2 つの通信パターンをもつ IoT 機器ネットワークに対する アノマリ型侵入検知手法の評価

2017SC056 落合琢斗　2017SC090 山田武徳
指導教員：石原靖哲

1　はじめに

近年, インターネットの発展に伴って, 医療, 自動車, 農業など世の中のあらゆる分野で多くの IoT 機器が利用されており, 我々の生活には必要不可欠のものになっている. しかし, 同時にセキュリティ対策不足により IoT 機器を狙ったサイバー攻撃が増加している [1]. サイバー攻撃の検出方法として, 2 種類の侵入検知システム (IDS) がある. シグネチャ型とアノマリ型である. シグネチャ型は不正パターンに一致するかで侵入を判断するため誤検出が少ない利点はあるが, 短期間で多様に変化するマルウェアなど今までにない攻撃だと検知が難しい. 一方でアノマリ型は正常パターンに一致するかで侵入を検知するため今までにないマルウェアに対して有効な手段ではある. しかし, 多種多様な動作をする PC などに対しては正常パターンのモデルを構築する必要があるため, 正常パターンの定義によっては誤検知が多くなってしまい検知精度を高めるのは難しい. そんななか, 瀧本ら [4] は限定的な振る舞いをするネットワークカメラのような IoT 機器なら通信ログは限られた種類のパケットのみからなると想定し, 比較的容易に正常パターンを定義することができると仮定した. そして, その仮定に基づき IoT 機器に特化したアノマリ型 IDS を提案した.

本研究では瀧本らの手法を元にアノマリ型 IDS の評価システムの構築を行う. そして 2 つの通信パターンをもつ IoT 機器への様々な攻撃に対するアノマリ型 IDS の有効性の評価を本研究の目的とする. 2 つの通信パターンをもつ IoT 機器を用いる理由として, 今後 IoT 機器が社会に増えていく中で, 特に 2 つもしくはそれ以上の機能をもつ IoT 機器の需要が高まるのではないかと考えたからである.

図 1 は対象とした IoT 機器に攻撃が行われている様子を示す. 2 つの通信パターンをもつ IoT 機器に対する攻撃を, アノマリ型侵入検知する環境を想定している.

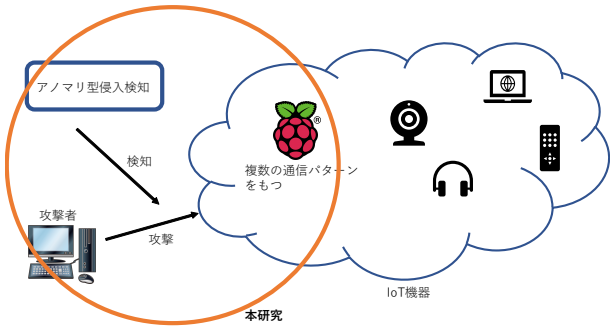

図 1　想定される環境

2　関連研究

本節では IoT 機器に対してのセキュリティ問題や異常検出の論文について紹介する.

1. 瀧本らによる文献 [4] は本研究の元になっている研究である. ネットワークカメラのような限られた動作をする IoT 機器にはアノマリ型 IDS が有効であることを提案し, 評価している.
2. 中原らによる文献 [3] では, ホームネットワークにおける IoT デバイス の異常検知手法として, ホームゲートウェイにて統計化した情報を解析サーバへと送付して検知を行うシステムを提案をしている.
3. 桂井らによる文献 [2] では, Raspberry Pi, snort を用いて, IoT ネットワーク向けの不正検知システム実現の検証を行っている.
4. 瀧本らの最新の論文 [5] では, 深層距離学習手法の 1 つである L2-SoftmaxNetwork を用いたアノマリ型攻撃検知システムを提案している.

3　評価システムの構築

3.1　システムの全体像

図 2 は図 1 の想定される環境を考慮した本研究の評価システムの全体像である. 本システムの開発環境は以下の通りである.

【ハードウェア】

- PC：名称 dynabook RX73/CBE
  プロセッサ Intel(R) Core(TM) i5-7200U CPU @2.50GHz 2.70GHz
  実装メモリ 8.00 GB
- IoT 機器：Raspberry Pi 4 Model B, カメラモジュール, 人感センサー

【ソフトウェア】

- OS：Windows 10
- プログラミング言語：Python3.7
- Weka：バージョン Weka 3.8.4
- ペネトレーションテストツール：metasploit
- Wireshark：バージョン Wireshark 3.2.5
- FileZilla：バージョン 3.48.1.0

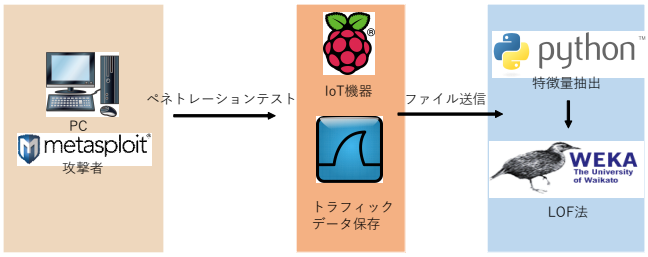

図 2　システムの全体像

表 1　通信パターン基準

通信先の IP アドレス	通信先	IP アドレス
	LINE	203.104.138.174
	Dropbox	162.125.80.14
通信量	通信量のパターン	例
	大 (3001KB 以上)	映像, システムのアップデート
	中 (101KB 以上 3000KB 以下)	画像, 検索
	小 (100KB 以下)	メッセージ
通信のタイミング	タイミングのパターン	例
	一定周期で通信	5 分おきに通信
	不定期に通信	ユーザーのアクション
	持続的に通信	監視カメラ

3.2　IoT 機器

3.2.1　通信パターンの定義

本研究における「通信パターン」の概念を, 通信パターン分析の論文 [6] を参考に定義する. まず, 通信パターンを定めるための基準として以下の 3 つを採用する.

- 通信先に基づく基準
- 通信量に基づく基準
- 通信のタイミングに基づく基準

そして, これらの基準ごとに表 1 のように通信パターンを定める.

通信先に基づく基準については, [6] では通信先がローカル内かローカル外かという 2 パターンとして定義されている. 本研究では通信先 IP アドレスごとに通信パターンが定まると定義する. よって本研究では, 異なる IP アドレスをもつ 2 つのホストと通信を行う IoT 機器は, 2 つの通信パターンをもつ.

通信量に基づく基準については, [6] と同様に, 通信量が 100KB 以下の場合, 101KB 以上 3000KB 以下の場合, 3001KB 以上の場合の 3 種類の通信パターンを定義する. よって本研究では, たとえば同じホストに 1KB 程度のテキストメッセージと 1000KB 程度の画像データを送信する IoT 機器は, 2 つの通信パターンをもつ.

通信のタイミングに基づく基準についても, [6] と同様に, 定期, 一定周期, 不定期, 常に通信という 4 種類の通信パターンを定義する. よって本研究では, たとえば同じホスト相手に 3 分間隔で通信しつつ不定期な通信も発生させる IoT 機器は, 2 つの通信パターンをもつ.

3.2.2　評価対象とする IoT 機器

本研究では評価対象として Raspberry Pi 4 を用いて作成した IoT 機器と既製品の IoT 機器として Qwatch（ネットワークカメラ）を用いた. 前節で定義した 3 つの通信パターン基準それぞれについて 2 つの通信パターンをもつよう, 以下の動作をする IoT 機器を作成した.

1. 通信先に基づく基準
   - Dropbox に一定周期で写真を送る.
   - LINE に一定周期で写真を送る.
2. 通信量に基づく基準
   - 一定周期で写真を撮り, その写真から顔を検知したら, メッセージ (通信量 小) のみ LINE に送る.
   - 一定周期で写真を撮り, その写真から顔を検知しなかったら, その写真とメッセージ (通信量 中) を LINE に送る.
3. 通信するタイミングに基づく基準
   - 一定周期で撮影した写真を LINE に送る.
   - 人感センサーにより人を感知出来なかったら撮影し, その写真を LINE に送る.(不定期)

Qwatch は, 通信先に基づく基準についての 2 つ通信パターンをもつよう, Qwatch で撮影した映像を 2 台の PC で常に確認できる状態にした.

3.3　通信ログからの特徴量抽出

異常検知を行うために通信トラフィックデータを一定時間のウインドウに区切り, 文献 [4] を参考にした特徴量を抽出する. この時 Wireshark で通信トラフィックデータを取得すると, pcapng 形式でファイル保存される. この pcapng 形式のファイルから特徴量抽出を行う為に, csv 形式に変換する. 本研究では, 特徴量として, パケットサイズ平均, パケットサイズ分散, 総パケットサイズ, 総パケット数, パケット到着間隔平均, パケット到着間隔分散, TCP パケットの割合, UDP パケットの割合を用いる.

3.4　LOF 法

LOF 法とは近くにデータがない, あるいは極端に少ないものを外れ値とみなす考え方である. 外れ値とは他と大きく異なるデータである. 今回の研究では, ペネトレーションテストを IoT 機器に行ったときの通信が, LOF 法を用いることによって外れ値として検出できるのではないかと考えている. ペネトレーションテストは攻撃者からの攻撃に見立てているため, IoT 機器に対して, アノマリ型 IDS が有効であることが示せる. LOF 法についてより具体的に記述すると以下のようになる.

$$RD_k(x, x') = \max(\left|\left|x - x^{(k)}\right|\right|, ||x - x'||)$$

$$LRD_k(x) = \{\frac{1}{k}\sum_{i=1}^{k} RD_k(x^{(i)}, x)\}^{-1}$$

$$LOF_k(x) = \frac{(\frac{1}{k})\sum_{i=1}^{k} LRD_k(x^{(i)})}{LRD_k(x)}$$

RD とはあるデータ x から別のデータ x' への到達可能距離である. この到達可能距離を用いて, x から k 番目までに近いデータとの到達可能距離の平均の逆数をとった LRD（局所到達可能密度）を求める. そして, この局所到達可能密度によって, x に対して k 番目までに近いデータの局所到達可能密度の平均と, x の局所到達可能密度の比である LOF（局所異常因子）を求めることが出来る. あるデータ x とは今回の研究では Wireshark でキャプチャした IoT 機器とのトラフィックデータを特徴量抽出, さらに主成分分析を行い, 次元圧縮したものである. Weka の LOF フィルタを用いることによって, 主成分分析, 次元圧縮が自動で行われ, 外れ値の検出を行うことが出来る.

4 実験

4.1 実験条件

- 作成した IoT 機器

  石原研究室のネットワークに Raspberry Pi を設置し通信ログを収集した. トラフィックデータとして, 異常データを含むデータ 6 時間分を各攻撃と通信パターン基準ごとに Wireshark で収集した. それぞれのデータは 1 分のウインドウサイズに区切り, 特徴量を抽出した. 各一定周期での LINE, Dropbox のやり取りは, 実際にシステムを使用することを考慮して, 3 分間間隔で定期実行を行った. 攻撃として今回の実験では, ペネトレーションテストを用いたブルートフォース攻撃, ping を用いた DoS 攻撃とする. マルウェアを用いた攻撃は本大学のセキュリティの関係上, 断念した. なお, 攻撃者は事前に IP アドレスは分かっているものとする. また, 正常データとみなす閾値（LOF の計算により求めた数値）を 2, 5, 8 とする.

- 既製品の IoT 機器

  石原研究室のネットワークに 2 台の PC と Qwatch を設置し通信ログを収集した. トラフィックデータの収集方法は作成した IoT 機器と同じである. 攻撃は DoS 攻撃, 閾値は 5 である.

本研究は石原研究室で実験を行うが, 照明点灯時と消灯時で, 画像のサイズが大きく変化し, 1 回の撮影で約 200KB の差が発生することが確認された. 実験への影響の可能性を排除するため, 照明点灯時のみ実験を行った.

4.2 実験結果

図 3, 図 4 は作成した IoT 機器による通信パターンが IP アドレスの DoS 攻撃とブルートフォース攻撃の割合の変化に伴っての見逃し率と誤検知率の推移を表している. 図 5 は既製品の IoT 機器による DoS 攻撃の割合の変化に伴っての見逃し率と誤検知率の推移を表している. 各攻撃の割合は, 6 時間分のデータのうちの異常データを含むデータを 6 時間で割った割合である. 見逃し率は異常なデータなのに正常と判断したデータの割合, 誤検知率は正常なデータなのに異常と判断したデータの割合のことである.

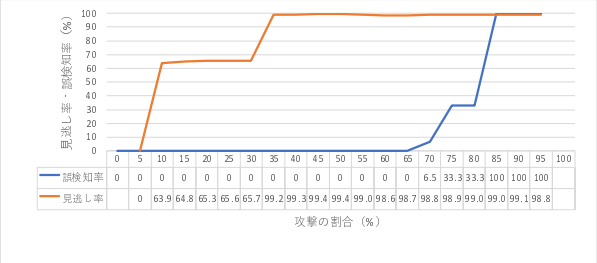

図 3 作成した IoT 機器による誤検知率と見逃し率の推移（DoS 攻撃）（IP アドレス）（閾値 5）

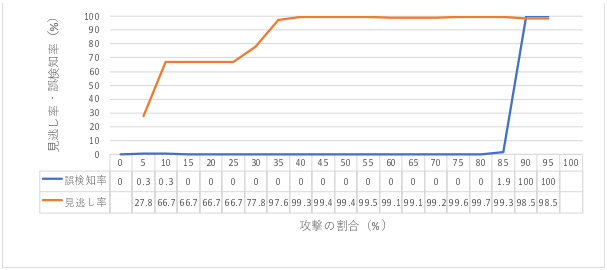

図 4 作成した IoT 機器による誤検知率と見逃し率の推移（ブルートフォース攻撃）（IP アドレス）（閾値 5）

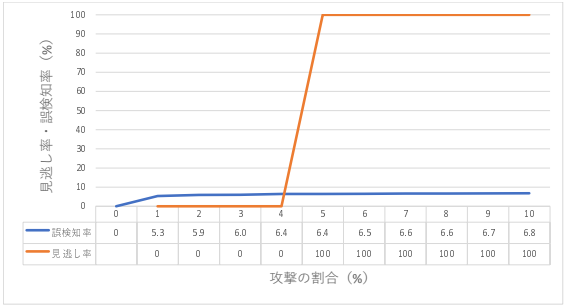

図 5 既製品の IoT 機器による誤検知率と見逃し率の推移

5 評価

5.1 評価基準

本研究では, 以下の 2 つの評価基準の元, アノマリ型侵入検知手法の評価を行う.

1. 攻撃の割合に対しての見逃し率の評価
2. 攻撃の割合に対しての誤検知率の評価

5.2 各通信パターンの評価

各通信パターンの評価をするにあたり, 閾値は 5, 攻撃は DoS 攻撃とし, 変化点に着目する. 変化点は, 見逃し率と誤検知率の数値が急激に変化する点である.

1. 表 2 の結果より, 見逃し率の観点からは以下の条件の時アノマリ型侵入検知手法が有効であるといえる.
   - IP アドレスは攻撃の割合が約 9% 以下の場合
   - 通信量は攻撃の割合が約 7% 以下の場合
   - タイミングは攻撃の割合が約 8% 以下の場合

表 2　通信パターンと攻撃の割合に対する見逃し率の変化点（閾値 5）

	変化点前	変化点後	変化点の攻撃の割合	変化点の誤検知率
IP アドレス	0%	約 64%	約 10%	0%
通信量	0%	約 83%	約 8%	0%
タイミング	0%	100%	約 9%	約 2%

表 3　通信パターンと攻撃の割合に対する誤検知率の変化点（閾値 5）

	変化点前	変化点後	変化点の攻撃の割合	変化点の見逃し率
IP アドレス	約 35%	100%	約 85%	約 99%
通信量	約 23%	100%	約 90%	約 99%
タイミング	約 11%	100%	約 91%	約 97%

2. 表 3 の結果より, 誤検知率の観点からは以下の条件の時アノマリ型侵入検知手法が有効であるといえる.
   - IP アドレスは攻撃の割合が約 84% 以下の場合
   - 通信量は攻撃の割合が約 89% 以下の場合
   - タイミングは攻撃の割合が約 90% 以下の場合

評価基準 1 と評価基準 2 より, 見逃し率と誤検知率の変化点に対する攻撃の割合が大きいほど, アノマリ型侵入検知手法に適した通信パターンといえる. 図 6 より, 攻撃の割合に対するアノマリ型侵入検知手法の有効範囲の広さは IP アドレスが 1 番広いため, どの通信パターンでもアノマリ型侵入検知手法は有効ではあるが IP アドレス, タイミング, 通信量の順に有効であることが分かった.

5.3　既製品の IoT 機器を用いての評価

既製品の IoT 機器と作成した IoT 機器によるアノマリ型侵入検知手法の有効性の評価方法は同じとする. 実験結果より, アノマリ型侵入検知手法の有効範囲は, 攻撃の割合が 4% 以下の場合であることが分かった. このことより既製品の IoT 機器でもある一定の範囲までアノマリ型侵入検知手法の有効性が確認できた. そして, 我々が本研究で定義した通信パターン基準の元で 2 つの通信パターンをもつ IoT 機器に対しアノマリ型侵入検知手法の有効性を示すことが出来た.

5.4　総合評価

今回の実験より攻撃の割合が増えるにつれ見逃し率, 誤検知率が以下のような振る舞いを行った.

1. 今回の実験の閾値や攻撃において, 攻撃の割合が 5% から 20% において見逃しが多くなり, 見逃し率が約 60% に達する. そこから攻撃の割合が 25% から 30% において, 見逃し率が上昇し, 100% に近づいていく.
2. 今回の実験の閾値や攻撃において, 攻撃の割合が 50% 未満だと誤検知率を約 25% 程度に抑えることが出来るが, そこから攻撃の割合が増加するにつれ, 誤検知率が上昇する. 攻撃の割合が 80% から 90% 以上になる時, 誤検知率が 90% から 100% に達することが多い.

結論として, どの 2 つの通信パターンをもつ IoT 機器,

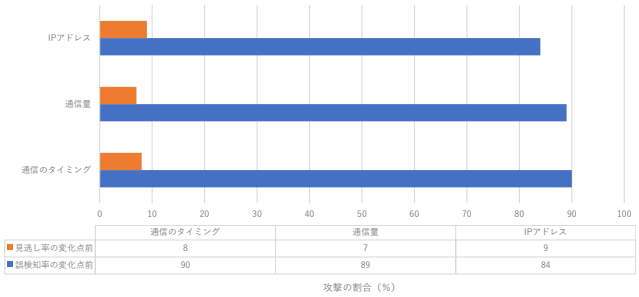

図 6　各通信パターンのアノマリ型侵入検知手法の有効範囲

閾値, 攻撃でも, 攻撃の割合 1%（6 時間のうちなら 3.6 分）までなら, 見逃し率 1% 未満, 誤検知率 10% 未満となることが分かった. なお, 誤検知率が見逃し率より割合が高い結果となった. これはアノマリ型の欠点である誤検知が多い点が反映されており, 閾値の選定が重要だと考えられる.

6　まとめ

本研究では, 瀧本らの手法を元にアノマリ型侵入検知手法の評価システムの構築を行い, Raspberry Pi 4 を用いて作成した IoT 機器と既製品の IoT 機器を評価対象とした. 結果として, アノマリ型侵入検知手法は, 本研究で定義した通信パターン基準の元で 2 つの通信パターンをもつ IoT 機器に対して攻撃の割合 1%（6 時間のうちなら 3.6 分）までなら, 見逃し率 1% 未満, 誤検知率 10% 未満となることが分かった.

今後の課題としては, マルウェアを用いた実験が本大学のセキュリティの関係上出来なかったので, その検証が必要である. また, 本研究ではアノマリ型侵入検知手法の評価のみなので, 今後アノマリ型検知手法を実装する試みが必要になっていくと考えられる.

参考文献

[1] トレンドマイクロ株式会社 is702, 2020. https://www.is702.jp/news/3748/.

[2] 桂井銀河, 向井宏明. IoT ネットワーク向け侵入検知システム. 信学技報, CS2020-12, pp. 49–52, 2020.

[3] 中原正隆, 奥井宜広, 小林靖明, 三宅優. Isolation Forest を用いた IoT デバイス向けマルウェア感染検知. 暗号と情報セキュリティシンポジウム 2020 論文集, 2020.

[4] 瀧本達也, 稲葉宏幸. IoT 機器に特化したアノマリ型侵入検知システムの提案. コンピュータセキュリティシンポジウム 2018 論文集, pp. 443–447, 2018.

[5] 瀧本達也, 稲葉宏幸. 深層距離学習を用いた IoT 機器に対するアノマリ型攻撃検知システムの提案. 信学技報, CS2020-12, pp. 13–18, 2020.

[6] 丹羽美乃, 梶克彦. IoT デバイスの時系列通信パターンの分析. 情報処理学会, pp. 227–228, 2018.