情報処理学会論文誌 Vol.59 No.12 2246–2260 (Dec. 2018)

標的型攻撃に対する マルウェア波及範囲推定手法の提案と評価

島川 貴裕[1,a)] 佐藤 信[1] 佐々木 良一[1]

受付日 2018年2月23日, 採録日 2018年9月7日

概要：近年，金銭や知的財産などの重要情報の窃取を目的として特定の企業や組織を攻撃対象とする標的型攻撃が社会的な問題となっている．標的型攻撃では，初期段階で乗っ取った攻撃基盤をベースに，次々と端末を乗っ取りながら侵害を拡大していくため，マルウェアに感染する端末が 1 台だけでなく組織内の複数端末が感染している可能性が高い．システム停止の組織に対する影響は膨大であり，早急に復旧したいが，すべての感染端末に対する対策をしてからでないとシステムを復旧できない．そのため，攻撃の発覚後には，複数の端末のログを組み合わせた解析により被害範囲の想定が必要となる．しかし，各ログの関連付けは時間を要するうえに高度な技術力が必要である．そこで本稿では，複数の端末のプロセスと通信試行のログを解析・突合することでマルウェアの波及範囲を推定する手法を提案する．また，提案手法を実現するプログラムを開発し評価実験を行った．その結果，提案手法により感染経路が分岐している場合や感染先の端末でマルウェアが実行されずに潜伏状態にある場合にも対応可能であることを確認した．そのため，提案手法により侵入源の端末から感染拡大行為によって感染した端末群をすべて特定することができ，網羅的に感染拡大を追跡できると考える．また，調査に必要な情報を抽出した後のログは元のログに対し 2%未満にまで圧縮でき，調査処理時間は感染端末の増加に対しおおよそ線形的な増加に抑えられた．

キーワード：デジタルフォレンジック，ネットワークフォレンジック，標的型攻撃，ログ解析

Proposal and Evaluation on Estimation Method of Spreading Range of Malware Caused by Targeted Attacks

Takahiro Shimakawa[1,a)] Makoto Sato[1] Ryoichi Sasaki[1]

Received: February 23, 2018, Accepted: September 7, 2018

Abstract: In recent years, targeted attacks aiming at specific companies and organizations for the purpose of stealing important information such as money and intellectual property are becoming a social problem. In targeted attacks, infringement expands while taking over the terminal one after another from the attack base taken over at the initial stage, so there is a high possibility that a terminal infected with malware infects not only one terminal but also multiple terminals in the organization. The influence on organization due to system outages is enormous, and the organization wants to recover it as soon as possible, but the system can not be recovered unless measures are taken against all infected terminals. Therefore, after the attack is detected, it is necessary to estimate the damage range by analyzing combinations of logs of multiple terminals. However, the association of each log is time-consuming and requires a high level of technical skill. In this paper, we propose a method to estimate spreading range of malware by analyzing and matching logs of multiple terminals process and communication attempt. We developed a program to realize the proposed method and conducted an evaluation experiment. As a result, we confirmed that it is possible to deal with cases where the infected route is diverging or when the infected terminal is in a latent state without malware being executed by the proposed method. Therefore, it is possible to identify all infected terminals by infection spreading behavior from the infected source terminal in the proposed method, and it is considered that the proposed method can comprehensively trace infection spread. Also, the log after extracting the necessary information for the survey can be compressed to less than 2% of the original log, and the survey processing time was suppressed to a roughly linear increase with respect to the increase of the infected terminal.

Keywords: digital forensic, network forensic, targeted attack, log analysis

1. はじめに

近年，特定の企業や組織を攻撃対象とする標的型攻撃が問題となっている．標的型攻撃とは，金銭や知的財産などの重要情報の窃取を目的として特定の標的に対して行われるサイバー攻撃である [1]．そのなかでも，攻撃対象の組織にマルウェア付きのメールを送り込んでから攻撃を行う標的型メール攻撃が問題となっている．日本では，2011 年の衆議院事務局，三菱重工業などに対する攻撃を境に年々増加傾向にある [2]．2015 年には日本年金機構が被害に遭い 125 万件の個人情報が流出した [2]．

IPA の報告書 [3] によると，標的型メール攻撃の攻撃シナリオは，以下の 6 段階で定義されており，攻撃全体が計画的に進行されていく．

(1) 計画立案段階：標的組織を設定し関連情報の収集
(2) 攻撃準備段階：攻撃に必要となる環境の準備
(3) 初期潜入段階：偽装メールによるマルウェア感染
(4) 基盤構築段階：感染端末を起点にして環境の調査
(5) 内部侵入・調査段階：端末間での侵害の拡大
(6) 目的遂行段階：窃取した機密情報の外部送信

攻撃が本格的に始まるのは，初期潜入段階であり計画立案段階で収集した情報を基に標的ユーザを確実に騙し，標的組織の端末をマルウェアに感染させる．次に，基盤構築段階で感染した端末を起点にして標的組織のネットワーク環境の調査を行う．その後，内部侵入・調査段階で他端末への乗っ取りを繰り返し，侵害を拡大していき機密情報のある端末への乗っ取りを試みる．そして，目的遂行段階で機密情報のある端末から機密情報を収集し外部に送信する．

この攻撃段階のなかでも攻撃の核心部となっているのが内部侵入・調査段階であり，基盤構築段階で確保した攻撃基盤をベースに，次々と端末を乗っ取りながら侵害を拡大していく [3]．そのため，マルウェアに感染する端末が 1 台だけでなく組織内の複数端末に感染が拡大している恐れがあり，攻撃発覚後は被害範囲の想定が重要となっている．被害範囲の想定を誤るとマルウェアが駆除されずにいた端末から攻撃が継続され，さらに被害が拡大することも考えられる．したがって，標的型攻撃の対策には，攻撃の痕跡が残っている可能性の高い各機器のログを組み合わせた分析により攻撃を検知し，被害範囲の想定のために一端末内で起きた事象の解析だけでなく，複数端末の事象を組み合わせた解析が必要である．しかし，ログには，膨大な量の攻撃に関係のない内容も含まれているため，各ログの関連付けは時間を要するうえに高度な技術力が必要である．

そこで本研究では，内部侵入・調査段階に焦点をあて，複数の端末のプロセスとその通信試行のログを解析・突合することでマルウェアの波及範囲を推定する方法を提案する．これにより，応急対策により事象が収束した後，被害範囲の想定や優先して調査すべき端末の特定が可能となり，組織にとって最も重要な運転を再開するまでの時間の大幅な短縮が可能となる．

[1] 東京電機大学
Tokyo Denki University, Adachi, Tokyo 120–8851, Japan
[a)] takahiro.shimakawa@gmail.com

2. 先行研究・関連技術

2.1 先行研究

これまで，著者の 1 人の佐藤を中心に標的型攻撃における内部侵入・調査段階で感染経路を検知する手法を検討してきた [4]．感染経路を追跡することで発見した端末が初期潜入段階または内部侵入・調査段階で感染したのかを判断できる．特に，内部侵入・調査段階で感染した端末には感染元の端末が存在するため，その感染元の端末からのさらなる感染拡大の恐れがあり，攻撃発覚後には迅速な感染経路の調査が必要となる．

これまで，マルウェアに感染した端末を発見する手法は数多く研究されてきたが，自動的に他の端末への感染経路まで調査する手法は少ない．

そこで，佐藤らは後述するプロセスログを用いてマルウェアが起動した子プロセスまで調査するプロセスレベルでの感染経路を検知する手法を検討してきた．この手法では，各端末のプロセスログに内部侵入・調査段階で用いられるツールのプロセスとそのプロセスによる通信試行が記録されているかを調査し，各端末のプロセスログを突合していくことで感染経路を検知する．また，プロセスログの調査は，既存のマルウェア検知手法や IDS によるアラートなどを基点として，マルウェアの感染が検知された端末のプロセスログから侵入源の端末が特定されるまで遡上的に調査をする．これまで，検証実験により，感染範囲の拡大の際に発生する内部通信の特徴を用いることでプロセスレベルでの感染経路が適切に追跡可能であり，侵入源の端末を発見可能であることが確認されている．しかし，佐藤らの手法は感染元の端末のみを特定していくため，感染先の端末の特定は行わない．そのため，一連の感染経路のみの特定はできるが，感染範囲の特定ができないという問題がある．これができないと組織は運転の再開をできず膨大な損失が生じるという重大な問題があった．そこで本研究では，佐藤らの研究により特定された侵入源の端末を起点にして感染拡大を追跡していき感染端末群の全体を特定することでマルウェアの波及範囲を推定する．また，感染先が複数である可能性が高いため，感染先の端末から感染元の端末を特定する遡上的な調査をする場合と比較し感染元の端末から感染先の端末を特定する前進的な調査をする場合では，調査処理時間も増加する可能性が高い．そのため，本研究ではログから調査に必要となる情報をあらかじめ抽出するようにし，調査の際には必要情報を抽出後のログを用いることで調査処理時間の急激な増加を抑えられるようにする．

2.2 関連技術

2.2.1 プロセスログ記録ツール：Onmitsu

Onmitsu とは，不審な通信の原因特定に有用な情報源である揮発性情報を記録するために三村らが開発したプロセスログ記録ツールである [4], [5]．Onmitsu は，Windows の標準 API を利用しカーネルドライバという形でシステム内に導入する．そして，プロセス情報とそのプロセスが発した通信に関する動作ログを常時記録し続ける．そのため，マルウェアによるプロセス情報の隠匿処理も回避できる可能性が高い．

また，標的型攻撃では，攻撃者がマルウェアを通して正規のツール・コマンドを起動しながら攻撃を進行していくため，攻撃挙動を把握するにはプロセス間の関係やプロセスが発した通信の把握が必要となるが，検証実験により記録したログからマルウェアのプロセスとマルウェアに関するプロセスが発した通信とを結び付けられることが確認されている．そのため，Onmitsu ではプロセスレベルでの攻撃挙動の把握が可能であり，標的型攻撃における攻撃挙動の把握にも有用である．

次に Onmitsu に記録されるログについて説明する．Onmitsu で記録する対象はプロセスにおける起動・終了・モジュール読み込み・ネットワーク通信の 4 つの挙動（ログタイプ）である．また，ログは**図 1** に示す情報が CSV 形式で記録される．

本研究では，以下の理由からプロセスと通信試行のログの記録に Onmitsu を用いた．

- 攻撃挙動をプロセスレベルで把握が可能．
- プロセスとその通信試行を 1 つのログに記録するため複数のツールが不要．
- マルウェアによるプロセスの隠匿処理を回避できる可能性が高い．
- 出力されるログファイルが CSV 形式であり汎用的に処理が可能．

2.2.2 オントロジ

オントロジとは，知識をあるドメイン内の概念と概念間の関係として形式的に表現する手法である．オントロジを具体的に表現する一手法として RDF（Resource Description Framework）が存在する [6]．RDF では，主語，述語，目的語という 3 つの要素（RDF トリプル）でリソースに関する情報を表現する．主語は記述対象のリソース，述語は主語の特徴や主語と目的語の関係，目的語は主語との関係のあるリソースや述語の値を表現している．RDF トリプルは，任意の粒度で情報を表現できる．また，主語と目的語をノードに，述語を矢印にした有向グラフで表現でき視覚化できる．さらに，RDF トリプルの集合と推論規則を組み合わせることで，異なる種類のデータを柔軟につなぎ合わせて，その部分和以上の総体を作ることができる．

また，佐藤らの研究 [4] によりプロセスログの情報の表現にオントロジを用いることで検知時間がプロセスログのみの場合と比べ約 1/24 となることが確認されている．

本研究では，以下の理由からプロセスログの情報を表現する手法としてオントロジを採用した．

- 各端末のログの関係性を柔軟に表現が可能
- 共通する述語をつなぎ合わせることで各端末のログの突合が容易
- 有向グラフで表現できるため視覚的な把握が容易
- プロセスログのみの場合と比べ検知時間の短縮が可能

3. 関連研究

本章ではまず，標的型攻撃における内部侵入・調査段階に着目した関連研究との差異を述べる．次に，被害状況の把握を目的とした関連研究との差異について述べる．

標的型攻撃における内部侵入・調査段階に着目した研究として，川口らは複数の端末で行われるさまざまな種類の不審活動を関連付けることで拡散活動を検知する手法を提案している [7]．この手法では，攻撃者の拡散活動にともない不審性が高い端末が連鎖的に現れる現象を，被攻撃端末をノードとするグラフ構造として抽出する．そして，このグラフがある基準を満たすとき，標的型攻撃における拡散活動が発生していると判断してアラートをあげる．

また，類似の研究として，海野らは標的型攻撃におけるシステム内部の諜報活動を検知する手法を提案している [8]．この手法では，標的型攻撃において攻撃基盤を拡大する過程に攻撃者が使わざるを得ない共通の攻撃手法をチョークポイントと定義し，このチョークポイントによるシステムの振舞い解析によって諜報活動の検知を行っている．

これらの研究では，攻撃の検知を主な目的としているため攻撃の検知後の被害状況の把握について検討されていない．そのため，本研究はこれらの研究で攻撃を検知した後の被害状況の把握のための追加調査の研究として位置づけられる．

被害状況の把握を目的とした研究として，遠峰らは標的

```
年,月,日,時,分,秒,ミリ秒,ログタイプ,プロセスID,親プロセスID,ファイルパス,コマンドライン,接続元IPアドレス
,接続元ポート番号,接続先IPアドレス,接続先ポート番号,プロトコル番号
2017,12,01,17,58,39,0616,PROCESS_LAUNCH,1036,1584,¥??¥C:¥Windows¥ShinoBOT.exe,"ShinoBOT.exe" ,,,,,
2017,12,01,17,58,39,0616,PROCESS_MODLOAD,1036,,¥Device¥HarddiskVolume2¥Windows¥ShinoBOT.exe,,,,,,
2017,12,01,17,58,41,0797,NETWORKV4,1036,,,,10.0.3.41,51319,10.0.1.4,8080,6
2017,12,01,17,59,58,0594,PROCESS_QUIT,1036,,,,,,,
```

図 1 Onmitsu により記録されるログのサンプル
Fig. 1 Sample log recorded by Onmitsu.

型攻撃の被害状況の把握やインシデントの分析に利用できるログの可視化手法を提案している [9]．この手法では，複数の端末で発生したさまざまなイベントログを集約し，一覧できるよう同一時間軸上に並べて可視化を行う．これにより，解析者は複数の端末を横断して発生したイベントをとらえることができるため，効率的な被害状況の把握の支援が行える．しかし，遠峰らの手法では，端末がマルウェアに感染しているかどうかなどの判断は解析者が行わなければならないため，マルウェアの波及範囲の推定までに時間を要する．遠峰らの手法に対し，本研究では，感染範囲の拡大の際に使用されたプロセスのログを突合した結果を解析者に出力することにより，迅速な波及範囲の推定を可能とする．

類似の研究として，松本らは IPS/IDS やファイアウォール，プロキシなどの通信装置から出力される通信ログとサーバ，端末から取得される端末ログを組み合わせることで感染範囲を特定する手法を提案している [10]．この手法では，攻撃が検知された後に，攻撃に関する通信ログと端末ログの対応付けを行い，攻撃に関する端末ログを抽出し，攻撃ユーザを特定する．そして，感染端末を操作する攻撃ユーザが他端末へアクセスしたことを端末ログから特定することで感染範囲の特定を行っている．しかし，松本らの手法では，日付・時刻やアクセス元ホストおよびアクセス先ホストによりログを対応付け感染拡大を追跡していくため，攻撃とは無関係な端末を抽出するなど誤検知する可能性が高い．松本らの手法に対し，本研究では，プロセスレベルで調査を行うことによりマルウェアに起因した通信・プロセスを特定でき精度良く感染範囲を特定することができる．

また，Hossain らは，攻撃の検知から攻撃の影響範囲の特定までを行うシステム SLEUTH を提案している [11]．SLEUTH では，1 端末内の監査ログを解析し，タグの伝搬状況からリアルタイムに攻撃を検知する．その後，タグの伝搬状況から作成された依存グラフを分析し，検知の要因となったノードから攻撃の開始位置のノードを特定する．さらに，依存グラフを分析し，攻撃の開始位置のノードから攻撃に関係するノード群（プロセス，ファイルなど）を特定することで攻撃の影響範囲を特定する．そのため，Hossain らの手法では 1 端末内の詳細な影響範囲を把握することができる．しかし，Hossain らの手法では，複数端末のログを組み合わせた解析をしていないため，被害の全体像を把握することができない．Hossain らの手法に対し，本研究では，複数端末のログを組み合わせた解析により組織内の感染端末群を特定する手法であるため，被害の全体像を把握することができる．また，被害の全体像を把握することはどの端末を優先して調査すべきかという判断を行うことができ，攻撃の迅速な対応につながる．そして，優先して調査すべき端末群を決定後に各端末に対し詳細な影響範囲を把握することで原因究明が可能になると考えられる．そのため，本研究で調査すべき端末を絞り込んだ後に Hossain らの手法により端末内の調査を行うことでより総合的な対応が可能になると考える．

4. 提案手法

本章ではまず，マルウェアの波及範囲の推定までの大まかな流れについて説明する．次に，4.1 節で感染範囲の拡大の際に悪用される遠隔操作ツール・コマンドの特徴について述べ，4.2 節でプロセスログから調査に必要な情報の抽出について述べる．そして，4.3 節でマルウェアの波及範囲の推定手法について述べる．また，4.4 節で提案手法を実装したプロトタイプの開発について述べる．

マルウェアの波及範囲の推定までの大まかな流れは次のとおりである．

(1) マルウェアに感染した端末の検知
(2) 検知した端末を起点に佐藤らの研究 [4] により侵入源の端末の推定
(3) 推定された侵入源の端末を起点にした感染拡大の追跡によりマルウェアの波及範囲の推定

ネットワーク内の端末をやみくもに調査するのでは迅速な波及範囲の推定が困難である．そこで本研究では，既存のマルウェア検知手法や IDS などのアラートを利用し，マルウェアに感染した端末を検知した後，佐藤らの研究 [4] により侵入源の端末を推定し，その端末を起点に感染拡大を追跡していく．これにより，迅速なマルウェアの波及範囲の推定を目指す．

4.1 悪用される遠隔操作ツール・コマンドの特徴

JPCERT/CC の報告書 [12] によると，攻撃者が感染範囲を拡大する際に悪用する遠隔操作ツール・コマンドには同じものが使用されることが多いと分かっている．また，JPCERT/CC の報告書 [12] から悪用されることが多い代表的な遠隔操作ツール・コマンドによる内部通信時の特徴を**表 1** に示す．表 1 から，悪用される遠隔操作ツール・コマンドによる内部通信時には特徴的なプロセス，ポート番

表 1 悪用される遠隔操作ツール・コマンドの特徴
Table 1 Features of abused remote control and command.

ツール・コマンド	クライアント端末		リモート端末
	起動プロセス	通信試行時の宛先ポート番号	起動プロセス
PsExec	psexec	135	PSEXESVC
WMIC	WMIC	135	WmiPrvSE
PowerShell	powershell	5985	Wsmprovhost
at	at	445	taskeng

号が用いられていることが分かる．また，本研究では企業などで業務にも使用されることがある表 1 に示した遠隔操作ツール・コマンドを主な対象とした．

ここで，例として PsExec を用いた内部通信を行った場合の挙動について説明する．PsExec を用いた内部通信は以下の流れで行われる．

(1) クライアント端末で PsExec が起動
(2) リモート端末へ向けて宛先ポート番号 135 で psexec による通信が発生
(3) リモート端末でクライアント端末へ向けて対応する通信が発生
(4) リモート端末で PSEXESVC が起動
(5) PSEXESVC が親プロセスとなりリモートコマンドを実行

表 1 のツール・コマンドを用いて内部通信を行った場合であっても，起動するプロセスや通信試行時の宛先ポート番号などが変わるだけでリモートコマンドの実行までの大まかな流れ自体は変わらない．

4.2 プロセスログから必要情報の抽出

実組織を想定すると攻撃により記録されたログだけでなく通常業務などのログも記録されているため，ログが膨大な量になる．また，攻撃開始から攻撃発覚までの期間が長くなればなるほど感染端末の台数も増加する恐れがあり，調査に必要となるログも増加する．そこで本研究では，あらかじめプロセスログから調査に必要な情報を抽出しておくことでインシデント発生時に迅速な調査を行えるようにする．

また，抽出の際には感染拡大時のシーケンスを考慮して表 1 の悪用される遠隔操作ツール・コマンドのプロセスおよび内部通信，その親・子プロセスを抽出する．攻撃者が感染拡大を行う際には，外部の C&C（Command and Control）サーバからマルウェアに対しコマンドを送り表 1 のツール・コマンドを起動する．そして，他の端末へ内部通信し通信先の端末へマルウェアの転送と実行をし感染を拡大する．そのため，クライアント端末の特徴プロセスの親プロセスをたどっていくと必ず外部と通信を行っているプロセスがある．このことは表 1 のツール・コマンドを通常利用する際にはないことであると考えられる．そこで本研究では，クライアント端末の特徴プロセスの親プロセスが外部と通信を行っていた場合，不審と定義し，以下の流れで必要情報を抽出する．

(1) 不審なクライアント端末の特徴を抽出
(2) 発見したクライアント端末の特徴プロセスの親プロセスを抽出
(3) リモート端末の特徴を抽出
(4) 発見したリモート端末の特徴プロセスの子プロセスを抽出

ネットワーク構造の表現に利用
＊語彙：CybOXから引用

ネットワーク語彙	プロセス語彙
network interface	process name
ipv4 address	process id
ipv6 address	parent process id
mac address	service groupe name
default gateway	
host name	
port number	

内部侵入段階の表現に利用
＊語彙：独自定義，関係性：独自定義

語彙	定義
host name	CybOXと同じ
status	マルウェア感染状態を示す

関係性	使用目的
Penetration	ホスト名間をつなぐ
infect process	statusとプロセス名をつなぐ

図 2 佐藤らにより定義されたオントロジの一部
Fig. 2 A part of the ontology defined by Sato et al.

4.3 波及範囲推定手法

本研究で提案するマルウェアの波及範囲推定手法では，表 1 の悪用される遠隔操作ツール・コマンドの内部通信とその通信を行っているプロセスの関係を明確にすることで感染範囲の拡大挙動を追跡していく．本提案手法は，佐藤らの研究 [4] により推定された侵入源の端末を起点にネットワーク内の端末に対し，表 1 の特徴がプロセスログに存在するか調査する手法である．具体的には，以下の手順で調査を行っていく．

(1) 侵入源の端末で検知されたマルウェアの子プロセスのなかからクライアント端末の特徴を持つものを抽出
(2) 抽出されたプロセスに関する通信から通信先の端末を特定
(3) 通信先の端末のプロセスのなかからリモート端末の特徴を持つものを抽出
(4) (3) で抽出したプロセスの子プロセスのなかからマルウェアの起動を発見
(5) (4) で発見したマルウェアの子プロセスおよびマルウェアの子プロセス以外のプロセスのなかからクライアント端末の特徴を持つものを抽出
(6) 手順 (2)〜(5) を繰り返す．

4.4 プロトタイプの開発

提案手法を実現するプロトタイプの開発を行った．機能要件は次の 2 つである．

- 4.3 節で述べた手順の自動的な処理
- 各端末の調査結果の統合

各端末の調査結果を統合するために情報処理機器間の関係を RDF で表現した．情報処理機器間の関係を RDF で表現するために定義した語彙と関係性は，**図 2** に示す佐藤らの研究 [4] で使用されたものと同一のものを使用した．また，調査結果の可視化については，グラフ描画ツールである Graphviz [13] を用いて RDF により記述されたマルウェアの波及範囲を可視化した．

5. 実験

5.1 実験概要

実験では，内部侵入・調査段階における攻撃者の感染範囲の拡大の際の行動を模擬する．その後，Onmitsu により記録していたプロセスログに対し，提案手法を適用し，

その有効性を評価する．また，実験では以下のことを確認する．

実験 1　感染経路が分岐していた場合の分岐先の感染端末群を特定可能であるか．

実験 2　感染拡大時のプロセスのつながりが途切れていた場合であっても感染端末をすべて特定可能であるか．

実験 3　通信先の端末のうちマルウェアが実行されずに潜伏状態にある端末も特定可能であるか．

実験 4　ネットワーク内の端末のうち，感染端末のみが特定されるか．

実験 5　実時間に対応しうる時間で感染端末をすべて特定可能であるか．

5.2　実験手順

感染範囲の拡大を模擬するために，RAT/ボットマルウェアシミュレータである ShinoBOT [14] を標的型攻撃に使われるマルウェアに見立てて感染させ，感染端末で表 1 のツール・コマンドを用いて次の手順で感染範囲を拡大した．

(1) 侵入源の端末で ShinoBOT を実行
(2) 他端末へ向けて内部通信を実行
(3) 通信先端末でリモートコマンドを実行

実験 1 では，通信先端末へ ShinoBOT の転送と実行を繰り返し感染範囲の拡大を行っていく．また，感染経路が分岐していた場合の分岐先の感染端末群を特定可能であるかを確認するために，**図 3** のように感染範囲の拡大を模擬した．そして，感染範囲の拡大を模擬した後，侵入源である端末 1 が佐藤らの研究 [4] により特定されたと仮定し，侵入源である端末 1 を起点に提案手法を適用する．

実験 2 では，攻撃が進行中に端末の利用者が攻撃に気付かず端末を再起動することなどを考慮して**図 4** のように端末間の感染拡大時のプロセスを一連のもの（PSEXESVC.exe→マルウェア→PsExec.exe）とするのではなく，途中でプロセスのつながりが途切れた場合（PSEXESVC.exe→マルウェア，端末再起動後に起動したマルウェア→PsExec.exe）であっても感染端末をすべて特定可能であるかを確認する．今回の実験では，端末 1 から端末 2 への感染拡大後，端末 2 で端末自身を再起動する．その後，端末 2 で再度 ShinoBOT を起動し端末 3 へ感染を拡大するようにした．そして，侵入源である端末 1 が佐藤らの研究 [4] により特定されたと仮定し，侵入源である端末 1 を起点に提案手法を適用する．

実験 3 では，通信先の端末で ShinoBOT を実行せず転送のみを行いマルウェアが潜伏状態にある端末を用意する．これにより，通信先の端末のうちマルウェアが実行されずに潜伏状態にある端末も特定可能であるかを確認する．今回の実験では，**図 5** のように感染範囲の拡大を模擬し，端末 2 では ShinoBOT を実行し，端末 3 では転送のみを行うようにした．また，IPA の報告書 [3] によると，攻撃者はマルウェアなどの攻撃用ツールの転送には，Windows ファイル管理共有などを利用することが分かっている．そのため，端末 1 上に共有フォルダを用意し，その中に ShinoBOT を配置した．ファイルの移動には，JPCERT/CC の報告書 [12] からバックグラウンドでファイルの送受信を可能と

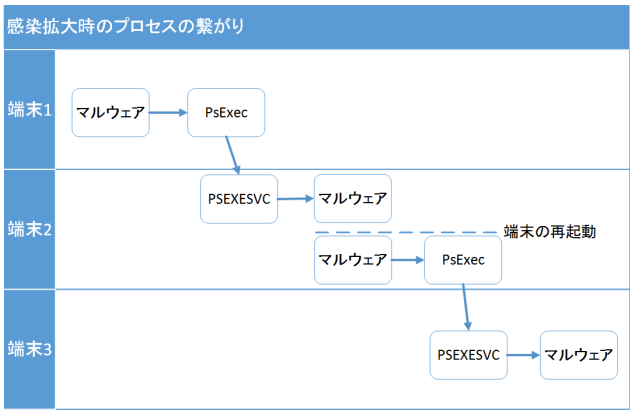


図 4　実験 2 における感染範囲の拡大
Fig. 4　Expansion of infected range in experiment 2.

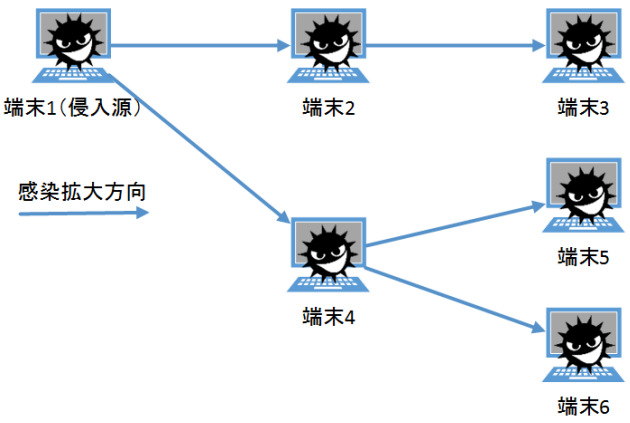


図 3　実験 1 における感染範囲の拡大
Fig. 3　Expansion of infected range in experiment 1.

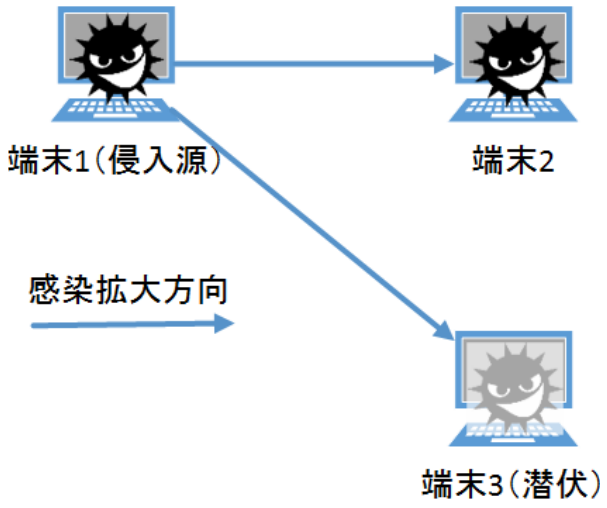


図 5　実験 3 における感染範囲の拡大
Fig. 5　Expansion of infected range in experiment 3.

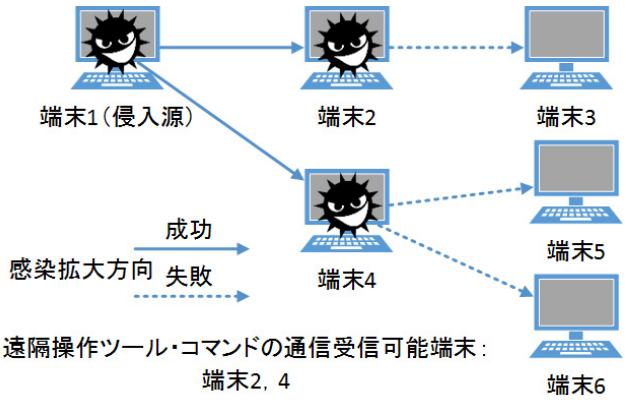


図 6 実験 4 における感染範囲の拡大
Fig. 6 Expansion of infected range in experiment 4.

するサービスである BITS が利用されることが分かっている．そのため，端末 3 へのリモートコマンドでは，BITS を利用するために bitsadmin を実行し端末 1 上の共有フォルダから ShinoBOT をダウンロードするようにした．そして，感染範囲の拡大を模擬した後，侵入源である端末 1 が佐藤らの研究 [4] により特定されたと仮定し，侵入源である端末 1 を起点に提案手法を適用する．

実験 4 では，表 1 のツール・コマンドによる通信が受信可能な端末を限定することにより，感染拡大が可能な端末と不可能な端末を用意する．そして，すべての端末が感染されるように感染範囲の拡大を試みる．これにより，ネットワーク内の端末のうち，感染端末のみが特定されるかを確認する．また，今回の実験では，端末 2，端末 4 の 2 台のみ表 1 のツール・コマンドによる通信が受信可能とし，**図 6** のように感染範囲の拡大を模擬した．そして，感染範囲の拡大を模擬した後，侵入源である端末 1 が佐藤らの研究 [4] により特定されたと仮定し，侵入源である端末 1 を起点に提案手法を適用する．

実験 5 では，日本年金機構の事例 [2] を参考に感染範囲を拡大し侵入源の端末から感染端末をすべて特定するまでの調査処理時間を計測する．実験環境は，VMWare ESXi 6.5 を用いて**図 7** のように実際の組織のネットワークを模した環境を構築した．また，日本年金機構の事例 [2] では，攻撃者による感染拡大行為により感染した端末数は 27 台であったため，クライアント PC を 27 台用意した．各クライアント PC では，Web 閲覧などの通常業務でも行われる操作をログの量が約 30MB となるまで行った．その後，**図 8** のように感染範囲の拡大を模擬した．そして，感染範囲の拡大を模擬した後，侵入源である端末 1 が佐藤らの研究 [4] により特定されたと仮定し，侵入源である端末 1 を起点に提案手法を適用する．

5.3 実験結果

実験 1 の結果，提案手法により感染端末をすべて特定することができた．**図 9** は，実験 1 の結果の一部を抜き出し

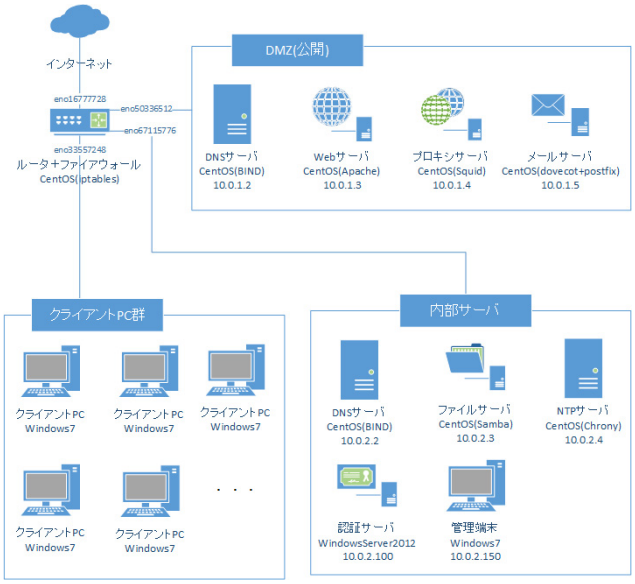


図 7 実験環境
Fig. 7 Experiment environment.

たものである．図 9 中に示している RDF トリプルは以下のとおりである．

- RDF トリプル（ホスト名，PID，プロセス ID）
- RDF トリプル（ホスト名，ipv4Address，IP アドレス）
- RDF トリプル（ホスト名，status，マルウェア感染状態）
- RDF トリプル（プロセス ID，name，プロセス名）
- RDF トリプル（プロセス ID，ParentPID，親プロセス ID）
- RDF トリプル（プロセス ID，launch_time，起動時間）
- RDF トリプル（プロセス ID，com_by，送信元ポート番号）
- RDF トリプル（プロセス ID，com_time，通信時間）
- RDF トリプル（IP アドレス，port，送信元ポート番号）
- RDF トリプル（送信元ポート番号，TCP，宛先ポート番号）
- RDF トリプル（送信元ポート番号，com_time，通信時間）
- RDF トリプル（マルウェアの感染状態，infected_process，マルウェアのプロセス ID）

図 9 中の RDF トリプル群の主語と述語を照合していくことにより以下のことが分かる．

- K-W7X64I1（端末 1 のホスト名）から PID という述語をたどることでプロセス ID が 1992 のプロセスが起動していたということが分かる（①→②→③）．
- プロセス ID：1992 から name という述語をたどることでプロセス名が PsExec.exe であるということが分かる（③→④→⑤）．
- プロセス ID：1992 から ParentPID という述語をたどることでプロセス ID：624 が親プロセスのプロセス

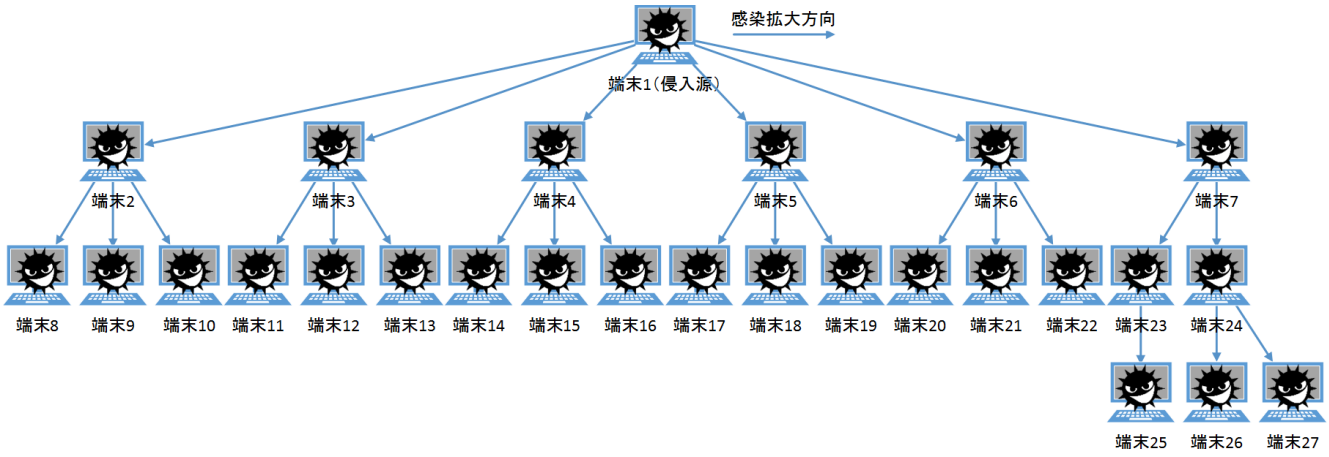
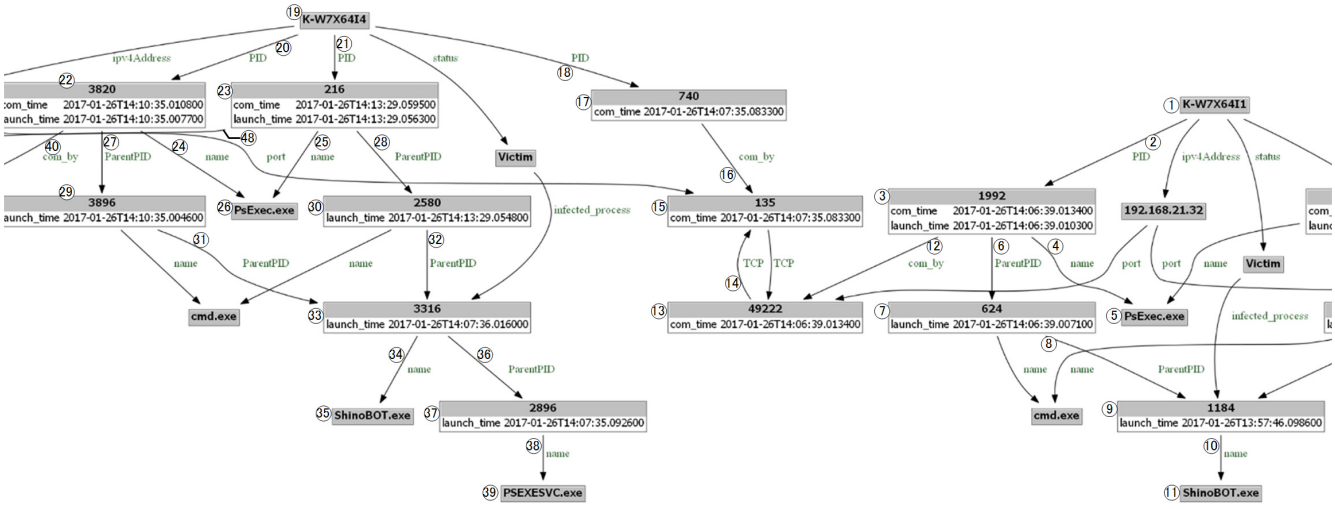
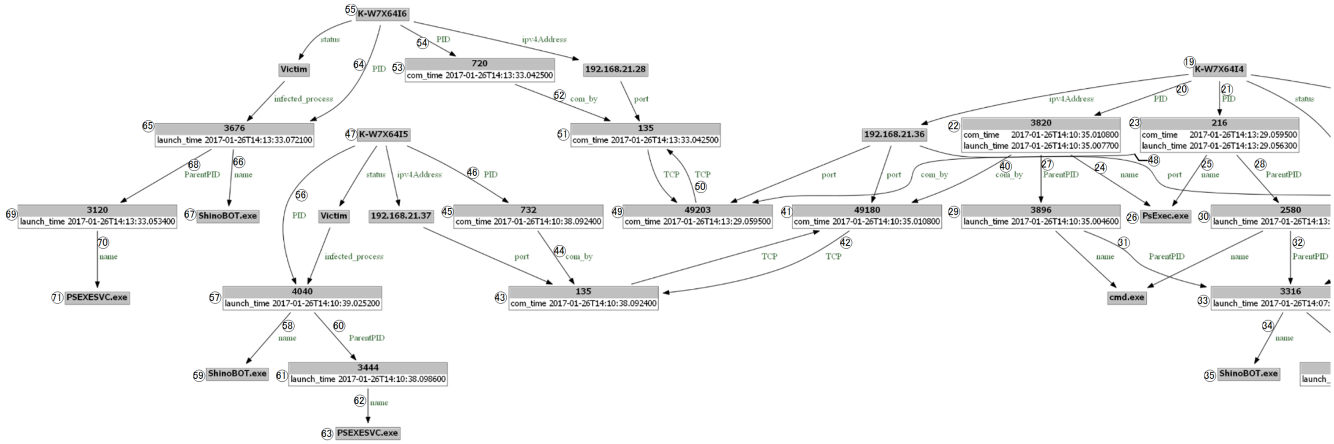


図 8 実験 5 における感染範囲の拡大

Fig. 8 Expansion of infected range in experiment 5.



(a) 結果 1



(b) 結果 2

図 9 実験 1 の結果

Fig. 9 Illustrated result of experiment 1 using RDF triple.

ID であるということが分かる（③→⑥→⑦）.

- プロセス ID：624 から ParentPID という述語をたどることでプロセス ID：1184 が親プロセスのプロセス ID であるということが分かる（⑦→⑧→⑨）.
- プロセス ID：1184 から name という述語をたどることでプロセス名が ShinoBOT.exe であることが分かる（⑨→⑩→⑪）.

これらのことから，端末 1 で ShinoBOT.exe（プロセス

ID：1184）により PsExec.exe（プロセス ID：1992）が起動されたということが分かる．

- プロセス ID：1992 から com_by という述語をたどることでこのプロセスがポート番号：49222 を送信元ポート番号として通信を行ったということが分かる（③→⑫→⑬）．
- ポート番号：49222 から TCP という述語をたどることでポート番号：135 が宛先ポート番号であるということが分かる（⑬→⑭→⑮）．
- ポート番号：135 への com_by という述語を逆にたどることでプロセス ID：740 のプロセスが通信を行ったということが分かる（⑮→⑯→⑰）．
- プロセス ID：740 への PID という述語を逆にたどることで K-W7X64I4（端末 4 のホスト名）で起動していたということが分かる（⑰→⑱→⑲）．

これらのことから，端末 1 で起動された PsExec.exe（プロセス ID：1992）により送信元ポート番号：49222，宛先ポート番号：135 として端末 4 へ通信が行われたということが分かる．

- K-W7X64I4 から PID という述語をたどることでプロセス ID が 3820 と 216 のプロセスが起動していたということが分かる（⑲→⑳→㉒，⑲→㉑→㉓）．
- プロセス ID：3820，216 から name という述語をたどることでプロセス名が PsExec.exe であるということが分かる（㉒→㉔→㉖，㉓→㉕→㉖）．
- プロセス ID：3820，216 から ParentPID という述語をたどることでプロセス ID：3896，2580 がそれぞれの親プロセスのプロセス ID であるということが分かる（㉒→㉗→㉙，㉓→㉘→㉚）．
- プロセス ID：3896，2580 から ParentPID という述語をたどることでプロセス ID：3316 が親プロセスのプロセス ID であるということが分かる（㉙→㉛→㉝，㉚→㉜→㉝）．
- プロセス ID：3316 から name という述語をたどることでプロセス名が ShinoBOT.exe であるということが分かる（㉝→㉞→㉟）．
- プロセス ID：3316 から ParentPID という述語をたどることでプロセス ID：2896 が親プロセスのプロセス ID であるということが分かる（㉝→㊱→㊲）．
- プロセス ID：2896 から name という述語をたどることでプロセス名が PSEXESVC.exe であるということが分かる（㊲→㊳→㊴）．
- プロセス ID：2896 の lunch_time とポート番号：135 の com_time を比較することでプロセス ID：2896 は通信後に起動したということが分かる（㊲，⑮）．

これらのことから，端末 1 からの通信後に端末 4 で PSEXESVC.exe（プロセス ID：2896）が起動され，ShinoBOT.exe（プロセス ID：3316）を起動しているということが分かる．そして，この ShinoBOT.exe により PsExec.exe（プロセス ID：3820，216）が起動されたということが分かる．

- プロセス ID：3820 から com_by という述語をたどることでこのプロセスがポート番号：49180 を送信元ポート番号として通信を行ったということが分かる（㉒→㊵→㊶）．
- ポート番号：49180 から TCP という述語をたどることでポート番号：135 が宛先ポート番号であるということが分かる（㊶→㊷→㊸）．
- ポート番号：135 への com_by という述語を逆にたどることでプロセス ID：732 のプロセスが通信を行ったということが分かる（㊸→㊹→㊺）．
- プロセス ID：732 への PID という述語を逆にたどることで K-W7X64I5（端末 5 のホスト名）で起動していたということが分かる（㊺→㊻→㊼）．
- プロセス ID：216 から com_by という述語をたどることでこのプロセスがポート番号：49203 を送信元ポート番号として通信を行ったということが分かる（㉓→㊽→㊾）．
- ポート番号：49203 から TCP という述語をたどることでポート番号：135 が宛先ポート番号であるということが分かる（㊾→㊿→(51)）．
- ポート番号：135 への com_by という述語を逆にたどることでプロセス ID：720 のプロセスが通信を行ったということが分かる（(51)→(52)→(53)）．
- プロセス ID：720 への PID という述語を逆にたどることで K-W7X64I6（端末 6 のホスト名）で起動していたということが分かる（(53)→(54)→(55)）．

これらのことから，端末 4 で起動された PsExec.exe（プロセス ID：3820）により送信元ポート番号：49180，宛先ポート番号：135 で端末 5 へ，PsExec.exe（プロセス ID：216）により送信元ポート番号：49203，宛先ポート番号：135 で端末 6 へ通信が行われたということが分かる．

- K-W7X64I5 から PID という述語をたどることでプロセス ID が 4040 のプロセスが起動していたということが分かる（㊼→(56)→(57)）．
- プロセス ID：4040 から name という述語をたどることでプロセス名が ShinoBOT.exe であるということが分かる（(57)→(58)→(59)）．
- プロセス ID：4040 から ParentPID という述語をたどることでプロセス ID：3444 が親プロセスのプロセス ID であるということが分かる（(57)→(60)→(61)）．
- プロセス ID：3444 から name という述語をたどることでプロセス名が PSEXESVC.exe であるということが分かる（(61)→(62)→(63)）．
- プロセス ID：3444 の lunch_time とポート番号：135 の com_time を比較することでプロセス ID：3444 は

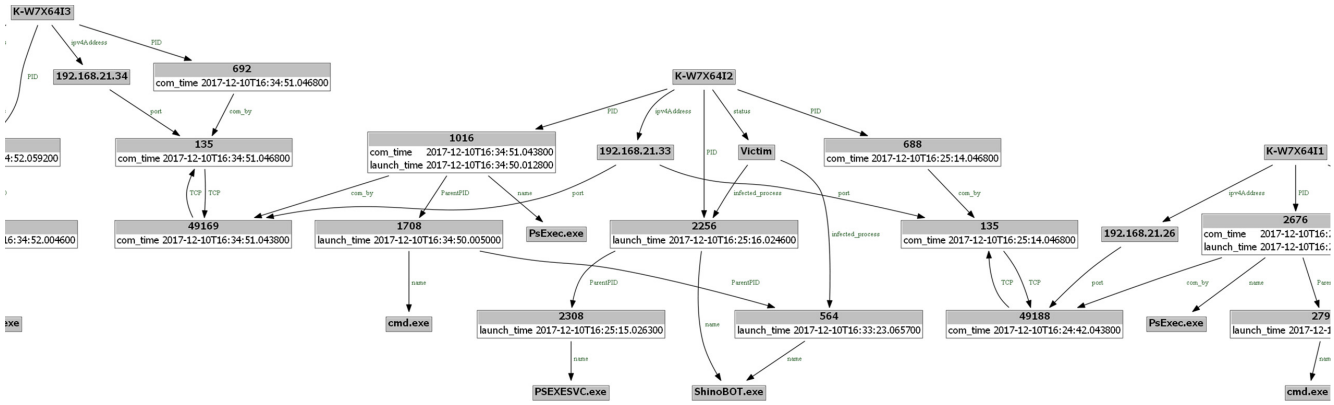


図 10 実験 2 の結果
Fig. 10 Illustrated result of experiment 2 using RDF triple.

通信の後に起動したということが分かる（㊶，㊸）．これらのことから，端末 4 からの通信後に端末 5 で PSEXESVC.exe（プロセス ID：3444）が起動され，ShinoBOT.exe（プロセス ID：4040）を起動しているということが分かる．

- K-W7X64I6 から PID という述語をたどることでプロセス ID が 3676 のプロセスが起動していたということが分かる（㊺→㊹→㊺）．
- プロセス ID：3676 から name という述語をたどることでプロセス名が ShinoBOT.exe であるということが分かる（㊺→㊻→㊼）．
- プロセス ID：3676 から ParentPID という述語をたどることでプロセス ID：3120 が親プロセスのプロセス ID であるということが分かる（㊺→㊽→㊾）．
- プロセス ID：3120 から name という述語をたどることでプロセス名が PSEXESVC.exe であるということが分かる（㊾→㊿→(71)）．
- プロセス ID：3120 の lunch_time とポート番号：135 の com_time を比較することでプロセス ID：3120 は通信の後に起動したということが分かる（㊾，㊶）．

これらのことから，端末 4 からの通信後に端末 6 で PSEXESVC.exe（プロセス ID：3120）が起動され，ShinoBOT.exe（プロセス ID：3676）を起動しているということが分かる．

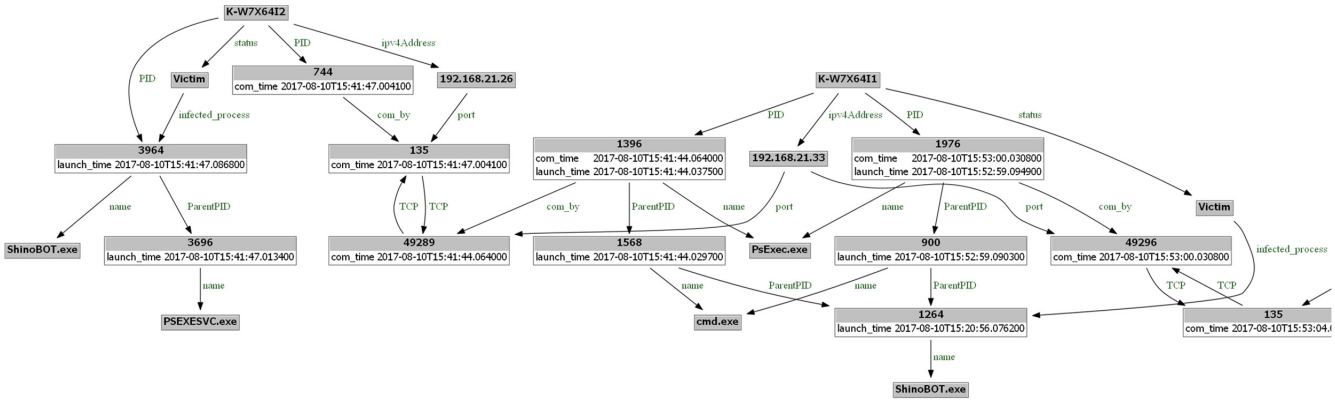
以上より，侵入源である端末 1 内で起動されたマルウェア（ShinoBOT.exe）を起因として感染範囲が拡大したということが分かる．

実験 2 の結果，提案手法により感染拡大時のプロセスのつながりが途中で途切れていた場合であってもマルウェアと遠隔操作ツール・コマンドの特徴プロセスの接続関係を基に，感染端末をすべて特定することができた．**図 10** に実験 2 の結果の一部を示す．図 10 中に示している RDF トリプルは図 9 と同様である．図 10 中の RDF トリプル群の主語と述語を照合していくことにより以下のことが分かる．

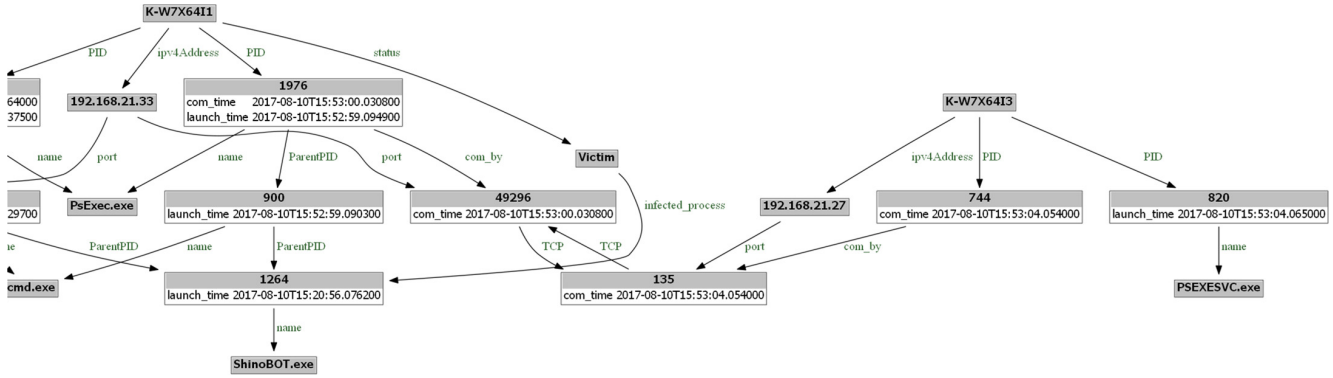
- 端末 1（K-W7X64I1）で起動された PsExec.exe（プロセス ID：2676）による通信の後，端末 2（K-W7X64I2）で PSEXESVC.exe（プロセス ID：2308）が起動されている．
- 端末 2（K-W7X64I2）で PSEXESVC.exe により ShinoBOT.exe（プロセス ID：2256）が起動されている．
- 端末 2（K-W7X64I2）で PsExec.exe（プロセス ID：1016）が起動されている．
- PsExec.exe（プロセス ID：1016）の親プロセスをたどっていくと再起動された ShinoBOT.exe（プロセス ID：564）により起動されている．
- PsExec.exe（プロセス ID：1016）により端末 3（K-W7X64I3）へ通信が行われている．

実験 3 の結果，提案手法によりマルウェアが潜伏状態にある端末も含めて感染端末をすべて特定することができた．**図 11** に実験 3 の結果を示す．図 11 中に示している RDF トリプルは図 9 と同様である．図 11 中の RDF トリプル群の主語と述語を照合していくことにより以下のことが分かる．

- 端末 1（K-W7X64I1）で ShinoBOT.exe（プロセス ID：1264）により PsExec.exe（プロセス ID：1396）が起動されている．
- PsExec.exe（PID：1396）により端末 2（K-W7X64I2）へ通信が行われている．
- 端末 2（K-W7X64I2）で端末 1（K-W7X64I1）からの通信後，PSEXESVC.exe（プロセス ID：3696）が起動されている．
- 端末 2（K-W7X64I2）で PSEXESVC.exe（プロセス ID：3696）により ShinoBOT.exe（プロセス ID：3964）が起動されている．
- 端末 1（K-W7X64I1）で ShinoBOT.exe（プロセス ID：1264）により PsExec.exe（プロセス ID：1976）が起動されている．
- PsExec.exe（プロセス ID：1976）により端末 3（K-



(a) 結果 1



(b) 結果 2

図 **11** 実験 3 の結果

Fig. 11 Illustrated result of experiment 3 using RDF triple.

```
①
2017,08,10,15,53,04,0540,NETWORKV4,744,,,,192.168.21.27,135,192.168.21.33,49296,6
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
②
2017,08,10,15,53,04,0650,PROCESS_LAUNCH,820,544,¥??¥C:¥Windows¥PSEXESVC.exe,C:Windows¥PSEXESVC.exe,,,,,
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
③
2017,08,10,15,53,04,0837,PROCESS_LAUNCH,3188,820,¥??¥C:¥Windows¥bitsadmin.exe,"bitsadmin.exe" /TRANSFER dl
¥¥192.168.21.33¥SHARE¥ShinoBOT.exe C:¥Users¥Yui¥AppData¥Local¥Temp¥ShinoBOT.exe,,,,,
```

図 **12** 端末 3 のプロセスログ

Fig. 12 Process log of terminal 3.

W7X64I3）へ通信が行われている．

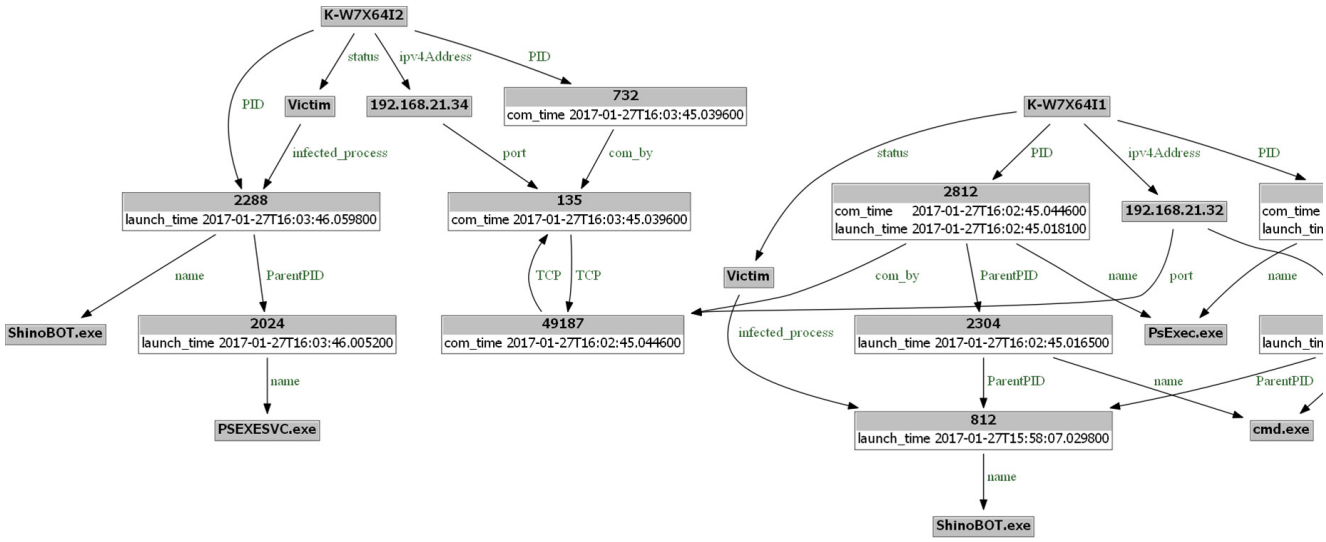
- 端末 3（K-W7X64I3）で端末 1（K-W7X64I1）からの通信後，PSEXESVC.exe（プロセス ID：820）が起動されている．

端末 1（K-W7X64I1）から端末 2（K-W7X64I2），端末 3（K-W7X64I3）へマルウェア（ShinoBOT.exe）を起因とした通信が行われた後，端末 2（K-W7X64I2）では ShinoBOT.exe の起動を確認できるが，端末 3（K-W7X64I3）では確認できない．しかし，**図 12** の端末 3 のプロセスログを確認すると端末 1 からの内部通信（①）後に PSEXESVC.exe が起動（②）し，親プロセスとなり bitsadmin.exe が起動（③）され，端末 1 から ShinoBOT.exe をダウンロードしていることが確認できる．

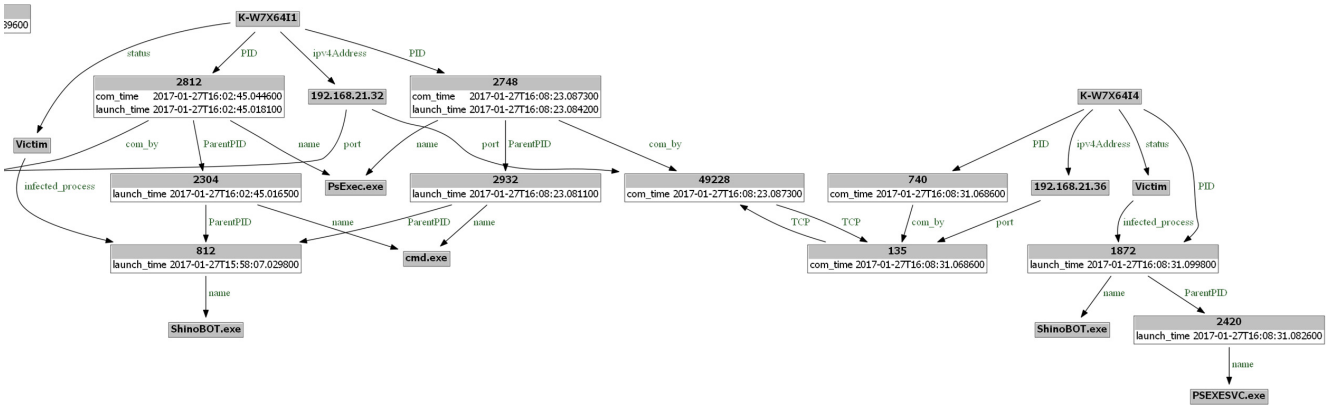
以上の結果から，プロセスレベルで追跡を行うことにより，マルウェアに起因した内部通信を特定することができ感染範囲の拡大を特定可能であることが分かった．

実験 4 の結果，提案手法によりネットワーク内の端末のうち，マルウェアに感染した端末のみを特定することができた．**図 13** に実験 4 の結果を示す．図 13 中に示している RDF トリプルは図 9 と同様である．図 13 中の RDF トリプル群の主語と述語を照合していくことにより以下のことが分かる．

- 端末 1（K-W7X64I1）で ShinoBOT.exe（プロセス ID：812）により PsExec.exe（プロセス ID：2812）が起動されている．
- PsExec.exe（プロセス ID：2812）により端末 2（K-W7X64I2）へ通信が行われている．
- 端末 2（K-W7X64I2）で端末 1（K-W7X64I1）からの通信後，PSEXESVC.exe（プロセス ID：2024）が起動されている．
- 端末 2（K-W7X64I2）で PSEXESVC.exe により ShinoBOT.exe（プロセス ID：2288）が起動されている．
- 端末 1（K-W7X64I1）で ShinoBOT.exe（プロセス ID：812）により PsExec.exe（プロセス ID：2748）が



(a) 結果 1



(b) 結果 2

図 13 実験 4 の結果

Fig. 13 Illustrated result of experiment 4 using RDF triple.

起動されている．

- PsExec.exe（プロセス ID：2748）により端末 4（K-W7X64I4）へ通信が行われている．
- 端末 4（K-W7X64I4）で端末 1（K-W7X64I1）からの通信後，PSEXESVC.exe（プロセス ID：2420）が起動されている．
- 端末 4（K-W7X64I4）で PSEXESVC.exe（プロセス ID：2420）により ShinoBOT.exe（プロセス ID：1872）が起動されている．

また，端末 2（K-W7X64I2），端末 4（K-W7X64I4）から他の端末への感染拡大は確認できない．これは，実験 4 では遠隔操作ツール・コマンドによる通信を受信可能と設定していた端末が端末 2（K-W7X64I2），端末 4（K-W7X64I4）のみであり，他の端末への感染拡大が行えなかったためである．その結果，図 13 中には，マルウェアに感染した端末 1（K-W7X64I1），端末 2（K-W7X64I2），端末 4（K-W7X64I4）のみが出力されている．

実験 5 の結果，すべての感染端末を特定するまでの時間は約 45 秒であり 1 端末あたりの平均は約 1.6 秒であった．調査処理時間のグラフを**図 14** に示す．縦軸は調査処理時間，横軸は調査に用いた各端末の台数を示している．図 14 から調査処理時間は感染端末台数の増加に対しおおよそ線形的な増加に抑えられた．また，プロセスログから調査に必要な情報を抽出した後の容量と元のログに対する圧縮率（抽出後/抽出前）の表を**表 2** に示す．圧縮率は最大で 1.95%，最小で 0.08%であり，平均で約 0.42%まで圧縮された．

6. 考察

6.1 マルウェアの波及範囲の推定について

今回の実験で提案手法により，感染経路が分岐している場合や，感染拡大時のプロセスが途中で途切れていた場合であっても感染拡大を追跡でき感染端末をすべて特定することができた．さらに，内部通信の通信先の端末でマルウェアが実行されずに潜伏状態にある端末も提案手法により特定することができた．これらのことから，提案手法に

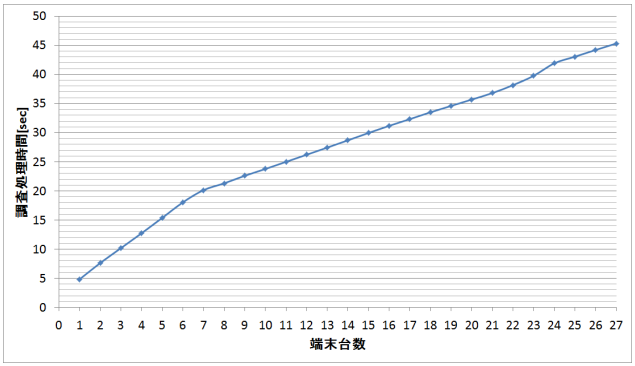


図 14 調査処理時間グラフ
Fig. 14 Survey processing time graph.

表 2 圧縮率
Table 2 Compression ratio.

端末	ログ容量（KB）	抽出後容量（KB）	圧縮率（抽出後/前）
1	36385	262	0.72%
2	37173	377	1.01%
3	34981	449	1.28%
4	33477	654	1.95%
5	34659	333	0.96%
6	36542	489	1.34%
7	34944	509	1.46%
8	35456	33	0.09%
9	34675	34	0.10%
10	37097	32	0.09%
11	34010	31	0.09%
12	35228	30	0.09%
13	35085	31	0.09%
14	35511	32	0.09%
15	34993	34	0.10%
16	36762	34	0.09%
17	34152	31	0.09%
18	34021	30	0.09%
19	34512	32	0.09%
20	37389	33	0.09%
21	35786	31	0.09%
22	34841	31	0.09%
23	34232	136	0.40%
24	35912	207	0.58%
25	34545	29	0.08%
26	33629	31	0.09%
27	35679	33	0.09%
平均	35247.26	147.70	0.42%
合計	951676	3988	0.42%

より侵入源の端末から感染拡大行為によって感染した端末群をすべて特定することができ，網羅的に感染拡大を追跡できると考える．また，提案手法によりネットワーク内の端末のうちマルウェアに感染した端末のみを特定することができた．そのため，提案手法により高い精度でマルウェアの波及範囲を推定できると考える．また，提案手法により特定された端末を詳細に調査することで駆除されずに潜伏していたマルウェアの早期発見につながり，再侵入の防止にも貢献することができると考える．

6.2 外部通信を起点とした感染端末特定手法との比較

感染端末を特定するためには，マルウェアの通信先の URL に対する通信が行われていないかを調査するといった外部通信を調査する手法がある．しかし，この手法では，外部への該当する通信が発生していなければ感染範囲が拡大していてもただちに認知することはできない．これに対して提案手法では，感染範囲を拡大する際に行われる内部通信を調査するため，外部へ該当する通信が発生していなくても感染端末を特定することができる．そのため，提案手法は外部通信を調査する手法を補完することができる．

6.3 調査処理時間

今回の実験 5 で参考にした日本年金機構への標的型攻撃の事例 [2], [15] では，攻撃者は攻撃が検知されるまで 27 台の端末に感染拡大を行っている．この事例における対応では，不審な通信の検知後に特定した感染端末は 2 台のみにとどまっており，翌日に他の 21 台の感染端末を特定している．その後の調査で残りの感染端末を特定しており，感染端末をすべて特定するまでにかなりの時間を要している．本提案手法による調査処理時間は，27 台の感染端末を特定するまでに約 42 秒であり，1 端末あたり約 1.6 秒であった．また，調査処理時間は感染端末台数の増加に対しおおよそ線形的な増加に抑えられた．このことから，1,000 端末に感染範囲が拡大していた場合であっても，1,600 秒程度ですべての感染端末を特定できると予想される．そのため，本提案手法による感染範囲の特定に要する時間は実時間に対応しうるものであると考える．

6.4 ログの保存領域への影響

標的型攻撃の対策としてログの保存領域に明確な定めはないが，JPCERT/CC では 1 つの参考値として 1 年分のログを保存することを推奨している [16]．1 年分のログを保存するとした場合，ある程度の保存領域が確保されると考えられる．しかし，元のログとともに追加情報を保存することを考慮すると追加情報の容量が大きいと必要期間のログを保存できない可能性がある．本提案手法による必要情報の抽出処理の結果，抽出後のログが元のログに対し 2%未満に圧縮されることを確認した．そのため，元のログとと

もに必要情報を抽出後のログを保存する際，必要以上にログの保存領域を圧迫せずに済むと考える．

6.5 攻撃者が利用するツール・コマンドについて

今回の実験で用いた表 1 のツール・コマンドと同様の機能を持つ既知のツールであれば，あらかじめ Onmitsu によりプロセスログを取得し，内部通信時の特徴を抽出後，開発したプログラムに抽出した特徴を追加することで対応可能となる．仮に，表 1 のツール・コマンドを難読化ツールやパッカーなどを利用してバイナリを書き換えられた場合であっても，本提案手法では内部通信時のプロセスの挙動を基に感染拡大を追跡しているため対応可能である．また，今回の実験 3 では，攻撃用ツールを転送する際に Windows ファイル管理共有を使用したが，攻撃者が表 1 のツール・コマンドを使用して転送用ツールを実行した場合，攻撃者が用意した外部サーバなどからのダウンロードであっても，感染先の端末を特定できる．攻撃者が独自に用意した転送手段を使用した場合であっても，転送後に攻撃用ツールをリモート実行する際に表 1 のツール・コマンドを使用した場合，感染先の端末を特定できる．すなわち，本提案手法は，攻撃者が表 1 のツール・コマンドを使用して攻撃用ツールの転送またはリモート実行を行った場合，感染拡大を追跡できる．

6.6 今後の課題

今回の実験では，攻撃の再現を提案手法の適用直前であるとしているが，標的型攻撃の発見までに平均 156 日という調査結果 [17] も出ている．その場合にはログが残っていなかったりログの入手に時間がかかったりする可能性がある．したがって，本提案手法は早期の攻撃検知機能と組み合わせて用いるべきものである．そのうえで，提案手法をより実用的な手法とする必要がある．攻撃者が表 1 のツール・コマンドと同様の機能を持つ独自ツールを作成し感染拡大時に使用した場合，あらかじめプロセスログを取得し，内部通信時の特徴を抽出することができないため，開発したプログラムに抽出した特徴を追加できず，感染拡大を追跡できない．しかし，新しい攻撃に事前の対応が困難であることは，他の方式でも同様であり，新しいツール・コマンドが使われるようになったらそれに対応する仕組みを提案方式に組み込むことで対応したいと考えている．また，解析者が調査結果を基に迅速な状況判断を行えるように調査結果の可視化のパターン増加についても今後の課題として検討する．本提案手法は，侵入源の端末を起点に調査を行っていくものであるため，侵入源の端末の特定を誤った場合や侵入源の端末が複数あった場合など正確なマルウェアの波及範囲の推定が行えない．そのため，今後の課題として確実に侵入源の端末を特定する方法の検討が必要である．また，本提案手法は，すべての端末で Onmitsu によりプロセスログが記録されている場合を前提とするものであるが，ログが欠損している場合の感染拡大の追跡手法も今後，重要となるので将来の課題として検討していきたい．

7. おわりに

本研究では，標的型攻撃における内部侵入・調査段階に焦点をあて，複数の端末のプロセスログを解析・突合することでマルウェアの波及範囲を推定する手法を提案した．

標的型攻撃では，内部侵入・調査段階で組織内の複数端末にマルウェアを感染させることが知られており，被害範囲の想定が重要となっている．佐藤らによりマルウェアの感染が検知された端末から侵入源の端末までの感染経路を特定する手法が提案されているが，被害範囲の想定には感染経路だけでなくマルウェアの波及範囲を把握する必要がある．そこで本研究では，侵入源の端末から感染拡大を追跡していき感染端末群の全体を特定することでマルウェアの波及範囲を推定する手法を提案し，提案手法を実現するプログラムを実際に開発して実験を行った．その結果，標的型攻撃における内部侵入・調査段階で感染範囲が拡大された際に，提案手法によりマルウェアの波及範囲を推定可能となる見通しが得られた．

今後は，実際に起こりうるケースを想定した実験により提案手法を評価するとともに提案手法をより実用的な手法とするために以下のことを検討していく．

- 新しい攻撃が出現した場合の対応
- 調査結果の可視化パターンの増加
- 確実な侵入源の端末の特定
- ログが欠損している場合の対応

謝辞 本研究に際して，さまざまなご指導いただきました LIFT プロジェクトの関係者に深謝いたします．

参考文献

[1] 総務省：「サイバー攻撃（標的型攻撃）対策防御モデルの解説」の公表，総務省（オンライン），入手先 ⟨http://www.soumu.go.jp/menu_news/s-news/01ryutsu03_02000125.html⟩（参照 2017-07-10）．
[2] サイバーセキュリティ戦略本部：日本年金機構における個人情報流出事案に関する原因調査結果，サイバーセキュリティ戦略本部（オンライン），入手先 ⟨https://www.nisc.go.jp/active/kihon/pdf/incident_report.pdf⟩（参照 2015-08-20）．
[3] IPA 独立行政法人情報処理推進機構：「高度標的型攻撃」対策に向けたシステム設計ガイド，IPA 独立行政法人情報処理推進機構（オンライン），入手先 ⟨https://www.ipa.go.jp/files/000046236.pdf⟩（参照 2015-02-17）．
[4] 佐藤 信，杉本暁彦，林 直樹，磯部義明，佐々木良一：マルウェアによるネットワーク内の挙動を利用した標的型攻撃における感染経路検知ツールの開発と評価，情報処理学会論文誌，Vol.58, No.2, pp.366–374 (2017).
[5] 三村聡志，佐々木良一：プロセス情報と関連づけた通信情報保全手法の提案，情報処理学会論文誌，Vol.57, No.9, pp.1944–1953 (2016).
[6] World Wide Web Consortium: RDF 1.1 Primer, avail-

able from ⟨https://www.w3c.org/TR/rdf11-primer/⟩ (accessed 2016-11-19).
[7] 川口信隆，築地原護，井手口恒太，谷川嘉信，冨岡英勤：不審活動の端末間伝搬に着目した標的型攻撃検知方式，情報処理学会論文誌，Vol.57, No.3, pp.1022–1039 (2016).
[8] 海野由紀，森永正信，山田正弘，鳥居 悟：標的型サイバー攻撃におけるシステム内部の諜報活動検知の提案，コンピュータセキュリティシンポジウム 2012 論文集，Vol.2012, No.3, pp.360–367 (2012).
[9] 遠峰隆史，津田 侑，神薗雅紀，杉浦一徳，井上大介，中尾康二：複数ホストを横断可能なタイムライン型イベントログ閲覧システム，信学技報，Vol.113, No.502, pp.125–139 (2014).
[10] 松本光弘，高橋洋一，白木宏明，大松史生：標的型サイバー攻撃の感染範囲特定方式に関する提案，第 76 回全国大会講演論文集，Vol.2014, No.1, pp.539–540 (2014).
[11] Hossain, M.N., Milajerdi, S.M., Wang, J., Eshete, B., Gjomemo, R., Sekar, R., Stoller, S.D. and Venkatakrishnan, V.N.: SLEUTH: Real-time Attack Scenario Reconstruction from COTS Audit Data, *Proc. USENIX Secur.*, pp.487–504 (2017).
[12] JPCERT/CC：インシデント調査のための攻撃ツール等の実行痕跡調査に関する報告書，JPCERT/CC（オンライン），入手先 ⟨https://www.jpcert.or.jp/research/ir_research.html⟩（参照 2016-07-01）.
[13] AT&T Research: Graphviz – Graph Visualization Software Envisioning connections, Graphvizq (online), available from ⟨http://www.graphviz.org/⟩ (accessed 2016-11-19).
[14] Shinogi, S.: ShinoBOT – The rat/bot malware simulator, ShinoBOT Can you detect an APT like me? (online), available from ⟨http://shinobot.com/top.php⟩ (accessed 2016-07-23).
[15] 日本年金機構における不正アクセスによる情報流出事案検証委員会：検証報告書，厚生労働省（オンライン），入手先 ⟨http://www.mhlw.go.jp/file/05-Shingikai-10201000-Daijinkanbousoumuka-Soumuka/0000095309.pdf⟩（参照 2016-10-05）.
[16] JPCERT/CC：高度サイバー攻撃への対処におけるログの活用と分析方法，JPCERT/CC（オンライン），入手先 ⟨https://www.jpcert.or.jp/research/apt-loganalysis.html⟩（参照 2016-02-29）.
[17] トレンドマイクロ：標的型サイバー攻撃最新動向，トレンドマイクロ（オンライン），入手先 ⟨http://sp.trendmicro.co.jp/jp/trendpark/apt/201606-1/20160617004204.html⟩（参照 2016-08-20）.

島川 貴裕

2016 年東京電機大学未来科学部情報メディア学科卒業．同年 4 月より同大学大学院未来科学研究科情報メディア学修士課程．ネットワークフォレンジックの研究に従事．

佐藤 信 （正会員）

2013 年創価大学大学院工学研究科修士課程修了．同年東京電機大学大学院先端科学技術研究科入学．センサネットワーク，ネットワークフォレンジック技術に関する研究に従事．

佐々木 良一 （正会員）

1971 年 3 月東京大学卒業．同年 4 月日立製作所入社．システム開発研究所にてシステム高信頼化技術，セキュリティ技術，ネットワーク管理システム等の研究開発に従事．2001 年 4 月～2018 年 3 月まで東京電機大学教授，4 月より東京電機大学特命教授．工学博士（東京大学）．2002 年情報処理学会論文賞受賞．2007 年および 2017 年に総務大臣表彰等．著書に，『IT リスクの考え方』（岩波新書，2008 年）等．日本セキュリティ・マネジメント学会会長，内閣官房サイバーセキュリティ補佐官等を歴任．本会フェロー．