Arm TrustZoneのワールド間における POSIX APIを用いた安全なプロセス協調

佐藤 太陽[1]　光来 健一[1]

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Trusted Execution Environment (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．クラウドアプリケーションをセキュアワールドに隔離して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，クラウドアプリケーションに脆弱性があるとシステム全体に影響が及ぶ恐れがある．そのため，TAとして実行する部分をできるだけ小さくして，CAと連携しながらクラウドアプリケーションを実行することが望ましい．しかし，CAとTAを連携させるには専用の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，さらに安全性を高める必要がある場合にはWebAssemblyを用いて実行する [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．

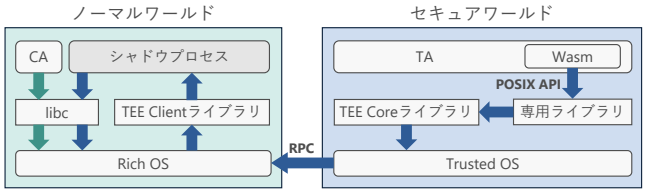

図1　TZmediatorのシステム構成

セキュアワールドで動作するTAはTZmediatorが提供する専用ライブラリ経由でPOSIX APIを利用する．このライブラリはTAに提供されるTEE Internal Core APIを用いて遠隔手続き呼び出し (RPC) を実行し，ノーマルワールドのシャドウプロセスを呼び出す．シャドウプロセスはCAに提供されるTEE Client APIを用いてRPCのリクエストを受け取り，標準Cライブラリを用いて指定されたPOSIX APIを代理実行する．これらのセマンティクスは通信相手のCAと同じであるため，POSIXに対する高い互換性を実現することができる．実行結果はRPC経由でセキュアワールドのTAに返送される．このように，専用APIを用いたワールド間通信は専用ライブラリとシャドウプロセスによってクラウドアプリケーションから隠蔽される．

ノーマルワールドで動作するCAは標準Cライブラリによって提供されるPOSIX APIを利用して，TAの代理として作成されるシャドウプロセスと通信を行う．さらに，シャドウプロセスがTAと通信を行うことによって，CAとTA間の通信を実現する．

3. 実験

TZmediatorを用いてCAとTA間で4KBのデータを

[1] 九州工業大学
Kyushu Institute of Technology

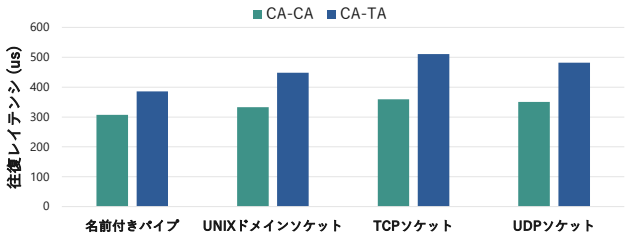

図 2 通信の往復レイテンシ

送受信する往復レイテンシを測定した．実験には Arm Cortex-A53 の CPU を搭載した Armadillo-X2 を用いた．通信に用いた POSIX API は名前付きパイプ，UNIX ドメインソケット，TCP ソケット，UDP ソケットの 4 つであった．比較として，ノーマルワールド内の CA 同士の往復レイテンシについても測定した．本実験では Wasm を用いずに TA を実行した． **図 2** に示すように，TZmediator は CA 同士でのデータの受け渡しよりも 1.26〜1.42 倍の実行時間がかかることが分かった．これはセキュアワールド内の TA がデータを読み書きする際に，ノーマルワールド内のシャドウプロセスを RPC で呼び出してデータの転送を行うためである．ネットワークソケットを用いた通信のオーバヘッドが，名前付きパイプや UNIX ドメインソケットを用いた通信よりも大きくなった原因については，現在調査中である．

4. まとめ

本研究では，クラウドアプリケーションを 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 を提案した．今後の課題は，名前付きパイプやソケット以外の通信に対応することである．

謝辞 本研究の一部は，JST，CREST，JPMJCR21M4 の支援を受けたものである．また，本研究成果の一部は，国立研究開発法人情報通信研究機構（NICT）の委託研究（JPJ012368C05501）により得られたものである．

参考文献

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