STRING COMPRESSION TECHNIQUE WITH MODIFIED AES ENCRYPTION

Manish Kumar Aery
Assistant Professor IET Bhaddal Department of MCA
manish.im66@ietbhaddal.edu.in

ABSTRACT

In this work of research, the three layered architecture is been developed with introduction of a strong encryption algorithm called AES RINJDAEL. The three layered architecture is been developed for the purpose of high security which is needed to secure the highly confidential database systems. Each level have its own significance. All the parameters of security is fulfilled by the approach on four levels. No malicious users can interfere with the system, attacks such as brute force are successfully overcomes. The AES RINJDAEL will save all the data in form of private and public key along the network saving the data from man in middle attacks. Below is the brief summary about what thesis outlines.

INTRODUCTION

Cloud computing is the service that provide the user the power of sharing computer resources. Instead of running applications yourself, the applications runs on the cloud, a shared data center, you have to just plug in like a utility. It is the type of internet based computing where services like servers, storage, applications are delivered to a computer or device through the internet. In cloud computing one does not need to buy any application, just have to pay according to the usage and need not to worry about storage space and computation speed of device or computer. It is very useful for micro and medium organizations because it is fast to get started; it costs less and provides more reliability, scalability and security.

In the traditional model of computing, both data and software are fully contained on the user's computer; in cloud computing, the user's computer may contain almost no software or data (perhaps a minimal operating system and web browser, display terminal for processes occurring on a network). Cloud computing is based on five attributes: multi-tenancy (shared resources), massive scalability, elasticity, pay as you go, and self-provisioning of resources, it makes new advances in processors, Virtualization technology, disk storage, broadband Internet connection, and fast, inexpensive servers have combined to make the cloud a more compelling solution.

The main attributes of cloud computing is illustrated as follows:

- Multi-tenancy (shared resources): Cloud computing is based on a business model in which resources are shared (i.e., multiple users use the same resource) at the network level, host level, and application level.
- Massive scalability: Cloud computing provides the ability to scale to tens of thousands of systems, as well as the ability to massively scale bandwidth and storage space
- Elasticity: Users can rapidly increase and decrease their computing resources as needed.
- Pay as you used: Users to pay for only the resources they actually use and for only the time they require them.
- Self-provisioning of resources: Users self-provision resources, such as additional systems (processing capability, software, storage) and network resources. [12]

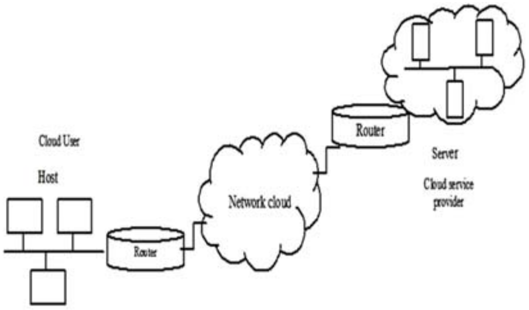


Figure 1.1 Cloud computing

1.1.1ARCHITECTURE OF CLOUD COMPUTING

Cloud computing architecture specifies the components and subcomponents required in delivery of cloud services. These components typically consist of front end platform, back end platform, cloud based delivery, and network and these components make up the cloud computing architecture. The front end platform includes fat client, thin client, and mobile

Devices, systems etc. the back end platform includes server, storages. Cloud architecture,the systems architecture of the software systems involved in the delivery of cloud computing, typically involves multiple cloud components communicating with each other over a loose coupling mechanism such as a messaging queue. Elastic provision implies intelligence in the use of tight or loose coupling as applied to mechanisms such as these and others.

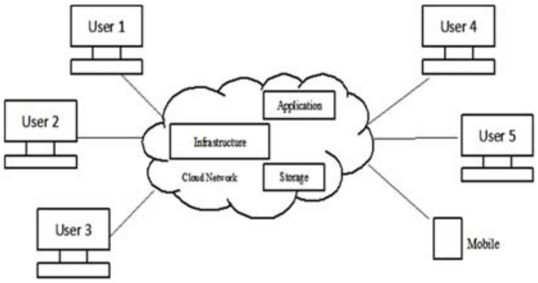


Figure 1.2 Cloud Computing Architecture

1.1.2 SERVICE MODELS

Cloud computing providers offer their services according to several fundamental models:

Infrastructure as a service (IaaS)

IaaS clouds often offer additional resources such as a virtual-machine disk image library, raw block storage, and file or object storage, firewalls, load balancers, IP addresses, Virtual local area networks(VLANs), and software bundlesIaaS-cloud providers supply these resources on-demand from their large pools installed in data centers. For wide area connectivity, customers can use either the Internet or carrier clouds (dedicated virtual private networks).

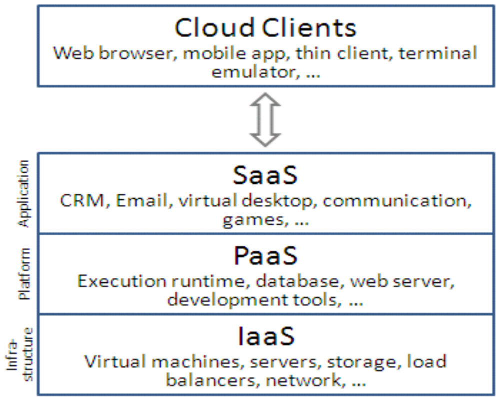

Cloud Clients
Web browser, mobile app, thin client, terminal emulator, ...
Application
SaaS
CRM, Email, virtual desktop, communication, games, ...
Platform
PaaS
Execution runtime, database, web server, development tools, ...
Infra-structure
IaaS
Virtual machines, servers, storage, load balancers, network, ...

Figure 1.3 Service model

Platform as a service (PaaS)

In the PaaS models, cloud providers deliver a computing platform, typically including operating system, programming language execution environment, database, and web server. Application developers can develop and run their software solutions on a cloud platform without the cost and complexity of buying and managing the underlying hardware and software layers. With some PaaS offers like Microsoft Azure and Google app engine, the underlying computer and storage resources scale automatically to match application demand so that the cloud user does not have to allocate resources manually. The latter has also been proposed by an architecture aiming to facilitate real-time in cloud environments.

Software as a service (SaaS)

In the business model using software as a service (SaaS), users are provided access to application software and databases. Cloud providers manage the infrastructure and platforms that run the applications. SaaS is sometimes referred to as "on-demand software" and is usually priced on a pay-per-use basis. SaaS providers generally price applications using a subscription fee.

1.1.3 DEPLOYMENT MODEL

Private cloud

Private cloud is cloud infrastructure operated solely for a single organization, whether managed internally or by a third-party, and hosted either internally or externally.Undertaking a private cloud project requires a significant level and degree of engagement to virtualize the business environment, and requires the organization to reevaluate decisions about existing resources. When done right, it can improve business, but every step in the project raises security issues that must be addressed to prevent serious vulnerabilities.

Public cloud

A cloud is called a "public cloud" when the services are rendered over a network that is open for public use. Public cloud services may be free or offered on a pay-per-usage model.Generally, public cloud service providers like Amazon AWS, Microsoft and Google own and operate the infrastructure at their data center and access is generally via the Internet.

Hybrid cloud

Hybrid cloud is a composition of two or more clouds (private, community or public) that remain distinct entities but are bound together, offering the benefits of multiple deployment models.

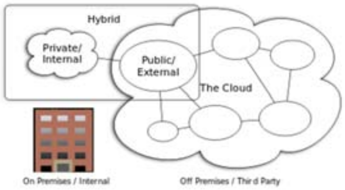


Figure 1.4 Deployment Model

1.1.4. CLOUD COMPUTING FEATURES

Cloud computing brings an array of new features compared to any other computing paradigms. These are briefly described in this section [31].

- **Scalability and On-Demand Services -** Cloud computing provides resources and services for users on demand. The resources are scalable over several data centers.
- **Quality of Service (QoS) -** Cloud computing can guarantee QoS for users in terms of hardware or CPU performance, bandwidth, and memory capacity.
- **User-Centric Interface -** Cloud interfaces are location independent and they can be accessed by well established interfaces such as Web services and Web browsers.
- **Autonomous System -** Cloud computing systems are autonomous systems managed transparently to users. However, software and data inside clouds can be automatically reconfigured and consolidated to a simple platform depending on user's needs.
- **Pricing -** Cloud computing does not require upfront investment. No capital expenditure is required. Users may pay and use or pay for services and capacity as they need them.

1.1.5. CLOUD COMPUTING CHALLENGES

The new paradigm of cloud computing provides an array of benefits and advantages over the previous computing paradigms and many organizations are migrating and adopting it. However, there are still a number of challenges, which are currently addressed by researchers, academicians and practitioners in the field [31].

a. Performance

The major issue in performance can be for some intensive transaction-oriented and other data intensive applications, in which cloud computing may lack adequate performance. Also, users who are at a long distance from cloud providers may experience high latency and delays.

b. Security and Privacy

Companies are still concerned about security when using cloud computing. Users are worried about the vulnerability to attacks, when information and critical IT resources are outside the firewall.

c. Control

A quantity of IT wings or departments are concerned because cloud computing providers have a full control of the platforms. Cloud computing providers typically do not design platforms for specific companies and their business practices.

d. Bandwidth Costs

Cloud computing, companies can save money on hardware and software; however they could incur higher network bandwidth charges. Bandwidth cost may be low for smaller Internet-based applications, which are not data intensive, but could significantly grow for data-intensive applications.

e. Reliability

Cloud computing still does not always offer round the clock reliability. There were cases where cloud computing services suffered few hours' outages. In the present and future days to expect more cloud computing providers, richer services, established standards and best practices.

1.2CLOUD COMPUTING SECURITY ISSUES

In distributed cloud database another issue that we consider is security. High growth in the field of networks leads a common problem for changing of the data at very fast rate. Hence it will be appropriate if we duplicate the data, hence our major concern is that our information must be protected while transferring the important information like banking transaction for this purpose we use many encryption techniques and also protect the confidential data from the unauthorized use. Security means protection of information and information system from unauthorized access, modification and misuse of information or destruction. Moreover distributed system poses four main components t hat are security authentication (creates password), authorization (prove his identity), and encryption. Hence encryption plays very vital role in the security. Hence we use some encryption techniques that will be helpful in dealing with some of the security issues

There is a number of security issues/concerns associated with cloud computing but these issues fall into two broad categories

i) Cloud provider

ii) Client

The responsibility goes both ways, however: the provider must ensure that their infrastructure is secure and that their clients' data and applications are protected while the user must ensure that the provider has taken the proper security measures to protect their information, and the user must take measures to use strong passwords and authentication measures.

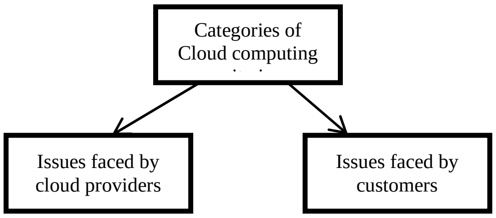

Categories of Cloud computing
Issues faced by cloud providers
Issues faced by customers

Figure 1.5 Categories of cloud computing security issues

One method to solve this problem is Encryption. Encryption can be done on three ways:

1. Server encryption – It also means provide security to data at rest. It provides security to data that is stored on the server from one or more threat like losing data.
2. Client Encryption – It also means providing security to moving data i.e., encrypting data while transmission. To encrypt data before sending it.
3. Proxy Encryption - Encryption is done using a third part alliance that either resides on your network or public or private cloud.

1.3CLIENT SERVER FRAMEWORK AND CRYPTOGRAPHY

1.3.1 CLIENT SERVER ARCHITECTURE

Client-server architecture (client/server) is network architecture in which each computer or process on the network is either a client or a server. Servers are powerful computers or processes dedicated to managing disk drive (file servers), printers (print servers), or network traffic (network servers). Clients are Personal computers or workstations on which users run applications. Clients rely on servers for resources, such as files, devices, and even processing power. Another type of network architecture is known as peer-to-peer architecture because each node has equivalent responsibilities. Both client/server and peer-to-peer architectures are widely used and each has unique advantages and disadvantages. Client-server architectures are sometimes called two-tier architectures.

1.3.2 CRYPTOGRAPHY

The art of protecting information by transforming it (encrypting it) into an unreadable format, called cipher text. Only those who possess a secret *key* can decipher (or decrypt) the message into plain text. Encrypted messages can sometimes be broken by cryptanalysis, also called codebreaking, although modern cryptography techniques are virtually unbreakable.

As the Internet and other forms of electronic communication become more prevalent, electronic security is becoming increasingly important. Cryptography is used to protect e-mail messages, credit card information and corporate data.

Cryptography is closely related to the disciplines of cryptology and cryptanalysis. Cryptography includes techniques such as microdots, merging words with images and other ways to hide information in storage or transit. However, in today's computer-centric world, cryptography is most often associated with scrambling plaintext (ordinary text, sometimes referred to as clear text) into cipher text (a process called encryption), then back again (known as decryption). Individuals who practice this field are known as cryptographers.

The main feature of the encryption/decryption program implementation is the generation of the encryption key. Now a day, cryptography has many commercial applications. If we are protecting confidential information then cryptography is provide high level of privacy of individuals and groups. However, the main purpose of the cryptography is used not only to provide confidentiality, but also to provide solutions for other problems like: data integrity, authentication, non-repudiation. Cryptography is the methods that allow information to be sent in a secure from in such a way that the only receiver able to retrieve this information. Presently continuous researches on the new cryptographic algorithms are going on. However, it is a very difficult to find out the specific algorithm, because we have already known that they must consider many factors like: security, the features of algorithm, the time complexity and space complexity. [9]

Modern cryptography concerns itself with the following four objectives:

1) **Confidentiality** (the information cannot be understood by anyone for whom it was unintended)

2) **Integrity** (the information cannot be altered in storage or transit between sender and intended receiver without the alteration being detected)

3) **Non-repudiation** (the creator/sender of the information cannot deny at a later stage his or her intentions in the creation or transmission of the information)

4) **Authentication** (the sender and receiver can confirm each other's identity and the origin/destination of the information)

Cryptography systems can be broadly classified into symmetric-key systems that use a single key that both the sender and recipient have, and public key systems that use two keys, a public key known to everyone and a private key that only the recipient of messages uses.These schemes are:

- **Symmetric key algorithm:** In this cryptographic scheme a common key is used for enciphering and deciphering the message.
- **Asymmetric key algorithm:** This cryptographic scheme uses two keys for encryption and decryption known as Public key and Private Key.

1.3.2.1 Public Key Encryption

A cryptographic system uses two keys - a public key known to everyone and a private or secret key known only to the recipient of the message. E.g. When John wants to send a secure message to Jane, he uses Jane's public key to encrypt the message. Jane then uses her private key to decrypt it. An important element to the public key system is that the public and private keys are related in such a way that only the public key can be used to encrypt messages and only the corresponding private key can be used to decrypt them. Moreover, it is virtually impossible to deduce the private key if you know the public key. Public-key systems, such as Pretty Good Privacy (PGP), are becoming popular for transmitting information via the Internet. They are extremely secure and relatively simple to use. The only difficulty with public-key systems is that you need to know the recipient's public key to encrypt a message for him or her. What's needed, therefore, is a global registry of public keys, which is one of the promises of the new LDAP technology. One of the weaknesses some point out about symmetric key encryption is that two users attempting to communicate with each other need a secure way to do so; otherwise, an attacker can easily pluck the necessary data from the stream. In November 1976, a paper published in the journal IEEE Transactions on Information Theory, titled "New Directions in Cryptography," addressed this problem and offered up a solution: **public-key encryption**.

Also known as **asymmetric-key** encryption, public-key encryption uses two different keys at once - a combination of a private key and a public key. The private key is known only to your computer, while the public key is given by your computer to any computer that wants to communicate securely with it. To decode an encrypted message, a computer must use the public key, provided by the originating computer, and its own private key. Although a message sent from one computer to another won't be secure since the public key used for encryption is published and available to anyone, anyone who picks it up can't read it without the private key. The key pair is based on prime numbers (numbers that only have divisors of itself and one, such as 2, 3, 5, 7, 11 and so on) of long length. This makes the system extremely secure, because there is essentially an infinite number of prime numbers available, meaning there are nearly infinite possibilities for keys. One very popular public-key encryption program is **Pretty Good Privacy (PGP)**, which allows you to encrypt almost anything. In symmetric-key encryption, each computer has a secret key (code) that it can use to encrypt a packet of information before it is sent over the network to another computer. Symmetric-key requires that you know which computers will be talking to each other so you can install the key on each one. Symmetric-key encryption is essentially the same as a secret code that each of the two computers must know in order to decode the information. The code provides the key to decoding the message. Think of it like this: You create a coded message to send to a friend in which each letter is substituted with the letter that is two down from it in the alphabet. So "A" becomes "C," and "B" becomes "D". You have already told a trusted friend that the code is "Shift by 2". Your friend gets the message and decodes it. Anyone else who sees the message will see only nonsense. The same goes for computers, but, of course, the keys are usually much longer. The first major symmetric algorithm developed for computers in the United States was the Data Encryption Standard (DES), approved for use in the 1970s. The DES uses a 56-bit key. Because computers have become increasingly faster since the '70s, security experts no longer consider DES secure -- although a 56-bit key offers more than 70 quadrillion possible combinations (70,000,000,000,000,000), an attack of brute force (simply trying every possible combination in order to find the right key) could easily decipher encrypted data in a short while. DES has since been replaced by the Advanced Encryption Standard (AES RINJDAEL), which uses 128-, 192- or 256-bit keys. Most people believe that AES RINJDAEL will be a sufficient encryption standard for a long time coming: A 128-bit key, for instance, can have more than 300,000,000,000,000,000,000,000,000,000,000,000 key combinations. The sending computer encrypts the document with a symmetric key, and then encrypts the symmetric key with the public key of the receiving computer. The receiving computer uses its private key to decode the symmetric key. It then uses the symmetric key to decode the document.

To implement public-key encryption on a large scale, such as a secure Web server might need, requires a different approach. This is where digital certificates come in. A digital certificate is basically a unique piece of code or a large number that says that the Web server is trusted by an independent source known as a certificate authority. The certificate authority acts as a middleman that both computers trust. It confirms that each computer is in fact who it says it is, and then provides the public keys of each computer to the other.

1.4 CRYPTOGRAPHY ALGORITHM

1.4.1 AES RIJNDAEL

AES is based on the Rijndael cipher developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, who submitted a proposal to NIST during the AES selection process.Rijndael is a family of ciphers with different key and block sizes. AES has block size of 128 bits, but three different key lengths: 128, 192 and 256 bits. AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data.

AES is based on a design principle known as a substitution-permutation network, combination of both substitution and permutation, and is fast in both software and hardware. Unlike its predecessor DES, AES does not use a Fiestel network (In Cryptography, a **Feistel cipher** is a symmetric structure used in the construction of block ciphers, it is also commonly known as a **Feistel network**. The Feistel structure has the advantage that encryption and decryption operations are very similar, even identical in some cases, requiring only a reversal of the key schedule. Therefore the size of the code or circuitry required to implement such a cipher is nearly halved. A Feistel network is an iterated cipher with an internal function called a round function).

AES operates on a 4×4 column major order matrix of bytes, termed the *state*, although some versions of Rijndael have a larger block size and have additional columns in the state. Most AES calculations are done in a special finite field.

In this algorithm, substitute bytes indicate that the algorithm should substitute the byte of the state with a byte from the S-box, which replaces each byte with the inverse transformation. The shift row procedure indicated that it does not change the value of the row elements but changes their order and does a circular left shift to the rows.

The algorithm begins with an **Add round key** stage followed by 9 rounds of four stages and a tenth round of three stages. This applies for both encryption and decryption with the exception that each stage of a round the decryption algorithm is the inverse of its counterpart in the encryption algorithm.

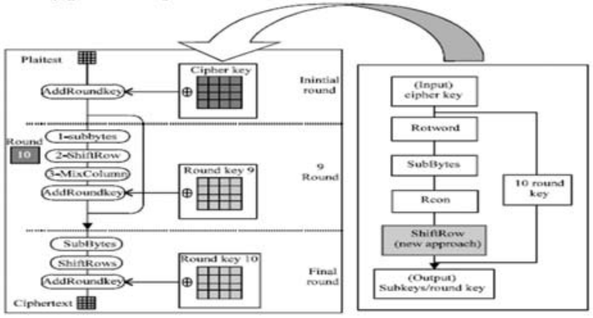


Figure 1.6 AES algorithm working

The four stages are as follows:

1. Substitute bytes

2. Shift rows
3. Mix Columns
4. Add Round Key

The tenth round simply leaves out the **Mix Columns** stage.

The first nine rounds of the decryption algorithm consist of the following:

1. Inverse Shift rows
2. Inverse Substitute bytes
3. Inverse Add Round Key
4. Inverse Mix Columns

Again, the tenth round simply leaves out the **Inverse Mix Columns** stage.

AES Algorithm has following steps:

Step1: Key Expansion: In this step round keys are derived from the cipher key using Rijndael’s key schedule.

Step2: Initial Round: This step consists of following sub-steps

i)Add Round Key:- In this step each byte of the state is combined with the round key using bitwise XOR

Step 3: Rounds: This step consists of following sub-steps

1. Sub Bytes:- It is a non-linear substitution step where each byte is replaced with another according to a lookup table.
2. Shift Rows:- This is a transposition step where each row of the state is shifted cyclically a certain number of steps.
3. Mix Columns:- it is a mixing operation which operates on the columns of the state, by combining the four bytes in each column.

Step 4: Final Round (no Mix Columns):It has following substeps:

1. Sub Bytes
2. Shift Rows

1. Add Round Key

AES algorithm Encryption process: The flowchart of the AES algorithm encryption process is given below

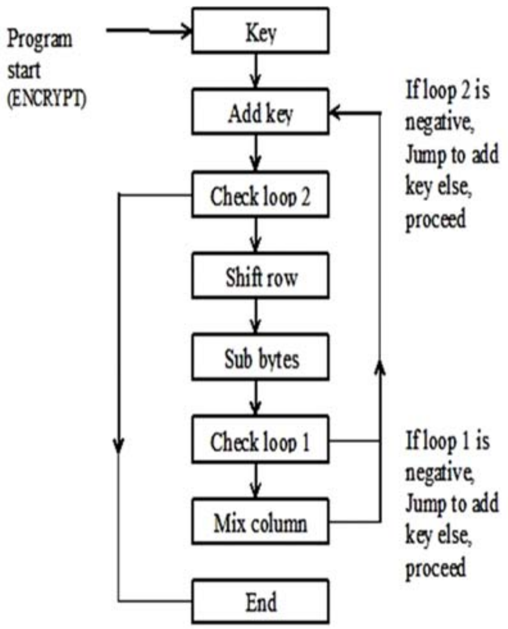

Program start (ENCRYPT)
Key
Add key
Check loop 2
Shift row
Sub bytes
Check loop 1
Mix column
End
If loop 2 is negative, Jump to add key else, proceed
If loop 1 is negative, Jump to add key else, proceed

Figure 1.7 AES Encryption Process

AES algorithm Decryption process: The flowchart of the AES algorithm decryption process is given below

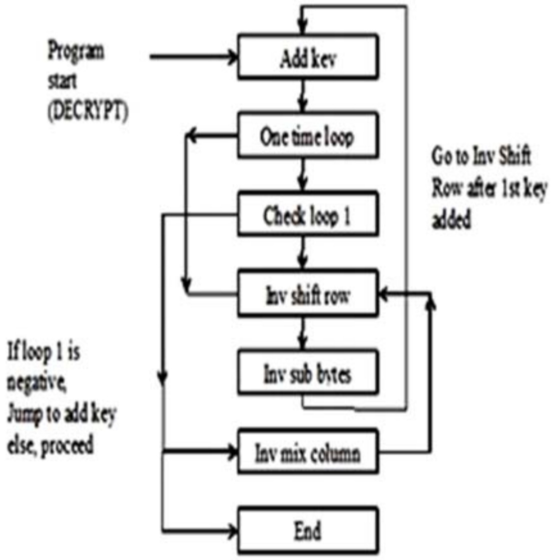


Figure 1.8 AES Decryption process

ADVANTAGES OF THE AES ALGORITHM ARE:

Security: AES is one of the more secure algorithms. Due to the ease of access to various components of the code, the security breaches that may arise can be detected very easily and corrected efficiently

Memory: AES does not require excessive memory to complete its functionality. The maximum bits used for encryption is 128 bits and this allows memory saving for both hardware and the software features.

Flexibility: AES algorithm works for a combination of a large number of blocks and bits. This encryption algorithm is very versatile and customizable since it can be modified depending on the problem to which it is applied.

1.4.2 BASE64 ENCODER

Base64 is a group of similar binary-to-text (A binary-to-text encoding is encoding of data in plain text. More precisely, it is an encoding of binary data in a sequence of characters. These encodings are necessary for transmission of data when the channel does not allow binary data (such as email or NNTP) or is not 8-bit clean) encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation. The term *Base64* originates from a specific MIME content transfer encoding.

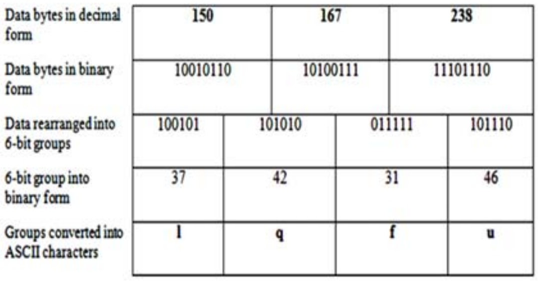
Base64 encoding schemes are commonly used when there is a need to encode binary data that needs to be stored and transferred over media that are designed to deal with textual data. This is to ensure that the data remains intact without modification during transport. Base64 is commonly used in a number of applications including email via MIME, and storing complex data in XML.

This encoding generally works in very less negligible time. This algorithm is very popular I e-commerce websites and application as lot of files are needed to come from server to your device in which if bandwidth is less it will take more time in loading those files as file size can vary from small to large and very large. This encoder converts the text into string value or whole data into string.

Base64 encoding takes three bytes, each consisting of eight bits, and represents them as four printable characters in the ASCII standard. It does that in essentially two steps.

Step 1: The first step is to convert three bytes to four numbers of six bits. Each character in the ASCII standard consists of seven bits. Base64 only uses 6 bits (corresponding to 2^6 = 64 characters) to ensure encoded data is printable and humanly readable. None of the special characters available in ASCII are used. The 64 characters (hence the name Base64) are 10 digits, 26 lowercase characters, 26 uppercase characters as well as '+' and '/'. For example, the three bytes are 150, 167 and 238; the corresponding (and frightening) bit stream is 100101101010011111101110, which in turn corresponds to the 6-bit values 37, 42, 31 and 46.

Step 2: These numbers are converted to ASCII characters in the second step using the Base64 encoding table. The 6-bit values of our example translate to the ASCII sequence "lqfu".



Data bytes in decimal form	150	167	238	
Data bytes in binary form	10010110	10100111	11101110	
Data rearranged into 6-bit groups	100101	101010	011111	101110
6-bit group into binary form	37	42	31	46
Groups converted into ASCII characters	l	q	f	u

Figure 1.9 Base64 Encoder working example

This two-step process is applied to the whole sequence of bytes that are encoded. To ensure the encoded data can be properly printed and does not exceed any mail server's line length limit, newline characters are inserted to keep line lengths below 76 characters. The newline characters are encoded like all other data.

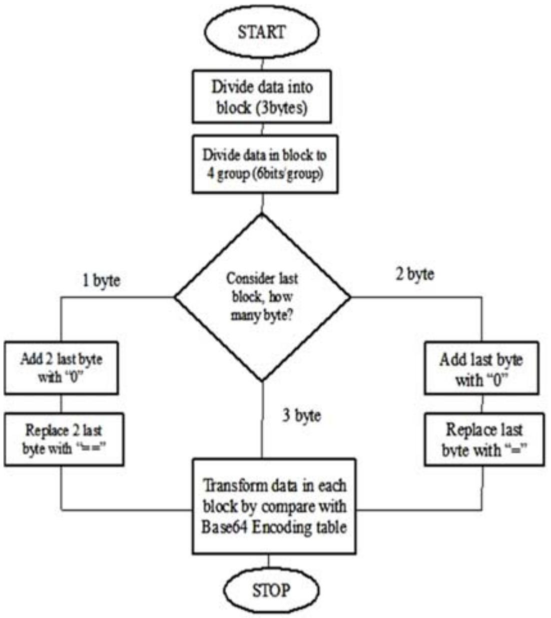

START
Divide data into block (3bytes)
Divide data in block to 4 group (6bits group)
Consider last block, how many byte?
1 byte
2 byte
Add 2 last byte with "0"
Add last byte with "0"
3 byte
Replace 2 last byte with "=="
Replace last byte with "="
Transform data in each block by compare with Base64 Encoding table
STOP

Figure 1.10 Base64 Encoder Flowchart

At the end of the encoding process we might run into a problem. If the size of the original data in bytes is a multiple of three, everything works fine. If it is not, we might end up with one or two 8-bit bytes. For proper encoding, we need exactly three bytes, however.

The solution is to append enough bytes with a value of '0' to create a 3-byte group. Two such values are appended if we have one extra byte of data; one is appended for two extra bytes.

Of course, these artificial trailing '0's cannot be encoded using the encoding table. They must be represented by a 65th character. The Base64 padding character is '='. Naturally, it can only ever appear at the end of encoded data.

2. PRESENT WORK

2.1 PROBLEM STATEMENT

We are going to analyze the AES RINJDAEL algorithm based encryption to cloud server. The AES RINJDAEL being fast and secure hashing algorithm will provide the more security to the database in cloud computing system. The system will have number of databases connected through them. The users do not knowing where the content has been uploaded on which particular database but the data of the user kept safe without the danger of being used by the unauthorized user and also further the database should be secured so that user can put his or her content. The proposed work is been extended by adding a compression technique known as lossless data compression which deals with audio and text data types. In future this work can be extended to public cloud database on Wide Area Network also since now public want to store their data into the cloud storage.

2.2 ISSUES IN CLIENT SERVER ARCHITECTURE

Since this thesis outlines the problem of security in cloud computing. There are various issues related to security in Cloud Computing which are been studied are as under.

1. Man in Middle attack: This attack is performed by a hacker on Network Layer 1 of ISO model. The information which is passed in bits between 2 devices (client and server) is passed through network cables. If a hacker in middle attacks the network layer the data shall get vulnerable to b hacked. The hacker can manipulate the data, edi the data, steel the data etc.
2. Low throughput and high encryption an decryption timings: Though cloud application uses data, and data passing capacity along th network is known as throughput. The throughpu is directly proportional to data size. If data size i large and it takes maximum time to convert int cipher text yielding more execution timings o encryption/decryption as a result it yields lowe throughput.
3. Privacy Leakage and Authentication: Since th research deals with Cloud service as storage i.e the user will upload its confidential data to the cloud system. The authentication problem can lead to leakage of privacy of particular user of the cloud application. The authentication system of this cloud based system must be strong enough to build a trust level towards a cloud server company who is offering a service of storage based services to general public. The information uploaded to cloud servers can range from very ordinary to very confidential even a pin number of debit/credit card.

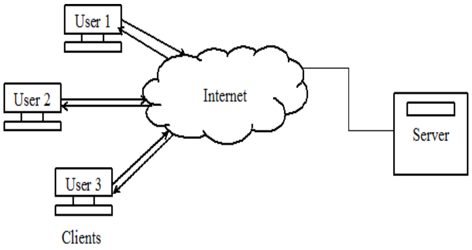


Figure 2.1 Client server architecture

2.3 OBJECTIVE

The main objective of this research work is to develop an efficient encryption system combined with compression technique that can encrypt the data and thus saving time and increasing the throughput.

The main contributions of the research work are:

1. A survey of existing AES algorithm in order to understand it's working.

2. Development of method that combine encryption feature of AES and compression feature of base 64 encoder.

3. Comparing the performance of AES and AES with compression to check the timings reduced and increased throughput

We are going to implement security architecture of the base paper in WLAN on real world using java web socket programming, consider it a real working protocol implementation.

There will be 3 levels of this protocol developed in java

1. Client Server Architecture: This is a main backbone of network. Internet – LAN/WAN are all relies on client and server architecture. This is developed via socket programming in java which a main basic approach of this thesis.
2. We will create the WLAN (wireless LAN) of system and existing authorized users will make connection to server of the organization where they work. In case the worker is unauthorized it will not be able to make the connection to server. This will need the database verification. The database will be present on server and all the data of existing user will be present on server's database which will be cross

verified dynamically. In the detail of particular authorized user we will maintain the unique ID of the user which will be hidden and will be asked when the user tries to establish the connection. Every user will have his unique key with them.

3. Encryption at client end via AES takes place on this 3rd level along with compression technique known as lossless data compression, which will pass the data along the network by reducing the actual size of data

   Our research or part of thesis is to implement this 3 layer system architecture on real world Wlan system. The users have to cross this 3 layered architecture. We will implement this layered system on LAN which is a real world example and will test it on network we will use the web socket programming in java to implement this protocol in real world which also a part of future work of this base paper. Looking at above six points this thesis successfully overcomes the objectives and aims stated above. The security issue is been successfully resolved which leads to data leakage and man in middle attack. The architecture followed was secured with AES RINJDAEL Encryption Algorithm where the throughput of devices is needed to be maximized for ruling out the scalability issues reducing the size of actual data and passing it along the network will obviously takes less time. Cloud computing systems generally have a front end, which is what the user sees, and a back end, which does all the work. Cloud computing shares some similarities with an older model of computing called timesharing

2.4 PROPOSED SCHEME

In our work we mainly deal with the cloud database, and in that we try to solve the security and improve the efficiency (throughput) by reducing the size and reducing the time required for encryption and decryption. We will work on three parameters of AES viz. time required to encrypt and decrypt the data, size of the file and throughput. Throughput refers to the performance of tasks by a computing service or device over a specified period. It measure the amount of completed work against time consumed and may be used to measure the performance of processor, memory and/or network communications. Security plays important role in this work as to protect our sensitive information from the unauthorized user this will be conducted with the help of certain algorithm and in that algorithm

The proposed system exhibits the solid frameworks which were designed during the planning phase of this thesis. Firstly the framework of client-server architecture was been studied which has a Linux based kernel.

The encryption process uses a set of specially derived keys called **round keys**. These are applied, along with other operations, on an array of data that holds exactly one block of data that is to be encrypted. This array we call the state array.

We can take the following steps to encrypt a 128-bit block:

1. Derive the set of round keys from the cipher key.
2. Initialize the state array with the block data (plaintext).
3. Add the initial round key to the starting state array.
4. Perform nine rounds of state manipulation.
5. Perform the tenth and final round of state manipulation.
6. Copy the final state array out as the encrypted data (cipher text).

The reason that the rounds have been listed as "nine followed by a final tenth round" is because the tenth round involves a slightly different manipulation from the others. The block to be encrypted is just a sequence of 128 bits. AES RINJDAEL works with byte quantities so we first convert the 128 bits into 16 bytes. We say "convert," but, in reality, it is almost certainly stored this way already. Operations in RSN/AES RINJDAEL are performed on a two-dimensional byte array of four rows and four columns. At the start of the encryption, the 16 bytes of data, numbered D_0-D_{15}, are loaded into the array.

Method	DES	AES RINJDAEL	Modified AES RINJDAEL
Approach	Symmetric	Asymmetric	Asymmetric
Encryption	Faster	Slow	Fastest
Decryption	Fastest	Slow	Faster
Key Distribution	Difficult	Easy	Easy

Complexity	O(log N)	O(N^3)	O(N log N)
Security	Moderate	Highest	Highest

Table 2.1: Performance analysis and comparison of symmetric and asymmetric key in DES, AES RINJDAEL, and Modified AES RINJDAEL

Following are the parameters on the basis of which working of this technique and already existing technique is measured.

1. **Time**: In this method, time depicts the time required to encode, decode, encrypt and decrypt the plain text. The time is measured in milliseconds. The time taken by AES modified (AES + Base 64 Encoder) is far less than the time taken by simple AES.
2. **Size**: Size depicts the size of file that is being fed for encryption and decryption. After encoding the size of file is further reduced and is then sent to encryption that further reduces the time for processing. The time used in encoding and encryption depends on the size of the file.
3. **Throughput**: Throughput refers to the performance of tasks by a computing service or device over a specified period. It measure the amount of completed work against time consumed and may be used to measure the performance of processor, memory and/or network communications.

   Throughput = size/time

The efficiency of the research work is examined by calculating these parameters and comparing these parameters with the existing techniques.

2.5 WORKING METHODOLOGY

With the help of AES RINJDAEL we can not only encrypt the data but also protect it from unauthorized access. AES RINJDAEL works on 12 bit of data therefore it is faster as compared to the other algorithms. Encryption requires more time but the data remain safe because decryption become difficult. The work methodology is explained below that describes how plain text will be converted into cipher text using AES and AES with string compression technique.

Step 1. Cloud Server will be made from where the communication will take place.

Step 2. Peer to peer communication application or other cloud service applications will be made.

Step 3. Connection of established cloud server with local client is established.

Step 4. The file to be sent is selected after logging in with appropriate username and password.

Step 5: The file is first encoded using base 64 encoder, then this encoded file will be sent to AES algorithm foe encryption

Step 6. Encryption process is as below

It is the process of converting the original text into the cipher text data.

Following are some of the steps:

1. Provider should transmit the public key (n, e) to the user who wants to store the data with him or her. Public key is the key that can be shared easily
2. User data is now mapped to an integer by using an agreed upon reversible protocol, known as padding scheme
3. Data is encrypted and the resultant cipher text (data) c is:

   C = me (mod n)
4. This cipher text or encrypted data is now stored so that later on can be used when required

Step 7. At the receiver end decryption process is performed.

Decryption is the process of converting the cipher text (data) to the original plain text (data)

Following are some of the steps:

1. User request the service provider for the data
2. The service provider verifies the authenticity of the user and then gives the encrypted data i.e. C
3. The user decrypts the data by computing

   m = Cd (mod n).
4. Once the m is obtained the user can get back the original data by reversing the padding scheme

Step 8. This decrypted file is then sent to base64 encoder for decoding process to obtain the plain text.

Step 9. Reduces encryption and decryption time along with increased throughput is obtained.

2.5.1 SYSTEM FLOW DESIGN

Following is the system flow design of the proposed system. In this a file is uploaded from the computer and sent to the server. First the file is encoded by Base 64 encoder and then encrypted by AES algorithm; finally cipher text is generated and saved in the database. From there the user will select the file to be decrypted and reverse process starts. First the file is decrypted and then decoded and finally plain text is obtained.

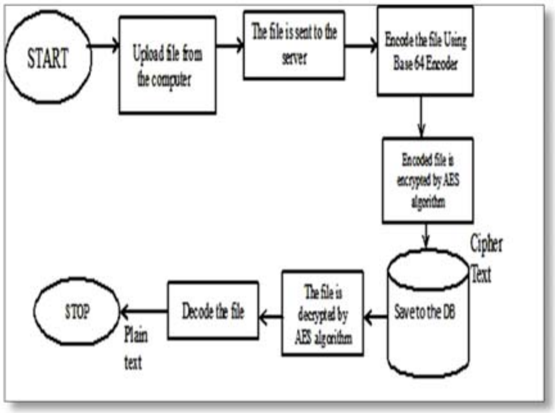

START
Upload file from the computer
The file is sent to the server
Encode the file Using Base 64 Encoder
Encoded file is encrypted by AES algorithm
Cipher Text
Save to the DB
The file is decrypted by AES algorithm
Decode the file
Plain text
STOP

Figure 3.2 System Flow Design

3. RESULTS AND DISCUSSIONS

The result obtained using AES Modified technique is shown following:

3.1 RESULTS USING MODIFIED TECHNIQUE

The Table 3.1 represents the six different sizes of files and corresponding encryption execution time taken by AES and AES Modified algorithms in seconds. By analyzing the Table 3.1, we conclude that the encryption time taken by AES Modified is very small as compare to AES. The encryption time taken by AES Modified and AES and six different size input files

Table 3.1Encryption Execution Time (Seconds)

Input File Size (MB)	Encryption Execution Time(Seconds)	
	AES	AES Modified
5	31.600	30.588
10	72.839	64.649
15	100.201	90.180
20	136.978	125.382
25	162.795	154.465
30	190.847	178.471

The Table 3.2 represents the six different sizes of files and corresponding decryption execution time taken by AES Modified and AES algorithms in seconds. By analyzing the Table 3.2, we conclude that the decryption time taken by AES Modified is very small as compare to AES. The decryption time taken by AES Modified and AES and six different size input files

Table 3.2Decryption Execution Time (Seconds)

Input File Size (MB)	Decryption Execution Time(Seconds)	
	AES	AES Modified
5	33.135	31.871
10	69.353	60.489
15	94.794	87.506
20	126.381	111.735
25	154.715	148.656
30	191.437	188.754

The Table 3.3 represents the six different sizes of files and corresponding throughput execution time taken by AES Modified and AES algorithms in KB/Seconds. By analyzing the Table 3.3, we conclude that the throughput time taken by AES Modified is large as compare to AES. The throughput time taken by AES Modified and AES and six different size input files.

Table 3.3Throughput Execution Time (Seconds)

Input File Size (MB)	Throughput Execution Time(MB/Seconds)	
	AES	AES Modified
5	3.7130	3.7134
10	7.4185	7.4401
15	1.1110	1.1132
20	1.4836	1.4858
25	1.8528	1.8550
30	2.2220	2.2242

4. CONCLUSION AND FUTURE WORK

4.1 CONCLUSION

During research on issues on secured client-server architecture and after its successful implementation, it's been concluded that AES RINJDAEL was successful and provides a strong point of security to existing client-server cloud architecture. The Purpose of adding three layered system in client and server side was successful and gave marginal better outcomes than previous research. In architecture while adding secured layers we kept in mind the different scenarios of authentication and authorization. The AES RINJDAEL at last level gave this research a brilliant security that this architecture is fully secured for any kind of confidential data preservation along with good results than previous basic AES

4.2 FUTURE WORK

Since AES RINJDAEL provides a strong security measure to existing system. It will be always an area of research as AES RINJDAEL takes less of power during generation of public and private key and yields higher throughput. On mobile OS AES RINJDAEL performance can be analyzed in terms of battery consumption and throughput. Preserving the power of smart phones can be new area of research in mobile cloud computing since every application back-end phase is shifting from clusters/grid to cloud based system. These applications usually take lot of battery power and can affect the battery life of particular phones. Since smart phones processor and RAM runs 24 hours if it's not on switched off mode the application running in background can eat up RAM and Processor which leads to decrease in battery life of a smart phone.

REFERENCES

1. SanjoliSingla, Jasmeet Singh., "Implementing Cloud Data Security by Encryption Using Rijndael Algorithm", Global Journal of Computer Science and Technology Cloud and Distributed, Vol 13, Issue 4, version 1.0, 2013
2. G. Jai Arul Jose, C.Sajeev, Dr.C.Suyambulingom., "Implementation of Data Security in cloud Computing", International Journal of P2P Network Trends and Technology, Vol 1, Issue 1, 2011.
3. Navraj Khatri, Jagtar Singh, Rajeev Dhanda., "Comparison of Performance of AES standards Based upon Encryption/Decryption Time and Throughput", International Journal of Engineering Research and Technology (IJERT), Vol 1, Issue 5, July 2012.
4. Tanzilur Rahman, Shengyi Pan, Qi Zhang., "Design of a High Throughput 128-bit AES (Rijndael Block Cipher)", International Multiconference of Engineers and Computer Scientists (IMECS), Vol 2, March 2010.
5. Daniel Fowles., "Implementation and analysis of the Rijndael encryption algorithm in different programming languages", Bachelor of Science in Computer Science with Honors, The University of Bath, April 2008.
6. Jasmeet Singh, SanjoliSingla., "Survey on Enhancing Cloud data security using EAP with Rijndael Encryption Algorithm", Global Journal of Computer Science and Technology Software and Data Engineering, Volume 13 Issue 5 Version 1.0, 2013
7. PrashantRewagad, YogitaPawar., "Use of Digital Signature and Rijndael Encryption algorithm to Enhanced security of data in Cloud Computing Services", International Journal of Computer Applications, Emerging trends in Computer Science and Information Technology, 2012
8. Vishwagupta, Gajendra Singh, Ravinder Gupta., "Advance Cryptography algorithm for improving data security", International Journal of Advanced Research in Computer Science and Engineering, Volume 2, Issue 1, January 2012.
9. DiaaSalamaAbdElminaam, Hatem Mohamed Abdual Kader, Mohiy Mohamed Hadhoud., "Evaluating the Performance of Symmetric Encryption Algorithms", International Journal of Network Security, Volume 10, No.3, May 2010.
10. Amit Sangroya, Saurabh Kumar, JaideepDhok, VasudevaVarma. "Towards analyzing data security risks in Cloud Computing Environments".
11. EmanM.Mohamed, HatemS.Abdelkader, Sherif El-Etriby., "Data security model for Cloud Computing", The twelfth international conference on networks, 2013.
12. Navraj Khatri, Rajeev Dhanda, Jagtar Singh., "Comparison of Power Consumption and Strict Avalanche criteria at Encryption/Decryption side of Different AES standards", International Journal of Computational Engineering Research (IJCER), Volume 2 Issue 4, August 2012.
13. Dr. Chander Kant, Yogesh Sharma., "Enhanced Security architecture for cloud security", International Journal of Advanced research in computer science and software engineering (IJARCSSE),Volume 3, Issue 5, May 2013.
14. Bruce Schneier, John Kelseyy, Doug Whitingz, David Wagnerx, Chris Hall, NielsFrguson, "Performance Comparison of the AES submissions", Version 2, February 1, 1999.
15. SubedariMithila, P. Pradeep Kumar., "Data security through Confidentiality in cloud computing environment", International Journal of Computer Science and Information Technologies (IJCSIT), Volume 2 (5), 2011.