パケットキャプチャーから感染種類を判定する発見的手法について

桑原 和也 †　　菊池 浩明 †　　寺田 真敏 ††　　藤原 将志 ††

† 東海大学情報理工学部情報メディア学科
259-1292 神奈川県平塚市北金目 1117，mulberry，kikn@tokai.ac.jp
†† 日立製作所 Hitachi Incident Response Team (HIRT)
212-8567 神奈川県川崎市幸区鹿島田 890 日立システムプラザ新川崎

あらまし 本研究では，研究用データセット CCC DATAset 2009 の攻撃通信データからマルウェアのダウンロード状況，ポートスキャンの振る舞いについて解析を行い，通信データからマルウェアの感染の有無や多重に感染する際のいくつかの特徴を発見した．これらの発見的手法とその精度について報告する．

Heuristics for Detecting Types of Infections from Captured Packets

Kazuya Kuwabara†　　Hiroaki Kikuchi†　　Masato Terada††
Masashi Fujiwara††

†Graduate School of Engineering, Tokai University, 1117 Kitakaname, Hiratsuka, Kanagawa 259-1292
††Hitachi, Ltd. Hitachi Incident Response Team (HIRT), 890 Kashimada, Saiwai-ku, Kawasaki, Kanagawa 212-8567

Abstract This paper studies the analysis on the CCC Data Set 2009 for behavior of downloads of the malware and the port-scans . The analyses show that some new features of the single and the federated infections. Based on the analysis, the paper proposes heuristic techniques for detection of malwares infection and reports the accuracy of the proposed heuristics.

1　はじめに

本研究では，サイバークリーンセンター（CCC）から提供された，研究用データセット CCC DATA set 2009 の攻撃通信データを解析する．攻撃通信データとはハニーポットの通信を tcpdump でパケットキャプチャーした libpcap 形式のファイルである．ハニーポットはホスト OS 上の 2 台のゲスト OS がそれぞれインターネット接続されており，パケットキャプチャーはホスト OS 上で行われている [1].

攻撃通信データは定期的にリセットされる．本稿では，Windows XP が送信する NTP パケットを利用して，攻撃通信データを観測単位時間に分割した（以下，**スロット**）[2]．2 日間で 145 個のスロットが生じる．このパケットキャプチャーデータ（以下，pcap データ）は情報量が膨大であり，手作業で全体を把握することは困難である．

そこで本研究では，pcap データからボットに関するいくつかの特徴を調査し．それらに基づいて与えられたスロットを選別し，マルウェア名の特定と未来予測をする発見的手法を提案する．

マルウェアの感染活動に関する調査結果が報告されているが [3]，最近のボットでは機能ごとにマルウェアが分割されていて，複数のマルウェアを連携してダウンロードしてから不正行為を行うことがある．それらの特徴をはじめとして，UDP 感染，連携した感染などのいくつかの特徴を発見した．しかし，マルウェアの亜種が次々に作り出されるため，本発見的手法では種別特定に至らなかった．ボットネットには C&C サーバーを利用する種類があり，ハーダーの特定には C&C の検出が必要であるが難しい．そこで，ボ

表 1: 特徴量一覧

$slot$	スロット ID(0, . . . , 145)
$Time$	スロットの開始時間
P_I, P_O	総入力 (出力) パケット数 [pkt]
MZ	文字列”MZ”の出現
PE	文字列”PE”の出現
DOS	“!This program cannot be run in DOS mode.” の出現
win	“!Windows Program” の出現
N, J	文字列”NICK”かつ”JOIN”の出現
$ip1$	“#las6 * ipscan s.s.s.s dcom2 -s” の出現
$ip2$	“#last * ipscan s.s.s.s dcom2 -s” の出現
ST	ポートスキャンの種類
DL	感染の有無
MW 数	マルウェアのダウンロード数
MW	マルウェア名

ット通信検出を行うための snort[4]，BotHunter[5]，BotSniffer[6] というツールの利用を検討する．

最後に，提案手法を実験データに適用した精度を報告する．

2 解析方法

2.1 特徴量抽出

文字列検索によって感染特定を考え，Network Grep を使い MZ，PE，DOS という文字列を抽出した．

“!This program cannot be run in DOS mode.” は exe ファイルをダウンロードするときに現れる．“!Windows Program” は tftp でファイルをダウンロードするときに現れる．MZ，PE はマルウェアをダウンロードするときに現れる．NICK は接続時，このコマンドでニックネームを申請するときのコマンド，JOIN はチャンネルに入るためのコマンド，ipscan はスキャンをする命令である．

ポートスキャンのタイプを第 4 オクテットが 1 つずつ変化するものを s4 とした．スロット内における感染判定のための特徴を抽出するための項目を表 1 に整理する．

2.2 攻撃通信データ内のマルウェアとハッシュ値

攻撃通信データ 2 日分の解析結果から得た総マルウェア数は 200 個あり，そのうちユニークハッシュ値は 24 種類，ユニークなマルウェアは 13 種類であった．表 2 に整理した．(攻撃元データより抽出した数値である．)

表 4: UDP

	MW 名	MW 数
TCP	PE_VIRUT.AV	91
	PE_BOBAX.AK	4
	PE_VIRUT.AT	1
	BKDR_POEBOT.GN	30
	TROJ_AGENT.ARWZ	6
	TROJ_BUZUS.AGB	24
	WORM_ALLAPLE.IK	1
	WORM_POEBOT.AX	1
	WORM_SWTYMLAI.CD	27
	WORM_AUTORUN.CZU	3
	WORM_IRCBOT.CHZ	1
	UNKNOWN	5
UDP	BKDR_MYBOT.AH	1
	BKDR_RBOT.ASA	5

2.3 ボット通信検出の検討

ボット通信の検出においては、snort、BotHunter、BotSniffer を利用する．snort はネットワーク型 IDS である. ホスト型 IDS では対応できない攻撃も検出することが可能である．snort の活用例としては抽出したホスト名とブラックリストを照らし合わせることが出来る [7].

BotHunter はマルウェアに感染したコンピュータの通信パターンを認識し，ボットトラフィックを発見でき，ボット感染 PC を特定することができるなど，様々な機能を持ち合わせているツールである．

BotSnifer はボットネットの C&C 検出システムであり，ボットネット C&C の相関関係と類似性の特性を利用する．コンポーネントには無関係の通信を遮断するもの，怪しい HTTP と IRC 通信の検出をするものがある．IP とポートの宛先に従って，クライアントをグループに分類し，時間と空間の相関関係と類似性を調べるためのグループ分析を行い，C&C を検出する．

3 解析結果

解析して得られた各スロットの特徴を表 3 に示す．

また，全 145 のスロットの中で，マルウェアをダウンロードしている 58 のスロットがあり，UDP を使った tftp での感染は 6 スロットあった．

表 2: MW リスト

MW 名	ラベル	ユニークハッシュ数	DL 数	スキャン数 (s4)	プロトコル	通信方式
PE_VIRUT.AV	PE1	8	91	18	TCP	PULL
PE_BOBAX.AK	PE2	1	4	4	TCP	PULL
PE_VIRUT.AT	PE3	1	1		TCP	PULL
BKDR_MYBOT.AH	BK1	1	1	6	UDP	PULL
BKDR_POEBOT.GN	BK2	1	30		TCP	PULL
BKDR_RBOT.ASA	BK3	4	5		UDP	PULL
TROJ_AGENT.ARWZ	TR1	1	6		TCP	PULL
TROJ_BUZUS.AGB	TR2	1	24		TCP	PULL
WORM_ALLAPLE.IK	WO1	1	1		TCP	PUSH
WORM_POEBOT.AX	WO2	1	1		TCP	PULL
WORM_SWTYMLAI.CD	WO3	1	27		TCP	PULL
WORM_AUTORUN.CZU	WO4	1	3		TCP	PULL
WORM_IRCBOT.CHZ	WO5	1	1		TCP	PULL
UNKNOWN	UK	1	5		TCP	PULL

表 3: スロットと各種特徴量

$slot$	P_I	P_O	MZ	PE	DOS	$NICK$ /$JOIN$	$ip1$ /$ip2$	ST (s4)	感染	MW
0	276	17774	9	13	3	3		1	1	PE1, TR2, WO3
1	61	352	0	4	0				0	
2	7488	178491	10	16	3	3	ip2*1	1	1	WO1, PE1TR2, WO3
3	350	240148	12	10	4	3	ip2*1	1	1	PE1, TR2, WO3, PE1
4	2	55	0	0	0				0	
5	5	59	0	0	0				0	
14	354	135725	9	10	3	3	ip1*3	1	1	BK1, TR2, WO3
46	379	791	0	0	0				1	BK2
83	571	74286	15	15	5	1		1	1	PE1*2, TR2, WO3
139	450	96211	13	18	3	1		1	1	PE2, WO4, WO3
total	44452	3038276	691	966	219	60	33	28	58	200
ave	306.57	20953.63	4.77	6.66	1.51	0.41	0.23	0.19	0.4	1.38

そのうちマルウェア名は 5 スロットが BKDR_RBOT.ASA で，残り 1 スロットは BKDR_MYBOT.AH であった．ダウンロードごとに表 4 にまとめた．また，exe ファイルは全て同じであった．UDP を使った tftp での感染の場合，TCP で exe ファイルをダウンロードする時に現れる，“!This program cannot be run in DOS mode.” という文字列はなく，“!Windows Program” という文字列が現れる．

図 1 は TFTP でのダウンロードのスロットの出力パケット数を表している．UDP では TCP に比べダウンロード時の 1 秒間あたりののパケット数が少なく，長い時間をかけてダウンロードする傾向がある．

これらの解析結果から発見的手法の Rule を導き出した．

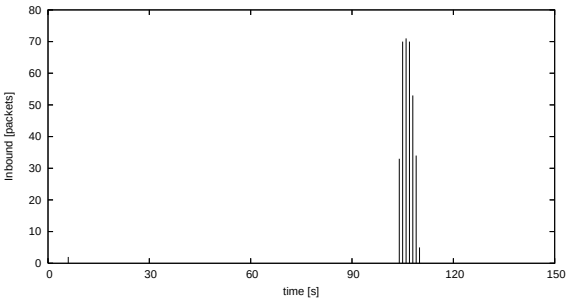

図 1: 入力パケット数 (UDP)

3.1 ポートスキャンに関する発見的手法

発見的手法の Rule は感染判定のための解析から発見した規則であり，感染判定，マルウェア名同定，未来予測に用いる．

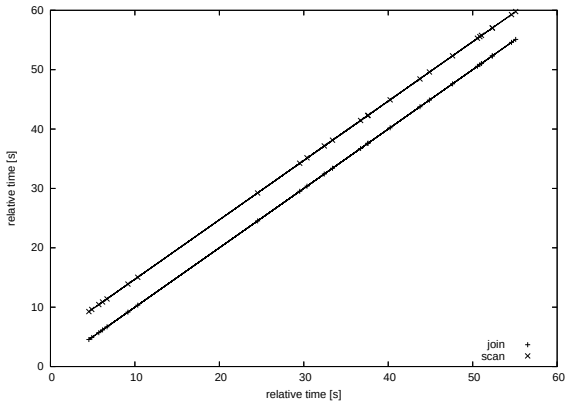

図 2: JOIN とスキャン開始時間の遅延

表 5: 連携感染パターン

パターン	スロット数
PE1 → TR2, WO3	17/58
BK1 → TR2, WO3	5/58
PE2 → WO4, WO3	4/58

コマンド”JOIN”が送られてからポートスキャンが起きるまでの時間差を図 2 に示す．X 軸は JOIN の時間，Y 軸は JOIN とポートスキャンのタイムラグ (時差) を表している．この場合のポートスキャンのタイプは第 4 オクテットを 1 ずつ増加させる s4 タイプであった．発見した s4 のポートスキャン 26 のすべてで JOIN に対しスキャン開始時間が正確に遅延している事が分かる．開始時間の分だけを取り出し，時間は無視して 2 つのイベントの差を可視化している．

Rule 1a. NICK，JOIN は NICK が 3 回現れた後，JOIN も 3 回現れる．

Rule 1b. 1 つめの JOIN から約 5 秒後にポートスキャンを開始する．

Rule 1c. ポートスキャンは 1 秒間に 256 パケットの通信を連続して行う．

3.2 連携感染に関する発見的手法

スロット内に複数のマルウェアを感染する時に特徴があった．全 145 のスロットの中で，マルウェアをダウンロードしている 58 のスロットがあり，共通したダウンロードパターンがあった結果を表 5 に示す．MW 名は表 2 を元にしている．

表 6 に各パターンの代表的なスロットを示す．

表 7: DL とスキャンの IP

slot	*DL* の発信元 *IP*	スキャンあて先 *IP*
0	124.86.C1.D1	124.86.E1.F1
2	124.86.C2.D2	124.86.E2.F2
3	124.86.C2.D2	124.86.E2.F2
16	114.145.C3.D3	114.145.E3.F3
29	114.164.C4.D4	114.164.E4.F4
例	A.B.C.D	A.B.E.F

Rule 2a. PE_VIRUT.AV をダウンロードしたあとに WORM_SWTYMLAI.CD と TROJ_BUZUS .AGB を同時刻にダウンロードを開始する．

Rule 2b. WORM_SWTYMLAI.CD と TROJ_BUZUS .AGB のあて先 IP は同じ．

Rule 2c. WORM_SWTYMLAI.CD と TROJ_BUZUS .AGB は 80 番を使い，PE_VIRUT.AV は 5 桁のポート番号使う．

Rule 2d. PE_VIRUT.AV の第 1，2 オクテットが同じあて先 IP でポートスキャンを開始する．

1 つ目のマルウェアをダウンロードしたあとに，次にマルウェアをダウンロードするまでの時間を時差とした．このダウンロードの時差をパターンごとに表 8 にまとめた．また，PE_VIRUT.AV のダウンロード時，あて先 IP は異なるものが多いが，WORM_SWTYMLAI.CD と TROJ_BUZUS.AGB のあて先 IP は全てのスロットで同じであった．PE_VIRUT.AV をダウンロードする時のあて先 IP の関係を表 7 にまとめた．

Rule2c は，表 6 に示すポート番号の関係から裏づけられる．

Rule2d は，表 7 に示すダウンロードとポートスキャンの関係に見られる振る舞いである．ダウンロード後に PE_VIRUT.AV の第 1，2 オクテットが同じあて先 IP でポートスキャンを開始している．

3.3 マルウェアの種類に関する発見的手法

通信方向には 2 種類あり，PULL では，脆弱性に対する攻撃成立後に，ハニーポットからダウンロードホストにマルウェア検体を要求する．PUSH では，C&C サーバからの命令によりポートを開き，ダウンロードホストからマルウェア検体が転送されるのを待つ．攻撃通信データには PUSH でのマルウェアのダウンロードは Windows XP で 1 スロット，Windows

表 6: 連携感染例

スロット	時間	$srcIP$	$dstPort$	MW 名
0	0:02:11	124.86.A1.B1	47556	PE_VIRUT.AV
0	0:03:48	67.215.C1.D1	80	TROJ_BUZUS.AGB
0	0:03:48	72.10.E1.F1	80	WORM_SWTYMLAI.CD
2	0:36:46	124.86.A2.B2	33258	PE_VIRUT.AV
2	0:36:52	72.10.E1.F1	80	WORM_SWTYMLAI.CD
2	0:36:52	67.215.C1.D1	80	TROJ_BUZUS.AGB
3	0:46:56	124.86.A2.B2	33258	PE_VIRUT.AV
3	0:48:52	67.215.C1.D1	80	TROJ_BUZUS.AGB
3	0:48:52	72.10.E1.F1	80	WORM_SWTYMLAI.CD
16	5:17:25	114.145.A3.B3	15224	PE_VIRUT.AV
16	5:18:37	67.215.C1.D1	80	TROJ_BUZUS.AGB
16	5:18:38	72.10.E1.F1	80	WORM_SWTYMLAI.CD

表 8: 多重感染パターン

		スロット例	スロット回数	平均時差	標準偏差
パターン 1	PE1 → TR2，WO3	0，2，3，16，29，30，50...	17	127.24	158.75
パターン 2	BK1 → TR2，WO3	14，55，56，125，126	5	176.4	147.36
パターン 3	PE2 → WO4，WO3	66，139，140，141	4	253.25	176.25

2000 で 1 スロットの 2 スロットあり，マルウェアをダウンロード後，図 3 のように一定の割合で通信を続ける特長があった．図 3 では 1 秒間に 20 から 30 パケットの通信を 5 秒間行った後，通信を止めまた，1 秒間に 20 から 30 パケットの通信を 5 秒間行うという特徴が見られた．平均 8.34 パケットの通信が行われた．この特徴から，PULL と PUSH を判別できる．

Rule 3a. PUSH でのマルウェアのダウンロードは，一定の割合で行う．

Rule 3b. 文字列”MZ”かつ”PE”を含むならばTCPによる感染である．

Rule 3c. PUSH のダウンロードは，WORM_ALLAPLE である．

Rule 3d. UDP で win という文字列があれば，TFTP のダウンロードである．

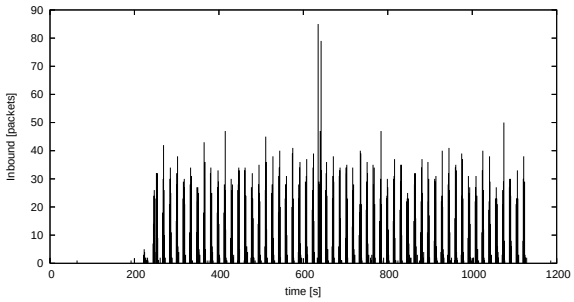

図 3: 入力パケット数 (PUSH)

3.4 感染判定のアルゴリズム

スロット表の特徴と発見的手法の Rule から，図 4 に示す感染判定のアルゴリズムを提案する．

初めにスロット内に総出力パケットが 85 パケット以上かどうかで分岐する．85 パケットはマルウェアをダウンロードしているスロットの中で最も総出力パケット数の少ない値である．85 パケット以上ある場合左に分岐する，Rule 3b，DOS，Rule 3d の分岐では文字列”MZ”，”PE”，”DOS”，”win”有無を見る．文字列”MZ”かつ”PE”がある場合は 96.8%の確率でスロット内にマルウェアを感染している．DOS の数とマルウェアの数は 95.5%の確率で一致する．Win の数とマルウェアの数は 100%一致，また tftp での UDP 感染である．（UDP 感染の場合，tftp で DL が始まる前に同じあて先 IP に TCP で連続して通信している特徴がある）感染の可能性にたどり着いた場合，NICK，JOIN が 3 回づつ現れる場合は 100%感染である．アルゴリズムは 2009 年の攻撃通信データのみを使い作成したものであり，このアルゴリズムの精度を表 9 にまとめた．

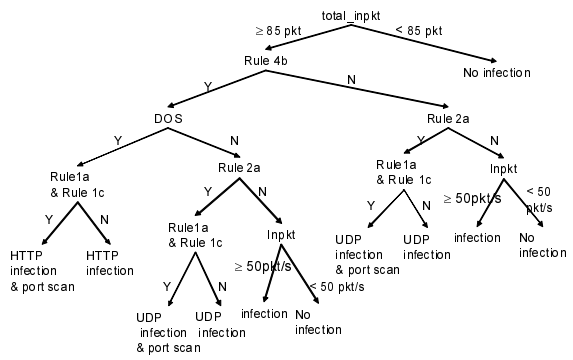

図 4: 感染を判定するアルゴリズム

表 9: 精度

判定結果 \ 真	感染あり	感染なし	total slot
感染あり	58	0	58
感染なし	0	87	87
total slot	58	87	145

3.5 その他の規則

同じマルウェアで同じ IP からスロットをまたいでダウンロードが行われる特徴が見られた．IRC セッションが成功し，ポートスキャンが行われるとダウンロードがとまった．この特徴を図 5 に示す．

同じ IP からのダウンロード間隔に特徴が見られ，IP122.18.A1.A2 のダウンロードは 289 秒と 384 秒の交互の間隔で繰り返された．想定外な事に最後のダウンロード間隔のみ大きく変化し，ポートスキャンを開始した．

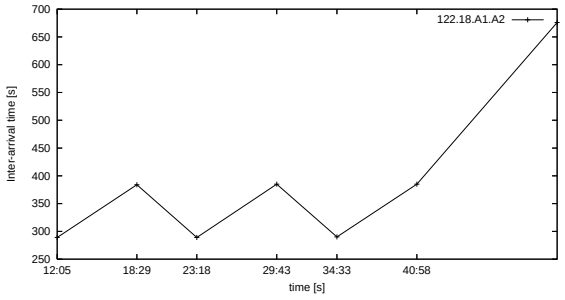

図 5: DL パターン

4 結論

本論文では，CCC DATAset 2009 攻撃通信データにおける，感染種類を判定する発見的手法を研究した．そのなかで UDP 感染，連携感染などのいくつかの特徴を発見した. マルウェアをダウンロードにもいくつか種類が存在し，Rule としてまとめた. その中で 3.3 の連携感染のように特徴が分かりやすいものもあれば，判断が難しいものも存在した．この研究ではマルウェアのダウンロードにおいて文字列検索に重点をおいて行ったが，文字列だけでは通常の通信を含めた場合の感染判定は難しい. 今後は，2.3 で述べたツールを活用して新たな特徴を検出する事により，感染判定の精度を上げていきたいと考えている．

謝辞

マルウェアのダウンロードに関する技術について助言を頂いた日立製作所の仲小路博史氏，鬼頭哲郎氏に感謝する．

参考文献

[1] 畑田，中津留，寺田，篠田，“マルウェア対策のための研究用データセットとワークショップを通じた研究成果の共有”，MWS2008，2008

[2] 小堀，菊池，寺田，“マルウェアの通信履歴と定点観測の相関について”，MWS2008，pp.67-74，2008

[3] 藤原，寺田，安部，菊池 “マルウェアの感染動作に基づく分類に関する検討”，情報処理学会，pp 177-182，2008

[4] snort，http://www.snort.org/

[5] BotHunter，http://www.bothunter.net/

[6] BotSniffer，http://www.networkworld.com/community/node/25105

[7] 東角，鳥居，“DNS 通信の挙動からみたボット感染検知方式の検討”，MWS2008，pp.13-18，2008