平成 30 年度 修士論文

ファームウェア解析によるIoT機器のデフォルトパスワードに関する実態調査

Research on the default password of IoT devices through the firmware analysis

指導教員　森 達哉 教授

早稲田大学　基幹理工学研究科　情報理工・情報通信専攻

学籍番号 5116F074

飯田　雄太

平成 31 年 2 月 1 日

概要

IoT(Internet of Things) の進展にともない，そのセキュリティが問題となってきた．本研究では，IoT 機器によく使われるパスワードを試す攻撃に対して，現状どの程度のリスクがあるのかをファームウェア解析によって明らかにしている．また，そこで用いたファームウェア解析自体の有用性の評価も行っている．その結果，ベンダ側の対策は進んでいるものの，攻撃に対して脆弱であり，対策が不十分である現状が明らかになった．同時に，ファームウェア解析は有用であるが，様々な制約とともに課題もあることが分かった．

目次

第 1 章　序論　11

1.0.1　はじめに 11

1.1　研究背景 12

1.2　研究の目的 13

1.3　貢献 13

1.4　論文の構成 13

第 2 章　関連研究　15

2.1　IoT のセキュリティ 15

2.2　ファームウェア解析 15

第 3 章　手法　17

3.1　ファームウェアの収集 17

3.2　ファームウェアの展開 17

3.3　静的解析 17

3.4　動的解析 18

第 4 章　実験　19

4.1　データの収集および解析の準備 19

4.2　パスワードファイルの調査 19

4.3　動的解析による Web インタフェースの調査 19

4.4　Web インタフェースのログイン情報の調査 21

4.5　Web インタフェースのパスワード変更手順の調査 22

第 5 章　結果　23

5.1　統計 23

5.2　パスワードファイルの調査結果 23

5.3　Web インタフェースのログイン情報 24

5.4　Web インタフェースのパスワード変更方法 25

5.5 Web インタフェースの脆弱性 26

第 6 章 議論 29
6.1 考察 29
6.2 制約 30
6.3 今後の課題 31
6.4 倫理的な議論 32

第 7 章 まとめ 33

参考文献 37

図目次

4.1 CSRF 20
4.2 DNS Rebinding 攻撃 21

表目次

5.1 収集および展開したファームウェアに関する統計 23
5.2 ファームウェアのアーキテクチャ 23
5.3 パスワードファイルの調査結果..... 24
5.4 Web インタフェースのログイン名 (2018 年 12 月) 24
5.5 Web インタフェースのパスワード (2018 年 12 月) 25
5.6 Web インタフェースのログイン名 (2017 年 2 月)..... 25
5.7 Web インタフェースのパスワード (2017 年 2 月)..... 25
5.8 Web インタフェースのパスワード変更方法 (2018 年 12 月) 26
5.9 Web インタフェースのパスワード変更方法 (2017 年 2 月)..... 26
5.10 CSRF および DNS Rebinding 攻撃への対策状況 (2018 年 12 月)..... 26
5.11 CSRF および DNS Rebinding 攻撃への対策状況 (2017 年 2 月) 27

第 1 章　序論

本章では，研究の背景と目的およびその概要について述べる

1.0.1　はじめに

あなたの家には，インターネットにつながったモノがいくつあるだろうか．IoT (Internet of Things) と呼ばれる概念は，あらゆるモノがインターネットにつながる未来を見据えた世界の青写真である．様々なモノがインターネットにつながると，わたしたちの社会はより便利になる．しかし，こうしたモノが別の人によって好きなように悪用されてしまう世界は恐ろしいものになるだろう．たとえば，ペットを見守るために設置したカメラが他人から見られてしまうとどういうことが起こるか．あなたのプライバシーがのぞかれるだけでなく，カメラから外出中であることを確認し，空き巣に入られるかもしれない．こうしたリスクは空想のものではなく，現実に考えられなければならない．2016 年には Mirai とよばれる IoT 機器を標的としたマルウェアによって，何台ものネットワークカメラが乗っ取られ，他の攻撃に悪用される事件があった．その攻撃が成功した根底にあるのは，パスワードをデフォルト設定のまま運用されている等，基本的なセキュリティがおろそかになっていた点である．Mirai の事例が示すように，IoT の概念が普及し，発展していく上では，セキュリティの問題は避けて通れないものである．

本研究は，よく使われるパスワードを試す攻撃 (パスワードリスト攻撃) に対して，現状の IoT 機器がどのくらいリスクがあるかを明らかにすることを目的とする．そのような調査をする上で，実際の機器を入手するアプローチはスケーラビリティに欠ける．そこで，機器に組み込まれたソフトウェアであるファームウェアの解析を行うことに本研究の特徴がある．その結果，設定がデフォルトのままで運用される機器が多いこと，またそれらの機器がパスワードリスト攻撃に対して脆弱である現状を明らかにした．また，外部に直接公開されていない機器に対する攻撃手法への対策が不十分な機器が多いことも明らかにした．その一方で，初期設定時に強制的にパスワードを変更させる対策を施している機器が増えており，パスワードリスト攻撃に対するベンダの認識が進んでいることも明らかにした．同時に，IoT 機器の現状調査にファームウェア解析がある程度は有効であることも示した．加えて，ファームウェアを入手することが難しい場合や，解析することが容易ではないケースが存在することも明らかになった．そのような場合，実機の調査も組み合わせることが必要である．

1.1 研究背景

■IoT　IoT(Internet of Things) という言葉が広く聞かれるようになった．IoT とは，今までインターネットにつながれてこなかった様々なモノをインターネットにつなげることで，より柔軟な使い方をしようという考え方である．身近にあるスマートフォンやタブレット端末から農業，ヘルスケア，自動車，住宅，建設，製造，小売，金融とその活用範囲を上げればきりがない．IoT を活用することで，様々な機器が相互に情報のやり取りを行うことができ，その活用の幅が広がる．ドイツのように国家プロジェクトとして IoT を推進している国もある．IoT 機器自体の増加も著しく，2016 年時点では約 60 億個であったのが，2020 年には約 200 億個まで増えると予想されている (Gartner による予想)[1]．IoT が様々な場面で実現されるようになった背景には，ハードウェアを安価に利用できるようになったことや，無線通信技術の発達，人工知能などを利用した大規模データ分析技術の発達が挙げられる．IoT は第 4 次産業革命ともいわれており，単にモノをインターネットにつなげるということを越えて，社会を根本から変える潜在性を秘めているともいえる．

■IoT のセキュリティ　しかし，IoT がより広く社会で利用されていくには解決されなければならない様々な問題がある．そのひとつはセキュリティの問題である．今までインターネットにつながれてこなかった機器であるだけに，セキュリティが考慮されていないものも多い．また，こうした機器はセキュリティ機能を搭載するだけの余裕がない場合も多い．このような無防備な機器がインターネット上に広く存在することにより，それらが別の攻撃の踏み台になりうる．さらに IoT 機器の中には，制御機器のように現実世界にフィジカルな影響を与えるものもあり，こうした機器が攻撃された場合の被害は計り知れない．実際に多種多様な IoT 機器がインターネットからアクセス可能であることが指摘されている．たとえば，SHODAN[2] というサイトでは，インターネットに公開状態である機器を検索することができる．似たような例として，insecam[3] というサイトでは，公開状態にあるネットワークカメラを検索することができる．機器によっては，インターネット越しに操作できてしまうものもある．こうした機器では，パスワードをデフォルトのままで運用しているといった根本的なセキュリティがおろそかにされている．

■マルウェア Mirai の登場　セキュリティの甘い IoT 機器が攻撃に悪用される事件が実際に起きている．2016 年 10 月，Amazon や Twitter，Netflix といった有名なサイトが約 5 時間にわたって接続不良を起こすという事件が発生した．原因は，Mirai とよばれるマルウェアに感染した IoT 機器からの攻撃である．Mirai は，パスワードがデフォルトのままで運用されているルータやネットワークカメラに対して，よく使われるパスワードを総当りで試すこと (パスワードリスト攻撃) で侵入し，乗っ取り，広まっていく．乗っ取られた機器は，標的とするサーバに一斉にアクセスをかけることでダウンさせる．この事件では，Dyne という DNS サーバが標的にさ

れ，それを利用していた各種サイトも影響を受けることとなった．その後，Mirai のソースコード [4] が公開されたことにより，様々な亜種が出現している．IoT 機器のセキュリティが改善されない限り，こうした事件は再び起こることが予想される．

1.2 研究の目的

本研究の目的は以下の 2 点である。

- IoT 機器へのパスワードリスト攻撃に対し，現状どのくらいのリスクがあるのか明らかにすること
- 実機を用いない手法として，ファームウェアの分析がどの程度有用であるのかを評価すること

1.3 貢献

本研究における貢献は以下の通りである。

- ファームウェアの分析を通し，パスワードリスト攻撃に対する対策の現状を明らかにした
- ファームウェア分析の有用性とその制約や課題を明らかにした

1.4 論文の構成

まず，第 1 章では研究の背景と目的について述べる．第 2 章では，本研究に関連する研究について述べる. そして，第 3 章では研究中で用いた手法について述べている．続く第 4 章，第 5 章では研究で行った実験およびその結果，第 6 章では実験結果に対する考察について述べる. 最後の第 7 章においては，全体のまとめとして結論を述べている．

第 2 章　関連研究

本章では，IoT のセキュリティやファームウェア解析に関連する研究について述べる

2.1　IoT のセキュリティ

IoT 機器のセキュリティに焦点を当てた研究として Yin Minn Pa Pa らの研究がある [5]．この研究では，IoT 機器の telnet ポートに見せかけたハニーポットを作成し，実際に運用している．そこでは，IoT 機器に対してどのように攻撃が行われるかが示されている．また，個別の機器を狙ったセキュリティリスクを評価した研究もいくつかある．たとえば，Ian Foster らは自動車の内部ネットワーク CAN(Car Area Network) と接続することで様々なサービスを提供する TCU(Telematic Control Unit) への攻撃の可能性が示している [6].

2.2　ファームウェア解析

firmware の静的解析を大規模に行った研究として Daming D. Chen らの研究がある [7]．その中で，ファームウェアの収集から展開，静的解析，ファームウェア間の相関関係を調べるフレームワークを実装し，評価を行っている．この研究を通して，同じ脆弱性が複数の機器にわたって存在することが指摘されている．一方，動的解析を行ったものとして Andrei Costin らの研究がある [8]．こちらは，Linux ベースのファームウェアの動的解析ツール Firmadyne を提案し，実装，評価を行っている．このツールにより，14 の未発見の脆弱性が見つかっており，動的解析の有用性が示されている．以上に挙げた研究は，解析手法そのものの評価であるが，そうした手法を実際の脆弱性発見に応用した研究もある．Andrei Costin らは，COTS 製品の Web インタフェースに焦点を当て，静的解析，動的解析を行っている [9]．ファームウェアのソースコードを調査する手法を提案している研究もある．Drew Davidson らは，FIE というツールを提案，実装している [10]．このツールは MSP430 プロセッサファミリー向けのファームウェアのソースコードをシンボリック実行によって調査をするツールである．

第 3 章　手法

本章では，研究中で用いたファームウェア解析技術について述べる

3.1　ファームウェアの収集

ファームウェアの収集は各ベンダの公式 Web サイトから行った．ベンダの公式サイトでは，各製品のページからファームウェアをダウンロードすることができる．本研究では，大規模に収集する目的から，手動ではなく自動で収集することにした．しかし，ダウンロードサイトは JavaScript で動的に生成されている例も多く，自動収集が困難であった．そこで，Web ブラウザを自動で操作することができる Selenium[11] というツールを利用した．Selenium は，主に web アプリケーションのテストに使われるツールであるが，Web サイトのクロールにも利用される．このツールは C#や Java，Perl，PHP，Python，Ruby と様々なプログラミング言語から利用できるが，データベースアクセス用などのその他ライブラリも利用したかったことから Python を用いた．収集したファームウェアは，ダウンロード先 URL やファイルの更新時間，バージョンなどのメタデータとともにデータベースで管理するようにした．ベンダによっては更新時間やバージョンの情報が得られないこともあった．

3.2　ファームウェアの展開

ダウンロードされたファームウェアは通常，圧縮されている．そのため，解析するためにはまず，展開する必要がある．使用されている圧縮フォーマットとしては，各ベンダが独自に実装していると思われるものや，一般的に用いられるフォーマットのものもある．ファームウェアを展開するツールとしては，様々なものが存在するが，本研究では，ファームウェアの動的解析プラットフォーム Firmadyne[12] に含まれる展開用ツールを用いた．この展開ツールには binwalk[13] というファームウェア分析ツールの API が利用されている．

3.3　静的解析

静的解析は，コードを実行せずに解析を行う手法である．ファイルのメタデータを分析したり，実行ファイルのバイナリを解析するなどの方法が含まれる．一般に，後述する動的解析に比

べて解析に時間がかかるが，その分詳細な情報を得ることができる．本研究ではおもに，ファイルシステムからパスワードファイルを抽出し分析している．この範囲では，ある程度自動化することができ，必要な情報を分析することができる．解析には，ファームウェア収集と同様，Python を利用している．

3.4 動的解析

動的解析は，コードを実際に実行して解析を行う方法である．コードを実行する際に仮想的な環境上で解析を行うことも多い．一般に，静的解析に比べて解析の手間はかからないが，あまり詳細な情報が得られない場合も多い．本研究では，ファームウェア中に含まれる Web インタフェースの脆弱性を分析するために動的解析を行っている．解析には，ファームウェアの動的解析プラットフォーム Firmadyne を利用している．Firmadyne では，各アーキテクチャに対応した QEMU[14] というオープンソースのエミュレータが用いられている．そのため，実環境での解析と異なる部分もあることには注意が必要である．

第 4 章　実験

本章では，COTS 製品のファームウェアを収集し，行った実験について述べる

4.1　データの収集および解析の準備

■ファームウェアの収集　COTS(Commercial Off-The-Shelf) 製品を提供する 18 のベンダの Web ページからファームウェアを収集した．収集した範囲としては，ルータ，ネットワークカメラ，無線 LAN アクセスポイント，無線 LAN 中継機器などが含まれる．

■ファームウェアの展開　収集したファームウェアは圧縮されているので，展開を行った．このとき，展開が成功し，ルートファイルシステムが得られたものに関して，パスワードファイルの調査を行った．同時に，展開が成功したものに関して，アーキテクチャの特定を行った．そのうち，アーキテクチャが判明したものから 50 件ほど抽出し，動的解析やユーザマニュアルの分析によって，CSRF や DNS Rebinding 攻撃への脆弱性の有無，Web インタフェースのログイン情報，Web インタフェースのパスワード変更方法を調査した．また，過去に収集した 45 件のファームウェアに対しても，同様の調査を行う．

4.2　パスワードファイルの調査

展開されたファームウェアのうち，ルートファイルシステムが得られたものに関して，/etc/passwd や/etc/shadow ファイルにどのような root のパスワードが含まれるか調査した．/etc/passwd や/etc/shadow は Linux のログイン情報を管理しているファイルであり，パスワードのハッシュ値が保存されている．得られたハッシュ値はパスワードハッシュ解析ツール John the Ripper[15] を利用し，解析結果のパスワードがマルウェア Mirai の利用するパスワードリストに含まれるか調査した．

4.3　動的解析による Web インタフェースの調査

■動的解析　展開されたファームウェアのうち，アーキテクチャが判明したものを 50 件ほど抽出し，動的解析を行った．ここでは，ファームウェアに含まれる Web インタフェースに CSRF

や DNS Rebinding 攻撃への脆弱性が含まれているか調査した．

■CSRF　CSRF は，Web アプリケーションに対する代表的な攻撃手法のひとつである．CSRF の原理について，図 4.1 に示す．ここでは，攻撃者が用意した悪性の Web ページにユーザが訪れるという状況を想定する．攻撃者はこの場合，攻撃対象の他のサイトに対して POST リクエストを飛ばすような JavaScript を実行させることができる．それが IoT 機器のもつ Web アプリケーションに対して行われた場合，機器の設定を変更することが可能となる．本研究では，POST を送信する JavaScript が実行されるページを用意し，そこにアクセスすることで，実際に機器の情報を外から変更できてしまうかを調べた．CSRF に対して対策が施されている場合，機器の情報を変更することはできない．

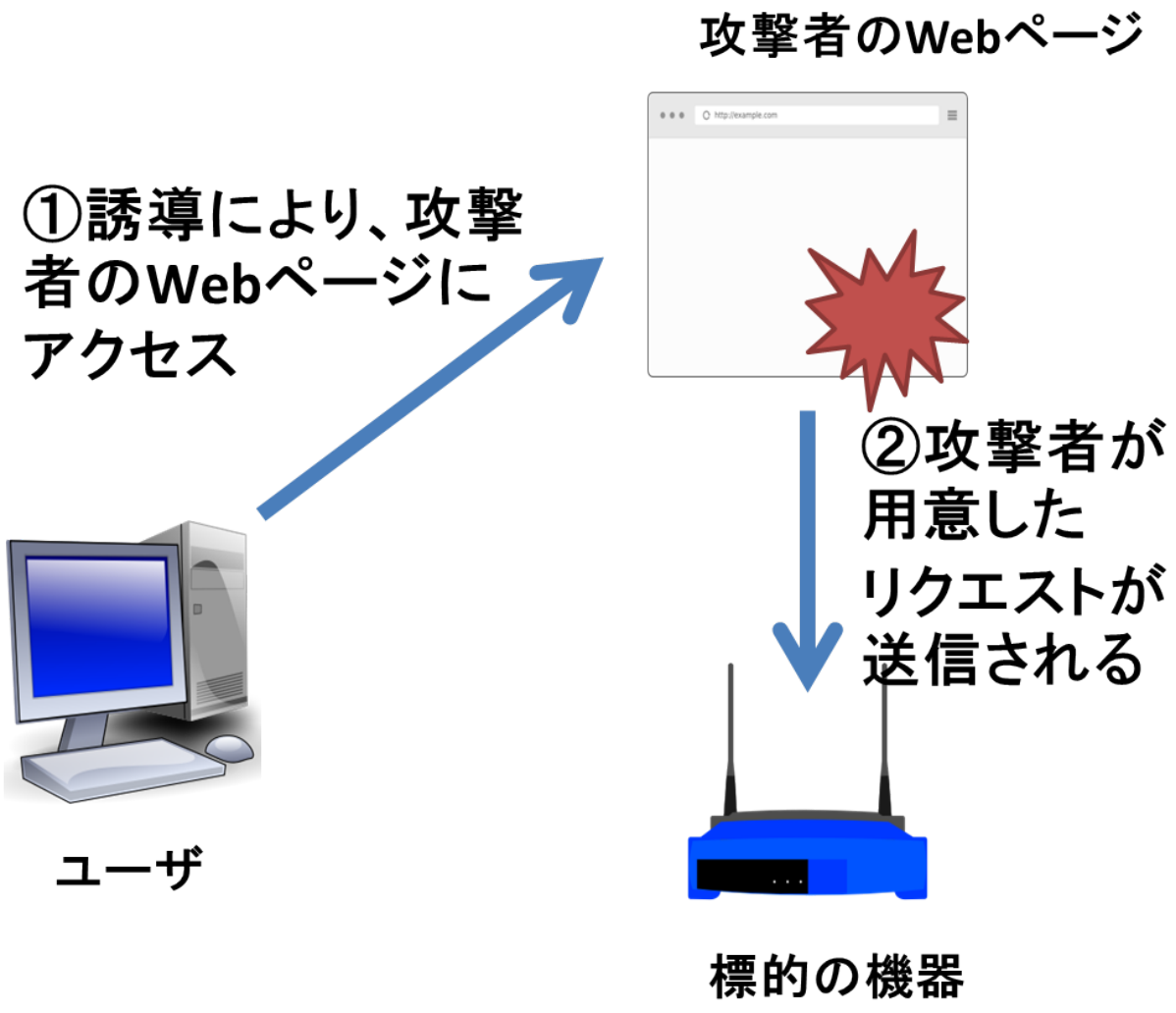

図 4.1　CSRF

■DNS Rebinding 攻撃　DNS Rebinding 攻撃は Same Origin Policy とよばれるセキュリティ機構を破るための手法のひとつである．Same Origin Policy はあるドメインのコンテンツから

別のドメインの情報にアクセスできないように Web アプリケーションに施されているセキュリティ機構である．これが破られた場合，攻撃者が用意した悪性 Web ページ上の JavaScript から攻撃対象の機器の情報を読み取ることが可能となる．DNS Rebinding 攻撃の原理について，図 4.2 に示す．DNS Rebinding 攻撃では，DNS の情報をたくみに変更することで，攻撃者のサイトと攻撃対象のサイトのドメインが同一のものとなるようにし，Same Origin Policy を突破する．ここでは，hosts ファイルに機器の IP アドレスとあるドメイン名の対応関係を記述をした場合に，そのドメイン名で通常どおりアクセスできるかを調べた．DNS Rebinding 攻撃に対して対策が施されている場合，異なるドメイン名では正常にアクセスすることができない．

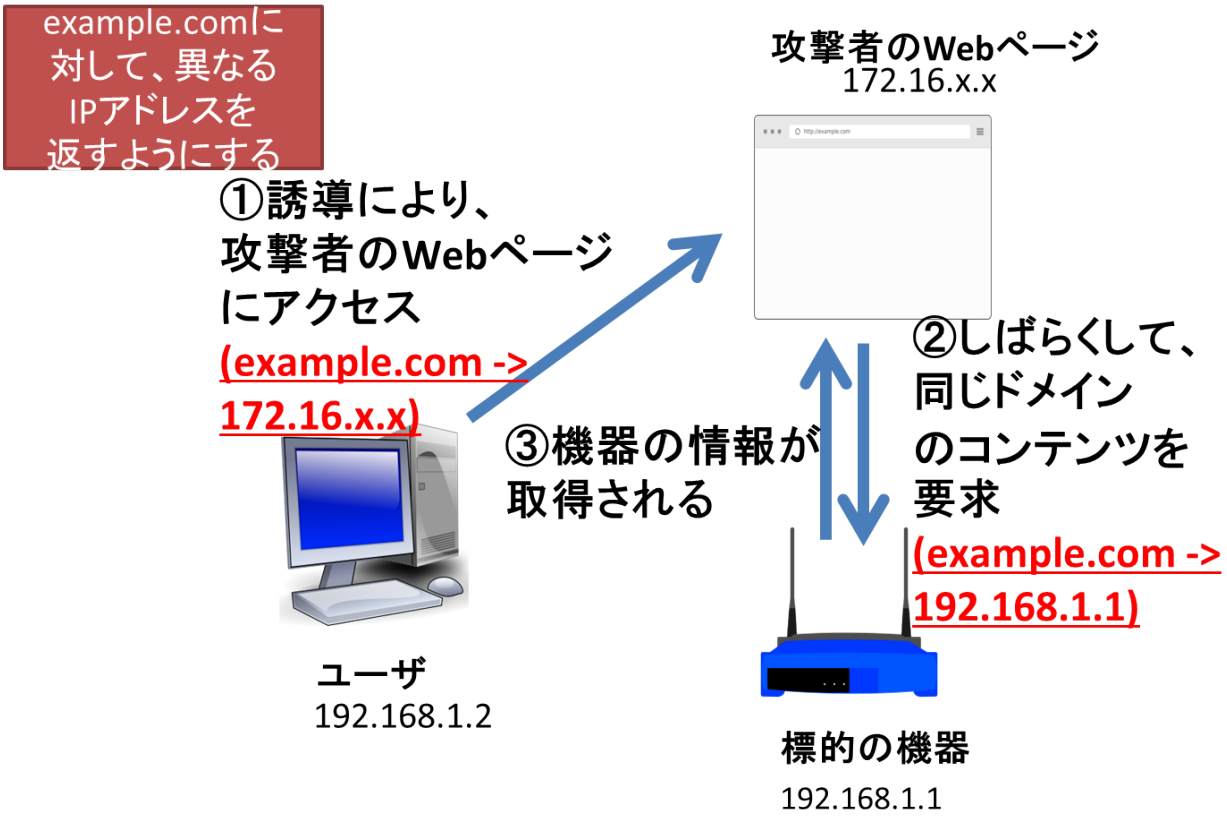

図 4.2　DNS Rebinding 攻撃

4.4　Web インタフェースのログイン情報の調査

動的解析を行った 50 件に関しては，同時にベンダが提供しているユーザマニュアルも収集し，どのようなデフォルトログイン名とパスワードが使用されているのか調べた．そして，得られた情報がマルウェア Mirai の持つログイン名とパスワードのリストに含まれるか調査した．

4.5 Web インタフェースのパスワード変更手順の調査

Web インタフェースのログイン情報を調査した 50 件について，同時にパスワードの変更手順についても調査した．通常，Web インタフェースにはパスワードを変更する機能が搭載されているが，それがどのタイミングで行えるかを，ユーザマニュアルの分析によって調べた．ここでは，機器固有のパスワードを使用しているものを一番強度の高いものとし，初回設定時に変更画面が現れるもの，後から任意のタイミングで変更が行えるもの，変更できないものという順で強度が弱くなっていくものとして評価を行う．

第 5 章　結果

本章では，第 4 章の実験で得られた結果について述べる

5.1　統計

ダウンロードしたファイルおよび展開したファイルに関しての統計を表 5.1 に示す．アーキテクチャが判明したものに関しては，表 5.2 のような内訳になった．なお，収集時期は 2018 年 12 月 18 日から 31 日までである．

表 5.1　収集および展開したファームウェアに関する統計

種類	数 (展開を行ったファイルに対する割合)
ダウンロードしたファイル	10,541
展開を行ったファイル	7,458
展開に成功したファイル	1,936 (26.0%)
アーキテクチャが判明したファイル	1,234 (16.5%)

表 5.2　ファームウェアのアーキテクチャ

アーキテクチャ	数 (アーキテクチャが判明したファイルに対する割合)
arm(little endian)	435 (35.3%)
mips(little endian)	406 (32.9%)
mips(big endian)	393 (31.8%)

5.2　パスワードファイルの調査結果

/etc/passwd と/etc/shadow から root パスワードを調べた結果，表 5.3 の通りになった．不明となっているものに関しては，大半がパスワードファイル自体が存在しないものである．これについては第 6 章で詳しく議論する．不使用となっているものはアカウントにロックがかけられているなど，パスワードファイル自体を使用していないと思われるものである．ハッシュ値

に関しては，ユニークなものが 17 種類得られた．そのうち，マルウェア Mirai のリストに含まれているものはハッシュ値としては 9 種類，パスワードとしては 5 種類あった．

表 5.3　パスワードファイルの調査結果

パスワード	数 (割合)
ハッシュ値が得られたもの	380 (19.6%)
不使用	89 (4.6%)
NULL パスワード	82 (4.2%)
不明	1,385 (71.5%)

5.3　Web インタフェースのログイン情報

収集したファームウェアのうち，アーキテクチャが判明したものから 50 件を抽出し，Web インタフェースのログインユーザ名を調べた．その結果，5 件は不明であり，判明したものに関しては表 5.4 のようになった．パスワードに関しては，表 5.5 の通りになった．未設定となっているものは，初期設定時にユーザにパスワード変更を強制する機器を示す．また，2017 年 2 月 1 日から 28 日の間に収集した 49 件に関しても調査を行った．その結果，4 件が不明で，残りのログイン名とパスワードはそれぞれ表 5.6，表 5.7 のようになった．パスワードがその他のものは，ベンダ名をそのままパスワードにしていた．機器固有のパスワードは，パスワードが記載されたシールが本体に貼ってある機器を示す．なお，集計はすべてのアカウントを合算して示している．それぞれの調査で得られたログイン名，パスワードは機器固有のものやベンダ名をもとにしたものが 3 件あったほかはすべて，マルウェア Mirai のログイン名，パスワードのリストに含まれていた．

表 5.4　Web インタフェースのログイン名 (2018 年 12 月)

ログイン名	数 (割合)
admin	24 (48.0%)
guest	5 (10.0%)
user	4 (8.0%)
NULL(なし)	1 (2.0%)
未設定	16 (32.0%)

表 5.5　Web インタフェースのパスワード (2018 年 12 月)

パスワード	数 (割合)
admin	15 (30.0%)
NULL(なし)	6 (12.0%)
guest	5 (10.0%)
password	4 (8.0%)
user	4 (8.0%)
未設定	16 (32.0%)

表 5.6　Web インタフェースのログイン名 (2017 年 2 月)

ログイン名	数 (割合)
admin	38 (65.5%)
guest	7 (12.1%)
user	7 (12.1%)
NULL(なし)	4 (6.9%)
root	2 (3.4%)

表 5.7　Web インタフェースのパスワード (2017 年 2 月)

パスワード	数 (割合)
admin	19 (32.8%)
NULL(なし)	12 (20.7%)
password	11 (19.0%)
user	5 (8.6%)
guest	5(8.6%)
1234	3 (5.2%)
機器固有のパスワード	1 (1.7%)
その他	2 (3.4%)

5.4　Web インタフェースのパスワード変更方法

ログイン名とパスワードが判明した 45 件に対して，パスワードの変更方法を調査した．その結果，表 5.8 のようになった．同様に，2017 年 2 月に収集したもののうち，パスワードが判明した 45 件に関しても調査を行った．その結果を表 5.9 に示す．どちらも、パスワードを変更できない機器は存在しなかった．初回設定時に変更画面が現れるものに関して，2017 年 2 月の調

査では，変更画面をスキップできる機器が多かったのに対して，2018 年 12 月の調査では，スキップできない機器のほうが多かった．

表 5.8　Web インタフェースのパスワード変更方法 (2018 年 12 月)

変更方法	数 (割合)
初回設定時に変更画面が現れる	27 (60.0%)
任意のタイミングで変更可能	18 (40.0%)
機器固有のパスワードで任意のタイミングで変更可能	0 (0%)
変更できない	0 (0%)

表 5.9　Web インタフェースのパスワード変更方法 (2017 年 2 月)

変更方法	数 (割合)
任意のタイミングで変更可能	34 (75.6%)
初回設定時に変更画面が現れる	10 (22.2%)
機器固有のパスワードで任意のタイミングで変更可能	1 (2.2%)
変更できない	0 (0%)

5.5　Web インタフェースの脆弱性

50 件のファームウェアのうち，動的解析によって Web インタフェースを調査できたものは 1 件だけであった．そのため，ほかに解析を行うことができるもの 8 件を追加し，計 9 件に対して調査を行った．解析の結果を表 5.10 に示す．また，2018 年 2 月に収集した 49 件に関しても動的解析を行った．このうち，解析ができたのは 6 件であった．その結果を表 5.11 に示す．

表 5.10　CSRF および DNS Rebinding 攻撃への対策状況 (2018 年 12 月)

対策状況	数 (割合)
CSRF への対策がされていない	4 (44.4%)
CSRF への対策がされている	5 (55.6%)
DNS Rebinding 攻撃への対策がされていない	9 (100%)
DNS Rebinding 攻撃への対策がされている	0 (0%)

表 5.11　CSRF および DNS Rebinding 攻撃への対策状況 (2017 年 2 月)

対策状況	数 (割合)
CSRF への対策がされていない	5(83.3%)
CSRF への対策がされている	1 (16.7%)
DNS Rebinding 攻撃への対策がされていない	6 (100%)
DNS Rebinding 攻撃への対策がされている	0 (0%)

第 6 章　議論

本章では, 得られた結果に関する考察および制約，今後の課題について述べる

6.1　考察

■パスワードの調査を通して　パスワードファイルの解析では，ファイル自体が存在しないものも多く，どういうパスワードが使われているのか不明なものが半数以上あった．こうした機器では，パスワードファイルへのシンボリックリンクは存在するが，その実体ファイルが存在しなかった．おそらく，起動時に実機側にあるファイルシステムをマウントしているものと思われる．以上のように, 解析対象が限定されたものとなり，全体としての傾向をつかむことはできなかった．しかし，中にはハッシュ値を得られたものもあり，そのいくつかは元のパスワードが判明した．こうしたパスワードはマルウェア Mirai のパスワードリストにも含まれているものであった．ここから，少なくともいくつかの機器に関しては，単純な root パスワードが用いられていることが判明した．Web インタフェースのデフォルトのログイン名とパスワードも同様に Mirai のリストに含まれているような単純なものであった．そのため，デフォルトのままにしておくことは，よく使われるパスワードを試す攻撃に対して脆弱であるといえる．パスワードの変更方法の調査結果をみると，2017 年 2 月の結果と比べて，初期設定時にパスワードを強制的に変更させるような機能のある機器が多くなっていることが分かった．2017 年 2 月の時点でも，初期設定時にパスワード変更を促す機器はあったが，ほとんどはスキップできる仕様となっていた．以上のことから，パスワードリスト攻撃が問題となっている現状に対するベンダの対策が進んでいることが見て取れる．しかし，調査した範囲では 4 割がそうした対策がされておらず，こうした機器はパスワードがデフォルトのまま運用される可能性が高い．

■攻撃の可能性　機器のパスワードがデフォルトのままで運用されており，かつ外部からのアクセスが可能である場合，容易に攻撃されてしまう．ネットワークカメラなどはこうした運用方法をなされる場合が多いので，攻撃されやすいといえる．実際、マルウェア Mirai でもこうしたネットワークカメラは標的になっている．しかし，外部から直接アクセスできないような機器でも CSRF や DNS Rebinding 攻撃を通して，外部からの攻撃が可能となる．調査の結果，CSRF に関しては，対策されている機器は増えていたが，解析ができた数が少ないので全体の傾向としては一概にはいえない．ひとついえることとしては，CSRF によって設定を変更できて

しまう機器がまだまだ存在するということだ．DNS Rebinding 攻撃に対しては，2017 年 2 月，2018 年 12 月のどちらの時点でも対策されていた機器は見つからなかった．以上のことから，こうしたリスクに対する認識がまだまだ不十分であると考えられる．実際，CSRF を行うことができなかった機器についても，それが CSRF を意識した対策によるものかは疑問が残る．

■考えられる対策　パスワードの変更方法を調査した結果をみると，初期設定時にパスワードの変更を促す機器が増えてきている．今後，こうした仕様が一般的になると，デフォルトのパスワードのまま運用される危険性も減少する．ほかにも，今回の調査の範囲では 1 件だけ確認できたが，機器固有のパスワードを利用するという方法もある．この方法では，よく使われるパスワードを狙った攻撃には有効であると考えられる．しかし，固有のパスワードを生成するアルゴリズムを推測しやすいものにしないといった注意も必要である．

■ファームウェア解析の評価　本研究では，ファームウェアの解析を行うことで，実機が手元にない状態であっても Web インタフェースの脆弱性を発見することができた．それにより，IoT 機器に対するセキュリティ対策がおろそかな点は見出せた．この点では，有用であると考える．本研究のように，IoT 機器全体の傾向を把握するという観点では，大規模な調査が必要となり，この点でファームウェア解析は有用である．しかし，パスワードファイルの解析を行った際に，パスワードファイル自体が存在しないものも多く，うまく解析できなかった．また，動的解析を行った際には，実際に解析が行えた機器はわずかであった．このように，様々な制約が存在することも分かった．これに関しては，次の制約の項目で論じる．以上からいえることとしては，ファームウェア解析は IoT 機器を解析するひとつの選択肢として有用だが，得たい情報によっては実機解析などと組み合わせるなどをする必要がある．また，静的解析をするのか，動的解析をするのかによっても得られる情報が大きく異なる．今回の実験では，動的解析によって Web インタフェースの脆弱性を調べたが，静的解析でも得られる可能性はある．しかし，動的解析ではよりその動作を簡便にとらえることができる．このように，ファームウェアを解析するにしても，調べるレベルにより適切な解析方法を選択する必要がある．また，ユーザマニュアルを調べたように，ファームウェア解析を単体で行うよりも，得られる別のメタデータを組み合わせることも重要だと考える．

6.2　制約

本研究で行ったファームウェア解析に関して，様々な制約があった．

■ファームウェアの入手性　まず，ファームウェアの入手性が挙げられる．本研究では COTS(Commercial Off-The-Shelf) 製品を対象にしたが，その理由のひとつは，ファームウェアの入手が容易であったからである．しかし，専門的な領域で用いられる機器のファームウェアなどはベンダが一般公開していない場合も多く，入手するには実機から取り出すといった操作が必要になる．大量にファームウェアを収集し，自動的に分析を行うという手法においては，

この入手性の問題が大きな壁となる．対策としては，ベンダと協力して研究をおこなうなどしてファームウェアを入手することも必要となるだろう．

■独自の圧縮フォーマット　次に，研究手法の章でも述べたが，機器に施されている圧縮フォーマットがベンダ固有のものである場合，展開が難しいという問題がある．こうした場合，既製の展開ツールではうまく展開を行うことができず，解析を行うことができない．こうした場合も，ベンダと協力して研究をするなどの方法が必要となる．

■実機が必要な場合　また，ファームウェアの解析では得られない情報もある．たとえば，機器の ROM に書き込まれた情報などは，実機を入手しないと得ることができない．ファームウェアの動的解析では，あくまで仮想的な環境で機器の動作をエミュレートしているだけなので，機器の物理的な振る舞いを調べる際などには実機での解析が必要となってくる．また，ファームウェア解析で得られた脆弱性が実際に機器にどの程度影響を与えるかを評価するには，実機での検証が必要となる．

■評価の限界　以上のような問題から，ファームウェアを収集・展開できた一部の機器のみが評価対象となるという点で注意が必要だと考える．ファームウェアを収集できなかったり，展開できなかったものに関しては，ファームウェア解析という手法では評価できないため，実機を入手して解析するなど別の方法で分析する必要がある．

6.3　今後の課題

本研究を通して得られた，今後の課題について述べる．

■解析の自動化　まず，課題のひとつとして，解析の自動化を進めることが挙げられる．ファームウェアの収集や展開，パスワードファイルの分析においてはある程度の自動化ができた．しかし，ファームウェアを収集する際に，ベンダの Web サイトの構造を分析する必要があり，ここでは手作業が要求された．また，動的解析やより詳細なコード分析のレベルにまで入ると，かなり手動での分析が必要となる．これがある程度自動で分析できるようになれば，ファームウェア分析をより簡便に行えるようになる．

■解析の大規模化　次に，より大規模な解析が挙げられる．今回はリソースの制約などから 1,000 件近くのファームウェアの分析にとどまったが，規模を拡大することで，こうした領域のセキュリティをより大きな視点で見ることができるだろう．特に、動的解析できた数が少なかったので，この点の改良が必要である．前述した自動化が進めば，本研究で用いた手法の大規模化が可能であると考えている．

■解析対象の多様化　そして，解析対象をより多様な機器に広めることもひとつの課題である．本研究では，身近な機器として COTS(Commercial Off-The-Shelf) 機器を対象にした．しかし，

COTS 機器は IoT 機器のごく一部にすぎない．IoT は，多種多様な機器がインターネットにつながることでより，その柔軟性を発揮すると考えている．本研究で主眼とした，パスワードがデフォルトのままの機器が狙われるといった問題はほかに種類の機器にも存在するであろう．また，違った種類の機器との比較を通して，また別の問題も浮かび上がってくることが期待される．本研究で得られた知見を別の機器の分析にも応用することができればいいと考えている．前述したように入手性の問題などもあるが，今後 IoT 機器のセキュリティを考えていく上では，多種多様な領域の機器を分析する必要があると考える．

6.4 倫理的な議論

本研究では，各ベンダのサイトからファームウェアを収集し，その解析を行った．こうした行為が法律でどう扱われているかをここに述べておく．平成 30 年 5 月 18 日に成立し，平成 31 年 1 月 1 日に施行された「著作権法の一部を改正する法律」[16] では，「当該著作物に表現された思想又は感情を自ら享受し又は他人に享受させることを目的としない場合」における著作物の利用を認めており，その中には，サイバーセキュリティ確保を目的としたソフトウェアの調査解析も含まれるとしている．実際の問題として，こうした解析行為は，IoT 機器に対するセキュリティリスクを正しく評価するためには必要な行為であると考えている．本研究を通しても，IoT 機器のセキュリティの不十分さが明らかになった．また，解析行為によって脆弱性を発見した場合は，ベンダに報告するなどしてセキュリティの向上に役立てるだろう．本研究中で見つかった CSRF や DNS Rebinding への対策の不備は，具体的な脅威に結びつくかは判断を待つところもあり，報告は行っていない．しかし，今後より重大なセキュリティリスクが見つかった場合は報告する義務があると考えている．

第 7 章　まとめ

本章では，本研究のまとめを述べる

本研究では，IoT 機器 (特に COTS 製品) を狙った，パスワードリスト攻撃に対するリスク評価を行った．そこでは，デフォルトパスワードが単純なもので，なおかつ初期設定時に変更をさせるような機能がない機器が多く，こうした攻撃に対して脆弱である現状が判明した．また，外部に直接接続されていない機器に対しても攻撃が可能な CSRF や DNS Rebinding 攻撃に対する対策も不十分であることが分かった．しかし，初期設定時にパスワードを強制的に変更させるような機器も増えてきており，パスワードリスト攻撃に対するベンダの対策が進んでいることも分かった．今回，大規模な傾向をつかむため，手法としてファームウェア解析を選んだ．そこでは，パスワードファイルの解析や脆弱性の調査を通して，ファームウェア解析が有用であることが示された．しかし，入手できないファームウェアがあり，展開が行えないものも多いことが分かった．また，動的解析がうまく行えないものも多く，ファームウェア解析には制約が多いことも分かった．今後は，自動化を進めることで，より大規模に解析を行うことを課題としている．また，COTS 製品以外にも目をむけ，新たなリスクを探って行きたいと考えている．

謝辞

本研究を進めるにあたって，熱心にご指導いただいた森達哉教授には大変感謝いたします．また，研究に対するアドバイスや有益な指摘の数々をくださった森研究室の皆様にも感謝いたします．

参考文献

[1] Iot デバイス数は 2017 年、世界人口を上回る？ –ガートナー - zdnet japan. https://japan.zdnet.com/article/35096854/.

[2] Shodan. https://www.shodan.io/.

[3] Insecam - world biggest online cameras directory. https://www.insecam.org/.

[4] >github - jgamblin/mirai-source-code: Leaked mirai source code for research/ioc development purposes. https://github.com/jgamblin/Mirai-Source-Code.

[5] Yin Minn Pa Pa. Iotpot: Analysing the rise of iot compromises. 9th USENIX Workshop on Offensive Technologies(WOOT), 2015.

[6] Ian Foster. Fast and vulnerable: a story of telematic failures. 9th USENIX Workshop on Offensive Technologies, 2015.

[7] Daming D. Chen. Towards automated dynamic analysis for linux-based embedded firmware. Network and Distributed Systems Security Symposium(NDSS), 2016.

[8] Andrei Costin. A large-scale analysis of the security of embedded firmwares. 23nd USENIX Security Symposium, 2014.

[9] Andrei Costin. Automated dynamic firmware analysis at scale: A case study on embedded web interfaces. 11th ACM Asia Conference on Computer and Communications Security(ASIACCS), 2016.

[10] Drew Davidson. Fie on firmware: Finding vulnerabilities in embedded systems using symbolic execution. 22nd USENIX Security Symposium, 2013.

[11] Selenium - web browser automation. https://www.seleniumhq.org/.

[12] Github - firmadyne/firmadyne: System for emulation and dynamic analysis of linux-based firmware. https://github.com/firmadyne/firmadyne.

[13] Github - refirmlabs/binwalk: Firmware analysis tool. https://github.com/ReFirmLabs/binwalk.

[14] Qemu. https://www.qemu.org/.

[15] John the ripper password cracker. https://www.openwall.com/john/.

[16] 著作権法の一部を改正する法律（平成 30 年法律第 30 号）について | 文化庁. http://www.bunka.go.jp/seisaku/chosakuken/hokaisei/h30$_h okaisei/$.