2005－ＣＳＥＣ－30（３）
2005／7／21

OS混在環境におけるマルウェア検疫方式の検討

馬場 達也　藤本 浩　角 将高　稲田 勉

株式会社NTTデータ 技術開発本部
〒104-0033 東京都中央区新川 1-21-2 茅場町タワー
E-mail: {babatt, fujimotohr, kadom, inadatt}@nttdata.co.jp

あらまし 現在、ワームやスパイウェアなどのマルウェアの感染による被害が大きな問題となっている。そこで、PCを企業ネットワークに接続する際に、そのPCがマルウェアに感染していないことをチェックする検疫技術が必要とされている。しかし、クライアントOSとしてWindowsを対象とした技術は存在するが、Mac OSやLinuxなどの非Windows OSが混在した環境への対応は十分であるとはいえない。本稿では、Windows以外のOSが混在した環境でも適切な検疫を行う方式について検討した結果を報告する。

キーワード 検疫ネットワーク，マルウェア，ワーム，スパイウェア，VLAN

A Study on a Method for Malware Quarantine in a Multi-OS Environment

Tatsuya BABA　Hiroshi FUJIMOTO　Masataka KADO　and　Tsutomu INADA

Research and Development Headquarters, NTT Data Corporation
Kayabacho Tower, 1-21-2, Shinkawa, Chuo-ku, Tokyo, 104-0033 Japan
E-mail: {babatt, fujimotohr, kadom, inadatt}@nttdata.co.jp

Abstract Recently, infections of malware such as Internet worms and spyware are becoming a serious problem. To prevent damage from these malware, there are "quarantine systems" that check the installed anti-virus software and the applied security patches on the client PCs when they are connected to the enterprise network. They have a problem, however, that almost all existing quarantine systems work on Microsoft Windows systems only. In this paper, we propose a network quarantine system which has functionalities such as malware quarantine, extermination, protection on network side without depending on client software in Multi-OS environment.

Keyword Quarantine System, Malware, Internet Worm, Spyware, VLAN

1. はじめに

近年、システムに感染するワームやスパイウェアなどのマルウェアの被害が増加してきている[1]。マルウェアによる被害を防ぐためには、企業のイントラネットにワームが侵入することを防ぐことが重要である。イントラネットへのマルウェアの侵入経路は、図1に示すものがあると考えられる。

インターネットから侵入するマルウェアに対しては、インターネットとイントラネットの境界にファイアウォールやIPS（Intrusion Prevention System：侵入防止システム）、ゲートウェイ型アンチウイルスソフトを導入することで防ぐことができる。しかし、最近は、外部でマルウェアに感染したノートPCなどを、有線LANや無線LAN、リモートアクセスVPN経由などでイントラネットに接続することによって感染が広まる、「持ち込みPC」からの侵入が問題となっている。

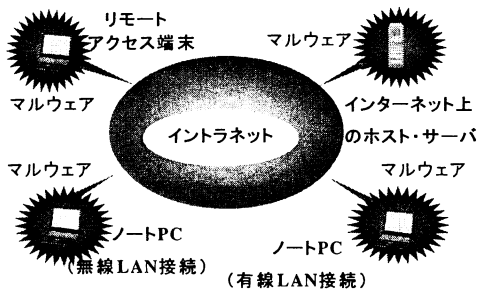

図1 イントラネットへのマルウェアの侵入経路

現在、この持ち込みPCからの侵入を防ぐ技術として、「検疫システム」が注目されている。検疫システムは、イントラネットに接続されるPCをチェックし、マルウェアに感染していないことを確認してから接続を許可するものである。また、同時に接続PCの脆弱

性をチェックすることにより、脆弱性を攻撃して感染するマルウェアがイントラネットに侵入した場合でも、被害が拡大しないように対策を行うこともできる。

2. 従来の検疫システムとその問題点

現在の多くの検疫システムでは、接続 PC 上で、ウイルス対策ソフトが最新のウイルス定義ファイルを使用して適切に動作しているかどうかということや、接続 PC に最新のパッチが適用されているかどうかということをチェックする。そして、これらのチェックの結果、イントラネットに接続するには不適切であると判断した場合には、接続を許可させないというアプローチをとっている。これにより、イントラネットに接続する PC のウイルス対策ソフトのウイルス定義ファイルのアップデートやパッチの適用を徹底させることができ、マルウェアの被害の拡大を防ぐことができる。

しかし、この方式では、クライアント PC に、検疫システム側で対応しているウイルス対策ソフトをインストールしておかなければイントラネットに接続できないという問題がある。例えば、お客様などが持ち込んだ PC を、一時的にイントラネットへ接続させたいような場合が考えられるが、使用しているウイルス対策ソフトが検疫システム側で対応していない場合には接続できないという問題がある。また、マルウェアの中には、自身が検知されたり、駆除されたりしないようにするために、ウイルス対策ソフトなどの機能を無効としてしまうものも多い。このようなマルウェアに感染した場合は、検疫によって隔離されたとしても、その後、ユーザがウイルス対策ソフトの機能を使用して駆除することができないという問題がある。

さらに、パッチを適用すると、動作しなくなるアプリケーションが存在するという問題もある。このようなアプリケーションを使用している PC には、該当するパッチを適用することができないため、イントラネットへの接続が許可されない。

そこで著者らは、ActiveX を使用して、Windows のレジストリの Run エントリやスタートアップフォルダの内容をスキャンし、ワームを含むマルウェアに感染していないかどうかをチェックし、感染していた場合には、駆除を行う方式を提案した[2]。提案した方式では、さらに、パッチの適用状況をチェックし、適用されていないパッチが存在した場合には、その脆弱ポートをネットワーク側で保護しつつ、業務ネットワークに接続させる機能も持っている。提案方式により、クライアント PC で使用しているウイルス対策ソフトの機能に依存せずに検疫を行うことができ、さらに、パッチが適用できない PC でも脆弱性を保護しながらネットワークに接続することができるようになる。しかし、検疫処理に ActiveX を使用しているため、他の検疫システムと同様に、Windows を搭載したクライアント PC のみにしか対応できていないという問題がある。

2004 年 6 月に Google にアクセスした OS を調査した結果によると、Windows 系が 91%、Mac OS 系が 3%、Linux 系が 1%、その他が 5%となっている[3]。現在は、Windows のシェアが圧倒的であるため、マルウェアの攻撃対象は Windows に絞られているが、Mac OS X の人気が上昇し、デスクトップ用の Linux 製品が各社からリリースされるなど、Windows 以外の OS のシェアが増加し、これらの OS もマルウェアの攻撃対象になると思われる。実際に、2004 年 10 月に、Mac OS X で動作するマルウェアである Renepo（別名 Opener）が出現しており、早急な対策が望まれている。

3. マルウェア検疫方式の検討

著者らは、複数の OS が混在した環境での検疫方式について検討した。

3.1. 対象 OS

OS 混在環境における検疫方式を検討するにあたり、表 1 の OS を対象とした。選択の基準は以下のとおりである。

- Windows に加えて、Mac OS および Linux を対象とする
- 持ち込み PC の検疫を目的とすることから、主にデスクトップ用途に使用されるものであること（サーバ用途のものは除く）
- 企業での利用を想定するため、ベンダによる有償サポートが受けられること
- 現在入手可能であるか、比較的最近まで販売されていたものであること

表 1 対象とするデスクトップ OS

Windows 系	Windows XP
Mac OS 系	Mac OS X 10.3（Panther）
	Mac OS X 10.4（Tiger）
Linux 系	Novell Linux Desktop 9
	Red Hat Enterprise Linux WS 4
	Sun Java Desktop System, Release 2 for Linux
	SUSE LINUX Professional 9.3
	Turbolinux 10 Desktop
	Vine Linux 3.1CR

3.2. 検疫方式

それぞれの OS に対して個別の検疫システムを用意し、ユーザに適切なものを選択させるという方式も考えられるが、それでは、それぞれの検疫システムを個別にメンテナンスしなければならず、管理コストが上がってしまう。また、ユーザが検疫時のアクセス先を

OS にあわせて変更する必要があるなどの不便さもある。そのため、単一の検疫システムで検疫対象マシンの OS の種類を自動判別し、それぞれの OS にあった検疫方式を自動適用するシステムとするのが望ましい。

このため、OS 混在環境に対応した検疫システムは、次の機能を持つ必要がある。

- 検疫対象マシンの OS の種類を自動判別する
- OS の種類に応じた検疫処理を行う
- 導入時および運用時のコストを低く抑えるためにクライアントレスで実現する

3.2.1. 動作プラットフォームの選択

クライアント PC に特別なソフトウェアをインストールすることなく検疫処理を行う方式として、従来から ActiveX を使用する方式が使用されている。しかし、ActiveX は Microsoft Windows でしか利用することができない。このため、さまざまなプラットフォームをサポートしている Java Web Start[4]を用いることとした。各 OS における Java への対応状況は表 2 のとおりである。Mac OS X には標準で J2RE（Java 2 Runtime Environment）がインストールされており、Java Web Start が利用できる状態で提供されている。Linux では、Novell Linux Desktop 9 や Sun Java Desktop System, Release 2 for Linux などにおいて標準で J2RE がインストールされている。その他の Linux や Windows には標準ではインストールされていないが、http://java.sun.com/ より無償でダウンロードすることが可能である。また、再配布可能であるため、J2RE がインストールされていない PC は、検疫サーバからダウンロードしてインストールすることも可能である。

表 2 各 OS の Java への対応状況

OS	J2RE バージョン
Windows XP	別途入手
Mac OS X 10.3.7	J2RE 1.4.2_03
Mac OS X 10.4.0	J2RE 1.4.2_07
Novell Linux Desktop 9	J2RE 1.4.2_03
Red Hat Enterprise Linux WS 4	別途入手
Sun Java Desktop System, Release 2 for Linux	J2RE 1.4.2_04
SUSE LINUX Professional 9.3	J2RE 1.5.0_01
Turbolinux 10 Desktop	別途入手
Vine Linux 3.1CR	別途入手

3.2.2. OS の自動判別

検疫サーバ側では、OS の種類に応じた適切な検疫を行うために、最初にクライアント PC の OS の種類を判別する必要がある。これについては、Java で java.lang.System.getProperty メソッドを使用することで、OS 名称(os.name)および OS バージョン(os.version)が取得できる（表 3)。

表 3 System.getProperty メソッドの出力

OS の種類	os.name の出力
Windows XP	"Windows XP"
Mac OS X	"Mac OS X"
Linux	"Linux"

ただし、Linux の場合は、OS 名称として"Linux"、OS バージョンとして、カーネルのバージョンが取得できるのみであり、これでは、ディストリビューション毎の脆弱性をチェックするには不十分である。そこで、「/etc」ディレクトリに格納されている表 4 のリリース情報ファイルの内容を確認することで、ディストリビューション名およびバージョンを取得する。

表 4 各 Linux OS のリリース情報ファイル

Linux ディストリビューション	リリース情報ファイル
Novell Linux Desktop	/etc/novell-release /etc/SuSE-release
Red Hat Linux	/etc/redhat-release
Sun Java Desktop System for Linux	/etc/sun-release /etc/SuSE-release
SUSE LINUX	/etc/SuSE-release
Turbolinux	/etc/turbolinux-release
Vine Linux	/etc/vine-release

3.2.3. マルウェア感染チェック

デスクトップ PC を狙うマルウェアは、常時稼動しているサーバを狙うものと異なり、マシンが再起動してもマルウェアプロセスが起動するように設定を行うという特徴がある。そこで、各 OS の起動時の自動実行設定と、既知のマルウェアが設定する自動実行設定を比較することによって、マルウェア感染を検知する。

比較処理は、クライアント PC 上で行う方法と、検疫サーバ上で行う方法が考えられるが、クライアント PC 上で行うためには、毎回膨大な量のマルウェア情報をクライアント PC に送信しなければならなくなる。このため、各 OS の自動実行設定を検疫サーバに送信し、検疫サーバ上で各マルウェアの自動実行設定の内容と比較することにより検知を行うこととした。

Windows XP の場合

Windows XP では、プログラムを OS 起動時に自動実行させるためには、レジストリの Run エントリと呼ばれる以下の場所に設定を記述する。

```
HKEY_CURRENT_USER¥Software¥Microsoft¥Windows¥CurrentVersion¥Run
HKEY_CURRENT_USER¥Software¥Microsoft¥Windows¥CurrentVersion¥RunOnce
HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion¥Run
HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion¥RunOnce
```

Windows XP には、レジストリをコマンドラインから操作するための reg コマンドが用意されており、これを使用して、レジストリの Run エントリの内容を取得し、検疫サーバに送信する。また、同時に、OS 起動時に自動実行されるスタートアップフォルダの内容を検疫サーバに送信し、マルウェア感染チェックを行う。

Mac OS X の場合

Mac OS X が起動すると、最初に init プロセスが「/etc/rc.boot」、「/etc/rc」、「/etc/rc.common」の各シェルスクリプトを実行する。そして、SystemStarter プロセスが「/System/Library/StartupItems/」および「/Library/StartupItems/」ディレクトリに置かれたシェルスクリプトを実行する。このディレクトリには、OS 起動時に自動実行されるアプリケーションの起動スクリプトなどが置かれる。

また、cron プロセスによって、ある時刻に自動実行される場合がある。そこで、検疫時には、上記のシェルスクリプトの内容およびディレクトリの内容、そして、cron の設定ファイルである crontab の内容を検疫サーバに送信し、マルウェア感染チェックを行う。

Linux の場合

Linux が起動すると、最初に init プロセスが「/etc/inittab」ファイルを参照し、プロセスの起動あるいは終了時の動作を指定する action が「sysinit」、「boot」、「bootwait」、「wait」のいずれかである行に記述されているファイル（Red Hat 系では「/etc/rc.d/rc.sysinit」および「/etc/rc.d/rc」、SUSE 系では「/etc/init.d/boot」および「/etc/init.d/rc」）を実行する。そして、ランレベルに応じて「/etc/rc.d/rc[0-6].d/」ディレクトリに置かれたシェルスクリプトを実行する。このディレクトリには、OS 起動時に自動実行されるアプリケーションの起動スクリプトなどが置かれる。

また、cron プロセスによって、ある時刻に自動実行される場合がある。そこで、検疫時には、上記のシェルスクリプトの内容およびディレクトリの内容、そして、cron の設定ファイルである crontab の内容を検疫サーバに送信し、マルウェア感染チェックを行う。

3.2.4. マルウェア駆除

マルウェアが検知された場合には、マルウェアを駆除する必要がある。具体的には、「マルウェアプロセスの停止」、「マルウェアプログラムファイルの消去」、「自動実行設定の解除」を行う。

Windows XP の場合

Windows XP には、起動中のタスクを参照するための tasklist コマンドと、タスクを停止させるための taskkill コマンドが用意されており、これらを使用して該当マルウェアプロセスを終了する。また、reg コマンドを使用して該当レジストリを削除し、マルウェアプログラムファイルを削除する。ただし、ユーザが Administrator 権限を有している必要がある。

Mac OS X の場合

Java Web Start プログラムから、ps コマンドおよび kill コマンドを外部コマンドとして使用して該当マルウェアプロセスを終了し、該当起動スクリプトとマルウェアプログラムファイルを削除する。しかし、削除には root 権限が必要であるため、sudo コマンドを使用する。

Linux の場合

Java Web Start プログラムから、ps コマンドおよび kill コマンドを外部コマンドとして使用して該当マルウェアプロセスを終了し、該当起動スクリプトとマルウェアプログラムファイルを削除する。しかし、sudo コマンドを使用してファイルを削除する権限をユーザにあらかじめ与えておく必要がある。

3.2.5. 脆弱性チェック

次に、適用されているパッチやパッケージの情報をもとに、脆弱性のチェックを行う。

Windows XP の場合

レジストリに格納されているパッチ情報を検疫サーバに送信し、脆弱性がないかどうかをチェックする。

Mac OS X の場合

Mac OS X の場合は、適用した「Security Update」などの情報を「/Library/Receipts/」ディレクトリに格納している。そこで、このディレクトリの内容を検疫サーバに送信し、脆弱性がないかどうかをチェックする。

Linux の場合

RPM ベースの Linux ディストリビューションの場合は、「rpm -qa」コマンドを発行することで、インストールされているパッケージとそのバージョンをチェックすることができる。現在は、多くの Linux ディストリビューションが rpm コマンドをサポートしており、表 1 にリストアップした Linux ディストリビューションはすべてこのコマンドを使用することが可能である。そこで、このコマンドの出力結果を検疫サーバに送信し、脆弱性がないかどうかをチェックする。

4. マルチ OS 対応マルウェア検疫システム

これまでに検討した検疫方式をもとに、マルチ OS 対応マルウェア検疫システムについて提案する。

4.1. システム構成

図2に提案する方式を実現するためのシステム構成を示す。本システムでは、業務ネットワークを構成する業務用VLANと、検疫ネットワークを構成する検疫用VLANを用意し、レイヤ2レベルで通信を制限している。検疫用VLANには、接続PCのチェックや、VLANおよびフィルタリングの設定を動的に制御する検疫サーバと、PC接続時の認証を行うRADIUSサーバを設置する。さらに、ブリッジファイアウォールをVLAN対応レイヤ2スイッチにVLANトランクで接続する。

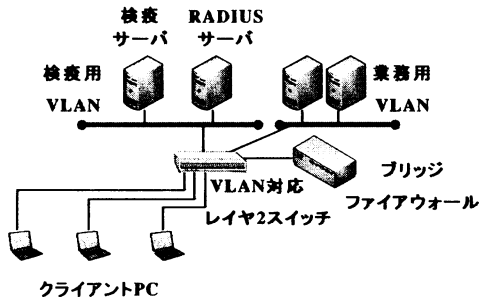

図2 システム構成例

接続PCを業務用VLANに直接接続する場合は、そのPCが接続されているポートに対して業務用VLANと同じVLAN IDを割り当てる。PCをブリッジファイアウォール経由で業務用VLANに接続させる場合には、そのPCが接続されているポートに対して、ポート毎に異なるユニークなVLAN IDを割り当てる。ブリッジファイアウォールは、異なるVLAN間をブリッジ接続する機能を備えているため、この場合は、必ずブリッジファイアウォール経由でアクセスされる。

4.2. 処理の流れ

本方式では、図3のように、「ユーザ認証」「マルウェア感染チェック」「脆弱性チェック」などを行う。

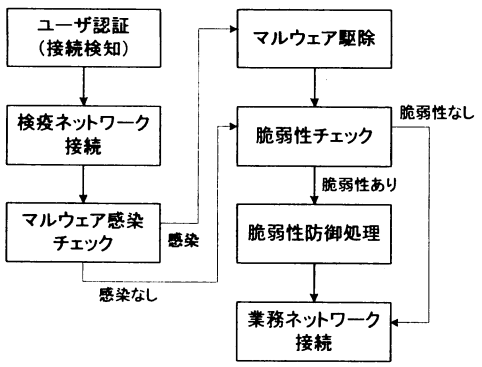

図3 マルウェア検疫処理の流れ

4.3. ユーザ認証/検疫ネットワーク接続処理

レイヤ2スイッチへのPCの接続を検知するために、標準のユーザ認証の仕組みであるIEEE 802.1X[5]を利用する。具体的には、検疫サーバにRADIUSプロキシ機能を持たせ、IEEE 802.1Xで用いられるRADIUS認証を、検疫サーバを経由して行うようにした。これにより、検疫サーバは、RADIUSリクエストメッセージに含まれている「Calling-Station-Id」、「NAS-IP-Address」、「NAS-Port」の各RADIUS属性から、接続PCのMACアドレス、接続先スイッチのIPアドレス、接続スイッチポートの情報を取得する。そして、RADIUS認証中に該当PCが検疫用VLANとのみ通信が可能となるように、ブリッジファイアウォールのフィルタリング設定を行う。検疫サーバは、このフィルタリング設定が完了するまでは、RADIUSサーバからの認証完了メッセージをレイヤ2スイッチへ転送しないように制御する。

IEEE 802.1Xに対応したIP電話などの非PC端末が接続された場合には、EAP-TLSなどの証明書による認証を行うことによって接続端末が非PC端末であることが判明するため、検疫処理をバイパスさせて業務用VLANに直接接続させる。

表5に各OSのIEEE 802.1Xへの対応状況をまとめる。Windows XPやMac OS X 10.3/10.4にはOS標準で提供されており、Linux系OSにはフリーで提供されているXsupplicant[6]を利用することができる。

表5 各OSのIEEE 802.1X対応状況

OS	IEEE 802.1X対応状況
Windows XP	標準装備
Mac OS X 10.3 (Panther)	標準装備
Mac OS X 10.4 (Tiger)	標準装備
Novell Linux Desktop 9	Xsupplicant 0.8bが付属
Red Hat Enterprise Linux 4	Xsupplicantを別途入手
Sun Java Desktop System, Release 2 for Linux	Xsupplicantを別途入手
SUSE LINUX Professional 9.3	Xsupplicant 1.0.1が付属
Turbolinux 10 Desktop	Xsupplicantを別途入手
Vine Linux 3.1CR	Xsupplicantを別途入手

4.4. マルウェア感染チェック/マルウェア駆除処理

認証完了後、ユーザは、検疫サーバに対してWebブラウザを使用してアクセスする。そして、チェック用Java Web Startプログラムをダウンロードし、実行する。

ダウンロードされたJava Web Startプログラムは、PCのOSの自動実行設定の内容を、あらかじめ定義された既知のマルウェアの自動実行設定の内容と比較することで、マルウェア感染の有無を判定する。もし、マルウェアに感染していると判断された場合には、該

当マルウェアプロセスを停止し、マルウェアプログラムを削除する。そして、該当自動実行設定を解除する。

4.5. 脆弱性チェック処理

マルウェア感染チェックが完了すると、脆弱性チェックを行う。具体的には、Java Web Start プログラムが、適用されているパッチやパッケージの情報を検疫サーバに送信し、検疫サーバに保管してある OS 毎の最新のリストと比較することによって、接続 PC に存在する脆弱性と脆弱ポートを割り出す。

4.6. 脆弱性防御/業務ネットワーク接続処理

最新のパッチやパッケージが適用されている場合は、検疫サーバは、該当 PC が接続されているレイヤ 2 スイッチのポートの VLAN ID を業務用 VLAN の ID に SNMP を用いて変更することで、ブリッジファイアウォールを経由せずに、直接業務用 VLAN に接続するようにする。最新のパッチやパッケージが適用されていない場合には、該当 PC が接続されているレイヤ 2 スイッチのポートの VLAN ID を業務用 VLAN とは異なるユニークな ID に SNMP を用いて変更することで、必ずブリッジファイアウォールを経由するようにし、さらに該当 PC の脆弱ポートへのアクセスをフィルタリングするようにブリッジファイアウォールの設定を行うことで、外部からの脆弱ポートを狙った攻撃から保護するようにする。

5. 実際のマルウェアの例

本章では、実際のマルウェアの例を示し、本方式によって検知することが可能であることを示す。

W32.Sasser.C.Worm

Windows に感染する脆弱性悪用型ネットワークワームである W32.Sasser.C.Worm[7]は、次のエントリをレジストリに追加し、Windows 起動時に自動実行されるようにする。

```
[キー名] HKEY_LOCAL_MACHINE¥SOFTWARE
         ¥Microsoft¥Windows¥CurrentVersion¥Run
[値]     "avserve2.exe"="%Windir%¥avserve2.exe"
```

検疫サーバ側で、上記のレジストリエントリを登録しておけば、本方式において Sasser.C を検知し、駆除することが可能である。

W32.Netsky.Q@mm

Windows に感染するマスメーリングワームである W32.Netsky.Q@mm[8]は、次のエントリをレジストリに追加し、Windows 起動時にワームプロセスが自動実行されるようにする。

```
[キー名] HKEY_LOCAL_MACHINE¥SOFTWARE
         ¥Microsoft¥Windows¥CurrentVersion¥Run
[値]     "SysMonXP"="%Windir%¥SysMonXP.exe"
```

検疫サーバ側に上記のレジストリエントリを登録しておけば、本方式において Netsky.Q を検知し、駆除することが可能である。

Keylogger.Cone.Trojan

Windows に感染するキーロガーである Keylogger.Cone.Trojan（別名：PerfectKeyLogger）[9]は、次のエントリをレジストリに追加する。

```
[キー名] HKEY_LOCAL_MACHINE¥SOFTWARE
         ¥Microsoft¥Windows¥CurrentVersion¥Run
[値]     "WIN HOST PROCESS"=
         "%system%¥WIN HOST PROCESS.EXE"
```

検疫サーバ側に上記のレジストリエントリを登録しておけば、本方式において Keylogger.Cone.Trojan を検知し、駆除することが可能である。

MacOS.Renepo.B

Mac OS X に感染するマルウェアである MacOS.Renepo.B[10]は、OS 起動時に自動実行されるように、次のディレクトリとファイルを作成する。

```
[ディレクトリ] /System/Library/StartupItems/opener
[ファイル]     opener, StartupParameters.plist
```

検疫サーバ側に、Renepo が作成するディレクトリとして「/System/Library/StartupItems/opener」を登録しておけば、本方式において Renepo.B を検知し、駆除することが可能である。

Backdoor.Dextenea

Backdoor.Dextenea[11]は、バックドアを開こうとする Linux ベースのトロイの木馬であり、OS 起動時に自動実行されるように、次のファイルを作成する。

```
/etc/rc.d/rc2.d/S80rpcmap
/etc/rc.d/rc3.d/S80rpcmap
/etc/rc.d/rc4.d/S80rpcmap
/etc/rc.d/rc5.d/S80rpcmap
```

検疫サーバ側に、Dextenea が作成する起動スクリプトとして「S80rpcmap」を登録しておけば、本方式において Dextenea を検知し、駆除することが可能である。

Linux.Lion.Worm

Linux.Lion.Worm[12]は、Linux に感染するワームであり、Red Hat 系 Linux に存在する起動スクリプトであ

る「/etc/rc.d/rc.sysinit」に自身を起動させるために以下の行を追加する。

```
# Name Server Cache Daemon..

/usr/sbin/nscd -q
/bin/in.telnetd

##########################################
```

検疫サーバ側に上記の行を登録しておくことで、本方式において Linux.Lion.Worm を検知し、駆除することが可能である。

6. 考察

6.1. 本方式によるマルウェア検疫の有効性

5章で示したとおり、デスクトップPCを狙うほとんどのマルウェアは、OS起動時に自動実行する設定を行う。このため、既知のマルウェアであれば、本方式によって多くのマルウェアを検知することが可能であると考えられる。また、自動実行設定をチェックするためだけであるため、高速に感染をチェックすることが可能であり、さらに、クライアントのプログラムに依存しないため、マルウェアがウイルス対策ソフトを無効化した場合でも駆除などの対処を行うことが可能である。

しかし、マルウェアの中には、プログラム名などをランダムで決定するものもあり、この場合は自動実行設定の内容が一定でないため、対応が難しいという問題がある。このようなマルウェアについては、自動起動設定以外の別の痕跡をもとに検出する必要がある。

6.2. 対応 OS の範囲

本方式では、IEEE 802.1X および Java Web Start という、広く利用されている汎用的な技術を使用しているため、これらの技術に対応した OS であれば、クライアントに特別なソフトウェアをインストールすることなく、検疫を行うことができる。

しかし、Java では、レジストリ操作や、プロセスの停止などの操作を直接行うことができないため、外部コマンドに頼らざるを得ない。しかし、Windows 2000 以前では、レジストリやタスクを操作するための CUI コマンドが用意されていないため、対応が難しいという問題がある。このため、これらの Windows を使用している場合は、従来どおり、ActiveX を使用した検疫プログラムも用意しておき、検疫サーバにアクセスした際に、利用者が使用している OS に応じてどちらかを選択できるようにしておく必要がある。しかし、今後は、これらの OS は減少傾向にあると思われ、逆に Mac OS X やデスクトップ Linux の利用が増加すると予想される。このため、Java Web Start を用いた本方式は将来的には有用であると考えられる。

また、J2RE が標準でインストールされていない OS が多いのも問題のひとつである。このため、検疫サーバから J2RE をダウンロードできるようにして、検疫前に J2RE をインストールしておいてもらうようにする必要がある。

7. まとめ

Windows、Mac OS、Linux などの OS が混在した環境において、接続 PC のマルウェア感染チェック、および、脆弱性チェックを行い、その結果から、ネットワーク機器の VLAN およびブリッジファイアウォールの設定を動的に制御して、マルウェアによる被害から防御するマルウェア検疫方式について提案した。

今後は、プロトタイプを実装し、実際のマルウェアを使用して検知精度などを評価していく予定である。

文　献

[1] “コンピュータウイルスの届出状況 [2005年5月分] について“, 独立行政法人 情報処理推進機構 セキュリティセンター, 2005 年 6 月 6 日, http://www.ipa.go.jp/security/txt/2005/documents/virus-full0506.pdf

[2] 馬場達也, 角将高, 稲田勉, “動的 VLAN 制御による統合ワーム対策システムの提案“, 第 122 回マルチメディア通信と分散処理研究会/第 28 回コンピュータセキュリティ研究会, 情報処理学会研究報告, Vol.2005, No.33, 2005-DPS-122/2005-CSEC-28, pp.43-48, 2005 年 3 月発行.

[3] "Google Zeitgeist - Search patterns, trends, and surprises according to Google - June 2004 Zeitgeist", http://www.google.com/press/zeitgeist/zeitgeist-jun04.html

[4] "Java Web Start Technology", Sun Microsystems, Inc., http://java.sun.com/products/javawebstart/

[5] "Port Based Network Access Control", IEEE 802.1X, Institute of Electrical and Electronics Engineers, Inc., http://www.ieee802.org/1/pages/802.1x.html

[6] "Open1x -- Open Source Implementation of IEEE 802.1x", Open1x Project, http://www.open1x.org/

[7] "W32.Sasser.C.Worm", Symantec Corporation, http://www.symantec.com/region/jp/sarcj/data/w/w32.sasser.c.worm.html

[8] "W32.Netsky.Q@mm", Symantec Corporation, http://www.symantec.com/region/jp/sarcj/data/w/w32.netsky.q@mm.html

[9] "Keylogger.Cone.Trojan", Symantec Corporation, http://securityresponse.symantec.com/avcenter/venc/data/keylogger.cone.trojan.html

[10] "MacOS.Renepo.B", Symantec Corporation, http://www.symantec.com/region/jp/avcenter/venc/data/jp-macos.renepo.b.html

[11] "Backdoor.Dextenea", Symantec Corporation, http://www.symantec.com/region/jp/avcenter/venc/data/jp-backdoor.dextenea.html

[12] "Linux.Lion.Worm", Symantec Corporation, http://www.symantec.com/region/jp/avcenter/venc/data/linux.lion.worm.html