that the whole incorporated programming framework meets necessities. It tests an arrangement to guarantee known and predictable outcomes. An illustration of system testing is the configuration-oriented system integration test. System testing depends on process descriptions and flows, emphasizing pre-driven process links and integration points.

Performance Test

The Performance test guarantees that the outcome is delivered as quick as possible, and the time taken by the framework for compiling, offering output to the clients and request being sent to the framework for to recover the outcomes.

Integration Testing

Software integration testing is the incremental integration testing of at least two coordinated programming segments on a solitary stage to deliver failures brought about by interface defects. The task of the integration test is to watch that segments or programming applications, for example segments in a software framework or – one stage up – programming applications at the organization level – associate without errors.

Acceptance Testing

User Acceptance Testing is a basic period of any project and requires huge cooperation by the end client. It additionally guarantees that the framework meets the utilitarian prerequisites.

Acceptance testing for Data Synchronization:

The Acknowledge will be received by the sender node after the packets are received by the destination node. The Route add activity is done just when there is a route demand in need. The status of nodes data is done consequently in the cache updating process.

Any project can be separated into units that can be additionally performed for detailed processing. At that point a testing system for every one of this unit is completed. Unit testing serves to identify the potential bugs in the individual segment, so the segment that has bugs can be recognized and can be redressed from errors.

FEASIBILITY STUDY:

The feasibility of the project is investigated in this stage and business agreement is advanced with an overall general plan for the project and some cost estimates. During framework analysis the feasibility study of the proposed system is to be completed. This is to guarantee that the proposed framework isn't a burden to the organization. For feasibility analysis, some understanding of the significant necessities for the framework is fundamental.

ECONOMICAL FEASIBILITY:

This study is done to check the financial effect that the framework will have on the association. The measure of asset that the organization can pour into the research and development of the framework is restricted. The expenditures should be defended. Thus, the developed system as well within the budget and this was accomplished on the grounds that the vast majority of the advances utilized are openly accessible. Just the customized products had to be purchased.

TECHNICAL FEASIBILITY:

This study is completed to check the technical feasibility, that is, the technical prerequisites of the framework. Any framework created should not have a popularity on the accessible technical assets. This will prompt high requests on the accessible technical assets. This will prompt high requests being set on the customer. The created framework should have an unobtrusive necessity, as just minimal or null changes are needed for executing this framework.

SOCIAL FEASIBILITY:

The aspect of study is to check the degree of acknowledgment of the framework by the user. This incorporates the way towards training the client to utilize the framework productively. The client should not feel undermined by the framework, rather should acknowledge it as a need. The degree of acknowledgment by the clients exclusively relies upon the strategies that are utilized to instruct the client about the framework and to make him acquainted with it. His degree of certainty should be raised so he is likewise ready to make some constructive criticism, which is invited, as he is the final user of the framework.

7. DATABASE DESIGN

Database design is characterized as an assortment of steps that assist with designing, creating, implementing, and maintaining a business's data management systems. The fundamental reason for planning a data set is to deliver physical and logical models of plans for the proposed data set framework. Database plan characterizes the database design utilized for planning, storing, and managing information. Precision in information can only be accomplished if a database is intended to store just important and fundamental data.

A well-designed database is basic in ensuring data consistency, disposing of repetitive information, effectively executing questions, and improving the presentation of the data set. Fastidiously planning a data set saves you from sitting around and getting disappointed during the information base improvement stage. A decent data set plan additionally permits you to handily get to and recover information at whatever point required. Here the database is designed to store the data about the users.

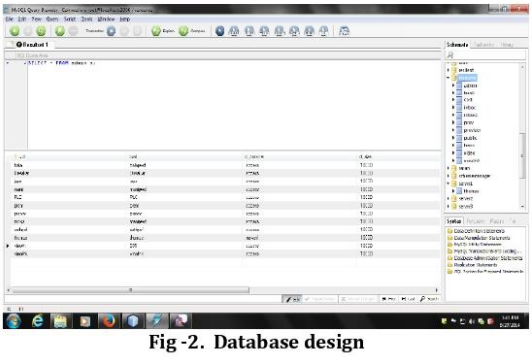

Fig -2. Database design

8. RESULT AND DISCUSSION

In order to make our investigations realistic we draw on a large number of data sources to create plausible demand models for long-duration video chat scenarios. The two scenarios chosen are from gaming, multi-party video poker, and education, a chat room for a massive open online course (MOOC). We gather data about participation in online poker and MOOCs both in terms of geographic distribution and in terms of time of day. To model the underlying cloud servers, we consider a scenario in which the locations and charges are based on Amazon EC2 and a future scenario in which 2,507 locations are taken from a data set of current data center locations. The modeling of delay is based on a global-scale delay measurement study we model millions of users per day joining and leaving chat rooms across the globe according to a stochastic demand model derived from analysis of user data. We combine different strategies for server selection and routing. The modeling produces estimates for the cost and delay for each user of the system over the course of many simulated days.

A dynamic choice of video routers will improve QoE when compared with an intelligent static choice. Choosing from a larger set of potential locations will improve QoE. The cost per user and the QoE will not worsen dramatically as the number of users in the system grows. Difference in demand and usage patterns will have significant impact on QoE and cost, even when the underlying infrastructure is the same. The routing policy used can have a large effect on delays within the system.
Using this application, the provider can get knowledge about the space consumption of the users. And migration between the servers is possible if one of the servers is overloaded.

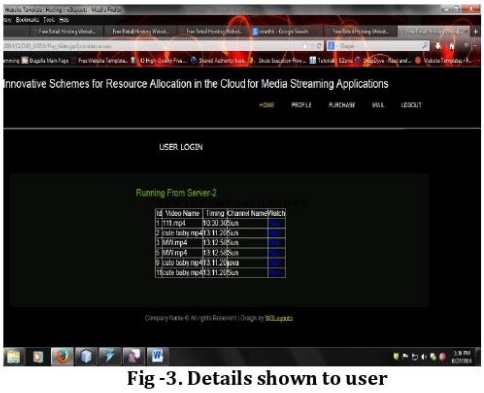

Fig -3. Details shown to user

Fig -4. Details accessed by provider

9. CONCLUSION

In this project considers recreations of a multi-client, cloud assisted worldwide video conferencing framework. We accept a video switch (for example as created by the Vconect venture) that can move between the cloud facilitated areas in a user transparent way. Utilizing sensible interest from genuine world information, we examined two diverse interest situations (one in view of gaming the other dependent on instruction). We thought about situations where video switch areas were chosen statically what's more, progressively. Our analyses were rushed to decide the postpone experienced by clients because of the decisions of cloud have area and courses for their video session.

10. FUTURE SCOPE

This paper studies the problem of resource allocations in the cloud for media streaming applications. We have considered non-linear time-discount tariffs that a cloud provider charges for resources reserved in the cloud.

In particular, there is scope to extend the existing capabilities of cloud-hosted IP Multimedia Subsystem (IMS) components—and to investigate the potential advantages of dynamic video routing within and between IMS clouds connected using IP Exchange (IPX) services. Another aspect of future work is to use our model to evaluate suitable locations for new data centers to host video routers to improve QoE beyond that achievable with current deployments. To aid this further investigation, the code and data to replicate our results are publicly available.

REFERENCES

[1] Shan-Hsiang Shen, "Efficient SVC Multicast Streaming for Video Conferencing with SDN Control," IEEE Transactions on Network and Service Management, vol. 16, pp. 403 - 416, June 2019.

[2] E. Coronado, R. Riggio, J. Villalón, and A. Garrido, "Efficient real-time content distribution for multiple multicast groups in SDNbased WLANs," IEEE Trans. Netw. Service Manag., vol. 15, no. 1, pp. 430–443, Mar. 2018.

[3] Mohammad H. Hajiesmaili, Lok To Mak, Zhi Wang, Chuan Wu, Minghua Chen, and Ahmad Khonsari, "Cost-Effective Low-Delay Design for Multiparty Cloud Video Conferencing," in IEEE Transactions on Multimedia, vol: 19, pp. 2760 - 2774, Dec. 2017

[4] A. Bentaleb, A. C. Begen, R. Zimmermann, and S. Harous, "SDNHAS: An SDN-enabled architecture to optimize QoE in HTTP adaptive streaming," IEEE Trans. Multimedia, vol. 19, no. 10, pp. 2136–2151, Oct. 2017.

[5] S. Y. Shah, B. Szymanski, P. Zerfos, and C. Gibson, "Towards relevancy aware service-oriented systems in wsns," IEEE Transactions on Service Computing, vol. 8, 2015.

[6] S. Agarwal, M. Kodialam, and T. V. Lakshman, "Traffic engineering in software defined networks," in Proc. IEEE INFOCOM, 2013, pp. 2211–2219.

[7] I. Papapanagiotou, M. Falkner, and M. Devetsikiotis, "Optimal functionality placement for multiplay service provider architectures," IEEE Trans. Network and Service Management, vol. 9, no. 3, pp. 359–372, 2012.

[8] D. Roberts, T. Duckworth, C. Moore, R. Wolff, , and J. O'Hare, "Comparing the end to end latency of an immersive collaborative environment and a video conference," in Proceedings of the IEEE/ACM international Symposium on Distributed Simulation and Real Time Applications, 2009, pp. 89–94.

[9] M. Chen, G. Su, and M. Wu, "Dynamic resource allocation for robust distributed multi-point video conferencing," IEEE Trans. on Multimedia, vol. 10, no. 5, August 2008.

[10] N. Laoutaris, G. Smaragdakis, K. Oikonomou, I. Stavrakakis, and A. Bestavros, "Distributed placement of service facilities in large-scale networks," in IEEE INFOCOM, 2007, pp. 2144–2152.

[11] Y. Chu, S. Rao, S. Seshan, and H. Zhang, "End-to-end delay analysis of videoconferencing over packet-switched networks," IEEE/ACM Transactions on Networking, vol. 8, no. 4, pp. 479–492, 2000.