VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
FUNDAMENTINIŲ MOKSLŲ FAKULTETAS

Cryptography and Information Security

Module: FMISB18500

Name Surname: Abdul Hannan Ayubi

Student Number : 20210290

Date: 05.10.2021

Lect. Paulius Narkevičius

Table of Contents

Diffie-Hellman Public-Key Algorithm

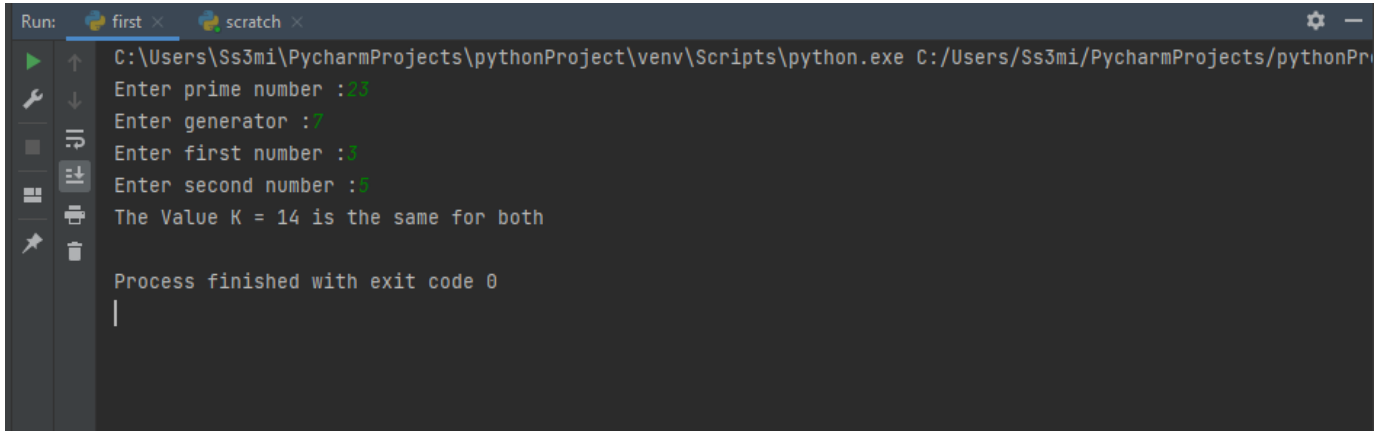
My Code:

```
def Calculate(firstNumber, secondNumber, generatorNumber, primeNumber):
    resultOne = pow(generatorNumber, firstNumber) % primeNumber
    resultTwo = pow(generatorNumber, secondNumber) % primeNumber

    finalResultOne = pow(resultOne, secondNumber) % primeNumber
    finalResultTwo = pow(resultTwo, firstNumber) % primeNumber

    return finalResultOne, finalResultTwo
```

Input/Output



I write a python application for this task. Firstly, I get the prime number, Generator and Two number from user. The algorithm calculates as below :

Example 1:

Prime number = 23,
Generator = 7, a1 = 3, a2 =5
7^3 mod 23 = 343 mod 23 = 21 (first value)
7^5 mod 23 = 16807 mod 23 = 17 (Second Value)
21^5 mod 23 = 4084101 mod 23 = 14 (Key)
17^3 mod 23 = 4913 mod 23 = 14 (Key)

Example 2:

prime number = 17
Generator = 3, a1 = 7, a2= 11
3^7 mod 17 = 2187 Mod 17 = 11
3^11 mod 17 = 177147 Mod 17 = 7
11^11 mod 17 = 285311670611 Mod 17 = 12
7^7 mod 17 = 823543 Mod 17 = 12

As a conclusion we calculate with this way that Bob and Alice can get the same shared key in this situation and my calculate function doing the same thing with different set of numbers.

KRSA Algorithm

My Code:

```
def Calculate(a, b, a1, b1):
    m = (a * b) - 1;
    e = (a1 * m) + a;
    d = (b1 * m) + b;
    n = ((e * d) - 1) / m;
    return e, d, n
```

Input/Output:

```
C:\Users\Ss3mi\PycharmProjects\pythonProject\venv
Enter a:9
Enter b: 11
Enter a1: 5
Enter b1: 8
Enter P: 512
P-Value: 512 Decrypt : 512.0
Result: 512.0

Process finished with exit code 0
```

I write a python function for this the function calculate the e,d,n values after getting the values from the user. After Getting a, b, a1, b1. The algorithm works as below:

Example 1:

A = 9, b = 11, a1 = 5, b1 = 8, P1 = 512

M = (9 * 11) – 1 = 98

E = (5 * 98) + 9 = 499

D = (8 * 98) + 11 = 795

N = ((499 * 795) – 1) / 98 = 4048

Example 2:

A = 5, b = 7, a1 = 9,b1 = 11, p = 538

M = (5 * 7) – 1 = 34

E = (9 * 34) + 5 = 311

D = (11 * 34) + 7 = 381

N = ((311* 381) – 1) / 34 = 3485

Encrypt = 512 * 499 Mod 4048 = 255488 Mod 4048 = 464 | 538 * 311 Mod 3485 = 38

Decrypt = 464 * 795 Mod 4048 = 368880 Mod 4048 = 512 | 38 * 381 Mod 3485 = 538

Conclusion: Here the Decrypted value is the same with the given P value, So the Encryption done successfully. Decrypting the message is depending on the value D and no one except the b1(who have the key) user can decrypt it.

RSA Algorithm

Input / Output

```
C:\Users\Ss3mi\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/Ss3mi/AppData/Roaming/JetBrains/PyCharmCE2021.2/scratches/scratch.py
Enter p :7
Enter q : 13
Enter e : 5
d :  29
Public key (5, 91)
Private key (29, 91)
Enter plane text : algorithm
The numeric values of alphabet
[1, 12, 7, 15, 18, 9, 20, 8, 13]
Encrypted Data
[1, 38, 63, 71, 44, 81, 76, 8, 13]
Decrypted Data
[1, 12, 7, 15, 18, 9, 20, 8, 13]

Process finished with exit code 0
```

I write a python function for doing RSA Algorithm. The RSA Algorithm Works as below:

P = 17, q = 23, e = 5
N = 17 * 23 = 391
Ph = (17 – 1) * (23 – 1) = 352
e= 1 < e < 352 and GCD(e,352) = 1
E = 1,3,5 we can pick one value from these and I get E = 5.
I = [1,2,3,4,5,6,7,8,9,10] a set of numbers

We can find the D from here: I * Ph / E = 2 * 352 / 5 = 141.0
Getting plane Text: vilnius
Alphabet of Plane Text: [22,9,12,14,9,21,19]
Encrypt = 22^5 Mod 391 = 252
9^5 Mod 391 = 8
12^5 Mod 391 = 156

14^5 Mod 391 = 199
9^5 Mod 391 = 8
21^5 Mod 391 = 106
19^5 Mod 391 = 287

Encrypt = [252,8,156,199,8,106,287]

Decrypt = 252 ^141 Mod 391 = 22
8 ^141 Mod 391 = 9
156 ^141 Mod 391 =12
199 ^141 Mod 391 = 14
8^141 Mod 391 = 9
106 ^141 Mod 391 = 21
287 ^141 Mod 391 = 19
Decrypt = [22,9,12,14,9,21,19]

Conclusion: As a conclusion of this algorithm, we create a mathematical expression based on the input of the user and after calculating the E, and D values based on the given formula we change the plane text to the alphabetics order. For example, A = 1, b = 2, c = 3, And after changing the whole plane text we Encrypt the numbers by using Enc = (numbers of plane text) ^(E) Mod (N). Decryption process is the same with the encryption formula we have done with Dec = (Numbers which Encrypted) ^ (D) Mod (N).