Central Washington University

Advanced Algorithms

Winter 2019

Project 4 Report

Author:
Hermann Yepdjio

Professor:
Dr. Razvan Andonie

February 28, 2019

Contents

1 Introduction

The purpose of this project was to experiment using the RSA public-key cryptosystem to encrypt and decrypt messages and Pollard-Rho's algorithm to break encryption codes. We experimented using different large prime numbers for p and q. The details and results of the experimentation are discussed below.

2 experimentation process

2.1 Self Experimentation

we experimented using primes of k bits (k = 30, 35, 40,...,55). The program uses Miller-Rabin's algorithm to check if the numbers are primes. we stopped at 55 because the pollard-rho implementation would take forever to factor the modulus if we were going beyond that. The experimentation starts with the user inputting a message to be encrypted. Then, for each value of k, we do the following:

- generate both a private-key and a public-key following the RSA public-key cryptosystem algorithm from the book
- using the public-key, we encrypt the message and record how long it takes,
- using the the private-key, we decrypt the message and record how long it takes ,
- using the public-key, we try to decrypt the message and record how long it takes,

2.2 Experimenting with a Classmate

For this part of the experiment, I teamed with Brian. I was able to break his code in about 7 seconds and he was also able to break mine.

3 Results

The results of the experimentation are contained in the table below

# bits for p and q	Time to encrypt (seconds)	Time to decrypt using private key (seconds)	Time to decrypt using public key(seconds)
30	0.00699996948242	0.0	0.0869998931885
35	0.0700001716614	0.0	2.71399998665
40	0.156999826431	0.0	26.3010001183
45	0.25200009346	0.0	81.8629999161
50	1.05800008774	0.000999927520752	860.194000006
55	0.885999917984	0.00100016593933	$>$ 1 hour

As we can see from the table above,

- the time spent for encrypting a message increases very slowly as the number of bits for p and q increases
- the time spent for decrypting a cipher-text using the private key is really small and increases extremely slowly as the number id bits for p and q increases
- the time spent for decrypting a cipher-text using the public key is big and increases really fast as the number of bits for p and q increases

4 Conclusion

From the experimentation above, we observed that increasing the values of p and q makes the encryption code harder to break. Since both encrypting and decrypting using the private key are fast, it makes therefore more sense to use very large values for p and q to produce the keys as it only really affect the people who might try to break the encryption code.