IoT 機器の脆弱性を考慮した IoT システムのアーキテクチャの設計

2014SE109 山口透子
指導教員：沢田篤史

1 はじめに

IoT の普及により，様々な機器がインターネットにつながるようになった．その一方で，家庭用カメラやテレビといった今までインターネットにつながることを想定されていなかった機器がネットワークに接続されるようになった．その中で，それら機器の脆弱性の顕在化が問題となっている．本研究の目的は，ホームネットワークシステムの脆弱性分析に基づいた IoT システムのアーキテクチャの設計である．研究指針として，IoT システムの参照アーキテクチャの各コンポーネントに対する脅威分析を行いセキュリティ機能を抽出する．セキュリティ機能を横断的関心事としてモジュール化し，IoT システムの参照アーキテクチャにおいて，セキュリティ機能を動的再構成するアーキテクチャを提案する．動的再構成を行うために，江坂らが提案した PBR (Policy Based Reconfiguration) パターン [1] を適用する．

2 技術背景

2.1 IoT (Internet of Things)

IoT とは様々なモノがインターネットに接続し，情報をリアルタイムで通信することで相互に制御する仕組みである [4]．利用者の目に見えないところで IoT 機器があらゆる脅威にさらされる可能性が高いので，十分なセキュリティ対策が必要である [4]．

2.2 コンテキスト指向

コンテキスト指向とはプログラミング実行時に外部環境の変化に依存する振る舞いをモジュール化する技術である [1]．コンテキスト指向とはプログラミング実行時に外部環境の変化に依存する振る舞いをモジュール化するためのプログラミング手法である．コンテキストとは，システムの環境や内部状態で，場所や状況に応じて変化する．変化する状況に応じてシステムの振る舞いを最適化することが可能となる．

2.3 PBR パターン

PBR パターンとは，江坂らが提案している，アスペクト指向とコンテキスト指向を統一的に扱うための自己適用のためのアーキテクチャパターン [1] である．PBR パターンを図 1 に示す． Policy は Component 間のメッセージ通信を横取りし， Context に応じて Configuration Builder へ再構成の命令を出す． Configuration Builder は Policy からのメッセージより New Component を生成する．New Component の生成後，New Component と Common Component からなる Updated Confiuration を生成し，再構成する．

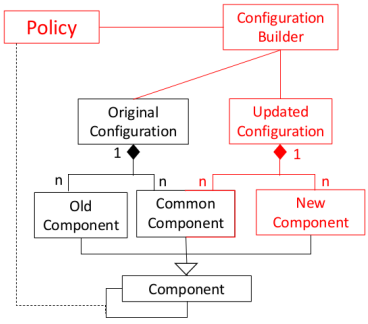

図 1 PBR パターン静的構造

3 IoT システムにおける脅威分析

本研究では，江坂が提案する IoT システムアーキテクチャ [2] に基づき，システムの構成要素であるデバイス，モバイル端末，クラウドの三つについて「IoT 開発におけるセキュリティ設計の手引き」[5] を参考にそれぞれを分析した．各コンポーネントの脅威に対し共通するセキュリティ機能をまとめたものを図 2 に示す．

モジュール/セキュリティ機能	FW機能	IDS･IPS	通信経路暗号化	ユーザー認証
クラウド	○	○	―	○
モバイル端末	―	―	―	○
デバイス	○	―	○	○
クラウド-モバイル端末間通信	―	―	○	○
クラウド-デバイス間通信	○	―	―	○
モバイル端末-デバイス間通信	―	―	○	○

モジュール/セキュリティ機能	データ暗号化	DoS対策	アンチウイルス
クラウド	○	○	―
モバイル端末	―	―	○
デバイス	○	―	○
クラウド-モバイル端末間通信	―	―	―
クラウド-デバイス間通信	―	○	―
モバイル端末-デバイス間通信	―	―	―

図 2 各コンポーネントの共通セキュリティ機能

4 脆弱性分析に基づく IoT システムのアーキテクチャ設計

本研究で提案するアーキテクチャを図 3 に示す．システムの環境変化をコンテキストとする．セキュリティ機能選択ポリシーがコンポーネント間のメッセージ通信を横取りし，コンテキストが変化したさいにセキュリティ機能選択活性機を活性化させ，選択したセキュリティ機能を付与する．

アーキテクチャの動的振る舞いを図 4 に示す．利用者によって行われる，デバイスの追加・変更，モバイル端末の追加・変更，さらに IoT システムのコンポーネント間の通信をシステム環境変化としてコンテキストとする．デ

バイスはデバイス情報を，モバイル端末はモバイル端末情報を，クラウドはクラウド情報を保持する．利用者によってデバイス，モバイル端末の変更・追加，フォグサービスの変更がおこなわれたさいに，情報が更新される (図 4 の 1)．セキュリティ機能選択ポリシーは更新されたシステム環境変化の更新メッセージを横取り (図 4 の 2) し，選択するセキュリティ機能インスタンスを生成するメッセージをセキュリティ機能付与 Behavior Activator に送る (図 4 の 4)．セキュリティ機能付与 Behavior Activator は付与するセキュリティ機能を生成 (図 4 の 5) し，セキュリティ機能を付与された IoT システムを再構成 (図 4 の 4) する．

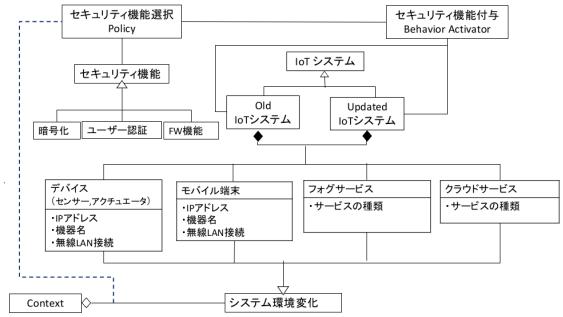

図 3 脆弱性を考慮した IoT システムのアーキテクチャ (静的構造)

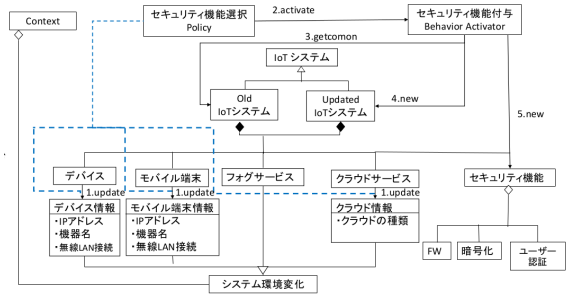

図 4 脆弱性を考慮した IoT システムのアーキテクチャ (動的振る舞い)

5 考察

本研究の関連研究に佐野の脆弱性分析に基づく IoT システムのアスペクト指向アーキテクチャ [3] が挙げられる．佐野の研究では，アスペクト指向に基づきアーキテクチャの静的再構成を行っている．本研究では，システムの環境変化をコンテキストとしたシステムの動的再構成を行った．システムの環境は利用者によってモバイル端末の変更や追加，デバイスの変更・追加が挙げられる．IoT システムのコンポーネント間で新たな接続が見られた場合も環境変化として挙げられる．そのため，実装するさいに，コンテキストの取得やセキュリティ機能を付与するさいの詳細な定義が必要となる．またコンテキストの定義を行い実装・検証するさいは，システムの処理時間，オーバーヘッドについて検証する必要がある．

6 おわりに

6.1 まとめ

IoT 技術はさらに普及がすると考えられているが，それに伴う危険性を十分に知る必要がある．利用者の目に見えない部分で安全に技術を利用するためのシステムを設計するために，セキュリティの観点から佐野の研究と比較しアーキテクチャを提案した．利用者のシステム使用状況に応じて付与するセキュリティ機能選択を動的再構成するために，自己適用のための PBR パターンを適用した．

6.2 今後の課題

各コンポーネントごとにコンテキストを取得する方法とセキュリティ付与の詳細な定義を行うことで実装が可能となる．実装後に処理時間の問題を検証する必要がある．またシステムの予期せぬデバイスが接続された場合，管理者に通報し，利用者に警告するなどの機能を考慮する必要がある．

7 参考文献

参考文献

[1] 江坂篤侍, 野呂昌満, 沢田篤史: インタラクティブシステムのための共通アーキテクチャの設計, コンピュータソフトウェア, Vol.35, No.4, pp.3-15 (2018).

[2] 江坂篤侍: 自己適応を目的としたソフトウェアアーキテクチャの構築と運用に関する研究, 南山大学 数理情報研究科 数理情報専攻 博士後期課程 博士論文, (2018).

[3] 佐野達也, “IoT システムの脆弱性分析に基づくアスペクト指向アーキテクチャの設計, ” 南山大学 2017 年度卒業論文, (2018).

[4] IPA 独立行政法人情報処理推進機構: “IPA 情報セキュリティ 10 大脅威” https://www.ipa.go.jp/security/vuln/10threats 2018.html,2018.

[5] IPA 独立行政法人情報処理推進機構 技術本部 セキュリティセンター:“IoT 開発におけるセキュリティ設計の手引き” https://www.ipa.go.jp/files/000052459.pdf,2016.