n-gram 解析と One-Class SVM を用いた IoT ボットネットワークの検知手法の提案

池田　良磨 [a)]・森　雅実 [b)]・岡崎　直宣 [c)]・山場　久昭 [d)]・油田　健太郎 [e)]

A Proposal of an IoT Bot Detection Method Using *n*-gram Analysis and One-Class SVM

Ryoma IKEDA, Masami MORI, Naonobu OKAZAKI, Hisaaki YAMABA, Kentaro ABURADA

Abstract

In recent years, IoT(Internet of Things) is expected various field, such as Industry, Medical and etc, with IoT becomes popular. On the other hand, at present, IoT security measures taken by individuals and companies are not enough. And attackers use IoT devices to cause cybercrime.
For DDoS(Distributed Denial of Service) attacks, there is a method of simultaneously transmitting packets from various devices by building a botnet. Such botnets are built by devices infected with malware. In addition, malware targeting IoT and building botnet such as "Mirai" is appearing one after another. We can be expected to suffer severe damage of botnet building malware infected with IoT because most IoT devices are vulnerable and not all general users have security knowledge and are not very interested in security incidents. In this paper, we propose IoT bot detection method using n-gram analysis and One-Class SVM, for the purpose of detecting whether the IoT device is infected so that the user can quickly deal with the malware when an IoT device is infected with malware that builds a botnet. This time, the difference between normal communication and communication with C&C(Command & Control) server when malware is infected is used for bot detection. And analyze packets with n-gram and detect outliers using One-Class SVM.

Keywords: IoT, malware, n-gram, SVM, C&C Server

1. はじめに

近年、家電製品や自動車など様々なモノがインターネットに繋がる Internet of Things(IoT) が流行し、それに伴い IoT は産業や医療のようなあらゆる場面での活躍が期待される分野となってきている。その一方で、現状個人や企業において行われている IoT のセキュリティ対策は十分なものとは言えず、攻撃者に IoT デバイスが利用されることによるサイバー犯罪に発展することが懸念される。

2016 年 10 月 26 日、米国の DNS サービスプロバイダである Dyn が大規模な DDoS(Distributed Denial of Service Attack:分散サービス妨害) 攻撃を受けたことで、様々なサイトにアクセスしづらくなるという事件が発生した。この DDoS 攻撃は「Mirai」と呼ばれるマルウェアに感染した 10 万台に及ぶ IoT デバイスによって構築されたボットネットによる攻撃であることが分かっている [1)]。

ボットネットとは、「Mirai」などのマルウェアに感染した端末同士がネットワーク上で連携して攻撃可能な体制を取り、サーバなど 1 つのターゲットに対して一斉に攻撃を仕掛ける攻撃アプローチのことである。一般的なマルウェアは PC 端末を乗っ取るが、「Mirai」の場合は IoT デバイスを標的としているのが特徴である。そして「Mirai」に感染しボット化した IoT デバイスは C&C サーバ (Command and Control Server) からの指令を受け次第、ターゲットに対して大量のパケットの送信を開始する。この攻撃を受けたサーバは負荷が増加しリソースが枯渇することで、サービスを正常に提供できない状況に陥る [2)]。

また、同年 10 月に「Mirai」の製作者として有力視されている「Anna-senpai」と名乗る人物が「Mirai」のソースコードを公開したことにより、「Mirai」の感染経路や手口が判明したが、これを機に「Mirai」の亜種が爆発的に登場し始めた [3)]。

「Mirai」は IoT デバイスにおいてデフォルトで使用されている認証情報のリストを利用して辞書攻撃により侵入を試みる。それに成功した場合、「Mirai」はメモリ上にのみ存在することになるためユーザは感染したデバイスの電源を切ることによって「Mirai」を駆除することが可能であるが、PC やスマートフォンに比べて IoT デバイスはユーザのセキュリティに対する関心や知識が低いことが考えられるため、IoT デバイスが正常に稼働し続ける限り感染したデバイスはそのまま放置されると思われる。DDoS 攻撃において IoT デバイスはあくまで攻撃の踏み台として利用されることが目的であるた

[a)] 工学専攻機械・情報系コース大学院生
[b)] 情報システム工学科学部生
[c)] 情報システム工学科教授
[d)] 情報システム工学科助教
[e)] 情報システム工学科准教授

め、仮に感染してしまっても IoT デバイスの機能を損なうことはない。その場合、ユーザは機能さえ損なわなければデバイスがマルウェアに感染したことに気が付かないだろう。これは「Mirai」に限った話ではなく他の IoT デバイスを標的としたマルウェアにも当てはまることである。

このことより本研究では、IoT デバイスがボットネットを構築するようなマルウェアに感染した際に、ユーザが迅速にマルウェアに対処を行えるようにその感染の有無を検出するシステムの提案を行う。

2. 関連研究

マルウェアに感染したデバイスを検知するシステムとして Wang らによって PAYL が提案されている [4]。PAYL はアノマリ型の検知システムであり学習フェーズとテストフェーズに分かれている。PAYL はパケットからの特徴量の抽出に $n-gram$ 解析を用いており学習フェーズでは、各ホストに対するパケットのポート、ペイロード長ごとに $n-gram$ の出現回数の平均と標準偏差を計算しそれをモデルとして記録する。テストフェーズでは、検知対象のパケットのペイロードにおける $n-gram$ の出現回数を学習フェーズと同様に計算し、そのパケットのポート、ペイロード長に該当するモデルとのマハラノビス距離を測定する。測定した値が閾値異常であれば異常として検出する。また、文献 [5] ではファイル感染型ウイルスに感染した機器が C&C サーバとの通信を行う際のパケットのペイロード情報が正常通信の場合と異なる特徴を持つことが述べられている。その異なる特徴の一つとしてペイロード内の ASCII 文字コードの出現頻度が挙げられており、$n-gram$ 解析によるペイロードからの特徴量を抽出することはマルウェアの検知に有効であると考えられる。そのため、本研究の提案システムにおける特徴量の抽出にはこの $n-gram$ 解析を採用する。

文献 [6] では、PAYL のバイト置換、パディングを用いた攻撃によって突破されてしまうという問題点を改善するために、$n-gram$ を改良した $2v-gram$ 法によって特徴量の抽出を行い、その特徴量を教師なし機械学習手法である One-Class SVM を用いて学習し検知を行うシステムの提案を行っている。この文献では PAYL において懸念される誤検知を低く抑えるために機械学習を用いており、また、$n-gram$ によるペイロードの解析との相性の良さから One-Class SVM を採用している。この $n-gram$ と One-Class SVM を組み合わせて検知システムを構築することが IoT における異常侵入検知システムにも適応できるのではないかと考える。

3. 提案手法

本研究では、$n-gram$ 解析と One-Class SVM を用いて IoT デバイスの正規通信とマルウェアに感染しボット化した後のデバイスの通信とを区別するシステムの提案を行う。

3.1 提案手法の実装環境

IoT デバイスの一般的な特徴として低コストかつ省電力であるという点がある。そのため各 IoT デバイスは最小限のハードウェアリソースしか積んでいないため、セキュリティ対策のために各々のデバイス全てに検知システムを実装することは技術的に難しく、ウイルス対策ソフトのようなツールを導入することは安価であるという利点を損ないかねない。また、国内外問わず流通している IoT デバイスから脆弱性を有する機器を完全に排除することは困難であるという点も考慮すると、検知システムの開発にあたり現実的であるのはゲートウェイなどの中継機器、及びクラウド上で実装可能なシステムであると考える。このことより、本研究ではスマートホームによる通信環境と、その中のスマートフォンや PC と IoT デバイスを接続するホームゲートウェイ上での検知システムの実装を想定する。図 1 に想定する IoT ネットワークを示す。

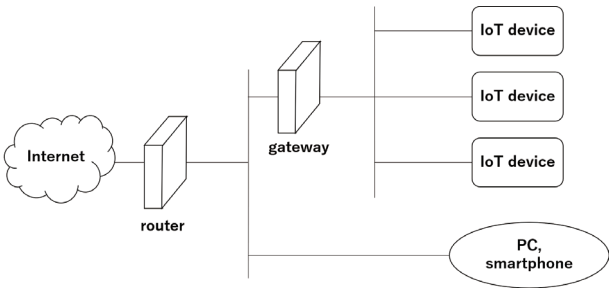

図 1. 想定する IoT ネットワーク

提案手法のシステムにはアノマリ型でのシステムの構築をする。マルウェアの検知手法は大きく分けてシグネチャ型とアノマリ型の 2 つに分類される。シグネチャ型は過去にマルウェアに感染した機器の通信パターンやバイナリをシグネチャとして記録し、このシグネチャと一致するものを異常として検出する手法である。そのためシグネチャ型検知手法はシグネチャとして定義してある既知のマルウェアを確実に検知することができるが、未知のマルウェアについては検知することができない。一方で、アノマリ型は正常時の通信パターンを正常として定義し、その定義にそぐわないものを異常パターンとして検出する。アノマリ型検知手法は正常パターンの定義を適切に行うことができれば未知のマルウェアであっても検知できる利点があるが、正常通信であっても外れ値として正常パターンから除かれてしまう誤検知を招くケースもある。今回は次々と登場する「Mirai」の亜種にも対応する必要があるため、アノマリ型での検知システムを採用する。

3.2 ペイロードからの特徴量抽出

本研究では機械学習に用いる特徴量の抽出を $n-gram$ 解析によって行う。$n-gram$ 解析は自然言語 (テキスト) を連続する n 個の文字、または n 個の単語単位で単語を切り出し、切り出した文字や単語の出現頻度を求めることである。これにより、テキスト中の任意の文字列の出現頻度パターンを知ることが可能となる。$n-gram$ 解析は自然言語処理において自然言語をデータとして機械学習をする際に用いられる手法であるが他分野でも頻繁に適用されている。

$n-gram$ 解析による文字列の分割の例として「あいうえお」という文字列の連続要素を以下に示す。

- **「あいうえお」の連続要素**
  - $n=1$ 「あ」「い」「う」「え」「お」
  - $n=2$ 「あい」「いう」「うえ」「えお」

– $n = 3$ 「あいう」「いうえ」「うえお」

提案システムでは、パケットのペイロード内の n 個の連続したバイト列の出現頻度から出現回数の総和と平均そして標準偏差を計算しこの 3 つを機械学習における入力データとして用いる。また、n の値を増やすと計算量が膨大になってしまうことを考慮し今回は $n = 2$ の $2-gram$ によって実装する。

3.3 One-Class Support Vector Machine

異常値を検出するために機械学習を用いるが今回は One-Class SVM を採用する。これはテキスト分類問題において利用されている $n-gram$ 解析が SVM と組み合わせて運用した場合に良好な性能を示すためである [6]。

一般的に、Support Vector Machine (SVM) は正常値 (+1) と異常値 (–1) の 2 クラスのベクトルを教師あり学習し、与えられたベクトルがどちらのクラスに属するかを分類する。一方で、One-Class SVM では正常値 (+1) のクラスのベクトルのみを学習し、与えられたベクトルがそのクラスに属するか否かで判定を行う。

通常、正常値はデータ空間において密度の高い領域にあり、異常値は密度が低い領域にある。そこで、データ間の距離を (1) 式のガウシアンカーネルを用いて特徴空間へ写像することで、異常値を原点近くに写像することが可能となる [7]。

この性質を利用して、原点付近のデータ群と他のデータ群を識別するマージンが最大となるような超平面を設定し、外れ値の外れ具合の指標を求める。(2) 式は識別関数 [8] であり、Φ で写像された非線形空間での内積は (1) 式となる。学習データのうち m の割合のデータが原点付近にくるような識別境界を (3) 式の最適化問題によって求める [9]。

$$K(x, x') = exp\left(-\frac{||x - x'||^2}{\sigma^2}\right) \tag{1}$$

$$f(\Phi(x)) = sgn(\omega \cdot \Phi(x) - \rho) \tag{2}$$

$$min_{\omega,\xi,\rho}\left(\frac{1}{2}||\omega||^2 + \frac{1}{ml}\sum_i \xi_i - \rho\right) \tag{3}$$

subject to

$$(w \cdot \phi(x_i)) \geq \rho - \xi_i, \xi_i \geq 0$$

(2) 式の識別関数が正であれば正常データ側、負であれば異常データ側であり、外れ値を検出できる。

本研究では、正常通信時のパケットの特徴量ベクトルを正常値として学習しておき、このクラスから外れたパケットをマルウェアに感染したデバイスの通信パケットと判定する。

3.4 提案システムの流れ

本研究の提案システムは学習フェーズとテストフェーズに分かれる。

まず学習フェーズでは、学習用データとしてマルウェアに感染していない IoT デバイスを稼働させた際の通信パケットから、$n-gram$ 解析によってペイロード内の $2-gram$ の出現頻度をカウントし、出現回数の総和と平均、標準偏差の 3 つを 1 パケットの特徴ベクトルとして抽出する。その後、抽出した特徴ベクトルを元に One-Class SVM の学習を行う。One-Class SVM では正常データの特徴ベクトルのみを学習するため、この段階で正常値と異常値を区別するための識別境界を生成し、正常クラスの領域が設定される。

そしてテストフェーズでは、検知対象の IoT デバイスの通信パケットから、学習フェーズと同様に $n-gram$ 解析によって特徴ベクトルを抽出する。抽出した特徴ベクトルが学習フェーズで生成した正常クラスの領域内に分類された場合、そのパケットは正常と判定される。また、正常クラスの領域外（外れ値）に分類された場合、そのパケットを異常と判定する。

表 1. False Positive 計測用データの概要

データセット名	キャプチャ期間 (hrs)	パケット数	IoTデバイス名
CTU-Honeypot-Capture-7-1	1.4	8,276	Somfy Door Lock
CTU-Honeypot-Capture-4-1	24	21,000	Philips HUE
CTU-Honeypot-Capture-5-1	5.4	398,000	Amazon Echo

表 2. True Positive 計測用データの概要

データセット名	キャプチャ期間 (hrs)	C&Cとのパケット数	マルウェア名
CTU-Malware-Capture-34-1	24	36,797	Mirai

4. 評価実験

4.1 実験目的

提案システムについて検知率と誤検知率を測定し、提案システムの有用性を確認する。

4.2 データセット

今回は IoT デバイスの通信トラフィックのデータセットとして Stratosphere Lab に公開されている IoT-23[10] を使用した。このデータセットは 20 種類の悪性シナリオと 3 種類の良性シナリオが用意されている。

このデータセットのうち、False Positive を計測するためのデータとして 3 種の良性シナリオを使用する。それぞれのシナリオで稼働させた IoT デバイスは異なっており、キャプチャしたトラフィックのパケット数は約 40 万パケットとなっている。良性シナリオのデータセットの概要は表 1 に示す。また、正常用データセットにおいては、学習用とテスト用の両方を用意する必要があるため、データセットのうち 8 割を学習用、2 割をテスト用になるようにデータを分割した。

悪性シナリオのデータセットはいくつか公開されていたが、今回は「Mirai」に感染させた Raspberry Pi の通信トラフィックをキャプチャした「CTU-IoT-Malware-Capture-34-1」を使用した。また今回はマルウェアに感染した IoT デバイスが C&C サーバと通信を行う際のトラフィックを用いて検出を行うため、このキャプチャデータから C&C サーバと通信を行っているパケットのみを抽出した。この抽出した C&C トラフィックのパケットを True Positive の計測用データとする。概要を表 2 に示す。

4.3 One-Class SVM のパラメータについて

本研究の提案システムでは One-Class SVM を採用しており、その実装は python の機械学習ライブラリである scikit-

表 3. $gamma = 0.001$ の検知結果

nu	TPR(%)	FPR(%)	ACC(%)
0.001	3.1	5.7	15.8
0.01	7.3	6.8	19.3
0.05	9.3	24.6	18.5
0.1	15.7	14.7	25.5
0.5	26.9	52.6	29.9

表 4. $gamma = 0.01$ の検知結果

nu	TPR(%)	FPR(%)	ACC(%)
0.001	89.6	7.7	90
0.01	88.9	14.1	88.5
0.05	91.5	26.4	89
0.1	94.6	8.7	94.2
0.5	91.9	57.2	91.9

表 5. $gamma = 0.1$ の検知結果

nu	TPR(%)	FPR(%)	ACC(%)
0.001	73.6	17	74.9
0.01	94.5	15.2	93.2
0.05	94.2	17.9	92.5
0.1	94.1	43.9	88.6
0.5	99.9	73.8	89.5

learn を用いた。

One-Class SVM を運用する上で学習の際に設定すべきパラメータに nu と $gamma$ がある。nu は学習データにどれだけの割合で異常値を含むか決める値であり、$gamma$ は識別境界の複雑さを決定するためのものである。本来であれば交差検証を行い、適切なパラメータを設定しなければならないが、今回は $gamma$ の値を 0.001、0.01、0.1 の 3 パターンとし、それぞれの $gamma$ に対して nu の値を適当に選定して実験を行っている。そのため、One-Class SVM のパラメータの調整は今後の課題と言える。

4.4 評価

本研究の提案システムの検知精度を検証した。また、本論では正常を Negative、異常を Positive とし、評価に用いる指標として以下を用いる.

- **評価指標**
  - $Accuracy(ACC) = \frac{TP+TN}{TP+FP+TN+FN}$
  - $TruePositiveRate(TPR) = \frac{TP}{TP+FN}$
  - $FalsePositiveRate(FPR) = \frac{FP}{FP+TN}$

3 パターンの $gamma$ の値について、それぞれの検知結果を表に示す。

まず、$gamma = 0.001$ の時、ACC はせいぜい 30%という低い数値となっている。また、TPR が 30%を超えているものがないことから、殆どの C&C トラフィックを検出できていないことがわかる。

次に、$gamma = 0.01$ の場合は、ACC は全体的に 90%近い値となっているが、FPR は一番低くても 7%ほどとなっており、2,000 パケットを超える誤検知が発生していることになる。

そして、$gamma = 0.1$ の場合、$nu = 0,001$ の時を除いて ACC は 90%近い値を得られている.$nu = 0.5$ では TPR が 99.9%と高い結果となっている。しかし、FPR は 10%以下にすら抑えることができていないことから誤検知が多いことがわかる。

今回の実験において、$gamma = 0.01$ かつ $nu = 0.1$ のとき、90%を超える TPR 得られ、FPR を 10%未満に抑える結果となったため最も良好な検知精度だったと考える。しかし、C&C サーバとの通信パケットを完全に検出することはできなかった。また、誤検知に関しても 1%未満に抑えることができなかったことからも、今後システムの改良を行う必要があると考える。

5. まとめ

本研究ではマルウェアに感染しボット化した IoT デバイスの検出をすることが可能な検知システムの提案を行った。ボット化した IoT デバイスが C&C サーバと通信を行うこと、そして C&C トラフィックのペイロード情報が正常トラフィックのペイロードと異なること [5] から、ペイロードによるマルウェアの検出が可能であると考えた。提案システムでは、ペイロードからの特徴量を $n-gram$ 解析によって抽出し、抽出した特徴ベクトルをもとに機械学習手法である One-Class SVM を用いることで C&C トラフィックの識別を行う方法を考案した。このシステムによる検知率は、True Positive 率 94.6%、False Positive 率 8.7%であり、検知精度の改善が求められる結果となった。

今後の課題として、まず 1 つに One-Class SVM のパラメータの調整がある。今回の実験ではパラメータの最適化を行っていないため、パラメータの値次第ではより高い検知精度を得ることができると考えている。また、C&C サーバとの通信の検出率を上げるために C&C セッションを特徴量として考慮することが有効であると考える。これはマルウェアに感染したデバイスが C&C サーバと通信を行う際にパケットの送受信間隔に決まった特徴があることが分かっているためである [11]。そして、今回は One-Class SVM を単体で使用したが、複数の分類器を組み合わせて使用することで分類精度が向上することが分かっている [6]。そのため、One-Class SVM だけでなく他の機械学習手法でも分類を行い、その分類結果を組み合わせて評価することができるシステムが実装できれば、TPR、FPR はさらに改善されると考えられる。

参考文献

1) https://dyn.com/blog/dyn-analysis-summary\\-of-friday-october-21-attack/, (2020-1-22 閲覧).

2) https://eset-info.canon-its.jp/malware_info/trend/detail/150120_3.html, (2020-1-22 閲覧).

3) https://cybersecurity-jp.com/security-measures/21815, (2020-1-22 閲覧).

4) Wang, Ke, and Salvatore J. Stolfo: Anomalous payload-based network intrusion detection, RAID. Vol.

4. 2004. Recent Advances in Intrusion Detection, pp. 203-222, 2004.

5) 大月優輔, 市野将嗣, 川元研治, et al.: マルウェア感染検知のためのトラヒックデータにおけるペイロード情報の特徴量評価, コンピュータセキュリティシンポジウム 2012 論文集, vol.3, pp. 691-698, 2012-10-23.

6) Roberto Perdisci, Guofei Gu, and Wenke Lee: Using an Ensemble of One-Class SVM Classifiers to Harden Payload-based Anomaly Detection Systems, Proceedings of the IEEE International Conference on Data Mining (ICDM 2006), pp. 488-498, 18-22 December 2006.

7) https://qiita.com/sergeant-wizard/items/c5f79adefe5016f4740e, (2020-1-22 閲覧).

8) https://home.hiroshima-u.ac.jp/tkurita/lecture/svm/node4.html, (2020-1-22 閲覧).

9) https://orizuru.io/blog/machine-learning/one-class-svm/, (2020-1-22 閲覧).

10) https://www.stratosphereips.org/datasets-iot23, (2020-1-22 閲覧).

11) 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, FIT2007, L-033, pp. 77–78, 2007.