5ZC-01

# スマートフォン・クラウドサーバ連携時における準同型暗号化手法の比較

松本 茉倫 †　　小口 正人 †
† お茶の水女子大学

## 1　はじめに

IoT デバイスで取得したセンサデータの中には，秘匿性が高いデータが存在しており，安全とは言えないクラウドサーバ上では情報漏洩に備え，個人情報を保護する必要がある．そこで，暗号文同士の加算・乗算が任意回数可能な Leveled 準同型暗号 (以下 LHE: Leveled Homomorphic Encryption) が注目されている． LHE によって，IoT デバイスで個人情報を暗号化しクラウドサーバで暗号化したまま分析，データ分析者が分析結果を復号することが可能になる．しかし，LHE による暗号化は時間がかかり，暗号文サイズが大きいため通信量が多くなってしまうことが計算能力の低い IoT デバイス上での実装の課題となっている． IoT デバイスの中でも，スマートフォンは近年ユーザにとって極めて使用頻度が高い重要なデバイスであり，ユーザのプライベートな情報を含んだ非常に多くのデータを取得している． 本稿では，スマートフォンで取得したデータのクラウドサーバを使ったデータ活用を想定してシステムデザインを既存手法と提案手法で 3 つ実装し，IoT デバイスの実行時間・クラウドサーバの実行時間・通信量の 3 つの指標で比較した．その結果，提案手法が既存手法に比べてスマートフォンの実行時間を約 589 分の 1 に削減可能であることを示した．

## 2　システムデザイン

図 1 は，システムを構成する IoT デバイス (IoT)・クラウドサーバ (CS)・復号サーバ (DS) の役割を示している．本稿では 3 つのシステムデザインを比較する．

### 2.1　デザイン 1 (Baseline)

Baseline として，シンプルに LHE で平文を暗号化するデザイン 1 は以下の処理を行う．

1. DS が LHE の公開鍵を IoT デバイスとクラウドサーバに配布．
2. IoT が LHE で平文を暗号化．
3. CS が LHE 暗号文を分析．
4. DS が分析結果を復号．

Comparison of Homomorphic Encryption Methods when Linking Smartphones and Cloud Servers
†Marin Matsumoto　†Masato Oguchi
†Ochanomizu University

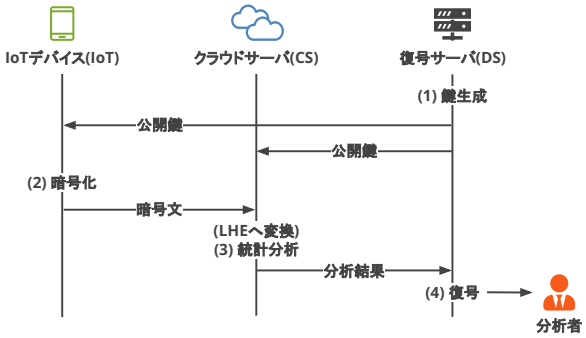



図 1: IoT デバイス (IoT)・クラウドサーバ (CS)・復号サーバ (DS) の役割

### 2.2　デザイン 2 (既存手法)

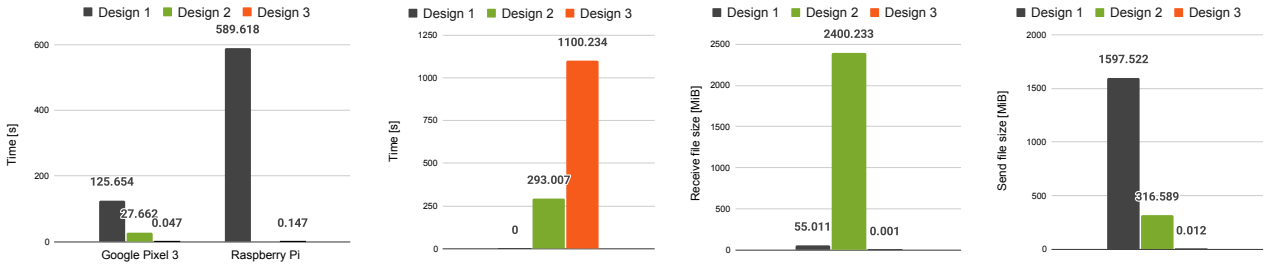
既存手法のデザイン 2 の処理を以下に示す．この手法は，Lauter ら [1] によって提案され，Gentry ら [2] によって実装手法が提案された．AES 鍵のみを LHE で暗号化することによって，高負荷な LHE を多く使わなくて済む．

1. DS が LHE の公開鍵を IoT と CS に配布．
2. IoT が AES で平文を暗号化し，LHE で AES 鍵を暗号化．
3. CS が AES 暗号文を LHE 暗号文へ変換して，LHE 暗号文を分析．
4. DS が分析結果を復号．

### 2.3　デザイン 3 (提案手法)

提案手法のデザイン 3 は IoT デバイスが LHE より軽量な RLWE 問題ベースの加法準同型暗号 (以下 AHE: Additive Homomorphic Encryption) で平文を暗号化し，代わりにクラウドサーバが LHE 暗号文へ変換する手法である．LHE 暗号文への変換は，LHE で暗号化された AHE の秘密鍵と AHE 暗号文との加算・乗算によって可能となる．以下に処理を示す．

1. DS が LHE で暗号化した AHE の秘密鍵は CS に，AHE の公開鍵を IoT に配布．
2. IoT が AHE で平文を暗号化．
3. CS が AHE 暗号文を LHE 暗号文へ変換し，LHE 暗号文を分析．
4. DS が分析結果を復号．




(a) IoT の実行時間　(b) CS の実行時間　(c) 通信量 (DS–IoT 間)　(d) 通信量 (IoT–CS 間) 量

図 2: 256B の平文を LHE 暗号化する場合の実験結果．デザイン 1 が Baseline，デザインが既存手法，デザイン 3 が提案手法．

## 3　実験

### 3.1　実験概要

3 つのデザインで 256B の平文を LHE で暗号化する場合を想定する．表 1 のように，IoT デバイスにはスマートフォンの Google Pixel 3 に加えて，低性能なデバイスである Raspberry Pi を使用した．IoT デバイスの実行時間・クラウドサーバの実行時間 (クラウドサーバで LHE 暗号文へ変換する場合の実行時間)・通信量 (IoT デバイス–クラウドサーバ間，復号サーバ–IoT デバイス間で IoT デバイスが送受信するファイルサイズ) という 3 つの指標で比較を行った．表 2 のパラメータについて，クラウドサーバが LHE 暗号文への変換処理後，Level は 4 程度残るように選択し，同程度のセキュリティレベルになるように設定した．Level が 4 程度残るとは，3 回程度の準同型乗算は可能であることを指す．デザイン 2(既存手法) では AES から LHE への変換処理で準同型乗算の回数が他 2 つのデザインよりも多いため Level は高く設定されている．

### 3.2　実験結果

図 2(a) より，デザイン 3(提案手法) は IoT デバイスの実行時間に関して最も有利であることが分かる．Raspberry Pi ではメモリ不足のためデザイン 2(既存手法) は実行不可能であった．この理由は，図 2(c) の DS-IoT 間の通信量，すなわち IoT デバイスが使う暗号化鍵のサイズが既存手法では約 2.3GiB と特に大きいためだと考えられる．既存手法では LHE パラメータの Level が大きいため鍵が大きくなる．提案手法は，図 2(c)・2(d) の通信量に関しても最も優れている．一方で，図 2(b) の変換処理には課題が残っていることが分かった．

## 4　まとめと今後の課題

IoT デバイスの LHE 暗号化の負荷を削減するため，IoT デバイス側では LHE よりも軽量な AHE によって平文を暗号化し，代わりにクラウドサーバ側で AHE から LHE の暗号文へ変換する手法を提案した．実験の結果，既存手法に比べて，提案手法では IoT デバイスへの負荷を格段に減らすことが可能であり，Raspberry Pi のようなメモリの限られたデバイスでも実行可能であることを示した．よって，提案手法の方が優れたシステムデザインであると考えられる．

今後の課題として，クラウドサーバにおける変換処理の高速化が挙げられる．

表 1: 実験環境

| | | |
|---|---|---|
| Google Pixel 3 (IoT デバイス) | OS | Android 9.0 |
| | CPU | ARM Cortex-A75(2.5 GHz) 4Cores<br>ARM Cortex-A55(1.6 GHz) 4Cores |
| | Main Memory | 4GB |
| Raspberry Pi 3 Model B+ (IoT デバイス) | OS | Raspbian GNU/Linux 10.0 |
| | CPU | ARM Cortex-A53(1.4 GHz) 4Cores |
| | Main Memory | 1GB |
| クラウドサーバ | OS | CentOS 6.10 |
| | CPU | Intel®Xeon®Processor E5-2643 v3 (3.4GHz) 6Cores × 2 Sockets |
| | Main Memory | 512GB |

表 2: パラメータ

| | | |
|---|---|---|
| デザイン 1 | LHE | Level: 7, Security:130 bit |
| デザイン 2 | LHE | Level: 45, Security: 131 bit |
| | AES | Key length: 128 bit |
| デザイン 3 | LHE | Level: 8, Security: 130 bit |
| | AHE | Degree of a polynomial: 128, Security: 128 bit |

## 謝辞

本研究は一部，JST CREST JPMJCR1503 の支援を受けたものである．

## 参考文献

[1] Kristin Lauter, Michael Naehrig, and Vinod Vaikuntanathan. Can homomorphic encryption be practical? In *CCSW '11: Proceedings of the 3rd ACM workshop on Cloud computing security workshop*, pp. 113–124, 2011.

[2] Craig Gentry, Shai Halevi, and Nigel P. Smart. Homomorphic evaluation of the aes circuit. In *Advances in Cryptology – CRYPTO 2012*, pp. 850–867, 2012.