Vol.2019-MPS-123 No.30
Vol.2019-BIO-58 No.30
2019/6/18

属性ベース暗号とIntel SGXを用いた堅牢かつ柔軟なアクセス制御を実現するデータ分析プラットフォームの構築

岩田 大輝[1,2,a)]　清水 佳奈[1,2,b)]

概要：近年，クラウド上でデータ分析を行うサービスの需要が高まっているが，このようなサービスを安全に運用するためには，個人情報や企業秘密を含むデータを適切に保護する仕組みが必要となる．そこで本研究では，属性ベース暗号と Intel SGX を用いてクラウド上のデータを秘匿したまま分析するシステムを提案する．提案システムでは，属性ベース暗号によってデータを暗号化して保管し，サーバーが分析リクエストを受けると，ユーザーの権限に合致するデータのみを Intel SGX の提供する保護領域内で復号化して必要な分析を実施する．このため，同様の目的を達成する秘密計算技術よりも大幅に計算量が少ない利点がある．提案システムを試験実装して一塩基多型の実データを含む複数のデータセットによる性能評価を実施し，生命情報解析への応用を検討した．

キーワード：属性ベース暗号, Intel SGX, プライバシー保護データマイニング

Secure and Fine-Grained Accessible Data Mining Platform Using Attribute-Based Encryption and Intel SGX

Abstract: Due to recent demand for cloud services that conduct data mining on personal data, it is important to develop an efficient system that enables to analyze data while keeping privacy. In this study, we propose a data analysis system that is based on attribute-based encryption (ABE) and Intel's SGX technology. In our system, all the data is encrypted by the ABE. When the server receives a request from a user, only the subset of data that the user is authorized to access is decrypted and analyzed in the enclave (a secure region provided by Intel SGX), and only the result is returned to the user. Since the system does not have any time-consuming task, it is computationally efficient compared to the multi-party computation, which aims for the similar purpose. We implemented the system and tested it on several datasets including a single nucleotide polymorphism dataset to investigate its performance for biological data analysis.

Keywords: Attribute-Based Encryption, Intel SGX, Privacy-Preserving Data Mining

1. はじめに

クラウドサービスはオンプレミスよりもシステム導入が容易であり，また運用コストを削減できるため，利用する企業やユーザが急速に増加している．さらに，クラウドサービスは計算資源の拡張が柔軟に行えるため，データ分析の分野において，その急速な発展を後押ししている．一方で，クラウドサービスでの個人情報や企業秘密を含むデータの保管や分析には，セキュリティやプライバシーを考慮する必要がある．

不正アクセス対策　クラウドサービスが考慮すべきセキュリティ課題の一つに不正アクセスへの対策がある．多くのアプリケーションでは，権限管理やログ監視によって不正アクセスを防止しているが，急速なユーザ数の増加に伴い，その管理が難しくなりつつある．

属性ベース暗号（Attribute-based Encryption: ABE）[1]は柔軟なアクセス制御とデータの暗号化を同時に実現できる手法で，従来の複雑な権限管理によるアクセス制御を

[1] 早稲田大学 基幹理工学研究科 情報理工・情報通信専攻, Department Computer Science and Computations Engineering, Waseda University
[2] 産総研・早大 生体システムビッグデータ解析オープンイノベーションラボラトリ, AIST-Waseda University Computational Bio Big-Data Open Innovation Laboratory; CBBD-OIL
[a)] d(underbar)iwata@ruri.waseda.jp
[b)] shimizu.kana@waseda.jp

データに付与する復号条件によって実現する．ABE では暗号化をする際に復号可能条件を暗号文に付加する必要があり，その復号可能条件を満たす秘密鍵を持つ者だけがデータを復号化できる．万が一データが外部に漏洩してしまっても，データは暗号化されているため，元データの漏洩を防止できる．

プライバシー対策　プライバシーへの対策もクラウドサービスが考慮すべき課題の一つである．近年は様々な産業で大量のデータが収集されるようになり，データに含まれるプライバシーや機密を考慮しながら，クラウド上で分析を行うサービスの需要が高まっている．ABE はクラウドサービスにおけるセキュリティ対策として有効であるが，データのアクセス制御のみ可能にする暗号化技術であるため，分析時にはデータを復号する必要があり，この要求を満たすことができない．

Intel 社が提供する Software Guard Extension（Intel SGX）[2] は，OS 等による攻撃を防ぐために開発された特殊なハードウェア機構である．Intel SGX は OS からも隔離された保護メモリ領域を定義することができ，アプリケーションで扱うデータを外部から保護することができる．そのため，Intel SGX を利用すると，データに含まれるプライバシーを考慮しながら分析が可能となる．

1.1 本研究の貢献

以上の例に見るように，セキュリティやプライバシーへの対策は喫緊の課題であり，これらを考慮したクラウド上でのデータ分析手法が求められている．このように，データの中身を保護したまま解析を行い，必要な情報のみを抽出する技術は，総称してプライバシー保護データマイニング (Privacy-Preserving Data Mining: PPDM) と呼ばれ，個別の技術に秘密計算や差分プライバシーなどがある．秘密計算や差分プライバシーは，それぞれ計算量と分析精度に課題を抱えており，実用には至っていない．そこで本研究では，ABE と Intel SGX を用いたクラウド上のデータを秘匿したまま分析するシステムを提案する．提案システムでは，ABE によってデータを暗号化して保管し，サーバーが分析リクエストを受けると，ユーザーの権限に合致するデータのみを Intel SGX が提供する保護メモリ領域内で復号化し，必要な分析を実施する．提案システムでは，従来の複雑なアクセス制御を ABE により実装することで，クラウドサービスにおける不正アクセスやアクセス制御の課題を解消した．さらに，Intel SGX による保護メモリ領域での実行により，分析精度を落とすことなく大幅に計算量を削減した．

本研究の新規性は以下の 3 点である．

- ABE によるデータ共有手法に分析を取り入れた点．
- 提案システムが従来手法よりも高速かつ精確な分析ができる点．
- 試験実装と実験を行い，提案システムが実用的であり，また生命情報解析へ応用できることを示した点．

本研究により，個人情報を利用したデータ分析の更なる拡大が期待できる．さらには，データ提供者が地理的に離れていたり，希少なデータを複数の研究機関から集めたりしたい場合に，個人を特定する情報を漏洩させずにデータを収集，分析することが可能となる．

2. 準備

2.1 属性ベース暗号（ABE）

柔軟なアクセス制御とデータの暗号化を同時に実現できる ABE は Sahai ら [1] によって提案された．ABE は個人が所持する “属性” と，復号可能条件（ポリシー）を暗号文，秘密鍵に埋め込むことで実現する．個人が所有する “属性” が，予め設定したポリシーを満たしている場合のみ，暗号文を復号できる．ABE には暗号文に各自の属性，秘密鍵にポリシーを埋め込む方式と，暗号文にポリシー，秘密鍵に各自の属性を埋め込む方式が存在し，それぞれ鍵ポリシー属性ベース暗号（Key-Policy ABE: KP-ABE）[3]，暗号文ポリシー属性ベース暗号（Ciphertext-Policy ABE: CP-ABE）[4] と呼ばれる．KP-ABE は動画などのコンテンツ配信など，CP-ABE は各自の属性が予め把握できる会社などでのデータ共有に応用が期待できる [5]．

本研究では CP-ABE の公開ライブラリの一つである OpenABE [6] を用いて実装を行った．OpenABE をはじめ，CP-ABE は一般的に以下の 4 つのアルゴリズムから構成される．

Setup$(\tau) \rightarrow PK, MSK$　セキュリティパラメータ τ を入力として，公開鍵 PK，マスター秘密鍵 MSK を出力する．ここで，MSK はユーザ用の秘密鍵の生成に利用するパラメータで，厳重に管理する必要がある．

KeyGen$(PK, MSK, \gamma) \rightarrow SK$　公開鍵 PK，マスター秘密鍵 MSK，ユーザの属性集合 γ を入力として，ユーザ用秘密鍵 SK を出力する．SK は作成したのち，ユーザに配布する．

Encrypt$(PK, M, T) \rightarrow CT$　公開鍵 PK，平文 M，アクセス構造 T を入力として，暗号文 CT を出力する．アクセス構造 T は暗号文 CT に付与するポリシーを指す．

Decrypt$(SK, CT) \rightarrow M$　ユーザ用秘密鍵 SK，暗号文 CT を入力として，平文 M を出力する．但し，M を出力するのは，ユーザの属性集合 γ が T を満たす場合のみである．

2.2 Intel SGX

Intel SGX とは，対応する Intel CPU に組み込まれているセキュリティ関連の命令セット群 [2] である．Intel SGX を利用するアプリケーションでは “Enclave” と呼ばれる CPU で生成される鍵で暗号化された保護メモリ領域を定義

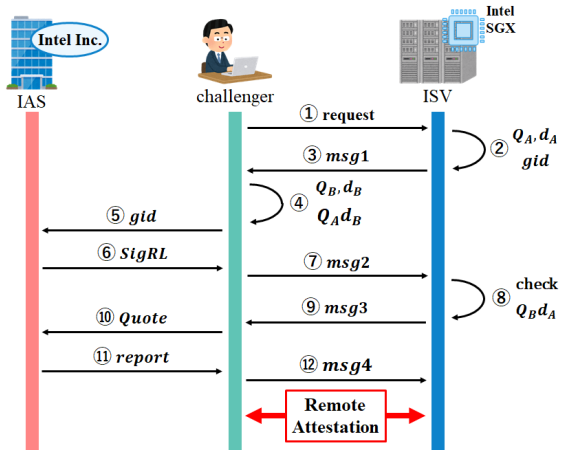

図 1　Remote Attestation の概要
Fig. 1　Overview of Remote Attestation

することができる．Enclave 内部の情報は OS やハイパバイザなど，いかなる権限でもアクセスできないため，アプリケーションで扱うデータやプログラムを OS やマルウェアから保護することができる．

Intel SGX は Intel 製の CPU と Enclave で実行するコードのみ信頼するモデルであり，その特性からプロセス内攻撃や OS・ハイパバイザによる攻撃，外部ハードウェアによる攻撃といった脅威に対して非常に有効である．その一方で，Intel SGX はコンピュータの物理的な事象を観測するサイドチャネル攻撃に対して脆弱であり，SgxPectre [7] や FORESHADOW [8] などの攻撃手法が報告されている．Intel 社らはこれらの対策に取り組んでいるが，開発者でも独自にサイドチャネル攻撃の対策を行う必要がある．

Remote Attestation

Remote Attestation は Intel SGX が提供する機能の一つで，安全なハードウェア環境で Enclave が生成したことを検証，証明し，通信の信頼レベルを高める仕組み [9] である．Remote Attestation は，以下の 3 つのエンティティで構成される．

- 接続を試みる challenger
- SGX 環境を持つサーバー（Independent Software Vendor: ISV）
- 検証を行う第三者機関（Intel Attestation Service: IAS）

Remote Attestation の概要について説明する（**図 1**）．

(1) challenger が ISV に request を送る．
(2) ISV は Enclave を立ち上げ，ECDSA-256bit 公開鍵 Q_A，秘密鍵 d_A，EPID-GID gid を生成する．EPID-GID とは CPU のグループ情報を含む ID で，検証時に利用する．
(3) ISV は Q_A と gid を $msg1$ として challenger に送付する．
(4) challenger は ECDSA-256bit 公開鍵 Q_B，秘密鍵 d_B を生成し，Q_A と d_B から ECDH 共通鍵 $Q_A d_B$ を生成する．
(5) challenger は ISV から取得した gid を IAS に送り，署名失効リスト $SigRL$ を要求する．
(6) IAS は gid を検証し，署名失効リスト $SigRL$ を challenger に返す．
(7) challenger は Q_B や $SigRL$, ECDSA 署名などを $msg2$ として ISV に送付する．
(8) ISV は ECDSA 署名や $SigRL$ を検証し，確認したら Q_B と d_A から ECDH 共通鍵 $Q_B d_A$　を生成する．楕円曲線暗号には，$Q_A d_B = Q_B d_A$ となる特性があり，この特性を利用することで鍵共有が実現する．
(9) ISV は実行中の Enclave の暗号学的ハッシュ等の情報をもつ $Quote$ などを $msg3$ として challenger に送付する．
(10) challenger は ISV から取得した $Quote$ を IAS に送り，検証を依頼する．
(11) IAS は検証結果として $report$ を返し，署名を行い challenger に送付する．
(12) challenger は $report$ の検証を行い，Enclave の信用可否を含む $msg4$ を ISV に送付する．Enclave が信用できる場合は，共通鍵 $Q_A d_B = Q_B d_A$ による Remote Attestation が実現する．

graphene-SGX

Intel SGX は，その特性からクラウドサービスにおけるプライバシー対策の一つとして非常に有効であるが，動的ライブラリのリンクが禁止されていたり，Enclave 内でのシステムコールが禁止されていたりと，その利用に様々な制約が設けられている．そのため，Intel SGX を用いた開発には従来のプログラムを大幅に変更する必要があり，利用拡大の足枷となっていた [10]．

この課題を解決するために，Tsai らはライブラリ OS の graphene [11] を SGX 環境で利用できるように拡張した graphene-SGX を開発した [10]．graphene-SGX はライブラリ OS で動作させるプロセスを Enclave 内で実行することで，ユーザが実行したプロセスを保護することができる．graphene-SGX のアーキテクチャと Enclave 内での動的ロードのプロセスを以下に示す．

(1) Enclave を生成する SGX ドライバを呼び出すため，untrusted Platform Adaption Layer(pal-sgx) に入る．
(2) 制御ライブラリ，実行バイナリ，マニフェストを読み込み，Enclave の初期化を行う．制御ライブラリとは，Enclave の境界に位置して様々なアクセス制御を行うライブラリである．また，マニフェストとはユーザが予め記述した設定ファイルのことで，Enclave への読み込みを許可するファイルやライブラリの指定を行う．
(3) 制御ライブラリが Linux ライブラリ OS と標準 C ライブラリを読み込む．
(4) マニフェストで指定した追加ライブラリを読み込む．

Vol.2019-MPS-123 No.30
Vol.2019-BIO-58 No.30
2019/6/18

ライブラリは制御ライブラリによって，予めマニフェストに登録された SHA-256 ハッシュ値と改竄がないか検証される．

提案システムでは，Graphene-SGX を利用し，ABE ライブラリ等を読み込ませることで実装した．Graphene-SGX は github で公開されている [12]．

3. 関連研究

ABE に関する研究

ABE は，計算時間が復号条件数に線形であることが知られており，その対策が研究されてきた．Li ら [13]，Zhang ら [14] は鍵発行や復号化で発生する膨大な計算を外部に委託することで，ユーザ側の計算量を削減し，計算時間を短縮するアウトソース属性ベース暗号 (Outsourced ABE: OABE) を提案している．また，ABE はその特性を活かした様々な応用研究がある．Qian ら [15] は ABE を生涯型電子カルテ（Personal Health Record: PHR）に利用する手法を提案している．個人情報を多く含む PHR を ABE で暗号化することで PHR システムのプライバシー問題を軽減し，さらに，鍵生成局を複数にして鍵生成局が秘密鍵の属性を把握できなくする工夫を施している．[15] はデータの保管方法に関する研究であり，PHR を利用してデータ分析を行うことは想定されていない．

Intel SGX に関する研究

Intel SGX を暗号化手法の汎用性向上に応用した手法が存在する．Contiu ら [16] は ID ベース放送型暗号と Intel SGX を組み合わせた手法を提案した．ID ベース放送型暗号の鍵生成を Enclave 上で実行することで，サーバーは鍵に関する情報を知り得ない．Fisch ら [17] は Intel SGX を用いた関数型暗号フレームワークを提案している．この手法では，Remote Attestation を応用することで端末操作による実行を可能にしている．また，AES-NI を利用してサイドチャネル攻撃への対策をしている．Chen ら [18] は Intel SGX を用いた川崎病に関する家族間の対立遺伝子の分析フレームワークを提案している．

PPDM に関する研究

ABE や Intel SGX を使用せずに，PPDM を行う研究がされている．Aono ら [19] は加法準同型暗号を用いたロジスティック回帰手法を提案している．この手法では，準同型暗号を実数演算できるように拡張しているが，計算の中間結果が漏れてしまい，データを予測されてしまう危険がある．また，暗号化したまま演算を行うため，実行時間が従来より極端に遅くなってしまう．google 社 [20] や Apple 社 [21] はデータにノイズを加え，ノイズ付加データを収集，分析するローカル型差分プライバシーを利用した手法を提案している．[20,21] は暗号化処理がないため，高速で動作するが，データにノイズを付加するため，データ数が少ないと分析精度が落ちてしまう．

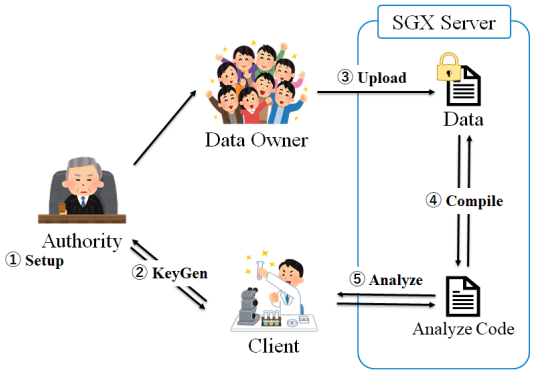

図 2　提案システムの概要
Fig. 2　Overview of proposed framework

4. 提案システム

4.1 提案システムの概要

提案システムの概要を図 **2** に示す．提案システムは，以下の 4 つのエンティティから成る．

- **Authority**: ABE 鍵生成局
- **SGX Server**: SGX 動作環境を持つサーバー
- **Client**: ユーザ用秘密鍵 SK をもつ者
- **Data Owner**: データ提供をする者

4.2 各プロトコル詳細

Setup$(\tau) \rightarrow PK, MSK$　セキュリティパラメータ τ を入力として，公開鍵 PK，マスター秘密鍵 MSK を出力する．実行は Authority が行い，PK を公開する．MSK は非公開にし，厳重に管理する．

KeyGen$(PK, MSK, \gamma) \rightarrow SK$　Authority は公開鍵 PK，マスター秘密鍵 MSK，ユーザの属性集合 γ を入力として，ユーザ用秘密鍵 SK を出力する．Client は Authority に SK を要求し，Authority は適切な Client の属性集合 γ を付与した SK を返す．この時，Authority は Client が悪意あるユーザでないか判別できるとする．

Upload(CT)　Data Owner が SGX Server に暗号化データ CT を提供する．Data Owner は自身で適切なアクセス構造 T を設定して ABE 暗号化（**Encrypt**$(PK, M, T) \rightarrow CT$）を行い，$CT$ を SGX Server に送る．

compile$() \rightarrow manifest$　SGX Server はマニフェスト $manifest$ を作成し，ファイルや共有ライブラリを読み込む．Data Owner が Upload した暗号化データはマニフェストにそのファイル名を記述し，再度コンパイルを行わないと Enclave 内部に取り込むことができない．そのため，$manifest$ を再作成する処理が必要となる．

Analyze$(SK, Fname) \rightarrow result$　ユーザ用秘密鍵 SK，暗号文ファイル名 $Fname$ を入力として，分析結果 $result$ を出力する．Client は SK と $Fname$ を SGX Server に送る．SGX Server はこれらを受け取り，Enclave 内でデー

タを復号化（**Decrypt**$(SK, CT) \rightarrow M$）する．復号化はユーザの属性集合 γ が T を満たす場合のみ実行され，満たさない場合は以降の処理は動作しない．SGX Server は，復号化したデータ M を用いてデータ分析を行い，分析結果を暗号化し（$result$），Client に返す．通信は Remote Attestation によって行われ，SK や $Fname$，$result$ が外部や SGX Server に漏れることはない．

5. 実験

5.1 実験環境

提案システムの試験実装と実験を行った．実装は Open-ABE [6]，Intel（R） SGX SDK for Linux 2.5 [22]，Intel（R）SGX 2.5 Linux Driver [23]，graphene-SGX [12] を用いて C++で行った．実験環境として Ubuntu 16.04.6 LTS，Intel Core i7-7700K CPU @ 4.20GHz，メモリ 16GB のマシンを用いた．実装したコードは `https://github.com/cBioLab/graphene_ABE` で公開している．

分析手法にはロジスティック回帰を利用し，回帰回数: 100 回，学習率: 0.01，正則化項 λ: 1 で固定した．

5.2 実験内容と結果

動作，精度確認実験

この実験では，提案システムの動作検証，及び分析精度を計測する．データセットは UCI で公開されている Breast Cancer Wisconsin (Diagnostic) Data Set [24] を利用した．このデータセットは，乳癌の診断結果 32 項目について 569 人分含んでおり，前処理として識別 ID の除去や標準化を行った．この UCI 乳癌データを訓練データ 8 割，テストデータ 2 割に分割し，その精度と AUC を提案手法，python で算出し，分析精度の変化を確認する．精度算出時の環境を統一するため，回帰変数 θ の初期値を，$\theta = 0$ で固定した．この実験から，提案システムは従来手法から精度を落とさず分析が可能であることが確認された（データ記載せず）．

実行速度測定実験

この実験では，提案システムの実行時間を測定する．データセットは IDASH PRIVACY & SECURITY WORKSHOP 2017, Task3 [25] で使用されたサンプルデータを利用した．このデータセットは一塩基多型を含む癌患者 1579 人分について 104 項目含まれている．この実験は提案システムの実用性を検証する実験であるため，データの分割や精度算出は行わず，データを全て分析に利用して実行速度を測定した．実験結果を**図 3** に示す．

5.3 考察

関連手法である完全準同型暗号を用いたロジスティック回帰分析では，9 次元 ×576 人のデータに対して回帰 100 回の演算に約 150 時間かかる（著者卒業論文より．データ

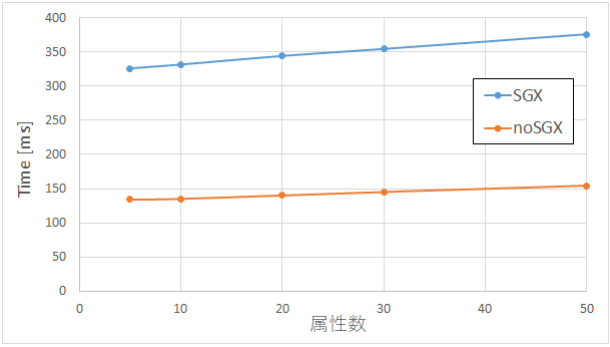

図 3 IDASH データにおける属性数と実行時間の関係
Fig. 3 Experimental result using IDASH data

記載せず）．そのため，図 3 から提案システムは提案システムが秘密計算技術よりも実行時間が短く，実用性が高いデータ分析プラットフォームであることが確認できる．また，提案システムが生命情報解析への応用が可能であることが示された．

一方で，実験結果から提案システムの課題が判明した．図 3 では，SGX 利用時と SGX 未使用時で実行時間に差異が見られたが，これは Enclave 使用時のメモリアクセスによるオーバーヘッドであると考えられる．Enclave のメモリサイズは，設定による消費を除いて最大で 96MB である．Enclave のメモリサイズが 96MB を超えると，不揮発性メモリへのアクセスが必要になり，オーバヘッドが発生する．提案システムではメモリ使用量を少なくする処理を施していないため，この対策は今後の課題である．また，秘密鍵 SK の取り扱いについても改善が必要である．提案システムでは，データ分析時に Client が秘密鍵 SK を SGX Server に送る処理が必要である．Remote Attestation を用いた暗号化通信を使用しているものの，この処理にはリスクが伴うため，Client が SK を送らずに分析する手法を検討する必要がある．さらに，クラウド上には多数のデータが存在することが想定されるため，ポリシーを満たす全てのデータを利用して分析ができるように改良すると，分析精度がさらに向上すると予想される．

6. 結論

本研究では，ABE と Intel SGX を用いてクラウド上のデータを秘匿したまま分析するシステムを提案した．また，提案システムを試験実装し，実用的な実行時間で収まることを実験により示した．提案システムと同様の目的を達成する秘密計算技術や差分プライバシーは，実行時間や精度に課題が残っていたが，提案システムでは，Enclave を用いてデータを秘匿したまま分析することで，分析精度を落とすことなく，秘密計算技術よりも大幅な計算量削減を実現した．Intel SGX の動作には，対応する CPU が必要であるが，提案システムでは Client が SGX 動作環境を用意する必要がない．さらに，ABE による暗号化したデータの

柔軟なアクセスにより，クラウドサービス普及に伴う不正アクセス発生時の課題やアクセス制御の課題を解決した．本研究により，個人情報を利用したデータ分析の更なる拡大が期待できる一方で，幾つか改善の余地があるため，引き続き研究に取り組む所存である．

参考文献

[1] Sahai, A. and Waters, B.: Fuzzy identity-based encryption, *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, pp. 457–473 (2005).
[2] Intel Corporation: Intel(R) Software Guard Extensions Tutorial Series: Part 1, Intel(R) SGX Foundation, Intel Corporation (online), available from ⟨https://software.intel.com/en-us/articles/intel-software-guard-extensions-tutorial-part-1-foundation⟩ (accessed 2019-04-06).
[3] Goyal, V., Pandey, O., Sahai, A. and Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data, *Proceedings of the 13th ACM conference on Computer and communications security*, ACM, pp. 89–98 (2006).
[4] Bethencourt, J., Sahai, A. and Waters, B.: Ciphertext-policy attribute-based encryption, *2007 IEEE symposium on security and privacy*, IEEE, pp. 321–334 (2007).
[5] 光成滋生：クラウドを支えるこれからの暗号技術，秀和システム (2015).
[6] Zeutro LLC: The OpenABE library, Zeutro LLC (online), available from ⟨https://github.com/zeutro/openabe⟩ (accessed 2019-04-02).
[7] Chen, G., Chen, S., Xiao, Y., Zhang, Y., Lin, Z. and Lai, T. H.: Sgxpectre attacks: Stealing intel secrets from sgx enclaves via speculative execution, *arXiv preprint arXiv:1802.09085* (2018).
[8] Van Bulck, J., Minkin, M., Weisse, O., Genkin, D., Kasikci, B., Piessens, F., Silberstein, M., Wenisch, T. F., Yarom, Y. and Strackx, R.: Foreshadow: Extracting the keys to the intel SGX kingdom with transient out-of-order execution, *27th USENIX Security Symposium*, pp. 991–1008 (2018).
[9] John M.: Code Sample: Intel Software Guard Extensions Remote Attestation End-to-End Example, Intel Corporation (online), available from ⟨https://software.intel.com/en-us/articles/code-sample-intel-software-guard-extensions-remote-attestation-end-to-end-example⟩ (accessed 2019-04-19).
[10] Tsai, C.-C., Porter, D. E. and Vij, M.: Graphene-SGX: A Practical Library OS for Unmodified Applications on SGX, *2017 USENIX Annual Technical Conference*, pp. 645–658 (2017).
[11] Tsai, C.-C., Arora, K. S., Bandi, N., Jain, B., Jannen, W., John, J., Kalodner, H. A., Kulkarni, V., Oliveira, D. and Porter, D. E.: Cooperation and security isolation of library OSes for multi-process applications, *Proceedings of the Ninth European Conference on Computer Systems*, ACM, pp. 1–14 (2014).
[12] OSCAR Lab: Graphene-SGX Library OS, OSCAR Lab (online), available from ⟨https://github.com/oscarlab/graphene⟩ (accessed 2019-04-02).
[13] Li, J., Chen, X., Li, J., Jia, C., Ma, J. and Lou, W.: Fine-grained access control system based on outsourced attribute-based encryption, *European Symposium on Research in Computer Security*, Springer, pp. 592–609 (2013).
[14] Zhang, R., Ma, H. and Lu, Y.: Fine-grained access control system based on fully outsourced attribute-based encryption, *Journal of Systems and Software*, pp. 344–353 (2017).
[15] Qian, H., Li, J., Zhang, Y. and Han, J.: Privacy-preserving personal health record using multi-authority attribute-based encryption with revocation, *International Journal of Information Security*, pp. 487–497 (2015).
[16] Contiu, S., Pires, R., Vaucher, S., Pasin, M., Felber, P. and Réveillère, L.: IBBE-SGX: Cryptographic Group Access Control using Trusted Execution Environments, *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, IEEE, pp. 207–218 (2018).
[17] Fisch, B., Vinayagamurthy, D., Boneh, D. and Gorbunov, S.: Iron: functional encryption using Intel SGX, *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ACM, pp. 765–782 (2017).
[18] Chen, F., Wang, S., Jiang, X., Ding, S., Lu, Y., Kim, J., Sahinalp, S. C., Shimizu, C., Burns, J. C., Wright, V. J. et al.: Princess: Privacy-protecting rare disease international network collaboration via encryption through software guard extensions, *Bioinformatics*, pp. 871–878 (2016).
[19] Aono, Y., Hayashi, T., Phong, L. T. and Wang, L.: Privacy-preserving logistic regression with distributed data sources via homomorphic encryption, *IEICE TRANSACTIONS on Information and Systems*, pp. 2079–2089 (2016).
[20] Erlingsson, Ú., Pihur, V. and Korolova, A.: Rappor: Randomized aggregatable privacy-preserving ordinal response, *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, ACM, pp. 1054–1067 (2014).
[21] Differential Privacy Team: Learning with Privacy at Scale, Apple Inc. (online), available from ⟨https://machinelearning.apple.com/docs/learning-with-privacy-at-scale/appledifferentialprivacysystem.pdf⟩ (accessed 2019-04-06).
[22] Intel Corporation: Intel(R) Software Guard Extensions for Linux OS, Intel Corporation (online), available from ⟨https://github.com/intel/linux-sgx⟩ (accessed 2019-04-03).
[23] Intel Corporation: Intel(R) Linux-sgx-driver, Intel Corporation (online), available from ⟨https://github.com/intel/linux-sgx-driver⟩ (accessed 2019-04-03).
[24] Kaggle Inc.: Breast Cancer Wisconsin (Diagnostic) Data Set, Kaggle Inc. (online), available from ⟨https://www.kaggle.com/uciml/breast-cancer-wisconsin-data⟩ (accessed 2019-04-02).
[25] IDASH team: IDASH PRIVACY SECURITY WORKSHOP 2017, the Department of Biomedical Informatics at UCSD and School of Informatics and Computing at Indiana University (online), available from ⟨http://www.humangenomeprivacy.org/2017/competition-tasks.html⟩ (accessed 2019-04-27).