2A-01

KVMにおける機密情報の拡散追跡機能における性能改善策

森山 英明† 山内 利宏‡ 佐藤 将也‡ 谷口 秀夫‡
有明工業高等専門学校† 岡山大学大学院自然科学研究科‡

1. はじめに

計算機内で管理されている機密情報は，外部に漏えいすることで，企業や個人にとって大きな損失となる．機密情報を保持したファイルへの誤操作や，外部からの機密情報へのアクセスを検知するために，仮想計算機モニタ(VMM: Virtual Machine Monitor)を利用した機密情報の拡散追跡機能を提案した．この機能では，機密情報を操作するシステムコールをフックして確認することで，機密情報が格納されているファイルへの操作内容を利用者に通知することができる．しかし，処理のオーバヘッドが大きいという問題がある．

本稿では，KVM(Kernel-based Virtual Machine)上に実現されている機密情報の拡散追跡機能について，オーバヘッドが大きい個所を明確化し，性能改善策を考察する．

2. KVMにおける機密情報の拡散追跡機能

2.1 機能の概要

計算機内の機密情報の利用状況を把握するために，仮想計算機モニタにおける機密情報の拡散追跡機能を提案し，KVM(Kernel-based Virtual Machine) 上に実現し，評価結果を報告した[1]．機密情報の拡散追跡機能は，機密情報を有する可能性のあるファイルとプロセス(以降，管理対象ファイルと管理対象プロセス)を拡散情報として記録し，管理する．この機能を VMM 上に実装することにより，オペレーティングシステム(OS)よりも攻撃が困難である VMM で機密情報を管理できる．また，ゲスト OS のソースコードを改変することなく VMM の改変のみで機能を提供できる利点がある．

VMM における機密情報の拡散追跡機能の処理流れを説明する．最初に，ゲスト OS 上でユーザプロセスがシステムコールを発行すると，システムコールの入り口(SYSCALL)で，VMM はシステムコールの発行を検知するためにフックする．このとき，システムコール番号から，機密情報の拡散に関係するシステムコールであるか否か判定する．もし，関係しないシステムコールの場合は，制御をゲスト OS へ戻し，システムコール処理を続行する．機密情報の拡散に関係するシステムコールの場合は，プロセスが発行したシステムコール番号，ページテーブル情報，扱うファイルのファイルディスクリプタの値など，各システムコールにおいて機密情報の拡散追跡に必要な情報を取得する．その後，制御をゲスト OS へ戻し，システムコール処理を続行する．また，各システムコールは，戻り値としてシステムコール処理の成否やシステムコールを発行したプロセスが扱うファイルの情報などを返却する．これらの情報を取得するために，システムコールの出口(SYSRET)もフックする．

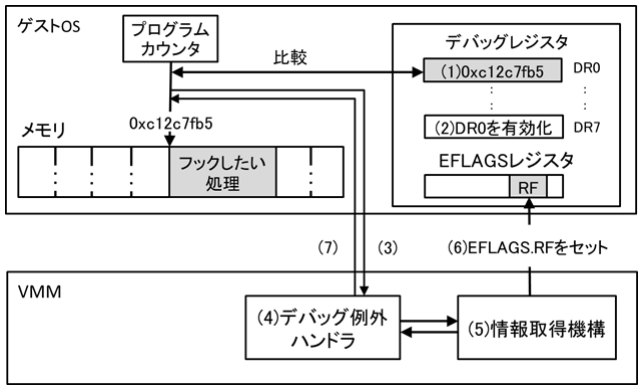

図1 ハードウェアブレークポイントによるフック

システムコールの入り口と出口のフックには，ハードウェアブレークポイントを利用する．図1 に，ハードウェアブレークポイントを用いたフックの手順を示す．最初に，フックしたい命令のアドレスをデバッグアドレスレジスタに設定し，有効化する．デバッグアドレスレジスタに設定したアドレスに格納された命令の実行を契機としてデバッグ例外が発生し，処理がゲスト OS から VMM へ移行する．VMM 側では，デバッグアドレスレジスタによるデバッグ例外であることを確認し，システムコールに応じて，機密情報の拡散追跡機能で必要な情報を取得する．

2.2 課題

機密情報の拡散追跡機能に関して，ソースコ

Method for Improving Performance of Function for Tracing Diffusion of Classified Information on KVM
† National Institute of Technology, Ariake College
‡ Graduate School of Natural Science and Technology, Okayama University

表1　フック箇所による処理時間の違いと得失の比較

フックする箇所	処理時間	長所	短所
(1)SYSCALL と SYSRET	104.0μs	機密情報の拡散経路を正確に把握することができ，警告の表示，処理の中断ができる	オーバヘッドが大きい．
(2)SYSCALL のみ	6.2μs	機密情報が漏えいする可能性がある場合に，警告の表示や処理の中断ができる．	管理対象ファイルへアクセスしたプロセスを正確に把握することができない．
(3)SYSRET のみ	97.8μs	機密情報の拡散経路を把握することができる．	機密情報が漏えいする可能性がある場合も，警告の表示や処理の中断ができない．

ード改変量やベンチマークプログラムを利用したオーバヘッドの評価を行った．この評価において，機密情報の拡散に関係しないシステムコールにおけるオーバヘッドは小さいものの，関係するシステムコールではオーバヘッドが大きくなることが示されている．このため，オーバヘッドの削減が課題となる．

3. 処理フローによる削減策の検討

図 2 に，システムコールをフックし，情報を取得する処理のフローを示す．図 2 のフローは，大きくシステムコール入り口(SYSCALL)をフックした際の処理とシステムコールの出口(SYSRET)をフックした際の処理に分割することができる．KVM における機密情報の拡散追跡機能において，オーバヘッドとなる処理を明確化するために，仮想マシン上にある管理対象ファイルに対して cp コマンドによるファイル複製を行った際の read()と write()システムコールの処理時間を，以下の 3 つの場合に分けて測定した．

(1) SYSCALL と SYSRET
(2) SYSCALL のみ
(3) SYSRET のみ

測定環境は，Fedora18(Linux Kernel 3.6.10)を搭載した計算機上に仮想マシンを 1 台用意し，仮想マシン上で管理対象ファイルの複製をした．

フック箇所による処理時間の違いと得失の比較を表 1 に示す．SYSRET フック時の処理は，管理対象ファイルやプロセスを参照していたかを判定するため，SYSCALL フック時の処理と比較してオーバヘッドは大きい．機密情報の拡散追跡機能の性能改善を行うには，この処理を改善する必要がある．

4. おわりに

本稿では，KVM における機密情報の拡散追跡機能に関して，ボトルネックとなる箇所を検討し，測定を行った．今後は，機密情報の拡散追跡を実現しつつ，オーバヘッドを削減する方法について検討する．

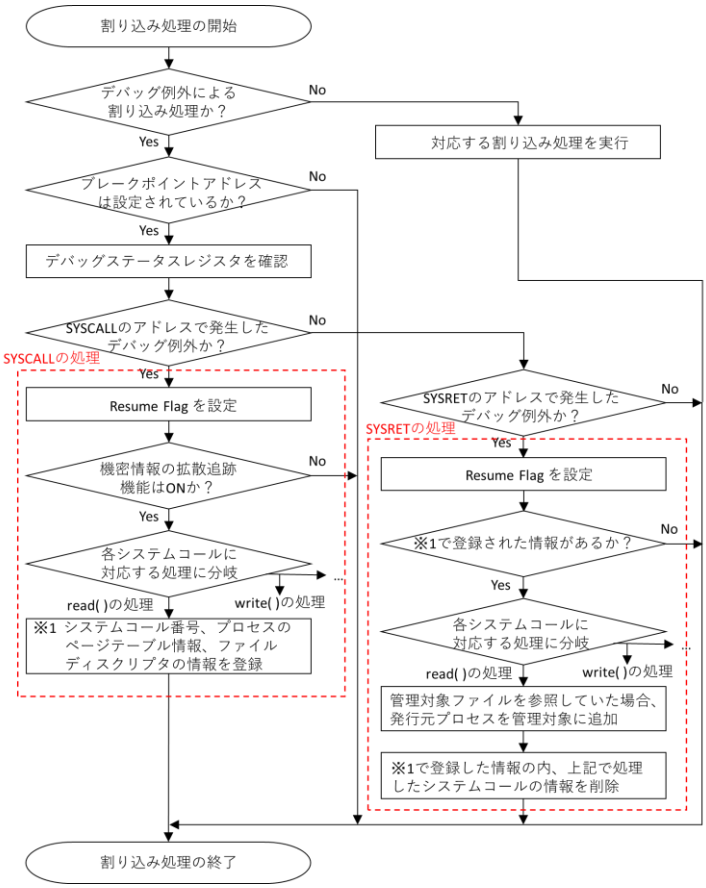

図 2　処理フロー

謝辞　本研究は JSPS 科研費 16H02829（基盤研究(B)）の助成を受けたものです．

参考文献

[1] Fujii, S., Sato, M., Yamauchi, T., and Taniguchi, H.: Evaluation and Design of Function for Tracing Diffusion of Classified Information for File Operations with KVM, The Journal of Supercomputing, Vol.72, Issue 5, pp.1841-1861 (2016).