Monthly Research

セキュアハードウェアの登場とその分析

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

Ver2.00.02

セキュアハードウェア

- ハードウェアレベルでのセキュリティ拡張や、
  それを実装したハードウェアが提案されている
  - 通常のマイクロプロセッサを拡張することで
    柔軟性を確保する試みもある
  - 今回は主に ARM TrustZone の仕組みから攻撃の可能性や
    “TEE” を通じたセキュリティハードウェア側の防御の可能性をさぐる

秘匿性を要求する処理

- 重要性の高いデータの分離
  - 重要データの保護、分離、分離したままでの保存
    - 認証情報
    - デジタル支払い処理に関する情報
- 広義の DRM
  - 秘匿したメディア処理
  - DRM のかかったメディアを秘匿（暗号化）したまま転送、出力（HDCP やバスの分離による）

ARM TrustZone

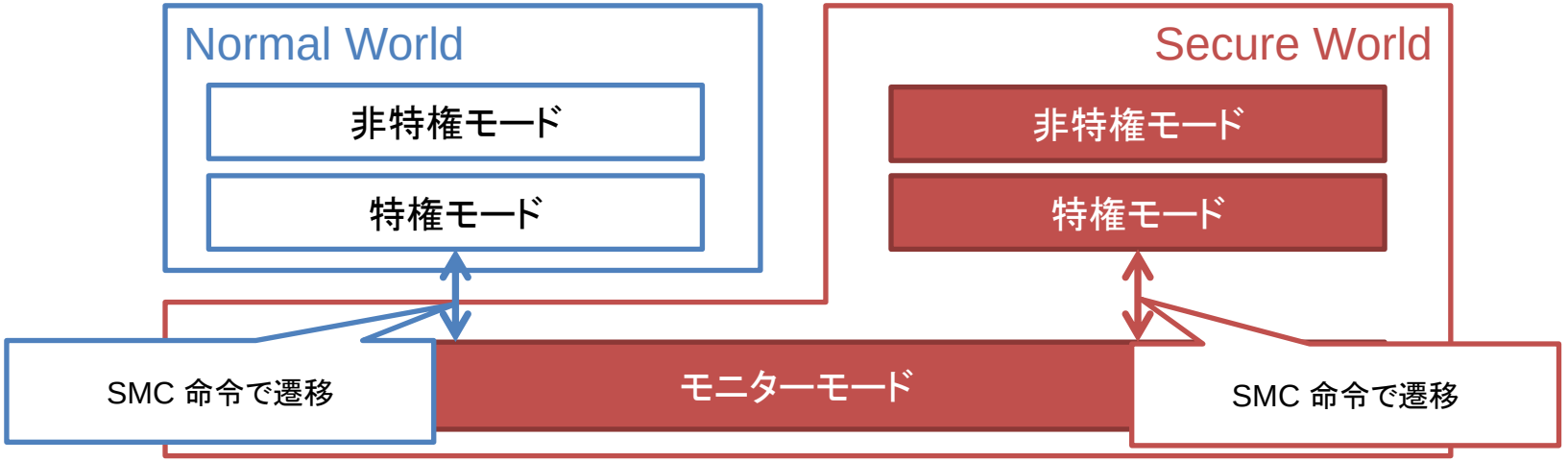
- ARM の実行環境を 2 つの world に分割し、
  秘匿性の高い情報を扱う Secure World からのみ必要な
  ハードウェア / 情報へアクセスできるようにする
  - TrustZone 仕様
    (準拠する仕様と対応する実装形態が複数存在する)
    - ソフトウェアによる論理的な分離
    - ハードウェア補助と ARM 拡張による一部物理的な隔離
  - Trusted Execution Environment [TEE]
    - Secure World 用のアプリケーションを作成するための
      共通 API と実行環境全体の標準仕様
    - TEE Client API (標準化仕様) や TrustZone API (ARM 提案) は
      Normal World と Secure World の間の通信を標準化する

TrustZone：ソフトウェア分離

- セキュア OS と、その上で動作するアプリケーション等の分離を行う技術としての TrustZone 準拠実装
  - 特権モードを Secure World の実行環境、非特権モードを Normal World の実行環境とみなす実装
  - タスクや CPU コアのひとつを Secure World の実行環境とみなす実装
- ソフトウェア的な分離を行うことで、データ分離を実施する
  - これは従来型の OS におけるデータ分離とそれほど変わらない
  - バスや割り込みなどは共有され、セキュア OS が world の分離を実施している
  - セキュア OS や Secure World 内プログラムのバグ、誤った DMA* などが world 分離を破ってしまうことが考えられる

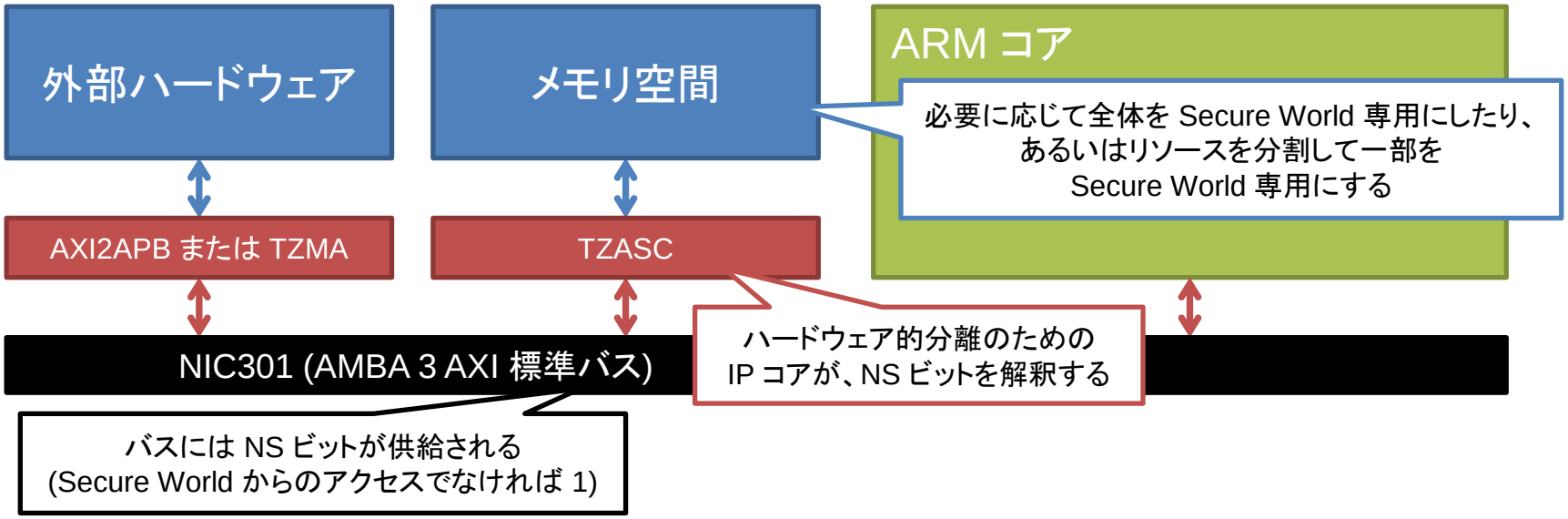
* Direct Memory Access : CPU を介さずハードウェア自身がメモリの読み書きを行う仕組み

TrustZone：ハードウェア分離（プロセッサ拡張）



- 今までの非特権モード、特権モードに加え、モニターモードを追加
  - 一種の（固定的な）仮想化機能による分離が提供される
- Secure World で動作しているか Normal World で動作しているかは ARM プロセッサ外から確認することができる
  - バスに対して NS（not secure）ビットという形で供給される
  - ハードウェアが自らアクセス制限を行うことが可能となる

TrustZone：ハードウェア分離（物理的な隔離）



- 各ハードウェアはこのバス信号を解釈することで、
  TrustZone で保護するべきハードウェアへの通信を制限する
  - このための数種類の IP コアが ARM より販売されている
  - ハードウェアを Secure World 専用にしたり、
    必要に応じてアドレス空間を World 別に分割することができる

TrustZone：ハードウェア分離（メリット / デメリット）

- メリット：通常の OS の上にレイヤーを追加することができる
  - 攻撃を受ける可能性が高い Linux などの通常 OS ではなく、形式的検証（機能の正当性の証明）が行われたセキュア OS を Secure World に置くことができる
- メリット：ハードウェア的に確実な分離が行われる
  - 一部サーバー環境で用いられている IOMMU のように、デバイスから直接メモリにアクセスできる状況であったとしても必要に応じてアクセスを禁止することができる
- デメリット：オーバーヘッドが大きい場合もある（※）
  - Normal World から Secure World へ遷移したい場合、一度モニターモードを介して Secure World の OS に移行する

※シングルスレッドの性能向上、かつマルチコアのうち1コアを専用に使うことで解決（2013/04/24追記）

TrustZone：モニター上で動作するソフトウェア

- TOPPERS Project / SafeG
  - Normal World 側 カーネル：Linux
  - Secure World 側 カーネル：TOPPERS/FMP
- Sierraware / Open Virtualization (SierraVisor)
  - Normal World 側 カーネル：Linux, BSD 系など
  - Secure World 側 カーネル：SierraTEE (TEE [後述] 準拠の独自 OS)
- また、Trustonic 社が商用製品を提供している
  - ARM、Gemalto、Giesecke & Devrient の合弁会社
  - Trusted Logic Mobility (元 Gemalto 社の一部門) を継承

攻撃の可能性 (1)

- Secure World で動作するのはあくまで “ソフトウェア” であるため、実装に脆弱性があれば攻撃することができる
  - Secure World のモニターへの攻撃
  - Secure World で動作するセキュアサービスへの攻撃
  - Secure World を動かすカーネルへの攻撃
- ただし、次の理由により攻撃を実施する、またそのための調査を行うことが非常に難しい
  - 特にモニターは機能が単純であり、形式的に脆弱性がないことが証明されている場合がある
  - Secure World で動作するプログラムの多くは単純な (脆弱性を作りこみにくい) 機能に特化する場合が多い
  - Secure World のリバースエンジニアリングが困難 (ROM へのアクセスが制限 / 暗号化されているなど)

攻撃の可能性 (2)

- それでも攻撃に成功した場合、深刻な情報漏洩や
  システムの乗っ取りなどにつながる可能性がある
  - 特にモニターの脆弱性によって権限昇格が成された場合、
    事実上ハイパーバイザー権限の rootkit が作成できる
  - そうでない場合も Secure World からアクセス可能な
    ハードウェアすべてにアクセス可能であるため、
    重要な情報や鍵が漏洩する可能性がある
  - それを防ぐためにも、Secure World 内で動作するプログラムが
    脆弱性を持たないことは非常に重要となる

TEE (Trusted Execution Environment)

- GlobalPlatform によって標準化された、認証された実行環境とそれに関わる API の仕様
  - これ自体は ARM TrustZone に依存していないが、TrustZone の Secure World 自体も TEE と呼ばれることがある
  - セキュリティプロセッサとその通信、アーキテクチャの標準化
- TEE System Architecture
  - TEE が前提とするアーキテクチャ仕様
- TEE Internal API
  - TEE 内で動作する C 言語で記述されたアプリケーション（セキュアサービス）のための API 仕様
- TEE Client API
  - TEE 内のアプリケーションと通信するための共通 API 仕様

TEE のメリットと今後の可能性

- TEE は、その環境内で動作する C 言語用の API を整備している
  (TEE Internal API)
  - つまり、TEE 準拠の環境であればある程度
    ソースコード互換のプログラムを作成することが可能
- Secure World の保護を行い続ける場合、TEE などの
  API を用いた設計の共有と実装の検証が重要になると予測する
  - 脆弱性の少ない設計のプログラムを作った上で検証し、
    それをセキュアハードウェア間で共有することが可能かもしれない

まとめ / 結論

- ARM TrustZone の一実装形態ではプロセッサ拡張との組み合わせで
  ハードウェアを含めたドメイン分離を実施することが可能である
- 数々の障害のために攻撃は難しくなっているが、
  それでもソフトウェアに脆弱性があれば攻撃の可能性は残る
  - TEE のような標準仕様と API をうまく用いることによって、
    実装に対する攻撃の可能性を最小限に減らすことが
    可能になると推測される
  - もちろん、セキュアサービス自体の設計にも気を使う必要がある

その他のセキュアハードウェアとその可能性

- Intel のセキュア実行のための技術
  - Intel Trusted Execution Technology (TXT)
  - Intel Virtualization Technology for Directed I/O (VT-d)
- Intel CE3100/4100

Intel のセキュア実行のための技術

- 信頼されたソフトウェアのみが実行される環境を実現するため、
  主に 2 つの技術を提供する
  - セキュアなハードウェアを提供する目的とは些か異なるが、
    信頼された実行環境を作る部分は共通している
  - 2 つの技術を組み合わせることで信頼された実行環境を提供できる
- Intel Trusted Execution Technology (TXT)
  - BIOS などのブート初期までに実行されるプログラムの
    電子署名を確認し、認証された (必要に応じて変更可能な)
    コードのみを実行する技術
- Intel Virtualization Technology for Directed I/O (VT-d)
  - ハードウェアからのメモリアクセス (DMA) を仮想化し、
    特権領域の上書きを防ぐことが可能な技術

Intel CE3100/CE4100

- 主にメディア処理を行う情報家電のための
  低消費電力版 x86 プロセッサとセキュリティ拡張を載せた SoC
  - Intel Media Processor CE3100 (Pentium M ベース)
  - Intel Atom Processor CE4100 (Atom ベース)
- セットトップボックスやスマート TV などへの利用を企図
  - ハードウェアによるビデオデコーダと専用ビデオ処理
  - 専用のセキュリティプロセッサ
  - HDCP によるバスの保護と、セキュリティ上のバス分離
- セキュリティ上の工夫により、秘匿性を要するメディアパスを
  分離し、不正なアクセスを防ぐことに成功している