asgn5 writeup

Ryan Hui

February 2023

Lessons Learned

Assignment 5 involved dealing with S.S encryption and was the hardest assignment for cse13s so far in my opinion. Being a 2-week assignment, this was to be expected however.

First, for numtheory.c file, we had to use the gmp library and mpz variables since the bits would be too large for regular unsigned ints to handle. Using mpz variables and functions was like learning a new language since it uses a lot of new syntax and rules that I needed to follow. A couple times I forgot to NULL out the variables and ran into seg faults but overall, it wasn't that bad since I started coding it in C first. The numtheory functions logic also weren't too bad. The hardest one was is prime since it was the longest and most complex. Overall, I learned how to use mpz variables and functions, improved on my coding logic, and learned how to run the test numtheory program on numtheory.c

Next, ss.c was the bulk of the assignment where I had to figure out how to actually implement the S.S encryption algorithm. It took a couple of rereadings over the assignment pdf and tutoring sessions for me to understand how what the functions were doing and how to code it. I learned that the functions were similar to their public/private version pairs and that understanding was helpful. Just like in numtheory, the usage of mpz variables in ss.c was also required. This file taught me a lot since I had to work with printing to files as well as improve on my mathematical logic with combining the numtheory functions to solve for the equations used for encryption.

Finally, the last 3 files that I learned a lot from were keygen, encrypt, and decrypt. These files utilized the ss functions in order to either generate the private and public keys, encrypt the message, or decrypt the message. I learned how to open and close files, creating bit sizes for verbose output, and take a username from the user. Learning where and when to use the ss.files was also something I had to learn.

Understanding of Cryptography

After finishing assingment 5, my understanding of cryptography has greatly increased. My understanding of the Schmidt-Samoa algorithm specifically is that it uses a public and private key where messages can only be decrypted if you have the private key. This method of cryptography is believed to be safe since it the algorithm relies on the difficulty of factoring "large composite integers into their constituent primes". Since only the n is revealed, it is hard to figure out the p and q primes used. By having a public and private key with a public and well known encryption method that is pretty much impossible to break, S.S. encryption highlights a secure encryption method that is simple enough to grasp. This information has taught me that I should be using well-studied and proven encryption algorithms instead of making my own. A public algorithm is safer than custom algorithm. Having a public key for encrypting is fine as long as the private key for decryption is safe.

Cryptography affects the world everyday in ways you may not think of. For example, the internet uses cryptography to secure information transfer on websites. This is vital for protecting against data breaches when doing online transactions. In the modern world, the military also uses cryptography to send important messages in order to avoid leaking the info. In conclusion, cryptography is used heavily in everyday life to protect users on the internet and to keep important messages a secret.

I utilize cryptography in everyday life by sending emails which get encrypted or making purchases with my credit card online which is also encrypted info. In addition, when I make withdraw or deposit cash from my bank account, that information also gets encrypted by my bank.