Volume 8, No. 5, May-June 2017
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Cryptography Based Security for Cloud Computing System

Prerna
Department of Computer Science and Engineering
Jamia Hamdard
New Delhi, India
prerna.prasad21@gmail.com

Parul Agarwal*
Department of Computer Science and Engineering
Jamia Hamdard
New Delhi, India
pagarwal@jamiahamdard.ac.in

Abstract: In this age of cloud computing, we tend to store data which we need frequently in web based cloud storage services so that it can be accessed whenever we need them. This not only provides an immense amount of flexibility to users but also makes our content accessible to us wherever we are and whenever we need them. We have many options in terms choosing web based cloud storage services for backing and archiving our data. There are many web based cloud storage services available out of which Amazon S3 and Google Drive are immensely popular among users. Backing up files so that they are not lost is an all-important step to ensure that nothing is ever lost. But, moving to the cloud is itself a big change and there are real concerns that make people pause before they sign up for any such service.This paper proposes and implements an algorithm which would encrypt the files uploaded on such web based cloud storage services and would decrypt the file once it has been downloaded using the keys that were generated during encryption. This would prevent unwanted intrusion into personal data and lack of standardization, i.e. one service provider may have end-to-end encryption while others do not.

Keywords: Cloud Computing, Encryption, Decryption, Standardization, Cloud Storage

I. INTRODUCTION

Cryptography can be referred to as the practice and study of hiding and securing information. It is the science of keeping information secret and safe. There has always been an urge among humans to keep sensitive information safe and secure so that it would not lead to unwanted intrusion into sensitive data which could lead to serious problems. Thus, cryptography has been practised by humans from ancient times to keep their information secure.

The earliest known use of cryptography was carved cipher texts on stone in Egypt. We know of the Caesar cipher, in which each letter in the plaintext was replaced by a letter which was a fixed number of positions down the alphabet [1][14]. The Egyptians used hieroglyphics, which were symbols used to decorate tombs of the deceased Pharaohs [2]. Even the Greeks were aware of cryptography and steganography. According to Herodotus, the messages were tattooed on a slave's shaved head and concealed under his regrown hair [3]. The Romans did know something of cryptography (e.g., the Caesar cipher and its variations). There is an ancient mention of a book about Roman military cryptography (especially Julius Caesar's), but, it has been lost [24][25]. In Europe during and after the Renaissance , citizens of various Italian states , including the Papacy was responsible for improvements in cryptographic practices (e.g. , polyalphabetic ciphers invented by Leon Alberti in the year 1465) [25].

Cryptography is one of the oldest fields of technical study that goes back almost 4,000 years[13]. The first known evidence of the use of cryptography was found in an inscription carved around 1900 BC [1].In today's times cryptography is as relevant as it was centuries ago because the digital revolution which started towards the end of the last century has introduced vulnerabilities which have often led to massive data loss and data theft and thus cryptography is an efficient way to deal with such vulnerabilities which could possibly lead to data loss and data theft.

The digital revolution has led to the rise of cloud computing, which is nothing but storing and managing data in remote servers on the Internet. The problems that we come across as far in cloud computing are cyber attacks, government intrusion, lack of standardisation and outages [4][16]. Therefore, encrypting our files before storing them in web based cloud storage services is an efficient way to overcome most of these problems.

II. EXISTING TECHNIQUES

Google Drive is a service that lets us store personal files on the cloud. Google Drive encrypts data using TLS (Transport Layer Security) standard even before it leaves the device. It is then uploaded to the drive. When the data reaches Google it is unencrypted and then re encrypted using 256-bit AES (Advanced Encryption Standard). The AES encryption keys used to encrypt the data are further encrypted with rotating master keys which adds a extra second layer of security, thus making the data more secure [5][17]. This process is simply reversed when we get data from Google Drive. Cloud Storage also allows us to enable versioning using which a history of modification and changes of all objects is kept in the bucket [18].

Amazon S3 stores objects redundantly across multiple facility in an Amazon S3 region. This redundancy helps is repairing data if there is a data corruption issue. In addition Amazon S3 also uses versioning to preserve every version of every object stored in our Amazon S3 bucket. Versioning allows us to easily recover from unintended user actions and application failures [6].

The server side encryption used by Amazon while the data is at rest i.e. stored in disks at Amazon S3 data centres, is similar to that to that of Google and it uses 256-bit AES to encrypt the data [20].

Although most of the service providers maintain high standards of encryption but encrypting data while when it is moved internally i.e. between the service providers own datacentres and also encrypting data in transit i.e. while the

data moves to and from the service providers remains an issue.

Also, there exists lack of standardization, i.e. one service provider may have end-to-end encryption while others do not [7].

Currently lack of resources/expertise is the number one cloud challenge (as of 2016) but security is the second most important concern when it comes to the cloud [29].

III. OUR CONTRIBUTION

A symmetric key encryption provides secrecy when two parties communicate. Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key [26]. The two parties first agree on a key and keep it secret. Before sending a message from one party to another the message is encrypted using the encryption algorithm and the key [8]. The cipher text obtained is sent to the other party who decrypts it using the decryption algorithm and the same key. The encryption and decryption algorithms are known, but the key is kept secret [9].

In this form of cryptography, the key must be known to both the sender and the receiver but the distribution of the key is the biggest difficulty [28].

We have developed and implemented a symmetric key algorithm where data is encrypted at client side and uploaded to a web based cloud storage service. Here we manage the encryption process and encryption keys. When the data is downloaded from the cloud storage service we decrypt it using the encryption keys.

The main aim of this algorithm is to secure the data while in transit, although SSL(Secure Sockets Layer) is used to keep the data private by establishing an encrypted link between a web server and a browser while the data is in transit [23] but by encrypting the data before it is sent provides an extra layer of security. Also, many service providers do not encrypt data when it is moved between their own datacentres which can lead to government intrusions, data loss and privacy risks, risk of intellectual property theft[27] and spying efforts and also many service providers do not have end-to-end encryption. Therefore, encrypting data at client side before it is uploaded to cloud storage service can help to deal with such threats.

A. Encryption algorithm

- Extract each character from a file and get its corresponding ASCII value.
- Convert the ASCII value to the corresponding binary value
- Check if the binary value is 8 bits or not.
- If not then add preceding 0's to make it a 8-bit binary value.
- Reverse the corresponding 8-bit binary value.
- Extract the first 4 bits from the reversed 8-bit binary value and reverse them.
- Similarly extract the last 4 bits and reverse them as well.
- Append the 4 bit binary values obtained in steps 6 and 7.
- The 8 bit binary value obtained after appending in step 8 is the cipher text.
- Convert this 8 bit binary value to ASCII and write the corresponding character to the encrypted file.
- The key is generated by adding 10 to the ASCII value in step 10, and the corresponding character is written to a separate encryption key file.

B. Example

Let the character be 'P', accordingly we will get:

- ASCII value of 'P' is 80.
- The binary value of 80 is 1010000.
- The value after appending a '0' to make the value 8 bit binary is 01010000.
- After reversing the 8 bit binary what we get is 00001010.
- After appending the first 4 bits and last 4 bits of the binary value what we get is 00000101.
- On converting 00000101 to decimal we get 5.
- The character corresponding to ASCII value 5 is the cipher text.
- Add 10 to 5 (ASCII value for cipher text).
- The character corresponding to ASCII value 15 is the key.

C. Decryption Algorithm

- Extract each character from the encrypted file and get its corresponding ASCII value.
- Get the ASCII value of each character from the encryption key file and subtract 10 from it.
- Check if the values in steps 1 and 2 are same or not.
- If they are not same then decryption will not be performed.
- If they are same decryption will be performed by reversing the encryption algorithm, i.e., by converting encrypted character to corresponding ASCII value and then from ASCII value to 8 bit binary value, breaking the binary value to 4 bits, reversing them individually and appending them and the reversing the appended binary value.
- The decrypted character is written to a separate decryption file which should be same with the content of the original file.

D. Example

- The ASCII value for the cipher text is obtained, which is 5.
- On subtracting 10 from 15(the ASCII value of the key), we get 5.
- Check for equality of values in steps 1 and 2 and update a count variable, initially set to 0.
- Here, after the equality check, count will be 1 (which is equal to the number of characters in encrypted and key file).
- Now 5 (cipher text ASCII value) is converted to 8 bit binary value which is 00000101.
- The first 4 bits and last 4 bits of the 8 bit binary value is are reversed, which gives 00001010.
- The 8 bit binary value obtained in the previous value is reversed, which gives 01010000.
- The decimal value of the corresponding binary value 01010000 is 80.
- The ASCII value for 80 is 'P'.(Original text that was encrypted).

E. Amazon S3

We have used Amazon S3 as the web based cloud storage service to test and implement this project. Amazon S3 is an object storage, which is simple, durable and scalable and with a simple web service which allows users to store and retrieve any amount of data, anytime and from anywhere on the web [10].

We upload objects to the Amazon S3 buckets after creating a bucket in any one of the AWS (Amazon Web Service) regions. We created our buckets in EU_WEST 1(Ireland) region. There are other regions as well like , EU_WEST 2(London) , US_EAST 1(North Virginia) , US_WEST 1 (North California) and others [21].

Amazon S3 is a key, value store designed to store as many objects as we need. Keys are the name we assign to an object. We use the key's name to retrieve the object. Value is the content that we are storing, which can range from 0 to 5 TB [11]. Every user can create upto 100 buckets and the bucket names are global [19].Every object resides in a bucket and a bucket represents a collection of objects. Each bucket is referred by a key (name), which is unique [15].

F. Functional Flow

- Initially a bucket and folder is created in our Amazon S3 account.
- A file is chosen which is to be encrypted and the encryption algorithm is executed on this file.
- The encrypted file and encryption key file is generated on our local systems.
- The encrypted file is uploaded to the bucket and folder we created in step 1.
- We download our encrypted files by specifying the key (name we assign to an object) for that file.
- The encrypted file will be downloaded to our local file system, we have to select the correct encryption key file for the encrypted file and execute the decryption algorithm on the encrypted file.
- The content of the decrypted file will be same as the original file.
-

G. Implementation Screenshots

- **Bucket and Folder Creation:**

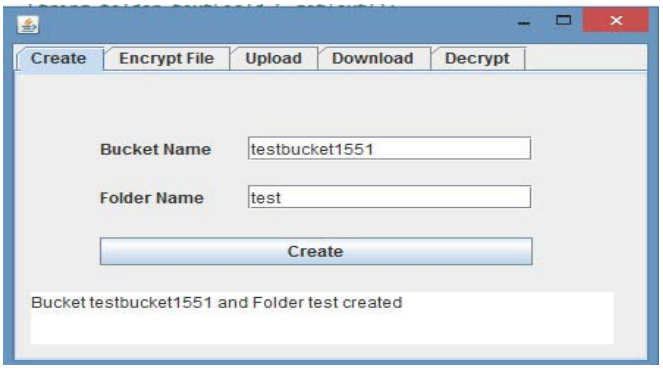

Figure 1. Bucket 'testbucket1551' and Folder 'test' creation in Amazon S3

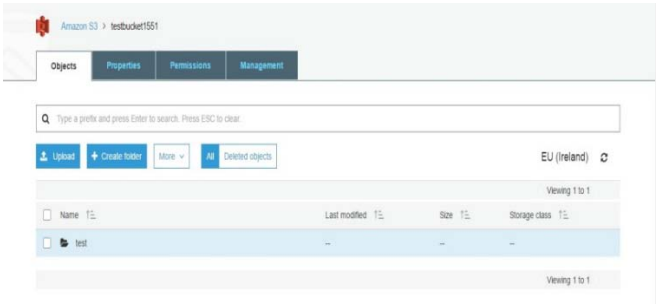

Figure 2. Bucket 'testbucket1551' and Folder 'test' created in Amazon S3

- **File Encryption:**

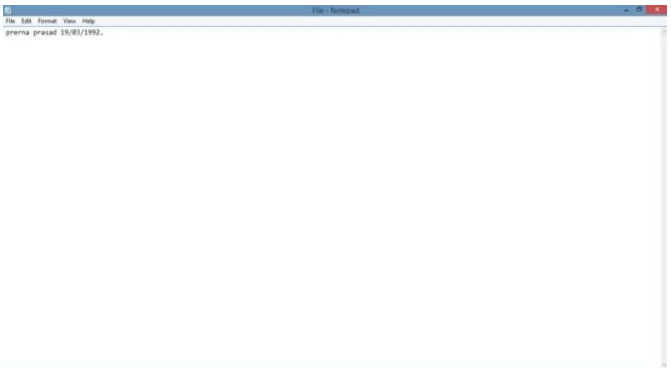

Figure 3. Original File (File.txt) which is to be encrypted

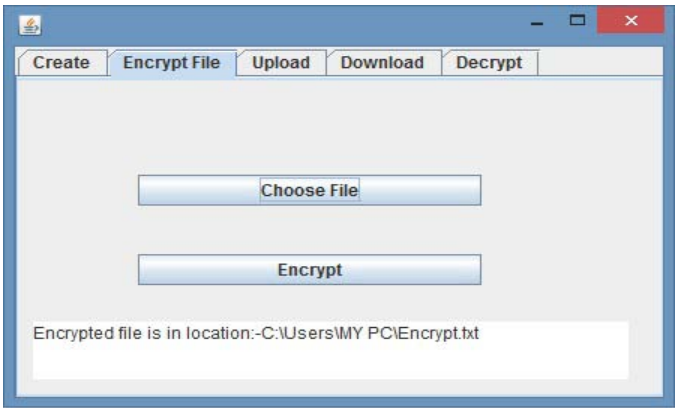

Figure 4. File (File.txt) encrypted as Encrypt.txt and present at C:\Users\MY PC\Encrypt.txt, also a Key (Key.txt) is generated during encryption at the same location

Figure 5. Content of the Encrypted File (Encrypt.txt) present at C:\Users\MY PC\Encrypt.txt

- **Encrypted File Upload :**

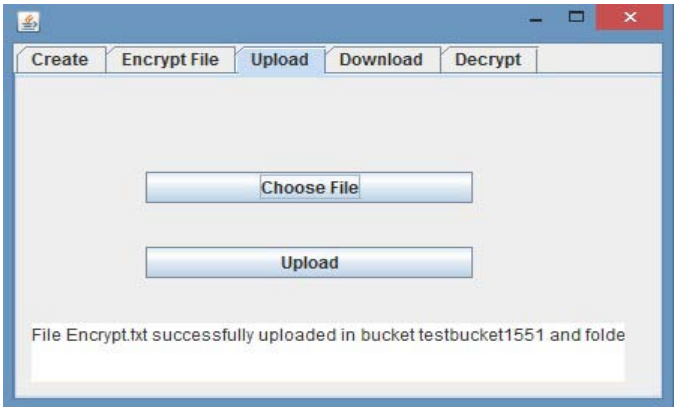

Figure 6. Uploading Encrypted File (Encrypt.txt) in folder 'test' inside bucket 'testbucket1551'

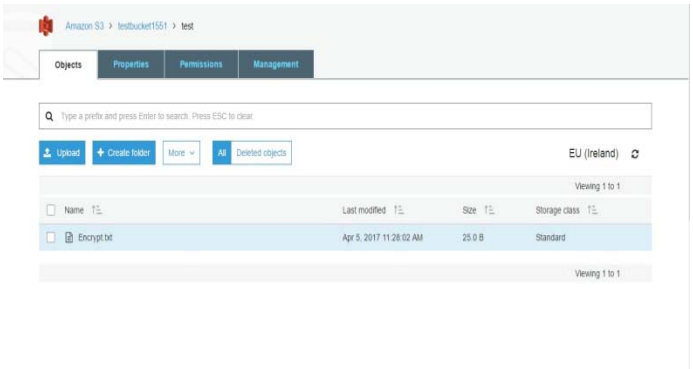

Figure 7. Encrypted File (Encrypt.txt) uploaded to the folder 'test' inside bucket 'testbucket1551' in Amazon S3

- **Encrypted File Download :**

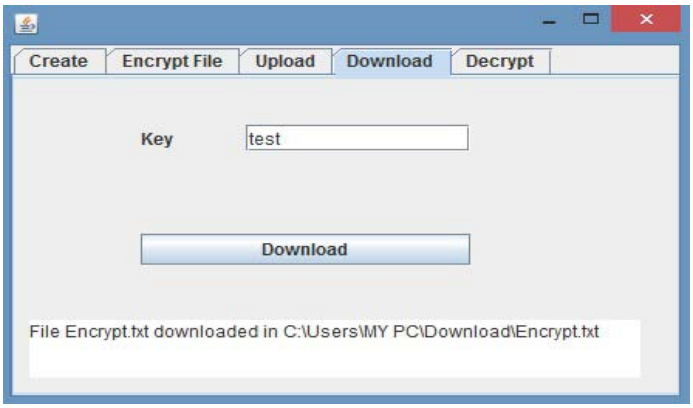

Figure 8. File (Encrypt.txt) downloaded from folder 'test' in bucket 'testbucket1551' to C:\Users\MY PC\Download\Encrypt.txt

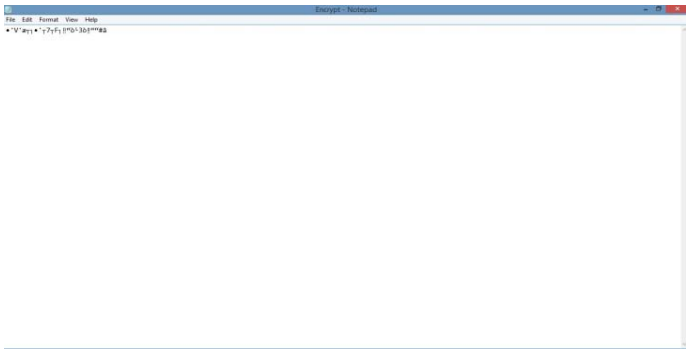

Figure 9. Content of the downloaded Encrypted File (Encrypt.txt) which is in C:\Users\MY PC\Download\Encrypt.txt

- **File Decrypt :**

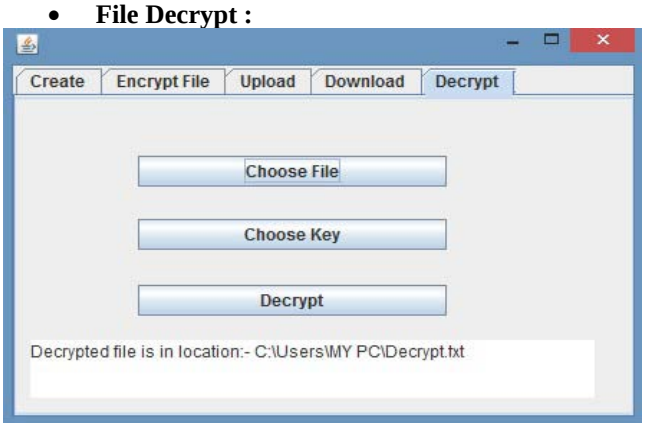

Figure 10. Encrypted File (Encrypt.txt) which was downloaded from Amazon S3 is decrypted using the Key (Key.txt). The Decrypted File (Decrypt.txt) is located at C:\Users\MY PC\Decrypt.txt

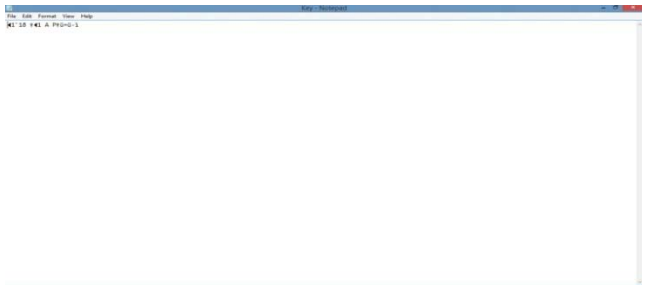

Figure 11. Content of the Key File (Key.txt), which was produced during encryption and is required during the decryption process

Figure 12. Decrypted File (Decrypt.txt) after the decryption process, the content of which is same as that of the original file (File.txt)

IV. RESULTS AND CONCLUSION

In this paper we have proposed and implemented a symmetric key encryption algorithm to encrypt data at client side before uploading it to a cloud storage service.

The main aim of this paper was to propose and implement an algorithm so that data can be encrypted at client side before it is uploaded to a cloud storage service because it provides an extra layer of security, minimises data theft in transit, minimises data intrusion and spying when data in moving within data centres of the service provider and also solves the problem of lack of standardisation [12] ,where some service providers guarantee end-to-end security but in reality their services are not secure. Hackers often trick a cloud into treating their illegal activity as a valid activity , and gain unauthorized access to the information stored in the cloud [22].

This algorithm has been currently tested for text files especially, for which the encryption and decryption processes worked as expected. But, it could be further enhanced for encrypting other file formats or even audio and video files and even larger files could be encrypted at client side before uploading to the cloud.

V. ACKNOWLEDGEMENT

I wish to thank my family for supporting me in this endeavour and also my guide, Ms. Parul Agarwal for her valuable suggestions and inputs without which it would have been difficult to proceed further.

VI. REFERENCES

1. A Brief History of Cryptography, an article available at https://access.redhat.com/ blogs/ 766093/ posts/ 1976023, March 2016.
2. [2] Advances in Cryptography by Dara Kirschenbaum, History of Mathematics, Rutgers, Spring 2000.
3. [3] The Art Of Cryptology: From Ancient Number System to Strange Number System by Debasis Das, U.A. Lanjewar, S.J. Sharma, International Journal of Application or Innovation in Engineering and Management, Volume 2, Issue 4, April 2013.
4. [4] Cloud Computing – Understanding Risk, Threats, Vulnerability and Controls: A Survey, Manish M. Potey,C A Dhote, Deepak H. Sharma, International Journal of Computer Applications, Volume 67– No.3, April 2013.
5. [5] Encryption At Rest In Google Cloud Platform, an article available at https://cloud.google.com/security/encryption-at-rest/default-encryption/ , April 2017.
6. Protecting Data Using Encryption, an article available at http://docs.aws.amazon.com/AmazonS3/latest/dev/UsingEncryption.html .
7. Top Ten Major Risks Associated With Cloud Storage, an article available at https://www.cloudwards.net/top-ten-major-risks-associated-with-cloud-storage/ , August 2015.
8. Introduction to Cryptography, Principles and Applications by Delfs, Hans, Knebl and Helmut.
9. Foundations of Security by Neil Daswani, Christoph Kern and Anita Kesavan.
10. What is Amazon S3 ? , an article available at http://docs.aws.amazon.com/AmazonS3/latest/dev/Welcome.html .
11. Working With Amazon S3 Buckets, an article available at http://docs.aws.amazon.com/AmazonS3/latest/dev/UsingBucket.html .
12. Top 10 Security Concerns for Cloud-Based Services an article by Joy Ma available at https://www.incapsula.com/blog/top-10-cloud-security-concerns.html , December,2015.
13. A Short History Of Cryptography, an article by Fred Cohen.
14. Past, Present, and Future Methods Of Cryptography And Data Encryption, A Research Review by Nicholas G. McDonald, Department of Electrical and Computer Engineering, University of Utah.
15. Creating, Listing and Deleting Amazon S3 Buckets, an article available at http://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/examples-s3-buckets.html.
16. What are the 12 biggest cloud computing security threats? , an article by Matthew Wilson available at https://www.ibm.com/blogs/cloud-computing/2016/04/12-biggest-cloud-computing-security-threats/ , April 2016.
17. Managing Data Encryption, an article available at https://cloud.google.com/storage/docs/encryption#rotating-keys , January 2017.
18. Object Versioning, an article available at https://cloud.google.com/storage/docs/object-versioning , April 2017 .
19. An Evaluation of Amazon's Grid Computing Services: EC2, S3 and SQS by Simson L. Garfinkel, Computer Science Group, Harvard University, Cambridge, Massachusetts.
20. Protecting Data Using Server-Side Encryption with Amazon S3-Managed Encryption Keys , an article available at http://docs.aws.amazon.com/AmazonS3/latest/dev/UsingServerSideEncryption.html .
21. AWS Regions and Endpoints , an article available at http://docs.aws.amazon.com/general/latest/gr/rande.html .
22. Security Threats On Cloud Computing Vulnerabilities by Te-Shun Chou , Department of Technology Systems, East Carolina University, Greenville, NC, U.S.A , International Journal of Computer Science & Information Technology (IJCSIT) Vol 5, No 3, June 2013 .
23. An analysis of security issues for cloud computing by Keiko Hashizume , David G. Rosado , Eduardo Fernandez-Medina , Eduardo B Fernandez , Journal of Internet Services and Applications,2013 .
24. The Cryptography, an article available at http://www.divini.net/tlm3/products0708/mathspedia/it/cryptography.pdf .
25. Cryptography / History , an article available at https://www.saylor.org/site/wp-content/uploads/2011/03/History.pdf
26. Study on symmetric key encryption: An Overview by Dharitri Talukdar, International Journal of Applied Research 2015 .
27. Advances and Trends in Cryptography an article by Dr. Tomislav Nad , SIGS Technology Summit , June 2015 .
28. An Overview of Cryptography an article by Gary C. Kessler, Embry-Riddle Aeronautical University - Daytona Beach, March 2016 .
29. 6 Cloud Computing Challenges Businesses are Facing These Days an article by Mona Lebied in Business Intelligence , January 2017