TEEを用いたProvenance AuditingのIoT機器への適用

竹村 太一[1,2,a)]　須崎 有康[1]　山本 嶺[2]

概要：近年，ステルス攻撃の数が増加しており，従来の検知システムでは検知できない状況にある．Provenance Auditing (PA) は，コンピュータに対する攻撃の影響を判断するための強力な手法である．PAでは，システムで発生したイベントのログからDAG（Directed Acyclic Graph）を構築し，管理者はそのDAGから攻撃の影響を判断することができる．しかし，PAはパーソナルコンピュータ用に設計されており，低消費電力のIoTデバイスには適合しない．
IoTデバイスのシステムコールログをTEE（Trusted Execution Environment）を用いて安全に収集し，リモートのPAサーバに安全に送信する方法を提案する．システムコールログは，カーネルからTEEに直接転送され，ユーザー空間には公開されない．カーネルはREEで動作するが，保護オプションによりTCB（Trusted Computing Base）として扱われるように強化する．提案手法のプロトタイプをRaspberry Pi3のArm TrustZoneに実装し，PAのフレームワークとしてSPADEを使用した．

キーワード：Arm TrustZone，Provenance Auditing

1. はじめに

IoT機器を対象とした初期のマルウェアであるMirai，BASHLITEは，DDoS（Distributed Denial of Service attack）やネットワークのスキャンを目的としていた．しかし，最新のマルウェアでは，IoT機器の破壊，暗号通貨の採掘，プロキシサービスの提供など幅広い機能が追加されている [1]．さらに，ファイルレスマルウェアやシェルコマンドを利用する攻撃といったステルス性の高い攻撃も増加しており，ブラックリストやIDS（Intrusion Detection System）を利用した従来の攻撃の検知手法では検知できない．これらの攻撃を検知するために，Provenance Auditing（PA） [2] に代表されるような調査手法を用いる必要がある．PAはOSレベルのログを収集し，システムで実行されるイベントの関係性をDAGとして構築する手法であり，攻撃の調査のために幅広く活用されている．しかし，端末資源等に制限があるIoT機器では，既存のPAをそのまま適用することができない [3]．本研究では，TEE（Trusted Execution Environment）を用いてIoT機器上で安全にログを収集しクラウド上のサーバに送信し，クラウド上のサーバでPAを行う手法を提案する．

[1] 国立研究開発法人 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology
[2] 電気通信大学大学院
The University of Electro-Communications
[a)] takemura@net.lab.uec.ac.jp

2. 背景

前述のPAは，システムで実行されたイベントを記録したシステムコールログを基に，実行されたイベントの因果関係をDAG（Directed Acyclic Graph）として表す．DAGのノードはプロセスやファイル，ソケットのイベントを表し，エッジはノード間の関係性を示す．管理者はDAGから攻撃の影響範囲の調査や攻撃者の侵入方法の特定を行う．また，PAはデスクトップパソコンで使われることを想定するため，OSカーネルは安全としている．そのため，攻撃者はLinux Auditへの攻撃や監査ログの削除や改竄は行えない．

TEEは，ハードウェアによって提供される隔離された実行環境であり，TEEの実現のために，CPU自体がOSから隔離した環境を用意する必要がある [4]．代表的なTEEとして，Arm TrustZone，Intel SGX，RISC-V Keystoneが挙げられる．本研究では，Arm TrustZoneを利用する．

3. 提案手法

図1に従来のPAを用いる従来手法と提案手法の動作概要を示す．なお，本図では灰色の濃淡によってセキュリティ強度を表現することとする．図1 (a) に示すように従来手法では，Linux Auditを用いてシステムで実行されたイベントを記録し，端末上でPAが行われる．まず，Linux Auditは，カーネル内で実行されるシステムコールに対して

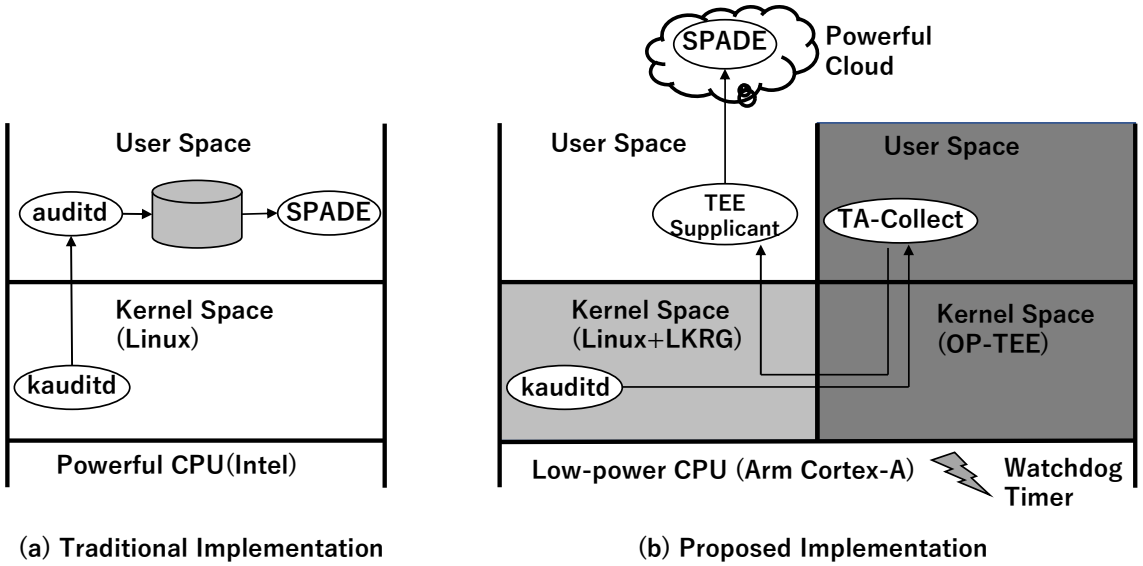

図 1 従来手法と提案手法の動作概要

Audit ルールに基づいて監査を行う．ここで Audit ルールとマッチしたシステムコールは，システムコールログとして生成される．カーネルスレッドである kauditd はシステムコールログを netlink を経由してユーザランドの auditd に送信され，auditd は受信したシステムコールログを監査ログとしてファイルに出力する．

ここで，PA を IoT 機器に適用するためには，次の二つの問題がある．一つ目は，十分な計算資源を有していない IoT 機器において，計算負荷の高い PA を端末上で行うことは困難であるということである．二つ目は，セキュリティが欠落している IoT 機器において，OS カーネルが安全であるという前提を用いることは困難であり，攻撃者が Linux Audit 自体への攻撃や監査ログの改竄を行う可能性があるということである．

図 1 (b) に提案手法の概要を示す．まず，一つ目の問題を解決するため，PA を IoT 機器の端末上ではなく，計算資源が豊富なクラウド上のサーバで行う．次に，二つ目の問題を解決するために，OS カーネルの堅牢化と隔離実行環境である TEE を用いて Linux Audit の保護を行う．OS カーネル内で生成されるシステムコールログはカーネルから直接 TEE に渡され，TEE から安全にクラウドサーバに送信する．ユーザランドを経由せずにシステムコールログをクラウドに送信するため，ユーザランドの攻撃から隔離されている．さらに，攻撃者が LA のログを削除や改変を行うと試みた場合，リモートからの Heartbeat と TEE に保護された Watchdog Timer(RO-IoT [5]) の連携によりその問題を検出し，問題が発生した際には IoT 機器のシステムリセットを行う．

4. 実装

本研究では，提案手法の有効性検証のため，実機実装を用いた動作実験を行う．本実装では，システムでの実行イベント記録のために Linux Audit，ログ収集とサーバへの送信に OP-TEE，送信されたログの解析に SPADE [2] を採用し，Raspberry Pi3 Model B（Arm Cortex-A53/4 コア/1.2GHz， RAM 1GB，ストレージ 16GB）上に実装した．

5. むすび

本研究では，TEE による Linux Audit の保護を行い，CPU 性能に制限がある IoT 機器に対して PA を適用する手法の提案を行った．これにより，提案手法を IoT システムに導入することで，ステルス性が高い攻撃を検出することが可能となり，インシデント発生時の管理者負担を低減したシステム構築が可能となると考えられる．

参考文献

[1] Omar Alrawi, Charles Lever, Kevin Valakuzhy, Kevin Snow, Fabian Monrose, and Manos Antonakakis. The Circle Of Life: A Large-Scale Study of The IoT Malware Lifecycle. In *30th USENIX Security Symposium (USENIX Security 21)*, 2021.

[2] Ashish Gehani and Dawood Tariq. SPADE: Support for Provenance Auditing in Distributed Environments. In *Proceedings of the 13th International Middleware Conference*, pp. 101–120. Springer, 2012.

[3] Anand Mudgerikar, Puneet Sharma, and Elisa Bertino. E-Spion: A System-Level Intrusion Detection System for IoT Devices. In *Asia Conference on Computer and Communications Security*, pp. 493–500. ACM, 2019.

[4] 須崎有康. Trusted execution environment の実装とそれを支える技術. 電子情報通信学会 基礎・境界ソサイエティ Fundamentals Review, Vol. 14, No. 2, pp. 107–117, 2020.

[5] Kuniyasu Suzaki, Akira Tsukamoto, Andy Green, and Mohammad Mannan. Reboot-Oriented IoT: Life Cycle Management in Trusted Execution Environment for Disposable IoT devices. In *Annual Computer Security Applications Conference*, pp. 428–441. ACM, 2020.