Sudan University of Science and Technology

College of Graduate Studies

Two-Factor Authentication and Role-based Access Control for Cloud Services

المصادقة ثنائية العامل والتحكم في الوصول استنادًا على الدور لخدمات الحوسبة السحابية

A Thesis Submitted in Partial Fulfillment for the Requirements of the Degree of M.Sc. In Electronics Engineering (Computer and Networks Engineering)

Prepared By:

Duaa Abdelmuniem Mohamed Osman

Supervised By:

Dr. Salaheldin Mohamed Ibrahim Edam

September 2019

Sudan University of Science & Technology

College of Graduate Studies

بسم الله الرحمن الرحيم

جامعة السودان للعلوم والتكنولوجيا

كلية الدراسات العليا

كلية الدراسات العليا

Approval Page

(To be completed after the college council approval)

Name of Candidate: Duaa | Abdelmoniem | Mohamed | Osman

Thesis title: Two-Factor Authentication and Role-based Access Control for Cloud Services

المصادقة الثنائية والتحكم في الوصول القائم على الدور لخدمات الحوسبة السحابية

Degree Examined for: .....

.....

Approved by:

1. External Examiner

Name: Dr. Elsadig Saeed Gebreel

Signature: ..... Date: 24/7/2019

2. Internal Examiner

Name: Ebtihal Haider Gismalla Yousif

Signature: ..... Date: 24/7/2019

3. Supervisor

Name: Dr. Salaheldin M. I. Edam

Signature: ..... Date: 24/7/2019

ص.ب 407 فاكس/ 83 769363 البريد الالكتروني cgs @ sustech edu.

ACKNOWLEDGMENT

I would like to gratefully thank **Allah** for his support and help to complete this work. For sometimes during the process, I felt it might be impossible to finish the whole thesis.

Apart from the efforts of myself, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to several individuals who have been instrumental in the successful completion of this work.

To my Supervisor, **Dr. Salaheldin Mohamed Ibrahim Edam**, thank you for your huge efforts and continued support. His guidance helped me in all the time. Without his encouragement and guidance, this thesis would not have materialized.

I would like to express my deepest appreciation to **Miss Sondos Wasfi** who dedicated her precious time to help me. She is always been there**.**

Last but not least, gratitude shall be expressed to ***my beloved family***; for their understanding & endless love, through the duration of my studies.

DEDICATION

To my loving parents, Abdelmuniem and Sameera

To my loving brothers; Mohammed and Ahmed

To my loving sisters; Rehab, Mona, and Fedaa

To my loving friends

They always inspired me to be a successful and educated person

Thanks all for supporting me spiritually throughout my life

May you also be motivated and encouraged to reach your dreams

ABSTRACT

Cloud computing became a tendency technology for IT and non-IT organizations. Many sectors and organizations shift towards cloud computing technology. They adopt this technology to deploy their services. Many services and applications gained benefits from cloud computing since these services become accessible over the whole world via the cloud. Cloud computing mitigates the burden of constructing and maintaining large data centers for non-IT sectors. Cloud providers offer On-Demand, scalable and measured services for cloud consumers to deploy their services and store their sensitive data. Accessing the cloud is one of the major security issues. Cloud provider should apply robust access model to gain the trust of their customers. They should restrict the access and grant the consumer the least privileges that are needed to accomplish his/her task. For that role-based access control model adopted in this thesis. This model is implemented to protect cloud services from unauthorized access. A mechanism of the time-based one-time password as two-factor authentication is also implemented to overcome the weakness of static password. A model of an online education system was created as an example of a cloud service. The system was developed using PHP language. The system has different roles each role has the least permission was needed. And for authentication BLOWFISH BCRYPT hashing function was used to implement time-based one time password. This password is renewed every session and has a timestamp of 30 seconds and then became unusable. The system has the capability to authenticate and authorize users of different roles. One time password raises the reliability of the authentication process by sending the password to the user's email.

المستخلص

أصبحت الحوسبة السحابية التقنية التي تتجه اليها جميع الجهات والمؤسسات سواء كانت متعلقة بمجال تكنولوجيا المعلومات او لا. تحولت العديد من القطاعات والمؤسسات نحو تكنولوجيا الحوسبة السحابية. ويعتمدون هذه التقنية لنشر خدماتهم. استفادت العديد من الجهات من مميزات الحوسبة السحابية، حيث بات من المستطاع نشر الخدمات واتاحتها عبر العالم من خلال السحابة. خففت الحوسبة السحابية عبء انشاء وصيانة مراكز البيانات الكبيرة للقطاعات الغير متعلقة بتكنولوجيا المعلومات. يوفر مقدمو خدمات السحابة خدمات عند الطلب وقابلة للتوسع وقياسية للعملاء لنشر خدماتهم وتخزين بياناتهم الحساسة. يعد الوصول إلى السحابة أحد المشكلات الأمنية الرئيسية. يجب على مزود السحابة تطبيق نموذج تحكم دخول للسحابة قوي لكسب ثقة عملائه. يجب أن تحد من الوصول للسحابة عن طريق منح المستهلك أقل الامتيازات اللازمة لإنجاز مهمته. لهذا تم اعتماد نموذج التحكم في الوصول استناداً على الدور في هذه الأطروحة. تم تطبيق هذا النموذج لحماية خدمات الحوسبة السحابية من الوصول غير المصرح به. وتم أيضًا تطبيق آلية كلمة مرور مؤقتة محددة بزمن معين كمصادقة ثنائية–العوامل للتغلب على ضعف كلمة المرور الثابتة. تم إنشاء نموذج لنظام تعليم عبر الإنترنت كمثال لخدمة على الحوسبة السحابية. تم تطوير النظام باستخدام لغة PHP. النظام له أدوار مختلفة لكل دور الامتيازات التي يحتاجها المستخدم فقط. وللمصادقة تم استخدام دالة BLOWFISH BCRYPT لتطبيق آلية كلمة مرور مؤقتة محددة بزمن معين. يتم تجديد كلمة المرور هذه في كل اتصال بالنظام ولها طابع زمني مدته 30 ثانية ثم يصبح غير صالح لتسجيل الدخول للنظام. النظام لديه القدرة على مصادقة وتفويض المستخدمين بامتيازات مختلفة كل

مستخدم حسب دوره. تزيد كلمة مرور مؤقتة محددة بزمن معين من موثوقية عملية المصادقة عن طريق إرسال كلمة المرور إلى بريد المستخدم الإلكتروني.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

CGI	Common Gateway Interface
DOS	Denial of Service
HMAC	Hash Message Authentication code
HOTP	Hashed-based One Time Password
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
IaaS	Infrastructure as a Service
IAM	Identity and Access Management
MD5	Message Digest
SQL	Structured Query Language
NIST	National Institute of Standards and Technology
OTP	One Time Password
PaaS	Platform as a Service
PHP	Hypertext Preprocessor
RBAC	Role-based Access Control
SaaS	Software as a Service
SHA	Secure Hashing Function
SME	Small and Medium-Sized Enterprise
SSO	Single Sign-On
TOTP	Time-based One Time Password
VM	Virtual Machine
WAMP	Windows, Apache, MySQL, PHP

LIST OF SYMBOLS

K	Key of One Time Password Algorithm
C	Counter of One Time Password Algorithm
T_0	Initial Counter Time for One Time Password Algorithm
T	Number of Time Steps Between The Initial Counter Time T_0 and The Current UNIX Time.

CHAPTER ONE

INTRODUCTION

Chapter One

Introduction

1.1. Preface

Cloud computing is one of the applied technologies for utility computing. It based on the concept of on-demand sharing resources over the internet. These resources are networks, servers, storage, applications, and services. Cloud computing is available in pay as you go manner [1].

Cloud computing has main three models. The infrastructure as a service (IaaS) offers hardware resources and physical assets like Amazon EC2. The Platform as a service (PaaS) offers a development environment for deploying customer's applications like Google App Engine. The software as a service (SaaS) offers applications like Microsoft online office [1].

In recent years cloud computing became a tendency technology in IT market. It became the most cost-effective architecture for the business environment in IT and non-IT sectors. It reduces the expenses of deploying and maintaining large data centers. Also, it eliminates the necessity of expertise IT staff for non-IT business.

Security of cloud computing is one of the crucial issues and challenges. It is important to deploy a secure cloud and resistant to attacks. Cloud providers should implement a robust access control mechanism for their infrastructure. Resources and data in the cloud are highly sensitive. Once a user wants to access any of cloud's resource she/she should authenticate his/her self to ensure that only authorized users can access the

cloud's resources [2]. A well-designed access control solutions need to be adopted in order to prevent unauthorized access to the cloud.

One time password (OTP) is a password used to authenticate the user for one session and for a limited time and then becomes expired. This mechanism is beneficial in two ways. Firstly, it protects the cloud from attackers even though the attackers compromise the password it's not valid for access. Secondly, it mitigates the overhead that the user suffering from remembering complicated passwords [3]. OTP mechanism was brought to overcome the weakness of static password since it is vulnerable to malicious adversaries [4]. Cryptography algorithm and hash function have been used to generate OTP [5].

1.2. Problem Statement

One of the major mechanisms to authenticate a user accessing the cloud is a username and password which is the biggest vulnerability used by attackers to break through the cloud. Internal users could do malicious attacks accidentally or intentionally, these users may have full access to the cloud that considered a security threat of the cloud.

1.3. Proposed Solution

This thesis proposed two-factor authentication and role-based access control solution. The time-based one-time password will be used to overcome static password weakness as two-factor authentication using the BLOWFISH hashing mechanism. Also, the PHP language will be used to set privileges according to the user role.

1.4. Objectives

This thesis aim to implement enhanced access control solution for cloud services, in addition to the following objectives:

- To enhance authentication of cloud users by adding a one-time password mechanism as two-factor authentication.
- To increase access control capability to set accounts privileges and permissions according to the user's role.

1.5. Methodology

The research methodology of this thesis divided into five stages. The first stage is the creation of a cloud service to test the proposed scenario using WampServer which composed of apache2, PHP, and MySQL. The second stage is the configuration of role-based access control and the integration of it with the cloud service using PHP language. The third stage is the implementation of time-based OTP using BLOWFISH BCRYPT hash function. The fourth stage is testing the designed system using hMailServer and Thunderbird interface to verify that the one time password sent to user's email. hMailServer is an open source e-mail server and Thunderbird is a free e-mail client application.

1.6. Thesis Layout

Chapter two contains a general overview of cloud computing, its models and characteristics, and security issues. In addition, some of the researches related to the one-time password and access control models in

cloud computing will be highlighted. Chapter three contains a detailed explanation of one-time password, role-based access control model and the developed scenario. Also, a brief definition of the tools used to implement the developed scenario. Chapter four illustrate the result and the validation of the proposed solution. Chapter five present the conclusion of the thesis work and recommendation for future work.

CHAPTER TWO

LITERATURE REVIEW

Chapter Two

Literature Review

This chapter contains a general overview of cloud computing, its models and characteristics, and security issues. In addition, some of the researches related to the one-time password and access control models in cloud computing will be highlighted.

2.1. Background

The term cloud has been used historically as a metaphor for the Internet. This usage was originally derived from its common depiction in network diagrams as an outline of a cloud, used to represent the transport of data across carrier backbones (which owned the cloud) to an endpoint location on the other side of the cloud [6].

NIST published its original definition in 2009, followed by a revised version after further review and industry input that was published in September of 2011: “Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.”[7]

The use of cloud computing has increased rapidly in many organizations. Cloud Computing was predicted to transform the computing world from using local applications and storage into centralized services provided by the organization. In addition, development of large-scale, on-

demand, flexible computing infrastructures is another benefit of cloud computing services [8, 9].

To users, cloud computing is a Pay-per-Use-On-Demand mode that can conveniently access shared IT resources through the Internet. Where the IT resources include network, server, storage, application, service and so on and they can be deployed with a much quick and easy manner, least management and interactions with service providers. Cloud computing can much improve the availability of IT resources and owns many advantages over other computing techniques. Users can use the IT infrastructure with Pay-per-Use-On-Demand mode; this would benefit and save the cost to buy the physical resources that may be vacant [10].

Cloud computing has brought a new paradigm shift in the technology industry and becoming increasingly popular day by day. The SMEs are adopting cloud computing for the low-cost implementation of total IT infrastructure and software system whereas the large enterprises are relying on their own infrastructure for data security, privacy and flexibility to access their own infrastructure. Cloud computing is a modern computing technology where software and hardware infrastructure of an enterprise can be placed over a network to access later in on-demand basis via the internet instead of having them locally within the enterprise. Cloud computing service provider holds the responsibility to manage and share all the hardware and software using virtualization among the clients, and the clients only pay for the subscribed services. Cloud computing turned out to be useful for small to medium enterprises (SMEs) in order to have low implementation cost for their total IT infrastructure and software systems. For SMEs, the pay per user basis service license drastically reduces the cost of both hardware and software. Cloud computing is adopted mostly by the SMEs due to its numerous benefits. An enterprise must check several things in order to move

into the clouds. Firstly, the customer must technically comply with the existing cloud system. Secondly, moving the data to the cloud should not violate any security law of the nation or break the customer data privacy policy. Thirdly, the internal must allow executing the workloads on the cloud environment. Finally, companies must prepare their business process in a way that on-demand cloud products can be acquired whenever required [11].

In a cloud computing environment, the traditional role of service providers is divided into two: cloud providers who own the physical data center and lease resources (e.g., virtual machines or VMs) to service providers; and service providers who use resources leased by cloud providers to execute applications [12].

The biggest challenges cloud and service providers face are secure data storage, high-speed access to the Internet, and standardization. Storing large amounts of data that is oriented around user privacy, identity, and application-specific preferences in centralized locations raise many concerns about data protection. These concerns, in turn, give rise to questions regarding the legal framework that should be implemented for a cloud-oriented environment. Cloud computing is untenable without high-speed connections (both wired and wireless). Unless broadband speeds are available, cloud computing services cannot be made widely accessible [6].

2.1.1. Cloud Computing Enabling Technology

The following technologies considered the primary technologies in cloud computing.

- **Clustering**

A cluster is a group of independent IT resources that are interconnected and work as a single system. System failure rates are reduced

while availability and reliability are increased since redundancy and failover features are inherent to the cluster. A general prerequisite of hardware clustering is that its component systems have reasonably identical hardware and operating systems to provide similar performance levels when one failed component is to be replaced by another. Component devices that form a cluster are kept in synchronization through dedicated, high-speed communication links. The basic concept of built-in redundancy and failover is core to cloud platforms [7].

- **Grid Computing**

A computing grid (or "computational grid") provides a platform in which computing resources are organized into one or more logical pools. These pools are collectively coordinated to provide a high performance distributed grid, sometimes referred to as a "super virtual computer". The technological advancements achieved by grid computing projects have influenced various aspects of cloud computing platforms and mechanisms, specifically in relation to common feature-sets such as networked access, resource pooling, and scalability and resiliency. These types of features can be established by both grid computing and cloud computing, in their own distinctive approaches. For example, grid computing is based on a middleware layer that is deployed on computing resources. These IT resources participate in a grid pool that implements a series of workload distribution and coordination functions. This middle tier can contain load balancing logic, failover controls, and autonomic configuration management. It is for this reason that some classify cloud computing as a descendant of earlier grid computing initiatives [7].

- **Virtualization**

Virtualization represents a technology platform used for the creation of virtual instances of IT resources. A layer of virtualization software allows physical IT resources to provide multiple virtual images of themselves so that their underlying processing capabilities can be shared by multiple users. Prior to the advent of virtualization technologies, the software was limited to residing on and being coupled with static hardware environments. The virtualization process severs this software-hardware dependency, as hardware requirements can be simulated by emulation software running in virtualized environments [7].

2.1.2. Cloud Computing Architecture

The architecture of the cloud computing environment can be divided into four layers: the hardware/datacenter layer, the infrastructure layer, the platform layer, and the application layer. These layers described in details as follows:

- The hardware layer, this layer is responsible for managing the physical resources of the cloud, including physical servers, routers, switches, power, and cooling systems. In practice, the hardware layer is typically implemented in data centers. A data center usually contains thousands of servers that are organized in racks and interconnected through switches, routers, or other fabrics. Typical issues at the hardware layer include hardware configuration, fault-tolerance, traffic management, and power and cooling resource management [12].
- The infrastructure layer Also known as, the virtualization layer, the infrastructure layer creates a pool of storage and computing resources by partitioning the physical resources using virtualization technologies such as Xen, KVM, and VMware. The infrastructure

layer is an essential component of cloud computing, since many key features, such as dynamic resource assignment, are only made available through virtualization technologies [12].

- The platform layer built on top of the infrastructure layer, the platform layer consists of operating systems and application frameworks. The purpose of the platform layer is to minimize the burden of deploying applications directly into VM containers. For example, Google App Engine operates at the platform layer to provide API support for implementing storage, database, and business logic of typical Web applications [12].
- The application layer, at the highest level of the hierarchy, the application layer consists of the actual cloud applications. Different from traditional applications, cloud applications can leverage the automatic-scaling feature to achieve better performance, availability, and lower operating cost. Compared to traditional service hosting environments such as dedicated server farms, the architecture of cloud computing is more modular. Each layer is loosely coupled with the layers above and below, allowing each layer to evolve separately. This is similar to the design of the protocol stack model for network protocols. The architectural modularity allows cloud computing to support a wide range of application requirements while reducing management and maintenance overhead [12].

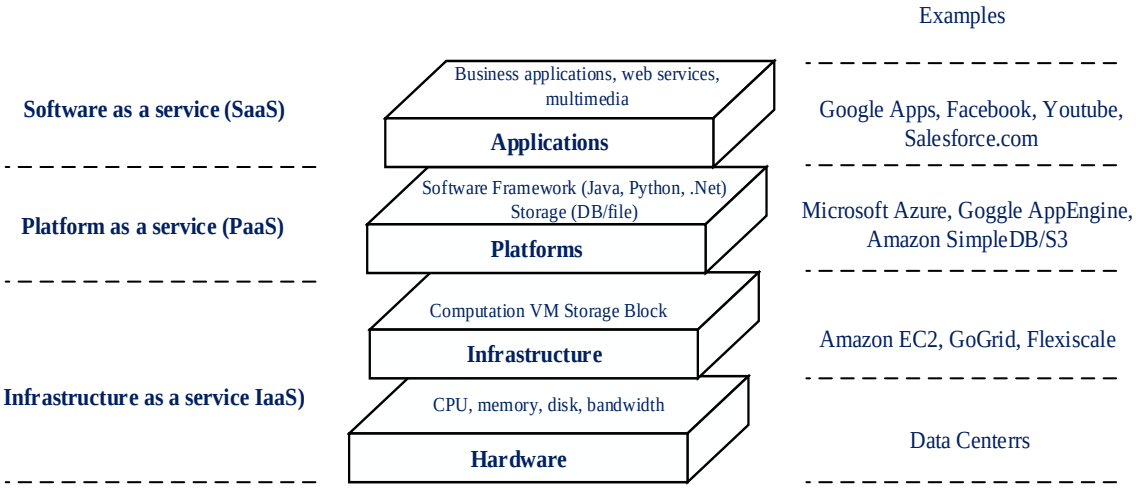

Figure 2.1. Cloud Computing Architecture

2.1.3. Cloud Delivery Models

Infrastructure-as-a-Service (IaaS), The IaaS delivery model represents a self-contained IT environment comprised of infrastructure-centric IT resources that can be accessed and managed via cloud service-based interfaces and tools. This environment can include hardware, network, connectivity, operating systems, and other "raw" IT resources. In contrast to traditional hosting or outsourcing environments, with IaaS, IT resources are typically virtualized and packaged into bundles that simplify up-front runtime scaling and customization of the infrastructure. The general purpose of an IaaS environment is to provide cloud consumers with a high level of control and responsibility for its configuration and utilization. The IT resources provided by IaaS are generally not pre-configured, placing the administrative responsibility directly upon the cloud consumer. This model is therefore used by cloud consumers that require a high level of control over the cloud-based environment they intend to create [7].

Platform-as-a-Service (PaaS), The PaaS delivery model represents a pre-defined "ready-to-use" environment typically comprised of already deployed and configured IT resources. By working within a ready-made

platform, the cloud consumer is spared the administrative burden of setting up and maintaining the bare infrastructure IT resources provided via the IaaS model. Conversely, the cloud consumer is granted a lower level of control over the underlying IT resources that host and provision the platform. PaaS products are available with different development stacks. For example, Google App Engine offers a Java and Python-based environment [7].

Software-as-a-Service (SaaS), a software program positioned as a shared cloud service and made available as a "product" or generic utility represents the typical profile of a SaaS offering. The SaaS delivery model is typically used to make a reusable cloud service widely available (often commercially) to a range of cloud consumers. An entire marketplace exists around SaaS products that can be leased and used for different purposes and via different terms. A cloud consumer is generally granted very limited administrative control over a SaaS implementation. It is most often provisioned by the cloud provider, but it can be legally owned by whichever entity assumes the cloud service owner role [7].

2.1.4. Cloud Services Deployment Models

Public Cloud is a publicly accessible cloud environment owned by a third-party cloud provider. The IT resources on public clouds are usually provisioned via the previously described cloud delivery models and are generally offered to cloud consumers at a cost or are commercialized via other avenues (such as an advertisement). The cloud provider is responsible for the creation and on-going maintenance of the public cloud and its IT resources [7].

Private Cloud is owned by a single organization. Private clouds enable an organization to use cloud computing technology as a means of centralizing access to IT resources by different parts, locations, or

departments of the organization. The actual administration of a private cloud environment may be carried out by internal or outsourced staff [7].

Hybrid Cloud is a cloud environment comprised of two or more different cloud deployment models. For example, a cloud consumer may choose to deploy cloud services processing sensitive data to a private cloud and other, less sensitive cloud services to a public cloud. The result of this combination is a hybrid deployment model. Hybrid deployment architectures can be complex and challenging to create and maintain due to the potential disparity in cloud environments and the fact that management responsibilities are typically split between the private cloud provider organization and the public cloud provider [7].

Community Cloud is similar to a public cloud except that its access is limited to a specific community of cloud consumers. The community cloud may be jointly owned by the community members or by a third party cloud provider that provisions a public cloud with limited access. The member cloud consumers of the community typically share the responsibility for defining and evolving the community cloud. Membership in the community does not necessarily guarantee access to or control of all the cloud's IT resources. Parties outside the community are generally not granted access unless allowed by the community [7].

2.1.5. Cloud Computing Characteristics

The five essential characteristics of cloud computing include the following:

- **On-Demand Self-Service**

On-demand self-service enables users to use cloud computing resources as needed without human interaction between the user and the

cloud service provider. With on-demand self-service, a consumer can schedule the use of cloud services such as computation and storage as needed, in addition to managing and deploying these services. In order to be effective and acceptable to the consumer, the self-service interface must be user-friendly and provide effective means to manage the service offerings. This ease of use and elimination of human interaction provides efficiencies and cost savings to both the user and the cloud service provider [13].

- **Broad Network Access**

Broad network access, which states that cloud services, can be accessed remotely from heterogeneous client platforms (e.g., mobile phones). For cloud computing to be an effective alternative to in-house data centers, high-bandwidth communication links must be available to connect to the cloud services. One of the principal economic justifications for cloud computing is that the lowered cost of high-bandwidth network communication to the cloud provides access to a larger pool of IT resources that sustain a high level of utilization [6, 13].

- **Location-Independent Resource Pooling**

The cloud must have a large and flexible resource pool to meet the consumer's needs, provide economies of scale, and meet service level requirements. Applications require resources for their execution, and these resources must be allocated efficiently for optimum performance. The resources can be physically located at many geographic locations and assigned as virtual components of the computation as needed. As stated by NIST,5 "There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter)."[13]

- **Rapid Elasticity**

Rapid elasticity refers to the ability of the cloud to expand or reduce allocated resources quickly and efficiently to meet the requirements of the self-service characteristic of cloud computing. This allocation might be done automatically and appear to the user as a large pool of dynamic resources that can be paid for as needed and when needed. One of the considerations in enabling rapid elasticity is the development and implementation of loosely coupled services that scale independently of other services and are not dependent on the elasticity of these other services [13].

- **Measured Service**

Because of the service-oriented characteristics of cloud computing, the number of cloud resources used by a consumer can be dynamically and automatically allocated and monitored. The customer can then be billed based on the measured usage of only the cloud resources that were allotted for the particular session. The NIST view of measured service is “Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported providing transparency for both the provider and consumer of the utilized service [13].

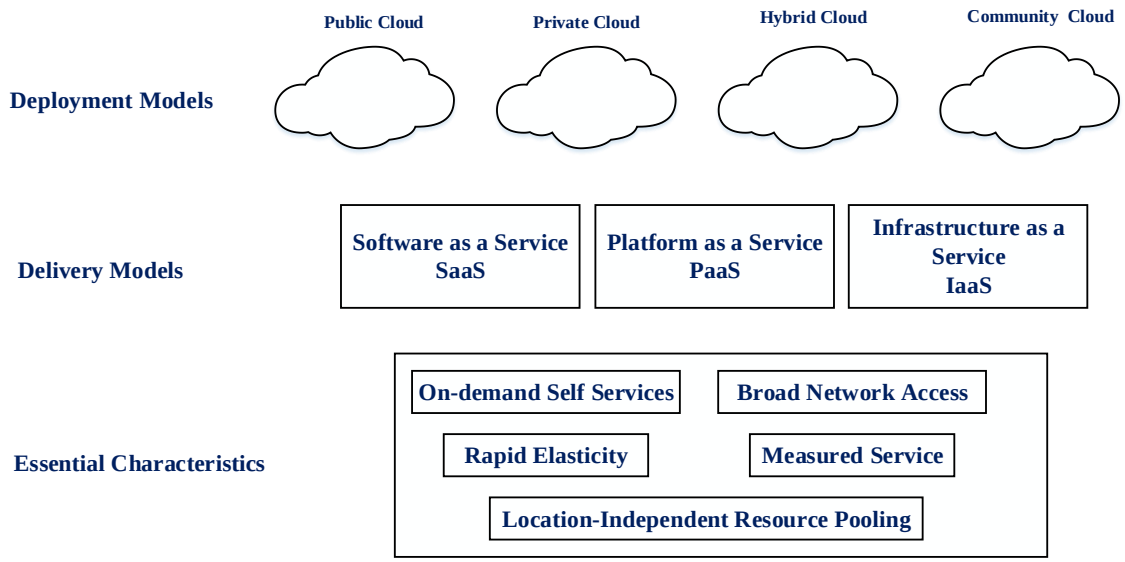

Figure 2.2. Cloud Computing Models and Characteristics

2.1.6. Key Driving Forces

There are several driving forces behind the success of cloud computing. The increasing demand for large-scale computation and big data analytics and economics are the most important ones. But other factors such as easy access to computation and storage, flexibility in resource allocations, and scalability play important roles.

Large-scale computation and big data: Recent years have witnessed the rise of Internet-scale applications. These applications range from social networks (e.g., Facebook, Twitter), video applications (e.g., Netflix, YouTube), enterprise applications (e.g., SalesForce, Microsoft CRM) to personal applications (e.g., iCloud, Dropbox). Large numbers of users commonly access these applications over the Internet. They are extremely large scale and resource intensive. Furthermore, they often have high-performance requirements such as response time. Supporting these applications requires extremely large-scale infrastructures. For instance, Google has hundreds of compute clusters deployed worldwide with hundreds

of thousands of servers. Another salient characteristic is that these applications also require access to huge volumes of data. For instance, Facebook stores tens of petabytes of data and processes over a hundred terabytes per day. Scientific applications (e.g., brain image processing, astrophysics, ocean monitoring, and DNA analysis) are more and more deployed in the cloud. Cloud computing emerged in this context as a computing model designed for running large applications in a scalable and cost-efficient manner by harnessing massive resource capacities in data centers and by sharing the data center resources among applications in an on-demand fashion [12].

Economics: To support large-scale computation, cloud providers rely on inexpensive commodity hardware offering better scalability and performance/price ratio than supercomputers. By deploying a very large number of commodity machines, they leverage economies of scale bringing per unit cost down and allowing for incremental growth. On the other hand, cloud customers such as small and medium enterprises, which outsource their IT infrastructure to the cloud, avoid upfront infrastructure investment cost and instead benefit from a pay-as-you-go pricing and billing model. They can deploy their services in the cloud and make them quickly available to their own customers resulting in a short time to market. They can start small and scale up and down their infrastructure based on their customers demand and pay based on usage [12].

Scalability: By harnessing huge computing and storage capabilities, cloud computing gives customers the illusion of infinite resources on demand. Customers can start small and scale up and down resources as needed [12].

Flexibility: Cloud computing is highly flexible. It allows customers to specify their resource requirements in terms of CPU cores, memory, storage, and networking capabilities. Customers are also offered the flexibility to customize the resources in terms of operating systems and possibly network stacks [12].

Easy access: Cloud resources are accessible from any device connected to the Internet. These devices can be traditional workstations and servers or less traditional devices such as smartphones, sensors, and appliances. Applications running in the cloud can be deployed or accessed from anywhere at any time [12].

2.1.7. Cloud Security Services

Additional factors that directly affect cloud services assurance include authentication, authorization, auditing, and accountability, as summarized in the following sections.

- **Authentication**

Authentication is the testing or reconciliation of evidence of a user's identity. It establishes the user's identity and ensures that users are who they claim to be. For example, a user presents an identity (user ID) to a computer login screen and then has to provide a password. The computer system authenticates the user by verifying that the password corresponds to the individual presenting the ID [13].

- **Authorization**

Authorization refers to rights and privileges granted to an individual or process that enable access to computer resources and information assets. Once a user's identity and authentication are established, authorization levels determine the extent of system rights a user can hold [13].

- **Auditing**

To maintain operational assurance, organizations use two basic methods: system audits and monitoring. These methods can be employed by the cloud customer, the cloud provider, or both, depending on asset architecture and deployment [13].

- A system audit is a one-time or periodic event to evaluate security.
- Monitoring refers to an ongoing activity that examines either the system or the users, such as intrusion detection.

- **Accountability**

Accountability is the ability to determine the actions and behaviors of a single individual within a cloud system and to identify that particular individual. Audit trails and logs support accountability and can be used to conduct postmortem studies in order to analyze historical events and the individuals or processes associated with those events. Accountability is related to the concept of nonrepudiation, wherein an individual cannot successfully deny the performance of an action [13].

2.1.8. Cloud Security Mechanisms

The following section briefly describes major security mechanisms that are used to mitigate cloud security threats.

- **Encryption**

Data, by default, is coded in a readable format known as plaintext. When transmitted over a network, the plaintext is vulnerable to unauthorized and potentially malicious access. The encryption mechanism is a digital coding system dedicated to preserving the confidentiality and integrity of data. It is used for encoding plaintext data into a protected and unreadable

format. Encryption technology commonly relies on a standardized algorithm called a cipher to transform original plaintext data into encrypted data, referred to as ciphertext. When encryption is applied to plaintext data, the data is paired with a string of characters called an encryption key, a secret message that is established by and shared among authorized parties. The encryption key is used to decrypt the ciphertext back into its original plaintext format. The encryption mechanism can help counter the traffic eavesdropping, malicious intermediary, insufficient authorization, and overlapping trust boundaries security threats [7].

- **Hashing**

The hashing mechanism is used when a one-way, non-reversible form of data protection is required. Once hashing has been applied to a message, it is locked and no key is provided for the message to be unlocked. A common application of this mechanism is the storage of passwords. Hashing technology can be used to derive a hashing code or message digest from a message, which is often of a fixed length and smaller than the original message. The message sender can then utilize the hashing mechanism to attach the message digest to the message. The recipient applies the same hash function to the message to verify that the produced message digest is identical to the one that accompanied the message. Any alteration to the original data results in an entirely different message digest and clearly indicates that tampering has occurred. In addition to its utilization for protecting stored data, the cloud threats that can be mitigated by the hashing mechanism include malicious intermediary and insufficient authorization [7].

- **Digital Signature**

The digital signature mechanism is a means of providing data authenticity and integrity through authentication and non-repudiation. A message is assigned a digital signature prior to transmission, which is then rendered invalid if the message experiences any unauthorized modifications. A digital signature provides evidence that the message received is the same as the one created by its rightful sender. The digital signature mechanism helps mitigate the malicious intermediary, insufficient authorization [7].

- **Identity and Access Management (IAM)**

The identity and access management (IAM) mechanism encompasses the components and policies necessary to control and track user identities and access privileges for IT resources, environments, and systems. Specifically, IAM mechanisms exist as systems comprised of four main components [7]:

- Authentication– Username and password combinations remain the most common forms of user authentication credentials managed by the IAM system, which also can support digital signatures, digital certificates, biometric hardware (fingerprint readers).
- Authorization– The authorization component defines the correct granularity for access controls and oversees the relationships between identities, access control rights, and IT resource availability.
- User Management– Related to the administrative capabilities of the system, the user management program is responsible for creating new user identities and access groups, resetting passwords, defining password policies, and managing privileges.
- Credential Management– The credential management system establishes identities and access control rules for defined user

accounts, which mitigates the threat of insufficient authorization. The IAM mechanism is primarily used to counter the insufficient authorization, denial of service, and overlapping trust boundaries threats.

- **Single Sign-On (SSO)**

The single sign-on (SSO) mechanism enables one cloud service consumer to be authenticated by a security broker, which establishes a security context that is persisted while the cloud service consumer accesses other cloud services or cloud-based IT resources. Otherwise, the cloud service consumer would need to re-authenticate itself with every subsequent request. The SSO mechanism essentially enables mutually independent cloud services and IT resources to generate and circulate runtime authentication and authorization credentials. The credentials initially provided by the cloud service consumer remain valid for the duration of a session, while its security context information is shared. The SSO mechanism's security broker is especially useful when a cloud service consumer needs to access cloud services residing on different clouds [7].

2.1.9. Cloud computing Security Threats

This section highlights security threats in cloud computing

- **Data Loss,** Data can be compromised because of deletion, modification, loss of encryption key and by other means like earthquakes, floods and fires etc. Organizations should maintain a comprehensive backup of their data to avoid such threats [14].
- **Data Breaches** refer to leakage of sensitive information to unauthorized users. Data breaches can occur because of improper

authentication and authorization mechanisms, unreliable use of encryption keys and operating system failure [14].

- **Account or Service Hijacking** occurs if an attacker gains access to login credentials, then the compromised account becomes a launching base and attacker can eavesdrop on the consumer businesses, refund false info, manipulate data and can reply to sessions and redirect the consumer to illegitimate sites and can launch various attacks [14].
- **Insecure Interfaces and APIs** refer to Application Programming Interfaces which are standards and protocols that consumers use to connect with cloud services. As the security of cloud services depends on these APIs so these should have secure certification standards, proper access controls and activity monitoring mechanisms to avoid threats like anonymous access, clear-text authentication, reusable tokens or passwords, improper authorization, limited monitoring, and logging capabilities [14].
- **Malicious Insiders** can be trusted people within an organization who can access organizational confidential assets. They can perform unprivileged activities to infiltrate organizational assets and can do brand damage, productivity and financial losses by means of conducting different activities like the firewall or Intrusion Detection System (IDS) pretending it to be a legal activity [14].
- **Insufficient Due Diligence** occurs when organizations jump into using services offered by service providers without having sufficient knowledge of the cloud models and its operations and without understanding which model fits for them along with the risks associated with it [14].

- **Abusive Use of Cloud Services** can be described as consumer's unethical and illegal actions to misuse the services. Low-cost infrastructure, high-resource, provisioning, weak registration procedures have facilitated anonymity to spammers, criminals, and other malicious users to achieve their target in attacking the system. Cloud services providers such as Amazon, Google, Facebook, Twitter etc. have been used to launch Trojans and Botnets [14].
- **Shared Technology Issues** occur in a multi-tenant framework, where on-demand services are delivered using shared infrastructure among different users having access to same VM. Vulnerabilities in virtualized hypervisors (use for isolation purpose) allow malicious consumers to have inappropriate access and to control legitimate consumers VMs [14].
- **Identity Theft** occurs when an attacker pretends to be someone else to get users credentials to gain access to its assets [14].
- **Changes to Business Model** is a severe threat where consumers' data may reside over different territories governed by different federal laws. The service provider provides various services to consumers without the knowledge of where they reside and the consumer loses control over the infrastructure which is indeed the biggest threat and can alter the life of cloud consumers.
- **Lock-IN** condition refers to the lack of ability to shift from one cloud to another. This threat occurs when organizations jump into the cloud service provider with poor knowledge of which cloud model is more suitable to them according to their needs to avoid Lock-IN [14].

2.1.10. Security Attacks in Cloud Computing

This section points represent attacks in cloud computing

- **Structured Query Language (SQL) Injection Attacks -** In standard SQL code, the attacker inserts malicious code to access unauthorized database to gain sensitive data about the user. In this case, the website allows hacker's data to be accessed by SQL Server considering it as user's data which leads the attacker to gain knowledge about how the website is functioning and so the attacker make changes into that [14].
- **Cross Site Scripting (XSS) Attacks -** The attacker inserts malicious code into the user's web page to redirect him to the attacker's website to access sensitive data. It can be done in two ways by either using Stored XSS (permanently stores malicious code into a resource managed by the web application) or Reflected XSS (immediately reflects back malicious code to the user and hence do not store it permanently) [14].
- **Phishing Attacks-** The attacker makes use of cloud service in phishing attacks, where the attacker manipulates a web link to redirect the user to a false link and so by hijacking users account gains access to sensitive data. Phishing attacks can be eradicated by identifying spam emails or pop-up which can be done by using anti-spam tools [14].
- **Domain Name Server (DNS) Attacks-** In many scenarios the user access a server by calling its domain name and instead of the domain he requests for being routed to some other malicious code. It happens in the case of DNS attacks where the attacker makes use of DNS to

translate the domain name into an IP address to access user's confidential data [14].

- **Man in the Middle Attacks (MITM) -** When an attacker attempts to intrude in an ongoing conversation with the aim to inject false information to access sensitive information being shared then it is known as MITM attack [14].
- **Denial of Service (DOS) Attacks -** In this kind of attack, the attacker tries to make the services unavailable which are assigned to authorize users by launching SYN flooding, UDP flooding, and ICMP flooding attacks etc. on the server. An attacker attempts to break the network or disable services provided by the server by sending data packets continuously to the target server and without changing the nodes, data packets, or decrypting encrypted data. These data packets occupy the network bandwidth and consume the server's resources [14].
- **Distributed Denial of Service (DDOS) Attacks -** DDOS is an advanced type of DOS attack in terms of flooding the target server with a large number of packets from multiple networks which have already been compromised to disable the services provided by the target server and generating more traffic than DOS so that the targeted server is unable to handle the requests which make it different from DOS attack [14].
- **Wrapping Attacks -** When a user request for a VM through a web browser as a result of this request the web server generates SOAP message which contains XML based information that will be exchanged between the server and browser. Before communication between server and browser, such XML based information are signed using signature values. In wrapping attack, the attacker duplicates the

SOAP message during translation and send it to the server as an authorized user and will be able to intrude the cloud services by running malicious code [14].

- **Hypervisor Attacks** It may happen that a guest operating system runs a malicious code on to the host system and tries to bring it down by taking full control of the system and blocks its access to other guest's operating systems. If the hacker succeeds he can make changes to any guest operating systems and get control over the data passing through the hypervisor [14].

2.2. Related Works

The following sections describe authentication methods and highlighted researches related to one-time password and access control models in cloud computing.

2.2.1. Authentication methods

The following points represent the mothods used for authentication

- **static Password Authentication**

The most widely used and oldest form of authentication is password. Users provide an id, a typed in word or name, along with a password. In the majority of the systems, the passwords are encrypted instead of storing it as a plain text [15]. Static authentication suffers from many drawbacks as it is considered as a low-security solution when used on its own or without precaution [16]:

- Password can be stolen online by an eavesdropper and used in order to gain access to the system (replay attack) [16].
- Physical attacks can easily be done, by a camera recording a PIN sequence as it is being typed, or by a keylogger (either software or hardware) to record a password on a computer [16].
- Passwords are also vulnerable to guessing attacks, like brute force attacks or dictionary attacks. However, it is possible to limit the scope of these attacks by adding a delay of about one second before entering the password and after entering a wrong password [16].
- A stolen password hash could sometimes be reverted, for example, by using a list of precomputed hashes of the most common passwords or more sophisticated attacks like using rainbows-tables [16].

- **One Time Password**

To overcome the drawback of password reuse, one-time passwords were developed. A one-time password (OTP) is valid for only a single transaction on a computer system or any other device such as a smartphone. OTP's are generated using random values and hash functions. Types of one-time passwords are a challenge-response password and a password list. The challenge-response password replies with a challenge value (e.g. a random number chosen by the authentication server) after getting a user identifier. The response is calculated using the response value or from a table based on that particular challenge. A one-time password list makes use of previous passwords, which are sequentially used by the user wanting to access a system. The values are generated such that it is very hard to predict the

next value from the previously generated values. The time synchronization is also one of the approaches for generating OTP. In this approach, a security token is used. The clock in the token and authentication server is synchronized as the generation of the password depends upon current time [15].

- **Public Key Cryptography**

  Public key cryptography, which is asymmetric cryptography, is a class of cryptographic protocols. The two keys are mathematically linked. The private key is kept as a secret and is used to decrypt and a public key is used to encrypt messages between the clients. Encryption and verification of signature both are completed using the public key. Advantage of public-key cryptography is that the public key is easily available to the public. They are often published on the Internet so that they can be easily retrieved. It is used to transfer a symmetrical encryption key by which the message is encrypted because of the computational complexity. It is based on simple algorithms and is much faster. The user keeps a private key, while the corresponding public key is made available in a certificate digitally signed by a respective certification authority. This certificate is made available to users [15].

2.2.2. OTP as Two-Factor Authentication

In [5] OTP generation algorithm used AES encryption algorithm and interpolation derivates. The derivates of interpolation is very complex in terms of guessing and predictability. The interpolation derivates derived the input message in terms of username and password, create a variable size matrix for the processing of input data of the hash algorithm, and randomly

select the 6-bit data for OTP transmission. JAVA development software used to implement a one-time password generation.

The paperwork in [17] is implemented the OTP and Role Based Access Control to provide different privileges to different users. Authors classified each user on the basis of three roles. Admin, manager, customer and assign different services as per their roles and responsibilities. The integrity of the file, which user is uploading on the server is calculated and maintained using the MD5 algorithm. The confidentiality of the message is maintained by using the RSA algorithm.

In the first stage of [18], the basic authentication mechanism is used to verify the user through "login-password" mechanism. The second stage of authentication would be triggered when the user performs any transaction (upload/ download/ view critical information) on the system. Then, a One Time Password (OTP) authentication would be a trigger and it would request the user to authenticate to proceed with the transaction. The system also employs one of the most successful encryption techniques of AES encryption for securing the Personal Health Record system, which is stored in the Third party semi-trusted cloud server. The fine-grained access control facility was also preserved and by shifting the security-related operation to the trusted Host server.

This paper [19] aims to ensure the protection of organizations' data from both the cloud provider and the third-party auditor and from some users who take advantage of the old accounts to access the data stored on the cloud. A third party auditor (TPA) is used to verify the data stored in the cloud and be sure that it does not tamper. The proposed system increases the authentication level of security by using two authentication techniques; time-based one-time password (TOTP) and automatic blocker protocol (ABP)

fully protect the system from unauthorized third party auditor. In the proposed system, the data owner controls all the privileges to be sure that who can access the outsourced data on cloud storage servers. To increase security, user authentication is verified by two-factor authentication: the first is exercised with a username and password while the second is caused by the implementation of TOTP.

This paper [20] proposes a two-factor new strong authentication scheme for cloud computing by using USB token with a combination of different techniques such as hash function and Diffie-Hellman key agreement. It consisting of three phases: Registration, login, authentication phase, and two activities: a change of password and USB token backup. In comparison with the existing methods, the present method is proved more and more effective in cloud computing because it achieves both functionality and security requirements. The functionality requirement it provides, involves mutual authentication, no verification table, user privacy, session key exchange, and choosing and updating the password freely. Moreover, it is very resistant to many familiar attacks: replay attack, password guessing, DOS, man-in-the-middle, insider, and USB Token loss attack.

This paper [21] proposed a two-factor access control protocol for web-based cloud computing services, using a lightweight security device. The device can compute some lightweight algorithms, e.g. hashing and exponentiation; and it is assumed that no one can break into it to get the secret information stored inside. With this device, the user secret key (which is usually stored inside the computer) is required. In addition, the security device should be also connected to the computer (e.g. through USB) in order to authenticate the user for accessing the cloud. The user can be granted access only if he has both items. Furthermore, the user cannot use his secret key with another device belonging to others for the access.

2.2.3. Access Control Models in Cloud Computing

The following points briefly define access control models in cloud computing:

- **Discretionary access control model**

  In Discretionary Access Control (DAS) model, the owner of the objects determine the permission rights on the objects or data that needs to be accessed based on the membership in a particular group or users identities [2].

- **Mandatory access control model**

  In Mandatory Access Control (MAC) model, only the administrator can determine and manage the access controls. He can decide and define the access policy which cannot be altered by any other users. In the MAC model, the administrator assigns different security labels to the subject and object. These security labels help to protect the flow of information from the higher security level to the lowest [2].

- **Role-based Access Control (RBAC)**

  In RBAC access decisions are based on the individual's roles and responsibilities within the cloud environment. It identifies the user role and based on this it manages the access of a user. The role is a set of objects or policies related to the subject. The role may vary from user to user. RBAC provided web-based application security. It allows users to execute multiple roles at the same time. RBAC decides what permission should be assigned to which user. A role manager responsibility is to assign a role to the user, and if the user is going out, then revoke a role from the user. Cloud Provider, users,

and others are not able to see the data if they are not assigned with proper roles. Data owner can revoke the role if they found as unauthorized user [22, 23].

- **Attribute-based access control**

In attribute-based access control, the attributes are considered based on the user's request and the type of access user wish to access and the needed resources of the user [22].

- **Attribute-based encryption**

In an attribute-based encryption scheme, a set of attributes are treated as a user identity and it's used for encryption and decryption techniques. Trusted agent generates keys for data owner and user. It generates key according to the attributes of the user. The trusted agent will generate public key and master keys for the user. Data owner role is to encrypt the data with user public key and user will decrypt the data with own private key [23].

- **Key-Policy based Encryption**

In key-policy attribute-based encryption, Ciphertext is associated with a set of attributes, Private Key which is issued by trusted authority is associated with access structure like a tree, which describes this user's identity. The user can recover the file if and only if access policy in the private key is satisfied with the attributes in the ciphertext [23].

- **Ciphertext policy based attributes based Encryption**

In CP-ABE, the private key is associated with a set of attributes, and a ciphertext are created with an access structure, which is used to specify the encryption policy. A user can decrypt the ciphertext if

and only if the attributes in the private key is satisfied the access tree specified in the ciphertext. CP-ABE is used to encrypt the data which can be kept confidential even if the storage server is untrusted [22, 23].

In this paper [24] the proposed solution is decentralized access control while several key distribution centers KDCs for key management were used. Anonymous authentication means cloud authenticate the credentials of the user and does not know the identity of the creator of data. This prevents replay attacks. The access control and authentication guarantee no two users can authenticate themselves if they are individually not authorized. The files are associated with file access policies, that used to access the files placed on the cloud.

The identity-based access control for digital content (iDAC) is proposed in this paper [25]. IDAC is based on ciphertext-policy attribute-based encryption (CP-ABE) which is identity-based encryption. In iDAC, the access structure is embedded into the encrypted digital content. Only users with the identity-based keys, which satisfy the access structure, could decrypt the digital content. There are three roles in iDAC, authority, content provider, and content user. Authority is responsible for key management. The authority will first set up security parameters. The content user could submit his/her security parameters to the authority for acquiring a secret key. The content provider is one who attempts to distribute digital content. The content provider will encrypt digital content using CP-ABE. The content user is one who attempts to access digital content. The content user could access the original digital content if and only if he/she could decrypt the protected content using CP-ABE with the secret key.

The scheme [26] allows users to delegate their access permissions by assigning user self-defined attributes to their delegatees and specifying access control rules in terms of the attributes. Access control enforcement and most of the tasks caused by changing access control rules and credentials are carried out by cloud providers. Users' tasks are limited to generating a credential key and policy keys. Each authorized user defines her own set of attributes and assigns the attributes to the users that she wants to delegate access permission. An authorized user issues credential certificate to her delegatees. A credential certificate states the attributes that an authorized user assigned to her delegatee. A user can acquire attributes from multiple authorized users. As a result, the user will be issued multiple credential certificates. An authorized user stores her delegatees' credential certificates on cloud providers. The credentials of a user are converted to an encryption key called credential key. If and only if the credentials satisfy the access control rules, the credential key and one of the rule keys form a matching key pair. That is, the information encrypted by the credential key can be decrypted by the rule key.

CHAPTER THREE

SYSTEM DESIGN

Chapter Three

System Design

This chapter contains a detailed explanation of one-time password, role-based access control model and the developed scenario. Also, a brief definition of the tools used to implement the developed scenario

3.1. One Time Password

The one-time password provides authentication for system access (login) and other applications requiring authentication that is secure against passive attacks based on replaying captured reusable passwords. The OTP system protects against external passive attacks against the authentication subsystem. It does not prevent a network eavesdropper from gaining access to private information and does not provide protection against either "social engineering" or active attacks [27]. One-Time Password is certainly one of the simplest and most popular forms of two-factor authentication for securing network access. For example, in large enterprises, Virtual Private Network access often requires the use of One-Time Password tokens for remote user authentication [28].

3.1.1. Hmac-based one time password

An algorithm to generate one-time password values, based on Hashed Message Authentication Code (**HMAC**). As the output of the HMAC-SHA-1 calculation is 160 bits, truncation of this value to something that can be easily entered by a user.

$$\mathbf{HOTP\ (K, C) = Truncate\ (HMAC\text{-}SHA\text{-}1(K, C))} \qquad \mathbf{(3.1)}$$

Where: - Truncate represents the function that converts an HMAC-SHA-1 value into a HOTP value. The Key (K), the Counter (C), and Data values are hashed high-order byte first [28].

3.1.2. Time-based one time password

TOTP is the time-based variant of the HOTP algorithm, where a value T, derived from a time reference and a time step, replaces the counter C in the HOTP computation. TOTP implementations may use HMAC-SHA-256 or HMAC-SHA-512 functions.

- **Algorithm Requirements**

The following points indicate the requirements taken into account for designing the TOTP algorithm

- The prover (e.g., token, soft token) and verifier (authentication or validation server) must know or be able to derive the current UNIX time (i.e., the number of seconds elapsed since midnight UTC of January 1, 1970) for OTP generation.
- The prover and verifier must either share the same secret or the knowledge of a secret transformation to generate a shared secret.
- The algorithm must use HOTP as a key building block.
- The prover and verifier must use the same time-step value X.
- There must be a unique secret (key) for each prover.
- The keys should be randomly generated or derived using key derivation algorithms.
- The keys may be stored in a tamper-resistant device and should be protected against unauthorized access and usage.

$$\mathbf{TOTP = HOTP\ (K, T),} \qquad \mathbf{(3.2)}$$

Where T is an integer and represents the number of time steps between the initial counter time T0 and the current UNIX time.

$$T = (\text{Current Unix time} - T0) / X, \quad (3.3)$$

Where the default floor function is used in the computation [29].

3.1.3. Blowfish BCRYPT

Blowfish bcrypt will be used for the hashing process. It will be used to hash the one time password.

Blowfish is an encryption algorithm which uses a variable key size of 32bits to 448 bits. Blowfish was designed by Bruce Schneier in 1993. it uses a different key size up to the length of 448 bits. Blowfish is a symmetric algorithm, which means it uses the same key for both the encryption process and the decryption process. Blowfish algorithm takes an input of 64 bits for encryption using a key of any length from 32 bits to 448 bits. The total number of rounds is 16 rounds.

BCRYPT makes heavy use of the Blowfish encryption function. This is a standard 16-round Feistel network, which uses SBoxes and subkeys K determined by the current state. Its block size is 64-bit and during every round, an f-function is evaluated: it uses the 32-bit input as four 8- bit addresses for the SBoxes and computes (S0(a)+S1(b) xor S2(c) + S3(d)). EksBlowfishSetup is a modified version of the Blowfish key schedule. It computes a state, which consists of 18 32-bit subkeys and four SBoxes – each 256 X 32 bits in size – which are later used in the encryption process. The state is initially filled with the digits of pi before an ExpandKey step is performed: After adding the input key to the subkeys, this step successively uses the current state to encrypt blocks of its salt parameter and updates it with the resulting ciphertext. In this process, ExpandKey computes 521

Blowfish encryptions. An important detail is that the input key is only used during the very first part of the ExpandKey steps. BCRYPT finally uses EncryptECB, which is effectively Blowfish encryption.

There are two phases in which the bcrypt algorithm is being executed. In the very first phase, the Eksblowfish Setup is called with the salt, password, and cost to process the Eksblowfish state. However, the expensive key schedule consumes lots of time. On the basis of 192-bit value of OrpheanBeholderScryDoubt is encrypted at 64 times from the previous phase to the particular state using Eksblowfish in ECB mode. The 128-bit salt would be concatenated with the final result of the encryption loop to provide the output.

Consider a 64 bits input ‘x’. This input bit is divided into two parts: xL and xR, each 32 bits. Then for each round, xL = xLXOR Ki(I is the number of round), F(xL) = (S0(a)+S1(b) xor S2(c) + S3(d)), xR = F(xL) XOR xR. Then swapping xL and xR. After the16th round, undo the last swap. After that xR = xR XOR K17 and xL = xL XOR K18. Finally, combine the 32 bits xL and xR to get the 64 bits output (cipher-text). The cost and 128-bit salt would be concatenated with the final result of the encryption loop to provide the output [30-32].

3.2. Role-Based Access Control

In Role-Based Access Control (RBAC), access decisions are based on an individual's roles and responsibilities within the organization or user base. The process of defining roles is usually based on analyzing the fundamental goals and structure of an organization and is usually linked to the security policy. An RBAC access control framework should provide administrators

with the ability to determine who can perform what actions, when, from where, in what order, and in some cases under what relational circumstances. The following aspects exhibit RBAC attributes to an access control model.

- Roles are assigned based on an organizational structure with emphasis on the organizational security policy
- Roles are assigned by the administrator based on relative relationships within the organization or user base.
- Each role is designated a profile that includes all authorized commands, transactions, and allowable information access.
- Roles are granted permissions based on the principle of least privilege.
- Roles are activated statically and dynamically as appropriate to certain relational triggers.
- Roles can be only be transferred or delegated using strict sign-offs and procedures.
- Roles are managed centrally by a security administrator [33].

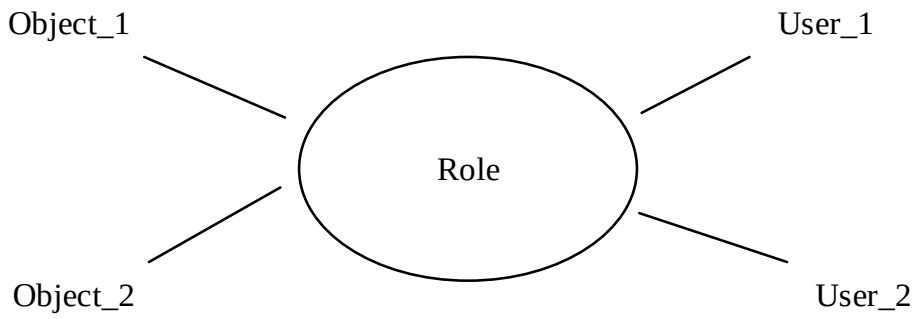

Figure 3.1. Role Relationship

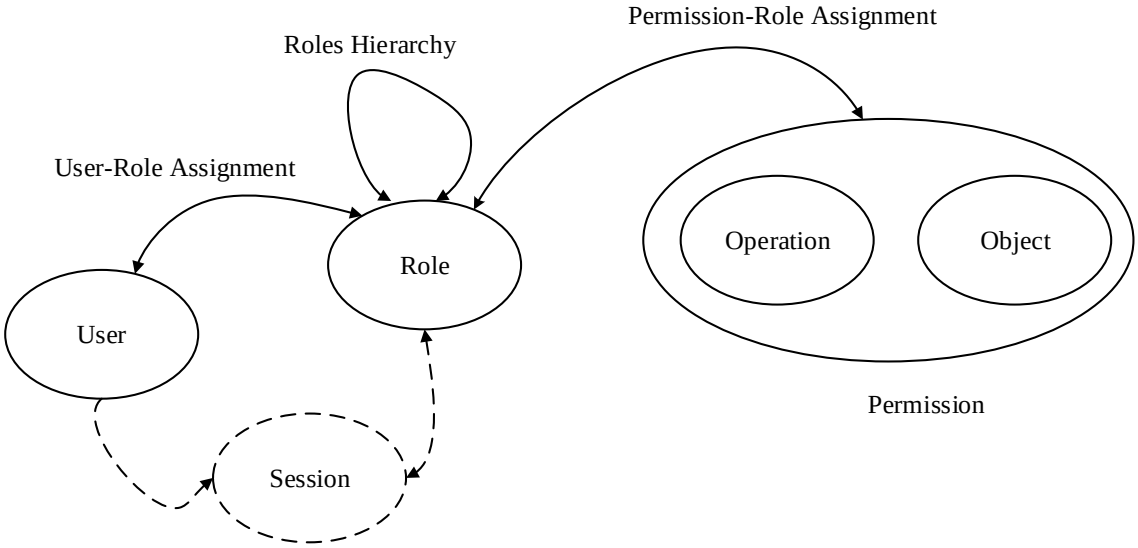

Figure 3.2. Role-based Access Control Model

RBAC was developed to overcome administration difficulties encountered in large commercial organizations. As the major part of access control decisions is based on the subjects' function or job, introducing roles greatly simplifies the management of the system. Since roles in an organization are relatively consistent with respect to user turn over and task reassignment, RBAC provides a powerful mechanism for reducing the complexity, cost, and potential for error in assigning permissions to users within the organization. An important feature of the RBAC model is that roles are hierarchical; roles inherit permissions from their parents. Thus, roles are not flat collections of groups of permissions. Hierarchy aims at increasing system administrator productivity by simplifying distribution, review, and revocation of permissions [34].

3.3. Developed Scenario

The idea behind the developed system is an online education contains short courses and diplomas in information technology. The scenario includes

sub-scenarios for different user privileges, including system administrator, instructors, and students. The system grants each of the users the ability to run the system with a privilege that varies between them such as add/delete/ update for the administrator's access, and for the instructors to add and view the content that the student can view only. Moreover, the system is secured through one-time password mechanism for the users in order to identify the person by his username and password and once again by a one time generated password to be delivered into the inbox of the Email of the student.

After the system checked the validity of the username and password, automatically generate a one-time password and send it to account's email. The system record the time of sending OTP. This value noted as T1. The user checks the email for the password and then type it into the system. The system record the time of entering OTP and noted it as T2. The system compares T1 and T2 to check that if the user enters the password before or after 30 seconds from T1. If the time of entering OTP is after 30 seconds from sending it to the email then this password is not valid to allow user login to the system otherwise, the user can log in to his/her profile.

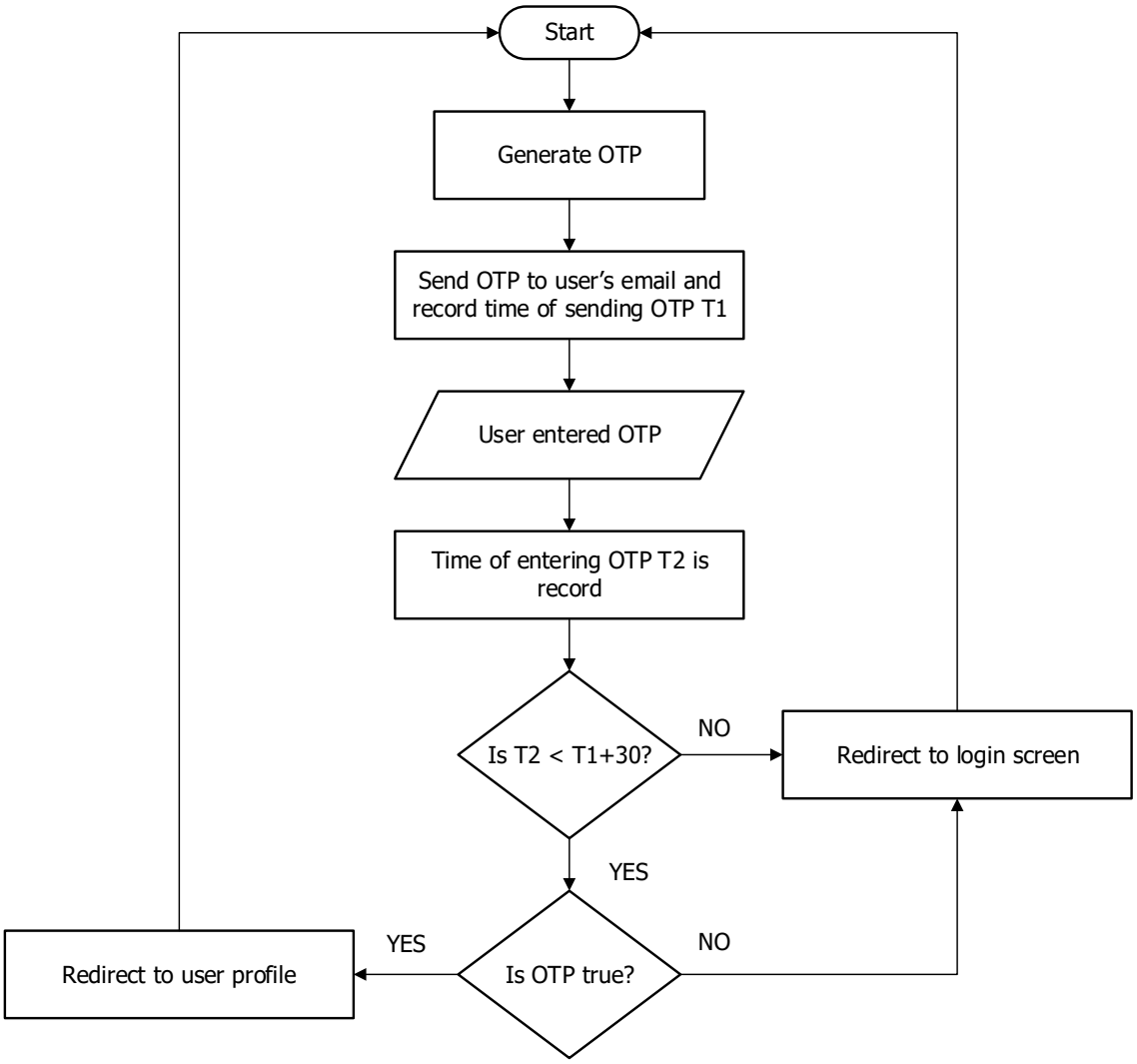

Figure 3.3. One Time Password Check

3.3.1. The First Scenario

A user is a student enrolled in one of the courses offered in the system.

- Registering a new user:

When the student attends to register into the system, a registration form appears when clicking on the signup button on the login form. The form asks for full name, age, gender, and email …etc. and a User Name/Password, the system registers the student into, every new student

will remain unable to access the system until the administrator activates their status.

- Administrator approve

After the approval of the administrator to student request of activation, the system allows students to log in to their sessions.

- Log in to the system

In order to log into the system, a username/password is required, the system checks the validity of the username and password, generate a one-time password and then send it to the student email.

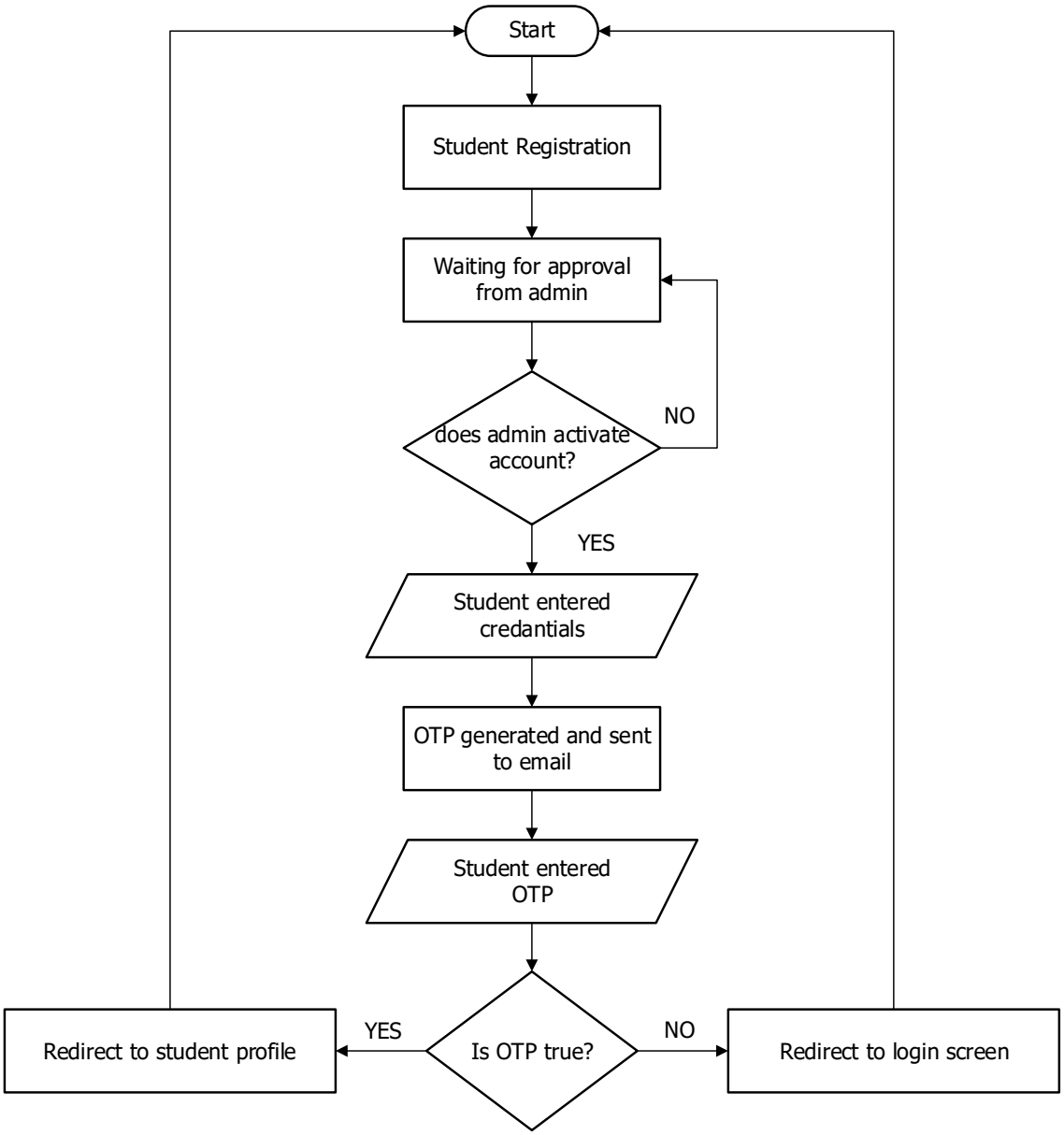

Figure 3.4. Student Registration and Login Scenario

3.3.2. The Second Scenario

An instructor has full access rights on only the courses that he/she responsible for. The instructor account is created by the administrator.

- Instructor login

Username and password are required for login with different privileges in order to upload lectures, assignments.

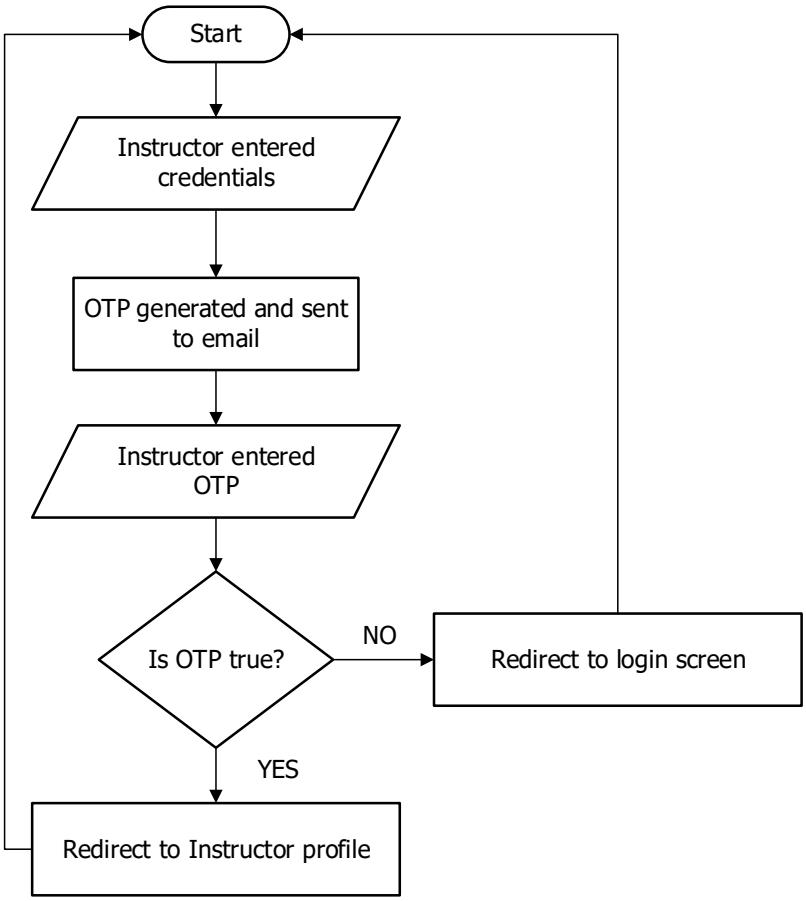

Figure 3.5. Instructor Login

3.3.3. The Third Scenario

An Administrator has full access rights on the system. He / She manage student and instructor accounts.

- Add instructor

The administrator creates an instructor account in the database and assigns permissions to the account that are uploading documents, lectures notes, assignments.

- Handling student account

The student's accounts once created will remain in a suspended state until the administrator activate the accounts. The administrator has the rights to activate, suspend and even delete a student account. The

activation and suspension of account depending on the status of the account. If the status equals 1 that means the account is activated otherwise it is suspended. The deletion of an account means to delete the record of an account from the database.

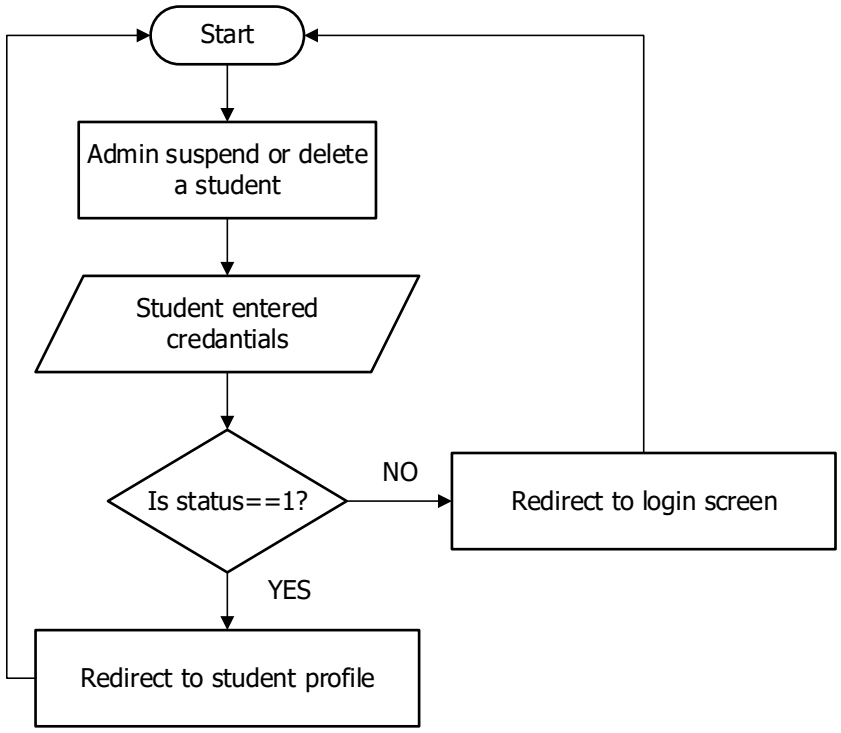

Figure 3.6. Suspend or delete a student account

3.4. Implementation

The proposed system composed of a web server, mail server, and database server. Web server contains the proposed online education system. Mail server contains the mail accounts created to users in order to send one time password to it. Database server contains both databases of the web server and mail server. Figure 3.7 represents the three servers which composed the proposed system and the relationship between them.

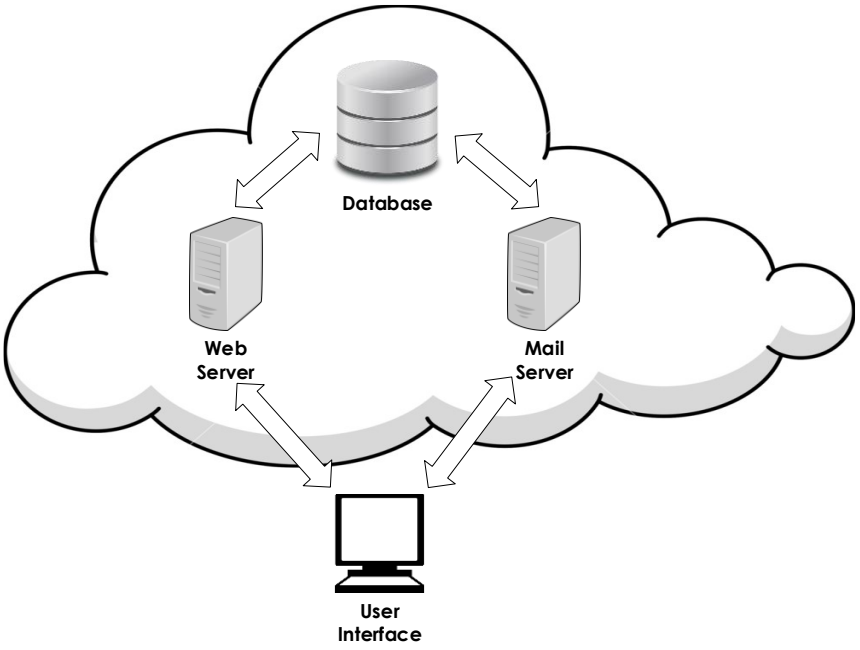

Figure 3.7. Block diagram of proposed system structure

The developed scenario with all sub-scenarios will be developed using PHP language, MySQL database and web development platform on windows. PHP language will be used to develop the overall system with role-based access control for three roles: Administrator, Instructor, and Student. The PHP script will be used to generate a time-based one-time password.

3.4.1. WampServer

WampServer is a Web development platform on Windows that allows creating dynamic Web applications with Apache2, PHP, MySQL, and MariaDB. WampServer is available free in both 32 and 64-bit versions. WampServer is not compatible with Windows XP, SP3, or Windows Server 2003.

- **Apache2**

Apache is an Open-source HTTP server for modern operating systems including UNIX and Windows. Apache is a secure, efficient and extensible

server that provides HTTP services in sync with the current HTTP standards. The Apache HTTP Server ("httpd") was launched in 1995 and it has been the most popular web server on the Internet since April 1996 [35].

- **PHP**

PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML[36]. PHP is mainly focused on server-side scripting, so you can do anything any other CGI program can do, such as collect form data, generate dynamic page content, or send and receive cookies. But PHP can do much more. PHP can be used on all major operating systems, including Linux, many UNIX variants (including HP-UX, Solaris, and OpenBSD), Microsoft Windows, macOS, RISC OS, and probably others. PHP also has support for most of the web servers today. This includes Apache, IIS, and many others. One of the strongest and most significant features in PHP is its support for a wide range of databases. Writing a database-enabled web page is incredibly simple using one of the database-specific extensions [37].

- **MySQL**

MySQL is the world’s most popular open source relational database management system database, enabling the cost-effective delivery of reliable, high-performance and scalable Web-based and embedded database applications. MySQL delivers the ease of use, scalability, and high performance, as well as a full suite of database drivers and visual tools to help developers and DBAs build and manage their business-critical MySQL applications. MySQL is developed, distributed, and supported by Oracle [38].

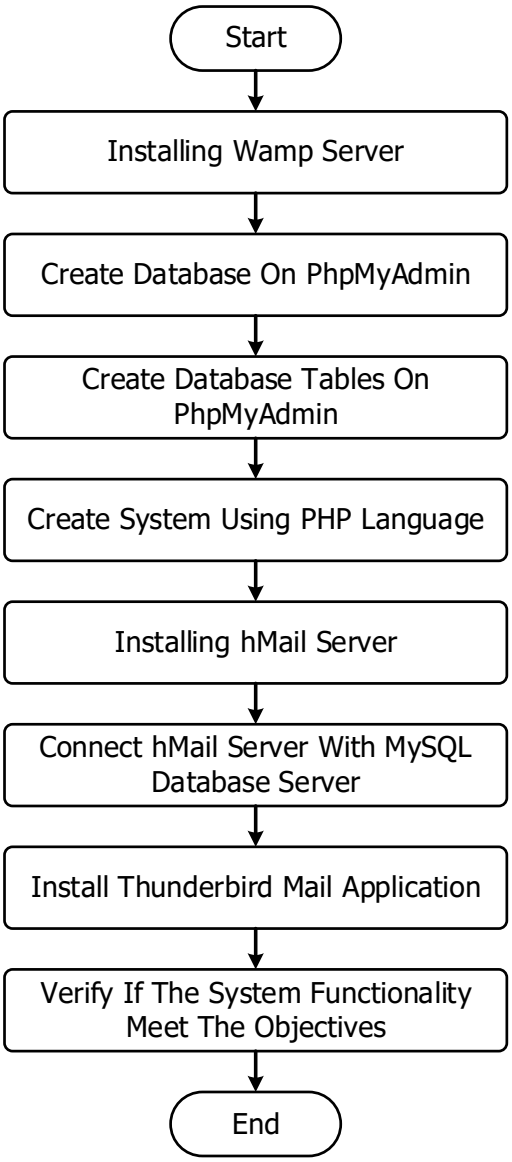

Figure 3.8. Proposed system implementation process flow

3.5. Testing

For testing the one-time password functionality, hmailserver and Thunderbird were used and integrated with wamp server. hMailServer is a free, open source, e-mail server for Microsoft Windows. It supports the common e-mail protocols (IMAP, SMTP, and POP3) and can easily be

integrated with many existing web-mail systems [39]. Thunderbird is a free cross-platform email application that's easy to set up and customize [40].

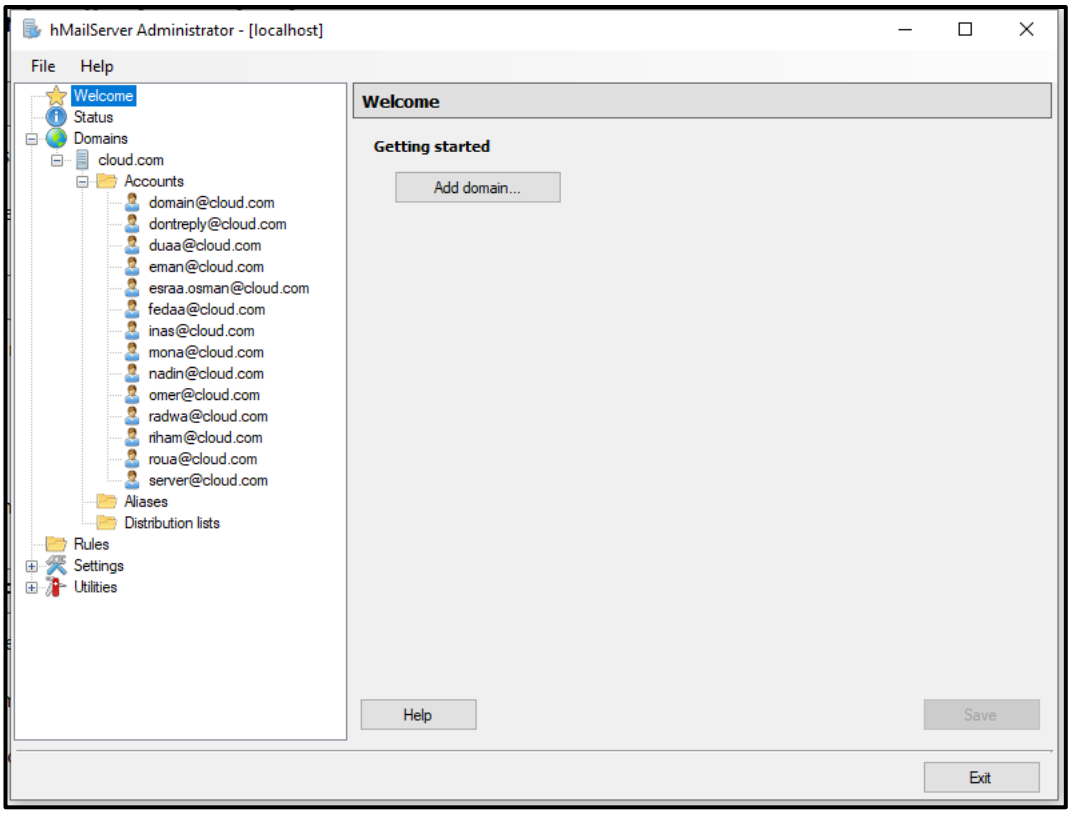

Figure 3.7. hMailServer interface

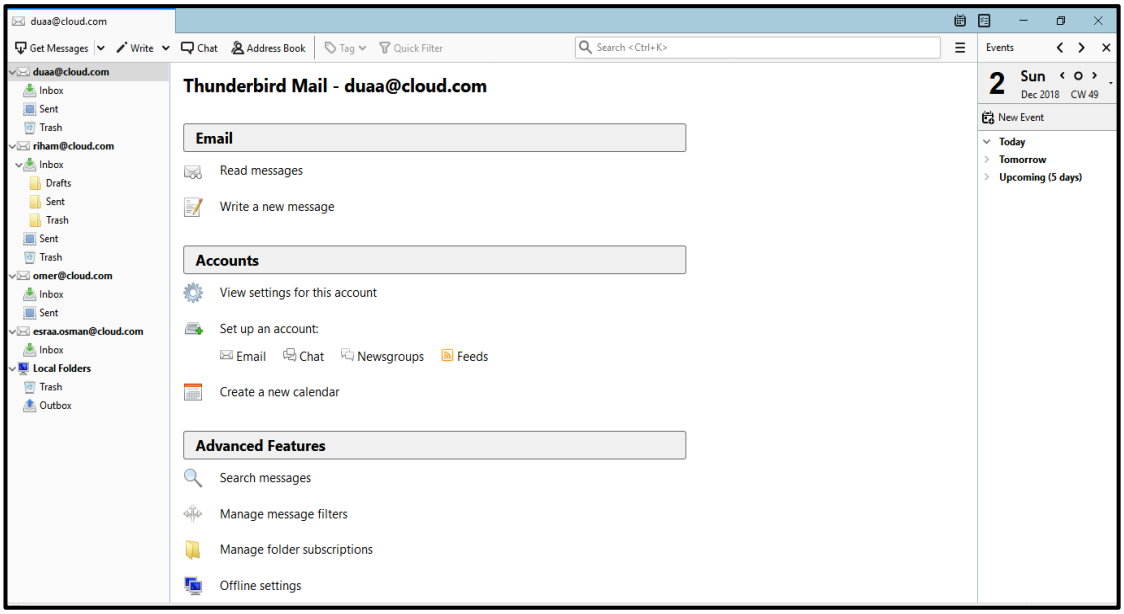

Figure 3.8. Thunderbird interface

CHAPTER FOUR

RESULTS AND DISCUSSION

Chapter Four

Result and Discussion

In this chapter, the developed system will be tested and all the results will be captured and discussed.

4.1. Results

The developed system is an online educational system hosted in the cloud. The system has three users, the administrator, the Instructor, the student. Each user has an account associated with permissions according to his role. The administrator has full access rights on the system. He / She manage student and instructor accounts. The student can only view (download) the data stored on the cloud and cannot upload files to protect the system from uploading files contain malicious scripts. The instructor which is a trusted user has the permission of storing data on the cloud. The following sections show in details the results which were obtained when testing the system.

4.1.1. Student Profile

Firstly the student should register into the system then can log in using his/her credentials in order to view (download) the lectures notes and assignments. The following sections illustrated the results which were obtained when testing the system for the student role. Firstly the student should register into the system then can log in using his/her credentials.

- **Student Registration**

Figure 4.1 shows the interface of the system. And figure 4.2 shows the student login page.

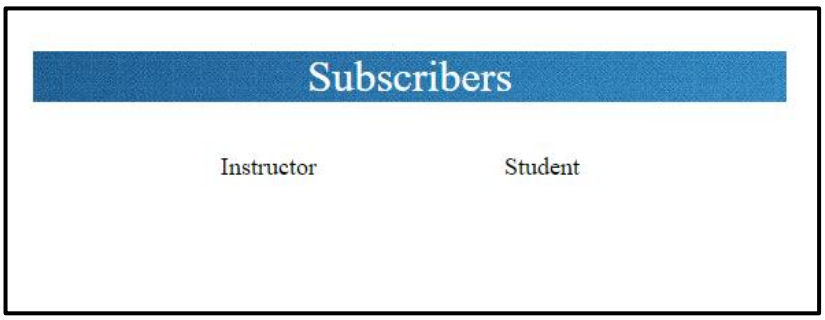

Figure 4.1. The interface of the designed system

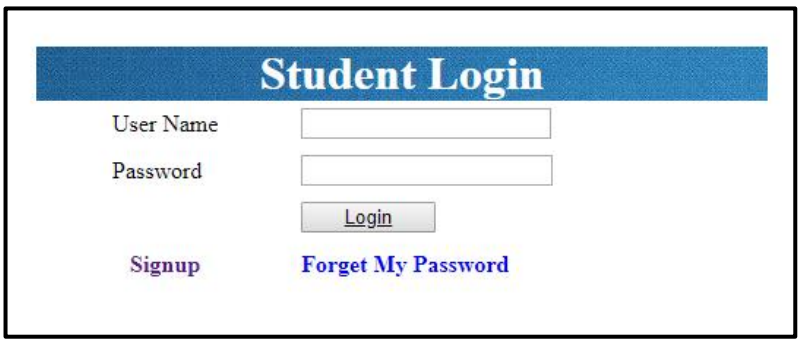

Figure 4.2. Student login page of the designed system

For the first time, the student must register on the system. To register the student click on signup.

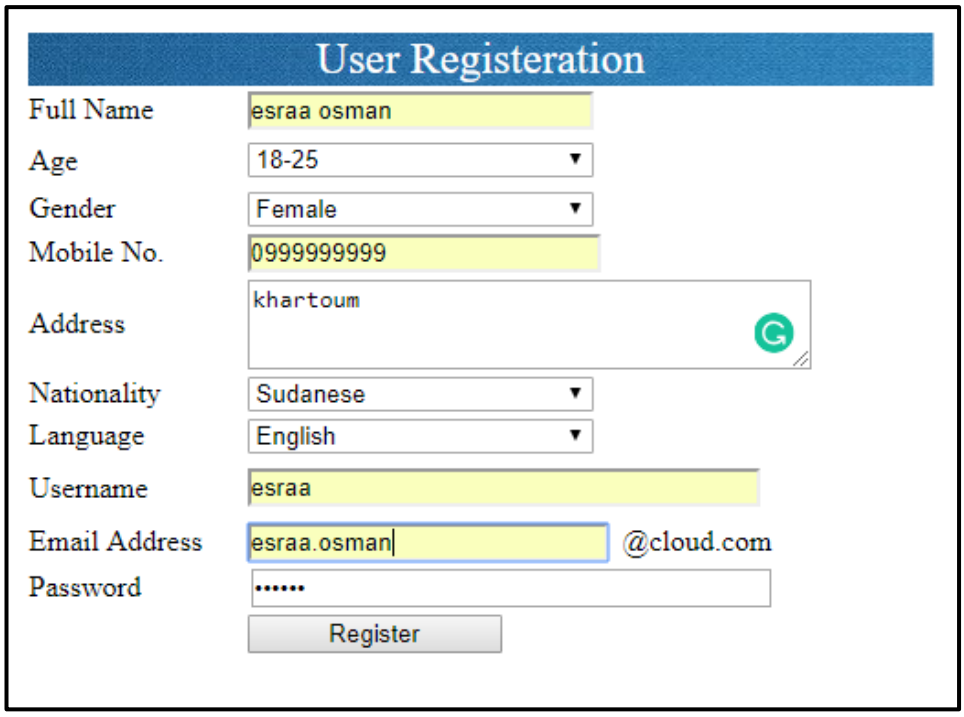

Figure 4.3. Registration page of the designed system

A new record was created in the database containing user information.

- **Admin approval**

The student cannot log in immediately to the system after registration unless of admin approve. Figure 4.4 illustrates a student attempt to login to the system before admin activates his/her account.

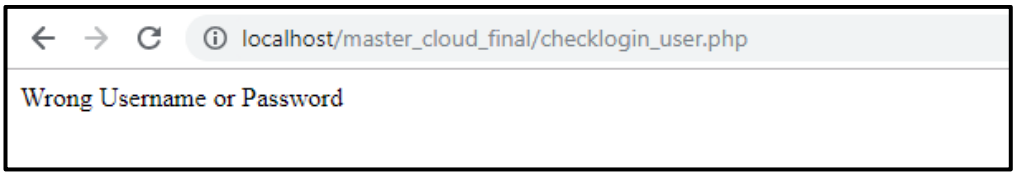

Figure 4.4. The system rejects credentials before admin approval

Figure 4.5 shows the admin page for managing the user in the system and illustrate that the user is waiting for activation. Figure 4.6 shows that the student account was activated.

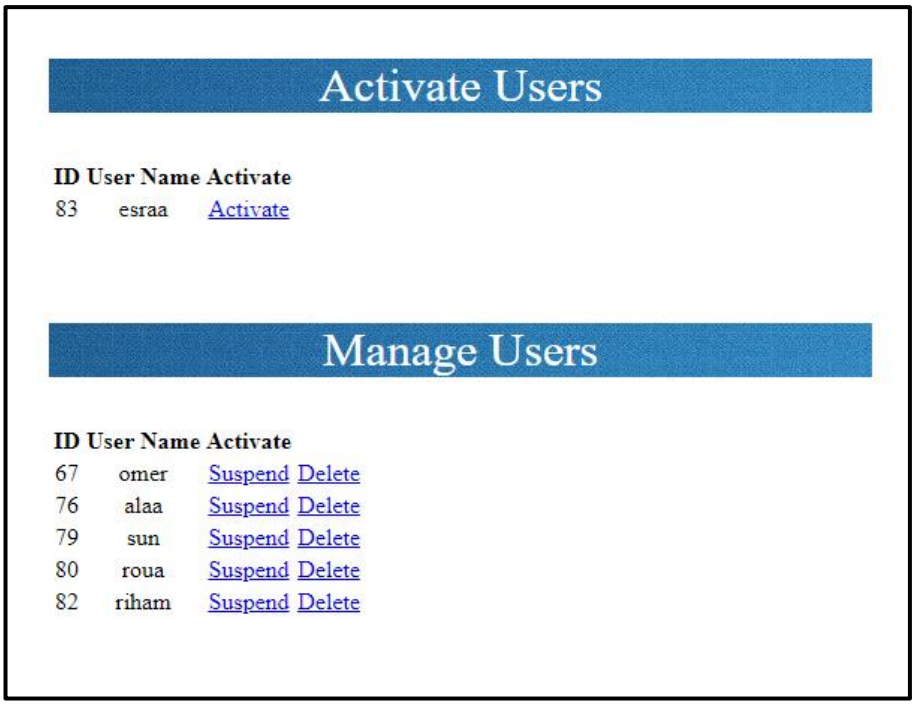

Figure 4.5. Student account waiting for activation

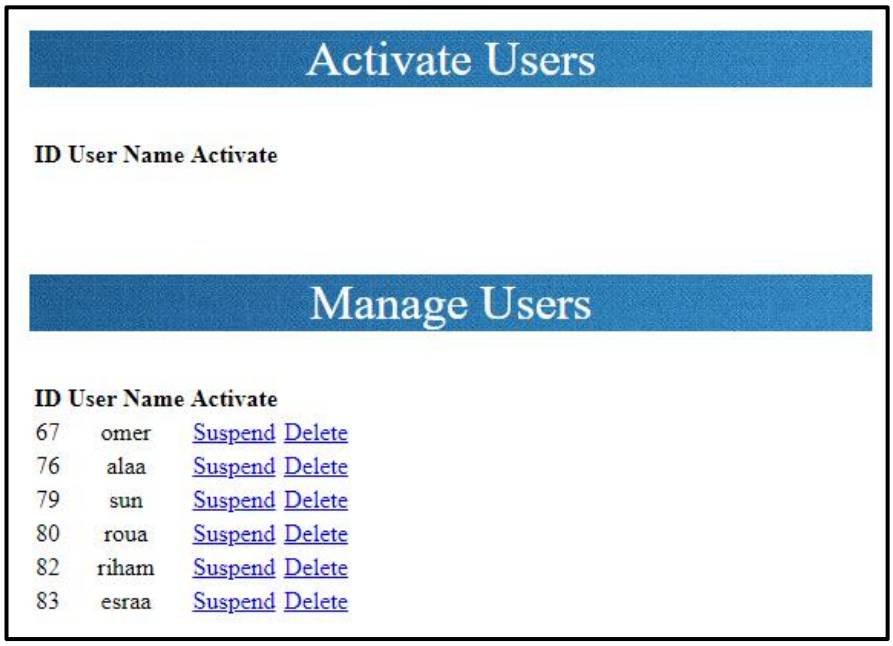

Figure 4.6. Student account was activated

Figures 4.5 and 4.6 illustrated that the admin can delete and even suspend student account.

➢ **Student login and one-time password generation**

After admin activated student account, the student can log in to the system. Once the student logs in to the system successfully then the system generates a one-time password, send it to the student's email and redirect him/her to enter the one-time password.

Figure 4.7 shows that a message was received in the student email contains the password. Figure 4.8 shows the page to enter OTP. The student must quickly enter the password before the completion of the 30 seconds time stamp. The system redirects the student to his/her profile. Figure 4.9 shows the student profile.

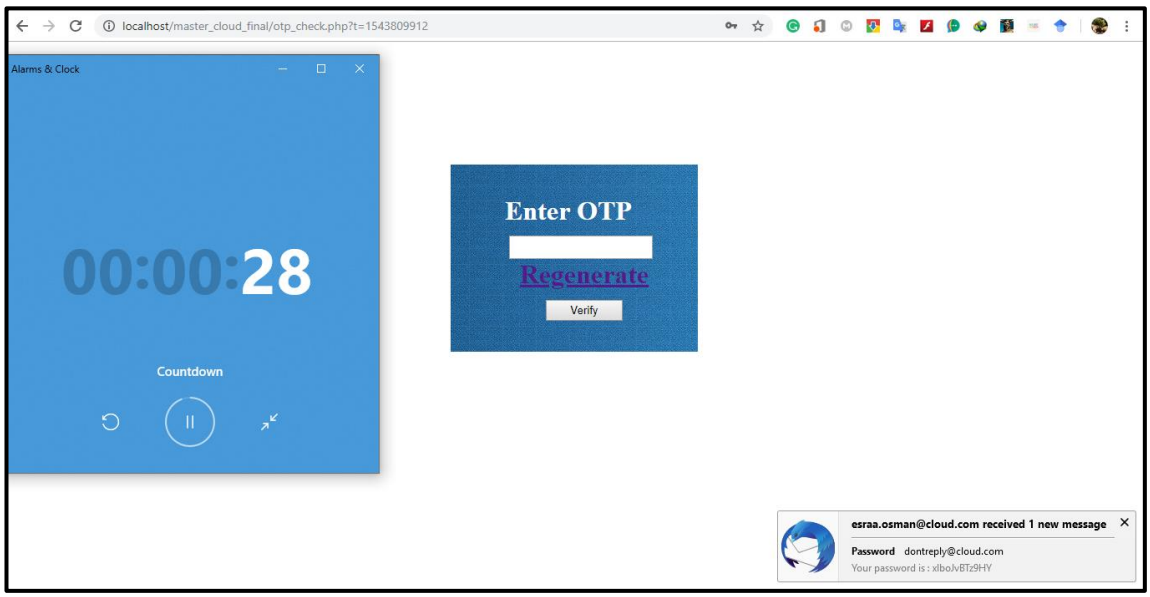

Figure 4.7. The password was generated and send via student email

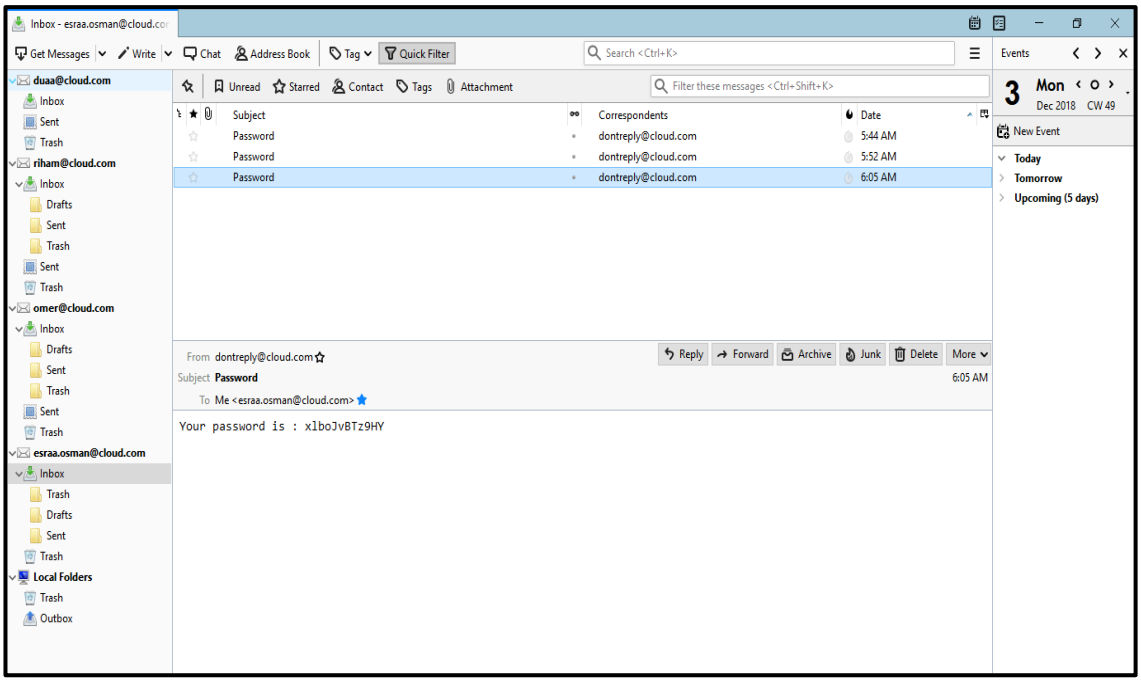

Figure 4.8. An email received contains password

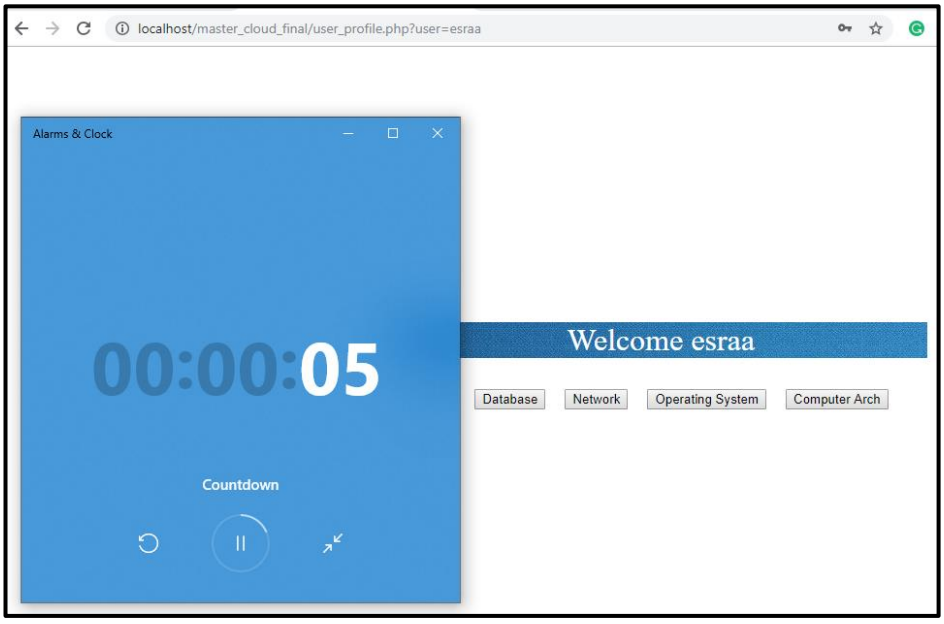

Figure 4.9. User profile of the designed system after successful login

If the student attempt to login after 30 seconds the system will show a message of regenerate the password again which illustrated in figure 4.10.

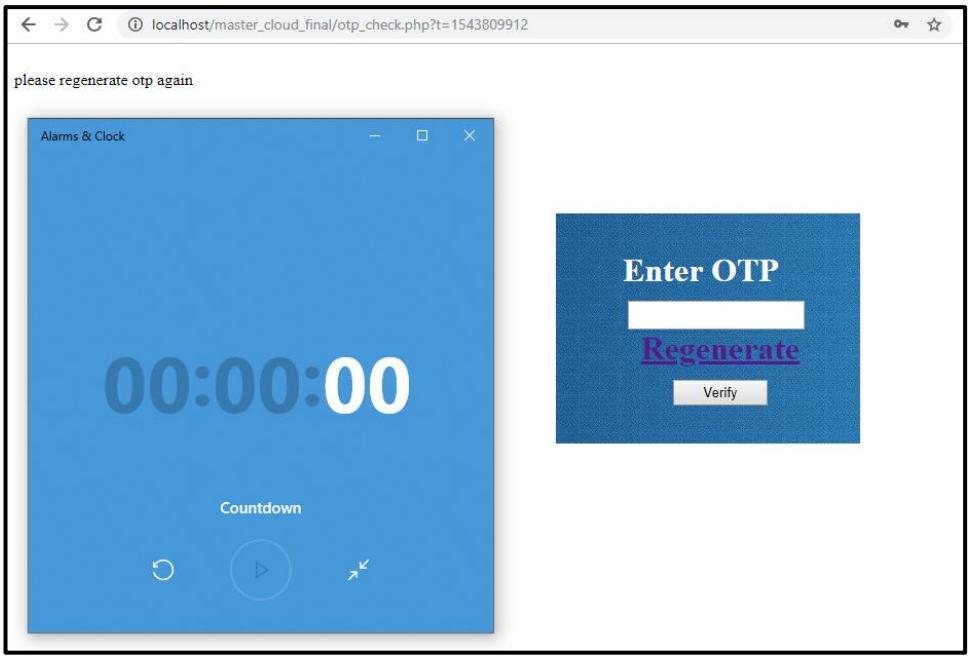

Figure 4.10. One time password entered after 30 seconds

- **Courses page**

After the successful login of the student accounts, the system redirects the student to his/her profile to choose a course. The student clicks on a course and will be redirected to the course page. Figure 4.11 shows the user profile and figure 4.12 represent the course page. The student can download a lecture or assignment.

Figure 4.11. The user profile of the designed system

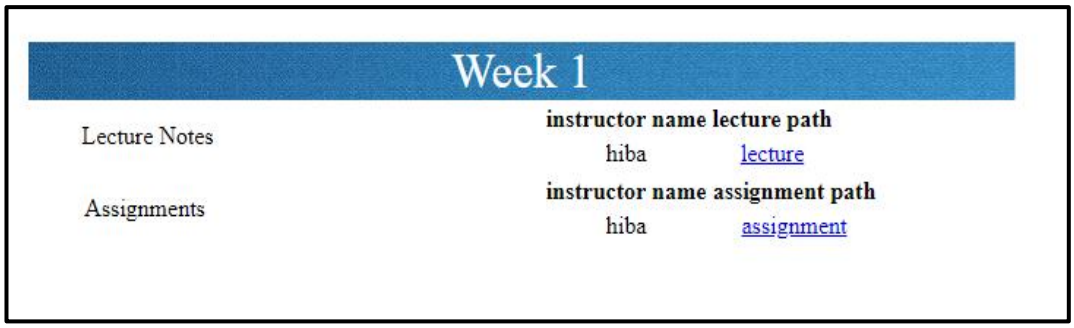

Figure 4.12. The course page of the designed system

4.1.2. Instructor profile

The instructor logins to the system with an account created by the administrator and the system redirected him/her to his/her profile. The following sections illustrated the results which were obtained when testing the system for the instructor role.

- **Instructor login**

Figure 4.13 shows the login page for the instructor.

Figure 4.13. Instructor login page of the designed system

The instructor login to the system by entering his/her credentials stored in the database. Then the system generates a one-time password and sends it to instructor email. The instructor checks his/her email to enter the one-time password which illustrated in figure 4.14.

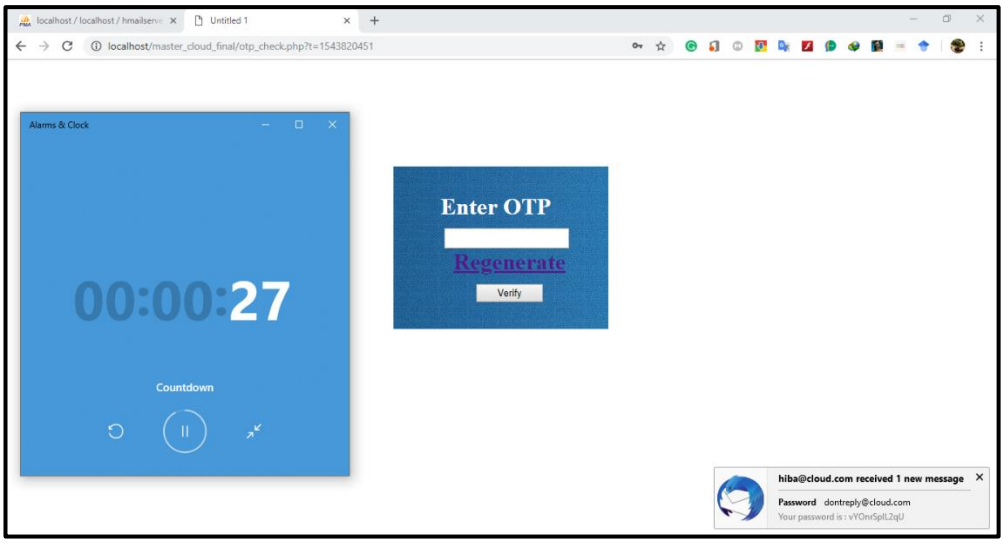

Figure 4.14. The password was generated and send via instructor email

- **Instructor upload files**

After the authentication process of the instructor done successfully, the system redirects the instructor to upload the lecture or assignment. Both lectures and assignments were uploaded in dedicated folders in the server. Figure 4.15 shows the page that enables the instructor to upload files.

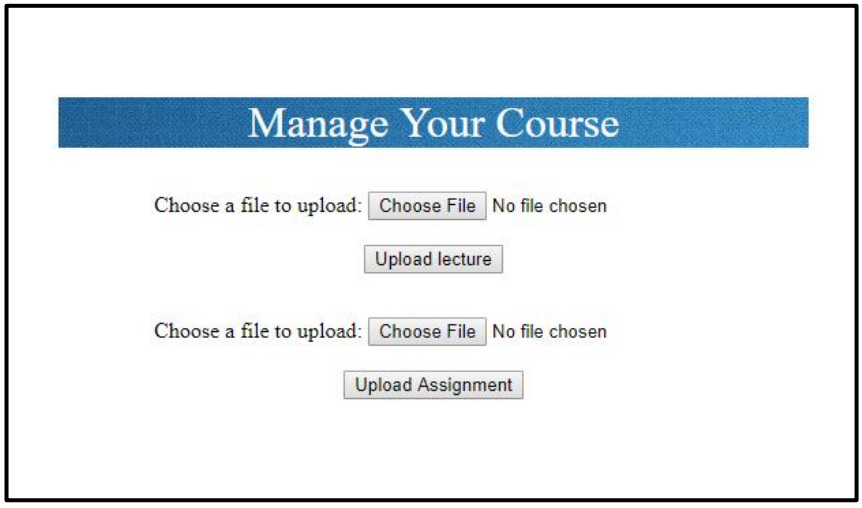

Figure 4.15. The instructor upload page of the designed system

4.2. Discussion

From the results illustrated in the previous section. The system has the capability to authenticate and authorize users of different roles. As shown above, the system has three roles, administrator, instructor, and student. Each role has the permissions needed to accomplish its task. The student can only view (download) the data stored on the cloud and cannot upload files to protect the system from uploading files contain malicious scripts. The instructor which is a trusted user has the permission of storing data on the cloud. One time password raises the reliability of the authentication process by sending the password to the user's email. This password is renewed every session and has a timestamp of 30 seconds and then became unusable.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

Chapter Five

Conclusion and Recommendation

This chapter concludes the work done in this thesis and presents the recommendation for future work.

5.1. Conclusion

A model of an online education system was created as an example of a cloud service. The system has three roles, the administrator who has full rights on the system and database, the instructor who has the rights to upload files in the system and student who has the rights to only view the data in the system. The proposed solution adopted in this thesis was carried out and achieved its goals whereas the authentication done by time-based one-time password and the user access only to cloud resources which were allowed to use. Role-based access control model was applied to the system in order to grant the least permission needed for the users to accomplish their task. This system designed and implemented using PHP language. A time-based one-time password also implemented using a blowfish hashing technique to authenticate the students. This password was generated and send to the user via email and has a time stamp of 30 seconds. After 30 seconds the password become expired and unusable to access the system.

5.2. Recommendation

The direction for future works for this thesis is to optimize the authentication and access control to the cloud resources by using a trusted

third party. A trusted third party has the user information and set access permissions to users based on their role. A trusted third party can be directory server acts as a repository of user information and credentials and set permissions on a group of similar users which has the same role. Also, an encryption technique can be implemented to save encrypted data on the cloud.

References

[1] S. Patidar, D. Rane, and P. Jain, "A survey paper on cloud computing," in *Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on*, 2012, pp. 394-398.

[2] P. Shynu and K. J. Singh, "A comprehensive survey and analysis on access control schemes in cloud environment," *Cybernetics and Information Technologies,* vol. 16, pp. 19-38, 2016.

[3] H. Sun, K. Sun, Y. Wang, and a. J. Jing, "TrustOTP: Transforming Smartphones into Secure One-Time Password Tokens," in *CCS '15 Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, Denver, Colorado, USA, 2015.

[4] J. Costa and A. Michalas, "Middle Man: An Efficient Two-Factor Authentication Framework," 2017.

[5] P. Patel and N. Gaud, "Access Control for Cloud Computing Through Secure OTP Logging as Services," *vol,* vol. 141, pp. 1-5, 2016.

[6] J. W. Rittinghouse and J. F. Ransome, *Cloud computing: implementation, management, and security*: CRC press, 2016.

[7] T. Erl, R. Puttini, and Z. Mahmood, *Cloud computing: concepts, technology & architecture*: Pearson Education, 2013.

[8] M. A. AlZain, B. Soh, and E. Pardede, "A new approach using redundancy technique to improve security in cloud computing," in *Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2012 International Conference on*, 2012, pp. 230-235.

[9] Z. Ghanbari, "A Literature Review on Cloud Computing Security Issues," *International Journal of Information, Security and Systems Management,* vol. 6, pp. 637-640, 2017.

[10] S. Kumar and R. Goudar, "Cloud computing-research issues, challenges, architecture, platforms and applications: A survey," *International Journal of Future Computer and Communication,* vol. 1, p. 356, 2012.

[11] M. M. Islam, S. Morshed, and P. Goswami, "Cloud computing: A survey on its limitations and potential solutions," *International Journal of Computer Science Issues (IJCSI),* vol. 10, p. 159, 2013.

[12] N. L. da Fonseca and R. Boutaba, *Cloud services, networking, and management*: John Wiley & Sons, 2015.

[13] R. L. Krutz and R. D. Vines, *Cloud security: A comprehensive guide to secure cloud computing*: Wiley Publishing, 2010.

[14] N. Amara, H. Zhiqui, and A. Ali, "Cloud Computing Security Threats and Attacks with Their Mitigation Techniques," in *Cyber-Enabled*

Distributed Computing and Knowledge Discovery (CyberC), 2017 International Conference on, 2017, pp. 244-251.

[15] S. T. D. Pandya, K. R. Narayan, S. Thakkar, T. Madhekar, and B. Thakare, "An overview of various authentication methods and protocols," *International Journal of Computer Applications*, vol. 131, pp. 25-27, 2015.

[16] E. C. S. Idrus, C. Rosenberger and J. Schwartzmann,, "A Review on Authentication Methods," *Australian Journal of Basic and Applied Sciences*, vol. 7, pp. 95-107, 2013.

[17] P. Pawar and R. Sheikh, "Implementation of secure authentication scheme and access control in cloud computing," in *ICT in Business Industry & Government (ICTBIG), International Conference on*, 2016, pp. 1-6.

[18] K. Ramesh and S. Ramesh, "Implementing One Time Password based security mechanism for securing personal health records in cloud," in *Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on*, 2014, pp. 968-972.

[19] S. A. El-Booz, G. Attiya, and N. El-Fishawy, "A secure cloud storage system combining time-based one-time password and automatic blocker protocol," *EURASIP Journal on Information Security*, vol. 2016, p. 13, 2016.

[20] B. S. Al-Attab and H. Fadewar, "Authentication scheme for insecure networks in cloud computing," in *Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC), 2016 International Conference on*, 2016, pp. 158-163.

[21] J. K. Liu, M. H. Au, X. Huang, R. Lu, and J. Li, "Fine-grained two-factor access control for web-based cloud computing services," *IEEE Transactions on Information Forensics and Security*, vol. 11, pp. 484-497, 2016.

[22] C. Langaliya and R. Aluvalu, "Enhancing cloud security through access control models: A survey," *International Journal of Computer Applications*, vol. 112, 2015.

[23] R. Charanya and M. Aramudhan, "Survey on access control issues in cloud computing," in *Emerging Trends in Engineering, Technology and Science (ICETETS), International Conference on*, 2016, pp. 1-4.

[24] A. Pathan and M. Ingle, "Survey Paper on User Anonymous Authentication Scheme for Decentralized Access Control in Clouds," *International Journal of Science and Research (IJSR)*, vol. 4, pp. 2024-2027, 2015.

[25] W.-B. Huang and W.-T. Su, "Identity-based access control for digital content based on ciphertext-policy attribute-based encryption," in

2015 International Conference on Information Networking (ICOIN), 2015, pp. 87-91.
[26] X. Ye, "Privacy preserving and delegated access control for cloud applications," *Tsinghua Science and Technology*, vol. 21, pp. 40-54, 2016.
[27] N. Haller, C. Metz, P. Nesser, and M. Straw, "A one-time password system," 2070-1721, 1998.
[28] D. M'Raihi, M. Bellare, F. Hoornaert, D. Naccache, and O. Ranen, "Hotp: An hmac-based one-time password algorithm," 2070-1721, 2005.
[29] D. M'Raihi, S. Machani, M. Pei, and J. Rydell, "Totp: Time-based one-time password algorithm," 2070-1721, 2011.
[30] G. Pradyumna, "Comparison of MD5 and Blowfish Algorithm," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 5, pp. 506-511, 2016.
[31] F. Wiemer and R. Zimmermann, "High-speed implementation of bcrypt password search using special-purpose hardware," in *ReConFigurable Computing and FPGAs (ReConFig), 2014 International Conference on*, 2014, pp. 1-6.
[32] L. Ertaul, M. Kaur, and V. A. K. R. Gudise, "Implementation and Performance Analysis of PBKDF2, Bcrypt, Scrypt Algorithms," in *Proceedings of the International Conference on Wireless Networks (ICWN)*, 2016, p. 66.
[33] M. Curphey, D. Endler, W. Hau, S. Taylor, T. Smith, A. Russell, *et al.*, "A guide to building secure web applications," *The Open Web Application Security Project*, vol. 1, 2002.
[34] R. Thion, "Access Control Models," in *Cyber Warfare and Cyber Terrorism*, ed: IGI Global, 2007, pp. 318-326.
[35] (November 2018). *HTTP Server Project*. Available: https://httpd.apache.org/
[36] (November 2018). *What is PHP?* Available: http://php.net/manual/en/intro-whatis.php
[37] (November 2018). *What can PHP do?* Available: http://php.net/manual/en/intro-whatcando.php
[38] (November 2018). *MySQL*. Available: https://www.oracle.com/technetwork/database/mysql/index.html
[39] (November 2018). *hMailServer - Free open source email server for Microsoft Windows*. Available: https://www.hmailserver.com/
[40] (November 2018). *Thunderbird — Software made to make email easier. — Mozilla*. Available: https://www.thunderbird.net/en-US/

Appendix

The following lines of code illustrate time-based one time password using blowfish hashing

A.1: Generation of one time password:

```
<?php

ob_start();

session_start();

$host="localhost"; // Host name

$username="root"; // Mysql username

$password=""; // Mysql password

$db_name="cloud"; // Database name

$tbl_name="users"; // Table name

mysql_connect("$host", "$username", "$password")or die("Cannot Connect To Localhost");

mysql_select_db("$db_name")or die("cannot select DB");

$username=$_SESSION['username'];

$sql="SELECT * FROM $tbl_name WHERE username='$username' and status='1' ";
```

```
$result=mysql_query($sql);

$row = mysql_fetch_assoc($result);

$email_id=$row['email'];

$string='123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'; //

$tmpString="";

$length=strlen($string);

$generatedPassword = "";

//generate one time password

for ($i = 0; $i < $length; $i++)

        $tmpString = str_shuffle( $string ); //generate the key

//truncate the key to be send to the user

        $generatedPassword = substr( $tmpString, 37, 8);

//send one time password via email

$to = $email_id;

$subject = "Password";

$txt = "Your password is: $generatedPassword";

$headers = "From: dontreply@cloud.com";

mail ($to,$subject,$txt,$headers);
```

```
//Blowfish Hashing

if (CRYPT_BLOWFISH == 1)

{

$generatedPassword=crypt($generatedPassword,'$2a$09$weprovideimmun
esystemm$');

}

//record time of one time password generation

$ otptime =0;

$otptime=time();

$_SESSION['generatedPassword']=$generatedPassword;

$_SESSION['otptime']=$otptime;

header("location:otp_check.php");

?>
```

A.2: One time password check

```
<?php
ob_start();
session_start();
$t1=$_SESSION['otptime'];
$t2=time(); // time of submitting one time password
if(isset($_POST['SubmitOTP'])){
        if (CRYPT_BLOWFISH == 1)
{
$chkPassword=crypt($_POST['OTP'],'$2a$09$weprovideimmunesystemm
$');
}
if(($t2-$t1) <= 30){
if($_SESSION['generatedPassword']==$chkPassword){
$_SESSION['generatedPassword']="";
$chkPassword="";
header("location:user_profile.php?user=".$_SESSION['myusername']."");
}
```

```
else{

echo "error";}

}else

{

$_SESSION['generatedPassword']="";

echo("please regenerate otp again");

}}?>
```