物理マシンを採用したマルウェア動的解析環境における仮想マシンと同等の復旧速度の実現

阿曽村一郎† 武田康博†
†株式会社 みずほフィナンシャルグループ
〒100-8176 東京都千代田区大手町 1-5-5
{ichirou.asomura,yasuhiro.takeda}@mizuhofg.co.jp

概要：マルウェアには自身の実行環境を判別する機能により，物理マシンでは動作するが仮想マシンでは動作しないものがあるため，動的解析環境には物理マシンを用いることが望ましい．一方で動的解析環境には解析後に OS を短時間で解析前の状態に戻す仕組みが求められるため仮想マシンが採用されていることが多い．物理マシンを用いた動的解析環境を構築するためには，OS を短時間で解析前の状態に戻す仕組みを実現させることが課題となる．我々は仮想マシンで動作しないマルウェアの動的解析を行うために，この課題を解決した物理マシンを用いた動的解析環境を構築した．本稿では，短時間で OS を解析前の状態に戻す仕組みについて述べる．

キーワード： マルウェア解析，物理マシン，仮想マシン，動的解析環境，ZFS，iSCSI，スナップショット

Sandbox: Proposal of bootable system snapshot for physical machine.

Ichiro Asomura† Yasuhiro Takeda†
†Mizuho Financial Group,Inc
Otemachi Tower, 1–5–5 Otemachi, Chiyoda–ku, Tokyo 100–8176, Japan
{ichirou.asomura,yasuhiro.takeda}@mizuhofg.co.jp

Abstract: Since some of malwares detect virtual machine, physical machine is preferable as a dynamic analysis environment. However, virtual machines are popular because malware can be executed and analyzed on them without having to reinstall operating system and application software every time. In order to create dynamic analysis environment using physical machine, there is a big challenge with instant system recovery. In this paper, we propose bootable system snapshot for physical machine which allows instant system recovery.

Keywords: Malware Analysis, Physical Machine, Virtual Machine, Sandbox, ZFS, iSCSI, Snapshot

1. はじめに

2016 年は 2015 年に比べてマルウェア総数は 29%増加[1]しており，情報セキュリティの脅威のうち 40%はマルウェアが関連しているとされている[2]．また，新型，あるいは亜種のマルウェアはウィルス対策ソフトでは検知できないものも多い[3]．このような状況下でマルウェアによる被害を未然に防ぐためにも，組織のメールサーバ等に到着するマルウェアや，新しいマルウェアの特性を迅速に解明し，可能であれば事前の対策を講じることが重要となる．また，万が一組織内でマルウェアによる被害が発生した場合にも，マルウェアの特性を解明の上，対策を講じる必要がある．このため，未知のマルウェアも含めて効率的に解析を行い，対策を迅速に行うためには適切な解析手法を用いることが重要である．マルウェアの解析手法は，マルウェアを動作させる解析方法とマルウェアを動作させないで解析する方法に分類できる．一般的に前者を動的解析，後者を静的解析と呼ぶ．

動的解析は，解析環境上でマルウェアを実際に動作させ，モニタリングした結果どのような振る舞いをするかを調べる手法である．動的解析の解析範囲は限定的ではあるが，一般的に静的解析よりも短時間でマルウェアの挙動を確認することができる．また，マルウェア自身がダウンロードした設定ファイルやモジュールがあって初めて有効になるような機能については，動的解析のほうが解析結果を得やすい．

マルウェアが持つ個々の機能に対して正確な解析結果を求める場合には，動的解析に加え，リバースエンジニアリング等の手法を用いる静的解析を行う必要がある．しかし，多くのマルウェアはパック（圧縮・難読化処理）されているため，解析には困難が伴い，時間を要する傾向にある．セキュリティ対策を担う部門では，急増する新種・亜種のマルウェアにも対応する必要があり，そのすべてを手動で解析することは，迅速に対策を行うという観点からは賢明とは言えない．

このような背景のため，効率的にマルウェアを解析するためには，全体のマルウェア解析プロセスの中の動的解析で多くの情報を収集し，静的解析に要する時間を削減するという点が重要である．

本稿では，解析対象となるマルウェアの動作に求められる動的解析環境の課題について述べ,その解決策を提案し，実装したうえで評価する．課題がどのように解決されたかを結論づけ，今後の展望を示す．

2. 関連研究

2.1 金融系マルウェアの攻撃手法

西田らによれば，金融系マルウェアの中には外部から設定情報を取得し，それにより挙動を変えるものが存在すると報告されている[4]．このようなマルウェアに対して迅速に対策を行うためには，先に述べたとおり動的解析が有効である．

2.2 マルウェアの対仮想化処理

マルウェアの中には，仮想化機構によって作られた環境では実行を終了して解析を妨害するものがある．大山によれば，FFRI Dataset 2016 に含まれるマルウェアの少なくとも 6.8%は対仮想化処理が実装されていると報告されている[5]．この中で個別の挙動として紹介されている TrojanSpy:Win32/Ursnif.HN も金融系マルウェアである．

これらから，特に金融系マルウェアの動的解析を行うためには対仮想化処理の回避を考慮する事が重要であると言える．

3. 動的解析環境の課題

3.1 課題

マルウェアの動的解析環境には，仮想マシンが採用されていることが多い．仮想マシンが持つ VM Introspection とスナップショット機能が動的解析に利用されることが多いことが理由としてあげられる．仮想マシンの VM Introspection を用いることによって，仮想マシンからゲスト OS のメモリやデバイスを監視することができる[6]．この機能を用いると直接仮想マシンを通してマルウェアの挙動をモニタリングできるため，仮想マシン上に構築したゲスト OS にエージェントを入れる必要がない点が優れている．仮想マシンを用いたエージェントレス型の動的解析環境は，マルウェアにエージェントや，エージェント関連のプロセスを検知されることがない．また，動的解析は，OS にマルウェアを感染させたうえで解析を行うため，解析完了後に感染前の状態に戻す必要がある．これは，解析前の状態のスナップショットを取得した仮想マシンを利用することによって，解析完了後ただちに解析前の状態に戻すことによって実現されている[7].

一方，マルウェアには，仮想マシンで動作させた場合その存在を検知し挙動を変えるものもあり[8]，仮想マシンに依存した動的解析環境では本来の動作結果を得られないことがある．

3.2 金融系マルウェアの特徴

金融系マルウェアとは，不正送金による金銭利益を最終目的として，インターネットバンキングの認証情報を搾取する「オンライン銀行詐欺ツール」を指し，高度に作りこまれているものが多い．例えばクラウド上に管理コンソールを持ち，マルウェア自身が感染後自分自身をアップデートしたりモジュールを追加する機能を有したり，設定ファイルを更新したりすることができるものもある．犯罪目的に作成されているため，対仮想化処理をはじめとした解析を回避する機能が備わっているものが多い[9]．一方，我々のこれまでの解析経験において，ランサムウェアには対仮想化処理が行われていなかった．

本稿でバンキングトロジャンと呼ばれる金融系マルウェアに加えて，ランサムウェアを比較対象としてテストした．

4. 仮想マシンと物理マシンの動作比較

マルウェアは必ずしも環境が整っていれば動作するというものではない．サンドボックスを避けるために一定時間潜伏した状態となっている場合や，感染がマルウェアの実行タイミングに依存する等，同じ環境においてもマルウェアを実行できたりできなかったりする場合がある．このような場合は，同じマルウェアを同じマシンにて条件を変えて何回か実行させて解析を行う．仮想マシンと物理マシンの比較検証を行うにあたって、動作タイミングや実行時間などの設定をマルウェア毎に個別に行う場合があるが，仮想マシンと物理マシンの間では条件を揃えることとする．また，マルウェアの動作判定は C2 サーバへの通信リクエストの有無で行うものとする．

比較を行う仮想マシンの動的解析環境としては，Linux 上で動作する KVM と VMWare ESXi の 2 種類用意した．大山の報告においてマルウェアによる signature 検出が報告されていなかったものとして KVM を，最も多くのマルウェアにより signature 検出されていると報告されていたものとして VMWare を選択した[5].

各環境のシステム構成を示す．

表 1　各環境のシステム構成

	仮想マシン 1	仮想マシン 2	物理マシン
ハードウェア	HP DL80 Gen9	HP DL80 Gen9	HP ProDesk 400G2 Mini
ハイパーバイザ	KVM	VMWare ESXi	N/A
OS	Windows 7	Windows 7	Windows 7

表 2　本稿でテストしたマルウェア一覧

#	検体名 (推定)	Hash 値 (MD5)	動作結果		
			仮想マシン 1	仮想マシン 2	物理マシン
金融系マルウェア					
1	ursnif/dreambot	6cfee3e546359d7a9de0928f175bc030	×	×	○
2	ursnif/dreambot	7b9043f66a8e435aefc1be92e0051e46	×	×	○
3	Trickbot	08ba011df60438ccb9462e819e7ec722	○	×	○
4	Bebloh	751fcf8b44d307072d4d42ed1b12053e	○	×	○
5	Bebloh	f4b61f8a65507699c098718bb720418a	○	○	○
6	Dridex	6233778c733daa00ce5b9b25aae0a3cb	○	○	○
7	Tinba	3d40e69d56a9cff6596b60cd131272d7	○	×	○
8	Rovnix	8f57e96532068c1febbaaa2827116807	○	×	○
9	Trickbot	1ce6a0cac1b8e0fbed2ae1030ff6c7e0	×	×	×
10	GozNym	2a9093307e667cdb71884ecc1b480245	×	×	×
11	Bebloh	53744067c00ccf2bc77fd8ce6b96de43	×	×	×
ランサムウェア					
12	Locky(lukitus)-ダウンローダ	c5fa2fdbfec21bef66143b6b4e8f6d7f	○	○	○
13	Locky(lukitus)-ダウンローダ	5551117adb3b2e7ce62406b1f411b1e4	○	○	○
14	Locky-ダウンローダ	f46f93f7ab4dbc396dabfb02008f673b	○	○	○
15	Locky(lukitus)-ダウンローダ	ccabf6f184355e030345845a4275c8c9	○	○	○
16	Locky(lukitus)-ダウンローダ	2f9836851f44ff1635ecb4218dd036f5	○	○	○

○：動作した (C2 サーバへの通信リクエストがあった)

×：動作しなかった (C2 サーバへの通信リクエストがなかった)

4.1 仮想マシンと物理マシンでの動作結果

仮想マシンと物理マシンを用いたマルウェアの動的解析結果の一覧を表 2 に示す. 金融系マルウェアについては，物理マシン上では 11 検体中 8 検体の動作が確認され，最も良い結果を得られている. 仮想マシンは KVM が 11 検体中 6 検体，VMWare ESXi が 11 検体 2 検体の動作が確認されている．物理マシンで動作しなかった 3 検体については仮想マシンにおいても動作しなかった．このことからも金融系マルウェアを動的解析する場合は，物理マシンを採用したほうが効率よく解析できることがうかがえる. 一方で，ランサムウェアの動的解析を行った結果，環境を問わず感染することが確認された．なお，検証結果については同じ環境上であっても OS のバージョンや C2 サーバの有無や観測時間の長さなど，条件が変わることによって結果が変わることがある.

5. 動的解析環境における物理マシンの復旧方式の提案

5.1 物理マシンの課題

動的解析において，仮想マシンが利用される理由のひとつに「スナップショット機能」がある．動的解析において「スナップショット機能」は，マルウェアに感染させた OS を高速に感染前に戻すために用いられている．一方，仮想マシンにおける「スナップショット機能」は物理マシンにおいては標準的な機能ではなく，マルウェアに感染させた OS を感染前に戻す仕組みを実現するためには，完全性と復旧速度について課題がある.

5.1.1 既存システム

物理マシンを選択できる自動解析環境に ThreatAnalyzer[a] と Cuckoo Sandbox[b] がある．ThreatAnalyzer は物理マシンの復旧方式に Deep Freeze[c]という復旧用のソフトウェアを採用している．Deep Freeze はコンピュータを再起動するだけで，変更されたすべてのファイルや環境を使用前の状態に復旧できるとされているソフトウェアだが，Rovnix のようにシステムを破壊するタイプのマルウェアに感染した場合には元の状態に復旧させることはできず，完全性に課題を残す．また，Cuckoo Sandbox を物理マシンで利用する場合に，復旧させるためのシステムとして選択できる FOG[d]は物理マシンの HDD をイメージ化し，物理マシンの複製やバックアップを可能とするソフトウェアである． FOG は，PXE と TFTP を利用し，サーバに保存されている HDD のイメージをネットワーク越しに物理マシンの HDD に展開することによって復旧させる． FOG を利用した場合，完全にクリーンな OS が物理マシンの HDD に上書きされるため Rovnix のようなシステム破壊型のマルウェアに感染したとしても問題なく復旧させることができる．ただし，物理マシンのシステムイメージ全体をサーバからクライアントに転送するため，時間を要する．例えば，物理マシン上にて 100GB 程度になるディス

a http://nextit.jp/threatanalyzer/
b https://cuckoosandbox.org/
c http://www.faronics.com/ja/products/deep-freeze/
d https://fogproject.org/

クイメージを Gigabit Ether を用いて転送する場合は 7-10 分程度かかるため，仮想マシンと比較して速度の面で課題が残る．

5.2 提案システム概要

本章では Deep Freeze や FOG が持つ課題を解決するべく，ZFS と iSCSI を組み合わせる事で，物理マシンにおいても仮想マシンにおけるスナップショットに相当する機能を実現する仕組みについて述べる．

5.2.1 ZFS

ZFS は Oracle Solaris 上で実装されている先進的なファイルシステムであり，「優れたスケーラビリティ」，「管理のしやすさ」，「データの堅牢性」といった特徴を備えたファイルシステムである． ZFS は多機能であるが，その中でも「スナップショット機能」は特徴的である．スナップショットは，ある時点におけるファイルシステムの状態を保持することができる．ZFS の「スナップショット機能」を多くの仮想マシンが有する「スナップショット機能」を物理マシンにて実現するために応用することとした．

5.2.2 iSCSI

iSCSI は SCSI プロトコルを TCP/IP 上で使用する規格であり，XP 以降の Windows は iSCSI ターゲットのストレージを直接 HDD としてマウントすることができる．動的解析に使用する Windows7 は iSCSI サーバのストレージに OS を直接インストールすることができ，システムドライブとすることが可能だ．

5.2.3 ZFS+iSCSI

iSCSI ターゲットのストレージのファイルシステムとして ZFS を採用することにより，iSCSI ターゲットのストレージから起動した Windows7 は仮想マシンが有する「スナップショット」機能と同等の機能を使うことができる．つまり，動的解析を行うためにマルウェアに感染した物理マシンを，再起動するだけで瞬時に感染前の状態に戻すことが可能となった．

5.2.4 システム構成

提案システムに利用したソフトウェアとそのバージョンを示す．

表 3 提案システムの構成

		ソフトウェア名	バージョン
サーバ側	OS	Ubuntu Linux	16.04 LTS
	ZFS	ZFS on Linux	0.6.5.6
	iSCSI	LIO+targetcli	3.0
	DHCP	ISC DHCP Server	4.3.3
	TFTP	HPA's tftp server	5.2
物理マシン側	PXE	iPXE[e]	2017-05-09

システム構成の概要図を示す．

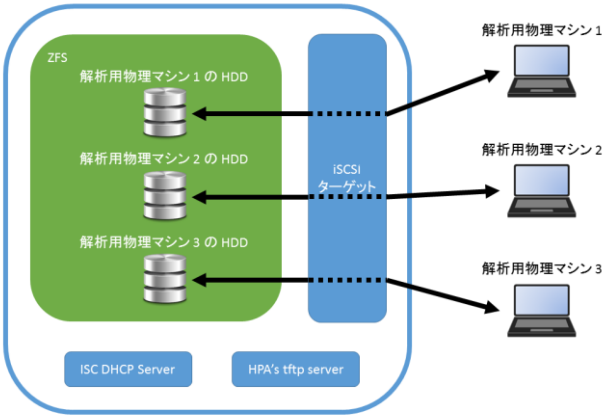

図 1 提案システム構成の概要図

5.3 起動プロセス

以下の図で起動プロセスを示す．

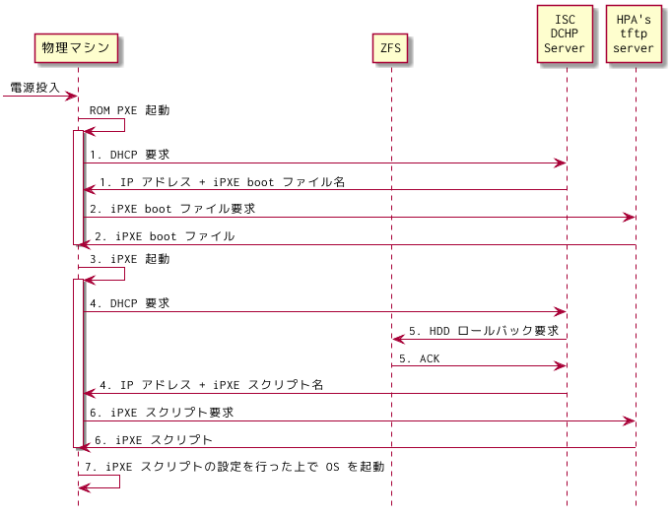

図 2 起動プロセスのシーケンス

起動プロセスは以下の流れで動作する．

1. NIC に内蔵された PXE クライアントにより DHCP 要求が行われ，IP アドレスと iPXE boot ファイル名が返却される．
2. PXE クライアントは TFTP サーバに対して iPXE boot ファイルを要求し，iPXE boot ファイルが返却される．
3. 返却された iPXE boot ファイルを iPXE クライアントとして実行する．
4. iPXE クライアントにより DHCP 要求が行われ，IP アドレスと iPXE スクリプトファイル名が返却される．
5. DHCP サーバは DHCP 要求に応答する前に ZFS の機能による HDD ロールバックを実行する．
6. iPXE クライアントは TFTP サーバに対して iPXE スクリプトを要求し，iPXE スクリプトが返却される．

e http://boot.ipxe.org/undionly.kpxe

7. 返却された iPXE スクリプト内容の設定を行ったうえで，OS の起動を開始する．

iPXE クライアントは 6 番目で返却される iPXE スクリプトの内容の通りに OS を起動するための設定を行うことができる．このシステムでは iSCSI ターゲットとして提供されるディスクを起動ディスクとして OS を起動するように設定した．

5.4 自動解析環境と組み合わせた使い方の例

前述のシステムは自動解析環境との連携が可能である．自動解析環境と組み合わせたシステム全体の概念図を示す．

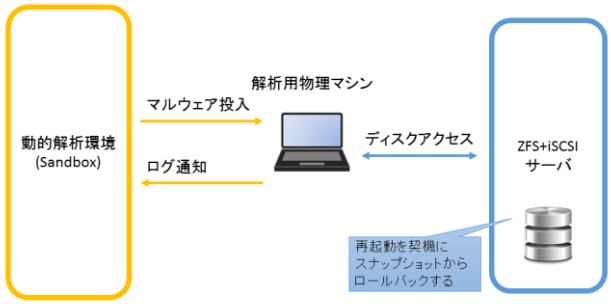

図 3 自動解析環境と組み合わせたシステムの概念図

物理マシンの再起動時に自動的に復旧されるため，連携のために特別な API を実行する必要はなく，自動解析環境側に修正を行う必要もない．

自動解析環境と組み合わせた場合の概要シーケンスを示す．

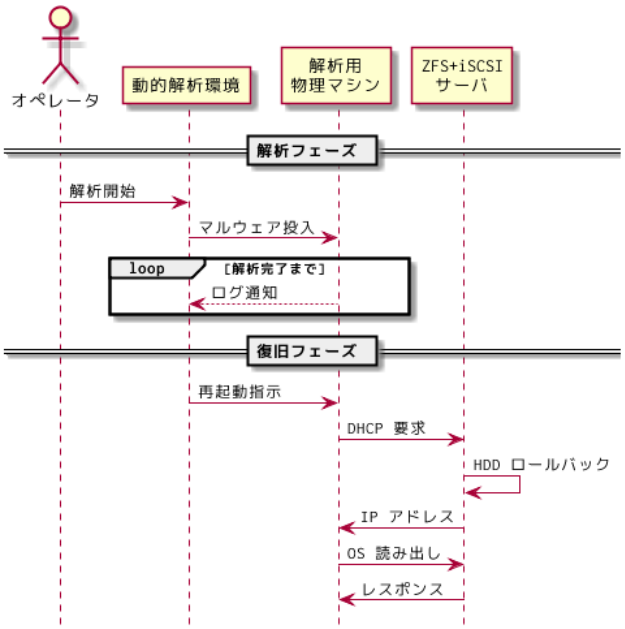

図 4 自動解析環境と組み合わせたシーケンス

6. 評価実験

本章では，物理マシンを利用した動的解析環境の性能を評価するために実施した評価実験の結果について述べる．

6.1 評価目的

評価実験を通じて次の点を確認することを目的とした．

1. 物理マシンを用いて動的解析を行うにあたって，解析完了後の復旧速度が仮想マシンと同等か
2. ZFS+iSCSI を採用した物理マシンの復旧方式は，動的解析に用いたマルウェアを物理マシンから除去しきれない可能性を否定することができるか

6.2 評価方法

評価は仮想マシン，物理マシンをマルウェアに感染させて動的解析を行い，解析後の復旧速度を計測する．動的解析に用いる Windows7 の RPC サービスへの接続が出来なくなった時点でシャットダウンとみなし，次に RPC に接続できた時点で起動完了とする．RPC への接続できない期間を 3 回計測し，平均を取った値を復旧速度とする．仮想，物理それぞれのマシン復旧方式として，仮想マシンは VMWare，KVM のスナップショット機能，物理マシンについては，Deep Freeze, FOG,そして ZFS+iSCSI の 3 種類を比較する．比較したマシンの種類については表 1.「各環境のシステム構成」に順じ，ネットワークは Gigabit Ether を用いるものとする．

6.3 評価結果

評価した結果，仮想マシンのスナップショット機能を利用した復旧方式の計測結果は，VMWare ESXi　の復旧速度は平均 10 秒，KVM においては平均 12.7 秒となった．物理マシンについては，Deep Freeze が平均 39 秒であり，復旧速度は物理マシンの中でもっとも速いが，Rovnix を感染させた場合にシステムが破壊され，再起動不能となった．FOG は，Rovnix 等の破壊型のマルウェアにも有効だが，サーバ上の HDD イメージをネットワーク越しに物理マシン上の HDD に展開する仕組みをとっているため，復旧速度は平均 627.3 秒であった．ZFS+iSCSI の復旧方式を採用した物理マシンは，平均 44 秒の復旧速度であった．複数の方式を比較した結果，ZFS+iSCSI の復旧方式を採用した物理マシンは Rovnix 等のシステムを完全に破壊するタイプのマルウェアの影響を受けずに高速に復旧されることを確認できた．

表 4　評価結果

	仮想		物理		
ソフトウェア名	VMWare ESXi	KVM	Deep Freeze	FOG	ZFS+iSCSI
バージョン名	6.0	Linux 4.4.0-89	8.38	1.2	表 3 に順ずる
1 回目(sec)	10	13	39	646	45
2 回目(sec)	10	12	39	625	42
3 回目(sec)	10	13	39	611	45
平均(sec)	10	12.7	39	627.3	44
備考			Rovnix を感染させた場合，再起動不能となる		

6.4 考察

6.4.1 物理マシンを用いた動的解析について

物理マシンを用いた動的解析を行う場合は，マルウェアが持つ解析回避機能のうち対仮想化処理については，原理的に回避可能だと考える．このため，表 2.の検証結果にて示されているように対仮想化処理機能が備わっている多くの金融系マルウェアを解析する場合の解析効率は上がる．

今回作成したシステムによって，課題となっていた復旧速度についても，仮想マシンと遜色ない速度を実現できることが分かった．また、ZFS+iSCSI を復旧方式として採用した場合，原理的に動的解析に用いたマルウェアを物理マシンから除去しきれない可能性を否定することができる．これらのことにより，特に物理マシンを採用した自動化された動的解析環境の大幅な解析効率向上に貢献できたと考える．

6.4.2 仮想マシンを用いた動的解析について

仮想マシンを用いた動的解析については，マルウェアが備えている対仮想化処理を回避する仕組みを導入することによって，解析可能なマルウェアを増やすことができると考えられる.仮想マシンを利用した場合のメリットとして，たとえば VM Introspection（あるいは類似の機能）を利用し，ハイパーバイザ側から直接マルウェアの軌跡を取得，トラップ可能な点があげられる．このため，物理マシンで課題となったエージェントレスの動的解析環境が実現可能である．また，ハードウェア部分は基本的に変更が困難な物理マシンと異なり，仮想マシンはハイパーバイザや仮想ハードウェアに機能を加えることも可能なため，より柔軟な解析を行える可能性がある．

6.4.3 今後の課題

物理マシンを自動化された動的解析に用いる場合，現段階では感染に用いる端末にエージェントを導入し，マルウェアの動作軌跡を取得する方式となる．このため，マルウェアの解析回避機能にエージェントやエージェントが用いるプロセス等を検知する機能が備わった場合には解析を検知される可能性があることや，仮想マシンであれば取得可能なメモリダンプを物理マシンで取得するためにはさらに工夫する必要があるなど，物理マシンを用いた動的解析，特に自動解析の分野はさらなる改善の余地がある．

謝辞　本稿を執筆するにあたって，仮想マシン，物理マシン別にマルウェアの動的解析を行っていただいた滝澤悦之さんに感謝いたします．

参考文献

[1] McAfee Labs 脅威レポート，https://www.mcafee.com/jp/resources/misc/infographic-threats-report-dec-2016.pdf
[2] IPA 情報セキュリティ 10 大脅威 2017 組織編，https://www.ipa.go.jp/files/000059212.pdf
[3] よくある相談と回答(FAQ) Q6-A6，https://www.ipa.go.jp/security/anshin/faq/faq-vi.html
[4] 西田雅太，太刀川剛，岩本一樹，遠藤基，奥村吉生，星澤裕二：静的解析と挙動観測による金融系マルウェアの攻撃手法の調査，コンピュータセキュリティシンポジウム 2014 論文集，Vol.2014，No.2，pp.859-866 (2014).
[5] 大山恵弘：マルウェアによる対仮想化処理の傾向についての分析，コンピュータセキュリティシンポジウム 2016 論文集，Vol.2016，No.2，pp.534-541 (2016).
[6] Survey: Virtual Machine Introspection Based System Monitoring and Malware Detection Techniques，https://www.cs.rochester.edu/u/hliao6/projects/other/os_survey.pdf
[7] 新井 悠，岩村 誠，川古谷 裕平，青木 一史，星澤 裕二，(2010)，アナライジング•マルウェア 3.6 章，オライリー・ジャパン
[8] An Overview of Malware Self-Defense and Protection，https://securingtomorrow.mcafee.com/mcafee-labs/overview-malware-self-defense-protection/
[9] Financial Threats Review 2017，https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/istr-financial-threats-review-2017-en.pdf