Project 2a - RSA decryption

Due July 26, 2017 at 5pm

1 Background

Your last project was on factoring numbers, with reference to the RSA encryption scheme. This extra part will test how efficient your factoring was (or if you factored at all, instead of finding patterns).

2 Project

For this project you will be factoring numbers, again. However this time there are only 4 numbers you need to factor they are provided separately as files `public-`k`.txt`, where $k = 1, 2, 3, 4$. In this file are two numbers, they are RSA public keys, the first number is larger this is n, the number you need to factor, the other is the encryption exponent of e, you'll need to pass this somewhere, but you do NOT need to factor it. Factoring this number will in no way help you factor n.

Additionally there are 4 message files `message-`k`.txt`, where $k = 1, 2, 3, 4$. These are the encrypted messages. Your goal is to decrypt them, via factoring n in the associated key files.

3 Expectations

None. There are no *requirements* for this part, it is completely optional. If you are able to factor some of these numbers (and thus decrypt the messages) you will receive hints towards the final, and some extra credit towards project 2.

You may use any method of factoring you'd like, as long as it is written by you. You may adapt algorithms from Project 2, or that you've found online, but ultimately the code you use must be written by you.

You should be able to factor the first number, and decipher the first message. These are small numbers, similar to ones found in Project 2. There is no extra credit for factoring this number, but can be used to help make sure you are using my provided code correctly.

In addition to the messages and public keys I am providing some Python code to decrypt the messages. If you are writing your code in Python, then you can simply copy my code into your file (or vice versa) and use it with your factoring algorithms.

If you are writing in another language, that is perfectly fine. All you'll need to do is type in the factorizations into the given Python code to do the decryptions.

If you do not know Python or can't get my code to work, talk to me quickly! This is not supposed to be the hard part of the project, so I will absolutely help you to get the messages decrypted if you factor the numbers.

4 Deliverables

For this you must submit a tarball of text files for each decrypted message, as well as the factorization. The files should be called `solution-k.txt` where $k = 1, 2, 3, 4$ corresponding to the message and public key. It should contain the plaintext English sentence(s) that the Python code spit out, as well as the factorization of n.

Additionally your tarball must contain all of the code used to factor the numbers, so I can see/read/evaluate it. You should also have a README file that describes changes you made from Project 2. No Makefile is necessary.

5 Restrictions

There are numerous libraries that have already implemented algorithms for use. You should not use these in your final turn in. All code that you submit should be yours. If your project just calls someonelse's library for factoring numbers you will not do well on this project.

Similarly for primes, there are numerous lists of primes online, it might be helpful to use them to practice factoring numbers but your final turn in should not have this. That is if you need a list of primes for your project you should generate that list yourself.

Absolutely nothing should be hard-coded. Once you factor some of the numbers you should not store the result so that future runs are faster. Though you may take advantage of any patterns you find.

6 Grading

Grading will be very easy for me, either you made a submission or not. If you submitted, you either submitted correctly factored numbers and successfully decrypted plaintext or you submitted nonsense. Each correct file (besides the first one) will give you some extra credit on Project 2 as well as a hint for the final.

7 Final thoughts

Absolutely nothing about this project is required, work on this if you like/are interested, but do NOT sacrifice studying time for the final. While the hints may help you on the final, they will not make up for not knowing the material.