LOVELY PROFESSIONAL UNIVERSITY

Enhancement in OTP generation process for cloud data security using Diffie-Hellman and HMAC

A Dissertation Submitted

By

Komalpreet Kaur

To

Department of Science & Technology

In partial fulfillment of the Requirement for the

Award of the Degree of

Master of Technology in Computer Science

Under the guidance of

Mr. Rohit Sethi

(May 2015)

INDIA'S LARGEST UNIVERSITY*

LOVELY PROFESSIONAL UNIVERSITY

Transforming Education, Transforming India

School of: Computer Science and Engineering

DISSERTATION TOPIC APPROVAL PERFORMA

Name of the Student: Komalpreet Kaur	Registration No: 11311418
Batch: 2013	Roll No. A75
Session: 14-15(I)	Parent Section: K2305
Details of Supervisor:	Designation: AP
Name ROHIT SETHI	Qualification: M.E (CSE)
U.ID 11542	Research Experience: 04 Year

SPECIALIZATION AREA: Networking and Security **(pick from list of provided specialization areas by DAA)**

PROPOSED TOPICS

1. Enhancement in OTP Process Generation in Cloud Computing
2. Security in MANET
3. Security in VANET

11542

Signature of Supervisor

PAC Remarks:

Topic 1 is approved.

25/9/14

11041

APPROVAL OF PAC CHAIRPERSON: **Signature:** **Date:** 18/09/14

*Supervisor should finally encircle one topic out of three proposed topics and put up for approval before Project Approval Committee (PAC)

*Original copy of this format after PAC approval will be retained by the student and must be attached in the Project/Dissertation final report.

*One copy to be submitted to Supervisor.

ABSTRACT

Cloud Computing is a technology or distributed network where user can move their data and any application software on it. With the use of this technology user need not to worry about the cost of hardware installation and their maintenance cost. Cloud computing provide different services to user. Cloud computing is now become most popular technology because of its availability, low cost and some other factors. But there is some issues in cloud computing, the main one is security because every user store their useful data on the network so they want their data should be protected from any unauthorized access, any changes that is not done on user's behalf. There are different encryption techniques used for security purpose like FDE and FHE. FDE encrypts the entire disk and FHE encrypts the particular functions. In representative architecture we use the term FHE for encryption. To solve the problem of Key management, Key Sharing various schemes have been proposed. The third party auditor is the scheme for key management and key sharing. The main advantage of this is the cloud service provider can offer the functions which were provided by the traditional third party auditor and make it trustful. The third party auditing scheme will be failed, if the third party's security is compromised or of the third party will be malicious. To solve this problem, we work on to design new modal for key sharing and key management in Fully Homomorphic Encryption scheme. In this we use the symmetric key agreement algorithm named Diffie Hellman, it is key exchange algorithm which create session key between two parties who want to communicate with each other and HMAC for the data integrity, OTP (One Time Password) is created which provides more security. Due to this the problem of managing the key is removed and data is more secured.

Certificate

This is to certify that Komalpreet Kaurhas completed M.tech dissertation titled **“Enhancement in OTP generation process for cloud data security using Diffie-Hellman and HMAC**” under my guidance and supervision. To the best of my knowledge, the present work is the result of her original investigation and study. No part of the dissertation has ever been submitted for any other degree or diploma.

The dissertation is fit for the submission and the partial fulfillment of the conditions for the award of M.tech Computer Science & Engg.

Date: **Signature of Advisor**

Name: Rohit Sethi

UID: 11542

ACKNOWLEDGEMENT

I would like to express my special thanks to GOD to give me this opportunity of writing thesis and providing such nice people who were there always to help me in my dissertation. Secondly, big thanks goes to my mentor "Mr. Rohit Sethi" who gave me this topic for my dissertation work, I am heartily thankful to Rohit Sethi for being my mentor and they also helped me in doing a lot of Research and I came to know about so many new things.

At last, I would also like to thank my parents and friends who helped me a lot in my research within the limited time frame. This is a complete work, I hope my mentor, parents and friends will help me till my thesis didn't complete.

DECLARATION

I hereby declare that the dissertation entitled, **“Enhancement in OTP generation process for cloud data security using Diffie-Hellman and HMAC**” submitted for the M.Tech Degree is entirely my original work and all ideas and references have been duly acknowledged. It does not contain any work for the award of any other degree or diploma.

Date: _______

Investigator
Reg. No. 11311418

TABLE OF CONTENTS

Chapter 3 Present work

Chapter 4 Result & Discussion

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Cloud Computing is the environment which provides on-demand & convenient access of the network to a computing resources like storage, servers, applications, networks and the other services which can be released minimum efficiency way. User retrieved data and modified data which is stored by client or an organization in centralized data called cloud. Cloud is a design, where cloud service provider provides services to user on demand and it is also known as CSP stands for “Cloud Service Provider”. It means that the user or the client who is using the service has to pay for whatever he/she is using or being used and served. It is a technique which gives a huge amount of applications under different topologies and each topology gives some new specialized services.

Example, Dropbox is a service of cloud where any user either with premium account (account with some extra features) or free account can use their cloud. There is one problem exist with it, that is any user can access any other user’s data without the knowledge of other user. Now the question arises that how we can prevent these kinds of issues. Standard enterprise security (such as firewall, antivirus and IDPS) is adopted by organizations to ensure the security for cloud computing. As the defense against the malicious services or services like identify frauds, almost all service provider organizations use the access control and user authentication mechanisms. The best feature mostly cloud service providers are providing is user can access the cloud from anywhere in the world.

To secure the user data, enterprises use the security mechanism, such as USB port control, Full Disk Encryption (FDE). But, are these mechanisms are good enough to secure the user data in cloud? The systems which runs 24*7 or all the time the above solutions are not effective that much. They cannot prevent the attackers to access data.

1.1 Cloud Computing

Cloud Computing is the environment which provides on-demand and convenient access of the network to a computing resources like storage, servers, applications, networks and the other services which can be released minimum efficiency way (Bhavna Makhija, 2013). Cloud Service Provider plays an important role in cloud. Users need not to buy software licenses or hardware to access any service. Users demand services from CSP or ISP and can access the internet services. This reduces the customer's expenditure and called "pay-per-use" service (Sean Carlin, 2011). There are four deployment models and three services models of cloud, theses are:-

1.1.1 Service Models: There are three service models of cloud-

- **Software as a Service (SaaS):** This is the capability of using applications which are running on cloud infrastructure. The users access these applications through internet connections. These kinds of clouds offer the implementation of some specific business threads that gives specific cloud capabilities. For E.g. GMAIL, Facebook. (Bhavna Makhija, 2013)
- **Platform as a Service (PaaS):** It gives the computational resources on which services and applications can be host and develop. For E.g. Online Photo Editing, Google Docs, YouTube. (Sanjoli Singla, 2013)
- **Infrastructure as a Service (IaaS):** This is the capability of doing processing, storing and run software which is given to the consumer. It's also referred as the "Resource Code" which provides resources as the services to a user. This work is done by the service provider. For E.g. Host Firewalls. (Young-Gi Min, 2012)

1.1.2 Deployment Models: Cloud services are mainly available in the three types of cloud. These clouds are as follows-

1. Public Cloud

2. Private Cloud

3. Hybrid Cloud

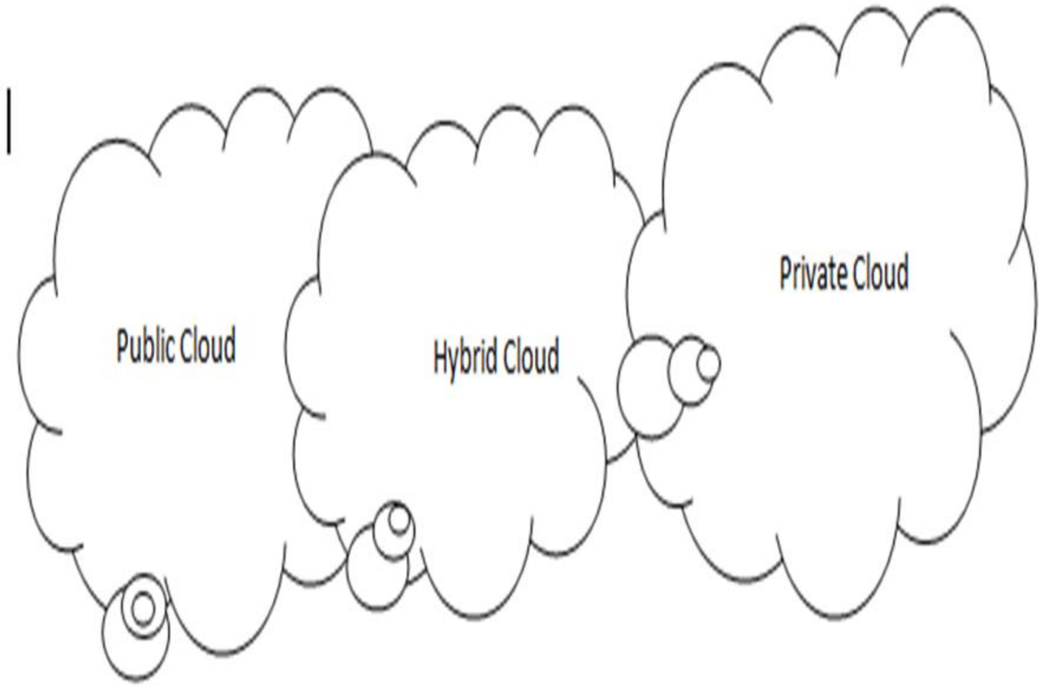


Figure 1.1 Deployment Diagram

- **Public Cloud:** In this cloud, resources allocated are publically. Applications in this cloud are on pay-per-use basis. Public clouds can be managed by government organizations or business. For E. g. Sky Drive and Google Drive. (Bhavna Makhija,2013)

- **Private Cloud:** In this cloud, resources are limited and used within an organization. It is more secure as employees in an organization can access the particular data only. For E. g. Banks. (Sanjoli Singla, 2013)

- **Hybrid Cloud:** In this cloud, there is a combination of both Public and Private cloud. The services within the organization are control by the customer and resources which need to be delivered externally are controlled by the service provider. (Sanjoli Singla, 2013)

- **Community Cloud:** This cloud is used by those organizations which have same concerns like security requirements; mission or policy. This is managed by organizations within a community or by the third party auditor. (Bhavna Makhija,2013)

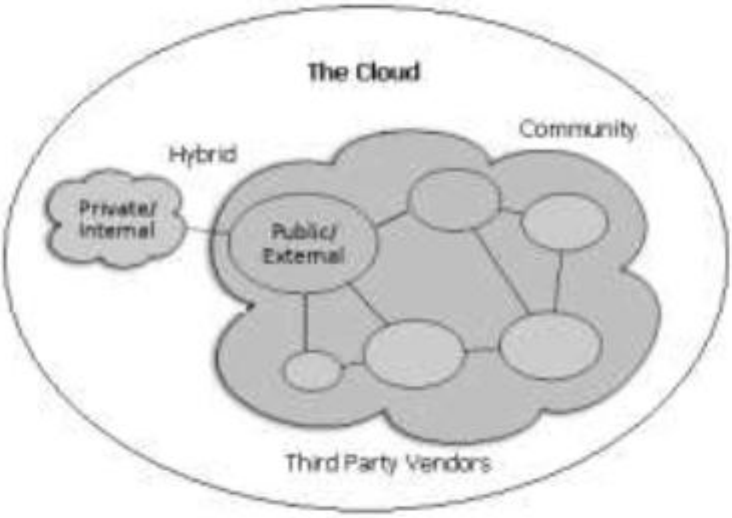


Figure 1.2 Deployment model of Cloud (Sanjoli Singla,2013)

1.1.3 Characteristics of cloud computing

- **On-Demand Self-Service:** It gives the "Graphical User Interface (GUI)" control panel (on websites etc) to use the computing resources like Network bandwidth, Additional computers or the user account.
- **Device & Location Independence:** It enables the user to use the online system using a web browser or the application (For eg. Dropbox, Mediafire) regardless the location of the user and whatever device he is using (Example, Tablet, I-Pad).
- **Broad Network Access:** It allow users to have the access on computing resources over the networks like the Internet from a wide range of devices such as tablet, PC, laptops.
- **Reliability:** It provides the reliable services.
- **Virtualization:** It allows sharing of storage media and servers which enhance the utilization.

- **E-pay-Per-Use:** The best feature ever, user has to pay only for what they are using and they can see their usage and can calculate that how much they has used and how much they need to pay according to usage.

1.2 Cloud Computing Security Issues

1.2.1 Cloud Computing Security

Network security, information security and many other security types like the computer security together make the term “Cloud Security”. Because it consist all of the security mechanism given above. It gives the broad set of technologies, policies and controls that are used to secure the data and applications exist with the cloud computing environment(Deyan Chen, 2012). It is not the product of computer security like anti-viruses and anti-spam’s. Security is the most concerning point to any service. External security or internal security required to each field. Only security ensures the privacy and integrity the cloud data. There are many security loopholes exist in the service. There are many types of security issues exist like DDOS, Man in the middle etc.

Some of the security types include:

- Data Loss
- Phishing
- Password Cracking
- Botnets and Other Malware

1.2.2 Security in Cloud

Security is the only issue which we need to focus in the cloud computing. You actually don’t know what is going on with your data, because it is stored by the third party like application and web browser.

1.2.3 Data Leakage

Centralized of data is the major benefit to cloud computing. When data is stored on the hard disk drive without any encryption, there are many chances of leakage of the data. When you upload your data to the cloud, there are proper encryption standard to encrypt the data and no one can take the advantage if got encrypted data. Overall conclusion is your data is secure on the cloud as compare to the local storage.

1.2.4 Offloading Work

The major advantage of the cloud is, you don't have to do anything yourself. Everything depends on the provider. They will provide you the security.

1.2.5 Forensics

If you think your data has been compromised, there is a service or online service for this. You can clone up the whole data offline and can analyze it.

1.3 Attacks in Cloud Computing

1.3.1 Denail of Service Attack (DoS)

Denial of service (DOS) is very common attack. It may totally interrupt or slow down the system. As cloud is used by many customers this may lead to possibility of DOS attack. In this attack attacker sends the number of zombie processes to the server that may be wrong query which may lead to eat up of resources or crash the server. These DOS attacks are possible in many different layers. The different attacks are possible in these layers. Jamming comes under physical layer. Sinkhole, wormhole attacks are possible in network layer attack. Flooding is one of the transport layer attack.(Young-Gi Min, 2012)

1.3.2 Cloud Malware Injection Attack

A challenger insert malicious code into a system in malware-injection attack. This attack can be viewed as scripts, code, active content, and other software. When an request of a legal user is ready to run in the cloud server, the individual service accepts the request for calculation in the cloud. The inspection done is to decide if the request matches a legal existing service. The truthfulness of the request is not checked. By penetrating the order and duplicating it as if it is a legal service, the malware activity succeeds in the cloud.

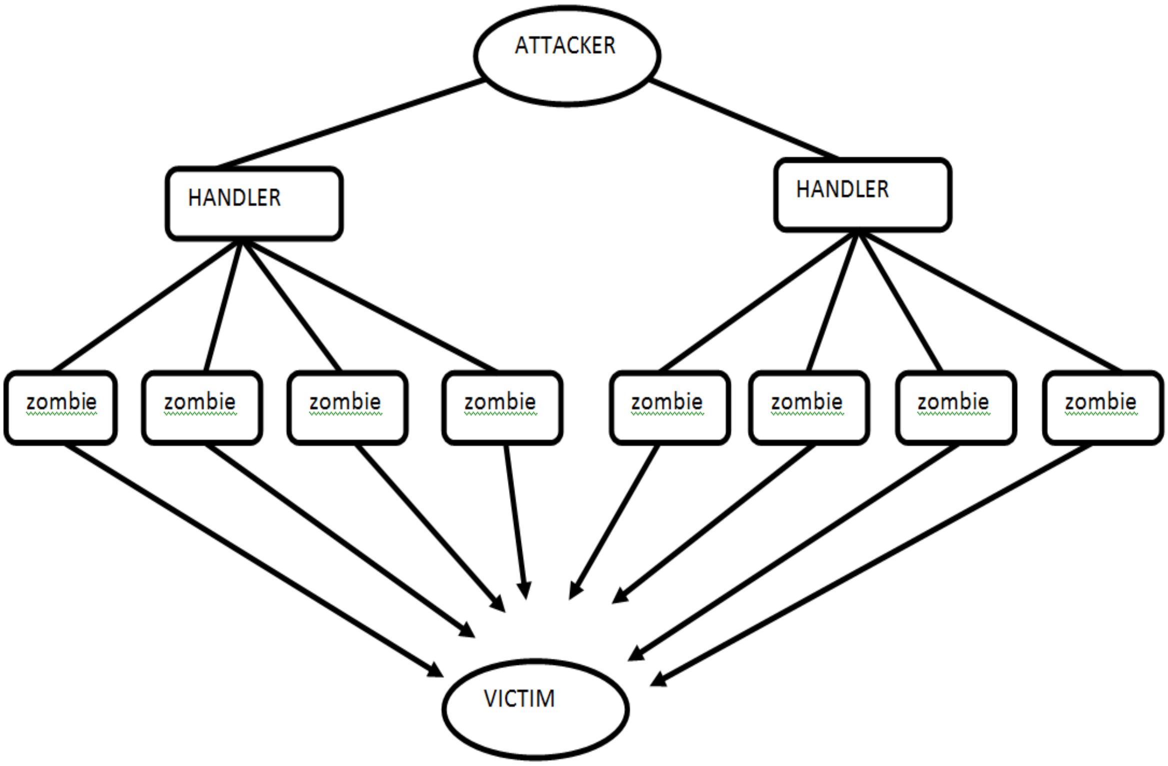


Figure 1.3 DDOS attack

1.3.3Side channel Attack:

An attacker can place the malicious virtual machine just near to actual virtual machine and pretend to be the real virtual machine and can get IDs of legitimate users. By using these legitimate IDs attacker can act as legitimate user to the cloud service provider and access all the information associated to that ID

1.3.4 Account Hijacking

Account hijacking is a process or method through which an computer account, email account or any other account of individual's, associated with a computing device or service is stolen or hijacked by a hacker or any unauthorized person. In this the hacker uses the stolen account information for the purpose of malicious or unauthorized activity.

This hijacking is done through many methods like phishing , guessing the password or number of hacking tactics .

1.3.5 Man-In-The-Middle Cryptographic Attacks

The man-in-the-middle attack are sometime called bucket brigade attack(often abbreviated MITM). This is done by any attacker when he place himself between two parties when they are communicate with each other. The attacker read all the message and also modify them.

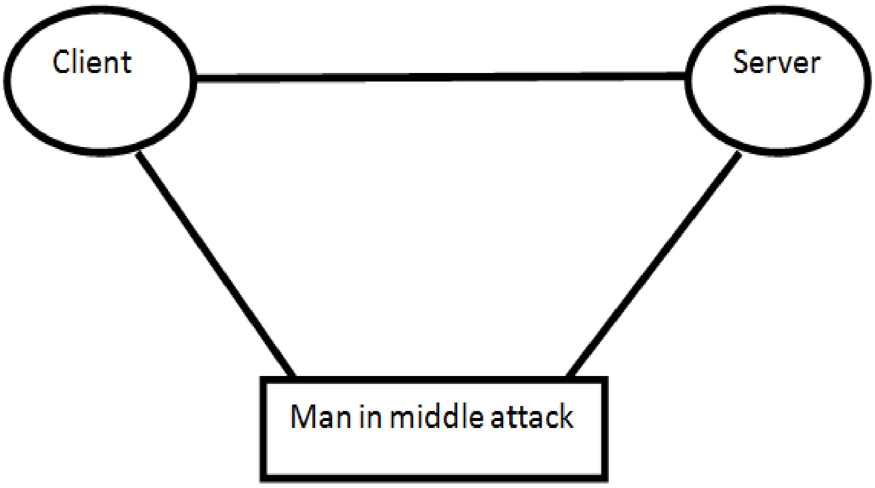


Figure 1.4 Man in the Middle attack

1.4 Fully Homomorphic Encryption

Fully Homomorphic encryption provides the better security than full disk encryption. Unlike FDE the encryption is not applied on full disk, encryption is applied on each function. The cipher text and plain text is not related but the emphasis is on the algebraic operation that works on both of them (Simarjeet Kaur, 2012).

After the invention of RSA, Rivest, Adleman and Dertouzos introduce the idea of Fully Homomorphic schemes. They asked for an encryption function that permits encrypted data to be operated on without preliminary decryption of the operands, and they called those schemes privacy homomorphism.

Fully Homomorphic Encryption can be used to inquiry a search engine, not including, what is being searched. More accurately, FHE has the many properties. Assume that cipher text ci decrypt to plaintexts mi.

Therefore, Decrypt (ci) = mi

Where the mi's and ci's are elements of some ring with two operations, addition and multiplication. In FHE one has

Decrypt (c1 + c2) = m1 + m2; Decrypt (c1 * c2) = m1 * m2

The operations are performed on encrypted data in fully homomorphic encryption and in this private key are not known. The secret key is only hold by the client. When we decrypt the result of some operation, it is the similar as if we had approved out the calculation on the raw data.

A fully homomorphic encryption scheme is a standard encryption scheme in which two functions "enc" and "dec" i.e. encrypt and decrypt with one extra function, which will term as "eval." This "eval" accepts as input the text of a program and a cipher text, and produces as output a cipher text as shown in the diagram.

$$\begin{tikzcd} \mathrm{enc}(m) \arrow[r, "\mathrm{eval}"] & \mathrm{enc}(f(m)) \\ m \arrow[u, "\mathrm{enc}"] \arrow[r, "f"] & f(m) \arrow[u, "\mathrm{enc}"'] \end{tikzcd}$$

Fig: 1.5 Encryption methods

1.5 Full Disk Encryption

The whole physical disk is encrypting with physical key for the better speed and simplicity in disk firmware in the case of fully disk encryption. In case of stolen laptop it is very effective technique to protect the data. It cannot fulfill the requirement of data protection goals in the cloud but physical theft is not the main threat. Full Disk Encryption is one of the most successful ways to protect our private data on laptops, tapes etc. Your data could be permanently lost when an encrypted hard drive goes bad.

FDE solution comprises a number of methods for receiving admittance to the drive when a consumer can no longer authenticate. This may be a recovery key, a recovery password or an emergency log-on account. Once common with the practice, make sure that the recovery information is centrally backed up, test your recovery strategies.

1.6 Difference between FDE and FHE

The difference between FDE and FHE is given below:

1. **Key management and trust**: In case of full disk encryption the keys may be located on or close to the physical drive. Key management is not handled by the user. On physical disk user data is encrypted. Consequently, FDE doesn't avoid online attacks. Any unauthorized party can leak the data in FDE. But in case of FHE data is not leaked by any unauthorized party. FHE encryption keys are managed by the user, but the data is not known to the user.

2. **Sharing**: Collaboration is often known as a "killer feature" for cloud applications. In FDE, cloud service provider is trustworthy for users as it provide correct access control because the key granularity i.e. the whole disk doesn't line up with access control granularity i.e. a single data unit. In FHE, as the user or third-party cloud provider manages the encryption keys, the best way of providing access control isn't clear yet.

1.7 Diffie Hellman key exchange

Diffie Hellman was the first public key algorithm or we can say that it is symmetric key agreement ever invented, in 1976. Diffie Hellman key agreement protocol is :

- It allows exchanging a secret key between two parties.
- Exponential key agreement

- Requires no prior secrets

Definition of Diffie Hellman: Before establishing a symmetric key, the both the two parties need to choose two numbers n and p. Let n be a prime number and p be an integer. The Diffie Hellman Problem (DHP) is the problem of computing the value of p^{ab}(mod n) from the known values ofp^{a} (mod n) and p^{b} (mod n).The setup of Diffie Hellman algorithm

- Suppose that we have two parties Alice (Master) and Bob (Slave), they want to communicate to each other.
- They do not want the eavesdropper to know their message.
- Alice and Bob agree upon and make public two numbers n and p, where n is a prime number and p is a primitive root mod n. Anyone has access to these numbers.

Table 1: private computations

Alice	Bob
Choose a secret number a. Compute $M \equiv p^{a}$ (mod n)	Choose a secret number b Compute $S \equiv p^{b}$(mod n).

- Generated public values are exchanged.
- Alice sends M to Bob ==M
- S= Bob sends S to Alice
- Alice calculate the number $K \equiv S^{a} \equiv (P^{a})^{b}$ (mod n).
- Bob calculate the number $K \equiv M^{b} \equiv (p^{b})^{a}$ (mod n).

Here Alice and Bob have the same key that is $K=p^{ab}$ (mod n).

In the Diffie-Hellman algorithm if two parties, say, Master and Slave wishes to exchange data, both agree on a symmetric key. For encryption or decryption of the messages symmetric key is used. We knows that Diffie Hellman algorithm is used for only key agreement or key exchange, but it does not used for encryption or decryption. Before starting the communication, secure channel is established between both the parties. Both parties select their own random number. On the basis of the selected random numbers, secure channel and shared key is established.

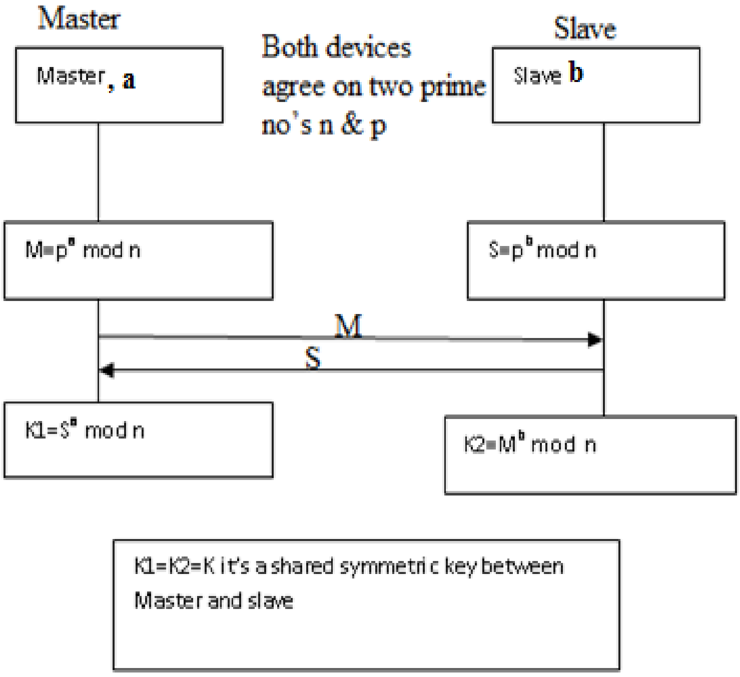

Master
Both devices agree on two prime no's n & p
Slave
Master, a
Slave b
M=p^a mod n
S=p^b mod n
M
S
K1=S^a mod n
K2=M^b mod n
K1=K2=K it's a shared symmetric key between Master and slave

Figure 1.6: Diffie-Hellman Key exchange

Figure shows that Master and Slave wants to communicate with each other. To startcommunication both parties need to establish secure channel. To establish secure channel, two random prime number p and n are selected, both devices are agreed on these two numbers. Selected p and n are the public numbers. Both parties, say device 1 become master and device 2 become slave; both master and slave select their private numbers ‘a’ and ‘b’ respectively. Master and slave use their public and private number and calculated their private keys.

Master computes:

M= p^a mod n

Slave computes:

S= p^b mod n

Now both master and slave exchange their private keys such as ‘M’ and ‘S’. After getting ‘M’ and ‘S’, master and slave calculates the secret keys such as K1, K2.

From S, master computes:

K1= S^a mod n

From M, slave computes:

K2= M^b mod n

If both master and slave calculate same values of K1 and K2, then secure channel is established between them. The combination of K1 and K2 becomes the shared symmetric key between master and slave.

To encrypt the messages, they used the public key or shared key (K) of both parties. For decryption of messages private key of both parties which is randomly chosen by the users i.e. ‘a’ and ‘b’ are used.

Chapter 2
Review of Literature

"Enhanced Data Security in Cloud Computing with Third Party Auditor" (Bhavna Makhija, 2013). In this paper they proposed different techniques and their merits and demerits like Message Authentication Code(MAC) which protect the data from integrity. The owner of any information verified the data integrity by recalculating the message authentication code of data received by others but recalculation is possible if the amount of data is very large. A hash tree is used for large files. Third party auditor is used to relieve the large data into small parts of maintenance and security. The proposed algorithm describes data integrity and dynamic data operations. They use encryption to ensuring the data integrity. Public key is also defined which is based on homomorphic authenticator. A hash function is used for proof of retriveability. The proposed algorithm has a main drawback that it require implementation of the higher resources cost.

"Securing the Cloud Environment Using OTP" (Vimmi Pandey, 2013). In this paper, Dynamic mobile token application is introduced. This is the application in mobile phones which is used to generate a code with the help of OTP (One Time Password). This OTP code is used only for one time to login session. In this paper, they describe one of the methods of OTP. There are two phases in it Registration phase and Login phase. User first register itself by fill credentials in the form and then enters to the Login phase. In login phase, OTP will generate for the login session. OTP is generated by three parameters: The current time, 4-digiti PIN code and Init-secret. This code is valid for three minutes only. This ensures protection against eavsdroppers attack and man-in-middle attack. Hence, they prove OTP is very secure.

"Cloud Data Security using Authentication and Encryption Technique" (Sanjoli Singla, 2013). In this paper, a design and architecture is proposed that can help to encrypt and decrypt the file at the user side which provides data security in both cases while user is at rest or is transferring data. In this paper they used the Rijndael Encryption Algorithm along with EAP-CHAP. This algorithm has five steps which need to be follow for the data security. The users are always concern about the privacy protection and security issues before storing their data on cloud. So in this the focus is on client side security in which only the authorized user can access the data. Even if some intruder

(Unauthorized user) gets access of the data then the data will not be decrypt. Encryption must be done by the user to provide better security Algorithm . For this, Rijndael Encryption algorithm is used.

“Cloud Computing Security” (Ankur Mishra, 2013). In this paper, two techniques are discussed: Virtualization and Muti-tenancy which provides security about cloud computing. As data is organized by third party organization that offer Saas and PaaS which is important for the security. So, Virtualization and Multi-tenancy techniques are used for the security purposes. Virtualization is a way of making a physical computer function as if it were two or more computers where each non-physical or virtualized. There are two types of virtualization: Full virtualization and Para virtualization and two architectures of virtualization: Hosted and Hypervisor architecture. Multi-tenancy is the ability to provide computing services to multiple customers by using a common infrastructure and code base. Multi-tenancy can be applied to different levels i.e. application level, middleware level, operating system, hardware level. Then security of virtualization and multi-tenancy has been discussed.

“A Novel Based Cross Domain Access Control Scheme for Cloud Storage” (Punithasurya, 2013). In this paper, when dealing with public cloud, security should be taken care. Security is very important in cloud computing. Security includes authentication, authorization and access control. In cloud storage, there are many access control schemes available. Control is the main part of cloud computing. Only access control that provide authorization to many users or authorized users. Access control has access privileges that are required by the user. Security is the major concern in cloud computing. For security purpose, Role Based Access Control (RBAC) is used. By the use of this technique time location and availability can be enhanced.

“Cloud Computing Security Case Studies and Research” (Chimere Barron, 2013). In this paper they discussed different issues related to cloud computing security. To protect cloud computing system and to prevent various attacks many security mechanisms have been developed. To improve the security of cloud computing new technologies has been developed by the researchers. Different types of attacks like SYN flood, malware injection, account hijacking are discussed in this paper. The main focus of this paper is on detecting and preventing SYN flood in cloud computing. The author developed two algorithms; detecting algorithm and preventing algorithm. These algorithms will implement and test these algorithms on cloud computing.

"Cloud Data Protection for the Masses" (Dawn Song, 2012) .In this paper the author described the data-protection-as-a-service where different services are provided for protecting data. Two techniques have discussed i.e. FDE (Full Disk Encryption) and FHE (Fully Homomorphic Encryption). There is a comparison in these techniques on the basis of key management, sharing, and ease of development, maintenance, aggregation and performance. The key management and access control are moved by DpaaS (Data-Protection-as-a-service) approach for purpose of balance easy maintenance and rapid development by user-side verification. Performance and ease of development offered by FDE is excellent.

"Data Security and Privacy Issues in Cloud Computing" (Deyan Chen, 2012). In this paper, it describes data security and privacy protection issues occurred in cloud computing in all the stages of data life cycle. There are seven stages of data lifecycle: Generation, use, transfer, share, storage, archival, destruction. They have discussed some current solutions like fully homomorphic technique, data integrity, client-based privacy management tool, etc. No doubt cloud has many advantages but still there are many issues like security need to be solved. According to the survey of Gartner for cloud computing, Public and Hybrid cloud has a revenue of $59 billion and by the year 2014 it will reach USD 149B with a annual growth of 20.The increase in the revenue of cloud with the time shows that cloud is a trustworthy industry. But still there are some issues in cloud regarding security of data which increases the threats from hackers.

"Cloud Computing Security Issues and Access Control Solutions" (Young-Gi Min, 2012). In this paper, three cloud computing models are introduced i.e. SaaS, PaaS, IaaS. There are five layers in cloud computing models are mentioned: Client, application, platform, and infrastructure and server layer. To solve security problems, every level should have security implementation. Security requirements of cloud computing and the solutions for the security problems are described. Different security attacks are defined which need to be overcome by applying security algorithms and another techniques. To have secured cloud deployment, areas like computing architecture, portability and interoperability, traditional security, business continuity, disaster recovery, data centre operations, Encryption and key management, identity and access management must be considered. The best way to minimize the unauthorized access is using Digital ID's for the employee, this also addresses the issue of non-repudiation.

"Cryptography and Encryption in Cloud Computing" (Simarjeet Kaur, 2012). In this paper various data encryption schemes like homomorphic encryption, searchable and structured encryption, Identity based encryption, signature based encryption etc. are discussed. These are emerging technique in cloud world security to provide day night full protection to critical data information.

"Applying Agents to the Data Security in Cloud Computing" (Dian-Yuan Han, 2012). In this paper, service interactions security module is introduced. To protect the cloud there are three layers considered: the first layer is traditional transport layer, second layer is cloud computing layer and requirements and third layer is application-driven layer. All the three layers do their work to ensure data security in cloud computing. Failures are normally occurred when there is high availability; high fault tolerance and high efficiency accesses to Internet based cloud data centers. These types of issues are significant and need to be handle carefully. These issues are more important and valuable then high performance. Agents are introduced to data security module in order to provide more efficient and useful services.

"A Proposed Model for Enhancing Data Storage Security in Cloud Computing Systems" (Dr Nashaat el-Khameesy, 2012). In this paper suggests a methodical application of "defence in depth" security technique that can help all security risks in networked storage. Defence in depth based networked storage security policy is more important because provides a comprehensive framework fot future attacks as the current technologies are more clearly understood. Storage security with defence in depth is growing well that will help in making storage much more resilient to future threats.

"Ensuring Data Storage Through A Novel Third Party Auditor Scheme in Cloud Computing" (Shui Han, 2011). In this paper, trustful third party is introduced in which the user can operate and store their data securely in cloud. There is a problem of data storage security in cloud computing. For more security of the cloud new scheme is introduced i.e. novel third party auditor. The advantage of this scheme is that the cloud service provider can offer the functions which were provided by the traditional third party auditor and make it trustful. It reduces the complexity in Cloud Computing. The third party auditor provides techniques like RSA and Bilinear Diffie-Hellman. By using RSA algorithm encrypted data is flow from sender to the receiver and by using Bilinear Diffie-Hellman keys are exchanged for security purposes. With the exchanging of keys, data is always sent to the valid and authorized users only.

"Cloud Computing Security" (Sean Carlin, 2011). In this paper, cloud computing is the distributed architecture that centralizes the resources of server on a scalable platform which provides services on demand. Various cloud deployment models are discussed i.e. public, private and hybrid. The main security issues and risks are discussed; sharing of resources is one of them. Customers are not satisfied with the data security on cloud. Cloud service providers must tell the customers about the deployment models. They need to use the third party auditor so that they can gain the trust of customers. For this, new techniques need to be developed and older should be removed for easy work in cloud architecture.

"**Cloud computing security using encryption Technique"(Geethu Thomas, 2010).** In this paper they presented that the cloud computing is very efficient technology or important field used for data storage due to its efficiency and flexibility. The cloud provides different types of services to the user and the architecture is also based upon those services. The data is stored on to data centers having a large size of data storage. The data as well as processing is somewhere on servers.

Chapter 3

Present Work

3.1. PROBLEM FORMULATION

At the very lower cost, cloud provides the services which can be accessible all the time, anywhere, anytime. User data privacy is the key challenge in cloud computing. Encryption is the only effective way to maintain the privacy and which provides the security in cloud computing. By this, two encryption schemes come into the picture. These are:-

1) Full-Disk Encryption (FDE)
2) Fully Homomorphic Encryption (FHE)

In FDE complete physical disks always encrypted with a symmetric key for simplicity and speed. FDE is very emphatic in maintaining the protection of private data like backup tapes and stolen laptops that key concern is that, it cannot fulfill the goal of data privacy in the cloud.

FHE generally provides the computations on cipher texts. Any function in plaintext can be converted into an equivalent function in cipher text: server performs the real work but it does not know the data which is computed by server. This property provides a privacy guarantees when it's computing on the data which is private but the practically implementation is still a constant question. On the basis on some factors, the comparisons have to make between them. Factors are:

- Key management and Trust
- Sharing
- Aggregation
- Performance
- Ease of development
- Maintenance

From the above factors, it is concluded that scheme of "Fully Homomorphic" is more reliable. It gives more privacy and security as compare to scheme of "Full Disk Encryption". The main problem which is there in Fully Homomorphic Encryption is a key storage, key management, Access control and Data Aggregation list maintaining. To solve problem of Key management, Key Sharing various schemes have been proposed in last years. The various security attacks are possible in these schemes. The third party auditor is the scheme for key management and key sharing. The third party auditing scheme will be failed, if the third party's security is compromised or of the third party will be malicious. To solve this problem, In this thesis we will work on to design new model for key sharing and key management in fully Homomorphic Encryption scheme.

3.2. Objectives of Research

1. To analyze FHE and FDE encryption schemes for data security in cloud computing.
2. To enhance FHE encryption scheme for key management and key sharing.
3. The enhancement will be based on Diffie-hellman, HMAC and OTP technique for cloud data security.
4. To implement proposed and existing schemes and compare results in terms of time, probability and space.

3.3 Research Methodology

This study is mainly focused on to develop modal for fully homomorphic disk encryption schemes. The new scheme will provide reliable key storage and key management services. This will enhance the reliability and security of the existing fully homomorphic encryption scheme. In this new modal, secure channel establishment algorithm will used for key management and key sharing. The secure channel establishment algorithm is Diffie- Hellman. The Diffie- Hellman algorithm is most secure and reliable algorithm. In the Diffie-Hellman algorithm if two parties, say, Master and Slave wishes to exchange data. Before starting the communication, secure channel is established. Both parties select their own random number. On the basis of the selected random numbers, secure channel and shared key is established.

We have embedded the Diffie Hellman key exchange algorithm for authentication procedure. In cloud network, it defines the source node and destination node. To establish secure channel between communicating parties, each party select a random prime number g and n, selected numbers become public keys of both parties. The source node become master and destination node become slave, master and slave select their private keys ‘a’, ‘b’ respectively.

The master calculates new value “M” from their selected public and private numbers.
$M= g^a \bmod n$
The Slave calculates new value “S” from their selected public and private numbers
$S= g^b \bmod n$
The Master and slave exchange their calculated “M” and “S” values through intermediate nodes. When Slave receives “M” and Master receives "S” both parties will calculate mode inverse value.
When master receive value “S” from slave and calculate new value “K1” from the received “S” value.
$K1=S^a \bmod n$
Slave receives value “M” from master and calculates new value”K2” from the received “M”
$K2=M^b \bmod n$
After calculating “K1” and “K2”, both parties establish secure channel, by calculated new key “K”. If both communicating parties have same “K1” and “K2” values, secure channel is made between Master and Slave.
$K=K1+K2$

Communication starts between both parties when secure channel is made between master and slave. The communication between Master and Slave is encrypted with public keys. Each parties use their own private keys to decrypt the communication.
The following flowchart explains the above work, how the embedded Diffie-Hellman works in security of cloud networks.

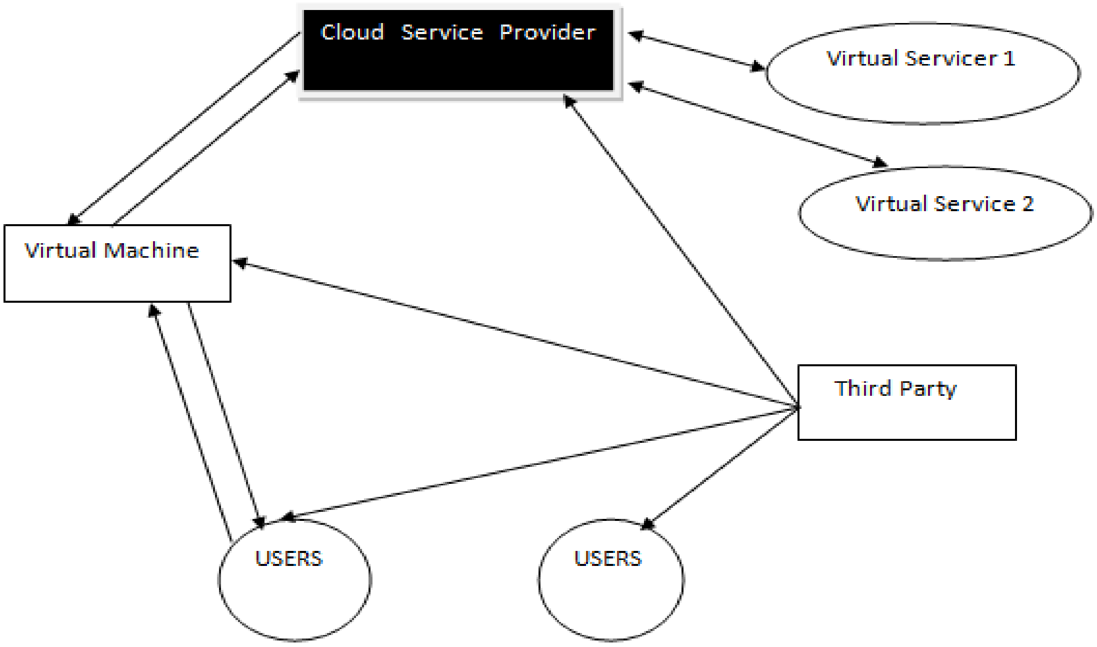


Fig. 3.1: Data communication diagram

In figure 3.1, a cloud service provider is present which is associated with virtual servers bidirectional. It is also associated with virtual machines. There are numbers of user are present in a network. These users are associated with virtual machines to exchange data between them. Third party is also present in the network. This party is attached or linked with virtual machines, users and cloud service provider. Now in the existing work homomorphic encryption technique is applied at the virtual machine. But this scheme has no key sharing and key management scheme. Due to lack of this technique security of the network is at risk level. The possibility of the attack is more in this scheme. To overcome this problem at virtual machine Diffie-Hellman algorithm is applied upon the virtual machines instead of homomorphic encryption scheme. In Diffie-Hellman technique key sharing and key management scheme is used in it so that proper security is provides to the network. Public and private key is shared between sender and receiver first. After the sharing of keys between both parties then communication begin between user and virtual machine. We can say that secure channel is established between them.

In our work, we are using Diffie-Hellman algorithm for secure channel establishment, OTP for providing more security to data and HMAC for data integrity. In our proposed scheme only two messages are needed to exchange between two devices and a secure channel will be established. It is more secure than the existing authentication procedure. It takes less time to authenticate the users and it enhances the performance of the mobile devices in the network. Diffie Hellman key exchange algorithm provides the safety against attack. In Diffie -Hellman algorithm, there is no provision for the storage or exchange of the PIN key. So it protects network devices from attacks.

One Time Password (OTP): Password is used for authentication by all the business and organization. Moreover Static passwords have many limitations. Password can be get hacked. Lackadaisical employee may note down passwords somewhere, system with saved passwords may be used by various users or a malicious user may reset all passwords just to create havoc. So it is very useful to use dynamic password i.e. one time password. Dynamic passwords are more secure as compared to static. There is no need to write down these passwords and remember these passwords. For each login session each time a new password is generated. One time passwords are more reliable and user friendly as well for authentication. OTP generation can be done by various OTP generation algorithms for generating strings of passwords. OTP ensures safety. This leads to authenticating them again and again over the period of time for each login session. To avoid the overhead we can use OTP for multi cloud environment.

HMAC: Data integrity is very important in cloud computing. As today's cloud is very vast and number of cloud users are increasing day by day, so it is the major challenge for cloud to have data integrity. Data integrity provides the assurance by the user that the data is not modified or corrupted by the service provider or other users. Data integrity in cloud can be done by using Message Authentication Code (MAC). The data is stored in the cloud by a user and the integrity of the data is checked by the auditor. For ensuring the integrity of the data, three approaches are there:

- Diffie-Hellman (DH) algorithm, which generates the symmetric key which is known only by the user and the auditor.

- Exclusive- OR (XOR) to perform a xor operation between the message and the key generated using DH algorithm.
- Secured Hashing Algorithm (SHA1) is hashed algorithm which generates a separate key using a one-way hash function by passing the original message to the hash function. This is done by both the user and the auditor and the value that is obtained from both of them is compared and hence the data integrity is verified.

In recent years, interest in developing a MAC is increasing which is derived from hash code like MD5, SHA-1 or RIPEMID-160. The reason of this increasing interest is MD5 which is hash function that was not designed to use it as MAC because it does not use the concept of secret key. HMAC incorporate a secret key into hash function, for this HMAC received the strong support. HMAC is mandatory for IP Security and is also used in many other proto0cols like Transport Layer Security (TLS) or Secure Socket Layer (SSL) and Secure Electronic Transaction (SET).

Hash based Message Authentication Code (HMAC) is a cryptographic hash function in which the message and the key are hash them together. Hash function of message authentication code can be calculated by secret key. SHA is a hash algorithm which is used to generate the authentication code for the message. This authentication code is used to check authentication of message which is to send by using secret key. This authentication code is also used for the data integrity which insures that the data is send to the valid receiver. More is the strength of hash function; hash output size and key size, more will be the strength of HMAC. HMAC supports hash algorithms like MD5, SHA-1, and SHA-256.

HMAC= Hash [(K+ XOR opad)] || Hash [(K+XOR ipad) || M]

K + is K padded with zeros on the left so that the result is b bits in length
ipad is a pad value of 36 hex repeated to fill block
opad is a pad value of 5C hex repeated to fill block
M is the message given as input to HMAC

The output binary authentication code which equals in the length to that of the hash function digest. By comparing the value of hash function at both sides i.e. sender and authenticator, the data integrity of the file can be checked. In HMAC, authentication codes are generated by using secret key and hash based algorithm. This code ensures the usage of hash function is expanded in many places. It conserves the performance of hash function.

Algorithm:

```
Selected node suppose user1
1. Login
2. Key generation
    2.1 Enter prime numbers
    2.2 Enter random numbers by client and cloud service provider
    2.3 Secret key generation and secure channel establishment
3. OTP (One Time Password) generation
    3.1 cloud server will set count1=0, count2=0...count5=0 for respective user at its side.
    3.2 Cloud Server will request for the OTP from user 1
    3.3 user1 enter (secret key+count) as OTP
    3.3 server match it because server knows both secret key and count of each user.
        3.3.1: count1++; // so for user 1 it will be count1=1; for remainig user their count will be still 0;
        3.3.2 if ( secret_key+count(x) == secret_key+count(y))
            {   Access granted;
            display message by server : print ("please enter the operation”);}
            else{   display message by server: print(" wrong password, your login number is count1);}
4.4 clinet will enter the operation using HMAC digest
        4.4.1 :  hmac(already generated secret key || v, file1,ver1 ||  sha1 )
                    if (ope==r)
                        {   data will read from server;
                        else { print(choose valid operation)}}
                if (ope==v)
                            {    server will check the file name and version;
                                 if (file1,ver1== file1,ver1)
```

```
                                        { printf("file is valid"); }
                                     else { print  ( file is invalid, please replace the file)
                                        }}
```

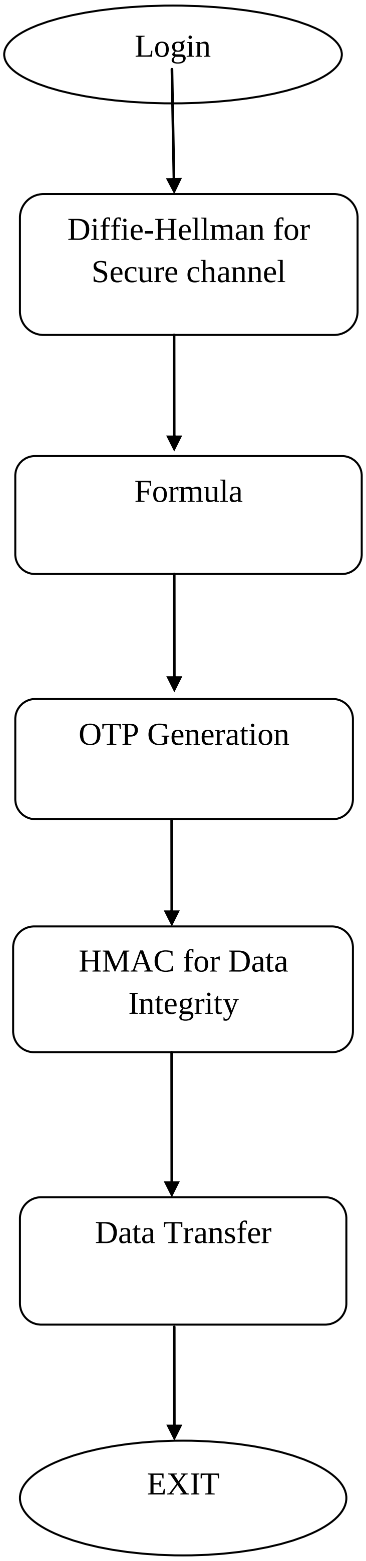
5. encryption/decryiton

6. Data operation

7. Logout;

note: // 1.at client side, user will enter prime number, random number for generating secret keys, once generating secret key user will enter otp , after inserting otp, user will enter operation(Insertion) with corresponding file name(file1 or file2).

FLOW CHART:


Login
Diffie-Hellman for Secure channel
Formula
OTP Generation
HMAC for Data Integrity
Data Transfer
EXIT

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction to MATLAB

MATLAB is a MATrix Laboratory. It is an interactive program which provides numerical computation and visualization of data. With the help of its programming capabilities it provides tool which is very useful for all areas of science and engineering.

The proposed idea is implemented in MATLAB which is broadly used in all areas of research universities, and also in the industry. MATLAB is valuable for mathematics equations (linear algebra) and also beneficial for numerical integration equations. These equations are solved by MATLAB. It is also a programming language, and is one of the simplest programming languages widely used for writing mathematical programs. It has a range of types of tool boxes that are very precious for optimization and many more.

Key Capabilities

- Launch computations on the cloud directly from MATLAB using Parallel Computing Toolbox.
- License MATLAB Distributed Computing Server with on-demand (hourly), perpetual.
- Easily manage MATLAB clusters running on Amazon EC2 on the MathWorks Cloud Center site.

Requirements

- MATLAB (R2013a or later) and Parallel Computing Toolbox for creating applications on the desktop
- Familiarity with Parallel Computing Toolbox concepts
- MATLAB Distributed Computing Server license for executing applications on the cloud

4.2 Problem Implementation

4.2.1 Network deployment

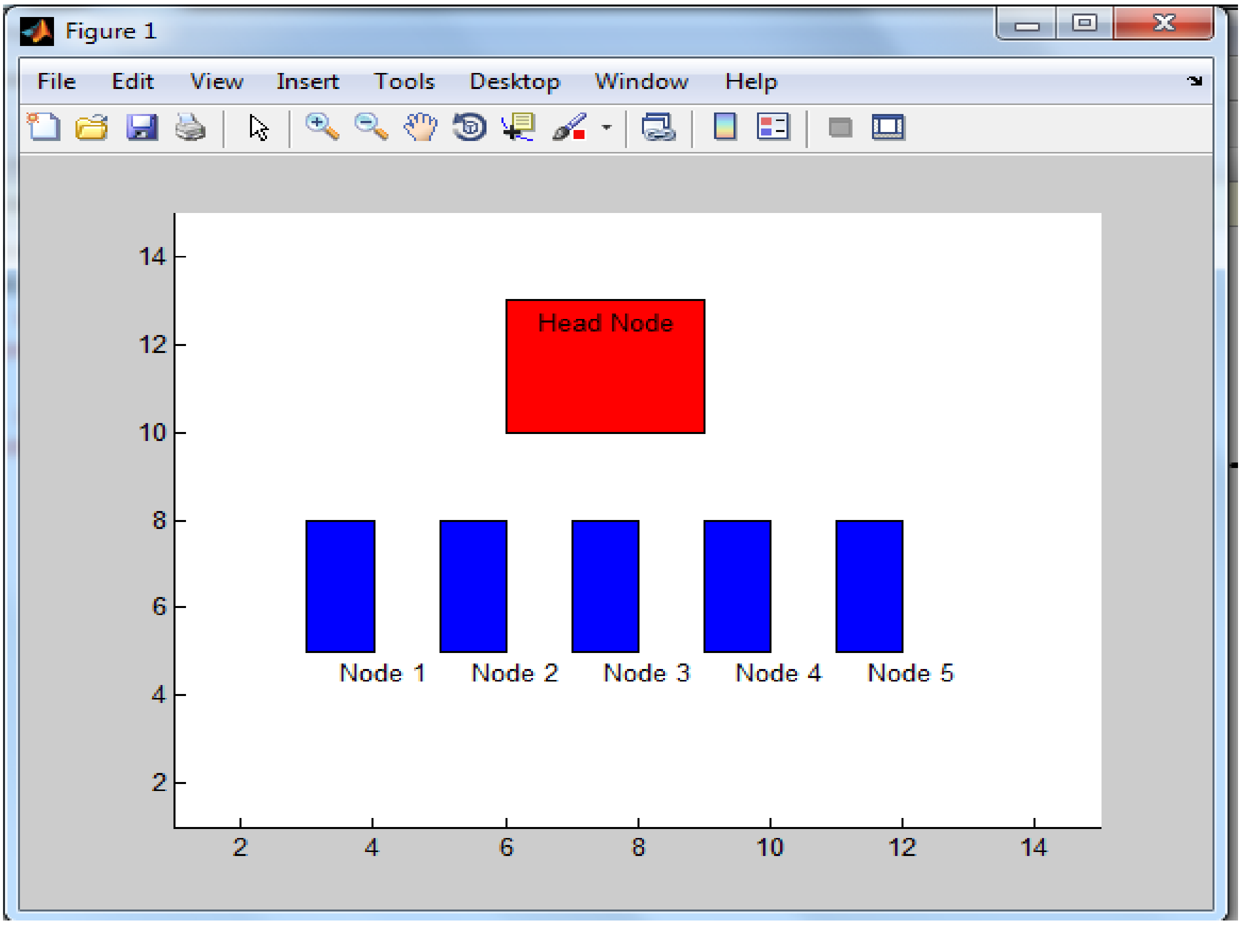
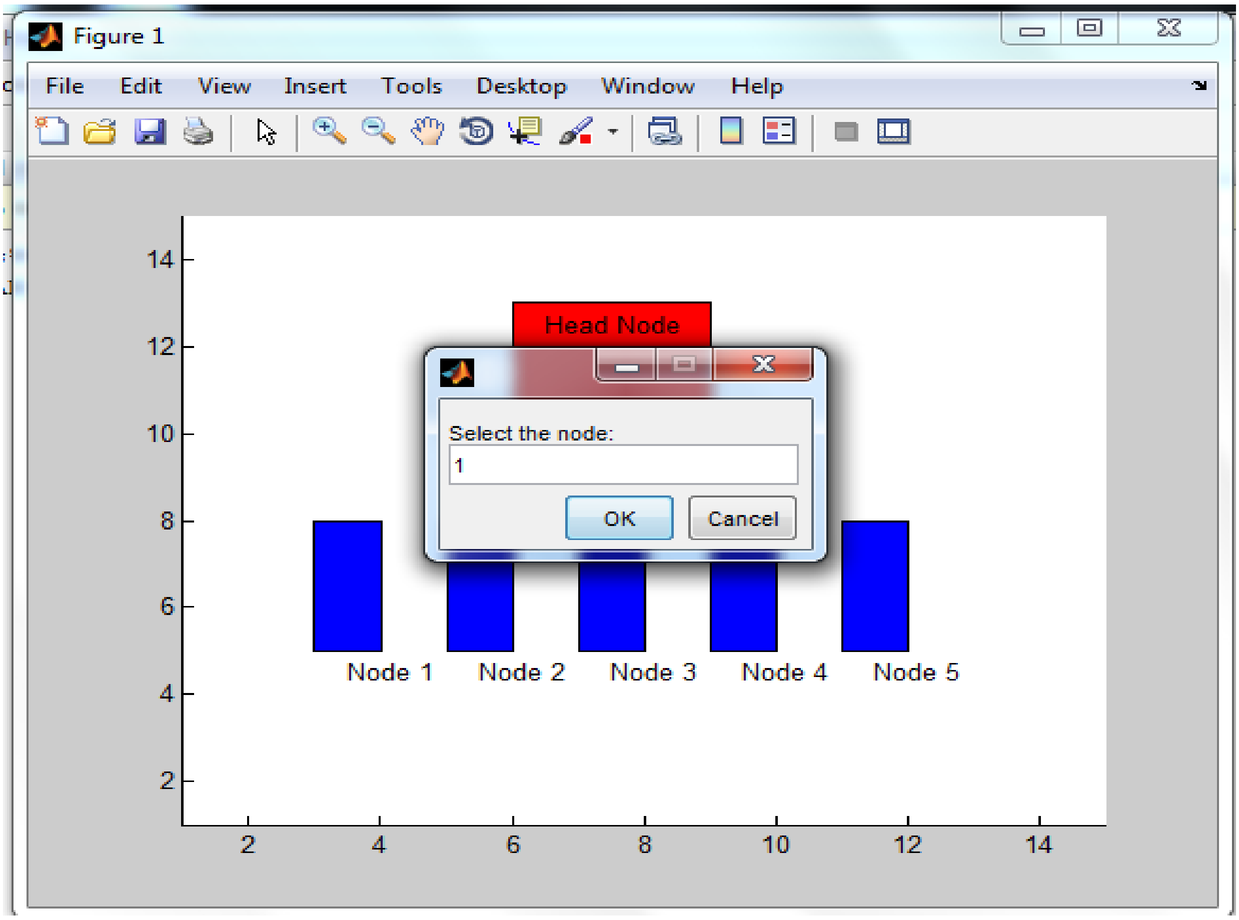


Figure 4.1:Network deployment

Figure 4.1 shows the network deployment with finite number of nodes. Here Head node is the master node i.e. the server to which the other nodes communicate. Nodes 1, Node2, Node 3, Node 4, and Node 5 are the users or the slaves. From these slave nodes, one of them will communicate with master node to read the data. Head node is represented by red color and other communicating nodes are represented by blue color. Head node is the cloud node on which the data is stored and other users read data from the cloud.

4.2.2 Select the communicating node



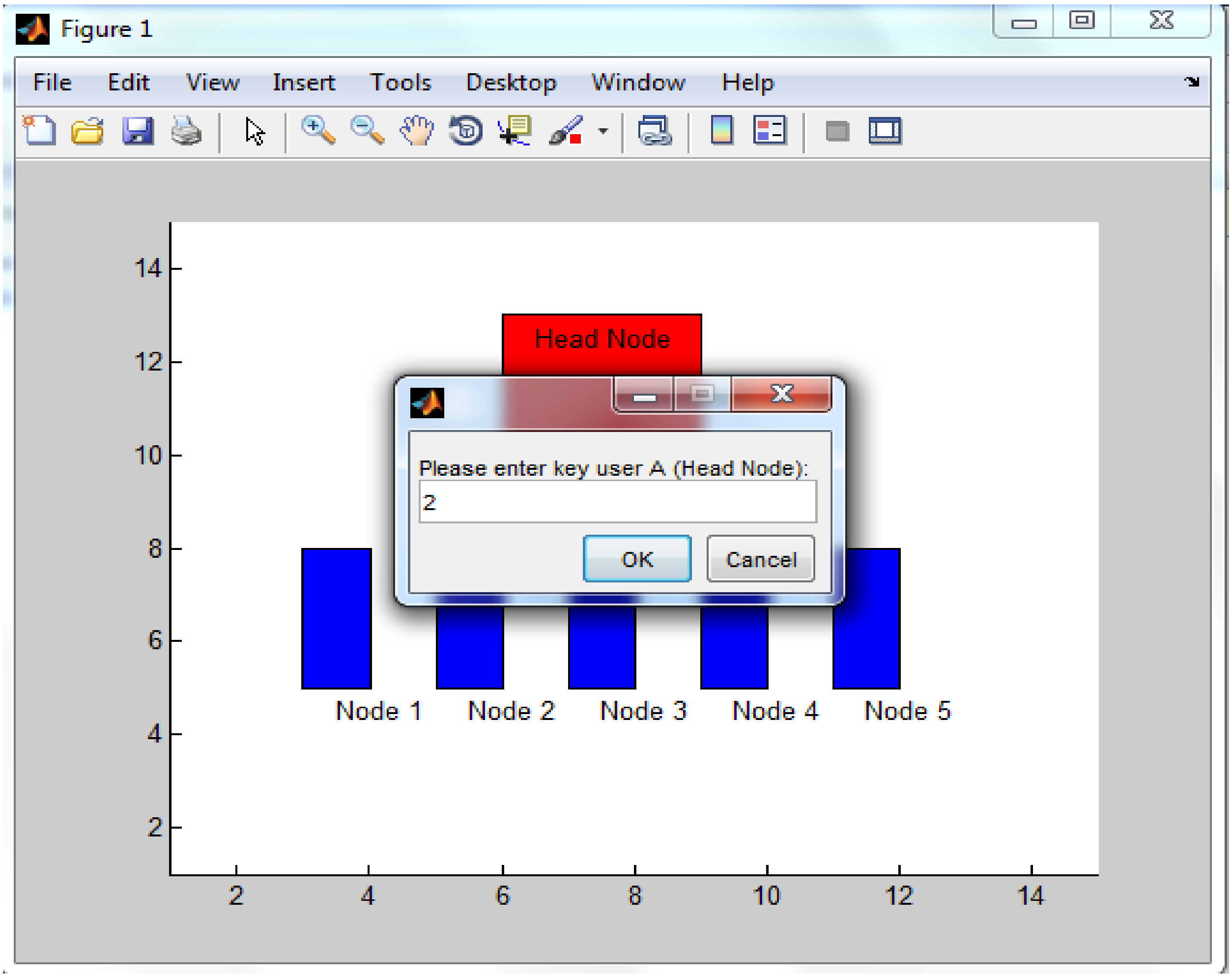
Select the node:

1

Figure 4.2: Select the communicating node

Figure 4.2 shows the selection of communicating node, which will communicate with the head node. Head node represented the server or cloud node and rest nodes are the users. Here node 1 is chosen for communication. When node 1 is chosen that means the node 1 wants to read the data stored on the cloud i.e. the head node.

4.2.3 PIN entered by Head node



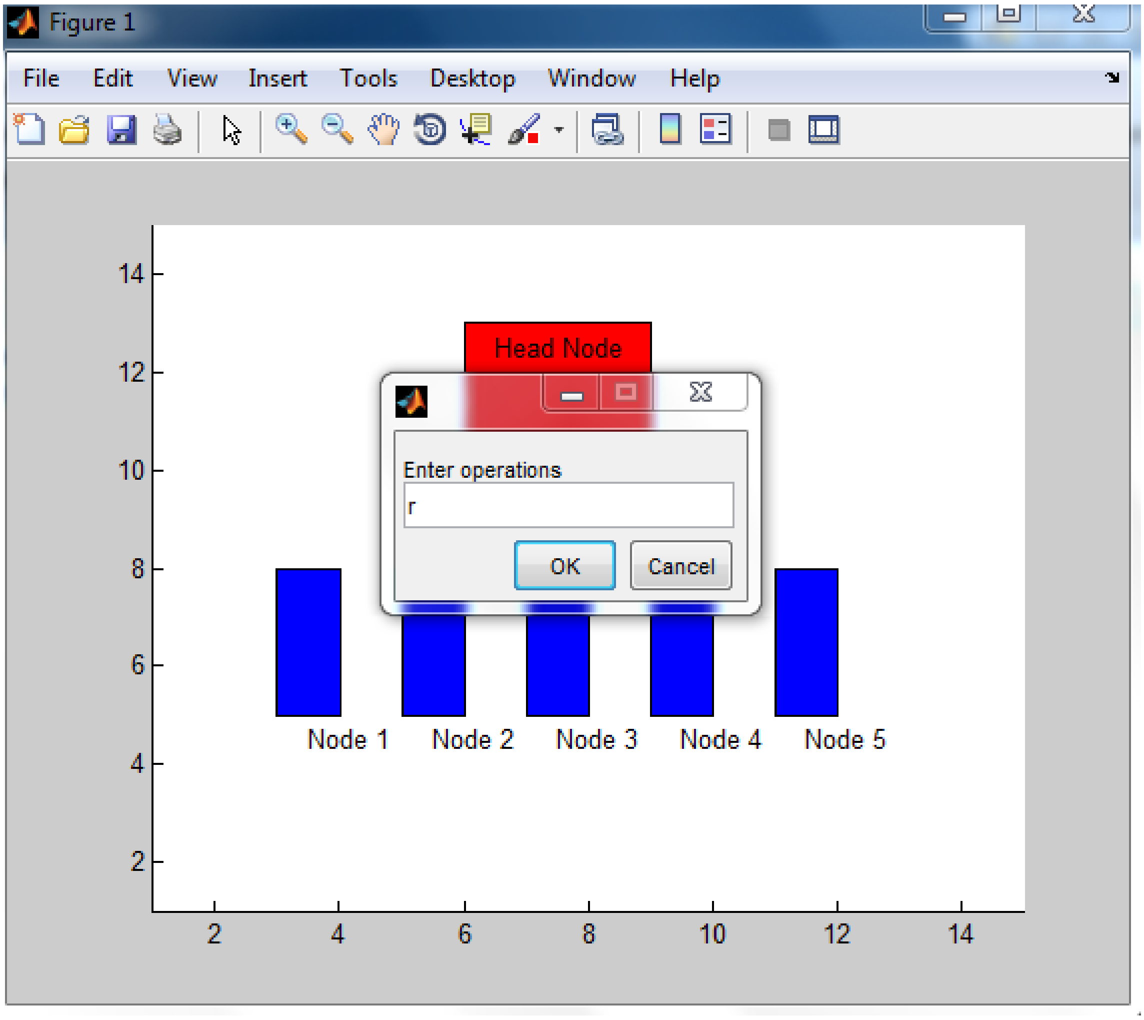
Please enter key user A (Head Node):

2

Figure 4.3: PIN entered by Head node

Figure 4.3 shows, master node enter the PIN. If the PIN entered at both sides is same then channel will be established between head node and node 1 and data will be transferred, else connection will not establish. Firstly, key will be selected by head node.

4.2.4 Selection of operation



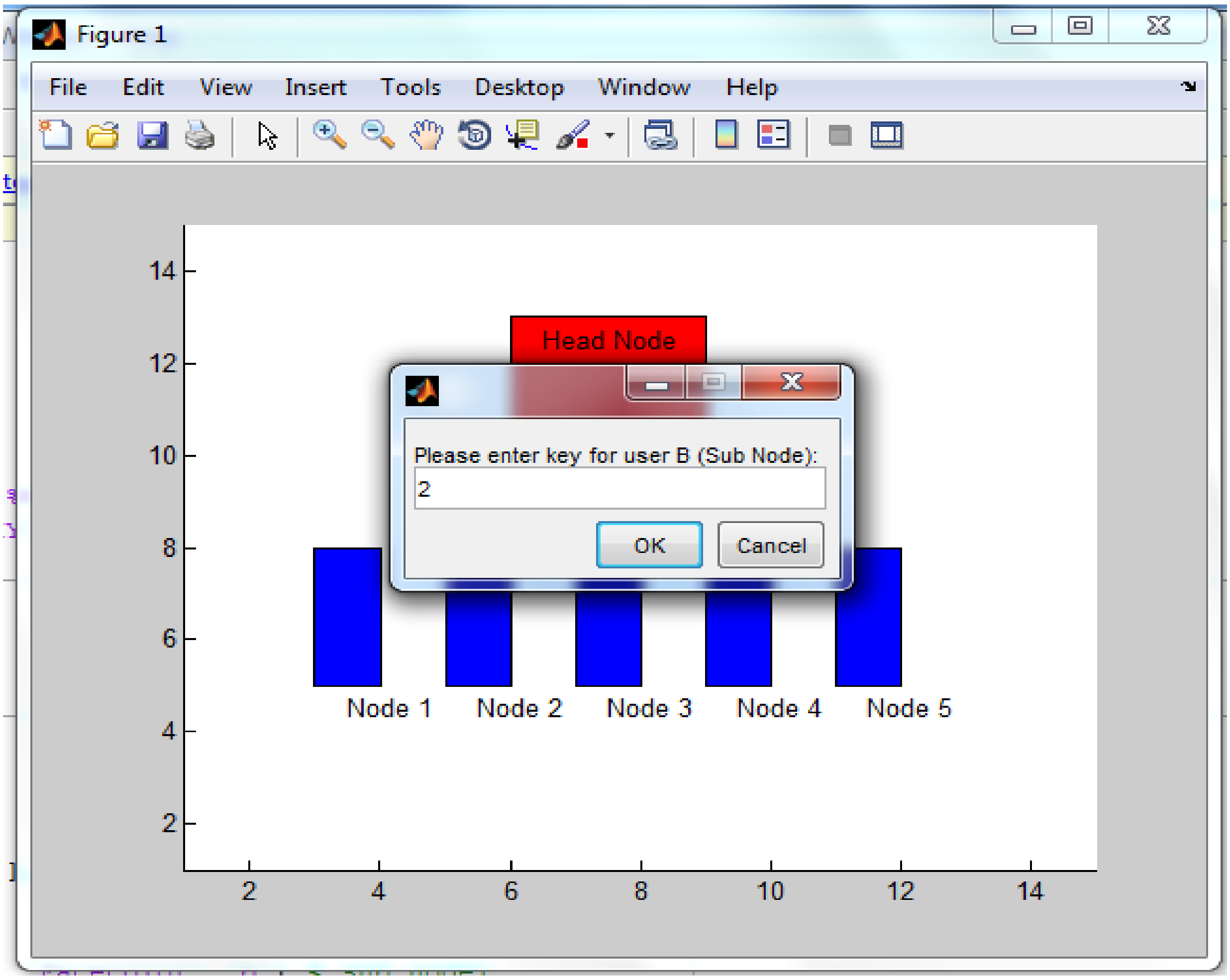
Enter operations

r

Figure 4.4: Operation selected

Figure 4.4 shows the selection of operation. In this operation 'r' is entered which means to read. User nodes are reading data from the head node i.e. the server node.

4.2.5 PIN enetered by node 1



Please enter key for user B (Sub Node):

2

Figure 4.5: PIN entered by node 1

Figure 4.5 shows, user B enter the PIN for successful authentication. Both, user and the server enter the same PIN and they perform the communication. The communication is possible only if both user A and User B enter the same key. If keys entered are not same then connection establishment will be failed.

4.2.6 Path established between Head node and Node 1

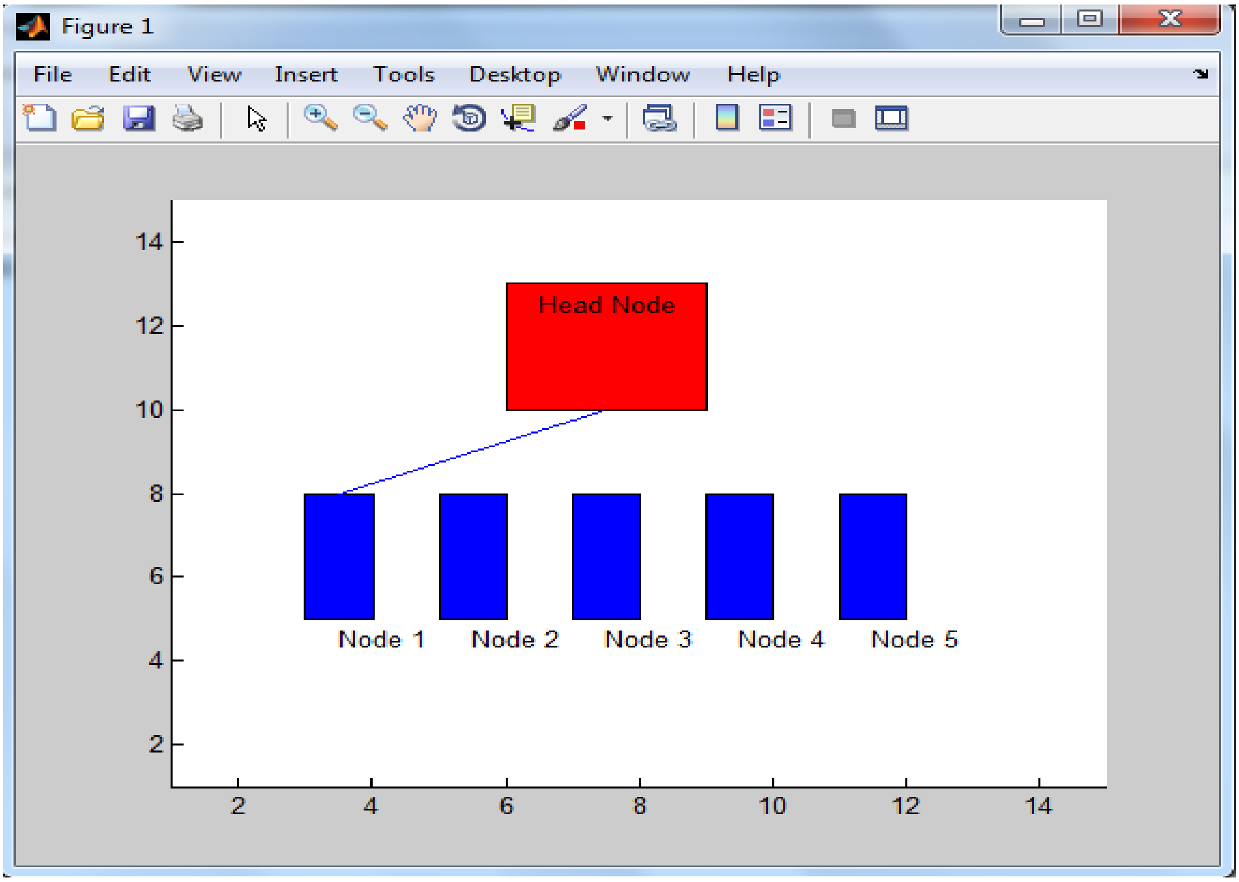


Figure 4.6: Path established between source to destination

Figure 4.5 illustrates that, when entered PIN is matched then path is established between source and destination. This is showing that data will transfer between these two nodes. Path is set for the communication between these nodes from which the data will transfer.

4.2.7 Data transfer between nodes

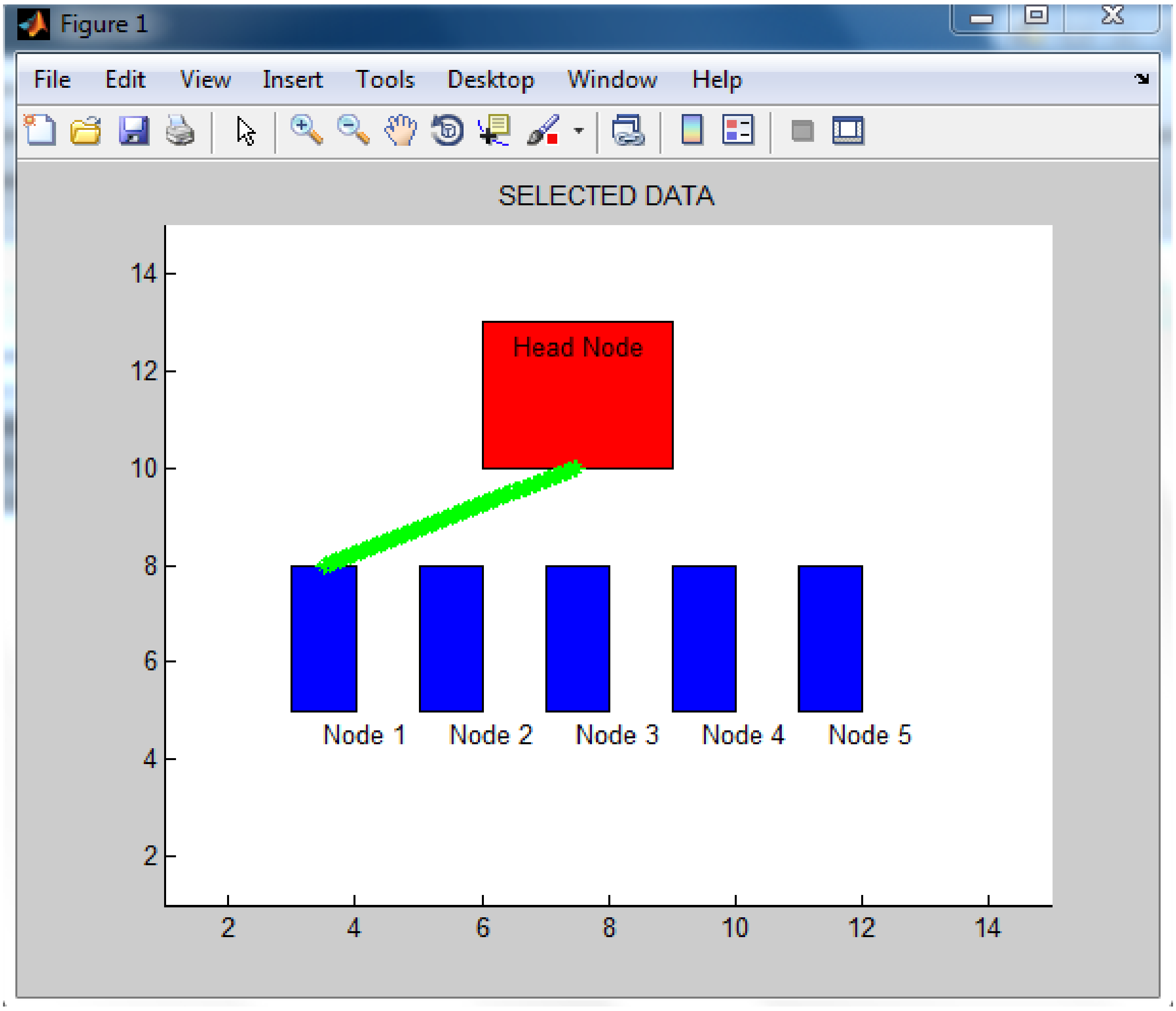


Figure 4.7: Data transfer between nodes

Figure 4.7 shows that the data is transferred from server node to user node after a successful authentication.

Data is transferred to node 1 from head node. But as there is no security of data in this, so new scheme is proposed in which data is more secured.

4.3 Solution implementation

4.3.1 Network deployment

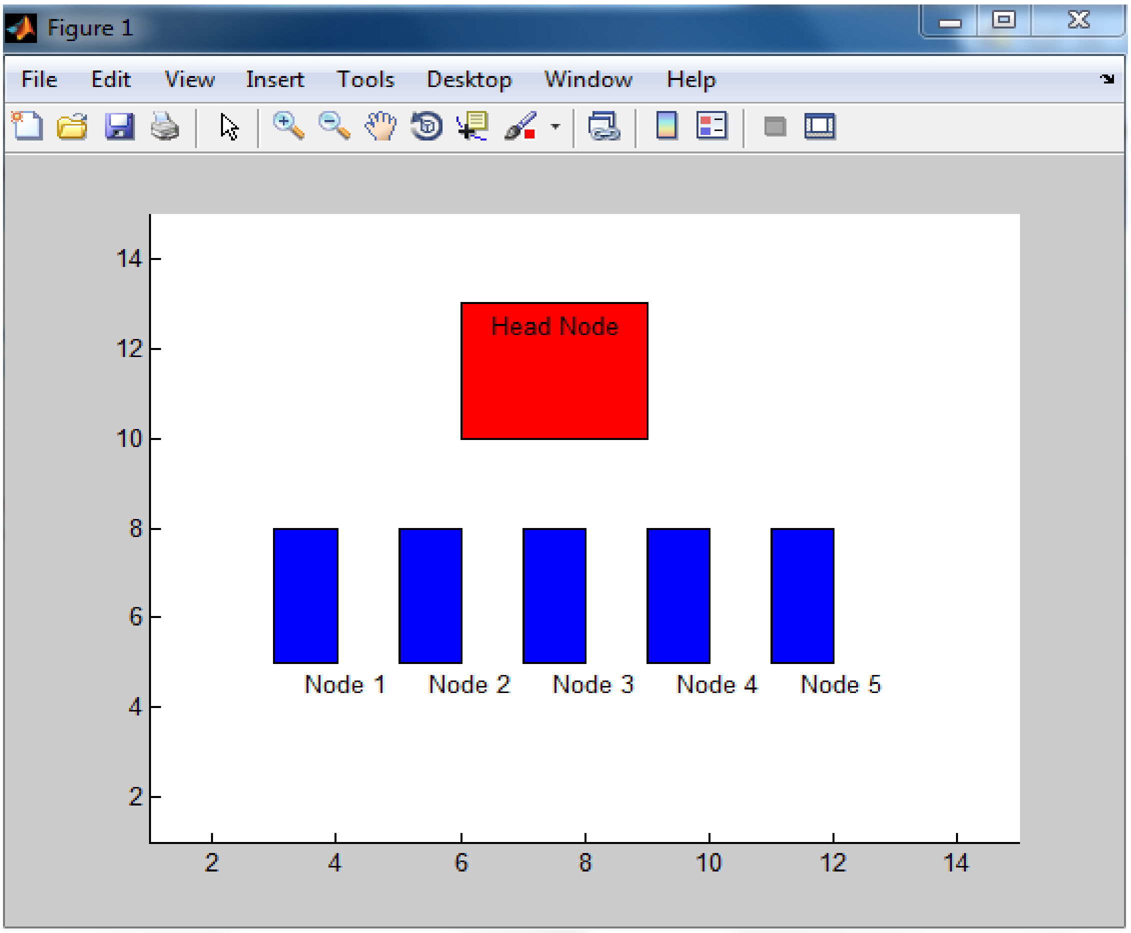
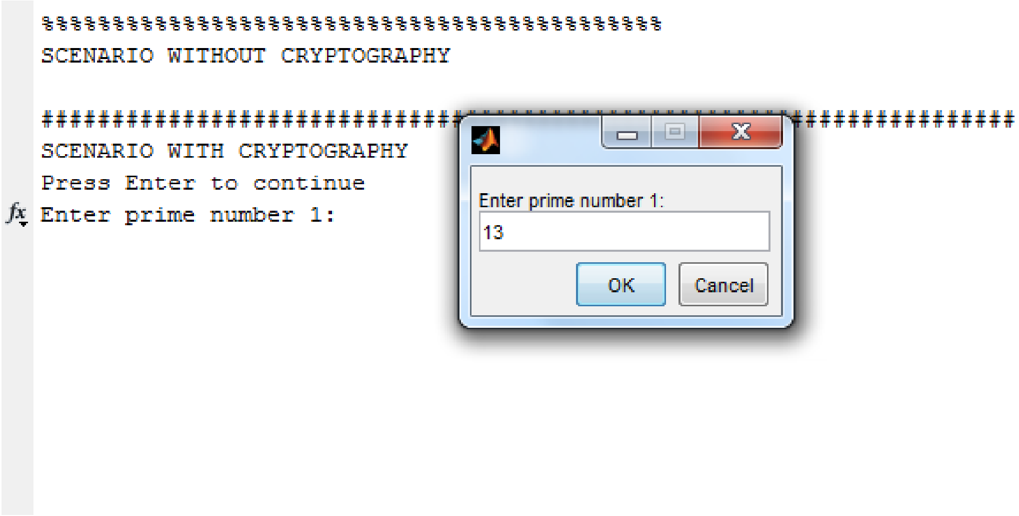


Figure 4.8:Network deployment

Figure 4.7 shows network deployment for implementation of our given solution. Deployment of network is same as before. But in the new method which is the solution to the problem, has different way to transfer data which is more secured. In this, as we are using Diffie-Hellman algorithm to communicate two nodes, so the head node is the master node and the other nodes are the slave nodes. Node selected before i.e. node 1 is slave node and head node is the master node and communication will be in these two.

4.3.2 First prime number is entered



```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
SCENARIO WITHOUT CRYPTOGRAPHY

#####################################################################
SCENARIO WITH CRYPTOGRAPHY
Press Enter to continue
Enter prime number 1:
```

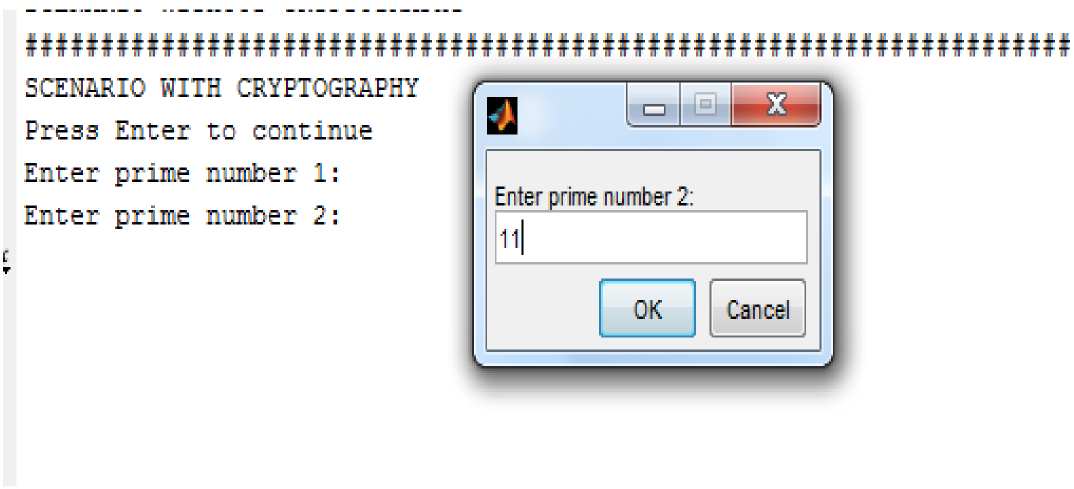
Enter prime number 1:

13

Figure 4.9:First prime number

Figure 4.8 shows, first prime no. is selected and both parties are agreeing on it. This is the prime no. which is used for calculating the secret key by Diffie-Hellman algorithm.

4.3.3 Enter second prime no



```
#####################################################################
SCENARIO WITH CRYPTOGRAPHY
Press Enter to continue
Enter prime number 1:
Enter prime number 2:
```

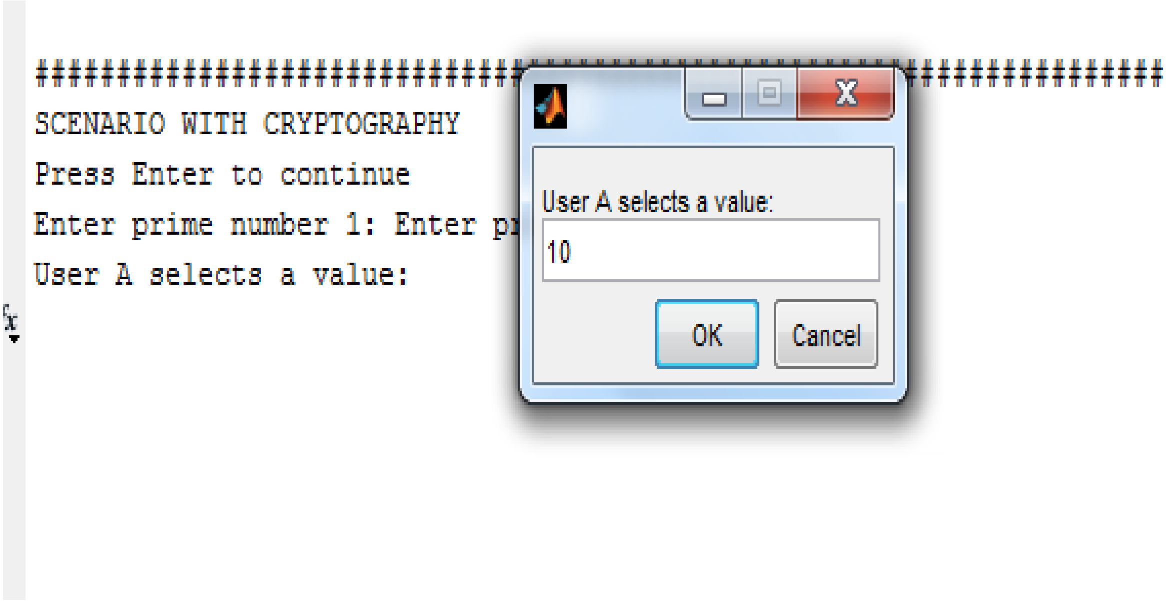
Enter prime number 2:

11

Figure 4.10: Enter second prime number

Figure 4.9 shows, the second prime no. entered. Before selected private numbers, both parties agree on two prime numbers ‘g’ and ‘n’, these numbers become public.

4.3.4 The master node choose their private number



```
##############################                    ################
SCENARIO WITH CRYPTOGRAPHY
Press Enter to continue
Enter prime number 1: Enter pr
User A selects a value:
```

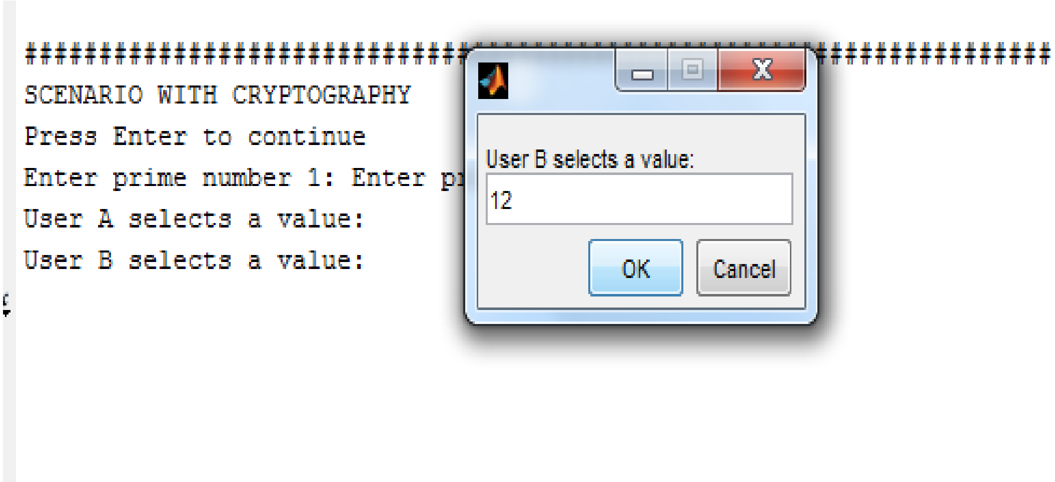
User A selects a value:

10

Figure 4.11: **Master node choose private number**

As illustrated in figure 4.10, the diffie Hellman key establishment procedure starts, the master node which wants to communicate with slave node, both parties select their private numbers. Before selected private numbers, both parties agree on two prime numbers ‘g’ and ‘n’, these numbers become public. This is the random number selected by master node. Private numbers are used to decrypt the data i.e. data while sending is encrypted by using public keys of users and then this data is decrypted at destination by using the private key.

4.3.5 Slave choose their private number



```
##############################                  ################
SCENARIO WITH CRYPTOGRAPHY
Press Enter to continue
Enter prime number 1: Enter pr
User A selects a value:
User B selects a value:
```

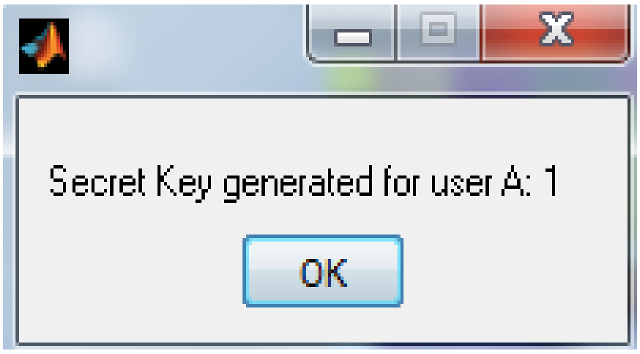
User B selects a value:

12

Figure 4.12: Slave node choose private numbers

As shown in figure 4.12, the destination node chooses their private number. This is the random number selected by user B i.e. the slave node which becomes the private key for user B. This is the private key which is used to decrypt the data when reached at destination.

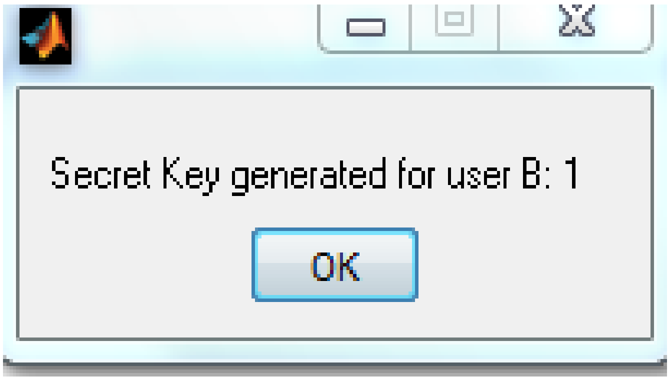
4.3.6 Keys Generated



Secret Key generated for user A: 1

Figure 4.13: Secret key generated for user A

By apply Diffie-Hellman algorithm, secret key for User A is generated. This key is not known to any of the user i.e. sender and receiver. This key is generated at the time of sending data and the key is generated on the basis of private keys, public keys and prime numbers decided by both the users.



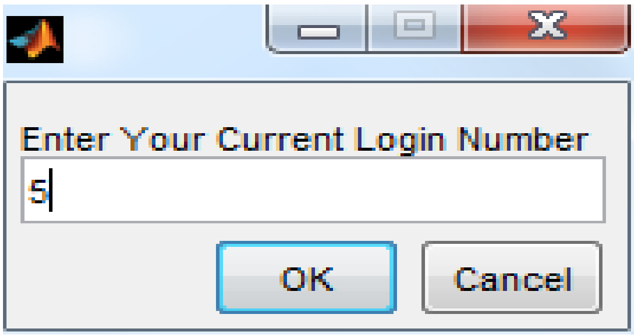
Secret Key generated for user B: 1

Figure 4.14:Secret Key generated for user B

This is the key generated for User B.

Keys generated for User A and User B are same, so successful communication will be establish between Head node i.e. master node and slave node. This key is always different every time the key is generated. This is the secret key which is not known to anyone i.e. master, slave or any intruder because this key is generated on run time and is always new basis on the private keys and random numbers selected by master node and the slave node.

4.3.7 OTP generation

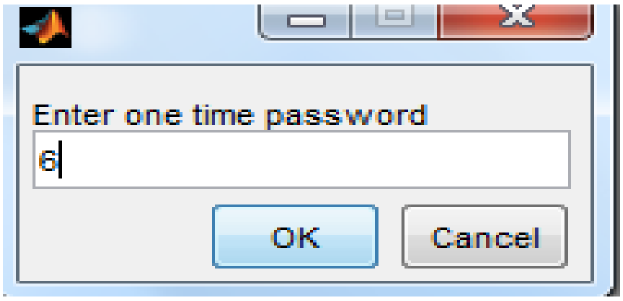


Enter Your Current Login Number

5

Figure 4.15: Current Login number of user

This is the login no. of user, i.e. how many times a user is login. This record of logging user is saved at server. Server keeps the login record of every legitimate user.



Enter one time password

6

Figure 4.16: OTP generated

This is the OTP generated by calculating the secret key and current login number. This is the one time password which is used only for once and is different every time. This is not known to anyone as this is generated on runtime.

4.3.8 Secure Channel Establishment between master node and slave node

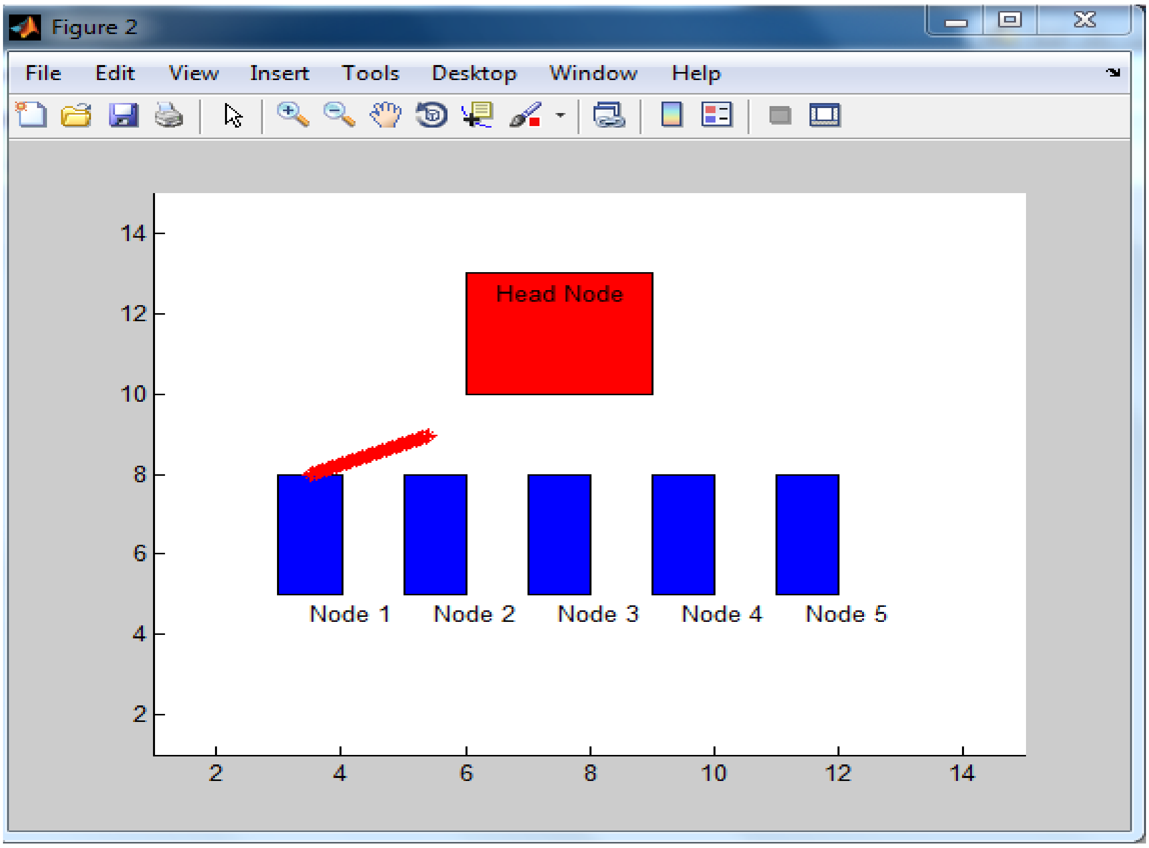


Figure 4.17: **Channel establishment**

Figure 4.13 shows that node 1 is ready to communicate with head node. When keys generated at sides master and slave then secure channel is establish between them and they are ready to transfer data.

4.3.9 Data transfer from master node to slave node

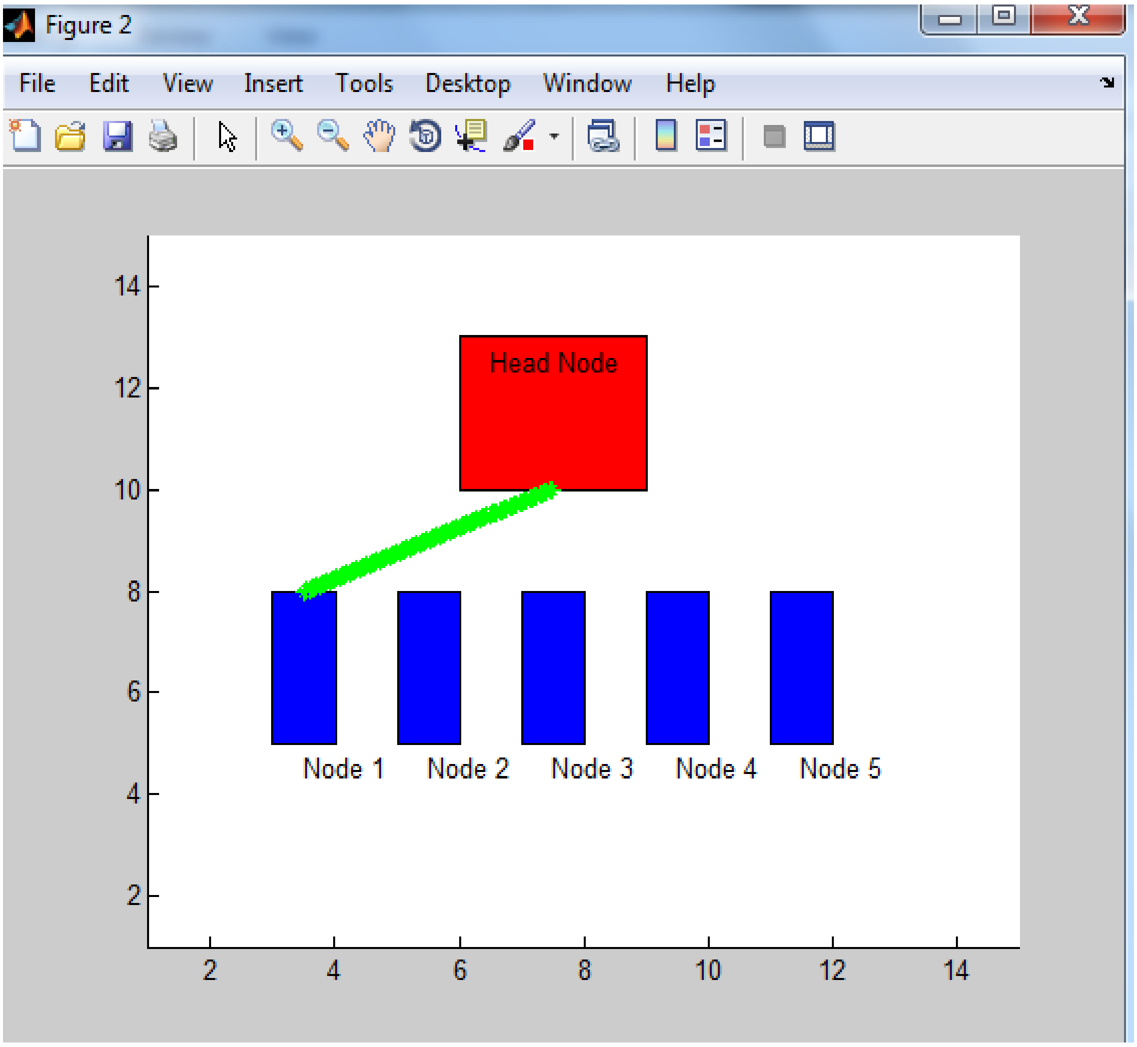


Figure 4.18 Data Transfer from master node to slave node

As shown in figure 4.14, green line indicates that successful data is transferred from master to slave node. This is possible only when keys which are generated before from diffie Hellman are same. When K1 and K2 are same i.e. keys are matched data can be successfully send from source to destination.

4.4 Results

4.4.1 Comparison graph of Delay

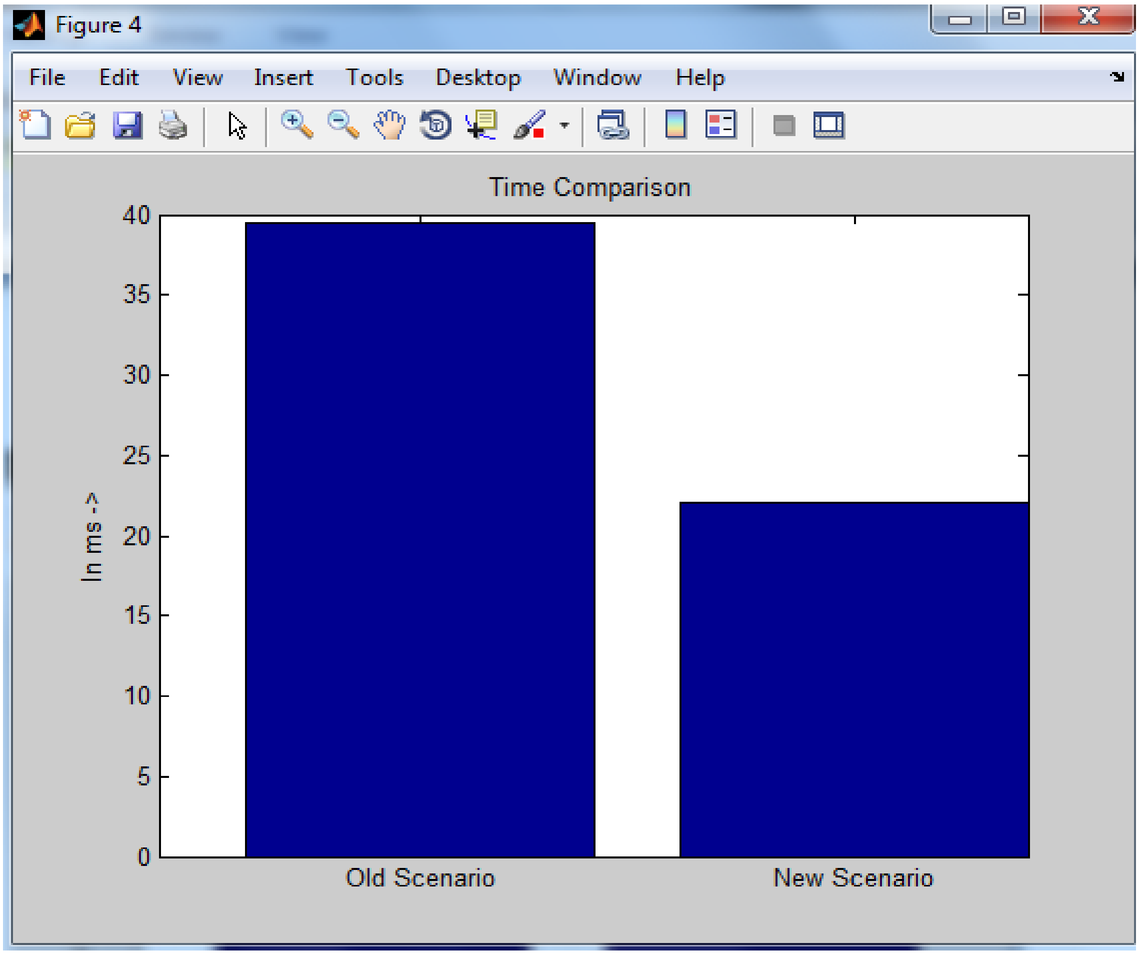


Figure 4.19: Comparison graph of delay

As shown in figure 4.16, the comparison between previous and proposed approach is shown in terms of delay. The delay in previous technique is increasing, when numbers of exchange messages are increased. In the proposed approach the delay is less due to increasing the number of message.

4.4.2 Comparison graph of Space

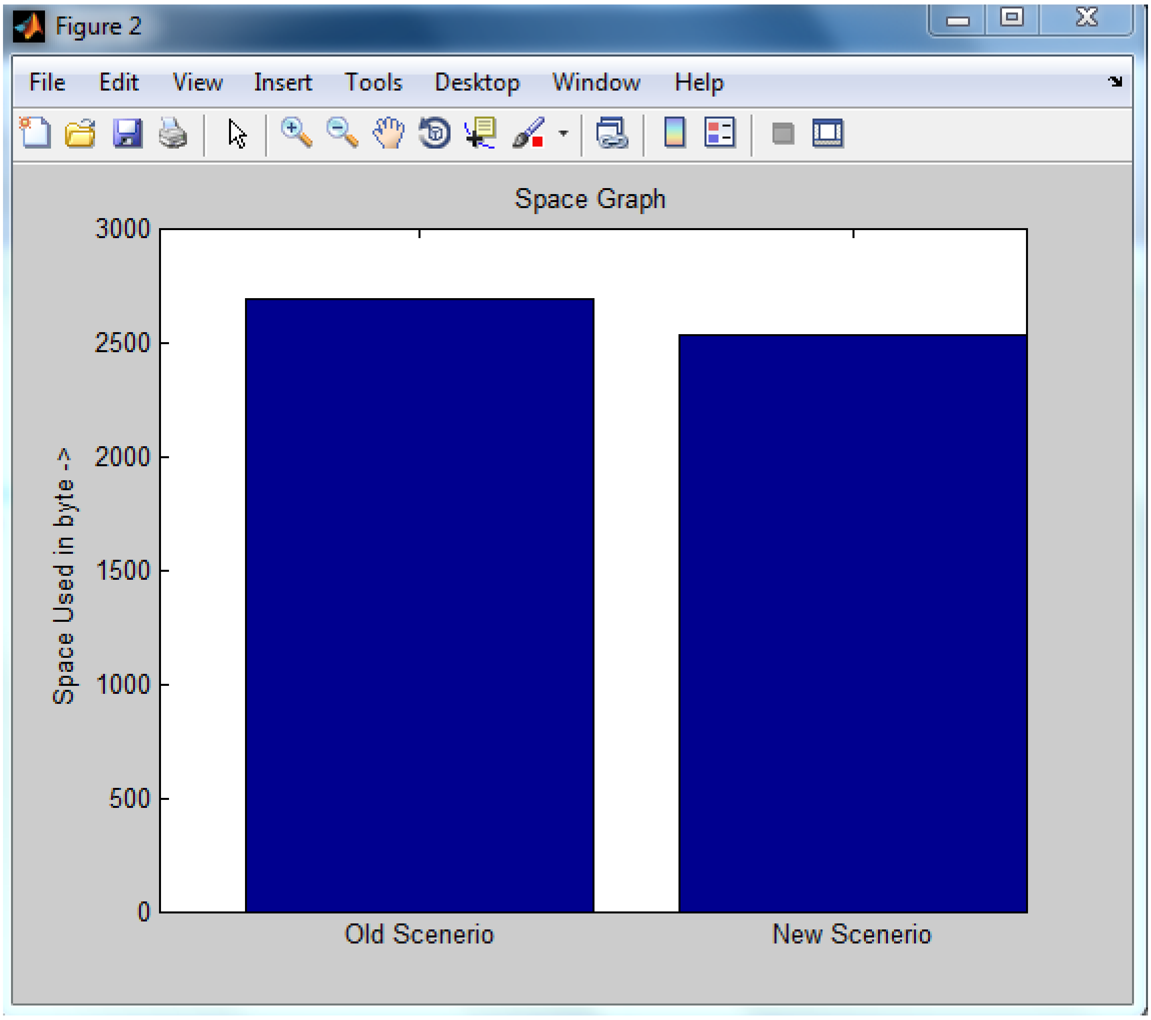


Figure 4.20: Comparison graph of Space

This is the comparison graph of space which is showing that the old scenario is covering more space than the new method which is proposed. This comparison shows better results.

4.4.3 Comparison graph of Probability

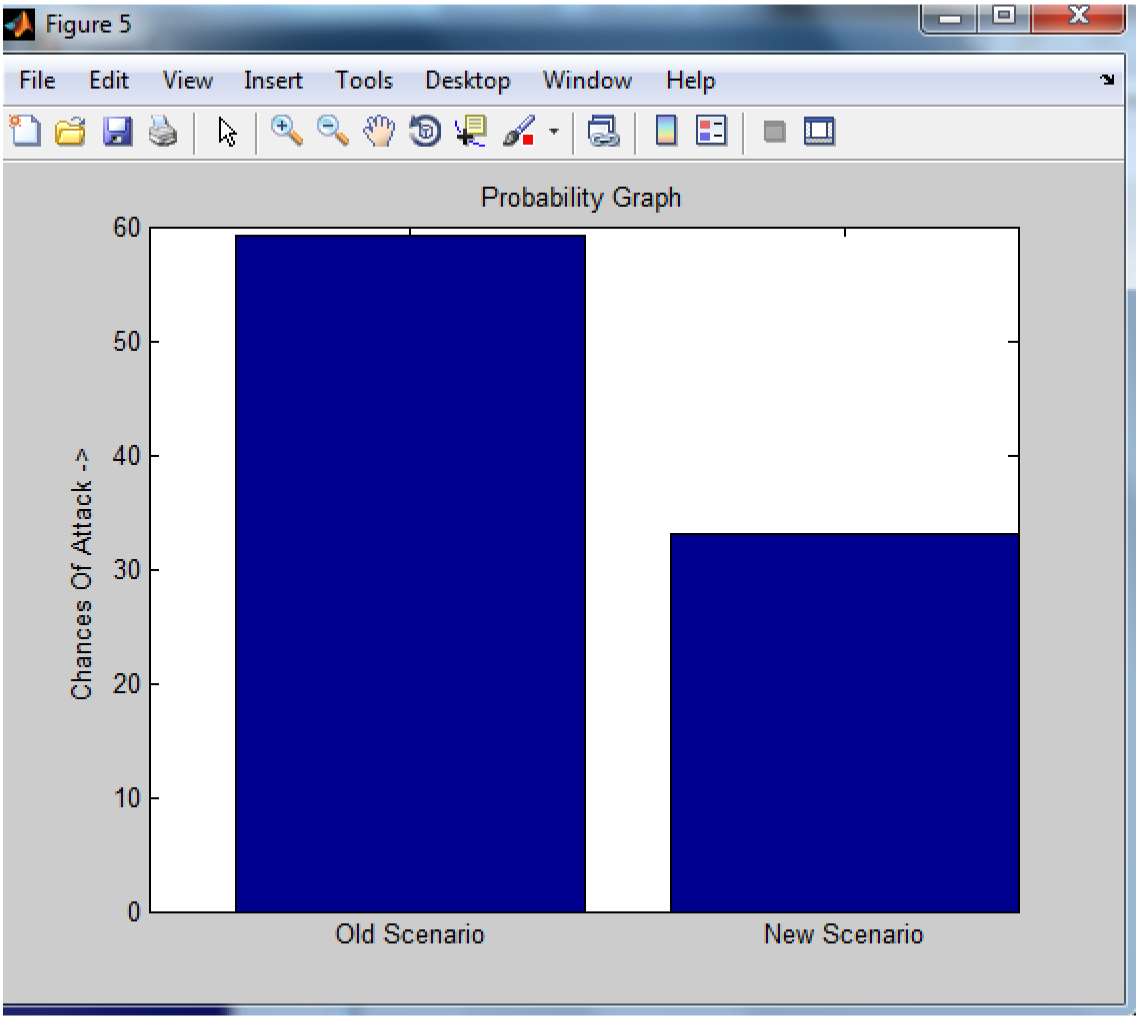


Figure 4.21: Comparison graph of probability

This is the probability graph which shows that by using old scenario more resources are used and in new scenario number of resources used are less as compared to old scenario.

Chapter 5

Conclusion and Future Scope

Cloud computing is the environment which provides on-demand & convenient access of the network to a computing resources like storage, servers, applications, networks and the other services which can be released minimum efficiency way. In this user can store their data and use different services and pay according to those services. The main factor is security that how we can store our data while storing into the cloud. In this thesis, we reviewed two most popular techniques for cloud data encryption. These techniques are full disk encryption and fully homomorphic encryption. In this work, we find that fully homomorphic encryption technique is more efficient than full disk encryption. But the main problem exists in fully homomorphic encryption is of key management and key sharing which reduces the reliability of the scheme. For key management and key sharing, enhancement is been proposed in the encryption scheme and enhancement is based on Diffie-hellman algorithm and HMAC and OTP is generated on the basis of secret key generated from diffie-hellman algorithm. This algorithm create session key between user and cloud. Each time new key is generated between two before communication. This reduces the time takes place in management and sharing of keys and secure channel is established between both i.e. user and the cloud service provider. The simulation shows that proposed enhancement is more efficient and reliable than the existing one. In future we work on to provide enhancement that further improve security features of fully homomorphic encryption scheme. In future role based access control scheme will be introduced with the proposed scheme to reduce chances of attack in cloud architecture

Chapter 6

References

Ankur Mishra, R. M. (2013). Cloud Computing Security. *International Journal on Recent and Innovation Trends in Computing and Computation, pp 36-39* .

Bhavna Makhija, V. G. (2013). Enhanced Data Security in Cloud Computing with Third Party Audito. *International Journal of Advanced Research in Computer Science and Software Engineering, pp 341-345* .

Chimere Barron, H. Y. (2013). Cloud Computing Security Case Studies. *Proceedings of the World Congress on Engineering* .

Dawn Song, E. S. (2012). Cloud Data Protection for the Masses. *IEEE Computer Society, pp 39-45* .

Deepanchakaravarthi Purushothaman, D. A. (2012). An Approach for Data Storage Security in Cloud Computing. *IJCSI International Journal of Computer Science Issues* .

Deyan Chen, H. Z. (2012). Data Security and Privacy Protection Issues in Cloud Computing. *International Conference on Computer Science and Electronics Engineering, pp 647-651* .

Dian-Yuan Han, F.-q. Z. (2012). Applying Agents to the Data Security in Cloud Computing. *International Conference on Computer Science and Information Processing(CSIP), pp 1126-1128* .

Dr Nashaat el-Khameesy, H. A. (2012). A Proposed Model for Enhancing Data Storage Security in Cloud Computing Systems. *Journal of Emerging Trends in Computing and Information Sciences* .

Gentry, C. (2009). A Fully Homomorphic Encryption Scheme.

John Harauz, L. M. (2009). Data Security in the World of Cloud Computing. *IEEE* .

Kaur, S. (2012). Cryptography and Encryption In Cloud Computing. *VSRD-IJCSIT, Vol. 2 (3), 2012, 242-249, pp 242-249* .

Marten van Dijk, C. G. (2010). Fully Homomorphic Encryption over the Integers.

Pandey, V. (2013). Securing the Cloud Environment Using OTP. *International Journal of Scientific Research in Computer Science and Engineering* .

Punithasurya K, E. D. (2013). A Novel Role Based Cross Domain Access Control Scheme for Cloud Storage. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 3, March 2013, pp 94* .

Sanjoli Singla, J. S. (2013). Cloud Data Security using Authentication and Encryption Technique. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 7, July 2013, pp 2232-2235* .

Sean Carlin, K. C. (2011). Cloud Computing Security. *International Journal of Ambient Computing and Intelligence, pp 14-19* .

Shui Han, J. X. (2011). Ensuring Data Storage Through A Novel Third Party Auditor Scheme in Cloud Computing. *IEEE computer science & Technology, pp 264-268* .

Young-Gi Min, H.-J. S.-H. (2012). Cloud Computing Security Issues and Access Control Solutions. *Journal of Security Engineering, pp 135-140* .

Zhiwei Yu, C. W. (2012). A novel watermarking method for software protection in the cloud. 409-430.

Chapter 7

Appendix

FHE	Fully Homomorphic Encryption
FDE	Full Disk Encryption
SaaS	Software as a Service
PaaS	Platform as a Service
IaaS	Infrastructure as Service
CSP	Cloud Service Provider