外部から不正侵入されたシステムのログ解析支援ツールの開発

中野 心太[1,*] 早稲田 篤志[2] 村上 洋一[2] 岸本 頼紀[2]
花田 真樹[2] 布広 永示[2] 関口 竜也[3] 折田 彰[3]

概要：マルウェアは，セキュリティサービスの無効化や，自身の動作に関連する痕跡情報を改ざんするなどの機能を用いて，ユーザ及びアンチマルウェアプログラムによる検知を免れようとしている．しかし，ファイルを作成または変更するマルウェアは，ファイル管理テーブル，レジストリ，イベントログ，および通信ログに痕跡情報を残すことがよく知られている．本研究では，システム上に残された痕跡情報から，マルウェアがシステムに侵入した後の挙動パターンを調べ，その調査結果を用いてマルウェアによる改ざん内容や攻撃者の目的などを解析するためのツールを開発している．本発表では，痕跡情報を人手で抽出する際の判断基準とその判断基準をログ解析支援ツールとして実装する開発内容について説明する．さらに，通常の作業中でも記録されているログ情報のフィルタリング処理，改ざん等の不正な操作が行われたと思われるログを抽出する痕跡情報抽出処理について報告する．

キーワード：マルウェア，ログ解析，デジタル・フォレンジック

Development of Log Analysis Support Tool for a System which was invaded illegally from Outside

Shinta Nakano[1,*] Atsushi Waseda[2] Yoichi Murakami[2] Yorinori Kishimoto[2]
Masaki Hanada[2] Eiji Nunohiro[2] Tatsuya Sekiguchi[3] Akira Orita[3]

Abstract: Malware tries to evade detection by users and anti-malware programs using its functions such as disabling security services and falsifying traces of its activities. However, it is well known that the malware which creates or modifies files leaves some trace in the file management table, registry, event log and communication log. In this study, we are analyzing the behavior pattern of malware based on the trace information left on the system. Also, based on the results of this analysis, we are developing a tool to analyze the contents of falsification by malware and the purpose of the attacker. In this presentation, we will explain the criteria of judgement for manually extracting the trace information and the details of the analysis tool which implements it. Furthermore, we will report the filtering process of the log information recorded even during normal work and the extraction process of the trace information which extracts logs of unauthorized operations such as falsification.

Keywords: Malware, Log Analysis, Digital Forensics

1. はじめに

2017 年以降，クリプトマイニングを行うマルウェアなどの台頭によって減少の一途をたどっていたランサムウェアだが，2019 年に入り，国内でランサムウェアの感染を目的とするばらまき型メールの攻撃件数が増加していることが報告されている[1]．RAT（遠隔操作ツール）によるシステムへの侵入後にランサムウェアを感染が行われた被害事例なども報告されており，無差別な攻撃による脅威のみならず，特定の組織・企業に対象を絞り，情報窃取を目的とした攻撃を行う標的型攻撃の脅威が増大しており，これらの脅威による被害が絶えないため，デジタル・フォレンジックの重要性が高まっている．

一般的にデジタル・フォレンジックでは，複数のツールを使用してマルウェアや攻撃者が残した痕跡を調査する．この時，システムログや MFT など OS 内に個別に残された

1 東京情報大学大学院 総合情報学研究科
Graduate School of Informatics, Tokyo University of Information Sciences.
2 東京情報大学 総合情報学部
Department of Information Sciences, Tokyo University of Information Sciences.
3 株式会社日立システムズ サイバーセキュリティリサーチセンタ
Hitachi Systems, Ltd. Cyber Security Research Center

* g19004sn@edu.tuis.ac.jp

痕跡をそれぞれのデータに対応したツールを使用し，それらの情報をまとめて全体像を得る．しかし，これらで使用されるツールは専門的なものが多く，詳細な分析はできるが全体像の概要を確認することには向かない．また，ラテラルムーブメントによる攻撃の調査などでは，ネットワーク上の複数コンピュータを横断的に調査する必要があり，詳細なデータから概要を得ることは煩わしい．

そこで，被害調査に際して全体の概要を得る支援ツールを提案する．膨大な痕跡の詳細を分析する前段階として，全体の痕跡の概要を視覚的に表現できれば，サイバー攻撃被害調査の初動の助けとなると考えられる．

本研究では，サイバー攻撃を想定したログに対するログ解析支援ツール（以下支援ツール）の試作を行った．本支援ツールはログ解析において一般的に手作業で行っていた知見（以下 know-how）について整理し，これらを機能化して実現することで，痕跡が残る複数のログを元に全体概要を可視化する．以下，2 章では関連研究について述べる．3 章では作成した支援ツールの概要とシステム構成について述べる．4 章では支援ツールに実装した know-how について述べる．5 章では本システムの適用例について述べる．6 章では考察を述べる．7 章ではまとめ，今後の展望について述べる．

2. 関連研究

ラテラルムーブメントの調査を行うツールとして，LogonTracer[2]などがあげられる，LogonTracer では，攻撃の被害を受けた範囲を調査するために，アカウントのログオンが行われたホスト間をリンクで繋ぎ可視化が可能である．このツールを使用することによって，感染が行われたコンピュータが判明している場合に，そのホストとリンクされているコンピュータを調査することが可能となり，不正ログオンに使用されたアカウントや感染の拡大が行われたホストなど，ラテラルムーブメントの痕跡を見つけることに繋がる．しかし，このツール単体ではログオンが行われたホストの情報を時系列で整理することができないため，通信の順序やラテラルムーブメントをひと目で調べることができない．

また，SysmonSearch[3]では，Microsoft が提供する Sysmon という通常では記録されない Windows OS の詳細なログとイベントログに記録するためのツールのログを解析することで，端末上で行われた不審な挙動やプロセスから関連するファイルやレジストリ情報を調査することが可能である．

そして，常時証跡収集と調査を行い，組織とシステムを完全に統合した形で運用することでデジタル・フォレンジックにかかるコストの低減と悪性なソフトウェアの動作に対して迅速な対処を可能としている．

しかし，Sysmon を予めインストールしておく，常時ログを記録しておく必要があるなど，事前の準備が必要である．

また，このツールでは特定のプロセスに関連するレジストリの洗い出しなど詳細な調査が出来る反面，調査したログの全体の流れをまとめるような機能はない．

先行研究の多くは，それぞれのログに対する分析が主であり，全体像を得るためには複数のツールを利用する必要があることがわかる．また，より詳細な分析には向いているが，全体概要のように情報を圧縮することなどには向いていないこともわかった．

3. 支援ツールの概要

一般的なログ解析ツールでは，異なる痕跡の組み合わせた分析や全体概要を得ることが難しい．そこで，本支援ツールでは，調査対象端末から得られた証跡から主要な情報を抽出し，攻撃の概要と全体像を把握することを目的とする．

全体の概要を得る方法として，手作業で行っていた know-how に着目する．概要を得るためには，痕跡情報のうち複数の情報を組み合わせる必要がある．例えば，MFT の FN 属性より SI 属性の方が古いファイルは改ざんの可能性があるため，このファイルに関する他のログファイルを調査し状況を把握する．このように，know-how の多くは 1 つの痕跡情報に関する他のログ情報を関連づけて抽出することで得ることができると考えられる．すなわち，他のログで現れた怪しい痕跡情報を基準にしたフィルタリングを実現できれば，概要を得る主要な情報を抽出できる．また，ブルートフォース攻撃のような大量のログは単純に抽出しただけでは見難くなる．そこで，類似情報をグループ化してまとめることで，概要を視覚的にわかりやすくできる．本試作では５つの know-how についてのみの実装とした．本支援ツールの具体的な処理として，システムは調査対象のハードディスク及び仮想イメージファイルから log2timeline/plaso[4]を用いて抽出し，ElasticSearch[5]へインデックスする．この時，EvtxToElk[6]を利用することで，直接調査対象のログをインデックスすることが可能となる．ユーザは，本支援ツールの，証跡のフィルタリング，グラフ化，及び全体像の把握を行うためのログ可視化などの機能を起動する．図 1 に支援ツールのシステム概要を示す．ここで，調査対象 A~C はそれぞれ同じ組織内の端末であり各端末に対する攻撃に関して保全した証跡である．

4 章で示す know-how を支援ツールの各機能として実装することで，大容量のログファイルからマルウェアによって行われたと推測される痕跡を抽出することが可能となり，マルウェアの検知，リモートログオンの記録，ソフトウェアのインストールなどの主要なイベントを時系列にマッピングすることが出来る．この結果，ひと目で攻撃の大まかな流れを把握することが可能となる．

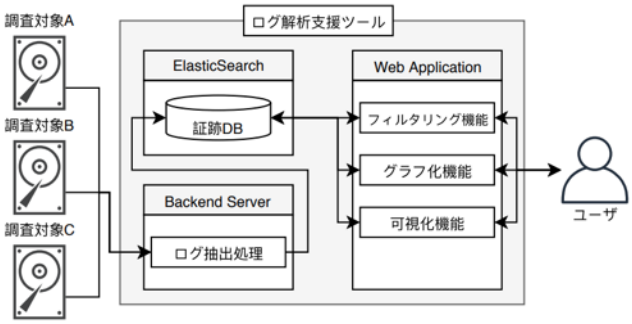

図 1 支援ツールのシステム概要

Figure 1 System overview of support tool

4. know-how の概要と実装

本支援ツールの開発にあたって，人手によるデジタル・フォレンジックを行う際に解析の指標とする次に示す 5 つの know-how を実装した．

① タイムスタンプの SI，FN 属性の活用
② 時間単位別 MFT レコード数の活用
③ ディレクトリ別のタイムスタンプ外れ値の検知
④ Prefetch の動作特性の活用
⑤ Security イベントログのリモートログオン記録の活用

これら 5 つの know-how の処理は，図 1 のフィルタリング機能の中に実装し，それらの処理をグラフ化機能，可視化機能から必要に応じて呼び出すことによって攻撃活動の大まかな把握が可能となる．

4.1 タイムスタンプの SI，FN 属性の活用

4.1.1 概要

Windows の MFT 内のレコードに記録されているタイムスタンプ情報のうち，SI 属性のものについては API から容易に変更が可能であることに対し，FN 属性の変更は困難である．SI 属性のタイムスタンプの多くは FN 属性のタイムスタンプと同時刻もしくはそれ以降の時刻である．また，マルウェアが作成したファイルのタイムスタンプを改ざんする際，同ディレクトリ内の他のファイルのタイムスタンプをコピーして自身のタイムスタンプに設定することがある．これらの特性に着目し，FN 属性のタイムスタンプよりも SI 属性のタイムスタンプが古くなっているレコードはタイムスタンプの改ざんが行われた可能性が高いと推測する．

4.1.2 実装方法

MFT から抽出した情報のうち，FN CreateTime，SI CreateTime の 2 つのタイムスタンプの情報を比較し，FN 属性よりも SI 属性が古いものをピックアップした．

4.2 時間単位別 MFT レコード数の活用

4.2.1 概要

MFT 内のレコードは，ファイルの書き込み，変更などの動作によって記録される．プログラムやアプリケーションのインストールなどによって大量のファイル書き込みが行われることで，時間単位別に集計した際，平常時よりも多くのレコードが作成されていることを確認できる．この特性に着目し，レコード数が増大している箇所のレコードからインストールされたプログラムを特定する．

4.2.2 実装方法

MFT のレコードを任意の単位時間でグルーピングし，件数のカウントを行った．

4.3 ディレクトリ別のタイムスタンプ外れ値の検知

4.3.1 概要

プログラムがインストールされたディレクトリ配下のファイル群のタイムスタンプは，同時期に作成されることから，近い日時になることが多い．この特性に着目し，同一ディレクトリ配下において，他のファイルと著しくタイムスタンプが異なるファイルに関連するレコードについてピックアップする．

4.3.2 実装方法

MFT に記録されたレコードの FilePath より，ディレクトリ別にグルーピングを行い，各グループ内で最も FN 属性のタイムスタンプが高いものをピックアップした．

4.4 Prefetch の動作特性の活用

4.4.1 概要

Windows 系 OS においてパスワードの窃取，認証情報の取得を行う際に用いられる mimikatz は，平常時には記録されない特徴的なイベントログが記録されることが報告されている[7]．mimikatz には，PowerShell からファイルレスで実行が可能なもの，exe 形式のものなど複数の実行形式が存在し，それぞれ残される痕跡が異なる．また窃取する情報によっても残される痕跡が異なり，パスワードハッシュを窃取する場合，イベントログには反映されず，Prefetch にのみ記録されることが報告されている[8]．この特性に着目し，悪性ソフトウェアが動作したか否かを特定する．

4.4.2 実装方法

Prefetch にはファイルの実行形跡（最終実行日時）が記録されるため，mimikatz に関するログの有無で判断を行った．

4.5 Security イベントログのリモートログオン記録の活用

4.5.1 概要

Windows 系 OS においてログオンの際に，Security イベントログにログオン情報が記録される．記録される情報のロ

グオンタイプを確認することで，ログオンの際にネットワークを経由したのか，対話型ログオンが行われたのかなどを判別することができる．特に，ログオンタイプが 10 のものは Windows 標準のリモートデスクトップ機能を用いてログオンが行われたものであり，mimikatz を用いて窃取した認証情報をもとにリモートデスクトップでラテラルムーブメントを行う際に残される痕跡を特定する．

4.5.2 実装方法

Security イベントログにイベント ID: 4624 が残されている場合はログオンの成功，イベント ID: 4625 ではログオンの失敗を示す，ログオンが成功しているもののうち，ログオンの種別を示すログオンタイプが 10 になっているレコードについて，リモートデスクトップ機能を用いてログオンしたものとして取得した，またこの know-how からログオンの失敗イベントが連続して出現する場合ブルートフォース攻撃の可能性も検出することが可能である．

5. 適用例

本支援ツールの可視化機能として，ログのフィルタを行った結果を簡略化して時系列順に一覧表示することが可能である．可視化機能の適用例として，ラテラルムーブメントが行われた被害端末に対して大まかな攻撃の流れを把握するための初動調査が挙げられる．

手作業での追跡に時間がかかるリモートログオン記録の分析や，Windows 標準のマルウェア対策ソフトである Windows Defender のマルウェア検知ログを時系列にマッピングし，4 章で述べた know-how によってフィルタされたログについても，タイムラインにマッピングが可能である．

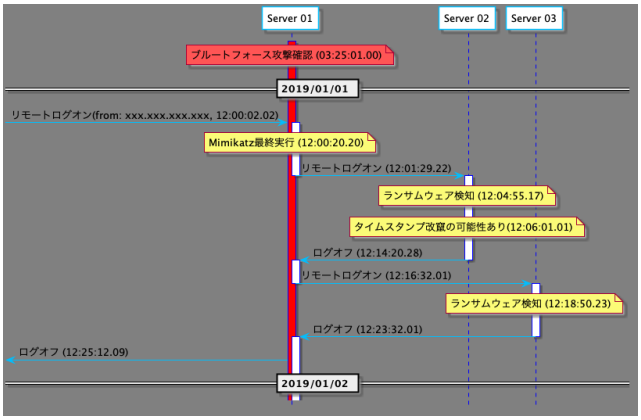

図 2　ラテラルムーブメントのタイムライン

Figure 2　Lateral Movement Timeline

図 2 にログを可視化した例を示す．図 2 では，server01~03 の各ライフラインが調査対象端末に対応し，縦軸の上から下へ向かって主要なイベントが時系列にマッピングされている．これらのイベントは，4 章で述べた know-how を基に絞り込んだログから得られた情報を簡略化して表示している．例として，4.1 の SI，FN 属性に着目したタイムスタンプ改竄がされた可能性を，図 2 の server02(12:06:01.01)に表示している，また 4.5 のリモートログオン履歴から，一定期間，一定間隔でログオンの失敗イベントが続いているログを簡略化し，server01(03:25:01.00)にブルートフォース攻撃が行われていることを表示している．

ユーザは，調査対象端末の仮想ハードディスクイメージや証跡ファイルなどを指定してツールを起動することで，自動で証跡がインデックスされ，事前に設定されたクエリや 4 章で述べた know-how によるフィルタが行われた結果得られたイベントをこのようなタイムライン形式で確認することが可能である．

6. 考察とまとめ

4 章で提案した 5 つの know-how の実装内容について考察する．

6.1 タイムスタンプの SI，FN 属性の活用

図 3 に，know-how（タイムスタンプの SI，FN 属性の活用）を実装することで確認されたタイムスタンプを改ざんされた可能性のあるレコードをフィルタしたビューを示す．

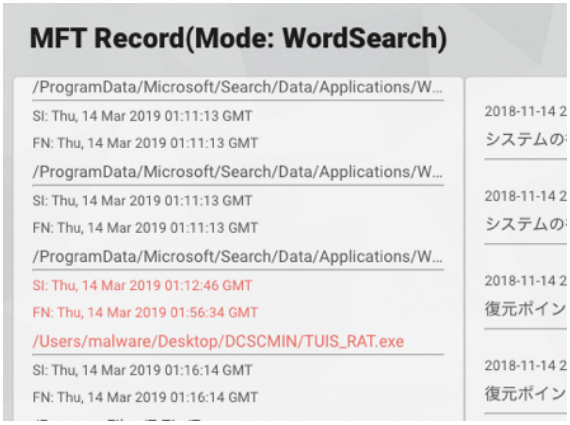

図 3　タイムスタンプを改竄された可能性のあるレコード

Figure 3　SI < FN Creation Time MFT Record

図 3 より，SI 属性のタイムスタンプを改ざんしたレコードのハイライトが行われている．しかし，Windows インストール時などにできるファイルにも同じ特徴を持つレコードが存在する場合があるため，精度を上げるために Windows インストール日時などから調査する対象の期間を狭める必要がある．

6.2 時間単位別 MFT レコード数の活用

図 4 に，know-how（時間単位別 MFT レコード数の活用）

を実装することで確認された時間単位別 MFT レコード数のグラフ化を行ったビューを示す．

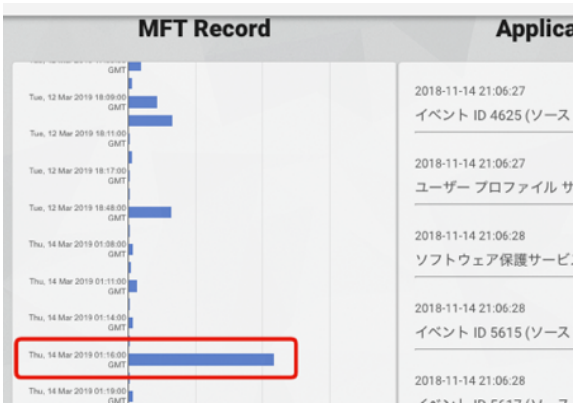

図 4　時間単位別 MFT レコード数
Figure 4　MFT Records Amount

図 4 より，RAT 用モジュールに感染した単位時間のグループで，多くの MFT レコード数が記録されていることが読み取れる．通常使用時においても MFT レコード数の増加が予想されるソフトウェアのインストールなどと区別を行うため，ソフトウェアのインストール日時と照らし合わせることで，より確認する範囲を狭めることができる．

6.3 ディレクトリ別のタイムスタンプ外れ値の検知

図 5 に，know-how（ディレクトリ別のタイムスタンプ外れ値の検知）を実装することで確認されたディレクトリ別の外れ値タイムスタンプをフィルタしたビューを示す．

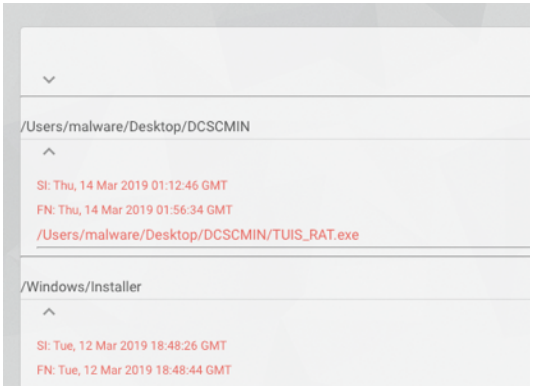

図 5　ディレクトリ別の外れ値タイムスタンプ
Figure 5　Outlier Timestamp by Directory

図 5 より，ディレクトリ別にグループ化し，同ディレクトリ内の他のファイルのタイムスタンプと著しくかけ離れているものをハイライトした．ディレクトリごとに調査を行うため，6.1，6.2 のフィルタ後にフィルタを行うことが望ましい．

6.4 Prefetch の動作特性の活用

know-how（Prefetch の動作特性の活用）を実装することで，プログラムの最終実行時間が記録される Prefetch を調査し，mimikatz のような攻撃に利用されるツールが実行されたことを知ることができる．

exe 版 mimikatz を起動し，認証情報を取得した端末において，Prefetch から mimikatz.exe に関するログを検索することで，実行の痕跡を得ることができた．

6.5 Security イベントログのリモートログオン記録の活用

Know-how（Security イベントログのリモートログオン記録の活用）を実装することで，イベントログから，イベント ID4624（ログオン）のタイプ 10 を検索し，リモートログオンの記録を抽出することができた．外部からブルートフォース攻撃などの方法でログオンが行われた場合，組織外の IP アドレスが記録されるため，ログオン元の IP アドレスでグルーピングすることで，普段から利用されているのか，攻撃に利用されているのかがわかる．

7. 今後の展望

今後の展望として，ログ取得用実験環境において，攻撃目標・シナリオを設定した複数の攻撃者に攻撃を実施させることでログのサンプル数を増やし，攻撃者や手順の違いによるログへの影響分析，攻撃手順とログ・実施目的の関連付けを行い，攻撃者の行動特性分析を行う．並行して，現状 MFT，イベントログ，Prefetch のみの解析を行っているため，ブラウザキャッシュやレジストリの調査についても検討し，システムの改良と新規 Know-How の実装を進めてログ抽出の精度上昇を図る．

謝辞 本研究を進め論文を執筆するにあたり，貴重なご意見と資料を提供して頂いた株式会社日立システムズサイバーセキュリティリサーチセンタのエンジニアの皆様，システム開発や機能評価などにご協力いただいた東京情報大学布広ゼミの学生の方々に深謝いたします．

参考文献

[1] Trend Micro．“2019 年 第 1 四半期セキュリティラウンドアップ：データを暗号化する標的型攻撃”．
https://resources.trendmicro.com/jp-docdownload-form-m121-web-2019q1-securityroundup.html

[2] JPCERT/CC “LogonTracer を用いた不正ログオンの調査”．
https://blogs.jpcert.or.jp/ja/2018/01/logontracer2.html

[3] JPCERT/CC “Sysmon ログを可視化して端末の不審な挙動を調査~SysmonSearch~”．
https://blogs.jpcert.or.jp/ja/2018/09/SysmonSearch.html

[4] “log2timeline/plaso”． https://github.com/log2timeline/plaso

[5] “Elasticsearch”． https://www.elastic.co/jp/

[6] “EvtxToElk: A Python Module to Load Windows Event Logs into ElasticSearch”． https://dragos.com/blog/industry-news/evtxtoelk-a-python-module-to-load-windows-event-logs-into-elasticsearch/

[7] IIJ．“IIJ Technical WEEK2017 Mimikatz 実行痕跡の発見手法”． https://www.iij.ad.jp/dev/tech/techweek/pdf/171108_02.pdf

[8] 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”．
https://www.jpcert.or.jp/research/20160628ac-ir_research.pdf