セキュア暗号ユニット搭載
シングルチップマイクロコントローラ
プロテクションプロファイル

バージョン 1.20

2022 年 6 月 15 日

国立研究開発法人
産業技術総合研究所

Contents

更新履歴

バージョン	発行日	説明
1.20	2022 年 6 月 15 日	評価機関からの指摘に対する修正

謝辞

このプロテクションプロファイル（PP）は、大学、国立研究所、セキュリティサービスプロバイダ、評価機関の代表者から構成される SCU インサイドシステム委員会によって開発された。この PP の開発に直接貢献した組織は次のとおりである。

国立研究開発法人 産業技術総合研究所

国立大学法人 横浜国立大学

セコム株式会社

株式会社 トッパン・テクニカル・デザインセンター

エス・ティー・マイクロエレクトロニクス株式会社

電子商取引安全技術研究組合

株式会社 ECSEC Laboratory

0 はじめに

0.1 本書の目的

この文書は、セキュア暗号ユニット（Secure Cryptographic Unit: SCU）搭載シングルチップマイクロコントローラのセキュリティ機能要件（Security Functional Requirement: SFR）とセキュリティ保証要件（Security Assurance Requirement: SAR）を定義するプロテクションプロファイル（Protection Profile: PP）である。

0.2 参照文書

[180-4] FIPS PUB 180-4 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION Secure Hash Standard (SHS)

[186-4] FIPS PUB 186-4 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION Digital Signature Standard (DSS)

[202] FIPS PUB 202 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions

[800-38A] NIST Special Publication 800-38A Recommendation for Block Cipher Modes of Operation Methods and Techniques

[800-38B] NIST Special Publication 800-38B Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication

[800-38C] NIST Special Publication 800-38C Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality

[800-38D] NIST Special Publication 800-38D Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC

[800-38E] NIST Special Publication 800-38E Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices

[800-38F] NIST Special Publication 800-38F Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping

[800-90B] NIST Special Publication 800-90B Recommendation for the Entropy Sources Used for Random Bit Generation

[800-108] NIST Special Publication 800-108 Recommendation for Key Derivation Using Pseudorandom Functions (Revised)

[800-133] NIST Special Publication 800-133 Revision 1 Recommendation for Cryptographic Key Generation

[1619] IEEE 1619-2018 - IEEE Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices

[5639] RFC 5639 Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation

[8032] RFC 8032 Edwards-Curve Digital Signature Algorithm (EdDSA)

[8439] RFC 8439 ChaCha20 and Poly1305 for IETF Protocols

[9797-2] ISO/IEC 9797-2:2011 Information technology - Security techniques - Message Authentication Codes (MACs) - Part 2: Mechanisms using a dedicated hash-function

[10116] ISO/IEC 10116:2017 Information technology - Security techniques - Modes of operation for an n-bit block cipher

[10118-3] ISO/IEC 10118-3:2018 IT Security techniques - Hash-functions - Part 3: Dedicated hash-functions

[14888-3] ISO/IEC 14888-3:2018 IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms

[18031] ISO/IEC 18031:2011 Information technology - Security techniques - Random bit generation

[18033-3] ISO/IEC 18033-3:2010 Information technology - Security techniques - Encryption algorithms - Part 3: Block ciphers

[19772] ISO/IEC 19772:2009 Information technology - Security techniques - Authenticated encryption

[AAPS] Joint Interpretation Library, Application of Attack Potential for Smartcards and Similar Devices, Version 3.1, June 2020

[AMSS] Joint Interpretation Library, Application of Attack Potential for Smartcards and Similar Devices, Version 2.4, January 2020

[CC1J] 情報技術セキュリティ評価のためのコモンクライテリア、パート 1:概説と一般モデル、バージョン 3.1 改訂 5 版、2017 年 4 月、平成 29 年 7 月翻訳第 1.0 版

[CC2J] 情報技術セキュリティ評価のためのコモンクライテリア、パート 2:セキュリティ機能コンポーネント、バージョン 3.1 改訂 5 版、2017 年 4 月、平成 29 年 7 月翻訳第 1.0 版

[CC3J] 情報技術セキュリティ評価のためのコモンクライテリア、パート 2:セキュリティ保証コンポーネント、バージョン 3.1 改訂 5 版、2017 年 4 月、平成 29 年 7 月翻訳第 1.0 版

[CEMJ] 情報技術セキュリティ評価のための共通方法、評価方法、バージョン 3.1 改訂 5 版、2017 年 4 月、平成 29 年 7 月翻訳第 1.0 版

[CPPFDE] collaborative Protection Profile for Full Drive 1 Encryption - Encryption Engine Version 2.0 September 09, 2016

[PP0096] Common Criteria Protection Profile FIDO Universal Second Factor (U2F) Authenticator BSI-PP-CC-0096-V3-2018

[PPTEE] GlobalPlatform Device Committee TEE Protection Profile Version 1.2.1

0.3 用語

用語	説明
アプリケーションソフトウェア	TOE から見ると利用者データであり、ソフトウェアゲート API を介して SCU の暗号機能を利用する。
外部エンティティ	TOE の外部にあって TOE と対話できる人間あるいは IT のエンティティ。

用語	説明
ガベージコレクション	プログラムが動的に確保したメモリ領域のうち、不要になった領域を自動的に解放する機能。
サブマスク	ある方法で生成し、保存するビット列。
ソフトウェアゲート	暗号機能に対するアクセス制御メカニズムのソフトウェアパートであり、ハードウェアパートを介して暗号機能をアプリケーションソフトウェアに提供する。
中間鍵	データ暗号化鍵（Data Encryption Key: DEK）を保護するために使用する鍵暗号化鍵（Key Encryption Key: KEK）、鍵包み鍵（Key Wrapping Key: KWK）、MAC 鍵を意味する。
ハードウェアゲート	暗号機能に対するアクセス制御メカニズムのハードウェアパートであり、暗号エンジンにアクセスする。
ハードウェアゲート鍵	信頼の起点（RoT）となる鍵。
秘密	本 PP では、事前共有鍵、鍵材料、あらかじめ計算している値のような、機密性保護を必要とするデータを指す。
保護ストレージ	ハードウェアゲート鍵を格納する特殊なストレージ。半導体テストプロセスによって HGK を書込む OTP 領域であったり、ハードコードして回路の一部であったりする。
ユーザ鍵	アプリケーションが使用する鍵であり、鍵暗号化鍵により暗号化され、完全性検証用データが付与される。

0.4 略語

略語	説明
AES	Advanced Encryption Standard。
API	Application Program Interface、TOE 外部のソフトウェアから呼び出されるインタフェース。
CBC	Cipher Block Chaining。
CCM	Counter with CBC-MAC。
CMAC	Cipher-based Message Authentication Code。
CPU	Central Processing Unit、中央処理装置。
CTR	Counter、カウンターモードの略称。
DEK	Data Encryption Key、データ暗号化鍵。
DRBG	Deterministic Random Bit Generator、決定論的乱数ビット生成器。
ECC	Elliptic Curve Cryptography、楕円曲線暗号。
ECDSA	Elliptic Curve Digital Signature Algorithm、楕円曲線デジタル署名アルゴリズム。
EdDSA	Edwards-curve Digital Signature Algorithm。
GCM	Galois/Counter Mode。
FIPS	Federal Information Processing Standard(s)、連邦情報処理標準。

略語	説明
HGK	Hardware Gate Key、ハードウェアゲート鍵。
HMAC	Keyed-Hash Message Authentication Code。
IEEE	Institute of Electrical and Electronics Engineers。
IEC	International Electrotechnical Commission、国際電気標準会議。
IoT	Internet of Things。
IP	Intellectual Property、LSI を構成するための部分的な回路情報。
ISO	International Organization for Standardization、国際標準化機構。
IT	Information Technology。
IV	Initialization Vector、初期化ベクタ。
KDF	Key Derivation Functions、鍵導出関数。
KEK	Key Encryption Key、鍵暗号化鍵の略。
KW	Key Wrap、鍵包み。
KWK	Key Wrapping Key、鍵包み鍵の略。
KWP	Key Wrap with Padding
MAC	Message Authentication Code、メッセージ認証コード。
NIST	National Institute of Standards and Technology、アメリカ国立標準技術研究所。
OTP	One Time Programmable。
PP	Protection Profile、プロテクションプロファイル。
RBG	Random Bit Generator、乱数ビット生成器。
RoT	Root of Trust、信頼の起点。
SAR	Security Assurance Requirement、セキュリティ保証要件。
SCU	Secure Cryptographic Unit、セキュア暗号ユニット。
SFP	Security Function Policy、セキュリティ機能方針。
SFR	Security Functional Requirement、セキュリティ機能要件。
SHA	Secure Hash Algorithm。
SoC	System on a Chip、一個の半導体チップ上にシステムの動作に必要な機能の多く、あるいは全てを実装するという設計手法、また、その手法を使って作られたチップ。
TOE	Target of Evaluation、評価対象。
TSF	TOE Security Function、TOE セキュリティ機能。
XTS	XEX (XOR – Encrypt – XOR) based tweaked-codebook mode with cipher text stealing。

1　PP 概説

1.1　PP 参照

タイトル　　セキュア暗号ユニット搭載シングルチップマイクロコントローラ　プロテクションプロファイル

バージョン　　1.20

発行日　　2022 年 6 月 15 日

1.2　TOE 概要

1.2.1　TOE の構成

TOE は、セキュリティ機能を提供する SCU を搭載した組込み機器用途向けシングルチップマイクロコントローラ（マイコンと称す）である。SCU は、SCU 内に実装する暗号エンジンと、その暗号エンジンに対して、ソフトウェアゲート API を介してアクセスできるソフトウェアゲートとハードウェアゲートで構成される。TOE はシングルチップマイコン内にメモリを搭載したメモリ内蔵型を想定する。また、TOE は、SCU を含むシングルチップマイコンの外に大規模なメモリを有するメモリ外付け型が想定されるが、この型式は将来の課題とする。

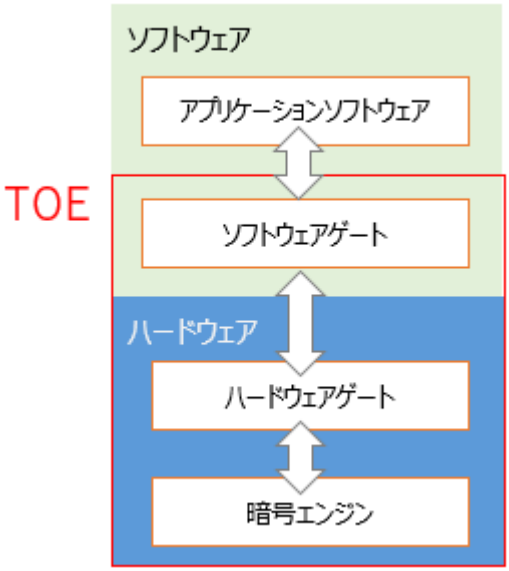


図 1-1　SCU の概念図

TOE は一般的に SoC で配付される。メモリ内蔵型 TOE の場合は、単一のダイに実装してパッケージにする場合がある。この SoC が、組込み機器の用途に必要なさまざまな回路を実装している基板にはんだ付けされ、組込み機器の筐体に収められる。

図 1-2 に TOE の構成例を示す。青線は TOE の物理的範囲を示し、SCU を搭載したマイコンの構成を示している。赤線は TOE の論理的境界を示す。図 1-2 では、アプリケーションソフトウェアは、論理的に TOE の外側に位置し、ソフトウェアゲートとハードウェアゲートを介して暗号機能を使用する。アプリケーションソフトウェアは TOE の物理的範囲の内部にある不揮発メモリに格納する。

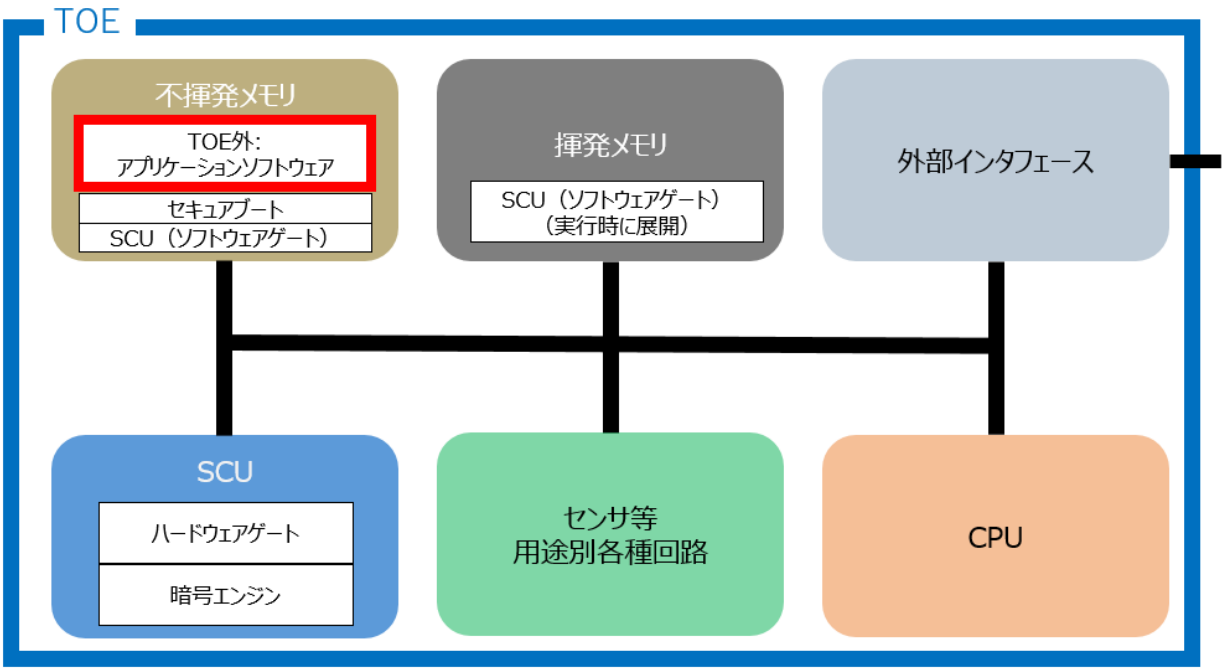


図 1-2　TOE の構成例

ソフトウェアゲートとハードウェアゲートは、暗号機能に対するアクセス制御のためのメカニズムであり、それぞれアクセス制御機能のソフトウェアパートとハードウェアパートを示す。アプリケーションソフトウェアによる暗号機能へのアクセスは、正当かつ許可されるべきアクセスであるが、それ以外のアクセスは拒否しなければならない。TOE のアクセス制御メカニズムは、この両者を区別するために実施される。

ソフトウェアゲートとハードウェアゲートは、次のように動作する。ハードウェアゲートは、自らに向けられた暗号操作コマンドデータを受け取り、そのたびに、ハードウェアゲートの内部状態を遷移させる。あるコマンドデータを受けたときの内部状態は、前の内部状態と現在の入力(暗号操作コマンドデータ)によって一意に決まる。もしハードウェアゲートが内部状態の正しい遷移データを知っていれば、現在の入力による内部状態の遷移結果を正しい遷移データと比較し、入力されたコマンドデータが正当なものかどうかを判定できる。

ソフトウェアゲートは、ハードウェアゲートの状態遷移を管理する情報を提供する。暗号機能にアクセスするすべてのパターンは、開発者があらかじめ予測した独自のものであり、開発者しか知らない。このアクセスパターンに伴うハードウェアゲートの内部状態遷移をあらかじめ計算し、ソフトウェアゲートに格納する。ソフトウェアゲートは、アプリケーションソフトウェアから暗号機能へのアクセスコマンドデータを受けると、そのアクセスコマンドデータとハードウェアゲートの次の内部状態遷移データを共にハードウェアゲートへ転送する。ハードウェアゲートは、渡された内部状態遷移データと、同時に受信したコマンドデータによる内部状態遷移結果と照合し、両者が一致すれば、受信したコマンドデータを正当なものと判断する。ハードウェアゲートへ渡す内部状態遷移データが格納されているソフトウェアゲートは、TOE 製造時に TOE 内の不揮発メモリに格納されるため、開発者以外は生成できない。つまり、開発者があらかじめ計算した暗号機能へのアクセスパターンは実行されるが、エミュレータなどがそれ以外のパターンで暗号機能を使用しようとしても、内部状態遷移データを提供できず、ハードウェアゲートによってアクセスを拒否される。

図 1-2 において、ソフトウェアゲート、ハードウェアゲートは、SCU の実装に依存する。SCU の外に位置するメモリのコンテンツの完全性、暗号機能の完全性を保護するためには、ソフトウェアゲートを適切に使用するようアプリケーションソフトウェアを開発する。

TOE は、SCU のセキュリティ機能を保護する自己保護機能を持つ。また、TOE は RoT となる HGK と外部エンティティが TOE を一意に識別する情報を SCU 内に格納する。TOE 開発者の責任において、十分なエントロピーを持つ乱数によって HGK を生成する。

TOE のセキュアブートプログラムは、起動時にソフトウェアゲートとアプリケーションソフトウェアを RAM に展開し、ソフトウェアゲートの完全性を検証し、アプリケーションソフトウェアの完全性（オプションで真正性）を検証する。また TOE は、アプリケーションソフトウェアのアップデート時に、アプリケーションソフトウェアの完全性と真正性を検証し、検証に成功したのちにアップデートする。ここで真正性とは、正当なアプリケーションソフトウェア開発者が開発したアプリケーションソフトウェア、という特性を言う。

1.2.2 TOE のセキュリティ機能

TOE は、SCU が担当し基盤となる暗号機能を、ソフトウェアゲートを介してアプリケーションソフトウェアに提供する。アプリケーションソフトウェアは暗号機能を利用した通信プロトコルやメモリ暗号化、識別認証といったセキュリティ機能を実装する。TOE は暗号機能と、暗号機能を利用したセキュリティ機能を保護する自己保護機能を実装する。

TOE は以下の主要なセキュリティ機能を持つ。この機能はベースライン機能であり、TOE の必須要件である。

- 暗号機能へのアクセス監視：ソフトウェアゲートとハードウェアゲートの協調により、暗号機能の不正な利用を検知して対応する機能
- 自己保護：SCU 動作中のノイズなど、漏洩ノイズの不正利用を妨げる機能と、物理的な攻撃を検知して対応する機能。
- セキュアブート：起動時にソフトウェアゲートとアプリケーションソフトウェアの完全性を検証する機能。
- 鍵の格納：鍵を SCU 外の TOE のメモリ領域に格納するため、暗号によって機密性と完全性を保護した鍵ストレージを提供する機能。
- ユーザ鍵のインポート：機密性を保護した状態で TOE の外部エンティティからユーザ鍵や秘密情報をインポートし、鍵ストレージに格納する機能。
- アップデート：アプリケーションソフトウェアの真正性と完全性を検証したのちにアップデートする機能。

TOE のベースライン機能を実現するための暗号機能は以下である。ST 作者は必要なセキュリティ機能要件を 7 章から選択する。

- 暗号化／復号：機密性を保護するため、平文を暗号化して暗号文にし、暗号文を復号して平文にする。
- デジタル署名検証：真正性と完全性検証のためにデジタル署名を検証する。
- ハッシュ値の計算：暗号学的ハッシュ関数によりハッシュ値を計算する。
- MAC の生成と検証：MAC を付与し、また MAC が付与されたデータの完全性を検証する。
- 乱数ビットの生成：RBG により乱数ビットを生成し、アプリケーションソフトウェアへ提供する。
- ソルト、ノンスの使用と IV の生成：暗号機能に必要なソルト、ノンスを適切に使用し、IV を生成する。
- 鍵の導出：鍵を導出する。

- 鍵の暗号化:鍵暗号化鍵を利用して鍵を暗号化する。

ソフトウェアゲートを介して TOE がアプリケーションソフトウェアに提供する暗号機能は以下である。この機能はオプション機能であり、ST 作者は必要なセキュリティ機能要件を 8 章から選択する。

- 鍵の生成:TOE が持つ乱数ビット生成器(RBG)により、用途と暗号アルゴリズムに適した鍵を生成する。
- 鍵および鍵材料の破棄:揮発メモリ上の鍵および鍵材料を復元不可能な状態にする。なお、不揮発メモリの鍵ストレージに格納されている鍵は暗号化されているため、破棄を想定していない。
- デジタル署名生成:真正性と完全性の保護のためにデジタル署名を生成する。

1.2.3 TOE のユースケース

SCU を搭載する TOE は、いわゆる IoT のエッジデバイスとなる、センサやアクチュエータ、監視カメラのような組込み機器のマイコンである。その組込み機器が収集した生データを TOE であるマイコンが処理し、処理した情報をセキュアに保管し、機器外部にセキュアに転送する機能をもつことが考えられる。SCU はそれらセキュアな処理の RoT となる。組込み機器製造者は、組込み機器のセキュリティ方針を定めて、機密性を保護する情報、完全性を保護する情報を定め、アプリケーションソフトウェアを実装する。

TOE は、アプリケーションソフトウェアの指示によって、TOE が持つサービスを用いて、アプリケーションソフトウェアが扱うデータの機密性、完全性、真正性を保護する。つまり、TOE から見ると、アプリケーションのどのデータを保護すべきか、判断できない。例えば、平文のままメモリに格納されたデータや、平文のまま外部に送信されたデータの機密性は保護できない。したがって TOE を利用してアプリケーションソフトウェアを開発する開発者は、保護したいデータを識別し、TOE のサービスを利用してデータを保護しなければならない。

1.2.4 役割

本 PP の主要な対象読者は TOE 開発者である。TOE 開発者は、SCU の IP を購入、あるいは自身で開発し、CPU やメモリなどの必要なコンポーネントを統合してシングルチップマイコンを開発・製造する。また、TOE 開発者は、TOE 製造時に HGK を SCU 内に保存する。TOE 製造は、TOE 開発者の工場で製造する場合もあれば、TOE 開発者が製造専門会社に委託する場合もある。

TOE の利用者は、TOE を購入し、TOE にアプリケーションソフトウェアを組込んで組込み機器を開発・製造する組込み機器開発者である。したがって、TOE のガイダンスは組込み機器開発者へ配付される。

TOE 開発者は、組込み機器を管理することはない。また、TOE 開発者は、HGK を組込み機器開発者へ通知する必要はない。ユーザ鍵は、組込み機器開発者に TOE を出荷する前にインポートすることを想定している。ユーザ鍵の更新のため、組込み機器開発者は、ユーザ鍵をインポートするためのソフトウェアゲート API の呼び出しをアプリケーションソフトウェアに実装することもできる。

最終利用者は組込み機器を購入し、所有して利用する。組込み機器の管理のため、TOE 外のアプリケーションソフトウェアを介した管理が発生するかもしれない。しかし TOE から見ると、アプリケーションソフトウェアの実装に依存するため、TOE の管理ではなく組込み機器の管理である。

1.2.5 TOE のライフサイクル

TOE 開発者は、フェーズ 1 の TOE 開発から、フェーズ 4 の TOE 製造までの工程と、フェーズ 6 の組込み機器開発者への TOE 配付をセキュアにしなければならない。また、フェーズ 5 のユーザ鍵書込みも TOE 開発者、または鍵インストールプロバイダがセキュアにしなければならない。

フェーズ 3 のアプリケーションソフトウェア開発、フェーズ 6 の組込み機器製造、フェーズ 7 の最終消費者への配付の環境は、本 PP の対象範囲外であるが、TOE を購入した組込み機器開発者が責任を持ちセキュアにすることを想定している。

フェーズ 1:ハードウェア開発

TOE の開発。TOE 開発者は、SCU の IP を購入し、または SCU を開発し、CPU などのコンポーネントと共に TOE ハードウェアを構築する。

フェーズ 2:ソフトウェア購入／開発

TOE 開発者は暗号エンジンをセキュアに使用するためのソフトウェアゲートを SCU IP ベンダから購入するか、あるいは開発する。セキュアブートプログラムは、SCU IP ベンダから購入するか、TOE 開発者が開発する。

フェーズ 3:アプリケーションソフトウェア開発

組込み機器開発者は、組込み機器の用途を実現するアプリケーションソフトウェアを開発する。TOE 製造者にアプリケーションソフトウェア搭載を依頼する場合は、TOE 製造者へアプリケーションソフトウェアを送付する。組込み機器開発者がアプリケーションソフトウェアを搭載する場合は、フェーズ 6 で搭載する。

フェーズ 4:TOE 製造

TOE 開発者は TOE を製造し、HGK を書込み、セキュアブートとソフトウェアゲートを TOE の不揮発メモリに搭載する。また、組込み機器開発者に依頼された場合、フェーズ 3 で組込み機器開発者が開発したアプリケーションソフトウェアを受け取り、TOE 内の不揮発メモリに搭載する。なお、アプリケーションソフトウェアは、フェーズ 6 で組込み機器開発者が TOE に搭載する場合もある。製造した TOE は、開発者テストを経て製品となる。

フェーズ 5:ユーザ鍵書込み

TOE 開発者は、ユーザ鍵や秘密情報を格納するデータオブジェクト(鍵ストレージ)を生成する。TOE 開発者は、組込み機器開発者からアプリケーションソフトウェアが使用するユーザ鍵や秘密情報を受け取り、鍵ストレージに格納し、鍵ストレージ全体を暗号化し MAC を付与する。ソフトウェアゲートを介してその鍵ストレージを TOE の不揮発メモリに書込む。このデータの書き込みは、TOE 開発者が鍵インストールプロバイダへ委託する場合もある。いずれにせよ、データの受け渡し、データの書込みはセキュアな環境で行われることが必要になる。データ書込み後、フェーズ 5 からフェーズ 6 へ、TOE を配付する。

フェーズ 6:組込み機器製造

組込み機器開発者は、TOE を基板へ搭載し、組込み機器を製造する。組込み機器開発者がアプリケーションソフトウェアを TOE に搭載する場合もある。この開発工程は、TOE を搭載する基板の開発、それを実装する組込み機器の開発の二つに分かれるかもしれない。この工程は TOE にとっては保護されている前提である。完成した組込み機器は、消費者へ配付される。

フェーズ 7:最終利用者による運用

TOE ライフサイクルの最終フェーズ。TOE が組込み機器に搭載されている状態で、想定する運用環境下で使用される。本 PP が想定する脅威は、この運用フェーズで発生する。

図 1-3　TOE のライフサイクル

1.2.6　鍵の保護

TOE は鍵格納サービスを提供し、暗号を使用することによって暗号エンジンで利用される暗号鍵の機密性を保ち安全に保管する。鍵の完全性は MAC によって保護する。一例として、ハードウェアゲートで利用さ

れるユーザ鍵が SCU 外部かつ TOE 内部の不揮発メモリに保持される際に、次のような処理を行う。まず、フェーズ 5 の TOE 開発者の工場、または委託先の鍵インストールプロバイダにおいて、SCU の内部に書き込まれる HGK を起点として、KEK と MAC 鍵を導出する。次に、TOE で利用される鍵ストレージ（ユーザ鍵、その他データが含まれる）を、KEK を用いて暗号化し、MAC 鍵を用いて、暗号化された鍵ストレージに対して MAC を付与する。このように、暗号化され、さらに MAC が付与された状態で、鍵ストレージは SCU の外部かつ TOE 内部のメモリに書き込まれる。

TOE には、製造時に生成されたデータオブジェクト（保護ストレージ）があり、TOE 開発者はそのデータオブジェクトに HGK を保管する。例えば、TOE 開発者が TOE 外部の乱数ビット生成器（RBG）によって HGK を生成して製造工程で TOE の中に埋め込む。HGK は、TSF の完全性を保証し、他のデータオブジェクト（鍵ストレージ）へのアクセス許可の基点になる。HGK は保護しなければならないため、TOE は自己保護機能が必要である。

2 適合主張

本 PP は CC バージョン 3.1 改訂第 5 版（日本語版）適合を主張する。

本 PP は CC パート 2 拡張を主張する。本 PP は CC パート 3 拡張を主張する。拡張するコンポーネントを 6 章に定義する。

本 PP は他の PP に適合していない。

本 PP において、TOE に対して適用する保証要件パッケージは EAL1 追加である。追加する保証コンポーネントは、ASE_SPD.1、ADV_ARC.1、ADV_FSP.2、ADV_TDS.1、ALC_FLR.1、AVA_VAN.2、AVA_SCU_EXT.1 である。AVA_SCU_EXT.1 は 6 章に定義する。

本 PP は ST または PP が本 PP に正確適合することを要求する。

この PP が満たす保証コンポーネントは APE_CCL.1, APE_ECD.1, APE_INT.1, APE_OBJ.1, APE_REQ.1, APE_SPD.1 である。

3 セキュリティ課題定義

3.1 資産

TOE は以下の資産 As を保護する。

As.SCU　　アプリケーションソフトウェアに提供するセキュリティサービスの完全性。

As.ConfUD　　機密性を必要とする利用者データの機密性。

As.IntegUD　　完全性を必要とする利用者データの完全性。

TOE は、TSF データの機密性をハードウェアの自己保護機能によって保護する。TOE は、不揮発メモリに格納する利用者データを TOE の暗号機能で保護する。

利用者データはユーザ鍵（データ署名生成のための秘密鍵、デバイス認証のための秘密鍵、アプリケーションソフトウェア検証用公開鍵、証明書検証用公開鍵など）、証明書、ソフトウェアゲート API を介して授受するデータ、そしてアプリケーションソフトウェアなどが含まれるだろう。TOE は、利用者データに必要な機密性、完全性を判断できない。したがって利用者が、機密性が必要な利用者データと、完全性を必要とする利用者データを識別し、TOE の機能を用いてセキュアに保存しなければならない。

TSF データは、HGK、鍵ストレージを復号する鍵、鍵ストレージの完全性を検証するデータ、ソフトウェアゲートの完全性を検証するためのデータ、IV、内部状態遷移データ、チップ ID である。

3.2 脅威

T.Internal_Access

攻撃者は、アプリケーションソフトウェアあるいはソフトウェアゲートを改ざんして、TOE の暗号機能を不正に使用するかもしれない。その結果、利用者データを暴露したり、改ざんしたりできるかもしれない。

根拠：以下のセキュリティ機能要件は、改ざんされたアプリケーションソフトウェアや改ざんされたソフトウェアゲートが TOE 内部のハードウェアゲートにアクセスし、TOE の暗号機能を使用するのを防止する。

- FDP_IFC.1/API, FDP_IFF.1/API は、外部エンティティであるアプリケーションソフトウェアによる暗号機能の操作要件を規定する。暗号機能へアクセスするソフトウェアゲートの内部状態遷移結果の正しさが検証された場合に限り、暗号機能から外部エンティティへ暗号機能の結果が出力される。すなわち、暗号機能の正しい利用だけが受け入れられる。
- FDP_MFW_EXT.1 は FPT_TST.1 から呼ばれ、起動時にアプリケーションソフトウェアの完全性を検証し、必要に応じて真正性を検証する。
- FPT_TST.1 は、起動時にソフトウェアゲートの完全性を検証することを定義し、FDP_IFC.1/API, FDP_IFF.1/API によるソフトウェアゲートの内部状態遷移データの検証を支援する。
- FPT_FLS.1/SG は、ソフトウェアゲートの内部状態遷移データの完全性が損なわれてもセキュアな状態を保持する。
- FPT_FLS.1/SB は、起動時にソフトウェアゲートの完全性、アプリケーションソフトウェアの完全性、必要に応じてアプリケーションソフトウェアの真正性が損なわれていてもセキュアな状態を保持する。

T.Weak_Import

攻撃者は、鍵ストレージへのインポート機能を悪用し、利用者データを暴露したり、改ざんしたりするかもしれない。

根拠：下記のセキュリティ機能要件により、あらかじめ暗号化され、完全性検証データを付与した鍵ストレージへのインポートと、完全性検証を行うことで、不正な利用者データのインポートを防止する。

- FDP_IFC.1/Import, FDP_IFF.1/Import は、外部エンティティであるアプリケーションソフトウェアによるインポート機能の操作要件を規定する。インポート機能へアクセスするソフトウェアゲートの内部状態遷移結果の正しさが検証された場合に限り、インポート機能を介して鍵ストレージへ格納される。すなわち、インポート機能の正しい利用だけが受け入れられる。
- FDP_UIT.1 は、鍵ストレージへインポートされる利用者データの完全性を検証する。

T.Unauthorized_Update

攻撃者は、組込み機器の利用者データを暴露したり、組込み機器のサービスを妨害したりするため、不正なアプリケーションソフトウェアを TOE にインストールするかもしれない。または、組込み機器の利用者データを暴露したり、組込み機器のサービスを妨害したりするため、セキュリティ機能の不具合があるバージョンに不正にロールバックするかもしれない。

根拠：下記のセキュリティ機能要件により、TSF は正しいアプリケーションソフトウェアのバージョンを得てアップデートを行い、アップデートしたアプリケーションソフトウェアを検証する。

- FPT_TUD_EXT.1 は、アプリケーションソフトウェアにアプリケーションソフトウェアの現在のバージョンを問い合わせ、アップデートを起動し、インストールの前にアップデートするアプリケーションソフトウェアを検証する。
- FPT_RPL.1 は、ロールバックの試みを防止する。
- FPT_FLS.1/UD は、アプリケーションソフトウェアの完全性エラーや真正性エラーの検出が起きてもセキュアな状態を保持する。

T.Weak_Crypto

攻撃者は、不適切に選択された鍵生成方法、暗号化アルゴリズム、鍵長、鍵破棄方法、または乱数ビット生成器を悪用することで、利用者データを暴露したり、改ざんしたりするかもしれない。

根拠：以下のセキュリティ機能要件に定義されている、承認された規格に基づく乱数ビット生成と十分なエントロピー源、十分な鍵長を持つ暗号アルゴリズムを実装し、アプリケーションソフトウェアに提供することで、脅威に対抗する。

- （オプション）FCS_CKM.1/AK は、非対称鍵を生成する。
- （オプション）FCS_CKM.1/SK は、対称鍵を生成する。
- （オプション）FCS_CKM.4 は、将来の回復を妨げるような方法で揮発メモリの鍵、鍵材料を破壊する。
- （選択）FCS_COP.1/SKC は、対称鍵アルゴリズムで暗号化・復号する。
- （選択）FCS_COP.1/KeyEnc は、鍵を暗号化・復号する。
- （選択）FCS_COP.1/Hash は、ハッシュを計算する。
- （選択）FCS_COP.1/MAC は、MAC を計算する。

- （オプション）FCS_COP.1/SigGen は、デジタル署名を行う。
- （選択）FCS_COP.1/SigVer は、デジタル署名を検証する。
- （選択）FCS_KDF_EXT.1 は、鍵導出を実行する。
- （選択）FCS_RBG_EXT.1 は、乱数ビットを生成する。
- （選択）FCS_SNI_EXT.1 は、TOE によって使用されるソルトとノンス、IV が鍵強度に悪影響を及ぼさないことを保証する。

T.Leak_Inherent

攻撃者は、暗号演算中の TOE の消費電力変化を観測し分析することによって、暗号鍵のような利用者データあるいは TSF データを暴露するかもしれない。

根拠：以下のセキュリティ機能要件は、SCU が利用者データ、TSF データを処理する際に生じる放射電磁波、消費電力に不要な情報が漏洩することを軽減し、攻撃者が統計処理による有益なデータの暴露を困難にする。

- FPT_EMS_EXT.1 は、TOE からのユーザデータ、TSF データの漏洩を軽減する。

T.Phys_Probing

攻撃者は、TOE 内部の物理的プロービングによって、暗号鍵のような TOE 内の利用者データ、あるいは他の攻撃に役立つ TSF データを暴露したり、改ざんしたりするかもしれない。

根拠：以下のセキュリティ機能要件は、半導体解析に用いる機器を使用して、メモリ素子を撮影したり、TOE の内部に物理的に接触したりして、利用者データを改ざん、あるいは取得したりする脅威に対抗することを目的とする。

- FPT_PHP.3 は、物理的プロービングに対抗する。
- FCS_STG_EXT.1 は、SCU 外に鍵ストレージを実装する。
- FCS_STG_EXT.2 は、暗号を使用して SCU 外にある鍵ストレージの機密性を保持する。
- FCS_STG_EXT.3 は、暗号を使用して SCU 外にある鍵ストレージの完全性を保持する。

T.Phys_Manipulation

攻撃者は、TOE 内部を物理的に操作することによって、そこに格納された利用者データや暗号鍵を改ざんしたり、あるいは他の攻撃のために TOE のセキュリティサービスを改ざんしたりするかもしれない。

根拠：下記のセキュリティ機能要件は、TOE の内部の物理的操作によって、直接的に情報資産を改ざんしたり、暴露するための他の攻撃の足がかりにしたりする脅威に対抗することを目的とする。

- FPT_PHP.3 は、TOE の物理的改ざんに対抗する。
- FCS_STG_EXT.1 は、SCU 外に鍵ストレージを実装する。
- FCS_STG_EXT.2 は、暗号を使用して SCU 外にある鍵ストレージの機密性を保持する。
- FCS_STG_EXT.3 は、暗号を使用して SCU 外にある鍵ストレージの完全性を保持する。

3.3　組織のセキュリティ方針

組織のセキュリティ方針はない。

3.4　前提条件

A.Trusted_User

組込み機器開発者は、TOE の外部に保持されているデータを適切に保護する。

運用環境のセキュリティ対策方針は、OE.Trusted_User である。

4 セキュリティ対策方針

本章では、APE_OBJ.1 に従い運用環境のセキュリティ対策方針のみ記載する。

4.1 運用環境のセキュリティ対策方針

OE.Trusted_User

組込み機器開発者は TOE の外部に保持されているデータの保護に関して提供されている TOE のガイダンスに従う。

根拠：組込み機器開発者が、TOE のガイダンスに従い、TOE 外部に保持するデータを保護するよう実装することによって、前提条件が満たされる。

5 セキュリティ要件

本 PP で使用するセキュリティ要件の表記法を次に示す。

(1) 変更のない要件:[CC2J]または拡張コンポーネント定義で使われている形式で記述している。

(2) PP 内での詳細化:追加した記述は**太字**で、削除した記述は~~取り消し線~~で示す。記述を置換すると**太字**の追加記述になる。

(3) 選択:

(ア) PP において完全、または部分的に完了:選択値(すなわち、PP で採用された選択値または ST で利用可能な残りの選択値)に下線付きの記述で示す。

例えば、拡張セキュリティ機能要件[選択:物理的、非物理的真性、決定論的、物理的ハイブリッド、決定論的ハイブリッド]では、PP 内において完全に完了している場合は[物理的]、部分的に完了している場合は[選択:物理的、非物理的真性]と示す。

(イ) 一部の SFR には、他の割付や選択を決定、または制限する選択を含む。これらの場合、表で表現し、表の各行を許可する選択肢のセットとして定義する。各行には、選択セットを示す一意の識別子を定義する。

(4) 完全な、あるいは部分的な割付:*斜体*で示す。

(5) PP において選択の中で完全に割付:*斜体下線付き*で示す。

例えば、[CC2J]または拡張セキュリティ機能要件[選択:デフォルト値変更、問い合わせ、改変、削除、[*割付: その他の操作*]]では、PP 内において選択も割付も完全に完了している場合は[デフォルト値変更、[*タグの選択*]]と示す。

(6) 繰返し:「/」で始まる文字列で示す。例えば、FCS_COP.1/Hash。

(7) 拡張した要件は、SFR または SAR 名称の末尾に「_EXT」をつけて示す。

5.1 セキュリティ機能要件

5.1.1 暗号サポート

FCS_STG_EXT.1 セキュア鍵ストレージ

下位階層:なし

依存性: なし

FCS_STG_EXT.1.1 TSF は、非対称プライベート鍵用と[選択:*対称鍵*、*秘密*、*その他の鍵なし*]用の[*ソフトウェアベース*]の鍵ストレージを提供しなければならない。

FCS_STG_EXT.1.2 TSF は、[*アプリケーションソフトウェア*]の要求に応じて、鍵／秘密を鍵ストレージにインポートする能力をサポートしなければならない。

FCS_STG_EXT.1.3 TSF は、[*アプリケーションソフトウェア*]の要求に応じて、安全な鍵ストレージ内の鍵／秘密を破壊することができなければならない。

適用上の注釈 1

TOE 内のメモリにソフトウェアで鍵ストレージが実装されているため、この要件でカバーする。

FCS_STG_EXT.2 鍵機密性保護

下位階層:なし

依存性: FCS_COP.1

FCS_STG_EXT.2.1 TSF は、[割付: *FCS_COP.1/KeyEnc で指定された暗号方法*]を使用して、[*すべてのソフトウェアベースの鍵ストレージ*]の機密性を保護しなければならない。

適用上の注釈 2

ST 作者は鍵ストレージを復号するための SFR を 7 章から選択すること。再暗号化する場合、暗号化・復号するための SFR を 7 章から選択すること。

FCS_STG_EXT.3 鍵完全性保護

下位階層:なし

依存性: FCS_COP.1

FCS_STG_EXT.3.1 TSF は、[*選択: 下記の完全性保護方法*]を使用して、[*すべてのソフトウェアベースの鍵ストレージ*]の完全性を保護しなければならない。

- *FCS_COP.1/Hash に従う保存されたハッシュ[選択: SHA-256, SHA-384, SHA-512, SHA3-224, SHA3-256, SHA3-384, SHA3-512] ;*
- *FCS_COP.1/MAC に従う MAC[選択: HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512, CMAC-AES-128, CMAC-AES-256] ;*
- *FCS_COP.1/SKC に従う対称暗号化[選択: AES_CCM, AES_GCM, AES_KW, AES_KWP] ;*
- *FCS_STG_EXT.2 に従って保護された非対称鍵を用いるFCS_COP.1/SigVer に従う保存された鍵のデジタル署名 ;*

適用上の注釈 3

ST 作者は鍵ストレージの完全性を保護するための SFR を 7 章から選択すること。

5.1.2 利用者データ保護

FDP_IFC.1/API サブセット情報フロー制御(API)

下位階層:なし

依存性: FDP_IFF.1 単純セキュリティ属性

FDP_IFC.1.1/API TSF は、[*サブジェクト:TOE の暗号機能、情報:サブジェクトへのソフトウェアゲート API 呼び出しと、サブジェクトからのソフトウェアゲート API レスポンス、操作:ソフトウェアゲート API 入力、暗号機能の実行とソフトウェアゲート API レスポンスの出力*]に対して[*組込み機器情報フロー制御 SFP*]を実施しなければならない。

FDP_IFC.1/Import サブセット情報フロー制御(インポート)

下位階層:なし

依存性: FDP_IFF.1 単純セキュリティ属性

FDP_IFC.1.1/Import TSF は、[*サブジェクト:TOE の不揮発メモリ、情報:鍵ストレージ、操作:インポート*]に対して[*インポート情報フロー制御 SFP*]を実施しなければならない。

FDP_IFF.1/API 単純セキュリティ属性(API)

下位階層:なし

依存性: FDP_IFC.1 サブセット情報フロー制御 FMT_MSA.3 静的属性初期化

FDP_IFF.1.1/API TSF は、以下のタイプのサブジェクト及び情報セキュリティ属性に基づいて、[*組込み機器情報フロー制御 SFP*]を実施しなければならない:[*サブジェクト:TOE の暗号機能、情報:サブジェクトへのソフトウェアゲート API 呼び出しと、サブジェクトからのソフトウェアゲート API レスポンス、サブジェクトのセキュリティ属性:あらかじめ埋め込まれたソフトウェアゲート API 呼び出しの完全性を検証するための内部状態遷移データ、情報のセキュリティ属性:ソフトウェアゲート API 呼び出しの完全性を検証する機能へ提供する内部状態の遷移結果*]

FDP_IFF.1.2/API TSF は、以下の規則が保持されていれば、制御された操作を通じて、制御されたサブジェクトと制御された情報間の情報フローを許可しなければならない: [

内部状態遷移データと TSF が計算した内部状態の遷移結果が一致し、ソフトウェアゲート API 呼び出しの完全性検証が成功したときのみ、サブジェクトへのソフトウェアゲート API 呼び出し/サブジェクトからのソフトウェアゲート API レスポンスの情報フロー、つまり暗号操作の出力を許可する。]

FDP_IFF.1.3/API TSF は、[*追加の情報フロー制御 SFP 規則:なし*]を実施しなければならない。

FDP_IFF.1.4/API TSF は、以下の規則、[*セキュリティ属性に基づいて情報フローを明示的に許可する規則:なし*]に基づいて、情報フローを明示的に許可しなければならない。

FDP_IFF.1.5/API TSF は、以下の規則、[*セキュリティ属性に基づいて情報フローを明示的に拒否する規則:なし*]に基づいて、情報フローを明示的に拒否しなければならない。

適用上の注釈 4

情報のセキュリティ属性は、ソフトウェアゲート API 呼び出しごとに計算される値であり、TSF が参照データとそれを比較することによって完全性を検証する。完全性を検証するための内部状態遷移の計算方法に暗号アルゴリズムを使用する場合、実装されたアルゴリズムを 7 章から選択する。

あらかじめ埋め込む内部状態遷移データを暗号によって保護する場合、実装されたアルゴリズムを 7 章から選択する。

FDP_IFF.1/Import 単純セキュリティ属性(インポート)

下位階層:なし

依存性: FDP_IFC.1 サブセット情報フロー制御 FMT_MSA.3 静的属性初期化

FDP_IFF.1.1/Import TSF は、以下のタイプのサブジェクト及び情報セキュリティ属性に基づいて、[*インポートフロー制御 SFP*]を実施しなければならない:[*サブジェクト:TOE の不揮発メモリ、情報:鍵ストレージ、サブジェクトのセキュリティ属性:なし、情報のセキュリティ属性:鍵ストレージの完全性検証に用いる値*]

FDP_IFF.1.2/Import TSF は、以下の規則が保持されていれば、制御された操作を通じて、制御されたサブジェクトと制御された情報間の情報フローを許可しなければならない: [

鍵ストレージの完全性検証に用いる値と TSF が計算した鍵ストレージの完全性を検証した値が一致したときのみ、サブジェクトへの鍵ストレージのインポートを許可する。]

FDP_IFF.1.3/Import TSF は、[*追加の情報フロー制御 SFP 規則:なし*]を実施しなければならない。

FDP_IFF.1.4/Import TSF は、以下の規則、[*セキュリティ属性に基づいて情報フローを明示的に許可する規則:なし*]に基づいて、情報フローを明示的に許可しなければならない。

FDP_IFF.1.5/Import TSF は、以下の規則、[*セキュリティ属性に基づいて情報フローを明示的に拒否する規則:なし*]に基づいて、情報フローを明示的に拒否しなければならない。

適用上の注釈 5

ST 作成者は、完全性の検証を実行するため、実装されたアルゴリズムを 7 章から選択する。選択されるのは、FCS_COP.1/Hash に従うハッシュ, FCS_COP.1/MAC に従う MAC, FCS_COP.1/SKC に従う認証付き暗号化である。

FDP_MFW_EXT.1 ソフトウェア完全性と真正性(セキュアブート)

下位階層:なし

依存性: FCS_COP.1

FDP_MFW_EXT.1.1 TSF は、**アプリケーションソフトウェア**の[完全性]を検証する能力を持たなければならない。

FDP_MFW_EXT.1.2 TSF は、**アプリケーションソフトウェア**の[完全性]の証拠を生成する能力を提供しなければならない。

適用上の注釈 6

TOE は、起動時にアプリケーションソフトウェアの完全性を検証する(セキュアブート)。ST 作成者は、完全性の検証を実行するため、実装されたアルゴリズムを 7 章から選択する。選択されるのは、FCS_COP.1/Hash

に従うハッシュ, FCS_COP.1/MAC に従うMAC, FCS_COP.1/SKC に従う認証付き暗号化, FCS_COP.1/SigVer に従うデジタル署名検証である。

FDP_UIT.1 データ交換完全性

下位階層: なし

依存性: [FDP_ACC.1 サブセットアクセス制御、または FDP_IFC.1 サブセット情報フロー制御] [FTP_ITC.1 TSF 間高信頼チャネル、または FTP_TRP.1 高信頼パス]

FDP_UIT.1.1 TSF は、利用者データを[*改変*]誤りから保護した形で[*受信*]を行うために、[*インポート情報フロー制御 SFP*]を実施しなければならない。

FDP_UIT.1.2 TSF は、利用者データ受信において、[*改変*]が生じたかどうかを判定できなければならない。

適用上の注釈 7

この要件は、インポートしたユーザ鍵の完全性を検証することを目的としている。ST 作成者は、完全性の検証を実行するため、実装されたアルゴリズムを 7 章から選択する。選択されるのは、FCS_COP.1/Hash に従うハッシュ, FCS_COP.1/MAC に従うMAC, FCS_COP.1/SKC に従う認証付き暗号化である。

5.1.3 TSF の保護

FPT_EMS_EXT.1 TOE 漏洩軽減

下位階層: なし

依存性: なし

FPT_EMS_EXT.1.1 TOE は、[*以下の TSF データのタイプのリスト*]および[*以下のユーザデータのタイプのリスト*]へのアクセスを可能にする [*TOE の電源ライン、TOE の外装*]への[*消費電力、放射電磁波*]の漏洩を軽減しなければならない。

表 5-1 サイドチャネル攻撃から保護するデータ

TSF データのタイプのリスト	ユーザデータのタイプのリスト
鍵ストレージの暗号鍵 *[割付:その他の TSF データ]*	*割付:ユーザデータのタイプのリスト*

FPT_FLS.1/SB セキュアな状態を保持する障害(セキュアブート)

下位階層: なし

依存性: なし

FPT_FLS.1.1/SB TSF は、以下の種別の障害が生じたときはセキュアな状態を保持しなくてはならない。:[*起動時のソフトウェアゲート完全性違反*、*起動時のアプリケーションソフトウェアの完全性違反*、**[選択:** ***起動時のアプリケーションソフトウェアの真正性違反*****、[割付:その他の違反**]]]。

適用上の注釈 8

このSFR は、FPT_TST.1 によるソフトウェアゲートとアプリケーションソフトウェアの検証に失敗したとき、セキュアな状態を維持する。

FPT_FLS.1/SG セキュアな状態を保持する障害(ソフトウェアゲート)

下位階層: なし

依存性: なし

FPT_FLS.1.1/SG TSF は、以下の種別の障害が生じたときはセキュアな状態を保持しなくてはならない。:[*FPT_TST.1 で実施される内部状態遷移データの完全性検証の失敗*]。

適用上の注釈 9

このSFR は、FPT_TST.1 により内部状態遷移の計算結果と、あらかじめ埋め込まれている内部状態遷移データの不一致が起きた場合をカバーする。

FPT_FLS.1/UD セキュアな状態を保持する障害(高信頼アップデート)

下位階層: なし

依存性: なし

FPT_FLS.1.1/UD TSF は、次のタイプの障害が発生したとき、セキュアな状態を保持しなければならない:[*アップデート用アプリケーションソフトウェアの真正性違反*、*完全性違反*]。

適用上の注釈 10

この要件は、アップデートするアプリケーションソフトウェアの完全性と真正性の検証に失敗したとき、セキュアな状態を維持する。

FPT_PHP.3 物理的攻撃への対抗

下位階層: なし

依存性: なし

FPT_PHP.3.1 TSF は、SFR が常に実施されるよう自動的に対応することによって、[*TSF*]への[*物理的な改ざん操作*、*物理的なプロービング*]に抵抗しなければならない。

詳細化:TSF は、物理的改ざん操作と物理的プロービングに継続的に対抗するための適切なメカニズムを実装する。この攻撃の性質(特に改ざん操作)により、TSF はすべての部品に対する攻撃を検出できない。し

たがって、セキュリティ機能要件が実施されていることを保証する、この攻撃に対する切れ目ない保護が必要である。ここでの「自動応答」とは、(i)いつでも攻撃を受ける可能性があると想定し、(ii)いつでも対策を講じることを意味する。

適用上の注釈 11

ST 作者はセキュリティターゲットに TOE の自動応答を記述すること。物理的な改ざん操作または物理的なプロービングを検出し、別の方法でセキュリティを確保できない場合、TOE が動作を停止するか動作させなくすると、セキュリティ機能要件が実施される。

FPT_RPL.1 リプレイ検出 (ロールバック)

下位階層: なし

依存性: なし

FPT_RPL.1.1 TSF は、以下のエンティティに対するリプレイを検出しなければならない:[*過去のバージョンのアプリケーションソフトウェア*]。

FPT_RPL.1.2 TSF は、リプレイが検出された場合、[*ロードされたアプリケーションソフトウェアの実行を防止し*、[選択:[*割付:他のアクション*]、他のアクションなし:から 1 つのみ選択]]する。

適用上の注釈 12

アプリケーションソフトウェアのロードが要求されると、TSF は検証されたバージョンのアプリケーションソフトウェアが以前に認証されたアプリケーションソフトウェアのバージョン以上であることを保証する。過去のバージョンのアプリケーションソフトウェアをロードすることで、既知の脆弱性にさらす危険性がある。

FPT_TST.1 TSF テスト

下位階層: なし

依存性: なし

FPT_TST.1.1 TSF は、[*暗号機能*]の正常動作を実証するために、**初期立ち上げ中**、[選択:*通常運用中定期的に*、***アプリケーションソフトウェア****の要求時に*、*条件[割付: 自己テストが作動すべき条件] 下で*]自己テストのスイートを実行しなければならない。

FPT_TST.1.2 TSF は、**ハードウェアゲート**に、[*内部状態遷移結果*]の完全性を検証する能力を提供しなければならない。

FPT_TST.1.3 TSF は、**初期立上げ中の自己テスト**に、[*ソフトウェアゲート*]の完全性を検証する能力を提供しなければならない。

適用上の注釈 13

この要件は、内部状態遷移結果と内部状態遷移データの一致により、状態遷移の完全性を検証し、その結果ソフトウェアゲート自身の検証が実施されることを目的としている。初期立上げ中の自己テストは、ソフトウェアゲートの完全性、アプリケーションソフトウェアの完全性、オプションで真正性を検証する(セキュアブー

ト)。ソフトウェアゲート検証手段のアルゴリズムを7 章から選択する。選択されるのは、FCS_COP.1/Hash に従う保存されたハッシュ, FCS_COP.1/MAC に従う MAC, FCS_COP.1/SKC に従う認証付き暗号化, FCS_COP.1/SigVer に従うデジタル署名検証である。

FPT_TUD_EXT.1　高信頼アップデート

下位階層: なし

依存性: FCS_COP.1

FPT_TUD_EXT.1.1 TSF は、**アプリケーションソフトウェア**の現在のバージョンを問い合わせる能力を[*ソフトウェゲー内*]に提供しなければならない

FPT_TUD_EXT.1.2 TSF は、**アプリケーションソフトウェア**のアップデートを開始する能力を[*ソフトウェアゲー内*]に提供しなければならない。

FPT_TUD_EXT.1.3 TSF は、**アプリケーションソフトウェア**のアップデートを製造者による[選択: *FCS_COP.1/SigVer のデジタル署名、FCS_COP.1/MAC に従う MAC*]を用いて、インストール前に検証しなければならない。

適用上の注釈 14

この要件は TOE がアプリケーションソフトウェアを更新する機能を提供する。TOE は、アップデート時、アップデートするアプリケーションソフトウェアを検証することによってアプリケーションソフトウェアの完全性と真正性を保証する。ST 作成者は、完全性検証、真正性検証を実行するために実装するアルゴリズムを7 章から選択する。

5.2 セキュリティ保証要件

この PP にて定義するセキュリティ保証要件は、EAL 1 + ASE_SPD.1 + ADV_ARC.1 + ADV_FSP.2 + ADV_TDS.1 + ALC_FLR.1 + AVA_VAN.2 + AVA_SCU_EXT.1 である。

5.3 セキュリティ要件根拠

5.3.1 セキュリティ機能要件根拠

表 5-2 にベースラインのセキュリティ機能要件において満たされる依存性を示す。

適用上の注釈 15

選択セキュリティ機能要件から必要なセキュリティ要件を選択したとき、そのセキュリティ機能要件の依存性を満たすように全てのセキュリティ機能要件を選択すること。

オプションベースのセキュリティ機能要件を追加した場合、ST 作者は満たされる依存性を示し、満たされない依存性はその根拠を記述すること。

表 5-2 ベースラインのセキュリティ機能要件依存性

機能要件	CC における依存性	PP において満たしている依存性
FCS_STG_EXT.1	依存性なし	なし
FCS_STG_EXT.2	FCS_COP.1	注:ST 作者が選択。
FCS_STG_EXT.3	FCS_COP.1	注:ST 作者が選択。
FDP_IFC.1/API	FDP_IFF.1	FDP_IFF.1/API
FDP_IFC.1/Import	FDP_IFF.1	FDP_IFF.1/Import
FDP_IFF.1/API	FDP_IFC.1 and FMT_MSA.3	FDP_IFC.1/API 注:FMT_MSA.3 は満たされない。
FDP_IFF.1/Import	FDP_IFC.1 and FMT_MSA.3	FDP_IFC.1/Import 注:FMT_MSA.3 は満たされない。
FDP_MFW_EXT.1	FCS_COP.1	注:ST 作者が選択。
FDP_UIT.1	[FDP_ACC.1 or FDP_IFC.1] and [FTP_ITC.1 or FTP_TRP.1]	FDP_IFC.1/Import 注:FTP_ITC.1 or FTP_TRP.1 は満たされない
FPT_EMS_EXT.1	依存性なし	なし
FPT_FLS.1/SB	依存性なし	なし
FPT_FLS.1/SG	依存性なし	なし
FPT_FLS.1/UD	依存性なし	なし
FPT_PHP.3	依存性なし	なし
FPT_RPL.1	依存性なし	なし
FPT_TST.1	依存性なし	なし
FPT_TUD_EXT.1	FCS_COP.1	注:ST 作者が選択。

FCS_STG_EXT.2、FCS_STG_EXT.3、FDP_MFW_EXT.1、FPT_TUD_EXT.1 の依存性 FCS_COP.1 は、ST 作者が使用している暗号機能要件をオプション要件から選択して記入する。

FDP_IFF.1/API の依存性 FMT_MSA.3 は満たされない。FMT_MSA.3 は、「情報のセキュリティ属性」の管理を規定する。本 TOE では、この属性は、API 呼び出しのたびに計算される内部状態遷移の結果であり、TOE の管理対象でない。そのため、FMT_MSA.3 は適用されない。

FDP_IFF.1/Import の依存性 FMT_MSA.3 は満たされない。FMT_MSA.3 は、「情報のセキュリティ属性」の管理を規定する。本 TOE では、この属性は、TOE の外部で計算される値であり、TOE の管理対象でない。そのため、FMT_MSA.3 は適用されない。

FDP_UIT.1 の依存性 FTP_ITC.1 または FTP_TRP.1 は満たされない。TOE 外部で暗号化され MAC 付与された利用者データをインポートするので高信頼パス／高信頼チャネルを適用しない。

5.3.2 セキュリティ保証要件根拠

TOE に適用するセキュリティ保証要件は、EAL1 + ASE_SPD.1 + ADV_ARC.1 + ADV_FSP.2 + ADV_TDS.1 + ALC_FLR.1 + AVA_VAN.2+ AVA_SCU_EXT.1 である。拡張セキュリティ保証要件 AVA_SCU_EXT.1 は 6 章に定義する。

これらセキュリティ保証要件の選択の決定は、TOE が扱う資産の重要性と TOE が使用される運用環境を考慮し、基本的な攻撃能力を持つ攻撃者の攻撃に耐えうること（すなわち AVA_VAN.1 または AVA_VAN.2）から導きだされた。次に、TOE は安価なデバイスに用いられるという、TOE の特徴を考慮し、開発者に求められる評価認証に必要なコスト、時間を削減しつつ、適切な保証を得られるレベルを考えた。

一方で、基本的な攻撃能力を持つ攻撃者の攻撃に耐えうること（すなわち AVA_VAN.1 または AVA_VAN.2）から、より高い AVA_VAN.2 を選択し、その依存性から ADV_ARC.1、ADV_FSP.2、ADV_TDS.1 を選択している。これにより、公知の脆弱性の探索に加えて、提供された評価用証拠資料に基づく脆弱性分析、TOE 仕様に基づく独立テストを実施することにより、EAL1 からの有意義な保証の増加を得ることができる。

また、近年の半導体開発、組込み機器開発において構成管理システムや量産システム、物流システムを使わないことはありえず、この観点から ALC クラスは EAL1 に含まれる最低限のものになっている。さらに、ALC_FLR.1 を加え、発見されたセキュリティ瑕疵が開発者によって追跡され訂正されることにより、TOE が将来にわたり維持されることを期待している。このほか、PP が想定しているユースケース、それに伴う脅威、前提条件について、読者の理解を助けるため、ASE_SPD.1 を追加している。

AVA_SCU_EXT.1 SCU 脆弱性調査について

TOE は暗号エンジン、ハードウェアゲートを、ソフトウェアゲートを介して制御するマイコンである。マイコンであるが、本 TOE はスマートカード用マイコンのように高い価値のある資産を扱わず、扱う資産価値は低いと想定している。そのため、本 TOE で想定する攻撃者は、スマートカードマイコンを攻撃する攻撃者より、低い攻撃能力を持つと想定する。

ここで、コモンクライテリアによる評価ではスマートカードと類似デバイスの脆弱性評定に[AAPS]を適用することが必須となっている。そのため、[PPTEE]を参考にして新たに拡張保証コンポーネント AVA_SCU_EXT を定義し、SCU を組み込む TOE 開発者が留意すべき、低い攻撃能力を持つ攻撃者が実行する攻撃と、SOG-IS サポート文書[AAPS]を参考にしたハードウェア攻撃の攻撃能力の計算表を 9 章に定義する。

表 5-3 に本 TOE のセキュリティ保証要件において満たされる依存性を示す。このように選択したセキュリティ保証要件の依存性は全て満たされている。

表 5-3 セキュリティ保証要件依存性

保証要件	CC における依存性	PP において依存性を満たす保証要件
ASE_CCL.1	ASE_ECD.1, ASE_INT.1, ASE_REQ.1	ASE_ECD.1, ASE_INT.1, ASE_REQ.1
ASE_ECD.1	依存性なし	なし
ASE_INT.1	依存性なし	なし
ASE_OBJ.1	依存性なし	なし
ASE_REQ.1	ASE_ECD.1	ASE_ECD.1
ASE_SPD.1	依存性なし	なし

保証要件	CC における依存性	PP において依存性を満たす保証要件
ASE_TSS.1	ASE_INT.1, ASE_REQ.1, ADV_FSP.1	ADV_FSP.2
ADV_ARC.1	ADV_FSP.1, ADV_TDS.1	ADV_FSP.2, ADV_TDS.1
ADV_FSP.2	ADV_TDS.1	ADV_TDS.1
ADV_TDS.1	ADV_FSP.2	ADV_FSP.2
AGD_OPE.1	ADV_FSP.1	ADV_FSP.2
AGD_PRE.1	依存性なし	なし
ALC_CMC.1	ALC_CMS.1	ALC_CMS.1
ALC_CMS.1	依存性なし	なし
ALC_FLR.1	依存性なし	なし
ATE_IND.1	ADV_FSP.1, AGD_OPE.1, AGD_PRE.1	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1
AVA_VAN.2	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1
AVA_SCU_EXT.1	AVA_VAN.1	AVA_VAN.2

6 付録:拡張コンポーネント定義

6.1 拡張セキュリティ機能コンポーネント

本 PP で定義する拡張コンポーネントとそれを含むファミリを以下に示す。これらは、CC パート 2 のファミリ及びコンポーネントをモデルとして構成し、[PP0096]と[CPPFDE]を参考にした。

6.1.1 **FCS_KDF_EXT 暗号鍵導出**

CC にはサブマスクから中間鍵を導出する SFR が定義されていないため必要である。

ファミリのふるまい

本ファミリは、中間鍵が規定されたセットのサブマスクから導出される手段を規定する。

コンポーネントのレベル付け

FCS_KDF_EXT.1 暗号鍵導出は、TSF に、規定された鍵導出関数を用いてサブマスクから中間鍵を導出することを要求する。

管理:FCS_KDF_EXT.1

特定の管理アクションは識別されていない。

監査:FCS_KDF_EXT.1

予見される監査対象事象はない。

FCS_KDF_EXT.1 暗号鍵導出

下位階層: なし

依存性: FCS_CKM.4 暗号鍵破棄

FCS_COP.1/MAC 暗号操作(メッセージ認証)

FCS_RBG_EXT.1 乱数ビット生成

FCS_KDF_EXT.1.1 TSF は、[選択:*FCS_RBG_EXT.1 で規定される乱数生成されたサブマスク、インポートされたサブマスク*]を[選択:*NIST SP 800-108* [*選択:カウンターモードを用いた KDF、フィードバックモードを用いた KDF、ダブルパイプライン繰り返しモードを用いた KDF]*、*NIST SP 800-132*] で定義されるとおり、その出力が少なくとも HGK と等しいセキュリティ強度(ビット数で)になるように、FCS_COP.1/MAC で規定される鍵付ハッシュ関数を用いて、中間鍵を導出しなければならない。

6.1.2 FCS_RBG_EXT 乱数ビット生成

CC にはランダムビット生成の SFR が定義されていないため必要である。

ファミリのふるまい

本ファミリのコンポーネントは、乱数ビット／乱数の生成についての要件に対処する。これは FCS クラスとして定義された新しいファミリである。

コンポーネントのレベル付け

FCS_RBG_EXT.1 乱数ビット生成は、乱数ビット生成に、選択された規格に従って実行され、エントロピー源によってシード値を供給されることを要求する。

管理:FCS_RBG_EXT.1

特定の管理機能は識別されていない。

監査:FCS_RBG_EXT.1

予見される監査対象事象はない。

FCS_RBG_EXT.1 乱数ビット生成

下位階層: なし

依存性: なし

FCS_RBG_EXT.1.1 TSF は、ISO/IEC 18031:2011 に従い、[選択: *Hash_DRBG (any)*、*HMAC_DRBG (any)*、*CTR_DRBG (AES)*] を用いて、すべての決定論的乱数ビット生成サービスを実行しなければならない。

FCS_RBG_EXT.1.2 決定論的 RBG は、[選択:

- *[割付:ソフトウェアベースのノイズ源の数] 個のソフトウェアベースのノイズ源*、
- *[割付:ハードウェアベースのノイズ源の数] 個のハードウェアベースのノイズ源]*

からエントロピーを蓄積するような、少なくとも 1 つのエントロピー源によってシード値を与えられなければならない。ここで、ノイズ源については、ISO/IEC 18031:2011 Table C.1「Security Strength Table for Hash Functions」に従い、生成する鍵やハッシュの最大セキュリティ強度と少なくとも等しく、かつ最小でも[選択:*128 ビット*、*192 ビット*、*256 ビット*]のエントロピーを蓄積する NIST SP 800-90B に従ったエントロピー源によってシードを供給されなければならない。

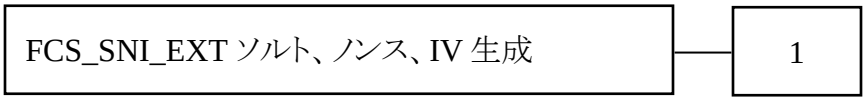
6.1.3 FCS_SNI_EXT ソルト、ノンス、及び IV 生成

CC にはソルト、ノンス、および IV の生成に関する SFR が定義されていないため必要である。

ファミリのふるまい

本ファミリは、ソルト、ノンス、及び IV が適格であることを保証する。

コンポーネントのレベル付け



FCS_SNI_EXT.1 暗号操作(ソルト、ノンス、及び IV 生成)は、ソルト、ノンス、及び IV の生成に、規定されたやり方で実行されるべき、TOE の暗号コンポーネントによって使用されることを要求する。

管理:FCS_SNI_EXT.1

特定の管理機能は識別されていない。

監査:FCS_SNI_EXT.1

予見される監査対象事象はない。

FCS_SNI_EXT.1 ソルト、ノンス、及び IV 生成

下位階層: なし

依存性: FCS_RBG_EXT.1 乱数ビット生成

FCS_SNI_EXT.1.1 TSF は、[選択:*ソルトを利用しない*、*[選択:FCS_RBG_EXT.1 で規定される DRBG*、*製造工程で TOE 外から提供されるサブマスク] によって生成されるソルトを使用する*]ようにしなければならない。

FCS_SNI_EXT.1.2 TSF は、[選択:*ノンスを利用しない*、*最小[割付:ノンスのサイズ]ビット長の一意のノンスを使用する*]ようにしなければならない。

FCS_SNI_EXT.1.3 TSF は、以下のやり方で IV を生成しなければならない[選択:

- *CBC: IV は、繰り返してはならない*、
- *CCM: ノンスは、繰り返してはならない*、
- *XTS: IV はなし。調整値(Tweak)は、非負の整数であり、連続に割り当てられ、かつ任意の非負の整数から始まらなければならない*、
- *GCM: IV は、繰り返してはならない。与えられた秘密鍵について GCM の呼び出し回数は 2^{32} 回を超えてはならない*]。

6.1.4 FCS_STG_EXT セキュア鍵ストレージ

CC には暗号用の鍵を格納する要件が定義されていないため必要である。

ファミリのふるまい

本ファミリは、SCU 外で TOE 内にセキュアに鍵を格納するための仕様を提供する。

コンポーネントのレベル付け

FCS_STG_EXT.1, セキュア鍵ストレージは、TSF が鍵ストレージを保持し、そのストレージの特性を指定することを要求する。

FCS_STG_EXT.2, 鍵機密性保護は、TSF が指定された方法で保存されたデータの機密性を保護することを要求する。

FCS_STG_EXT.3, 鍵完全性保護は、TSF が指定された方法で保存されたデータの完全性を保護することを要求する。

管理:FCS_STG_EXT.1、FCS_STG_EXT.2、FCS_STG_EXT.3

特定の管理機能は識別されていない。

監査:FCS_STG_EXT.1、FCS_STG_EXT.2、FCS_STG_EXT.3

予見される監査対象事象はない。

FCS_STG_EXT.1 セキュア鍵ストレージ

下位階層: なし

依存性: なし

FCS_STG_EXT.1.1 TSF は、非対称プライベート鍵用と[選択:*対称鍵*、*秘密*、*その他の鍵なし*]用の[選択:*変更不可能なハードウェアベース*、*変更可能なハードウェアベース*、*ソフトウェアベース*]の鍵ストレージを提供しなければならない。

FCS_STG_EXT.1.2 TSF は、[割付:*許可されたサブジェクト*]の要求に応じて、鍵／秘密を鍵ストレージにインポートする能力をサポートしなければならない。

FCS_STG_EXT.1.3 TSF は、[割付:*許可されたサブジェクト*]の要求に応じて、安全な鍵ストレージ内の鍵／秘密を破壊することができなければならない。

FCS_STG_EXT.2 鍵機密性保護

下位階層: なし

依存性: FCS_COP.1

FCS_STG_EXT.2.1 TSF は、[割付:*FCS_COP.1 で指定された暗号方法*]を使用して、[割付:*鍵データ*]を暗号化しなければならない。

FCS_STG_EXT.3 鍵完全性保護

下位階層: なし

依存性: FCS_COP.1

FCS_STG_EXT.3.1 TSF は、[割付: *完全性保護方法*]を使用して、[割付: *鍵データ*]の完全性を保護しなければならない。

6.1.5 FDP_MFW_EXT ソフトウェアの完全性と真正性

CC の利用者データを保護する要件は広範に適用できるが、特にソフトウェアの完全性と真正性の検証を必要とする場合の要件が定義されていないため必要である。

ファミリのふるまい

本ファミリは、セキュアブートと高信頼アップデートのためにソフトウェアの完全性と真正性検証についての要件に対処する。

コンポーネントのレベル付け

FDP_MFW_EXT ソフトウェア完全性と真正性	1

FDP_MFW_EXT.1 ソフトウェア完全性と真正性は、TSF がソフトウェアの完全性、または真正性、あるいはその両方を検証することを要求する。

管理: FDP_MFW_EXT.1

特定の管理機能は識別されていない。

監査: FDP_MFW_EXT.1

予見される監査対象事象はない。

FDP_MFW_EXT.1 ソフトウェア完全性と真正性

下位階層: なし

依存性: FCS_COP.1

FDP_MFW_EXT.1.1 TSF は、ソフトウェアの[選択: *完全性*、*真正性*]を検証する能力を持たなければならない。

FDP_MFW_EXT.1.2 TSF は、ソフトウェアの[選択: *完全性*、*真正性*]の証拠を生成する能力を提供しなければならない。

6.1.6 FPT_EMS_EXT TOE 漏洩軽減

TOE の消費電力や、電磁放射などへの不要な情報の漏洩を悪用する例として、単純電力解析（SPA）、差分電力解析（DPA）、タイミング攻撃などがある。本ファミリは、CC パート 2 の他のコンポーネントが直接対応していない、不要な情報の漏洩制限に関する機能要件を記述している。

ファミリのふるまい

本ファミリのコンポーネントは、不要な情報の漏洩軽減についての要件に対処する。

コンポーネントのレベル付け:

FPT_EMS_EXT.1 TOE 漏洩軽減は、TSF データとユーザデータの暴露につながる不要な情報漏洩の軽減を要求する。

管理:FPT_EMS_EXT.1

特定の管理機能は識別されていない。

監査:FPT_EMS_EXT.1

予見される監査対象事象はない。

FPT_EMS_EXT.1 TOE 漏洩軽減

下位階層: なし

依存性: なし

FPT_EMS_EXT.1.1 TOE は、[*割付:TSF データのタイプのリスト*]および[*割付:ユーザデータのタイプのリスト*]へのアクセスを可能にする[*割付:攻撃インタフェースのリスト*]への[*割付:漏洩のタイプのリスト*]の漏洩を軽減しなければならない。

6.1.7 FPT_TUD_EXT 高信頼アップデート

CC はソフトウェアのアップデートに適した要件は定義されていないため必要である。

ファミリのふるまい

本ファミリのコンポーネントは、ソフトウェアをアップデートするための要件に対処する。

コンポーネントのレベル付け:

FPT_TUD_EXT.1 高信頼アップデートは、インストール前にアップデートを検証する機能を含めソフトウェアをアップデートするために提供される機能を要求する。

管理:FPT_TUD_EXT.1

特定の管理機能は識別されていない。

監査:FPT_TUD_EXT.1

予見される監査対象事象はない。

FPT_TUD_EXT.1 高信頼アップデート

下位階層: なし

依存性: FCS_COP.1

FPT_TUD_EXT.1.1 TSF は、ソフトウェアの現在のバージョンを問い合わせる能力を[割付: *サブジェクトのリスト*]に提供しなければならない。

FPT_TUD_EXT.1.2 TSF は、ソフトウェアのアップデートを開始する能力を[割付: *サブジェクトのリスト*]に提供しなければならない。

FPT_TUD_EXT.1.3 TSF は、ソフトウェアのアップデートを製造者による[選択: *デジタル署名*、*MAC*、*[割付:その他の手段]*]を用いて、インストールする前に検証しなければならない。

6.2 拡張セキュリティ保証コンポーネント

6.2.1 AVA_SCU_EXT SCU の脆弱性評定

目的

SCU の脆弱性分析は、[AAPS]に識別された攻撃手法によって、攻撃者による SFR の侵害を許すかどうかを決定するための評定のことである。SCU を組み込む TOE 開発者が留意すべき、低い攻撃能力を持つ攻撃者が実行する攻撃と、SOG-IS サポート文書[AAPS]を参考にしたハードウェア攻撃の攻撃能力の計算表を定義するために必要である。

[AAPS]に記載のある攻撃手法に基づく脆弱性調査は、攻撃者が実現しうる可能性がある潜在的脆弱性を確認するために評価者が実行する。評価者は、侵入テストを実行して、TOE の運用環境で潜在的脆弱性を悪用できないことを確認する。侵入テストは、評価者が、基本的な攻撃能力を想定して実行する。

コンポーネントのレベル付け

このファミリは、ただ 1 つのコンポーネントからなる

AVA_SCU_EXT.1 SCU 脆弱性調査

依存性: AVA_VAN.1

開発者アクションエレメント:

AVA_SCU_EXT.1.1D 開発者は、テストのための TOE を提供しなければならない。

内容・提示エレメント:

AVA_SCU_EXT.1.1C TOE は、テストに適していなければならない。

評価者アクションエレメント:

AVA_SCU_EXT.1.1E 評価者は、SCU の特性に基づき低い攻撃能力に対応した侵入テスト方法を考案し、侵入テストを実施しなければならない。

6.2.2 SCU 脆弱性調査（拡張 – AVA_SCU_EXT）

6.2.2.1 サブアクティビティの評価（AVA_SCU_EXT.1）

6.2.2.1.1 目的

SCU の脆弱性調査は、CEM の AVA_VAN.1 評価アクティビティの一部として実施され、SCU 搭載マイコン固有の脆弱性を識別するため、AVA_VAN.1 の CEM ワークユニットの詳細化を目的としている。

6.2.2.1.2 入力

このサブアクティビティ用の評価証拠は次の通りである。

a) テストに適した TOE

このサブアクティビティのその他の入力は次の通りである。

a) 潜在的な脆弱性に関する現在の情報（[AMSS]）

6.2.2.1.3 アクション AVA_SCU_EXT.1.1E

AVA_SCU_EXT.1.1C TOE は、テストに適していなければならない。

AVA_SCU_EXT.1-1 評価者は、9 章に基づき低い攻撃能力に対応した侵入テスト方法を考案し、侵入テストを実施しなければならない。

9 章では、攻撃者が攻撃を達成するために必要な攻撃能力を算出する尺度を示す。また、1 つまたは多数の攻撃段階で構成される攻撃パスの概念を紹介する。脆弱性を認識するために、攻撃パスの攻撃段階ごとに解析及びテストを実行する必要がある。

7 付録: 選択セキュリティ機能要件

以下は、セキュリティ機能要件の内容に合致するよう選択するセキュリティ機能要件である。

7.1 暗号サポート

FCS_COP.1/SKC 暗号操作(対称鍵暗号)

下位階層: なし

依存性: [FDP_ITC.1 セキュリティ属性なし利用者データインポート、または FDP_ITC.2 セキュリティ属性を伴う利用者データのインポート、または FCS_CKM.1 暗号鍵生成] FCS_CKM.4 暗号鍵破棄

FCS_COP.1.1/SKC TSF は、**[選択:下記の標準のリスト]**を満たす、特定された暗号アルゴリズム**[選択:下記の暗号アルゴリズム]**と、暗号鍵長**[選択:下記の鍵長]**に従って、[**選択:暗号、復号**]を実行しなければならない。

表 7-1 対称鍵暗号

識別子	暗号アルゴリズム	鍵長	標準のリスト
AES-CCM	AES CCM モード。最小サイズ 64 ビットの予測不能で繰り返しのないノンス使用。	[選択: 128 ビット, 192 ビット, 256 ビット]	ISO/IEC 18033-3 (AES) ISO/IEC 19772, sec. 8 (CCM) NIST SP800-38C (CCM)
AES-GCM	AES GCM モード。長さが 96 ビットの繰り返しのない IV で、決定論的 IV 構成方法[SP800- 38D, Section 8.2.1]を使用しなければならない。MAC 長は 96, 104, 112, 120, 128 ビットのいずれかでなければならない。	[選択: 128 ビット, 192 ビット, 256 ビット]	ISO/IEC 18033-3 (AES) ISO/IEC 19772, sec.11 (GCM) NIST SP800-38D (GCM)
AES-CBC	AES CBC モード。繰り返しのない予測不能な IV を使用。	[選択: 128 ビット, 192 ビット, 256 ビット]	ISO/IEC 18033-3 (AES) ISO/IEC 10116 (CBC) NIST SP800-38A (CBC)
AES-XTS	AES XTS モード。一意の調整値 (tweak) [選択: 任意の非負整数で始まる連続した非負整数, データユニットのシーケンス番号]	[選択: 128 ビット, 192 ビット, 256 ビット]	ISO/IEC 18033-3 (AES) [選択: IEEE 1619, NIST SP800-38E] (XTS)
AES-KWP	AES ベースの KWP	[選択: 128 ビット, 256 ビット]	ISO/IEC 18033-3 (AES), NIST SP 800-38F, sec. 6.3 (KWP) ISO/IEC 19772, clause 7 (key wrap)
AES-KW	AES ベースの KW	[選択: 128 ビット, 256 ビット]	ISO/IEC 18033-3 (AES), NIST SP 800-38F, sec. 6.2 (KW) ISO/IEC 19772, clause 7 (key wrap)
CAM-CBC	Camellia CBC モード。繰り返しのない予測不能な IV を使用。	[選択: 128 ビット, 256 ビット]	ISO/IEC 18033-3 (Camellia) ISO/IEC 10116 (CBC)

識別子	暗号アルゴリズム	鍵長	標準のリスト
CAM-CCM	Camellia CCM モード。最小サイズ 64 ビットの予測不能で繰り返しのないノンス使用。	[選択: 128 ビット, 256 ビット]	ISO/IEC 18033-3 (Camellia) ISO/IEC 19772, sec. 8 (CCM) SP800-38C
CAM-GCM	Camellia GCM モード。長さが 96 ビットの繰り返しのない IV で、決定論的 IV 構成方法[SP800-38D, Section 8.2.1]を使用しなければならない。MAC 長は 96, 104, 112, 120, 128 ビットのいずれかでなければならない。	[選択: 128 ビット, 256 ビット]	ISO/IEC 18033-3 (Camellia) ISO/IEC 19772, sec.11 (GCM) NIST SP800-38D
CAM-XTS	Camellia XTS モード。一意の調整値(tweak) [選択: 任意の非負整数で始まる連続した非負整数, データユニットのシーケンス番号]	[選択: 256 ビット, 512 ビット]	ISO/IEC 18033-3 (Camellia) [選択: IEEE 1619, SP800-38E] (XTS)
ChaCha-Poly	ChaCha20-Poly1305。最小サイズ 96 ビットの予測不能で繰り返しのないノンス使用。	256 ビット	RFC 8439

適用上の注釈 16

Camellia、ChaCha20-Poly1305 の暗号アルゴリズムについては、暗号アルゴリズム試験、サイドチャネル情報の測定方法が検討中であるため、開発者はこれらを選択する前に認証機関に相談しなければならない。

FCS_COP.1/KeyEnc 暗号操作（鍵暗号化）

下位階層: なし

依存性: [FDP_ITC.1 セキュリティ属性なし利用者データインポート、または FDP_ITC.2 セキュリティ属性を伴う利用者データのインポート、または FCS_CKM.1 暗号鍵生成] FCS_CKM.4 暗号鍵破棄

FCS_COP.1.1/KeyEnc TSF は、以下の**[選択：FCS_COP.1/SKC の標準のリスト]**に合致する、指定された暗号化アルゴリズム**[選択：下表の暗号アルゴリズム]**と、暗号鍵長**[選択：下表の鍵サイズ]**に従って、[*鍵の暗号化と復号*]を実行しなければならない。

表 7-2 鍵暗号化

識別子	暗号アルゴリズム	鍵サイズ	標準のリスト
SE1	[選択: AES-CCM, AES-GCM, AES-CBC]	[選択: 128 ビット, 192 ビット, 256 ビット]	左記暗号アルゴリズムに対応する、FCS_COP.1/SKC の標準のリストを選択する
SE2	[選択: AES-KWP, AES-KW, CAM-CBC, CAM-CCM, CAM-GCM]	[選択: 128 ビット, 256 ビット]	

FCS_COP.1/Hash 暗号操作(ハッシュ)

下位階層: なし

依存性: [FDP_ITC.1 セキュリティ属性なし利用者データインポート、または FDP_ITC.2 セキュリティ属性を伴う利用者データのインポート、または FCS_CKM.1 暗号鍵生成] FCS_CKM.4 暗号鍵破棄

FCS_COP.1.1/Hash TSF は、**[選択:ISO/IEC 10118-3:2018、FIPS 180-4, FIPS202]**に合致する、特定された暗号アルゴリズム**[選択:SHA-1、SHA-256、SHA-384、SHA-512、SHA-3-224、SHA-3-256 、SHA-3-384、SHA-3-512, SHAKE256]**~~と暗号鍵長[割付:*暗号鍵長*]~~に従って[*ハッシュ*]を実行しなければならない。

適用上の注釈 17

ハッシュ選択は、署名生成に使用されるアルゴリズムの全体的な強さと一致している必要がある。たとえば、TOE は P-256 の ECC には SHA-256、P-384 の ECC の場合は SHA-384、そして P-521 の ECC には SHA-512 を選択する。

SHA-1 は、ハッシュベースのメッセージ認証コード、鍵導出関数、および乱数ビット生成器に利用できる。

FCS_COP.1/MAC 暗号操作(MAC)

下位階層: なし

依存性: [FDP_ITC.1 セキュリティ属性なし利用者データインポート、または FDP_ITC.2 セキュリティ属性を伴う利用者データのインポート、または FCS_CKM.1 暗号鍵生成] FCS_CKM.4 暗号鍵破棄

FCS_COP.1.1/MAC TSF は、[**選択:ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2", NIST SP 800-38B**]に合致する、特定された暗号アルゴリズム**[選択:HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512, CMAC-AES-128, CMAC-AES-256]**と **[選択:HMAC, AES]で使われる**暗号鍵長**[割付:*暗号鍵長(ビット)*]**に従って[*メッセージ認証*]を実行しなければならない。

適用上の注釈 18

1つ以上の HMAC アルゴリズムが選択される場合、ST 作成者は、3 番目の選択で「HMAC」を選択し、1 番目の選択で「ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"」を選択する。割付において、鍵長[k]は、L1 と L2 の間の範囲に入る(適切なハッシュ関数について、ISO/IEC 10118 で定義されている)。例えば、SHA-256 については、L1=512 かつ L2 =256 となる、ここで、$L2 \leq k \leq L1$ となる。

1つ以上の CMAC アルゴリズムが選択される場合、ST 作成者は、3 番目の選択肢で「AES」及び 1 番目の選択で「NIST SP 800-38B」を選択する。割付については、鍵長は、128 と 256 の間の範囲に入る。

FCS_COP.1/SigVer 暗号操作(デジタル署名検証)

下位階層: なし

依存性: [FDP_ITC.1 セキュリティ属性なし利用者データインポート、または FDP_ITC.2 セキュリティ属性を伴う利用者データのインポート、または FCS_CKM.1 暗号鍵生成] FCS_CKM.4 暗号鍵破棄

FCS_COP.1.1/SigVer TSF は、**[選択：以下の標準のリスト]**を満たす、特定された暗号アルゴリズム**[選択：以下の暗号アルゴリズム]**と、暗号鍵長**[選択：以下の鍵長]**に従って、[*デジタル署名検証*]を実行しなければならない。

表 7-3 デジタル署名検証

識別子	暗号アルゴリズム	鍵長	標準のリスト
ECDSA	[選択: SHA-256, SHA-512]を使用した ECDSA [選択: brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, NIST P-256, NIST P-384, NIST P-521]	[選択: 256 ビット, 384 ビット, 512 ビット]	[選択: ISO/IEC 14888-3; FIPS186-4 (Section 6)] [ECDSA] RFC5639 (Section 3) [Brainpool Curves] FIPS186-4 (Appendix D.1.2) [NIST Curves] [選択: ISO/IEC 10118-3, (Section 10, 11); FIPS180-4, (Section 6)] [SHA]
EdDSA	EdDSA [選択: SHA-512 を使用した Ed25519, SHAKE256 を使用した Ed448]	[選択: 256 ビット, 456 ビット]	RFC8032 [EdDSA] [選択: ISO/IEC 10118-3, (Section 10, 11); FIPS180-4, (Section 6), FIPS202 (section 6)] [SHA]

FCS_KDF_EXT.1 暗号鍵導出

下位階層: なし

依存性：FCS_CKM.4 暗号鍵破棄 FCS_COP.1/MAC 暗号操作(メッセージ認証) FCS_RBG_EXT.1 乱数ビット生成

FCS_KDF_EXT.1.1 TSF は、[選択：FCS_RBG_EXT.1 で規定される乱数生成されたサブマスク、インポートされたサブマスク]を[選択：NIST SP 800-108 [選択：カウンターモードを用いた KDF、フィードバックモードを用いた KDF、ダブルパイプライン繰り返しモードを用いた KDF]、NIST SP 800-132] で定義されるとおり、その出力が少なくとも HGK と等しいセキュリティ強度（ビット数で）になるように、FCS_COP.1/MAC で規定される鍵付ハッシュ関数を用いて、中間鍵を導出しなければならない。

適用上の注釈 19

評価者が製品の鍵管理を完全に理解し、鍵が適切に保護されることを保証するための要件をどのように満たすかを十分に理解するように、製品の暗号鍵管理は十分に詳述されるべきである。

FCS_RBG_EXT.1 乱数ビット生成

下位階層：なし

依存性: なし

FCS_RBG_EXT.1.1 TSF は、ISO/IEC 18031:2011 に従い、[選択：Hash_DRBG (any), HMAC_DRBG (any), CTR_DRBG (AES)]を用いて決定論的乱数ビット生成サービスを実行しなければならない。

FCS_RBG_EXT.1.2 決定論的 RBG は[[*割付:ハードウェアベースのノイズ源の数*]*個のハードウェアベースのノイズ源*] からエントロピーを蓄積するような、少なくとも1つのエントロピー源によってシード値を与えられなければならない。ここでノイズ源については、ISO/IEC 18031:2011 に従い、生成する鍵やハッシュの最大のセキュリティ強度に少なくとも等しく、かつ最小でも[選択:128 ビット、192 ビット、256 ビット] のエントロピーを蓄積する NIST SP 800-90B に従ったエントロピー源によってシードを供給されなければならない。

適用上の注釈 20

[800-90B], Appendix C では、製品が直ちに使用する必要がある最小エントロピー測定が説明されている。乱数ビット生成器は十分なエントロピーを生成しなければならない。ISO / IEC 18031:2011 には、乱数を生成する 4 つの異なる方法が含まれている。これらはそれぞれ、基礎となる暗号プリミティブ(ハッシュ関数／暗号)に依存している。この PP では、Hash_DRBG または HMAC_DRBG には SHA-1、SHA-224、SHA-256、SHA-384、および SHA-512 が許可され、CTR_DRBG には AES ベースの実装のみが許可される。

FCS_SNI_EXT.1 ソルト、ノンス、IV 生成

下位階層:なし

依存性:FCS_RBG_EXT.1 乱数ビット生成

FCS_SNI_EXT.1.1 TSF は、[選択:ソルトを利用しない、[選択:FCS_RBG_EXT.1 で規定される DRBG、製造工程で TOE 外から提供されるサブマスク] によって生成されるソルトを使用する]ようにしなければならない。

FCS_SNI_EXT.1.2 TSF は、[選択:ノンスを利用しない、最小[*96*]ビット長の一意のノンスを使用する]ようにしなければならない。

FCS_SNI_EXT.1.3 TSF は、以下のやり方で IV を生成しなければならない[選択:

- CBC: IV は、繰り返してはならない、
- CCM: ノンスは、繰り返してはならない、
- XTS: IV はなし。調整値(Tweak)は、非負の整数であり、連続に割り当てられ、かつ任意の非負の整数から始まらなければならない、
- GCM: IV は、繰り返してはならない。与えられた秘密鍵について GCM の呼び出し回数は 2^{32} 回を超えてはならない]。

8 付録: オプションベースのセキュリティ機能要件

8.1 暗号サポート

FCS_CKM.1/AK 暗号鍵生成(非対称鍵)

下位階層: なし

依存性: [FCS_CKM.2 暗号鍵配付、または FCS_COP.1 暗号操作] FCS_CKM.4 暗号鍵破棄

FCS_CKM.1.1/AK TSF は、以下の[**選択: 下表の標準のリスト**]に合致する、指定された暗号鍵生成アルゴリズム[**選択:下表の鍵生成アルゴリズム**]と指定された暗号鍵長[**選択:下表の鍵長**]に従って、**非対称**暗号鍵[**選択:下表の鍵名**]を生成しなければならない。

表 8-1 非対称鍵のリスト

鍵名	鍵生成アルゴリズム	鍵長	標準のリスト
ECC-N	FIPS PUB 186-4 (Section B.4)	[選択: 256 (P-256), 384 (P-384), 512 (P-521)]	FIPS PUB 186-4 (Section B.4 & D.1.2)
ECC-B	FIPS PUB 186-4 (Section B.4)	[選択: 256 (brainpoolP256r1), 384 (brainpoolP384r1), 512 (brainpoolP512r1)]	RFC5639 (Section 3) [Brainpool Curves] FIPS PUB 186-4 (Section B.4)

FCS_CKM.1/SK 暗号鍵生成 (対称鍵)

下位階層: なし

依存性: [FCS_CKM.2 暗号鍵配付、または FCS_COP.1 暗号操作] FCS_CKM.4 暗号鍵破棄

FCS_CKM.1.1/SK TSF は、以下の[**選択: 下表の標準のリスト**]に合致する、指定された暗号鍵生成アルゴリズム[**選択:下表の鍵生成アルゴリズム**]と指定された暗号鍵長[**選択:下表の鍵長**]に従って、**対称**暗号鍵[**選択: 下表の鍵名**]を生成しなければならない。

表 8-2 対称鍵のリスト

鍵名	鍵生成アルゴリズム	鍵長	標準のリスト
RSK	FCS_RBG_EXT.1 で指定された乱数ビット生成器による直接生成	[選択: 128 ビット, 256 ビット, 512 ビット]	承認された RBG として NIST SP 800-133 (Section 5)に挙げられたものに ISO/IEC 18031も加えた NIST SP 800-133 (Section 7.1).
DSK	[選択: FCS_COP.1/KDF で指定した鍵導出関数]	[選択: 128 ビット, 256 ビット, 512 ビット]	NIST SP 800-108

適用上の注釈 21

512 ビット鍵長の選択は、AES-256 を使用する XTS-AES の場合である。AES-128 と AES-256 の両方に対する XTS-AES の場合、開発者は、NIST SP 800-133 のセクションのように、RBG からの直接生成を使用して完全キーを確実に生成することが期待されている。

TOE がさらなるコンディショニングなしに RBG の出力から直接鍵を作成することをサポートする場合、ST 作成者は RSK を選択し、TOE が通常 RBG からシードされ、それからあとに適切な鍵長までコンディショニングされる鍵導出関数をサポートする場合、DSK を選択すべきである。

FCS_CKM.4 暗号鍵破棄

下位階層: なし

依存性: [FDP_ITC.1 セキュリティ属性なし利用者データインポート、または FDP_ITC.2 セキュリティ属性を伴う利用者データのインポート、または FCS_CKM.1 暗号鍵生成]

FCS_CKM.4.1 TSF は、以下の[*標準なし*]に合致する、指定された暗号鍵破棄方法**[**

揮発メモリ用に、破棄は[選択:

(ア) 次の方法による 1 回の上書き:[選択:

① **TSF の RBG を利用した疑似乱数パターン**

② **0**

③ **1**

④ **新しい鍵値**

⑤ **[割付:いかなる機密を含まない何らかの値]**

]によって実行されなければならない

(イ) メモリへの電源断

(ウ) 鍵への参照を破棄した後、ガベージコレクションを要求する。];

]に従って、暗号鍵**と鍵材料**を破棄しなければならない。

適用上の注釈 22

揮発メモリの場合、TSF が消去対象のデータを保持している特定の物理メモリ位置をアドレス指定できず、それゆえ論理アドレスをアドレス指定(古いデータを保持している関連する物理アドレスを解放する)に依存し、物理アドレス内のデータが読み取りできなくなることを要求する(つまり、SFR 文で参照される「ガベージコレクション」)状況の中で選択「鍵への参照を破棄した後、ガベージコレクションを要求する。」が使われる。「揮発メモリへの電源断」とは、TOE の主電源をオフにすることを想定しており、スイッチのような回路によって揮発メモリへの電力供給を断つ手段は想定していない。

FCS_COP.1/SigGen 暗号操作(デジタル署名生成)

下位階層: なし

依存性: [FDP_ITC.1 セキュリティ属性なし利用者データインポート、または FDP_ITC.2 セキュリティ属性を伴う利用者データのインポート、または FCS_CKM.1 暗号鍵生成] FCS_CKM.4 暗号鍵破棄

FCS_COP.1.1/SigGen TSF は、**[選択:以下の標準のリスト]**を満たす、特定された暗号アルゴリズム**[選択:以下の暗号アルゴリズム]**と、暗号鍵長**[選択:以下の鍵長]**に従って、[*デジタル署名生成*]を実行しなければならない。

表 8-3 デジタル署名生成

識別子	暗号アルゴリズム	鍵長	標準のリスト
ECDSA	[選択: SHA-256, SHA-512] を使用した ECDSA [選択: brainpoolP256r1, brainpoolP384r1, brainpoolP512r1, NIST P-256, NIST P-384, NIST P-521]	[選択: 256 ビット, 384 ビット, 512 ビット、521 ビット]	[選択: ISO/IEC14888-3; FIPS186-4 (Section 6)] [ECDSA] RFC5639 (Section 3) [Brainpool Curves] FIPS186-4 (Appendix D.1.2) [NIST Curves] [選択: ISO/IEC 10118-3, (Section 10, 11); FIPS180-4, (Section 6)] [SHA]
EdDSA	EdDSA [選択: SHA-512 を使用した Ed25519, SHAKE256 を使用した Ed448]	[選択: 256 ビット, 456 ビット]	RFC8032 [EdDSA] [選択: ISO/IEC 10118-3, (Section 10, 11); FIPS180-4, (Section 6), FIPS202(section 6)] [SHA]

9 付録: AVA_SCU_EXT - SCU の脆弱性調査

この章では、SCU インサイドシステム委員会に参加している業界の利害関係者（大学、SCU 開発者、インテグレータ、および評価機関）によって議論された SCU 搭載 TOE の攻撃の能力を評価するための方法論を記載する。

この方法論の目的は、特定の TOE に対して攻撃を成功させるために必要な作業の評価を支援することである。SCU 搭載 TOE 固有の攻撃見積もり表の定義と、攻撃者が攻撃を実行するために必要な能力を計算するためのガイダンスを提供する。

9.1 要素の識別と攻撃能力のレート付け

コモンクライテリアでは、識別フェーズと悪用フェーズの区別がなくなっている。しかし、スマートカードコミュニティ内では、リスク管理に「識別」（攻撃の実証）のコストと「悪用」（例えば、スクリプトがインターネットで公開された場合）のコストを明確に区別することが必要である。したがって、SCU 搭載 TOE の評価もそれを踏襲し、攻撃能力を算出する場合に、この区別を維持する。識別と悪用の区別は、評価においては、攻撃パスの理解と文書化のために不可欠であるが、両方のフェーズが合わさって完全な攻撃になるので、攻撃能力の最終的な総和はこの 2 つのフェーズの点数を合計して算出する。

9.1.1 攻撃の計算方法

攻撃パスの識別、及び悪用の解析やテストは、関連する要素（所要時間、専門知識、TOE の知識、TOE へのアクセス、攻撃を実行するのに必要な機器、ならびにオープンサンプルまたは既知の秘密を持つサンプルを使用したかどうか）にマッピングされる。攻撃がいくつかの段階で構成される場合でも、識別及び悪用は攻撃パス全体に対して算出すればよい。攻撃の要素、攻撃の識別フェーズと悪用フェーズの考え方は、[AAPS] 4 章を参照すること。

9.1.2 所要時間

所要時間とは、脆弱性を識別して悪用するために要する時間である。本書は[AAPS] 4.2 を踏襲し、以下の表のとおりとする。所要時間の詳細は[AAPS] 4.2 を参照すること。

表 9-1 所要時間

詳細化	識別	悪用
< 1 時間	0	0
< 1 日	1	3
< 1 週間	2	4
< 1 か月	3	6
> 1 か月	5	8
> 6 か月	6	10
実際的ではない	*	*

所要時間における 1 日を 8 時間と定める。1 週間は 40 時間とし、1 か月は 160 時間とする。

9.1.3 専門知識

専門知識とは、脆弱性を識別して悪用するために必要な技術的専門知識である。[AAPS] 4.3 を踏襲し、以下の表のとおりとする。専門知識の詳細は[AAPS] 4.3 を参照すること。

表 9-2 専門知識

詳細化	識別	悪用
素人	0	0
熟練者	2	2
エキスパート	5	4
複数のエキスパート	7	6

9.1.4 TOE の知識

TOE の知識とは、脆弱性を識別して悪用するために必要な TOE 設計と運用の知識である。[AAPS] 4.4 を踏襲し、以下の表のとおりとする。TOE の知識の詳細は[AAPS] 4.4 を参照すること。

表 9-3 TOE の知識

詳細化	識別	悪用
公開	0	0
制限	2	2
秘密	4	3
危機的	6	5
非常に危機的	9	*
実際的ではない	*	*

9.1.5 TOE へのアクセス

TOE へのアクセスとは、脆弱性を識別して悪用するために必要な TOE の数である。[AAPS] 4.5 を踏襲し、以下の表のとおりとする。TOE へのアクセスの詳細は[AAPS] 4.5 を参照すること。

表 9-4 TOE へのアクセス

詳細化	識別	悪用
< 10 サンプル	0	0
< 30 サンプル	1	2
< 100 サンプル	2	4
> 100 サンプル	3	6
実際的ではない	*	*

9.1.5.2　TOE パッケージの効果

TOE は様々なパッケージを選択することができる。TOE に攻撃を加えるにはパッケージを物理的に除去することが必要になり、その除去手順は攻撃パスの一部である。パッケージの除去をどれほど困難にさせているか、によって次のようなレート付けになる。詳しくは[AAPS] 4.6 を参照のこと。

表 9-5　TOE パッケージの除去

詳細化	識別	悪用
低い効果	0	0
中程度の効果	1	2
高い効果	2	4

9.1.6　機器

TOE を攻撃するための機器を以下の 4 分類に分けて重みづけを行う。機器の詳細化の定義は、[AAPS] 4.6 を参照すること。

表 9-6　機器

詳細化	識別	悪用
なし	0	0
標準	1	2
特殊	3	4
特別注文	5	6
複数の特別注文	7	8

標準、特殊、特別注文に詳細化するときの指標の一つとして購入価格がある。[AAPS] 4.6.1 によれば購入価格による機器の詳細化は次の通りになる。

表 9-7　機器と購入価格

購入価格	詳細化
10K€まで （1€120 円として 120 万円まで）	標準
10K€から 200K€ （120～2,400 万円）	特殊
200K€を超える （2,400 万円を超える）	特別注文

それぞれの詳細化された機器にあてはまる具体的な例は[AAPS]4.6.1 を参照のこと。

9.1.7　オープンサンプル／既知の秘密を持つサンプル

以下の定義は[AAPS] 4.7 を要約したものである。詳しくは[AAPS] 4.7 を参照すること。

オープンサンプルとは、テストソフトウェアをダウンロードまたは実行できる機能を備えたハードウェアサンプルを意味し、テストソフトウェアが例えばファームウェアの対策をバイパスしたり、IC ハードウェアの内部構

成を変更したりして、TOE の安全でない構成にする場合がある。これにはベンダによる特定のテスト環境のサポートが含まれる場合がある。しかし、ハードウェアそのものを変更してはならない。

既知の秘密を持つサンプルとは、TOE の通常操作では利用できない機能を利用して、PIN や鍵などの秘密を TOE に設定できるサンプルを意味する。ベンダが、内部の秘密への特定のアクセスを許可する場合、既知の秘密を持つサンプルであるか、検討することができる。

ベンダが評価機関に提供する、機能テストに必要な機能インタフェースや鍵は、オープンサンプル／既知の秘密を持つサンプルとはみなされない。

レート付け表では、「オープンサンプル／既知の秘密を持つサンプル」の要素が定義されており、点は識別フェーズでのみ提供される。これらが悪用フェーズで使用されることが禁止されている。

表 9-8 既知の秘密を持つサンプル

詳細化	識別	悪用
公開／不要	0	NA
制限	2	NA
機密	5	NA
危機的	9	NA
実際的ではない	*	NA

表 9-9 オープンサンプル

詳細化	識別	悪用
公開／不要	0	NA
制限	2	NA
機密	5	NA
危機的	9	NA

9.1.8 攻撃能力の計算

以下のような表を作成して、これまで述べた攻撃に必要な要素を加算して合計点を求める。

表 9-10 攻撃能力の計算表

要素	識別	悪用
所要時間		
専門知識		
TOE の知識		
TOE へのアクセス		
機器		

要素	識別	悪用
オープンサンプルまたは既知の秘密を持つサンプル		
小計		
合計		

下表に基づき合計点を攻撃能力に変換する。

表 9-11 攻撃能力のレート付け

点数	TOE が対抗する攻撃者が持つ攻撃能力
0～15	なし（レート付けなし）
16～20	基本
21～24	強化基本
25～30	中程度
31 から上	高い

本 TOE は基本的な攻撃能力を持つ攻撃者の攻撃に耐えられなければならない。すなわち、攻撃に必要な要素を加算して、16 点に達しなければならない。一方で、TOE は 20 点を超す攻撃能力を持つ攻撃者の攻撃を想定してもよいが、TOE の扱う資産とセキュリティ対策の開発コストのバランスを考えるべきである。

9.1.9 本 TOE における攻撃者の人物像

TOE の用途は、スマートカードの内部に保管されている金融情報や個人情報のような価値の高い資産は想定していない。そのため高い専門知識を持つ者が高価な機器を用いて TOE を攻撃し、低い資産価値を取得する、というケースは考えにくい。TOE が想定する攻撃者は、技術を誇示したい愉快犯と考えられる。この愉快犯は一般的な機器を所有しており、大学や企業にあるやや高価な特殊機器を使うことができると思われる。

それぞれの要素には関係性がある。素人が特殊機器を使いこなすことは考えにくいため、熟練者と特殊機器の組み合わせが妥当である。また、中程度の効果のあるパッケージを除去するため、特殊機器を使う場合はサンプル数が少なくても済むかもしれないが、素人が標準的な機器を使う場合は試行錯誤に多くのサンプルを使うと思われる。同じように、熟練者が特殊機器を用いてサイドチャネル攻撃を短い所要時間で成功させる場合もあれば、素人が一般的な機器を用いてある程度の期間、サイドチャネル攻撃試行を続けるかもしれない。

このように、本 TOE が想定する基本的な攻撃能力を持つ攻撃者の人物像を定め、TOE を攻撃するための所要時間、専門知識、TOE の知識、TOE へのアクセス、機器を想定する。

9.1.9.1 所要時間

TOE は高い資産価値を持たないため、一つの攻撃パスによって 4 か月を超えるコストをかけて攻撃に成功しても得られる対価、あるいは誇示する名声は小さい。よって TOE を攻撃する、基本的な攻撃能力を持つ攻撃者が攻撃にかける最大の時間は 1 か月を超え、4 か月未満と想定する。よって「＜1 時間」「＜1 日」「＜1 週間」「＜1 か月」「＞1 か月」のいずれかになる。

9.1.9.2 専門知識

TOE を攻撃する、基本的な攻撃能力を持つ攻撃者の専門知識を次のように想定する。

1. 物理攻撃、センサやフィルタの制圧

   TOE のパッケージを除去し、メモリコンテンツを撮影したり、配線を切ったり、配線をショートさせたりして回路のふるまいを変更するために必要な専門知識は、「熟練者」とする。「エキスパート」、「複数のエキスパート」は、攻撃に関与しないと仮定する。また、物理攻撃に必要なパッケージの除去に使われるツールは、精密研磨装置、エッチング（プラズマエッチングや化学エッチング）などであり、これらを使いこなすにはある程度の経験が必要であるため、「素人」は想定しない。

2. かく乱攻撃、RNG への攻撃、テスト機能の悪用

   TOE の動作仕様を超えた環境に TOE をさらし、TOE に誤動作を起こさせたり、あるいは TOE にエネルギーを印加して誤動作を起こさせたりする攻撃に必要な専門知識は、「素人」と「熟練者」とする。TOE に高い電圧パルスを印加したり、電源の瞬停を起こしたり、回路の一部を高い温度にすることは「素人」でも可能である。また、レーザーワークステーションや EMI/BBI ワークステーションは使いこなすために経験が必要なことから「熟練者」とする。

3. サイドチャネル攻撃

   サイドチャネル攻撃に必要な専門知識は「素人」と「熟練者」とする。シャント抵抗の接続による消費電力の取得と統計解析は、計測手法と解析プログラムがインターネット上で広く開示されており、「素人」でも攻撃試行が可能である。また、放射電磁波の測定や精密な取得波形アライメントは「熟練者」レベルの技量が必要となる。しかし、資産価値の観点からエキスパート以上の専門知識を持つ攻撃者は想定外とする。

4. ソフトウェア攻撃

   ソフトウェア攻撃に必要な専門知識は「素人」と「熟練者」とする。ソフトウェアゲート API は一般に開示されないが、外部インタフェースからアプリケーションのバグを探索するファジングは、無料のツールがインターネットで公開されており「素人」でも試行できる。ファジングの探査パラメタの絞り込みにはプロトコルの専門知識を必要とするがエキスパートの専門知識までは必要とせず、「熟練者」と想定する。

9.1.9.3 TOE の知識

TOE を攻撃する基本的な攻撃能力を持つ攻撃者の TOE の知識は「公開」と想定する。言い換えると公開レベル以上の情報は、TOE 開発者によって保護されている。

9.1.9.4 TOE へのアクセス

TOE を攻撃する、基本的な攻撃能力を持つ攻撃者がアクセスする TOE を次のように想定する。

1. 物理攻撃、センサやフィルタの制圧

   TOE のパッケージを除去するため、攻撃の試行錯誤が必要となる。攻撃者レベルが熟練者であり、FIB などの特殊に分類される攻撃機器の取り扱いに習熟していると想定すれば、機器と専門知識の合計点が 16 点となる。基本的な攻撃能力の上限 20 点を想定すると、サンプル数は 10 個に満たない。攻撃者が素人で、標準的な機器により試行錯誤すると、失敗する個数は多くなるが、所要時

間との関係からサンプル数は 30 個に満たないだろう。以上のような想定から、TOE へのアクセスは多くとも 30 個に満たないと想定する。よって「＜30 サンプル」「＜10 サンプル」のいずれかになる。

2. かく乱攻撃、RNG への攻撃、テスト機能の悪用

   高い攻撃能力を持つ攻撃者に耐えうるスマートカード用 IC はベロシティカウンタを持つ場合がある。これは、異常なエネルギーを印加され、センサが感知するとカウントアップしていき、ある閾値を超えると IC の動作を永久に止めたり、機密情報を消去したりする機構である。この場合、攻撃の試行錯誤のためにアクセスする TOE 数を多く必要とする。本 TOE の場合は、資産を保護するための動作停止より、動作継続を利用者が望むと想定されるため、アクセスする TOE 数は 1 個でよいと考えられる。よって「＜10 サンプル」となる。

3. サイドチャネル攻撃、ソフトウェア攻撃

   攻撃パスの構築には 1 個で可能であり、その攻撃パスを用いた悪用も 1 個で可能である。よって「＜10 サンプル」となる。

TOE パッケージの効果については、高い効果は想定外とする。高い効果は、複数のエキスパート、特別注文の機器を必要とし、基本的な攻撃能力を持つ攻撃者の想定から外れるからである。よって「低い効果」「中程度の効果」のいずれかになる。

9.1.9.5 機器

TOE を攻撃する、基本的な攻撃能力を持つ攻撃者が使用する機器は「標準」と「特殊」とする。TOE は高い価値のある資産を扱わない。価値の低い資産を暴露するため、日本円で約 2,400 万円以上の「特別注文」の装置を用いて攻撃を実施することは想定しない。スキルを誇示したい愉快犯や大学関係者が容易に購入できる標準的な機器や、大学や研究機関が所有する特殊な機器が使われると想定する。

9.1.9.6 オープンサンプル／既知の秘密を持つサンプル

オープンサンプル／既知の秘密を持つサンプルは想定しない。[AAPS]段落 77 より、機能テストに使用される機能インタフェースと鍵はオープンサンプル／既知の秘密を持つサンプルにあてはまらない。したがって「なし」となる。

9.1.10 本 TOE における攻撃能力のレート付け

本 TOE における攻撃の要素をレート付け表にすると次の通りとなる。

表 9-12 攻撃能力の計算表

要素	識別	悪用
所要時間		
1 時間未満	0	0
1 日未満	1	3
1 週間未満	2	4
1 か月未満	3	6
1 か月以上	5	8

要素	識別	悪用
専門知識		
素人	0	0
熟練者	2	2
TOE の知識		
公開	0	0
TOE へのアクセス		
<10 サンプル	0	0
<30 サンプル	1	2
パッケージの効果		
低い効果	0	0
中程度の効果	1	2
機器		
なし	0	0
標準	1	2
特殊	3	4
オープンサンプル／既知の秘密を持つサンプル		
なし	0	NA

攻撃能力の計算表における要素を、基本的な攻撃能力である 16～20 点になるよう組み合わせると次の通りとなる。ただし、組み合わせ 2 は強化基本の攻撃能力の組み合わせ例、組み合わせ 6 は基本的な攻撃能力にも満たない例である。ここで、特殊機器を扱える攻撃者は熟練者、中程度の効果のあるパッケージの除去には 30 個に満たない個数で試行錯誤できる、という組み合わせを前提とする。また、TOE の知識は「公開」で 0 点、オープンサンプル／既知の秘密を持つサンプルも「なし」の 0 点である。

表 9-13 攻撃能力の組み合わせ

要素	組み合わせ 1				組み合わせ 2			
	識別		悪用		識別		悪用	
所要時間	＞1 か月	5	＜1 か月	6	＜1 週間	3	＜1 日	3
専門知識	素人	0	素人	0	熟練者	2	熟練者	2
TOE へのアクセス	＜30	1	＜10	2	＜30	1	＜10	0
パッケージ除去	中程度	1	中程度	2	中程度	1	中程度	2
機器	標準	1	標準	2	特殊	3	特殊	4
小計	8		12		10		11	
計	20				21			

要素	組み合わせ 3				組み合わせ 4			
	識別		悪用		識別		悪用	
所要時間	＞1 か月	5	＜1 か月	6	＜1 か月	3	＜1 週間	4
専門知識	素人	0	素人	0	熟練者	2	熟練者	2
TOE へのアクセス	＜10	0	＜10	0	＜10	0	＜10	0
パッケージ除去	低い効果	0	低い効果	2	低い効果	0	低い効果	0
機器	標準	1	標準	2	特殊	3	特殊	4
小計	6		10		8		10	
計	16				18			

要素	組み合わせ 5				組み合わせ 6			
	識別		悪用		識別		悪用	
所要時間	＞1 か月	5	＜1 か月	6	＜1 週間	2	NA	0
専門知識	熟練者	2	素人	0	エキスパート	5	NA	0
TOE へのアクセス	＜10	0	＜10	0	＜10	0	NA	0
パッケージ除去	低い効果	0	低い効果	2	中程度	1	NA	0
機器	標準	1	標準	2	特殊	3	NA	0
小計	8		10		11		0	
計	18				11			

9.2 攻撃方法の例

以下の攻撃能力の例は、スマートカード製品の開発、製造、セキュリティ評価、および配布に関与するさまざまなアクタ（ハードウェアベンダ、カードベンダ、OS プロバイダ、評価機関、認証機関、サービスプロバイダ）で構成されるセキュリティエキスパートのグループである、JHAS によって議論された。

次に示す例は[AAPS] 5 章を要約したものである。詳細は[AAPS] 5 章を参照のこと。また攻撃詳細は非公開の[AMSS]に記載している。[AMSS]は ICSS-JC に問い合わせること。

TOE に適用できる攻撃能力の計算例を示した。これは読者の理解のために作成した。

9.2.1 物理攻撃

半導体故障解析に用いられる各種機器を使用して、半導体のパッケージを除去して半導体内部にアクセスしたり、または線材を追加して回路を改ざんしたりできる。この加工により内部信号を暴露したり、ふるまいを改変したりできる。また、メモリのビット値読み取りまたはビット値の強制セットのため物理アクセスを行う。

この攻撃の主な効果は次の通りである。

- 暗号鍵のような秘密へのアクセス。
- フォルト注入や DFA を容易にするためのセキュリティ機能の無効化。

- 内部信号の強制（強制的に 0(L)または 1(H)にする）。

攻撃者がこの攻撃を行う場合、想定される攻撃パスは次の通りである。

1. バスプロービング

   本攻撃を行うためには、パッケージを除去したあと、数層ある配線層や絶縁膜を除去して回路レイアウトを把握し、見当をつけたメモリブロックの周辺回路からバスを見つける。回路を動作させつつバスにタングステン線を接続させるために FIB による加工が必要になる。8 ビットバスの場合、8 回の加工が必要になる。バスから配線を引き出し、バスアナライザにより信号を分析する。特殊機器とエキスパートの技量を必要とし、所要時間も識別で一か月を超えると想定される。悪用フェーズが 0 点の場合が考えられ、その場合 16 点に満たない。

2. 不揮発メモリコンテンツの復元

   不揮発メモリのうち、FLASH メモリの撮影には原子間力顕微鏡や、高価な SEM が必要となるため、低い攻撃能力を超える。ROM の場合は、パッケージと配線層を除去し金属顕微鏡や安価な SEM で撮影して画像をつなぎ合わせ、ソフトウェアで解析するにはエキスパートレベルの技量を必要とする。表 9-13 の組み合わせ 6 のレート付けとなるため、ROM に鍵や PIN のような機密情報を格納してはならない。

9.2.2 センサやフィルタの制圧

この攻撃は、TOE の正しい動作を脅かす動作環境（温度、電圧、クロック等）を監視するセンサやフィルタを無効化したり、バイパスしたりしようとする。TOE のハードウェア部またはソフトウェア部は、センサやフィルタからの応答を使用して、TOE を安全にするためのアクションを実行する。

この攻撃の主な効果は次の通りである。

- メモリやレジスタの内容が改ざんされる。
- プログラムフローが改ざんされる。
- CPU や暗号エンジンなどの TOE コンポーネントが誤動作する。
- 動作モードやパラメタが変更される。

攻撃者がこの攻撃を行う場合、想定される攻撃パスは次の通りである。

1. 電圧センサの無効化

   本攻撃を行うためには、パッケージを除去したあと、数層ある配線層や絶縁膜を除去して回路レイアウトを把握し、見当をつけた電源ブロックの周辺回路からセンサを見つける。センサ回路を動作させつつ無効化するようショート配線を接続させたり、配線を切ったりするために FIB による加工が必要になる。特殊機器とエキスパートの技量を必要とし、所要時間も識別で一か月を超えると想定される。悪用フェーズが 0 点の場合が考えられ、その場合 16 点に満たない。

9.2.3 かく乱攻撃（フォルト注入攻撃）、RNG への攻撃、テスト機能の悪用

かく乱攻撃はフォルトインジェクション（フォルト注入）攻撃ともいわれる。TOE に悪用可能な誤動作を発生させるために行う攻撃で、前項に述べたように異常な動作環境にさらす方法と、TOE 外部から強い光や電磁波、電源への高電圧パルス波のようなエネルギーを印加して TOE をかく乱させる方法がある。

RNG への攻撃は主にフォルトを RNG 回路に注入して誤動作させる。

テスト機能の悪用は、セキュアブートプログラムをフォルト注入によって誤動作させたりして、テストモードへ遷移させる。

この攻撃の主な効果は次の通りである。

- メモリ読み出し中の読み出し値の改ざん。
- レジスタやメモリのコンテンツの改ざん。設定レジスタの設定値を変えたり、失敗カウンタの値を変えたり、命令レジスタの内容を改ざんしたり、プログラムカウンタの内容を改ざんしたりして TOE のふるまいを変更する。
- 暗号エンジンを誤動作させ、DFA を行う。
- RNG を誤動作させ、品質の低い乱数を出力させる。

攻撃者がこの攻撃を行う場合、想定される攻撃パスは次の通りである。

1. レーザー、EMFI、BBI によるフォルト注入

   本攻撃を行うためには、パッケージを除去したあと、SoC のサブストレートを露出させ、レーザーワークステーションよりレーザースポットを照射・移動させつつ TOE を動作させ誤動作を引き起こす。BBI も同じように BBI ワークステーションによりプローブをサブストレートに接触・印加・移動を繰り返させる。EMI の場合は微小コイルをサブストレートに近づけ、印加、移動を繰り返す。パッケージの除去に試行錯誤が必要で、ワークステーションは特殊機器に相当すると考えると表 9-13 の組み合わせ 2 あるいは 4 相当のレート付けを超えるような対抗策が必要になる。

2. 電圧グリッチによるフォルト注入

   本攻撃を行うためには、電源ラインに任意波形発生器からスパイク状に電圧を変動させたり、瞬停させたりして誤動作を引き起こす。任意波形発生器は価格にもよるが標準的な機器であり、素人が容易に試行できる。表 9-13 の組み合わせ 3 あるいは 5 相当のレート付けを超えるような対抗策が必要になる。

9.2.4 サイドチャネル攻撃

サイドチャネル攻撃は、暗号演算中の消費電力、放射電磁波を測定して、アルゴリズム実装の意図しない漏洩情報を利用して暗証番号や暗号鍵のような秘密情報を推測する攻撃である。

この攻撃の主な効果は次の通りである。

- 統計解析により鍵や PIN などの秘密情報を暴露する。
- EEPROM への書き込みタイミングや、PIN 比較などの TOE のふるまいを把握する。

攻撃者がこの攻撃を行う場合、想定される攻撃パスは次の通りである。

1. 測定した消費電流波形による解析

   本攻撃を行うためには、SoC の電源端子にシャント抵抗を接続して暗号演算実行時の電流を測定し、統計的解析で暗号鍵を推測する。標準的なオシロスコープで測定ができ、統計解析ソフトウェ

アもインターネットに公開されている。表 9-13 の組み合わせ 3 あるいは 5 相当のレート付けを超えるような対抗策が必要になる。

2. 測定した放射電磁波による解析

本攻撃を行うためには、パッケージを除去し、SoC にできるだけ測定コイルを近づけ、電磁波を測定する。ノイズを除去するため適切なノイズフィルタが必要となる。ノイズの中のリーク成分を取得するため、特殊に分類される高価なオシロスコープが必要となり、また識別時にコイルの配置、ノイズ除去に熟練者レベルの技量を必要とする。表 9-13 の組み合わせ 4 あるいは 5 相当のレート付けを超えるような対抗策が必要になる。

9.2.5 ソフトウェア攻撃

ソフトウェア攻撃は、オーバーフローを悪用して受信バッファのデータを暴露したり、プロトコルの再送コマンドを悪用してコマンドデータを暴露したり、ソフトウェアの不具合やプロトコルの RFU 実装を悪用する攻撃である。

この攻撃の主な効果は次の通りである。

- 秘密データを暴露する。

攻撃者がこの攻撃を行う場合、想定される攻撃パスは次の通りである。

1. プロトコル解析

標準になっているプロトコルの RFU 部分にデータを設定しふるまいを観察する。素人が標準的な機器を用いて試行できるため、表 9-13 の組み合わせ 3 あるいは 5 相当のレート付けを超えるような対抗策が必要になる。

2. ファジングによる解析

コマンドの取りうる値を網羅的に作成して TOE へ送信し、プロトコルの仕様範囲外のデータを受信したときの TOE のふるまいを観察する。TOE のふるまいは無応答になったり、リセットしたりする場合もあり、特殊機器を用いて挙動を観察しつつ繰り返しコマンド送信する。表 9-13 の組み合わせ 4 あるいは 5 相当のレート付けを超えるような対抗策が必要になる。

以上