# 組込みシステム向けのポータブルDoSテストツール

長柄 啓悟[†1,a)]　青木 克憲[†1]　松原 豊[†1]　高田 広章[†1]

**概要**：近年 IoT 機器が注目され，その脆弱性が明らかになっている．組込みシステムの脆弱性とその対策を検討するため，既存マルウェアである *Mirai* による攻撃性能を評価した．*Mirai* とは，組込み Linux 上で動作するコンピュータをボット化し，DDoS 攻撃を行うマルウェアである．そして，この Miai を利用してポータブル DoS テストツールを作成し，IoT 機器のプロトタイプを対象に DoS テストを行った．ツールには負荷の可視化を行う機能や攻撃パケット量を調整する機能を加えた．このようなツールを用いることで IoT 機器の脆弱性を確認したり，実際の DoS 攻撃の様子について学習したりすることができる．

**キーワード**：組込み/IoT デバイス，セキュリティ，テストツール開発，DoS 攻撃

## 1. はじめに

近年 IoT（Internet of Things）機器が注目され，多種多様な組込みシステムがネットワークに繋がるようになっている．同時にそれら機器の脆弱性が明らかになっている．その脆弱性の悪用例として，IoT マルウェア *Mirai* が挙げられる．*Mirai* とは，組込み Linux が動作する，デジタルビデオレコーダーやネットワークカメラ，家庭用ルータなどの IoT 機器を対象に感染し，それら機器から DDoS（分散型サービス妨害）攻撃を行うマルウェアである．2016 年 9 月にはセキュリティブログへの 620 Gbps の攻撃が観測された．また，同年 10 月の DNS サーバープロバイダである Dyn 社への攻撃では，史上最大規模である 1.2 Tbps の攻撃が観測された [2]．

公開されている *Mirai* のソースコード [3] を元に *Mirai* の攻撃性能について評価したところ，*Mirai* には多彩な攻撃方法や高い攻撃性能があることが確認された [4]．*Mirai* のような DoS（サービス妨害）攻撃による被害事例は増加しており，組込みシステムも，DoS 攻撃を行う踏み台となる他に，その組込みシステム自体が DoS 攻撃の対象になる可能性がある．そのため，組込みシステムにおいて，攻撃時でもセーフティに動作もしくは停止する必要がある．よって，本研究では，この *Mirai* を利用して IoT 機器に対して DoS 攻撃の耐性をチェックするためのポータブルテストツールを作成した．

既存のツールには，LOIC や Slowloris といった DoS ツールや，Load Impact や curl-loader といった Web サイト負荷テストツールがあり，これはパソコンをベースに動作するもので，その対象は主に Web サイトである [5][6][7]．これらのツールを参考に，*Mirai* にはない攻撃性能調整機能を追加し，組込みボードである odroid-c2 を用いた DoS テストツールを作成した．既存ツールと本ツールの機能について比較したものが次頁の表 1 になる．このように odroid-c2 を用いた本ツールにはメリットが多く存在する．また，Web サイトと比べ，DoS 攻撃とそれによる IoT 機器への影響は分かりづらいため，DoS に対する脆弱性が視覚的に把握しやすいように負荷の可視化機能を追加した．組込みボードを用いることで 3 つのメリットがある：1) モニタリングするパソコンのリソースがストレステストに割かれる必要がない，2) 小型で省電力なため持ち運びにも向き， 3) 安価にスケールする．

## 2. ポータブル DoS テストツール

### 2.1 概要

本節では，本研究で作成したポータブル DoS テストツールの全体像について説明する．本ツールは，図 1 のようにアタッカーとモニターからなる．アタッカーで，対象機器に DoS 攻撃を行い，そのトラフィック情報をモニターに送

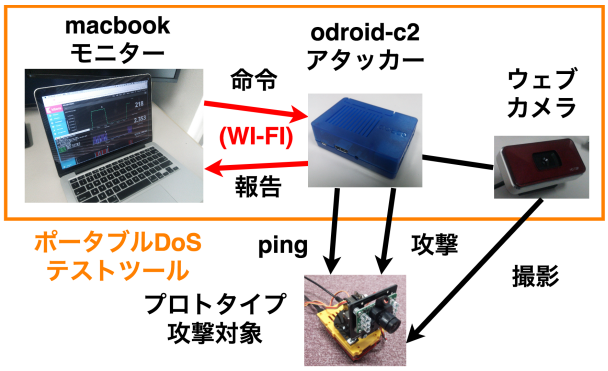


図 1　本ツールの概要図

†1　現在，名古屋大学
a)　nagara@ertl.jp

表 1　DoS テストツールの機能要件の比較

| ツール名 | LOIC | Load Impact | curl-loader | 本ツール |
|---|---|---|---|---|
| 攻撃プロトコル | UDP | HTTP | HTTP, HTTPS | UDP, TCP, HTTP |
| 攻撃調整機能（帯域と時間） | ✓ | ✓ | ✓ | ✓ |
| 可視化機能 | | ✓ | | ✓ |
| カメラ | | | | ✓ |
| サイズ | | | | ✓ |
| リソース消費 | | ✓ | | ✓ |
| コスト | PC | PC & 1 回 3 $ | PC | PC & odroid-c2 |

信した．モニターで，PC 上で動作し攻撃の指示と負荷の可視化を行った．攻撃指示はモニターからを通じ Command Line Interface (CLI) にてアタッカーに指示を送ることで行った．また，モニターで，攻撃対象に ping を送った時のその応答情報とアタッカーから送られてくるトラフィック情報について Elasticsearch と kibana を用いて，可視化を行った．Elasticsearch は Elastic 社のデータ格納及び解析ツールで，kibana は Elasticsearch のログデータをブラウザ上で，グラフや表で視覚的に表示するツールである．本研究では，アタッカーには組込みボードである odroid-c2（ubuntu 16.04 LTS Xenial Xerus）を使用し，モニターには PC（macOS Sierra 10.12.3）を使用した．アタッカーとモニターの IP アドレスは攻撃対象の機器と同じネットワーク下になるように設定する必要がある．

本ツールでは，DoS 攻撃に使用される大量のリソースを odroid-c2 が肩代わりするため，重い処理である DoS 攻撃中でも PC で他のタスクを行うことができた．また，本ツールのモニター機能によって，その PC のリソースを利用して，攻撃トラフィック量や ping 応答時間といった負荷情報を確認することができた．また，odroid-c2 は安価で省電力で小型（40 g，85 × 56 mm）であるため，持ち運びがしやすく，IoT 機器への DoS テストに向いている．

### 2.2　攻撃機能

本節では，本ツールの攻撃機能の詳細について説明する．

#### 2.2.1　*Mirai* の攻撃性能

DoS 攻撃の機能は *Mirai* の機能を利用した．*Mirai* は表 2 のような攻撃の種類を持つ [4]．ARM プロセッサを搭載したシングルボードコンピュータである Hardkernel 社の odroid-c2 を使用し，*Mirai* の攻撃性能を評価した結果が図 2 になった．攻撃台数が 1 台の場合の転送速度は 127.18 Mbps で，2 台の場合の転送速度は 247.04 Mbps で，3 台の場合の転送速度は 378.75 Mbps で，4 台の場合の転送速度は 510.60 Mbps で，5 台の場合の転送速度は 648.54 Mbps と，台数に比例して攻撃性能が増加していた．この時，攻撃を行う組込みボード上の 1 コアでの *Mirai* の CPU 使用率が 95 %を超えていることを top コマンドにより確認した．

また，他の攻撃方法について実機で行なった場合の攻撃性能をまとめたものが表 3 になった．この表から攻撃台

表 2　*Mirai* の攻撃の種類

| 攻撃名 | 詳細 |
|---|---|
| UDP 攻撃 | UDP パケットを大量に送る |
| VSE 攻撃 | ゲームエンジンに対して UDP パケットを大量に送る |
| DNS リゾルバ攻撃 | DNS に存在しないドメイン名の名前解決要求をする |
| SYN 攻撃 | SYN パケットを大量に送る |
| ACK 攻撃 | ACK パケットを大量に送る |
| STOMP 攻撃 | TCP セッション確立後に ACK パケットを大量に送る |
| GRE IP 攻撃 | GRE プロトコルによるパケットを大量に送る |
| GRE イーサネット攻撃 | イーサネットと GRE プロトコルによるパケットを大量に送る |
| プレーン UDP 攻撃 | 高速化のために最適化した UDP パケットを大量に送る |
| HTTP 攻撃 | HTTP リクエストを大量に送る |

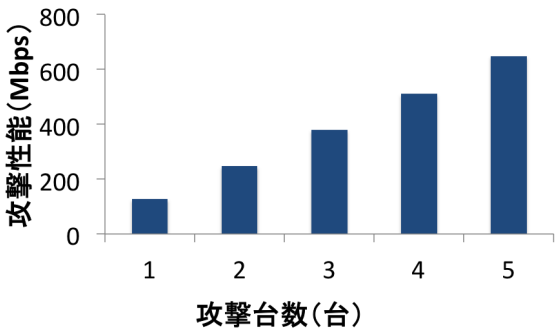


図 2　攻撃台数を変化させた時の UDP フラッド攻撃の性能

表 3　攻撃台数が 1 台と 5 台の場合の各攻撃の転送速度（Mbps）

| 攻撃方法 | 1 台 | 5 台 |
|---|---|---|
| UDP | 127.18 | 648.54 |
| VSE | 36.47 | 184.26 |
| SYN | 23.04 | 113.46 |
| ACK | 98.50 | 612.74 |
| GRE IP | 198.75 | 963.53 |
| GRE ETH | 200.67 | 966.12 |
| UDP PLAIN | 248.97 | 959.91 |
| HTTP | 2.22 | 11.04 |

数に比例して攻撃性能が推移していることが分かった．プレーン UDP フラッド攻撃においては，攻撃台数が 2 台の時は 462.43 Mbps，3 台の時は 749.74 Mbps と，攻撃台数に比例して攻撃性能が推移していたが，4 台の時は 959.74 Mbp であり，LAN ハブの回線の最大値（1 Gbps）が原因で，4 台目以降は測定できる攻撃性能が頭打ちになってしまったと考えられる．そのため，回線による制限を無視すれば，4 台以上の攻撃台数の場合も攻撃台数に比例した分のパケットが送ることが可能だと考えられ，実際のマルウェアの DDoS 攻撃の威力は，攻撃に使用される IoT 機器の性能と台数によると予測できる．また，*Mirai* の攻撃の中で最大のものはプレーン UDP フラッド攻撃であり，その数値は 247.84 Mbps であり，攻撃ボットの台数に比例して攻撃性能が増加していることが確認された．IoT 機器のネットワークインターフェースが主に 100 Mbps 程度であることを考えると，この攻撃性能値は十分であると言える．

#### 2.2.2 攻撃調整機能

新たな機能として攻撃帯域微調整機能を取り入れた．この機能は nanosleep 関数を DoS 攻撃の間に一定間隔で挟むことで実現した．この機能を使用した時の攻撃性能の推移は図 3 のようになった．nanosleep の呼び出しや判定により多少の誤差がある．攻撃性能を微調整することで IoT 機器の負荷の限界値を確認することができる．

### 2.3 可視化機能

#### 2.3.1 概要

DoS 攻撃とその影響を確認するには，ミラーリング機能で攻撃パケットを拾いその様子を横から Wireshark といったパケットキャプチャツールで観察するか，攻撃対象である IoT 機器の様子を観察するといった方法が考えられる．しかし，前者ではノウハウが必要で一般ユーザには向かず，後者では機器の種類に依存するため製品によっては負荷が把握しづらいことが予想される．そのため，Elasticsearch と kibana による視覚的に理解しやすくブラウザベースで使いやすい負荷の可視化機能を追加した．

#### 2.3.2 ping 応答時間の可視化

攻撃対象となった IoT 機器の負荷を調べる単純な方法として ping 応答時間が挙げられる．対象に ping を送った時に，プログラムが正常に稼働していれば応答がすぐに返ってくるが，DoS 攻撃による負荷がかかっていると応答が遅いことや返ってこないこと，ping を送ること自体ができないことがある．これらを指標にどれくらいの負荷が対象機器にかかっているのかということがわかる．

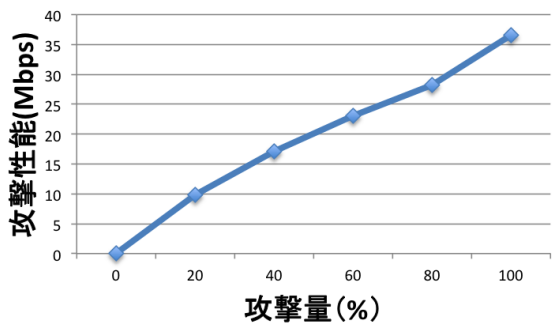


図 3 攻撃調整機能（VSE 攻撃）

python-ping[14] を参考にし，返ってきた応答時間と現在時刻を Elasticsearch に書き込むプログラムを実装した．応答がない場合はエラーメッセージと現在時刻を Elasticsearch に書き込み，kibana での表示では，応答時間が 10 秒であると固定して色を変え区別した．

#### 2.3.3 トラッフィック量の可視化

*Mirai* の DoS 攻撃は大量にパケットを送るフラッド型の攻撃であり，テスト時このパケット量は上記の実験の値から予測はできるが，実際の値を把握するには，ミラーリング機能を持った LAN ハブや Wireshark といったパケットキャプチャツールが入った PC が必要となる．また，パケットキャプチャツールにおいては大量のパケットを処理するために，PC にかかる負荷も大きくなる．そこで，アタッカーでトラフィック量を取得しモニターでそれを受け取り可視化する機能を追加した．

odroid-c2 上でトラフィック量を取得しモニターに送信するプログラムと，モニターで受信したデータを Elasticsearch に書き込むプログラムを実装した．アタッカーでは，通信量と時刻を読み取り，モニターにデータを送信した．モニターとの時刻のずれの補正も，このプログラムが行った．モニターで，アタッカーが送信してきた通信量と時刻情報を受け取った．このアタッカーとモニターの通信は，DoS 攻撃の大量のパケットによりロスすることがあるため，攻撃に使用する LAN ポートと異なるインターフェース（例えば Wi-Fi）を用いる必要がある．

2 ホスト間の時刻同期ではアタッカーの時刻をモニターの時刻に合わせるように実装した．前提として，モニターはアタッカーよりも時刻が dt だけ進んでいるとし，2 ホスト間のネットワークのレイテンシーは dn で一定とし，プログラムの処理時間を 0 とした．この場合に，アタッカーは現時刻 t1 をモニターに送信し，これを受信したモニターは直ちに現時刻 t2 を X に送信し，これを受信した時のアタッカーにおける現時刻を t3 とすると，t2 - t1= dt + dn，t3 - t1 = 2 * dn より，アタッカーの時刻が t の時モニターの時刻は t + dt = t + t2 - t1 / 2 - t3 / 2 となった．この時刻同期プロトコルは誤差が少数 3 桁以降で発生するが，これは十分な精度であるため，時刻同期方法として本手法を採用できると判断した．

#### 2.3.4 カメラ

IoT 機器によって，画面表示やランプからその動作を確認できる．アタッカーはカメラで 1 秒ごとに撮影し，その画像が前の画像と異なるならばモニターに送信した．最新

のキャプチャされた画像を表示するメトリックウィジェットと，数直線上に表示されている撮影ポイントの画像を表示する専用ウィジェットを作成した．これにより，負荷情報と製品の状態をモニターで同時に確認でき，アタッカーを遠隔操作する場合でもそれらを確認して記録できる．

## 3. 活用事例

この章では，静的な組込みシステムのプロトタイプに対する攻撃例について述べる．このプロトタイプのボードは GR-PEACH であり，OS は TOPPERS/ASP（Release 1.9.2）であり，TCP/IP スタックは LwIP（version 1.4.0）である．このプロトタイプは，カメラに映っている映像を http サーバ上で提供する機能を持つ．

このプロトタイプに対して UDP, VSE, UDP PLAIN 攻撃などを行うと攻撃中提供する映像が停止した．VSE フラッド攻撃を行い，その様子を kibana やプロトタイプが提供するサイト上の様子から確認した．VSE 攻撃は攻撃性能微調整機能を用いて，20%, 40%, 60%, 80%, 100%と性能を変え，それぞれ 30 s の攻撃を行なった．この攻撃の様子を kibana のダッシュボードで可視化した画面が図 4 になる．20%から 60%まで攻撃トラフィック量がスケールし，それに伴って ping 応答時間が長くなったり，応答がなくなる頻度が増加したりしている様子が確認できた．提供されるサイトの映像では，20%, 40%と攻撃性能が上がるほど，映像の提供が遅れ，60%では攻撃終了後まで映像が固まった．また，100%の攻撃中にサイトで更新を行うとシステムが破損し図 5 のように ping の応答がなくなり，サイトに接続することができなくなる様子が確認できた．プロトタイプ自体を再起動することでサービスは再び提供できるようになったが，DoS 攻撃中にユーザが動作の停止を見てサイトの更新を行う可能性は高く，この機器での DoS 攻撃による影響が大きいことが確認された．IoT 機器として，最低限攻撃終了後もシステムが再開できるという要件を満たす必要がある．

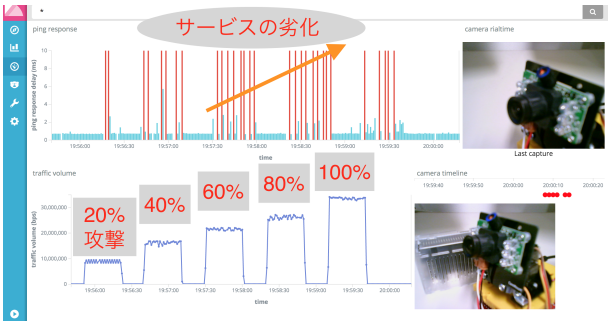


図 4　攻撃調整機能によるプロトタイプに対する攻撃

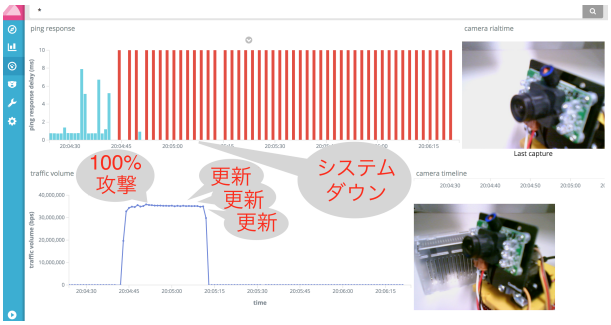


図 5　攻撃によってシステムがダウンする場合

## 4. おわりに

### 4.1 まとめ

本研究では，マルウェア *Mirai* の概要やその攻撃性能について調査し評価し，これを利用してポータブル DoS テストツールを作成した．攻撃指示機能における GUI への変更や IP アドレスの設定の簡略化といった本ツールの機能の充実が今後の課題といえる．

**謝辞**　本研究の一部は JSPS 科研費 16K21097 の助成を受けて行われた．

### 参考文献

[1] IPA：つながる世界のセーフティ＆セキュリティ設計入門，IPA（2015）．
[2] 齋藤衛：IIJ Technial WEEK 2016 セキュリティ動向 2016，入手先 ⟨http://www.iij.ad.jp/company/development/tech/tech-week/pdf/161111_01.pdf⟩（2017.1.27）．
[3] jgamblin/Mirai-Source-Code - GitHub，入手先 ⟨https://github.com/jgamblin/Mirai-Source-Code⟩（2016.10.27）．
[4] 長柄啓悟，松原豊，青木克憲，高田広章：組込みシステム向けマルウェア *Mirai* の攻撃性能評価，第 217 回 システム・アーキテクチャ研究発表会（2017）．
[5] Sauter, M: “LOIC Will Tear Us Apart” The Impact of Tool Design and Media Portrayals in the Success of Activist DDOS Attacks. American Behavioral Scientist, 57(7), 983-100.7. (2013).
[6] Load Impact: Performance testing for DevOps. 入手先 ⟨https://loadimpact.com/⟩，（2017.4.13）．
[7] Welcome to curl-loader. 入手先 ⟨http://curl-loader.sourceforge.net/⟩，（2017.4.13）．
[8] 新井悠，岩村誠，川古谷裕平，青木一史，星澤裕二：アナライジング・マルウェア，オライリー・ジャパン（2010）．
[9] IPUSIRON：ハッカーの学校，株式会社データハウス（2015）．
[10] Adam Shostack：*threat modeling designing for security*（2014）．
[11] 日経 BP マーケティング：IoT セキュリティ～インシデントから開発の実際まで～，日経 BP イノベーション ICT 研究所編，日経 BP 社（2016）．
[12] IPA：IoT 開発におけるセキュリティ設計の手引き，IPA（2016）．
[13] MMD-0056-2016 - Linux/Mirai, how an old ELF ... - Malware Must Die!，入手先 ⟨http://blog.malwaremustdie.org/2016/08/mmd-0056-2016-linuxmirai-just.html⟩（2017.1.27）．
[14] python-ping 2011.10.17.376a019 : Python Package Index，入手先 ⟨https://pypi.python.org/pypi/python-ping/⟩（2017.3.24）．