NDnano
Notre Dame Nanoscience and Technology

NDnano Summer Undergraduate Research
2022 Project Summary

1. Student name & home university: Jeremy Stevens, University of Notre Dame

2. ND faculty name & department: Ningyuan Cao, Department of Electrical Engineering

3. Summer project title: Computation modeling for homomorphic encryption in privacy-preserving vehicular network

4. Briefly describe new skills you acquired during your summer research:

During the summer, I've obtained many skills for maintaining and updating a GitHub repository for the encryption model I was developing. I learned how to implement algorithms from papers and mathematical concepts into Python. After working on the model, I learned how to effectively present my findings and discuss the importance of my research.

5. Briefly share a practical application/end use of your research:

The end use of my research is to develop a model for estimating computation cost for homomorphic encryption. This model will aid with implementing new algorithms which will accelerate homomorphic encryption computations which will eventually allow for practical uses of homomorphic encryption.

6. 50- to 75-word abstract of your project:

The aim of this project is to decrease computational cost of homomorphic encryption by developing an encryption model to implement improved algorithms. Models for BFV and CKKS encryption schemes were developed and tested. Accelerated algorithms such as Barrett Reduction and the Number Theoretic Transform were implemented, and the respective gains for each were recorded.

7. References for papers, posters, or presentations of your research:

[1] J. Fan and F. Vercauteren, "Somewhat Practical Fully Homomorphic Encryption," *Cryptology ePrint Archive*, 2012, Accessed: Jul. 28, 2022. [Online]. Available: https://eprint.iacr.org/2012/144

[2] H. Chen, K. Laine, and R. Player, "Simple Encrypted Arithmetic Library - SEAL v2.1," in *Financial Cryptography and Data Security*, Cham, 2017, pp. 3–18. doi: 10.1007/978-3-319-70278-0_1.

[3] J. H. Cheon, Y. Son, and D. Yhee, "Practical FHE parameters against lattice attacks," *Cryptology ePrint Archive*, 2021, Accessed: Jul. 28, 2022. [Online]. Available: https://eprint.iacr.org/2021/039

[4] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic Encryption for Arithmetic of Approximate Numbers," in *Advances in Cryptology* – ASIACRYPT 2017, Cham, 2017, pp. 409–437. doi: 10.1007/978-3-319-70694-8_15.

[5] Z. Liu *et al.*, "High-Performance Ideal Lattice-Based Cryptography on 8-Bit AVR Microcontrollers," *ACM Trans. Embed. Comput. Syst.*, vol. 16, no. 4, p. 117:1-117:24, Jul. 2017, doi: 10.1145/3092951.

One-page project summary that describes problem, project goal and your activities / results:

The focus of this research is to develop a model to determine the computational costs of different algorithms for homomorphic encryption schemes. Homomorphic encryption is an emerging technology which would allow centralized computation resources, such as cloud servers, to work with encrypted data without knowing the contents of the data it is performing operations on. Fully homomorphic encryption schemes were first proven in 2009, however every fully homomorphic encryption scheme is unfeasible for practical purposes because the computational overload required when implementing these encryption schemes. Research is being done to further decrease the overhead operations required when implementing fully homomorphic encryption schemes.

Before beginning development of the model, it was important to decide which encryption schemes we wanted to focus on. Since the creation of the first homomorphic encryption scheme, many distinct schemes have been theorized. With this research, we focused on two leveled homomorphic encryption schemes, BFV and CKKS. We decided to model these schemes because they are similar with their fundamentals concepts and are implemented in popular open-source libraries, such as Microsoft SEAL and PALISADE. In both schemes, the fundamental concept for the encryption schemes is to represent the ciphertext as polynomials in rings. A polynomial ring is a set of polynomials with the coefficients bounded in a field. Using these polynomial rings, both encryption schemes rely on the Ring Learning with Errors problem as security for the encryption scheme, which is a computational problem that is theorized to even be difficult for a quantum computer to solve.

After deciding to focus on BFV and CKKS, we started working on models for the encryption schemes in Python. When developing these models, most of the code was written with custom classes we created. To accurately access the computation overhead that comes with these encryption schemes, it was important that we knew how each of the different parts of the encryption scheme were implemented so we could record computational costs accordingly. As such, we sparingly used outside libraries in our repository, one library we did use was NumPy to only generate random numbers. To verify the validity of the encryption schemes, we would use the encryption parameters provided in the papers to see if it worked properly.

Once we were able to verify the validity of the different encryption schemes, we then focused on creating a class to record the computation count in the encryption schemes. This class would record the number of additions, multiplications, and modulo operations in different sections of the encryption scheme, such as encryption, decryption, and key generation. The different sections would provide insight to the most expensive parts of the encryption schemes, which would guide us in our focus for decreasing computation cost.

The first area we improved upon was decreasing modulo operation cost. We implemented Barrett Reduction in replacement of ‘schoolbook’ modulo operations. Barrett Reduction is designed to optimize large modulo operations when the modulus is constant, which is the case in BFV and CKKS. After the implementation of Barrett Reduction, we integrated the Number Theoretic Transform into the encryption schemes to reduce the cost of polynomial multiplication. The Number Theoretic Transform is a discrete Fourier transform over a ring, which would decrease the computational complexity of polynomial multiplications, from $O(n^2)$ to $O(n*\log(n))$. After implementing both Barrett Reduction and the Number Theoretic Transform, we saw a speed-up of about 200x when compared to an original encryption scheme with neither improved algorithm.

In the future, we continue to implement more advanced algorithms to decrease the computation cost required for the implementation of these schemes. We also plan to test improvements that can be seen when performing these operations on specialized hardware. The code for these models is public in GitHub repositories for other researchers to utilize. GitHub Link