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

吉川 亮太† 神薗 雅紀† 吉岡 克成† 松本 勉†

†横浜国立大学
〒240-8501 神奈川県横浜市保土ヶ谷区常盤台 79-7
{kikkwa-ryota-zg, kamizono-masaki-fn}@ynu.jp, {yoshioka, tsutomu}@ynu.ac.jp

あらまし 近年，特定の企業を対象とした標的型攻撃が脅威となっている．その対策技術として，組織内のサーバやセキュリティ製品が生成するログを収集し，相関分析することで不正侵入を検知する技術が注目されている．しかし，侵入を検知するルールが不十分であることや検知能力が管理者の設定に大きく依存するといった問題があるため，様々な企業内ネットワークで汎用的に利用可能な検知手法が重要である．本稿では，企業内ネットワークのホスト群の内部状態の類似性に着目し，逸脱した状態のプロセスを正規プロセスになりすますプロセスとして検知する手法を提案し，標的型攻撃の感染経路における提案手法の有効性を報告する．

Proposal of a Detection Method of Malicious Process by Focusing on the Similarity of the States of the Hosts

Ryota Kikkawa† Masaki Kamizono†

Katsunari Yoshioka† Tsutomu Matsumoto†

†Yokohama National University
79-7, Tokiwadai, Hodogaya-ku, Yokohama-shi, Kanagawa, 240-8501, JAPAN
{kikkwa-ryota-zg, kamizono-masaki-fn}@ynu.jp, {yoshioka, tsutomu}@ynu.ac.jp

Abstract In recent years, targeted attacks have become a bigger threat. One of the countermeasures against targeted attack in consideration is to collect various logs from endpoints and network devices for conducting correlation analysis so as to detect the sign of intrusions among the logs. But configuring detection rules suitable for each organization to be protected is still an open problem. In this paper, we propose a method to detect malicious process that pretends to be legitimate process by focusing on the similarity of the states of the hosts in an enterprise network. The experiment shows that several malware samples can be detected by the proposed method.

1 はじめに

近年，特定の組織を対象とした標的型攻撃が大きな脅威となっている[1]．標的型攻撃では，標的となる組織内のユーザへ標的型メール等を介して侵入し，マルウェアを用いて情報収集を行いつつ権限の高いユーザのホストへ感染を広げることによって，重要情報を取得し，攻撃者の基へ送信することが知られている[2]．

標的型攻撃の対策として標的型メールを検知するという入口対策が挙げられる[3][4]．しかし，攻撃者は標的組織の取引相手などの周辺情報を調べた上で巧妙に細工された標的型メールで人的ミスを誘う場合があり，正規の業務メールと標的型メールを完全に見分けることは困難といえる．さらに標的型攻撃で利用されるマルウェアは，アンチウィルス等によるシグネチャマッチングによって検知することが難しく[5]，感染後も正規プロセスになりすましたり，正規プロセスを改ざんすることでシステム内に存在するため，検知をより困難にしている．

このような背景から，マルウェアの侵入を防ぐという入口対策に加えて，感染後の活動を早期検知する内部対策や出口対策の重要性が指摘されている．内部対策として注目されている技術がログ解析による不正侵入検知である．組織内のサーバやセキュリティ製品，ファイアウォールなどから生成されるログを収集し，相関分析することによって攻撃者の侵入や侵入後の攻撃の痕跡を特定し早期検知を目指す．大量のログデータを高速で処理する技術の向上により，ログ解析による対策技術の発展は著しく，様々な機器でログを収集し保管することが重要となっている[6]．これらのログ解析の効果は，セキュリティ管理者による検知ルールの設定に大きく依存しており，様々な企業内ネットワークで汎用的に利用可能な検知手法が希求されている．

そこで，本稿では類似した状態や設定のホスト群が多数存在する企業内ネットワークのような環境において収集したログを利用し，ホスト群の各プロセスの状態の類似性に着目することで，他と逸脱した状態のプロセスを悪性プロセスとして検知する手法を提案する．具体的には，各エンドユーザのホストから生成されるログを収集し，同一名プロセスの持つ属性値(実行可能ファイルのパス，実行時のカレントディレクトリ，実行可能ファイルを実行したユーザ名，説明，会社名，親プロセス，自動実行場所，DEP の ON/OFF 状態)を比較することによって，正規プロセスになりすます悪性プロセス(以後，なりすましプロセスとする)を検知する．実マルウェアを用いた評価実験では，標的型攻撃において典型的なマルウェアの感染シナリオを再現し，提案手法の有効性を示す．

本稿の構成は次の通りである．第 2 章で正規プロセスになりすます悪性プロセスの検知手法について述べる．第 3 章で提案手法について説明し，第 4 章で評価実験について記述し，第 5 章で評価実験の考察を行う．最後に，第 6 章でまとめと今後の課題を述べる．

2 関連研究

不審な通信を生成するプロセスが正規プロセスか正規プロセスになりすます悪性プロセスであるかを判定するプロセス解析システムに関する研究がこれまで行われている[7][8]．

文献[7]では，ホストマシン上にマルウェアに感染していないことが保証された仮想マシンをリファレンスとして用意し，検査対象プロセスとメモリイメージの比較を行うことで悪性コードの挿入等を検知する手法を提案している．

文献[7]の手法では，検査対象のホストマシンにおいて新たにアプリケーションをインストールする際には，リファレンスである仮想マシンにも同様にインストールを行い，ホストマシンと仮想マシンを同じ環境を保つ必要があるため，仮想マシン側がマルウェア感染していないという前提をどのように保つかが重要といえる．

本提案手法では，保護対象組織内のホスト群の多くがマルウェア感染していないという前提の下，内部状態が類似するホスト群から状態が逸脱したなりすましプロセスを検知する手法を提案する．

3 提案手法

本章では，なりすましプロセスを検知する提案手法について説明する．本手法の前提条件として，検査対象となる組織内ネットワークのホスト群は管理されており，同一のソフトウェア群が同様の設定で導入されているものとし，少なくともこのようなホスト群は 3 台以上存在するとする．

標的組織内で諜報活動を行う際，攻撃者は OS で一般的に起動されるようなプロセスと同じ名前でマルウェアを実行することで正規プロセスになりすまし，検知されにくくすることが想定されるため，提案手法では複数のホスト上の同一名プロセス群の状態を比較することで，他のホストと状態が逸脱するなりすましプロセスの検知を行う．

3.1 提案手法の概要

提案手法では，各エンドユーザのホストから生成されるプロセスに関する情報をログとして定期的にログ解析サーバへ送る．このプロセスに関する情報とは，動作中のプロセスが持つ属性値(実行可能ファイルのパス，実行時のカレントディレクトリ，実行可能ファイルを実行したユーザ名，説明，会社名，親プロセス，自動実行場所，DEP の ON/OFF 状態)である．これらの属性に関する説明を表 1 に記載する．

表 1 プロセスの持つ属性値の説明

プロセスの属性	説明
実行可能ファイルのパス	実行可能ファイルが保存されているフルパス
実行時のカレントディレクトリ	プロセスを実行した際のディレクトリ
実行したユーザ名	ファイルを実行したユーザ，サービス
説明・会社名	実行可能ファイルが持つプロパティ情報
親プロセス	当該プロセスを生成した親プロセス名
自動実行場所	システムレジストリに書かれている値
DEP の ON/OFF 状態	データ実行防止が有効となっているか

各ホストから収集したログを解析サーバ上で解析を行い，なりすましプロセスを検知する．解析サーバ上で行う解析処理の流れを以下に示す．

(1) 3 台以上のホスト上で動作する同一名プロセスのリストを抽出する．
(2) リストに含まれる各プロセス名について，当該プロセス名と同一の名前をもつ全てのプロセスを同一名プロセス群として抽出する．
(3) 上記の(2)で得られた全ての同一名プロセス群に対してそれぞれ，プロセスの各属性に関する以下の検査を行う．
(3-1) 同じ属性値を持つプロセスでクラスタを作成する．
(3-2) 作成されたクラスタが 1 つである場合は正常であると判定する．クラスタが複数ある場合はクラスタに含まれるプロセス数が最少のクラスタを異常とし，そのクラスタ内のプロセスを全てなりすましプロセスとして検知する．

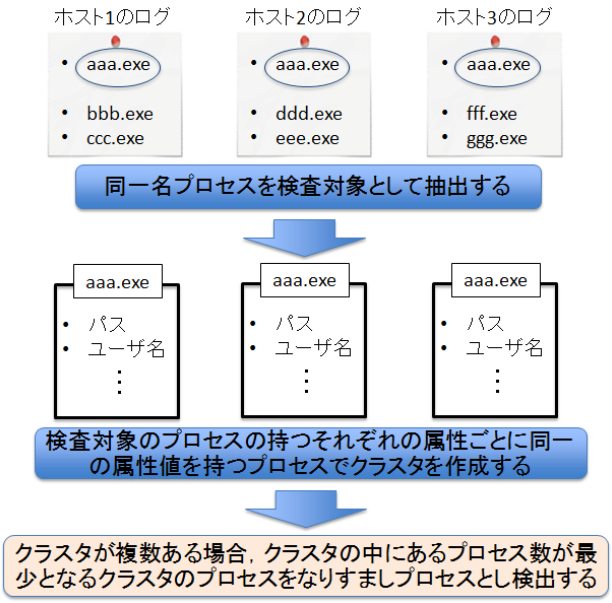

図 1 提案手法の概要

3.2 例外処理

正規プロセスにおいても，実行の状況に応じて属性値が固定でない場合がある．例えば「svchost.exe」の場合，状況に応じてユーザ名は「¥SYSTEM」，「¥NETWORK SERVICE」，「¥LOCAL SERVICE」となる．今回，なりすまし先の正規プロセスとしてよく使われる「svchost.exe」に対してユーザ名が「¥SYSTEM」，「¥NETWORK SERVICE」，「¥LOCAL SERVICE」となる場合は，検出を行わない設定とした．

また，提案手法では管理された企業内ネットワークのようにインストールされたアプリケーションのバージョンや設定が同一のホスト群を前提としているが，このようなネットワークにおいても業務の事情等により特殊な設定を行う場合があり得る．

これらの例外のうち，保護対象システム固有の例外については，システム更新時等にホワイトリスト化することが可能と思われるが，業務の事情等による設定変更に対しては，常時対応する必要がある．これらの例外対応の方法や効率化については今後の課題としたい．

4 評価実験

本章では，提案手法の評価実験について記述する．評価実験では，標的型攻撃で想定される典型的なマルウェアの感染シナリオを，実マルウェアを用いて再現し，提案手法により検知できるかを確認する．また，不正侵入が発生していないという前提で大学研究室内の PC6 台から収集したログに対して提案手法による解析を行い，誤検知の有無を検証する．

収集するプロセスに関するログは，プロセス監視ツールである Process Monitor[9]，Process Explorer[10] を組み合わせ，動作中のプロセスが持つ属性値を収集する．

4.1 実験の概要

評価実験では，同一の状態のホストを 3 台用意し，そのうちの 1 ホストをそれぞれの感染シナリオに基づきマルウェア検体に感染させた状態でログを収集する．感染シナリオについては 4.2 節で説明する．残りの 2 ホストは，検体に感染していない状態でログを収集する．そして，収集した 3 つのログに対して提案手法による解析を行い，感染ホスト上の悪性プロセスの状態が他の 2 ホストと比べて逸脱した状態にあり，提案手法により検知できるかを評価する．

本評価実験に用いたマルウェア検体は，事前調査によりなりすましプロセスを生成することが確認された検体の中から特徴的な 6 検体を利用した．

また，誤検知検証実験では，大学研究室内の PC6 台から起動後の状態でプロセスに関するログを収集し，提案手法による解析を行った．

表 2 評価実験に用いた検体

検体番号	MD5 ハッシュ値	検知名(Trend Micro)	作成するプロセス名
検体 1	ff02b26a532614b152ab520a566ef16e	Mal_DRPR-3	winlogon.exe
検体 2	e27c97959655f4d13258aacaf6f573da	BKDR_ANDROM.NIL	explorer.exe
検体 3	198ef662c19545b12469a1c63d20c012	TROJ_GEN.RCBCEL8	iexplore.exe
検体 4	f4522ee4da43be0c1853bc416c008620	BKDR_EVILOGE.SM	svchost.exe
検体 5	571dd3fba036334f634ecce e0e978319	Mal_DRPR-3	svchost.exe
検体 6	29915f304ed73d06db0d9adc221b23a8	BKDR_ANDROM.FWC	svchost.exe

4.2 感染経路の再現

感染シナリオとして，組織内のユーザへの標的型メールにより侵入されるケースがある．IPA が発行した「標的型攻撃メールの傾向と事例分析＜2013 年＞」[11]によると，標的型メールには，大きく「添付ファイル型」と「URL リンク型」に分けられ，「添付ファイル型」が全体の約 97%を占めている．添付ファイルや URL リンク先からダウンロードされる不審ファイルの種別として，実行ファイルが約 60%，文書ファイルが約 30%と報告されている．

そこで，標的型メールによる侵入の再現として，標的型メールに実行可能ファイルであるマルウェアの圧縮ファイルが添付されており，ユーザが誤って実行してしまう場合(シナリオ 1)，悪性 PDF ファイルが添付されており，このファイルを開くことによってアプリケーション(Adobe Reader)の脆弱性を突かれ，PDF ファイルに埋め込まれているマルウェア本体が実行されてしまう場合(シナリオ 2)，メール等の本文に含まれる URL リンク先が悪性サイトであり，これにアクセスすることでブラウザ(Internet Explorer)の脆弱性を突くドライブバイダウンロード攻撃が発生しマルウェアに感染してしまう場合(シナリオ 3)の 3 つの感染シナリオを再現する．

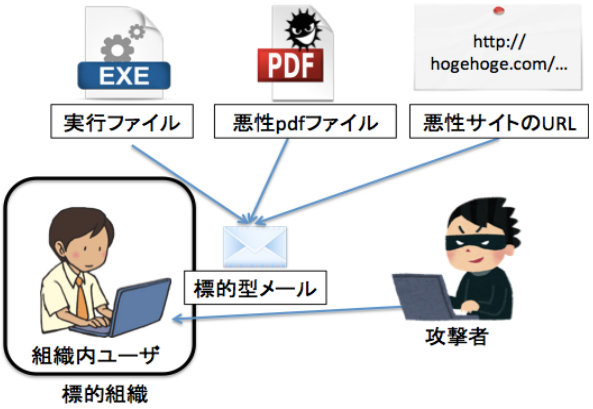

図 2 感染シナリオ 1~3 のイメージ図

また，侵入後に組織内で諜報活動や権限の高いユーザの持つアクセス権の窃取のために感染を広げるケースがある．文献[12]では，Windows のリモートホストで任意のコマンドやプログラムを実行することができるアプリケーションである PsExec[13]が標的型攻撃で利用されることが説明されている．そこで，侵入済のホスト(ホスト A とする)を踏み台ホストとし，同一ネットワーク内の他のホスト(ホスト B とする)へ感染を広げる攻撃の再現として，ファイル共有サービスの脆弱性を突きリモートエクスプロイト攻撃により感染を広げる場合(シナリオ 4)，踏み台ホストであるホスト A 上で PsExec を使いリモートホストであるホスト B へマルウェア検体を送り実行させることにより感染を広げる場合(シナリオ 5)を再現する．

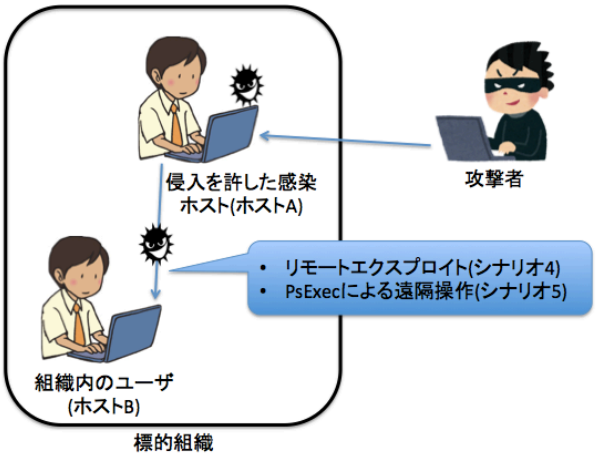

図 3 感染シナリオ 4，5 のイメージ図

4.2.1 シナリオ 1

メールに暗号化zipにより圧縮したマルウェア検体を添付し，ホストに送信する．本評価実験では，メーラソ

フトウェアとしてThunder Bird[14]を用いることとし，これにより標的型メールを受信し，添付ファイルを開くことによる感染を再現する．

4.2.2 シナリオ 2

脆弱性検証ツールである Metasploit Framework[15](以後，Metasploit とする)により Adobe Reader 9.2 の脆弱性(CVE-2010-1240)を突く悪性 pdf ファイルを作成する．当該 pdf ファイルは脆弱性のある Adobe Reader により閲覧される際に，ユーザ権限を奪取し内部に埋め込まれた実行可能ファイル(マルウェア検体本体)を実行させるものである．シナリオ 1 と同様にメーラから当該 pdf ファイルを開くことによる感染を再現する．

4.2.3 シナリオ 3

Metasploitにより悪性Webサイトを構築し，これにアクセスさせることによりブラウザ(Internet Explorer 6)の脆弱性(CVE-2010-0806)を突き，ユーザ権限を奪取し，検体をダウンロード・実行させる，ドライブバイダウンロード攻撃を再現する．シナリオ 1，シナリオ 2 と同様に，上記の悪性 Web サイトの URL が記載されたメールを受信後，ビューア上でハイパーリンクをダブルクリックすることでブラウザを起動し，悪性 Web サイトにアクセスさせることによる感染を再現する．

4.2.4 シナリオ 4

侵入済みのホスト A からファイル共有サービスの脆弱性(CVE-2008-4250)を突き，リモートホスト B の SYSTEM 権限を奪取した後，ホスト B へ検体を送り，実行することにより感染の拡大を再現する．

4.2.5 シナリオ 5

侵入済みのホスト A から PsExec を使いリモートホスト B に管理者権限ユーザとしてアクセスし，マルウェア検体を送り実行することで感染の拡大を再現する．

前提条件として，攻撃者はリモートホスト B の ID/Pass を取得済みである，または，Pass the Hash 攻撃等でリモートホスト B にアクセスできる状況であるとする．

4.3 実験環境

シナリオ 1~3 は，ゲストマシン 1 上で行い，シナリオ 4，5 では，ゲストマシン 2，3 が侵入済みホスト A となり，リモートホスト B に対応するゲストマシン 1 へ感染を広げるものとする．収集したログの解析はホストマシン上で行う．また，誤検知検証実験では，研究室内のホストのうち，OS が windows7 のもの 6 台を対象に行った．OS 情報は表 3 に記載する．

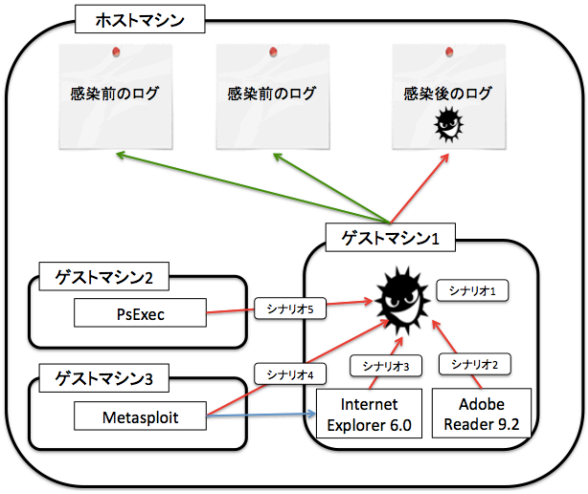

図 4 評価実験の概要

表 3 評価実験で用いたホストの OS 情報

マシンの名称	OS
ホストマシン	Ubuntu14.04
ゲストマシン 1	Windows XP SP3
ゲストマシン 2	Windows XP SP2
ゲストマシン 3	Kali Linux 1.0.7
誤検知検証用マシン	Windows 7 SP1

4.4 実験結果

実験結果を表 4~6 に示す．これらの表では，感染ホストのプロセスの属性値が他のホストと異なっており，提案手法により検知が可能な場合に感染ホストの属性値を記載している．なお，シナリオ 4，5 において，検体 2，6 はなりすましプロセスを生成しなかった．感染シナリオを再現した評価実験では，誤検知はなかった．

5 考察

実験の結果，今回用意した検体が生成するなりすましプロセスは，シナリオ 1~5 のいずれにおいても提案手法により検知できることを確認した．以下では，検知に用いたプロセスの属性毎に詳しく考察する．

- **パス**

検体 1，2，6 が生成するなりすましプロセスは，正規プロセスとパスが異なっており，提案手法により検知することができた．Windows の重要ファイルは WFP 機能により保護されているため，正規プロセスを生成するための実行可能ファイル(以後，正規ファイルとする)と同じディレクトリにファイルインジェクションすることは困難であり，これらの検体が生成したプロセスは正規ファ

表 4 評価実験で確認された正規プロセスとは異なる属性値(シナリオ 1~3)

検体番号	パス	カレントディレクトリ	ユーザ名	説明	会社	親プロセス	自動実行場所	DEP
検体1	C:¥Documents and Settings¥username¥Local Settings¥Temp¥winlogon.exe	C:¥Documents and Settings¥username¥Local Settings¥Temp¥	¥username	なし	なし	sample.exe		なし
検体2	C:¥Documents and Settings¥All Users¥explorer.exe			なし	Loihytgvfd		HKLM¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion¥Run¥Start WingMan Profiler	なし
検体3		C:¥WINDOWS¥system32¥	¥SYSTEM			svchost.exe		
検体4		シナリオにより変化※1	¥username			sample.exe		
検体5			¥username			2.8-install.exe		
検体6	C:¥Documents and Settings¥All Users¥svchost.exe	C:¥Documents and Settings¥username¥	¥username	なし	なし	なし	HKLM¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion¥Run¥SunJavaUpdateSched	なし

※1・・・シナリオ 1 では「C:¥Documents and Settings¥username¥」であり，シナリオ 2 は「C:¥Documents and Settings¥username¥My Documents¥」，シナリオ 3 は「C:¥Documents and Settings¥username¥デスクトップ¥」であった．シナリオ 1~3 ではその他の値は同値である．

表 5 評価実験で確認された正規プロセスとは異なる属性値(シナリオ 4)

検体番号	パス	カレントディレクトリ	ユーザ名	説明	会社	親プロセス	自動実行場所	DEP
検体 1		C:¥WINDOWS¥TEMP¥				sample.exe		なし
検体 2	なりすましプロセスを作成しない							
検体 3		C:¥WINDOWS¥system32¥	¥SYSTEM			svchost.exe		
検体 4						sample.exe		
検体 5						2.8-install.exe		
検体 6	なりすましプロセスを作成しない							

表 6 評価実験で確認された正規プロセスとは異なる属性値(シナリオ 5)

検体番号	パス	カレントディレクトリ	ユーザ名	説明	会社	親プロセス	自動実行場所	DEP
検体 1	C:¥Documents and Settings¥username¥Local Settings¥Temp¥winlogon.exe	C:¥Documents and Settings¥username¥Local Settings¥Temp¥	¥username	なし	なし	sample.exe		なし
検体 2	なりすましプロセスを作成しない							
検体 3		C:¥WINDOWS¥system32¥	¥SYSTEM			svchost.exe		
検体 4			¥username			sample.exe		
検体 5			¥username			2.8-install.exe		
検体 6	なりすましプロセスを作成しない							

表 7 誤検知検証実験で確認された正規プロセスの誤検知

誤検知された属性	誤検知の原因	プロセス例
パス，会社名，説明	アプリケーションのバージョンが異なる	IMJPCMNT.EXE，nvvsvc.exe，jusched.exe，
親プロセス	同じ名前の子プロセスを生成する	ccsvchst.exe，nvvsvc.exe
パス，カレントディレクトリ	インストールするディレクトリが異なる	procexp.exe，procexp64.exe
ユーザ名	複数のユーザ名，サービスが実行する	WmiPrvSE.exe，IMJPCMNT.EXE，ccsvchst.exe

イルとは異なるディレクトリに保存されていることからパスに差異がでたものと思われる. なお, 正規プロセスにコードを挿入するマルウェアについては, パスは正規プロセスと同様であることから検知できない.

・ カレントディレクトリ

検体 4 は, 感染シナリオによってカレントディレクトリが変化したため, 正規プロセスと異なるカレントディレクトリを示す場合が確認された. 検体 4 以外の全ての検体は感染シナリオに影響を受けずカレントディレクトリが決まっていたが, 正規プロセスと異なる場合も確認された. 感染シナリオ上の制限でカレントディレクトリを自由に設定できない場合には, 検知を行う上で有効な属性といえる.

・ ユーザ名

検体 3 は感染シナリオに影響を受けることなくユーザ名は¥SYSTEMとなった. それ以外の検体においては感染シナリオによってユーザ名が変化した. シナリオ 4 では, ¥SYSTEM となり, それ以外のシナリオでは感染時にログインしているユーザ名であった. カレントディレクトリの場合と同様に, 感染シナリオ上の制限から正規プロセスとなりすましプロセスのユーザ名を一致させることが困難な場合, ユーザ名に基づく検知は有効といえる. 但し, シナリオ 5 の場合, PsExec には¥SYSTEMにより実行するというオプションがあるため, ユーザ名を変化させることは容易である.

・ 説明/会社名

プロセスの説明/会社名は, 実行可能ファイルが持つプロパティ情報から得られる情報であり, 正規プロセスと同様の情報を攻撃者が設定することは技術的には容易であるが, 実際にはこれらの情報を設定していない検体も多く見られた. なお, 正規プロセスにコードを挿入することによりなりすましプロセスを生成する場合は, この属性は正規プロセスと一致するため, 検知はできない.

・ 親プロセス

検体 2 は, 生成するプロセスのなりすまし先が親プロセスを持たない正規プロセスであり, 自身の生成するなりすましプロセスも親プロセスを持たないため検知することができなかった.

検体 1, 4 は, 親プロセス名が今回の実験で用いた検体のファイル名 sample.exe となり, 正規プロセスとの差異が確認された. また, なりすましプロセスが, 既に起動している正規プロセスの子プロセスとして生成される場合や検体が独自で生成した子プロセスに生成される場合が確認され, これらも正規プロセスとの差異が確認された. しかし, 検体名や検体が生成する子プロセスを正規の親プロセス名と同じ名前とすることで, 検知回避は容易と思われる.

・ 自動実行場所

再起動時になりすましプロセスを起動する検体 2, 6 は, 自動実行場所が異なっていた.

このことより, 再起動後に自動実行するプロセスに関しても検知することが可能であることが確認できた.

・ DEP(データ実行防止)

自動実行場所と同様に検体 2, 6 はすべての感染シナリオにおいて検知することが確認できた. また, 検体 1 もすべての感染シナリオで検知することができた. 正規プロセスである「winlogon.exe」は, DEP により保護されているが, 検体 1 が生成する「winlogon.exe」では, 保護されておらず検知に成功した.

今回行った評価実験において, 検知が成功した属性のうち, 説明/会社名と親プロセスは攻撃者が注意深く設定し作成したなりすましプロセスであれば検知を回避されてしまう. しかし, カレントディレクトリとユーザ名は, 感染シナリオによって影響を受ける場合がある. パスについては, 検知を回避するために正規プロセスにコードを挿入するという攻撃にせざるを得ない. これらの属性を検知要素として使うことは, 攻撃者の感染活動を制限することに繋がると考えられる.

また, 上で述べた正規プロセスにコードを挿入するタイプのマルウェアは, 今回の属性だけでは検知することは難しい. この点を解決するためにプロセスが呼び出す dll ファイルの有無を検査対象属性とすることで検知する方法を検討中である.

また, 誤検知に関する検証実験では, 3 台以上のマシン上で動作する同一名プロセスが 56 件あり, 15 件のプロセスで誤検知が確認された. 誤検知のうちの 8 件がアプリケーションのバージョンの違いやインストールするディレクトリの違いによる誤検知であった. これらの誤検知に関しては, 今回の実験環境が大学研究室内のマシンであり, アプリケーションのバージョンが統一されていないことと, 利用者が独自にインストールできる環境であることが原因であると考えられ, 企業等の組織では, アプリケーションのバージョンや設定を統一するなどといった組織内ルールを設けることにより誤検知を減らすことができると考えられる. また, その他の 4 件は, 複数のユーザ名を持つプロセスによる誤検知であった. これらに関しては, 事前にプロセスの属性に関するシステム固有の例外についてのホワイトリストを用意することで誤検知を減らすことができると考えられる.

本稿では，アプリケーションのバージョンや設定が同一であることを前提としているが，業務の事情等により特殊な設定を行う場合やプロセスの属性に関するシステム固有の例外がある場合など，例外処理についての検討には課題が残されている．

本稿の提案手法では，各エンドユーザのホストにエージェント型のセキュリティアプライアンスを設置し，エージェントが生成するホストの状態に関するログを基に悪性状態のホストを検出するといった運用形態を想定している．現在では，AssetView[16]や Log Audit Tracker[17]などのエージェント型のセキュリティアプライアンスが数多く出てきており，これらの製品が生成するログの中には，本稿で挙げた一部のプロセスの属性に関する情報も含まれているため，提案手法が適用できる可能性が高いと考えられる．

6 まとめと今後の課題

類似した状態や設定の環境における悪性プロセスの検知手法について提案した．そして，評価実験では，典型的な感染シナリオを再現し提案手法により悪性プロセスを検知することができるか確認を行い，実験に用いた検体についてはすべて検知することができた．しかし，注意深くマルウェアを作成し，実行する場合や正規プロセスにコードを挿入する場合は，現状では検知が難しく改良が必要である．

また，同一名プロセスだけでなく，一文字違いや文字が入れ替わっておりユーザを視覚的に騙す正規プロセスに類似したプロセスを検知対象に加える改良も行う予定である．

謝辞 本研究の一部は，JSPS 科研費 24680006 の助成により行われた．

参考文献

[1] 独立行政法人情報処理推進機構(IPA)，“標的型サイバー攻撃の事例分析と対策レポート”，
http://www.ipa.go.jp/files/000024536.pdf，
(最終閲覧日：2014/08/17)

[2] 独立行政法人情報処理推進機構(IPA)，“「新しいタイプの攻撃」の対策に向けた設計・運用ガイド”，
http://www.ipa.go.jp/files/000017308.pdf，
(最終閲覧日：2014/08/19)

[3] 独立行政法人情報処理推進機構(IPA)“『標的型メール攻撃』対策に向けたシステム設計ガイド”，
http://www.ipa.go.jp/files/000033897.pdf，
(最終閲覧日：2014/08/17)

[4] 日本ネットワークセキュリティ協会(JNSA)“標的型攻撃メールへの対策”，
http://www.jnsa.org/ikusei/spam/07_02.html，
（最終閲覧日：2014/08/17）

[5] MANDIANT，“M-Trends: The Advanced Persistent Threat”，White Paper 2010，

[6] 内閣官房情報セキュリティセンター，“政府機関における情報システムのログ取得・管理の在り方の検討に係る調査報告書”，
http://www.nisc.go.jp/inquiry/pdf/log_shutoku.pdf，
（最終閲覧日：2014/08/21）

[7] 山本 匠，河内 清人，桜井 鐘治，“不審プロセス特定手法の提案”，マルウェア対策研究人材育成ワークショップ 2013，セッション 3B1-2，2013.

[8] 山本 匠，河内 清人，桜井 鐘治，“不審プロセス特定手法の実装および評価”，研究報告コンピュータセキュリティ(CSEC)，2014-CSEC-64 巻，33 号，pp.1-8，2014.

[9] Windows Sysintenals，Process Monitor，
http://technet.microsoft.com/ja-jp/sysinternals/bb896645.aspx，(最終閲覧日：2014/08/21)

[10] Windows Sysintenals，Process Explorer，
http://technet.microsoft.com/ja-jp/sysinternals/bb896653.aspx，(最終閲覧日：2014/08/21)

[11] 独立行政法人情報処理推進機構(IPA)，標的型攻撃メールの傾向と事例分析<2013 年>，
http://www.ipa.go.jp/files/000036584.pdf，
（最終閲覧日：2014/08/17）

[12] 山田 正弘，森永 正信，海野 由紀，鳥居 悟，武仲 正彦，“組織内ネットワークにおける標的型攻撃の諜報活動検知方式”，2014 年暗号と情報シンポジウム，セッション 3A4-3，2014.

[13] Windows Sysintenals，PsExec，
http://technet.microsoft.com/ja-jp/sysinternals/bb897553.aspx，(最終閲覧日：2014/08/17)

[14] Mozilla Japan，ThunderBird，
http://www.mozilla.jp/thunderbird/，
（最終閲覧日：2014/08/17）

[15] Rapid7 LLC，Metasploit Framework，
http://www.metasploit.com/，
（最終閲覧日：2014/08/17）

[16] 株式会社ハンモック，AssetView，
http://www.hammock.jp/assetview/，
（最終閲覧日：2014/08/21）

[17] 株式会社ディアイティ，Log Audit Tracker，
http://www.dit.co.jp/products/lat/index.html，
(最終閲覧日：2014/08/21)