AES BASED ENCRYPTION SCHEME FOR MOVING TARGET DEFENCE IN CLOUD STORAGE

A Thesis Submitted
in Partial Fulfillment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

In
Field of Specialization

by
Akanksha Singh
(Enrollment no:……)

Under the Supervision of
Assistant Professor
Abhinav Singh
CSE Department BBDU, Lucknow

to the

SCHOOL OF ENGINEEGING

BABU BANARASI DAS UNIVERSITY
LUCKNOW

JUNE, 2020

ACKNOWLEDGEMENTS

Firstly I would like to thank Er. Abhinav Singh for giving me the wonderful opportunity to complete my M.techthesis under his supervision, it is truly an honor. Thank you for all the advice, ideas, moral support and patience in guiding me through this project.
Without his invaluable guidance, this work would never have been a successful one.

We are likewise appreciative to Er.Abhinav Singh, Dr. Praveen Kumar Shukla and other in Department of Computer Science and Engineering for rousing us in improving the calculations.

At long last we might want to thank our folks for their help and allowing us remain for additional days to finish this task.

Date- 25/06/2020
BBD University

Akanksha Singh

AES Based Encryption Scheme for Moving Target Defence in Cloud Storage

ABSTRACT

Now days cloud computing become one of the main topicof IT and main point is cloud data storage security. Unlike prior efforts in cyber security research, aAdvanced Encryption Scheme based theory, called moving target defense (MTD), increases the complexity and costs for attacks by effectively restricting the vulnerability exposure and the attack opportunities through various continually-changing evaluation, development mechanisms and strategy. Advanced Encryption Standard (AES) was the classical scheme of the traditional symmetric-key encryption schemes. Now it has been gradually replaced by the triple DES or AES so that the encoder has a larger key space. In this paper, we propose a dynamic 3-layer encryption scheme based on DES and network coding, with a low-complexity partial key update mechanism. Based on the theoretical analysis, the new scheme is shown to have the benefit to achieve a dynamic transition between efficiency and security, which increases its adaptability to various cyber conditions. The simulation results also show that the running ratio of the new scheme is relatively lower than or comparable to the AES.Cloud storage provides user to access remotely store their data so it becomes necessary to protect data from unauthorized access, hackers or any type of modification and malicious behavior. Security is an important concern. The meaning of data storage security is to secure data on storage media. Cloud storage does not require any hardware and software management.We have developed a web application through which user can share data. This thesis enhanced advance security goal for cloud data storage.

TABLE OF CONTENT

LIST OF FIGURE

LIST OF ABBREVIATION

JVM	Java Virtual Machine
VOC	voice of customer
ACK	Acknowledgement
SRS	Software Requirement Specification

Chapter 1

INTRODUCTION

Moving target defense (MTD) is one of the cyberspace game-changing revolutionary technologies proposed by Federal Networking and Information technology Research and Development (NITRD) in recent years[1]. Nowadays, network security configurations are typically deterministic, static and homogeneous. These features reduce the difficulties for cyber attackers scanning the network to identify specific targets and gather essential information. Thus, the attackers take the asymmetric advantages of building up, launching and spreading attacks, and the defenders are at a passive position. The existing defense mechanisms and approaches cannot reverse this situation. Therefore, MTD is proposed as a new revolutionary technology to alter the asymmetric situation of attacks and defenses[2][3]. It keeps moving the attack surface of the protected target through dynamic shifting, which can be controlled and managed by the administrator. In this way, the attack surface exposed to attackers appears chaotic and changes all the time. Thus, the work effort, i.e., the cost and complexity for the attackers to launch a successful attack, will be greatly increased. As a result, the probability of successful attacks will be decreased, and the resiliency and security of the protected target will be enhanced effectively. The revolutions of MTD can be summarized from the following three aspects [3]: (i)Dynamic defense: the transformation from static to dynamic in system architecture. (ii)Active defense: the transformation from passive perception into actively setting blocks to the weakness and virus in security mechanism. (iii)Flexible defense: the transformation from regular into a flexible operation mode. The basic goal of MTD is to achieve the active defense to the external attacks based on unknown vulnerabilities and backdoors. To date, MTD has been studied in various contexts, including cloud computing [4], [5] and web applications [6], [7].

The similar dynamic idea can also be adopted in cryptography design. It is well known that Data Encryption Standard (DES) has been widely used as a mainstream symmetrical encryption. Meanwhile, DES has laid a foundation for the development and application of modern block cipher theory[8]. At present, with the rapid development of computing power, the classic iterated block cipher DES has

become very fragile, which causes the effective realization of DES crack by the exhaustive attack. So, it has gradually been replaced by the triple-DES algorithm or Advanced Encryption Standard (AES) so that the encoder has a large enough key space. However, due to the existence of S-box, DES still has benign incalculability to analysis attack[9]. Two of the most effective methods of iterated block-cipher attack are differential cryptanalysis (DC) and linear cryptanalysis (LC). DC is the first published method that can crack DES successfully in the average computational complexity of less than 2^{55}.

Although the above-mentioned algorithms such as the triple DES and AES have gradually replaced the classical DES, they still cannot meet the dynamic security requirements of the intelligent information network due to their static extension to the key space. In this paper, we present an encryption scheme to improve DES under the concept of MTD, by means of (linear) network coding (NC), which advocates linearly combining coding along with data propagation[14]. The following two reasons motivate us to choose NC. First, NC, which has been used in [15] and [16] for encryption scheme design, changes the static nature of network information transmission, so it is a good match to achieve the dynamic, active and random features of MTD as defined in [3]. Second, the use of NC as an encryption scheme has the potential to resist the exhaustive attack, as an *L*-bit plaintext may correspond to □(2*L*□*L*) possible ciphertexts.

The National Institute of Standards and Technology (NIST) define cloud computing as “a model for user convenience, on-demand network access contributes the computing resources (e.g. network, storage, application, servers and services) that can be rapidly implemented with minimal management effort or service provider interference” [5].

As we proposed the concept of cloud data storage security strategy capable to overcome the shortcomings of traditional data protection algorithms and improving security using encryption decryption techniques AES (Advanced Encryption standard) are adoptable to better security for the cloud. We have developed a web application through which user can share data. This thesis enhanced advance data security and user authorization in cloud.

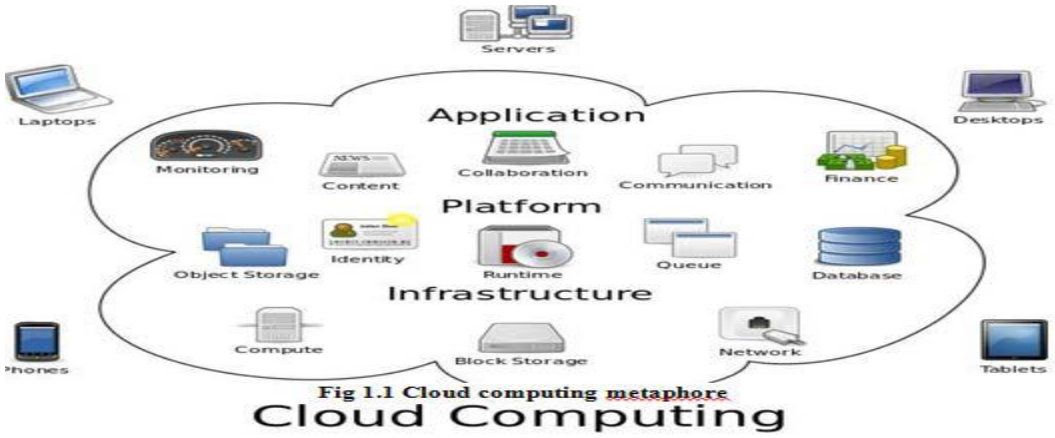


Fig 1.1 Cloud computing metaphore

Fig 1 represents cloud computing metaphore.

Cloud computing is a general term for anything that involves hosted services over the Internet. These services are broadly divided into three categories: [6]

Infrastructure as a Service (IaaS): In IaaS model computer resources such as storage, computing capabilities are made available to the customer on demand. It's cost saving model. In this model customer only pay to use IT infrastructure as needed.[14]

E.g: Amazon Web Services, Virtual machines, servers, storage, load balancers, network.

Platform as a Service (PaaS): In the PaaS model a development environment is offered to the customer which is managed by the provider. On which customer can develop and run their applications without building and managing complex infrastructure.[14]

E.g: Google Application Engine, Execution runtime, database, Web server, Development tools.

Software as a Service (SaaS): In the SaaS model an application is offered to the customer by the cloud service provider. In which application is hosted by the provider at their infrastructure and distributed over the network as a service on demand.[14]

E.g: Online word processing and spreadsheet tools, Microsoft office, Email, communication, Games.

Cloud computing is typically classified in four types.

Public cloud: Public cloud is publicly accessible cloud which is managed by third parties. All customers share a common infrastructure pool with limited configuration. The cloud provider is responsible for creation and ongoing maintenance of the public cloud.[6][14]

Private Cloud: Private cloud is accessible only by an organization and also managed by the organization. Private cloud enables an organization to use cloud computing by means centralizing access to IT resources from different geographical location. .[6][14]

Hybrid cloud: Hybrid cloud combines both public and private cloud models. With Hybrid cloud organization can utilize third party cloud provider service in a full or partial manner. Thus, Hybrid cloud increases flexibility of computing.[6][14]

Community Cloud: Community cloud is a multi-tenant infrastructure which is shared among several organizations. And it is managed, governed and secured by all the participating organization. These organizations have similar cloud requirements and their ultimate goal is to achieve business objective. It is beneficial in order to cost saving.

Cloud based environment there are many security issues such as authentication, integrity, privacy, virtualization, confidentiality, large amount data processing, scalability, access control etc.[8] Traditional security approaches are no longer suitable for data and application in cloud. [1][2][3][4]

PRINCIPLES OF DATA SECURITY

The four basic principles of confidentiality, availability, integrity and privacy are used in all data security techniques.

Confidentiality: confidentiality means that information be available only when people need. The principle of confidentiality means that data is available only to authorized persons and when needed.[2]

Availability: The principle of availability means that information should be made available when the authorized persons need it.[2][4]

Integrity: Data integrity is important in compliance with the laws governing data integrity and should guarantee high quality, accuracy, consistence, accessible data.[2]

1.1 PROPOSED PROBLEM STATEMENT

There are various issues are found which required further research. Since cloud computing is a utility available on net,[1] so various issues like user privacy, data theft and leakage, eaves dropping, unauthenticated access and various hackers attacks are raised.[9]

Under the above mentioned title, I am incorporating security control mechanism which supports to the authentication, authorization and data security techniques in the cloud computing system. Very important factor for secure cloud environment is insurance of adequate protection of data and information.

In order to secure the data in cloud computing, adequate controls are needed as: consideration of all forms of data and privacy requirements, appliance of confidentiality, creation of data asset catalog, integrity and availability, as well as appliance of identity and access management.

 Introduce a mechanism to enhance the data security and user authorization in cloud computing.

 To enhance extra storage information with smallest key size.

 To reduce the storage requirement using smallest key size.

 Provide the high security level with the help of smallest key size.

 To reduce bandwidth.

 To maintain the confidentiality and integrity of data.

1.2OBJECTIVE

We have proposed a system with following objectives.

- ✓ To understand the security issues related with cloud computing.
- ✓ To provide high quality services to the users.
- ✓ To provide high data security in cloud based environment using encryption techniques.
- ✓ To check the Authentication of users i.e. authorized person can access the data.
- ✓ To maintain the privacy of data.

1.3MOTIVATION

As per the literature survey, we analyzed that there are several attackers which do various activities such as- modification in transmitted data, read and misuse the confidential information an soon. As we know we can download & upload our information on which we want to keep confidential. There are various security algorithms presents such as encryption and decryption by which cloud try to maintain privacy of data and user authentication etc.

1.4 Thesis Organization

The thesis scope is to demonstrate the main issues that are related to the Tweeter Sentiment analysis, and plotting graph of its analysis, including the application, main duties, etc.

Chapter two demonstrates main theoretical topics and illustrations. It is a summary for the material related to clustering and protocols of the tweeter sentiment analysis, in addition to major structure.

The Software requirement analysis work is shown in Chapter three in details.

It illustrates the main steps of work, diagram of development process and program running, also, the program flow in details. Chapter four shows the system design details, with different measurements.

CHAPTER 2

LITERATURE SURVEY

Based on broad writing review identified with the A Network Coding and AES Based Encryption Scheme for Moving Target Defense has been thought about in this part.

Ronald S. Cordova et.al[2017][1]to analyze and compare the performance of AES, Blowfish, and RSA algorithms and also shows that Blowfish manifested a higher time efficiency ratio.

S.MahdiShariatiet.al[2015][2] investigated the security challenges and issues based on two perspectives that are –Data security, Privacy and protection. Authors also discussed about cloud computing, their services and related challenges such as availability, data location, data isolation and recovery , and also mention risks such as -insecure interface, data loss or leakage malware etc. And also analyzed protection & privacy related issues that are- loss of control, invalid storage etc. and also analysis of data storage issues like–service provider, data integrity recovery & backup with principles such as-transparency, integrity, minimization etc.

Neha A Puriet.al[2014][3] discussed about the appliation on Cloud computing and their service models and also about the Encryption Algorithms. In this paper author also increases the security level by implementing ECC Algorithm.

Subhashiniet.al[2011][2] have depicted all security related issues present in the distributed computing. The different organizations of the cloud and every one of the issues present in every sending are been characterized in the paper. They have characterized in regard to the administration conveyance where in each kind of SaaS, PaaS, and IaaS. They specifically characterized all the security issues in the product as an administration of distributed computing. In Issues of SaaS, there are classifications dependent on information, arrange, web applications and virtualization vulnerabilities.

Balachandrareddyet.al[2010][4] have talked about administration level understandings that are been issued by client to supplier before getting into cloud. This is the main trust a supplier will see from client, yet it insufficient to give security as it doesn't answers the issues to the misfortunes of the client, there ought to be sure changes as per the sort of administration a client is working and should be institutionalized with favored client get to, information isolation, area of information and so on.

DiaoZheet.al[2017][4] to achieve data security of Cloud storage and to formulating corresponding security policy and analyzing the security risks and relate them with the previous research results. And

also focus on the relevant security technology which is based on the structural characteristics of cloud storage system.

DebajyotiMukhopadhyay et.al[2014][5] discussed about the Cloud Computing, Cloud and Data Security and their related working model.And also discuss data security by implementing key agreement, encryption and signature verification with Hyperelliptic curve cryptography.

Subhash Chandra Patel et.al[2015][6]discussed about the overview of the cloud computing and also have to proposed a method to achieve security with combined approach of PGP and Kerberos and this method provides authentication, confidentiality, integrity, and privacy featues to Cloud Service Providers and Cloud Users.

KresimerPopovicet.al[2010] [7] have talked about various security concerns present in the cloud display which is losing privacy and uprightness of the information while exchange, stockpiling and recovery. They additionally examined on the things that will be think about where the dangers are available in distributed computing like from client to kind of administrations. With the above issues they reasoned that we have to take security and protection in giving cloud administrations.

Patrick Mc. Daniel et.al[2010 [10] portrayed about difficulties of security and upgrades that are to be made over cloud for secure information over cloud. They focused chiefly on security issues over cloud occurrences. The occurrences over cloud will keep running on some base framework which may trade off and causes a security issue. There are additionally outside foes over cloud which may need security of occasions from outsiders. They talked about specific open doors which are to the extraordinary difficulties for analysts. The distributed computing security concerns were examined in detail in [13] the primary issues talked about were protection worries because of outsider clients. As the security because of programmer's increment over web and the distributed computing is absolutely on web, there are diverse issues like assaults are examined on it.

SameeraAbdulrahmanAlmulla et.al[2010][11]have examined about administration in distributed computing ,the difficulties with respect to the data security worries in regards to classification, integrity and accessibility. They talk about security difficulties of distributed computing in regards to character and access the executives.

Steve Mansfield et.al[2008][12] has talked about with respect to the upsides of having the cloud in the meantime the issues present in cloud. When we use in our edge territory we utilize numerous security sides like firewalls DMZ'z and so on., where as in cloud all are on a remote framework with no security. Creator predominantly indicates out that we require have a lot of trust in the plan of framework with great validation and approval capacities.

ManpreetKauret. al[2016][8] reviewed the comparative analysis of various encryption algorithms. Cloud computing ,data security challenges related with their deployment, service and network related models are described. And also discussed about the encryption algorithms .

PalkeshSoniet. al[2016][14] proposed a survey paper in which deep analysis of security issues are described and also focus on the challenges in Cloud computing.

BASE PAPER

Moving target defense (MTD) is one of the cyberspace game-changing revolutionary technologies proposed by Federal Networking and Information technology Research and Development (NITRD) in recent years[1]. Nowadays, network security configurations are typically deterministic, static and homogeneous. These features reduce the difficulties for cyber attackers scanning the network to identify specific targets and gather essential information. Thus, the attackers take the asymmetric advantages of building up, launching and spreading attacks, and the defenders are at a passive position. The existing defense mechanisms and approaches cannot reverse this situation. Therefore, MTD is proposed as a new revolutionary technology to alter the asymmetric situation of attacks and defenses[2][3]. It keeps moving the attack surface of the protected target through dynamic shifting, which can be controlled and managed by the administrator. In this way, the attack surface exposed to attackers appears chaotic and changes all the time. Thus, the work effort, i.e., the cost and complexity for the attackers to launch a successful attack, will be greatly increased. As a result, the probability of successful attacks will be decreased, and the resiliency and security of the protected target will be enhanced effectively. The revolutions of MTD can be summarized from the following three aspects [3]: (i)Dynamic defense: the transformation from static to dynamic in system architecture. (ii)Active defense: the transformation from passive perception into actively setting blocks to the weakness and virus in security mechanism. (iii)Flexible defense: the transformation from regular into a flexible operation mode. The basic goal of MTD is to achieve the active defense to the external attacks based on unknown vulnerabilities and backdoors. To date, MTD has been studied in various contexts, including cloud computing [4], [5] and web applications [6], [7].

In this paper author has developed a high quality desktop application with the help of AES encryption technique and also maintain security issues such as- authentication, integrity, privacy and confidentiality. In this proposed system authors objective is-

 To understand the security issues and provide high quality services.

 To provide high data security by using Steganography, encryption and decryption techniques and also minimizing the uploading and downloading time on cloud storage.

Implementation of the proposed system has to be done into 8 steps these are-

- Registration module
- Login module

- Encryption
- Upload and download module
- FTP module

For encrypt the data AES(Advanced Encryption Algorithm)is used. AES support128, 192, 256 bits block cipher. AES algorithm mainly repeats four functions to encrypt data.

Algorithm

 Key Expansions

 Initial Round

 Round

 Final Round

DISADVANTAGES

 Less Data security.

 No user authentication for example-OTP.

 Less data storage capacity.

Conclusion

This part generally discusses the papers that are suggested while influencing this proposal to report. Each one of these papers give information related to learning of total lead, their present courses of action, and systems used moreover their central focuses and imperatives.

Chapter 3

SOFTWARE REQUIREMENT SPECIFICATION

This part depicts about the prerequisites. It determines the equipment and programming prerequisite that are needed for software to keeping in mind the end goal, to run the application appropriately. The Software Requirement Specification (SRS) is clarified in point of interest, which incorporates outline of this exposition and additionally the functional and non-practical necessity of this thesis.

3.1 General Description

The reason behind the framework prerequisites and determination record is to depict the assets and administration of those assets utilized as a part of the configuration of the Public Key Cryptosystem for information partaking in distributed storage. This framework necessity and particular will likewise give insights with respect to the utilitarian and non-useful prerequisites of the venture.

3.1.1 Users Perspective

The Characteristic of this task work is to give information adaptability security while sharing information through cloud. It gives a proficient approach to share information through cloud.

3.2 Non Functional Requirement

Non-utilitarian necessities are the prerequisites which are not straightforwardly having a place with the specific capacity gave by the framework. This gives the criteria that can be utilized to finish up the operation of a framework rather than particular practices.

This can be utilized to relate the rising structure properties, for instance, immovable quality, response time and store inhabitancies. Here again they ought to portray objectives on the system, for instance, the capacity of the data yield devices and data representation used as a piece of structure interfaces. In all probability all non-helpful essentials can be relating to the system as whole rather than to individual structure highlights. This suggests they are every now and again essential appear differently in relation to the individual commonsense necessities. Non utilitarian necessity gets through the client needs, in view of spending plan limitations, hierarchical approaches, and the requirement for interoperability with other programming and equipment frameworks.

The going with non-valuable requirements are meriting thought.

- Security: The framework ought to permit a secured correspondence between information proprietor and beneficiary.
- Reliability: The system should be trustworthy and ought not corrupt the execution of the present structure and should not to provoke the hanging of the structure.

3.3 System Requirement

3.3.1 Hardware Requirement

- **Processor** : intel/amd
- **Keyboard** : 104 Keys
- **Floppy Drive** : 1.44 MB MHz Pentium III
- **RAM** : 128 MB
- **Hard Disk** : 10 GB
- **Monitor** : 14” VGA COLOR
- **Mouse** : Logitech Serial Mouse

- **Disk Space** : 1 GB

3.3.2 Software Requirements

- **Operating System** : Win 2000/ XP
- **Server** : Apache Tomcat
- **Technologies used** : Java, Servlets, JSP, JDBC
- **JDK** : Version 1.7
- **Database** : My SQL 5.0

3.4 Feasibility Study

Believability is the determination of paying little respect to whether an undertaking justifies action. The framework followed in building their strength is called acceptability Study, these kind of study if a task could and ought to be taken.

Three key thoughts included in the likelihood examination are:

- Technical Feasibility
- Economic Feasibility
- Operational Feasibility

3.4.1 Technical Feasibility

Here it is considered with determining hardware and programming, this will effective fulfil the client necessity the specialized requires of the framework should shift significantly yet may incorporate

- The office to create yields in a specified time.
- Reaction time under particular states.
- Capacity to deal with a particular segment of exchange at a specific pace.

3.4.2 Economic Feasibility

Budgetary examination is the often used system for assessing the feasibility of a projected structure. This is more usually acknowledged as cost/favourable position examination. The method is to center the focal points and trusts are typical casing a projected structure and a difference them and charges. These points of interest surpass costs; a choice is engaged to diagram and realize the system will must be prepared if there is to have a probability of being embraced. There is a consistent attempt that upgrades in exactness at all time of the system life cycle.

3.4.3 Operational Feasibility

It is for the most part identified with human association and supporting angles. The focuses are considered:

- What alterations will be carried through the framework?
- What authoritative shapes are dispersed?
- What new aptitudes will be needed?
- Do the current framework employee's individuals have these aptitudes?
- If not, would they be able to be prepared over the span of time?

3.5 Resource Requirement

3.5.1 Java

Java is a stage autonomous programming dialect. It is outline to be basic and convenient crosswise over diverse stages.

The java programming vernacular is an unusual state tongue that can be portrayed by most of the going with in vogue expressions:

- Object oriented
- Simple
- Architecture neutral
- Portable
- Robust
- Dynamic

- Secure

The Java API is a broad social affair of moment programming fragments that give various profitable limits, for instance, graphical customer interface (GUI) contraptions. The Java API is accumulated into collections of correlated classes and interfaces; these collections are recognized as packs.

Java stage gives you the accompanying elements:

- **The essentials:** Items, strings, strings, numbers, info, yield, information structures, framework properties, date, time et cetera.
- **Applets:** The arrangement of traditions utilized by applets.
- **Networking:**URLs, TCP, UDP attachments, and IP addresses.
- **Internationalization:** Help for composing projects that could be restricted for clients around the world. Projects can naturally adjust to particular local people and be shown in the suitable dialect.
- **Security:** Mutually low level and abnormal state, together with electronic marks, open and private key administration, right of entry control and authentications.
- **Software components:** Recognized as JavaBeans, could connect to existing parts structural designs.
- **Object Serialization:** Let's Permits lightweight tirelessness and correspondence by means of Remote Method Invocation (RMI).
- **Java Database Connectivity (JDBC):**Give consistent entree to an extensive variety of social databases.

Advantage of java technology:

- **Get started quickly:** In spite of the fact that the java programming dialect is an intense article arranged dialect, it is anything but difficult to learn, particularly for software engineers effectively acquainted using C or C++.
- **Write less code:** Examinations of undertaking estimationsadvise that a framework built in the java language tongue shall be 4 times more diminutive compare tosimilar program in C++.
- **Write better code:** Java dialect energizes great coding rehearsals and its trash gathering serves to evade memory spills. Its item introduction, its javaBeans segment building design and its far

reaching, effectively extendible API let's to use again other individuals tried code and present less bugs.

- **Develop programs more quickly:** Headway time can be as much as twice as speedy against making the similar program in C++.
- **Write once, run anywhere:**Since 100% immaculate java projects are ordered into system autonomous byte codes, run reliably on whichever java stage.
- **Distribute software more easily:** Update applets smoothly from a middle server. The applets misuse the segment of allocating new classes could be stacked "on the fly", without recompiling the whole system.

3.5.2 Java Server Page

Java Server Pages development allows you to put scraps of servlet coding particularly into a substance based record. A java server page is a substance based record that holds two sorts of substance: static format data, which should be imparted in several substance based association, for instance, XML, WML, HTML and Java server page segments, which choose how the page fabricates component content.

Java Server Page (JSP): An extensible Web innovation that make use of layout information, custom components, scripting dialects, and server side Java programming language articles to homecoming element substance to a customer. As indicated by JSP model1 we can build up the application as:

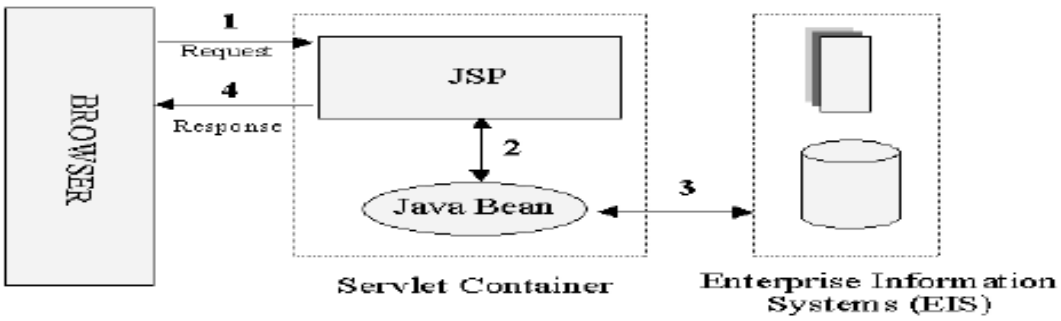


Figure 3.5.2.1 Architecture of jsp model 1

Commonly the layout information is XML/HTML components, and as a rule the customer is a Web programs.According to above replica the presentation method of reasoning must be executed in java

server page and the business justification must be realized as a component of Javabean and this model assist us in disconnecting the appearance and business basis.

For sweeping extent reaches out instead of using model1 it is perfect to use model2 Model View Controller. Struts structure is in light of model 2.JSP allows you to specific the dynamic bit of your pages with the static HTML.Here it just makes the steady HTML in the run of the mill way, using any Web page building mechanical assemblies it frequently uses. Here again encase the dynamic code for the parts in exceptional labels, the greater part of which begin with "<%" and end with "%>". You ordinarily give your record a .jsp augmentation, and regularly introduce it in wherever you could put a typical Web page.

Despite the fact that what you compose frequently seems more like a customary HTML record compare to a servlet, in the background, the java server page just acquires change over to an average servlet, through the static HTML basically being printed to the yield stream joined with the servlet's organization method.

3.5.3 JavaScript

Java Script is a one of the object oriented; lightweight, script based computer programming language dialect tongue which was made by Netscape Communication Corporation. JavaScript holds the change equally client and server parts of application of Web program and the client side; this could be employ to form programs which are implemented by a Web program inside the association of a Web page on the server side, this should be employed to make a Web server programs which could deal with information set up together by a Web project and thereafter overhaul the program's showcase moreover.

Despite the fact that JavaScript underpins client and server Web programming, this lean toward JavaScript at customer side programming subsequent to a large portion of the programs bolsters this one. JavaScript is as simple as to study as HTML, and JavaScript explanations could be incorporated in HTML reports by encasing the announcements among a couple of scripting labels.

Here there are a couple of issues we can do with JavaScript

- Approve the substance of a structure and build computations.
- Include looking over or changing information to the Browser's status line.

- Invigorate pictures or pivot pictures that change when we shift the mouse above them.
- Identify the program being used and presentation distinctive substance for diverse programs.
- Identify introduced modules and inform the client if a module is needed.
- JavaScript can do a great deal more with JavaScript, including making whole application.

JavaScript and Java are altogether diverse dialects. A couple of the obvious contrasts are:

- Java applets are usually indicated for a situation inside the web chronicle; JavaScript could impact every bit of the Web report itself.
- JavaScript is most suitable to essential applications and inserting instinctive components to Web pages and Java could be used for amazingly multifaceted applications.

Here there are various distinctive complexities yet the fundamental object to review is that JavaScript and Java are autonomous lingos. They are together useful for unmistakable objects; really they could be used both to join their ideal circumstances.

3.5.4 JDBC

JDBC with a last goal to set a self-administering database benchmark API for java made database blend, or JDBC. This offers a nonspecific SQL database access portion that provides an expected interface to a mixed pack of RDBMS. This trustworthy interface is refined in the use of "unit" database framework units, or drivers. On the off chance that a database merchant wishes to have JDBC strengthen, he or she canallot to the driver to every stage that the database and java keep running on. To get a more expansive attestation of JDBC, sun builds up JDBC's system concerning ODBC.Java database integration gives uniform access to an extensive variety of social databases. MS Access databases are utilized for rapidly overhauling the store table.

Plan objectives for JDBC are as per the following:

- **SQL Level API**

  The organizers experience that their essential objective was to portray a SQL interface for java. Yet not the most diminished database interface level achievable, this is an adequately low level for strange state devices and APIs to be made. Then again, it is at an adequately abnormal state for application programming architect to use it surely. Finishing this target looks into future

mechanical assembly shippers to "make" JDBC code and to cover countless difficulties from the end customer.

- **SQL Conformance**

  SQL sentence structure changes as this movement from database shipper to database vender.With an end target to support a wide grouping of traders;JDBC will authorizeeveryinvestigation articulation to be moved out through this to the basic database driver. This authorizes the integration unit to grip non-standard usefulness in approach that is appropriate for their clients.

- **JDBC must be implemented on top of basic database interfaces**

  The JDBC SQL API must "sit" on top of other fundamental SQL level APIs. This target grants JDBC to use active ODBC level drivers by the usage of an item interface. This interface could be making an elucidation of JDBC calls to ODBC and the other route around.

- **Give a java interface that is steady with whatever is left of the java framework**

  As of java's affirmation in the customer gathering thusly for, the makers feels that they should not to wander away from the present blueprint of the middle java structure.

- **Keep it simple**

  These objective likely shows up in all products outline objective postings. JDBC is no special case. Sun considered that the configuration of JDBC ought to be extremely basic, taking into account stand out strategy for finishing an undertaking for every component. Permitting copy usefulness just server to confound the client of the API.

- **Keep the common cases simple**

  Since as a general rule, the typical SQL calls utilize the software engineers are straightforward INSERT's, SELECTs, UPDATE's and DELETE's these request should be anything but difficult to perform with JDBC.Be that as it may, more intricate SQL explanations ought to likewise be conceivable.

Conclusion

This part gives subtle elements of the practical prerequisites, non-utilitarian necessities, asset prerequisites, equipment necessities, programming necessities and so on. Again the non-utilitarian prerequisites thus contain item necessities, authoritative prerequisites, client prerequisites, fundamental operational necessities and so on.

CHAPTER 4

SYSTEM ANALYSIS

4.1 Introduction to System Analysis

4.1.1 System

A system is an orderly group of interdependent components linked together according to a plan to achieve a specific objective. Its main characteristics are organization, interaction, interdependence, integration and a central objective.

4.1.2 System Analysis

System analysis and design are the application of the system approach to problem solving generally using computers. To reconstruct a system the analyst must consider its elements output and inputs, processors, controls feedback and environment.

4.2 Existing System

The basic goal of MTD is to achieve the active defense to the external attacks based on unknown vulnerabilities and backdoors. To date, MTD has been studied in various contexts, including cloud computing and web applications. The similar dynamic idea can also be adopted in cryptography design. It is well known that Data Encryption Standard (DES) has been widely used as a mainstream symmetrical encryption. Meanwhile, DES has laid a foundation for the development and application of modern block cipher theory. At present, with the rapid development of computing power, the classic iterated block cipher DES has become very fragile, which causes the effective realization of DES crack by the exhaustive attack. So, it has gradually been replaced by the triple-DES algorithm or Advanced Encryption Standard (AES) so that the encoder has a large enough key space.

Since security is a major concern in the cloud, it is of great importance for our Smart-Frame to provide a solution to address that. As mentioned earlier, one of the huddles for widely deploying security solutions based on public key cryptography is the high cost for maintaining PKI. Under traditional public key cryptography, each participating entity must locate and verify the public keys of the receivers. This is especially burdensome for end user devices in our Smart-Frame, which are usually assumed as limited in networking capacity.

4.3 DISADVANTAGES

- Traditional public key cryptography always requires a setup phase to generate public keys of the receiving parties.

- Under traditional public key cryptography, each participating entity must locate and verify the public keys of the receivers.

- Assumed as limited in networking capacity.

4.4 Proposed System

In this thesis, we present an encryption scheme to improve AES under the concept of MTD, by means of (linear) network coding (NC), which advocates linearly combining coding along with data

propagation. The following two reasons motivate us to choose NC. First, NC, which has been used in for encryption scheme design, changes the static nature of network information transmission, so it is a good match to achieve the dynamic, active and random features of MTD as defined. Second, the use of NC as an encryption scheme has the potential to resist the exhaustive attack, as an*L*-bit plaintext may correspond to possible ciphertexts.

We have proposed a system with following objectives.

• To understand the security issues related with cloud storage.

• To provide high quality services to the users.

• To provide high data security in cloud based environment using encryption and decryption.

4.5 Advantages of the Proposed System

- It provides secure communication services

- Availability will be assured by minimizing the time and frequencies for updating identities (public-keys).

4.6 Project Module Description

AdminModules

- Login
- UserDetails
    - AddUser
    - EditUser
    - DeleteUser
    - ViewUser Details
- CloudDetails

    - ViewDetails
- TransactionDetails
    - SelectUser
    - Viewlog Details
- SignOut

UserModules

- Login
- ShowProfile
- Upload aFile
    - Userhas to selectthe file from thelocal system
    - File Encrypted using AES Encryption
    - Upload to cloud storage
    - Insert aTransactionRecord
    - ShowUpload SuccessfulMessage to user

- Transaction
- Signout

4.7 Architecture

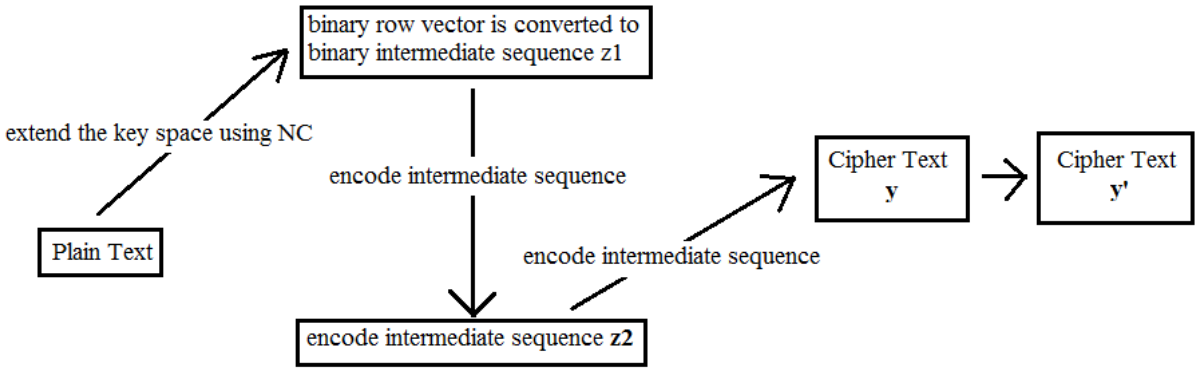


Figure 3. Architecture diagram

4.8 Data Flow Diagram

4.8.1 DFD Admin Session

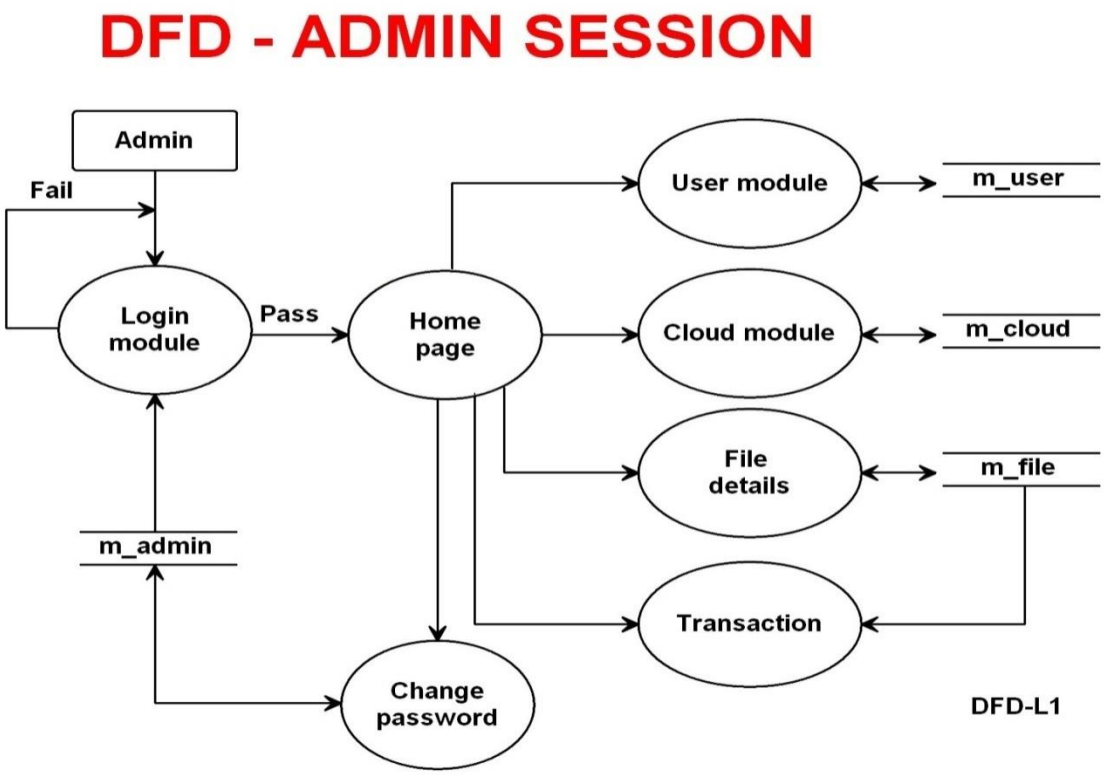


Figure 4.8.1 DFD Admin Session Diagram

4.8.2DFD User Session

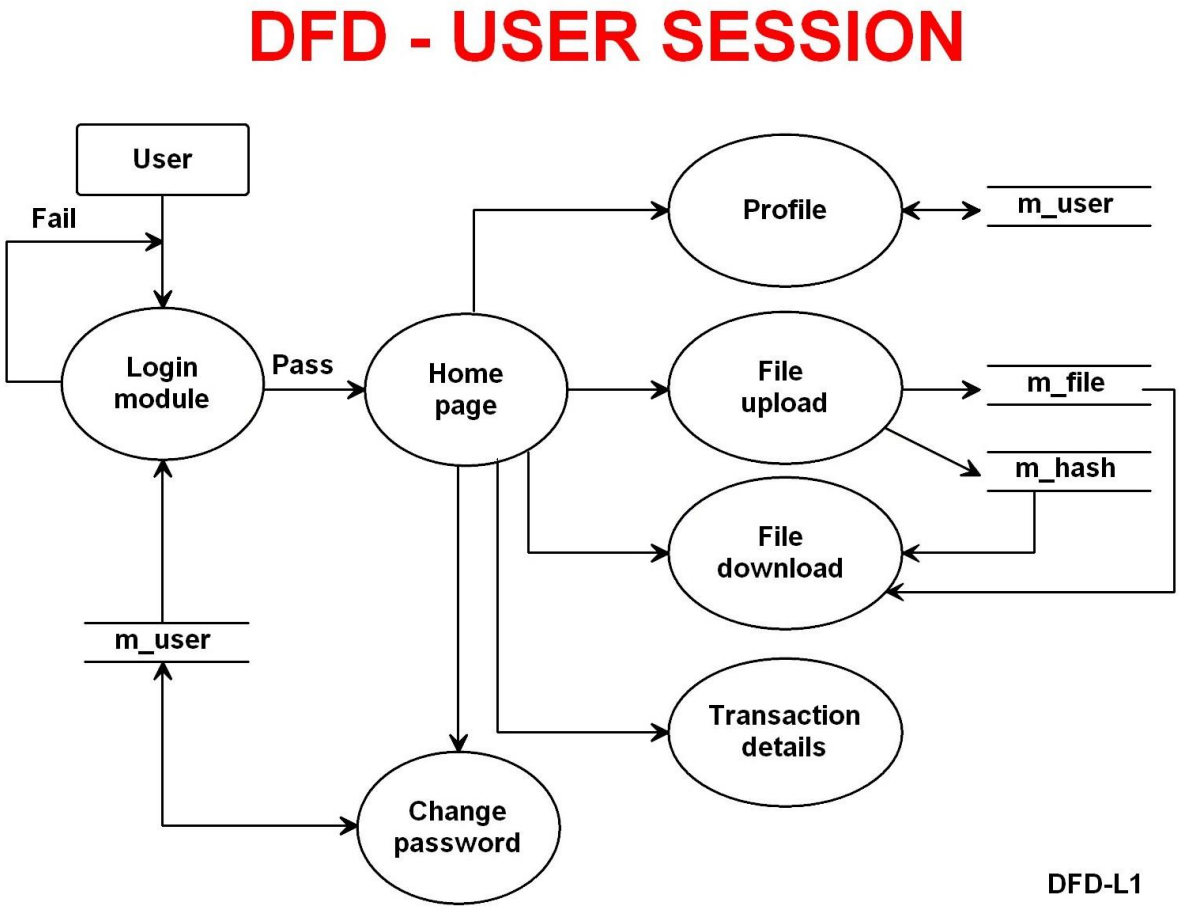


Figure 4.8.2 DFD User Session Diagram

4.8 TECHNIQUE AND ALGORITHMS

Algorithm

- **AES (Advanced Encryption Standard) Encryption algorithm.**
- **AES (Advanced Encryption Standard) Decryption algorithm.**

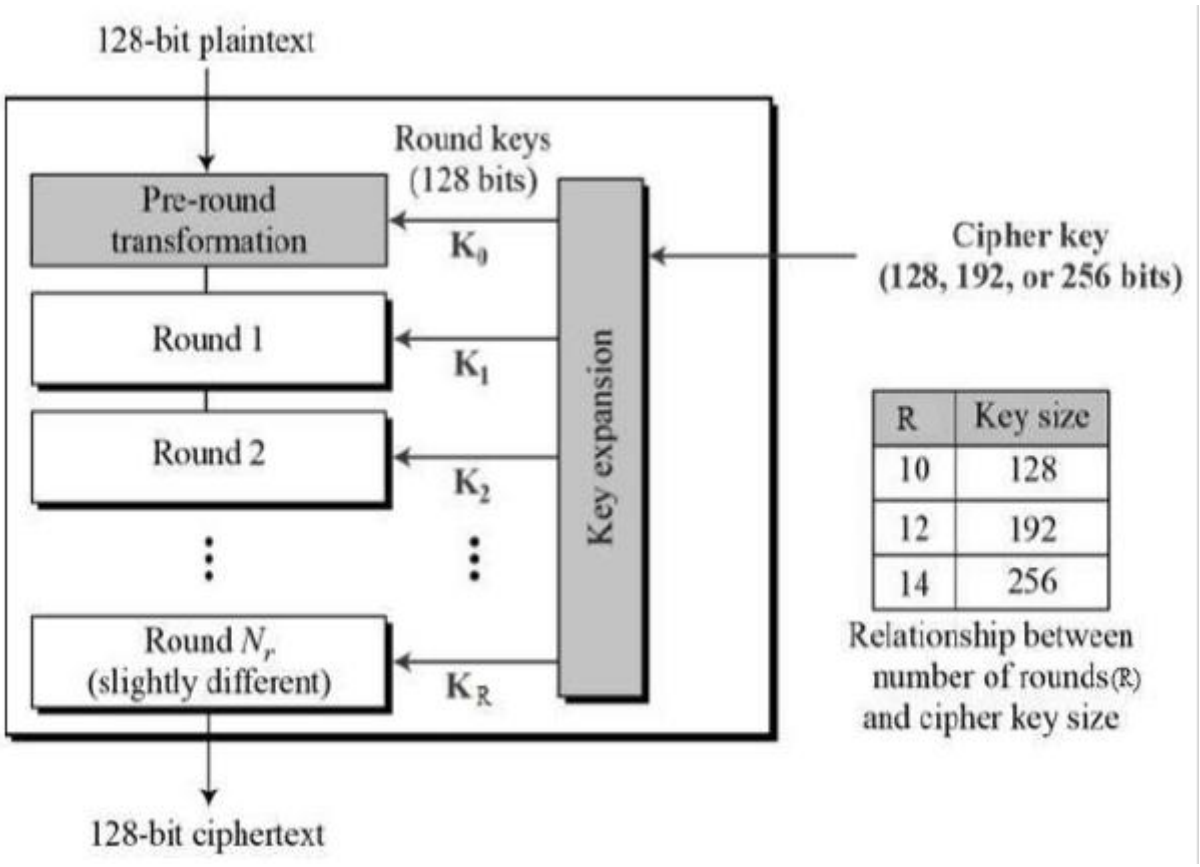
AES (Advanced Encryption Standard) Encryption algorithm

AES is an iterative rather than Feistel cipher. It is based on 'substitution–permutation network'. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations).

Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix −

Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

The schematic of AES structure is given in the following illustration –

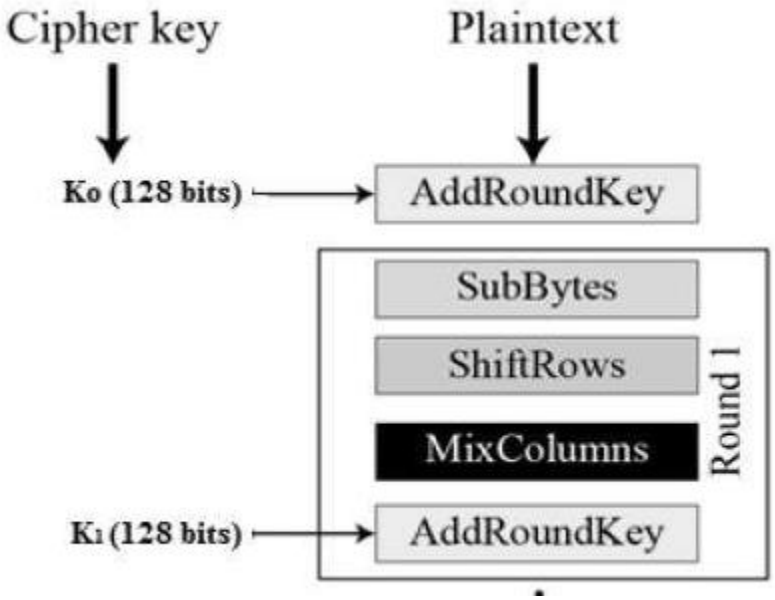


R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

Encryption Process

Here, we restrict to description of a typical round of AES encryption. Each round comprise of four sub-processes. The first round process is depicted below –



Byte Substitution (SubBytes)

The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

Shiftrows

Each of the four rows of the matrix is shifted to the left. Any entries that 'fall off' are re-inserted on the right side of row. Shift is carried out as follows −

- First row is not shifted.
- Second row is shifted one (byte) position to the left.
- Third row is shifted two positions to the left.
- Fourth row is shifted three positions to the left.
- The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

MixColumns

Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes,

which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

Addroundkey

The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

Decryption Process

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order −

- Add round key
- Mix columns
- Shift rows
- Byte substitution

Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms need to be separately implemented, although they are very closely related.

4.9 System Design

4.9.1 The activities of the Design process:

1. Interface outline portrays the structure and association of the UI. Incorporates a portrayal of screen format, a meaning of the methods of collaboration, and a depiction of route components. Interface Control instruments to actualize route choices, the originator chooses frame one of various connection system;

a. Navigation menus

b. Graphic symbols

c. Graphic pictures

Interface Design work process the work process starts with the ID of client, undertaking, and natural necessities. When client errands have been distinguished, client situations are made and broke down to characterize an arrangement of interface protests and activities.

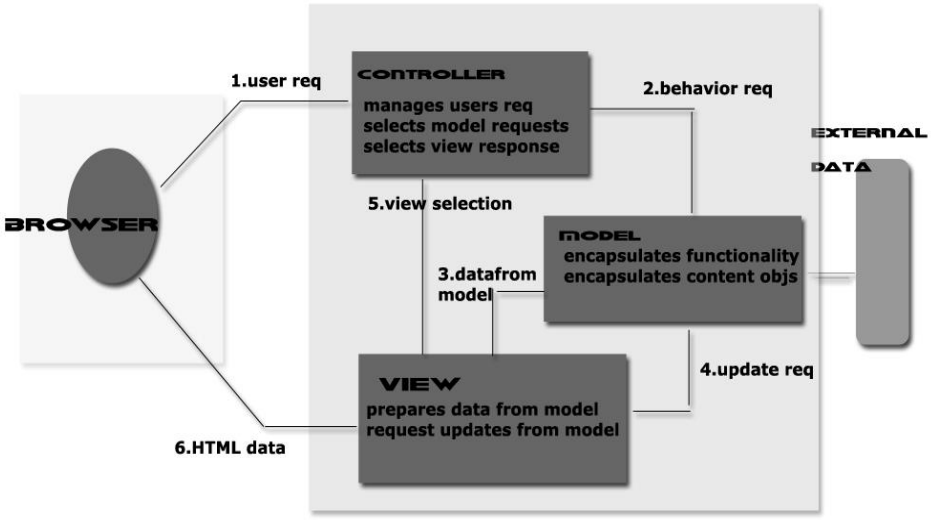
2. Aesthetic outline likewise called visual communication portrays the "look and feel" of the WebApp. Incorporates shading plans, geometric design.Content size, textual style and position, the utilization of designs, and related stylish choices.

3. Content plan characterizes the design, structure, and blueprint for all substance that is exhibited as a component of the WebApp. Builds up the connections between content items.

4. Navigation outline speaks to the navigational stream between substance objects and for all WebApp capacities.

5. Architecture outline distinguishes the general hypermedia structure for the WebApp. Engineering configuration is attached to the objectives set up for a WebApp, the substance to be exhibited, the clients who will visit, and the route logic that has been built up.

- Content engineering, centers around the way in which content protests and organized for introduction and route.
- WebApp design, addresses the way in which the application is structure to oversee client communication, handle inner preparing errands, impact route, and present substance. WebApp design is characterized inside the setting of the advancement condition in which the application is to be actualized.



J2EE uses MVC Architecture

6. Component design-develops the detailed processing logic required to implement functional components.

Chapter 5

RESULT

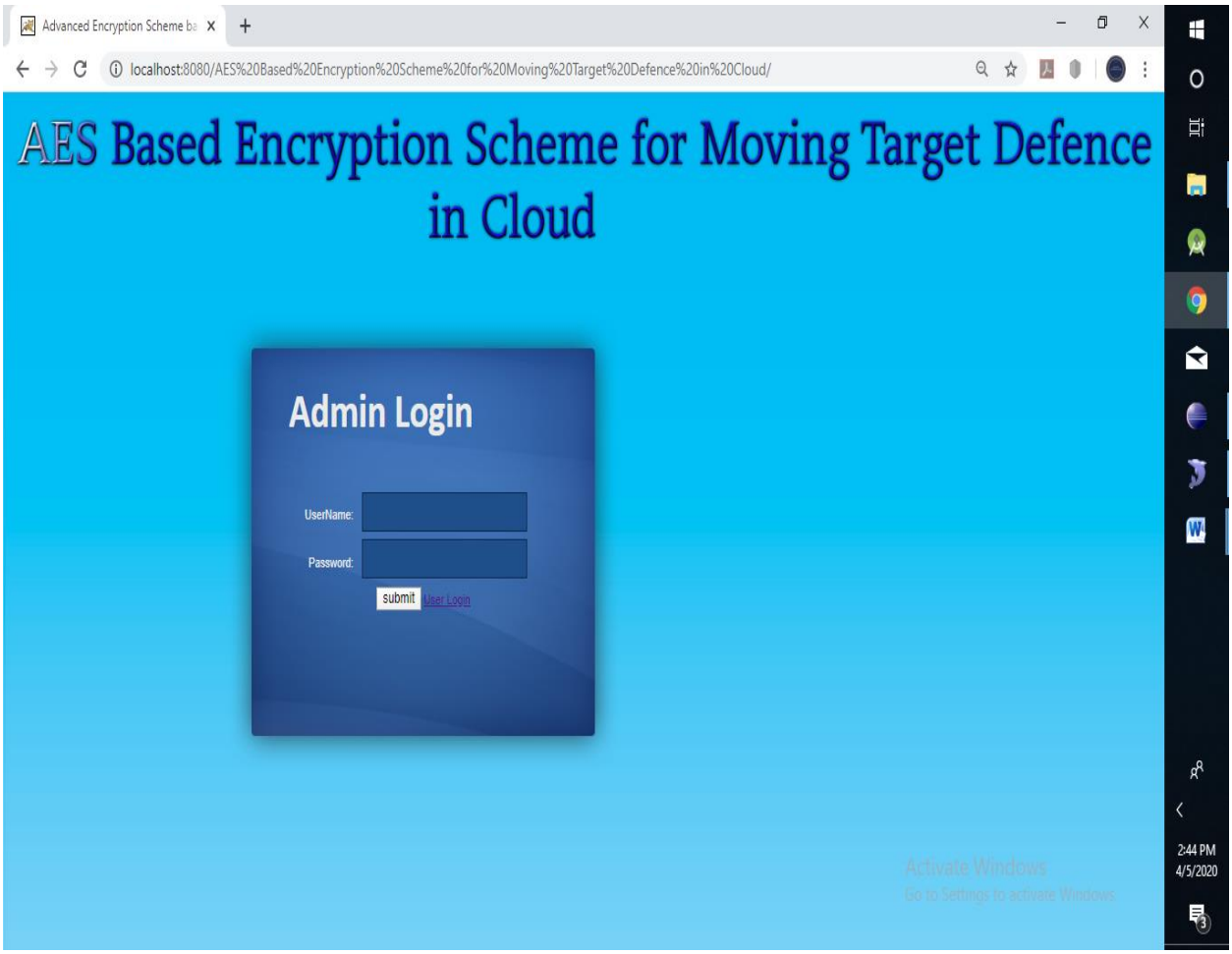
In this proposed work we have developed Web application. For implemented we developed a web page to register the user, data owner and admin. We created a method where user can share files to other users. We have designed a page in which user can simply enter the id of person whom to transfer the files and file gets uploaded to cloud server and name of the files get saved to MYSQL database table. When user want to download the file he must send a request to data owner and then data owner may give permission to download by sending a key mail to the requested user. RNS and DES algorithm is used for Encryption.

Table 1. Data Security Enhancement

SN	Existing System	Proposed System
1	No Encryption techniques is used	Proposed system provides high security for data.
2	Directly uploading data to cloud storage. For example Google drive	Before uploading data to cloud, data get encrypted, with AES encryption.
3	In cloud storage all get stored in single place / file / folder	Proposed system provides distributed storage in cloud. Data get stored in different folder / place / file

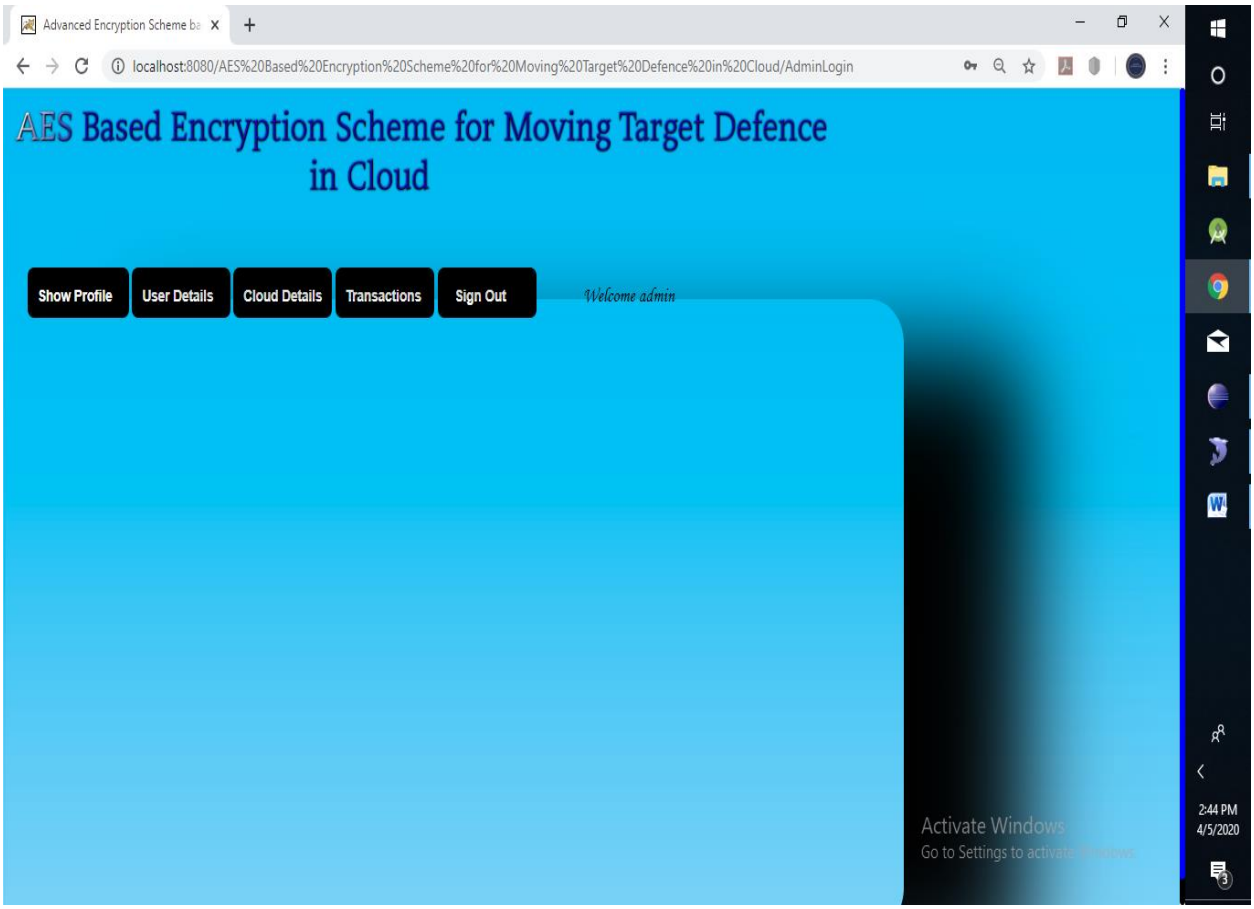
Theaccompanying depictions layout the outcomes or yields that we are going to get once regulated execution of the considerable number of modules of the framework.

Screenshot Results



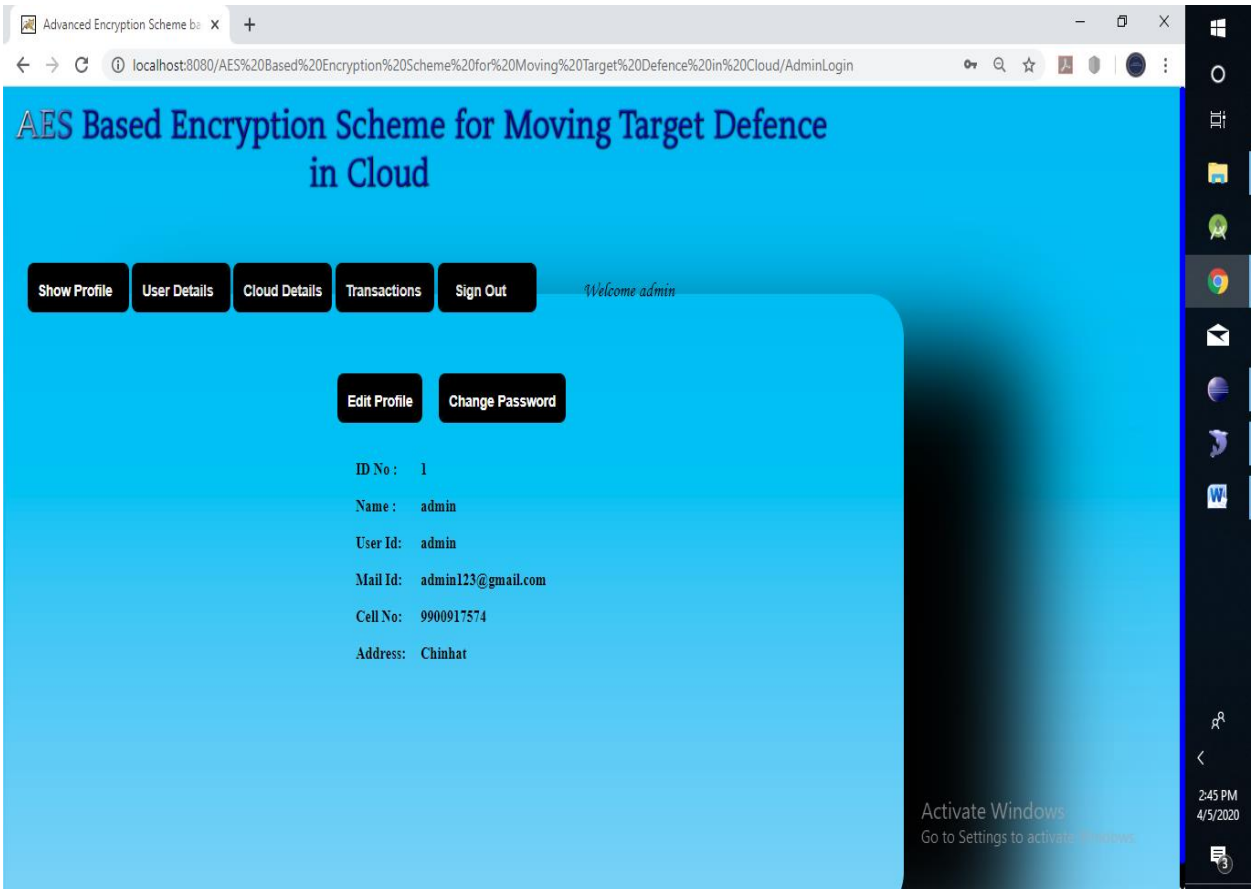
Screenshot-1 Login Page

The screenshot-1 shows the admin login page.



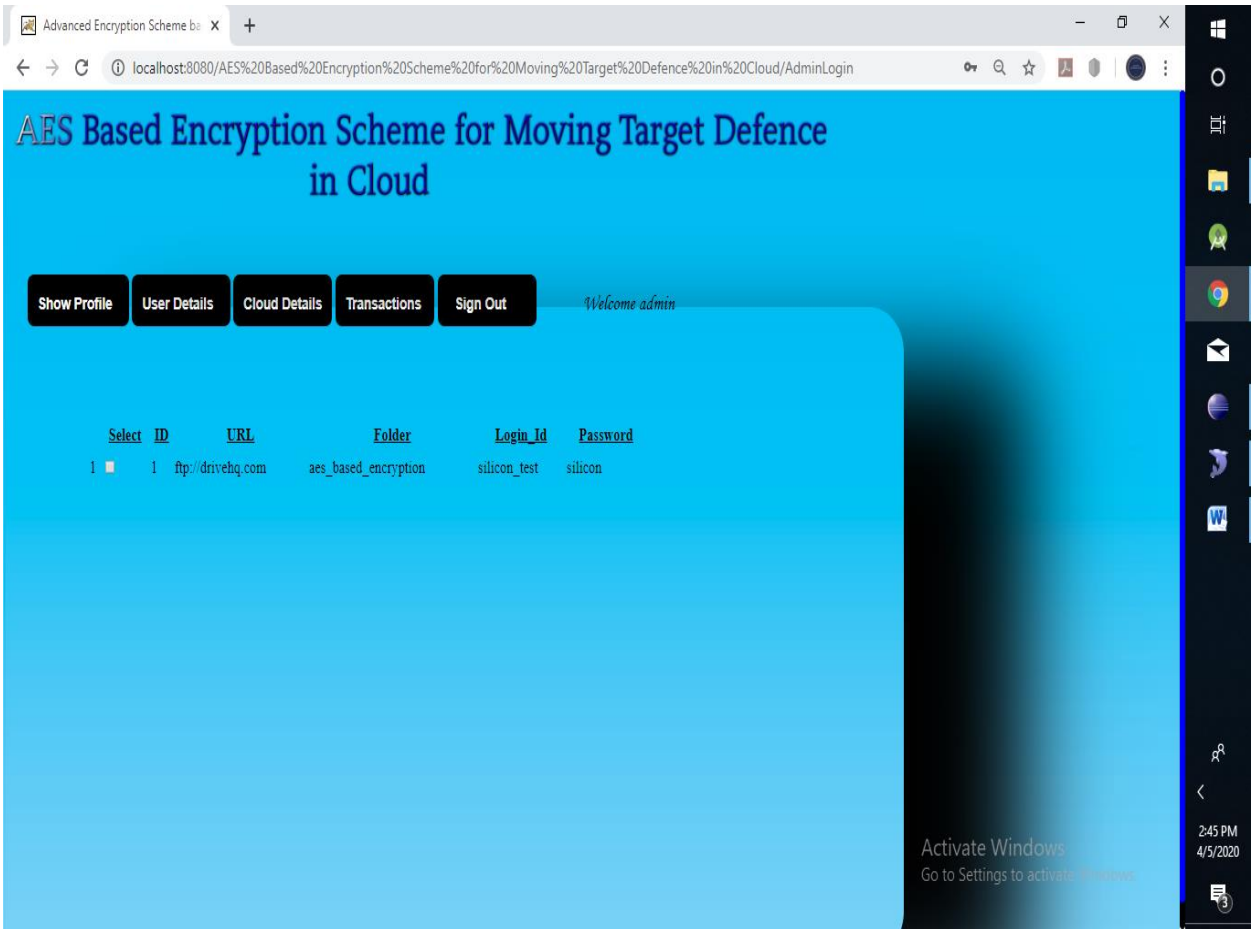
Screenshot-2 Admin Home Page

The screenshot-2 shows the admin home page.



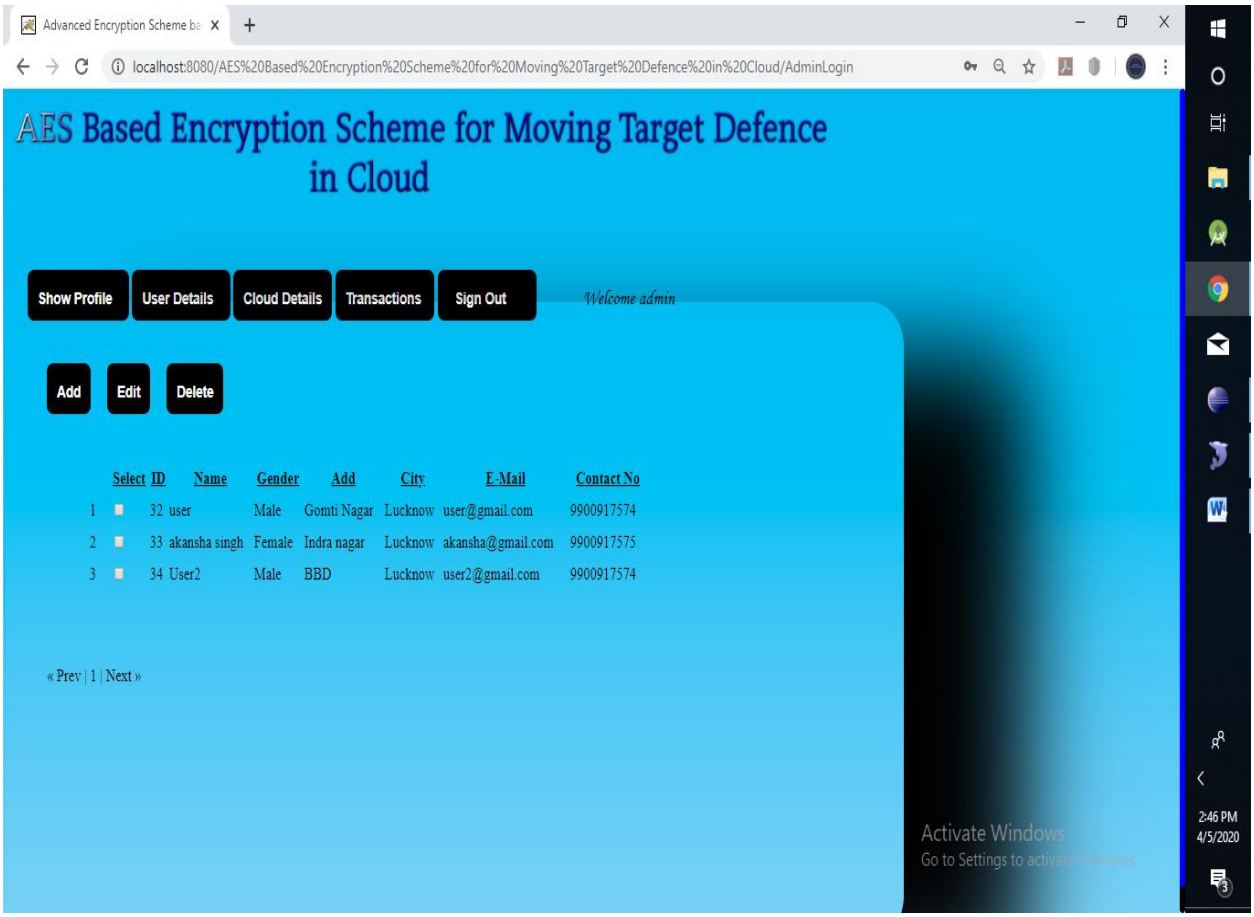
Screenshot-3 Admin Profile Page

The screenshot-3 shows the admin profile page details.



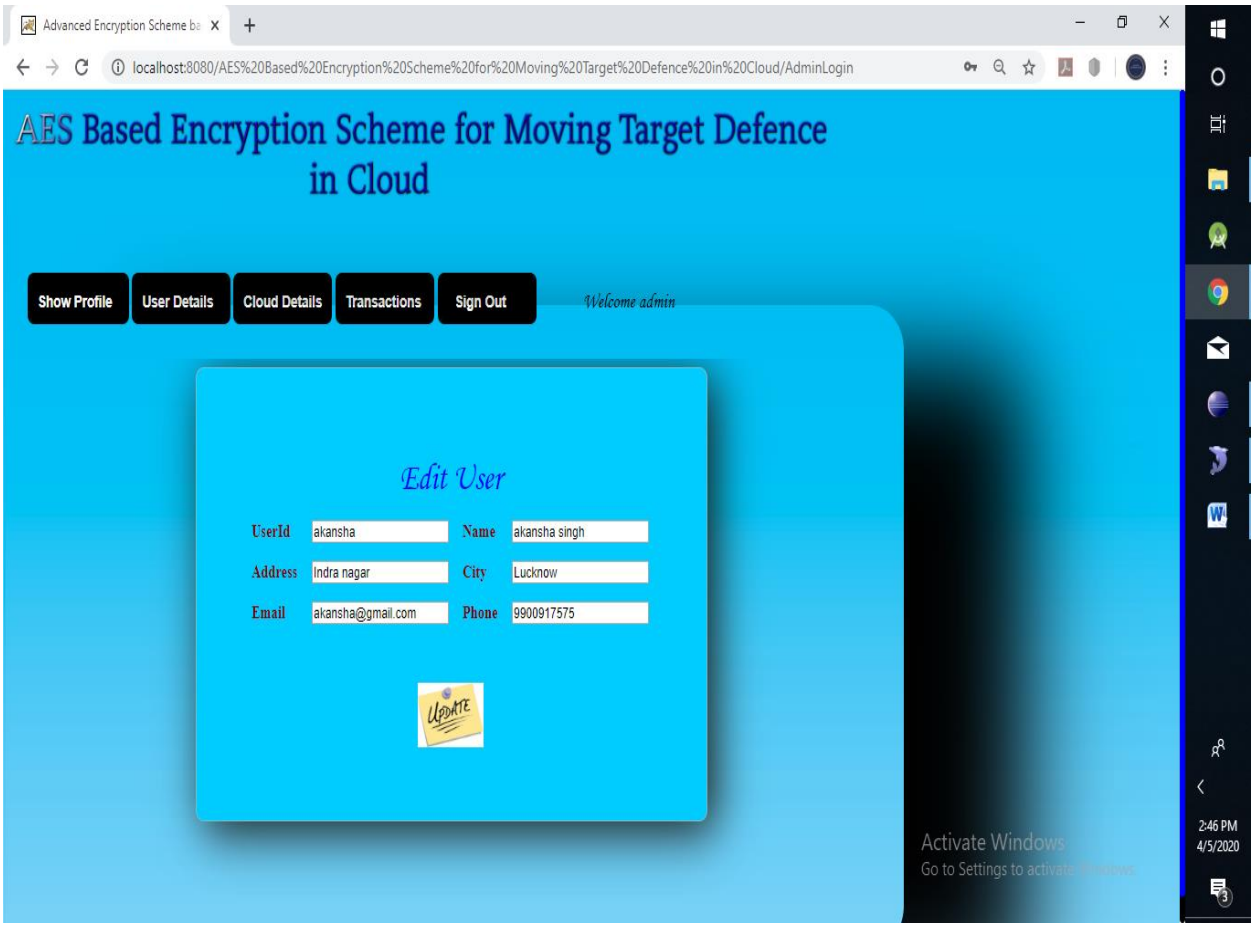
Screenshot-4 Cloud Server Details

The screenshot-4 shows the cloud server details



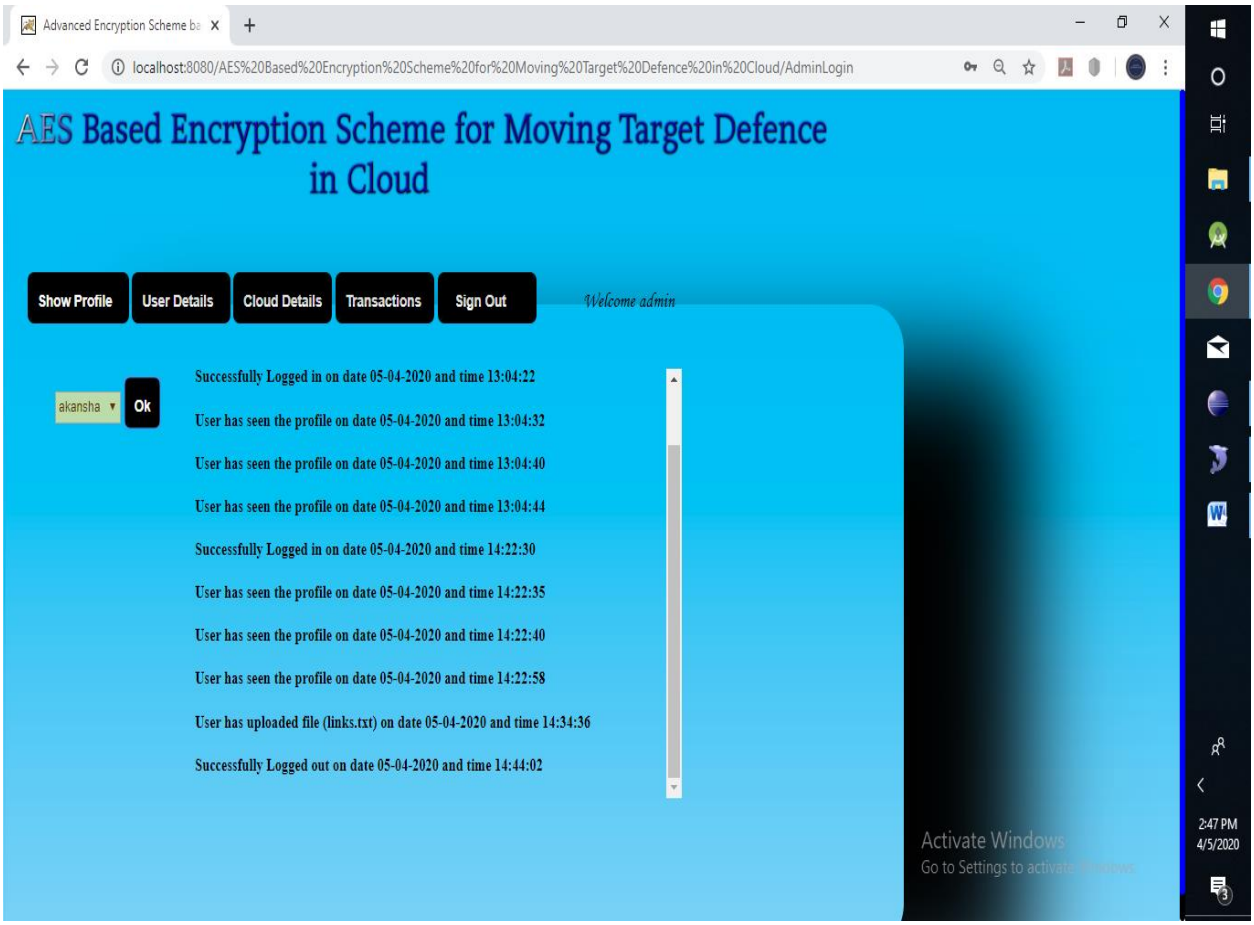
Screenshot-5 User Details

The screenshot-5 shows the User details.



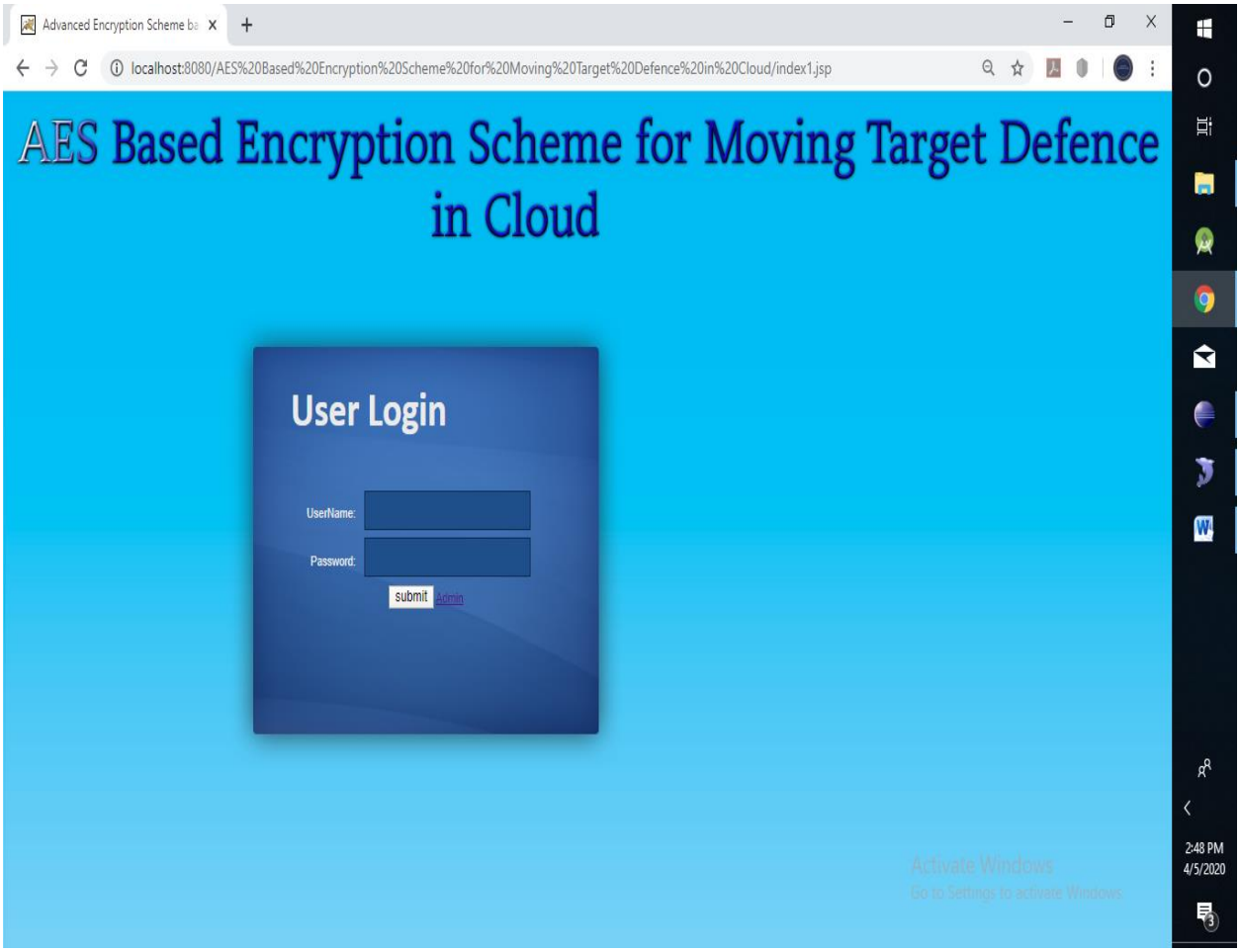
Screenshot-6 Edit User Details

The screenshot-6 shows the edit user details



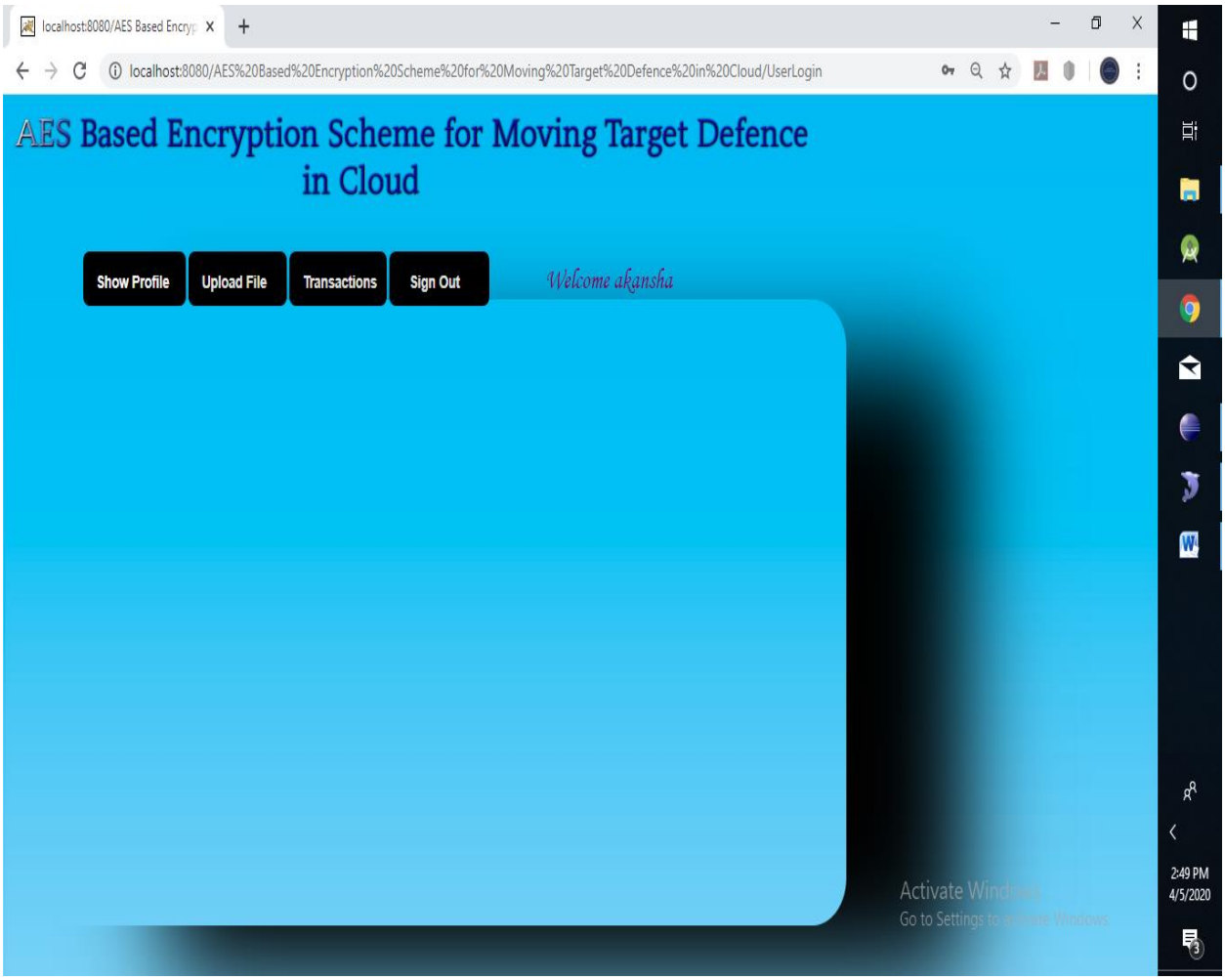
Screenshot-7 Transaction details

The screenshot-7shows the transaction details.



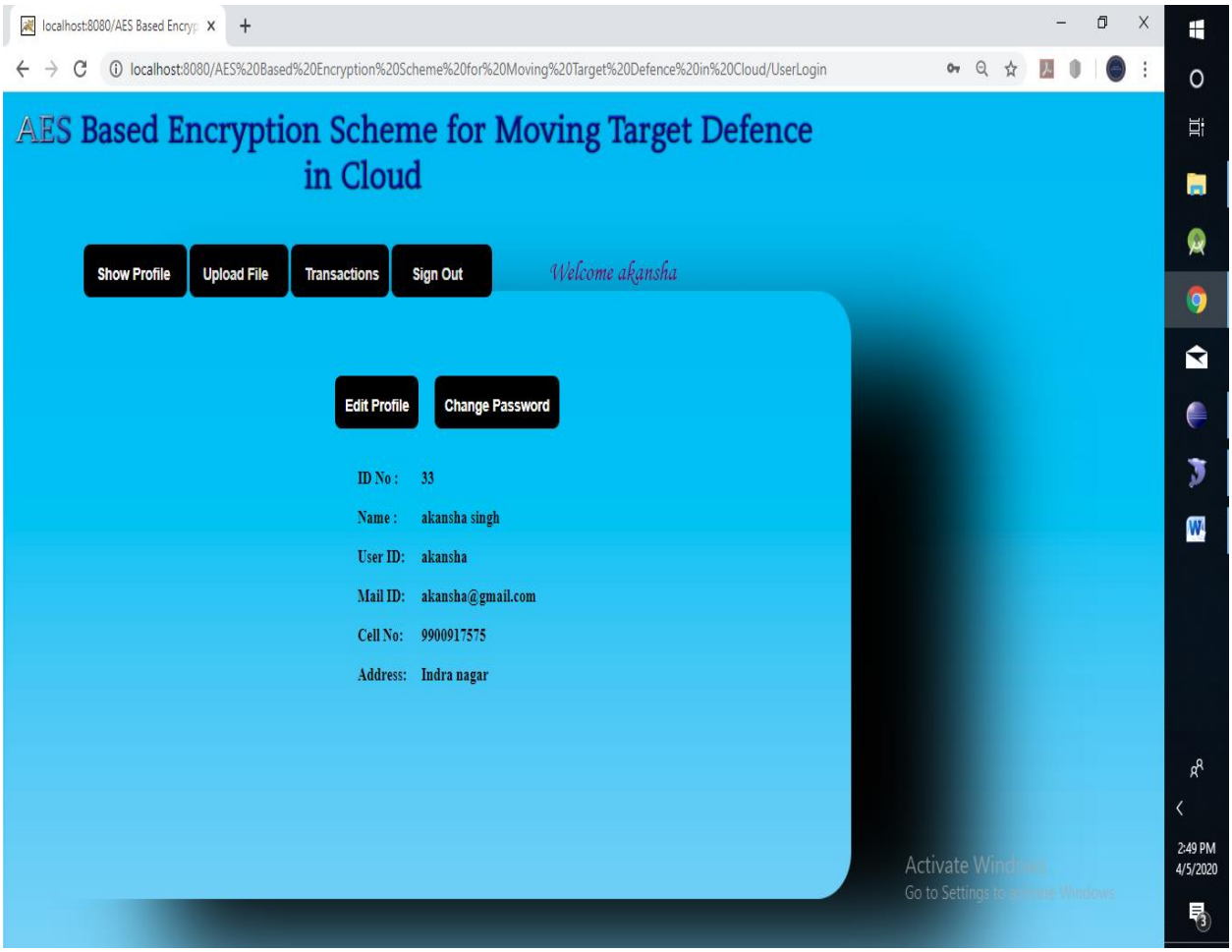
Screenshot-8 User Login

The screenshot-8 shows the User Login Page.



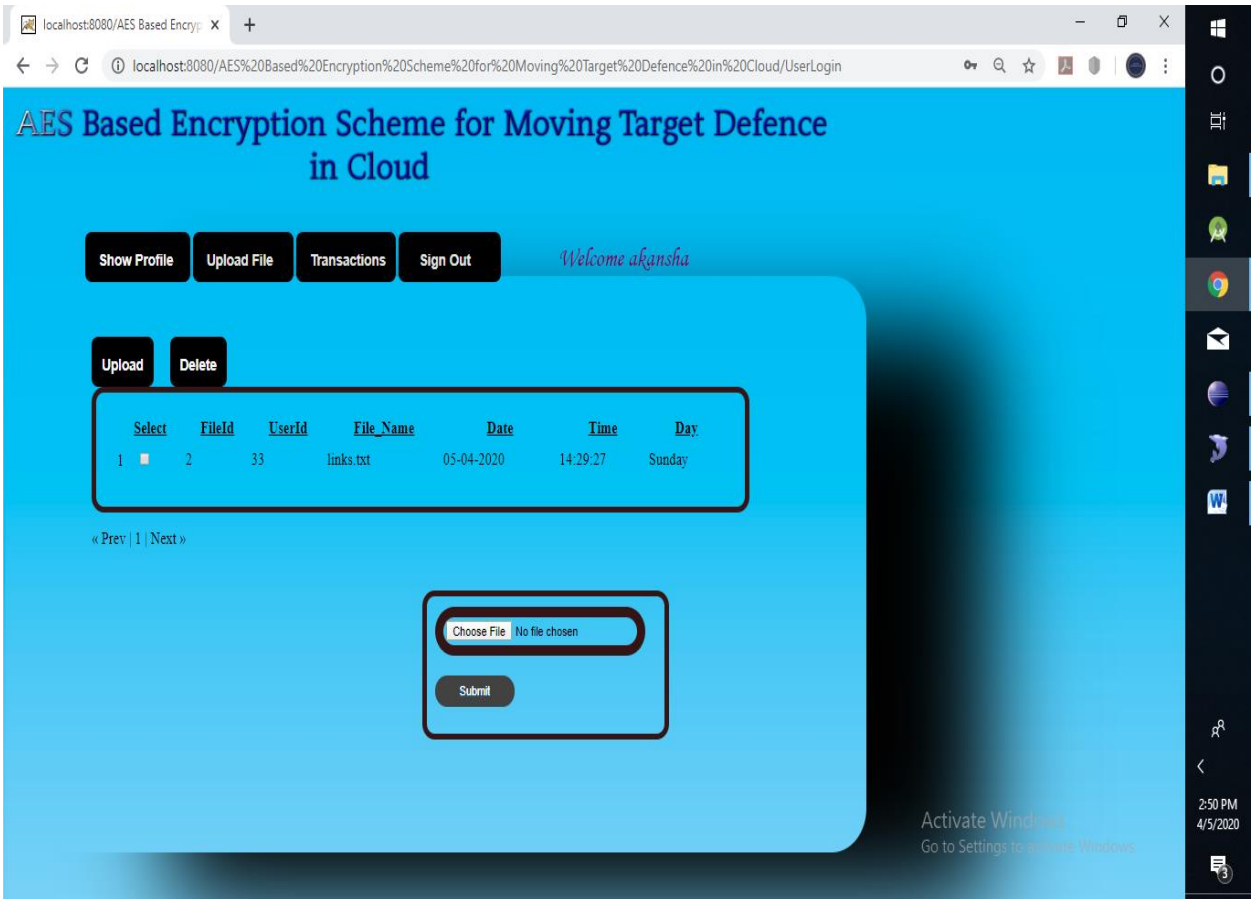
Screenshot-9 User Home

The screenshot-9 shows the User Home Page.



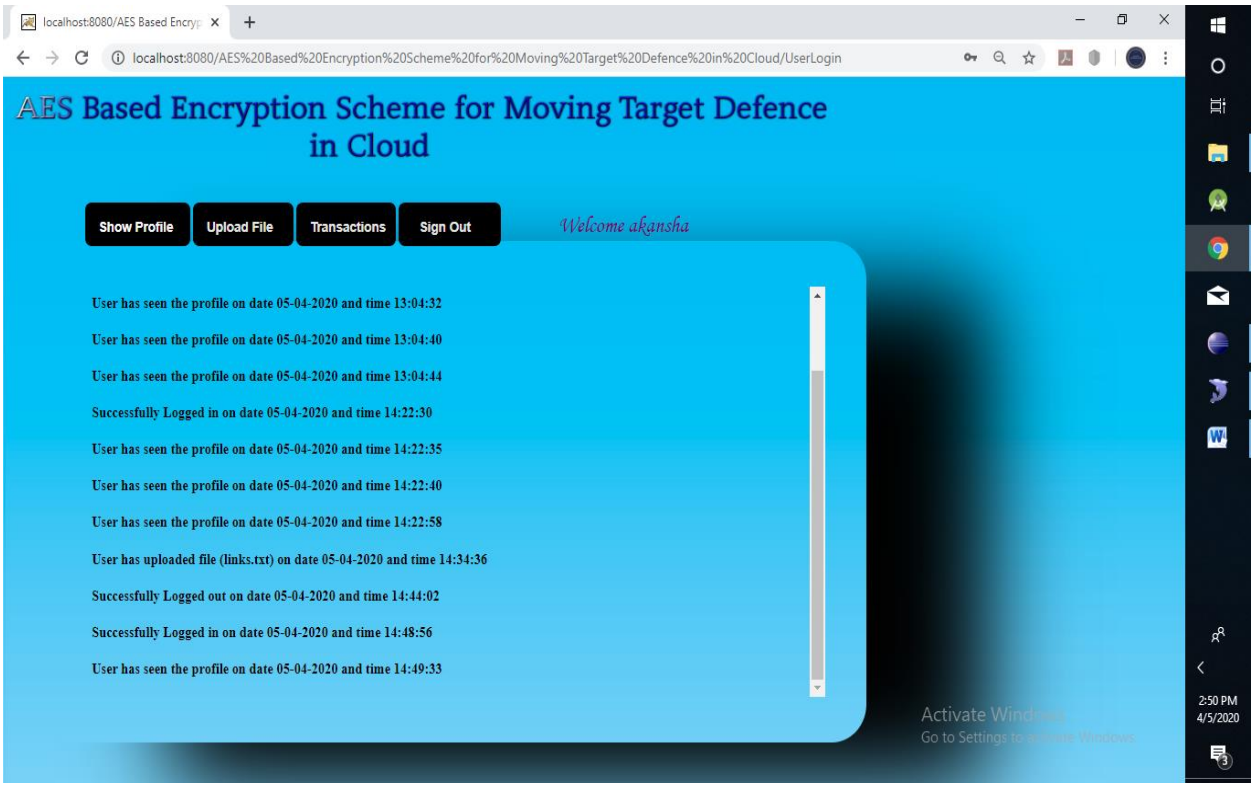
Screenshot-10 User Profile

The screenshot-10 shows the User Profile Page.



Screenshot-11 Upload file

The screenshot-11 shows the Upload File Page.



Screenshot-12 User Transaction Details

The screenshot-12 shows the user transaction details Page.

CONCLUSION AND FUTURE SCOPE

In this thesis, we proposed a novel encryption scheme which combines both the AES and the network coding characteristic, which has good behavior to resist both exhaustive and analysis attacks. The simulation results show that the running ratio of the proposed scheme is relatively lower than or comparable to the AES. The NC nature of the proposed scheme makes it endow the dynamic, active and random characteristics in the concept of Moving Target Defense (MTD). The security level of the proposed scheme will be tested in our future work.In proposed system we discussed about cloud storage security issues and challenges. In future we will try to deploy this in other cloud based environment and the best can be chosen. In future standard can be developed for cloud storage security. We will try to find out problems related to existing security algorithms and implement better version of existing security algorithms.

REFERENCES

[1] Lombardi F, Di Pietro R. Secure virtualization for cloud computing. Journal of Network Computer Applications (2010), doi:10.1016/j.jnca.2010.06.008.

[2] Subashini S, Kavitha V., "A survey on security issues in service delivery models of cloud computing," Journal of Network and Computer Applications (2011) vol. 34 Issue 1, January 2011 pp. 1-11.

[3] Sudha.M, Bandaru Rama Krishna rao, M.Monica, "A Comprehensive approach to ensure secure data communication in cloud environment" International Jornal Of computer Applications, vol. 12. Issue 8, pp. 19-23.

[4] Balachander R.K, Ramakrishna P, A. Rakshit, "Cloud Security Issues, IEEE International Conference on Services Computing (2010)," pp. 517-520.

[5] Cong Wang, Qian Wang, KuiRen, and Wenjing Lou, "Ensuring Data Storage Security in Cloud Computing" proceeding of International workshop on Quality of service 2009", pp.1-9.

[6] Gary Anthes, "Security in the cloud," In ACM Communications (2010), vol.53, Issue11, pp. 16-18.

[7] KresimirPopovic, ŽeljkoHocenski, "Cloud computing security issues and challenges," MIPRO 2010, pp. 344-349.

[8] KikukoKamiasaka, Saneyasu Yamaguchi, Masato Oguchi, "Implementation and Evaluation of secure and optimized IP-SAN Mechanism," Proceedings of the IEEE International Conference on Telecommunications, May 2007, pp. 272-277.

[9] Luis M. Vaquero, Luis Rodero-Merino, Juan Caceres1, Maik Lindner, "A Break in Clouds: Towards a cloud Definition," ACM SIGCOMM Computer Communication Review, vol. 39, Number 1, January 2009, pp. 50-55.

[10] Patrick McDaniel, Sean W. Smith, "Outlook: Cloudy with a chance of security challenges and improvements," IEEE Computer and reliability societies (2010), pp. 77-80.

[11] SameeraAbdulrahmanAlmulla, Chan YeobYeun, "Cloud Computing Security Management," Engineering systems management and its applications (2010), pp. 1-7.

[12] Steve Mansfield-Devine, "Danger in Clouds", Network Security (2008), 12, pp. 9-11.

[13] Anthony T. Velte, Toby J.Velte, Robert Elsenpeter, Cloud Computing: A Practical Approach, Tata McGrawHill 2010.

[14] Lizhe Wang, Jie Tao, Kunze M., Castellanos A.C., Kramer D., Karl W., "Scientific Cloud Computing: Early Definition and Experience," 10th IEEE Int. Conference on High Performance Computing and Communications, pp. 825-830, Dalian, China, Sep. 2008, ISBN: 978-0-7695-3352-0.

[15] Pring et al., "Forecast: Sizing the cloud; understanding the opportunities in cloud services," Gartner Inc., Tech. Rep. G00166525, March 2009.

[16] AmanBakshi, Yogesh B. Dujodwala, "Securing cloud from DDoS Attacks using Intrusion Detection System in Virtual Machine," ICCSN "10 Proceeding of the 2010 Second International Conference on Communication Software and networks, pp. 260-264, 2010, IEEE Computer Society, USA, 2010. ISBN: 978-0-7695-3961-4.

[17] H. KAMAL IDRISSI, A. KARTIT, M. EL MARRAKI FOREMOST SECURITY APPREHENSIONS IN CLOUD COMPUTINGJournal of Theoretical and Applied Information Technology 31 st January 2014. Vol. 59 No.3

[18] Kuyoro S. O, Ibikunle F. &Awodele O Cloud Computing Security Issues and ChallengesInternational Journal of Computer Networks (IJCN), Volume (3) : Issue (5) : 2011

[19] Chunye Gong, Jie Liu, Qiang Zhang, Haitao Chen and ZhenghuGongngThe Characteristics of Cloud Computing2010 39th International Conference on Parallel Processing Workshopse Brazilian Computer Society 2010

[20] SO, Kuyoro. Cloud computing security issues andchallenges. International Journal of Computer Networks, 2011, vol. 3, no 5.

[21] D. H. Rakesh, R. R. Bhavsar, and A. S. Thorve, "Data security over cloud," International Journal of Computer Applications, no. 5, pp. 11-14, 2012.

[22] J. Krumm, "A survey of computational location privacy," Personal and Ubiquitous Computing, vol. 13, no. 6, pp. 391-399, 2009.

[23] K. Hwang, S Kulkarni and Y. Hu, "Cloud security with virtualized defence and Reputation-based Trust management," Proceedings of 2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing (security in cloud computing), pp. 621-628, Chengdu, China, December, 2009. ISBN: 978-0-7695- 3929 -4.

[24] Marios D. Dikaiakos, DimitriosKatsaros, PankajMehra, George Pallis, Athena Vakali, "Cloud Computing: Distributed Internet Computing for IT and Scientific Research," IEEE Internet Computing Journal, vol. 13, issue. 5, pp. 10-13, September 2009. DOI: 10.1109/MIC.2009.103.

[25] AL.Jeeva, Dr.V.PalanisamyAndK.Kanagaram "Comparative Analysis Of Performance Efficiency And Security Measures Of Some Encryption Algorithms" International Journal Of Engineering Research And Applications (IJERA) ISSN: 2248-9622 Vol. 2, Issue 3, pp.3033- 3037, May-Jun 2012.

[26] Maha TEBAA, Saïd EL HAJJI, Abdellatif EL GHAZI, "Homomorphic Encryption Applied to the Cloud Computing Security", World Congress on Engineering, Volume I, ISBN: 978-988-19251-3-8; ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online) , 2012.

[27] Pratap Chandra Mandal, „Superiority of Blowfish Algorithm", International Journal of Advanced Research in Computer Science and Software Engineering. September (2012) ISSN: 2277-128X Vol. 2, Issue 7.

[28] G. Devi and M. Pramod Kumar, „Cloud Computing: A CRM Service Based on a Separate Encryption and Decryption using Blowfish Algorithm", International Journal of Computer Trends and Technology. (2012) Vol. 3 Issue 4, ISSN: 2231-2803, pp.592-596.

[29] Neha Jain and GurpreetKaur „Implementing DES Algorithm in Cloud for Data Security" VSRD International Journal of CS & IT Vol. 2 Issue 4, pp. 316-321, 2012.

[30] G. Devi , M. Pramod Kumar, "Cloud Computing: A CRM Service Based on a Separate Encryption and Decryption using Blowfish algorithm" International Journal Of Computer Trends And Technology Volume 3 Issue 4, ISSN: 2231-2803, pp. 592-596,2012.

[31] D. S. Abdul. Elminaam, H. M. Abdul Kader and M. M. Hadhoud ,"Performance Evaluation of Symmetric Encryption Algorithms", Communications of the IBIMA Volume 8, 2009.

[32] Gurpreet Singh, SupriyaKinger "Integrating AES, DES, and 3-DES Encryption Algorithms for Enhanced Data Security" International Journal of Scientific & Engineering Research, Volume 4, Issue 7, July-2013.

[33] Mr. Gurjeevan Singh, , Mr. AshwaniSingla and Mr. K S Sandha " Cryptography Algorithm Compaison For Security Enhancement In Wireless Intrusion Detection System" International Journal of Multidisciplinary Research Vol.1 Issue 4, August 2011.

[34] Gurpreet Singh, SupriyaKinger"Integrating AES, DES, and 3-DES Encryption Algorithms for Enhanced Data Security "International Journal of Scientific & Engineering Research, Volume 4, Issue 7, July-2013.

PUBLISHED PAPER AND CERTIFICATE

Journal of Xidian University

An UGC-CARE Approved Group 2 Journal

ISSN NO: 1001-2400, Impact Factor : 5.4

http://xadzkjdx.cn/, Mail : editorjxu@gmail.com

CERTIFICATE ID : JXU-R2591

CERTIFICATE OF PUBLICATION

This is to certify that the paper entitled

AES Based Encryption Scheme for Moving Target Defense in Cloud Storage

Authored by

Akanksha Singh

From

BBD University, Uttar Pradesh

Has been published in

VOLUME 14, ISSUE 4, 2020.

Jenny Corbett

JXU JOURNAL

AES Based Encryption Scheme for Moving Target Defense in Cloud Storage

Akanksha Singh[#1], Abhinav Singh[*2]

[#1]*Dept. of Computer Science & Engineering, BBD University, Uttar Pradesh, India*

[*2]*Assistant Prof. of Dept. of Computer Science & Engineering, BBD University, Uttar Pradesh, India*

[1]akankshasingh@gmail.com

[2]abhinav@bbdu.org

Abstract— **Now days cloud computing become one of the main topic of IT and main point is cloud data storage security. Cloud is the fastest growing technology. This technology provides access to many different applications. Cloud computing is used as data storage so data security and privacy issues such as confidentiality, availability and integrity are important factor associated with it. Cloud storage provides user to access remotely store their data so it becomes necessary to protect data from unauthorized access, hackers or any type of modification and malicious behaviour. Security is an important concern. The meaning of data storage security is to secure data on storage media. Cloud storage does not require any hardware and software management. It provide high quality applications. As we proposed the concept of cloud data storage security strategy capable to overcome the shortcomings of traditional data protection algorithms and improving security using Advanced Encryption Standard (AES) encryption and decryption technique adoptable to better security for the cloud. We have developed a web application through which user can share data. This thesis enhanced advance security goal for cloud data storage.**

Keywords— **Cloud data, Security, encryption, decryption, privacy.**

I. INTRODUCTION

The National Institute of Standards and Technology (NIST) define cloud computing as "a model for user convenience, on-demand network access contributes the computing resources (e.g. network, storage, application, servers and services) that can be rapidly implemented with minimal management effort or service provider interference" [5]. The users can access the cloud data and application at anytime and anywhere. The cloud contains large number of servers required to deliver scalable and reliable on-demand services [5]. Cloud Computing is an emerging information technology that change the way of IT architectural solution. It is a new pattern It is a new pattern of business computing. Computing is refers to manipulating Cloud, Configuring and Accessing the Applications online[2]. It offers the online data storage, Infrastructure and applications. It overcomes the Platform dependency issues because it need not to install the software on our local PC. Cloud computing provides information resources for users in "CLOUD" through the Internet [1][2]. As we proposed the concept of cloud data storage security strategy capable to overcome the shortcomings of traditional data protection algorithms and improving security using encryption decryption techniques DDES(Double Data Encryption standard) and RNS(Random Number System) are adoptable to better security for the cloud. We have developed a desktop application through which user can share data. This paper enhanced advance data security and user authorization in cloud.

Code

Admin Login

```
<%@pageimport="com.util.*"%>


<html>

<head>

<title>Advanced Encryption Scheme based on AES for Moving Target
Defense</title>

<%

//int no=Utility.parse(request.getParameter("no"));

%>

<linkhref="<%=request.getContextPath()%>/Files/CSS/style.css"

      rel="stylesheet"type="text/css"/>

<linkhref="<%=request.getContextPath()%>/Files/CSS/message.css"

      rel="stylesheet"type="text/css"/>

<linkrel="stylesheet"

      href="<%=request.getContextPath()%>/Files/CSS/login.css"

      type="text/css"/>

<scripttype="text/javascript"

      src="<%=request.getContextPath()%>/Files/JS/style.js"></script>




<linkhref="login-box.css"rel="stylesheet"type="text/css"/>

<linkhref="<%=request.getContextPath()%>/Files/CSS/styles.css"rel="stylesheet
"type="text/css"/>

<linkhref="<%=request.getContextPath()%>/Files/CSS/logins.css"rel="stylesheet
"type="text/css"/>
```

```
<linkhref="<%=request.getContextPath()%>/Files/CSS/popup.css"rel="stylesheet"
type="text/css"/>

<scriptsrc="<%=request.getContextPath()%>/Files/JS/jquery-
1.6.4.min.js"type="text/javascript"></script>









</head>


<bodyonload="startTimer()">

<div  style="position: centre;  left: -10px;"   >

      <imgsrc="<%=request.getContextPath()%>/Files/Images/tt.png"width=100%><
/img>

</div>


<divstyle="padding: 20px 0 0 250px;">


<formclass="login"action="<%=request.getContextPath()%>/AdminLogin"

                        method="post">

<divid="login-box">


<H2>Admin Login</H2>

<br/>

<br/>

<divid="login-box-name"style="margin-top:20px;">UserName:</div><divid="login-
box-field"style="margin-top:20px;"><inputname="name"class="form-
login"title="Username"value=""size="30"maxlength="2048"/></div>

<divid="login-box-name">Password:</div><divid="login-box-
field"><inputname="pass"type="password"class="form-
login"title="Pass"value=""size="30"maxlength="2048"/></div>
```

```
<br/>


<br/>

<br/>

<center>

<inputtype="submit"name="submit"value="submit"/>


<ahref="<%=request.getContextPath() %>/index1.jsp"> User Login </a>

</center>




</div>

</form>


</div>




            <%

            intnoo=Utility.parse(request.getParameter("no"));

            if(noo==1)

{

      %>


                  <divclass="error"id="message"style="height: 65; width:
250px; top:180" >

                  <p>Opp's,your Id or password is wrong ..!</p>

            </div>
```

```
	<%
}
		%>
</body>
</html>
```

User Login code

```
<%@pageimport="com.util.*"%>

<html>
<head>
<title>Advanced Encryption Scheme based on AES for Moving Target
Defense</title>
<%
//int no=Utility.parse(request.getParameter("no"));
%>
<linkhref="<%=request.getContextPath()%>/Files/CSS/style.css"
	rel="stylesheet"type="text/css"/>
<linkhref="<%=request.getContextPath()%>/Files/CSS/message.css"
	rel="stylesheet"type="text/css"/>
<linkrel="stylesheet"
	href="<%=request.getContextPath()%>/Files/CSS/login.css"
	type="text/css"/>
<scripttype="text/javascript"
	src="<%=request.getContextPath()%>/Files/JS/style.js"></script>
```

```
<linkhref="login-box.css"rel="stylesheet"type="text/css"/>

<linkhref="<%=request.getContextPath()%>/Files/CSS/styles.css"rel="stylesheet
"type="text/css"/>

<linkhref="<%=request.getContextPath()%>/Files/CSS/logins.css"rel="stylesheet
"type="text/css"/>

<linkhref="<%=request.getContextPath()%>/Files/CSS/popup.css"rel="stylesheet"
type="text/css"/>

<scriptsrc="<%=request.getContextPath()%>/Files/JS/jquery-
1.6.4.min.js"type="text/javascript"></script>







</head>


<bodyonload="startTimer()">

<div  style="position: centre;  left: -10px;"   >

      <imgsrc="<%=request.getContextPath()%>/Files/Images/tt.png"width=100%><
/img>

</div>


<divstyle="padding: 20px 0 0 250px;">


<formclass="login"action="<%=request.getContextPath()%>/UserLogin"

                        method="post">

<divid="login-box">


<H2>User Login</H2>

<br/>
```

```
<br/>

<divid="login-box-name"style="margin-top:20px;">UserName:</div><divid="login-
box-field"style="margin-top:20px;"><inputname="name"class="form-
login"title="Username"value=""size="30"maxlength="2048"/></div>

<divid="login-box-name">Password:</div><divid="login-box-
field"><inputname="pass"type="password"class="form-
login"title="Pass"value=""size="30"maxlength="2048"/></div>

<br/>

<%-- <span class="login-box-options"><input type="checkbox" name="1"
value="1"> Remember Me <a href="#" style="margin-left:30px;">Forgot
password?</a></span> --%>

<br/>

<br/>

<center>

<inputtype="submit"name="submit"value="submit"/>


<ahref="<%=request.getContextPath() %>/index.jsp">Admin  </a>

</center>







</div>

</form>







src="<%=request.getContextPath() %>/Files/Images/close.png" height="50"

                    width="50"/></a>
```

```
                <form class="login"
action="<%=request.getContextPath()%>/AuditorLogin"
                        method="post">

                        <center>
                              <label><font
                                    style="font-family: MonotypeCorsiva;
font-size: 35px; color: #DAA520;">Auditor's
                                          Login</font></label>
                        </center>
                        <br><label><font style="color: #336633;">Auditor's
Id</font></label>  
                        <input type="text" tabindex="1" class="input"
placeholder="Tpa Id"
                              name="name" required><br><br><label><font
                              style="color:
#336633;">Password</font></label><input type="password"
                              class="input" tabindex="2" name="pass"
placeholder="Password"
                              required><br><br>
                              <input
type="image" id=""
                              value="submit" src="<%=request.getContextPath()
%>/Files/Images/login.png" alt="submit Button"
                              class="gradientbuttons"
                              width="80" height="80">
                </form>
            </div></div>-->
```

```
            <%
            intnoo=Utility.parse(request.getParameter("no"));
            if(noo==1)
{
      %>

                  <divclass="error"id="message"style="height: 65; width:
250px; top:180" >
                  <p>Opp's,your Id or password is wrong ..!</p>
            </div>

      <%
}
            %>
</body>
</html>

/**
```

```
 *
 */
packagecom.action.user;

import java.io.IOException;
importjava.io.PrintWriter;
importjava.sql.ResultSet;

importjavax.servlet.RequestDispatcher;
importjavax.servlet.http.HttpServlet;
importjavax.servlet.http.HttpServletRequest;
importjavax.servlet.http.HttpServletResponse;

importcom.DAOFactory.CommonDAO;
importcom.DAOFactory.DAO;
importcom.DAOFactory.DAOFactory;
importcom.util.Utility;

public class UploadFile extends HttpServlet
{

        public void doGet(HttpServletRequestrequest,HttpServletResponse response)throws
IOException
        {
                PrintWriter out = response.getWriter();
                try
```

```
{
		String submit=request.getParameter("submit");
		String name=request.getParameter("name");
		//System.out.println("submit---"+submit);
		//System.out.println("name----"+name);
		boolean result=false;
		ResultSetrs=CommonDAO.getFiles(CommonDAO.getUserID(name),"uploaded");
		RequestDispatcherrd=null;
		if(submit.equals("get"))
		{
			if(rs.next())
			{
rs=CommonDAO.getFiles(CommonDAO.getUserID(name),"uploaded");
				request.setAttribute("rs", rs);
				request.setAttribute("name", name);
				rd=request.getRequestDispatcher("/Files/JSP/User/files.jsp");
				rd.forward(request, response);
			}
			else
			{
rd=request.getRequestDispatcher("/Files/JSP/User/files.jsp?no=-1");
				rd.forward(request, response);
			}
		}
```

```
else if(submit.equals("Upload"))
{
	rs=CommonDAO.getFiles(CommonDAO.getUserID(name),"uploaded");
	request.setAttribute("rs", rs);
	rd=request.getRequestDispatcher("/Files/JSP/User/files.jsp?no=1");
	rd.forward(request, response);
}
else if(submit.equals("Delete"))
{
	String []chk=request.getParameterValues("chk");
	request.setAttribute("name", name);
	if(chk==null)
	{
rs=CommonDAO.getFiles(CommonDAO.getUserID(name),"uploaded");
	request.setAttribute("rs", rs);
rd=request.getRequestDispatcher("/Files/JSP/User/files.jsp?no=3");
		rd.forward(request,response);
	}
	else
	{
		for(int i=0;i<chk.length;i++)
		{
			String fname=CommonDAO.getFileName(chk[i]);
```

```
String hashblocks=CommonDAO.gethashblknos(fname);

String[] temp;

 /* delimiter */
 String delimiter = "-";
 /* given string will be split by the argument delimiter provided. */
temp = hashblocks.split(delimiter);
 /* print substrings */
System.out.println("======== "+hashblocks);
 String blockname="";
System.out.println("-----Temp Size--"+temp.length);
for(int j =0; j <temp.length ; j++)
 {
        System.out.println("--===-======-= "+temp[j]);
          //int id2=Integer.parseInt(temp[i]);
        blockname=CommonDAO.getblocks(temp[j]);
        System.out.println("-----BLOCK NAME------ "+blockname);

        boolean flag=false;
        flag=CommonDAO.getInstance(temp[j]);
        if(flag==true)
          {

    result=Utility.deleteFile(Utility.getPro("server"),Utility.getPro("user"),Utility.getPro("pass"), blockname);
```

```
                                                result=CommonDAO.
Updatehashtable(temp[j]);
                                                    }
                                                }


                                            result=CommonDAO.UpdateTrans(chk[i], "uploaded");
                                            Utility.writeOnFile(name+".txt", "User has deleted file
("+fname+") on date "+Utility.getDate()+" and time "+Utility.getTime()+"",
getServletContext().getRealPath("/"));


                                        }

        rs=CommonDAO.getFiles(CommonDAO.getUserID(name),"uploaded");
                                            request.setAttribute("rs", rs);

        rd=request.getRequestDispatcher("/Files/JSP/User/files.jsp?no=4");
                                            rd.forward(request,response);
                                        }
                            }
                    }
                    catch(Exception e)
                    {
                            System.out.println("Opps's Error is in User UploadFile Servlet....."+e);
                            out.println("Opps's Error is in User UploadFile Servlet....."+e);
                    }
        }
}
```