

University of San Carlos

School of Arts and Sciences

Department of Computer, Information Sciences, and Mathematics

Talamban Campus, Cebu City, Philippines

IT 3104N – Information Assurance and Security

**PROJECT: CRYPTOGRAPHY**

**Submitted by:**

Van AJ Vanguardia

**Submitted to:**

Godwin Monserate

**30 November 2022**

**Overview**

The project is a combination between a modified RSA (Rivest-Shamir-Adleman) and the Atbash cipher. RSA proposed a method for implementing a public-key cryptosystem whose security rests in part on the difficulty of factoring large numbers. RSA was modified to use four keys of prime numbers (p, q, r, s) instead of two (p, q), which improves security due to its added complexity.

The Atbash Cipher is a simple substitution cipher that is sometimes called mirror code and it was modified by taking the extra step of reversing the result string after the reversal or mirroring of an alphabet. This project intents to provide a more complex cipher algorithm owing to large factorials and mappings in order to attain more secured encryption and decryption process. Furthermore, the methods and modifications used in the project are broken down into four sections: key generation, encryption, decryption, and reverse process.

**Key Generation**

1. Choose four prime numbers where P, Q, R, S are distinct
2. Calculate N (Product of the 4 prime numbers)

* Let N = p \* q \* r \* s

1. Calculate T (Totient of Product of Primes)

* Let T = (p -1) \* (q - 1) \* (r - 1) \* (s - 1)

1. Select e (encryption) such that:

* 1 < e < T
* Coprime of N and T or gcd(T, e) = 1;

1. Select d (decryption) such that:

* d = de(mod T) = 1

1. Generate public key

* Let KU = {e, N}

1. Generate private Key

* Let KR = {d, N}

**Encryption Process**

1. Modified Rivest-Shamir-Adleman – Encryption

Formula: C = M ^ e mod N

Where:

C = Ciphertext

M = Message

e = Encryption

N = Product of the 4 prime numbers

1. Modified Atbash

(1) First Phase

Formula: N – L

Where:

N = Max length of the alphabet (25, starting with 0)

L = Index of the alphabet (0 – 25)

Example:

25 – 0 (A) = 25 (Z)

25 – 25 (Z) = 0 (A)

(2) Second Phase

Formula: REV(S)

S = String of atbash characters

REV = String reverse function

Example:

Khoor -> rookH

**Decryption Process**

1. Modified Atbash

(1) First Phase: Normal Atbash

Formula: N – L

Where:

N = Max length of the alphabet (25, starting with 0)

L = Index of the alphabet (0 – 25)

Example:

25 – 0 (A) = 25 (Z)

25 – 25 (Z) = 0 (A)

(2) Second Phase: Reversal of Atbash

Formula: REV(S)

S = String of atbash characters

REV = String reverse function

Example:

elloH -> Hello

1. Modified Rivest-Shamir-Adleman Cryptography – Decryption

Formula: M = C ^ d mod n

Where:

M = Message

C = Ciphertext

d = Decryption

N = Product of the 4 prime numbers

**Reverse Process**

Swap the positions of the string element using loop and iterate until i is equals to j.

Formula:

temp = text[i]

text[i] = text[j]

text[j] = temp

Where:

i = 0; starting index of a string

j = len(string); the length of a string

temp = temporary storage of a character

text = array of characters