軽量ハイパーバイザにおける IoT 向け入出力記録機構の設計と実装

2019SE011 飯尾和史
指導教員：宮澤元

1 はじめに

Internet of Things(IoT) の普及により，様々なデバイスがインターネットから攻撃を受ける可能性にさらされている．IoT デバイスへの攻撃では，外部からの不正アクセスや Input/Output(I/O) 機器のデータ盗聴が行われる．そのため，I/O 機器への不正アクセスの検知や入出力の記録が IoT デバイスのセキュリティ向上において必要となる．

IoT デバイスにおけるセキュリティ問題の解決策の一つとして，入出力の監視がある．入出力の監視によって，不正アクセスの検知やアクセスの遮断といったセキュリティ対策を取ることができる．また，マルウェア感染などの危険性を考慮すると，アプリケーションからは透過的な入出力監視を行うことが望ましい．

透過的に入出力監視を行う方法として，ハイパーバイザの準仮想化ドライバを用いる方法や低レベル I/O を監視する方法がある．いずれの方法も，仮想化技術を用いてアプリケーションとハードウェアを隔離しセキュリティを向上できる [1]．準仮想化ドライバを用いる方法では，ドライバに入出力記録機構を実装して様々な I/O 機器との入出力を記録できる．しかし，この方法では I/O 機器の種類ごとに準仮想化ドライバを実装する必要がある．一方，低レベル I/O を監視する方法は I/O 機器の種類ごとの実装が必要ないので IoT デバイスでの利用に適している．その反面，従来の低レベル I/O の監視は主にデバイスの動作の確認のために利用されており，セキュリティのために特定のアクセスを監視するといった用途は想定されていない．

本研究の目的は，IoT 向けのセキュリティ向上手法の一部として入出力記録機構を設計・実装することである．特定の低レベル I/O を監視することで，不正アクセス監視を低コストで実現できる．

本稿では軽量ハイパーバイザ上の IoT 向け入出力記録機構を提案する．入出力に用いられる I/O アドレスを捕捉して，特定の I/O 機器へのアクセスをゲスト OS に対して透過的に記録する．この機構によって，準仮想化ドライバによる入出力記録機構のように I/O 機器の種類ごとに実装する必要がなくなる．BitVisor の PCI デバイス I/O モニターを用いた提案機構の実装についても述べる．

研究課題は以下の 2 点である．

1. IoT 向け低レベル入出力記録機構の設計
2. IoT 向け低レベル入出力記録機構の実装

2 関連研究

大野らは，ゲスト OS による不正なファイル操作を検知するために，ハイパーバイザ内で VirtIO を利用してブロックデバイスへの操作を監視する手法を提案した [2]．この手法により，ブロックデバイスへの不正アクセスを検知できるので，セキュリティの向上が期待される．

大野らの提案手法は，VirtIO を用いて様々なブロックデバイスを一つの仮想デバイスに抽象化する．これによって，ブロックデバイスの種類が異なってもブロックデバイス操作の監視方法は変わらないので，提案手法を様々なブロックデバイスに適用できる．しかし，ブロックデバイス以外の入出力装置に対しては，異なるデバイスドライバ上に同様の仕組みを実装する必要がある．

3 IoT 向け低レベル入出力記録機構

本研究では，ゲスト OS と様々な I/O 機器の入出力を記録できる IoT 向け低レベル入出力記録機構を提案する．I/O 機器への不正アクセスは特定のアドレスを用いて行われるので，すべてのアクセスを記録する必要はない．提案機構は特定のアドレスに対するアクセスのみを記録する．

3.1 想定する IoT デバイス

IoT デバイスには，様々な種類のデバイスが存在する．本研究において想定する IoT デバイスは，I/O 機器を制御し，エッジサーバとデータの送受信を行うデバイスである．これらのデバイスは OS や軽量のハイパーバイザを動作させることができる程度の計算資源を持っており，本研究の提案機構を用いてデバイス内部でのセキュリティ対策の強化を実現できる．一方，センサ機能だけを持つ IoT デバイスのように，OS やハイパーバイザを動作させることができないデバイスは対象としない．

3.2 IoT 向け低レベル入出力記録機構の構成

提案する IoT 向け低レベル入出力記録機構はハイパーバイザ内のアクセス記録機構からなる (図 1)．I/O 機器のレジスタが Memory-Mapped I/O(MMIO) によってゲスト OS のメモリ空間にマップされている．利用者は記録したいアドレスを入出力記録機構にあらかじめ設定しておく．アプリケーションが入出力を行うと，これらのアドレスに対するアクセスが発生する．アクセス記録機構はこれらのアクセスを捕捉して記録する．

アクセス記録機構は，特定のアクセスを捕捉する仕組みによって実現できる．この仕組みと I/O 機器の種類には関係がないので，準仮想化ドライバによる入出力記録機構のように，I/O 機器の種類ごとに実装する必要がない．

4 IoT 向け低レベル入出力記録機構の実現

本研究では，BitVisor の PCI デバイス I/O モニターを利用して IoT 向け低レベル入出力記録機構を実現する．

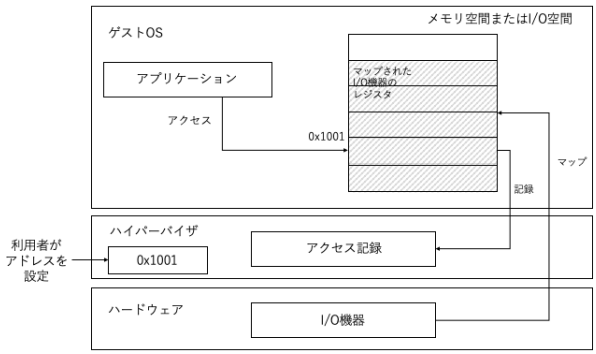

図 1 IoT 向け低レベル入出力記録機構の構成図

BitVisor は，単一のゲスト OS だけを動作させることができる軽量のハイパーバイザである [3].

PCI デバイス I/O モニターは，ゲスト OS と PCI デバイスの入出力を記録する準パススルードライバである．PCI デバイスの Base Address Register(BAR) を用いて，ゲスト OS との入出力を記録する．BAR には I/O 機器のレジスタのベースアドレスが格納されているので，これを用いてゲスト OS と PCI デバイスの入出力を記録する．

提案機構では，PCI デバイス I/O モニターに対して記録するアドレスを指定する部分と，指定に基づいてアクセスするアドレスを選択する部分を追加で実装した．

5 実験

提案機構の有効性を確認するため，ゲスト OS が I/O 機器へのアクセスに要する時間を計測する実験を行う．本節では，実験環境や実験内容，実験結果について述べる．

5.1 実験環境

VMware で BitVisor，Ubuntu を動作させて計測を行う．I/O 機器へのアクセスは Ubuntu 上で実行する自作プログラムが行う．使用したソフトウェアのバージョンを表 1 に示す．使用した PC の仕様は，CPU が Intel Core i5-7200U 2.50GHz，メモリが 16.0GB である．I/O 機器は，Adafruit FT232H を搭載した General-Purpose I/O(GPIO) ボードを使用する．

表 1 使用したソフトウェアのバージョン

ソフトウェア	バージョン
VMware	VMware Workstation 16 Player
BitVisor	BitVisor 2.0 (GitHub 2022/5/23 コミット版)
Ubuntu	Ubuntu 22.04 LTS
Linux カーネル	Linux 5.15.0-56-generic

5.2 実験内容

GPIO ボードに接続された LED を点灯させる自作アプリケーションの実行に要する時間を，GPIO ボードのオープン直前とクローズ直後に clock_gettime() を呼び出し計測する．入出力記録に何も使用しない場合，PCI デバイス I/O モニターを使用する場合，提案機構を実現したアクセス記録機能を使用する場合の結果を比較する．PCI デバイス I/O モニターで記録する I/O アクセスは 1024 アドレス分であり，提案手法は 1024 アドレスのうちアドレス 0xfc051010, 0xfc051014, 0xfc051018 番地に対応するアクセスを記録する．また，計測は各場合で 10 回ずつ行う．

5.3 実験結果

計測した結果を表 2 に示す．モニターは PCI デバイス I/O モニターを，提案機構はアクセス記録機能を意味する．

表 2 I/O 機器へのアクセスに要する時間の計測結果

	使用しない [s]	モニター [s]	提案機構 [s]
平均	0.45	0.49	0.50
標準偏差	0.03	0.03	0.03

実験結果から，PCI デバイス I/O モニターの方がアクセス記録機能よりも I/O 機器との入出力時のオーバーヘッドが小さいことがわかる．

6 おわりに

本稿では，IoT 向けのセキュリティ向上手法の一部として入出力記録機構を設計・実装することを目的に，IoT 向け低レベル入出力記録機構を提案した．また，BitVisor の PCI デバイス I/O モニターを用いて提案機構を実現した．提案機構は PCI デバイス I/O モニターよりも記録に要するメモリ量が少ないが，実験の結果からアクセスに要する時間が長いことが判明した．

今後の課題は，入出力記録を用いた不正アクセス検知の実現である．IoT デバイス上で実現する場合のオーバーヘッドを考慮して，入出力記録を外部のコンピュータへ送信して実現する手法について検討する．

参考文献

[1] C. Moratelli, et al., “Embedded Virtualization for the Design of Secure IoT Applications,” in *Proceedings of the 27th International Symposium on Rapid System Prototyping: Shortening the Path from Specification to Prototype*, pp.2-6, 2016.

[2] 大野 仁 他, “組込み型ハイパーバイザにおける VirtIO を利用した不正ファイルアクセス監視方法,” in *The 39th Symposium on Cryptography and Information Security(SCIS2022)*, 2C3-1, 2022.

[3] T. Shinagawa, et al., “BitVisor: A Thin Hypervisor for Enforcing I/O Device Security,” in *Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments*, pp.121-130, 2009.