2ZA-07

KVMを利用した機密情報の拡散追跡機能における ファイルパス取得処理削減の評価

荒木　涼†　　森山　英明†　　山内　利宏‡

†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．管理者の誤操作や外部からの不正アクセスによる機密情報を保有するファイルの外部への拡散を検知するために，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が提案され，実現されている[1]．この機能では，機密情報を保有するファイルを操作するシステムコールをフックして情報を取得することで，ファイルの拡散を検知し，利用者に機密情報の拡散経路を通知することを可能としている．

本稿では，KVM(Kernel-based Virtual Machine)を用いた機密情報の拡散追跡機能について，ファイルパス取得処理を削減した際の評価結果について述べる．

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，VMMを用いた機密情報の拡散追跡機能を提案している[1]．機密情報の拡散追跡機能では，機密情報を保有してい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)を登録し，管理する．この機能をVMM上に実装することにより，機能への攻撃が困難となり，堅牢なシステムを実現している．

KVMにおける機密情報の拡散追跡機能では，仮想計算機上で発行されるシステムコールをフックするために，ハードウェアブレークポイントを用いてシステムコール発行前(SYSCALL命令)と終了直前(SYSRET命令)でデバッグ例外を発生させる．これにより，仮想計算機上の処理から拡散追跡機能の処理へ移行する．VMM側では，SYSCALL命令とSYSRET命令のどちらの実行前にデバッグ例外が発生したかを確認する．次に，フックされたシステムコールの番号から，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を判定する．このとき，機密情報の拡散に関係するシステムコールの際は，SYSCALL命令実行前の拡散追跡処理として，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，および扱うファイルのファイルディスクリプタの値などを取得する．また，SYSRET命令実行前の拡散追跡処理では，システムコール処理の成否，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取得する．もし，機密情報の拡散に関係しないシステムコールの場合は，拡散追跡にともなう処理を行わず，システムコール処理を続行する．

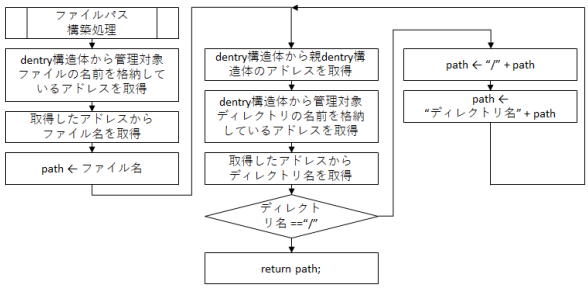

図1 ファイルパス取得処理

2.2 ファイルパス取得処理

機密情報の拡散追跡機能では，管理対象ファイルの情報を取得する際，管理対象ファイルの位置情報(以降，ファイルパス)を仮想計算機上から取得する．このファイルパスの取得処理は，管理対象プロセスからwriteシステムコールが発行され，新たに管理対象ファイルの登録される際に行われる．具体的には，writeシステムコールのSYSRET命令実行前の拡散追跡処理において，ファイルパスを取得する．

ファイルパス取得処理を図1に示す．まず，管理対象ファイルのファイル名を取得する．この際，ファイル名やディレクトリ名を取得するために，仮想ファイルシステム(VFS)の機能の一つであり，ファイル名やディレクトリ名，および親dentryアドレスを保有しているデータ構造であるdentry構造体にアクセスする．ファイル名の取得では，ファイル名を格納しているアドレスをdentry構造体から取得し，このアドレスからファイル名を取得する．次に，カレントディレクトリ名を取得するために，同じdentry構

Evaluation of Reduction of Processing for Detecting File Path in 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 測定環境

測定用計算機	
CPU	Intel Xeon CPU E5-2620 v4@2.10GHz
コア数	8 個
メモリ	32GB
OS	Fedora18(Linux Kernel 3.6.10)(64bit)
VMM	kvm-kmod-3.6
仮想計算機	
コア数	1 個
メモリ	2GB
OS	Fedora18(Linux Kernel 3.6.10)(64bit)

造体から親 dentry 構造体のアドレスを取得し，この親 dentry 構造体からディレクトリ名を格納しているアドレスを取得する．このアドレスからディレクトリ名を取得する．このカレントディレクトリ名の取得をルートディレクトリまで行うことで，ファイルパスを構築し，取得する．

機密情報の拡散追跡機能におけるファイルパス取得の処理時間は，文献[2]において評価している．文献[2]において，ファイルパス取得処理は，管理対象プロセスが write システムコールを発行した際に実行される機密情報の拡散追跡処理の全体の処理時間の内，約 37.0%を占めていることを明らかにしており，ファイルパス取得処理によるオーバヘッドが大きいことを示している．以降では，ファイルパス取得処理を削除することにより，機密情報の拡散追跡機能の処理時間をどの程度低減できるか検討する．

3. ファイルパス取得処理削減による評価

3.1 評価の概要

測定環境を表 1 に示す．評価では，以下の 2 つの場合における機密情報の拡散追跡機能を導入して測定を行い，結果を比較することで，ファイルパス取得処理の有無による処理時間の違いを明確化する．

(1) ファイルパス取得処理を行う場合
(2) ファイルパス取得処理を行わない場合

また，測定は，管理対象ファイルに対して，ファイルを複製するプロセスを実行することで，機密情報の拡散追跡機能における write システムコールにおける処理時間の測定を行う．具体的には，write システムコールを発行した際の SYSCALL 命令実行前の拡散追跡処理と SYSRET 命令実行前の拡散追跡処理のそれぞれの処理時間を測定した．

3.2 write システムコールの実行時間

write システムコールにおける処理時間を測定した結果を表 2 に示す．表 2 より，write システムコールの SYSRET 処理において，ファイルパス取得処理を行わない場合の処理時間は，ファイルパス取得処理を行う場合の処理時間と比較すると，約 15600 clock 減少しており，これは，write システムコールの SYSRET 処理において，約 41.15%の処理時間の削減となる．write システムコールの SYSCALL 処理において比較すると，30 clock 減少しており，これは， write システムコールの SYSCALL 処理において，約 9.43%の処理時間の削減となる．

表 2 write システムコールフック時の拡散追跡機能の処理時間（clock）

	ファイルパス取得処理を行う場合		ファイルパス取得処理を行わない場合	
	SYSCALL	SYSRET	SYSCALL	SYSRET
処理時間	318.1	37901.7	288.1	22303.4

また，write システムコール全体の処理において比較すると，約 15630 clock 減少しており，これは，SYSCALL 時と SYSRET 時を含む，write システムコール発行時における，機密情報の拡散追跡機能の処理全体の約 40.89%の処理時間の削減となる．write システムコール全体の処理においても約 40.89％削減できているのは，SYSCALL の処理時間が SYSRET の処理時間に比べ非常に小さいためである．以上の結果より，ファイルパス取得処理の削除により，機密情報の拡散追跡機能において，管理対象ファイルの登録処理を含む場合における処理の高速化が可能であると言える．

このとき，inode 番号の情報から，管理対象ファイルを登録し機密情報の拡散追跡を行うことが可能である．

4. おわりに

機密情報の拡散追跡機能におけるファイルパス取得処理を削除することで，write システムコールにおいて約 40.89%の処理時間の削減が可能であることを，測定により明らかにした．今後は，機密情報の拡散追跡機能における拡散追跡処理外でのファイルパス取得手法について検討する．

謝辞 本研究の一部は 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).

[2] 森山英明，山内利宏，佐藤将也，谷口秀夫，“KVM を利用した機密情報の拡散追跡機能におけるファイルアクセス性能の評価，”第 17 回情報科学技術フォーラム(FIT2018)講演論文集，第 1 分冊，pp. 147-148 (2018-9-12)．