Improvement of CBC Encryption with Combination of Merkle - Hellman Knapsack Cryptosystem with Shifting Cipher Algorithm

Alisha Anjum Aleem
Masters of Computer Science
Chicago, USA
aanjum1@hawk.iit.edu

Purna Sahithi Adduri
Masters of Computer Science
Chicago, USA
padduri@hawk.iit.edu

***Abstract*—The Merkle-Hellman invented in 1978 is based on the superincreasing subset problem. Ralph Merkle and Martin Hellman used the subset problem to create a cryptosystem to encrypt data. A superincreasing knapsack vector s is created and the super-increasing property is hidden by creating a second vector M by modular multiplication and permutation. The vector M is the public key of the cryptosystem and s is used to decrypt the message. This paper demonstrates how to strengthen the encrypted message being sent by use of discrete logarithms so that only the intended recipient of the message is able to decipher the message.**

***Keywords*—Security; cryptography; cryptosystem; knapsack problem; superincreasing vector.**

I. Introduction

In CBC mode, The first block of plaintext is XORed with the Initialization vector and its encrypted with the symmetric key. Each block of plaintext is XORed with the previous ciphertext block before being encrypted. This way, each ciphertext block depends on all plaintext blocks processed up to that point. To make each message unique, an initialization vector must be used in the first block. Here the processing of the sequence of plain text blocks is chained together. The input to the encryption function for each plaintext block bears no fixed relationship to the plain text block.

In the knapsack problem, It is an NP-complete problem in combinatorial optimization. In Knapsack Problem, Given a set of items, each with a weight and a value, determine the number of each item included in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible. It derives its name from the problem faced by someone who is constrained by a fixed-size knapsack and must fill it with the most valuable items.Merkle–Hellman is an asymmetric-key cryptosystem, meaning that two keys are required for communication: a public key and a private key. Furthermore, unlike RSA, it is one-way: the public key is used only for encryption, and the private key is used only for decryption. Thus it is unusable for authentication by cryptographic signing.

The Merkle–Hellman system is based on the subset sum problem (a special case of the knapsack problem). The problem is as follows: given a set of numbers A and a number b, find a subset of A which sums to b. In general, this problem is known to be NP-complete. However, if the set of numbers (called the knapsack) is superincreasing, meaning that each element of the set is greater than the sum of all the numbers in the set lesser than it, the problem is "easy" and solvable in polynomial time with a simple greedy algorithm.

This can be denoted as –

$$\text{Maximize} \sum_{i=0}^{n} b_i x_i \qquad (1)$$

$$\text{Subject to} \sum_{i=0}^{n} w_i x_i \leq W \qquad (2)$$

$$x_i = \begin{cases} 1, \text{ if the item is included in the knapsack} \\ 0, \text{if the item is not included in the knapsack} \end{cases} \quad (3)$$

where,

'b' is the value associated with each item i
'w' is the weight associated with each item i
'W' is the maximum capacity of the knapsack
'n' is the number of items

The subset sum problem is a special case of the knapsack problem. This problem finds a group of integers from a list vector V, where V = (v1, v2, v3, … vn), with the subset of elements in the vector V gives a sum of S. It also determines if a vector X = (x1, x2, x3… xn) exists where xi element of {0,1} so that V*X = S [5]. Then A superincreasing knapsack vector s is created and the super-increasing property is hidden by creating a second vector M by modular multiplication and permutation. Vector M is the public key of the cryptosystem and is used to decrypt the message [2].

Meanwhile, Caesar Cipher is a type of shift cipher. Shift Ciphers work by using the modulo operator to encrypt and decrypt messages. The Shift Cipher has a key K, which is an integer from 0 to 25. We will only share this key with people that we want to see our message. For Shift Cipher encryption: For every letter in plaintext, We convert the letters a-z to 0-25 and call this number X, calculate Y= (X+K) mod 26. convert this number Y into a letter that matches its order in an alphabet starting from 0. For Shift Cipher decryption: For every letter in ciphertext, we convert the letter into the number that matches its order in the

alphabet starting from 0, and call this number Y. Now we calculate Y=(X-K) mod 26. Convert this number X into a letter that matches its order in the alphabet starting from 0.

II. EXISTING SYSTEM

The existing system uses only Merkle-Hellman Knapsack Cryptosystem and CBC Mode to encrypt the messages and decrypt them in reverse order. In Figure 1, The existing system shows the encryption process with two techniques, CBC and Merkle-Hellman and in Figure 2, the decryption process is shown. Through the encrypted message is difficult to break, it can be found out if we get to know the pattern of the message being sent.

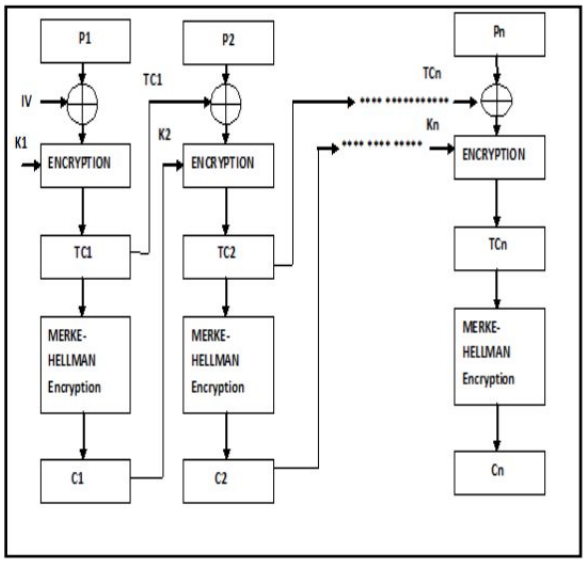


Figure 1: Block diagram of the encryption process

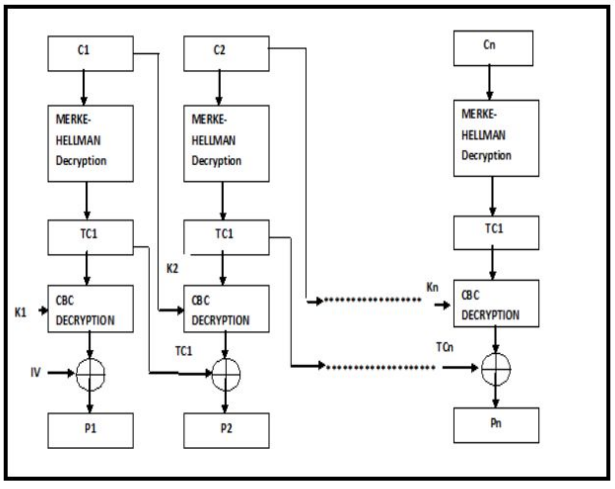


Figure 2: Block diagram of Decryption Process

III. PROPOSED SYSTEM

Cipher Block Chaining (CBC) is one of the ways Block Ciphers gets n-blocks of plain text and encrypts all blocks simultaneously while the Merkle-Hellman algorithm developed in 1978 is based on the problem of a superincreasing subset. In cryptography, a Caesar cipher, also known as Caesar's cipher, the shift cipher, Caesar's code or Caesar shift, is one of the simplest and most widely known encryption techniques. This is a type of substitution cipher, where each letter is replaced in plain text for a certain number of positions in the alphabet. This paper shows how to use a combination of three cryptographic algorithms to encrypt messages so that only the recipient can decrypt the message.

IV. ENCRYPTING MESSAGES

The 3 levels of encryption are done the proposed cryptosystem. Firstly shift cipher is performed. The shift cipher works by using the modular operation to encrypt the message. The shift has a key K, which is an integer from 0 to 25. Firstly CBC mode of encryption is performed on it by breaking the plain text into individual characters, then converted to binary equivalent. Each binary equivalent of the character represents a block. CBC encryption is done by XORing the block with the initialization vector and key. Each block is encrypted to form the temporary ciphertext. Secondly, the temporary ciphertext is encrypted through Merkle-Hellman encryption by choosing a superincreasing sequence of positive numbers. A superincreasing sequence is one where every number is greater than the sum of the preceding number. The two secret number is chosen, one with is greater than the sum of all the numbers in the sequence and its coprime as the other one. After all the blocks of code have been they are combined together to get the cipher to form the ciphertext.

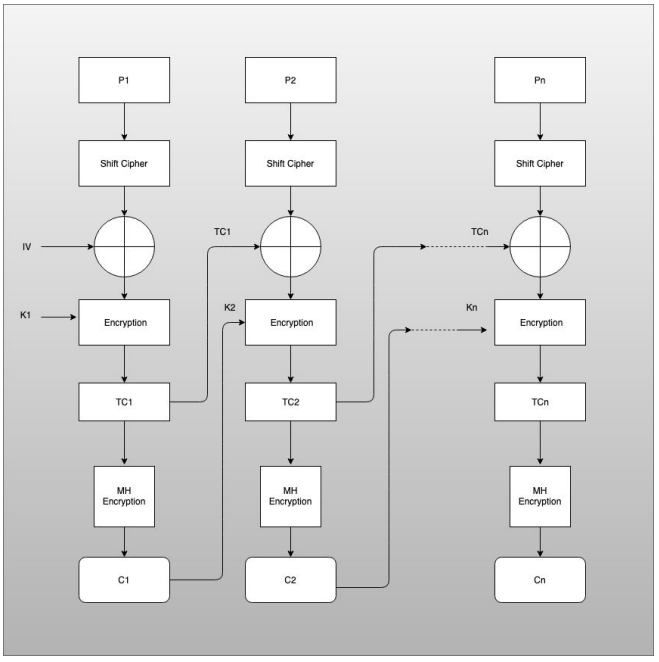


Figure 1: Block diagram of the encryption process with a shift cipher technique

A. Mathematical Explanation

Choose Key to perform Caesar cipher. Then choose IV and key of 7 bits. To perform encryption in CBC mode. After performing the CBC mode of encryption, the result is feed to perform Merkle-Hellman.

Steps to perform Merle-Hellman encryption,

- Choose the superincreasing sequence where every number is greater than the sum of all preceding number

  $w = (w_1, w_2, ..., w_n)$

- Choose the secret number "q"

$$q > \sum_{i=1}^{n} w_i,$$

and a random integer, r, such that gcd(r,q) = 1 (i.e. r and q are coprime).

q is chosen this way to ensure the uniqueness of the ciphertext. If it is any smaller, more than one plaintext may encrypt to the same ciphertext. Since q is larger than the sum of every subset of w, no sums are congruent mod q and therefore none of the private key's sums will be equal. r must be coprime to q or else it will not have an inverse mod q. The existence of the inverse of r is necessary so that decryption is possible.

Now calculate the sequence

$\beta = (\beta_1, \beta_2, ..., \beta_n)$

where

$\beta_i = rw_i \bmod q$.

The public key is β, while the private key is (w, q, r).

Encryption

To encrypt an n-bit message

$\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)$,
where α_i is the i-th bit of the message and {0, 1}, calculate

$$c = \sum_{i=1}^{n} \alpha_i \beta_i.$$

The ciphertext then is c.

Example :

Encrypting the string "hi"

Step 1: Performing a Caesar cipher technique

Choose a random key Kc=19

h i

7 8

\+ 19 19

(26 27) mod 26

0 1

a b

the caesar cipher for "hi" is ab

Step 2: Encryption for the first block(letter) 'a'

Perform **CBC** encryption mode

Generate the random IV and Key K1

let, IV = 0 1 0 0 1 0 0

K = 1 0 0 0 0 0 1

the characters "ab" are converted to binary format

a = 1100001

b = 1100000

binary characters of each and every block are considered as a separate block

For the first block,

The binary value of 'a' is XORed with IV and then the result is again XORed with the key K.

p1, p2,…., pn represents the blocks of plain text. Each block of plain text is then encrypted by using a vector and a key, thereby producing a temporary code (TC). TC1,TC2,… TCn represents the temporary codes for the plain text blocks p1,p2,..pn respectively.

Thus,

TC1= ((1100001) XOR (0100100))XOR (1000011)

= 0000100

This completes the CBC stage of 1st block.

Perform **Merkle-Hellman scheme** on the temporary output of 1st block.

The first step is to choose a superincreasing sequence. In this case the sequence is s = (3, 5, 15, 25, 54, 110, 225)

The two secret numbers chosen are a = 439, r = 10.

The sequence vector is b = b1, b2,… bn

Where bi = r * si mod a.

The message is encrypted by multiplying all the elements of sequence b with the corresponding elements of sequence s and adding the resulting sum.

Therefore, the encrypted message

$C = \sum^{n}_{i=0} b_i * (TCj)_i$

Where, j represents the jth block

b1 = 3 * 10 mod 439 = 30

b2 = 5 * 10 mod 439 = 50

b3 = 15 * 10 mod 439 = 150

b4 = 25 * 10 mod 439 = 250

b5 = 54 * 10 mod 439 = 101

b6 = 110 * 10 mod 439 = 222

b7 = 225 * 10 mod 439 = 55

Encrypting the character 'a'

b = (30, 50, 150, 250, 101, 222, 55) and

TC1= (0000110)

C1 = 101+222 = **323**

Step 3: Encryption for the second block(letter) 'b'

Here the temporary encrypted value of the previous block 'b' is taken as the Initial vector IV,

from previous step and perform XOR operation with the input block.

The previous temporary encrypted value was

TC1= 0000100.

The key is obtained by taking the final output of the 1st block and then calculating the remainder value by dividing it by 128.

i.e. k2= binary equivalent of (C1 mod 128)

= binary equivalent of (323 mod 128)

= binary equivalent of (67) = 1000011

Now,

TC2=P1 XOR TC1 XOR C1

=1100010 XOR 0000110 XOR 1000011

TC2= ((0000100) XOR (1000011)) XOR (1100010)

= 0100111

b = (30, 50, 150, 250, 101, 222, 55) and

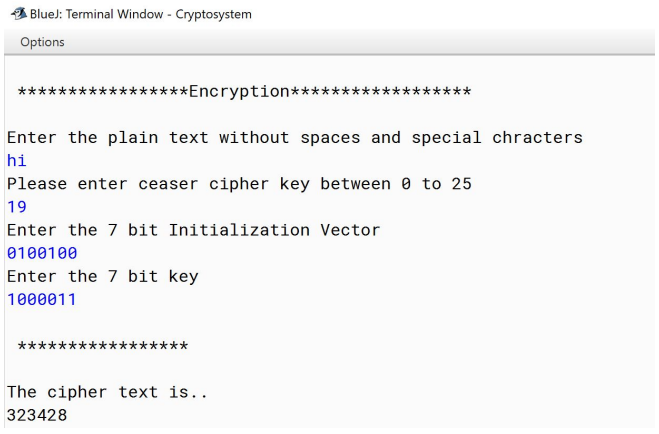
TC2= (0100111)

Therefore, C2 = 50 + 101 + 222+55 = **428**.

Therefore, the actual encrypted message value is

C= 323428.

B. Experimental Results for encryption



V. DECRYPTING MESSAGES

The received message is decrypted by separating the ciphertext into groups of 3 digits from the starting position. Each group is considered as the separate plain text. The decryption is performed in 3 steps on each group. Each block is decrypted by Merkle- Helman, then with CBC mode of decryption by XORing the temporary ciphertext with key and initialization vector choose in the encryption. Then shift cipher is performed to decrypt the original message.

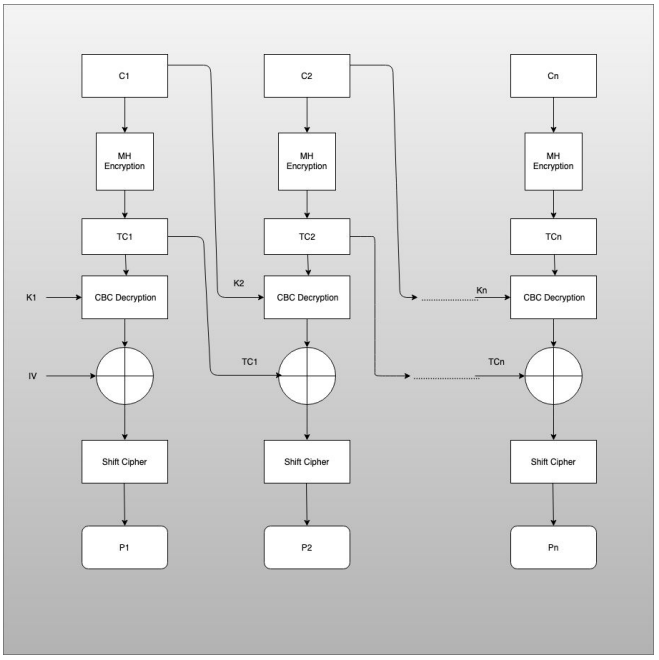


Figure 2: Block diagram of the Decryption Process with shift cipher technique

A. Mathematical Explanation

Decryption

Perform the Merkle-Hellman scheme by separating the ciphertext into groups of 3 digits from the starting position. Then perform CBC encryption mode on it by XOR in with key and IV. Finally, perform Caesar cipher on the outcome of the previous step.

Steps to perform Merle-Hellman decryption,

In order to decrypt a ciphertext *c,* a receiver has to find the message bits α_i such that they satisfy

$$c = \sum_{i=1}^{n} \alpha_i \beta_i.$$

This would be a hard problem if the β_i were random values because the receiver would have to solve an instance of the subset sum problem, which is known to be NP-hard. However, the values β_i were chosen such that decryption is easy if the private key (w, q, r) is known.

The key to decryption is to find an integer s that is the modular inverse of r modulo q. That means s satisfies the equation $s\,r \bmod q = 1$ or equivalently there exist an integer k such that $sr = kq + 1$. Since r was chosen such that $\gcd(r,q)=1$ it is possible to find s and k by using the Extended Euclidean algorithm. Next, the receiver of the ciphertext c computes

$$c' \equiv cs \pmod{q}.$$

Hence

$$c' \equiv cs \equiv \sum_{i=1}^{n} \alpha_i \beta_i s \pmod{q}.$$

Because of rs mod q = 1 and $\beta_i = rw_i \bmod q$ follows

$$\beta_i s \equiv w_i rs \equiv w_i \pmod{q}.$$

Hence

$$c' \equiv \sum_{i=1}^{n} \alpha_i w_i \pmod{q}.$$

The sum of all values w_i is smaller than q and hence $\sum_{i=1}^{n} \alpha_i w_i$ is also in the interval [0,q-1]. Thus the receiver has to solve the subset sum problem

$$c' = \sum_{i=1}^{n} \alpha_i w_i.$$

This problem is easy because w is a superincreasing sequence. Take the largest element in w, say w_k. If $w_k > c'$, then $\alpha_k = 0$, if $w_k \leq c'$, then $\alpha_k = 1$. Then, subtract $w_k \times \alpha_k$ from c', and repeat these steps until you have figured out α.

Example:

Decrypting the message: **C = 323428**

Step 1: Performing decryption on the first block

Separate the ciphertext into groups of 3 digits from the starting position.

Therefore, C1 = 323 C2 = 428

The encrypted message C1 is 323 and

The two secret numbers chosen are a = 439, r = 10.

In this case the sequence is s = (3, 5, 15, 25, 54, 110, 225)

The modular inverse of 10 in 10 mod 439 is calculated by using the extended Euclidean algorithms and which was found to be 44

So, 323 * 44 mod 439 = 164

The largest number in the sequence s, which is smaller than 28 is 25.

164 - 110 = 54

54 – 54 = 0

Thus, the binary sequence becomes TC1 = 0 0 0 0 1 1 0.

After performing Merkle-Hellman decryption we apply CBC on the first block on the first block. The ciphertext is XORed with the key and then with the IV.

P1= TC1 XOR K1 XOR IV

=(0 0 0 0 1 1 0 XOR 0 1 0 0 1 0 0) XOR 1 0 0 0 0 1 1

= 1100001

the character equivalent of this binary value is ‘a’

the perform caesar cipher for Key Kc=19

a

0

- 19

-19+26= 7 which is equivalent to alphabet **‘h’**

Step 2: Performing decryption on the second block

The encrypted message C2 is 428

The two secret numbers chosen are a = 439, r = 10.

In this case the sequence is s = (3, 5, 15, 25, 54, 110, 225)

The modular inverse of 10 in 10 mod 439 is calculated by using the extended Euclidean algorithms and which was found to be 44

So, 428 * 44 mod 439 = 394

The largest number in the sequence s, which is smaller than 194 is 110

394 - 225 =169

169 - 110 = 59

59 - 54 = 5

5 - 5 =0

Thus, the binary sequence becomes TC2 = 0 1 0 0 1 1 1.

After performing Merkle-Hellman decryption we apply CBC on the first block on the first block. The ciphertext is XORed with the C1 and then with the P1.

C2 = Binary equivalent of 428 = 110101100

P1 = (C2 XOR TC2) XOR C1

=(110101100 XOR 0100111) XOR 101000011

= 1010000

the character equivalent of this binary value is ‘b’

the perform caesar cipher for Key Kc=65

b

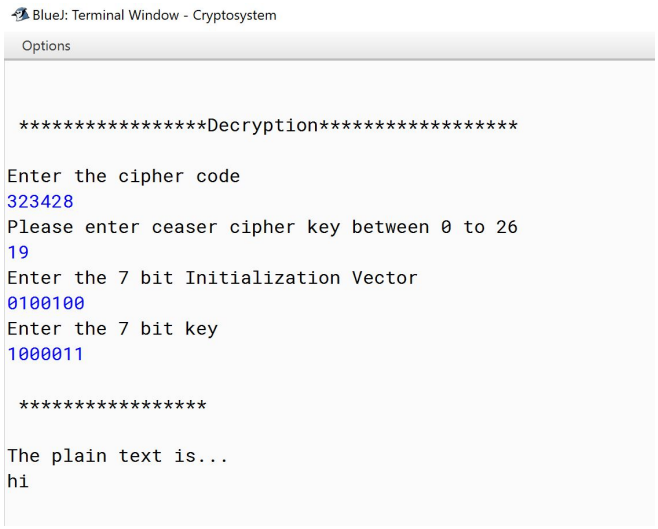
1

- 19

-18+26 = 8 which is equivalent to the alphabet **‘i’**

Therefore, the decrypted message is ‘hi’

C. *Experimental Results for Decryption*



VII. CONCLUSION

To enhance the confidentiality of the current existing system and to get rid of the loopholes, we propose a system where we use the combination of Shift cipher with CBC mode of encryption and Merkle-Hellman Knapsack cryptosystem to encrypt the message so that only the intended recipient or receiver can decipher the messages sent by the sender.

FUTURE SCOPE

The future scopes include implementation of encryption and decryption of the plaintext that contains spaces and special characters and sentences.This can be achieved by extensive use of shifting and hashing algorithms. The encryption algorithm can be strengthened by combining it with other encryption schemes also.

REFERENCES

[1] A.Menezes, P.vanOorschot and S.Vanstone, “Handbook of Applied Cryptography”, CRC Press, 1996

[2] Ashish Agarwal, “Encrypting Messages using the Merkle Hellman Knapsack Cryptosystem”

[3] W.Diffie and M.Hellman, “New directions in cryptography”, IEEE Transactions on Information Theory – 22, 6, pp 644 – 654.

[4] R.Merkle and M.Hellman, “Hiding information and signatures in trapdoor knapsacks”, IEEE Transactions on Information Theory – 24, 5, pp 525 – 530.

[5]William Stallings,” Cryptography and Network Security”, fifth edition,pg225-227.

[6]http://mathworld.wolfram.com/SubsetSumProblem.html

[7]http://www.mast.queensu.ca/~math418/m418oh/m418oh04.pdf