sphinx.

## What is it?

Sphinx stands for “a password Store that Perfectly Hides from Itself (No eXaggeration!)”. It is a novel implementation of a password store which is secure even when the password manager itself has been compromised!

Something Awesome

## How does it work?

Sphinx relies on device-enhanced password authenticated key exchange, by storing part of the password on an external device, such that an attacker breaking into this device learns no information about the master password or the user’s individual passwords.

## Result

The original sphinx paper works with an android phone as the Device. The authors proposed a possible extension for an online instantiation of sphinx rather than a smartphone-based instantiation. I thought that was *something awesome*, so here is my implementation of the password manager + encryption.

### Python implementation of core mathematics

Interactive Python Jupyter Notebook containing my implementation of core algorithms.

* Octet String to Integer Primitives: convert between a positive integer and a byte array
* HashToBase function: hash a byte array into an element of a prime group
* Simple SWU Hashing algorithm: Hashes a byte array into an elliptic curve
* Oblivious Pseudo-Random Function: recreates bytes for server password rwd without knowing the master password itself
* API flask server which provides the “device” backend web services (computes )

### JavaScript front-end

* Mathematically complete implementation of the above functions in JavaScript with sjcl.js for front-end cryptographic capability
* Fully featured React.js front-end chrome extension providing full user experience for MVP.

## Reflection

I am quite happy with what I was able to achieve this semester in COMP6841. I had to learn everything from how elliptic curve cryptography primitives work, programming them in python and JavaScript, to the development of the front to interact with the backend all within the span of 8 weeks.

The initial goal when I set out with this project was to possibly use this solution as my main password manager, and although mathematically the solution is solid and cryptographically secure, I now learn that rolling your own cryptography is unsafe. The learning experience of playing with elliptic curves was worthwhile and improved my understanding of cryptography and how it is implemented in code.

I would like to have spent more time on the project, to add further functionality or regularly post on my blog. I would also like to have proven that sphinx remains cryptographically secure even when the device has been compromised.