九州工業大学 情報工学部 情報・通信工学科 2023 年度 卒業論文 概要

コース	ソフトウェアデザイン	指導教員	光來 健一
学生番号	202C1066	氏　　名	佐藤 太陽

論文題目	Arm TrustZone のワールド間での POSIX API を用いた協調実行

1　はじめに

近年，クラウドのアプリケーションをユーザの近くで実行することによりサービスの品質を向上させるエッジコンピューティングが普及してきている．エッジデバイスにおいては OS すら信頼できない場合があるが，CPU が提供する隔離実行環境 (TEE) を用いることでクラウドアプリケーションを安全に実行することができる．エッジデバイス向けの TEE である Arm TrustZone は 2 つのワールドを提供し，セキュアワールドで Trusted Application (TA) が動作し，ノーマルワールドで Client Application (CA) が動作する．セキュアワールドでクラウドアプリケーションを実行することが考えられるが，セキュアワールドは高い権限を持っているため，TEE での実行がシステム全体に影響を及ぼす恐れがある．また，TEE を利用するには専用の API を用いる必要があり，CA と TA の柔軟な協調は容易ではない．

本研究では，クラウドアプリケーションを TrustZone の 2 つのワールドに分割し，ワールド間で POSIX API を用いて協調実行を行うシステム TZmediator を提案する．

2　TZmediator

TZmediator は図 1 のようにクラウドアプリケーションを TrustZone の 2 つのワールドに分割し，保護する必要のない処理はノーマルワールドで CA として実行する．そして，セキュアワールドで実行する必要がある処理のみを TA として実行する．その際に，WebAssembly を用いることで安全に実行することができる [1]．この CA と TA は並列に実行され，ワールド間にまたがって POSIX API を用いて協調動作する．そのために，ノーマルワールド内に TA に対応するシャドウプロセスを作成し，CA と TA はシャドウプロセスを介して通信を行う．

セキュアワールドで動作する TA は専用ライブラリが提供する POSIX API を利用する．このライブラリは遠隔手続呼び出し (RPC) を用いてノーマルワールドのシャドウプロセスを呼び出す．その際に，API の種類と引数を渡し，実行結果を受け取る．シャドウプロセスは標準の C ライブラリを用いて指定された POSIX API を代理実行する．CA と同じ OS に対してシステムコールを発行することができるため，CA との間の通信の互換性は高い．

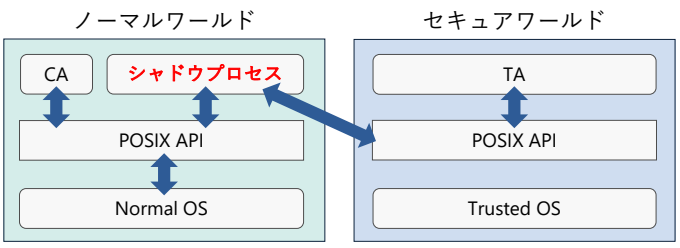

図 1. TZmediator のシステム構成

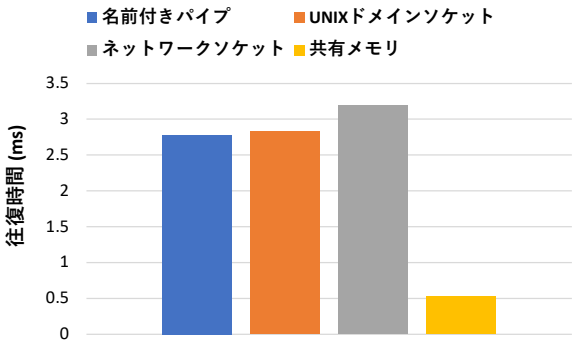

図 2. CA と TA 間の通信の往復時間

ノーマルワールドで動作する CA も TA の代理であるシャドウプロセスとの通信を行うために，標準の C ライブラリによって提供される POSIX API を利用する．それに加えて，CA に提供される専用ライブラリは TA を実行するための API を提供する．この API はスレッドを生成してから TA をロードし，専用 API を用いて TA の処理を呼び出す．サブスレッドを用いることにより，CA のメインスレッドは TA が実行されている間，並列に動作することができる．

3　実験

TZmediator を用いて CA と TA が双方向に通信できることを確認する実験を行った．この実験では，名前付きパイプ，UNIX ドメインソケット，ネットワークソケットを用いて CA と TA それぞれからデータを送信し，通信先から返送されたデータを受信した．その結果，正常に通信を行えることが確認できた．次に，256 バイトのデータを送受信する往復時間を測定した．比較として，CA と TA の間に確立されている共有メモリを用いて通信する時間も測定した．図 2 に示すように，TZmediator は共有メモリを用いた通信よりも 5.2〜6.0 倍の実行時間がかかることが分かった．TA とシャドウプロセス間の RPC において共有メモリを用いることによって高速化が可能であると考えられる．

4　まとめ

本研究では，クラウドアプリケーションを TrustZone の 2 つのワールドに分割し，ワールド間で POSIX API を用いた協調実行を可能にするシステム TZmediator を提案した．今後の課題は，他の通信機構に対応することや，WebAssembly を用いて安全に TA を実行できるようにすることである．

参考文献

[1] Ménétrey et al. WaTZ: A Trusted WebAssembly Runtime Environment with Remote Attestation for TrustZone. ICDCS, 2022.