IoT機器の安全性を高める動的通信分別手法の研究

長嶋 秀幸[1] 落合 秀也[1] 江崎 浩[1]

概要：IoT とはありとあらゆるモノがインターネットに接続され様々なサービスを提供するインフラス トラクチャー・システムのことである。IoT システムに関する研究開発と社会実装が展開される中、 IoT 機器に関するサイバーセキュリティ対策が問題になっている。具体的には、適切にセキュリティ 機能の実装と設定が管理されず、不十分・不適切な状態のままインターネットを含む様々なネット ワークに接続されている IoT 機器が多く存在する。その結果、それらの機器が不正にアクセスされ DDoS 攻撃や e メールスパムなど様々な攻撃に悪用される事例が多数発生している。このような IoT 機器の悪用を防止するためにはファイアウォールや IoT 機器自体でのアクセス制 限を行う、あるいは IoT 機器への平易なパスワードでのログインを許可しない設定にしておくこと やデフォルトのパスワードから推測されにくいパスワードに変更しておくなどの対処法が一般的であ る。しかしこれらの対処法があるにも関わらず IoT 機器によるインシデントは多数発生している。本研究では、ある特定の通信相手に対して定期的な通信を行っているような IoT 機器を想定し、そ の通信データを分析することで、正常な通信と不正な通信を判別する手法を提案・実装した。実際の 通信データに提案手法を使用し、その有効性を検証した。

1. はじめに

IoT とは Internet of Things の略称であり、ありとあらゆるモノがインターネットに接続されサービスを提供するインフラストラクチャー・システムのことである。モノとしてはセンサやマイク、カメラなどが挙げられ、それらで取得された周囲の温度や映像、静止画像などの情報をインターネットを介して集積、分析し、その結果に基づいてモノが動作しサービスを提供する。この IoT のシステムは様々な分野で社会実装が展開されている。例えば医療分野では体にセンサを付けて体温や心拍数、血圧などの情報を計測し患者の健康状態を把握したり、家電分野では家の中の電化製品の制御、窓や扉の施錠の確認などをタブレット端末から行えるようにしたりなど、他にも農業分野や自動車分野などその利用は多岐にわたる。このようなインターネットに接続される IoT 機器は 2017 年の Gartner 社の予測によると設置ベースで 2017 年には約 84 億台あると推測され、2020 年には約 200 億台にもなると予想されている [1]。

その利便性の一方、IoT 機器の普及と共にそのサイバーセキュリティ対策が問題になっている。IoT 機器はインターネットに接続されることによりこれまでのコンピュータと同様のリスクに晒されることになる。しかし一般的に IoT 機器では使用できる CPU やメモリ、消費電力などに制限があるため従来と同じ方法でのコストの高いセキュリティは導入するのが困難である。またそれだけではなくその爆発的な増加やインターネットに接続されているという意識の低さなどに起因すると考えられる適切に管理されていない設定の不十分な IoT 機器がインターネット上には多く存在する。

それらにより引き起こされたインシデントの事例として 2016 年に発生したダイン社に対する DDoS 攻撃 [2] や 2015 年に Incapsula が報告したルータによる DDoS 攻撃 [3] などが挙げられる。前者のダイン社に対する攻撃では Mirai というマルウェアに感染したウェブカメラなどの機器が踏み台にされ、Twitter、Netflix、Amazon、Github、Reddit などを含む多数のウェブサイトが数時間に渡りダウンした。Mirai の感染方法は約 60 ほどのデフォルトやよく使用されるユーザ名とパスワードの組み合わせでログイン出来る機器を探すというものだった。後者の攻撃では Small Office/Home Office 向けのルータが乗っ取られ悪用された。感染したルータに関しては HTTP と SSH によるリモートアクセスがデフォルトポート経由で可能になっていたうえ、ほとんどのルータがデフォルトのログイン認証情報を使っていたことが分かった。

このように設定の不十分な IoT 機器は攻撃に利用される可能性がある。現在上記のような不正なログインを防ぐためにはファイアウォールや IoT 機器自体でのアクセス制限を利用する。または IoT 機器へのパスワードでのログイン

[1] 東京大学大学院情報理工学系研究科

を許可しない設定にすることやデフォルトのパスワードから推測されにくいパスワードに変更することなどが必要である。しかしこれらの方法があるにも関わらず攻撃に利用される機器が多くある。そのことからファイアウォールでの設定や IoT 機器自体のセキュリティ設定ではなく通信データを利用し動的に不正アクセスや DoS 攻撃から IoT 機器を守る方法を提案する。

最後に本論文の構成を述べる。本章以降では 2 章で関連技術と関連研究として本研究に使用した Local Outlier Factor による外れ値検出の説明と他の異常検知手法についての紹介を行う。次に 3 章で提案手法の説明をする。4 章で実施した実験の説明、そしてその結果と評価を述べ、最後に 5 章でまとめと今後の課題を述べる。

2. 関連技術・関連研究

2.1 LOF による外れ値検出

この節では本研究に用いた LOF による外れ値検出 [4] について概説する。まず説明に必要な値の定義を説明する。あるデータセットを D、それに含まれるデータを p とし、次の 2 つの条件を満たすデータ $o \in D$ と p の距離をデータ p の k 距離 ($k-distance(p)$) と定義する。

(1) 少なくとも k 個のデータ $o^{'} \in D \setminus \{p\}$ に対して $d(p,o^{'}) \leq d(p,o)$ が成り立つ

(2) 高々 $k-1$ 個のデータ $o^{'} \in D \setminus \{p\}$ に対して $d(p,o^{'}) < d(p,o)$ が成り立つ

$d(p,o)$ は p と o のユークリッド距離である。そしてこの p の k 距離近傍のデータ群を $N_{k-distance(p)}(p)$ といい、次のように定義される。

$$N_{k-distance(p)}(p) = \{q \in D \setminus \{p\} \mid d(p,q) \leq k-distance(p)\}$$

LOF による外れ値検出は n 次元空間上であるデータの局所密度とあるデータの近傍のデータの局所密度を計算し、その比率の平均によって外れ値を検出する手法である。局所密度とはあるデータの近傍 k 個のデータとの距離の平均の逆数である。データ p の局所密度 ($lrd_k(p)$) は次式で定義される。

$$lrd_k(p) = 1/\left(\frac{\sum_{o \in N_{k-distance(p)}(p)} reach-dist_k(p,o)}{\left|N_{k-distance(p)}(p)\right|} \right)$$

$reach-dist_k(p,o)$ は到達可能距離といい、p と o の間の距離としてユークリッド距離ではなくこの到達可能距離を使用する。

$$reach-dist_k(p,o) = max\{k-distance(o),\ d(p,o)\}$$

図 1 においてデータ p 自身の局所密度は p の近傍のデータの局所密度と比べて小さくなり、そのためデータ p は外れ値と判定される。p 以外のデータは自身の局所密度と近傍のデータの局所密度の差が小さく、正常値と判定される。この局所密度の比率の平均が LOF であり

$$LOF_k(p) = \frac{\sum_{o \in N_{k-distance(p)}(p)} \frac{lrd_k(o)}{lrd_k(p)}}{\left|N_{k-distance(p)}(p)\right|}$$

と定義される。

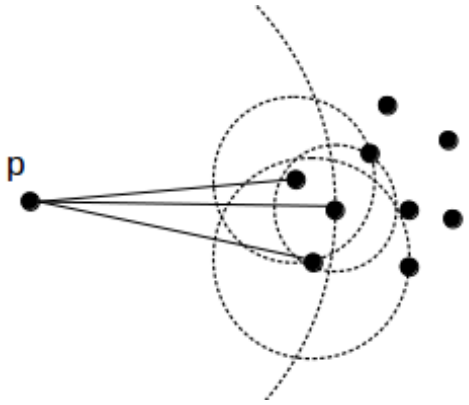

図 1　LOF による外れ値検出の概要

2.2 関連研究

この節では他の異常検知手法について述べる。システムやネットワーク上で行われる不正な行為を検出してシステム管理者への通知など指定された動作をするを Intrusion Detection System(IDS) という。IDS はさらにその検出手法でシグネチャ型 (不正検出型) とアノマリ型 (異常検出型) の 2 種類に分類することができる [5]。

すでに知られている攻撃のパターンや特徴を定義したものをシグネチャと呼び、そのシグネチャと通信パケットがマッチングする場合に不正な通信であると判断するのがシグネチャ型の IDS である。シグネチャ型の IDS では既知の攻撃方法に対しては優れた検出能力を持つが、シグネチャの存在しない未知の攻撃は検知出来ないため常にシグネチャを更新し続けなければいけないという欠点がある。代表的なシグネチャ型の IDS としてはオープンソースソフトウェアである Snort[6] などがある。

アノマリ型の IDS では通常の動作や正常な通信を定義しておき、その定義から外れたものを不正な通信と判断する。正常な状態は管理者によって統計情報などの閾値で設定する。アノマリ型ではシグネチャ型のように新しい攻撃が見つかるたびに設定の更新をする必要は無くなるが、閾値を適切に設定できずに攻撃を許したり誤検知が多くなってしまうなどの問題がある。

また機械学習を用いて通信の判定を行う研究も多く行われてきた。Umer らの教師なしクラスタリングの手法である k-means、self-organizing map、DBSCAN を用いて通信の判定を行ったもの [7] や Sabhnani らの多層パーセプトロ

ンなど 9 種類の機械学習の手法を KDDCup99 データセットを用いて評価したもの [8] などがある。

3. 提案手法

3.1 想定環境

本研究で想定する IoT 機器の性質を述べる。想定する IoT 機器は監視カメラなどある特定の通信相手を持ち、定期的な通信を行っているものとし断続的な通信を行うプリンタなどの機器は考えない。また通信相手が特定の端末であるため同じプロトコルを使用して同時に多数の相手と通信するような状況はないものとする。

IoT 機器は通信を取得する初期状態ではクリーンであるとし、不正なプログラムをインストールされるなどして他の端末との通信を行ってはいないものとする。

3.2 手法の概要

ここでは提案手法の概要を説明する。IoT 機器で送受信される通信パケットのみを全て取得でき、IoT 機器への通信を制御できるような位置において以下に述べる手順で通信の正常、異常の判別と遮断を行う。

IoT 機器の通信パケットは一定時間の間取得され、その一定時間分のデータを分析することでどの通信を遮断するかを判定する。まず取得した一定時間分のパケットのデータを通信に使用しているポート番号によって分割する。そして分割したそれぞれのデータで通信相手ごとの統計量を計算する。本研究で対象とする IoT 機器は少数の特定の通信相手を持ち通信を行っているものと想定しているので、あるポート番号についての通信相手の構成は IoT 機器への攻撃を行っている端末や通信する必要のない端末が多数存在し、正常な通信を行っている端末が少数であると考えられる。そこで計算した統計量のなかで外れ値となっている値を持っている通信相手を正常な通信相手であると判定する。外れ値の発見は LOF による外れ値検出で行う。

次に LOF による外れ値検出を使用して判定を行う方法について、図 2 にデータの取得から正常な通信かどうかの判定までの流れを示し、各段階の処理について説明する。

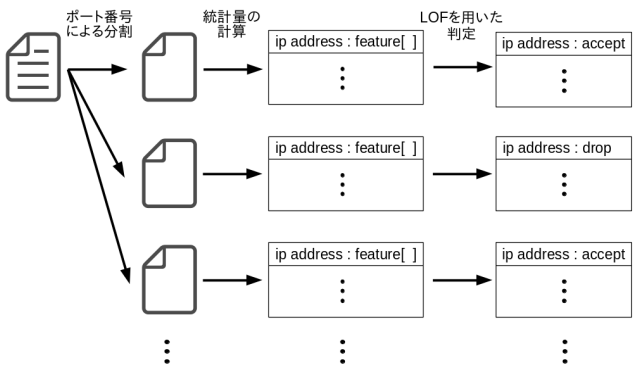

図 2 LOF による外れ値検出を用いた判定までの流れ

1.IoT 機器の通信の取得

IoT 機器が送信しているパケット、受信しているパケットを取得する。一定時間間隔で区切ったデータを分析に使用する。通信パケットを取得するのは IoT 機器の送受信パケットが全て観測できる位置で行う。

2. ポート番号による分割

取得したデータのうち TCP、UDP で通信しているデータをポート番号ごとに分割する。分割のためのポート番号は送信元ポート番号と宛先ポート番号で値の小さいものを使用する。

3.IP アドレスごとの統計量の計算

ポート番号ごとにデータ分割した後、各 IP アドレスとの通信について統計量を計算する。統計量は送受信したパケットの回数、通信したデータ量の合計、パケットあたりのデータ量の平均、パケットの時間間隔の平均とした。

4. 統計量を使用した通信の判定

計算した統計量をもとに各 IP アドレスとの通信が正常なものかどうか判定する。4 つ全ての統計量を使用し n 次元での外れ値検出手法である LOF による外れ値検出を行い、外れ値と判定されたものを正常な通信とする。対象の IoT 機器は多数の相手との通信を行わないと想定しているので、あらかじめ 1 つのポート番号での通信相手数の上限を設定しておき、外れ値と判定されたものの数がその上限を超えた場合はそのポート番号での通信は正常な通信が無いと判定する。

4. 実験・考察

4.1 検証データ

検証に用いたデータの説明を述べる。東京大学江崎落合研究室内に設置されているディジインターナショナル社製の XStick ZB を挿入した PC を計測対象として、それが送受信するパケットを取得した。XStick ZB はワイヤレス PAN アダプタであり、USB ポートに差しこむだけでワイヤレスネットワークと PC との通信を可能にする機器である。計測対象の PC は IP アドレス 203.178.135.67 の端末と HTTP を用いて定期的に通信をしていた。通信パケットの取得は計測対象とそれが接続しているスイッチの間で tcpdump を使用して行った。パケットを取得した期間は 2017 年 11 月 2 日から 2017 年 11 月 6 日までの 4 日間である。

4.2 考察

本節では提案手法の実験結果と評価を述べる。提案手法を適用する時間間隔として 6、12、24 時間分の 3 通りを設定した。代表的な値としてポート番号 80 の HTTP の通信とポート番号 53 の DNS の IP アドレスに対する判定の結果を表 1、表 2 に示す。一部省略している部分は正常でな

いと判定された通信である。

まずポート番号 80 の通信の判定結果について述べる。表 1 を見ると全ての時間間隔で 203.178.135.67 を正常な通信であると判定している。**??**で述べたとおり計測対象の機器は IP アドレス 203.178.135.67 とポート番号 80 を利用して通信しているのでこの判定は成功したと言える。しかし 203.178.135.67 以外に正しいと判定した通信相手が 3 つ存在する。計測対象の機器はこれらの IP アドレスは正しい通信相手ではないため不必要な通信を誤って正しいと判定していることが分かる。

次にポート番号 53 の通信についての判定について述べる。表 2 を見ると全ての時間間隔で 203.178.135.2 を正しい通信と判定している。203.178.135.2 は計測対象の機器の接続しているネットワークセグメントの DNS サーバであり正しい通信相手であるためこの判定は成功していると言える。セカンダリ DNS である 203.178.135.36 については 12 時間分のデータを使用した際にを正しい通信と判定できているがそれ以外では判定に失敗している。また 24 時間分のデータで判定した時に 185.188.207.15 を正しい通信と誤って判定している。

最後にポート番号 123 番の判定結果について述べる。ポート番号 123 番はネットワークに接続される機器の時刻を正しいものに同期するために使用される。計測対象の機器も NTP サーバと通信を行い時刻を同期している。通信している NTP サーバの IP アドレスは 160.16.75.242、157.7.154.29、129.250.35.251、172.104.105.31 の 4 つである。表 3 を見ると、24 時間分のデータでは判定に成功しているが 12 時間分のデータでは 2 つの NTP サーバのみ判定に成功、6 時間分のデータでは全ての判定に失敗している。これは正しい通信を行っている相手が不正な通信を試みている相手に比べて多かったことが原因と思われる。本来正常な通信相手が不正な通信相手に比べて少ないという想定で外れ値を正常な通信のものと判定していたが 6 時間分のデータを見てみると、正常な相手と不正な相手の数が等しいという状況になっている。このような状況では提案手法では正しい判定が行えないことが分かった。

5. おわりに

本研究では IoT 機器から取得した通信データをポート番号ごとに分割し外れ値を検出することで正しい通信かどうかの判定を行う方法を提案し実際のデータに適用して検証した。通信量が他と大きく異なる通信については正しく判定できたが、本来許可する必要の無い通信についても正しい通信と判定してしまう傾向が高かった。

今後の課題としては第一にポート番号 123 番の通信データの判定結果から分かるように提案手法を適用する際、パケットの取得時間を小さくしていくとデータに含まれる通信相手数が減少し、外れ値検出による通信の判定が行えない場合があるということである。第二に本研究では外れ値検出は 4 つの統計量を用いて行ったが、これに関してさらに詳細な統計量を追加し様々な要素で通信を評価できるようになれば現在よりも判定の精度が向上する可能性がある。

参考文献

[1] Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016 入手先 ⟨http://www.gartner.com/newsroom/id/3598917⟩（参照 2017-1-25）.

[2] Kolias, Constantinos, et al. "DDoS in the IoT: Mirai and other botnets." Computer 50.7 (2017): 80-84.

[3] Lax Security Opens the Door for Mass Scale Abuse of SOHO Routers 入手先 ⟨https://www.incapsula.com/blog/ddos-botnet-soho-router.html⟩（参照 2017-1-25）

[4] Breunig, Markus M., et al. "LOF: identifying density-based local outliers." ACM sigmod record. Vol. 29. No. 2. ACM, 2000.

[5] Gangwar, Mrs Anshu, and Mr Sandeep Sahu. "A survey on anomaly and signature based intrusion detection system (IDS)." International Journal of Engineering Research and Applications 4.4 (2014): 67-72.

[6] Snort - Network Intrusion Detection & Prevention System 入手先 ⟨https://www.snort.org/⟩

[7] Umer, Muhammad Fahad, and Muhammad Sher. "Automatic Clustering of Malicious IP Flow Records Using Unsupervised Learning." Enterprise Security. Springer, Cham, 2017. 97-119.

[8] Sabhnani, Maheshkumar, and Grsel Serpen. "Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context." MLMTA. 2003.

表 1　ポート番号 80 番の通信の判定結果

6 時間		12 時間		24 時間	
IP アドレス	判定	IP アドレス	判定	IP アドレス	判定
203.178.135.67	accept	203.178.135.67	accept	203.178.135.67	accept
141.212.122.69	drop	13.80.148.254	accept	208.100.26.228	accept
187.122.241.46	drop	176.63.90.83	drop	13.80.148.254	accept
185.172.110.216	drop	24.192.163.197	drop	221.194.44.161	drop
176.63.90.83	drop	141.212.122.72	drop	45.55.24.20	drop
208.100.26.228	drop	141.212.122.122	drop	119.94.91.38	drop
141.212.122.72	drop	139.162.111.147	drop	185.172.110.216	drop
13.80.148.254	drop	141.212.122.69	drop	151.80.79.184	drop
		208.100.26.228	drop	104.238.169.57	drop
		37.59.49.136	drop	74.82.47.13	drop
		141.212.122.151	drop	141.212.122.69	drop
		185.172.110.216	drop	193.203.234.216	drop
		151.80.79.184	drop	24.192.163.197	drop
		221.194.44.161	drop	⋮	⋮
				72.21.92.20	drop

表 2　ポート番号 53 番の通信の判定結果

6 時間		12 時間		24 時間	
IP アドレス	判定	IP アドレス	判定	IP アドレス	判定
203.178.135.2	accept	203.178.135.2	accept	203.178.135.2	accept
203.178.136.36	drop	203.178.136.36	accept	185.188.207.15	accept
133.11.124.164	drop	133.11.124.164	drop	203.178.136.36	drop
208.100.26.228	drop	71.6.216.60	drop	133.11.124.164	drop
74.82.47.2	drop	209.126.136.2	drop	71.6.216.60	drop
185.94.111.1	drop	185.94.111.1	drop	185.94.111.1	drop
45.55.0.154	drop	208.100.26.228	drop	208.100.26.228	drop
209.126.136.2	drop	141.212.122.164	drop	74.82.47.2	drop
		74.82.47.2	drop	139.162.126.103	drop
		139.162.126.103	drop	45.55.0.154	drop
		45.55.0.154	drop	141.212.122.164	drop
				209.126.136.2	drop

表 3　ポート番号 123 番の通信の判定結果

6 時間		12 時間		24 時間	
IP アドレス	判定	IP アドレス	判定	IP アドレス	判定
129.250.206.86	accept	160.16.75.242	accept	160.16.75.242	accept
184.105.139.80	accept	157.7.154.29	accept	157.7.154.29	accept
185.94.111.1	accept	184.105.139.80	drop	129.250.35.251	accept
185.188.207.15	accept	172.104.105.31	drop	172.104.105.31	accept
160.16.75.242	drop	129.250.35.251	drop	185.188.207.15	drop
129.250.35.251	drop	185.94.111.1	drop	129.250.206.86	drop
172.104.105.31	drop	129.250.206.86	drop	123.249.35.56	drop
157.7.154.29	drop	185.188.207.15	drop	185.35.62.232	drop
				184.105.139.80	drop
				⋮	⋮
				185.82.203.58	drop