端末からの不正メール送信を防止するための検討

平田 祐二

近年，ボットネットによるスパムメールの大量送信や DDoS 攻撃，情報の奪取などの不正行為が蔓延している．ボットは Herder の命令を受けて初めて活動を開始するために感染していることに気が付きにくいという特徴がある．本研究ではポートの制御を行い，プロセスツリーを監視することによりメーラを呼び出したのが正規なユーザか否かを判別する．正規なユーザと確認できた場合にのみ，ポートを開放しメールの通信を許可する．

Examination to prevent illegal mail sending from terminal

Yuji Hirata

Recently, the misbehavior of a large amount of transmission of the spam mail by bottonet, the DDoS attack, and the seizure of information spreads. Bot has the feature that the nature does not adhere easily to the infection for the activity to begin only after herder instruction is received. It is distinguished whether it is a regular user to have called mailer by controlling the port in this research, and observing the process tree. The port is opened only when it can be confirmed a regular user and the communication of mail is permitted.

1. はじめに

インターネットの発展に伴い，ウィルスの被害が大きな問題となっている．近年では，ボットネットによるスパムメールの大量送信や DDoS(Distributed Denial of Service)攻撃, 情報の奪取などの不正行為が蔓延している．ボットネットとは一種のバックドアを埋め込まれた多数の PC で構成されるネットワークの総称であり, 現在では多くの場合 IRC(Internet Relay Chat)の仕組みを利用して指令を受け制御されている．ボットネットは，従来のワームやウィルスのように自動的に感染を拡大せず, Herder の命令を受けて初めて活動を開始するため，感染していることに気が付きにくいという特徴がある．さらに，Malware の作成や配布の意図が従来の愉快犯的な動機から犯罪組織と結びついた営利目的へと変貌している．

Herder はボットに命令を出すために, 公開されている IRC サーバを使用するか，あらかじめボットに感染させた IRC サーバを使用する．また，ボット自身にサーバの機能を持たせ，IRC サーバとして使用する場合もある．Herder は複数の IRC サーバと接続しているため，仮に一つのサーバを停止できたとしても他のサーバを介して命令を送り続けることができる．またサーバを踏み台にして命令を出すので, Herder を見つけることが困難とされている．

本稿では，ボットが Herder の命令を受けて初めて行動を起こすことに着目し，クライアント側で対策を施す．クライアントからメールが送信さ

れる時に，プロセスツリーを監視することにより正常なメール送信か否かを判断し，ポート制御を行うことによりボットによるスパムメール送信を遮断する方式を提案する．

以降，2 章で既存のボット対策となるアンチウィルスソフトと OP25B について述べる．3 章では提案方式のポート制御とプロセスツリーの監視について述べる．4 章で今後の検討課題を述べる．5 章でまとめる．

2. 既存技術とその課題

2.1. アンチウィルスソフト

ウィルス対策ベンダーなどが提供しているアンチウィルスソフトは，コンピュータウィルスの特徴などを記録したデータファイルとコンピュータでやり取りされるデータを照合し，ボットを取り除く．しかし，定義ファイルに情報のあるボットに対しては問題なく対応できるが，定義ファイルに情報のないボットには対応できない．

ボットはオープンソースになっていることに加えて，知識のないユーザでも容易に改変できるツール類も公開されているため，新種のボットが数多く出回っている．1 時間に 1 件は新種のボットが出現しているという報告もある[1]．また検知されるのを防ぐため定期的にボットがアップデートされるなど，アンチウィルスソフトでは対応しきれないという問題がある．

2.2. OP25B(Outbound Port 25 Blocking)

OP25B とは，ISP(Internet Services Provider)による対策で，契約している ISP のコンピュータから契約外 ISP のメールサーバを使用したメール送信を防止するためのスパムメール対策技術である[2]．

図 1.に OP25B の詳細を示す．通常メールを送信する際にはポート 25 番が使用される．ISP_A 網の端末から ISP_A メールサーバには，ポート 25 番の通信が許可される．ISP_A 網の端末が ISP を変更した場合，その端末は ISP_A

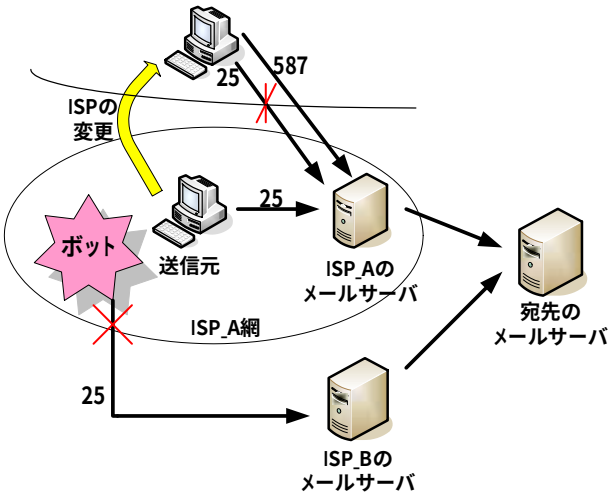


図 1. OP25B の詳細

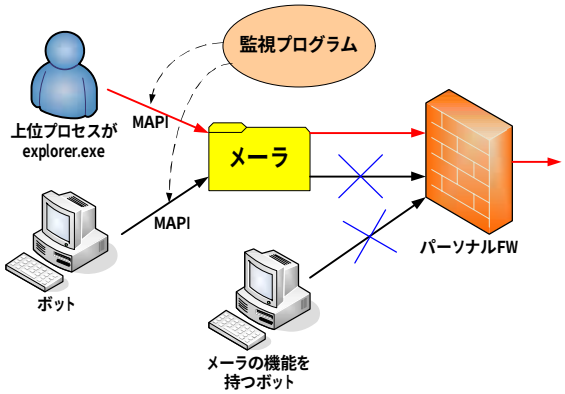


図 2. 提案方式の構成

のメールサーバへのポート 25 番の通信は遮断されてしまう．この場合には，サブミッションポートと呼ばれるポート 587 番を使用しユーザ認証を行うことにより通信を可能にさせることができる．

この技術によりボットが独自の SMTP エンジンを使用してのポート 25 番の通信は遮断することができる．しかし，ボットに感染しているコンピュータが，メーラを用いてユーザの契約しているメールサーバへスパムメールを送信するような場合には対応できない．また，情報収集機能を持つボットがパスワードなどを取得し，正規のユーザを装ってメールを送信した場合には，ポート 587 番を使用してもメール送信を防止できない．

3. 提案方式

ボットは亜種が多く存在することに加え, 新たな機能を持つボットも次々に発見されている. そのため本提案では, 二次被害を防止するためにもクライアント側での対策を検討した.

本提案では, メーラの呼び出し元が正規なユーザと判断できた場合にのみポートを開放しメールの送信を許可する. 正規なユーザかどうか判断するために, MAPI(Messaging API)を監視しプロセスツリーにより上位プロセスの確認を行う(図 2)[3].

3.1. MAPI(Messaging API)

MAPI は Windows 上で電子メールを扱うための標準仕様でメールメッセージを作成, 転送するための関数群である. 一般的に Windows 上では MAPI によりメールを送信する.

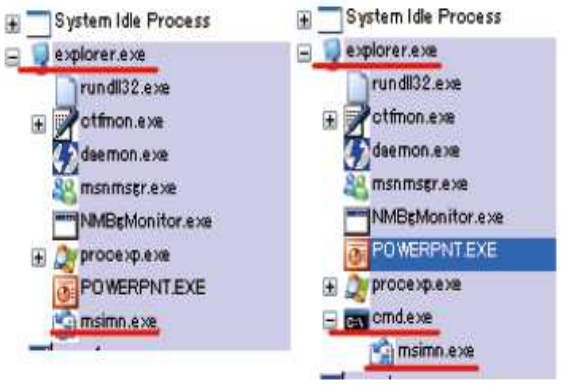
3.2. パーソナルファイアウォールの利用

アンチウィルスソフトの機能としてパーソナルファイアウォールが含まれているものがある. パーソナルファイアウォールは, 外部のネットワークからの侵入およびコンピュータ内部から外部ネットワークへの異常な通信を検知または遮断する. ユーザが定義したルールに従って, パケットやプロトコルを許可または拒否することができる. また, 細かいポート制御も可能である[4].

一般にメールを送信する際, SMTP ポート 25, 587 番を使用する. 提案方式では, パーソナルファイアウォールの設定で常に SMTP ポートを遮断しておく. メーラを呼び出したのが正規なユーザと確認できた場合にのみポートを開放する. 通信終了後に再度ポートを遮断する. この方法により不正なメール送信を防止する.

3.3. プロセスツリーによる確認

3.2.で述べたメーラを呼び出したのが正常なユーザかどうかを判断するためにプロセスツリーを監視する. プロセスツリーとは, 実行



(1)正常時　　(2)異常時

図 3. プロセスツリーによる上位プロセスの確認

中のプロセスをツリー上に表現したものである. メーラが実行されたとき, 通常は explorer.exe が上位プロセスとなる.

図 3 はプロセスを可視化するアプリケーションである Process Explorer v11.04 を用いて, プロセスツリーを表示したものである. メーラのプロセス名は msimn.exe である. 正常時にはメーラの上位プロセスは explorer.exe となる. 一方, ボットに感染していると, メーラを呼び出す上位プロセスは正常時とは異なる. 図 3.(2)では, メーラを呼び出しているのが cmd.exe であり, 異常とみなせる. 図 3 は一つの例であり, ボットに感染していれば必ずメーラの上位プロセスが cmd.exe になると限らない.

プロセスツリーにより, メーラの上位プロセスが explorer.exe と確認できた場合は正規なユーザがメーラを呼び出したと判断し, explorer.exe 以外の場合は不正なプログラムがメーラを呼び出したと判断できる.

3.4. 監視プログラムの動作

提案方式を実現するために, MAPI をフックし, プロセスツリーの検査を行う監視プログラムが必要である. 監視プログラムはコンピュータが起動したときに同時に起動する.

MAPI 関数は 19 種類あり, その中でメール送信に使用される関数を表 1 に示す. 監視プロ

表 1.　メール送信に使用される MAPI 関数

セッション名	機能
MAPILogon	メールサーバへログオン. ユーザ名とパスワードを指定し, 成功時にセッションハンドルを返す.
MAPILogoff	メールサーバからログオフ. MAPILogon にて返ってきたセッションハンドルを指定する.
MAPISendMail	MapiMessage 構造体のメールコンテンツを送信する.

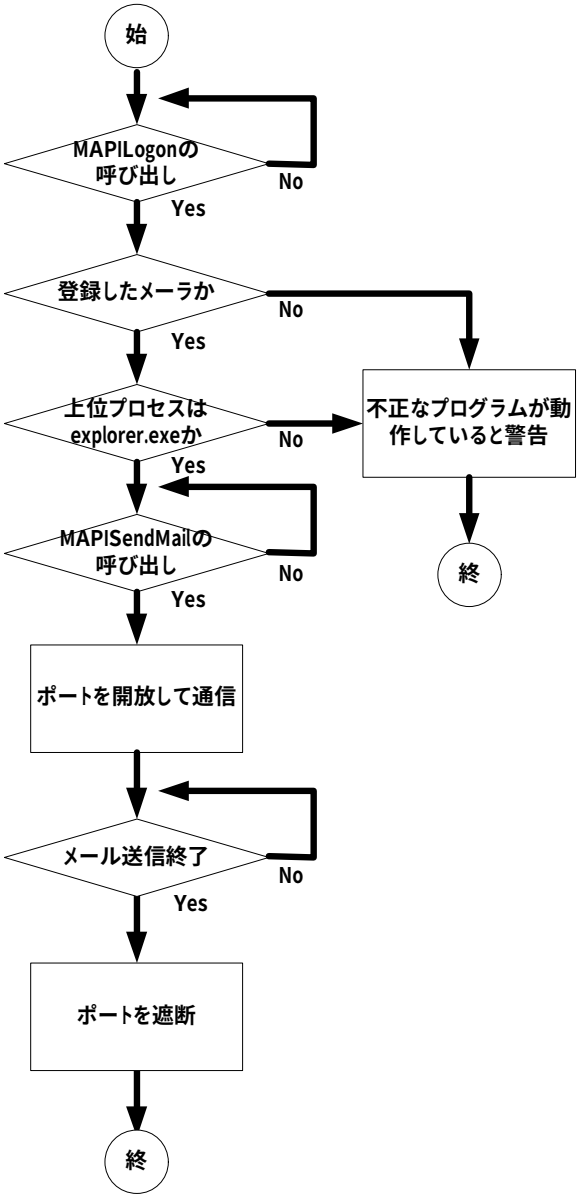


図 4.　監視プログラムのフローチャート

グラムでは MAPILogon と MAPISendMail を監視する.

MAPILogon が呼び出された時にメーラが起動したと判断できるため, まず呼び出されたメーラが登録してあるメーラと一致するかを確認する. 一致しなかった場合は不正なプログラムが動作している恐れがあり, ユーザにアラームをあげて, ポートを開放しない. 呼び出されたメーラと登録してあるメーラが一致した場合は, メーラの上位プロセスをプロセスツリーにより確認する. メーラの上位プロセスが explorer.exe 以外のプログラムだった場合, 不正なプログラムが動作している恐れがあるため, ユーザにアラームをあげて, ポートを開放しない. メーラの上位プロセスが explorer.exe の場合は, 次に MAPISendMail が呼び出されるのを待つ. メールの通信要求がされる時に MAPISendMail が呼び出されるので, この時点でポートを開放する. メール送信の終了を確認したら再びポートを遮断する(図 4).

以上により端末からの不正なメール送信を防止し, ユーザに対して危険にさらされていることを知らせることができる.

4. 実装の検討

(1)登録メーラの確認方法

登録してあるメーラの値を得るためにはレジストリを操作する必要がある. レジストリとは Windows 系 OS 上の, システムやアプリケーションの設定を記録するデータベースである.

HKEY_LOCAL_MACHINE¥SOFTWARE¥Clients¥Mail キーに登録してあるメーラの値が格納されている. キーとは, フォルダのようなもので, その下に多数のキーやエントリを持つ. エントリとは, レジストリの中で実際にデータを表示する要素のことで, 各キーは必ず一つのエントリを持つ[6]. 格納された値を読み込むためには, レジストリを開閉する必要がある. RegOpenKeyEx 関数を使ってキーを開き, RegQueryValueEx 関数で格納されている

値を読み込む．その後，RegCloseKey 関数でキーを閉じる．この方法により登録してあるメーラの値を得ることができる．

(2)呼び出されたメーラの確認方法

呼び出されたメーラを確認するために，MessageBox 関数をフックするプログラム[7]を利用する。フックとは，プログラム中の特定の箇所に利用者が独自の処理を追加できる仕組みのことであり，主に元のプログラムの機能追加や拡張などの手段として使われる．[7]のサンプルプログラムでは、関数の呼び出し元のパス名を表示することができる。サンプルプログラムでフックする関数を MAPI 関数に書き換えることにより、呼び出し元のパス名を取得できる。サンプルプログラムの開発環境は Windows XP，コンパイラは Visual C++.NET である．

表示されたパス名から，拡張子なしのファイル名を取得する．まず，パス名の先頭から'¥'，':'，'/'を検索し，発見した時に次のポインタを保存する．この動作を NULL が検索されるまで繰り返すことにより，拡張子ありのファイル名を取得することができる．次に，拡張子ありのファイル名の先頭から'.'を検索し，最後に現れた'.'の位置に NULL を代入する．この方法により，拡張子なしのファイル名を取得することができる．この方法により呼び出されたメーラのプロセス名を得ることができる．

5. 今後の検討課題

本稿では端末からの不正なメール送信を防止し，ユーザに対して危険にさらされていることを警告する対策を提案した．しかし MAPI に対応していないメーラを使ったメール送信はポートが開放されず通信を行うことができないという課題がある．本研究では，使用するメーラが MAPI に対応している必要があるため今後検討が必要である．

6. まとめ

端末からの不正なメール送信を防止するための対策としてプロセスツリーを監視し，メーラを呼び出したのが正常なユーザと判断できた場合にのみ，パーソナルファイアウォールの SMTP ポートを開放する手法を検討した．今後はこの手法の有効性を確認するための実装を進める．

参考文献

[1]“インターネット上の脅威「ボット」”
(http://itpro.nikkeibp.co.jp/free/ITPro/OPINION/20041216/153951)

[2]“OP25B とは”
(http://bb.watch.impress.co.jp/cda/special/14369.html)

[3]“不正メールの送信防止とボット感染検知の検討”
(http://www.wata-lab.meijo-u.ac.jp/file/gthesis/2007/2007-GT_Abst-Ryoichi_Mamiya.pdf)

[4]“パーソナルファイアウォールについて”
(http://support.microsoft.com/kb/321050/ja)

[5]“Process Explorer v11.04”
(http://technet.microsoft.com/en-us/sysinternals/bb896653.aspx)

[6]“レジストリの概要”
(http://www.2ken.no-ip.com/kaihou/h15/yokoyama.pdf)

[7]“API_Hook.zip”
(http://ruffnex.oc.to/kenji/text/api_hook)

謝辞

本研究を進めるにあたり，多大なるご指導，ご鞭撻を賜りました渡邊晃教授に心より感謝いたします．また有益なご助言，ご検討を頂きました渡邊研究室の皆さんには深く感謝いたします．