ボットネットの多段追跡システムの構想と CCC DATAset 2008 の利用手法

三原　元[†]　名雲　孝昭[††]　芦野　佑樹[†††]　上原　哲太郎[§]　佐々木　良一[‡]

[†,‡]東京電機大学大学院　工学研究科　情報メディア学専攻
[††]東京電機大学　工学部　情報メディア学科
[†††]東京電機大学大学院　先端技術研究科　情報通信メディア学専攻
[§]京都大学　学術情報メディアセンター

あらまし　近年，ボットネットによる被害が問題になっている．ボットネットは，ボットになってしまった PC を特定して隔離したとしても別の PC がボットになってしまい，根本的な解決にならない．著者らは，これの問題に対して，ボットネットによって被害を受けた PC が存在するネットワークと，ボット PC が存在するネットワークの管理者間で情報共有を行うことで，ボット PC，C&C(Command&Control)サーバ，そしてハーダーが操作する PC までを追跡する多段追跡システムの開発を構想している．本論文では，この多段追跡システムについて述べると共に，このシステムにおける，研究用データセット CCC DATAset 2008 を解析して得られたデータを元にした，C&C サーバの特定方法について述べる．

Plan of Multistep Pursuit System of Botnet And Use Technique of CCC DATAset 2008

Hajime Mihara[†]　Takaaki Nagumo[††]　Yuki Ashino[†††]　Tetsutarou Uehara[§]
Ryoichi Sasaki[‡]

[†, ‡]Information Systems and Multimedia Design, Graduate School of Engineering, Tokyo Denki University
[††]Information Systems and Multimedia Design, Tokyo Denki University
[†††]Information, Communication and Media Design Engineering, Graduate School of Advanced Science and Technology, Tokyo Denki University
[§]Academic Center for Computing and Media Studies, Kyoto University

Abstract　Recentry, damage caused by the botnet becomes a big problem. There is a problem that the other bot PCs are produced, even if one bot PC could be specified. Therefore, it is not a fundamental solution. To solve this problem, authors propose the multistep trace back system that pursues not only bot PCs but the C&C server and Harder. Moreover, authors describe the C&C server detection technique using Data set for research CCC DATAset 2008 for the second step in this system.

1 はじめに

近年，ボットネット[1]による被害が問題になっている．ボットネットとは，ボットと呼ばれるマルウェアに感染した PC(以下，ボット PC とする)が複数組み合わさって構成されるネットワークである．ボット PC は，C&C サーバと呼ばれる中継サーバを介して，ボットネットを操作する攻撃者(以下，ハーダーとする)からの命令を受け取ることで様々な活動を行う．

現在では，数百台から数万台のボット PC から構成されるボットネットが確認されている[1]．このボットネットは，ハーダーからの命令により，複数のボット PC が一斉に DoS(Denial of service)攻撃を行う，DDoS(Distributed DoS)攻撃に利用される．

ボットからの攻撃パケットは送信元 IP アドレスが偽装されていることがあり，ボット PC の特定は困難である．その問題に対処するため，IP トレースバックシステム[2]が提案されている．しかしながら，IP トレースバックシステムだけでは，攻撃パケットを送信しているボット PC の特定はできても，ハーダーや，ハーダーの命令を中継する C&C サーバを特定することはできない．

そこで，著者らは，ネットワーク管理者同士が情報共有を行い，ボット PC の特定だけではなく，C&C サーバやハーダーの操作する PC の特定を目的とした，多段階トレースバックシステムを構想している．

本論文では，この多段追跡システムの構想について述べると共に，研究用データセット CCC DATAset 2008(以下，CCC2008 データとする)の解析結果と，解析結果を用いた多段追跡システムの実現方法について述べる．

2　多段追跡システムの構想と CCC2008 データ

2.1　多段追跡システム概要

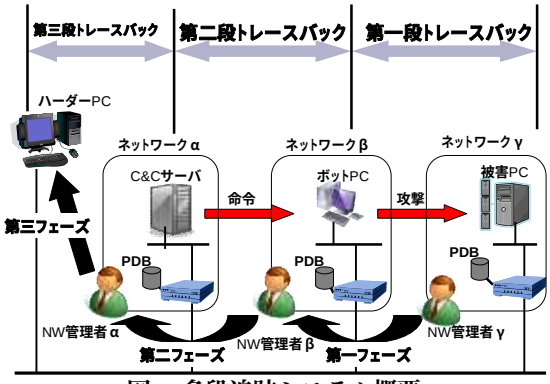

図 1 多段追跡システム概要

著者らが構想している多段追跡システムは図 1 のように 3 つのフェーズから構成されている．ネットワークβにあるボット PC が，ネットワークγ内の PC に対して攻撃を行っていると仮定し以下に各フェーズについて説明する．

表 1 多段追跡の通達の流れ

① ネットワーク管理者γが，ネットワーク管理者βに通達を行う．
② 通達を受けたネットワーク管理者βはネットワーク管理者αに通達を行う
③ 通達を受けたネットワーク管理者αがハーダーによって操作する PC を特定する

(1)第一フェーズ：第一段トレースバックは，IP トレースバックシステムを用いて，ボット PC が存在するネットワーク(以下ネットワークβ)まで追跡する(図 1①，表 1①)．

(2)第二フェーズ：第二段トレースバックでは，PDB を用いて，ネットワークβから C&C サーバが存在するネットワーク(以下ネットワークα)までを追跡する(図 1②，表 1②)．

(3)第三フェーズ：第三段トレースバックは，ネットワークαからハーダーの操作する PC までを追跡する(図 1③，表 1③)．

本論文では，第一フェーズから第二フェーズについて検討を行った．

2.2　第一段トレースバックシステム

IP トレースバックシステムとは，パケットの送信元 IP アドレスが偽装されていても，正しい送信元 IP アドレスが特定可能な技術の総称である．

本構想では，第一段トレースバックシステムには出国印方式 IP トレースバックシステム[2]を使用する．出国印方式 IP トレースバックシステムは，パケットにマーキングを行う方式[3]の一つである．各ネットワークに設置されたエッジルータによって，通過するパケットにルータ自身の IP アドレス等の情報を書き込む．この方式を用いることで，ネットワークβのボット PC が，送信パケットの送信元 IP アドレスを偽装しても，ネットワーク管理者γはボット PC の IP アドレスを知ることができる．システムには専用のエッジルータの導入のみであるため，他の方式と比較して導入時の変更点が少ない．上記の理由により，第一段トレースバックシステムには，出国印方式 IP トレースバックシステムを使用する．

2.3　第二段トレースバックシステムの構想と CCC2008 データ

第二段トレースバックシステムでは，パケットデータベース(以下 PDB とする)を用いる．PDB は，　エッジルータに設置し，一定時間の間エッジルータを通過する全ての通信データを保存する．

PDB によって，ネットワーク管理者βは，①ボット PC の IP アドレス，②C&C サーバの IP アドレス，③ボット PC の感染から他 PC

攻撃までの一連の流れを知ることができる．本構想での③の特定については，ネットワーク管理者βによって PDB に記録されているボット PC の通信データを調査し特定する物とする．①，②は第 3 章で記述する方法で解析することにより，それぞれの IP アドレスの特定を行う．

3　CCC2008 データの解析と実験結果

3.1　CCC2008 データに関して

著者らは，CCC2008 データの解析を行い，2.3 節の①，②の特定方法について検討を行った上で，実験を行い，その有用性の検証を行った．

今回のワークショップで CCC(サイバークリーンセンター)から提供された CCC2008 データは，以下の 3 種類のデータで構成される．

データ 1) マルウェア検体
　ハニーポットで収集したマルウェア検体のハッシュ値

データ 2) 攻撃通信データ
　ボットに感染したハニーポット（以下，ハニーポット）の 2 台の通信を libcap 形式で 2 日分キャプチャしたデータ

データ 3) 攻撃元データ
　ハニーポット 112 台の 6 ヶ月のマルウェア取得時のログデータ

著者らは，上記のデータ 2)の解析を行った．

3.2　CCC2008 データの解析

表 2 検出したボット名称(Symantec の分類による)

ボット名称	ボット名称
W32.IRCBot	W32.Bobax!dr
W32.IRCBot.Gen	Backdoor.IRC.Bot
W32.Spybot.Worm	名称不明1
W32.Virut.A	名称不明2
W32.Virut.B	名称不明3
W32.Virut.H	名称不明4
W32.Virut.W	名称不明5
W32.Virut!gen	

データ 2)を解析した結果，166 個の検体が存在していることを確認した．著者らは，アンチウィルスソフトを用いてこれらの検体を調査したところ，161 個のボットの存在を確認した．検出された検体は，10 種類に分類することができた．しかし，アンチウィルスソフトは，他の 5 個の検出ができなかった．これら 5 個の検体は，それぞれバイナリデータが異なることから，5 種類のボットとして取り扱うことにした．アンチウィルスソフトにより分類できた 10 種類と，分類できなかった 5 種類のボットを含めた計 15 種類のボットを表 2 に示し，以下 15 種類のボットと呼ぶ．

3.3.　特定方法の検討

2.3 節①，②を特定する方法として，以下の方法を検討した．

(1)SYN パケットの再送間隔

ボットは，自動的に動作するプログラムであることから，何らかの規則正しい動作をしているのではないかと考え，この動作を利用できないかと考えた．

(2)C&C サーバ

ボットは，C&C サーバから命令を受けることから，ボットと C&C サーバ間で行われる通信の特徴を見つけ，その特徴を使用できないかと考えた．

(3)ダウンロードサーバ

ボットは，自身のプログラムの更新などのため，特定のサーバからデータをダウンロードする．従って，このサーバの特徴を見つけ，その特徴を利用できないかと考えた．

(4)接続先サーバ

ボットは，C&C サーバ及びダウンロードサーバに接続するので，15 種類のボットに共通する接続先を見つけ，その接続先の特徴を利用できないかと考えた．

以上の検討をもとにして，以下の解析を行った．

3.2.1　SYN パケットの再送間隔

表 3 SYN パケット再送間隔パターン 1

ボート	1～2回目(秒)	2～3回目(秒)	3回目～ポート変更(秒)	1～3回目合計(秒)
1031	2.834	5.978	42.069	8.812
1040	3.045	6.043	42.009	9.088
1045	成功	–	–	–
平均	**2.940**	**6.011**	**42.039**	8.950

表 4 SYN パケット再送間隔パターン 2

ボート	1～2回目(秒)	2～3回目(秒)	3回目～ポート変更(秒)	1～3回目合計(秒)
1033	2.861	6.049	12.292	8.910
1038	2.918	5.994	12.285	8.912
1039	3.055	5.953	12.168	9.008
1040	3.046	5.932	12.287	8.978
1044	3.030	5.992	12.289	9.022
1047	2.942	5.991	12.330	8.933
1052	2.964	6.048	12.170	9.012
1053	2.991	5.936	12.287	8.927
1054	2.911	5.984	12.284	8.895
1055	成功	–	–	–
平均	**2.969**	**5.987**	**12.266**	8.955

著者らは，ハニーポットが，外部のサーバに接続する際に行われる SYN パケットを送信する間隔を測定した．一例として，表 3 と表 4 には，同一のボットから送信される SYN

パケットの再送間隔を示した．その結果，SYN パケットはどのネットワークアドレスに対しても一定間隔で行われることが確認できた．しかし，その間隔時間は，ネットワークアドレス毎に異なることが分かった．

結果として，再送間隔を用いて 2.3 節①，②の特定の可能性について検討を行ったが，宛先ネットワークアドレス毎の特徴であるため，適切ではないという結論に至った．

3.2.2 C&C サーバ

C&C サーバが使用するポート及び命令コマンドを，nothink.org[4]が公開している情報を元にして調査した(表 5，表 6)．表 5 の出現数は，ハニーポットが C&C サーバに対して接続を試みた回数を指す．また，表 6 の 1)，2)は感染指示命令であり，3)，4)はダウンロード命令である．

表 5 C&C サーバ使用ポート

TCP PORT番号	出現数	割合
6667	51	18
8080	36	12
65520	89	32
5190	23	8
10324	30	10
1863	12	4
18067	7	2
2293	6	2
3938	6	2
7763	8	3
その他	18	7
合計	286	100

表 6 ボットへのダウンロード指示

1) ipscan s．s．s．s dcom2 –s
2) advscan dcom135 160 5 0 –b –r –s
3): !get http://ダウンロードサーバ URL
/~grander/unpr．exe
4) #rs2:=XIa0ZhVFU3q69d0a8Df5/betV/
8WnIWAV9LI/B8t8K9Iwq5+Ttdc7
+yHIKyxzLPV6tJ

表 5 より，C&C サーバに共通するポートがないために，C&C サーバが使用するポートを利用した 2.3 節①，②の特定は現実的ではない．

また，今回確認した表 5 の命令コマンドをベースに，命令コマンドを利用した 2.3 節①，②の特定方法について検討を行った．しかし，LURHQ Threat Intelligence Group のレポートでは，90 種類の命令コマンドが実装されたボットも存在することが分かっている[5]．そのため，命令コマンドを利用した 2.3 節①，②の特定方法は現実的ではない．

3.2.3 ダウンロードサーバ

ハニーポットが，自動的にもしくは C&C サーバに命令されてからデータを取得するために接続するサーバ(以下，ダウンロードサーバ)について調査した．また，ハニーポットが，ダウンロードサーバからダウンロードした各ボットプログラムとトロイの木馬のデータサイズを調査した(表 7)．その結果，ダウンロードサーバが使用するポートは，80 がほとんどであった．さらに，ダウンロードしたどのデータもサイズが小さいことが確認できた．

表 7 マルウェアのデータサイズ

マルウェア名称	サイズ (KB)	マルウェア名称	サイズ (KB)
W32.IRCBot	122	W32.Bobax!dr	67
W32.Spybot.Worm	62	**未検出1**	45
W32.IRCBot.Gen	45	**未検出2**	165
Backdoor.IRC.Bot	65	**未検出3**	104
W32.Virut.A	124	**未検出4**	214
W32.Virut.B	312	**未検出5**	87
W32.Virut.H	228	lorder.exe	27
W32.Virut.W	47	lox.exe	45
W32.Virut!gen	51	**平均**	**106**

以上の結果から，ダウンロードサーバのポート番号や http 通信の内容を用いて，2.3 節①，②の特定は，ボットの通信とは関係ない通信とダウンロードサーバの通信の区別が困難であるため，不適切であると結論づけた．

3.2.4 接続先サーバ

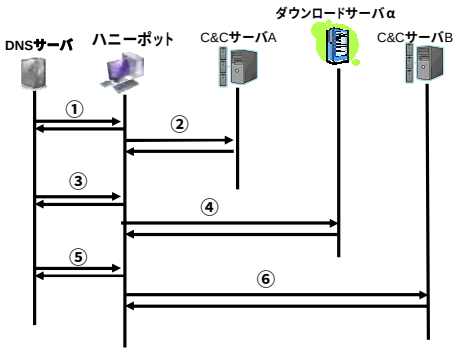

図 2 ボットに感染したハニーポットの動作例

著者らは，ボットに感染したハニーポットの接続先サーバを，15 種類のボットごとに調査した．その結果，複数回異なる C&C サーバ及びダウンロードサーバに接続する通信を確認した(図 2)．図 2 の説明を以下に示す．

① DNS サーバに C&C サーバ A の名前

解決を行う

② C&C サーバ A に接続し，ダウンロード命令を受ける

③ DNS サーバにダウンロードサーバαの名前解決を行う

④ ダウンロードサーバに接続し，データをダウンロードする

⑤ ④によって，DNS サーバに C&C サーバ B の名前解決を行う

⑥ C&C サーバに接続する

この調査では，②は 3.2.2 項と同じ方法で確認し，⑤は，(1)ネットワークから隔離された表 8 の PC を用意し，(2) 表 8 のゲスト OS 上で，④のデータを実行する．(3)そしてホスト OS の通信データを，パケットをキャプチャする機能を持つ Wireshark[6]を用いて取得して確認を行った．

表 8 ダウンロードデータ実行用 PC 環境

ホスト OS	OS	Windows XP Professional
	CPU	Pentium4 3GHz
	メモリ	2GB
	仮想 PC ソフト	Microsoft Virtual PC 2007
ゲスト OS	OS	Windows XP Professional
	メモリ	512MB

この調査結果より，15 種類のボットは，表 9 に示す 4 つのサーバに一つ以上接続していることがわかった(表 10)．また，表 9 のサーバに接続する際，ボットは DNS サーバに名前解決を行い，接続していることを確認した．

以上より，著者らは表 9 の 4 つのサーバの名前解決を行う通信を利用し，2.3 節①，②の特定を行う方法が有用であるのではないかと考え，4 章の実験を行った．

表 9 共通する 4 つの通信先ホスト

1) C&C サーバ P
2) C&C サーバ H
3) ダウンロードサーバ Y
4) ダウンロードサーバ N

表 10 ボットごとの表 9 の接続先サーバ

ボット名称	接続先サーバ	ボット名称	接続先サーバ
W32.IRCBot	1)	W32.Bobax!dr	2)
W32.IRCBot.Gen	1)	Backdoor.IRC.Bot	1)
W32.Spybot.Worm	2)	名称不明1	1)
W32.Virut.A	3)	名称不明2	1)
W32.Virut.B	3),4)	名称不明3	1)
W32.Virut.H	3)	名称不明4	1)
W32.Virut.W	4)	名称不明5	1)
W32.Virut!gen	2)		

4 実験

今回の実験では，IDS(侵入検知システム)である Snort[7]を使用した．Snort を使用することで，検出したいパケットをルールファイルに設定し，対象のパケットを検出することができる．従って，以下の実験では，Snort に対し，2.3 節①の特定をするために表 11 の 1)を検出するルールを設定し，2.3 節②の特定をするために表 11 の 2)を検出するルールを設定して実験を行った．

また，Snort は，第二段トレースバックシステムにおいてでも，2.3 節①，②の特定に使用する予定である．これは，Snort の機能を 2.3 節①，②の特定に利用するのが目的であり，第二段トレースバックシステムに IDS を使用するものではない．

表 11 Snort で検出するパケット

1) ハニーポットが表 9 のドメインのクエリを DNS サーバに送信するパケット(ボット PC の特定)
2) 1)のパケットの DNS サーバからの返答パケット(C&C サーバの特定)

4.1 検知実験

4.1.1 実験環境

本実験の目的は，ボットの検知と C&C サーバの特定の実験を行うことである．以下に実験の構成を示す．

(1) ボットの通信データ

(2) Snort

(1)は，データ 2)の中に含まれるボットが感染してから C&C サーバ及びダウンロードサーバへの接続を行うまでものである．実験は，それぞれ各 15 種類のボットの通信データで行った．また，(2)Snort を導入した PC の仕様を表 12 に示す．

表 12 Snort を導入した PC の仕様

CPU	Pentium4 3GHz
メモリ	2GB
OS	Windows XP Professional
Snort	Snort 2.8.3

4.1.2 実験方法

本実験の手順を以下に示す．

(a) Snort に表 11 の 1)と 2)を検出するルールを設定する

(b) Snort に(1)のデータを入力する

(c) Snort が表 11 の 1)と 2)を検出するか確認する

本実験では，(c)で Snort から表 11 の 1)

と 2)が検出されれば 2.3 節①，②の特定が行えたものとする．

4.1.3 実験結果

実験の結果，全 15 種類のボットに関して，データ 2)における 2.3 節①，②の特定に成功したことが確認できた（表 13）．

表 13 検知実験の結果

ボット名称	実験1	ボット名称	実験1
W32.IRCBot	検出可	W32.Bobax!dr	検出可
W32.IRCBot.Gen	検出可	Backdoor.IRC.Bot	検出可
W32.Spybot.Worm	検出可	名称不明1	検出可
W32.Virut.A	検出可	名称不明2	検出可
W32.Virut.B	検出可	名称不明3	検出可
W32.Virut.H	検出可	名称不明4	検出可
W32.Virut.W	検出可	名称不明5	検出可
W32.Virut!gen	検出可		

4.2 誤検知実験

4.2.1 実験環境

本実験の目的は，表 11 の方法で誤検知の有無について確認を行うことである．以下に実験の構成を示す．

(1) 非ボット感染ネットワーク通信データ
(2) Snort

ここで，(1)とは本研究室の LAN 内の通信を指す．その概要を表 14 に示す．(2)は 3.4.1 項で使用した Snort の構成と同じである．

表 14 非ボットネットワーク通信データ

PC台数	20台 (OS:WindowsXP)
通信データ取得時間	24時間
パケット数	約50万パケット

4.2.2 実験方法

本実験の手順を以下に示す．

(a) Snort に表 11 の 1)を検出するルールを設定する
(b) Snort に(1)の通信データを入力する
(c) Snort が表 11 の 1)を検出するか確認する

本実験では，(c)で表 11 の 1)が検出されれば，誤検知が発生したものとする．

4.2.3 実験結果

表 15 誤検知実験の結果

ホスト名	誤検知
C&Cサーバ P	無
C&Cサーバ H	無
ダウンロードサーバ Y	無
ダウンロードサーバ N	無

実験の結果，非ボット PC で構成されるネットワークで誤検知が発生しないことが確認できた（表 15）．

4.3 考察

4.1 節の実験結果から，期待した結果を得られたので，実環境下でも同様の結果が得られるものと期待できる．また，4.2 節の実験結果から，第二段トレースバックシステムの実運用にも誤検知を少なくできるのではないかと考えられる．

以上の結果から，3.2.4 項のデータを利用した 2.3 節①，②の特定方法は，第二段トレースバックシステムでも有用であることが期待でき，システム実現の見通しを立てることができた．

第二段トレースバックシステムのさらなる精度の向上には，今回の調査で行った 15 種類のボットの他に，さらにボットの調査を行っていく必要があると考える．

5 まとめ

本論文では，多段追跡システムの構想と第二段トレースバックシステムにおける CCC2008 データの解析結果の適用方法に関して述べた．また，適用方法の有用性を実験によって得られた．

今後は多段追跡システムの第三段の方式の提案を行っていくと共に，引き続きボットに対する調査を行い，第二段トレースバックシステムへの適用を目指す．

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.8,2007

[2]藩博文，佐々木良一，「IP トレースバックのための出国印方式の試作と評価」，情報処理学会論文誌，Vol. 49，No. 9，2008

[3]D. Songetal.:”Advanced and Authenticated Marking Schemes for IP Trace back”,Proc. IEEE INFO-COM,Apr.2001

[4] nothink. org
http://www. nothink. org

[5] LURHQ Threat Intelligence Group: Phatbot Trojian Analysis
http://www. lurhq. com/phatbot. html

[6]wireshark. org
http://www. wireshark. org/

[7] snort. org
http://www. snort. org