

**RSA Algorithm Analysis**

|  |  |
| --- | --- |
| **Name** | **Code** |
| **Abdelaziz Salah Mohammed** | 173041 |

**Delivered to:**

**DR\_ Samir Shahin**

**Eng. Khaled Moataz**

**Date: 13/4/2023**

**Sender Explanation:**

This is an RSA sender program. It takes a message and a public key

    and encrypts the message using the public key. The encrypted message

    is then sent to the receiver.

    @Algorithm is as follows:

    1. we define a map, which converts the 36 available characters to a number.

    as follows

        [0 -> 0, 1 -> 1 ... 9 -> 9, a -> 10, b -> 11, ... z -> 35]

    2. if we find any different character, we deal with it simply as space.

    3. Read the message from the user.

    4. before implementing the RSA, we must code the input into numbers.

    5. our scheme is as follows:

        - Convert any extra characters to spaces as specified above.

        - Group the plaintext into sets of five characters per group

        - If the last grouping does not have exactly five characters,then append some space to the end of the plaintext message to fill out the last grouping

        - Convert each group into a separate number as follows

            - group one is [c4,c3,c2,c1,c0]

            - then the number = c4\*37^4 + c3\*37^3 + c2\*37^2 + c1\*37^1 + c0\*37^0

        - A decoding operation should be performed in the same way except that the process is reversed.

          A number is converted to its character grouping by using mod and div

**Receiver Explanation:**

 '''

       This file act as a receiver server

It Accept the message coming from the sender

Then it decrypt it using its private key

After that it decode the message

This is done by

1. Loop for i from 0 to 5
2. Get the number mod 37
3. Then apply integer division by 37
4. Then append the character to the list in which you cascade letters in.

Then we show the message.

    '''

**Attack Explanation:**

 here we want to apply the attack

        so the algorithm will be as follows:

            1- read the public key of the server

            2- apply prime factorization on n

            3- calculate phi(n)

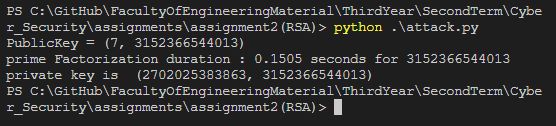
            4- calculate d

            5- decrypt the message

**Attack notes:**

* As the number of bits used in generating the pair of keys increases, the time taken to be able to apply the attack will increase.
* The increase in the time increases exponentially.
* I have computed the time taken for apply the attack for some values of n, and here are the results:

N = 21:



N = 22:



N = 23:



N = 24:



N = 25: Text

Description automatically generated

N = 26:

Text

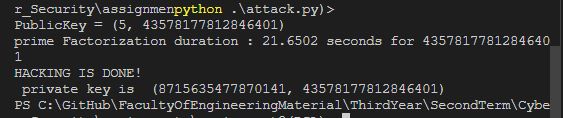
Description automatically generated

N = 27:

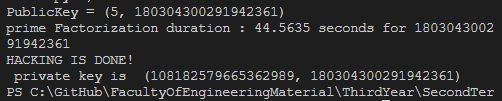
Text

Description automatically generated

N = 28:



N = 29:



N = 30:

Text

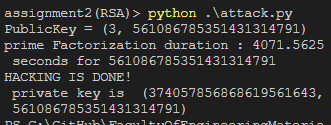
Description automatically generated

N = 32:

Text

Description automatically generated

N = 35:



Chart, line chart

Description automatically generatedTime vs number of bits: