IoTマルウェアと通信可能な悪性サーバの探索

篠宮 一真[1]　山村 翔[1,2]　荒木 翔平[1]　張 一凡[1]　胡 博[1]　神谷 和憲[1]　谷川 真樹[1]　浜田 泰幸[3]
高橋 健司[3]

概要：サイバー攻撃において，攻撃者は作成したマルウェアを感染させたホストに対し，C&C (Command and Control) サーバなどの悪性サーバを通じて操作することで任意の攻撃を成立させる．一般的な防護策としてブラックリストによる通信検知・遮断があるが，攻撃者は複数のサーバを用意し，短期間に使用するサーバを切り替えることで対策している．そのため攻撃者の通信検知回避に追従するためには，いち早い新規に用いられる悪性サーバの発見と，使用されていた悪性サーバの撤廃の確認が重要である．本稿では，近年増加が顕著なIoTマルウェアを対象に，インターネット上に存在する悪性サーバを効率的に発見する手法の検討を行う．マルウェアを動的解析し，悪性サーバとの通信時に送出するパケットのペイロードを乗せたプローブと，それに対するレスポンスから悪性サーバを検知するためのシグネチャを作成し，探索を行う．大規模に収集したxFlowを分析することで悪性サーバの可能性が高い宛先を優先的に探索し，近傍探索では発見できない悪性サーバを効率的に発見できることを示す．

キーワード：プロービング，C&Cサーバ，マルウェア，IoT，ハニーポット

Internet-wide Probing for Malicious Servers Communicable with IoT Malware

Kazuma Shinomiya[1]　Natsuru Yamamura[1,2]　Shohei Araki[1]　Iifan Tyou[1]　Bo Hu[1]
Kazunori Kamiya[1]　Masaki Tanikawa[1]　Yasuyuki Hamada[3]　Kenji Takahashi[3]

Abstract: The majority of cyber attacks is executed by controlling a number of hosts compromised with malware through malicious servers. Blacklisting the malicious servers is a effective countermeasure, however, attackers prepare multiple servers and switch to another one to help being detected. To follow this movement, it is significant to discover servers that attacker uses afresh and confirm obsolete servers. We propose a method for probing the Internet for malicious servers efficiently. The probing is performed with transmitted packet which includes payload transmitted in malware's communication with a malicious server, and signature to detect the response. Our method probes for malicious servers not found by neighbors-probing utilizing xFlow data, that was collected in a large-scale network.

Keywords: Probing, C&C server, malware, IoT, Honeypot

1. はじめに

情報通信技術の発展とともに，サイバー攻撃は増加および巧妙化の一途を辿っており，社会的な脅威となっている．攻撃者は作成したマルウェアを感染させたホストに対し，C&C (Command and Control) サーバなどの悪性サーバを通じて任意の攻撃を成立させる．そのためマルウェアへの感染を発端としたサイバー攻撃に対抗するための技術として，悪性サーバ検知はセキュリティ分野において重要であり，多くの研究が行われてきた．

既存の悪性サーバ検知はHIDSやアンチウイルスソフトが導入されたホスト，NIDSやハニーポットが設置された

[1] NTTセキュアプラットフォーム研究所
NTT Secure Platform Laboratories
[2] 警察大学校
National Police Academy
[3] NTT Security

ネットワークの挙動やマルウェアを解析する手法が主流である．これらは観測者が設置したシステムを用いて攻撃を待ち受ける手法であり，受動的な観測手法であるといえる．受動的な観測手法では，取得できる情報はあるホストやネットワークにおいて実際に取得することに成功したものに限定されていることから，無数に存在する悪性サーバすべての情報を観測することができない．この問題を解決するためには，IDS やハニーポット等の観測装置の数を増やすことが考えられるが，そのコストは大きく，増設してもすべての悪性サーバを発見することは不可能である．

このような状況のもとで，能動的な観測手法の研究が注目されつつある．能動的な観測手法は積極的に任意のネットワーク空間やホストに対して特定のパケットを送信してその応答を分析することで情報収集を行う手法であり，受動的な観測手法と比べコストが低く，観測範囲も限定されないという利点がある．しかし，観測するための能動的な行為によって法律的および倫理的問題を引き起こさないよう十分に考慮した設計を行う必要性がある．

本研究では，適切な実験設計のもと，能動的な観測手法であるプロービング (probing) を用いて，近年その増加が顕著となっている [1]，マルウェアに感染した **IoT** 端末と通信を行う悪性サーバを発見する手法の検討を行う．

2. 関連研究と本研究の新規性

2.1 能動的な観測の関連研究

Zmap [2] は実行環境が整えば 45 分以内ですべての IPv4 アドレスのポートスキャンが可能なネットワークスキャン方法を提案し，スキャンツールは OSS として公開されている．Censys [3] は Zmap を用いて特定のサービスを提供していることを確認したホストに対し，アプリケーションスキャナの ZGrab を用いて設定情報等を収集・分析し，その結果をインターネット上に公開している．これらの研究では高速に情報収集ができる点が優れているが，収集できる情報は限られている．

BotProbe [4] は IRC 通信において特定のコマンドを送ることでチャット相手が人間であるかボットであるかを判別し，C&C サーバと通信をする可能性がある端末を検知する手法を提案した．PeerPress [5] は P2P でマルウェアと通信する端末の LAN 内探索を，マルウェアの静的解析をもとに行っている．

CyberProbe [6] はマルウェアの動的解析を基にフィンガープリントを生成し，IPv4 空間の中からマルウェアへの命令の送信等の通信を行う悪性サーバを探索し，未知悪性サーバを含む悪性サーバを発見する手法を提案した．Zmap の高速なスキャン手法を取り入れ，BotProbe や PeerPress のような特定のプロトコルではなく，汎用的に対応可能としている．

2.2 IoT 端末に感染するマルウェア

近年，世の中の様々な物がネットワークに接続されるようになり，この状態は IoT (Internet of Things) と呼ばれている．インターネット接続を有する組込み機器は IoT 機器と呼ばれ，その数は 2020 年には 200 億台に上ると予測されている [1]．それに伴い，IoT 機器に感染するマルウェアの数やその活動数も急増しており，感染した IoT マルウェアは悪性サーバを通じた攻撃者からの指令により分散型サービス拒否攻撃 (Distributed Denial of Service; DDoS) 攻撃等に利用される [7].

2.3 関連研究の課題と本研究の新規性

CyberProbe は一般の PC 端末に感染するマルウェアを対象に悪性サーバを探索する手法を提案した．この研究ではプロトコル非依存であるとされているが，HTTP を重視する手法であり，HTTP 以外のプロトコルを用いる悪性サーバの発見には至っていない．またフィンガープリント生成時にマルウェアの送出パケットをすべて送出しており，インターネット上のホストへ悪影響を及ぼしている可能性がある．

本研究では広域なネットワークの中から未知の悪性サーバを発見するという観点から CyberProbe を参考に設計し，近年の急増に伴いその対策が急務となっている IoT マルウェアを対象に，感染端末が通信し得る悪性サーバの効率的な探索手法を提案する．特定のプロトコルを重視せず，プロトコル非依存で悪性サーバを発見するためのフィンガープリント生成について検討する．また，研究倫理面での検討を慎重に行い，提案手法において法律的および研究倫理的にリスクが最小限になるような手法を提案する．さらに，探索手法に関して，従来の既知悪性サーバの近傍探索に加え，近傍外の探索において，大規模なネットワークフローデータを活用した探索手法を提案する．

3. 提案手法

3.1 概要

本研究は IoT マルウェアの取得からそのマルウェアが通信可能な悪性サーバをプロービングによって発見するまでを一つのシステムとして設計し，評価をする．プロービングとは，プローブ (probe) と呼ばれる調査用パケットを送出し，それに対するレスポンスからプローブの送信先が任意の特性を有するか否かを判定する探索手法である．なお本稿では，任意のサービスの疎通確認をするものをスキャン，特定の調査を行うためのペイロードを乗せたパケットを送信しその応答を見るものをプロービングと呼ぶ．

提案手法の概要を図 1 に示す．本研究では IoT マルウェアの収集から IoT マルウェアと通信可能な悪性サーバの探索までを自動で行うシステムを提案する．攻撃者は攻撃に用いるサーバの IP アドレスやドメインを一定の期間ごとに変更することで検知を回避する．そのため，マルウェア

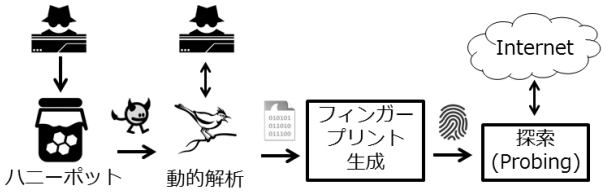


図 1　提案手法の概要

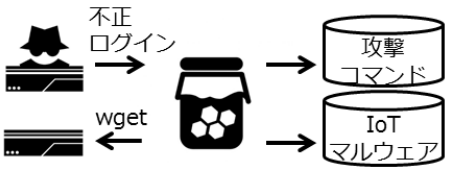


図 2　IoT ハニーポットの概要

収集から探索を自動化することにより，攻撃者が検知回避を施す前に素早く悪性サーバを発見することができる．提案システムは IoT マルウェアの収集，フィンガープリント生成，探索の 3 つの機能に大別される．以下，各機能について説明する．

3.2　IoT マルウェアの収集

本稿では IoT ハニーポットに SSH および Telnet ハニーポットの cowrie [8] を用いて攻撃を観測・収集する．図 2 に IoT ハニーポットの概要を示す．cowrie はシェルの対話を記録するように設計されており，攻撃者による実際のコマンドを記録する．このコマンドを実際に実行し攻撃者のマルウェア配布サーバからマルウェアをダウンロードする．

3.3　フィンガープリント生成

IoT ハニーポットで収集したマルウェアをサンドボックスの cuckoo [9] 上で動的解析を行った際の悪性通信データ用いてフィンガープリントを生成する．本節では悪性通信データから，悪性サーバのフィンガープリントを生成方法について述べる．フィンガープリントとは一般的に同一性を確認するための値や手法を指し，本研究では任意のサーバがマルウェアと通信可能である悪性サーバであることを確認するために，送信する調査用のパケット (プローブ) とその応答から結果を判定するためのシグネチャの組をフィンガープリントとする．

3.3.1　Request Response Pairs の抽出

まず，端末からのリクエストとサーバからのレスポンスのペアである Request Response Pairs (RRP) を抽出する．ここでは同一 IP アドレス・ポートを使用し，送受信が 3 秒以下の間隔で行われるリクエストとレスポンスの集合のことを RRP と呼ぶ．なお，マルウェアは疎通確認等のために正常な宛先へ通信を行うことがあるため，Alexa Top Sites [10] の上位 10 万のドメインとの通信は取り除く．先行研究 [6] ではこの後，この RRP ごとにすべてのリクエストパケットを複数の宛先に送信して応答パターンを複数得ることで検知漏れ (False Negative) の低減を図っている．しかし，これは攻撃行為や不法行為を引き起こす可能性があると考えられるため本研究では実施しない．

3.3.2　クラスタリング

本提案手法では，プロトコルに依存せずにフィンガープリントの生成を行うため，宛先 IP アドレス，パケット長，各 RRP のリクエストの IP ペイロードの byte 値列を特徴量とする．クラスタリングのアルゴリズムは DBSCAN [11] を用いる．クラスタリングにより，マルウェアが行う通信活動ごとに RRP が分類され，スキャン等の悪性サーバ以外との通信と，C&C サーバからの命令の送受信や，悪性ファイル配布サーバからの悪性ファイルのダウンロードを行うものを区別することができる．

3.3.3　フィンガープリントの選定

クラスタリングによって出力された各クラスタの中の代表となる一つの RRP を抽出し，研究倫理上の観点から，(1) 可読性があり，(2) 目視によって攻撃性やログインする挙動等が無いことができたものをフィンガープリントとする．フィンガープリントはリクエストとレスポンスのペイロード部分から成り，前者をプローブのペイロード部分として送信し，それに対する応答と後者とのマッチングにより，悪性判定を行う．

3.4　探索

フィンガープリントを用いて，実際に調査用のパケットであるプローブを送信し，その応答とシグネチャとのマッチングにより悪性サーバであるか否かを判定する．本稿では実際に IPv4 空間の探索を行い，探索効率や研究倫理面から探索範囲の決定方法について考察する．

3.4.1　全探索

探索方法として最も単純なものに，IPv4 アドレスの内，BGP によって広告されているアドレスすべてを探索する全探索がある．全探索では IPv4 空間において探索可能なすべての悪性サーバの発見が期待できるが，探索の必要があるホスト数は約 26 億存在するため，一度の探索に必要な時間が長いことや，ネットワークへの負荷，正常サーバへのプローブ送信数が大きくなるという研究倫理的問題がある．そのため本稿の実験においては研究倫理面に配慮し，全探索は実施しない．

3.4.2　近傍探索

本提案手法ではマルウェアの解析を伴うため，その解析の段階で悪性サーバの IP アドレスが少なくとも一つは取得可能である．近傍探索は，このシードとなる既知 IP アドレスの周辺を探索するものであり，攻撃者は保有する複数の悪性サーバを同じホスティングサービス内や ISP のネットワーク内に設置傾向がある [6] ため，近傍外探索よりも効率的と考えられる．近傍探索における探索範囲の決定は BGP の経路情報を利用し，同一アドレス帯探索と同

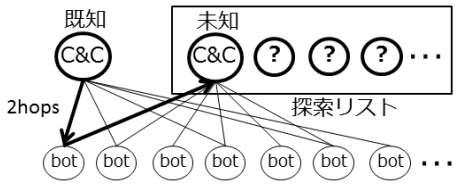


図 3　2hop 先探索の調査対象

一組織探索の 2 種類の近傍探索を実施する．

3.4.2.1　同一アドレス帯探索

既知悪性 IP アドレスが含まれる経路情報を取得し，既知悪性 IP アドレスが所属する IP アドレス帯を探索範囲とする．

3.4.2.2　同一組織探索

既知悪性 IP アドレスが含まれる経路情報を取得し，description 情報から組織名を得る．そして，BGP 経路情報から description として同じ組織名を持つものを抽出し，それらの IP アドレス帯をすべて足し合わせたものを探索範囲とする．

3.4.3　近傍外探索

既知悪性 IP アドレスの近傍外の探索は近傍探索に比べ範囲が広大であるため，効率的に探索するには工夫が必要となる．本稿では近傍外探索において悪性サーバを効率的に探索する 2hop 先探索を提案し，比較のために無作為探索を実施する．

3.4.3.1　大規模ネットワークフロー情報を活用した 2hop 先探索

大規模なネットワークで取得されたフロー情報を活用し，実際に過去に行われた通信から調査対象を絞りこむ．図 3 に 2hop 先探索の調査対象決定方法を示す．ネットワーク上に存在する端末をノードとしたグラフで表現したとき，既知悪性サーバの 1hop 先はマルウェアに感染しボット化した端末である可能性が見込まれる．そして，そのボットからさらに 1hop 先にある端末は同一攻撃者の別の悪性サーバである可能性が見込まれる．そのため，既知悪性サーバの 2hop 先のホストを調査対象とすることで未知悪性サーバ発見を効率化することができる．本稿ではシードとなる悪性 IP アドレスとして 2018 年 8 月にハニーポットで入手した IP アドレスを用いる．

3.4.3.2　無作為探索

提案手法との比較対象に無作為探索を用いる．任意の探索範囲から無作為 (ランダム) に探索対象を抽出することで，探索範囲を限定せずに任意のホスト数を探索可能であるが，網羅的な探索はできない．

3.4.4　ポートの開閉状況調査

調査対象の IP アドレスに対して，マルウェア動的解析時に使用していた宛先ポートが開いているかを調べる．この段階において，サービスが利用可能な端末のみにプローブを送信することによって，探索に必要な通信量を抑えるとともに，正常な端末へのプローブ送信を抑えることができる．

表 1　フィンガープリント中の特徴的文字列

フィンガープリント	特徴的文字列
I	BUILD DONGS\n
II	BUILD B00G4YMAN : x.x.x.x\n
III	Connected ->x.x.x.x ->x86_32\n
IV	BUILD STREAMS\n

3.4.5　プローブ送出の制限

プローブ送出は，最大送信レートと最大同時コネクション数に制限をかけ，プローブ送信先に負荷をかけないよう配慮する．最大送信レートと最大同時コネクション数はそれぞれ 1Mbps，100connections/s とし，予備実験によって現実的な時間内に探索が完了することを確認しており，先行研究 [6] の 26Mbps，400connections/s と比較して低い値である．

3.4.6　実装

ポートの開閉状況調査はネットワークスキャンツールの Zmap [2] を用いる．探索は python を用いて実装し，asyncio [12] ライブラリを用いて宛先ごとに非同期で悪性判定を行うことで高速化する．探索を行うホストの CPU のクロック周波数は 2GHz (1 コア)，メモリ容量は 4.0GB である．

4.　実験結果および考察

4.1　フィンガープリント生成結果

2018 年 8 月 13 日から 8 月 17 日の間に IoT ハニーポットを用いて得られた検体を用いる．なお，ログインをする機能を持つプローブは不正アクセスを引き起こす可能性があるため，プロトコルとして Internet Relay Chat (IRC) を用いるものや，可読性がなく安全性が確保できないフィンガープリントは取り除いた．その結果，一週間の実験期間に取得した 4 つの検体から 4 つのフィンガープリントを生成した．実際に生成されたフィンガープリント内でのプローブに含まれる特徴的な文字列を表 1 に示す．なお，表中では送信元 IP アドレスを x.x.x.x と表記してある．

プロービングでは，悪性サーバがこれらの文字列を認識し，プローブに対する特有な応答が送信されることを利用している．例えば，フィンガープリント I のプローブに対して悪性サーバは “SCANNER ON\n” という文字列を含む応答を返す．シグネチャマッチングではこの文字列を検知し，悪性サーバと判定した．

4.2　探索結果

2018 年 8 月 13 日から 8 月 17 日の間に探索を行った．探索方法は近傍外探索の無作為探索と 2hop 先探索，近傍探索の同一アドレス帯探索と同一組織探索である．表 2 に各探索によって発見された悪性サーバ数 / 探索ホスト数，表 3 に発見した IP アドレスを VirusTotal [13] と Censys [3]

表 2　探索結果

	近傍外探索		近傍探索	
	無作為探索	2hop 先探索	同一アドレス帯探索	同一組織探索
I	0 /218,479	2 /218,479	2 /65,536	3 /2,187,264
II	0 /218,479	2 /218,479	1 /4,096	3 /1,519,104
III	0 /218,479	1 /218,479	1 /4,096	1 /1,519,104
IV	0 /218,479	0 /218,479	2 /4,096	2 /5,888

を用いて照会した結果と，発見に用いた探索方法を示す．Loader は VirusTotal の Downloaded Files で 1 つでも悪性判定がされた場合，C&C は Communicating Files で 1 つでも悪性判定がされた場合，初めて発見された日付を記す．

近傍外探索では 2hop 探索による 5 個，近傍探索では同一アドレス帯探索による 6 個，同一組織探索による 9 個，重複を除くと合計で 13 個の悪性サーバが発見され，その内 5 個は未知のものであった．また，悪性サーバは telnet で命令のやりとりを行う TCP23 番ポートと悪性ファイルを配布する TCP80 番ポートの疎通が確認できたことから，C&C と Loader を兼ねているものが多いと考えられる．表 2 に示すホスト数を探索した場合の各探索の平均探索時間は無作為探索が 180 秒，2hop 先探索が 175 秒，同一アドレス帯探索が 16 秒，同一組織探索が 1,045 秒であった．

4.3　考察

近傍探索と近傍外探索を比較すると，近傍探索の方が悪性サーバの発見数と発見率が高いことから，一般の PC 端末に感染するマルウェアと同じように，攻撃者は既設の悪性サーバと同一のアドレスレンジや同一組織内のネットワークに別の悪性サーバを設置している．

また複数のフィンガープリントで発見可能な悪性サーバが存在することから，同一攻撃者が所有する悪性サーバが複数種のマルウェアに対応するよう設計されている可能性や，プローブの内容に依らずマルウェアに命令を送信するよう設計されている可能性が考えられる．

探索効率は同一アドレス帯探索，同一組織探索，近傍外探索の順に高く，シードである既知の悪性サーバに近いレンジの探索ほど探索効率が高い．また，悪性サーバの発見がより難しい近傍外探索に関して無作為探索と 2hop 先探索を比較すると，本研究が提案した大規模なネットワークフローデータを用いた 2hop 先探索によって，効率的に近傍外の悪性サーバを発見できていることが確認できた．

4.4　今後の課題

本稿では実験期間が 5 日間と短期間であったため少量の検体での評価となった．今後はより長期に渡る検体の確保を行い，本稿では目視で行なっていた，攻撃や不法行為発生の可能性のあるプローブの除外を自動で行う手法を検討することで，より大規模な悪性サーバ探索を行う．

また近傍外探索における本研究で提案した 2hop 先探索においては，hop 数だけでなく送信バイト数やフラグなどの共起分析によって探索効率が向上する可能性があるため，今後提案手法の更なる効率化を行う．

さらに，本稿ではマルウェアの動的解析結果を基にフィンガープリントを生成したが，静的解析による手法 [14] の併用により，発見可能な悪性サーバ数の向上を目指す．

5.　研究倫理

本研究の提案手法では，マルウェアが動的解析時に発生させたペイロードを用いてプローブを作成し，インターネット上を探索する．探索の性質上，正常なサーバへプローブを送信することは避けられないため，事前の適切な実験設計を行わなければ実ネットワークや実ホストに被害を与える可能性がある．

我々は事前に過去の類似研究 [2,6] や ICT 研究における研究倫理原則を示した Menlo Report [15]，情報処理学会倫理綱領 [16] を参考に，実験中に発生し得る問題点とそれらに対する対応策をまとめ，CSS2018 倫理相談窓口に相談し，賛同を得た後，著者所属組織 [1] における認可を受けた．

なお，本研究では探索対象のホスト数が大きいためインフォームドコンセントの実施は非現実的であると判断し，行なっていない．プロービングは機密情報の収集や攻撃を引き起こすものではなく，プローブの事前検査によりその可能性を最小化している．本研究提案手法による悪性サーバ情報は既存の攻撃検知システムへの導入が容易かつ有用であると考えており，公益性を有すると判断した．

本稿では以上から実験の妥当性を判断し，実施した．以下に発生し得る問題点と対応策を示す．

5.1　発生し得る問題点

(1) 宛先での悪影響（不正アクセス等）や，宛先を仲介した第三者への悪影響（マルウェア感染拡大，DoS 攻撃等）が発生する可能性がある
(2) 送信先ネットワークの帯域を圧迫する可能性がある
(3) ホスティングサービス上サーバからプローブ送信を行う場合，レピュテーション低下等を招く可能性がある

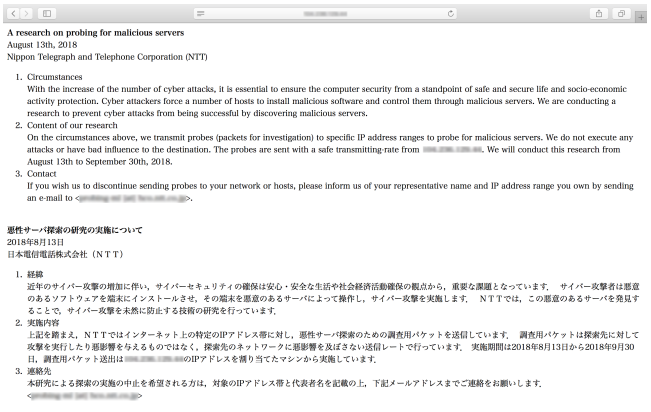
5.2　対応策

(1) 事前に送出するパケットを検査し，危害を加える可能性のあるものをフィルターする
(2) 送信レートは必要最小限に設定する
(3) パケット送信元の IP アドレス上に公開 web サーバを立て，我々の身元・研究目的等を公開し，必要があれば当該組織宛てにプローブ送出を行わない

対応策の一つとして，本稿では実験開始前に公開 web サーバによる連絡先の公開を行い，送信先からプローブ送信の中止要望等を受け付けた．図 4 に実際に公開した web ページのスクリーンショットを示す．なお，本稿の実験期間中に公開したメールアドレス宛てに連絡を受けることはなかった．

表 3　発見した悪性 IP アドレスの詳細

IP アドレス	AS 番号	フィンガープリント	Virus Total		Censys (Open Port)	近傍外探索		近傍探索	
			Loader	C&C		無作為探索	2hop 先探索	同一アドレス帯探索	同一組織探索
1	A	Ⅲ	-	-	23	×	○	×	×
2	B	Ⅱ	7/31	8/16	21, 22, 23, 80	×	×	×	○
3	B	Ⅱ	-	-	22, 25, 110, 143	×	×	×	○
4	B	Ⅰ, Ⅱ	-	-	21, 22, 23	×	○	○	○
5	B	Ⅰ	8/7	8/14	21, 22, 23, 80	×	○	×	×
6	C	Ⅱ	7/23	7/23	21, 22, 23, 80	×	○	×	×
7	B	Ⅲ	8/5	8/5	21, 22, 3306	×	×	○	○
8	D	Ⅳ	8/15	8/15	21, 22, 3306	×	×	○	○
9	D	Ⅳ	8/15	8/15	-	×	×	○	○
10	A	Ⅱ	8/10	-	23	×	○	×	×
11	E	Ⅰ	8/10	8/10	21, 22, 23, 80	×	×	○	○
12	E	Ⅰ	-	-	21, 22, 23, 80	×	×	○	○
13	F	Ⅰ	-	-	21, 22, 23, 80	×	×	×	○



A research on probing for malicious servers
August 13th, 2018
Nippon Telegraph and Telephone Corporation (NTT)

1. Circumstances
   With the increase of the number of cyber attacks, it is essential to ensure the computer security from a standpoint of safe and secure life and socio-economic activity protection. Cyber attackers force a number of hosts to install malicious software and control them through malicious servers. We are conducting a research to prevent cyber attacks from being successful by discovering malicious servers.
2. Content of our research
   On the circumstances above, we transmit probes (packets for investigation) to specific IP address ranges to probe for malicious servers. We do not execute any attacks or have bad influence to the destination. The probes are sent with a safe transmitting-rate from [redacted]. We will conduct this research from August 13th to September 30th, 2018.
3. Contact
   If you wish us to discontinue sending probes to your network or hosts, please inform us of your representative name and IP address range you own by sending an e-mail to <[redacted]>.

悪性サーバ探索の研究の実施について
2018年8月13日
日本電信電話株式会社（NTT）

1. 経緯
   近年のサイバー攻撃の増加に伴い，サイバーセキュリティの確保は安心・安全な生活や社会経済活動確保の観点から，重要な課題となっています． サイバー攻撃者は悪意のあるソフトウェアを端末にインストールさせ，その端末を悪意のあるサーバによって操作し，サイバー攻撃を実施します． NTTでは，この悪意のあるサーバを発見することで，サイバー攻撃を未然に防止する技術の研究を行っています．
2. 実施内容
   上記を踏まえ，NTTではインターネット上の特定のIPアドレス帯に対し，悪性サーバ探索のための調査用パケットを送信しています． 調査用パケットは探索先に対して攻撃を実行したり悪影響を与えるものではなく，探索先のネットワークに悪影響を及ぼさない送信レートで行っています． 実施期間は2018年8月13日から2018年9月30日，調査用パケット送出は[redacted]のIPアドレスを割り当てたマシンから実施しています．
3. 連絡先
   本研究による探索の実施の中止を希望される方は，対象のIPアドレス帯と代表者名を記載の上，下記メールアドレスまでご連絡をお願いします．
   <[redacted]>

図 4　公開した web ページ

6. おわりに

本稿では，IoT マルウェアを収集・動的解析・分析することで IoT マルウェアと通信する可能性のある悪性サーバのフィンガープリントをプロトコル非依存で生成し，そのフィンガープリントを用いてインターネット空間から悪性サーバを効率的に探索する手法の提案を行った．

研究倫理面でのリスクを最小化した実験設計をし，5 日間の実験期間で収集した 4 個の検体を用いて実験を行った結果，クレームを受けることはなく，13 個の悪性サーバを発見し，うち 5 個は未知のものであった．

IoT マルウェアを利用した攻撃においても，悪性サーバは既存悪性サーバの近傍に多い傾向や，命令の送信と悪性ファイルの配布の役割を兼ねている場合が多いことがわかった．さらに近傍外探索においては大規模ネットワークで収集したフロー情報を分析して探索先を決定することにより，効率的に悪性サーバを発見できることを示した．

参考文献

[1] Gartner. Gartner Says 6.4 Billion Connected “Things” Will Be in Use in 2016, Up 30 Percent From 2015. https://www.gartner.com/newsroom/id/3165317
[2] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. ZMap: Fast Internet-Wide Scanning and its Security Applications. USENIX Security Symposium, August 2013.
[3] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J. Alex Halderman. Censys: A Search Engine Backed by Internet-Wide Scanning. ACM Computer and Communications Security. October 2015.
[4] G. Gu, V. Yegneswaran, P. Porras, J. Stoll, and W. Lee. Active botnet probing to identify obscure command and control channels. Annual Computer Security Applications Conference, Honolulu, HI, December 2009.
[5] Z. Xu, L. Chen, G. Gu, and C. Kruegel. Peerpress: Utilizing enemies’ p2p strength against them. ACM Computer and Communications Security. October 2012.
[6] Antonio Nappa, Zhaoyan Xu, M. Zubair Rafique, Juan Caballero, and Guofei Gu. CyberProbe: Towards Internet-Scale Active Detection of Malicious Servers. Proceedings of NDSS, February 2014.
[7] Manos Antonakakis et al. Understanding the Mirai Botnet. USENIX Security Symposium, August 2017.
[8] Michel Oosterhof. Cowrie. https://github.com/micheloosterhof/cowrie
[9] Cuckoo Sandbox. https://cuckoosandbox.org/
[10] Alexa Internet, Inc. https://www.alexa.com/topsites
[11] Martin Ester et al. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. KDD-96 Proceedings. 1996.
[12] Python Software Foundation, asyncio - Asynchronous I/O, event loop, coroutines and tasks. https://docs.python.org/3/library/asyncio.html
[13] Virus Total. https://www.virustotal.com
[14] Zhaoyan Xu, Antonio Nappa, Robert Baykov, Guangliang Yang, Juan Caballero, and Guofei Gu. AUTOPROBE: Towards Automatic Active Malicious Server Probing Using Dynamic Binary Analysis. Proceedings of the 21st ACM CCS. 2014.
[15] M. Bailey, D. Dittrich, E. Kenneally, and D. Maughan. The Menlo report. IEEE Security and Privacy, 2012.
[16] 情報処理学会. 情報処理学会倫理綱領. https://www.ipsj.or.jp/ipsjcode.html