



Jasst 2011 北海道 ライトニングトーク

Several Barriers to QA -B/W Interface-

Fourteenforty Research Institute, Inc.

株式会社 フォティーンフォティ技術研究所

<http://www.fourteenforty.jp>

ソフトウェア開発部

クウォリティアシュアランスエンジニア

内藤史郎

仕様理解の齟齬

開発担当者⇔ステークホルダー間で発生する
開発後期における仕様理解の齟齬

仕様理解の齟齬が発生する原因

両者のスキルセットも然ることながら、両者が目的を達成するための視点が異なることに概ね起因する

目的を達成するための両者の視点	
開発担当者	ステークホルダー
コード仕様や実装可能な構造など	システム全体でできる機能
↓	↓
LPPACKET PacketAllocatePacket(void); VOID PacketInitPacket(LPPACKET IpPacket,PVOID Buffer,UINT Length); BOOLEAN PacketReceivePacket(LPADAPTER AdapterObject,LPPACKET IpPacket,BOOLEAN Sync); BOOLEAN PacketSendPacket(LPADAPTER AdapterObject,LPPACKET pPacket,BOOLEAN Sync); (上記APIを使用して実装可能なことや制限 etc...)	<ul style="list-style-type: none"> ○DoS攻撃を検知できること ○フィッシングサイトにはアクセスさせないこと ○マルウェアが入り込む通信をブロックすること ○組み込みデバイスのためリソースに配慮すること ○社内Demoに使用するため説得力のあるGUIを要する
関数・API等、内部構造に精通しないと理解が困難である 実装以外の回避策が適切に定義できないことがある	散発的な要求のため実装上の辻褄が合わなくなることがある 内部構造上の制限事項を理解できないことがある

ここで仕様理解の齟齬が発生する



では、どう解決するのか？



複雑な事物をシンプルに

システム概念図にマッピングして
仕様ベース (Black Box) ⇔ 構造ベース(White Box)に置換する

構造ベース→仕様ベースの変換例

Input

コード仕様や実装可能な構造など



```
LPPACKET PacketAllocatePacket(void);
VOID PacketInitPacket(LPPACKET lpPacket,PVOID
Buffer,UINT Length);
BOOLEAN PacketReceivePacket(LPADAPTER
AdapterObject,LPPACKET lpPacket,BOOLEAN Sync);
BOOLEAN PacketSendPacket(LPADAPTER
AdapterObject,LPPACKET pPacket,BOOLEAN Sync);
(上記APIを使用して実装可能なことや制限 etc...)
```

Output

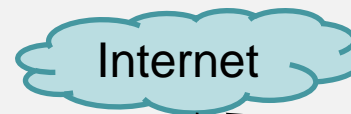
構造ベース→仕様ベース変換例



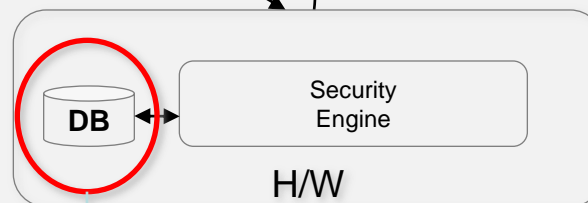
- 大量のリソースを使用できない
- ログも定期的に外部リソースに退避させなければならない

Mapping

システム概念とのマッピング



システム概念図



リソースに配慮する必要がある

仕様ベース→構造ベースの変換例

Input

散発的な仕様



- DoS攻撃を検知できること
- フィッシングサイトにはアクセスさせないこと
- マルウェアが入り込む通信をブロックすること
- 組み込みデバイスのためリソースに配慮すること
- 社内Demoに使用するため説得力のあるGUIを要する

Output

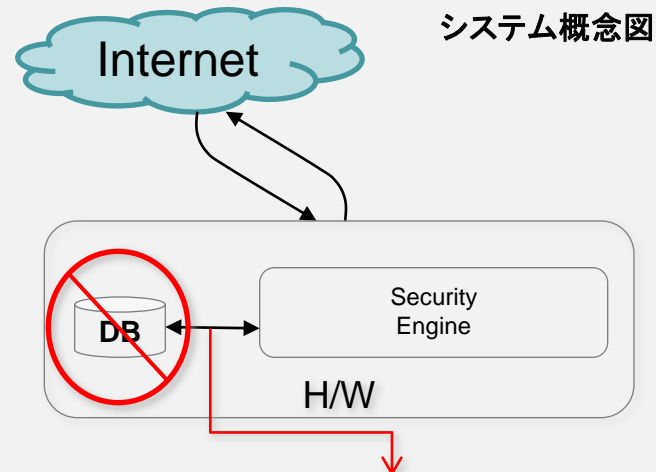
仕様ベース→構造ベース変換例



IDS/IPS_Flag = 1のとき、通信をブロック後squidで暫定的にREDIRECT設定された通信先パケットを生成・送信し、既定のサイトにリダイレクトさせる

Mapping

システム概念とのマッピング

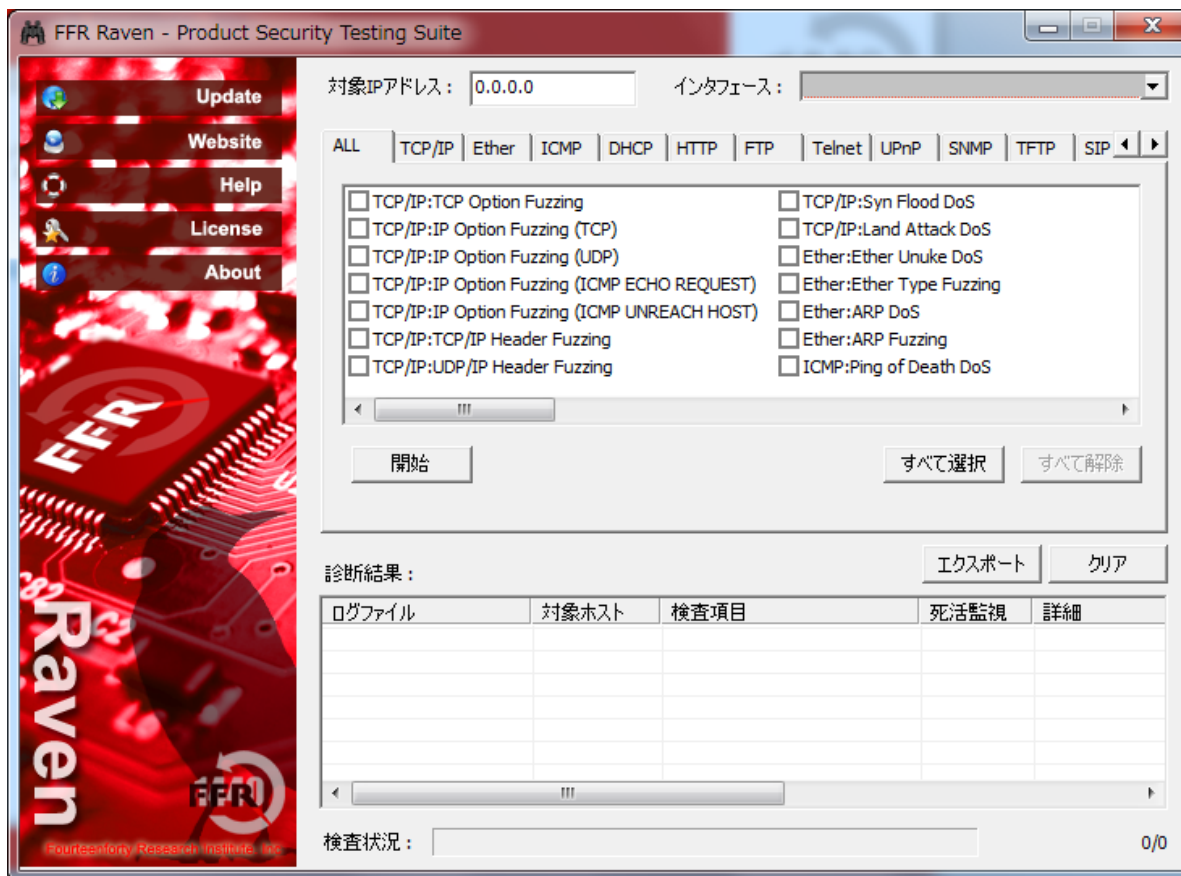


暫定処置としてここで対応可能

仕様をシンプルに理解するために必要なことは
システム概念図にマッピングするだけ

複雑なテストをシンプルに行うFFRの製品

FFR Raven®



ネットワーク製品・プロトコルスタックのテスト

通常プロトコル種別ごと、各フィールドごとにテストケース設計

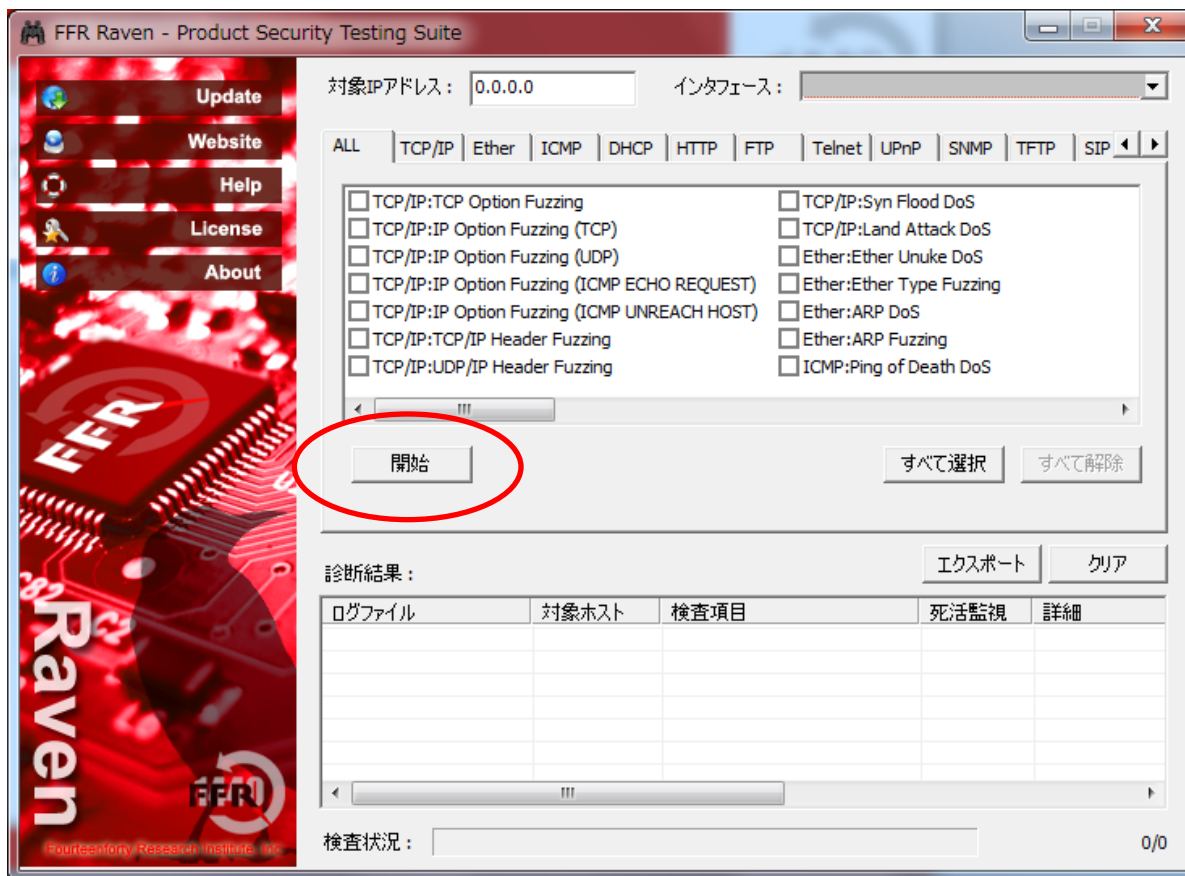
- ・ TCP/IP Option
- ・ TCP Header
- ・ UDP/IP Header
- ・ Ether Type
- ・ ICMP Code
- ・ ICMP Type
- ・ DHCP Option
- ・ HTTP GET
- ・ HTTP Host Request Header
- ・ HTTP Content-Length
- ・ SNMP Community Name
- ・ . . .

手間・工数が相当量掛る

これらのテストをシンプルに一度に行うために
QA担当者はRavenの開始ボタンを押すだけ

複雑なテストをシンプルに行うFFRの製品

FFR Raven®



ありがとうございました



Fourteenforty Research Institute, Inc.

株式会社 フォティーンフォティ技術研究所

<http://www.fourteenforty.jp>

ソフトウェア開発部

クウォリティアシュアランスエンジニア

内藤史郎