Vol.2011-DPS-146 No.54
Vol.2011-CSEC-52 No.54
2011/3/11

マルウェア動的解析オンラインサービスの脆弱性解消方法の検討

村上 洸介* 織井 達憲* 笠間 貴弘*
吉岡 克成* 松本 勉*

近年，任意のユーザから実行ファイル等の投稿を受け付け，解析環境（サンドボックス）内で実行し，その挙動を解析して結果をユーザに提供する「マルウェア動的解析オンラインサービス」が人気を集めている．我々はこれまで，特別に設計したデコイを解析対象としてサービスに投稿することで，サンドボックスの情報を暴露させ，その情報を基に解析環境の検知・回避を行う攻撃に対して，当該サービスが脆弱であることを指摘している．特に，インターネット接続型のサンドボックスが用いる IP アドレスは，容易に変更することが難しい場合も多く，攻撃者に特定された場合にサービスの解析結果に重大な影響を及ぼす．そこで本稿では，サンドボックスをインターネットに接続する際に，検体を投稿するユーザのホストをプロキシとして用いることで，サンドボックスの IP アドレスを攻撃者から隠蔽する手法を提案する．また，提案手法の具体的な実現例について示し，考察を行う．

How to Resolve the Vulnerability of Public Malware Sandbox Analysis Systems

Kousuke Murakami* Tatsunori Orii* Takahiro Kasama*
Katsunari Yoshioka* Tsutomu Matsumoto*

Recently, the use of public Malware Sandbox Analysis Systems (public MSASs), which receive online submissions of possibly malicious files or URLs from an arbitrary user, analyze their behavior by executing or visiting them by a testing environment (i.e., a sandbox), and send analysis reports back to the user, has increased in popularity. In previous study, we have pointed out a vulnerability of public MSASs that the host information (i.e., Windows product key, MAC address, IP address, etc.) of a sandbox used in public MSAS can be easily disclosed by an attacker who submits a decoy sample dedicated to this purpose, and an attacker can detect public MSAS and conceal potential malicious behavior of malware by using the disclosed information. In particular, if the IP address used by an Internet-connected sandbox is identified by an attacker, then it causes serious influence on an analysis result of the service. In this paper, we propose a method that uses a service user as a proxy when the sandbox connects the internet for hiding its IP address. We also show an implementation example of the proposed method.

1. はじめに

近年，コンピュータウィルスやワーム，ボット，トロイの木馬などといった悪意のあるソフトウェア（マルウェア）による被害が深刻となっている．マルウェアへの対策として，解析対象のマルウェア検体を解析環境（サンドボックス）内で実行し，その挙動を観測するマルウェア動的解析の研究が広く行われている[2, 7, 9, 10, 14-20, 22, 28, 29]．また，インターネット上でユーザからの実行ファイル等の検体の投稿を受け付け，自動的に動的解析を行い，解析レポートを投稿ユーザに提供するサービスが多くのセキュリティベンダや研究グループにより公開され，運用されている[16-20, 22, 24, 28, 29, 33]．これらのサービスの多くは Windows 上で動作する実行ファイルを解析対象としているが，JavaScript[33]や，Flash[33]，DLL[22]，PDF[22, 33]，Web サイト[16, 17, 20]の解析を行うサービスも存在する．本稿では，このようなマルウェア動的解析オンラインシステムを Public MSAS（Public Malware Sandbox Analysis System）と呼ぶ．

我々は先行研究[11-13]において攻撃者が特別に設計した検体（これをデコイと呼ぶこととする）をサービスに投稿することで，インターネットに接続されたサンドボックスの IP アドレスを特定し，サンドボックスの検知を行うデコイ挿入を行う攻撃法を指摘し，これに対して実運用中の Public MSAS が脆弱であることを実証実験により示した．さらに，先行研究[8]では，Windows プロダクトキー，MAC アドレス，OS インストール日時といったサンドボックス固有の情報(ホスト情報)をデコイ挿入攻撃により取得し，これらの情報を基にサンドボックス検知を行う攻撃法を指摘し，同様に実運用中の Public MSAS に対して有効であることを示した．

本稿では，これらの攻撃への対策を検討する．まず，ホスト情報に基づくサンドボックス検知に対しては，情報取得に用いる API をフックし，解析を行う毎に異なる偽装値を返す対策や，毎回の解析時に異なるホスト情報を用いる対策を示す．さらに，ホスト情報を取得する動作自体が通常のマルウェアの動作として稀であることに着目し，デコイ挿入攻撃が行われている事実を検出する方法を示す．

一方，IP アドレスに基づく検知においては，サンドボックスが実際に攻撃者のサーバにアクセスすることで行われるため，IP アドレスの偽装が難しく，さらに，マルウェアが遠隔の攻撃者のサーバにアクセスすること自体はボット等で見られる典型的な挙動であるため，デコイ挿入攻撃を検知することが難しい．そこで，プロキシとして検体投稿者のマシンと受付用サーバを利用し，これらを介してインターネット上のホストと通信することで，サンドボックスの IP アドレスを攻撃者から秘匿する対策手法を提案する．また，OpenVPN[25]と UPnP によるポート開放機能[32]を用いた提案手法の実現例を示す．

本稿の構成は以下の通りである．まず 2 章で，前提とする Public MSAS のモデルとデコイ挿入攻撃を説明する．3 章でデコイ挿入攻撃への対策を提案する．4 章で提案対策手法に関する考察を行い，5 章でまとめを述べる．

* 横浜国立大学，〒240-8501 横浜市保土ヶ谷区常盤台 79-7, Yokohama National University, 79-7 Tokiwadai, Hodogaya-ku, Yokohama, 240-8501, Japan.

Vol.2011-DPS-146 No.54
Vol.2011-CSEC-52 No.54
2011/3/11

2. Public MSASとデコイ挿入攻撃

本章では Public MSAS のモデル，Public MSAS の評価項目，Public MSAS に対するデコイ挿入攻撃について説明する．

2.1 Public MSASのモデル

Public MSAS は Windows の実行ファイルなどのファイルを解析対象とするものと，Web サイトを対象とするものに分けられる．本稿では前者を「Public MSAS-F」，後者を「Public MSAS-W」と表記する．投稿ユーザは解析対象のファイルをサービスに投稿し解析を依頼する．受付は解析対象のファイルを受理するために公開されたインターフェイスであり，典型的には Web サイトとして実現される．サンドボックスは解析対象のファイルを実行し挙動を解析するための環境であり，内部ではマルウェアの挙動を把握するために各種ログ（通信ログやレジストリアクセスログ，API 呼び出し履歴など）が取得され，それらをまとめた解析レポートが投稿者に提供される．サンドボックスから外部へ攻撃が流出することを防ぐため，サンドボックスをインターネットから隔離し，代わりに多様なダミーサーバからなる擬似インターネットを用意する方法が従来から行われている．これを隔離型 Public MSAS-F と呼ぶこととする．隔離型 Public MSAS-F のモデルを図 1 に示す．一方，近年では，ボットのように，インターネット上の様々なホストと連携して動作するマルウェアの解析を行うため，サンドボックスをインターネットに接続する，インターネット接続型 Public MSAS-F が主流となっている．図 2 にインターネット接続型 Public MSAS-F のモデルを示す．

一方， Public MSAS-W では，投稿者は解析対象の URL を受付に投稿し解析を依頼する．解析対象の URL を受け取ると，サンドボックスは実際にその URL のコンテンツを取得し，解析を行う．解析対象の Web サイトからコンテンツ取得を行うため，Public MSAS-W は通常インターネット接続型である．

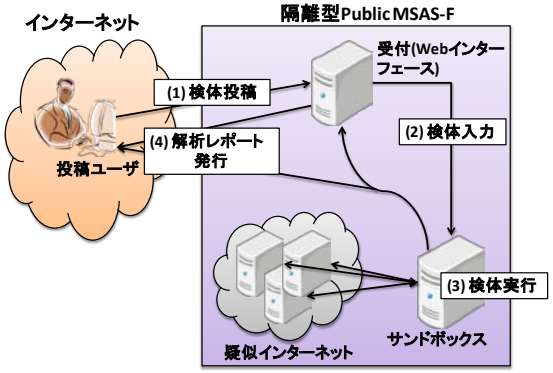

図 1　隔離型 Public MSAS-F のモデル図

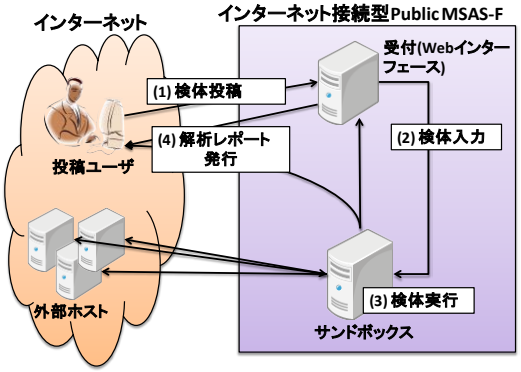

図 2　インターネット接続型 Public MSAS-F のモデル図

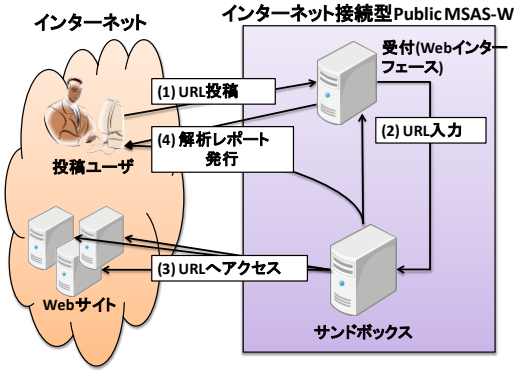

図 3　インターネット接続型 Public MSAS-W のモデル図

2.2 Public MSASの評価項目

マルウェア動的解析システムの評価項目として，文献[14]では以下の 3 つの項目が挙げられている．これらの 3 項目は Public MSAS の評価項目として適用できる．

① **Observability**：動的解析によってマルウェアの様々な挙動を観測できる性質に関する評価項目である．近年のマルウェアにはデバッガや仮想化システムを検知することで解析の回避を試みるものが存在する[3]．これらのマルウェアは解析環境を検知して実行停止したり，異なる挙動を示したりするため，マルウェアの本来の挙動の観測が難しくなっている．サンドボックスの IP アドレスを特定することによるサンドボックス検知[11]や，Windows プロダクトキー，MAC アドレス，OS インストール日時といったサンドボックス固有のホスト情報を用いてサンドボックスを検知する攻撃[8]などによっても，Observability が低下する．

Vol.2011-DPS-146 No.54
Vol.2011-CSEC-52 No.54
2011/3/11

② **Containment**：解析環境自体がマルウェアに感染したり，解析環境の外部に攻撃が流出することなく，安全に解析を行えるかどうかの指標を Containment と呼ぶ．サンドボックスを外部のネットワークから完全に隔離する方法は高い Containment を有するが，マルウェアの外部ホストとのやり取りを解析環境内で完全に再現することは難しいため，Observability が低下することが一般的であり，トレードオフの関係にある．

③ **Efficiency**：マルウェアの挙動を安定的かつ効率的に観測できるかどうかの指標を Efficiency と呼ぶ．解析に要する時間や解析自動化の可能性も Efficiency の重要な評価指標である．

2.3 デコイ挿入攻撃

本節では，文献[8, 11-13]で示されたデコイ挿入攻撃を説明する．デコイ挿入攻撃はサンドボックス検知の際に用いる情報によって「ホスト情報に基づくサンドボックス検知」と「IP アドレスに基づくサンドボックス検知」の二つに分けられる．以下，それぞれの検知方式について述べる．

2.3.1 ホスト情報に基づくサンドボックス検知

ホスト情報に基づくサンドボックス検知では，まず攻撃者は Public MSAS に対してデコイを挿入（投稿）し，サンドボックス情報の暴露を試みる．ここでの攻撃者の目的は，デコイを挿入することで対象の Public MSAS を特定するためのサンドボックス情報を取得し，デコイから攻撃者にその情報を伝えさせる（暴露させる）ことである．先行研究[8, 11]では情報の暴露方法として，

- ネットワーク経由
- 解析レポート経由

の 2 通りが示されている．ネットワーク経由での暴露方法[11]では，攻撃者は自身で管理可能なサーバを用意し，投稿したデコイとサーバ間の通信によってサンドボックス情報の暴露を行う．当然ながら，隔離型 Public MSAS-F ではネットワーク経由の暴露方法は適用できない．一方，解析レポートを経由した暴露方法[8]では Public MSAS がユーザに提供する解析レポートに注目し，取得した情報を解析レポートに埋め込むことで暴露を行う．

以下に Public MSAS-F と Public MSAS-W からのサンドボックス情報暴露の手順をそれぞれ述べる．

Public MSAS-F の場合　Public MSAS-F からのサンドボックス情報暴露では，攻撃者はまず，デコイ（実行ファイル）を用意し，サービスへ投稿する．投稿されたデコイは，実行されるとサンドボックスの情報を取得し，それをネットワーク経由または解析レポート経由で攻撃者に暴露する．文献[8]では Windows プロダクトキー，MAC アドレス，OS のインストール日時などの情報が暴露対象の情報となっている．図 4 に Public MSAS-F からのサンドボックス情報暴露の手順を示す．

Public MSAS-W の場合　一方，Public MSAS-W に対しては，攻撃者はまず囮 Web サーバを用意し，その Web サーバ上のコンテンツに対応したデコイ URL を解析サービスへ投稿する．投稿を受けると Public MSAS-W はデコイ URL が示す囮 web サーバ上のコンテンツにアクセスする．サンドボックスからのアクセスに対して，サンドボックスの Web ブラウザの脆弱性を突く攻撃コードを送信することで，デコイ（実行ファイル）をダウンロード・実行させる．実行後の流れは，Public MSAS-F と同様である．また，JavaScript 等のスクリプトを送信することで，サンドボックスで用いられているモニタの解像度やタイムゾーン，ブラウザのプラグインなどといったサンドボックス固有の情報を取得[18]し，それらの情報を基にサンドボックスを特定する方法がある．

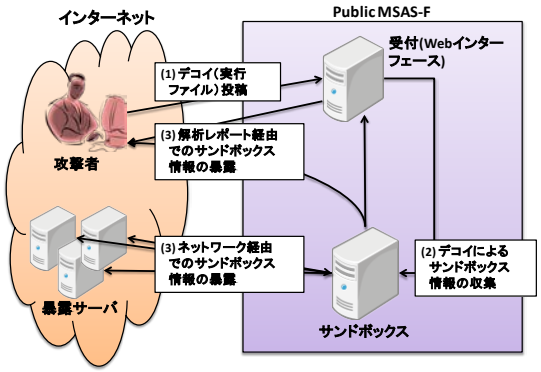

図 4　Public MSAS-F からのサンドボックス情報暴露の手順

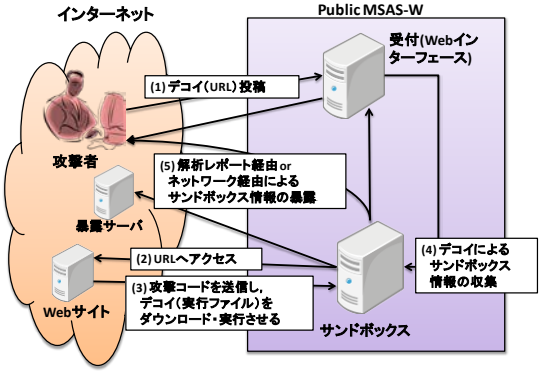

図 5　Public MSAS-W からのサンドボックス情報暴露の手順

次に攻撃者は，収集したサンドボックス情報を基に Public MSAS による解析の検知及び回避を行う．検知手法としては，

- ホストベースの検知
- ネットワークベースの検知

の 2 通りが示されている．ホストベースの検知では，マルウェア自身が検知用サンドボックス情報を保持し，検知を行う．一方ネットワークベースの検知では，マルウェアから情報を受信したサーバ側が検知を行い，マルウェアへの応答を変化させる．ホストベースの検知は隔離型のサンドボックスでも検知ができるという点が攻撃者にとってメリットであり，一方，ネットワークベースの検知は，サンドボックスであるか否かの判定を攻撃者のサーバ上で行うため，判定条件の変更などに柔軟

Vol.2011-DPS-146 No.54
Vol.2011-CSEC-52 No.54
2011/3/11

に対応できる点が攻撃者にとってメリットとなる．

2.3.2 IPアドレスに基づくサンドボックス検知

IP アドレスに基づくサンドボックス検知では，まずは前項のホスト情報に基づくサンドボックス検知と同じく，攻撃者は Public MSAS に対してデコイを挿入（投稿）し，サンドボックスの IP アドレスの暴露を行う．

以下に Public MSAS-F と Public MSAS-W からの IP アドレス情報暴露の手順をそれぞれ述べる．

Public MSAS-F の場合　Public MSAS-F からの IP アドレス暴露では, 攻撃者は同じく，デコイを用意し，サービスへ投稿する．投稿されたデコイは，実行されると暴露サーバと通信を行い，IP アドレスを暴露する[11].

Public MSAS-W の場合　Public MSAS-W に対しては, デコイを実行させるまでの手順は前項の Public MSAS-W の手順と同様である．サンドボックスでデコイが実行されると，上記 Public MSAS-F の場合と同じく，デコイは暴露サーバと通信を行い IP アドレスを暴露する．

暴露した IP アドレスを基にサンドボックスを検知，解析回避を行う際は，前項で述べたネットワークベースの検知により解析回避を行う．

3. デコイ挿入攻撃への対策

本節ではデコイ挿入攻撃への対策方法を提案する．まず，3.1 節では，サンドボックスのホスト情報に基づく検知への対策を示す．次に 3.2 節では，IP アドレスに基づく検知への対策について検討し，3.3 節と 3.4 節で投稿ユーザと受付をプロキシとして用いることで，攻撃者から IP アドレスを秘匿する具体的手法を提案する．

3.1 ホスト情報に基づく検知への対策

2.3.1 節に示した通り，ホスト情報に基づくサンドボックス検知の手順は (1)デコイによるホスト情報の取得 (2)ホスト情報の暴露 (3)ホスト情報によるサンドボックス検知，となる．そこで各手順における対策を考える．

まず，(1)デコイによるホスト情報の取得への対策としては，ホスト情報を偽装する方法が考えられる．例えば，Windows プロダクトキーや OS インストール情報を取得する際には，通常 API が用いられるため，これらをフック[5]し，戻り値を変更することでホスト情報を偽装できる．このような対策を行う際は，偽装により，デコイではないマルウェアの解析に与える影響についても考慮する必要がある．また，デコイではない通常のマルウェアがこれらの API を用いることが稀な場合には，当該 API の使用を監視することでデコイの疑いが高い検体を検出できる．類似の対策として，変更が可能なホスト情報については，解析を行う前に毎回変更する対策が考えられる．

次に(2)ホスト情報の暴露への対策として，ネットワーク経由の暴露と解析レポート経由の暴露の場合についてそれぞれ検討する．ネットワーク経由の暴露を防ぐ方法として暴露の可能性のある通信をフィルタリングする方法や，さらに極端な対策として隔離型のサンドボックスを用いる方法が考えられる．解析レポート経由の暴露についても，解析レポートに含まれる情報を制限したり，解析レポートを提供しないことで暴露を防ぐ方法が考えられる．例えば，検体がレジストリの作成を行った場合，どのレジストリにどのような値を追加したのかという詳細な情報はレポートに掲載せず，レジストリの種類などの限定された情報のみを提供する方法が考えられる．上記の対策はいずれも，Public MSAS の Observability を大きく低下させる点が欠点である．

最後に(3)ホスト情報によるサンドボックス検知への対策としては, (1)デコイによるホスト情報の取得への対策と同様に, ホスト情報の偽装や変更が考えられる. 加えて，サンドボックス検知機能を有する検体を検知する方法として，Public MSAS で通常用いているサンドボックスとは異なるサンドボックスを用意し，両方の環境で解析を行い，その差異からサンドボックス検知機能の有無を判定する対策が考えられる．

3.2 IPアドレスに基づく検知への対策

2.3.2 節に示した通り，サンドボックスの IP アドレスは攻撃者が事前に用意した暴露サーバ側で特定されるため，ホスト情報のようにサンドボックス側で簡単に偽装することはできない．また，IP アドレスを暴露するためにデコイが行う動作は暴露サーバへの接続のみであり，この通信を C&C 等と区別することは困難である．したがって，唯一の対策は，暴露サーバを含む外部ホストに接続する際に用いる IP アドレスを十分な頻度で変更することである．以下ではその具体的方法として(1)複数の固定 IP アドレスを用いる方法 (2)商用 ISP を用いる方法 (3)匿名通信路を用いる方法をそれぞれ示し，その問題点を説明する．

まず，最も基本的な対策である(1)複数の固定 IP アドレスを用いる方法について検討する．文献[11]の実証実験でも，実運用中の Public MSAS において複数の IP アドレスを用いるシステムが確認されている．しかしながら，Public MSAS を運営する組織が IP アドレスを用意するコストに比べて, 攻撃者がデコイ挿入攻撃を実施するコストは小さいため, 用意された IP アドレスを暴露するのに十分な数のデコイを挿入することで攻撃者は用意された IP アドレス情報を取得できると思われる. なお, Public MSAS の使用を有料にすることでデコイ挿入攻撃のコストを増加させることができるが，有料化は Public MSAS の人気を大きく低下させ，Public MSAS の目的の 1 つである検体収集の能力は大きく低下するものと思われる．

次に，(2)商用 ISP 回線を用いる方法について説明する．多くの商用 ISP では IP アドレスの効率的な利用のため, 契約回線上の各ホストに対して動的に IP アドレスを割り当てている．さらに，PPPoE セッションの再確立など，一定の初期化作業により割当アドレスをユーザ側から強制的に変更できる場合がある[12, 13]．これを利用し，解析を行う度に ISP から異なる IP アドレスの割当を受け，解析を行うことで IP アドレスに基づく検知を防ぐ方法が考えられる. 少ないコストで多数の IP アドレスを利用できる点で当該方法は有効であるが, IP アドレスの割当方法が ISP に依存しており, Public MSAS 側から制御できない点， ISP 側でフィルタリングなどを行っている場合に解析結果に影響が出る可能性がある点, 頻繁に IP アドレスの再割り当てを行うことにより ISP 側へ与える負荷，割り当てられた IP アドレスの DNS 逆引きにより得られるドメイン名の関連性から検知される可能性がある点などが問題である．

次に，(3)匿名通信路を用いる方法について説明する．これは，サンドボックスが外部ホストと通信を行う際，Tor[31]のような匿名通信路を用いることで，サンドボックスの IP アドレスを秘匿する方法である. 匿名通信路によりサンドボックスは通信相手に IP アドレスを知られずに通信を行うことができる. しかし匿名通信路を利用している事実は通信相手に知られる可能性がある．実際に，Tor を構成する Onion ルータの IP アドレスの一部はリスト化されてインターネット上で公開されている[23]．通常，マルウェアの被害にあうような一般ユーザが Tor を利用しているとは考えにくいため, Tor を利用しているという事実から攻撃者に解析環境であることが検知される可能性がある．

このように IP アドレスに基づく検知は効果的で実現可能性の高い対策がこれまで

Vol.2011-DPS-146 No.54
Vol.2011-CSEC-52 No.54
2011/3/11

示されていなかった．そこで，本稿では，投稿ユーザをプロキシとして用いることでサンドボックスの IP アドレスを攻撃者から秘匿する手法を提案する．

3.3 投稿ユーザをプロキシとして用いるPublic MSASの提案

本節では投稿ユーザをプロキシとして用いる Public MSAS を提案する．以下に提案手法の概念図を示す．

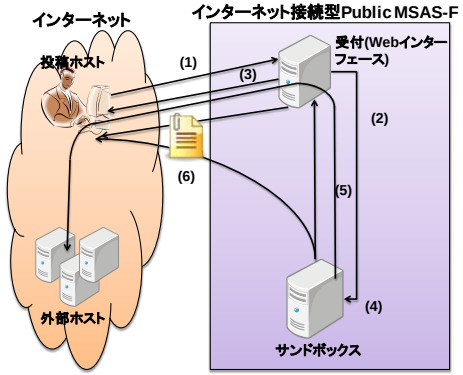

図 6　提案手法概念図

図 6 中の各コンポーネントの機能・要件は次のとおりである．

投稿ホスト：投稿ホストは投稿ユーザのマシンであり，マルウェア検体を受付に投稿し，解析終了後に解析レポートを受信する．また，投稿した検体がサンドボックス内で実行される際にはプロキシとして働き，マルウェアがインターネット側へ行う通信を中継する役割を持つ．インターネット上の外部ホストからは，マルウェアの行う通信は投稿ホストから発信されているように見える．プロキシ機能を実現するためのツール(プロキシツール)は受付からダウンロードしインストールする．近年のマルウェアの中にはサーバとして振る舞い，外部ホストからの接続を待ち受けるタイプが存在するため[15]，サンドボックスから開始されるセッションだけでなく，外部ホストからサンドボックスに対して確立されるセッションも適切に転送する必要がある．また，投稿ホストはサンドボックスからのプロキシ接続を受け付けられるようにファイアウォールなどの設定を適切に行う必要がある．

受付：受付は投稿ホストからマルウェア検体を受け付け，サンドボックスへマルウェア検体を入力する．サンドボックスでの解析が始まると，マルウェアが行う通信は受付を経由して投稿ホストのプロキシまで届けられる．一方，外部ホストから投稿ホストに届いた通信は受付を介して，サンドボックスに送られる．解析終了時には解析結果である各種ログ情報をサンドボックスから受け取り投稿ホストへ渡す．

サンドボックス：サンドボックスでは投稿されたマルウェアが実行され，動的解析が行われる．サンドボックス内部で実行されたマルウェアによる通信は受付を経由し，投稿ユーザホストのプロキシを通じてインターネット上のホストへ届けられる．解析終了後には解析結果をユーザまたは受付に出力する．

以下に投稿ホストからマルウェア検体が投稿されてから投稿ホストに解析レポートが出力されるまでのイベントを時系列順に説明する．なお，以下のステップ(1)〜(6)は図 6 内の番号と対応している．また，投稿ホストが初めて当該 Public MSAS を利用するときに限り，プロキシツールのダウンロードとインストールが必要であるが，以下では，プロキシツールは既にインストールされているものとする．

(1)　投稿ホストはマルウェア検体を受付に投稿する．受付は検体を受理すると，当該投稿ホスト用のプロキシ設定情報を返信する．投稿ホストは受信したプロキシ設定情報をプロキシツールに反映する．

(2)　受付は投稿された検体をサンドボックスへ入力する．

(3)　検体の実行が始まる直前に，受付から投稿ホスト内のプロキシツールへ起動指示が送信される．プロキシツールが起動すると，サンドボックスと投稿ユーザの間でプロキシのセッションが確立される．

(4)　サンドボックス内で検体が実行される．

(5)　サンドボックス内で実行された検体が外部ホストに対して行う通信は，受付，投稿ホストを介してインターネット上の外部ホストへと到達する．逆に，外部ホストから投稿ホストに届いた通信は，受付を介してサンドボックスに送られる．

(6)　検体の実行が終了したのち，プロキシツール停止指示と解析結果がユーザのホストへ送信される．

なお，上では Public MSAS-F を例に説明を行っているが，Public MSAS-W においてもマルウェアの通信を投稿 URL へのアクセスと読み替えることで本手法の適用が可能である．

3.4 投稿ユーザをプロキシとして用いるPublic MSASの実現例

本節では，3.3 節で提案した手法の実現例を示す．図 7 は提案手法の全体図である．なお，本実装例は実際に動作確認を行い，設計通りに動作することを確認している．

以下，図 7 の各構成要素について述べる．

投稿ホスト：投稿ホストは Windows OS を想定している．投稿ホストで動作するプロキシは OpenVPN[25]を用いる．投稿ホスト上では OpenVPN はサーバとして動作する．3.3 節で述べたように，サンドボックス内で実行された検体が行う通信は受付を経由し，投稿ホストをプロキシとして通過した後，外部ホストへと送信される．これらの一連の通信が正しく行えるように，まず投稿ホストのネットワーク環境を正しく設定する必要がある．例えば，商用 ISP 回線を用いてインターネットに接続する環境では，

Vol.2011-DPS-146 No.54
Vol.2011-CSEC-52 No.54
2011/3/11

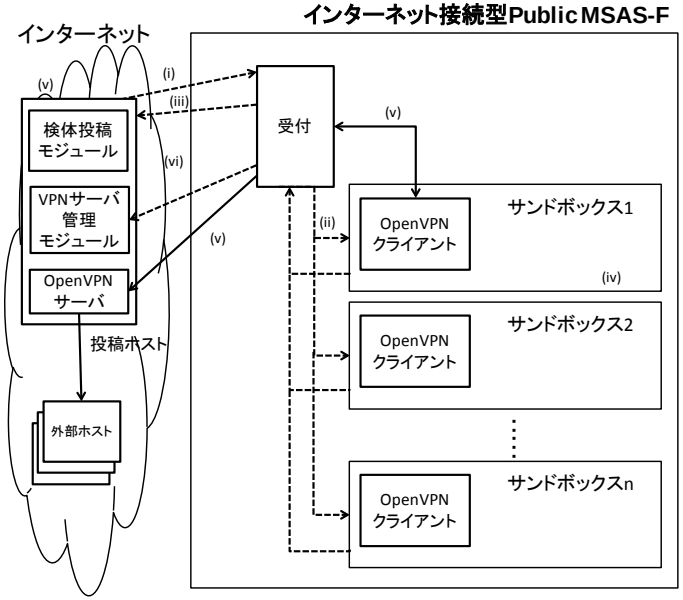

図 7　提案手法全体図

投稿ホストの上流にブロードバンドルータなどのネットワーク機器が設置され，パケットフィルタリングが行われている可能性がある．このような環境では Public MSAS からの VPN 接続が投稿ホストへと到達できない，そのため，プロキシツールには，UPnP[27]を用いて OpenVPN が使用するポートの開放・ポートフォワーディングの設定を行う機能を実装する．プロキシツールは以下の構成要素からなる．

a)　検体投稿モジュール
b)　VPN サーバ管理モジュール
c)　OpenVPN サーバ

検体投稿モジュールは，ユーザからの指示に従い，解析対象の検体を受付に送信する．VPN サーバ管理モジュールは検体の投稿と同時に起動され，受付からの OpenVPN サーバ起動指示を待ち受ける．また，受付から OpenVPN サーバ終了指示を受信した際は OpenVPN サーバを終了した後，自身を停止させる．

次に，受付より提供されるプロキシ設定情報の内容について述べる．プロキシ設定情報は以下の要素により構成される．

I.　OpenVPN 設定ファイル
II.　VPN サーバ管理モジュール設定ファイル

OpenVPN 設定ファイルには OpenVPN サーバが待ち受けるポートや VPN セッション確立の際に必要となる ID・パスワードなどの情報が記載されている．VPN サーバ管理モジュール設定ファイルには VPN サーバ管理モジュールが待ち受けるポート，管理モジュールに指示を送信する際に使用する ID・パスワードなどの情報が記載されている．この際，セキュリティ上の理由から，ID・パスワード，ポートなどの項目は，検体を投稿する度に受付が作成し投稿ユーザに提供する．この際，投稿ユーザのホスト上で動作しているその他のサービスと重複することが無いよう，ポート番号の選定には注意が必要である．

受付：受付は投稿ホストに対して，プロキシツール，プロキシ設定情報の提供を行う．この際，受付は各投稿ユーザの IP アドレスと送付した OpenVPN の ID・パスワードの対応を記憶し，サンドボックスに入力する．さらに，投稿ホスト上の VPN サーバ管理モジュールにアクセスし，OpenVPN の起動及び停止指示を行う．

サンドボックス：サンドボックスは検体が実行される環境であり，多くの場合は Windows OS が動作している．サンドボックスの動的解析環境としての具体的構成の説明は本稿では割愛し，提案手法に特化した機能のみを説明する．サンドボックスの具体的構成については文献[14]を参照頂きたい．まず，サンドボックスは，受付から検体，投稿ホストの IP アドレス，ID・パスワードを受け取る．サンドボックス内では OpenVPN クライアントが動作しており，受付より受け取った IP アドレスに対して，ID・パスワードを用いてアクセスし，VPN セッションを確立する．その後，検体を実行し，その挙動を解析する．あらかじめ定められた時間が経過すると，検体の観測結果である各種ログを外部に出力し，終了する．

以下に，提案手法による Public MSAS-F が検体投稿を受けユーザに解析レポートを出力するまでの流れを示す．各項目は図 7 の数字と対応している．なお，初回に限り投稿ホストは，プロキシツールをインストールする必要がある．この際，投稿ホストがインターネットに接続しているインターフェイスのインターネット接続の共有を有効化し，OpenVPN サーバが用いる仮想インターフェイスから共有できるようにする．

i.　投稿ホストはマルウェア検体を受付に投稿する．検体を受信すると，受付は当該投稿ホスト用のプロキシ設定情報を生成する．投稿ホストはプロキシ設定情報を受付から受信した後，VPN サーバ管理モジュールを起動する．投稿ホストの上流にルータが存在する場合は，VPN サーバ管理モジュールは，UPnP により当該ルータにアクセスし，自身が受付との通信に使用するポートを開放する．
ii.　受付は投稿された検体をサンドボックスへ入力する．
iii.　受付から投稿ホスト上の VPN サーバ管理モジュールへ OpenVPN サーバの起動指示が送信される．投稿ホストで OpenVPN サーバが起動されると，VPN サーバ管理モジュールは UPnP により上流のルータにアクセスし，OpenVPN が使用するポートを開放する．OpenVPN サーバの起動後，サンドボックス内の OpenVPN クライアントより VPN セッションが確立される．この際，i で生成された ID とパスワードを用いて認証が行われる．
iv.　サンドボックス内で検体が実行される．
v.　サンドボックス内で実行された検体が行う通信は受付を経由して投稿ホストからインターネット上の外部ホストへと送信される．
vi.　サンドボックス内でのマルウェアの実行が終了した後，受付よりユーザの VPN サーバ管理モジュールへ OpenVPN サーバの終了指示が送信される．VPN サーバ管理

モジュールは OpenVPN サーバの終了を確認した後，UPnP により開放されたポートを閉じ自身を終了させる．同時に受付は解析レポートをユーザへ提供する．

4. 考察

本章では，まず 3.3 節で説明した提案方式の有効性について考察する．提案手法は投稿ホストをプロキシとして用いているため，外部ホストからサンドボックスの IP アドレスを知ることは原理的にできない．したがって，サンドボックスの IP アドレスを秘匿するという観点では 3.2 節で説明した匿名通信路を用いる方法と同程度の効果が期待できる．さらに前述の通り，Tor をはじめとする匿名通信路は送信者の匿名性を保証することを目的としており，Tor ノードの検出[30]には十分な対策が取られていないことから，攻撃者は Tor による通信をサンドボックスからの通信であると判断することで，解析を回避できると思われる．これに対して，提案方式は透過的に働くプロキシを適用するため，解析の事実を攻撃者に察知されにくい点が特長といえる．

一方，提案方式では，検体から外部ホストに攻撃通信が発生した場合には，外部ホストから容易に投稿ホストの IP アドレスが確認できるため，攻撃の責任を問われることを懸念するユーザがいるものと思われる．しかしながら，実運用中の Public MSAS では使用上の合意事項として，外部ホストへの攻撃の責任は投稿者にあることを明記している場合も多く，アクセスした IP アドレスに関わらず責任が発生するものと思われる．また，外部ホスト向け通信は実運用中のサンドボックスにおいてはフィルタリングが行われており，脆弱なポートへの通信は遮断されたり，サンドボックス内部のダミーサーバへ転送されるようになっているため，外部ホストへの攻撃のリスクは軽減されている．さらに，提案手法を実際に運用する際には，ユーザが隔離型で利用するか自身のホストをプロキシとし，インターネット接続型で利用するかを選択できるようにすることもできる．

次に通信の遅延時間に基づく，提案方式に対する高度な検知手法の可能性について述べる．

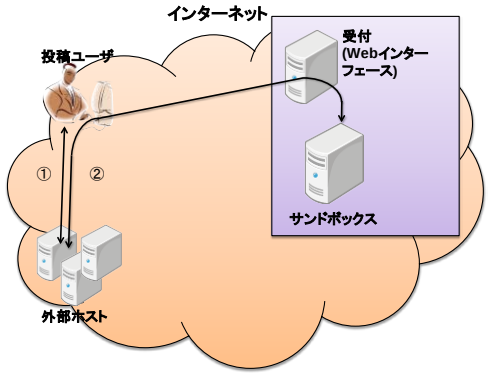

図 8　プロキシを用いた場合の遅延

図 8 の経路①は外部ホストと投稿ユーザが直接通信を行う場合，すなわち，従来の public MSAS における通信の経路である．一方，経路②は提案手法における通信の経路である．このように，経路②では，投稿ユーザとサンドボックス間の分だけ経路①と比較して余分な遅延が発生することとなる．この遅延により攻撃者がサンドボックスを検知するシナリオとして以下が考えられる．

まず，前提として，攻撃者は自らのもつIPアドレスとインターネット上の任意のIPアドレスとの間の通信の遅延時間を計測可能であるとする 1．このとき，経路①と経路②の遅延の差が，これらの経路上のネットワーク状態の変化および通信相手の通信処理能力の違いにより発生しうる誤差と比較して十分に大きい場合，攻撃者は経路①と経路②を区別できる可能性がある．このような攻撃の実現可能性に関する詳細な検討は今後の課題とする．

また，従来の Public MSAS ではユーザホストに特別なツールなどは必要なく，Web ブラウザがインストールされていればサービスを利用できた．しかし，3.4 節で示した実現例ではユーザホストに OpenVPN をインストールし，ネットワークの初期設定を行う必要があり，ユーザに対する負担である．しかし，OpenVPN のインストール自体はインストーラが配布されており，またインストール時にはデフォルト設定からの変更は特に必要ないため，一般的なインストーラ付属のアプリケーションをインストールできるユーザならば導入は容易であるといえる．ネットワークの初期設定についても 1 回だけ行えばよいため，比較的負担は少ないが，自動化によりユーザの負担をさらに削減できると思われる．

最後に，3.4 節の実現例における各要素については，実装と動作確認を行っており，設計通りに動作することを確認している．

5. まとめ

本稿では，我々が先行研究で指摘してきた Public MSAS の脆弱性の対策を提案し，特にサンドボックスの IP アドレスを特定する攻撃への対策を施した Public MSAS について具体的な実現方法を示した．

今後の課題としては，4 章で考察した通信の遅延によるサンドボックス検知の検証や，サンドボックスのホスト情報を用いた検知に対する具体的な対策の検討が挙げられる．

参考文献

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

2) U. Bayer, C. Kruegel, and E. Kirda, "TTAnalyzer: A Tool for Analyzing Malware," 15th Annual Conference of theEuropean Institute for Computer Antivirus Research (EICAR), 2006.

3) X. Chen, J. Andersen, Z. M. Mao, M. Bailey, J. Nazario, "Towards an Understanding of Anti-virtualization and AntidebuggingBehavior in Modern Malware," The 38th Annual IEEE/IFIP International Conference on DependableSystems and Networks (DSN 2008), 2008.

4) Roger Dingledine, Nick Mathewson, and Paul Syverson "Tor: The Second-Generation Onion Router" Proc. 13thUSENIX Security Symposium, 2004.

1 実際には，同一 ISP 内のホストなどネットワークトポロジ的に近いホストとの通信遅延をリファレンス情報として利用したり，GeoIP[23]のように IP アドレスから地理情報を得るサービスや AS トポロジ情報を利用して通信遅延を推測する攻撃が想定される．

5) H. Father, "Hooking Windows API . Technics of Hooking API Functions on Windows," CodeBreakers Journal, Vol. 1, No. 2, 2004.
6) P. Gill, Y. Ganjali, B. Wong and D. Lie, "Dude, where's that IP? Circumventing measurement-based IP geolocation, " 19th USENIX Security Symposium, 2010.
7) D. Inoue, K. Yoshioka, M. Eto, Y. Hoshizawa, and K. Nakao, "Automated Malware Analysis System and its Sandbox for Revealing Malware's Internal and External Activities," IEICE Trans. Vol. E92D, No. 5, pp. 945 -954, 2009.
8) 笠間 貴弘, 織井 達憲, 吉岡 克成, 松本 勉, "マルウェア動的解析オンラインサービスの脆弱性（その 2）, " コンピュータセキュリティシンポジウム 2010 (CSS 2010), 2010.
9) S. Miwa, T. Miyachi, M. Eto, M. Yoshizumi, and Y. Shinoda, "Design and Implementation of an Isolated Sandbox with Mimetic Internet Used to Analyze Malwares," Proc. DETER Community Workshop on Cyber Security Experimentation and Test, 2007, pp. 6 -6, 2007
10) C. Willems, T. Holz, and F. Freiling, "Toward Automated Dynamic Malware Analysis Using CWSandbox," Security& Privacy Magazine, IEEE, Volume 5, Issue 2, pp. 32 -39, 2007.
11) K. Yoshioka, Y. Hosobuchi, T. Orii, and T. Matsumoto, "Vulnerability of Malware Sandbox Analysis as an OnlineService," IPSJ (Information Processing Society of Japan)Computer Security Symposium 2009, Session F5-1, Oct. 2009.
12) K. Yoshioka, Y. Hosobuchi, T. Orii, and T. Matsumoto, "Vulnerability in Public Malware Sandbox Analysis Systems," 10th Annual International Symposium on Applications and the Internet, SAINT2010, pp. 265 - 268, 2010.
13) K. Yoshioka, Y. Hosobuchi, T. Orii, and T. Matsumoto, "Your Sandbox is Blinded: Impact of Decoy Injection to Public Malware Analysis Systems," IPSJ Journal, March, 2011 (accepted).
14) K. Yoshioka and T. Matsumoto, "Multi-pass Malware Sandbox Analysis with Controlled Internet Connection," IEICE Trans. E93A No. 1, pp. 210-218, 2010.
15) 吉岡克成, 村上洸介, 松本勉, "マルウェア感染ホスト検出のためのネットワークスキャン手法と検出用シグネチャの自動生成" 情報処理学会論文誌 Vol.51, No 9, pp. 1633 - 1644, 2010.
16) agues, http://www.aguse.jp
17) Anubis, http://analysis.seclab.tuwien.ac.at/.
18) Comodo Instant Malware Analysis, http://camas.comodo.com/cgi-bin/submit
19) CWSandbox, http://www.cwsandbox.org/
20) gred, https://www.gred.jp/?tab=goleo
21) Peter Eckersley, "How Unique is Your Web Browser?,", https://panopticlick.eff.org/browser-uniqueness.pdf
22) Joebox, http://www.joebox.org/.
23) MaxMind - GeoIP, http://www.maxmind.com/app/ip-location
24) Norman Sandbox, http://www.norman.com/technology/norman_sandbox/
25) OpenVPN, http://openvpn.net/
26) Proxy.org -Tor servers -Tor IPList, http://proxy.org/tor.shtml
27) SOCKS Protocol Version 5, ftp://ftp.rfc-editor.org/in-notes/rfc1928.txt
28) Sunbelt CWSandbox, Malware Research Labs, http://www.sunbeltsecurity.com/
29) Threat Experts, http://www.threatexpert.com/
30) Tor or not Tor: How to tell if someone is coming from a Tor exit node, in PHP, http://www.irongeek.com/i.php?page=security/detect-tor-exit-node-in-php
31) Tor Project, http://www.torproject.org/
32) UPnP Forum, http://www.upnp.org/
33) Wepawet, http://wepawet.iseclab.org/