B-003

機密情報の拡散追跡機能における procfs を用いた管理対象の把握および変更手法の評価
Evaluation of a procfs-Based Method for Listing and Modifying Managed Targets in a Function for Tracing the Diffusion of Classified Information

森山 英明 * 山内 利宏 † 佐藤 将也 ‡ 谷口 秀夫 §
Hideaki Moriyama Toshihiro Yamauchi Masaya Sato Hideo Taniguchi

1. はじめに

計算機内で管理されている機密情報が外部に漏えいすると，企業や個人の大きな損失となる．これに対し，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（以降，拡散追跡機能と略す）を提案し，KVM（Kernel-based Virtual Machine）上で実現した[1]．この機能は，機密情報を操作するシステムコールを VMM が検知し解析することで，機密情報を有する可能性のあるプロセスやファイル（以降，管理対象プロセスと管理対象ファイルと呼ぶ）の情報をログとして出力する．ここで，ホスト OS 上で拡散追跡機能を用いて監視を行う者（以降，利用者と呼ぶ）は，現在の管理対象プロセスとファイルの一覧の取得，および追加・削除を行う必要がある．本稿では，procfs（Process Filesystem）を用いて実現したこれらの機能[2][3]について，評価結果を報告する．

2. 機密情報の拡散追跡機能

機密情報の拡散は，あるプロセスが管理対象ファイルの内容を読み込み，他のプロセスやファイルに内容を伝えることによって発生する．拡散追跡機能は，これらのプロセスやファイルを管理対象とし，拡散情報として出力する．拡散追跡機能の全体図を図 1 に示し，以下で説明する．

(1) 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．
(2) VMM 上でシステムコールの発行を検知し，発行されたシステムコールを判定する．
  (A) 機密情報の拡散に関係しないシステムコールの場合，制御をゲスト OS へ戻し，システムコールの処理を続行する．
  (B) 機密情報の拡散に関係するシステムコールの場合，機密情報の拡散追跡に必要な情報を取得し，取得した情報をもとに拡散情報を更新する．
(3) （2-B）で新たな管理対象プロセスや管理対象ファイルを検知した場合は，取得した情報をシステムログ（`/var/log/messages`）に出力する．
(4) 制御をゲスト OS へ戻す．

＊有明工業高等専門学校 創造工学科
Department of Creative Engineering, National Institute of Technology, Ariake College
†岡山大学 学術研究院 環境生命自然科学学域
Faculty of Environmental, Life, Natural Science and Technology, Okayama University
‡岡山県立大学 情報工学部
Faculty of Computer Science and Systems Engineering, Okayama Prefectural University
§岡山大学大学院 環境生命自然科学研究科
Graduate School of Environmental, Life, Natural Science and Technology, Okayama University

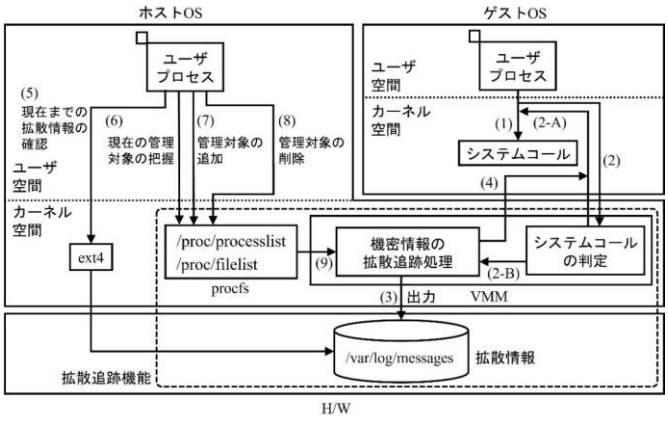

図 1　KVM を用いた拡散追跡機能

この実装により，ゲスト OS のソースコードを改変することなく機密情報の拡散を追跡することができる．また，利用者が`/var/log/messages` を参照することで，検知したすべての拡散情報を確認することができる（図 1 の(5)）．

3. procfs を用いた管理対象の把握および変更手法

拡散追跡機能が現在管理している管理対象プロセスとファイルを把握するために，procfs を用いた管理対象の把握手法を提案している[2]．この手法では，現在の管理対象プロセスとファイルの一覧を管理するために，仮想ファイルとして`/proc/processlist` と`/proc/filelist` をそれぞれ作成する．これらの仮想ファイルに対して，ユーザプロセスからそれぞれ読み込みを行うことで（図 1 の(6)），拡散追跡機能から現在管理している管理対象プロセス，または管理対象ファイルの一覧を取得し（図 1 の(9)），各仮想ファイルへ出力する．これらの処理は，各仮想ファイルのファイル操作構造体の読み込み処理としてカーネルモジュール上で実装している．

また，`/proc/processlist` と`/proc/filelist` を介することで，利用者から管理対象プロセスとファイルの追加と削除を行う機能を実現している[3]（図 1 の(7)と(8)）．管理対象プロセスの追加や削除は，`/proc/processlist` に追加・削除の操作種別，PID，プロセス名の情報を書き込むことで実行できる．同様に，管理対象ファイルの追加や削除は，`/proc/filelist` に追加・削除の操作種別，inode 番号，ファイルの絶対パス名を書き込むことで実行できる．利用者から，各仮想ファイルへこれらの情報の書き込みが行われる際に，拡散追跡機能は情報を解析し，管理対象プロセスとファイルの情報を更新する．これらの処理は，各仮想ファイルのファイル操作構造体における書き込み処理として，カーネルモジュール上で実装している．

拡散追跡機能において，管理対象プロセスは PID をインデックスとした 32,768 個の要素を持つ配列で管理されている．管理対象プロセスを新たに検知すると，この配列の対応するインデックスの値を 1 に設定する．一方，管理対象

表 1　測定環境

CPU	Intel Xeon Processor E5-2609 (8 コア)
メモリ	64 GB
OS	Fedora 18 (Linux kernel 3.6.10, 64 bit)
VMM	KVM-kmod-3.6

ファイルは，inode 番号とファイルの絶対パス名をメンバとして持つ構造体の配列で管理されており，新たに検知するたびに，この配列の末尾にエントリを追加する．

4. 評価

procfs を用いた管理対象の把握手法，および管理対象の追加・削除による変更手法について，以下の 3 項目より評価する．本評価の測定環境を表 1 に示す．

(評価 1) 現在の管理対象プロセス，または管理対象ファイルの一覧を仮想ファイルへ出力する処理時間
(評価 2) 管理対象プロセスの追加・削除処理の処理時間
(評価 3) 管理対象ファイルの追加・削除処理の処理時間

処理時間の測定は，測定対象の処理の前後で rdtscll 命令によるタイムスタンプを取得して差分を求め，10 回測定した平均の clock 数と CPU の動作周波数から算出する．

(評価 1)として，管理対象プロセス，または管理対象ファイルの一覧を仮想ファイルへ出力する処理時間を測定した結果を図 2 に示す．これらの処理は，扱う管理対象情報の数により増減するため，1 から 100 まで段階的に変更して処理時間を測定している．図 2 より，管理対象プロセス数 1 の際の処理時間は 67.92 μs であり，管理対象プロセス数が増加するにつれ処理時間は緩やかに増加し，100 の際は 106 μs となる．管理対象ファイルを対象とした出力処理では，管理対象ファイル数 1 の際は 3.84 μs，100 の際は 45.27 μs となり，ほぼ線形に処理時間が増加する．管理対象の違いによる処理時間の差異は，管理対象プロセスの一覧の取得において，PID をインデックスとする配列の走査処理が含まれることが原因と考える．管理対象の一覧の出力処理は，利用者が任意の契機で行うもので頻度は高くないことから，拡散追跡機能へ与える影響は小さい．

(評価 2)として，管理対象プロセスの追加と削除の処理時間を測定した結果を図 3 に示す．(評価 1)と同様に，扱う管理対象プロセス数を 1 から 100 まで段階的に変更して測定する．削除処理では，あらかじめ管理対象プロセスを 100 個登録し，削除数を 1 から 100 まで段階的に増やして処理時間を測定する．図 3 より，追加の処理時間は 4.11 μs から 351.5 μs へ線形に増加し，削除の処理時間は 3.52 μs から 349.97 μs へ線形に増加する．

(評価 3)として，管理対象ファイルの追加と削除の処理時間を測定した結果を図 4 に示す．扱う管理対象ファイル数を 1 から 100 まで段階的に変更して測定する．図 4 より，追加の処理時間は 4.34 μs から 414.5 μs へ線形に増加し，削除の処理時間は 4.1 μs から 421.1 μs へ線形に増加する．(評価 2)と(評価 3)より，管理対象プロセスとファイルを 1 個追加，または削除する処理時間は 3.5 μs から 4.4 μs となり，拡散追跡機能へ与える影響は小さい．

5. おわりに

機密情報の拡散追跡機能において，procfs を用いた管理対象プロセスとファイルの一覧取得機能，および追加・削除機能の評価を行った．管理対象プロセス一覧の出力処理では，配列の走査処理を伴うため，管理対象ファイル一覧の出力処理と比較して，処理時間が長くなることを確認した．一方，管理対象の追加・削除の処理時間は，1 エントリあたり約 4 μs であった．これらの結果より，拡散追跡機能へ与える影響は小さい．

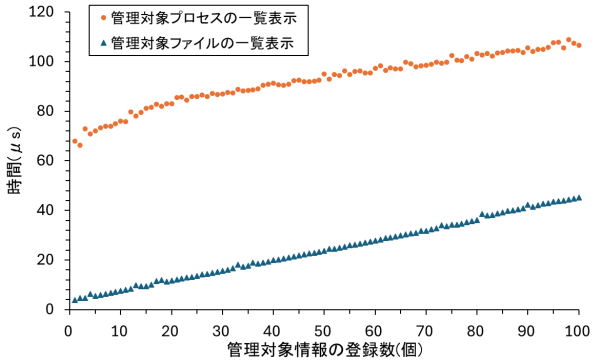

図 2　管理対象プロセス，または管理対象ファイルの一覧を仮想ファイルへ出力する処理時間

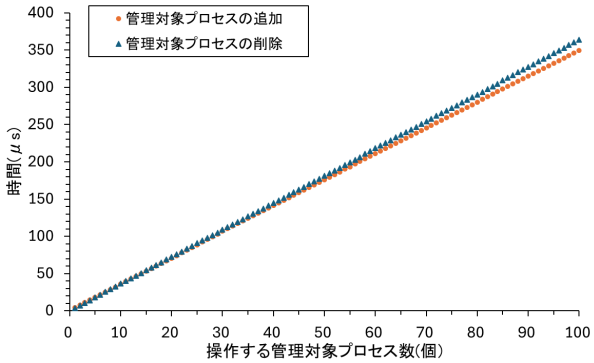

図 3　管理対象プロセスの追加と削除の処理時間

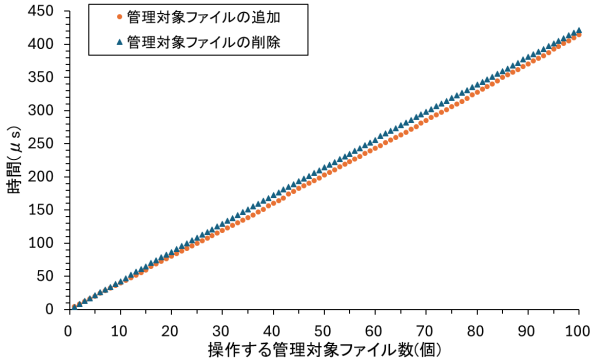

図 4　管理対象ファイルの追加と削除の処理時間

謝辞　本研究の一部は，JSPS 科研費 JP23K24848，JP25K03119 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, No. 5, pp. 1841–1861(2016).
[2] 森山英明，山内利宏，佐藤将也，谷口秀夫，“機密情報の拡散追跡機能における proc ファイルシステムを用いた管理対象把握手法の検討，”電気関係学会九州支部連合大会講演論文集，vol.2023，pp.357-358(2023).
[3] 森山英明，山内利宏，佐藤将也，谷口秀夫，“拡散追跡機能における procfs を用いた管理対象の動的追加・削除機構の提案，”電気・情報関係学会九州支部連合大会講演論文集，vol.2024，pp.63-64 (2024).