# CYBR 434 Network Security

Lab 3, RSA - Due, 10/13

Fall 2025

## Overview

The objective of this lab is to learn more about RSA. You'll finish writing a basic RSA python encryption program (given a mostly complete file) and learn how the key size affects the effectiveness of RSA.

**Part 1:**

Finish implementing the code in `practiceRSA.py` and make sure you can encrypt and decrypt data. Encrypt a message with **both** your public and private keys. Code for encrypting with a public key is given, **do not forget to provide code for encryption with your private key**. Include a screen shot of the output in your write up.

**Part 2:**

Given my `public key`(n,e), decrypt the cipher text $C$ (represented as an integer) created with my public key :

```
C = 131557800652976364014135900992901925579 5
public key = (303995681913192380046930326078271705385 3,65537)
```

Make a separate python file to calculate your solution to part 2. Explain what you did and how you cracked the cipher text using the public key. Be sure to state the decrypted text in your write up.

**Part 3:**

Create keys of increasing size and plot the time required to crack them. Start with the number of bits for each prime number at ~64 and include at least 10 steps, increasing the number of bits each step. Determine how large the key needs to be for cracking to be too difficult for your computer. Include the plot in your PDF. **Be sure to crack 3-5 keys per size interval and use the average of time taken to crack them for your data points.** You may find the `sympy` library helpful.

**Extra credit (15)**

Crack:

```
cipher text =  384064655929475133376253368712717288470700006955283702011833979565434672114882524865492201938 4
public key = (107548704267580420894374726952509122827468523354173751930242826450807529858788238147657648933 07,65537)
```

You must write the cracking code for this and I must be able to run it.

## Turn in

You'll create a tarfile to turn in your lab. Make sure to include:

1. A PDF write up answering any questions/providing conclusions (in red), requested screen shots, and plots (in orange).
2. Source code. Your python files **must** be named `lab3-part1.py`, `lab3-part2.py`, and `lab3-part3.py` and your code must run.
3. A **detailed** `README` describing your code. The `README` **must** describe how to run each python file and what they do.