IPA

2024 年度 未踏 IT 人材発掘・育成事業 採択案件評価書

1. 担当 PM

田中 邦裕（さくらインターネット株式会社 代表取締役社長）

2. クリエータ氏名

九鬼 琉（横浜国立大学 理工学部 数物・電子情報系学科／株式会社オプシス）

3. 委託金支払額

2,880,000 円

4. テーマ名

ゼロデイ攻撃の対象となる IoT 機器を早期特定するシステムの開発

5. 関連 Web サイト

なし

6. テーマ概要

本プロジェクトでは、事前に IoT 機器のファームウェアを大規模に収集・解析し、ゼロデイ攻撃の観測時にファームウェアの解析結果と攻撃情報を照合することにより、観測されたゼロデイ攻撃のターゲットとなっている機器を早期に特定するシステムを開発した。本プロジェクトが想定するシナリオに基づいたシステムの概要を図 1 に示す。

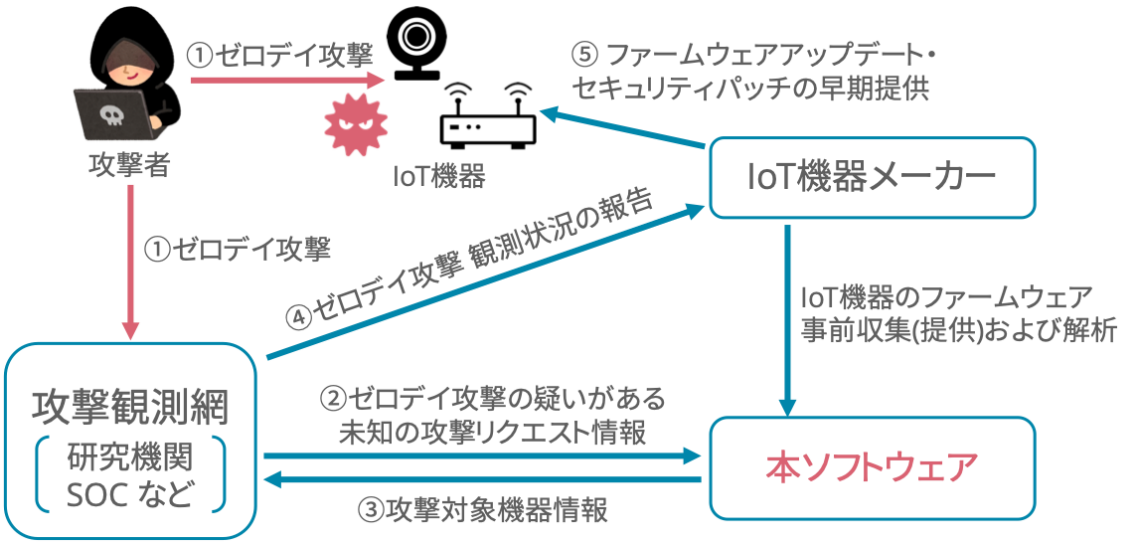

図 1：システム概要

まず、攻撃者がインターネットに直接接続されている IoT 機器に存在する脆弱性を悪用し、ゼロデイ攻撃を行う。このとき、研究機関や企業の SOC 部門などが運用する攻撃観測網においても当該ゼロデイ攻撃が観測される（①）。しかし、従来の攻撃分析手法ではこの時点で、当該ゼロデイ攻撃がどのような機器やサービスを標的としているかを特定することが困難であり、脆弱性の修正や注意喚起などの対策に直結させることができなかった。そこで本プロジェクトでは、観測されたゼロデイ攻撃の疑いがある未知の攻撃リクエスト情報（②）をもとに、事前に大規模に収集した IoT 機器のファームウェアの解析結果と照合することにより、攻撃の標的となっている可能性のある機器を特定し、その情報を出力する（③）。この情報を用いることで当該 IoT 機器のベンダーや開発者に対して、ゼロデイ脆弱性の存在とその悪用状況を報告することができ（④）、脆弱性を修正するためのファームウェアアップデートやセキュリティパッチの早期提供に貢献することができる（⑤）。

7. 採択理由

本プロジェクトは、インターネットに接続された Wi-Fi ルータやウェブカメラなどの IoT 機器への攻撃を観測し、ゼロデイ攻撃とそのターゲットを早期に発見し、脆弱性の修正を速やかに行うことを目的としている。いままでにも同様のアプローチはなされていたが、本提案においてはファームウェアの収集範囲を大幅に拡大させるとともに、ファームウェアの動的解析と静的解析を組み合わせて、よりゼロデイ攻撃に対する対応速度と精度を向上させる取り組みをしている部分が革新的である。

提案者はすでに大学において同じ内容の研究を進めており、実現可能性は高い。その上で未踏プロジェクトとして進めることによって、より産業界のニー

ズをとらえながら、実効性のあるアプローチを取ることができる。加えて、IPA や NISC などの政府機関とも連携しながらプロジェクトを進めることで、公共の安全とセキュリティの強化にも繋がり、プロジェクトの社会的影響力が増すと共に、セキュリティ研究の重要性が国内外で認識される機会が増えると考え、採択した。

8. 開発目標

本プロジェクトの開発目標は、IoT 機器を標的としたゼロデイ攻撃に対して、脆弱性が公表される前の段階でその対象機器を特定し、迅速な対処を可能とする解析・照合システムを構築することである。従来、観測された未知の攻撃リクエストがどの機器を狙ったものかを特定することは極めて困難であり、結果として脆弱性の修正や注意喚起に時間を要していた。

本プロジェクトではこの課題を解決するため、IoT 機器のファームウェアを事前に大規模に収集・解析し、その静的・動的な解析結果をデータベース化したうえで、サイバー攻撃観測網（例：ハニーポット）で取得された攻撃リクエストの内容と照合を行うシステムを開発することとした。この照合作業により、攻撃パスや特徴的な文字列を含むファイル構造との一致を検出し、対象機器のベンダー名・型番・ファームウェアバージョンなどの詳細情報を迅速に提示できる。

さらに、攻撃が影響を及ぼすと考えられる実装箇所のファイルパスやソースコードを特定したり、仮想環境で当該機器のファームウェアを起動して再現実験したりすることが可能な機能も備え、開発者による迅速な脆弱性修正と、被害拡大防止のための判断材料を提供することとした。このような一連の機能により、本システムがサイバー攻撃の初動対応力を飛躍的に高め、IoT セキュリティの強化に資する社会的意義の高い基盤となることを目標とした。

9. 進捗概要

本プロジェクトでは、観測されたゼロデイ攻撃の攻撃リクエスト情報と、事前に収集・解析した IoT 機器のファームウェアとの照合によって攻撃対象機器を特定するシステムの開発にあたり、4 つのコンポーネントからなるソフトウェアの実装を行った。

まずは、ファームウェアの収集である。IoT 機器ベンダーのホームページなどで公開されているファームウェアを、スクレイピングによって自動的に収集する処理を構築し、機器の型番やバージョン、公開日、収集元 URL、ファイルハッシュなどのメタ情報とともに保存する仕組みを整備した。ベンダーの Web サーバに過度な負荷をかけないように連続リクエストの間隔を空けるなどの配慮を行いつつ、既に収集済のベンダーに対しても定期的にアクセスを実施することで、新しい機器の発売やバージョン更新があった場合には追加収集を行うようにした。また、各ベンダーのホームページに対するスクレイピングコードを効率

よく実装・運用するため、LLM（Large Language Model）を活用してコードを自動生成する機能を構築した。具体的には、ファームウェアが掲載されているページの URL を入力すると HTML を取得し、スクレイピング要件とともにプロンプトとして LLM に入力し、出力された Python スクリプトを保存することでコードを自動生成する仕組みを構築した。一部のページでは生成されたコードが正常に動作しない場合もあるため、人手による動作確認と必要に応じた修正を行う工程も併せて設けたが、大部分が自動化されたため、すべてを人手で実装する場合と比較して実装の所要時間を大幅に短縮することができている。このような LLM 活用によるスクレイピングコードの生成は、プロジェクトの「試行錯誤の自動化」の好例であり、再利用性のある仕組みとして他用途にも展開可能と見込まれる。

次に、収集したファームウェアに対して静的解析を行う処理を実装した。ファームウェアを展開（アンパック）して内部に含まれるファイルやバイナリなどを抽出する処理には、unblob というオープンソースツールを採用した。unblob は、多数の圧縮形式やファイルシステムに対応しており、フォーマット標準に基づいたルールにより高速にデータチャンクやバイナリパターンを検出してファイル構造を復元できる点から、binwalk などの既存の他のツールと比較して高い展開成功率と処理速度を実現できることを確認の上で選定した。展開成功率とパフォーマンスを検証した上でのツール選定は、解析全体の信頼性・スケーラビリティを左右する重要な判断であり、実験と定量評価に基づいた妥当な選定であると評価している。展開されたファイルおよびディレクトリ構造は、収集時に得られたファームウェアのメタ情報と紐づけて MariaDB 上に記録されるよう設計した。記録先のテーブルには、ファイルパスの逆順文字列を格納するカラムを Generated Columns 機能により定義し、これに対してインデックスを貼ることで、攻撃リクエストに含まれるパスとの後方一致照合を効率よく行えるようにした。また、ファイルパスが更新された場合にも自動的に逆順カラムに反映されるように設定した。これら後方一致に特化した DB 設計は、攻撃リクエストの性質を踏まえた現実的かつ技術的に優れた工夫であり、実運用の可視性・検索性能の観点でも今後の応用余地がある。

さらに、動的解析部においては、仮想環境上でファームウェアを起動して動作を観測する処理を構築した。このために FirmAE という IoT ファームウェアの動的解析に特化した解析ツールを導入し、これに対して複数の拡張機能を実装した。拡張機能には、複数の仮想環境を同時起動して並列にファームウェアを解析できる機能、起動成功時に ping 応答やネットワークサービスの稼働状況、各サービスのレスポンスやフィンガープリントを自動的に収集・記録する機能、必要な依存関係やカーネルモジュールが不足している場合にそれらを自動補完またはスタブモジュールで代替する機能、初回起動時の初期設定やアカウント作成を自動化する機能などが含まれる。ファームウェアが仮想ネットワーク上で起

動した際には、その IP アドレスを取得し、Nmap を用いてポートスキャンを実施することで、サービスの起動状況やバージョン情報を収集する処理を追加した。Nmap のスキャンでは以下のようなオプションを指定している。

```
nmap -sV -Pn -p- [host_ip]
```

-sV オプションはサービス検出を有効にし、各開放ポートに様々なペイロードを送信することにより、各ポートで動作しているアプリケーション層のサービス名やそのバージョンを推測する。-Pn オプションは Nmap が通常のスキャンを開始する前にホストの到達性を確認するために行う ping スキャンを無効化するオプションである。Nmap のデフォルトの挙動では、事前の ping スキャンに応答がない場合、ホストの存在が確認できないと判定し、ポートスキャンを行わずに処理を終了する設計となっている。しかし、IoT 機器の中には ping に応答しない機器も多く、そのような機器に対しても必ずポートスキャンを実行するようにするために指定している。そして-p-オプションを指定することにより、1～65535 番までの全てのポートをスキャン対象としている。

ネットワークサービスの起動状況の収集後、続いて各サービスのレスポンスやフィンガープリントを自動的に収集・記録するようにした。具体的には、HTTP サービスの場合、トップページを起点として外部リンクを除いた内部リンク先へ再帰的にアクセスし、各ページのレスポンスとリクエストパス情報を記録するようにした。

次に、攻撃照合部では、ハニーポットなどの観測網で観測された攻撃リクエストを入力として受け取り、そのリクエストの宛先ポート番号やリクエストパス、パラメータ名、リクエストヘッダ内の Cookie や認証情報などを抽出し、それを基にファームウェアの静的解析および動的解析の結果と照合する処理を実装した。照合の第一段階では、ファームウェア中に存在するファイルパスが攻撃リクエストパスと後方一致するものをリストアップし、対象機器の候補として出力する処理を行う。出力にはベンダー名、製品名、ファームウェアバージョン、公開日、照合に一致したファイルパスが含まれる。照合の第二段階では、静的解析で抽出されたファイルの可読文字列と攻撃リクエストから抽出された特徴的な文字列の一致確認、動的解析で得られたネットワークサービスのポート番号やレスポンスの照合を行う。これらにより、対象機器候補の絞り込みを行うとともに、攻撃対象である可能性のある脆弱な実装箇所として、ファイルパスや該当周辺の実装内容を出力することを可能にした。

また、検証のために、照合により対象機器候補としてリストアップされたファームウェアを仮想環境上で起動し、起動後の機器にコマンドラインやブラウザ経由で接続できる環境を提供する機能を実装した。これにより、観測された攻撃リクエストが当該ファームウェアに対して有効かどうか、つまり脆弱性が再現

可能であるかを詳細に調査するための環境を整えた。以上の実装を通じて、ファームウェア収集・静的解析・動的解析・攻撃照合・仮想環境による再現の一連の処理が可能となるソフトウェア基盤を構築した。

各構成要素の完成度は高く、個々の技術選定や自動化の工夫が行われており、将来的には外部のサイバー観測網との連携や、脆弱性対応支援ツールとしての発展も視野に入れられる基盤となる成果となった。

10. プロジェクト評価

本プロジェクトは、IoT 機器を対象としたゼロデイ攻撃の分析・特定に向けた解析基盤を、実運用に耐える水準でまとめ上げた点で高く評価できる。各構成要素において、目的に即した技術選定と設計の工夫が随所に見られ、処理の精度と効率の両立が意識されていた。特に収集の自動化では、LLM によるコード生成を導入することで、多様なサイト構造への対応力と保守性が確保されており、実装のスケーラビリティに対する視野が感じられた。

照合処理では、複数の解析結果を活用し、単純な一致ではなく意味的な関連性を追う設計が実装されている点が印象的である。さらに、得られた結果を再現環境で検証できる仕組みまで一体的に構築されており、分析と実証の橋渡しを明確に意識したアーキテクチャといえる。ツール開発にとどまらず、観測・照合・再現というサイバー攻撃対応の一連の流れを自動化した点に、本プロジェクトの独自性と実践的価値が集約されている。

また、技術的な完成度に加え、既に国内の観測機関と連携して実際に 4 件のゼロデイ攻撃対象機器候補を特定・報告しており、成果の実証性と社会的波及性も高い。今後の拡張可能性についても、他プロトコルへの適用や外部連携 API の整備、非公開ファームウェアへの対応に向けたベンダー交渉の構想まで視野に入っており、継続発展に向けた道筋が明確に描かれている。

開発体制についても、実質的に一人で収集・解析・連携・検証までを担っており、単なる研究開発にとどまらない高い遂行力が発揮されている。こうした実行力と自律性は、未踏プロジェクトにおける重要な価値のひとつといえる。総じて、本プロジェクトは、現在の成果のみならず今後の展開可能性も含めて、極めて高い水準に達していると判断する。

11. 今後の課題

本プロジェクトは、IoT 機器のゼロデイ攻撃に対する照合と再現の実行可能性を実証し、技術的には一定の完成度を示している。他方で、今後の発展と実装段階への移行を見据えると、複数の側面で改良および体制整備が課題となる。

まず、ファームウェアのエミュレーション成功率は約 3 割にとどまり、暗号化や非対応アーキテクチャへの対応が制約となっている。静的解析との併用により一部は補完されているものの、解析対象をさらに広げるためには技術的な

改良が引き続き求められる。また、動的解析で取得可能な非 HTTP 系サービスの照合処理への統合は進行途上にあり、活用可能なデータを照合設計に効果的に還元するための再設計が課題である。

照合処理においては、現在主に文字列一致やパスの後方一致を用いた設計となっているが、今後はファイル構造や通信プロトコルの意味的な対応関係に基づいた分析手法の導入により、判定精度を高める余地がある。さらに、仮想環境による再現機能についても、操作性や導入手順の簡素化が望ましく、検証フェーズの実用性を高める UI/UX の改良は今後の検討事項である。

加えて、システムを持続的に運用・拡張していくための体制面の課題の解決が必要となる。現在は個人ベースでの開発・運用が中心となっているが、今後は外部からの支援や連携を制度的に取り込み可能な運営母体などを整備し、NICT や IPA などの公的機関や、機器ベンダー、通信事業者などからの資源投入に応じた受け皿として機能させる必要がある。これは資金的な観点のみならず、継続的に人材や技術が集まる仕組みとしても重要である。

ファームウェア収集においても、OTA によって公開されない製品が増加する中で、ベンダーとの信頼関係に基づく協力体制の構築が必須となる。とりわけ、国内ベンダーだけでなく海外ベンダーとの関係性強化については、技術的信頼性と公益性を丁寧に説明し、協調的な情報提供関係を築けるかどうかが今後のカバレッジ拡大の鍵を握る。

このように、技術水準は高い一方で、実用規模での定着には、技術的改良と体制構築の両面で段階的かつ戦略的な対応が求められる。持続性・拡張性・信頼性を備えた社会実装型の仕組みへと進化させていくためには、これらの課題に対する取り組みが期待される。