2B-04

# セキュリティ検証における開発情報に基づく検証項目の順序決定手法

跡部　悠太†　　千田　修一郎†　　伊藤　益夫†　　西山　博仁†

三菱電機株式会社　情報技術総合研究所†

## 1. はじめに

近年，組込み機器がネットワークに接続されていく一方で，セキュリティ攻撃に対する懸念から，セキュリティ機能の実装が必要とされるようになっている．

セキュリティ機能を実装した機器に対して，セキュリティ機能が動作しているか検証を行うことをセキュリティ検証と呼ぶ．セキュリティ検証方法の 1 つに，検証対象機器に対して実際にセキュリティ攻撃を実行する方法がある．このセキュリティ検証において、検証項目である攻撃を自動で生成・実行するセキュリティ検証ツール [1] [2] [3]がある．これにより，人的ミスによる間違いや見落としを避けることができる．しかし，大規模な組込みシステムに対しては，セキュリティ検証に時間がかかる問題がある．

以上の背景から，本稿では，セキュリティ検証ツール実行後早期に脆弱性を検出可能な検証方法を提案する．特に，設計・開発時に得られる開発情報と，開発情報を脆弱性と関連付ける情報から，脆弱性を検出する可能性を予測することによる検証項目の順序決定手法を提案する．

## 2. 関連研究

ソフトウェアのシステムテストにおいて，機能毎に算出した優先度から検証順序を決定し，システムバグの早期検出を図る手法が提案されている [4]．しかし，セキュリティ検証において必ずしも脆弱性の早期検出に有効ではない．なぜなら，機能内に含まれる脆弱性に対し，過去の実績から優先度を下げてよい検証項目や，逆に実装方法の問題で優先度を上げる必要のある検証項目があったとしても，機能単位では全て同等と扱われるからである．また，優先度計算に用いるメトリクスが人手によって点数化されており，開発作業のオーバーヘッドの発生及び属人的な優先度付けが問題である．

A Consideration on Sequencing of Security Verification Based on Development Information

† Yuta Atobe, Shuichiro Senda, Masuo Ito and Hirohito Nishiyama,
Information Technology R&D Center, Mitsubishi Electric Corporation.

## 3. 開発情報に基づく検証項目の順序決定

### 3.1. 提案手法の解決する課題

提案手法が解決する課題は以下の 2 つである．

A) 脆弱性に対して検証優先度を算出すること

B) 属人的な要素を排除すること

### 3.2. 提案手法の概要

脆弱性に対する優先度は，脆弱性に関連する機能の優先度と攻撃の優先度から算出する．このため，最初に機能と攻撃の各優先度を定める．脆弱性の優先度が決定された後，脆弱性の優先度順に各脆弱性に関連する攻撃を列挙し，最初に定めた攻撃の優先度から攻撃の順序を定める．

### 3.3. 脆弱性における依存関係の明確化

3.2 節で示した提案手法の内容を実施するために，脆弱性における依存関係を明確化させる．脆弱性を利用しシステムの財産を脅かす事象が脅威であり，その脅威を具現化する手段が攻撃であることから攻撃は脆弱性に依存する関係が存在する．また，セキュリティ検証における脆弱性は，検証対象機器が持つ機能に対して洗い出す．脆弱性には，物理的な要因などによる脆弱性も存在するが，セキュリティ検証ツールによって検証可能な脆弱性は機能に関連付けられ，脆弱性は機能に依存する関係が存在する．

上記より，脆弱性には図 1 に示す依存関係が存在する．この依存関係と機能，攻撃に関する情報を用いることで，脆弱性に対して優先度を決定することが可能である． 例えば，図 1 において，機能 A が複雑に設計された機能であり，攻撃①が対策の不備を生じやすい攻撃である場合，関連する脆弱性 a は高い優先度を算出する．

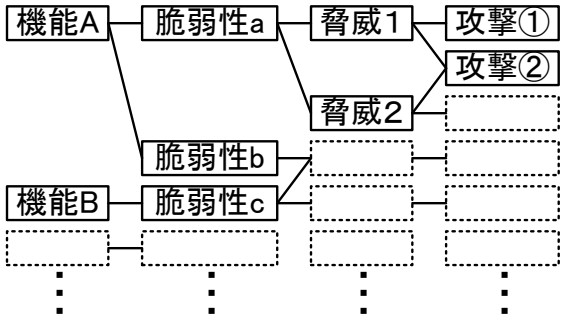


図 1 脆弱性における依存関係.

表 1 品質管理情報.

| | テスト密度 | 関数の行数 | … |
|---|---|---|---|
| 基準値 | 25 | 160.0 | |
| 機能 A | 24 | 159.0 | |
| 機能 B | 22 | 225.0 | |
| 機能 C | 30 | 231.0 | |
| … | | | |

表 2 検証実績情報.

| | NG 件数 | 平均改修時間 | … |
|---|---|---|---|
| 攻撃① | 5 | 4 | |
| 攻撃② | 1 | 2 | |
| 攻撃③ | 0 | 0 | |
| … | | | |

### 3.4. 優先度算出のための開発情報

3.3 節で明確化した依存関係において，機能に関連する情報は設計・開発時に取得でき，攻撃に関する情報は検証時に取得可能である．よって，提案手法では，以下の 2 つの情報を用いる．

(1) 品質管理情報

(2) 検証実績情報

品質管理情報とは，テスト密度や関数の行数などであり，機能の品質管理をするためのメトリクスである．表 1 のように各機能に対して品質管理メトリクスを保持する．品質メトリクスは，[5]で推奨している品質指標などを用いる．一般的に品質の悪い機能にはバグが含まれている．バグは攻撃の対象となることが多いため，品質の悪い機能は脆弱性の存在可能性が高いと考えられる．

検証実績情報とは，過去の検証実績を蓄積した情報であり，攻撃に対応する．例えば，NG と判定された回数やその改修に要した平均時間などがある．表 2 のように各攻撃に対して検証実績を取得し，保持する．過去に NG 判定と導いた攻撃は，対策の不備を生じやすく，脆弱性の存在可能性が高いと考えられる．

### 3.5. 脆弱性の優先度に基づく検証順序決定手法

脆弱性における依存関係及び 3.4 節で示した 2 つの情報から脆弱性に対する優先度を算出し，検証項目の実行順序を決定する手順を示す．提案手法は，主に以下の 4 つの Step で構成される．図 2 に提案手法の各 Step の流れとデータフローを示す．

Step1. 機能に対する優先度計算処理

機能に対する優先度は機能毎の各品質管理情報から計算する．各品質管理情報の品質目標などの基準値を基に品質の良し悪しを判定し，品質が悪いと判定された品質管理情報に設定した重みを加算することで優先度を決定する．

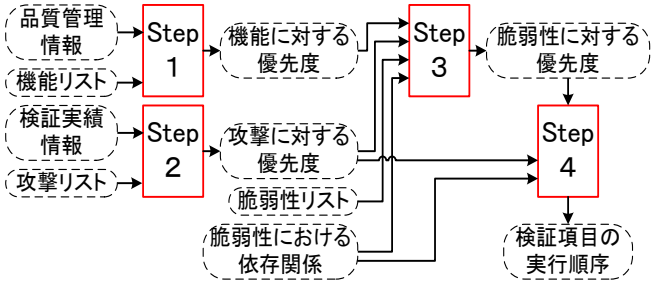


図 2 提案手法の実行 Step とデータフロー.

Step2. 攻撃に対する優先度計算処理

攻撃に対する優先度は攻撃毎の各実績情報から計算する．各実績情報に重みを設定し、重みと実績情報の値を乗算した値の和を優先度とする．

Step3.脆弱性に対する優先度計算処理

脆弱性に対する優先度は Step1 及び Step2 で算出した各優先度から計算する．脆弱性における依存関係のある機能の優先度の最大値及び攻撃の優先度の最大値の和を脆弱性の優先度とする．計算式を以下に示す．このとき，機能の優先度に対する攻撃の優先度の重み係数を設定する．

脆弱性の検証優先度

＝関連攻撃の検証優先度の最大値 × 重み係数

＋関連機能の検証優先度の最大値優先度

Step4.脆弱性の優先度に基づく実行順序決定

Step3 によって算出された脆弱性の優先度に基づいて検証項目の実行順序を決定する．

## 4. おわりに

本稿では，開発情報を基に算出した脆弱性の検証優先度に基づいて，検証の実行順序を決定する手法を提案した．提案手法では，脆弱性に対して検証優先度を決定することを可能にし，セキュリティ検証ツール実行後早期に脆弱性が検出可能な検証項目の実行順序を決定することが可能である．また，明確な基準値に基づいて検証優先度を算出することにより，属人的な要素を排除し，機械的に処理可能な手法である．

## 参考文献

[1] Paros project, http://parosproxy.org/index.shtml.

[2] OWASP (The Open Web Application Security Project), https://www.owasp.org/index.php/Category:OWASP_WebScarab_Project.

[3] 小菅祐史，河野健二，“Amberate: Web アプリケーションの脆弱性自動検出フレームワーク，” コンピュータソフトウェア, 2011.

[4] 平山雅之ほか，“機能モジュールに対する優先度に基づいた選択的ソフトウェアテスト手法の提案，” 信学技報, SS-2001-6, pp.1--8, 2001.

[5] 【改訂版】組込みソフトウェア開発向け品質作り込みガイド, IPA SEC, 2012.