Understanding Homomorphic Encryption
Daniel Duncan, H18B, 5418388, COMP6841

**What was produced?**

1. A report explaining Homomorphic Encryption (HE), targeted at anyone who has done/remembers first year maths.

Adding two messages $m_1$ and $m_2$ can be achieved by multiplying their two ciphertexts,

$$\begin{aligned} Encrypt(m_1) \cdot Encrypt(m_2) = c_1 \cdot c_2 &= (x_1, y_1) \cdot (x_2, y_2) \\ &= (x_1 \cdot x_2, y_1 \cdot y_2) \\ &= (\alpha^{k_1} \mod p \cdot \alpha^{k_2} \mod p, \alpha^{m_1}\beta^{k_1} \mod p \cdot \alpha^{m_2}\beta^{k_2} \mod p) \\ &= (\alpha^{k_1+k_2} \mod p, \alpha^{m_1+m_2} \cdot \beta^{k_1+k_2} \mod p) \\ &= Encrypt(m_1 + m_2). \end{aligned}$$

Hence the product of two ciphertexts is equivalent to the ciphertext of the sum of the respective plaintexts.

*Report Excerpt A*

Homomorphic addition follows relatively simply. Adding two ciphertexts by adding their decryptions gives

$$\begin{aligned} Decrypt(c_1) + Decrypt(c_2) &= \Delta \cdot m_1 + v_1 + \Delta \cdot m_2 + v_2 \\ &= \Delta(m_1 + m_2) + (v_1 + v_2). \end{aligned}$$

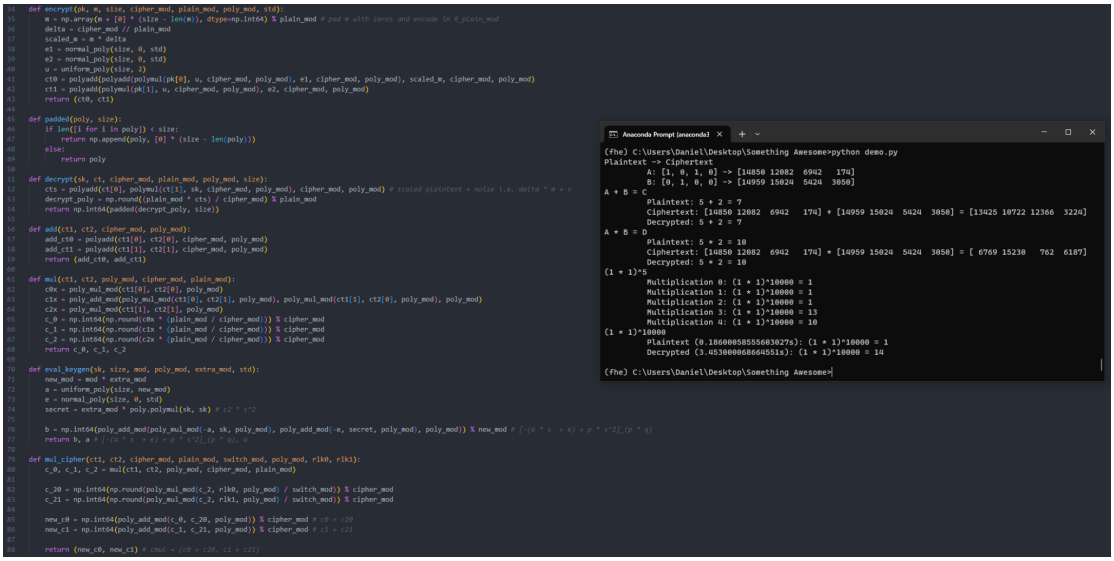
Addition needs to be done $\mod t$, so noting that $m_1 + m_2 = t \cdot \epsilon + [m_1 + m_2]_t$ for some binary polynomial $\epsilon$, and that the remainder of q divided by t is $q - \Delta \cdot t$ i.e. $\Delta \cdot t = q - r_t(q)$,

$$\begin{aligned} \Delta(m_1 + m_2) + (v_1 + v_2) &= \Delta(t \cdot \epsilon + [m_1 + m_2]_t) + (v_1 + v_2) \\ &= \Delta \cdot [m_1 + m_2]_t + \Delta \cdot (t \cdot \epsilon) + (v_1 + v_2) \\ &= \Delta \cdot [m_1 + m_2]_t + (q - r_t(q)) \cdot \epsilon + (v_1 + v_2) \\ &= \Delta \cdot [m_1 + m_2]_t + v_{add} = Decrypt(c_1 + c_2), \end{aligned}$$

where $v_{add}$ :=some small noise. Hence, in this scheme, homomorphic addition is as simple as $m_1 + m_2 = Decrypt(c_1 + c_2)$.

*Report Excerpt B*

2. A working Somewhat Homomorphic Encryption implementation, and a demo to help explain HE to my tutorial/others.



```
(fhe) C:\Users\Daniel\Desktop\Something Awesome>python demo.py
Plaintext -> Ciphertext
        A: [1, 0, 1, 0] -> [14850 12082  6942   174]
        B: [0, 1, 0, 0] -> [14959 15024  5424  3050]
A + B = C
        Plaintext: 5 + 2 = 7
        Ciphertext: [14850 12082  6942   174] + [14959 15024  5424  3050] = [13425 10722 12366  3224]
        Decrypted: 5 + 2 = 7
A * B = D
        Plaintext: 5 * 2 = 10
        Ciphertext: [14850 12082  6942   174] * [14959 15024  5424  3050] = [ 6769 15230   762  6187]
        Decrypted: 5 * 2 = 10
(1 + 1)^5
        Multiplication 0: (1 * 1)^10000 = 1
        Multiplication 1: (1 * 1)^10000 = 1
        Multiplication 2: (1 * 1)^10000 = 1
        Multiplication 3: (1 * 1)^10000 = 13
        Multiplication 4: (1 * 1)^10000 = 10
(1 + 1)^10000
        Plaintext (0.18600858555603027%): (1 * 1)^10000 = 1
        Decrypted (3.4530000686645511s): (1 + 1)^10000 = 14

(fhe) C:\Users\Daniel\Desktop\Something Awesome>
```

*Code Excerpt and Demo Output*

**Results**

I have gained a strong understanding of HE. The main product was my report (available in the appendix), which is a good representation of my understanding, and is backed up by thorough research (samples available in the appendix). Additionally, implementing an SHE scheme in Python was insightful, and although I followed a guide, I followed the maths instead of copying their code. This implementation provides a solid basis for a demo, to help educate others (i.e. my tutorial) about how HE works and its benefits and shortcomings.

**What I did**

The majority of my time (>20 hours) was spent researching. This involved reading, annotating, and understanding relevant academic papers. Notable academic papers with annotations are available in the appendix. I spent ~5 hours doing maths, including example calculations, proofs, and other ways of understanding what was outlined in many of the papers. Some calculations are available in the papers in the appendix. I spent ~5 hours programming a toy SHE implementation, and a demo using the toy SHE implementation to help explain homomorphic encryption and its benefits and shortcomings. Both are linked in the appendix. The rest of my time (>10 hours) was spent writing the explainer report, and this report. The explainer report is linked in the appendix.

My main time management problem was overfixation on research. I felt that the priority was to be an expert on HE, as opposed to producing something. Ultimately producing a report was the best way to recover from this, as it demonstrates the knowledge gained from all of the research in a tangible manner.

**How I was challenged**

*A few key things that I learned:*

- Homomorphic encryption is a vast field with no single structure of scheme.
- Ring theory, lattices, and where the security of advanced encryption schemes comes from - i.e. concepts including RLWE, and the shortest nonzero vector in a high-dimensional lattice problem.
- How one goes about developing an encryption scheme - the motivation for this project was to ‘roll my own crypto’ because Richard Buckland kept saying not to in the lectures. I didn’t make enough progress during the term to ‘roll my own scheme’, but I feel like I’ve developed a deep enough understanding to do it these holidays to (jokingly) spite Richard. My ‘aha’ moment was when following a guide on implementing toy SHE, and it was devising homomorphic multiplication. The process was simply to multiply the two noisy scaled ciphertexts, and modify the result until it was something workable.

*What was hard:* Connecting abstract maths concepts like rings to encryption in practice - after a few papers and guides I found a few amazing explanations that made perfect sense. Additionally, time management - as always, my project was over-ambitious given time constraints. Balancing it with two other demanding subjects, part-time work, and other commitments was challenging, but worthwhile.

*Problems I faced:* Being sick for the majority of the term meant poor time management up until flex week, which made things much harder. My strategy for dealing with this was putting in whatever effort I could, whenever I was able to - for example, I did most of my research on the bus to and from work.

*What have I learned about myself and project management:* I get trapped in the research phase, and I’m bad at refining my scope as I progress. I’ve done this in many of my projects - I set out with a goal to deeply understand something, find out that it’s actually a broad field, and then fail to refine my scope.

*What would I do differently next time:* prioritising this over other work, and trying harder to not get sick - it sounds dumb, but my time management was so bad because I constantly overexert myself - if I were to ‘groundhog day it’, I would take this term easier and focus more on the projects I enjoy (this one!).