様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　４ 年　 ５ 月　 ９ 日現在

機関番号：３２６１２
研究種目：基盤研究(C)（一般）
研究期間：2019～2021
課題番号：１９Ｋ１１９０６

研究課題名（和文）レガシーコードの実行回避によるハイパーバイザの安全性向上

研究課題名（英文）Avoiding Legacy Code Execution to Improve Hypervisor Security

研究代表者
河野　健二（Kono, Kenji）

慶應義塾大学・理工学部（矢上）・教授

研究者番号：９０３０１１１８

交付決定額（研究期間全体）：（直接経費）　　3,400,000円

研究成果の概要（和文）：クラウド環境の基盤であるハイパーバイザの安全性向上はすべてのユーザの安全性に直結する．本研究課題では，ハイパーバイザの脆弱性をあらかじめすべて取り除くことは困難であるという立場から，脆弱性があっても安全に運用できることを目指す．ハイパーバイザの脆弱性の多くが，レガシーな命令やデバイスをエミュレーションするコードにあることに着目し，エミュレーションする必要のないレガシー・コードの実行要求を効果的にフィルタリングする手法を確立した．特に命令エミュレータと仮想デバイスエミュレータに対する研究・開発をすすめ，これまでに報告されている多くの攻撃に対して有効であり，実行時オーバーヘッドも低いことを示した．

研究成果の学術的意義や社会的意義
マルチテナント型のクラウド環境はすでに広く普及しており，社会基盤のひとつとなっている．テナント間での情報の秘匿性・完全性などを保証する基盤となっているのはハイパーバイザという低レイヤのソフトウェアである．ハイパーバイザに脆弱性があれば，そこが攻撃の起点となり，クラウド上のすべてのサービス，すべての利用者のセキュリティが損なわれる．本研究は，クラウド環境の根幹であるハイパーバイザの安全性向上に寄与するものであり，その社会的意義は高い．ハイパーバイザそのものの機能拡張は行っておらず，既存のハイパーバイザと仮想マシンの間に薄いソフトウェアレイヤを仕込めばよく，デプロイも容易であると期待される．

研究成果の概要（英文）：Improving the safety of the hypervisor, which is the basis of the cloud environment, is directly linked to the safety of all the users of the cloud environments. In this research project, from the standpoint that it is difficult to eliminate all the vulnerabilities lurking in the hypervisors in advance, we aim to manage the hypervisors safely even if they have serious vulnerabilities. Focusing on the fact that many of the hypervisor vulnerabilities are in the code that emulates legacy instructions and devices, we have established a method for effectively filtering the execution requests of legacy code that does not need to be emulated. Our focus is especially on the instruction emulators and virtual device emulators for research and development, and it has been shown that the proposed method is effective against many attacks reported so far and has low run-time overhead.

研究分野：システムソフトウェア

キーワード：ハイパーバイザ　仮想マシン　仮想デバイス　脆弱性　セキュリティ

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９（共通）

１．研究開始当初の背景

仮想化技術はクラウド環境の基盤として広く使われており，その安全性はクラウド利用者全ての安全性に直結する．　特に，ハイパーバイザ（仮想マシンモニタともいう）の脆弱性は致命的なセキュリティホールとなりやすく，ひとたびその脆弱性が悪用されると，権限昇格などにより情報漏洩や改ざんなどの被害をもたらしうる．実際，2015 年に発見された VENOM というマルウェアは，ハイパーバイザの脆弱性を突き権限昇格を可能にするものであった．このような脆弱性は，クラウドにおけるプライバシ・データの利活用を阻害し，機械学習やビッグデータ解析による新たな産業の創出すら阻害する危険性を孕んでいる．

現在，広く実用的に用いられている Xen や KVM などのハイパーバイザは，残念ながら，多くの脆弱性が報告されている．2018 年の 11 月の段階で合計 350 件を超える脆弱性が報告されており，その件数は着実に増加している．今後，多くの貴重なデータがクラウド環境に集約されるようになるにつれ，その基盤となるハイパーバイザが格好の攻撃ターゲットになることは間違いがない．

２．研究の目的

クラウド環境の基盤をなすハイパーバイザの安全性の向上を目指す．ハイパーバイザの安全性を向上させる既存のアプローチは，その脆弱性を効率よく発見することに注力しているものが多い．それに対し，本研究のアプローチは「脆弱性があっても安全に運用する」ことを狙ったものである．ハイパーバイザに限らず，一般に，ソフトウェアの脆弱性を開発段階ですべて取り除くことは難しい．ましてや，ハイパーバイザのような複雑なソフトウェアの場合はなおさらである．したがって，「ハイパーバイザには脆弱性があるもの」という前提の上で，いかにして脆弱性のあるハイパーバイザを安全に運用するかという視点から安全性の向上をはかる．

３．研究の方法

本研究の着眼点は，ハイパーバイザに残る脆弱性の多くがレガシーな実行環境に対応するためのコードに存在するという点にある．ハイパーバイザの実装は，レガシーな環境に対応するためのコードを多く含んでおり，最新の仮想化支援ハードウェアを搭載した CPU や I/O デバイス（SR-IOV 対応の NIC など）では不要なコードも多く存在する．仮想化支援のための機能拡張がなされるたびに，ハイパーバイザに対する機能要件が簡略化されている．たとえば，メモリ仮想化支援のための EPT (extended page table) という機能拡張がなされる以前は，シャドウページテーブル (shadow page table) というソフトウェア的な技法を使ってメモリを仮想化していた．EPT を有効にした環境では，シャドウページテーブルに関する一連のコードは不要となっている．

ゲスト環境を乗っ取った攻撃者は，さまざまな攻撃ベクタを用いてレガシーコードの強制実行を試みる．たとえば，ハイパーバイザの一部である命令エミュレータを強制的に起動し，命令エミュレータ内のレガシーコード (現在の環境では実行するはずのないコード) の脆弱性を悪用することができることが報告されている．レガシーコードは，複雑である上に
最新の環境では十分に検査されておらず脆弱性の温床である．

本研究では「実行する必要のないレガシーコードの実行を阻止する」ことで，レガシーなコードに残る脆弱性の悪用を防止する．本研究で提案するシステムは，ゲスト環境とハイパーバイザの界面で動作し，「ゲスト環境からのイベントを横取りし，不適切なイベントをフィルタリング」してレガシーコードの実行を回避する．たとえば，EPT が有効な環境であるにも関わらず，シャドウページテーブルのためのイベントが発行されれば，それは不正であるとみなしてフィルタリングする．

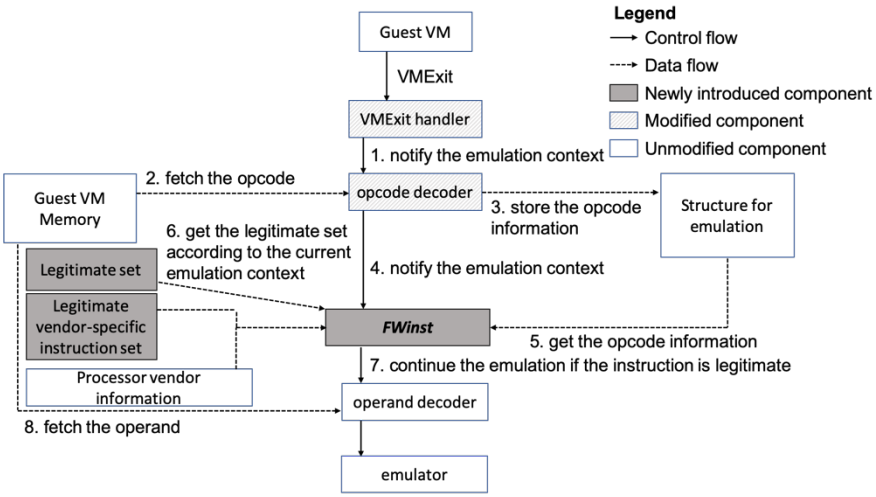
４．研究成果

（1）命令エミュレータにおけるレガシーコードの実行回避

ハイパーバイザでは多くの命令を CPU 上でネイティブに実行するようになっているものの，一部の命令に関してのみ，命令のエミュレーションを行う．エミュレーションが必要な命令は極めて限定されており，特に，仮想化のためのハードウェア・サポートが実用化されてからは，エミ

ュレーションの必要な命令は大きく削減されている．にもかかわらず，ハイパーバイザはレガシーなハードウェア上でも動作するように実装されており，比較的最近の CPU であればエミュレートする必要のない命令であってもエミュレーションできるようになっている．ゲスト環境において，エミュレーションの必要な命令が実行されると，VMExit というハードウェア・イベントが発生し，ハイパーバイザに制御が移行する．本研究で提案する手法では，1）ハイパーバイザが動作する CPU のマイクロ・アーキテクチャおよび，　2）ハイパーバイザの設定から導出されるエミュレーション・コンテキストに基づいて，命令のエミュレーションが必要な環境であるかどうかを判定する．もし，エミュレーションの必要のない環境であれば，脆弱性を突くための不正攻撃であるとみなして，フィルタリング・アウトする．このような仕組みを導入することで，脆弱性を突くことを狙ったレガシー・コードの実行を事前に阻止するようになっている．

下図に提案手法の概略を示す．命令エミュレーションを要請する VMExit が発生すると，エミュレーション・コンテキストに基づいて，エミュレーションが必要であるかどうかを判断する．エミュレーションが必要であればエミュレーションを行い，不要であれば不正攻撃と判断してエミュレーションは行わない．



下図にエミュレーション・コンテキストと，そのコンテキストでエミュレーションする必要のある命令の一覧を示す．エミュレーション・コンテキスト，そのコンテキストを識別するための情報，およびそのコンテキストでエミュレーションを許可する命令が一覧となっている．これは Intel 社の提供する CPU のマイクロ・アーキテクチャの進化に合わせて精緻な分析を行った結果として得られたものである．

Emulation Context	Context Identification	Legitimate Instructions
PIO	I/O instruction	`in, out`
MMIO	EPT violation or EPT misconfig	`mov, movsx stosx, or`
Shadow page table	Exception or NMI (#PF)	memory access instructions
Real mode	VCPU status (No VMExit)	all real-mode instructions
Migration	Exception or NMI (#UD)	`vmcall, vmmcall syscall, sysenter sysexit, rsm, movbe`
UMIP	Access to GDTR or IDTR or Access to LDTR or TR	`sgdt, sidt, sldt smsw, str`

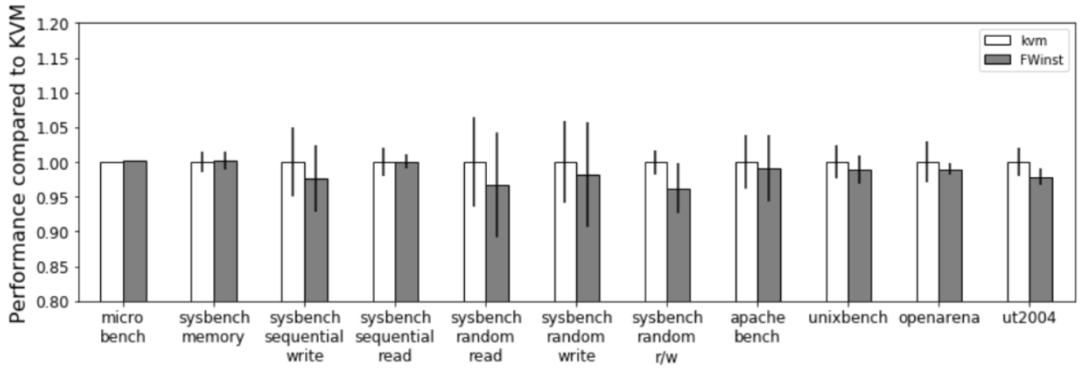
本提案手法を Linux のハイパーバイザである KVM に実装した．Intel Xeon Silver 4110 (Skylake) および Intel Xeon X5650 (Westmere) 上で動作する Linux Kernel 4.8.1 を用いて評価実験を行った．次のページにその詳細を示す．提案手法を用いることで，多様な脆弱性に対して正しくフィルタリングを行うことができ，不正攻撃を未然に防止できることがわかる．いく

つかの攻撃に対しては提案手法では防止できないとなっているものの，現在の標準的な環境では問題とならない．

CVE #	vulnable instruction	Intel Westmere	Intel Haswell	AMD	vul. comp.	emu. context
2018-10853	`fxrstor, fxsave, sgdt, sidt`	√	√	√	emu.	UMIP, Real Mode
2017-17741	`vmmcall, vmcall`	d	d	d	emu.	Migration
2017-7518	`syscall`	√	√	√	emu.	Migration
2017-2584	`fxrstor, fxsave, sgdt, sidt`	×	×	×	emu.	UMIP, Real Mode
2017-2583	`mov SS`	√	√	√	emu.	Real Mode
2016-9756	`far jump` or `far ret`	√	√	√	emu.	Real Mode
2016-8630	illegal instruction	√	√	√	operand	None
2015-0239	`sysenter`	√	√	d	emu.	Migration
2014-8481	`movbe`	d	√	√	operand	Migration, Real Mode
2014-8480	`clflush, hint-nop, prefetch`	√	√	√	operand	Real Mode
2014-7842	unsupported instructions by the instruction emulator	√	√	√	emu.	None
2014-3647	`far jump` or `far ret`	√	√	√	emu.	Real Mode
2014-0049	`pusha`	√	√	√	emu.	Real Mode
2012-0045	`syscall`	√	√	√	emu.	Migration
2010-5313	unsupported instructions by the instruction emulator	√	√	√	emu.	None
2010-0435	`mov DR`	√	√	√	emu.	Real Mode
2009-4031	instruction that contains too many bytes	×	×	×	opcode	All

√: defended, ×: not defended, d: depends on migration contexts, emu.: emulation, operand: operand decoder

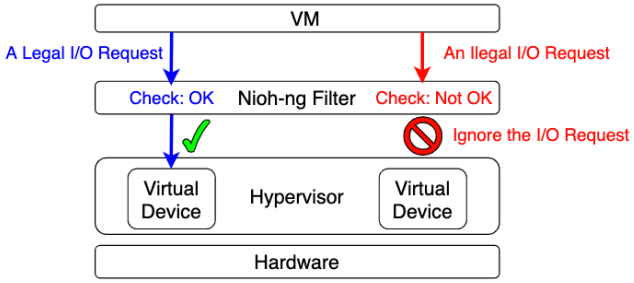
提案手法による実行時オーバーヘッドも十分に小さい．詳細な実験環境およびベンチマークの説明は省略するものの，提案手法なし（白い棒グラフ）と提案手法あり（グレーの棒グラフ）を比較すると，極端な性能低下は発生していない．



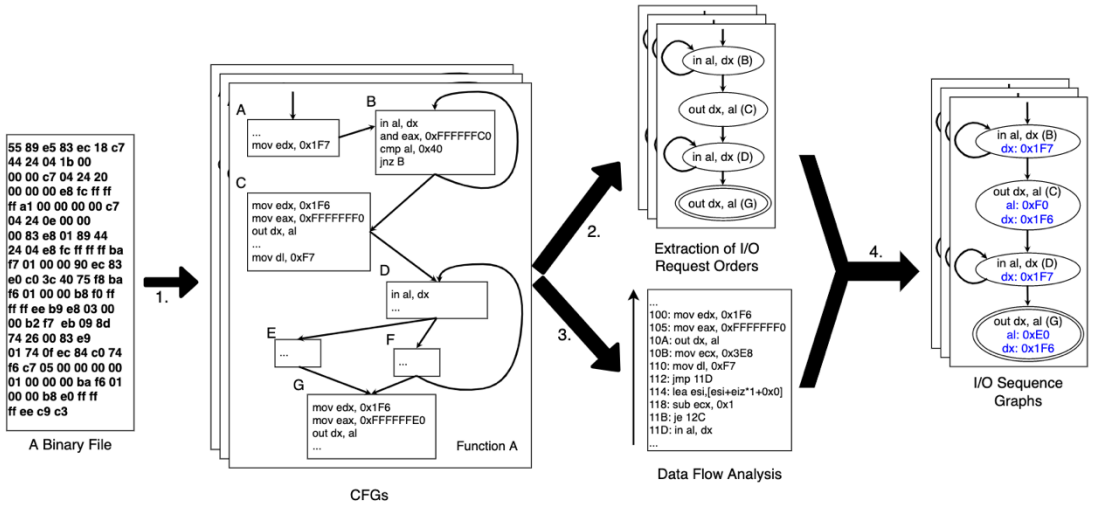
（2）仮想デバイスにおけるレガシーコードの実行回避

ハイパーバイザが仮想化している資源の 1 つに I/O がある．仮想マシン内で動作するゲスト・オペレーティングシステムは物理マシンを直接操作することはできないため，仮想マシンが I/O を要求すると，ゲスト環境からハイパーバイザへと制御が遷移する．その後，ハイパーバイザ内のデバイスエミュレータがデバイスのエミュレートを行う．ハイパーバイザが仮想化するデバイスの中にはレガシーなものも多く含まれており，そのようなデバイスをエミュレーションするコードの脆弱性を突く攻撃が多く知られている．

本研究では，デバイスエミュレータの脆弱性からハイパーバイザを防御する仕組みの実現を行った．仮想マシンとハイパーバイザの間で不正な I/O 要求をフィルタリングする階層を導入する．不正な I/O 要求の多くは，ゲスト環境のデバイスドライバが要求する I/O シーケンスとは異なっていることに着目し，ゲスト環境のデバイスドライバが生成しうる I/O シーケンスを事前に抽出し，その I/O シーケンスと一致しない I/O 要求はフィルタリング・アウトする．下に概要図を示す．



ゲスト環境で動作するデバイスドライバのバイナリ・コードから I/O 要求のシーケンスを静的解析により抽出するという方法をとった．下図に示すように，デバイスドライバのバイナリ・コードからコントロール・フロー・グラフを生成し，そこから I/O シーケンスの抽出を行う．さらにデータフロー解析を行うことで，I/O 要求におけるレジスタの値などが確定できる時は，そうした値の確定まで行う．最後にそれらをマージして I/O シーケンス・グラフを生成する．



ゲスト環境から I/O 要求があると，その I/O 要求が I/O シーケンス・グラフに合致するかどうかを検査し，合致すれば正当な I/O 要求とみなして実行する．合致しなければ不正 I/O 要求とみなしてフィルタリング・アウトする．

CVE ID	CVSS v2 Score	Emulated Device	Cause of Vulnerability	Can Reject This Exploitation?
CVE-2016-7909	4.9	AMD PCNet-PCI II Ethernet Controller	Inadequate Validation of Values	Yes
CVE-2015-5279	7.2	NE2000 Network Card (RTL8029AS etc.)	Buffer Overflow	Yes
CVE-2020-13361	3.3	Ensoniq ES1370 Audio PCI	Out-of-bounds access	Partially Yes
CVE-2020-13800	4.9	Rage 128 and RV100 ATI SVGA	Inadequate Validation	Partially Yes
CVE-2020-15863	7.2	XGMAC Ethernet Controller	Buffer Overflow	No

上の表に提案方式の有効性を評価した結果を示す．バイナリ解析の精度に依存するため，すべての攻撃に対して完全な防御を行うまでには至らないものの，一定の有効性があることがわかる．現在，本研究テーマの発展形として，セキュリティ・パッチがリリースされるまでの一時繋ぎとして利用できるように手動でフィルタを記述する方式の研究・開発を進めつつある．

５．主な発表論文等

〔雑誌論文〕　計1件（うち査読付論文　1件／うち国際共著　0件／うちオープンアクセス　0件）

１．著者名 Kenta Ishiguro, Kenji Kono	４．巻 E103-D
２．論文標題 Instruction Filters for Mitigating Attacks on Instruction Emulation in Hypervisors	５．発行年 2020年
３．雑誌名 IEICE Trans. on Information and Systems	６．最初と最後の頁 -
掲載論文のDOI（デジタルオブジェクト識別子） 10.1587/transinf.2019EDP7186	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 －

〔学会発表〕　計4件（うち招待講演　0件／うち国際学会　0件）

１．発表者名 伊東 拓海，石黒 健太，河野 健二
２．発表標題 関数ポインタの使用有無に着目した Intel CETによる攻撃ガジェットの削減
３．学会等名 情報処理学会 OS 研究会
４．発表年 2020年

１．発表者名 庄司 豊，石黒 健太，河野 健二
２．発表標題 バイナリ解析に基づく仮想デバイスの不正 I/O 要求のフィルタリング
３．学会等名 情報処理学会
４．発表年 2020年

１．発表者名 石黒 健太，河野 健二
２．発表標題 ハイパーバイザに対するファジングの効果的な初期シード生成
３．学会等名 情報処理学会
４．発表年 2020年

１．発表者名 平松 勇人，石黒 健太，河野 健二
２．発表標題 カーネル内インタプリタに特化したファジングの提案
３．学会等名 情報処理学会
４．発表年 2021年

〔図書〕　計0件

〔産業財産権〕

〔その他〕

―

６．研究組織

	氏名 （ローマ字氏名） （研究者番号）	所属研究機関・部局・職 （機関番号）	備考

７．科研費を使用して開催した国際研究集会

〔国際研究集会〕　計0件

８．本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関