情報処理学会論文誌 Vol.57 No.2 597–610 (Feb. 2016)

重要情報へのファイルアクセス失敗挙動に基づく情報探索型マルウェア検知手法

田辺 瑠偉[1,a)] 笠間 貴弘[3] 吉岡 克成[2] 松本 勉[2]

受付日 2015年6月1日, 採録日 2015年11月5日

概要：近年，感染先マシン内に保存されている重要情報を狙うマルウェアによる被害が増加しており，対策が求められている．マルウェア作成者は，盗み出す情報が多いほどより大きな利益を生み出すことができるため，多様な情報の収集を試みる．しかし，マルウェアがアクセスする情報が感染先マシン内に保存されているとは限らず，ファイルアクセスに失敗する場合がある．そこで本稿では，重要情報の探索時に発生するファイルアクセス失敗挙動に基づく情報探索型マルウェア検知手法を提案する．一般に，ファイルアクセス失敗挙動は正規プロセスからも発生するため，マルウェア検体を動的解析する際にファイルアクセスログを収集し，重要情報へのファイルアクセス失敗挙動を調査することで，情報探索型マルウェアを検知するためのシグネチャを作成する．評価実験では，情報漏洩を行うことが予想される実マルウェア検体に対して提案手法を適用し，情報探索型マルウェアを検知できることを示す．また，正規ユーザが利用しているマシンに対して提案手法を適用したところ，正規ユーザのログとマルウェアのファイルアクセス失敗ログには違いが見られ，情報漏洩の有無を判断する閾値を調整することで検知能力を保ったまま誤検知を十分に小さくできることを確認した．

キーワード：マルウェア対策，情報漏洩

Detecting Credential Search Focused on File Access Failure

Rui Tanabe[1,a)] Takahiro Kasama[3] Katsunari Yoshioka[2] Tsutomu Matsumoto[2]

Received: June 1, 2015, Accepted: November 5, 2015

Abstract: In recent years, malware that steal credential information at target host have become a great threat and therefore countermeasures are needed. The more information the malware steals the more benefit the malware author earns and so malware tries to steal various types of information. However, credential information that the malware steals does not always exist at target host and that file access failure occurs. In this paper, we propose a method to detect credential search malware by focusing on file access failure to credential information. In general, file access failure also occurs from benign processes. Our method generates signatures to detect credential search malware using malware sandbox analysis, in which malware sample's file access log is monitored and file access failure to credential information are investigated. Based on the result of experiments with real malware samples which are known as information stealer, we show our method can detect malware that search credential information. And by applying our method to hosts of benign user, we show the difference between file access failure of benign user and malware.

Keywords: malware detection, information leak

1. はじめに

近年のサイバー攻撃では，ID やパスワードなどのアカウ

[1] 横浜国立大学
Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
[2] 横浜国立大学大学院環境情報研究院/横浜国立大学先端科学高等研究院
Graduate School of Environment and Information Sciences and Institute of Advanced Sciences, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
[3] 国立研究開発法人情報通信研究機構
National Institute of Information and Communications Technology, Koganei, Tokyo 184–8795, Japan
[a)] tanabe-rui-nv@ynu.jp

ント情報，メールアドレス，クレジットカード情報といった，ユーザの重要情報を狙った攻撃が増加している．これらの攻撃において，高度に機能化された悪意のあるソフトウェア，通称マルウェアが果たす役割は大きく，攻撃者はマルウェアを利用して感染先マシン内に保存されている重要情報を収集するとともに，取得した情報を悪用して不正活動を拡大する場合が多い．こうしたマルウェアの対策技術は多岐にわたるが，保護対象システム内へのマルウェアの侵入を検知・防止する「入口対策」，保護対象システム内でのマルウェア感染や不正活動を検知・防止する「内部対策」，外部にいる攻撃者とマルウェアの通信を検知・防止する「出口対策」の 3 つに分類することができる．すべての対策が重要であることはいうまでもないが，複数の対策技術を組み合わることで多層防壁を実現することが望ましい．

本稿では，上記対策技術の中で特に「内部対策」に注目し，保護対象マシン内で発生した重要情報へのファイルアクセス失敗挙動を監視することで情報探索を行うマルウェアを検知する手法を提案する[*1]．提案手法では，重要情報が保存されるファイルやそのファイルが保存されるディレクトリへアクセスすることで，感染先マシン内で重要情報の収集を行うマルウェア（以後，情報探索型マルウェアと呼ぶこととする）を検知対象とする．攻撃者は，マルウェアを利用して感染先マシン内に保存されている重要情報を外部に漏洩させた後，取得した情報を悪用することで不正活動を拡大する場合が多い．たとえば，Gumblar [25] 攻撃では，マルウェアが感染先マシン内に保存されている FTP アカウントの情報を漏洩し，攻撃者が取得した情報を悪用して不正な Web コンテンツをアップロードすることで正規サイトがマルウェア配布サイトに改竄され，被害が拡大する．また，PONY [23] は 100 近くのクライアントソフトから ID やパスワードをはじめとするアカウント情報を漏洩させることで知られている．攻撃者は，取得したアカウント情報を攻撃者の間で売買する場合や，漏洩した情報を悪用してアカウントを乗っ取り，不正サイトの URL が記載されているメッセージの送信や他人になりすますなどして被害を拡大させる [20]．このほかにも近年では，異なるサービス間で同じパスワードを使いまわしているユーザを対象に，事前に収集したパスワード情報を利用して不正ログインを試みるパスワードリスト攻撃 [21] による被害も増加している．標的型攻撃においても，長期にわたり情報収集が行われる場合や感染ホストが攻撃者の最終目的の情報を有していない場合も考えられるため，情報探索型マルウェアへの感染を早い段階で検知することで情報漏洩による被害を軽減できる可能性がある．このため，マルウェアによる情報漏洩を早い段階で検知し，不正活動の拡大を防ぐことが重要である．マルウェア作成者は，盗み出すデータが多いほどより大きな利益を生み出すことができるため，多様な情報の収集を試みる．しかし，感染先ホストにマルウェア作成者の目的とする情報が保存されているとは限らない．このため，重要情報へのファイルアクセス失敗挙動を監視することで，情報探索型マルウェアの感染を検知できる可能性がある．そこで，マルウェア動的解析を用いて情報探索型マルウェアのファイルアクセスログを収集し，ファイルアクセス失敗挙動を分析することで情報収集型マルウェアを検知するためのシグネチャを作成する．特に，ファイルアクセス失敗時のアクセス先の種類数に着目し，マルウェアによる情報探索活動と正規プロセスを区別する．このようにして作成したシグネチャを保護対象マシンのファイルアクセス失敗ログとマッチングすることで，マルウェア感染の有無を判断する．

評価実験では，分析対象のファイルに対してウィルス対策ソフトによる検知結果を提供するサービスである VirusTotal [26] から，2012 年 9 月 6 日から 2014 年 6 月 28 日までの間に，Web レポートなどで重要情報を狙うことが報告されているマルウェアの名前をキーワードに収集した 378 検体の実マルウェアに対してマルウェア動的解析を行い，提案手法を適用したところ多数の検体から情報収集活動が発生し，情報探索型マルウェアのファイルアクセス失敗挙動を観測することができた．また，正規ユーザのファイルアクセス失敗ログと情報探索型マルウェアのファイルアクセス失敗ログを比較したところ，情報探索型マルウェアからは重要情報が保存されるファイルやそのファイルが保存されるディレクトリを探索するファイルアクセスが見られた．さらに，特定のディレクトリへのファイルアクセス失敗数にも特徴が見られた．提案手法は，情報探索型マルウェアが重要情報の探索を試みたディレクトリ群と，そのディレクトリへのファイルアクセス失敗数からマルウェア感染検知を行っており，ファイルアクセス失敗挙動に注目している点が特徴である．

以降の本稿の構成は次のとおりである．2 章で先行研究を紹介し，3 章で提案手法の説明を行う．そして，4 章で提案手法を実装したシステムによる評価実験について述べ，5 章で考察を行う．最後に，6 章でまとめと今後の課題について述べる．

2. 先行研究

攻撃者が狙う重要情報は様々であるが，重要情報の収集はユーザに気付かれないように行われ，被害者の多くは被害にあっていることにすら気付いていない．このため，攻撃者やマルウェアによる情報収集活動を早い段階で検知することが重要である．たとえば，不審なファイルを保護対象マシン内で実行する前に，ユーザ環境を模擬した環境内

[*1] 本稿では，ファイルを格納するための保存場所をディレクトリと呼び，あるプロセスがファイルやディレクトリへのアクセスに失敗することをファイルアクセス失敗挙動と呼ぶこととする．

で実行し，マルウェアが行う情報収集や情報漏洩に見られる特徴をテイント解析により検出する手法 [13] が提案されている．また，広く普及しているウィルス対策ソフトやセキュリティアプライアンスの中にも，不審なファイルを解析環境内で実行することで，マルウェアの情報収集活動を検知できるものが存在する．これらは，未知の検体も検知できる点で有効であるが，解析環境を検知するマルウェアに対しては有効に働かない．このため，保護対象システム内で発生したマルウェアによる不正活動を検知することが重要である．

保護対象マシン内で発生した情報収集活動を検知する研究は，ホストベースまたはネットワークベースという観点で分類できる．前者の例として，正規ユーザや正規プロセスの挙動をプロファイリングし，保護対象マシン内で通常では起こりえない挙動が発生した場合に攻撃を検知する手法 [10], [11], [12] や，マルウェアと正規マシンの挙動を比較し，正規プログラムの挙動をモデル化することで，保護対象マシン内で異常が発生した場合にマルウェア感染を検知する手法 [7] など，保護対象システム内のポリシに反する挙動を攻撃として検知する手法が多数提案されている．また，保護対象マシン内のプロセスが行う通信に注目し，正規ユーザの操作がともなわない通信が発生した場合にマルウェアを検知する手法 [3], [8] など，情報収集を行うマルウェアの特徴が保護対象マシン内で観測された場合に攻撃を検知する手法が提案されている．このように，ホストベースの検知手法の多くはヒューリスティックな検知手法であり，様々な種類のマルウェア検体を検知することができる．しかし，既存手法の多くはシステムコールやレジストリ操作，プロセスの監視など膨大なログを分析する必要があり，実際にユーザがいる環境への適用が難しい場合がある．

一方で後者の例として，保護対象マシンのシステムコールのログをネットワーク上に存在する中央管理サーバに集約することで，各マシンの感染状況をネットワーク越しに確認する手法 [6] や，保護対象マシン群のプロセス情報を取得し，プロセスの親子関係や実行パス，プロセス名などを比較することでマルウェア感染を検知する手法 [5], [9] が提案されている．ネットワークベースの検知手法はホストベースの検知手法に比べ，シグネチャやルールの更新が簡単に行える反面，正規プロセスになりすましたマルウェアなど，すべての攻撃を検知できるとは限らない．これら以外にも，保護対象システム内に囮ファイルやホストを用意しておき，囮へのアクセスが発生した場合や，囮情報が悪用された場合に攻撃を検知する手法 [2], [14], [18], [19] が提案されている．また，攻撃者やマルウェアによるディレクトリやファイルなどの情報列挙要求に対し，実体のない仮想リソースを応答することで攻撃を妨害する手法 [4] が提案されている．こうした手法は，正規プロセスになりすましたマルウェアも検知できるが，ウィルス対策ソフトによるスキャンなど，正規プログラムによるアクセスを誤検知する可能性がある．また，既存手法の多くは実マルウェア検体を用いた評価実験を行っておらず，保護対象マシン内で攻撃者やマルウェアがアクセスする可能性の高い囮をどう用意するかが課題である．

上記関連研究の状況をふまえ，本稿では重要情報の探索時に発生するファイルアクセス失敗挙動に注目し，マルウェア動的解析により情報探索型マルウェアを検知するためのシグネチャを生成する手法を検討する．

3. 重要情報へのファイルアクセス失敗挙動に基づく情報探索型マルウェア検知

本章では，保護対象マシン内で発生した重要情報へのファイルアクセス失敗挙動を監視することで情報探索型マルウェアを検知するための手法を提案する．提案手法の基本アイディアは，情報探索型マルウェアが感染先マシン内で行う重要情報の探索活動を検知するためのシグネチャを，保護対象マシン内で発生したファイルアクセス失敗ログとマッチングすることでマルウェア感染の有無を判断するというものである．そのため，まず初めに実際のマルウェア検体が感染時にどのようなファイルアクセスを行うか調査し，ファイルアクセス失敗時のアクセス先の種類数に着目することで，情報探索型マルウェアに共通して見られる特徴をシグネチャとする．作成したシグネチャは保護対象マシン内，あるいは管理者のマシン上で保護対象マシンのファイルアクセス失敗ログと比較することでマルウェアの情報探索活動を検知する．以降では，3.1 節で提案手法の基本アイディアを説明し，3.2 節で提案手法の実現形態である重要情報へのファイルアクセス失敗挙動に基づく情報探索型マルウェア検知システムについて説明する．そして，3.3 節でシステムの実装について説明する．

3.1 基本アイディア

提案手法は，情報探索型マルウェアが感染先マシン内で行う重要情報の探索活動を検知するためのシグネチャを用いて，保護対象マシンのマルウェア感染の有無を判断するというものである．以下では，ある検体（MD5 ハッシュ値：ab5ec8f2acd42b635a79503701eb5d41，Symantec Norton による名称：W32.Qakbot [27]）が行うファイルアクセスの実例を用いて提案手法の基本アイディアを説明する．

4.1 節の評価実験で用いた動的解析環境において当該検体を実行すると，**図 1** のようなファイルアクセスが発生した．これは，多くのユーザが利用するクライアントソフトやそのクライアントソフトが設定情報をデフォルトで保存するファイルへのファイルアクセスであり，当該検体が感染先マシン内で重要情報の探索を行っていることが分かる．ここで，当

該検体がファイルアクセスに失敗したファイルパスのうち，“sm.dat” や “sitemanager.xml” のようにアクセスを試みたファイルを重要情報ファイル，“CuteFTP” や “Pocomail”，“Chrome” などのように重要情報ファイルが保存されているディレクトリを重要情報ディレクトリ，“C:¥Documents and Settings¥user name¥Application Data” などのように重要情報ディレクトリが保存されているディレクトリを監視対象ディレクトリと呼ぶこととする．マルウェア作成者は，盗み出す情報が多いほどより大きな利益を生み出すことができる．このため，感染先マシンから多様な情報を盗み出そうとする場合が多い．しかし，目的とする情報が感染先マシン内に保存されているとは限らない．実際に当該検体のファイルアクセスの結果に注目すると，重要情報へのファイルアクセスは失敗している．当該検体を実行した環境は，マルウェア検体の挙動を観測するための環境であり，重要情報はほとんど保存されていないが，正規ユーザが利用している環境においても，このようなファイルアクセス失敗挙動が発生することが予想される．そこで，ファイルアクセス失敗挙動に基づき情報探索型マルウェアの検知を行う．

Activity	Path	Result
OPEN,	C:¥Documents and Settings¥user name¥Application Data¥**CuteFTP¥sm.dat**,	NOT FOUND
OPEN,	C:¥Documents and Settings¥user name¥Application Data¥**FileZilla¥sitemanager.xml**,	NOT FOUND
OPEN,	C:¥Documents and Settings¥user name¥Application Data¥**Pocomail¥**,	NOT FOUND
OPEN,	C:¥Documents and Settings¥user name¥Application Data¥**BatMail¥**,	NOT FOUND
OPEN,	C:¥Documents and Settings¥user name¥Application Data¥Google¥**Chrome¥**,	NOT FOUND
OPEN,	C:¥Documents and Settings¥user name¥Application Data¥**RockMelt¥**,	NOT FOUND
OPEN,	C:¥Documents and Settings¥user name¥Application Data¥**CoffeeCup Software¥SharedSettings.ccs**,	NOT FOUND
	…	

図 1 W32.Qakbot [27] の重要情報探索活動（MD5 ハッシュ値：ab5ec8f2acd42b635a79503701eb5d41）

Fig. 1 Credential search of W32.Qakbot [27] (MD5hash: ab5ec8f2acd42b635a79503701eb5d41).

一方，マルウェア検体から発生したファイルアクセス失敗挙動がすべて情報探索を目的としているとは限らない．実際に，当該検体を解析環境内で実行したときに観測したファイルアクセスログから失敗したファイルアクセス先を分析したところ，様々なファイルやディレクトリへのファイルアクセス失敗挙動が観測された．各ファイルパスを「¥」記号ごとに分割し，最も左に現れたディレクトリをルートノード，最も右に現れたディレクトリやファイルをリーフ，間にあるディレクトリをノードとして，当該検体が失敗したファイルアクセス先を木構造で表した結果を**図 2** に示す．なお，この結果には当該検体を実行したときに解析環境内で動作していた正規プロセスによるファイルアクセス失敗挙動も含まれている．この結果から，特に “Application Data” ディレクトリへのファイルアクセスが頻繁に発生していることが分かる．当該ディレクトリへのファイルアクセスを分析したところ，動的解析時に実行したマルウェア本体（malware.exe）から数十種類以上に及ぶ重要情報ディレクトリや重要情報ファイルへのファイルアクセスであり，重要情報の探索を目的とするファイルアクセスである．また，このような挙動を複数種類のマルウェア検体から確認することができた．一般に，正規プロセス

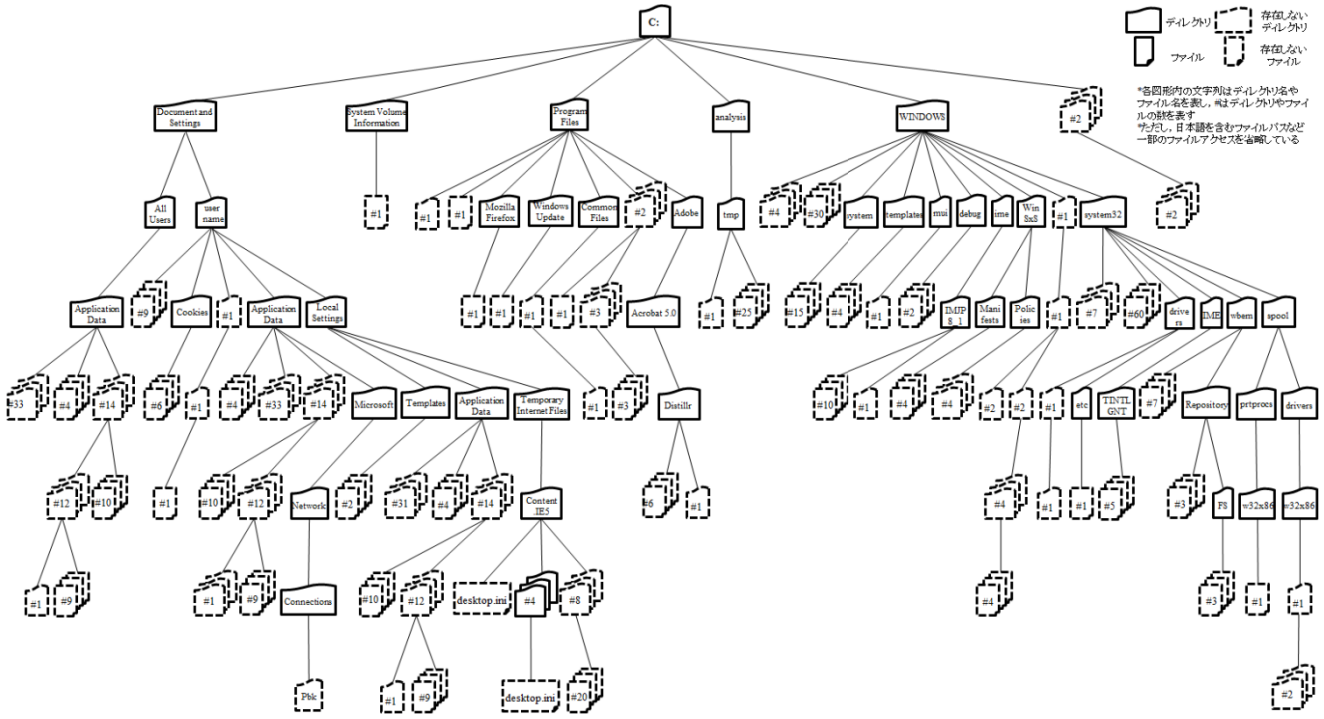


図 2 W32.Qakbot [27] を解析環境内で実行したときのファイルアクセスログから失敗したファイルアクセス先を抽出した結果（MD5 ハッシュ値：ab5ec8f2acd42b635a79503701eb5d41）

Fig. 2 Files access failure path abstracted from file access log of W32.Qakbot [27] executed at analysis environment (MD5hash: ab5ec8f2acd42b635a79503701eb5d41).

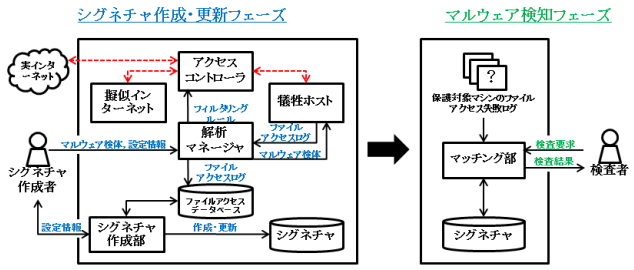


図 3 重要情報へのファイルアクセス失敗挙動に基づく情報探索型マルウェア検知システム

Fig. 3 Malware detection system focused on file access failure to credential files.

からもファイルアクセス失敗挙動は発生する．しかし，あるプロセスから複数種類の重要情報へのファイルアクセスが発生することは考えにくい．そこで，マルウェアが行うファイルアクセス失敗挙動のアクセス先に着目し，マルウェアが探索する監視対象ディレクトリと重要情報ディレクトリや重要情報ファイルの種類数を抽出することで，情報探索型マルウェアを検知するためのシグネチャを作成する．

ただし，マルウェア検体がどの環境でも同じファイルアクセスを行うとは限らない．実際に当該検体を Windows XP と Windows7 の 2 つの異なる OS 上で実行した場合にファイルアクセス先に違いが見られた．おそらく，感染先マシンのファイルシステムの構造に応じてアクセス先のディレクトリやファイルを変更していることが予想される．このため，同一検体を複数の異なる環境下で実行することで，情報探索型マルウェアをより多くの環境下で検知可能なシグネチャのセットを作成する．そして，作成したシグネチャと保護対象マシン内で発生したファイルアクセス失敗ログをマッチングすることで，マルウェアによる情報探索活動を検知する．

3.2 重要情報へのファイルアクセス失敗挙動に基づく情報探索型マルウェア検知システム

本節では，3.1 節で説明した基本アイディアの実現形態である重要情報へのファイルアクセス失敗挙動に基づく情報探索型マルウェア検知システム（以降，単にシステムと呼ぶこととする）について説明する．まず，**図 3** に本システムの全体図を示す．本システムは，情報探索型マルウェアの検知に用いるシグネチャを作成，更新するための**シグネチャ作成・更新フェーズ**と，作成したシグネチャを用いて保護対象マシンのマルウェア感染の有無を判断する**マルウェア検知フェーズ**からなる．シグネチャの作成や更新は，AV ベンダの解析者などマルウェアに関する広い知識を有する技術者が行い，作成されたシグネチャは AV ソフトのシグネチャのように，複数の異なる環境下で使われるユニバーサルなシグネチャを想定している．一方，マルウェア検知はネットワーク管理者のマシンや保護対象マシン上で行われる．シグネチャマッチングでは，誤検知と見逃しの関係を考慮してシグネチャを複数用意しておくことで，検査者のニーズに合わせた検査を行う．また，ホワイトリストを用意しておくことで，誤検知を減らすことが可能である．以降では，本システムの構成要素について述べ，それぞれの処理について説明する．

シグネチャ作成者：シグネチャ作成者は，マルウェア動的解析や情報探索型マルウェアを検知するためのシグネチャの作成を行う．シグネチャ作成者は，まず初めに解析マネージャにマルウェア検体と設定情報を入力し，マルウェア動的解析を行う．その後，シグネチャ作成部に設定情報を入力し，シグネチャを作成する．

犠牲ホストと解析マネージャ：犠牲ホストは，マルウェア動的解析において検体を故意に実行して感染させるための環境である．解析マネージャは，シグネチャ作成者から受取った情報をもとに，アクセスコントローラにフィルタリングルールを適用するとともに，マルウェア検体を犠牲ホスト上で実行し，マルウェア動的解析を開始する．その後，マルウェア検体が犠牲ホスト上で行ったファイルアクセスログを受け取り，犠牲ホストをマルウェア感染前の状態に復元してマルウェア動的解析を終了する．取得したファイルアクセスログはファイルアクセスデータベースに蓄積され，シグネチャの作成に利用される．

アクセスコントローラと擬似インターネット：アクセスコントローラは，犠牲ホスト上で実行されたマルウェア検体が外部に対して行う通信を，危険性が低いと判断された場合には実インターネットへ，それ以外の場合には擬似インターネットへ転送するか通信を遮断する．一方，擬似インターネットやインターネット側から犠牲ホストに対して行われる通信は無条件に犠牲ホストに転送する．擬似インターネットには，ハニーポットプログラムである Nepenthes [1] を用いており，135/TCP，445/TCP などエクスプロイト攻撃の対象となるポート上で脆弱なサービスを模擬することでマルウェア検体の攻撃を観測する．

ファイルアクセスデータベース：ファイルアクセスデータベースは，シグネチャを作成するのに必要なデータを蓄積するためのデータベースであり，マルウェア動的解析環境内で発生したファイルアクセス失敗挙動を蓄積する．実体としては，（検体の MD5 ハッシュ値，プロセス名，ファイルパス）の組みからなるレコードの集合である．

シグネチャ作成部とシグネチャ：シグネチャ作成部は，シグネチャ作成者から受け取った情報をもとに，ファイルアクセスデータベースから情報探索を目的とするファイルアクセス失敗挙動を抽出し，情報探索型マルウェアを検知するためのシグネチャを作成，更新する．シグネチャの実体としては，（監視対象ディレクトリ，保護対象マシン内で情報探索型マルウェアを検知するためのしきい値）の組みで

ある（以後，保護対象マシン内で情報探索型マルウェアを検知するためのしきい値を，情報探索型マルウェア検知用しきい値と呼ぶこととする）．

シグネチャ作成・更新の流れ：シグネチャの作成では，まず初めにマルウェアが重要情報の探索を目的にアクセスする監視対象ディレクトリ群を抽出し，その後，重要情報ディレクトリや重要情報ファイルの種類数から情報探索型マルウェア検知用しきい値を決定する．以下にシグネチャ作成，更新の流れを示す．

A) シグネチャ作成者が解析マネージャにマルウェア検体，および設定情報（フィルタリングルール，実行時間 S [秒]）を入力し，動的解析を開始する．

B) 解析マネージャは実行時間経過後，ファイルアクセスログを取得し，犠牲ホストを感染前の状態に復元して動的解析を終了する．取得したファイルアクセスログはファイルアクセスデータベースに転送され，ファイルアクセス失敗挙動が蓄積される．

C) 動的解析を繰り返し，マルウェア検体のファイルアクセス失敗挙動をデータベースに蓄積する．

D) シグネチャ作成者がシグネチャ作成部にシグネチャ設定情報（ファイルアクセスデータベースから監視対象ディレクトリを決定するためのしきい値）を入力し，シグネチャの作成を開始する．シグネチャ作成部は，まず初めにファイルアクセスデータベースに以下の操作を行い，監視対象ディレクトリ群の抽出を行う．ここで，ファイルアクセスデータベースから監視対象ディレクトリを決定するためのしきい値を，監視対象ディレクトリ抽出用しきい値，ファイルアクセスデータベースに登録されている MD5 ハッシュ値の集合を H，各 MD5 ハッシュ値を $h \in H$，ファイルアクセスデータベースに登録されているプロセス名の集合 P，各プロセス名を $p \in P$，監視対象ディレクトリの集合を D，各ディレクトリを $d \in D$ と呼ぶこととする．
   (ア) ファイルアクセスデータベースから，MD5 ハッシュ値が $h \in H$，プロセス名が $p \in P$ であるレコードのファイルパスの一覧を取得する．ただし，同じファイルパスは 1 つのファイルパスとし，重複がないようにする．
   (イ) 抽出したファイルパスをディレクトリやファイルごとに分割し，各ファイルパスで最も左に現れたディレクトリをルートノード，最も右に現れたファイルやディレクトリをリーフ，間に現れたディレクトリをノードとする木構造として考え，すべてのノードに対して前順走査を行う．そして，あるノードに接続している子ノードの数が監視対象ディレクトリ抽出用しきい値を超えた場合に，ルートノードから当該ノードまでのファイルパスを監視対象ディレクトリ Dir として，監視対象ディレクトリ Dir が監視対象ディレクトリの集合 D に含まれていなければ追加する．

E) 続いて，シグネチャ作成部はファイルアクセスデータベースから，MD5 ハッシュ値が $h \in H$，プロセス名が $p \in P$，ファイルパスに監視対象ディレクトリ $d \in D$ を含むレコードから，監視対象ディレクトリ $d \in D$ へのファイルアクセス一覧を取得し，重要情報ディレクトリや重要情報ファイルの種類数を抽出する．

F) 情報探索型マルウェア検知用しきい値は，抽出した重要情報ディレクトリや重要情報ファイルの種類数に応じて決定するが，値が低くなるとマルウェア感染を見逃すことは低くなるが，反対に誤検知が増加してしまう．同様に，値が高くなると誤検知は減少するが，見逃しが増えてしまうという特性を持つ．そこで，各監視対象ディレクトリ $d \in D$ に対して，情報探索型マルウェア検知用しきい値を複数用意しておく．具体的な値の決定方法についてはさらなる検討が必要であり，5 章において考察する．

G) シグネチャ作成部は，作成されたシグネチャの監視対象ディレクトリがシグネチャ内に存在しない場合にシグネチャの追加を行い，作成されたシグネチャの監視対象ディレクトリがシグネチャ内にすでに存在した場合には，シグネチャ作成者の判断に応じてシグネチャの追加，更新を行う．

マルウェアの挙動は不確定であり，マルウェア検体が動的解析時に必ずしも重要情報探索活動を行うとは限らない．このため，同一検体を複数回動的解析することで，ファイルアクセスデータベースにより多くの情報を蓄積することが望ましい．同様に，ファイルシステムの構造は OS ごとに異なっており，マルウェア検体の多くも実行される OS に応じて重要情報の探索を試みるディレクトリ（監視対象ディレクトリ）を変更するため，様々な OS 上でマルウェア検体を動的解析することが望ましい．一方で，提案手法で作成するシグネチャはマルウェア検体ごとに固有のものではなく，マルウェアの情報探索活動を広くとらえたものである．そのため，1 つのシグネチャで複数種類のマルウェア検体を検知することができる．また，情報探索型マルウェアの多くは “Application Data” ディレクトリなど，あるディレクトリ内に保存される情報を狙う．これまでに観測したことのない新たなクライアントソフトを狙う情報探索型マルウェアが登場した場合にも，クライアントソフトが重要情報を監視対象ディレクトリ内に保存する場合には，提案手法により作成したシグネチャを用いて当該検体を検知することができる可能性がある．

実際に作成したシグネチャを用いて情報探索型マルウェアへの感染を検査する場合，動的解析時のファイルアクセス失敗ログと保護対象マシンのファイルアクセス失敗ログ

を用いて，検査者のニーズに合わせたシグネチャを用いる．多くの保護対象マシンを定常的に監視する場合は，大量の誤検知を生じるシグネチャは対策のコストを増大させるため使用が困難であるが，厳重な保護が必要な重要情報を有するホストやマルウェア感染が疑われるホストを詳細に監視する場合には見逃しが少ないシグネチャの利用も想定される．また，提案手法によりマルウェア感染を検知できた場合，重要情報がすでに外部に漏洩している可能性がある．提案手法は，不正侵入を未然に防ぐ技術や攻撃者が収集した情報を外部に漏洩させることを検知する技術との併用により高い防御効果が期待できる．

検査者とファイルアクセス失敗ログ：検査者は，マッチング部へ検査要求を入力し，マッチング部が保護対象マシンのファイルアクセス失敗ログとシグネチャをマッチングした検査結果を受け取る．ファイルアクセス失敗ログの収集は各保護対象マシン内で単位時間ごとに収集する．

マッチング部：マッチング部では，受け取ったファイルアクセス失敗ログを作成したシグネチャとマッチングし，マルウェア感染の有無を判断する．検査に使うシグネチャは検査者のニーズに応じて変更することが可能である．

マルウェア検知の流れ：マルウェアの検知は，作成したシグネチャをファイルアクセス失敗ログとマッチングすることで行われる．具体的には，検査者がネットワークの管理者などである場合には，ファイルアクセス失敗ログとシグネチャをマッチングするためのマシンを保護対象ネットワーク上に用意し，保護対象マシン群のファイルアクセス失敗ログを受け取り，マルウェア感染の有無を判断する．一方，検査者が保護対象マシンのユーザである場合には，保護対象マシン内でファイルアクセス失敗ログを収集し，保護対象マシン内でマルウェア感染の有無を判断する．マッチングでは，ファイルアクセス失敗ログから監視対象ディレクトリへのファイルアクセス失敗挙動を抽出し，重要情報ディレクトリや重要情報ファイルの種類数が情報探索型マルウェア検知用しきい値より小さければ (1) マルウェアによる情報探索活動は検知されなかったと判断する．一方，情報探索型マルウェア検知用しきい値より大きい場合には (2) マルウェアによる情報探索活動が検知されたと判断する．マルウェア検体が感染先マシン内で情報探索活動を行うタイミングはバラバラであるため，保護対象マシンのファイルアクセスログとシグネチャを継続的にマッチングすることが望ましい．

3.3 実装

本節では，本システムの実装について説明する．まず初めに図 4 に全体図を示す．提案手法を適用した本システムは，評価実験を効率的に行う理由からマシン A とマシン B の 2 台の実マシン上に実装した．マシン A の犠牲ホストには VMware Server 1.0.6 のゲスト OS（Windows XP Professional SP1）を用いており，ホスト OS には CentOS 5.3 を用いた．マシン B の犠牲ホストには Virtual Box のゲスト OS（Windows 7 64 Bit Professional）を用いており，ホスト OS には Ubuntu 12.0.4 を用いた．また，犠牲ホスト以外のすべての構成要素はホスト OS 上に実装した．以下，各構成要素の実装について述べる．

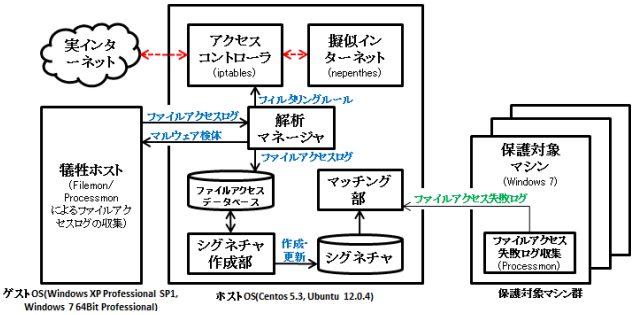


図 4 提案手法の実装

Fig. 4 Implementation of proposal method.

犠牲ホストと解析マネージャ：マシン A の犠牲ホストは VMware Server 1.0.6 のゲスト OS として実現した．犠牲ホストの OS は Windows XP Professional SP1 としたが，任意の OS との入れ替えが可能である．マシン B についても同様である．解析マネージャは Perl により実装した．解析マネージャはシグネチャ作成者から受け取った設定情報をもとにフィルタリングルールをアクセスコントローラに適用し，通信の制御を行う．解析マネージャが犠牲ホストを起動すると，犠牲ホストは Windows のタスクスケジューラによってあらかじめ設定したタスクにより，Filemon または Processmon [24] を起動してファイルアクセスログの収集を開始するとともに，マルウェア検体を解析マネージャから SSH によりダウンロードして実行する．設定時間経過後，ファイルアクセスログの収集結果を保存するとともに，解析マネージャに結果を返す．その後，解析マネージャがゲスト OS を感染前の状態に復元して動的解析を終了するとともに，ファイルアクセスデータベースに結果を蓄積する．

擬似インターネットとアクセスコントローラ：擬似インターネットには，ハニーポットプログラム Nepenthes 0.2.2 [1] をホスト OS 上で起動し，犠牲ホストからの要求に応答できるようにした．アクセスコントローラには，Linux のパケットフィルタリングツールである iptables を用いた．アクセスコントローラでは NAT を行っており，ゲスト OS から実インターネット上のホストへの接続要求を許可する場合には iptables の PREROUTING チェインで ACCEPT ターゲットを用い，POSTROUTING チェインにおいて MASQUERADE ターゲットを適用することで IP アドレスの変換を行い，実インターネット上のホストと通信できるようにした．また，ゲスト OS から実インターネット上のホストへの接続要求を許可しない場合に

```
### Record ###
020f7636117313a1d685c522b7e77da6, malware.exe, C:¥ProgramData¥CuteFTP¥
020f7636117313a1d685c522b7e77da6, malware.exe, C:¥ProgramData¥pocomail¥
020f7636117313a1d685c522b7e77da6, malware.exe, C:¥ProgramData¥Google¥Chrome¥
...

### Signature ###
C:¥ProgramData¥, 5
C:¥Users¥user name¥AppData¥Local¥, 10
C:¥Users¥user name¥AppData¥Roaming¥, 10
...
```

図 5 ファイルアクセスデータベース内のレコードと提案手法を用いて作成されるシグネチャの例

Fig. 5 Example of file access database records and signature made by proposal method.

は，Nepenthes [1] が対応するポートへの通信については，iptables の REDIRECT ターゲットを用いて擬似インターネットへ通信を転送し，それ以外のポートへの通信についてはすべて遮断した．同様に，ゲスト OS からホストマシンと同じネットワークに所属しているホストへの通信についてはすべて遮断した．アクセスコントローラと擬似インターネットの実装の詳細については文献 [15], [16], [17] をご参照いただきたい．

シグネチャ作成部とマッチング部：シグネチャ作成部とマッチング部は Perl により実装した．また，同一マシン上に実装した．それぞれの詳細については 3.2 節で説明したとおりであるが，マルウェアの中には，感染先マシンのファイルシステムの構造に応じてファイルアクセス先を変えるものが存在する．このため，同一検体を Windows XP と Windows 7 の 2 つの異なる解析環境下で実行し，マルウェア感染ホスト検知フェーズではそれぞれの環境下で作成されたシグネチャを合わせて使う．

ファイルアクセスデータベースとシグネチャ：ファイルアクセスデータベース内のデータや作成されたシグネチャは入れ替えや追加など，結果を容易に共有することができる．以下，ファイルアクセスデータベース内のレコードとシグネチャの書式について説明する．ファイルアクセスデータベースの各レコードは，動的解析を行ったマシン内で収集したファイルアクセスログからファイルアクセス失敗挙動を抽出した結果である．このため，解析環境内で動作していたプロセスによるファイルアクセス失敗挙動や，情報探索を目的としないファイルアクセス失敗挙動も含まれている．**図 5** にファイルアクセスデータベース内のレコードの例を示す．各レコードは左から順に，（実行した検体の MD5 ハッシュ値，操作を行ったプロセスの名前，ファイルパス）である．なお，実験に使用した Filemon や Processmon [24] の出力結果は（操作を行った時刻，プロセス名，プロセス ID，操作内容，操作を行ったファイルやディレクトリ，操作の結果）であり，ファイルアクセスを試みたファイルやディレクトリが存在しない場合には，操作結果に “NOT FOUND”，“NAME NOT FOUND” が，アクセスを試みたファイルパスが存在しない場合には “PATH NOT FOUND が” 操作結果に出力される．4 章の評価実験では，上記の場合にファイルアクセスに失敗したと判断している．

保護対象マシンのファイルアクセス失敗ログ：保護対象マシンには，研究室内のメンバに協力していただき，4 台のマシンから Processmon [24] を用いてファイルアクセス失敗ログを収集した．また，収集した保護対象マシンのファイルアクセス失敗ログと作成したシグネチャのマッチングは，シグネチャを作成したマシン上のマッチング部で行った．シグネチャマッチングは保護対象マシン内で行うことも可能であるが，このような監視機構の実装は今後の課題とする．なお，これらのマシンにはエンドユーザ向けのウィルス対策ソフトが導入されており，マルウェアには感染していないものとする．また，ファイルアクセス失敗ログの収集は，ユーザが利用している時間を中心に収集した．

4. 評価実験

提案手法は，情報探索型マルウェアへの感染の有無を検査する手法である．このため，まず初めに実験 1 において，実マルウェア検体を用いてシグネチャを作成し，シグネチャの作成に用いた検体セットと同一の検体セットに対して提案手法を適用することで，提案手法の検知能力を評価した．次に実験 2 において，提案手法が異なる OS 下でも有効に働くことを検証するため，実験 1 で使用した検体セットからランダムに抽出した検体セットに対して，動的解析環境におけるゲスト OS を変更した状態で実験 1 同様の実験を行い，提案手法の検知能力を評価した．最後に実験 3 において，提案手法による誤検知を検証するため，正規ユーザのファイルアクセス失敗ログと実験 2 で作成したシグネチャを比較し，提案手法の検知能力を評価した．

4.1 シグネチャ作成時と同一の検体セットに対する検知精度の評価（実験 1）

実験方法：実験 1 では，VirusTotal [26] から 2012 年 9 月 6 日から 2014 年 6 月 28 日までの間に収集した 1,070 検体の実マルウェア（以降では，評価用検体セットと呼ぶこととする）からシグネチャを作成し，提案手法を用いて情報探索型マルウェアを検知できることを確認した．検体の収集には，Web レポートなどで重要情報を狙うことが報告されているマルウェアの検知名を利用した．実験には，3 章で述べたマルウェア動的解析システムを用いており，動的解析やシグネチャの作成に用いた設定は**表 1** のとおりである．また，評価用検体セットの内訳を**図 6** に示す．なお，検体の分類には Symantec Norton による検知結果を用いている．

実験結果：評価用検体セットを動的解析したところ，数百回以上に及ぶファイルアクセス失敗挙動が観測された検体が存在した一方で，ファイルアクセス失敗挙動が数十回未満

表 1 動的解析やシグネチャの作成に用いた設定（実験 1）
Table 1 Settings of malware dynamic analysis and signature creation (experiment 1).

解析環境	Windows XP SP2
ネットワーク設定	53/udp,80/tcpのみ外部接続を許可
検体実行時間	300秒
ファイルアクセスログ取得方法	Filemon
解析期間	2014/11/8~2014/11/14
監視対象ディレクトリ抽出用しきい値	40

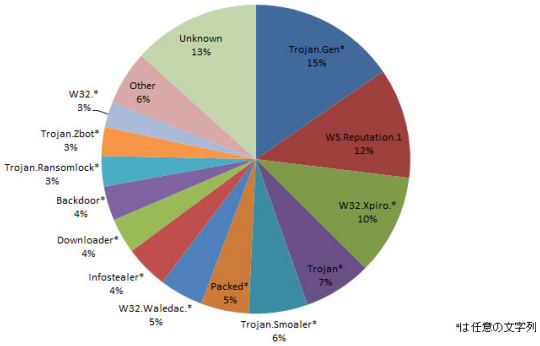


図 6 評価用検体セットの内訳（実験 1）
Fig. 6 Detail of malware samples used for experiment 1.

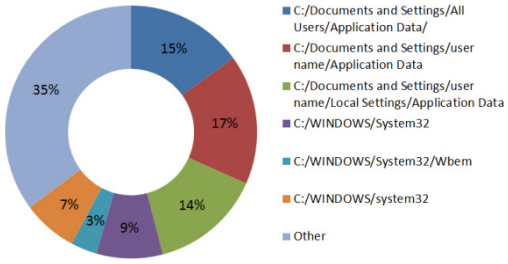


図 7 失敗したファイルアクセス先の分布（実験 1）
Fig. 7 Distribution of file access failure path (experiment 1).

である検体も存在した．マルウェアの挙動は不確定であり，当該検体群が実験期間内で情報探索活動を行うとは限らない．また，評価用検体セットには情報探索活動を行わない検体も含まれている．このため，実験 1, 2 では，動的解析環境内で情報探索活動を行ったマルウェア検体を検知することを目指す．評価用検体セットのファイルアクセス失敗挙動を分析したところ，“Application Data” ディレクトリなどいくつかのディレクトリにファイルアクセス失敗挙動が集中していた．また，このような挙動は検体ファミリが異なる検体間で観測された．**図 7** にファイルアクセス失敗先の分布を，**図 8** に監視対象ディレクトリを “C:/Documents and Settings/user name/Application Data/” としたときにしきい値を 1～50 までとした場合の，評価用検体セットに対する提案手法の検知能力を検証した．この結果，情報探索型マルウェア検知用しきい値を 1 とした場合，1,070 検体のうち約 89%（957 検体）を検知することができた．情報探索型マルウェア検知用しきい値を高くしていくと検知できる検体の数は減少していき，4～15 の範囲では約

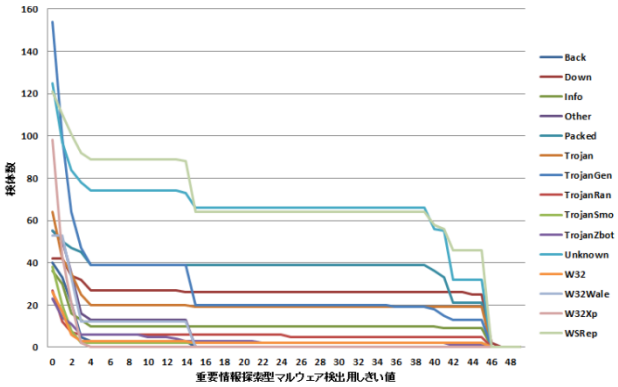


図 8 シグネチャの作成に用いた情報探索型マルウェア検知用しきい値と検知できる検体数の関係（実験 1）
Fig. 8 Relationship between threshold used for making a signature and number of malware samples detected by the signature (experiment 1).

30%～35%の検体を検知することができた．情報探索型マルウェア検知用しきい値が 16～42 の範囲では検知できる検体は約 20%～24%であり，さらに値を高くしていき，情報探索型マルウェア検知用しきい値が 43～46 の範囲では検知できる検体は約 16%であった．最後に，情報探索型マルウェア検知用しきい値を 47 とした場合，検知できるのは約 0.2%であった．一方で，情報探索型マルウェア検知用しきい値を 1～4 の範囲にした場合，13 種類以上の検体ファミリを検知することができた．同様に，情報探索型マルウェア検知用しきい値を 5～15 の範囲にした場合，12 種類以上の検体ファミリを検知することができ，情報探索型マルウェア検知用しきい値を 16～46 の範囲にした場合，図 6 において検知結果が Other となっている検体が検知できなくなり，10 種類の検体ファミリを検知することができた．最後に，情報探索型マルウェア検知用しきい値を 47 とした場合，1 種類の検体ファミリを検知することができた．なお，本実験ではマルウェア本体が行ったファイルアクセスログのみを検査対象としている．

4.2 シグネチャ作成時と同一検体セットに対する検知精度の評価（実験 2）

実験方法：実験 2 では，実験 1 で使用した評価用検体セットの中からランダムに抽出した 378 検体に対して，実験 1 と異なる解析環境下で実行した場合にも提案手法が有効であることを確認し，提案手法の検知能力を評価した．実験には，3 章で述べたマルウェア動的解析システムを用いた．動的解析やシグネチャの作成に用いて設定は**表 2** のとおりである．

実験結果：当該検体群を動的解析したところ，いくつかのディレクトリの下の重要情報にファイルアクセスが集中していた．そこで，これらのディレクトリを監視対象ディレクトリとし，情報探索型マルウェア検知用しきい値を 1～

表 2 動的解析やシグネチャの作成に用いた設定（実験 2）
Table 2 Settings of malware dynamic analysis and signature creation (experiment 2).

解析環境	Windows 7 64bit
ネットワーク設定	53/udp,80/tcpのみ外部接続を許可
検体実行時間	300秒
ファイルアクセスログ取得方法	Processmon
解析期間	2015/2/20〜2015/3/9
監視対象ディレクトリ抽出用しきい値	40

表 3 監視対象ディレクトリごとのシグネチャの作成に用いた情報探索型マルウェア検知用しきい値と検知できる検体数の関係（実験 2）
Table 3 Relationship between information stealing malware detection threshold used for making signature and number of malware samples detected by the signature for each monitoring target directory (experiment 2).

情報探索型マルウェア検知用しきい値	監視対象ディレクトリ		
	C:/ProgramData	C:/Users/username/AppData/Local	C:/Users/username/AppData/Roaming
1	378	378	378
2	202	241	378
3	190	189	203
4	190	186	200
5	186	186	186
6	186	181	186
7	186	181	186
8	186	181	186
9	181	181	186
10	181	181	186
15	181	181	186
20	181	181	181
25	181	181	181
30	181	181	181
35	181	181	181
40	180	180	180
45	171	171	171
50	2	0	0

50 までとした場合の提案手法の検知能力を検証した．**表 3** に監視対象ディレクトリごとに設定したしきい値に対する検知結果を示す．この結果，情報探索型マルウェア検知用しきい値を 1 とした場合，378 検体のうちすべての検体を検知することができた．情報探索型マルウェア検知用しきい値を高くしていくと検知できる検体の数は減少していき，5〜15 の範囲ではどのシグネチャでも約 47%〜49%の検体を検知することができた．情報探索型マルウェア検知用しきい値が 16〜48 の範囲ではどのシグネチャでも約 45%〜47%の検体を検知することができた．最後に，情報探索型マルウェア検知用しきい値を 50 とした場合，シグネチャとして有効な監視対象ディレクトリは “C:/ProgramData” のみであり，検知できたのは 0.5%であった．実験 1 と 2 では，監視対象ディレクトリこそ異なるものの，マルウェア検体がファイルアクセスに失敗した重要情報ファイルや重要情報ディレクトリの種類やアクセス数は類似しており，実際に存在する Web ブラウザやメールソフト，FTP クライアントソフトへのファイルアクセスを試みていた．

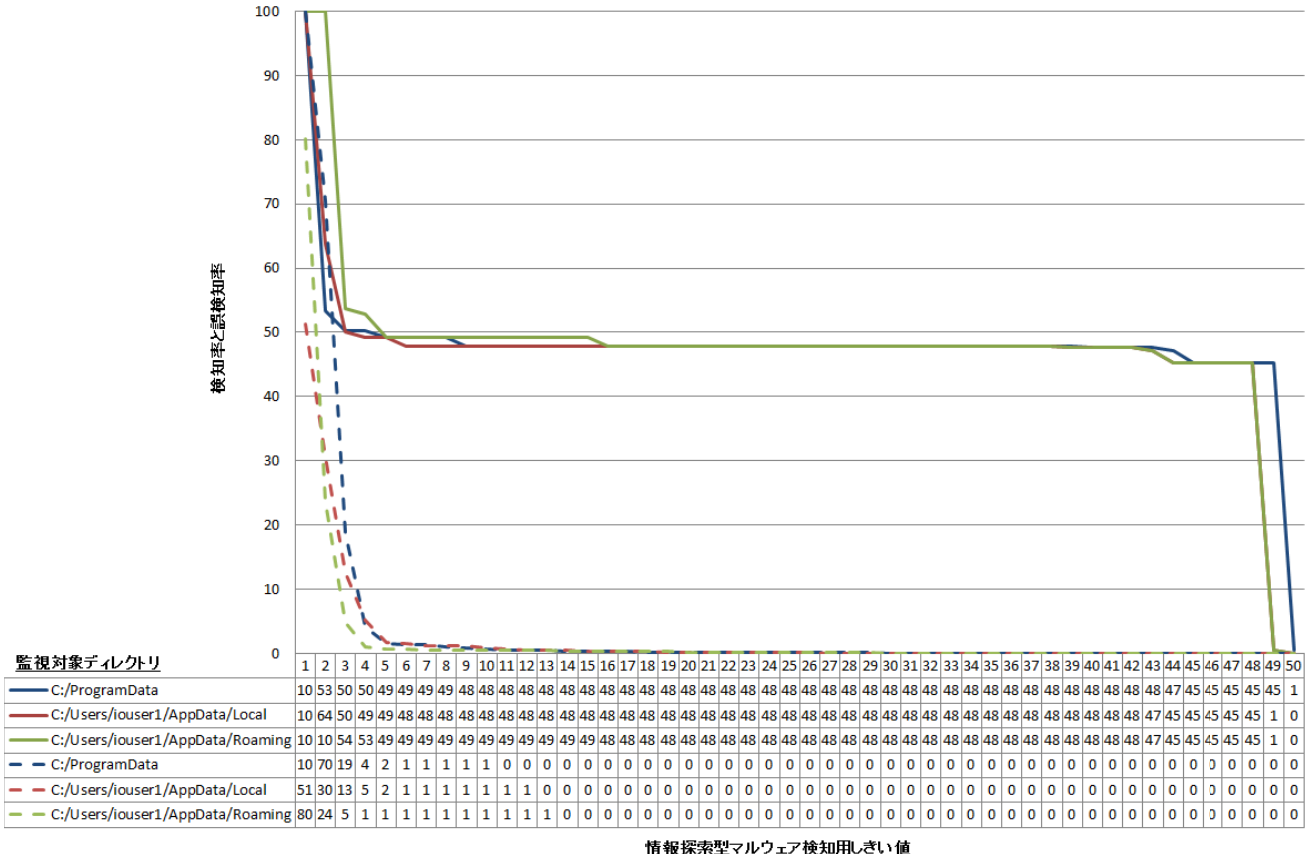
表 4 正規ユーザのファイルアクセス失敗ログ（実験 3）
Table 4 File access fail logs of benign users (experiment 3).

	観測期間(date)	観測時間(hour)
マシン1	2015/2/24~2015/2/28	48
マシン2	2015/2/18~2015/2/22	48
マシン3	2015/2/14~2015/2/23	39
マシン4	2015/2/20~2015/2/25	32
合計	2015/2/14~2015/2/28	167

4.3 正規ユーザのファイルアクセス失敗ログに対する検知精度の評価（実験 3）

実験方法：実験 3 では，実験 2 で作成したシグネチャと正規ユーザのファイルアクセス失敗ログを比較することで，提案手法の検知精度を評価した．実験には，正規ユーザが使用している 4 台のマシンから収集したファイルアクセス失敗ログを用いた．これらのマシンは，書類や資料の作成などの日常業務を行うために利用されているマシンであり，OS として Windows7 が利用されている．エンドユーザ向けのウィルス対策ソフトが導入されているほか，OS の更新プログラムやウィルス対策ソフトのシグネチャは最新の状態である．本実験では，これらのマシンはマルウェアに感染していないマシンであると想定する．なお，ファイルアクセス失敗ログはユーザがマシンの電源を入れた直後から収集を開始し，ユーザが作業を行っている間も収集し続けた．また，ログの収集には Processmon を用いた．**表 4** に正規ユーザのファイルアクセス失敗ログの詳細を示す．

実験結果：実験 2 で作成したシグネチャと正規ユーザのファイルアクセスログを比較した．**図 9** に提案手法をマルウェア動的解析結果と正規ユーザのファイルアクセス失敗ログに適用したときの検知精度の評価結果を示す．正規ユーザが観測期間内で利用していた 4 台のマシンのファイルアクセスログを 5 分ごとに分割し，実験 2 で作成したシグネチャとマッチングした場合に，情報探索型マルウェアに感染していると判断したファイルアクセスログ数を全ファイルアクセスログ数で割った結果（以後，誤検知率と呼ぶこととする）を点線で表す．また，実験 2 で作成したシグネチャと実験 2 のマルウェア動的解析時のファイルアクセスログをマッチングした場合に，マルウェアに感染していると判断したファイルアクセスログ数をマルウェア検体の総数で割った結果（以後，検知率と呼ぶこととする）を実線で表す．この結果から，情報探索型マルウェア検知用しきい値が 1〜3 の範囲では，どの監視対象ディレクトリでも誤検知が大量に発生した．情報探索型マルウェア検知用しきい値が 4〜26 の範囲では，どの監視対象ディレクトリも誤検知率は約 0.06%〜5.1%（検知されるファイルアクセスログ数 1〜76）であった．一方で，検知率は約 47%〜



監視対象ディレクトリ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
C:/ProgramData	10	53	50	50	49	49	49	49	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	47	45	45	45	45	45	1
C:/Users/iouser1/AppData/Local	10	64	50	49	49	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	47	45	45	45	45	1	0
C:/Users/iouser1/AppData/Roaming	10	10	54	53	49	49	49	49	49	49	49	49	49	49	49	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	47	45	45	45	45	1	0
C:/ProgramData	10	70	19	4	2	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C:/Users/iouser1/AppData/Local	51	30	13	5	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C:/Users/iouser1/AppData/Roaming	80	24	5	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

図 9 提案手法をマルウェア動的解析結果と正規ユーザのファイルアクセス失敗ログに適用したときの検知精度の評価結果（実験 3）

Fig. 9 Evaluation result of proposal method using malware dynamic analysis result and benign user file access fail log (experiment 3).

52%（検知できた検体数 181～200）であった．情報探索型マルウェア検知用しきい値が 27～29 の範囲では，どの監視対象ディレクトリも誤検知率は 0%～0.06%（検知されるファイルアクセスログ数 0～1）であった．一方で，検知率は 47%（検知できた検体数 181）であった．情報探索型マルウェア検知用しきい値を 30 とすると，どの監視対象ディレクトリも誤検知率が 0%となった一方で，検知率は 47%のままであった．

5. 考察

実験 1 では，1,070 に及ぶ実マルウェア検体に対して提案手法を適用し，情報探索型マルウェア検体を検知した．また，1 つのシグネチャで複数のマルウェアファミリを検知できることを確認した．実験 2 では，実験 1 と OS が異なる解析環境で実行した検体群に対し提案手法を適用し，情報探索型マルウェアを検知した．また，実験 1，2 の結果から，OS の種類の違いよってファイルシステムの構造が異なるため監視対象ディレクトリが異なったものの，重要情報ファイルや重要情報ディレクトリについては，同一のものが多かった．実験 3 では，正規ユーザのファイルアクセス失敗ログに対して提案手法を適用し，検知できる検体数と誤検知数について調査した．これらの結果から，提案手法は情報探索型マルウェアの特徴をとらえており，ファイルアクセス失敗挙動を用いてマルウェア感染を検知できる可能性があるが，提案手法を実用化するにはさらなる工夫が必要であることが分かった．このため，本章では提案手法の有効性や課題について考察する．

情報探索型マルウェア検知用しきい値の決定方法：情報探索型マルウェア検知用しきい値は，値が低くなると検知できる検体数や検知できる検体ファミリの種類数が増えるが誤検知も増えてしまう．反対に，値が高くなると誤検知は減少するが検知できる検体数や検知できる検体ファミリの種類数も減少してしまうという特性を持つ．このため，情報探索型マルウェア検知用しきい値は見逃しと誤検知の関係を考慮しながら決定すべきである．また，実験 2 の結果から，監視対象ディレクトリによって検知できる検体数や誤検知数が異なるため，監視対象ディレクトリごとに決定すべきである．加えて，実ユーザが利用している環境には様々なソフトウェアがインストールされており，情報探索型マルウェアがマルウェア動的解析環境においてファイルアクセスに失敗した重要情報ファイルや重要情報ディレクトリでも，アクセスに成功する可能性がある．そのため，

情報探索型マルウェア検知用しきい値は，このことも考慮しながら決定すべきである．マルウェア作成者は，感染先マシンから漏洩する情報が多いほど大きな利益を生み出すことができるため，情報探索型マルウェアが狙う重要情報の種類が増加することがあっても，減少することは考えにくい．また，マルウェア作成者は，より多くのマシンから情報を漏洩することで大きな利益を生み出すことができるため，作成したシグネチャは多数のマシンに対して有効に働くことが予想される．

提案手法の限界：提案手法はファイルアクセス失敗挙動に基づいてマルウェア感染を判断するが，実際にユーザが利用しているマシンが情報探索型マルウェアに感染した場合，重要情報へのファイルアクセスが成功することも予想される．しかし，マルウェア作成者は，感染先マシンから収集する情報が多いほど大きな利益を生み出すことができるため，多様な情報の収集を試みる．このため，ある重要情報へのファイルアクセスが成功した場合にも，マルウェアが他の重要情報へファイルアクセスに失敗することが予想され，提案手法で検知できる可能性がある．ただし，ユーザが多数の重要情報を複数のクライアントソフトを利用してマシン内に保存している場合，提案手法による検知が難しくなるケースもある．一方，提案手法は監視対象とするディレクトリを動的解析の結果から抽出しているため，マルウェア動的解析で観測したことのないディレクトリを狙った攻撃は検知することができない．したがって，ユーザの挙動の監視やディレクトリ内にあるファイルのリストを取得するなどして，感染先マシン内に実際に保存されている重要情報のみを狙うマルウェアについては検知することができない．また，情報探索型マルウェアであっても，特定の重要情報しか狙わないマルウェアも検知することはできない．このため，提案手法はユーザの重要情報を狙った攻撃をすべて防げるわけではなく，既存技術と組み合わせることで多層防壁を実現することが望ましい．これらのマルウェアに対しては筆者らが論文 [14] で提案している，保護対象マシン内にダミーの認証情報を保存しておくことで，ダミー認証への不正アクセスやダミー認証情報が悪用された場合に，不正侵入を事後検知する方法による検知が有効である．本稿で提案した手法を用いることで，マルウェア作成者が狙う重要情報を特定することができ，論文 [14] の課題である，マルウェアがアクセスする可能性の高い，効果的なダミー認証情報の作成方法を解決できる可能性がある．

提案手法の検知精度：マルウェアが感染先マシン内で情報探索を行うタイミングは様々である．評価実験では，マルウェア検体を実行した直後のファイルアクセスログを対象に検査を行っている．また，解析環境のインターネット接続は限られている．このため，動的解析環境の設定の変更や，再起動後のファイルログ，複数回動的解析を行ったときのファイルアクセスログなど，より多くの環境下で動的解析を行うことで提案手法の検知精度を高くすることができる可能性がある．同様に，保護対象マシンに対しても定期的に検査を行うことで，情報探索型マルウェアを検知できる可能性が高くなる．

提案手法では，監視対象ディレクトリを監視対象ディレクトリ抽出用しきい値により決定しており，監視対象ディレクトリの選び方によって検知精度に影響が出ることは筆者らも認識している．しかし，様々なセキュリティ対策を行っている環境であっても，マルウェアによる情報漏洩が発生してしまう現状において情報漏洩を未然に防ぐことは困難である．このため，情報探索型マルウェア感染を検知できる提案手法は有効であるといえる．

提案手法のコスト：多くの場合，ユーザのファイルアクセスログは短時間のデータでも膨大になる．このため，ファイルアクセスログの中からマルウェアの挙動を抽出することは容易ではない．一方，提案手法ではファイルアクセス失敗にのみ注目しており，ログの収集やマッチングにかかるコストは高くない．実際に，4 章の評価実験に用いた正規ユーザのファイルアクセス失敗ログは最大でも 5 分あたり 123 MB であった．また，提案手法を用いて作成するシグネチャは，1 つのシグネチャで多数のマルウェア検体や複数のマルウェアファミリを検知できる点は提案手法の利点である．作成したシグネチャが，本実験に使用した情報探索型マルウェア検体に共通して見られる特徴をとらえているといえる．

提案手法の拡張：提案手法は，重要情報が保存されたディレクトリへの，ファイルアクセス失敗挙動を監視することで情報探索型マルウェアを検知する．しかし，マルウェアがつねにファイルシステム上に保存された情報を狙うとは限らない．たとえば，Windows マシンの多くはログインしているユーザのパスワードをレジストリ上に保持しており，こうした情報を狙うマルウェアの存在が報告されている [22]．提案手法のコンセプトは，重要情報が保存されているリソースに対して広く適用可能であり，提案手法を拡張することでさらに多くの攻撃を検知できる可能性がある．

提案手法は，情報探索型マルウェアから発生するファイルアクセス失敗挙動を用いてマルウェア検知を行う．このとき，ファイルアクセスを行ったプロセスや実際のアクセス先ファイルなどについては考慮していない．しかし，これらの情報がマルウェアの検知に有効な可能性がある．そこで，今後はアクセス先とアクセス元に注目することで提案手法の拡張を目指す．一方で，提案手法は正規プロセスと同じ名前のプロセスを立ち上げるマルウェアや正規プロセスにインジェクションを行うマルウェアの情報探索活動を検知することができる点は，提案手法の利点である．

6. まとめと今後の課題

本稿では，重要情報へのファイルアクセス失敗挙動に基づく情報探索型マルウェア検知手法を提案した．実マルウェア検体を用いた評価実験の結果，提案手法を用いてマルウェアの情報探索活動を検知することができた．また，正規ユーザのファイルアクセス失敗ログに対して提案手法を適用したところ，情報漏洩の有無を判断するしきい値を調整することで検知能力を保ったまま誤検知を十分に小さくすることができた．今後の課題は，より多くの検体と正規ユーザのログを用いた評価，提案手法の拡張である．

謝辞 本研究の一部は，文部科学省国立大学改革強化推進事業の支援を受けて行われた．A part of this work was conducted under the auspices of the MEXT Program for Promoting the Reform of National Universities.

参考文献

[1] Baecher, P., Koetter, M., Holz, T., Dornseif, M. and Freiling, F.C.: The Nepenthes Platform: An Efficient Approach to Collect Malware, *Proc. 9th International Symposium on Recent Advances in Intrusion Detection* (*RAID 2006*), pp.165–184 (2006).

[2] Bowen, B., Hershkop, S., Keromytis, A. and Stolfo, S.: Baiting Inside Attackers Using Decoy Documents, *Proc. Security and Privacy in Communication Networks*, *ICST'09*, LNICST, Vol.19, pp.51–70 (2009).

[3] Cui, W., Katz, R. and Tan, W.: Design and implementation of an extrusion-based break-in detector for personal computers, *The 21st Annual Computer Security Applications Conference* (*ACCSA 2005*), pp.360–370 (2005).

[4] 角丸貴洋，島成 佳，吉岡克成：組織ネットワークにおける内部攻撃に対する模擬的欺瞞方式，情報処理学会コンピュータセキュリティシンポジウム 2014（CSS 2014），2B4-4 (2014).

[5] 吉川亮太，神薗雅紀，吉岡克成，松本 勉：保護対象ホスト群の状態の類似性に着目した悪性プロセスの検知手法の提案，情報処理学会コンピュータセキュリティシンポジウム 2014（CSS 2014），2B4-3 (2014).

[6] Kührer, M., Hoffmann, J. and Holz, T.: CloudSylla: Detecting Suspicious System Calls in the Cloud, *Proc. Stabilization, Safety, and Security of Distributed Systems*, *SSS2014*, LNCS, Vol.8756, pp.63–77 (2014).

[7] Lanzi, A., Balzarotti, D., Kruegel, C., Christodorescu, M. and Kirda, E.: AccessMiner: using system-centric models for malware protection, *Proc. 17th ACM Conference on Computer and Communications Security*, pp.399–412 (2010).

[8] Maloof, M. and Stephens, G.: ELICIT: A System for Detecting Insiders Who Violate Need-to-know, *Proc. Recent Advances in Intrusion Detection*, *RAID'07*, LNCS, Vol.4637, pp.146–166 (2007).

[9] 中里純二，津田 侑，高木彌一郎，衛藤将史，井上大介，中尾康二：ホスト型 IDS を用いた標的型攻撃対策，情報処理学会コンピュータセキュリティシンポジウム 2014（CSS 2014），2B2-3 (2014).

[10] Nguyen, N., Reiher, P. and Kuenning, G.: Detecting insider threats by monitoring system call activity, *Proc. 2003 IEEE Information Assurance Workshop on Systems, Man and Cybernetics Society*, pp.45–52 (2003).

[11] Salem, M., Hershkop, S. and Stolf, S.: A Survey of Insider Attack Detection Research, *Proc. Insider Attack and Cyber Security*, *AIS*, Vol.39, pp.69–90 (2008).

[12] Stolf, S., Apap, F., Eskin, E., Heller, K., Hershkop, S., Honig, A. and Svore, K.: A Comparative Evaluation of Two Algorithms for Windows Registry Anomaly Detection, *International Journal of Computer Security 2005*, Vol.13, No.4, pp.659–693 (2005).

[13] Yin, H., Song, D., Egele, M., Kruegel, C. and Kirda, E.: Panorama: Capturing System-wide Information Flow for Malware Detection and Analysis, *Proc. 14th ACM Conference on Computer and Communications Security*, pp.116–127 (2007).

[14] 米持一樹，田辺瑠偉，吉岡克成，松本 勉：ダミーの認証情報を用いて不正侵入を事後検知する方法，電子情報通信学会 暗号と情報セキュリティシンポジウム 2013（SCIS2013），4C2-5 (2013).

[15] Yoshioka, K., Inoue, D., Eto, M., Hoshizawa, Y., Nogawa, H. and Nakao, K.: Malware Sandbox Analysis for Secure Observation of Vulnerability Exploitation, *IEICE Trans.*, Vol.E92D, No.5, pp.955–966 (2009).

[16] Yoshioka, K., Kasama, T. and Matsumoto, T.: Sandbox Analysis with Controlled Internet Connection for Observing Temporal Changes of Malware Behavior, *Proc. 4th Joint Workshop on Information Security*, *JWIS 2009*, 3a-2 (2009).

[17] Yoshioka, K. and Matsumoto, T.: Multi-pass Malware Sandbox Analysis with Controlled Internet Connection, *IEICE Trans.*, Vol.E93-A, No.1, pp.210–218 (2010).

[18] Yuill, J., Zappe, M., Denning, D. and Feer, F.: Honeyfiles: Deceptive Files for Intrusion Detection, *Proc. 5th Annual IEEE Information Assurance Workshop on Systems, Man and Cybernetics Society*, pp.116–122 (2004).

[19] Yuill, J., Denning, D. and Feer, F.: Using Deception to Hide Things from Hackers: Processes, Principles and Techniques, *Journal of Information Warfare* (2006).

[20] CANNON：SNS が新たなマルウェア感染ルートになりつつある—どうする SNS セキュリティ，入手先 ⟨http://canon-its.jp/eset/malware_info/threat/150428/⟩（参照 2015-11-12）.

[21] JPCERT：STOP!! パスワード使い回し!!パスワードリスト攻撃による不正ログインに向けた呼びかけ，入手先 ⟨https://www.jpcert.or.jp/pr/2014/pr140004.html⟩（参照 2015-11-12）.

[22] HKTL_PWDUMP, available from ⟨http://about-threats.trendmicro.com/Malware.aspx?language=jp&name=HKTL_PWDUMP⟩ (accessed 2015-11-12).

[23] IIJ-SECT：BHEK2 を悪用した国内改ざん事件の続報，入手先 ⟨https://sect.iij.ad.jp/d/2013/03/225209.html⟩（参照 2015-11-12）.

[24] Process Monitor, available from ⟨https://technet.microsoft.com/ja-jp/sysinternals/bb896645⟩ (accessed 2015-05-25).

[25] McAfee：Gumblar（ガンブラー）について，入手先 ⟨http://www.mcafee.com/japan/security/gumblar.asp⟩（参照 2015-11-12）.

[26] VirusTotal, available from ⟨https://www.virustotal.com/⟩ (accessed 2015-05-25).

[27] W32.Qakbot, available from ⟨http://www.symantec.com/ja/jp/security_response/writeup.jsp?docid=2009-050707-0639-99⟩ (accessed 2015-05-03).

田辺 瑠偉 （学生会員）

2014 年 3 月横浜国立大学大学院環境情報学府情報メディア環境学専攻博士課程前期修了．修士（情報学）．同年 4 月横浜国立大学大学院環境情報学府情報メディア環境学専攻博士課程後期に進学．情報セキュリティ，特にネットワークセキュリティの研究に従事．

笠間 貴弘 （正会員）

2014 年 3 月横浜国立大学大学院環境情報学府情報メディア環境学専攻博士課程後期修了．博士（工学）．2011 年 4 月より情報通信研究機構に研究員として入所．マルウェア解析やネットワーク攻撃観測・分析等サイバーセキュリティの研究開発に従事．2011 年情報処理学会山下記念研究賞受賞．

吉岡 克成 （正会員）

2005 年 3 月横浜国立大学大学院環境情報学府情報メディア環境学専攻博士課程後期修了．博士（工学）．同年 4 月独立行政法人情報通信研究機構にて研究員として勤務．2007 年 12 月より横浜国立大学学際プロジェクト研究センター特任教員（助教）．2011 年 4 月横浜国立大学大学院環境情報研究院准教授．マルウェア解析やネットワーク攻撃観測・検知等の情報システムセキュリティの研究に従事．2009 年文部科学大臣表彰・科学技術賞（研究部門）受賞．

松本 勉 （正会員）

1986 年 3 月東京大学大学院工学系研究科電子工学専攻博士課程修了，工学博士．同年 4 月横浜国立大学講師．2001 年 4 月より同大学院環境情報研究員教授．2014 年 12 月より同大学先端科学高等研究員（IAS-YNU）情報物理セキュリティ研究ユニットリーダーを兼務．ネットワーク・ソフトウェア・ハードウェアセキュリティ，暗号，耐タンパー技術，生体認証，人工物メトリクス等の「情報・物理セキュリティ」の研究教育に 1981 年より従事．1982 年にオープンな学術的暗号研究を目指した「明るい暗号研究会」を 4 名で創設．2005～2010 年国際暗号学会 IACR 理事．1994 年第 32 回電子情報通信学会業績賞，2006 年第 5 回ドコモ・モバイル・サイエンス賞，2008 年第 4 回情報セキュリティ文化賞，2010 年文部科学大臣表彰・科学技術賞（研究部門）各受賞．