SECURING DATA EXCHANGE THROUGH CLOUD WITH CRYPTOGRAPHY

Vishwam Chepuri
Student,
Electronics and Communication Engineering
NIT Warangal
vchepuri@student.nitw.ac.in

Srinandh Ramidi
Student,
Electronics and Communication Engineering
NIT Warangal
rsrinandh@student.nitw.ac.in

Abstract*— *As business regulations and information security expand at an asymmetrical pace, corporate executives often end up facing privacy and security challenges because they do not have the knowledge or experience to address. While encryption is the baseline technology that privacy experts agree is the cornerstone of security, encryption in the cloud can be daunting. So the user is left with two ways to maintain integrity of his data when it is passed over the cloud, selecting a cloud provider that will allow the customer to encrypt the data before it is sent to the cloud for storage or processing, or partner with a software as a service (SaaS) provider that will manage the encryption and decryption of the data. The proposed work focuses on implementation of bidirectional Hex Cryptography to use with Unicode characters.

Keywords—Data security issues, Bidirectional Hex Cryptography, Cloud computing

I. INTRODUCTION

Cloud computing has recently reached popularity and developed into major trend in IT. One of the major challenges faced in Cloud computing is data privacy. Sensitive and personal information that is kept in the cloud should be defined as being for internal use only, not to be shared with third parties. Businesses must have a plan to securely and efficiently manage the data they gather. So Cryptography plays a vital role to maintain integrity of data, There are mainly two types of cryptographic algorithms namely Secret Key cryptography, which uses single key for both encryption and decryption, also called symmetric encryption and Public Key Cryptography, which uses one key for encryption and another key for decryption, also called asymmetric encryption. here we implement a two level Bi directional Hex symmetric encryption which helps in enhancing the protection of data from third party applications. The technique proposed here uses same key for both encryption and decryption.

II. LITERATUE SURVEY

In cloud computing the major issue is to provide the security of data. In Cloud computing data security is achieved by the Authentication, Encryption & Decryption, Message authentication code, Hash function, and Digital signature and so on. So here we discuss about some security problems and their solutions.

Use of Digital Signature with Diffie Hellman Key Exchange and AES Encryption Algorithm to Enhance Data Security in Cloud Computing [1].

-Mr.PrashantRewagad and Ms.YogitaPawar [1].
Here in this paper, the researcher used three way architecture protection schemes. Firstly Diffie-Hellman algorithm is used to generate keys for key exchange step. Then digital signature is used for authentication, thereafter AES encryption algorithm is used to encrypt or decrypt user's data file. Diffie-Hellman key exchange algorithm is vulnerable to main in the middle attack. The most serious limitation is the lack of the authentication.

Implementation of Digital signature with RSA Encryption algorithm to enhance the Data security of cloud in Cloud Computing [3].

-Uma Somani, Kanika Lakhani, and Manish Mundra [3].
In this paper, there are two enterprises A and B. An enterprise A has some data that are public data and enterprise has public cloud. Now B wants some secure data from A's cloud. So RSA algorithm and Digital signature are used for secure communication. In this method, enterprise A takes data from cloud, which B wants. Now the data or document is crushed into little line using Hash code function that is called Message digest. Then A encrypts the message digest within private key and the result is in the Digital signature form. Using RSA algorithm, A will encrypt the digital signed signature with B's public key and B will decrypt the cipher text to plain text with his private key and A's public key for verification of signature.

III.PROPOSED WORK

Previous section describes about cloud computing and importance of encryption and decryption in exchange of data through cloud. Here in this paper bidirectional Hex encryption algorithm is performed to provide two level of security.

HEX encryption algorithm:

In this algorithm , Binary data is grouped into four bits and then each group is assigned an alphabet as shown below:

Binary value	Code
0000	A
0001	B
0010	C
0011	D
0100	E
0101	F
0110	G
0111	H
1000	I
1001	J
1010	K
1011	L
1100	M
1101	N
1110	O
1111	P

Here in this work, we are using 16 alphabets namely A, B , C, D, E, F, G, H, I, J, K, L, M, N, O, P as a key. Every bit have 4 bits like A=0000, B =0001, C=0010, D=0011, E=0100, F=0101, G=0110, H=0111, I=1000, J=1001, K=1010, L=1011, M=1100, N=1101, O=1110, P=1111. Unicode data is initially represented using above alphabets according to table shown above. According to key given in table below, data is encrypted and is sent to cloud. This key is shared between sender and receiver so that receiver can decrypt it from the cloud.

Table 2: Key Combination

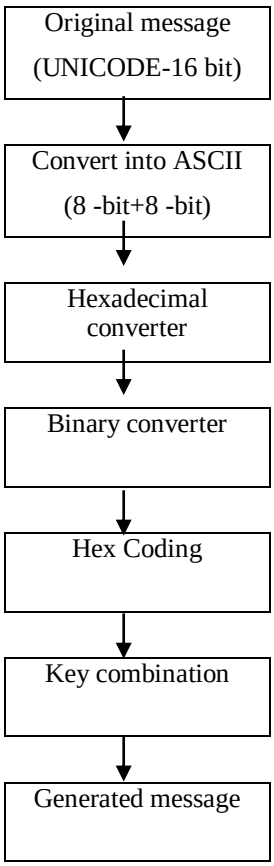
Key Combination	Binary Value	Value
A	0101	5
B	1001	9
C	1101	13
D	0001	1
E	1111	15
F	0110	6
G	1010	10
H	0000	0
I	1000	8
J	1100	12
K	1110	14
L	0010	2
M	1011	11
N	0100	4
O	0111	7
P	0011	3

Above table can be used to achieve two level encryption as described below.

Data 1011 is initially encoded into L, then L is encrypted into 0010. As it can be observed data is securely encrypted by using two stages. The major issue in this procedure is sharing the key between sender and receiver. The key between sender and receiver is shared through diffie-Hellman key sharing algorithm. The key shown in above table gets updated for every data transfer, this process ensures that algorithm is safe and secure.

Encryption is divided into number of stages where first stage consists of conversion from Unicode to ascii, second stage consists of conversion from ascii to Hexadecimal, third stage consists of conversion from Hexadecimal to binary which are shown below.

ENCRYPTION PROCESS:



As it can be seen from above encryption flow chart, there are six stages that original message should pass to generate encrypted message. To understand clearly, we take an example Unicode and perform encryption on it as shown below.

Unicode:

à³†à³¶à³¿à³·

ASCII:

\u0e0\u0aa\u02020\u0e0\u0aa\u0b6\u0e0\u0aa\u0bf\u0e0\u0a
a\u0b7

Hexadecimal value:

5c753065305c753061615c7530323032305c753065305c7530
61615c753062365c753065305c753061615c753062665c7530
65305c753061615c75306237

Binary value:

0101110001110101001100000110010100110000010111000111010100110000011000001011000010101110001110101001100000110000101100001010111000111010100110000011000010110000101011100011101010011000001100001011000010101110001110101001100000110001000110110011101110000011001010011000001100010011001100101110001110101001100000101100001110101001100000110000101100001010111000111010100110000011000010110000101011100011101010011000001100001011000010101110001110101001100000110001000110111

Hex coding:

FMHFDAGFDAFMHFDAGBGBFHBNEMAMIMAMIMBHBNEMBJEMBHBNEMBIFIFHBNEMBIINJHBNEMBJEMBHBNFDADBGBGBFMHFDAGCGGFMDKJIDJEMBHBNEMBIFIFHBNEMBIGH

Generated Message:

0110101100000110000101010110101100000110000101011010100110101001011000001001010011111011010110111000101110101101110001011100100000100101001111101110011100111110111100100000100101001111101110011000010000100110000001001010011111011100111001111101110010000100101001111101110011000010000100110000001000101010001100110101001101010010110101100000110000101010110101101110101010011010110000111101100100000011100111110111001000010010100111111011100110000110100001100101010011111011100110000100001001100000010010100111110111001100001000010011000000100010101000110011010100110101001011010110000011000010101101101110101010011010110000111101100100000011100111110111001000010010100111111011100110000110100001100101010011111011100110000100001001100000010

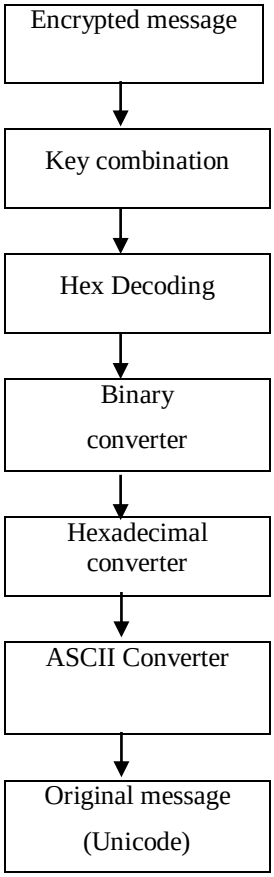
Key combination

Generated message

DECRYPTION PROCESS:

Generated message along with key is received at receiver side, flow chart of decryption process is shown below.

Flow Chart for Decryption process:



Encrypted message:

01101011000001100001010101101011000001100001010110
10100110101001011000001001010011111011010110111000
10110101101110001011100100001001010011111011100111
00111110111001000010010100111110111001100001101000
01100000100101001111101110011000100001001100000010
01010011111011100111001111101110010000100101000110
00010101000110011010100110101001011010110000011000
01010110101101101010100110101100011110110010000001
11001111101110010000100101001111101110011000011010
00011000001001010011111011100110001010000
(As obtained in above example)

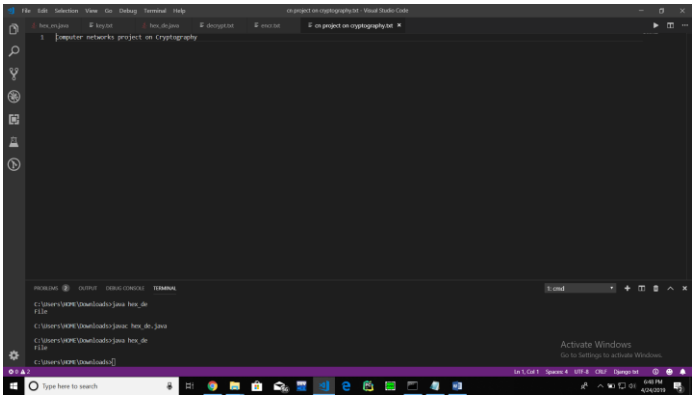
This message is passed through above stages and finally original message is recovered.

Recovered message:

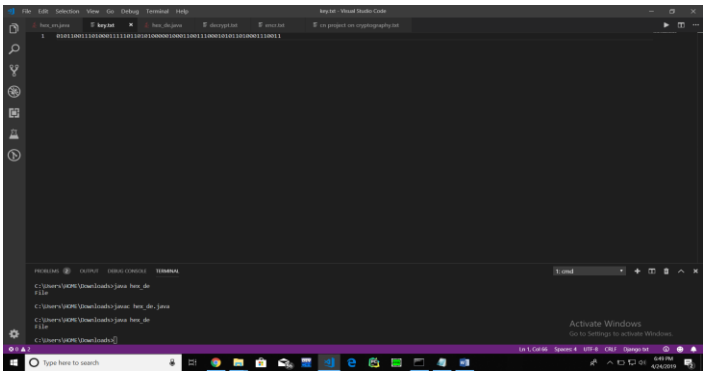
àª†àª¶àª¿àª·

OBSERVATIONS:

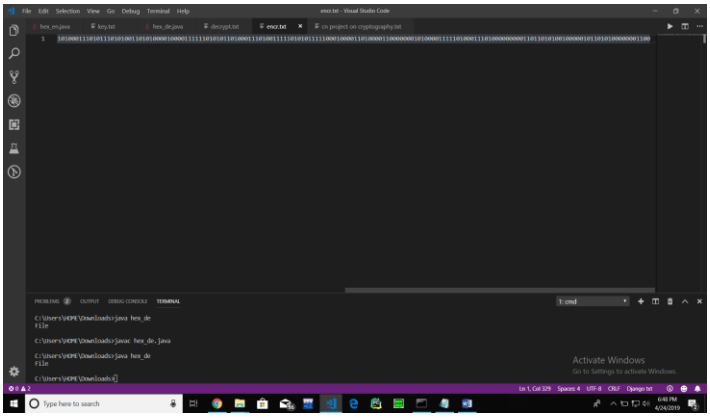
File to be encrypted:



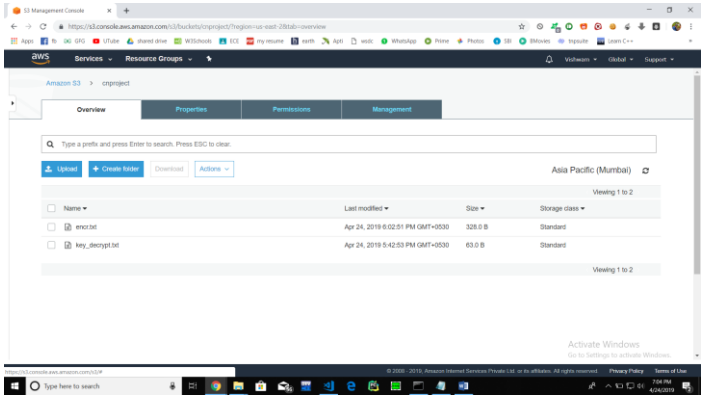
Key:



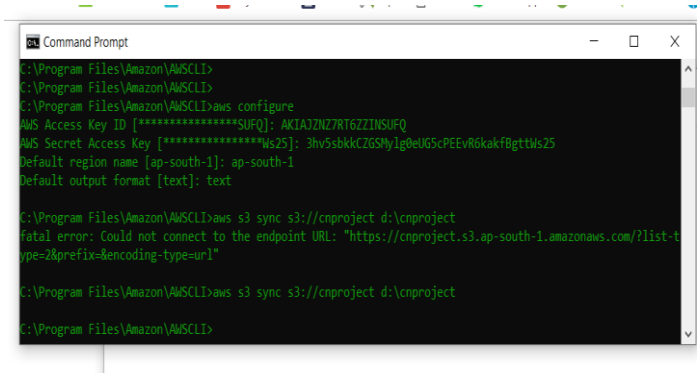
Encrypted file:



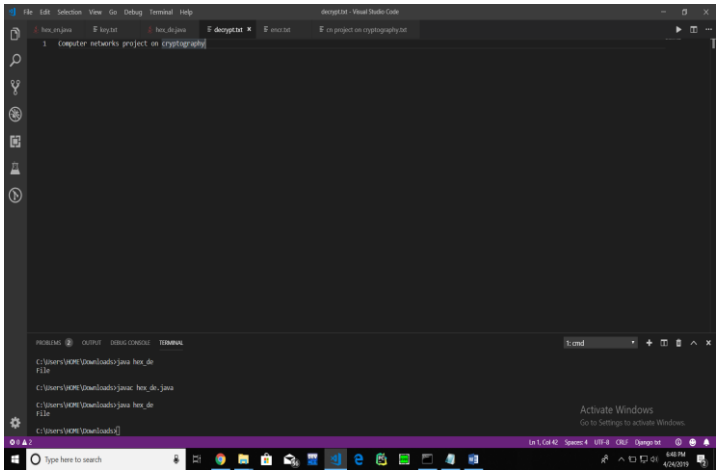
Uploading to AWS cloud:



Setting up AWS:



Decrypted file:



REFERENCES:

[1] PrashantRewagad, YogitaPawar, “Use of Digital Signature with Diffie-Hellman Key Exchange and AES Encryption Algorithm to Enhance Data Security in Cloud Computing” 2013 International Conference on Communication System and Network Technologies (IEEE Computer Society).

[2]UmaSomani,KanikaLakhani,ManishaMundra,’’Implementi ng Digital Signature with RSA Encryption Algorithm to Enhance the Data Security of Cloud in Cloud Computing’’- 2010 IEEE 1st International Conference on Parallel, Distributed and Grid Computing (PDGC-2010).