동형암호 기술의 연구 동향

안상우, 임한섭, 이태호, 조영진, 박준우
한국정보통신기술협회
swan06@tta.or.kr, hslim@tta.or.kr, lth1124@tta.or.kr, choyj@tta.or.kr, junusee@tta.or.kr

Studies on Homomorphic Encryption: A Survey

An Sang Woo, Lim Han Seop, Lee Tae Ho, Cho Young Jin, Park Jun Woo
Telecommunications Technology Association (TTA)

요 약

암호화된 데이터를 안전하게 보호하면서도 유용하게 활용할 수 있는 동형암호 기술은 수많은 분야에서의 관심 주제가 되었다. 초기에는 매우 비효율적이었던 동형암호 연산 성능이 지속적인 연구에 의해 개선되었으며, 덧셈과 곱셈만 지원했던 초기 부분동형암호의 연산 기능 또한 완전동형암호를 통해서 모든 연산을 지원할 수 있게 발전하였다. 현재도 동형암호의 추가 연산 부하를 최적화하고 병렬 연산 플랫폼 상에서 고속화하기 위한 다양한 연구가 수행되고 있으며, 동형암호를 실용화하고자 하는 연구 및 개발 결과들이 제시되고 있다. 본 논문에서는 동형암호 기술에 대한 연구 동향을 살펴보고, 동형암호 기술을 활용한 연구 및 개발 분야를 소개함으로써 동형암호의 현황과 활용 전망을 제시한다.

Ⅰ. 서 론

동형암호(Homomorphic Encryption)는 각 평문에 대한 연산을 수행한 뒤 암호화한 결과가 각 평문을 암호화한 암호문에 대한 연산을 수행한 결과가 같은 암호 기술이다. 동형암호 기술 자체는 1978년에 제안되었지만 [1], 최근 개인정보 유출 방지의 중요성이 대두되면서 암호화된 정보 상에서 연산이 가능한 동형암호 기술의 관심이 증대되었다. 현재 의료 산업, 금융 데이터, 통계 시스템 등 다양한 분야에서 동형암호 기술을 활용하기 위한 연구가 수행되고 있으며, 동형암호 기술의 한계점인 성능 및 소요 자원을 최적화하는 연구 결과 또한 제시되고 있다. 본 논문에서는 동형암호 기술 및 활용 연구에 대한 동향을 정리하고, 미래 디지털 산업에서의 동형암호 기술의 전망을 소개하고자 한다. 본 논문의 구성은 다음과 같다: 2장에서는 동형암호에 대한 개념을 서술한다. 3장에서는 동형암호 기술에 대한 연구 동향을 분석한다. 4장에서는 동형암호 기술에 대한 응용 및 활용 연구 동향을 소개한다. 이후, 5장에서 결론으로 정리한다.

Ⅱ. 동형암호 개념

기존 보안 시스템 상에서는 암호화된 데이터일지라도 데이터를 사용하거나 처리하기 위해서는 복호화 과정을 수행한 뒤 평문에 대한 연산을 수행한 뒤에 다시 암호화를 수행해야 하였으나, 동형암호 기술의 등장으로 인하여 중요한 데이터를 추출/복호화 하지 않고도 작업을 수행할 수 있는 가능성이 제시되었다. 동형 암호의 개념도는 다음 수식으로 요약된다.

$$E(M_1) \circ E(M_2) = E(M_1 \circ M_2) \quad \cdots (1)$$

초기 부분동형암호의 경우 덧셈과 곱셈의 경우에만 위의 수식이 성립하였으나 [2], Gentry [3]로부터 제안된 완전동형암호(Fully Homomorphic Encryption, FHE) 기술이 발전하면서 최근에는 전체 사칙연산과 XOR과 AND 등의 모든 논리 연산에도 동형암호 기술을 적용할 수 있게 되었다. 그러나 이러한 동형암호는 연산 시 마다 암호문에 Noise가 발생하여 일정 한계를 넘어서면 복호화가 불가능해지기 때문에, 재부팅(Bootstrap) 기법을 통해 Noise를 중간에 제거하는 방향을 적용하였고, 현재 Noise 발생을 줄이거나 재부팅의 추가 부하를 줄이는 최적화 연구가 수행되고 있다.

Ⅲ. 동형암호 기술 연구 동향

완전동형암호를 바탕으로 제시된 대표적인 Scheme은 BGV/BFV [4], FHEW/TFHE [5, 6], CKKS [7]가 있다. BGV는 2012년에 제안된 유한체 정수 기반 동형암호 Scheme이며, FHEW는 2015년에 제안된 비트 연산 기반 동형암호 Scheme이다. 2017년에는 최초의 실수 연산을 지원하는 CKKS 동형암호 Scheme이 제안되었다. 완전동형암호 Scheme들이 활용된 다양한 오픈 소스 구현물들은 다음과 같다.

- HEaaN [7]: Seoul National Univ.에서 개발되었으며 CKKS를 지원함.
- SEAL [8]: Microsoft에서 개발되었으며 BFV, CKKS를 지원함.
- HElib [9]: IBM에서 개발되었으며 BGV, CKKS를 지원함.
- Concrete [10]: Rust로 작성된 TFHE 변형 격자 기반 라이브러리.
- NFLlib [11]: BGV/BFV를 최적화한 NTT 기반 라이브러리.
- cuFHE [12]: 동형암호를 GPU 상에서 가속화하기 위해 개발됨.
- Lattigo [13]: Golang으로 작성되었으며 BGV, TFHE, CKKS를 지원함.
- OpenFHE [14]: 다양한 Scheme을 지원하는 격자 기반 라이브러리. PALISADE [15]가 포함되어 있음.

1. 동형암호 연산에 대한 최적화 연구 동향

동형암호에 대한 다양한 연구들은 주로 동형암호의 성능을 개선시키기 위한 다양한 최적화 기법이나 병렬 연산 환경에서의 가속화 등을 주제로 수행되고 있다. 2016년 [16]에서는 FHEW Scheme 상에서 재부팅 시간을 0.1초로 줄이는 연구 결과를 제시했으며, 2019년 [17]에서는 CKKS Scheme 상에서 재부팅 시간을 0.01초까지 줄일 수 있음을 소개하였다. 2021년에는 [18]에서 Lattigo를 타겟으로 더욱 더 효율적인 재부팅 절차를 제안하였으며, [19]과 [20]에서는 THFE의 재부팅 내의 구성 함수들을 효

과적으로 연산할 수 있는 기법을 제안하였다.

2. 하드웨어 장비를 활용한 가속화 연구 동향

하지만 소프트웨어 구현만으로는 동형암호의 추가 연산 부하를 해소하기 힘들기 때문에, GPU나 FPGA와 같은 하드웨어 장비를 활용하여 동형암호를 병렬적으로 가속화하기 위한 연구도 수행되고 있다. 2019년 [21]에서는 FPGA를 활용하여 BGV Scheme 기반 동형암호 곱셈 연산을 병렬적으로 처리할 수 있는 방안을 제안하였다. [22]에서는 CKKS Scheme 기반에서 FPGA를 활용해 실수 기반 동형암호 곱셈 연산을 가속화하였다. [23]에서는 GPU를 활용하여 재부팅 시간을 고속화하는 결과를 제시하였으며, [24]에서는 TFHE를 GPU 상에서 최적화한 연구를 수행하였다. [25]에서는 ASIC 레벨의 연산 가속기를 개발하여 동형암호를 고속화하였다.

3. 동형암호에 특화된 컴파일러 개발 및 동형암호 기술 표준화 동향

가속화 이외에도 동형암호를 위한 다양한 컴파일러가 개발되고 있다. [26]에서는 Microsoft SEAL을 대상으로 CKKS를 위한 컴파일러 및 최적화 프로그램을 제안하였다. [27]에서는 SEAL, HElib, FHEW, TFHE, PALISADE 등 대부분의 동형암호 라이브러리를 지원하는 E3 컴파일러가 소개되었고, HEaaN을 지원하는 CHET 컴파일러가 [28]에서 제시되었다.

각종 국제 기관과 기업들은 동형암호를 표준화하기 위해 각종 워크숍과 표준화 회의를 진행하고 있다. ISO/IEC에서는 2019년 동형암호에 대한 표준화 작업이 수행된 바 있으며 [29], 2024년 완전동형암호 표준화 발간을 목표로 표준화 Draft를 진행하고 있다 [30]. 또한 최신 연구 주제와 개발 분야를 공유하는 Homomorphic Encryption Standardization Consortium에서는 다양한 학계/산업계/정부 기관이 협력하여 각종 보안 파라미터 등을 정립한 표준 [31]을 제정한 바 있으며, Workshop on Encrypted Computing & Applied Homomorphic Cryptography(WAHC)를 매년 개최하고 있다 [32].

Ⅳ. 동형암호 기술 응용 및 활용 동향

동형암호는 서비스 제공자가 사용자의 정보를 암호화된 상태로 분석 및 활용할 수 있다는 점에 따라 다양한 데이터 프라이버시 분야에서 활용될 수 있다. 구글은 동형암호 라이브러리 SHELL을 이용하여 기기 정보를 보호하는 Private Set Membership 서비스를 개발하고 있다 [33]. Microsoft Edge의 경우 준동형암호 기법을 사용하여 사용자의 패스워드를 노출하지 않은 채 사용자 암호의 노출 여부를 알려주는 패스워드 모니터링 서비스를 제공하고 있다 [34]. 국내 국민연금공단에서는 HEaaN을 적용하여 국민연금 데이터와 KCB 신용데이터를 결합 후 분석한 사례가 있다[35]. OpenFHE의 경우 Google Transpiler 프로젝트의 백엔드에 사용되고 있다 [36]. [37]에서는 동형암호를 활용하여 개인의 건강 데이터를 손상시키지 않으면서 인공 췌장 장치 시스템의 보안성을 확보할 수 있는 방법을 제안하였다. 최근에 동형암호의 성능 최적화 및 가속화 연구가 적극적으로 수행되면서, 동형암호 기술이 인공신경망 등의 기계학습에 충분히 활용될 수 있음을 보였다. 2016년 [38]에서 암호화된 데이터에 대한 신경망 연구를 수행한 바 있으며, 2020년에는 기계학습 방법 중 하나인 Support Vector Machine(SVM) 상에서 동형암호를 통한 개인정보 보호 기술이 연구된 바 있고 [39], 2022년에는 클라우드가 암호화 데이터를 운용할 수 있도록 학습시킨 Private AI를 제시하였다 [40]. 또한 근래 양자 컴퓨터의 개발로 인하여 대두되고 있는 양자내성암호 또한 대부분 격자기반 암호알고리즘을 기반으로 구성되어 있기 때문에 동형암호 기술과의 연구 시너지가 발생할 전망이다.

Ⅴ. 결론

ISO/IEC에서는 2019년 동형암호에 대한 표준화에 이어 2020년부터 완전동형암호에 대한 표준화를 추진하고 있으며, 2024년 표준안 발간을 목표로 진행하고 있다. Microsoft, Google, IBM, Naver 등의 기업들 또한 차세대 암호 기술인 동형암호를 활용한 개인정보 보호, 통계 분석, 프라이버시 보호 기반 데이터 학습 시스템을 구축하고 있다. 동형암호에 대한 다양한 연구가 수행됨에 따라 동형암호에 대한 실용화는 현재 진행형이 되고 있으며, 동형암호의 실용화는 차세대 암호 기술로서 정보보안 분야 전체에 필수적인 요소로 자리매김함을 의미한다. 따라서, 동형암호 기술은 모든 시스템에 증가하는 수요에 따라 앞으로도 적극적인 연구가 수행되어야 하며, 추후 다양한 플랫폼에서 동형암호를 최적화하고 활용하는 방안이 연구되어야 한다.

참 고 문 헌

[1] Rivest, R. L., and Dertouzos, M. L. “ON DATA BANKS AND PRIVACY HOMOMORPHISMS.” (1978).

[2] Gamal, T. E. “A public key cryptosystem and a signature scheme based on discrete logarithms.” CRYPTO, 10–18. (1984).

[3] Gentry, C. “Computing arbitrary functions of encrypted data. Commun.” ACM 53, 3 (March 2010), 97–105. (2010).

[4] Brakerski, Z., Gentry, C., and Vaikuntanathan, V. “(Leveled) fully homomorphic encryption without bootstrapping”. The 3rd Innovations in Theoretical Computer Science Conference (ITCS '12). Association for Computing Machinery, New York, NY, USA, 309–325. (2012).

[5] Ducas, L., and Micciancio, D. “FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second.” Advances in Cryptology – EUROCRYPT 2015, Lecture Notes in Computer Science, vol 9056. Springer, Berlin, Heidelberg. (2015).

[6] Chillotti, I., Gama, N., Georgieva, M., and Izabachène, M. “TFHE: Fast Fully Homomorphic Encryptionover the Torus.” In Journal of Cryptology, vol 33, 34–91. (2020).

[7] Cheon, J.H., Kim, A., Kim, M., and Song, Y. “Homomorphic Encryption for Arithmetic of Approximate Numbers.” Advances in Cryptology – ASIACRYPT 2017, Lecture Notes in Computer Science, vol 10624, Springer, Cham. (2017).

[8] Microsoft Research. “Microsoft SEAL.” release 4.0, github; https://github.com/Microsoft/SEAL. (2022).

[9] IBM. “HElib.” release 2.2.1, github; https://github.com/homenc/HElib. (2021).

[10] Zama. “Concrete.” release 0.3.0, github; https://github.com/zama-ai/concrete. (2022).

[11] Aguilar-Melchor, C., Barrier, J., Guelton, S., Guinet, A., Killijian, M. O., and Lepoint, T. “NFLlib: NTT-Based Fast Lattice Library.” In Proceedings of the RSA Conference on Topics in Cryptology – CT-RSA 2016, vol 9610, Springer-Verlag, Berlin, Heidelberg, 341–356. (2016).

[12] Vernam Group, “cuFHE.” beta release 1.0, github; https://github.com/vernamlab/cuFHE. (2018).

[13] Mouchet, C., Bossuat, J., Troncoso-Pastoriza, J. R., and Hubaux, J. “Lattigo: a Multiparty Homomorphic Encryption Library in Go.” (2020).

[14] Badawi, A. A., Bates, J., Bergamaschi, F., Cousins, D. B., Erabelli, S., Genise, N., Halevi, S., Hunt, H., Kim, A., Lee, Y., Liu, Z., Micciancio, D., Quah, I., Polyakov, Y., RV., S., Rohloff, K., Saylor, J., Suponitsky, D., Triplett, M., Vaikuntanathan, V., and Zucca, V. “OpenFHE: Open-Source Fully Homomorphic Encryption Library.” Cryptology ePrint Archive, Paper 2022/915. (2022).

[15] PALISADE Project. “PALISADE Homomorphic Encryption Software Library.” release 1.11.8, gitlab; https://gitlab.com/palisade/palisade-release. (2022).

[16] Chillotti, I., Gama, N., Georgieva, M., and Izabachène, M. “Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds.” Advances in Cryptology - ASIACRYPT 2016, 3-33, Berlin, Heidelberg. (2016).

[17] Chen, H., Chillotti, I. and Song, Y. “Improved Bootstrapping for Approximate Homomorphic Encryption.” Advances in Cryptology - EUROCRYPT 2019, 34-54. (2019).

[18] Bossuat, J.-P., Mouchet, C., Troncoso-Pastoriza, J., and Hubaux, J.-P. “Efficient Bootstrapping for Approximate Homomorphic Encryption with Non-sparse Keys.” Advances in Cryptology - EUROCRYPT 2021, 587-617. (2021).

[19] Guimarães, A., Borin, E., and Aranha, D. F. “Revisiting the functional bootstrap in TFHE.” IACR Transactions on Cryptographic Hardware and Embedded Systems, vol 2021(2), 229-253. (2021).

[20] Chillotti, I., Ligier, D., Orfila, J.-B., and Tap, S. “Improved Programmable Bootstrapping with Larger Precision and Efficient Arithmetic Circuits for TFHE.” Advances in Cryptology - ASIACRYPT 2021, 670-699. (2021).

[21] Roy, S.S., Turan, F., Järvinen, K., Vercauteren, F., and Verbauwhede, I.M. “FPGA-Based High-Performance Parallel Architecture for Homomorphic Computing on Encrypted Data.” 2019 IEEE International Symposium on High Performance Computer Architecture (HPCA), 387-398. (2019).

[22] Riazi, M.S., Laine, K., Pelton, B., and Dai, W. “HEAX: High-Performance Architecture for Computation on Homomorphically Encrypted Data in the Cloud.” IACR Cryptol, ePrint Arch, 2019/1066. (2019).

[23] Jung, W., Kim, S., Ahn, J. H., Cheon, J. H., and Lee, Y. “Over 100x Faster Bootstrapping in Fully Homomorphic Encryption through Memory-centric Optimization with GPUs.” IACR Transactions on Cryptographic Hardware and Embedded Systems, vol 2021(4), 114-148. (2021).

[24] Morshed, T., Aziz, M., and Mohammed, N. “CPU and GPU Accelerated Fully Homomorphic Encryption.” 2020 IEEE International Symposium on Hardware Oriented Security and Trust, San Jose, CA, 142-153. (2020).

[25] Samardzic, N., Feldmann, A., Krastev, A., Devadas, S., Dreslinski, R., Peikert, C., and Sanchez, D. “F1: A Fast and Programmable Accelerator for Fully Homomorphic Encryption.” MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture, 238-252, Virtual Event, Greece. (2021).

[26] Chowdhary, S., Dai, W., Laine, K., and Saarikivi, O. “EVA Improved: Compiler and Extension Library for CKKS.” 9th on Workshop on Encrypted Computing & Applied Homomorphic Cryptography (WAHC '21), Association for Computing Machinery, New York, NY, USA, 43-55. (2021).

[27] Chielle, E., Mazonka, O., Tsoutsos, N.G., and Maniatakos, M. “E3: A Framework for Compiling C++ Programs with Encrypted Operands.” IACR Cryptol, ePrint Arch, 1013. (2018).

[28] Dathathri, R., Saarikivi, O., Chen, H., Laine, K., Lauter, K., Maleki, S., Musuvathi, M., and Mytkowicz, T. “CHET: An Optimizing Compiler for Fully-Homomorphic Neural-Network Inferencing.” 40th ACM SIGPLAN Conference on Programming Language Design and Implementation, 142-156, Phoenix, AZ, USA. (2019).

[29] ISO/IEC 18033-6:2019: IT Security techniques - Encryption algorithms - “Part 6: Homomorphic encryption.” Published, International Organization for Standardization. (2019).

[30] ISO/IEC 18033-8: Information security - Encryption algorithms - “Part 8: Fully Homomorphic encryption.” Under development, International Organization for Standardization. (2022).

[31] Albrecht, M., Chase, M., Chen, H., Ding, J., Goldwasser, S., Gorbunov, S., Halevi, S., Hoffstein, J., Laine, K., Lauter, K., Lokam, S., Micciancio, D., Moody, D., Morrison, T., Sahai, A., and Vaikuntanathan, V. “Homomorphic Encryption Security Standard.” HomomorphicEncryption.org, Toronto, Canada. (2018).

[32] ACM. “WAHC 2022 - 9th Workshop on Encrypted Computing & Applied Homomorphic Cryptography.” Available Online; https://homomorphicencryption.org/workshops/wahc22/. (2022).

[33] Yeo, K., Patel, S., and Private Computing Team. “Protecting your device information with Private Set Membership.” Google Security Blog, Available Online; https://security.googleblog.com/2021/10/protecting-your-device-information-with.html. (2021).

[34] Lauter, K., Kannepali, S., Laine, K., and Moreno, R. C. “Password Monitor: Safeguarding passwords in Microsoft Edge.” Microsoft Research, Security, privacy, and cryptography. (2021).

[35] 전하경. “동형암호, 데이터 프라이버시를 위한 완벽한 기술.” 한국금융, (2016). Available Online; https://www.fntimes.com/html/view. php?ud=2020061610214591018a55064dd1_18. (2016).

[36] Naik, A. R. “Google Launches General Purpose Transpiler For Fully Homomorphic Encryption.” Analytics India Magazine (2021).

[37] Weng, H., Hettiarachchi, C., Nolan, C., Suominen, H., and Lenskiy, A. “Ensuring security of artificial pancreas device system using homomorphic encryption.” Biomedical Signal Processing and Control, 79, 104044. (2022).

[38] Gilad-Bachrach, R., Dowlin, N., Laine, K., Lauter, K., Naehrig, M., and Wernsing, J. “CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy.” The 33rd International Conference on Machine Learning, Machine Learning Research, 48:201-210. (2016).

[39] Park, S., Byun, J., Lee, J., Cheon, J. H., and Lee, J. “HE-Friendly Algorithm for Privacy-Preserving SVM Training.” IEEE Access, 8, 57414-57425. (2020).

[40] Lauter, K. “Private AI: Machine Learning on Encrypted Data.” Recent Advances in Industrial and Applied Mathematics Cham: Springer International Publishing. (97-113). (2022).