Vol.2010-GN-75 No.4
2010/3/18

USB メモリを介したファイル移動の監視とそのログ視覚化

小崎真寛[†1]　芝口誠仁[†2]
中山佑輝[†2]　岡田謙一[†2]

情報化社会の発展に伴い，可搬記憶媒体からの情報漏洩が問題化している一方で，企業側の対策は暗号化やパスワードロックなどの機能を有した USB メモリに使用を制限する，もしくは可搬記憶媒体の使用を全面的に禁止するなどの措置がとられている場合が多い．しかしこのような措置は，USB メモリの使用を制限することで生じる可能性のある社内の生産性・利便性の低下といった副作用に比べ，セキュリティという観点ではそれに見合う効果が得られるかについては疑問の余地がある．本稿では，USB メモリの操作ログを取得した後にログを視覚化することで，管理者による迅速な監視とフィードバックを実現した．評価実験の結果，USB メモリの利便性を確保しつつもセキュリティを維持する対策に貢献できた．

Monitoring File-Migration with USB flash drive and Visualization of its Log

Masahiro Kozaki,[†1] Seiji Shibaguchi,[†2]
Yuki Nakayama[†2] and Ken-ichi Okada[†2]

In this paper, we describe a method of monitoring and visualization of USB flash drive's log information. Recently, with rapid progress toward an information society, the increasing damage caused by information leakage via USB flash drive has been concerned about. Corporate security against it is to keep employees to use a USB memory with password lock and encryption feature or to keep employees from using removable media. But as compared with reduction of productivity and convenience, it is doubtful that we can get improves of security. Therefore, we aim to develop a method of method of monitoring file-migration with USB flash drive and visualization of its log. Lastly, through an experiment we verify that our mothod is effective.

1. はじめに

情報化社会の発展に伴ない，近年 USB メモリは急激な低価格化・大容量化・小型化が進んでいる．半導体の世界では 18ヶ月で集積密度が倍増するというムーアの法則[1)] が知られており，これは単なる経験則であるがゴードン・ムーア（Intel 共同創始者）が 1965 年に提唱してから現在までのところほぼ同法則に従って集積密度は増加しており，この傾向は今後も続くとみられている．

このように USB メモリはデータの持ち運びという点で利便性が高く，その需要は今後も大きいと見込まれる一方で情報漏洩といった問題が顕在化しつつある．価格の低下により誰でも気軽に持つことが可能になり，小型化したことで盗難や紛失の危険が増した．大容量化したことは企業情報や顧客の個人情報といった機密情報の大量漏洩につながりやすい．また 2005 年に完全施行された個人情報保護法きっかけとして情報漏洩の事件が注目されるようになり，同法に対応すべく企業側の組織体制が整備されるようになった．

こうした状況をうけ，セキュリティの強化が重要となり企業の危機管理やガバナンスが説かれる中，その場合のガバナンスとは単に法律を順守することや情報の機密性を強めることを意味していることが多い．特に日本企業には個人情報保護にみられるように，費用対効果を考えないで「完璧」を求める傾向がある[2)]．企業内で USB メモリもハードディスクも使えないというような状況は，その弊害と言えるだろう.[3)4)5)] Bruce Schneier は，その著書の中でセキュリティの本質はトレードオフであり，経済的な問題であると指摘している.[6)]

また，仮に企業内で USB メモリの使用を許可していたとしても，その情報漏洩対策は暗号化やパスワードロックなどの機能を有した USB メモリに使用を制限する，もしくは可搬記憶媒体の使用を全面的に禁止するなどの措置がとられている場合が多い．したがって情報が漏洩した場合に，どの USB メモリからどういった情報が漏洩したのかという点において追跡が難しいうえに，情報の漏洩を全て防ぎきるものではない．

本稿は，監視プログラムを USB メモリ内に仕込み，WindowsAPI をフックすることで USB メモリの操作ログ (USB メモリが挿されたホストの Mac アドレス，IP アドレス，挿入されていた時間，ファイルの移動などの情報）を取得し，ログを各 USB メモリから集約

†1 慶應義塾大学理工学部
Faculty of Science and Technology, Keio University
†2 慶應義塾大学大学院理工学研究科
Graduate School of Science and Technology, KeioUniversity

した後に視覚化することで，管理者による迅速な監視とフィードバックを目指すものである．これによって，従来の USB メモリから情報が漏洩することを防ぐことを目的としながらも，結果として利便性の低下を招いていたセキュリティ対策ではなく，利便性を確保しつつセキュリティを維持することに貢献できる．

以降，2 章で関連研究を紹介し，3 章でログの取得とその視覚化，システムアーキテクチャなどについて詳述する．4 章では実際にテキストログと視覚化ログとの比較を評価実験として行った．最後に 5 章で本稿のまとめとする．

2. 関 連 研 究

2.1 APIHook を用いた USB メモリからの情報漏洩対策システム

本研究は WindowsAPI をフックすることで USB メモリ－ホスト間のファイル移動を監視し，ログを取得した後視覚化する．そこで APIHook を用いた USB メモリからの情報漏洩対策，および情報の視覚化に関する関連研究を紹介する．

日本ネットワークセキュリティ協会は，漏洩媒体・経路比率の「USB 等可搬記憶媒体」の割合が増加していること，情報漏洩の原因として誤操作というものの割合が一番多くなっていることを報告している．そこで，古澤ら[7] は USB メモリからの情報漏洩に着目し，誤操作による情報漏洩を防ぐことを可能とする情報漏洩対策システムの開発を行った．これは USB メモリのファイルを USB メモリ外に持ち出すことを WindowsAPI をフックすることで防ぎ，また許可されたソフトウェアのみが USB メモリ内のファイルを編集可能とするシステムであり，ユーザが誤って機密情報を外部 PC に漏洩してしまう事態を防止する．しかしこのシステムは，対策システムが導入された PC 以外の PC では対応することができず，情報漏洩を防ぐことはできない．またこのように USB メモリの使用を制限することは利便性・生産性の低下に繋がってしまう．

2.2 事前対策と事後対策

上記で述べた関連研究は，事前対策に着目した一例として挙げることができる．これは事前に何らかの対策を導入することで情報の漏洩を未然に防ぐことを目的としたものだが，利便性や生産性が落ちてしまうという問題点があった．また対策システムを施したからといって事前に全ての脅威を防ぐことは不可能である．そこで各種操作をログで取っておき，万が一の際にログを解析し，犯人特定・復旧などを行う「事後対策」の観点からセキュリティを高める考え方がある．[8] 裁判などの法的な場においては電磁気的記録を物的証拠の解析結果として専門知識のない人にも提示する必要があり，近年頻繁に議論されているデジタルフォレンジックとして注目されている．

2.3 情報視覚化技術

セキュリティの分野においても，情報の視覚化とは膨大で多様なテキストログの解析を支援するために用いられる．ログ情報は基本的に大量に発生することがほとんどであり，ログから今現在自分の欲しい情報を抽出したり，また特徴・傾向をつかんだりする場合，文字情報のログでは非常に複雑であり理解が困難である．そのため解析に非常に時間がかかるという問題点がある．このような問題を解決するのが視覚化である．そこでネットワークを経由した不正アクセスを防ぐことを目的として，小池ら[9] はネットワーク型不正侵入検知システム（Network-based Intrusion Detection System:NIDS）のログを情報視覚化技術によって管理者が即座に状況を把握できるシステムを構築した．NIDS は不正アクセスの検知に有効なシステムであるが，NIDS がもたらす誤検知（False Positive）の扱いが困難であることが問題であり，視覚化によって NIDS の調整作業の負担を削減することを目的としている．この研究では NIDS のログに絞って視覚化が行われていたが，本研究ではより USB メモリの使用履歴が分かりやすいようにファイル操作・挿入先ホストの情報・ホストへの挿入時間などのログ情報を対象とすることに主眼を置いた．

3. ログの取得とその視覚化

3.1 提案のコンセプト

1 章でも書いたように，本提案は監視プログラムを USB メモリ内に仕込み，WindowsAPI をフックすること（APIHook）で USB メモリの操作ログを取得し，その後視覚化することで，管理者による迅速な監視とフィードバックを目指すものである．これによって，利便性を確保しつつもセキュリティの維持に貢献する．

3.2 システム全体像

APIHook によって USB メモリ－ホスト間のファイル移動を監視し，ログを取得する．あらかじめ監視プログラムを USB メモリ内に仕込んでおき，ユーザが USB メモリを使用するたびにファイルを操作した情報を監視し，そのログをテキストデータとして保存する．その後，各 USB メモリ内のログを集約して視覚化することで管理者による解析を支援する（**図 1**）．これによって組織全体で USB メモリを介してファイルがいつ企業外部へ移動したのかを直感的に把握できるように表示することが可能となる．

3.3 想定環境と運用

本提案は企業内での運用を想定している．まず，従業員は監視プログラムを仕込んだ指定

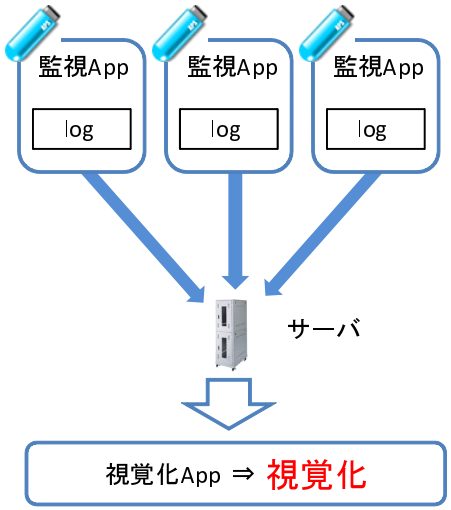

図 1 システム全体像
Fig. 1 system architecture

USB メモリに使用を限定する．管理者は，予めすべての従業員に監視プログラムの入った USB メモリを配布し，従業員はその USB メモリを使用することを仮定する．また規定の USB メモリは耐タンパ性を持つ領域を仮定し，第三者による物理的・論理的な侵食を受けることはないと仮定する．そしてログファイルの完全性に関して対象外とした．すなわち，この監視アプリケーションによって作成されたログファイルが第三者の手によって改竄されることは無いということを予め想定する．

実際の運用は，まず管理者が事前準備をしておき，各従業員が USB メモリを使用，その後視覚化という流れなる．管理者が行う事前準備として，

（1） 予め安全なホストと安全でないホストを規定する
（2） 予め機密ファイルを登録しておく（監視定義ファイル）

がある．安全なホストとは，エンタープライズ内のコンピュータおよび従業員が日常的に使用する家庭用コンピュータ（一台）とする．管理者はこれらのコンピュータの Mac アドレスを予め取得し，データベースとして保存しておくものとする．USB メモリが取得した Mac アドレスがこのデータベースと照合して一致すれば，安全なホストに接続されていたということが分かる．照合して一致しなければ，安全でないホスト（外部のホスト）に接続されていたということである．また「機密ファイル」と「非機密ファイル」では，ファイルが流出した際に機密ファイルが流出した方がより危険であり，2 つのファイル移動は視覚化の際に区別するべきであるとの観点から，管理者は予め初期段階での企業内における機密ファイルの所在を把握しておく．

3.4 監視アプリケーション

今回の提案では APIHook を用いてファイル移動を監視する．フックは IAT（インポートアドレステーブル）の書き換えによって実現している．この APIHook によって WindowsAPI 関数に様々な機能を加えることが可能となる．

APIHook を用いたファイル監視 APIHook とは，アプリケーションが WindowsAPI を呼ぶ途中でフック（横取り）し，任意の機能を付け加えるものである．WindowsAPI とはアプリケーションが OS の各機能を実行するための接点となるものであり，各 API は DLL ファイルとして保存されている．

ここである実行ファイルがあるとする．実行ファイルのフォーマットは PE(Portable Executable) 形式と呼ばれるもので，PE ファイルではヘッダ後にネイティブコード，リソース情報，デバッグ情報など様々なデータが配置されており，これらは種類ごとにセクションとして区切られ格納されている．このセクションの一つが IAT であり，ここで外部 DLL と動的リンクを行っている．インポートセクションにはインポートされる関数のメモリアドレス格納されており，あるモジュールから外部の関数を呼び出す際には，そのモジュールの IAT を参照してアドレスを取得することによって呼び出している．よって IAT を書き換えることで API のフックを実現することが可能である．**図 2** に実際に API をフックする様子の概略図を示す．あるファイルを移動したとすると Windows の内部では WindowsAPI が呼ばれるが，これを途中でフックし必要なログを取得した後に本の関数を呼び出すという操作を行っている．具体的に疑似コードを書いて説明すると．ある API 関数 API_FUNC() があったと仮定する．

```
API_FUNC(引数 1, arg1, ...){
   // 本来の関数の機能
}
```

そして，この API_FUNC() のフック関数を HOOK_API_FUNC() とする．すると HOOK_API_FUNC() は以下のようになる．

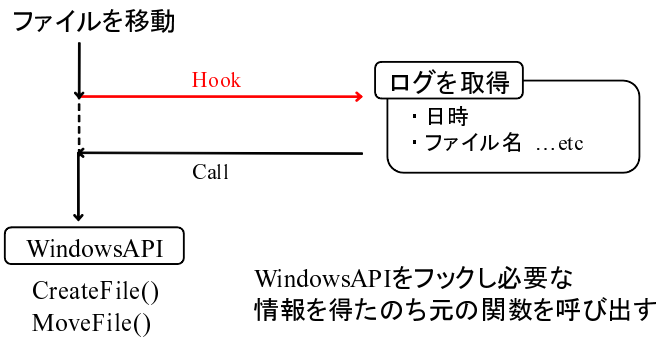

図 2 APIHook の概略図
Fig. 2 APIHook

```
HOOK_API_FUNC(引数 1, arg1, ...){
   // ここに付け加えたい機能を追加する
   API_FUNC(引数 1 arg1,...);  //最後にオリジナル関数を呼ぶ
}
```

このように実装すれば，既存のプログラムからの API 呼び出しを監視することが可能になり，またそこに新しい機能を追加，あるいは機能を変更することが可能になる．例えば，文書保存時に元文書のバックアップ機能を持たないワープロソフトにその機能を追加することや，特定のプログラムの起動を禁止したりすることなどが出来る．フックを行う際の注意としては，オリジナル関数とフック関数の引数は完全に同等でなければならない．またアプリケーションは参照先アドレスが書き換えられ，違う API 関数が呼ばれていることを知らない．したがって，最後にはオリジナル関数を呼んで，通常の処理を終えなければならない．

3.5 視覚化アプリケーション

ここでは視覚化アプリケーションについて詳述する．一般的にログ情報のフォーマットはテキスト形式であるが，文字情報の羅列であるテキストログを視覚化することの利点は，[10] の整理に従えばデータの直感的な把握を可能にする点と，大量のデータを容易に扱うことを可能にする点が挙げられる．この二点を解決するにはスケーラビリティの問題と，Small Screen Probrem と呼ばれる描画領域の大きさという物理的制約から生じる問題に取り組む必要がある．本提案が想定している企業内での使用においても，従業員の数だけ USB メモリが存在することになり，その全てを一つの画面で視覚化することは現実的でない．

そこで本提案では，まず事前準備の段階で，各 USB メモリを所持する従業員の「所属」ごとに分類し，視覚化の際にその所属に属する従業員の USB メモリを表示するようにした．具体的には，アプリケーションの「組織図のツリー」画面である部署を選択することで，その部署に所属する従業員の USB メモリに関するログがメイン画面に視覚化される仕組みとなっている．

また 2 つの View Mode（Unit View Mode/File View Mode）を搭載し，ある部署に所属する従業員全員の USB メモリに関してその使用状況の全体像を把握するモードと，ある USB メモリ一個に絞ってその USB メモリと挿されたホスト間でどのようなファイル移動があったのかについて詳細に視覚化するモードを使い分けることができる．(後述)

アプリケーションのメイン画面左にはタイムラインが表示されており，これは縦軸が時系列になっていることを示している．(**図 3**)．中央部は各 USB メモリから取得したログに基づいた視覚化がなされているが，色付きの太線は，その USB メモリがホストに挿されていた事実とその挿入時間，そして挿入先のホストが安全なホストかどうかを示している．また，各 USB メモリがどの時間帯にどのようなホストに挿されていたのかを直感的に把握できるようにした．図のグレーで表示される部分は，その時間帯に USB メモリが安全なホストに挿されていたことを示しており，赤で表示される部分は，その時間帯に安全でないホストに挿されていたことを示している．グレー・赤それぞれの領域を右クリックすることで，詳細表示画面に挿入先ホストの MacAddress や IPAddress，挿入開始時間，挿入終了時間などを表示できるようにした．

アプリケーションの 2 つの View Mode を，以下で説明する．

Unit View Mode 各 USB メモリに着目し，いつどのようなファイルが USB メモリに流入（またはホストへ流出）したのかについて視覚化する．ここでホストから USB メモリへファイルが移動することを「ファイルの流入」，また USB メモリからホストへファイルが移動することを「ファイルの流出」とする．ここでは USB メモリを中心に考えて，**図 4** のようにファイルが流入する場合は USB メモリに対して矢印が向き，ファイルが流出する場合は反対に USB メモリから矢印が出て行くようにした．

そして移動したファイルが機密ファイルかそうでないか，流出先（流入先）のホストが安全なホストかそうでないかなどによって矢印の色分けを行った．例えばあるファイルが

USB メモリからあるホストへ流出したとすると，その場合の矢印の色は**表 1** のようになる．非機密ファイルが安全なホストへ流出することは比較的危険ではないため矢印の色は黒で表示し，反対に機密ファイルが安全でないホストへ流出することは危険であるため赤で表示した．

以下，具体的な視覚化アプリケーションの読み取りについて解説する．各矢印を右クリックすることで，詳細表示画面にファイル名や流出先（流入元）ホストの情報，ファイル操作が行われた時間などを表示できるようにした．例えばある矢印を右クリックして以下のような表示を得られた場合，

ファイル名：a.txt
取得時間：3 時間 10 分　12 月 02 日　2009 年
USB_ID：usb1
送信元：00-24-A5-36-38-A9
送信先：この USB メモリ

これは，3 時 10 分に a.txt というファイルが 00-24-A5-36-38-A9 という Mac アドレスを持つホストからこの USB メモリに流入したことを示している．矢印がある場所が仮にグレーの太線内であったとすると，この 00-24-A5-36-38-A9 というホストは安全なホスト（エンタープライズ内のホストか，もしくは従業員が登録した家庭内ホスト）であることが分かる．

また，**図 5** を見ると，usb2 は 5:00〜6:00 の間にあるファイルの流出が起きたことが分かる．その時間帯に挿入されていたホストは，グレー表示であるため安全なホストである．また，下側の流出では矢印の色が青であるため，前述した表から機密ファイルに関するものだということが分かる．usb3 と usb5 を見ると両方とも赤い太線が表示されている時間帯がある．これは，安全でないホストに一定時間挿入されていたが，ファイル移動は生じていなかったことを示している．usb4 は 2:00〜7:00 までの間，ホストに挿入されていたことも，ファイル移動が起きたこともなかったことを示している．

File View Mode　USB メモリごとの各ファイルに着目し，いつどのようなファイル移動があったのかについての詳細を視覚化する．ファイルがどのホストから USB メモリへ流入し，そしてどのホストへと流出したのかという遷移状況，またどのようなファイル操作（ファイル名変更，ファイル消去，ファイル編集など）が行われたのかを知る必要がある

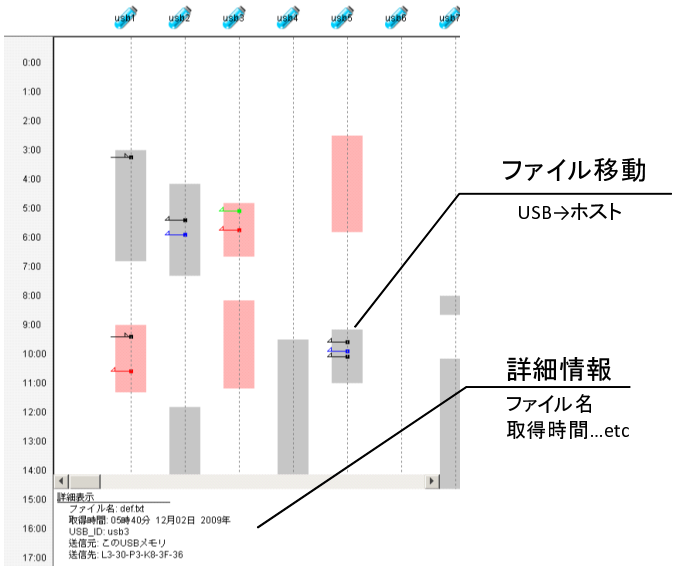

図 3　Unit View Mode
Fig. 3　Unit View Mode

図 4　矢印の向き
Fig. 4　arrow direction

ため，まず USB メモリ内に存在する全ファイルのハッシュ値を USB メモリ内の領域に取得し保存しておく．
そして実際にファイルの移動や操作が行われると，そのファイル移動ログやファイル操作ログを，ハッシュ値のデータベースと照合しながら矢印や点線等で視覚化していく．
例えば confident.pdf という機密ファイルが，ある時間にある「安全なホスト（企業内ホスト）」から USB メモリに流入し，別の時間にある「安全でないホスト（外部ホスト）」へ流出したとすると，図 5 のようになる．図 5 の左図ではある時間に安全なホストから機密ファイル confident.pdf が流入し Rename した後，ある時間に安全でないホストから CONFIDENT.pdf として流出したことを示している．また右図では同様にある時間に安全なホストから機密ファイル confident.pdf が流入し Copy された後，ある時間にある安全なホストに一方はそのまま流出，もう一方は一度編集されてから流出したことを示している．図 5 のように視覚化することで，ファイルがどのホストから USB メモリに流入し，どのホストへと流出していったかといったファイルの「伝搬経路」を直感的に把握しやすくなる．

4. 評　　価

4.1 評価の概要

視覚化がより情報漏洩の際の解析を支援するのかを確かめるためにファイルの追跡ログの視覚化アプリケーションを使った評価を行った．想定する状況はある情報が流出し，流出した情報はわかっているが，流出先ホストがわかってない状態を考える．被験者はテキスト形式のログデータを解析してもらい，次に視覚化アプリケーションを使った解析をしてもらう．比較項目は解析時間 (s) と正答率 (%) である．なお，今回のログデータはすべて手作業によって作成した．被験者は情報工学を専攻する大学生/大学院生 10 名（男 8 名，女 2 名）にお願いした．

4.2 評 価 手 順

まずテキスト形式のログデータ（10 個の USB メモリをそれぞれ一日使用したと仮定し

表 1　矢印の色の例
Table 1　arrow color

	機密ファイル	非機密ファイル
安全ホスト	←	←
非安全ホスト	←	←

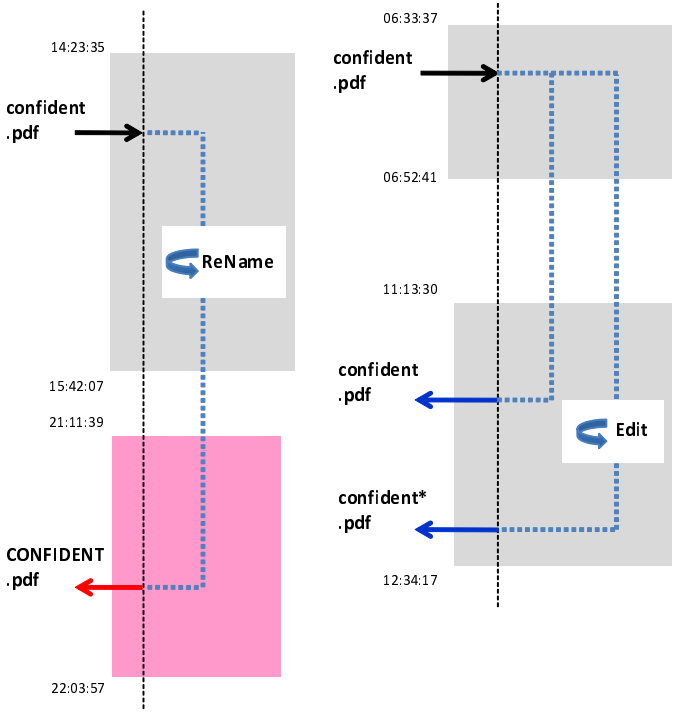

図 5　File View Mode
Fig. 5　File View Mode

たログ）を作成し，それを解析して答える質問形式の問いに答えてもらう．

問いは全部で 10 問作成した．被験者は情報工学を専攻する大学生 10 名（男 8 名，女 2 名）であり，被験者の慣れへの影響を考慮しタスクを 2 パターンに分けた．パターン A では「テキストログ→視覚化ログ」の順で解析してもらい，パターン B では「視覚化ログ→テキストログ」の順で解析してもらう．そして被験者 10 名を 5 名，5 名の 2 グループに分け，前者にはパターン A，後者にはパターン B を割り当てた．問いは，

（1） 時刻 5:20 にあるファイルが usb2 から流出または流入しました．そのファイル名・流出先のホストの Mac アドレスを調べてください．また，そのファイルは機密ファイルであるかどうか答えてください．

という形式を 6 題，また

（2） usb3 がホストに挿されていた時間と，そのホストの Mac アドレスを調べてください．また，そのホストは安全かどうかを判定してください．

という形式を 4 題作成し，それぞれ 3 題と 2 題を選び，計 5 題をパターン A，パターン B に割り当てた．

パターン A 被験者はまず実験方法の説明を受ける．この時間は解析時間に含めていない．次に被験者はテキスト形式のログデータを与えられる．その後，視覚化アプリケーションを使って解析を始めてもらう．

パターン B 被験者はまず実験方法の説明を受ける．この時間は解析時間に含めていない．次に被験者はアプリケーションを使って解析を始めてもらう．その後，テキストログデータを参照しながら問いに答えてもらう．

結果と考察 パターン A のテキストログ（正答率/解析時間）およびパターン B のテキストログ（正答率/解析時間）のそれぞれに F 検定を行ったところ，等分散であることが分かった．よって等分散を仮定した 2 標本による t 検定を行ったところ，そろぞれ $p > 0.05$ より有意差が見られなかった．またパターン A の視覚化ログ（正答率/解析時間）およびパターン B の視覚化ログ（正答率/解析時間）のそれぞれに F 検定を行ったところ，同様に等分散であることが分かった．よって等分散を仮定した 2 標本による t 検定を行ったところ，それぞれ $p > 0.05$ より有意差は見られなかった．以上から，被験者の慣れへの影響は無視出来るものとしてよい．テキストログと視覚化ログにおける各被験者の解析時間と正答率を**表 2**, **表 3** としてまとめた．また，結果をグラフ化したものを**図 6** に示す．

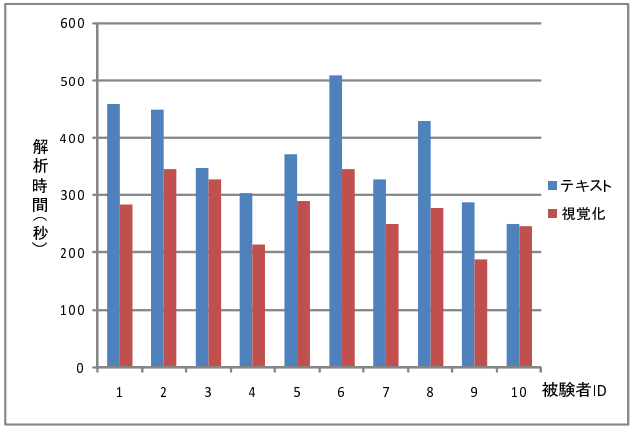

図 6 実験結果
Fig. 6 experimental result

本提案手法は情報漏洩が発覚したときに，ファイルの移動という観点から，解析を支援するツールを開発した．結果，解析時間を均約 25%削減できた．視覚化アプリケーションではインタフェースの悪さからか，若干解析に戸惑った被験者もいたようだ．特にホストの挿入時間は，スクロールバーを使って画面外の太線があるかどうかを調べる必要があったのだが，これを気付かず現在見えている範囲のみで解答してしまったケースが多く見られた．結果として平均解析時間は短縮されたが，正答率は変わらなかった．ただし，今回使用したテキスト形式のログデータは行数の少ない短いものであり，かつ 5 分程度で終わる比較的やさしいタスクであった．実際に扱われるログは非常に長い期間（1 年，2 年単位）であるためにテキストデータによる解析は数日，もしくはそれ以上を要する．その点において，少なくともより迅速に解析を行うことのできる本解析ツールは有効であると言える．

5. ま と め

情報化社会の発展に伴い，情報の記録媒体が紙から電子ファイルに移り変わってきている．またムーアの法則により，半導体の集積密度は増加し同時に価格は急激に低下した．これによって USB メモリは誰でも気軽に持ち運びが可能でかつ大容量な情報を安価に扱えるようになった反面，どこでも気軽に抜き差しできるため情報漏洩というリスクが無視出来

Vol.2010-GN-75 No.4
2010/3/18

表 2 実験結果（パターン A）
Table 2 experimental result(pattern A)

被験者 ID	Ttxt	Ptxt	Tapp	Papp
1	458	100	284	96
2	449	84	345	88
3	348	100	328	100
4	304	100	213	96
5	371	84	284	100

Ttxt: テキストログの解析時間（s）
Tapp: 視覚化ログの解析時間（s）
Ptxt: テキストログの正答率（%）
Papp: 視覚化ログの正答率（%）

表 3 実験結果（パターン B）
Table 3 experimental result(pattern B)

被験者 ID	Ttxt	Ptxt	Tapp	Papp
6	508	84	345	100
7	328	100	250	100
8	429	100	277	100
9	288	90	188	100
10	250	84	245	100

Ttxt: テキストログの解析時間（s）
Tapp: 視覚化ログの解析時間（s）
Ptxt: テキストログの正答率（%）
Papp: 視覚化ログの正答率（%）

ないようになった．そこで本稿では，USB メモリからの情報漏洩という点に着目し，USB メモリとホスト間でのファイルの移動を APIHook によって監視し，そのログを取得した後に視覚化することで，情報が漏洩した場合に管理者による迅速な解析を支援する仕組みを提案した．情報漏洩が起こったと仮定し，解析の評価を行った結果，視覚化アプリケーションを用いた場合，テキストログを用いた場合と比べて 25 %程度時間を短縮できることがわかった．

参 考 文 献

1) ムーアの法則，http://bit.ly/cDlL4e，2010 年 2 月 15 日確認.
2) 池田信夫：リスク，不確実性およびセキュリティ，
http://www003.upp.so-net.ne.jp/ikeda/iisec.pdf，2010 年 2 月 15 日確認.
3) 会社に潜む情報セキュリティの落とし穴，2010 年 2 月 19 日確認
http://www.itmedia.co.jp/enterprise/articles/0902/03/news006.html
4) 企業における USB メモリ活用ガイドライン，2010 年 2 月 19 日確認
http://www.atmarkit.co.jp/fwin2k/operation/usbmemory/usbmemory_01.html
5) 池田信夫：本当はいらない「個人情報保護法」，2010 年 2 月 19 日確認
http://ascii.jp/elem/000/000/130/130476/
6) ブルース・シュナイアー：セキュリティはなぜやぶられたのか 日経 BP 社（2007）
7) 古澤麻衣子，鈴木大輔，芦野佑樹，佐々木良一：APIHook を用いた USB メモリからの個人情報漏洩対策システムの開発，DICOMO2009，pp976-982（2009）
8) 辻井重夫，萩原栄幸（著）：デジタルフォレンジック辞典， デジタルフォレンジック研究会（2006）
9) 小池英樹，大野一広：SnortView: NIDS の誤検知判別を目的とした視覚化システム，情報処理学会論文誌，2003 Vol. 44, No.11, pp. 2757 - 2766
10) 小池英樹:情報視覚化の本質的問題点と情報セキュリティへの応用，http://www.kgt.co.jp/avsconso/event/vc13/summary/data/2-2.pdf，2010 年 2 月 18 日確認