6A-05

機密情報の拡散追跡機能におけるタイムリーな管理対象把握法

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 創造工学科†　　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，外部へ漏えいすることで，企業や個人にとって大きな損失となる．この問題への対処として，KVM による仮想計算機環境を利用し，機密情報の外部への拡散を検知し追跡する機能を提案した．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情報を取得することで，拡散経路の把握を可能としている．一方，機密情報の拡散追跡機能は，機密情報の拡散の検知を契機として拡散経路をログに出力するため，任意の時点における拡散経路の把握が難しい．

本稿では，機密情報の拡散追跡機能について，任意の時点においてタイムリーに拡散経路を把握する手法について，検討した結果を述べる．

2. KVM における機密情報の拡散追跡機能

2.1 拡散追跡処理の流れ

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（VMM: Virtual Machine Monitor）を用いた機密情報の拡散追跡機能（以降，拡散追跡機能）を提案している．また，KVM（Kernel-based Virtual Machine）上に機能を実現し，評価結果を報告した[1]．拡散追跡機能では，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 上に実現することで，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．

拡散追跡機能における拡散追跡の処理を，図 1 に示す．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（図 1 の(1)）．これにより，処理をゲスト OS から VMM へ移行させる（図 1 の(2)）．VMM 側では，ハードウェアブレークポイントによるデバッグ例外であることを確認し，SYSCALL 命令と SYSRET 命令のどちらの実行前に発生したものか確認する．次に，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を判定する．機密情報の拡散に関係するシステムコールの場合は，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，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(図 1 の(2-B))．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．これらの情報を基に，機密情報の拡散経路の情報を，ログとして保存し（図 1 の(3)），ゲスト OS 側に処理を移行して，システムコール処理を続行する（図 1 の(4)）．もし，機密情報の拡散に関係しないシステムコールをフックした場合は，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 の(2-A)）．

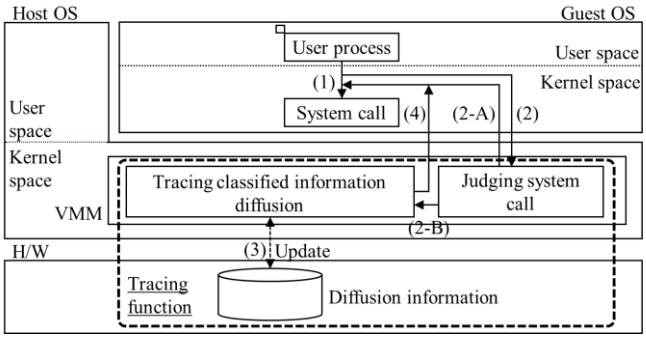

図 1 機密情報の拡散追跡機能

2.2 ログ出力処理における問題と対処

現状の機密情報の拡散追跡機能において，拡散経路として出力されるログの例を図 2 に示す．図 2 は，機密情報を含むファイルとして /root/secret.txt というファイルを登録し，このファイルに対して，cp コマンドによるファイルの複製を 5 回行った際に生成される拡散経路のログである．拡散追跡機能では，新しく管理対象プロセスやファイルを検知する度に，拡散追跡機能の起動時から取得したすべての管理対象プロセスとファイルの拡散経路をシステムログとして/var/log/messages に出力する．図 2 の例では，cp コマンドを 5 回実行した際に，管理対象プロセスとして取得した PID=773，774，775，776，777 のプロセス情報と管理対象ファイルと

Method of Timely Detecting for Tracing Diffusion of Classified Information

† National Institute of Technology, Ariake College

‡ Graduate School of Natural Science and Technology, Okayama University

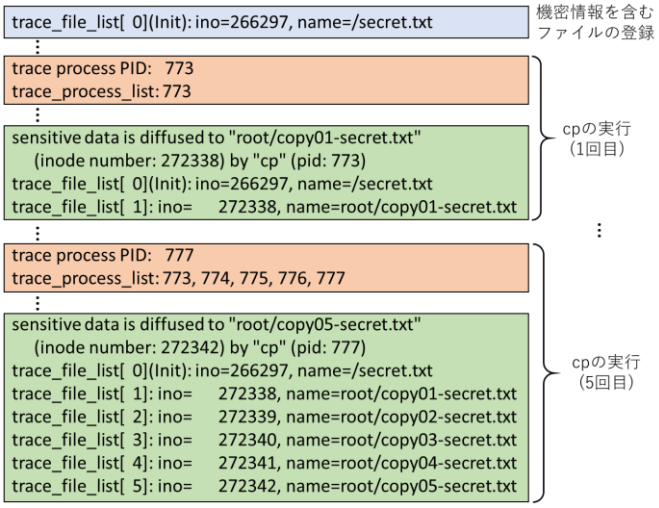

図 2 機密情報の拡散経路のログ

して取得した copy01-secret.txt から copy05-secret.txt のファイル情報が出力される．この手法には，以下の利点がある．

(利点 1) 新たに検知した管理対象プロセスとファイルを即座に把握することができる．

(利点 2) システムログ上の最後に出力されている機密情報の拡散経路のログを確認することで，すべての拡散経路を確認できる．つまり，システムログを遡って確認する必要がない．

一方，以下の問題がある．

(問題 1) 新たに管理対象プロセスやファイルを検知する度に，これまで検知した管理対象プロセスとファイルをすべて出力するため，出力処理によるオーバヘッドが大きい．

(問題 2) 管理対象プロセスの終了時や管理対象ファイルの削除時に，これらのプロセスやファイルが管理対象から外れることを，機密情報の拡散経路のログとして出力しない．このため，拡散経路のログを確認する際に登録されている管理対象プロセスとファイルを，タイムリーに把握することができない．

3. 対処

(問題 1)への対処として(対処 1)を行う．

(対処 1) 新たに管理対象プロセスやファイルを検知する際に，新たに検知した管理対象プロセスやファイルのみを，機密情報の拡散経路のログとして，システムログに出力する．これにより，オーバヘッドを低減する．

また，(問題 2)への対処として(対処 2)と(対処 3)を行う．

(対処 2) 拡散追跡機能の起動時から取得した管理対象プロセスとファイルの拡散経路情報を，システムログから抽出し，利用者に一括して表示する機能（以降，一括ログ出力機能）を実現する．

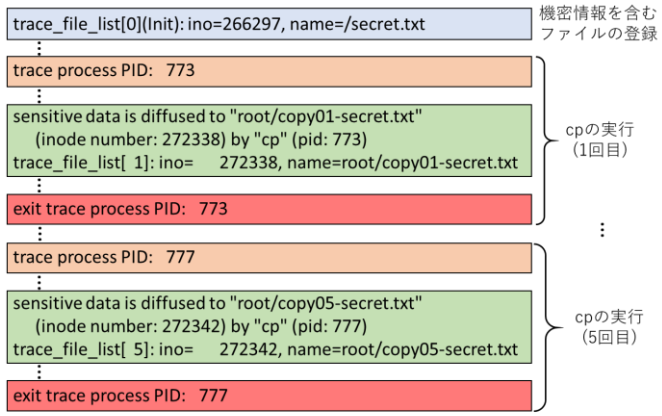

図 3 対処後における機密情報の拡散経路のログ

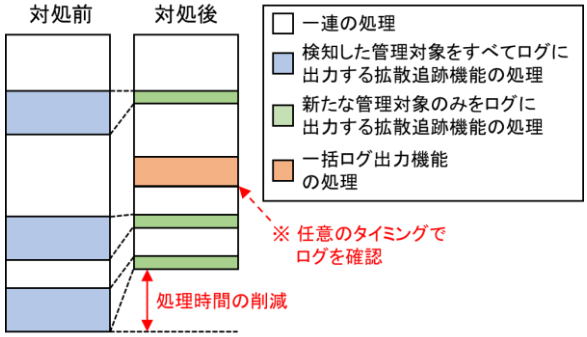

図 4 ログ出力処理削減による効果

(対処 3) 管理対象を外れたプロセスファイルを即座に表示する．これにより，タイムリーな把握が可能となる．

機密情報の拡散経路のログについて，(対処 1)から(対処 3)を適用した際の例を図 3 に示す．5 回目に cp コマンドを実行した際に，新たに管理対象として検知した PID=777 のプロセスの情報のみが表示されており，PID=777 の管理対象プロセスが終了したことを拡散経路のログとして表示されている．また，対処により期待される効果を図 4 に示す．一括ログ表示機能の処理時間を短く実現することで，機能を減らすことなくオーバヘッドの削減が可能であると考える．

4. おわりに

本稿では，機密情報の拡散追跡機能について，拡散経路のログを取得する処理における問題点を明確にし，任意の時点においてタイムリーに拡散経路を把握する手法を検討した結果を述べた．残された課題は，提案手法の評価である．

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).