様式 C-19

科学研究費補助金研究成果報告書

平成２２年　５月１８日現在

研究種目：若手研究（B）
研究期間：2007～2009
課題番号：１９７００００２
研究課題名（和文）　仮想計算機を用いたきめ細かいセキュリティ機構の研究

研究課題名（英文）　Fine-grained Security Mechanisms Using Virtual Machines

研究代表者
光来　健一（KOURAI KENICHI）
九州工業大学・大学院情報工学研究院・准教授
研究者番号：６０３７２４６３

研究成果の概要（和文）：攻撃によってOSの制御を奪われた場合でも不正なファイルアクセスや不正な通信だけを防げるようにするために、仮想計算機モニタ（VMM）でOSレベルの情報を用いてきめ細かいアクセス制御を行うシステムを開発した。また、サービス妨害攻撃に対処するためにVMMが直接OSのスケジューリングを変更するシステムを開発した。これらのシステムの開発を通して、VMMがOSレベルの情報を扱う際に必要となるインタフェースの定義を行った。

研究成果の概要（英文）：We have developed fine-grained access control systems using OS-level information in the virtual machine monitor (VMM). The developed systems prevent only illegal file accesses and communication even if the attackers take the control of an OS. We have also developed a system in which the VMM directly changes OS-level scheduling. Through the development of these systems, we defined a new interface necessary for using OS-level information in the VMM.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
２００７年度	1,100,000	0	1,100,000
２００８年度	1,100,000	330,000	1,430,000
２００９年度	1,000,000	300,000	1,300,000
年度			
年度			
総　計	3,200,000	630,000	3,830,000

研究分野：総合領域
科研費の分科・細目：情報学・ソフトウェア
キーワード：ディペンダブルシステム、仮想計算機、セキュリティ、アクセス制御、スケジューリング、オペレーティングシステム

１．研究開始当初の背景

従来、計算機システムに対する攻撃はオペレーティングシステム（OS）を中心として対処が行われており、OS によるセキュリティ機構についてこれまでに様々な研究がなされてきた。例えば、サーバソフトウェアの脆弱性を利用してシステムに侵入されても、OS のアクセス制御により不正アクセスを防ぐことができる。また、OS が資源の使用量を制限することで、サービス妨害攻撃を緩和す

ることができる。申請者もこれまでに OS レベルで攻撃者の侵入を確実に検知するシステムについて研究してきた。

しかし、OS 自体が攻撃を受けた場合、OS によって提供されているこのようなセキュリティ機構は機能しなくなる危険性がある。近年、OS の脆弱性が数多く報告されており、OS 自体が攻撃を受ける機会は増加する傾向にある。攻撃により OS の制御を奪われると、OS が提供している計算機資源に対するアクセス制御を無効化されてしまう。その結果、攻撃者による計算機資源への不正アクセスを許すことになる。また、サービス妨害攻撃により OS 自体が過負荷になると、OS 上で提供されている全てのサービスが停止してしまう。このようなサービス停止はサーバ計算機にとっては致命的である。

２．研究の目的

本研究では、攻撃により OS のセキュリティ機構が機能しなくなった場合でも、計算機資源への不正アクセスおよびサービスの妨害を防ぐことを目的とする。そのために、OS を仮想計算機上で動作させるようにし、仮想計算機モニタ（VMM）と呼ばれる仮想計算機を管理するソフトウェアにセキュリティ機構を実装する。VMM は OS から独立しており、OS が計算機資源にアクセスする時には必ず VMM を経由するという特徴を利用して、VMM 上で OS に依存しないセキュリティ機構を実現する。

さらに、VMM できめ細かいセキュリティ機構を実現できるようにするために、OS と VMM の間のインタフェースを再定義することも目的とする。従来、VMM はハードウェアレベルのインタフェースのみを OS に提供していたため、計算機資源に関する高レベルの情報（ファイル名など）は失われていた。このような高レベルの情報はきめ細かい制御には欠かせないため、VMM が OS から高レベルの情報を取得できるようにする。その際に、VMM が OS から偽の情報を渡されないようにする。

３．研究の方法

VMM で以下のセキュリティ機構を実現し、その設計・実装を通して OS と VMM の新しいインタフェースを定義する。

(1) きめ細かいアクセス制御

攻撃により OS の制御を奪われ、OS のアクセス制御が無効化された場合でも、VMM のアクセス制御により不正アクセスを防げるようにするシステムを開発する。このアクセス制御では、従来、VMM レベルでは提供するのが難しかったファイル単位の制御を行えるようにする。OS を介さずに確実に認証を行えるようにするために、VMM がユーザを直接、認証できるようにする。また、OS 内のファイアウォールが無効化されても侵入者が外部と行う不正な通信だけを禁止できるように、VMM レベルのパケットフィルタで OS の情報を利用できるようにする。

(2) 資源要求のスケジューリング

サービス妨害攻撃により OS が過負荷になった場合でも、VMM が資源割り当てを制御することで OS の負荷を下げられるようにする。このような場合、OS が過負荷になっているためプロセスレベルでの制御は難しかったため、従来は VM 単位での資源割り当ての制御が行われていた。本研究では VMM からプロセス単位の制御を行えるようにする。また、VMM 本体に対するサービス妨害攻撃として、間接的にメモリリークを起こさせることで動作を遅くしたりクラッシュさせたりする攻撃がある。このような攻撃は対処が難しいため、高速に VMM を若化させることで正常な状態に戻せるようにする。

４．研究成果

(1) VMM によるファイルアクセス制御

VMM でファイルアクセス制御を行うことにより、攻撃によって OS の制御を奪われた場合でもファイルに対する不正アクセスを防ぐことができるシステムを開発した。このシステムでは仮想計算機にローカルディスクを持たせず、仮想計算機がファイルにアクセスする時には VMM に対してファイル要求を行わなければならないようになっているため、VMM で確実にアクセス制御を行うことができる。

VMM はファイル要求を出したユーザを確認するために、仮想計算機を使っているユーザに対して直接ダイアログを出して認証を行うことで、OS に頼らないセキュリティを実現した。このダイアログはユーザが使っている画面に自然な形で表示されるため、ユーザに違和感を感じさせないようにすることができる。攻撃者が出す偽のダイアログと区別できるようにするために、VMM とユーザだけが知っている情報をダイアログに表示する。

VMM が仮想計算機に対してファイルを提供できるようにすることで、ファイル単位のきめ細かいアクセス制御が可能となった。従来の VMM ではディスクのブロックレベルのインタフェースを提供していたため、きめ細かいアクセス制御を行うのは難しかった。そのため、VMM に OS のファイルシステムレベルのインタフェースを定義した。

OS の制御を奪われた時に OS のファイルキャッシュ上にある機密情報が漏洩する危険性を減らすために、認証の有効期限が切れた時に OS のファイルキャシュを強制的に消去

する仕組みを開発した。VMM が OS のファイルキャッシュとして使われているメモリ領域を見つけて内容を消去し、ファイルキャッシュとして使われないようにする。OS のファイルキャッシュを消去する仕組みを OS-VMM 間のインタフェースとして定義した。

このようなアクセス制御の下で setuid されたプログラムを実行できるようにするために、VMM に依頼して実行するためのインタフェースを定義した。管理者に setuid されたプログラムは管理者の持つファイルにアクセスするが、VMM による認証ではユーザが管理者である必要がある。この問題を解決するために、setuid プログラムは VMM を介して別の仮想計算機で実行できるようにした。

(2) VMM によるきめ細かいパケットフィルタリング

仮想計算機に侵入された後、仮想計算機からの踏み台攻撃を緩和するために、VMM から OS 内部のプロセスやユーザといった情報を用いてきめ細かいパケットフィルタリングを行うシステムの設計を行った。VMM から仮想計算機上で行われている通信の情報を取得するために、OS 内の通信に関するデータ構造を参照するためのインタフェースを定義した。

この設計に基づいて、パケットフィルタを仮想化ソフトウェア Xen のドメイン 0 と呼ばれる特権を持った仮想計算機上に実装した。この実装を通じて、パケットフィルタとドメイン 0 の OS カーネルの間に必要となるインタフェースを新たに定義した。このシステムの性能評価を行ったところ、ドメイン 0 から仮想計算機内の OS の情報を解析する部分に非常に大きなオーバヘッドがあることが分かった。

そこで、このシステムの高性能化のために、パケットをまとめて一括で検査する機構、および、同一コネクション内を流れるパケットの検査を省略する機構を開発した。前者の機構を実現するために、パケットフィルタとドメイン 0 のカーネルの間のインタフェースの修正を行った。これらの機構により、性能を大幅に改善することができた。

さらなる高速化を検討したところ、開発したパケットフィルタを VMM 内に組み込むことが必要であるという結論に達した。Xen においてはパケット処理はドメイン 0 と VMM の両方が関わっているため、パケットフィルタを VMM 内に組み込む場合でもシステム全体のインタフェースを考える必要がある。そこで、パケットフィルタを VMM に組み込む際に必要になるドメイン 0 と VMM の間のインタフェースの設計を行った。

(3) VMM によるプロセススケジューリング

VMM から仮想計算機内の OS のスケジューリングを変更することにより、特定のプロセスがシステムを過負荷にするサービス妨害攻撃を防ぐシステムを開発した。開発したシステムでは、サービス妨害攻撃を行っているプロセスを指定して、スケジューリング対象から外すことで実行を阻止する。OS のスケジューラが使っているデータ構造を VMM から変更できるようにするために、VMM が OS のデータ構造にアクセスするためのインタフェースを定義した。

VMM が特定のプロセスを強制的にスケジューリング対象から外すための手法として、スケジューラのランキューで待っているプロセスについてはランキューから取り除くようにした。この際に OS の一貫性を損わないようにするために、OS がランキューを操作中でないことをチェックするようにした。一方、現在実行中のプロセスや I/O 待ちを行っているプロセスについては、VMM がプロセスの状態を書き換えることによりスケジューリング対象から外せるようにした。

研究の初期段階ではオープンソースで広く使われている Linux を対象として研究を進めたが、Linux のスケジューリングの変更ができるようになった時点で、対象 OS を Windows に拡大した。Windows の内部構造は Linux とは異なる部分が多く、内部情報が非常に少なかったため、様々な間接的な手法を組み合わせることにより、プロセスをランキューから外せるようになった。

一方、Windows の I/O 待ちプロセスの制御には Linux と同じ手法は使うことができなかった。Windows に適用できる手法を模索したが、Windows で I/O 待ちしているプロセスを VMM から制御するのは困難であるという結論が得られた。そこでランキュー待ちをしているプロセスだけを制御する手法でどのくらい正確に制御ができるか調べたところ、ある程度正確に制御を行うことができることが分かった。

(4) VMM の高速な若化

VMM の資源を不足させられることによるサービス妨害攻撃への対処として、VMM を高速に再起動することで正常な状態を回復する若化手法を開発した。VMM の性能低下はその上で動作しているすべての仮想計算機に影響を及ぼす。また、VMM を再起動する際にはすべての仮想計算機を一旦終了させ、VMM の再起動後に起動させる必要があり、非常に長い時間がかかっていた。開発した手法では、VMM 上の仮想計算機の状態をメモリ上に保持したまま、仮想計算機モニタだけを高速に再起動することを可能にした。

この手法により、VMM の若化時に仮想計算機上で動いているサービスのダウンタイムを大幅に削減することができた。提案手法に

似た従来手法として、仮想計算機を終了させるのではなく、ディスクに実行状態を一時保存しておくという手法もある。しかし、ディスクがボトルネックになり仮想計算機を終了させる場合よりもはるかに長い時間がかかることが分かった。提案手法は仮想計算機の数や使用しているメモリ量にほとんど影響されない非常に高速な手法である。

一方、クラスタ環境ではVMMを再起動する前に仮想計算機を別の計算機にマイグレーションすることで同等の効果が得られると考えられるため、これら２つの手法の性能を比較した。実際に大規模なクラスタ環境を用いるのは困難であったため、可用性やトータルスループットを表現するモデルを作成して、実験に基づいて性能を評価した。その結果、同じ規 模のクラスタを用いた場合、提案システムのほうがトータルスループットが高くなることが示された。

５．主な発表論文等
（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計５件）

① K. Kourai and S. Chiba, Fast Software Rejuvenation of Virtual Machine Monitors, IEEE Transactions on Dependable and Secure Computing, 査読有, 採録決定.
② K. Kourai, H. Hibino, and S. Chiba, Application-Level Scheduling Using AOP, Transactions on Aspect-Oriented Software Development V, 査読有, 5490巻, 2009年, pp.1-44.
③ 滝澤裕二, 光来健一, 千葉滋, 柳澤佳里, SAccessor：デスクトップPCのための安全なファイルアクセス制御, 情報処理学会論文誌:コンピューティングシステム, 査読有, 1巻, 2008年, pp.275-285.
④ 田所秀和, 光来健一, 千葉滋, 仮想マシン間にまたがるプロセススケジューリング, 情報処理学会論文誌:コンピューティングシステム, 査読有, 1巻, 2008年, pp.124-135
⑤ 柳澤佳里, 光来健一, 千葉滋, XenLASY：XenのI/O処理を追跡するためのアスペクト指向プロファイラ, 情報処理学会論文誌:プログラミング, 査読有, 49巻, 2008年, p.51-61.

〔学会発表〕（計８件）

① 安積武志, Xenにおけるゲスト OS の解析に基づくパケットフィルタリング, ソフトウェア科学会第26回大会, 2009年9月18日, 島根大学.
② 光来健一, VMMのソフトウェア若化を考慮したクラスタ性能の比較, 第7回ディペンダブルシステムワークショップ, 2009年7月15日, 函館大沼プリンスホテル.
③ 田所秀和, 仮想マシン間プロセススケジューリングの実環境への適用にむけて, 第111回OS研究会, 2009年4月24日, 沖縄県青年会館.
④ 新井昇鎬, セキュリティ機構のオフロードを考慮した仮想マシンスケジューラ, ミニ VM ワークショップ, 2009年3月18日, 秋葉原ダイビル.
⑤ 安積武志, 仮想マシンに対する高いサービス可用性を実現するパケットフィルタリング, 第 6 回ディペンダブルシステムワークショップ, 2008年7月4日, 函館大沼プリンスホテル.
⑥ 田所秀和, 仮想マシン間にまたがるプロセススケジューリング, 第 6 回先進的計算基盤システムシンポジウム, 2008 年 6 月 13 日, つくば国際会議場.
⑦ K. Kourai, Preventing Performance Degradation on Operating System Reboots, International Service Availability Symposium, 2008年5月20日, 東京大学.
⑧ 滝澤裕二, 光来健一, 千葉滋, 柳澤佳里, SAccessor：デスクトップPCのための安全なファイルアクセス制御システム, 第19回コンピュータシステム・シンポジウム, 2007年11月27日, 東京ファッションタウン.

６．研究組織

(1)研究代表者

光来　健一（KOURAI KENICHI）

九州工業大学・大学院情報工学研究院・准教授

研究者番号：６０３７２４６３

(2)研究分担者

（　　）

研究者番号：

(3)連携研究者

（　　）

研究者番号：