# DESIGN AND SIMULATION OF CLOUD COMPUTING SECURITY FOR DATA AT REST AND DATA ON TRANSIT

**BY:**

**DR. AGBASONU VALERIAN C.**

**DEPARTMENT OF COMPUTER SCIENCE, IMO STATE UNIVERSITY, OWERRI.**

**E-MAIL: valpraise@gmail.com Tel: +2348038727760**

**DR. NDUNAGU JULIANA NGOZI**

**SCHOOL OF SCIENCE AND TECHNOLOGYNATIONAL OPEN UNIVERSITY OF NIGERIA,**

**OWERRI STUDY CENTRE.**

**MR. CHIDIEBERE UGWUEGBULAM**

**SCHOOL OF SCIENCE AND TECHNOLOGY, NATIONAL OPEN UNIVERSITY OF NIGERIA, OWERRI STUDY CENTRE.**

**Abstract**

*This research focuses on cloud computing security for data stored in the database*

*Called data at rest and data being transmitted called data on transit. The purpose is to find out why access control and data confidentiality are being violated despite the numerous measures put in place to checkmate unauthorized access. It was at last found out that these measures such as encryption and password have loopholes and hackers know how to get their way through them. Encryption outputs (Cryptext) usually do have patterns with which to recognize the algorithm that produced it. Just by looking at a cryptext, cryptanalysts can tell whether it is BlowFish, Rijndael, MD5 or which other algorithm that produced it, and quickly they design a reverse algorithm to decrypt and read the data. In the course of Passwords too, it was found out that passwords can be guessed, copied or hijacked. This research therefore is providing a solution by designing an encryption model that can generate inconsistent cryptext with no pattern. Since illegal decryption is hinged on pattern matching, this encryption is therefore hack-proof. The research also designed a system of authentication that uses Biometrics instead of Password for access control. The advantages of having Biometric Systems in the cloud were given, such as reduction of cost and ubiquitous access. New paradigm shifts were also suggested, especially in the area of having Biometric as a Service (BaaS).*

*These designs are simulated using a web-system developed with PHP, MySQL, JavaScript and other Programs. The System Design followed the SSADM methodology for componentization of the system Modules giving room for coupling, decoupling, updation, modification, encapsulation and reuse, as well as easy maintainability. Unified Modeling Language was extensively used to simplify the explanation of the system Modules. Recommendations were also made for application of these designs in other online activities where data is set to rest or set on transit such as eMails, SMS, social Media inbox and any other Privacy-demanding activity. Government Agencies can also adopt it for application in any top secret and classified Information.*

## Introduction

The Internet Users have experienced violation of privacy at one time or the other. Unauthorized access to sensitive and private data is no longer alien-news for today's Internet users. The Internet, as a global Network with its hyperlinked resources improves communication and exchange of files and business data. People now connect to the internet to do business, to search for information or to send messages. According to the International Telecommunication Union, there are over three billion (3,035,749,340) users of Internet worldwide (ITU, 2014). The evolutionary trend and permeation of the Internet into almost every facet of life, has catalysed the emergence of cloud computing, which is open for a multi-tenant transaction and simplification of technological sophistication from a user's focal point.

In the simplest terms, cloud computing means processing, storing and accessing data and programs over the Internet instead of in the user's computer (Catteddu, D. and Hogben, G., 2009). The business world is moving fast into the cloud because of the gains, including capital cost reduction, elimination of hardware failure-risk, Reduction of spending on Technology Infrastructure, Improve accessibility and flexibility, etc**.** However as genuine business people are migrating to the Cloud, so also are the Cyber Criminals (Hackers), after all that's where they feel the big business is (Cretu, 2012). Hackers have devised new techniques to even make their nefarious operations in the cloud easier. Many preventive measures have been applied to stop them from reading confidential and private data. One of such measure is Encryption.

Encryption is an Information Security Measure that renders data unintelligible to unauthorized readers. It produces as output a coded transformation of data in a form unreadable to intruders and interlopers who lack the appropriate key to decipher the encoding (Y. Ching-Nung, 2013). Encryption technique is important not just for the data, but for the database (data at Rest) and the communication (data on Transit) channels as used in the secure Socket layers (SSL) and Secure Hypertext Transfer Protocol (HTTPs). Data is the most important resource to a user, and in a public cloud where communal computing and multitenancy is practiced, encryption must be inevitable to ensure confidentiality and integrity of data and data banks. Encryption is implemented with a hope to curb impersonation, wiretapping, piracy, spoofing and data diddling which are common threat in a multi-user environment. There has been Blowfish, Rijndeal, MD5, SHA-1 and so many other encryption algorithms presently in existence to combat some of the known threats yet Users with confidential data are gripped with fear of insecurity, even the service providers are not confidently sure of the data security despite the encryption used (G. Ateniese, 2007).

Hackers have developed several machinations for breaking cloud securities. They have botNets or Robot-Networks which are intelligent computers which guess and apply passwords in a bid to gain unauthorised access, in what they call brute-force attack. The Oracle padding attack has been used to attack encryptions; they also have Trojan horses for creating security backdoors and weakening of security systems. All these threaten and sometimes neutralize cloud security (Baltzan, 2014).

And In the course of this research, it was found out that encryption outputs have patterns. That's why hackers have succeeded so far. If two hundred different data (plaintexts) are encrypted using the same algorithm, the output will be two hundred different encrypted data (cryptext) but with same pattern, same header or same length. This consistent output with recognizable pattern betrays the encryption algorithm to the Hackers. Once the hackers see a cryptext, they can tell which algorithm produced it, or worst still, develop a reverse algorithm to decrypt it. These hackers that specialize on illegal decryption are usually cryptanalysts. They have various tools and methods with which to attack encrypted data once the pattern is analyzed. Most cryptanalyst has what they call Rainbow-Table that contains several terabytes of pre-computed data for pattern

matching (Prashant, 2012). The boast that with Rainbow-Table, every cryptext is in danger is true, but only so long as the cryptext has a recognizable pattern; Because even if the Algorithm is not known, Cryptanalysts can use what they call Oracle Padding Attack (OPA) to fish it out; What is just needed is that uniform pattern (Prashant, 2012). No wonder data leakages and unauthorized reading of private information abound despite numerous encryption algorithms.

From the foregoing, it is glaringly clear that a paradigm shift is required to keep data safe from these crime-thirsty cryptanalysts. The shift is to deflect from Consistent pattern to an inconsistent pattern. Since these cryptanalysts doesn't do decryptions manually but rely on computers for pattern matching, they cannot successfully and easily decrypt what is called un-patterned cryptext. This means that if there can be an encryption algorithm that can produce outputs with inconsistent and unique cryptext, then cryptanalyst would be defeated flat.

Encryption protects data from unauthorized usage but not access. The best known way to prevent unauthorized Access is by Biometric means. Using Password for authentication is no longer enough to protect data. Recent events and reports have shown that password is no longer sufficient as it can be manipulated, forgotten or hijacked among other failings (Shinder, 2011). Ndunagu, et al (2012) emphatically agrees with this and recommends that Biometric systems be used. Biometric systems are systems that authenticate a user based on unique biological traits such as Finger Print, Voice Print, Facial morphology, Retina of the Eyes, and so on. Central to all computer security is the concept of authentication - verifying that the user is who he claims to be. (D. W. Chadwick, 2012).Biometric authentication has been widely regarded as the most foolproof - or at least the hardest to forge or spoof. Since the early 1980s, systems of identification and authentication based on physical characteristics and unique biological traits have been available to enterprise IT. But the problem is that implementing Biometric authentication is very expensive and reliable devices are difficult to come by. Small and medium-sized businesses and single users may not afford it. in fact this cost issue has been the major challenge that has delayed the implementation of Biometrics authentication, not the knowledge of it (Moore R.,2005). However we know that Cloud computing cuts down every computing cost for both hardware and software, coupled with its on-demand provision and pay-only-as-you-use methodology that's why H. Shuai and X. Jianchuan (2011) recommends having what they call Biometric as a Service (BaaS) where a cloud Provider specializes in Biometric Authentication for all cloud users.

This Project is therefore poised to search into the existing Encryption algorithms to debunk why they have not been very faithfully effective in securing data. The project will then propose a new unbreakable algorithm and merge it with Biometric system to guarantee effective and fortified system of security in the cloud computing. This project if adopted will prevent impersonation, curb unauthorized access to private data and stop illegal decryption of data in the cloud.

**Aims and Objectives:**

The major thrust of this study is to model a 'secure' cloud-based framework for data exchange in a cloud platform. It also aims to propagate a new security idea for both Cloud infrastructure and data migration in the virtual world.

The study is specifically carried out to:

- Investigate the *how* and *why* of hacker's attack and unauthorized access to data stored in a virtual Network
- Design a hack-proof encryption algorithm for data in a cloud's communication platform
- Simulate this design using a simple PHP-MySQL-JavaScript platform

**Significance of the Study**

The Study when completed would provide security assurance to cloud users thus motivating many to subscribe for cloud services, especially Security Agents, Virtual Organizations, Banks and others whose information requires absolute confidentiality.

The Project will design and simulate an encryption algorithm model that outputs a pseudo-header encryption which will be unbreakable by any hacker.

This model can be applied to "big Data" and all data as a Service provisions including email and SMS Messages, Social Media inbox, other classified communication platforms.

The project will design Authentication that uses Biometric system for providing an impersonation-free security system, and thus stop unauthorized access to sensitive data stored in the cloud.

This can be applied to Bank ATM Machines, and other Private Information (PI) Systems; as well as in Device access Control.

Cloud Service Providers, especially those into Platform as a Service (PaaS) and Data as a Service (DaaS) can use these models in this project to secure their infrastructure and Users' data.

**Keywords:** Cryptext, Cryptanalyst, Hash, Decrypt,Hackers, Hack-proof, Encryption, Rainbow Table, Biometric, Cloud Service Provider.

**Design Methodology**

The design methodology used for this project is Structured System Analysis and Design Methodology (SSADM). The SSADM makes it feasible to build the system component by component for easy debugging, maintenance and deploying using the waterfall model.

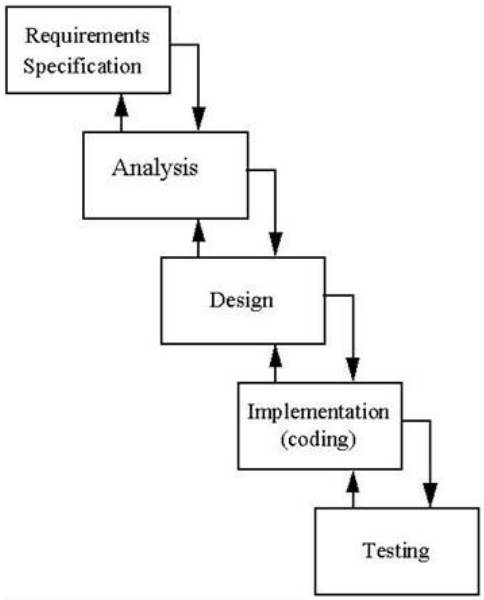



Fig 1.1 The waterfall Model adopted for the system

The New System followed the new five-tier architecture of system Development, which is an expansion of the three-tier model. The five-tier architecture has five layers and each layer has the ability to exist on a physically independent system and be able to communicate over a network (Internet). The layers have a clearly defined interface or API through which it communicates with the Layer below and above it. Moreover, because of the modular structure of the system, each layer is replaceable with other equivalent technologies. One major benefit of the five-tier system is that the Application has a better chance of coping with future traffic and advance technological integrations as it is able to scale well within each tier and thus make developers independent of specific vendors.

The five-tier is diagrammatically illustrated below:

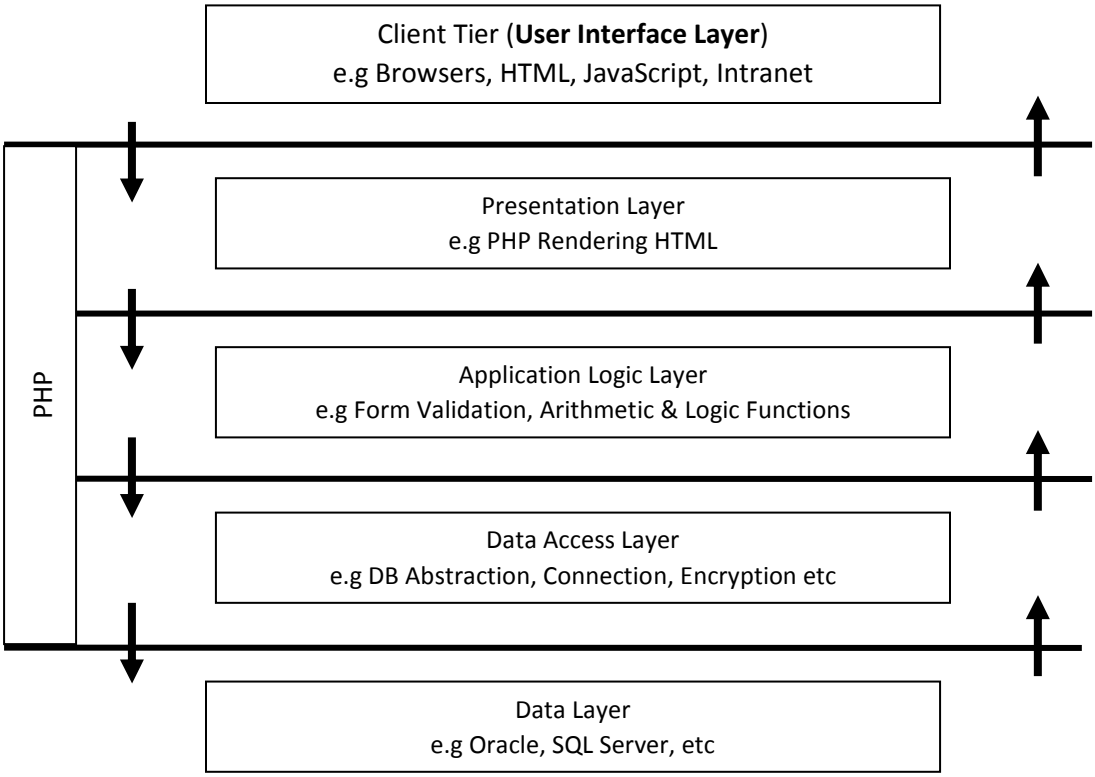



Fig 1.2: The 5-Tier Architecture of the New System

Unified Modeling Language (UML) is used extensively to illustrate the designs of the new system. UML is adopted because of its simplicity, clarity and ease of use in system designs. It also integrates easily with SSADM which is the core methodology for this new project. The designs are presented in many formats including Operation Procedure: showing how the system interacts with the user; System block diagram: showing the design modules in block form; Component Chart showing the component interfaces of the system, Sequence Diagram: showing the order of flow of processes and Deployment Diagram: showing the hardware and Interaction systems.

**The Operation Procedure Chart:**

The procedure chart shows what the steps taken when a User wants to secure data using the system. The user Logs in with UserName or Biometric or any authentication, the user submits the data along with a salt-key. The salt-key is a word, phrase or any combination of alphanumeric text that is not more than 32-digits.

The system uses this key to "salt" the encryption. Salting hashes the encryption output (the cryptext) and makes it to be different at each round of encryption, thereby producing outputs that are not uniform. The Salting also hardens the encryption. The higher the digits of the salt, the harder the encryption. A 12-digit salt is harder than a 5-digit salt, and more difficult to break.

After the encryption, the cryptext is reversed (to harden it the more) and stored.

During retrieval of the Information, The user logs in and indicates the data to read and supplies a key. The system decrypts with that key.

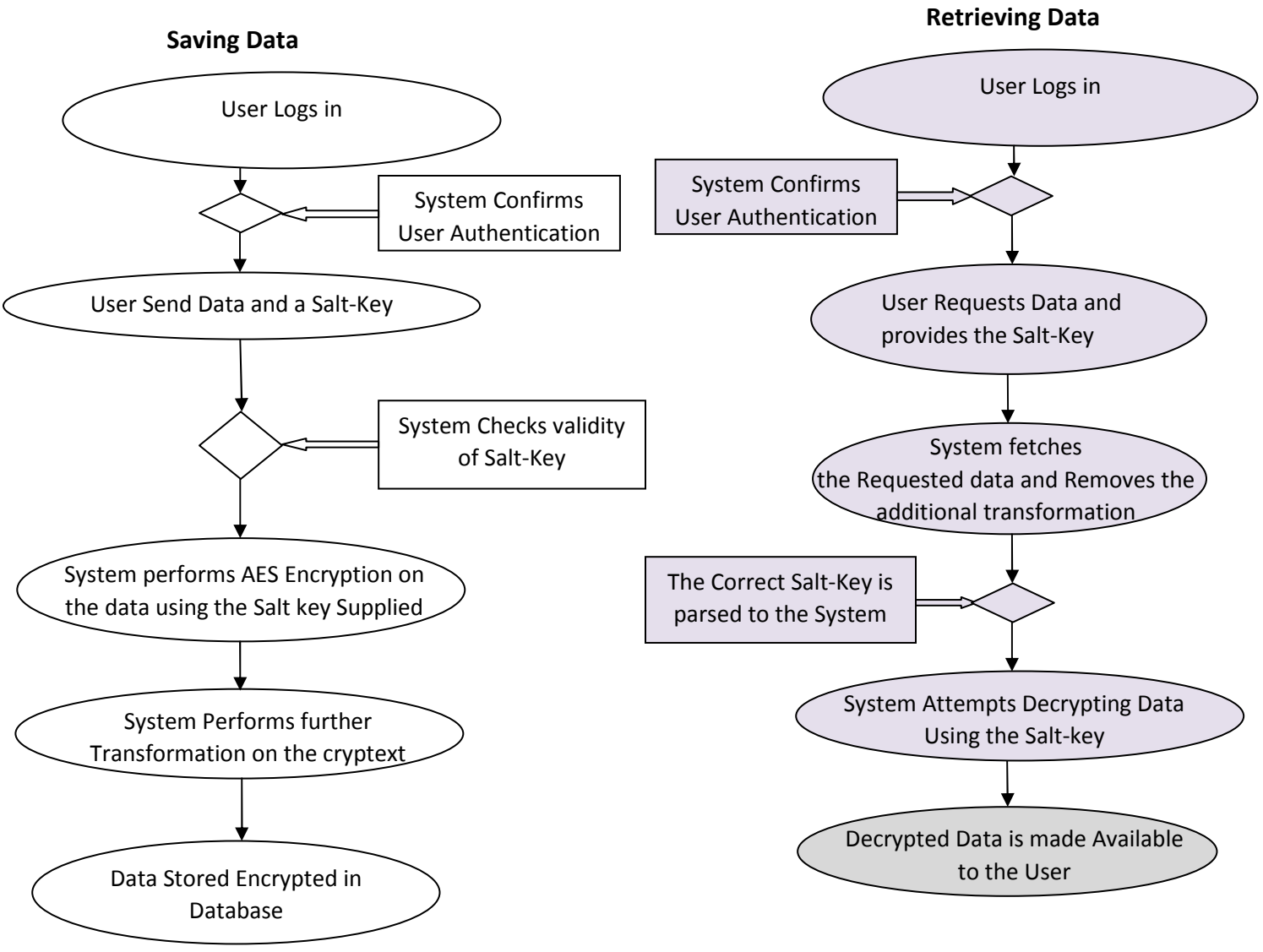
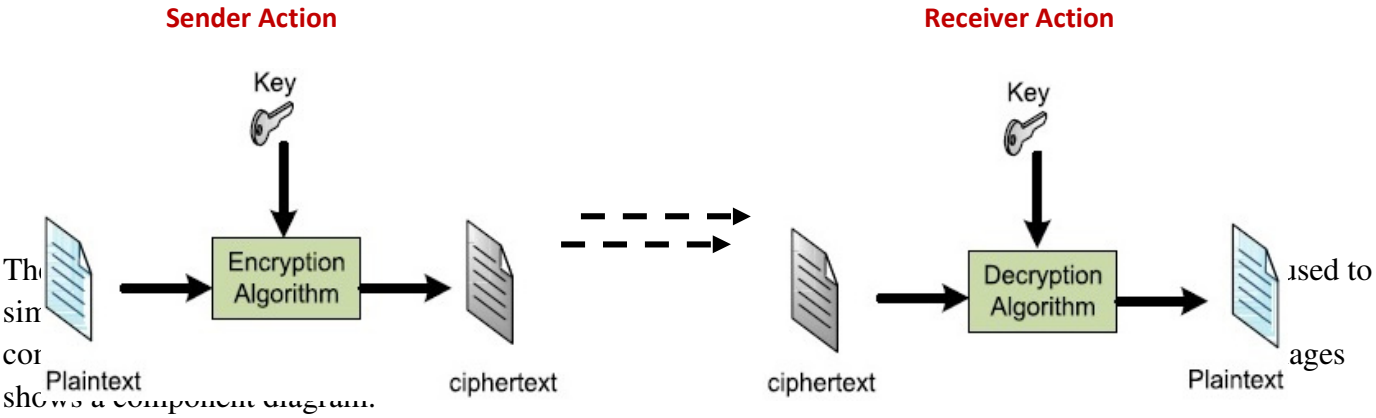



Fig 1.3: **The Operation Procedure Chart**

**Communication Diagram**

The communication diagram displays the relationship of components of this new system. It is used to simplify the complexity of the new system showing the communication link within the system.




Th sim cor sho ised to ages

Fig 1.4: The Internal Communication Diagram

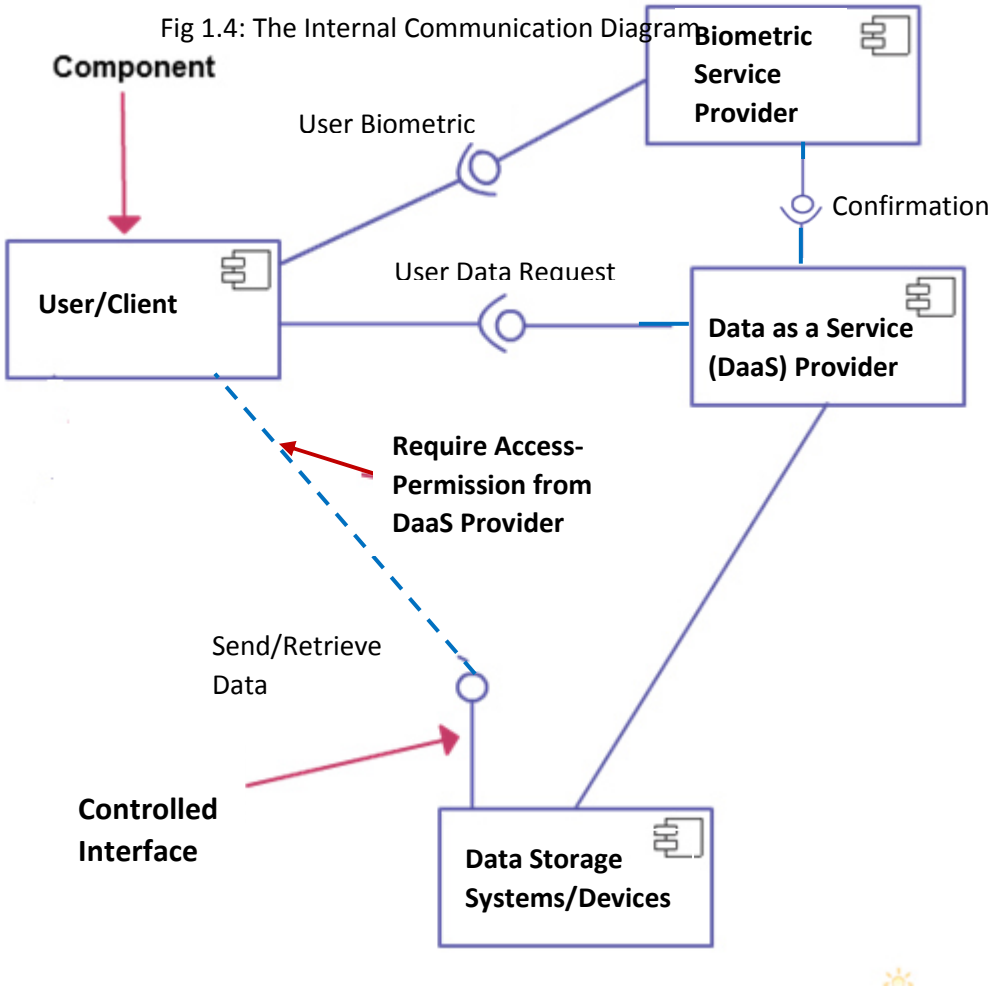



Fig 1.5: The Component Diagram of the new system

**Use Case Diagram**

The Use case Diagram is the behavioral UML diagrams used to give a graphic overview of the actors and processes involved in the new system, including different functions needed by those actors and how these different functions are interacted. The Diagram is given below:

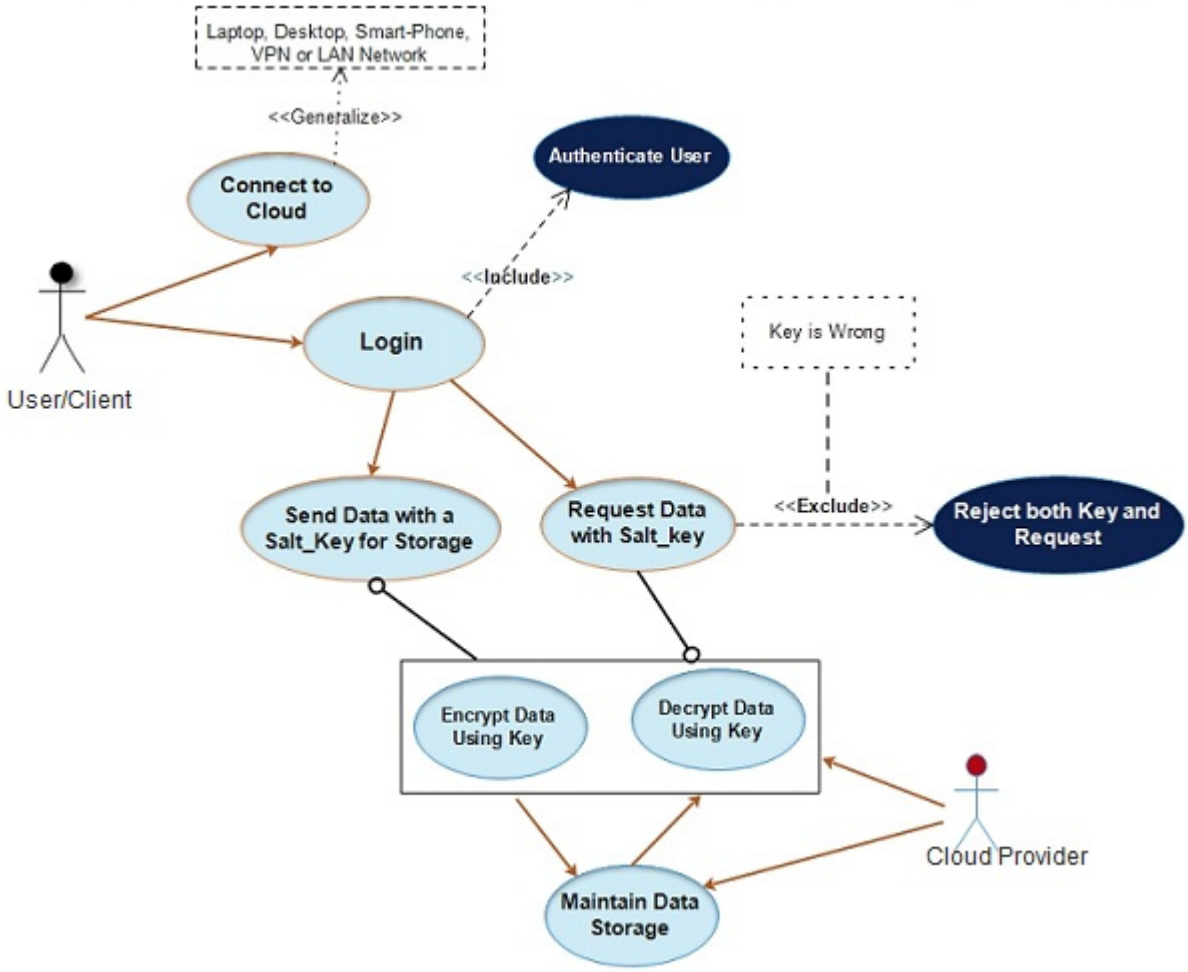



Fig 1.6: The Use case Diagram

**The Design of the Encryption System**

The encryption system is the main heart of this project. The Encryption for this new system is called Pseudo-crypt. This is because it has a deceiving header, variable length and inconsistent pattern. This will prevent illegal decryption because according to Y. Ching-Nung (2013) Decryptions require pattern-matching and fixed length harmonization.

The Block diagram of the design is shown below:

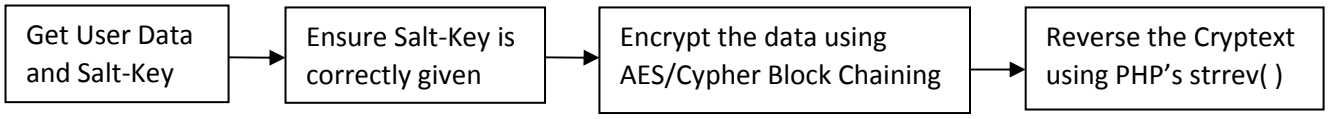



Fig 1.7: The Block Diagram of the Encryption System

The block diagram shows that when a user sends his data, he will also send a Salt_key which is required to encrypt/decrypt the data. The salt has to meet certain standards before it is used. In this project, it is simplified to be less than 32-digits. Using the Cypher Block Chaining will make the Encryptor see the data as a block and transform them simultaneously. The output of the transformation called cryptext is then reversed. This is the beginning of the pseudo-crypt. Reversing the word "come" for instance, will give "emoc" as output. This reversal plays a major role in the hardness of the encryption.

**The UML Deployment Diagram**

The deployment diagrams shows the hardware of requirement of the new system and the how they are deployed across multiple stakeholders. In the diagram, the user represents the cloud client, The provider is the Cloud DaaS provider while the Biometric server includes all the systems (hardware/Software) required to authenticate a user. This might be the same Daas provider or a different provider. The Cloud Controller represents the Cloud back end, the Hypervisor, and all that makes up the cloud. The deployment diagram is shown below**:**

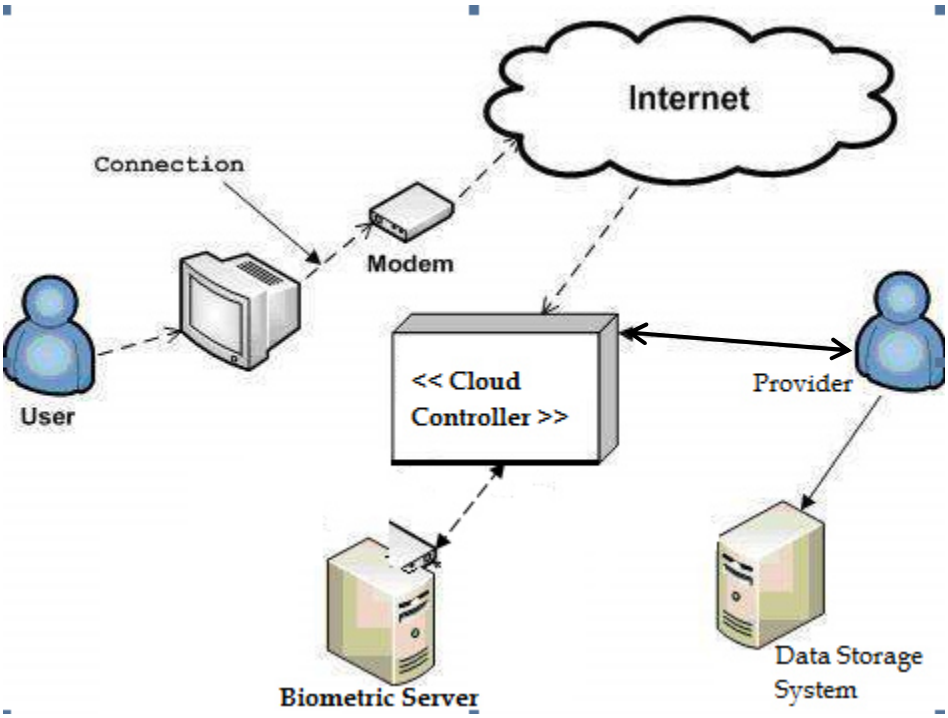



Fig 1.8: the Deployment Diagram of the new System

**The flow chart of the new system:**

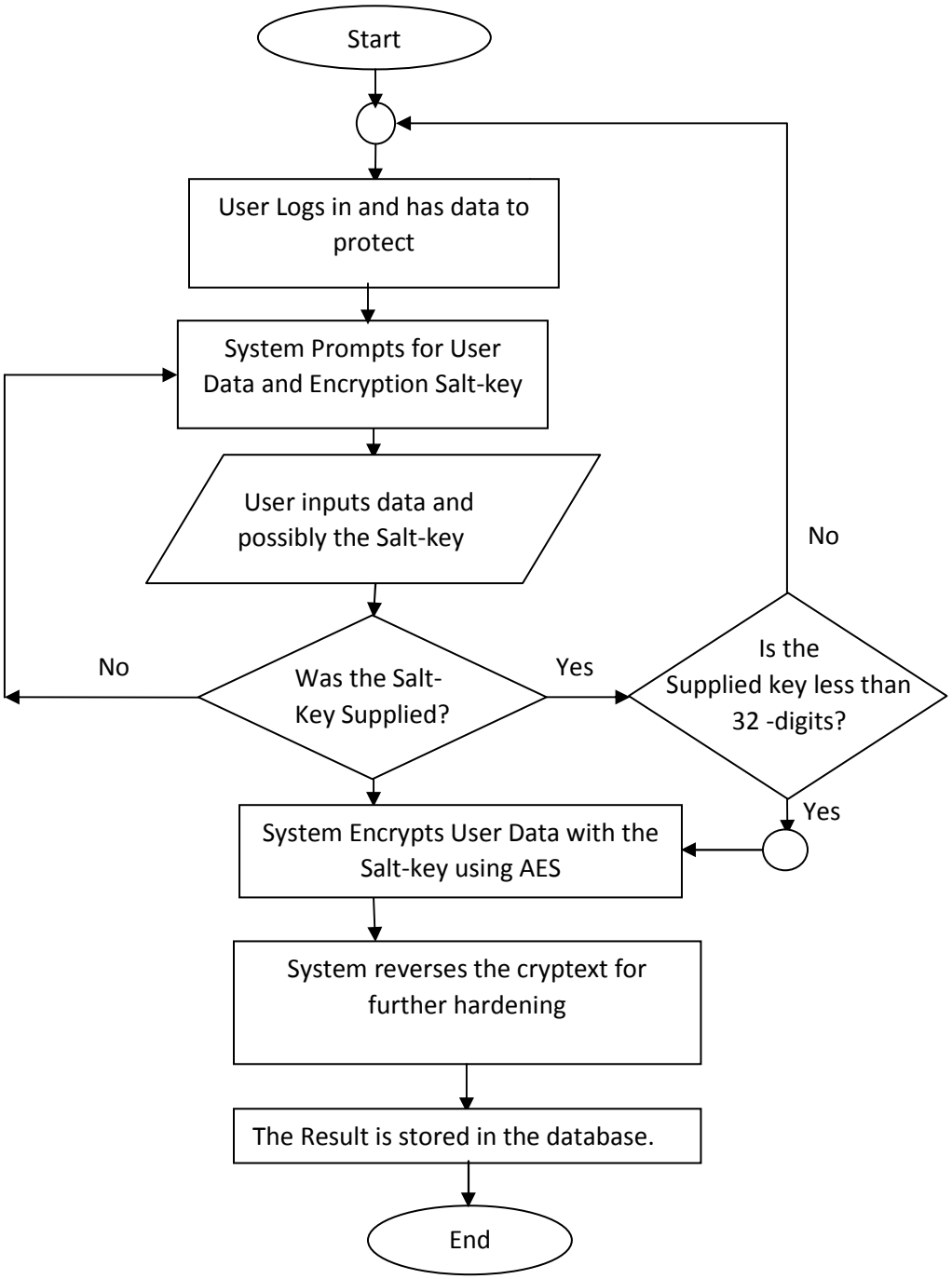



Fig 1.9: the Flow chart of the new system

**Rijndael Encryption Design**

Rijndael is the crypto-engine used for this project. Rijndael implementation has 128,192or 256 bit key lengths. Size of data blocks to be encrypted with Rijndeal is always 128 bits. Initial round of Rijndeal is AddRoundKey, this is followed by four iterative rounds including subBytes, shiftRows, mixColumns and add round key. Rijndeal with 128 bit key length has 10 rounds,192-bit has 12 rounds and 256 bit has 14 rounds. This project made use of the 128-bit key length to speed up the processing.

Each round consists of the following steps.
1. Initial AddRoundKey
2. SubBytes () Transformation
3. Substitutional Box Created For Subbytes
4. MixColumns () Transformation
5. AddRoundKey () transformation
The inverse process of encryption gives decryption text.

**1) The Sub Bytes step**
The SubByte step is a non-linear byte substitution that operates on each of the 'state' bytes independently, where a state is an intermediate cipher result. Here each byte in the state matrix is replaced with a SubByte using an 8-bit substitution box, the Rijndael S-box.

**2) The Shift Rows step**
The ShiftRows step operates on the rows of the state; it cyclically shifts the bytes in each row by a certain offset. For AES, the first row is left unchanged. Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively. For blocks of sizes 128 bits and 192 bits, the shifting pattern is the same. Row n is shifted left circular by n-1 bytes

**3) The Mix Columns step**
During this operation, each column is multiplied by the known matrix that for the 128-bit key is:
The multiplication operation is defined as: multiplication by 1 means no change, multiplication by 2 means shifting to the left, and multiplication by 3 means shifting to the left and then performing xor with the initial unshifted value. After shifting, a conditional xor with 0x11B should be performed if the shifted value is larger than 0xFF. In more general sense, each column is treated as a polynomial over **GF**(*28*) and is then multiplied by modulo x4+1 with a fixed polynomial
$c(x) = 0x03 \cdot x3 + x2 + x + 0x02$

**4) The Add Round Key step**
In the AddRoundKey step, the subkey is combined with the state. For each round, a subkey is derived from the main key using Rijndael's key schedule; each subkey is the same size as the state. The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR.

**Rijndael Algorithm**

```
Rijndael(data, CipherKey)
{
KeyExpansion(CipherKey, ExpandedKey);
AddRoundKey(State, ExpandedKey);
For( i=1; iFinalRound(Data, ExpandedKey + Nb*Nr);
}
```

And the round function is defined as:

```
Round(data, RoundKey)
{
ByteSub(data);
ShiftRow(data);
MixColumn(data);
AddRoundKey(data,RoundKey);
}
```

**Algorithms for the new system**

**Algorithm -1: Data Retrieval:**

Start

1. User Connects to the Cloud
2. The Access Interface is Opened
3. System requests Username and Biometric
4. If any of these (supplied) is wrong, reject the User and close the Access Interface
5. Open the data Access Interface
6. User Indicates the data to read
7. System fetches the data and demands for salt_Key to decrypt the data
8. Accept user input and decrypt the data with it.
9. If the key is wrong, give wrong and unreadable output.

End

**Algorithm-2: Pseudo-Crypt Algorithm:**

Start

1. Request for user data and a Salt-Key
2. Get and check the salt_key
3. If the salt_key is less than 4-digits, or more than 32-digits, give error report and return to step 1.
4. Encrypt data using Rijndael and the salt-Key with Cypher Block Chaining
5. Get the output 'y' of rijndael function
6. Reverse y using PHP's strrev( ) to get $Y_{rev}$
7. Output $Y_{rev.}$

**Discussion of Results**

Hackers attack data stores by guessing the Passwords or getting it by tricks from the Owner. The end target is to have access to private data.

D. W. Chadwick (2012) and Shinder (2011) sufficiently justified that password do not provide the most secure authentication mechanism as it is guessable and forgettable, Also Nduagu (2012) asserts that a biometric method is infinitely more difficult to "undo" using a brute force attack than ordinary password. Fingerprints for instance are unique and part of each individual and do not change over time. And so, they can be accessed at any time without requiring the customer to carry an additional device or memorizing (or forgetting) it. But Moore R. (2005) argues that Biometric infrastructure is very expensive and can be only available to those businesses that could afford it, thus supporting that Password authentication should remain.

However with cloud Biometric Service Provision, the cost of Infrastructure acquisition is removed as small and big businesses can access Biometric Authentication with little fee through the Service Provider that runs Biometric as a Service (BaaS).

Hacking or unauthorized decryption of data is done by analysing the abducted data, looking for uniform patterns that will help the hackers develop a rainbow Table, Brute-Force or any other attack system that will decipher the cryptext. That is why Prashant (2012) asserts that cryptexts should be varied or be made inconsistent using "Salt" or Encryption algorithms that support salting like Blow-Fish and AES. In agreement, Y. Ching-Nung (2013) suggests that randomly generated salts will give the desired inconsistent output (cryptext). This Inconsistency of output will make it near impossible to develop a system that will perform unauthorized decryption since automatic decryptions are based on Pattern Matching.

L. Popa et al (2010) suggests taking security key out of the Network for a new change and stronger security. He insists that User inclusion in the security of user-data is essential.

The Pseudo-Crypt designed in this project uses salt supplied by each user, producing a kind of random salt, since different users give different salt. This therefore aids in outputting inconsistent cryptext that cannot be decrypted unauthorized, and thus hack-free.

**Summary, Conclusion and recommendation**

**Summary of Results**

The System's encryption's cryptext is unique and cannot be decrypted with the conventional and contemporary illegal decryption systems. The system does not decrypt without additional input (Salt_key) from the user. This distinguishes the cryptosystem from others.

Two types of results are obtainable from the system: Result obtained by using the wrong key and the one obtained by using the correct key.

When the correct key is used, the expected output is obtained but if the wrong key is used, the data becomes hashed and an unreadable output is obtained.

**Conclusion**

It is feasible and important to have secure cloud computing system for data at rest and data on transit using Encryption and biometric system. The future of big Data and user's continual patronizing of cloud services especially those offering Data as a Service, depends on the guarantee of data security in the cloud. Using Pseudo-crypt, as established in this project is one sure way of provisioning data Security in a Cloud computing platform.

**Recommendations**

Based on the results and findings from this project the following recommendations are made to the generality of readers, especially to data-centric organizations, cloud service providers and the Internet/Cloud administrators:

1. That Cloud-based Biometric Processing Service Centres(i.eBaaS) be established with access and linkages to all cloud service providers for proper and effective authentication of cloud users
2. That Mobile Device/Computer Manufacturers consider adding biometric hardware (Scanners, readers, etc) on their products for seamless integration and communication with cloud systems.
3. That Cloud Service Providers rendering Data as a Service adopt and start using the Pseudo-Crypt methods for data privacy, Integrity and security.
4. That this system be applied to communication systems like email, SMS, social media inbox etc to secure messages for adequate confidentiality.

**Suggestions for Future Research**

The following suggestions are made for other investigators to carry out research in this Psuedo-crypt/Biometric authentication area:

The key for salting the encryption is user dependent. A user having different keys for different data may forget or confuse the keys, since it is not stored in the database for security sake, there is need therefore to find out how to manage/retrieve these keys in case of user forgetfulness.

**Hint:** Since biometric data can be converted to digital data, and they are unique for each individual, is there no way they can be used as salt_keys to encrypt data? This will solve the problem of key storage and enhance identification and authentication Accuracy.

## References

A. Juels and B. A. Kaliski Jr., "PORs: Proofs of retrievability for large files", Proc of the 14th Conf on Computer and Communications Security (CCS'07). New York:ACM, pp. 584-597, (2007).

Al Beshri, A. M. (2013) Outsourcing data storage without outsourcing trust in cloud computing. PhD thesis, Queensland University of Technology. Available online at http://eprints.qut.edu.au/61738/(Accessed: June 05, 2014)

Brunette, G. and Mogull, R. (2009) Security Guidance for Critical Areas of Focus in Cloud Computing V2.1. Technical Report. Cloud Security Alliance.

Catteddu, D. and Hogben, G. (2009) Cloud Computing: benefits, risks and recommendations for information security. Technical Report.European Network and Information Security Agency.

Cretu, L. G (2012) Cloud-based Virtual Organization Engineering. *Informatica Economică*vol. 16, no. 1/2012.Available online at http://www.revistaie.ase.ro/content/61/09%20-%20Cretu.pdf (Accessed 12/12/2012).

Delgado, V. (2010): Exploring the limits of cloud computing, KungligaTekniska Högskolan (KTH) Stockholm, Sweden..Master of Science Thesis presented to KTH Information and Communication Technology, Stockholm, Sweden.

D. W. Chadwick and K. Fatema(2012) "A privacy preserving authorization system for the Cloud", Journal of Computer and System Sciences,), vol. 78, no. 5,

D. Zissis and D. Lekkas(2012) "Addressing Cloud Computing Security Issues", Future Generation Computer Systems, Vol. 28, Issue 3, pp.583-592.

E. M. Mohamed, H. S. Abdelkader and S. El-Etriby (2012), "Enhanced data securitymodel cloud computing", Paper presented at the Informatics and Systems (INFOS), 2012 8th International Conference on (2012, 14-16 May 2012). pp. 1359-1373.

G. Ateniese, R. Burns and R. Curtmola(2007) "Provable data possession at untrusted stores", Proc of the 14th Conf on Computer and Communications Security (CCS'O7). New York: ACM,),pp. 598-609.

H. Hacigumus, B. Iyer, L. Chen and S. Mehrotra(2002) "Executing SQL over encrypted data the database service provider model", Proc of the 2002 ACM SIGMODIntConf on Management of Data (SIGMOD'2002). New York: ACM, pp. 216-227.

Higgins, R. (2011). Securing a multi-tenant environment. Available online athttp://searchcloudsecurity .techtarget.com/tip /Securing-a-multi-tenant environment (Accessed: August 16, 2014)

H. Shuai and X. Jianchuan (2011) "Ensuring data storage security through a novel third party auditor scheme in cloud computing", Paper presented at the Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference on (2011, 15-17 Sept. 2011).

http://en.wikipedia.org/wiki/Advanced_Encryption_Standard.

ISACA (2009). Cloud Computing: Business Benefits With Security, Governance and Assurance Perspectives: Emerging Technology White Paper. Available online at http://www.isaca.org/...Center /.../Cloud-Computing-28Oct09-Research.pdfAccessed: 08 June 2011

Imogokate (2011) Cloud Computing! 4 Security Advantages, 5 Characteristics & 10 Benefits – A Presentation for Business Posted on May 16, 2011. Available online at http://imogoblog.wordpress.com/2011/05/16/cloud-computing-4-security-advantages-5-characteristics-10-benefits-a-presentation-for-business/ Accessed: 07 June 2011

J. Taeho, L. Xiang-Yang, W. Zhiguo and W. Meng (2013) "Privacy preserving cloud data access with multi-authorities", Paper presented at the INFOCOM, 2013 Proceedings IEEE, (2013, 14-19 April 2013).

.

J. B. Bernabe, J. M. Marin Perez, J. M. AlcarazCalero, F. J. Garcia Clemente and G. M. Perez (2014) "Semantic- Aware – multitenancy-authorization system for cloud architectures", Future Generation Computer Systems, vol. 32, pp. 154-167.

Jackson, P. (1999). Virtual teams and lost proximity: Consequences on trust relationships. *Virtual Working,* 1 (3), 46 – 56. Available online at http://www.informaworld.com/smpp/content~db=all~content=a729852051?words=virtual*|enterprises* (Accessed: October 25, 2007)

Joshi, H., &Dhyani, P. (2011). Cloud Computing & Rise of Virtual Organizations. Available online at http://www.niit.com/investorrelations/Investor%20Newsletter/Q2Jul2011/document/cloud-computing1.pdf. (Accessed:12/12/2012).

K. Hashizume, D. Rosado, E. Fernández-Medina and E. Fernandez (2013) "An analysis of security issues for cloud computing", Journal of Internet Services and Applications, vol. 4, no. 1, pp.1-13.

L. Popa, M. Yu, S. Y. Ko, S. Ratnasamy and I. Stoica (2010) "Cloud Police: taking access control out of the Network", ACM Sigcomm Workshop, 2010.

Maggi, F. and Zanero, S. Rethinking (2010) security in a cloudy world.Technical report, Dipartimento di Elettronica e Informazione, Politecnico di Milano.

Moore, R. (2005) "Cybercrime: Investigating High-Technology Computer Crime," Cleveland, Mississippi: Anderson Publishing

Ndunagu Juliana Ngozi and OkunadeOluwasogo A. (2012) Diffusion of Cybercrime in the Nigerian Cash-Less Economy: Using Double Level Authentication. Journal of Information and Communication Technologies, ISSN 2047-3168, Volume 2, Issue.

Nolle, T. (2012). Pros and cons of a non-VM-based IaaS model. Available online at http://searchcloudcomputing.techtarget.com/tip/Pros-and-cons-of-a-non-VM-based-IaaS-model (Accessed: August 20, 2012)

PrashantRewagad, YogitaPawar (2012) "Use of Digital Signature and Rijndael encryption Algorithm to Enhanced Security of data in Cloud computing Services", Proceeding published in International Journal of Computer Applications (IJCA)

P. Rewagad and Y. Pawar (2013) "Use of Digital Signature with Diffie Hellman Key Exchange and AES Encryption Algorithm to Enhance Data Security in Cloud Computing. Paper presented at the Communication Systems and Network Technologies (CSNT), 2013 International Conference on. (2013, 6-8 April 2013).

Reilly, D.; Wren, C. & Berry, T. (2011).*Cloud Computing: Pros and Cons for Computer Forensic Investigations:*International Journal Multimedia and Image Processing (IJMIP), Volume 1, Issue 1, March 2011. Available online at http://www.infonomics-society.org/IJMIP/Cloud%20Computing_Pros%20and%20Cons%20for%20Computer%20Forensic%20Investigations.pdf Accessed: 20 May 2011

Sample, C. (2010). IaaS security puts spotlight on hypervisor security and Tenant management. Available online at http://searchcloudsecurity.techtarget.com/tip/IaaS-security-puts-spotlight-on-hypervisor-security-tenant-management(Accessed: August 16, 2014)

Shinder, D. (2011). Security considerations for Infrastructure as a Service cloud computing model. Cloud Computing.Available online at http://www.windowsecurity.com/articles/Security-Considerations-Infrastructure-Service-Cloud-Computing-Model.html (Accessed: August 16, 2014)

SanjoliSingla, Jasmeet Singh (2013) "Survey on Enhancing Cloud Data Security using EAP with Rijndael Encryption Algorithm", Global Journal of Computer Science and Technology (GJCST), Vol. 13, Issue 5.

S. Kamara and K. Lauter(2010), "Cryptographic cloud storage", LNCS 6054:Financial Cryptography and Data Security. Berlin: Springer, pp. 136-149.

S. K. Sood (2012) "A combined approach to ensure data security in cloud computing", Journal of Network and Computer Applications, vol. 35, no. 6, pp. 1831-1838.

Siddiqui, M. (2011).*Cloud Computing Security:* Final paper submittedspring 2011. Available online at http://blogs.techconception.com/manny/content/binary/Manny%20Siddiqui%20-%20Cloud%20Computing%20Security.pdf (Accessed: 20 May 2011).

Santos, N., Krishna, P., & Rodrigues, G. (nd). Towards Trusted Cloud Computing. *MPI-SWS Available online at http://*www.mpisws.org/~gummadi/papers/trusted_cloud.pdf. (Accessed: 07 June 2011)

Sample, C. (2012). IaaS security puts spotlight on hypervisor security, tenant management. Available online at http://searchcloudsecurity.techtarget.com/tip/IaaS-security-puts-spotlight-on-hypervisor-security-tenant-management(Accessed: August 16, 2012)

Tao Sun and Xinjun Wang (2013) International Journal of Security and Its Applications Vol.7, No.6 Research of Data Security Model in Cloud Computing Platform for SMEs

U. Khalid, A. Ghafoor, M. Irum and M. AwaisShibli(2013) "Cloud Based Secure and Privacy Enhanced Authentication and Authorization Protocol", Procedia Vol.22, pp. 680-688.

W. C. David (2010) "Cloud computing Key Initiative Overview", http://www.gartner.com/resources/ 173600/173626/cloud_computing_key_initiati_173626.pdf.

Warren G. Kruse, Jay G. Heiser (2002) Computer forensics: incident response essentials Addison-Wesley. p. 392. ISBN0201707195

Y. Ching-Nung and L. Jia-Bin (2013) "Protecting Data Privacy and Security for Cloud Computing Based on Secret Sharing", Paper presented at the Biometrics and Security Technologies (ISBAST), Internationa Symposium on. (2013, 2-5 July 2013).

Z. Lan, V. Varadharajan and M. Hitchens(2013), "Achieving Secure Role-Based Access Control on Encrypted Data in Cloud Storage", Information Forensics and Security, IEEE Transactions on, vol. 8, no. 12, pp. 1947-1960.