システムに潜む未知の脆弱性を検出するセキュリティテスト技術

Security Testing Technology to Detect Unknown Vulnerabilities Hidden in Systems

兒島　尚　　坂口　昌隆　　大津　昭彦　　永井　和哉

あらまし

様々なサイバー攻撃が活発になっている中で，ソフトウェア製品の脆弱性の報告数も年々増加の一途をたどっている。また，製品やシステムに潜む脆弱性の存在は，サイバー攻撃を受ける大きな要因となっており，攻撃に先んじた脆弱性対策の整備が急務となっている。富士通研究所は，攻撃者の視点やマインドを取り入れたファジングと呼ばれるテスト手法に着目し，未知の脆弱性を事前に検出するセキュリティテスト技術を開発した。本技術は，より短時間で脆弱性を検出できるテストデータを生成する「最適テストデータ生成」や，未知のプロトコルを自動的に解析して適切に評価する「プロトコル自動順応」といった機能を備えている。また，近年増加するIoT製品にも対応している。富士通では，製品・システムの出荷前検証に本技術を活用しており，組織的なセキュリティ品質の強化に役立てている。

本稿では，セキュリティテスト技術の機能について述べる。

Abstract

As various cyberattacks become increasingly rampant, the number of reported vulnerabilities in software products is steadily increasing year by year. The presence of vulnerabilities hidden in products and systems is a major factor behind cyberattacks, and vulnerability measures to forestall attacks urgently need to be put in place. Fujitsu Laboratories has turned its attention to a testing technique called fuzzing, which incorporates the perspectives and mentalities of attackers, and developed a security testing technology that detects unknown vulnerabilities in advance. This technology provides functions such as “optimum test data generation” to generate test data that allow for the detection of vulnerabilities in a shorter amount of time and “automatic protocol adaptation” to automatically analyze and appropriately evaluate unknown protocols. It also supports IoT products, which are increasing in recent years. Fujitsu makes use of this technology for pre-shipment verification of products and systems, helping to systematically strengthen security quality. This paper describes the functions of this security testing technology.

1. まえがき

近年，様々なサイバー攻撃が活発になっている中で，ソフトウェア製品の脆弱性の報告数も年々増加の一途をたどっている。独立行政法人情報処理推進機構（IPA）が公開する脆弱性対策情報データベースJVN iPediaの登録状況によれば，脆弱性の報告数は毎年20％程度増加し，累計で約9万件近くに上っている。(1)

また，製品やシステムにおける脆弱性の存在は，サイバー攻撃を受ける大きな要因となっている。このため，安心・安全な製品やシステムをお客様に提供する上で，脆弱性対策の整備が急務となっている。更に，IoT製品の普及に伴い，より高度に自動化された攻撃事例が増えている。例えば，IoTマルウェア「Mirai」は，数百Gbps規模のトラフィックを伴うDDoS（Distributed Denial of Service）攻撃を発生させたと言われている。(2) IoT製品は，単機能ながらネットワークへの接続が前提となっており，脆弱性がサイバー攻撃につながりやすいため，その対策が欠かせない。

富士通では，ファームウェアを含めたソフトウェア製品のセキュリティ品質を向上させるために，セキュア開発推進チームを中心として様々な取り組みを行っている。(3)各開発工程では，セキュリティ強化を目的としてセキュリティアーキテクトと呼ばれる専門の担当者を割り当てて，組織としてセキュリティテストなどを確実に実行している。そして，脆弱性の影響を受ける可能性があるサーバ製品，ネットワーク製品，IoT製品については，出荷前のテスト工程で問題があれば改修するルールを定めている。

富士通研究所は，未知の脆弱性検出に有効なファジング手法を活用したセキュリティテスト技術を開発した。これにより，攻撃者が行うであろう攻撃に先んじて，出荷・稼働前に製品やシステムの脆弱性を検証できる。

本稿では，セキュリティテスト技術の代表的な機能の特長，注力した開発のポイント，および脆弱性検出の効果について述べる。

2. ファジング手法によるセキュリティテスト技術

本章では，未知の脆弱性検出に有効なファジング手法を活用したセキュリティテスト技術について述べる。

2.1 ファジング手法とは

ファジング手法とは，テスト対象の製品やシステムに対して，通常想定しないような大量のデータを入力して，ソフトウェアの不具合を発見するテスト手法である。この手法は従来，テスト対象の内部仕様を知らない攻撃者が主に利用するものであった。近年では，通常の機能テストで検出できないような未知の脆弱性検出に有効な手法として，大手ベンダーの開発プロセスへの採用が進んでいる。

富士通研究所では，早くからその効果に着目し，独自のファジングツールの研究開発を行っている（図-1）。本ツールから様々な製品やシステムに大量の攻撃データを入力することで，脆弱性を検出できる。

2.2 ファジングツールの機能

富士通研究所が開発したファジングツールは，主な機能として「最適テストデータ生成」と「プロトコル自動順応」を備えている。

(1) 最適テストデータ生成機能

多くの脆弱性検出に効果の高い最適なテストデータを，より短時間で生成する機能である（図-2）。ここでは，多数の製品・システムのセキュリティテストを行ってきた富士通研究所の経験に基づいた，独自のノウハウを活用している。

従来のファジング手法では，まず正常データを徐々に変異させて異常データを生成し，順々にテストしていくことで脆弱性を検出していた。しかし，正常データのサイズが大きかったり変異パターンが多かったりすると，変異が特定の箇所だけに局所化するため，テストデータの異常度がなかなか上がらず，脆弱性検出に時間がかかることがあった。

そこで，まず異常データからテストを開始して，徐々に正常データに復元していくという新しい手法

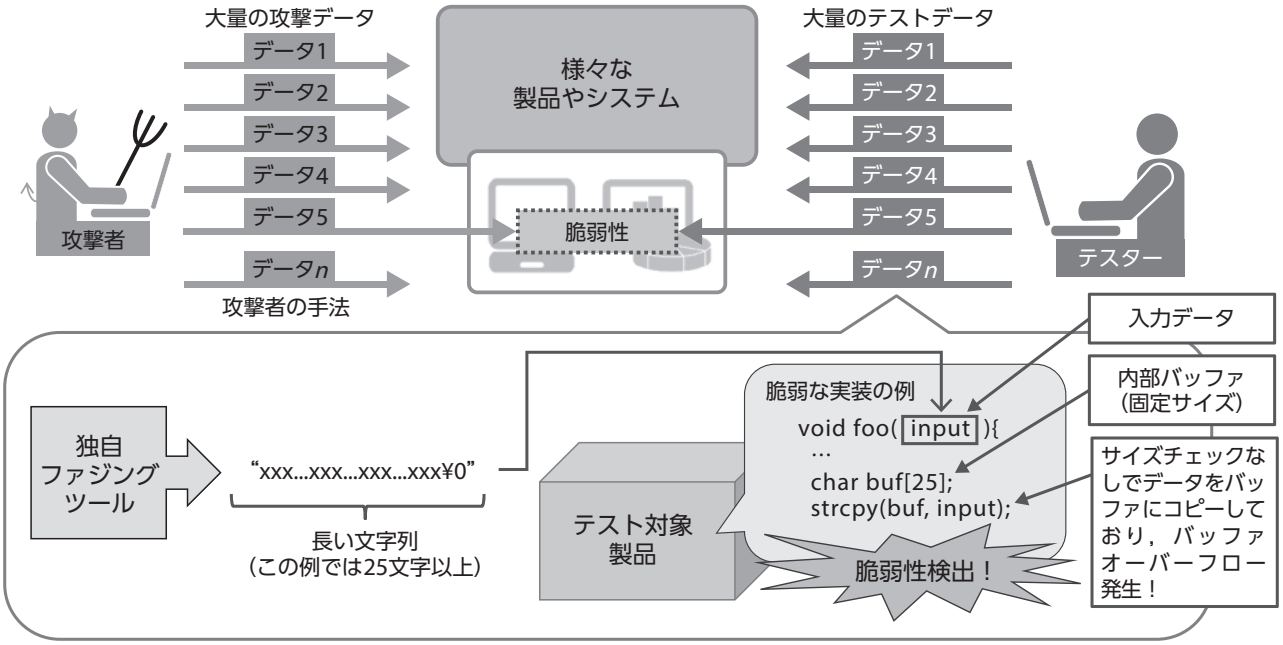

図-1　開発したファジングツールの概要

図-2　最適テストデータ生成機能

を考案し，より短時間で脆弱性を検出することに成功した。

(2) プロトコル自動順応機能

テスト対象の運用環境において，キャプチャーした正常パケット群からパケットのデータ構造を自動的に解析し，プロトコルに順応したテストデータを生成する機能である（図-3）。従来のツールでは，プロトコルごとに専門家が手動で解析した順応エンジンを組み込むというアプローチが主流であった。しかし，このアプローチは，仕様がRFC（Request For Comments）などで公開された既知の標準プロトコルにしか対応していない。

一方，開発したツールでは，全く未知のプロトコルからヒューリスティクス分析(注)などを駆使して，プロトコル一般の共通要素である文字列・整数などを表す箇所を特定することで，多種多様なプロトコルに対応できる。

(注)　人間の様々なノウハウをそのままプログラムとして表現した方式一般のこと。富士通研究所の長年のセキュリティテストの経験に基づいたノウハウを活用している。

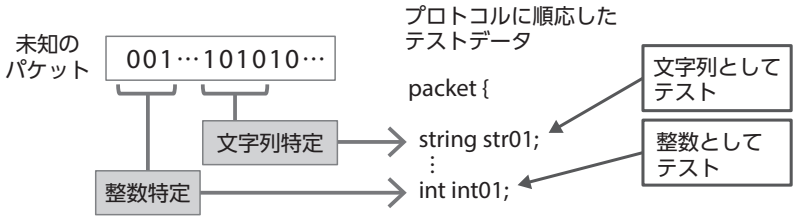

図-3　プロトコル自動順応機能

2. 3　SSHプロトコル順応エンジン

開発したツールは，前節で述べたようにプロトコルの種類を問わず適用できる。更に，テスト精度の向上のために，重要性の高いプロトコルについては専用の順応エンジンを開発している。ここでは代表的な例として，SSH（Secure Shell）プロトコル用の順応エンジンについて説明する。SSHは，通信路の認証や暗号化を実現するネットワークプロトコルであり，[4] サーバ製品やネットワーク製品などに搭載されているコマンドラインベースのリモート管理コンソールの保護などに広く利用されている。

開発したエンジンは，現在の主流バージョンであり，攻撃にさらされる機会が多いSSH-2のサーバソフトウェアの評価を対象としている。従来のファジングツールでは，SSH-2メッセージの評価に対応していなかった。富士通研究所のファジングツールでは，これに対応したランダムテストという方法によって，約76万件のテストパケットに対するサーバの耐性を評価できる。

生成されるテストパケットの例としては，バッファオーバーフローの誘発を狙いとしたヌル文字（例：図-1の¥0）で終了する長い文字列などがある。また，整数オーバーフローの誘発を狙いとした境界値付近の整数値（例：255や65535）や，予測不可能な挙動を引き起こすランダムな値なども生成できる。更に，サーバのプロトコル状態やコネクションの接続方法の不一致を狙いとした高度なテストにも対応しており，実装上の様々な問題を網羅的に評価できる。

2. 4　脆弱性検出の効果

開発したエンジンを利用して，ネットワークスイッチ製品を評価したところ，6種類の製品のうち5種類から未知の脆弱性を検出した。[5] いずれの場合も，テストパケットの送信直後に製品が応答不能になり，しばらくすると自動的に再起動されるものであった。これは，実運用において攻撃を受けた場合には，サービス停止などの深刻な被害が生じる可能性があるものであった。

これらの脆弱性については，情報セキュリティ早期警戒パートナーシップ[6] に基づき，IPAの脆弱性報告窓口に正規の手順で報告済みであり，各ベンダーによる修正対応が完了している。また，注意喚起を目的として脆弱性公開サイトでも公開されている。[7]

3.　IoT向けMQTTプロトコルファジングツール

本章では，前述のファジングツールをベースとして，IoT製品やシステムのセキュリティ強化を目的に開発した，IoT向けMQTT（Message Queuing Telemetry Transport）プロトコルファジングツールについて述べる。

3. 1　MQTTとは

MQTTとは，多数機器間の同報通信を単純・軽量に実現するPub/Sub（Publish/Subscribe）型のメッセージングプロトコルである。[8] M2M（Machine to Machine）やIoTなど，機器の計算資源やネットワーク帯域の制約が強い環境でも，十分な性能が出せるように設計されている。IoT向けのデファクトスタンダードなプロトコルとして，様々なクラウドベンダーをサポートしている。

MQTTでは，サーバであるBrokerと，クライアントであるPublisher（IoT製品など）および

Subscriber（サービスなど）との間で任意のメッセージによる通信が行われる。

Publisherは，トピック名をラベルとして送信メッセージに付与し，Brokerに対してPUBLISHパケットを送信する。SubscriberはBrokerに対して，自身が購読したいトピック名の通知をあらかじめ依頼しておく。そしてBrokerは，PublisherからPUBLISHパケットを受け取った後，自身に接続しているSubscriberの中で，該当するトピック名を購読している全てのSubscriberを対象にPUBLISHパケットを一斉に送信する。

このように，Brokerを経由することで，MQTTでは1対多のメッセージ送信を実現している。

3. 2 開発ツールの特長

富士通研究所は，MQTTの仕組みに合わせてIoT向けのMQTTプロトコルに対応した機能を備えたファジングツールを開発した。

(1) 実装ソフトウェアの評価対応

MQTTプロトコルを実装するソフトウェアの評価に当たっては，プロトコル処理ロジックとペイロード処理ロジックの2種類の異なる対象が存在する。

プロトコル処理ロジックは，MQTTの仕様で定義された標準のメッセージ形式の構文解析や意味解釈，およびBroker，Publisher，Subscriber間の接続関係の管理などを行っている。これらの処理の実装は，Broker用のサーバ側ソフトウェアおよびPub/Sub用のクライアント側ライブラリで構成される。これらのソフトウェアやライブラリには，mosquitto[9]をはじめとするオープンソースソフトウェア（OSS）が広く利用されているが，既に多くの稼働実績があるため，改めてセキュリティ評価を行う必要性は低い。

一方，ペイロード処理ロジックは，Subscriberとなるアプリケーション側でメッセージの中身であるペイロードを受信した後に行うものである。これらは，個々のアプリケーションごとに特有の処理となるため，個別にソフトウェア実装の安全性を評価する必要がある。

富士通研究所が開発したツールでは，ペイロード処理ロジックを評価の対象としている。ツールはBrokerとして動作し，Subscriberとして接続してきたアプリケーション側に対してテストデータを含むペイロードを送信する。ペイロードはアプリケーションごとに異なる独自の形式となるが，前述のプロトコル自動順応機能を活用することで，未知のペイロード形式に対しても有効なテストケースを生成できる（図-4）。

(2) クライアント側へのファジングの対応

一般的に，ファジングの実施はテスト対象がサーバ側で動作するため，クライアント側にあるファジングツールから能動的にテストデータを送信してテストを行う。しかし，テスト対象がクライアント上

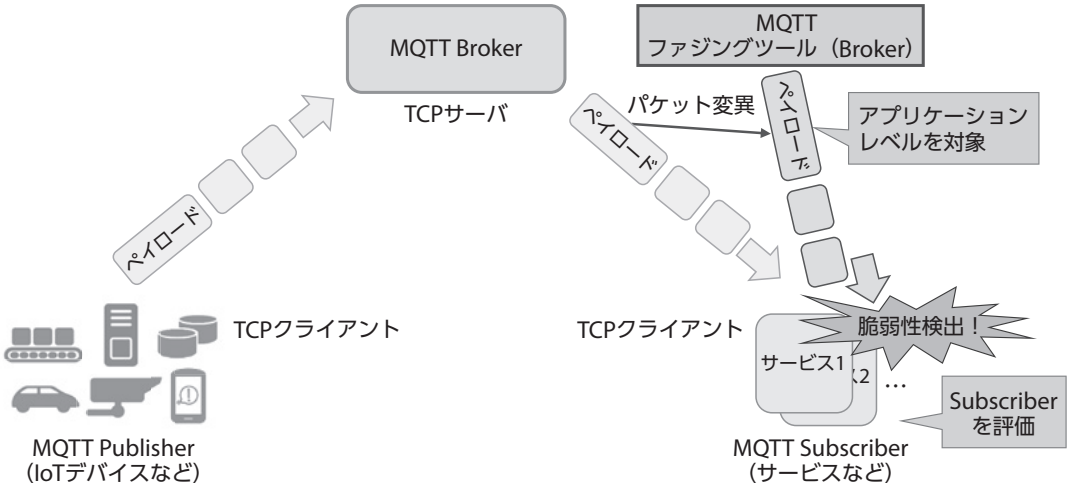

図-4　MQTTペイロードに対するファジング機能

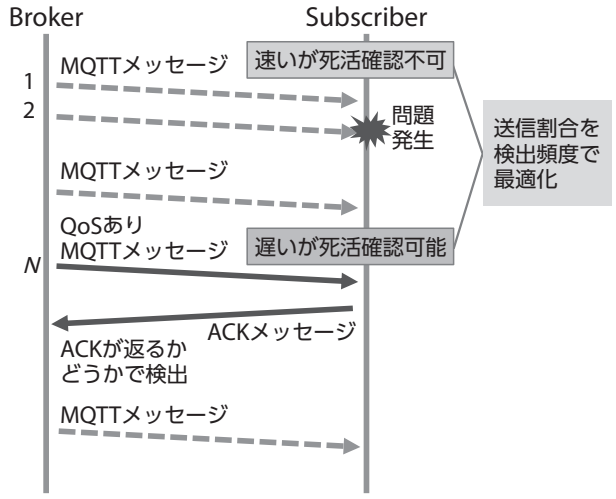

図-5　独自の死活確認手法

で動作する場合，ファジングツールはサーバ側にあるため能動的にテストを実施できない。そこで，開発したファジングツールでは，クライアント側のテスト対象をSSH経由のリモート制御で起動してMQTT接続を開始させることによって，能動的なテストを実現している。

また，テスト対象の死活確認も必要となる。サーバ側へのファジングの場合，テストデータに対する応答の有無によって，サーバの死活確認を行う。一方で，クライアント側ではサーバへの応答は行われないことが多く確認が難しい。これについては，MQTTの標準機能であるQoS（Quality of Service）を利用して解決できる。

QoSとは，MQTTメッセージの到達可能性を高めるために付与される属性情報であり，メッセージの受信側に到達確認メッセージ（Ackメッセージ）の送信を要求することで，メッセージ消失時の再送を実現する。QoSによって，クライアント側の死活確認が可能になるが，単純に全ての送信メッセージに対してAckメッセージの送信を要求すると，待ち時間が生じるためテストの効率が悪くなってしまう。そこで，開発したツールでは，QoSありメッセージによる死活確認のタイミングを，エラー発生率に応じて自動的に調整し最適化している（図-5）。これによって，効率的なクライアント側へのファジングを実現している。

3. 3　脆弱性検出の効果

開発したツールのターゲットは，アプリケーション独自のペイロード処理ロジックの脆弱性である。これらの脆弱性検出の有無は，アプリケーションの仕様や実装に依存する。そのため，アプリケーションの脆弱性検出の実績は公開されていないが，検出した脆弱性は開発者に通知し，修正されている。ただし，アプリケーションを対象とした評価の過程で，それらが利用するコンシューマ向けソフトウェアや，クラウドベンダーが提供するAPI（Application Programming Interface）においても脆弱性を検出している。これらについては，前述のパートナーシップに基づいた対応によってIPAに報告済みである。[10]

4.　セキュリティテスト技術を活用したサービス

本章では，開発したセキュリティテスト技術を活用したサービスについて述べる。

富士通では，開発した技術や脆弱性検出の知見・ノウハウをベースとして，お客様向けの有償のセキュリティ脆弱性検証サービスを提供している。[11], [12] 本サービスは，ファジング基本サービスとファジング再現支援・修正確認サービスで構成されている。

基本サービスでは，テスト計画の立案からファジングの実施，検出した脆弱性の再現性確認，結果報告まで，ファジングに必要な全ての工程を実施する。検出した脆弱性は，お客様に問題の特定と修正を行っていただく。その後，再現支援･修正確認サービスで確認するために再度のファジング実施と結果報告を行う。この再現支援・修正確認サービスは，IoT製品にも対応済みである。今後も総合ICTベンダーである富士通の社会的使命として，長年にわたり培ったセキュリティ品質向上の取り組みの経験を，本サービスを通じてお客様にも還元していく。

5.　むすび

本稿では，システムに潜む未知の脆弱性を検出するセキュリティテスト技術について述べた。

2004年，OSSを主な対象とした既知の脆弱性の公開・共有システムが，政府・業界団体主導で整備された。[6] 一方，ベンダーやシステムインテグレーターが開発する個々の製品・システムに潜む未知の脆弱性は，開発側組織において担保する必要がある。富士通では，本稿で紹介したファジング手法を活用したセキュリティテスト技術を中心として，進化する攻撃に対応し，お客様に安心・安全な製品およびシステムを提供していく。

参考文献

(1) 情報処理推進機構：脆弱性対策情報データベースJVN iPediaの登録状況 [2018年第3四半期(7月～9月)].
https://www.ipa.go.jp/security/vuln/report/JVNiPedia2018q3.html

(2) 日経BP社：DNSサービス「Dyn」への大規模DDoS攻撃、発信源は10万台のIoT機器.
https://tech.nikkeibp.co.jp/it/atcl/idg/14/481542/102800290/

(3) 富士通：富士通グループ 情報セキュリティへの取り組み (第2版).
http://www.fujitsu.com/jp/solutions/business-technology/security/secure/concept/index.html

(4) IETF：RFC4251 - The Secure Shell（SSH）Protocol Architecture.
https://tools.ietf.org/html/rfc4251

(5) 兒島 尚ほか：SSH-2サーバ向けファジングツール. コンピュータセキュリティシンポジウム (CSS) 2013.

(6) 情報処理推進機構：情報セキュリティ早期警戒パートナーシップガイドライン.
https://www.ipa.go.jp/security/ciadr/partnership_guide.html

(7) JVN：Japan Vulnerability Notes.
https://jvn.jp/

(8) OASIS：Message Queuing Telemetry Transport (MQTT).
http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html

(9) Eclipse Foundation：mosquitto.
https://mosquitto.org/

(10) 坂口昌隆ほか：MQTTファジングツールの開発とその評価. 暗号と情報セキュリティシンポジウム (SCIS) 2019.

(11) 富士通コンピュータテクノロジーズ：未知の脆弱性を検出する「ソフトウェア脆弱性検証サービス」を提供開始.
http://www.fujitsu.com/jp/group/fct/resources/news/press-releases/2016/1101.html

(12) 富士通コンピュータテクノロジーズ：MQTTプロトコルに対応した「ソフトウェア脆弱性検証サービス」を提供開始.
http://www.fujitsu.com/jp/group/fct/resources/news/press-releases/2018/0122.html

著者紹介

兒島　尚（こじま　ひさし）
富士通 (株)
サイバーセキュリティ事業戦略本部
システムインテグレーション全般のセキュリティ品質強化の推進に従事。

坂口　昌隆（さかぐち　まさたか）
(株) 富士通研究所
セキュリティ研究所
システムセキュリティ技術の研究開発に従事。

大津　昭彦（おおつ　あきひこ）
富士通 (株)
共通ソフトウェア開発技術本部
ミドルウェアおよびファームウェアを含むソフトウェア全般のセキュリティ品質強化の推進に従事。

永井　和哉（ながい　かずや）
(株) 富士通コンピュータテクノロジーズ
TMP事業部
テスト，自動化，および運用監視などに関するソフトウェア製品開発に従事。