2006－CSEC－35 (3)
2006／12／8

Botnetの命令サーバドメインネームを用いたBot感染検出方法

朝長 秀誠† 田中 英彦†

†情報セキュリティ大学院大学
〒221-0835 神奈川県横浜市神奈川区鶴屋町 2-14-1

{mgs051103,tanaka}@iisec.ac.jp

あらまし 近年，コンピュータウィルスの一種であるBotが大きな問題となっている．その理由は，Botに感染したコンピュータ(Bot感染コンピュータ)が攻撃者の命令を仲介する命令サーバを中心にネットワーク(Botnet)を形成し，攻撃者からの命令を受信することである．Botはそのプログラムがインターネット上で公開されているものが多く，それを用いることで誰でもその亜種を作成することができる．このため，現在主流となっているシグネチャマッチングタイプのアンチウィルスソフトウェアでは，パターンファイルの作成が追いつかず，Botの検出が困難になっている．そのため，現在はBotの挙動を用いる検出方法に対する研究が主流になっている．本研究では，BotがDNSサーバに命令サーバのFQDN(Fully Qualified Domain Name)をクエリする挙動を観測することでBotを検出する方法を提案する．

A method of detecting Bot infection using Botnet C&C server domain name

Syusei Tomonaga† Hidehiko Tanaka†

†INSTITUTE of INFORMATION SECURITY
2-14-1, Tsuruya-cho, Kanagawa-ku, Yokohama-shi, Kanagawa 221-0835, Japan

{mgs051103,tanaka}@iisec.ac.jp

Abstract Recently, Bot which is a kind of computer virus becomes a big problem, that is to form a network around the C&C server by which Bot infection computer relays the attacker's instruction, and to receive the instruction of the attacker. Every Bot program has a lot of open source on the Internet, and can make the subspecies easily by using it. Therefore, making the pattern file of virus doesn't catch up, and the detection of the Bot is difficult through the anti virus software of the signature match type. The research on the detection method using the Bot behavior is a main stream. In this paper, we propose the method of detecting Bot by using behavior that Bot make a query to the DNS asking FQDN of the C&C server.

1 はじめに

近年，コンピュータウイルスの一種であるBotが大きな問題となっている．その理由の一つは，Bot感染コンピュータが攻撃者の命令を仲介する命令サーバを中心にBotnetと呼ばれるネットワークを形成し，攻撃者からの命令を受信することである．これによって攻撃者はBot感染コンピュータにDoS攻撃やSPAMメール送信などの攻撃命令を送信し，一斉に同じ攻撃を仕掛けることができる．また，攻撃者はBotのプログラムを更新し，動作変更することができるので，従来のウイルス検知システムによる検出を回避することができる．

Botnetの多くはIRC(Internet Relay Chat)プロトコルを利用して命令サーバとBot感染コンピュータ間の通信(スター型)を行う(図1)．また，Bot感染コンピュータ同士が命令を送受信するPeer to Peerのようなの通信(ランダム型)を行う場合もあるが，ランダム型の通信は，スター型の補足的な役割とし

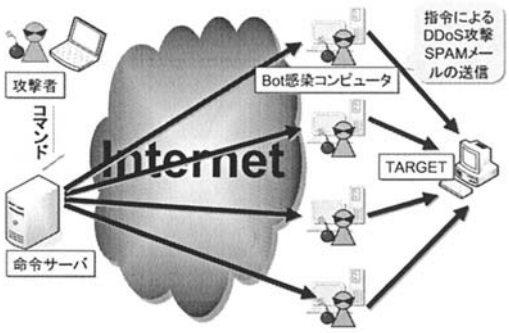

図 1: オーソドックス Botnet

て利用されている．命令サーバは攻撃者が Dynamic DNS を利用して独自に用意したもので，主に第三者のサーバが無断で利用される．Bot 感染コンピュータはこの命令サーバの FQDN を元に DNS サーバで名前解決をして得られた IP アドレスが示す命令サーバに接続する．攻撃者は命令サーバを介して複数の Bot に命令を出すことで Bot 感染コンピュータに何らかの攻撃動作を同時に実行させることができる．

また，Bot の種類がその亜種を含めて非常に多いことも Bot が問題になっている一つの理由である．Bot はそのプログラムがインターネット上で公開されているものが多く，これを用いることでウイルスについて知識がない人でもその亜種を作成することができる．また，公開されている Bot プログラムの中にはウイルス作成のための GUI が含まれているものも存在する．このため，現在主流となっているシグネチャマッチングタイプのアンチウィルスソフトウェアでは，パターンファイルの作成が追いつかなくなり，Bot の検出が難しい．定点観測によって収集された Malware の 80 %が未知のものである [1] という結果からもパターンファイルによる Bot の検出の難しさが分かる．このため，現在は Bot の挙動を用いる検出方法に対する研究が主流になっている．

本稿では，Bot の挙動分析による検出方法の一つとして，Bot が命令サーバの FQDN をクエリする点に着目し，Bot 感染を検出する方法を提案する．すなわち，Botnet の命令サーバの FQDN を Bot から自動的に抽出して FQDN ブラックリストを作成し，DNS の中に組み込む．このブラックリストを元にネットワークで発生する様々な DNS アクセスの中から Bot 感染を検出する．以下，提案方式による Bot の検出方法と検出システムについて説明した後，提案方式の有効性を示す．最後にまとめと今後の課題について述べる．

2 提案方式

2.1 提案方式の特徴

現在筆者らは，他の機関と連携して 11 個の IP アドレスを用いてハニーポットを構築し，Malware の収集を行っている．この定点観測によって，感染元ホストの IP アドレス最上位 8 ビットまたは 16 ビットが一致するアドレス群へネットワーク感染を広めるといった Bot の感染傾向が分かった (表 1)．この結果からネットワークアドレスレンジによって感染する Malware の種類が異なると考えられる．従って，自らの内部ネットワークにウイルス検知システムを適用することにより，自分自身のネットワーク内に感染する可能性のある Bot の感染を阻止できる可能性が高いといえる．本方式では自らのネットワークに感染を仕掛けてくる Malware を対象に Bot 検出のためのシグネチャを作成する方法を用いる．

Malware を検出するシグネチャとして現在主流となっているシグネチャマッチングタイプのアンチウィルスソフトのシグネチャは，専門のウイルス解析者がバイナリに含まれる特長的な文字列を元に作成するので，ウイルスについて知識のない人にはこの方法でオリジナルのシグネチャを作成し，ウイルス検出を行うことが難しい．そこで本方式では，Bot に含まれる命令サーバの FQDN をシグネチャとして使用することでウイルス解析者でなくてもシグネチャの作成を行えるようにする．また，命令サーバの FQDN という決められた値を利用するので自動的にシグネチャを作成することも可能である．

表 1: Honeypot へアクセスのあった Malware の IP アドレス分布
(Honypot 動作期間:6 月 22 日〜11 月 1 日)

IP アドレス:60.32.xxx.xxx

	上位 8bit 一致	上位 16bit 一致	総数
アドレス数	5,697	5,118	6,132

IP アドレス:220.157.xxx.xxx

	上位 8bit 一致	上位 16bit 一致	総数
アドレス数	1,725	669	1,992

本稿で検出対象となるBotはBotnetの通信トポロジとしてスター型を対象とする．また，OSやサービスの脆弱性を突いてネットワークから感染するBotを対象にしてその検出方法を検討する．

2.2 提案方式の構成

2.2.1 Malware収集ステップ

Malware収集ステップでは，FQDNブラックリストを作成するためにBotを収集する．Malwareの収集にはハニーポットを用いる．ハニーポットとは，クラッカーの侵入手口やMalwareの振る舞いを研究するために，ネットワーク上に設置された，脆弱性を持つシステムのことである．このハニーポットはローインタラクション型とハイインタラクション型に分類することができる．

ローインタラクション型ハニーポットとは，システムやサービスをエミュレートすることによって機能する．ハイインタラクション型ハニーポットとは，侵入可能な実際のアプリケーションを備えたシステムである．ローインタラクション型は送られてきたコマンドに対して考えられるレスポンスを返すので，侵入される危険性は少ない．それに対してハイインタラクション型はオペレーティングシステム全体とアプリケーション全体を提供するので，システムに侵入したクラッカーの行う詳細な侵入動作情報を得ることができるが，実際に侵入されるというリスクを持っている．

2.2.2 Malware解析ステップ

Malware解析ステップでは，Malware収集ステップで収集されたMalwareの中からBotのバイナリを解析し，得られたBotnet命令サーバのFQDNをBot検出ステップに転送する．Malwareの解析には静的解析と動的解析を組み合わせた方法を用いる．その理由は，Botに複数のFQDNが含まれている場合があり[1]，動的解析または，静的解析のみでFQDNを抽出しようとする際，全てのFQDNを抜き出せない可能性があるからである．

Malware解析は，動的解析から行う．Windows上でMalwareを実行し，Botとその他のMalwareに分ける．そして，Bot実行時のDNSサーバとの通信をキャプチャし，その中から命令サーバのFQDNを抽出する．次に静的解析を行う．Botのバイナリを実行しメモリ上に展開されたプロセスをメモリダンプし，デバッガを用いてバイナリを逆アセンブルする．そのコードの中に含まれる動的解析で抽出したFQDNとその他FQDNとのメモリアドレスによる近接関係から，命令サーバのFQDNのみを抽出する．静的・動的解析で抽出したFQDNをFQDNブラックリストとしてBot検出に利用する．動的・静的解析時にはMalwareの耐解析機能がFQDN抽出の妨げとなるため，全てのMalwareの解析が可能な状態ではないが，今後Malwareの耐解析機能を無効化する機能を組み込むことで，提案方式の精度を向上することができる．3.2.1では，現在行っている耐解析機能を無効化する方法について述べる．Malware解析システムの流れを図2で示す．

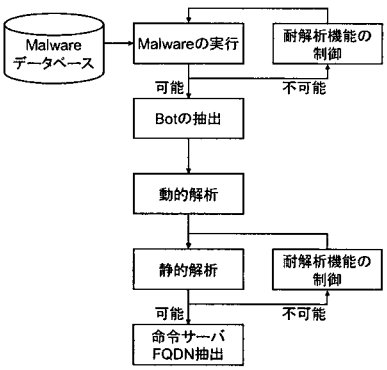

図 2: Malware解析ステップの流れ

2.2.3 Bot検出ステップ

Bot検出ステップでは，Malware解析ステップで得られたFQDNを元にFQDNブラックリストを作成し，それを元にDNSサーバへのアクセスからBotのアクセスを検出する．DNSサーバへの正引きアクセスとFQDNブラックリストを照合し，一致した場合はBotからのアクセスと考えられる．Bot検出ステップの流れを図3で示す．

3 検出システム

3.1 Malware収集システム

本研究ではMalware収集用のハニーポットとしてローインタラクション型ハニーポットであるNe-

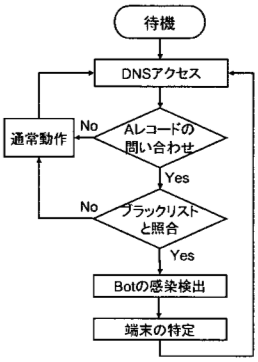

図 3: Bot 検出ステップの流れ

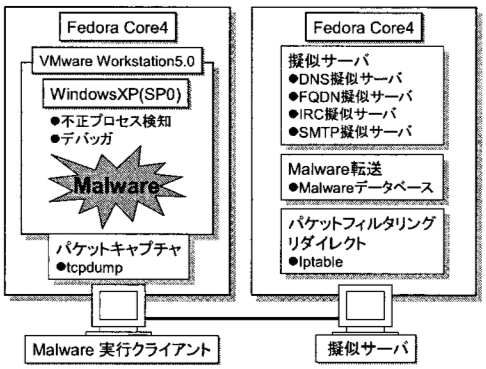

図 4: Malware 解析環境

penthes v0.1.7 [2] を用いる．これは Malware から攻撃を受けた場合，受け側のエミュレートを行い，仕込まれるファイルを取得する．その際，知られている脆弱性を模倣して，Malware と通信を行うことで Malware のファイルをダウンロードするが実行することはない．ローインタラクション型ハニーポットを用いた場合 Malware が感染するようなリスクは低くなる．しかし，既知の脆弱性を用いる Malware しか収集することができず，実際の Windows をハニーポットとして利用した場合と比較して，Malware の収集能力が低下してしまう問題点がある．

ハニーポットで収集した Malware は随時 Malware 解析システムに転送する．

3.2 Malware 解析システム

Malware 解析システムでは Malware 解析環境を構築し，その中で Malware の解析を行い命令サーバの FQDN を抽出する．Malware 解析環境の構成を図 4 に示す．

3.2.1 Malware 実行クライアント

Malware 実行クライアントでは Malware を実行し，実行している不正プロセスのメモリダンプと，パケットキャプチャから FQDN を抽出する．Malware 実行クライアント上では VMware Workstation5.0 上に仮想ホストとして，WindowsXP(SP0) をインストールし，その上で Malware を実行する．VMware Workstation はスナップショット機能があり，Malware 実行後の Windows を感染以前の状態に容易に戻すことが可能である．また，コマンドラインでの実行が可能なため，自動的に仮想マシンを立ち上げて，解析を行うことができる．

しかし，現在の Malware は仮想ホスト上で実行されることを検知することで，プロセスを停止させるものが存在する．これは仮想ホスト上で Malware を実行させて解析行うことが多く，解析された Malware の挙動を検出に利用されることを阻止するためである．Malware は仮想ホスト上で受信するパケットや，Windows の内部情報から得られる以下の値を元に VMware 上で実行されているか判断する．

1. VMware Backdoor I/O
2. 仮想ホストの MAC アドレス
3. Video BIOS
4. SCSI デバイスのデバイス名
5. IDE デバイスのデバイス名
6. vmnat の MAC アドレス

Malware の持つ VMware を検知する耐解析機能を回避するためには，これらの情報が仮想ホスト上から得られないようにする必要がある．それを可能にするために [3] では，VMware のバイナリコードを書き換えるパッチを提供している．このパッチの特徴としては，(1)〜(5) の偽装が可能となる．しかし，これだけでは vmnat の MAC アドレスは変更できないので，バイナリを直接書き換える必要がある．それ以外にも vmx ファイルにオプションとして付け加えることで，耐解析機能を回避することが可能である．

表 2 では，3.1 のハニーポットを利用して収集した Malware を用いて，VMware Workstation5.0 に (1)〜(6) の特徴を無効化した状態で Malware を実行した場合と，無効化しない状態で実行した場合の比較を行った．結果として，実行可能な Malware の中で，35 個が動作するようになった．動作しなかった多くの Malware は命令サーバからの指令が受信可能かなどのネットワーク環境を検知して停止するものだったため，実際の命令サーバなどに接続できるような環境にすることで，より多くの解析が可能になると思われる．

実行された不正プロセスのコードを得るためにはデバッガを用いる．しかし，Malware はデバッガの解析を阻止する機能を持つものが存在する．また，難読化によって FQDN 文字列を抽出できない場合も存在する．従って，この耐解析機能を回避する方法の検討が必要である．

3.2.2 擬似サーバ

擬似サーバは，各種サーバを模擬した機能と，Malware 実行クライアントに Malware を転送する機能を持つ．サーバの模擬は Malware の実行時にネットワークアクセスを把握するために，実際のネットワークに近い状況を作り出し，Malware と通信を行う．擬似サーバは DNS サーバ，IRC サーバ，SMTP サーバ，FTP サーバの 3 つから構成される．

DNS サーバはクエリがあった場合，偽装した IP アドレスを返す．この場合，感染ホストの利用する DNS サーバ以外のアドレスにクエリしたとしても，この模擬サーバでアクセスを返す．IRC・FTP サーバは，ポートに対してアクセスがあった場合，3Way ハンドシェイクを行う．SMTP サーバはメールの送信が行われた場合，SMTP のアクセスを模擬する．

また，Malware 実行クライアントで実行した Malware の通信を制御するために iptable を用いる．52，80 などいくつかのポートを ACCEPT し，Bot からの IRC 通信があると考えられるポートを擬似サーバの 6667 ポートにリダイレクトする．そして，DNS アクセスを含めた通信を tcpdump でキャプチャする．

表 2: VMware 耐解析 Malware の調査

	動作	停止	実行不能	総数
対策なし	1,169	177	308	1,654
対策あり	1,204	142	308	1,654

3.3 Bot 検出システム

Bot 検出システムは DNS サーバへ名前解決の問い合わせがあった場合，そのメッセージから FQDN を抽出し，FQDN ブラックリストとマッチングを行う．そして，一致した場合にそのメッセージの送信元を Bot 感染クライアントと特定し，名前解決の結果を 127.0.0.1 として返す．127.0.0.1 にする理由は，Bot のアクセスを感染ホスト上に留める事で，命令サーバへ接続させないためである．現在の Bot は DNS サーバへクエリする際，感染クライアントが利用するローカルの DNS サーバを利用するが，今後インターネット上の他の DNS サーバを利用する場合も考えられる．従って，ローカルの DNS サーバへのアクセス監視だけでなく，外向きの DNS サーバアクセスをネットワーク上で監視する必要がある．それ以外の方法として，外向きの DNS サーバアクセスを全てリダイレクトして，ローカルの DNS サーバへアクセスを変えてしまう方法も考えられる．本研究では，後者を採用する．

DNS サーバへのアクセスと対応付ける FQDN ブラックリストは Malware の収集数に応じて増加すると考えられるので，線形検索では DNS サーバのレスポンスタイムが減少してしまう．

そこで本研究では，ハッシュ法などの高速検索アルゴリズムを利用する．また，FQDN ブラックリストは古くなると Botnet に利用されていない FQDN が増加し，不要な検索が増加してしまうので，一定期間ハニーポットに感染がない Bot の FQDN は消去することで，FQDN ブラックリストの巨大化を防止する．

4 提案方式の有効性

同じネットワークアドレスレンジで収集された Malware から作成される FQDN ブラックリストに基づいて Bot の検出を行った場合，どの程度の精度を持つかその有効性を調べるために以下のような簡単な実験を行なった．

まず，二つの地点 (地点 P，地点 Q) でハニーポットを設置して一定期間 (1 週間)Malware の収集を行った．各場所には 4 つの連続する IP アドレスでハニー

表 3: Malware 検出率

地点 P (収集期間:8 月 20 日〜26 日 測定日:8 月 26 日)

Malware 収集地点	A B	B C	A C	A B C
Malware 検出率	0.94	0.78	0.84	1.00

地点 Q (収集期間:9 月 7 日〜13 日 測定日:9 月 13 日)

Malware 収集地点	X Y	Y Z	X Z	X Y Z
Malware 検出率	0.94	0.94	0.94	0.94

ポット (A, B, C, D) があり，その中 3 つ (A, B, C) は FQDN ブラックリスト作成のために，残り 1 つ (D) は Bot 検出用ホストとして用いられる．3 つのハーニポットから収集された Malware の FQDN ブラックリストの有効性を調査するために，4 種類のブラックリスト (A と B，A と C，B と C，A と B と C からのブラックリスト) を作成した．作成したブラックリストを用いて D にアクセスしてくる Malware の検出を行った．同様に比較のため地点 Q でも同じ実験を行った．その結果を表 3 に示す．2 つのハニーポットの組合せから得られるブラックリストによる Bot の検出率より，3 つのハニーポットの組合せから得られるブラックリストによる検出率が高いことが分かった．すなわち，同じネットワークアドレスレンジでハニーポットを 3 つ以上設置すると提案方式が有効であると考えられる．

5 まとめと今後の課題

提案方式の精度を上げるために，今後以下の課題について検討する．

耐解析機能の回避 現在の Malware 解析システムでは，仮想ホストやネットワーク環境を検知する Malware を解析することはできない．仮想ホストについては他の仮想化技術を用いることで，解析率を向上することができる可能性がある．ネットワーク環境検知については，直接命令サーバや Web サーバにアクセスできるような環境に変更することで回避できる可能性がある．その際，ポートスキャンなどの攻撃動作を外部へ出さない環境の構築が必要になる．

FQDN 抽出 本方式では実行された不正プロセスのコードから FQDN を抽出する．Bot には命令サーバの FQDN 以外に複数の FQDN が含まれる可能性がある．その中から命令サーバの FQDN を抽出する方法としては，動的解析から得られた命令サーバ FQDN とのメモリ上の近接関係から判断している．しかし，この方法が全ての Bot に対応できるかは未確認である．従って，全ての Bot に対応できる方法の検討が必要である．

Bot 検出の限界 本方式では全ての Bot を検出することはできない．Bot の特徴としてプログラムのアップデートを行うことができる．これにより，1 つの Bot 検出の手法が開発されたとしても，その検出を回避するアップデートが行われた場合，その検出方法は無効となってしまう．従って，既存の Bot 感染拡散を検出する手法では全ての Bot を検出できないといえる．そのため，Bot 感染検出のためには様々な Bot の特徴を捉える方法を融合することで検出精度の向上をはかることができると考える．本方式は，その 1 つをになう方法であるといえる．

謝辞

本研究では情報セキュリティ大学院大学の堀合啓一氏に一部データの提供を頂いた．本研究を進めるにあたり，有益な助言と協力を頂いた IISEC Bot 研究チームの関係者各位に深く感謝します．

参考文献

[1] 高橋正和, 村上純一, 須藤年章, 平原伸昭, 佐々木良一, “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 p2512-p2523, Aug.2006.

[2] Nepenthes - finest collection - `http://nepenthes.mwcollect.org/`

[3] French Honeynet Project VMware fingerprinting counter measures `http://honeynet.rstack.org/tools/vmpatch.c`