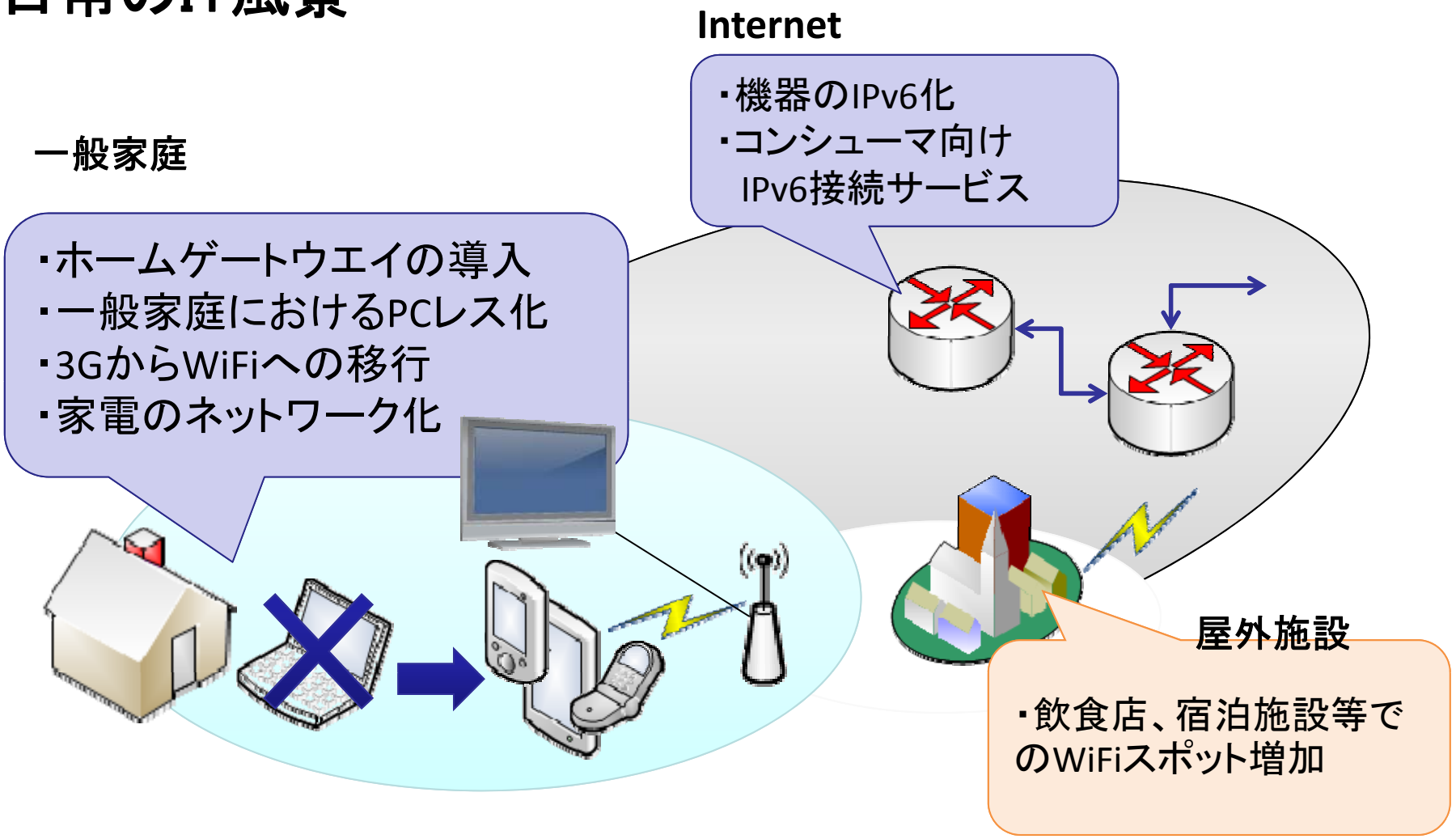
IPA®
INFORMATION-TECHNOLOGY PROMOTION AGENCY, JAPAN

# 今、情報家電のセキュリティ対策の課題は何か

独立行政法人　情報処理推進機構
セキュリティセンター 研究員
鵜飼裕司 博士(工学)
y-ukai@ipa.go.jp

# 日常のIT風景


Internet
・機器のIPv6化
・コンシューマ向け
IPv6接続サービス
一般家庭
・ホームゲートウエイの導入
・一般家庭におけるPCレス化
・3GからWiFiへの移行
・家電のネットワーク化
屋外施設
・飲食店、宿泊施設等で
のWiFiスポット増加

# 非PC端末のセキュリティの現状

**近年、組み込みシステムの脆弱性報告や、システムを狙う攻撃が急増**
(ルーター、モバイルデバイス、IP電話、etc…)

・影響が広範囲になりつつある。
・古典的な脆弱性を持つ組み込み機器が数多く存在。
・ベンダーの体制が不十分 (技術的、社会的対応)。
・対策ソリューションが不十分 (アップデート問題、検出困難)。
・効率的なセキュリティ・テスト手法は研究途上。
・セキュリティ・テスト可能な人材は世界的にも希少。現状、セキュリティ研究者のみ。

**組み込みは、攻撃者にとって格好のターゲットとなりつつある**

# アンダーグラウンドによる攻撃の傾向

- **組み込みシステムも、脆弱性があれば攻撃可能**

  - 対象が組み込みシステムでも、PCと同様の攻撃が可能。
  - セキュリティ脆弱性の本質、検査手法、対処手法は同じ。
  - アップデートは困難。製品回収のリスク。

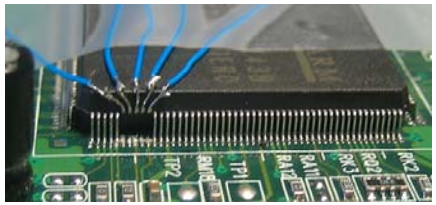
- **攻撃の傾向と実際**

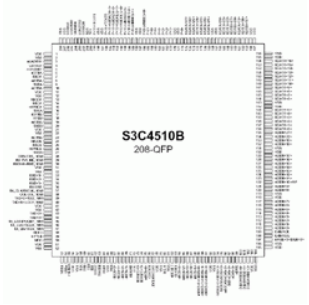
  - ハッカー・アンダーグラウンド・ビジネス台頭。
    （DDoSやSPAMの踏み台、情報搾取など多発）
  - WindowsやUNIXの対策が進む中、次のターゲットは組み込み。
  - 組み込みに対する攻撃ノウハウも普及。
  - テストが不十分。古典的な脆弱性を含む製品が大量に存在

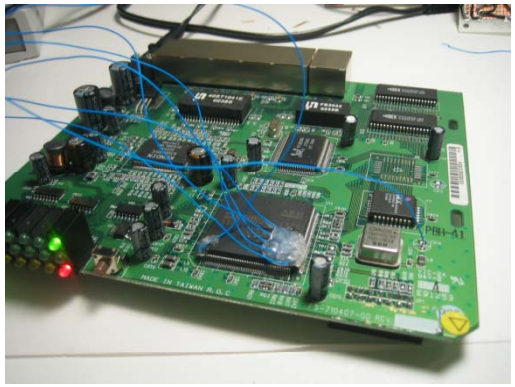
**アンダーグラウンドにとって、「組み込みシステムは次のターゲット」**

# 脆弱性解析例 – “クローズド”でも安全ではない

JTAG強制接続

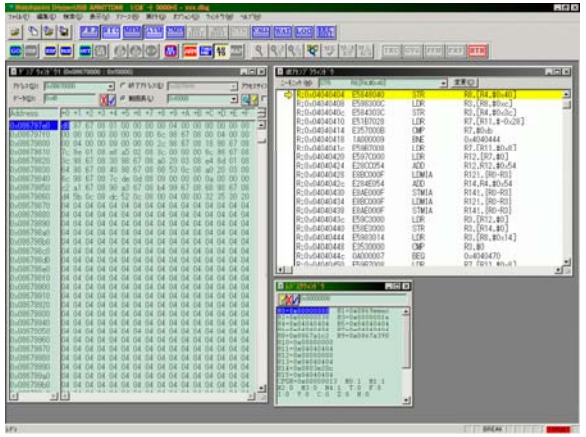





汎用JTAG


デバッガで解析



# 事例：Router Hacking

- D-Link Router UPNP Stack Overflow

  概要:　http://www.blackhat.com/html/bh-europe-06/bh-eu-06-speakers.html#Jack
  発表資料:　http://www.blackhat.com/presentations/bh-europe-06/bh-eu-06-Jack.pdf

- 個人向けブロードバンドルータに脆弱性が発見される。
- 「M-SEARCH」コマンドに長いオプションを指定すると落ちる。
- 実際には、攻撃者が任意のコードを実行可能となる脆弱性。
- D-Link を解析し、攻撃コードを作成。
  - 脆弱性を突き、管理者パスワードをクリア、リモート管理を強制的に有効にする
  - その後、攻撃者は改竄したファームをリモートからアップロード
  - これは、ユーザがexeファイルをダウンロードしたら、ウイルス感染のように特定のコードをそのexeに注入するファーム

→ ダウンロードするexeがすべてウイルス感染してしまう

# 事例：ATM Hacking

- リモート監視機能を利用してATMに外部から接続
  - ベンダーの既定状態で、リモート監視機能が有効
- 認証回避の脆弱性を悪用し、ﾊﾞｯｸﾄﾞｱ付きﾌｧｰﾑｳｪｱに差替え
- ATMのフロントパネルを操作し、不正に現金を引き落とし

※http://www.youtube.com/watch?v=qwMuMSPW3bU

# 事例：Coffee Maker Hacking

- Jura F90 Coffee maker
  - ネットワーク経由でコーヒーのパラメーター調整が可能
  - 自分好みのコーヒーを淹れることができる
- ソフトウェア脆弱性が存在
  → リモートから不正に操作することが可能

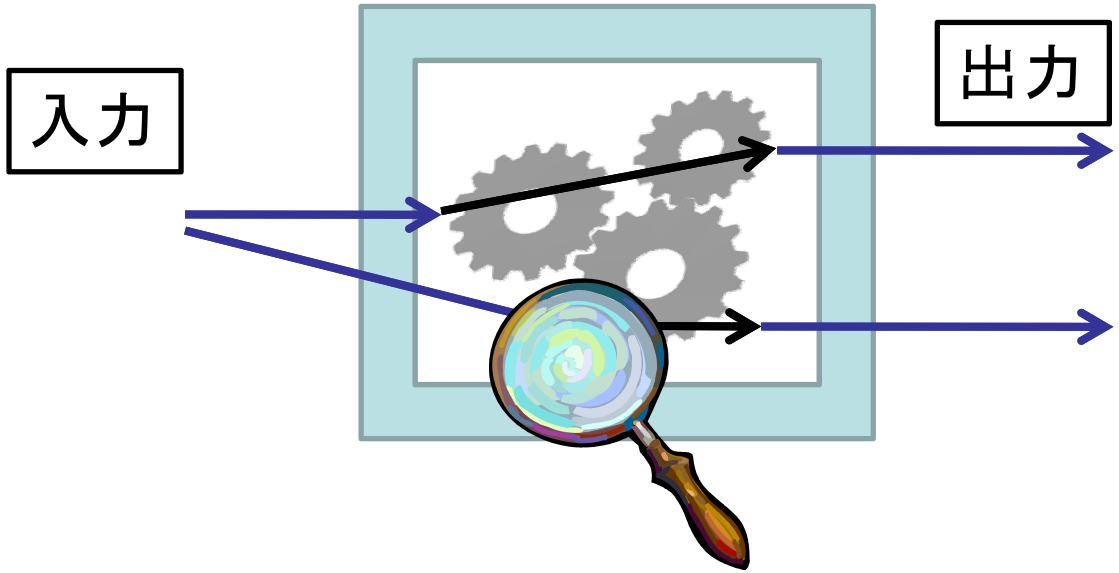
Guess what - it can not be patched as far as I can tell ;) It also has a few software vulnerabilities.

Fun things you can do with a Jura coffee maker:
1. Change the preset coffee settings (make weak or strong coffee)
2. Change the amount of water per cup (say 300ml for a short black) and make a puddle
3. Break it by engineering settings that are not compatible (and making it require a service)

http://www.securityfocus.com/archive/1/493387

# 脆弱性検査手法：ホワイトボックス検査

- 構造に着目
  - 条件分岐などの内部構造を考慮
  - コーディングエラーの検出に利用される

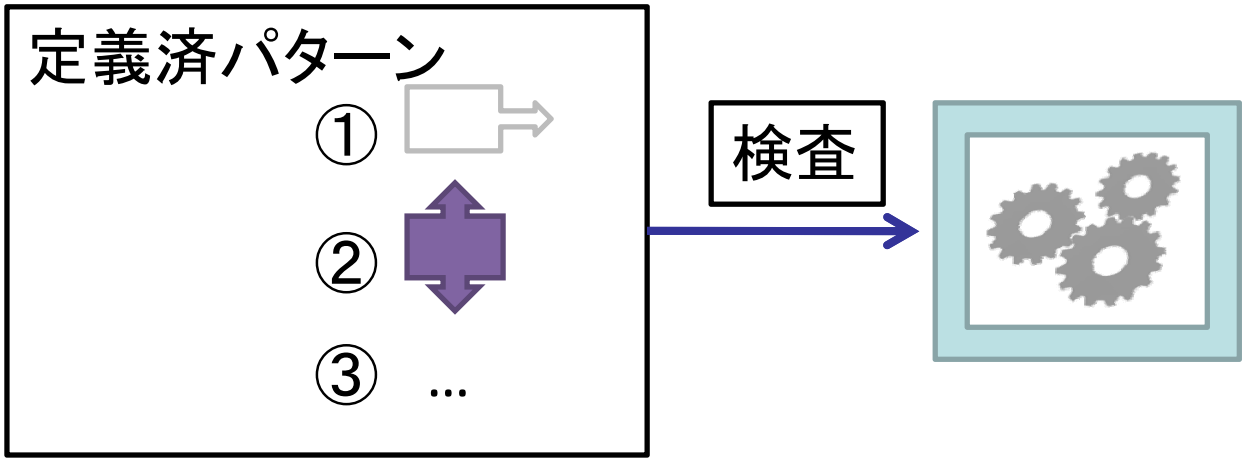


# 脆弱性検査手法：ブラックボックス検査

- 入出力結果に着目
  - 仕様に沿った動作をするか否かを検査
  - 内部構造が不明でも適用できる

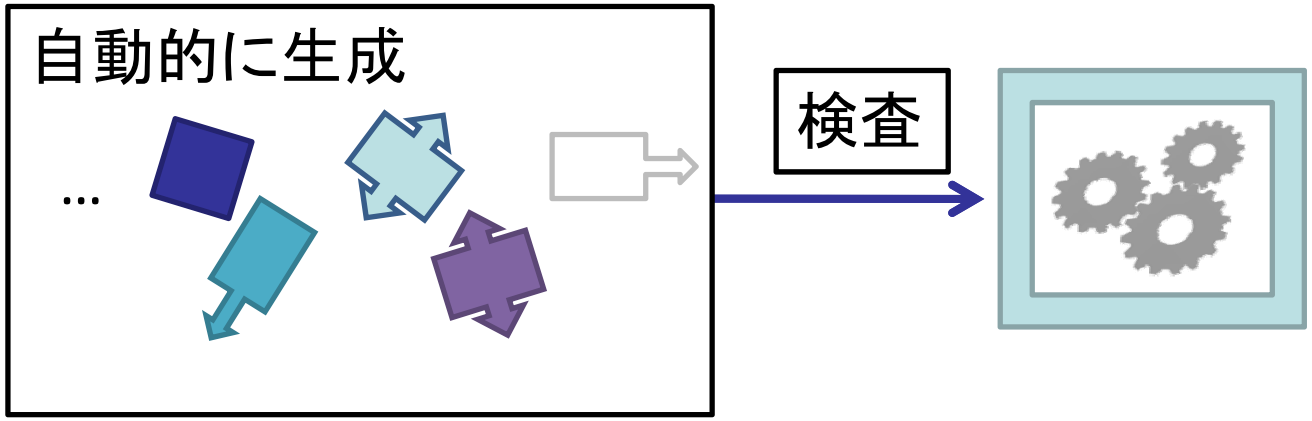
# パターン検査

- 既知、または指定したパターンで検査
  - 典型的な問題パターンをあらかじめ定義し検査
  - 既知の脆弱性を効率よく洗い出すことができる
  - 定義していないパターンは検知できない



# Fuzzing

- 大量のデータを使った検査
  - 大量のデータを自動的に生成・検査
  - 脆弱性の内容をあらかじめ想定する必要がない
  - 想定しない脆弱性の検出が可能



# パターン検査 vs Fuzzing

- **パターン検査**
  - 基本的には既知の問題を取り扱う手法のため、**開発段階のテストには不向き。**

- **Fuzzing**
  - Fuzzingは未知の脆弱性も発見可能なため、**開発段階のテストにも向いている。**

**Fuzzingでの検査**

- 2004年頃からセキュリティ脆弱性検出での利用が本格化
- 大手ソフトウェアベンダの製品においても多数の脆弱性を検出
- IPパケット、HTMLレンダリングエンジン、Javascriptエンジン、圧縮展開コンポーネント、画像処理ライブラリ等

# Fuzzingの難しさ

時間と検査性能のトレードオフ

4byteのデータをファジングする場合
→　　0x00000000～0xffffffffの約40億パターン
→　約50日(1000pkts/秒）

効率の良い検査を行う必要がある

Fuzzingの性能は網羅性ではなく「絞込み」

現実的な時間で多数の脆弱性を洗い出せるFuzzingアルゴリズムが優良

# 情報家電におけるセキュリティの現状と対策

1. OS/ミドルウエア等は、汎用的なものが利用されている事が多い
   - 組み込みシステムにおけるセキュリティ問題と共通点が多い
   - 対策も同様

2. 組み込みシステムに残留する脆弱性の除去が重要

3. 機器の品質管理プロセスにセキュリティ検査を
   - 開発コストとインシデント対策コストとのバランス
   - 簡易的な検査でも効果大
   - ソースコード検査やFuzzingなどで少なくとも基本的な検査を