German University in Cairo
Faculty of MET (CSEN 1001 Computer and Network Security Course)
Dr. Amr El Mougy

Hacking Assignment 2 : Break Simple RSA Schemes

Discussion 03/03/2018 - 08/03/2018 **Task Grade** 5%

1 Introduction

In this task you're required to implement an algorithm to break Simple RSA Schemes. This can be done by first getting the prime factors of n using the Sieve of Eratosthenes algorithm discussed in class. Once you have obtained the two prime factors p & q, you can then use the extended Euclidean Algorithm to find the private key d using $\Phi(n)$ and the public key e.

After finding the private key, you can simply follow the decryption equation to get the decrypted value;

$$\mathbf{M} = \mathbf{C}^{\mathbf{d}} \bmod \mathbf{n}$$

2 Details

You are given a zip folder containing the base implementation of the task.

You are required to implement an algorithm that given the values; n, e and E encrypted using RSA, returns the both the private key d and the decrypted message. The name of your class should be "OldRSACipherBreaker".

To compute the prime factors of n, you will apply the Sieve of Eratosthenes algorithm to get all the prime numbers till n. Once you get all the prime numbers till n, you can start to check whether n is divisible by any two of the prime numbers. Once you find p & q; two prime factors of n, you can calculate Euler's Totient, $\Phi(n)$.

Once you acquire $\Phi(n)$, you can use it alongside e to calculate the private key d. This can be done by using the Extended Euclidean Algorithm, where gcd(e,$\Phi(n)$) = [1, d, k]. You can read more on the Extended Euclidean Algorithm here:
`https://brilliant.org/wiki/extended-euclidean-algorithm/`

Once you have found the private key d, you can use it along with n to decrypt the provided cipher by following the equation:

$$\mathbf{M} = \mathbf{C}^{\mathbf{d}} \bmod \mathbf{n}$$

You are also provided with a set of unit test cases in order to check your implementation once you are done. The "TestOldRSABreaker"tests getting the both the correct d and the correct decrypted message.

You will find the source code in the following link; `https://goo.gl/dXjxFs`.

3 Submission

You will be required to submit your project by maximum one week from the tutorial slot (e.g. if your tutorial slot is on Sunday, your deadline is the following Saturday at 23:59) . Export the project in a zip format with the unit tests included and upload it to the MET website in the corespondent submission link for your tutorial group. The zip folder should be named as $[ID]\_[TutoiralNumber]\_[Task\_Number]$ (e.g. $[31-1111]\_[T01]\_[Task2]$).

In case there was a problem in the submission through the MET website, then send an email to the following gmail account csen1001.17@gmail.com with the title same as the name of the zip folder.