マルウェア通信活動抑制のためのネットワーク制御

竹森 敬祐† 酒井 崇裕‡ 西垣 正勝‡ 安藤類央* 三宅 優†

†株式会社 KDDI 研究所 〒356-8502 埼玉県ふじみ野市大原 2-1-15

‡静岡大学創造科学技術大学院 〒432-8011 静岡県浜松市中区城北 3-5-1

*独立行政法人情報通信研究機構 〒184-8795 東京都小金井市貫井北町 4-2-1

あらまし 昨今のマルウェアは，外部ホストから指令を受けるもの，新たなコードを取得して活動を変化させるもの，外部のメールサーバを利用して迷惑メールを送付するものなど，インターネット上のホストと連携した活動が行われている．マルウェアには，こうした通信要素が組み合わされた通信シナリオがあり，一部のシナリオを阻止することでマルウェアとしての本来の挙動の抑制が期待される．そこで本研究では，CCC DATAset 2009 の攻撃通信データ（CCC 攻撃通信データ）から，マルウェアの通信要素と通信シナリオに関する調査を行い，その傾向から通信シナリオを進展させない対策を提案する．そして，この対策を施したネットワークを構築し，CCC DATAset 2009 のマルウェア検体（CCC マルウェア検体）を実行させたときの通信活動の抑制の程度について評価する．これにより，ネットワーク上での効果的なマルウェア通信活動の抑制方式を明らかにする．

キーワード マルウェア通信シナリオ，ネットワーク防御，動的解析

Network-based Prevention against Malware Communications

Keisuke TAKEMORI †, Takahiro SAKAI ‡, Masakatsu NISHIGAKI ‡, Ruo ANDO*,

and Yutaka MIYAKE †

† KDDI R&D Laboratories Inc. 2-1-15 Ohara, Fujimino-shi, Saitama, 356-8502, JAPAN

‡ Shizuoka University 3-5-1 Jyohoku, Naka-ku, Hamamatsu-shi, Shizuoka, 432-8011, JAPAN

* NICT 4-2-1 Nukui-Kitamachi, Koganei-shi, Tokyo, 184-8795, JAPAN

Abstract A malware infected host receives command & control packets and new malware code from malicious servers. The infected hosts send spam mail messages via internet mail servers. A malware communication scenario is defined by a state machine that is composed of malicious packets. When the communication scenario is blocked by a router, it is expected that communication activities of the malware are restrained. In this research, we investigate both the malicious packets and the communication scenario using the CCC DATAset 2009 attack communication data, and propose a network-based prevention technique against the malware communication. Also, we apply the prevention technique to a test-bed network that monitors the malware packets and communication scenario. As the suppression level of the malware communication is evaluated, the effective prevention technique is demonstrated using the CCC DATAset 2009 malware code.

Keywords Malware Communication Scenario, Network-base Prevention, Active Monitor

1. はじめに

昨今のマルウェアは，脆弱なホストに侵入・常駐して，外部ホストから指令を受けるもの，新たなコードを取得して活動を変化させるもの，外部のメールサーバを利用して迷惑メールを送付するもの，他のホストへ感染を拡げるものなど，インターネット上のホストと連携して活動を行っている．マルウェアには，脆弱性を突く侵入フェーズ，指令や新たなマルウェアを受け取る指令･配布フェーズ,外部へ二次感染を試み

る攻撃フェーズなどが組み合わされた通信シナリオがある．よって，一部のシナリオを阻止することで，外部ホストとの連携が損なわれ，ネットワーク上での通信活動を抑制できるものと期待される．

これまで，指令を受け取る IRC 通信や攻撃パケットなどの通信要素に着目した検知手法[1-2]や，通信要素が組み合わされた通信シナリオを用いて検知する手法[3-6]が提案されている．しかし，こうして検知されたマルウェアの通信活動を抑制する手法についての検討はなされていない．ネットワーク上での攻撃対策として，スパムメールの送信規制に利用される DNS Blacklist（DNS-BL）[7]や Out Bound Port 25 Block（OP25B）[8]，フィッシングサイトへの誘導を阻止する OpenDNS[9]などがあるが，マルウェアの個々の通信要素を防ぐ技術であり，通信シナリオ全体に対する抑制効果は不明である．よって，通信シナリオの進展を阻止する適切な技術の選定と，その抑制効果について評価する必要がある．

そこで本研究では，CCC 攻撃通信データ[10]を用いて，侵入フェーズ，指令･配布フェーズ，攻撃フェーズに含まれる通信要素と，これらの組み合わせである通信シナリオの具体的な調査を行い，通信シナリオを進展させないための効果的な対策について提案する．そして，提案対策を施したネットワークを構築し，CCC マルウェア検体[10]を安全に実行したときの，通信シナリオの抑制効果について評価する．

以下 2 章において，本論文で注目するマルウェアの通信要素と通信シナリオについて整理する．3 章で，CCC 攻撃通信データの通信要素と通信シナリオに関する統計調査を行い，通信シナリオの初期に現れる重要な通信要素を明らかにして，マルウェアの通信活動の効果的な抑制手法について提案する．4 章で対策の施されたネットワーク環境で CCC マルウェア検体を実行したときの通信活動の抑制の程度について評価を行い，最後に 5 章で適切な抑制手法についてまとめる．

2. 注目する通信要素と通信シナリオ

本研究では，インターネット上の既存のルータや Firewall でも容易に実現できるマルウェア通信の抑制対策の提案を目指すこととし，パケットのヘッダ情報のみから抽出されるマルウェアの通信要素と通信シナリオについて検討する．

2.1. マルウェアの通信要素

我々は，CCC 攻撃通信データについて，ハニーポットへの侵入フェーズ，感染後の指令･配布フェーズ，外部への攻撃フェーズに含まれる，特徴的な通信要素について調査した．これは，脆弱性 Port に関する知識と，ヒューリスティックな調査を繰り返し，ハニーポットが感染したことによって発生しうる通信要素に着目したものである．結果を図 1 に示す．ここに示した通信要素は，既存の通信機器による検知とブロック制御を容易に適用できるものであり，パケットのペイロード部の詳細解析が必要な通信要素を抽出していないことに注意されたい．

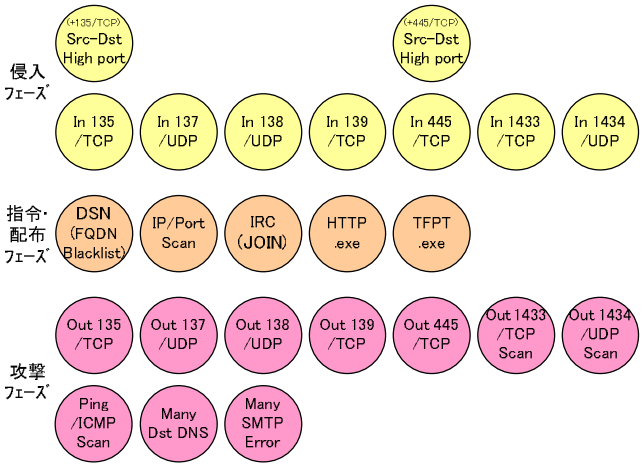


図 1. 注目するマルウェアの通信要素

【侵入フェーズ】

Windows OS の RPC や Netbios 関連の通信は LAN 上でのみ観測されるため，WAN から LAN に向けた In135/TCP, In137/UDP, …, In445/TCP は，攻撃通信とみなせる．また，殆どのクライアントホストには SQL サービスは起動していないため，In1433/TCP と In1434/UDP も攻撃通信とみなせる．尚，In135/TCP と In445/TCP には，データの送受信専用に Source/Destination Port が 1024 以上の通信チャネルを利用するモードがあり，昨今のマルウェアの中には，このモードで検体をホストに送り込むものもある．

【指令･配布フェーズ】

感染ホストは，指令を受け取るために指令サーバと通信を行う．指令フェーズの多くは IRC サービスを利用しているが，その制御コマンドの JOIN パケットに着目すると，その Source IP が感染ホスト，Destination IP が指令サーバと判断できる．また，感染ホストは新たなマルウェ

アを取得するために，配布サーバと HTTP/TFTP 通信を利用することも多い．こうした指令サーバや配布サーバのホスト名から IP アドレスへの名前解決に，DNS を利用することが多い．昨今，DNS を利用せずにインターネット上の指令サーバを自力で探索するマルウェアも多くなり，宛先 IP と Port をランダムに変化させながら，しらみつぶしに通信を試みる活動も観測される．尚，IRC による指令フェーズの中には，JOIN パケットを利用しないまま指令を受け取るものもある．この場合は，複数の指令サーバと通信を行うときには IP/Port Scan として観測される．

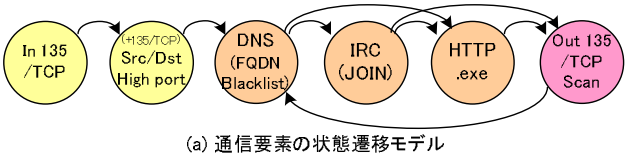
【攻撃フェーズ】

攻撃フェーズでは，脆弱性を持つ Port を狙い，多数の IP アドレスを探索する．中には Ping パケットでホストの存在を確認した後に，攻撃を試みるマルウェアもある．ところで，CCC 攻撃通信データには含まれていないが，スパムメール送信型マルウェアの場合，踏み台として利用する多数の SMTP サーバへの通信が試みられる．その際，プライマリ／セカンダリ DNS 以外にも，送信先のドメインを代表する DNS に，SMTP サーバの名前解決を要求する特徴も現れる．

2.2. マルウェアの通信シナリオ

図 1 で，マルウェアの通信要素を列挙した．ここでは，通信要素が組み合わされた通信シナリオと，その観測手法について説明する．

図 2(a)に，CCC 攻撃通信データに見られる通信シナリオの一例を示す．左の丸印から順に通信要素の出現順位となる．感染ホストの多くは，マルウェアに多重感染しており，攻撃パケットを送信しながら定期的に指令フェーズや配布フェーズを繰り返す．これにより，観測時間の経過とともに，状態遷移を表す矢印が複雑化してしまう．そこで我々は，マルウェア感染ホストの通信シナリオの統計調査を行うにあたり，複雑化する状態遷移を無視して，通信要素の出現順位のみに注目する．この様子を図 2(b)に示す．



(a) 通信要素の状態遷移モデル

In 135 /TCP	(+135/TCP) Src/Dst High Port	DNS (FQDN Blacklist)	IRC (JOIN)	HTTP . exe	Out135 /TCP

(b) 通信要素の出現順位に注目した通信シナリオ

図 2. マルウェア感染ホストの通信シナリオ

3. CCC 攻撃通信データの通信要素・通信シナリオの調査および抑制手法の提案

ここでは，マルウェアの通信シナリオに現れる通信要素の確率と，その出現順位を調査して，通信シナリオの早い段階で活動を阻止できうる通信要素の選定と，抑制手法を提案する．

まずは，CCC 攻撃通信データから，図 1 の通信要素と図 2(b)の通信シナリオを自動的に抽出する調査支援システムを実装した．このシステムは，ハニーポットの再起動周期で，通信シナリオをリセットする設計になっている．

3.1. 通信要素の調査

調査は，2009 年 3 月 13-14 日の CCC 攻撃通信データのうち，ハニーポットの再起動がほぼ定時刻に完了して，前の周期の通信要素が残っていない延べ 182 周期(=台)のハニーポットを対象にした．表 1 に，図 1 の通信要素が観測されたハニーポットの延べ台数と出現確率を報告する．

表 1. CCC 攻撃通信データに含まれる通信要素

侵入フェーズ	100% (182台)	指令･配布フェーズ	73% (133台)	攻撃フェーズ	29% (53台)
In135/TCP	87%(159)	DNS(FQDN BL)	27%(49)	Out135/TCP	16%(28)
S/D HighPort	19%(35)	IP/Port Scan	69%(126)	Out137/UDP	0%(0)
In137/UDP	2%(3)	IRC(JOIN)	17%(31)	Out138/UDP	0%(0)
In138/UDP	2%(4)	HTTP.exe	21%(39)	Out139/TCP	0%(0)
In139/TCP	41%(75)	TFTP.exe	6%(11)	Out445/TCP	0%(0)
In445/TCP	66%(121)			Out1433/TCP	0%(0)
S/D HighPort	23%(41)			Out1434/UDP	0%(0)
In1433/TCP	15%(27)			Ping/ICMP	16%(30)
In1434/UDP	14%(25)			Many DNS	0%(0)
				Many SMTP	0%(0)

【侵入フェーズ】

狙われやすい Port として，135/TCP や 445/TCP が挙げられる．これらの Port が狙われた場合，約 20%の確率で Source/Destination の High Port を利用した通信モードで，マルウェアの埋め込みも行われる．

【指令･配布フェーズ】

侵入フェーズの後に，指令･配布フェーズに至ったハニーポットは延べ 133/182 台あり，73%が指令や新たなマルウェアを受け取った．

指令サーバや配布サーバとの通信において，DNS サーバを利用して名前解決を行うマルウェアは，27%(49 台)であることがわかった．ちなみにこの 49 台のハニーポットから発信された指令サーバと配布サーバに関するホスト名は，11 種類しか観測されておらず，ホスト名(FQDN)の Blacklist を作成・管理することは容易であった．DNS を利用しないで自らインターネット上

の指令サーバを探索する IP/Port Scan は 69%(126 台)あり，DNS クエリに注目したマルウェア検知を逃れる仕組みも拡がっている．IRC(JOIN)を含む指令通信は 17%(31 台)あり，明確な指令フェーズを持つマルウェアは少ない．代わりに，HTTP や TFTP でマルウェアを受け取るものもそれぞれ 21%(39 台)と 6%(11 台)あり，マルウェア内部に感染後の挙動がハードコーディングされているものと推定される．

【攻撃フェーズ】

攻撃フェーズに至ったハニーポットは延べ 53/182 台あり，29%が二次感染活動を行った．逆に言えば，感染後に黙り込むマルウェアは 71%にものぼる．これについて，仮想マシンモニタを見抜いて停止するマルウェアがあることを想定し，著者らは安全な実マシンモニタと仮想マシンモニタを構築して，10 個の CCC マルウェア検体を実行した．その結果，わずか 1 検体でのみ，実マシン上の通信シナリオが仮想マシン上のシナリオよりも進展した．攻撃フェーズに至らないマルウェアでも，感染ホストの再起動時に自身のコピーを自動起動するように HKEY_CURRENT_USER，HKEY_LOCAL_MACHINE を改ざんしており，再起動のない動的解析システムを避ける工夫がなされている．こうしたマルウェア検知には，ホスト内部で異常なファイル操作に着目する手法が適している[11].

ちなみに攻撃は，Outbound 135/TCP の通信と，Ping/ICMP の探索活動しか観測されなかった．

3.2. 通信シナリオの調査

3.1 節と同様に，延べ 182 台のハニーポットについて，通信シナリオを調査した．その結果，最小 1 の通信要素から，最大 10 の通信要素までを持つ延べ 101 種類の通信シナリオが観測された．図 3 に，指令･配布フェーズから先へ進展した通信シナリオの一例を示す．

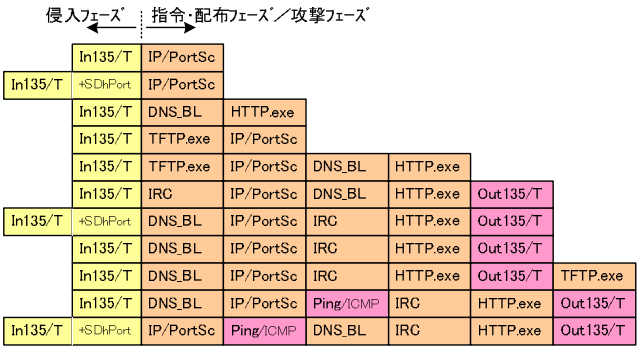


図 3.CCC 攻撃通信データに含まれる通信シナリオ

頻繁に観測される通信シナリオとして，*Port135 侵入⇒DNS 名前解決⇒IP/Port ランダム探索⇒IRC 指令受信⇒HTTP コード取得⇒Port135 攻撃* がある．たとえ侵入を許した場合でも，初期段相で出現する *DNS 名前解決*を止めることで，その後の *IRC 指令受信*や *HTTP コード取得*も阻止でき，最終的な *Port135 攻撃*も抑制できると推測される．

3.3. 通信活動の抑制手法の提案

図 3 の可能な限り左側に出現する通信要素を阻止することで，その後の通信シナリオの抑制を効果的に行える．侵入フェーズは最左に出現するのは自明であるため，指令･配布フェーズ以後の通信要素について，CCC 攻撃通信データにおける侵入フェーズを除いた，各通信要素の平均出現順位と出現確率を表 2 にまとめておく．

表 2. 指令･配布，攻撃フェーズにおける通信要素の出現順位と出現確率

通信要素	出現順位	出現確率
IP/Port Scan	1.26	95%
DNS(FQDN Blacklist)	1.69	37%
Ping/ICMP	2.17	23%
IRC(JOIN)	3.03	23%
HTTP.exe/TFTP.exe	3.28	38%
Out135/TCP	5.18	21%

3.3.1. 脆弱 Port のブロック

侵入フェーズを阻止することで，感染ホストの増加を抑えられる．早急に取り組むべき Port は，135/TCP, 445/TCP, 139/TCP である．これらの通信は LAN に限られたサービスであるため，WAN 上のルータや LAN への Gateway(GW)でのブロックが有効である．1433/TCP や 1434/UDP もブロックすべきであるが，WAN 上にはこれらのサービスを提供する Web サーバもあるため，LAN への Inbound 通信をブロックすれば良い．

次に感染した場合を想定して，LAN から WAN への Outbound 135/TCP, 445/TCP, 139/TCP Port をブロックするのも有効である．

3.3.2. 宛先数制限

宛先 IP/Port や Ping/ICMP による宛先のランダムスキャンは，WAN 上のルータでは制御が難しいが，LAN の GW に，単位時間中で接続できる宛先数を制限する装置を設けると良い．

3.3.3. 悪性ホストの名前解決ブロック

表 2 によると指令･配布フェーズ以降に至る通信シナリオの 37%に，指令･配布サーバのホ

スト名に関する DNS 名前解決が平均 1.69 位という初期段階で現れる．このため，OpenDNS に類する技術（以降，DNS-Block 方式と呼ぶ）で悪性ホストのホスト名の解決処理を阻止することが有効と言える．

3.3.4. 悪性ホスト宛て通信ブロック

指令･配布フェーズ以降に至る 23%のマルウェアは，IRC 通信が平均 3.03 位で出現する．これを阻止するために，Reputation DB と呼ばれる悪性ホストリスト[12]を用いて，WAN や LAN で指令サーバの IP アドレスへ向かう通信をブロックすれば良い（以降，IRC-Block と呼ぶ）．

38%のマルウェアは，平均 3.28 位で HTTP や TFTP で新たなコードを取得している．これについても悪性ホストリストを用いて，WAN や LAN で配布サーバの IP アドレスへの通信をブロックすれば良い．プロキシサーバを利用して，配布サーバとの HTTP と TFTP 通信をブロックする手法もある（以降，HTTP.exe-Block と呼ぶ）．

3.3.5. 未認定 Mail/DNS への通信ブロック

CCC 攻撃通信データには観測されなかったが，スパムメールの送信を抑制するためにも，LAN 上のホストは決められたプライマリ／セカンダリ DNS サーバのみと通信する制御や，OP25B のアクセス制御を，LAN の GW に適用することも有効である．

4. CCC マルウェア検体を用いた抑制手法の有効性に関する評価

ここでは，3 章で提案したマルウェア通信活動の抑制手法を想定したネットワークを構築し，その上でマルウェア検体を安全に実行させたときの抑制の程度について検証する．

4.1. 実験環境

実験用のネットワーク構成を図 4 に示す．多数の通信事業者と連携して WAN 上に提案対策を施すことは難しいため，LAN の Firewall の設定を適宜変更することで，同様の効果を期待した．実験では，Windows XP の仮想マシン上で検体を 15 分間活動させたときの通信パケットを，LAN 側で観測した．尚，評価は DNS-Block，IRC-Block，HTTP-Block について行った．

DNS-Block は，Firewall に 53/ UDP 通信を棄却するルールを設定した．*IRC-Block* は，IRC で利用する Port と指令サーバ IP への通信を棄却するルールを設定した．*HTTP.exe-Block* は，マルウェア配布サーバの IP アドレスを棄却するルールを設定した．尚，実行したマルウェアがアクセスしたマルウェア配布サーバは複数存在したため，その全てへのアクセスを棄却するルールと，50%を棄却するルール，残りの 50%を棄却するルールの 3 パターンを適用した．

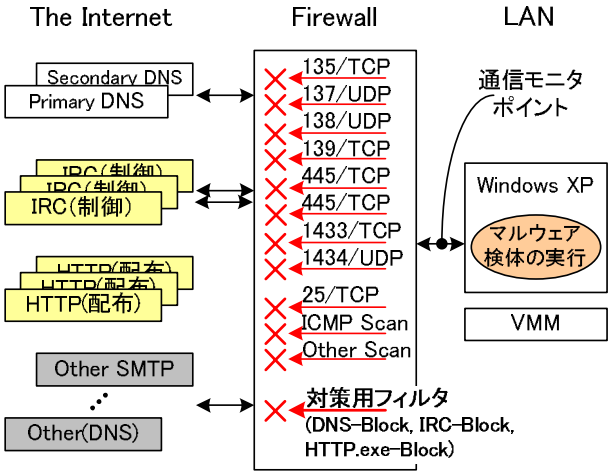


図 4. 安全性と抑制を考慮したネットワーク

4.2. マルウェア通信活動の抑制実験の結果

2009 年 9 月 4 日現在で，10 個の CCC マルウェア検体のうち，図 4 のネットワーク上で指令･配布フェーズ以降へと通信シナリオが展開した検体は 5 個あり，その通信シナリオは 3 種類に分類された．この 3 種類の検体について，DNS-Block，IRC-Block，HTTP.exe-Block を個々に適用して実行したときの，通信活動の抑制の様子について観測した．その結果を表 3 に示す．

表 3. マルウェア通信活動の対策と抑制の程度

(a) マルウェア検体A

	DNS BL	IRC (指令)	HTTP (exe)	SMTP (スパム)	Other Scan	Total
対策無	372	71	2862	156	1164	4625
DNS-Block	170	0	0	0	0	170
IRC-Block	46	270	0	0	0	316
HTTP.exe 100% Block	8	79	24	0	0	111
HTTP.exe 50% Block (1)	4	74	12	0	0	90
HTTP.exe 50% Block (2)	361	79	1001	158	803	2402

(b) マルウェア検体B

	DNS BL	IRC (指令)	HTTP (exe)	445/TCP Scan	Total
対策無	6	58	10	1376	1450
DNS-Block	154	0	0	0	154
IRC-Block	8	318	7	0	333
HTTP.exe 100% Block	6	44	11	1936	1997
HTTP.exe 50% Block (1)	6	49	11	1776	1842
HTTP.exe 50% Block (2)	6	50	10	1738	1804

(c) マルウェア検体C

	DNS BL	HTTP (exe)	445/TCP Scan	Total
対策無	24	599	1705	2328
DNS-Block	30	0	0	30
IRC-Block	26	471	1982	2469
HTTP.exe 100% Block	12	48	0	60
HTTP.exe 50% Block (1)	6	36	0	42
HTTP.exe 50% Block (2)	6	178	1738	1922

【DNS-Block】

全ての検体において，指令･配布サーバへの DNS 名前解決をブロックすると，その後の指令･配布フェーズと攻撃フェーズの全てを抑制できることがわかる．よって，DNS 名前解決を伴うマルウェアに対しては，悪性ホストリストを用いた DNS-Block を施すことで，通信活動を効果的に封じ込めることができる．

【IRC-Block】

検体 A と B については，指令サーバへの IRC 接続を繰り返し試みるため IRC パケットは増加するが，その後の HTTP 通信によるコード取得やスパムメール送信，他ホスト探索が停止する．検体 C については，IRC-Block は効果を発揮しないことがわかる．

【HTTP.exe-Block】

検体 A については，HTTP.exe 100%Block と HTTP.exe 50%Block(1)では，スパムメール送信と他ホスト探索が停止している．しかし，HTTP.exe 50% Block(2)では，スパムメール送信と他ホスト探索を担う検体の取得に成功したことが推測され，本来の通信活動を取り戻している．このことより，スパムメールの宛先や本文は，IRC 通信ではなく，検体中にハードコーディングされていることがわかる．検体 B については，HTTP.exe-Block によって，新たな検体の取得を阻止しても，実行された検体自体が Out445/TCP 攻撃まで行うため，通信活動の抑制には寄与しない．検体 C については，HTTP.exe 100%Block と HTTP.exe 50%Block(1)では，Out445/TCP Scan を抑制できている．しかし，HTTP.exe 50%Blcok(2)では，Out445/TCP Scan を担うマルウェアを取得できたものと推測され，通信活動を抑制できなかった．

4.3. 考察

DNS で名前解決を行う検体に対しては，悪性ホストリストを用いた DNS-Block 方式が，指令･配布フェーズと攻撃フェーズの全てを抑制できるため，最も効果的な方式であることがわかった．指令フェーズを持つ検体については，悪性ホストリストを用いた IRC-Block 方式を適用することで，その後の配布フェーズや攻撃フェーズを効果的に抑制できることもわかった．また，悪性ホストリストによる WAN やプロキシによる HTTP.exe-Block 方式では，多数の配布サーバのうち，スパムメール送信や外部攻撃を担うマルウェアの配布を阻止できた場合にのみ，通信活動の抑制が期待される．よって，新たなマルウェア配布を阻止する悪性ホストリストの網羅性を高める必要があり，効果には疑問が残る．

5. おわりに

本論文では，CCC 攻撃通信データから，マルウェアの通信要素と通信シナリオに関する調査を行い，その傾向を把握した．また，通信シナリオを進展させないネットワーク側での対策を提案し，CCC マルウェア検体を安全な環境で実行させたときの通信活動の抑制の効果について評価を行った．その結果，DNS で悪性ホストの名前解決を阻止する手法が最も効果的にマルウェアの通信活動を抑制できることがわかった．

文　献

[1] Jan Goebel, “Rishi: identify bot contaminated hosts by IRC nickname evaluation,” USENIX, Proc. of HotBots’2007, Apr., 2007.

[2] Ping Wang, Sherri Sparks, and Cliff C. Zou, “An Advanced Hybrid Peer-to-Peer Botnet,” USENIX, Proc. of HotBots’2007, Apr., 2007.

[3] Guofei Gu, Junjie Zhang, and Wenke Lee, “BotSniffer: Detecting botnet command and control channel,” Internet Society, Proc. of NDSS 2008, Feb. 2008.

[4] Guofei Gu, Phillip Porras, Vinod Yegneswaran, Martin Fong, and Wenke Lee, “BotHunter: Detecting Malware Infection Through IDS-Driven Dialog Correlation,” USENIX, Proc. of 16th USENIX Security Symposium, 2007.

[5] 阿部義徳，田中英彦，“C&C セッション分類によるボットネットの検出手法の一検討”，情処，FIT2007, L-033, 2007 年 9 月．

[6] BotHunter: http://www.bothunter.net/

[7] DNSBL: http://www.dnsbl.info/

[8] OP25B 連絡会: http://www.op25b.jp/isp/

[9] OpenDNS: http://www.opendns.com/

[10] 畑田充弘，中津留勇，寺田真敏，篠田陽一，"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"，情処，MWS2009, 2009 年 10 月．

[11] 酒井崇裕，竹森敬祐，安藤類央，西垣正勝，“侵入挙動の反復性によるボット検知方式”，情処，MWS2009, 2009 年 10 月．

[12] MalwareDomainList: http://www.malwaredomainlist.com/