Data SandboxによるMulti-Party Confidential Computing実現方式の提案

石倉 禅[1,a)]　柏木 啓一郎[1,2]　神谷 弘樹[1]　武藤 健一郎[1]　馬越 健治[1]　泉 雅巳[3]

概要：TEE(Trusted Execution Environment) を用いた安全な処理技術が発展し，近年では使用中のデータを保護するため，Confidential Computing の開発が活発に進められている．これまで奥田らは，Intel SGX を用い，データとプログラムを秘匿しつつ実行結果を得られ，且つ，実行毎にデータとプログラムを，実行基盤上における TEE の外部領域に格納する必要がない Confidential Program Execution の実現方式を提案し，この方式の安全性を形式検証で評価した．本研究では，先行手法の発展として AMD SEV-SNP を用い，さらにストレージの暗号化やネットワーク閉塞の要塞化などを施すことにより，データとプログラムを秘匿しながら実行結果を得つつも，悪意のあるプログラムが TEE の外部にデータを持ち出すことができないようにすることで，Multi-Party Confidential Computing を実現しながら，汎用性のある TCB(Trusted Computing Base) をセキュアに提供する方式を提案する．

Proposition of Multi-Party Confidential Computing by Data Sandbox

Zen Ishikura[1,a)]　Keiichiro Kashiwagi[1,2]　Koki Mitani[1]　Kenichiro Muto[1]　Kenji Umakoshi[1]
Masami Izumi[3]

Abstract: The secure processing technology using TEE (Trusted Execution Environment) has been developed, and the development of confidential computing is actively advanced recently to protect the data in use. So far, Okuda et al. proposed an implementation method of Confidential Program Execution using Intel SGX, in which execution results can be obtained while data and program are concealed, and data and program need not be stored in the external region of TEE on the execution infrastructure for every execution, and evaluated the security of this method by formal verification. In this study, we use AMD SEV-SNP as a development of the previous method, and propose a method to securely provide a general-purpose TCB (Trusted Computing Base) while realizing Multi-Party Confidential Computing by preventing malicious programs from taking data out of TEE while obtaining execution results while hiding data and programs by encrypting storage and fortifying network blockage.

Keywords: Data Sandbox, Multi-Party Confidential Computing, TEE(Trusted Execution Environment), AMD SEV-SNP, Remote Attestation, TCB(Trusted Computing Base)

1. はじめに

近年, クラウドコンピューティングにおいて, Confidential Computing の活用が注目されている．Confidential Computing は，ハードウェアベースの TEE(Trusted Execution Environment) を用いて処理中のデータを保護する技術である．その要素技術としては，信頼できる領域を提供する TEE に加えて，それらが適切に動作していることを遠隔で検証できる Remote Attestation などが用いられる．また，Confidential Computing を活用することで，クラウドサービスの利用者は，クラウド基盤上で処理中の自身の機密データなどをクラウド事業者から秘匿することができる．新たな市場の創出を狙うハードウェアベンダ，

[1] NTT 社会情報研究所
NTT Social Informatics Laboratories
[2] NTT ソフトウェアイノベーションセンタ
NTT Software Innovation Center
[3] 日本電信電話株式会社 技術企画部門
[a)] zen.ishikura@ntt.com

クラウド事業者，ソフトウェアベンダなどが中心となり Confidential Computing Consortium を発足し，Microsoft などのクラウド事業者がクラウド基盤上で関連サービスを開始するなど，事業での活用に向けた取り組みが加速しつつある [10][11]．現時点でも様々な企業が Confidential Computing を活用したサービスを開始しつつあるが，それらの多くはクラウドサービスの利用者とクラウド事業者が 1 対 1 の関係で利用するユースケースを基本としている．しかし，秘匿性の高いデータの活用が進むにつれ，データやそれを処理するプログラムをすべて 1 社で用意することは難しくなり，今後は複数のエンティティが秘匿性の高いデータやプログラムをお互いに提供し活用するユースケースが増えることが予想される．例えば，データを提供するエンティティ（データ提供者）が複数存在する場合，データ提供者同士が互いのデータを秘匿できることが求められる．また，クラウド基盤上で動作するデータ分析プログラムを作成し提供するエンティティ（プログラム提供者）が，データ提供者と異なる場合，クラウド基盤の運用者に対してデータやプログラムを秘匿することに加え，データをプログラム提供者に対して秘匿できること，および，プログラムをデータ提供者に対して秘匿することが求められる．とくに後者については，AI による予測モデルを複数企業のデータをもとに更新するユースケースなどにおいて，顕著なニーズがあると考えられる．

1.1 前提とする技術

本節では，本研究において前提となる技術について述べる．

1.1.1 TEE

TEE とは，CPU に具備された機能を用いて，メモリ上に保持するデータが常に暗号化され外部から読み取ることができない状態となるように計算処理を実行することができるソフトウェア実行環境のことである．TEE には Remote Attestation に基づき利用者との間で安全な通信経路を確立するための機能が具備されている．認証の保護，暗号化，モバイルデバイス管理，決済，DRM 等に用いられている．

1.1.2 Remote Attestation

Remote Attestation とは，利用者がネットワーク経由で利用するサーバーサイドで TEE を構成する場合に，当該 TEE が正しく構成されていることを遠隔で確認するための機能である．TEE が正しく構成されていることを利用者が確認するための情報としては，TEE を構成する CPU を提供するハードウェアベンダの鍵を用いた署名付きのアテステーションレポートなどが，利用者に提供される．なお，Remote Attestation のプロシージャは，RFC 9334[5] にて標準化されており，アーキテクチャ，および，さまざまな既存および新興のリモートアテステーション方式に適用できる一般的な用語セットを定義している．

1.1.3 AMD SEV-SNP

AMD SEV-SNP[2] は，仮想マシン（VM）の単位を TEE として構成することができるハードウェア実装の 1 つである．仮想マシン単位で隔離が可能なメモリ暗号化機能，および，Remote Attestation に必要な構成情報の収集および署名に関する機能などを提供する．サーバーサイドで構成する TEE として利用することができる．Intel SGX[3] もまた，サーバーサイドで構成する TEE であるが，TEE の構成単位はプロセスである．

1.1.4 Confidential Computing

Confidential Computing[4] は，TEE を用いて処理中のデータを保護する技術群である．TEE により処理を実行中のプログラムやデータが許可なく閲覧・変更されることを防ぐことができるため，秘匿性の高いデータの処理への活用が見込まれている．

1.1.5 Multi-Party Confidential Computing

複数のデータ提供者やプログラム提供者など，複数のエンティティが秘匿性の高いデータやプログラムをお互いに提供し活用するユースケースに対応が可能な Confidential Computing の活用形態を，本稿では Multi-Party Confidential Computing と定義した．

2. 先行手法

本章では，奥田らの手法 [1] について述べる．奥田らは，Intel SGX の TEE を活用して，事前に信頼関係のないパートナー企業間で，互いの有するプログラムとデータを秘匿したまま，プログラムの実行結果のみを享受できるプログラム秘匿実行の方式を提案した．この方式においては，データおよびプログラムが暗号化された状態で TEE 内に格納することができる手法が導入されている．これにより，各エンティティが秘匿したいデータやプログラムが，実行基盤における TEE の外側にて平文で取り扱われる懸念がなくなり，クラウド事業者でさえも，データやプログラムを閲覧することができないという利点がある．具体的には，共通鍵暗号と公開鍵暗号を組み合わせ，安全に TEE 内にてデータやプログラムを展開するための鍵配送方式として Dan-Twist という手法を提案した．また，提案方式を対象とした形式手法による安全性評価を実施した．本稿の内容は，奥田らの手法を基礎とする．

3. 本研究での課題

本研究では，奥田らの手法を基にしながら，AMD SEV-SNP を活用して，Multi-Party Confidential Computing システムを構築するにあたって，以下 2 つの課題に取り組んだ．奥田らの手法は Enclave の単位を TEE として構成するものであるが，本研究では比較的複雑な仮想マシンの単位を TEE として構成することから，直面した課題である．

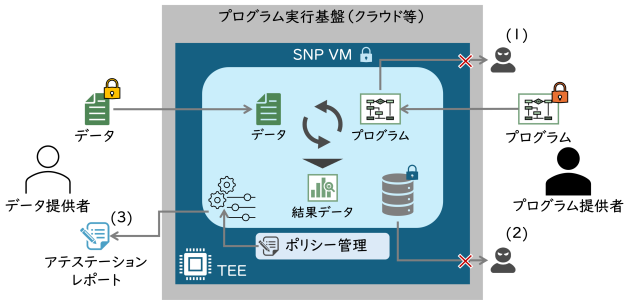

図 1　提案構成の概要

3.1　悪意あるプログラムによる外部へのデータ漏洩の防止

Multi-Party Confidential Computing において，正当なプログラム提供者が，悪意あるプログラムを TEE である仮想マシンに格納し実行した場合，仮想マシンに具備されているネットワーク通信機能を用いて，本来 TEE 内に留めて秘匿すべきデータ（データ提供者が提供したデータだけでなく，計算過程の中間データも含む）を TEE の外部に持ち出すことができてしまう．

3.2　データやプログラムを格納するディスク領域の拡張と保護

Multi-Party Confidential Computing を AI 学習などの領域で活用する場合，データやプログラムのサイズが大きくなることが予想される．しかし，TEE は物理メモリ領域を暗号化してデータを保護する仕組みであるため，TEE 内のプログラムが読み込み可能なデータ容量は，物理メモリの上限に制限されてしまう．一方で，仮想マシンはディスク領域を活用してデータ処理を行うことが可能な仕組みであるが，ディスク領域は TEE の機能だけでは保護されない．

4.　提案手法

本章では，前章で述べた課題を解決するために，悪意あるプログラムが外部にデータを漏洩することを防止するための手法，および，TEE が読み込み可能なデータ容量の制限を緩和するための手法について述べる．提案手法の概要について**図 1** に示す．

4.1　要塞化によるデータの外部への漏洩の防止

本節では，悪意あるプログラムが外部にデータを持ち出すことを防止する（図 1 中 (1)）ため，TEE である仮想マシンにおける，ネットワーク，および，コマンド実行制御に関する保護策について述べる．

4.1.1　ネットワークを介した外部漏洩への対策

悪意あるプログラムが実行された場合に，関係するエンティティが事前に合意していない通信先にネットワーク通信を行いデータを持ち出すことができないようにする対策として，仮想マシンに設定を行うことで機能制限を設けた．具体的には，仮想マシンの Kernel config を変更することにより，loopback アドレス以外のネットワークインタフェースが利用できないように無効化した．

4.1.2　コマンド実行制御

前述のネットワーク制御を施した場合，仮想マシンは外部とネットワーク通信を行うことができなくなる．しかし，Multi-Party Confidential Computing を実現するためには，ネットワーク通信以外のなんらかの方法で，仮想マシンの外部との間でデータやプログラムの入出力，および Remote Attestation などを行う必要がある．例えば，OSS の Kata Containers[8] では，仮想マシン内で動作する Kata Agent と呼ばれるソフトウェアが仮想マシン外部からの制御を受ける機能を提供している．この Kata Agent に対する外部からの制御をあらかじめ与えた Kata Agent Policy[9] により制限することで，Multi-Party Confidential Computing の制御に必要な情報の入出力のみを仮想マシンに対し許可することが可能となる．

4.2　ディスク保護による読み込み可能なデータ容量の拡大

TEE が読み込み可能なデータ容量の制限を緩和するためには，物理メモリ領域だけでなく，仮想マシンが利用可能なディスク領域を活用してデータ処理を行えるようにすることが有効である．しかし，ディスク領域は TEE の機能だけでは保護されないことが課題である．そこで，本節では，ディスク上に配置したデータが外部に漏洩することを防止する（図 1 中 (2)）ため，TEE である仮想マシンにおける，ディスクに関する保護策について述べる．

4.2.1　ディスクを介した外部漏洩への対策

ディスク上のデータを保護するためには，ディスク上のデータを入出力のたびに暗号化・復号する方式が有効である．例えば，LUKS (Linux Unified Key Setup-on-disk-format) [7] は，ブロックデバイスを暗号化でき，暗号化したデバイスの管理を簡素化するツールセットを提供している．TEE 内の仮想マシンにおいて暗号化ディスクをマウントするためには，仮想マシンが暗号化ディスクの暗号鍵を保持し，かつ，クラウド事業者がこの鍵を窃取できないようにする必要がある．このため，TEE 内にて暗号鍵を生成し，これを暗号化ディスクのマウントに利用する．

4.3　各保護策が有効になっていることの検証

本節では，保護策が有効になっていることを参加エンティティが検証（図 1 中 (3)）する仕組みについて述べる．前節の対策が施された仮想マシンについては，ソースコードを含めて公開し，データ提供者やプログラム提供者といった参加するエンティティが検証できるようにした．これにより，各エンティティは，AMD SEV-SNP の Remote Attestation で用いられるアテステーションレポートの中の Measurement の値を確認することにより，起動時に読

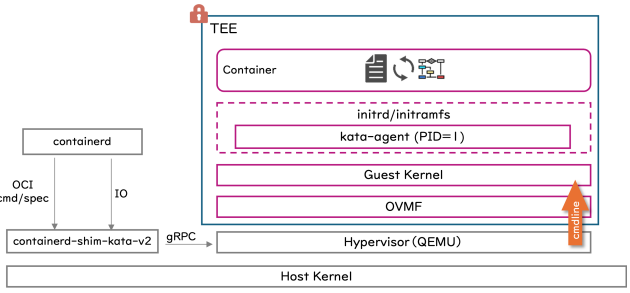

図 2 Measurement にて検証可能な領域

み込まれるファームウェアや OS が改ざんされていないことを検証することが可能となる（**図 2**）[6]．また，アテステーションレポート中に REPORT_DATA という，ユーザが TEE において任意で指定できるパラメータフィールドが設けられている．AMD SEV-SNP のアテステーションレポートには AMD による署名が付与されており，これを検証することにより，ユーザーはアテステーションレポートが改ざんされていないことを確認することができる．例えば，TEE 内で作成した暗号鍵のハッシュ値を加えることにより，遠隔のユーザーは，TEE で作成された暗号鍵が改ざんされていないことを検証することが想定されている．

5. 提案手法を踏まえた処理実行フロー

本章では，提案手法を踏まえた Multi-Party Confidential Computing 処理の実行フローについて述べる．以降，プログラムを実行する基盤システムを，MPCC システムと呼称し，MPCC システムの TEE で保護されたプログラムを実行する VM 領域を，DSB(Data Sandbox) と呼称する．

5.1 DSB の起動

参加エンティティの代表者が，MPCC システムに対して，DSB を起動するよう要求し，MPCC システムは，VM イメージを読み込み，DSB を起動する．
この際，DSB は，4.1 節の要塞化が施された状態で起動する．

5.2 データおよびプログラムの登録

データ提供者は，データを暗号化するための暗号鍵を作成し，この鍵でデータを暗号化し，MPCC システムの暗号化ディスク領域に登録する．同様の操作で，プログラム提供者は，プログラムを暗号化するための暗号鍵を作成し，この鍵でプログラムを暗号化し，MPCC システムの暗号化ディスク領域に登録する．また，プログラム実行結果データを受け取るエンティティ（本稿ではデータ提供者を想定）は，プログラム実行プログラム実行結果データを暗号化するための暗号鍵を作成する．

5.3 処理実行ポリシーの提案および合意

データ提供者，プログラム提供者が，どのようなプログラムでデータ分析を行うかについてのポリシーを確認し，合意する．データ提供者，および，プログラム提供者は，ポリシー提案者による提案を受け確認し，合意する．なお，ポリシー提案者は，任意のエンティティであってよい．

5.4 Remote Attestation

ポリシー合意を契機に，DSB 内で，暗号鍵（TEE 鍵）を生成し，データ提供者，および，プログラム提供者に，TEE 鍵を，DSB が作成したアテステーションレポートと共に送付する．
この際，4.3 節のアテステーションレポート検証を用いて，データ提供者，および，プログラム提供者は，VM 上の OVMF，Kernel, Initrd が改ざんされていないかを確認し，さらに，アテステーションレポートに DSB が作成した TEE 鍵のハッシュ値を加えることにより，TEE 鍵が改ざんされていないことを確認する．もし仮に検証が失敗した場合は，データ提供者もしくはプログラム提供者により，処理を中断することが可能である．

5.5 鍵登録

アテステーションレポートの検証が検証した場合，データ提供者は，データを復号するための復号鍵，および，プログラム実行結果データを暗号化するための暗号鍵を，TEE 鍵で暗号化し，DSB に送付することにより，DSB 内に鍵登録をする．同様の操作で，プログラム提供者は，プログラムを復号するための復号鍵を，TEE 鍵で暗号化し，DSB に送付する．

5.6 復号処理

DSB は，TEE 鍵を用いて，データを復号するための鍵，および，プログラムを複合するための鍵を復号し，MPCC システムの暗号化ディスク領域に登録されているデータおよびプログラムを TEE 上で復号する．復号処理は TEE 上で行われるため，メモリ暗号化等により，TEE 外部の全てのエンティティはデータやプログラムにアクセス不可である．
この際，4.2 節のディスク制御により，暗号化ディスクをマウントするためには TEE 内で作成した暗号鍵を用いる必要があるため，DSB が動作する TEE のみが暗号化ディスクをマウントすることが可能である．また，暗号化ディスクを登録されているデータおよびプログラムを復号するための複合鍵は，データ提供者，および，プログラム提供者が，DSB に対して，TEE 鍵で暗号化したものであり，TEE 内でのみ復号が可能となる．

5.7 プログラム実行

復号したデータおよびプログラムを，DSB 内で実行する．

この際，4.1 節の要塞化により，プログラムが外部にデータを漏洩することはできない．4.1.1 節のネットワーク制御により，DSB は外部と通信することができないため，MPCC システム内において，データおよびプログラムが外部に漏洩することはない．

5.8 プログラム実行結果データの取得

DSB は，プログラム実行結果として得られたプログラム実行結果データを，プログラム実行結果データを暗号化するための暗号鍵で暗号化し，データ提供者は，暗号化されたプログラム実行結果データを DSB から取得する．データ提供者は，暗号化されたプログラム実行結果データを，プログラム実行結果データを復号するための復号鍵を用いて復号することで，プログラム実行結果データを取得する．この際，DSB は，4.1.1 節のネットワーク制御，および，4.1.2 により，エンティティが合意したポリシーによって許可された通信先にのみ，プログラム実行結果データを送信することが可能である．

6. まとめ

本稿では，AMD SEV-SNP を用いた Multi-Party Confidential Computing の実現方式を提案した．提案方式により，TCB を拡張した場合においても，悪意あるプログラムが外部にデータを持ち出すことを防止するための手法，および，TCB の拡張に対応可能なデータ容量の制約を解決するための手法について述べた．今後は，形式検証等を用いた理論的な観点を含む，精緻な安全性評価を拡充し，Multi-Party Confidential Computing における，多種多様なユースケースに対する適用性について検討を行い，研究開発を進める予定である．

参考文献

[1] 奥田哲矢，中林美郷，荒井研一，菊池亮，千田浩司：Confidential Program Execution の提案と安全性評価，SPT(2021).

[2] AMD Inc.: AMD SEV-SNP: Strengthening VM Isolation with Integrity Protection and More, (2020)

[3] Intel Corporation: Intel® Software Guard Extensions (Intel® SGX) Developer Guide, (2018)

[4] Confidential Computing Consortium: A Technical Analysis of Confidential Computing v1.3, (2022)

[5] Internet Engineering Task Force (IETF): RFC 9334 Remote ATtestation procedureS (RATS) Architecture, (2023)

[6] Jeremy Powell: AMD SEV-SNP Attestation: Establishing Trust in Guests, Linux Security Summit Europe(2022).

[7] Joachim Metz: LUKS Disk Encryption format specification, (2020)

[8] Open Infra Foundation: Kata Containers Architecture, https://github.com/kata-containers/kata-containers/tree/main/docs/design/architecture

[9] Open Infra Foundation: Kata Agent Policy, https://github.com/kata-containers/kata-containers/blob/main/docs/how-to/how-to-use-the-kata-agent-policy.md

[10] Microsoft Corporation: Azure confidential computing partners, https://learn.microsoft.com/en-us/azure/confidential-computing/partner-pages/partner-pages-index

[11] Howard, Heidi, Alder, Fritz, Ashton, Edward, Chamayou, Amaury, Clebsch, Sylvan, Costa, Manuel, Delignat-Lavaud, Antoine, Fournet, Cédric, Jeffery, Andrew, Kerner, Matthew, Kounelis, Fotios, Kuppe, Markus A., Maffre, Julien, Russinovich, Mark, Wintersteiger, Christoph M.: Confidential Consortium Framework: Secure Multiparty Applications with Confidentiality, Integrity, and High Availability, Proceedings of the VLDB Endowment, Volume 17, Issue 2, pp.225-240(2023)