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 **3:** セキュリティ保証コンポーネント

2006 年 **9** 月

バージョン **3.1**
改訂第 **1** 版

CCMB-2006-09-003

平成 19 年 3 月翻訳第 1.2 版
独立行政法人 情報処理推進機構
セキュリティセンター
情報セキュリティ認証室

IPA まえがき

はじめに

本書は、「IT セキュリティ評価及び認証制度」において、「認証機関が公開する評価基準」の規格として公開している Common Criteria(以下、CC という)を翻訳した文書である。

原文

Common Criteria for Information Technology Security Evaluation
Part1: Introduction and general model Version 3.1
September 2006 CCMB-2006-09-001
Part2: Security functional components Version 3.1
September 2006 CCMB-2006-09-002
Part3: Security assurance components Version 3.1
September 2006 CCMB-2006-09-003

まえがき

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の本バージョン(CC v3.1)は、2005 年に CC v2.3 が公開されて以来、最初の主要な改訂版である。

CC v3.1 は、重複する評価アクティビティを排除し、製品の最終保証にあまり役立たないアクティビティを削減または排除し、誤解を減らすために CC 用語を明確にし、セキュリティ保証が必要である領域に対する評価アクティビティを再構築し焦点を当て、必要に応じて新しい CC 要件を追加することを目的としている。

CC バージョン 3.1 は、次のパートから構成される:

- パート 1: 概説と一般モデル
- パート 2: セキュリティ機能コンポーネント
- パート 3: セキュリティ保証コンポーネント

商標:

- UNIX は、米国及びその他の諸国の The Open Group の登録商標である。
- Windows は、米国及びその他の諸国の Microsoft Corporation の登録商標である。

法定通知:

*以下に示す政府組織は、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の本バージョンの作成に貢献した。これらの政府組織は、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、バージョン*3.1 *のパート1から3(CC* 3.1 *と呼ぶ)の著作権を共有したまま、ISO/IEC 15408 国際標準の継続的な開発/維持の中で、CC* 3.1 *を使用するために ISO/IEC に対し、排他的でないライセンスを許可している。ただし、適切と思われる場合に CC* 3.1 *を使用、複製、配布、翻訳及び改変する権利は、これらの政府組織が保有する。*

オーストラリア/ニュージーランド:	*The Defence Signals Directorate and the Government Communications Security Bureau;*
カナダ:	*Communications Security Establishment;*
フランス:	*Direction Centrale de la Securite des Systemes d'Information;*
ドイツ:	*Bundesamt fur Sicherheit in der Informationstechnik;*
日本:	*独立行政法人 情報処理推進機構(Information-technology Promotion Agency);*
オランダ:	*Netherlands National Communications Security Agency;*
スペイン:	*Ministerio de Administraciones Publicas and Centro Criptologico Nacional;*
英国:	*Communications-Electronics Security Group;*
米国:	*The National Security Agency and the National Institute of Standards and Technology*

目次

図一覧

表一覧

1 序説

1 この CC パート 3 に定義されているセキュリティ保証コンポーネントは、プロテクションプロファイル(PP)またはセキュリティターゲット(ST)に表されているセキュリティ保証要件に対する基礎である。

2 これらの要件は、TOE 保証要件を表現する標準的な手段を規定している。この CC パート 3 は、保証コンポーネント、保証ファミリ、及び保証クラスのセットをカタログ化している。また、PP 及び ST の評価基準も定義しており、評価保証レベル(EAL)と呼ばれる、TOE の保証をレート付けするための既定の CC 尺度を定義する評価保証レベルも示している。

3 パート 3 の対象読者には、セキュアな IT 製品の消費者、開発者、評価者が含まれる。CC パート 1 の 7 章は、CC の対象読者及び対象読者からなるグループによる標準の使用についての追加情報を提供している。これらのグループは、パート 3 を次のように使うことができる:

- 消費者は、PP または ST に記述されているセキュリティ対策方針を達成するための保証要件を表すコンポーネントを選択する際に、この CC パート 3 を使用して、TOE で要求されるセキュリティ保証レベルを決定する。
- 開発者は、TOE を構成するときに実際のまたは認識された消費者のセキュリティ要件に応じ、TOE の保証要件のステートメントを解釈する際、及び TOE の保証アプローチを決定する際にこの CC パート 3 を参照する。
- 評価者は、TOE の保証を確定する際、及び PP と ST を評価する際に、不可欠な評価基準のステートメントとして、このパートで定義されている保証要件を使用する。

2 適用範囲

4 この CC パート 3 は、CC の保証要件を定義している。ここには、コンポーネント TOE の保証を測定するための尺度を定義する評価保証レベル(EAL)、統合 TOE の保証を測定するための尺度を定義する統合保証パッケージ(CAP)、これらの保証レベルとパッケージを構成する個々の保証コンポーネント、及び PP と ST の評価基準が含まれている。

3 規定の参照

5 以下の参照文書は、本文書の適用のために不可欠である。日付の付いている参照資料については、指定した版のみが適用される。日付のない参照資料については、(修正を含む)最新版の参照文書が適用される。

CC-1 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、バージョン 3.1、改訂第 1 版、2006 年 9 月 パート 1: 概説と一般モデル

CC-2 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、バージョン 3.1、改訂第 1 版、2006 年 9 月 パート 2: 機能セキュリティコンポーネント

4 用語と定義、記号と略語

6 本文書の目的のために、CC パート 1 で使用された用語、定義、記号、及び略語を適用する。

5 概要

5.1 CC パート 3 の構成

7 6 章では、CC パート 3 のセキュリティ保証要件で使われるパラダイムを記述している。

8 7 章では、保証クラス、ファミリ、コンポーネント、及び評価保証レベルの提示構造とそれらの関係、及び統合保証パッケージの構造を記述している。また、10 章から 17 章に記述されている保証クラスとファミリの特性も記述している。

9 8 章では、EAL を詳細に定義している。

10 9 章では、CAP を詳細に定義している。

11 10 章から 17 章では、CC パート 3 の保証クラスを詳細に定義している。

12 附属書 A では、開発クラスの背景にある概念について詳しく説明し、例を示している。

13 附属書 B では、統合 TOE 評価と統合クラスの背景にある概念を説明している。

14 附属書 C では、保証コンポーネント間の依存性を要約している。

15 附属書 D では、PP と、APE クラスのファミリとコンポーネントの間の相互参照を示している。

16 附属書 E では、EAL と保証コンポーネントの間の相互参照を示している。

17 附属書 F では、CAP と保証コンポーネントの間の相互参照を示している。

6 保証のパラダイム

18 この章の目的は、保証に対する CC のアプローチを支持する原理を示すことである。この章を理解することにより、読者は、CC パート 3 保証要件の合理的根拠を理解できる。

6.1 CC の原理

19 CC の原理は、セキュリティ及び組織のセキュリティ方針を犯す脅威を明確に表現し、提案するセキュリティ手段が意図する目的に対して明らかに十分であることである。

20 そこで、脆弱性の可能性、脆弱性を実行させる能力(意図的悪用または意図しない誘発)、及び脆弱性が実行されることにより引き起こされる損害の範囲を軽減する手段が採用されるべきである。さらに、脆弱性のその後の識別、及び脆弱性が悪用または誘発されることの排除、緩和、及び/または通知を容易にする手段が採用されるべきである。

6.2 保証アプローチ

21 CC の原理は、信頼されるべき IT 製品の評価(能動的な調査)に基づいて保証を提供することである。評価は、保証を提供する伝統的な手段であり、先行する評価基準書の基礎である。既存のアプローチと調和を取るために、CC は、同様の原理を採用している。CC は、適用範囲、深さ、及び厳格性を一層強調することにより、専門の評価者による、証拠資料及び結果としての IT 製品の有効性を測定することを提案している。

22 CC は、保証を得るための他の手段の相対的利点を排除しておらず、またそれらについての注釈も行っていない。保証を得るための別のアプローチに関する調査が継続されている。成熟した別のアプローチがこれらの調査アクティビティから明らかになれば、それらをこの CC に含めることが検討される。現在の CC は、将来それらを取り入れることができるように構成されている。

6.2.1 脆弱性の重要性

23 不正利益の取得及び善意ではあるがセキュアでない行為のいずれかであれ、セキュリティ方針を侵害する機会を積極的に利用しようとする脅威エージェントが存在すると想定される。脅威エージェントは、意図せずにセキュリティの脆弱性を誘発し、組織に損害を与えることがある。機密に関わる情報を処理する必要性と、十分に信頼された製品の可用性の欠如のために、IT の障害をもたらす重大なリスクが存在する。したがって、IT セキュリティの違反が重大な損失をもたらすことがある。

24 IT セキュリティの違反は、ビジネスでの IT の適用時に、脆弱性の意図的悪用または意図しない誘発によって引き起こされる。

25 IT 製品で生じる脆弱性を阻止する手順を踏むべきである。可能な限り、脆弱性には次のように対処するべきである:

- 排除 -- つまり、すべての実行可能な脆弱性を明らかにし、排除または無効にする有効な手順を踏むべきである;
- 最小化 -- つまり、脆弱性の実行による潜在的な影響を、容認できる残存レベルにまで軽減するための有効な手順を踏むべきである;
- 監視 -- つまり、残存脆弱性を実行させる試みを検出し、損失を抑える手順を踏むことができるようにする有効な手順を踏むべきである;

6.2.2 脆弱性の原因

26 脆弱性は、以下の障害により起きることがある:

- 要件 -- つまり、IT 製品は、必要とされるすべての機能と特徴を所有しているが、なお、セキュリティに関してその製品を不適切または無効にする脆弱性を含む;
- 開発 -- つまり、IT 製品がその仕様を満たしていない、及び/または開発上の標準が不十分であるか、設計上の選択が不適切であるために脆弱性が導入される;
- 運用 -- つまり、IT 製品は正しい仕様に従って正しく構成されているが、運用の管理が不適切であるために脆弱性が導入された。

6.2.3 CC 保証

27 保証は、IT 製品がそのセキュリティ対策方針を達成しているという確信の根拠である。保証は、実証されていない主張、これまでの関連する経験、または特別の経験などのソースを参照することで得られる。ただし、この CC は、能動的な調査を通して保証を提供する。能動的な調査とは、セキュリティ特性を決定するための IT 製品の評価である。

6.2.4 評価を通した保証

28 評価は、保証を得るための伝統的な手段であり、CC のアプローチの基礎となっている。評価技法には次のものが含まれるが、必ずしもこれだけに限定されない:

- プロセス及び手続きの分析とチェック;
- プロセス及び手続きが適用されていることのチェック;
- TOE 設計表現の間の対応分析;
- 要件に対する TOE 設計表現の分析;
- 証拠書類の検証;
- ガイダンス文書の分析;
- 開発された機能テストと提供された結果の分析;
- 独立機能テスト;

- 脆弱性(欠陥仮説法を含む)の分析;
- 侵入テスト。

6.3 CC 評価保証の尺度

29 CC 原理は、評価のための労力が大きいほど、大きな保証結果が得られること、及び必要最小限の労力で必要な保証レベルを提供することが目標であることを主張している。労力のレベルは、次のことに基づいて増加する:

- 適用範囲 -- つまり、IT 製品のより多くの部分が対象になると、労力は大きくなる;
- 深さ -- つまり、詳細な設計や詳細な実装を使用すると、労力は大きくなる;
- 厳格性 -- つまり、より構造化された形式的な方法で適用されると、労力は大きくなる。

7 セキュリティ保証コンポーネント

7.1 セキュリティ保証クラス、ファミリ、及びコンポーネントの構造

30 次の節では、保証クラス、ファミリ、及びコンポーネントを表す際に使用される構造について記述する。

31 図 1 は、この CC パート 3 に定義されている SAR を示している。SAR の最も抽象的なセットは、クラスと呼ばれることに注意のこと。各クラスには保証ファミリが含まれ、保証ファミリには保証コンポーネントが含まれ、保証コンポーネントには保証エレメントが含まれる。クラスとファミリは、SAR を分類するための分類方法を提供するために使われる。一方、コンポーネントは、PP/ST で SAR を特定するために使われる。

7.1.1 保証クラスの構造

32 図 1 は、保証クラスの構造を示す。

7.1.1.1 クラス名

33 各保証クラスには一意の名前が割り当てられる。この名前は、保証クラスが扱うトピックを示す。

34 保証クラス名の一意の短い形式も提供される。これは、保証クラスを参照するための主な手段である。採用された規則では、「A」の次にクラス名に関係する 2 文字が続く。

7.1.1.2 クラスの概説

35 各保証クラスには、クラスの構成が記述され、クラスの意図を扱う補足説明を含む概説の節がある。

7.1.1.3 保証ファミリ

36 各保証クラスには、少なくとも 1 つの保証ファミリが含まれる。保証ファミリの構造については、次の節で説明する。

図 1　　保証クラス/ファミリ/コンポーネント/エレメントの階層

7.1.2　　保証ファミリの構造

37　　図 1 は、保証ファミリの構造を示す。

7.1.2.1　　ファミリ名

38　　各保証ファミリには一意の名前が割り当てられる。この名前は、保証ファミリが扱うトピックについての記述情報を提供する。各保証ファミリは、同じ意図を持つ他のファミリが含まれている保証クラスの中に置かれる。

39　　保証ファミリ名の一意の短い形式も提供される。これは、保証ファミリを参照するために使われる主な手段である。採用されている規則では、クラス名の短い形式が使われ、その後に下線文字が続き、次にファミリ名に関係する 3 文字が続く。

7.1.2.2 目的

40 保証ファミリの目的の節は、保証ファミリの意図を表す。

41 この節は、ファミリが扱うように意図されているCC保証のパラダイムに特に関係する目的を記述する。保証ファミリの記述は、全般的なレベルにとどめている。目的に必要な特別な詳細は、特定の保証コンポーネントに組み込まれる。

7.1.2.3 コンポーネントのレベル付け

42 各保証ファミリには、1 つまたは複数の保証コンポーネントが含まれる。保証ファミリのこの節では、使用可能なコンポーネントについて記述し、それらの区別を説明する。主な目的は、保証ファミリが、PP/ST に対する SAR の必要な、または有用な部分であることが決定された後に、これらの保証コンポーネントを区別することである。

43 複数のコンポーネントが含まれている保証ファミリは、レベル付けが行われ、コンポーネントにレベルを付ける方法の根拠が示される。この根拠は、適用範囲、深さ、及び/または厳格性に関して示される。

7.1.2.4 適用上の注釈

44 保証ファミリに適用上の注釈の節が存在する場合、その節には保証ファミリの追加情報が含まれる。これは、保証ファミリの利用者(例えば、PP と ST の作成者、TOE の設計者、評価者)が特に関心を持つ情報である。表現は非形式的であり、例えば、使用上の制約及び特別の注意が必要となる領域に関する警告が扱われる。

7.1.2.5 保証コンポーネント

45 各保証ファミリには、少なくとも 1 つの保証コンポーネントが含まれる。保証コンポーネントの構造については、次の節で説明する。

7.1.3 保証コンポーネント構造

46 図 2 は、保証コンポーネント構造を示す。

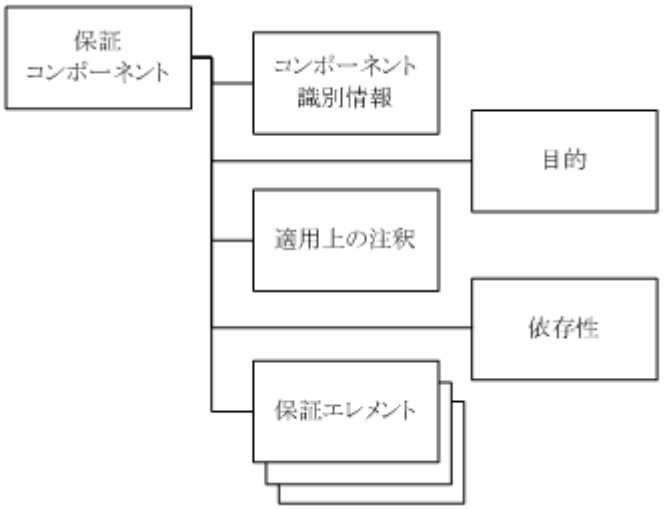

図 2 保証コンポーネント構造

47 ファミリ内のコンポーネント間の関係は、ボールド表記を用いて強調表示される。新規、及び階層内でこれまでのコンポーネントの要件を越えて拡張または修正されている要件のこれらの部分は、ボールドで表示される。

7.1.3.1 コンポーネント識別情報

48 コンポーネント識別情報の節は、コンポーネントを識別、分類、登録、及び参照するために必要な記述情報を提供する。

49 各保証コンポーネントには一意の名前が割り付けられる。この名前は、保証コンポーネントが扱うトピックについての記述情報を提供する。各保証コンポーネントは、セキュリティの目的を共有する保証ファミリの中に置かれる。

50 保証コンポーネント名の一意の短い形式も提供される。これは、保証コンポーネントを参照するために使われる主な手段である。使用される規則では、ファミリ名の短い形式が使用され、次にピリオドが続き、次に数字が続く。各ファミリ内のコンポーネントに対する数字は、1 から順に割り付けられる。

7.1.3.2 目的

51 保証コンポーネントに目的の節が存在する場合、その節には特定の保証コンポーネントの特定の目的が含まれる。この節を持つ保証コンポーネントについて、コンポーネントの特定の意図を示し、目的のさらに詳細な説明を提供する。

7.1.3.3 適用上の注釈

52 保証コンポーネントの適用上の注釈の節が存在する場合には、コンポーネントを容易に使用するための追加情報が含まれる。

7.1.3.4 依存性

53 保証コンポーネントの間の依存性は、コンポーネントが自己完結型ではなく、他のコンポーネントの存在に依存するときに起きる。

54 各保証コンポーネントは、他の保証コンポーネントに対する依存性の完全なリストを提供する。コンポーネントによっては、「依存性なし」を示してもよい。これは、識別される依存性は存在しないことを示す。依存されているコンポーネントは、他のコンポーネントに依存してもよい。

55 依存性リストは、必要とされる最小限の保証コンポーネントのセットを識別する。依存性リストで、あるコンポーネントの下位階層にあるコンポーネントが、依存性を満たすために使用される場合もある。

56 特別の状況では、示された依存性が適用できない場合がある。PP/ST の作成者が、特定の依存性を適用できない理由の根拠を提供することで、その依存性を満たさないことを選択してもよい。

7.1.3.5 保証エレメント

57 各保証コンポーネントには、保証エレメントのセットが提供される。保証エレメントは、それ以上分割しても意味のある評価結果が得られないセキュリティ要件である。これは、CC で認められている最小のセキュリティ要件である。

58 各保証エレメントは、保証エレメントの以下の 3 つのセットの 1 つに属するものとして識別される。

- 開発者アクションエレメント: 開発者が行わなければならないアクティビティ。このアクションのセットは、次に続くエレメントのセットで参照されている証拠資料によってさらに評価付けされる。開発者アクションの要件は、エレメント番号の後に「D」の文字を追加することによって識別される。

- 証拠の内容・提示エレメント: 必要とされる証拠、証拠が示さなければならないもの、及び証拠が伝えなければならない情報。証拠の内容・提示の要件は、エレメント番号の終わりに「C」の文字を追加することによって識別される。

- 評価者アクションエレメント: 評価者が行わなければならないアクティビティ。このアクションのセットには、証拠の内容・提示エレメントに記述されている要件が満たされていることの確認が明示的に含まれる。また、開発者がすでに行っているものに加えて実行しなければならない明示的なアクションと分析も含まれる。暗黙の評価者アクションも、証拠の内容・提示要件に示されていない開発者のアクションエレメントの結果として実行される。評価者アクションの要件は、エレメント番号の終わりに「E」の文字を追加することにより識別される。

59 開発者アクションと証拠の内容・提示は、PP または ST の SFR を満たしている TOE に保証を示す開発者の責任を表すために使用される保証要件を定義する。

60 評価者アクションは、評価の 2 つの側面での評価者の責任を定義する。第 1 の側面は、「APE: プロテクションプロファイル評価」及び「ASE: セキュリティターゲット評価」の章の APE クラスと ASE クラスに従った PP/ST の妥当性の確認である。第 2 の側面は、TOE がその SFR と SAR に対する適合性の検証である。PP/ST が妥当であり、要件が TOE によって満たされていることを実証することにより、評価者は、定義されたセキュリティの課題を TOE がその運用環境で解決するという信頼の基礎を提供できる。

61 開発者アクションエレメント、証拠の内容・提示エレメント、及び明示的評価者アクションエレメントは、TOE の ST においてなされるセキュリティ主張の検証に費やされなければならない評価者の労力を識別している。

7.1.4 保証エレメント

62 各エレメントは、満たす必要がある要件を表す。要件のこれらのステートメントは、明確かつ簡潔で、曖昧でないことが意図されている。したがって、重文は存在せず、分離可能な要件はそれぞれ個別のエレメントとして記述される。

7.1.5 コンポーネントの分類

63 この CC パート 3 には、関係する保証に基づいてグループ化されたファミリのクラスとコンポーネントが含まれている。各クラスの冒頭に、クラス内のファミリと各ファミリのコンポーネントを表す図が示される。

図 3 サンプルクラスのコンポーネント構成図

64 上記の図 3 には、単一のファミリを含んだクラスが示されている。このファミリには、直線的に階層化された 3 つのコンポーネントが含まれている(つまり、コンポーネント 2 は、特定のアクション、特定の証拠、あるいはアクションまたは証拠の厳格性の観点から、コンポーネント 1 以上を必要とする)。この CC パート 3 の保証ファミリはすべて直線的に階層化されているが、将来追加される保証ファミリにとって直線性は必須の基準ではない。

7.2 EAL 構造

65 図 4 は、CC パート 3 に定義されている EAL 及び関連する構造を示す。図は、保証コンポーネントの内容を示しているが、この情報は、CC に定義されている実際のコンポーネントを参照することにより、EAL に含まれていることが意図されていることに注意のこと。

図 4 EAL 構造

7.2.1 EAL 名

66 各 EAL には一意の名前が割り付けられる。この名前は、EAL の意図についての記述情報を提供する。

67 EAL 名の一意の短い形式も提供される。これは、EAL を参照するために使われる主な手段である。

7.2.2 目的

68 EAL の目的の節は、EAL の意図を表す。

7.2.3 適用上の注釈

69 EAL に適用上の注釈の節が存在する場合、その節には EAL の利用者(例えば、PP と ST の作成者、この EAL を目標とする TOE の設計者、評価者)が特に関心を持つ情報が含まれる。表現は非形式的であり、例えば、使用上の制約及び特別の注意が必要となる領域に関する警告が扱われる。

7.2.3.1 保証コンポーネント

70 各 EAL には、保証コンポーネントのセットが選択されている。

71 与えられた EAL により提供されているものより上位の保証レベルは、以下のことにより達成させることができる:

- 他の保証ファミリから追加の保証コンポーネントを取り込む;
- 保証コンポーネントを同じ保証ファミリの上位レベルの保証コンポーネントで置き換える。

7.2.4 保証と保証レベルの関係

72 図 5 は、CC に定義されている SAR と保証レベルの関係を示す。保証コンポーネントはさらに保証エレメントに分解されるが、保証エレメントを保証レベルによって個々に参照することはできない。図の矢印は、EAL からクラス内で定義されている保証コンポーネントへの参照を表すので注意のこと。

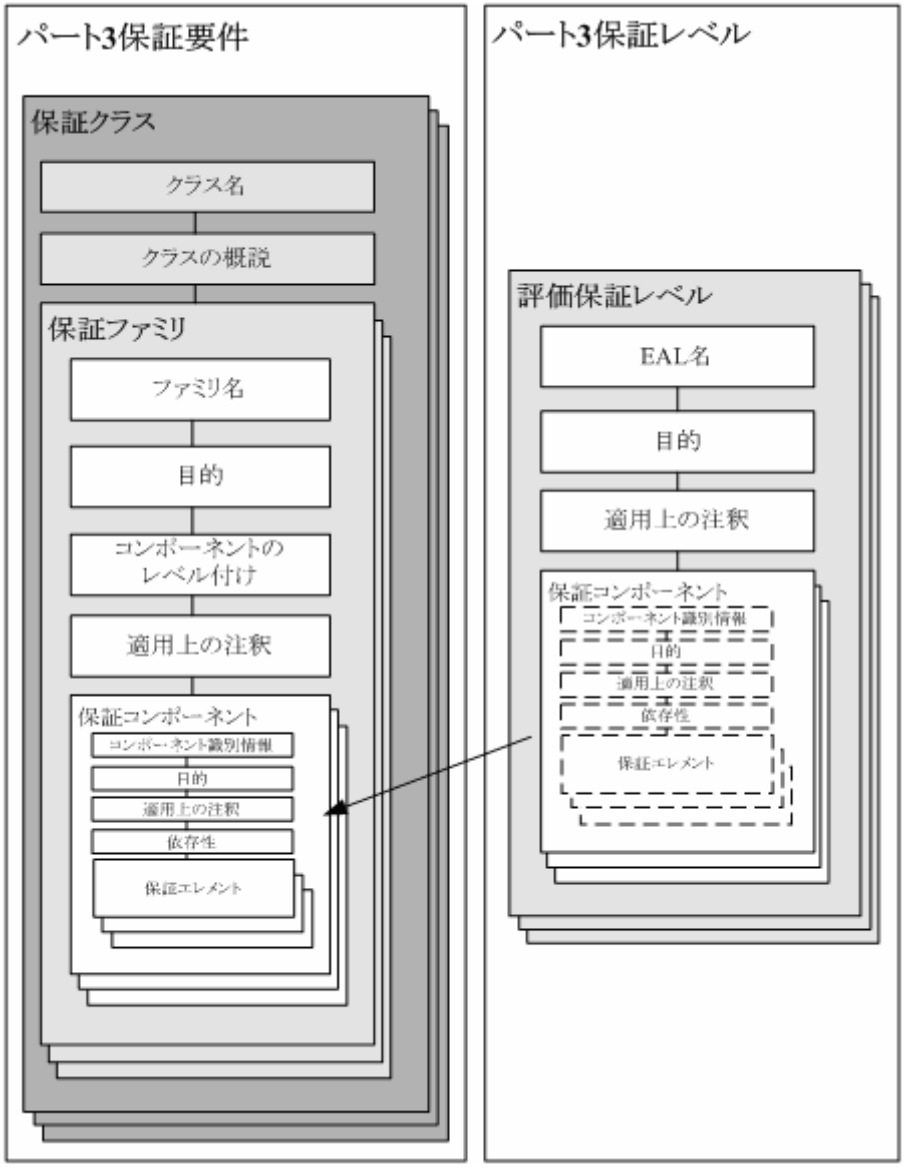

図 5 保証及び保証レベルの関連

7.3 CAP 構造

73 CAP の構造は、EAL の構造と似ている。この 2 種類のパッケージの主な違いは、それぞれが適用される TOE の種類にある。つまり、EAL はコンポーネント TOE に適用され、CAP は統合 TOE に適用される。

74 図 6 は、CC パート 3 に定義されている CAP 及び関連する構造を示す。図は、保証コンポーネントの内容を示しているが、この情報は、CC に定義されている実際のコンポーネントを参照することにより、CAP に含まれていることが意図されていることに注意のこと。

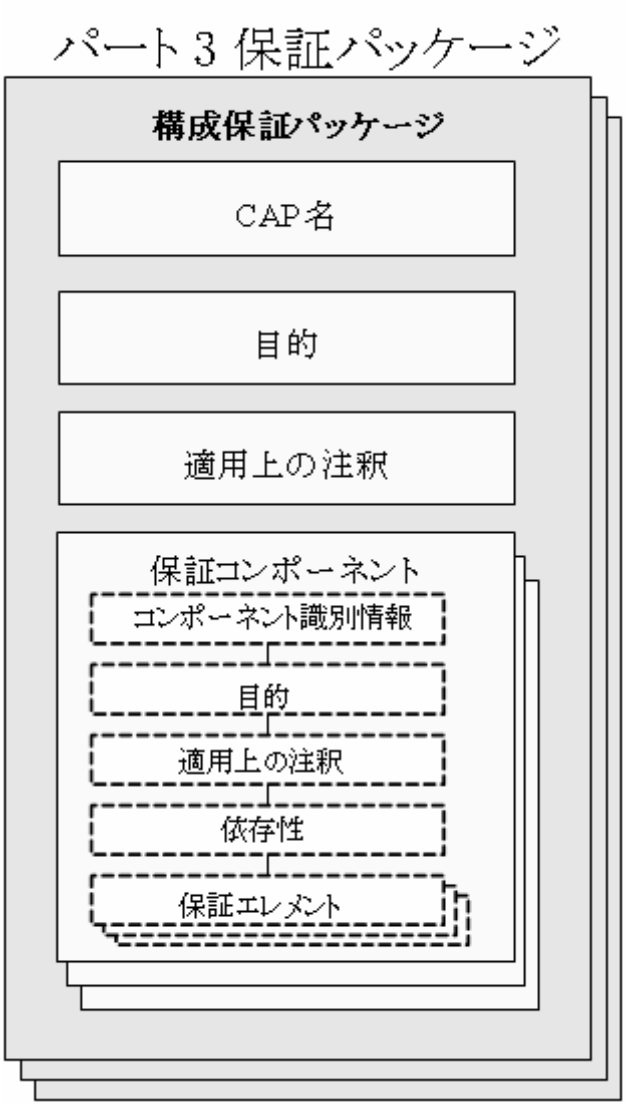

図 6 CAP 構造

7.3.1 CAP 名

75 各 CAP には一意の名前が割りあてられる。この名前は、CAP の意図についての記述情報を提供する。

76 CAP 名の一意の短い形式も提供される。これは、CAP を参照するために使われる主な手段である。

7.3.2 目的

77 CAP の目的の節は、CAP の意図を表す。

7.3.3 適用上の注釈

78 CAP に適用上の注釈の節が存在する場合、その節には CAP の利用者(例えば、PP と ST の作成者、この CAP を目標とする統合 TOE のインテグレータ、評価者)が特に関心を持つ情報が含まれる。表現は非形式的であり、例えば、使用上の制約及び特別の注意が必要となる領域に関する警告が扱われる。

7.3.3.1 保証コンポーネント

79 各 CAP には、保証コンポーネントのセットが選択されている。

80 一部の依存性は、統合 TOE アクティビティが依存する依存コンポーネントの評価中に実行されるアクティビティを識別する。依存性が依存コンポーネントアクティビティ上にあることが明示的に識別されていない場合、依存性は、統合 TOE の別の評価アクティビティに対するものである。

81 与えられた CAP により提供されているものより上位の保証レベルは、以下のことにより達成させることができる:

- 他の保証ファミリから追加の保証コンポーネントを取り込む;
- 保証コンポーネントを同じ保証ファミリの上位レベルの保証コンポーネントで置き換える。

82 ACO: CAP 保証パッケージに含まれる統合コンポーネントは、コンポーネントに対して意味のある保証を提供しないため、コンポーネント TOE 評価に対する追加として使用されるべきではない。

7.3.4 保証と保証レベルの関係

83 図 7 は、CC に定義されている SAR と統合保証パッケージの関係を示す。保証コンポーネントはさらに保証エレメントに分解されるが、保証エレメントを保証パッケージによって個々に参照することはできない。図の矢印は、クラス内で定義されている保証コンポーネントへの CAP からの参照を表すので注意のこと。

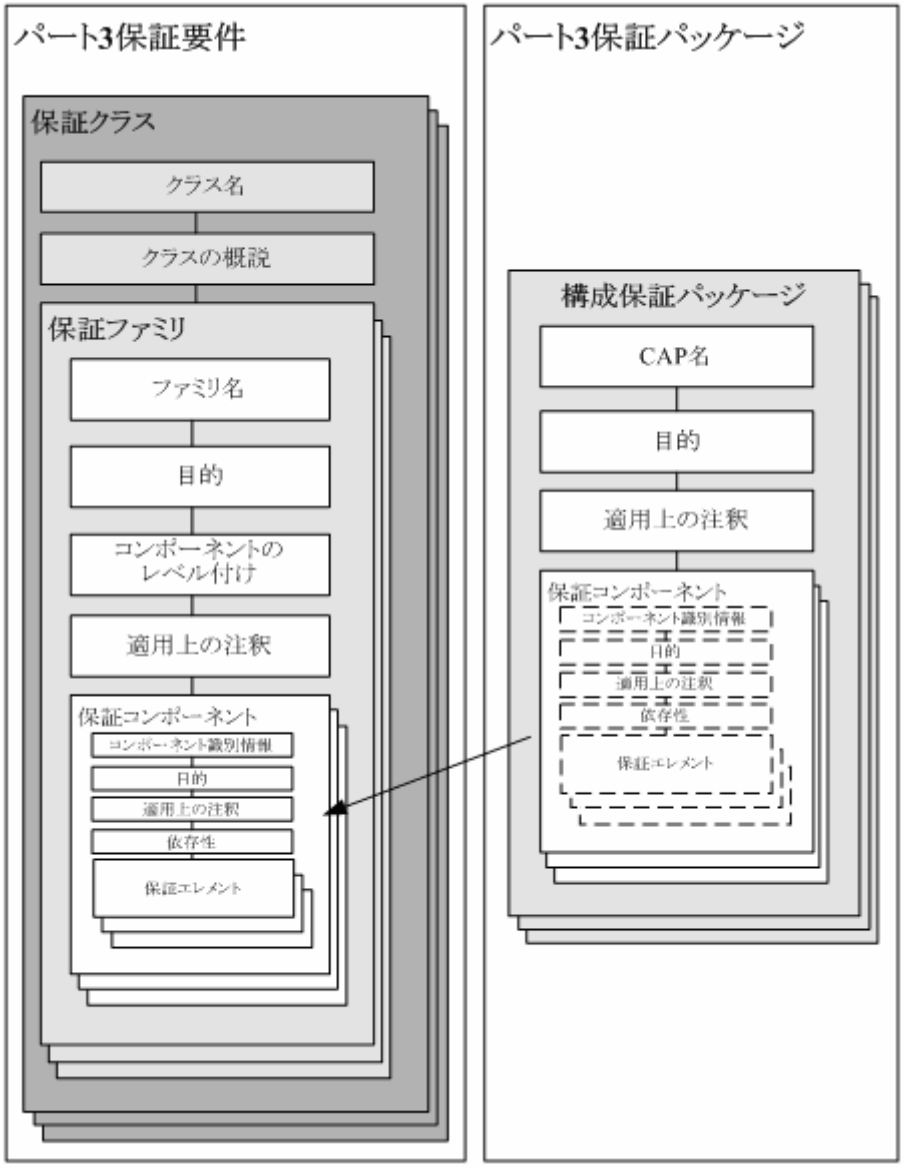

図 7　保証及び統合保証パッケージの関連

8 評価保証レベル

84 評価保証レベル(EAL)は、得られる保証のレベルと、そのレベルの保証を得るためのコスト及び可能性とを比較考量する段階的な尺度を提供する。CC のアプローチは、評価終了時における TOE の保証の概念と、TOE が運用されている間のその保証の維持の概念を区別して識別している。

85 CC パート 3 のファミリとコンポーネントが、必ずしもすべて EAL に含まれないことに注意しなければならない。これは、これらが有意義な望ましい保証を提供しないことを意味するものではない。これらのファミリとコンポーネントは、それらが有用性を提供する PP や ST の EAL の追加として考慮されることが期待される。

8.1 評価保証レベル(EAL)の概要

86 表 1 は、EAL の要約を示している。列は、階層的に並べられた EAL のセットを表し、行は保証ファミリを表す。その結果として得られるマトリックスの各数字は、適用すべき特定の保証コンポーネントを識別している。

87 次の節に概説されているように、階層的に並べられた 7 つの評価保証レベルが、TOE の保証のレート付けのために CC に定義されている。それらは、各 EAL がそれより下位のすべての EAL よりも多くの保証を表すため、階層的に並べられている。EAL から EAL への保証の増加は、同じ保証ファミリから階層的に上位の保証コンポーネントへの置換(つまり、厳格性、適用範囲、及び/または深さを拡大する)及び他の保証ファミリからの保証コンポーネントの追加(つまり、新しい要件を追加する)によって達成される。

88 これらの EAL は、この CC パート 3 の第 7 章に記述されている保証コンポーネントの適切な組み合わせからなる。さらに正確には、各 EAL は各保証ファミリから 1 つ以下のコンポーネントを取り込んでおり、あらゆるコンポーネントのすべての保証依存性を扱っている。

89 EAL は、CC に定義されているが、保証の他の組み合わせを表すことも可能である。特に、「追加」(augmentation)の概念によって、(EAL にまだ取り込まれていない保証ファミリからの)EAL に対する保証コンポーネントの追加、または(同じ保証ファミリで階層的に上位の他の保証コンポーネントによる)保証コンポーネントの置換が許可される。CC に定義されている保証構造において、EAL は要件を追加されることのみが可能である。「その構成する保証コンポーネントを欠いた EAL」の概念は、有効な主張として標準では認められない。追加に伴って、EAL に追加された保証コンポーネントの有効性と付加価値を正当化する義務が主張者の側に課せられる。EAL には、拡張された保証要件を追加することもできる。

保証クラス	保証ファミリ	評価保証レベル別の保証コンポーネント						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
開発	ADV_ARC		**1**	1	1	1	1	1
	ADV_FSP	**1**	**2**	**3**	**4**	**5**	5	**6**
	ADV_IMP				**1**	1	**2**	2
	ADV_INT					**2**	**3**	3
	ADV_SPM						**1**	1
	ADV_TDS		**1**	**2**	**3**	**4**	**5**	**6**
ガイダンス文書	AGD_OPE	**1**	1	1	1	1	1	1
	AGD_PRE	**1**	1	1	1	1	1	1
ライフサイクル サポート	ALC_CMC	**1**	**2**	**3**	**4**	4	**5**	5
	ALC_CMS	**1**	**2**	**3**	**4**	**5**	5	5
	ALC_DEL		**1**	1	1	1	1	1
	ALC_DVS			**1**	1	1	**2**	2
	ALC_FLR							
	ALC_LCD			**1**	1	1	1	**2**
	ALC_TAT				**1**	**2**	**3**	3
セキュリティ ターゲット評価	ASE_CCL	**1**	1	1	1	1	1	1
	ASE_ECD	**1**	1	1	1	1	1	1
	ASE_INT	**1**	1	1	1	1	1	1
	ASE_OBJ	**1**	**2**	2	2	2	2	2
	ASE_REQ	**1**	**2**	2	2	2	2	2
	ASE_SPD		**1**	1	1	1	1	1
	ASE_TSS	**1**	1	1	1	1	1	1
テスト	ATE_COV		**1**	**2**	2	2	**3**	3
	ATE_DPT			**1**	**2**	**3**	3	**4**
	ATE_FUN		**1**	1	1	1	**2**	2
	ATE_IND	**1**	**2**	2	2	2	2	**3**
脆弱性評定	AVA_VAN	**1**	**2**	2	**3**	**4**	**5**	5

表 1　評価保証レベルの要約

8.2　評価保証レベルの詳細

90　次の節では EAL を定義する。その際、特定の要件とそれらの要件の一律的な特性との差異を、ボールド体を用いて強調する。

8.3　評価保証レベル 1(EAL1) - 機能テスト

目的

91　EAL1 は、正しい運用についてある程度の信頼が要求されるが、セキュリティへの脅威が重大とみなされない場合に適用される。個人情報または同様の情報の保護に関して当然の配慮がなされているとの論旨をサポートするために、独立の保証が要求されるところで価値がある。

92　EAL1 は、限定されたセキュリティターゲットのみを必要とする。TOE が満たさなければならない SFR を記述すれば十分であり、セキュリティ対策方針を通して脅威、OSP、及び前提条件から SFR を派生させる必要はない。

93　EAL1 は、仕様に対する独立テスト、提供されたガイダンス証拠資料の調査など、顧客に対して有効な TOE の評価を提供する。EAL1 評価は、TOE の開発者の支援を受けずに、最小の費用で実施できるように意図されている。

94 このレベルの評価は、TOE の機能がその証拠資料に対していわば一貫しているという証拠を提供するべきである。

保証コンポーネント

95 **EAL1 は、限定されたセキュリティターゲットとその ST 内の SFR の分析により基本レベルの保証を提供する。この分析は、セキュリティのふるまいを理解するために、機能とインタフェースの仕様及びガイダンス証拠資料を使用して行われる。**

96 **分析は、公知の潜在的な脆弱性の探索及び TSF の独立テスト(機能及び侵入)によってサポートされる。**

97 **また EAL1 は、TOE 及び関連する評価文書の一意の識別情報を通して保証を提供する。**

98 **この EAL は、評価されていない IT に比べ、有意義な保証の増加を提供する。**

保証クラス	保証コンポーネント
ADV: 開発	ADV_FSP.1 基本機能仕様
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクルサポート	ALC_CMC.1 TOE のラベル付け
	ALC_CMS.1 TOE の CM 範囲
ASE: セキュリティターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.1 運用環境のセキュリティ対策方針
	ASE_REQ.1 主張されたセキュリティ要件
	ASE_TSS.1 TOE 要約仕様
ATE: テスト	ATE_IND.1 独立テスト - 適合
AVA: 脆弱性評定	AVA_VAN.1 脆弱性調査

表 2 EAL1

8.4 評価保証レベル 2(EAL2) - 構造テスト

目的

99 EAL2 は、設計情報とテスト結果の提供に関して開発者の協力を必要とする。ただし、正常な商業的習慣を越える労力を開発者側に要求するべきではない。したがって、コストまたは時間の投資の大幅な増加を要求するべきではない。

100 そこで、EAL2 は、開発者または利用者が完全な開発記録を簡単に使用できない場合に、低レベルから中レベルの独立に保証されたセキュリティを必要とする環境に適用できる。そのような状況は、従来のシステムの安全性を高めるとき、または開発者へのアクセスが制限されるところで生じる。

保証コンポーネント

101 **EAL2** は、**完全な**セキュリティターゲット及びその ST 内の SFR の分析により保証を提供する。この分析は、セキュリティのふるまいを理解するために、機能とインタフェースの仕様、ガイダンス証拠資料、**及び TOE のアーキテクチャの基本的な記述**を使用して行われる。

102 分析は、TSF の独立テスト、**機能仕様に基づく開発者テストの証拠、開発者テスト結果の選択的で独立した確認、及び基本的な攻撃能力を持つ侵入攻撃者に対する抵抗力を実証する(提供された機能仕様、TOE 設計、アーキテクチャ設計、及びガイダンス証拠に基づく)脆弱性分析**によってサポートされる。

103 また、**EAL2** は、**構成管理システムの使用**と**セキュアな配付手続きの証拠**を通して保証を提供する。

104 この EAL は、**開発者テスト、(公知の脆弱性の探索に加えて)脆弱性分析、さらに詳細な TOE 仕様に基づく独立テストを要求することにより、EAL1 からの**有意義な保証の増加を**表す**。

保証クラス	保証コンポーネント
ADV: 開発	ADV_ARC.1 セキュリティアーキテクチャ記述
	ADV_FSP.2 セキュリティ実施機能仕様
	ADV_TDS.1 基本設計
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクル サポート	ALC_CMC.2 CM システムの使用
	ALC_CMS.2 TOE の一部の CM 範囲
	ALC_DEL.1 配付手続き
ASE: セキュリティ ターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様
ATE: テスト	ATE_COV.1 カバレージの証拠
	ATE_FUN.1 機能テスト
	ATE_IND.2 独立テスト - サンプル
AVA: 脆弱性評定	AVA_VAN.2 脆弱性分析

表 3　EAL2

8.5 評価保証レベル 3(EAL3) - 方式テスト、及びチェック

目的

105 EAL3は、良心的な開発者が、既存の適切な開発方法を大幅に変更することなく、設計段階で有効なセキュリティエンジニアリングから最大の保証を得られるようにする。

106 EAL3 は、開発者または利用者が、中レベルで独立して保証されたセキュリティを必要とし、大幅なリエンジニアリングを行わずに TOE とその開発の完全な調査を必要とする状況で適用される。

保証コンポーネント

107 **EAL3** は、完全なセキュリティターゲット及びその ST 内の SFR の分析により保証を提供する。この分析は、セキュリティのふるまいを理解するために、機能とインタフェースの仕様、ガイダンス証拠資料、及び TOE の**設計**の**アーキテクチャ**記述を使用して行われる。

108 分析は、TSF の独立テスト、機能仕様**及び TOE 設計**に基づく開発者テストの証拠、開発者テスト結果の選択的で独立した確認、及び基本的な攻撃能力を持つ侵入攻撃者に対する抵抗力を実証する(提供された機能仕様、TOE 設計、アーキテクチャ設計、及びガイダンス証拠に基づく)脆弱性分析によってサポートされる。

109 また、**EAL3** は、**開発環境管理**の使用、**TOE** 構成管理の使用、及びセキュアな配付手続きの証拠を通して保証を提供する。

110 この EAL は、**セキュリティ機能性**の**さらに完全な**テスト**カバレージ**、及び TOE が**開発中に改ざんされないというある程度の信頼を提供するメカニズム及び/または手続き**を要求することにより、**EAL2** からの有意義な保証の増加を表す。

保証クラス	保証コンポーネント
ADV: 開発	ADV_ARC.1 セキュリティアーキテクチャ記述
	ADV_FSP.3 完全な要約を伴う機能仕様
	ADV_TDS.2 アーキテクチャ設計
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクル サポート	ALC_CMC.3 許可の管理
	ALC_CMS.3 実装表現の CM 範囲
	ALC_DEL.1 配付手続き
	ALC_DVS.1 セキュリティ手段の識別
	ALC_LCD.1 開発者によるライフサイクルモデルの定義
ASE: セキュリティ ターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様
ATE: テスト	ATE_COV.2 カバレージの分析
	ATE_DPT.1 テスト: 基本設計
	ATE_FUN.1 機能テスト
	ATE_IND.2 独立テスト - サンプル
AVA: 脆弱性評定	AVA_VAN.2 脆弱性分析

表 4　EAL3

8.6 評価保証レベル 4(EAL4) - 方式設計、テスト、及びレビュー

目的

111 EAL4 は、厳格ではあるが、多大な専門知識、スキル、及びその他の資源を必要としない正常な商業的開発習慣に基づいて、有効なセキュリティエンジニアリングから最大の保証を開発者が得られるようにする。EAL4 は、既存の製品ラインへのレトロフィットが経済的に実現可能であると思われる最上位レベルである。

112 そこで、EAL4 は、開発者または利用者が従来の商品としての TOE に独立して保証された中レベルから高レベルのセキュリティを必要とし、セキュリティ特有のエンジニアリングコストを追加で負担する用意ができている状況で適用される。

保証コンポーネント

113 **EAL4** は、完全なセキュリティターゲット及びその ST 内の SFR の分析により保証を提供する。この分析は、セキュリティのふるまいを理解するために、機能と**完全な**インタフェースの仕様、ガイダンス証拠資料、TOE の**基本モジュール**設計の記述、**及び実装のサブセット**を使用して行われる。

114 分析は、TSF の独立テスト、機能仕様及び TOE 設計に基づく開発者テストの証拠、開発者テスト結果の選択的で独立した確認、**開発者による脆弱性の探索の証拠**、及び**拡張された基本的な**攻撃能力を持つ侵入攻撃者に対する抵抗力を実証する(提供された機能仕様、TOE 設計、**実装表現**、アーキテクチャ設計、及びガイダンス証拠に基づく)脆弱性分析によってサポートされる。

115 また、**EAL4** は、開発環境管理の使用、**自動化を含む追加の** TOE 構成管理の使用、及びセキュアな配付手続きの証拠を通して保証を提供する。

116 この EAL は、さらに多くの**設計記述**、**実装のサブセット**、及び TOE が開発中または**配付中に**改ざんされないという信頼を提供する**向上した**メカニズム及び/または手順を要求することにより、**EAL3** からの有意義な保証の増加を表す。

保証クラス	保証コンポーネント
ADV: 開発	ADV_ARC.1 セキュリティアーキテクチャ記述
	ADV_FSP. 4 完全な機能仕様
	ADV_IMP.1 TSF の実装表現
	ADV_TDS.3 基本モジュール設計
AGD: ガイダンス文書	AGD_OPE.1: 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクル サポート	ALC_CMC.4 製造支援、受入れ手続き、及び自動化
	ALC_CMS.4 課題追跡の CM 範囲
	ALC_DEL.1 配付手続き
	ALC_DVS.1 セキュリティ手段の識別
	ALC_LCD.1 開発者によるライフサイクルモデルの定義
	ALC_TAT.1 明確に定義された開発ツール
ASE: セキュリティ ターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様
ATE: テスト	ATE_COV.2 カバレージの分析
	ATE_DPT.2 テスト: セキュリティ実施モジュール
	ATE_FUN.1 機能テスト
	ATE_IND.2 独立テスト - サンプル
AVA: 脆弱性評定	AVA_VAN.3 焦点を置いた脆弱性分析

表 5　EAL4

8.7 評価保証レベル 5(EAL5) - 準形式的設計、及びテスト

目的

117 EAL5 は、専門的なセキュリティエンジニアリング技法を中程度に適用することによりサポートされる厳格な商業的開発習慣に基づいて、セキュリティエンジニアリングから最大の保証を開発者が得られるようにする。そのような TOE は、おそらく EAL5 保証を達成する意図を持って設計され、開発される。専門的な技法を適用しない厳格な開発と比較して、EAL5 要件による追加のコストは、大きくはないと思われる。

118 そこで、EAL5 は、開発者または利用者が計画された開発において独立して保証される高レベルのセキュリティを必要とし、専門的なセキュリティエンジニアリング技法による非合理的なコストを負担することのない厳格な開発アプローチを必要とする状況で適用される。

保証コンポーネント

119 **EAL5** は、完全なセキュリティターゲット及びその ST 内の SFR の分析により保証を提供する。この分析は、セキュリティのふるまいを理解するために、機能と完全なインタフェースの仕様、ガイダンス証拠資料、TOE の設計の記述、及び実装を使用して行われる。**また、モジュール化された TSF 設計も必要となる。**

120 分析は、TSF の独立テスト、機能仕様に基づく開発者テストの証拠、TOE 設計、開発者テスト結果の選択的で独立した確認、及び**中程度の**攻撃能力を持つ侵入攻撃者に対する抵抗力を実証する**独立した**脆弱性分析によってサポートされる。

121 また、**EAL5** は、開発環境管理の使用、自動化を含む**包括的な** TOE 構成管理の使用、及びセキュアな配付手続きの証拠を通して保証を提供する。

122 この EAL は、**準形式的**設計**記述**、**完全な**実装、**さらに構造化された(それによって分析可能な)アーキテクチャ**、及び開発中に TOE が改ざんされないという信頼を提供する向上したメカニズム及び/または手続きを要求することにより、**EAL4** からの有意義な保証の増加を表す。

保証クラス	保証コンポーネント
ADV: 開発	ADV_ARC.1 セキュリティアーキテクチャ記述
	ADV_FSP.5 追加の誤り情報を伴う完全な準形式的機能仕様
	ADV_IMP.1 TSF の実装表現
	ADV_INT.2 適切に構造化された内部
	ADV_TDS.4 準形式的モジュール設計
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクル サポート	ALC_CMC.4 製造支援、受入れ手続き、及び自動化
	ALC_CMS.5 開発ツールの CM 範囲
	ALC_DEL.1 配付手続き
	ALC_DVS.1 セキュリティ手段の識別
	ALC_LCD.1 開発者によるライフサイクルモデルの定義
	ALC_TAT.2 実装標準への準拠
ASE: セキュリティ ターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様
ATE: テスト	ATE_COV.2 カバレージの分析
	ATE_DPT.3 テスト: モジュール設計
	ATE_FUN.1 機能テスト
	ATE_IND.2 独立テスト - サンプル
AVA: 脆弱性評定	AVA_VAN.4 方法的脆弱性分析

表 6　EAL5

8.8 評価保証レベル 6(EAL6) - 準形式的検証済み設計、及びテスト

目的

123 EAL6 は、重大なリスクに対して価値の高い資産を保護するためのプレミアム TOE を作り出すために、セキュリティエンジニアリング技法の厳格な開発環境への適用から、高い保証を開発者が得られるようにする。

124 そこで、EAL6 は、保護される資産の価値が追加コストを正当化するリスクの高い状況で適用するセキュリティ TOE の開発に適用される。

保証コンポーネント

125 **EAL6** は、完全なセキュリティターゲット及びその ST 内の SFR の分析により保証を提供する。この分析は、セキュリティのふるまいを理解するために、機能と完全なインタフェースの仕様、ガイダンス証拠資料、TOE の設計、及び実装を使用して行われる。**追加の保証が、選択 TOE セキュリティ方針の形式的モデル、及び機能仕様と TOE 設計の準形式的表現を通して得られる。**また、モジュール化され、**階層化された** TSF 設計も必要となる。

126 分析は、TSF の独立テスト、機能仕様に基づく開発者テストの証拠、TOE 設計、開発者テスト結果の選択的で独立した確認、及び**高い**攻撃能力を持つ侵入攻撃者に対する抵抗力を実証する独立した脆弱性分析によってサポートされる。

127 また、**EAL6** は、**構造化された**開発**プロセス**の使用、**開発**環境管理の使用、**完全な**自動化を含む包括的 TOE 構成管理の使用、及びセキュアな配付手続きの証拠を通して保証を提供する。

128 この EAL は、**さらなる包括的分析**、実装の**構造化された表現**、さらなる**アーキテクチャ構造(例えば階層化)**、**さらに包括的な独立した脆弱性分析**、及び向上した**構成管理**と開発**環境管理**を要求することにより、**EAL5** からの有意義な保証の増加を表す。

保証クラス	保証コンポーネント
ADV: 開発	ADV_ARC.1 セキュリティアーキテクチャ記述
	ADV_FSP.5 追加の誤り情報を伴う完全な準形式的機能仕様
	ADV_IMP.2 TSF の実装
	ADV_INT.3 最小限複雑な内部
	ADV_SPM.1 形式的 TOE セキュリティ方針モデル
	ADV_TDS.5 完全な準形式的モジュール設計
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクル サポート	ALC_CMC.5 高度なサポート
	ALC_CMS.5 開発ツールの CM 範囲
	ALC_DEL.1 配付手続き
	ALC_DVS.2 セキュリティ手段の十分性
	ALC_LCD.1 開発者によるライフサイクルモデルの定義
	ALC_TAT.3 実装標準への準拠 - すべての部分
ASE: セキュリティ ターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様
ATE: テスト	ATE_COV.3 カバレージの厳格な分析
	ATE_DPT.3 テスト: モジュール設計
	ATE_FUN.2 順序付けられた機能テスト
	ATE_IND.2 独立テスト - サンプル
AVA: 脆弱性評定	AVA_VAN.5 高度な方法的脆弱性分析

表 7　EAL6

8.9 評価保証レベル 7(EAL7) - 形式的検証済み設計、及びテスト

目的

129 EAL7 は、リスクが非常に高い状況及び/または資産の高い価値によってさらに高いコストが正当化されるところで適用するセキュリティTOEの開発に適用される。現在、EAL7の実際的な適用は、広範な形式的分析に従うセキュリティ機能性が強く重要視されているTOEに限られる。

保証コンポーネント

130 **EAL7** は、完全なセキュリティターゲット及びその ST 内の SFR の分析により保証を提供する。この分析は、セキュリティのふるまいを理解するために、機能と完全なインタフェースの仕様、ガイダンス証拠資料、TOE の設計、及び**構造化された**実装の**提示**を使用して行われる。追加の保証が、選択 TOE セキュリティ方針の形式的モデル、及び機能仕様と TOE 設計の準形式的表現を通して得られる。モジュール化され、**階層化された、簡潔な** TSF 設計も必要となる。

131 分析は、TSF の独立テスト、機能仕様に基づく開発者テストの証拠、TOE 設計と**実装表現**、開発者テスト結果の**完全で**独立した確認、及び高い攻撃能力を持つ侵入攻撃者に対する抵抗力を実証する独立した脆弱性分析によってサポートされる。

132 また、**EAL7** は、構造化開発プロセスの使用、開発環境管理の使用、完全な自動化を含む包括的な TOE 構成管理の使用、及びセキュアな配付手続きの証拠を通して保証を提供する。

133 この EAL は、**形式的表現と形式的対応、及び包括的テストを使用する**さらに包括的な分析を要求することにより、**EAL6** からの有意義な保証の増加を表す。

保証クラス	保証コンポーネント
ADV: 開発	ADV_ARC.1 セキュリティアーキテクチャ記述
	ADV_FSP.6 追加の形式的仕様を伴う完全な準形式的機能仕様
	ADV_IMP.2 TSF の実装
	ADV_INT.3 最小限複雑な内部
	ADV_SPM.1 形式的 TOE セキュリティ方針モデル
	ADV_TDS.6 形式的な上位レベルの設計提示を伴う完全な準形式的モジュール設計
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクルサポート	ALC_CMC.5 高度なサポート
	ALC_CMS.5 開発ツールの CM 範囲
	ALC_DEL.1 配付手続き
	ALC_DVS.2 セキュリティ手段の十分性
	ALC_LCD.2 測定可能なライフサイクルモデル
	ALC_TAT.3 実装標準への準拠 - すべての部分
ASE: セキュリティターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様
ATE: テスト	ATE_COV.3 カバレージの厳格な分析
	ATE_DPT.4 テスト: 実装表現
	ATE_FUN.2 順序付けられた機能テスト
	ATE_IND.3 独立テスト - 完全
AVA: 脆弱性評定	AVA_VAN.5 高度な方法的脆弱性分析

表 8　EAL7

9 統合保証パッケージ

134 統合保証パッケージ(CAP)は、統合 TOE について、得られる保証のレベルと、そのレベルの保証を得るためのコスト及び可能性とを比較考量する段階的な尺度を提供する。

135 CAP には、CC パート 3 のファミリとコンポーネントが少数しか含まれないことに注意しなければならない。これは、すでに評価されたエンティティ(基本コンポーネントと依存コンポーネント)の評価結果に基づくという CAP の性質によるもので、CAP が有意義な望ましい保証を提供しないことを意味するものではない。

9.1 統合保証パッケージ(CAP)の概要

136 CAP は統合 TOE に適用される。統合 TOE は、コンポーネント TOE 評価が実施された(評価中)コンポーネントで構成される(付属書 B を参照のこと)。個別のコンポーネントは、EAL、または ST で特定された別の保証パッケージに対して認証される。統合 TOE の保証の基本レベルは EAL1 の適用により得られることが期待される。これは、一般的に公知に利用できるコンポーネントについての情報を使用して達成できる(EAL1 は、コンポーネントと統合 TOE の両方に対して、それらの仕様どおりに適用できる)。CAP は、統合 TOE に対して EAL1 より上の EAL を適用するよりも上位のレベルの保証を得る代替アプローチを提供する。

137 依存コンポーネントは、環境内の IT プラットフォーム要件を満たすために、以前に評価、認証された基本コンポーネントを使用して評価を受けることができるが、これによりコンポーネント間の相互作用の形式的な保証または統合の結果による脆弱性の持ち込みの可能性は提供されない。統合保証パッケージは、これらの相互作用を考慮し、より上位レベルの保証で、コンポーネント間のインタフェースそれ自体がテストのサブジェクトとなることを保証する。統合 TOE の脆弱性分析も、コンポーネントを統合した結果として脆弱性が持ち込まれる可能性を考慮して実行される。

138 表 9 は、CAP の要約を示している。列は、階層的に並べられた CAP のセットを表し、行は保証ファミリを表す。その結果として得られるマトリックスの各数字は、適用すべき特定の保証コンポーネントを識別している。

139 次の節に概説されているように、階層的に並べられた 3 つの統合保証パッケージが、統合 TOE の保証のレート付けのために CC に定義されている。それらは、各 CAP がそれより下位のすべての CAP よりも多くの保証を表すために、階層的に並べられている。CAP から CAP への保証の増加は、同じ保証ファミリから階層的に上位の保証コンポーネントへの置換(つまり、厳格性、適用範囲、及び/または深さを拡大する)及び他の保証ファミリからの保証コンポーネントの追加(つまり、新しい要件を追加する)によって達成される。このような増加によって、個々のコンポーネント TOE について得られる評価結果への影響を識別できるように、構成の分析が強化される。

140 これらの CAP は、この CC パート 3 の第 7 章に記述されている保証コンポーネントの適切な組み合わせからなる。さらに正確には、各 CAP は各保証ファミリから 1 つ以下のコンポーネントを取り込んでおり、あらゆるコンポーネントのすべての保証依存性を扱っている。

141 CAP では、最高でも拡張された基本的な攻撃能力を持つ攻撃者に対する抵抗力のみが考慮される。これは、ACO_DEV から提供される設計情報のレベルに起因しており、攻撃能力に関連付けられたいくつかの要因(統合 TOE の知識)を制限し、評価者が実行可能な脆弱性分析の厳格性に影響を与える。したがって、統合 TOE の保証レベルは制限されるが、統合 TOE 内の個別のコンポーネントの保証はかなり高くなる場合がある。

保証クラス	保証ファミリ	統合保証パッケージ別の保証コンポーネント		
		CAP-A	CAP-B	CAP-C
統合	ACO_COR	**1**	1	1
	ACO_CTT	**1**	**2**	2
	ACO_DEV	**1**	**2**	**3**
	ACO_REL	**1**	1	**2**
	ACO_VUL	**1**	**2**	**3**
ガイダンス文書	AGD_OPE	**1**	1	1
	AGD_PRE	**1**	1	1
ライフサイクルサポート	ALC_CMC	**1**	1	1
	ALC_CMS	**2**	2	2
	ALC_DEL			
	ALC_DVS			
	ALC_FLR			
	ALC_LCD			
	ALC_TAT			
セキュリティターゲット評価	ASE_CCL	**1**	1	1
	ASE_ECD	**1**	1	1
	ASE_INT	**1**	1	1
	ASE_OBJ	**1**	**2**	2
	ASE_REQ	**1**	**2**	2
	ASE_SPD		**1**	1
	ASE_TSS	**1**	1	1

表 9　統合保証レベルの要約

9.2 統合保証パッケージの詳細

142 次の節では CAP を定義する。その際、特定の要件とそれらの要件の一律的な特性との差異を、ボールド体を用いて強調する。

9.3 統合保証レベル A (CAP-A) - 構造的統合

目的

143 CAP-A は、統合 TOE が統合され、その結果である複合物の正しいセキュリティ運用に信頼が要求される場合に適用される。これには、依存コンポーネントからの設計情報とテスト結果の提供に関して依存コンポーネントの開発者に協力を求める必要があるが、基本コンポーネントの開発者の関与は要求されない。

144 そこで、CAP-A は、開発者または利用者が完全な開発記録を簡単に使用できない場合に、低レベルから中レベルの独立に保証されたセキュリティを必要とする環境に適用される。

保証コンポーネント

145 **CAP-A は、統合 TOE のセキュリティターゲットの分析によって保証を提供する。統合 TOE の ST 内の SFR は、セキュリティのふるまいを理解するために、コンポーネント TOE の評価からの出力(例えば、ST、ガイダンス証拠資料)及び統合 TOE 内のコンポーネント TOE 間のインタフェースの仕様を使って分析される。**

146 **分析は、依存情報に記述されているように依存コンポーネントが信頼する基本コンポーネントのインタフェースの独立テスト、依存情報、開発情報、及び統合の根拠に基づく開発者テストの証拠、及び開発者テスト結果の選択的で独立した確認によってサポートされる。分析は、評価者による統合 TOE の脆弱性レビューによってもサポートされる。**

147 **また CAP-A は、統合 TOE の一意の識別情報(つまり、IT TOE とガイダンス証拠資料)を通して保証を提供する。**

保証クラス	保証コンポーネント
ACO: 統合	ACO_COR.1 統合の根拠
	ACO_CTT.1 インタフェーステスト
	ACO_DEV.1 機能記述
	ACO_REL.1 基本依存情報
	ACO_VUL.1 統合の脆弱性レビュー
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクルサポート	ALC_CMC.1 TOE のラベル付け
	ALC_CMS.2 TOE の一部の CM 範囲
ASE: セキュリティターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.1 運用環境のセキュリティ対策方針
	ASE_REQ.1 主張されたセキュリティ要件
	ASE_TSS.1 TOE 要約仕様

表 10 CAP-A

9.4 統合保証レベル B (CAP-B) - 方式的統合

目的

148 CAP-B は、良心的な開発者が、統合 TOE に統合されたコンポーネント TOE 間の相互作用の影響をサブシステムレベルで理解することから最大の保証を得られるようにする一方で、基本コンポーネントの開発者に要求される関与を最小限に抑える。

149 CAP-B は、開発者または利用者が、中レベルで独立して保証されたセキュリティを必要とし、大幅なリエンジニアリングを行わずに統合 TOE とその開発の完全な調査を必要とする状況で適用される。

保証コンポーネント

150 **CAP-B** は、統合 TOE の**完全な**セキュリティターゲットの分析によって保証を提供する。統合 TOE の ST 内の SFR は、セキュリティのふるまいを理解するために、コンポーネント TOE の評価からの出力(例えば、ST、ガイダンス証拠資料)、コンポーネント TOE 間のインタフェースの仕様、及び統合**開発情報に含まれている TOE 設計(TSF サブシステムの記述)**を使って分析される。

151 分析は、依存情報**(TOE 設計も含む)**に記述されているように依存コンポーネントが信頼する基本コンポーネントのインタフェースの独立テスト、依存情報、開発情報、及び統合の根拠に基づく開発者テストの証拠、及び開発者テスト結果の選択的で独立した確認によってサポートされる。分析は、**基本的な攻撃能力を持つ攻撃者に対する抵抗力を実証する**評価者による統合 TOE の脆弱性**分析**によってもサポートされる。

152 **この CAP は、セキュリティ機能性のより完全なテストカバレージを要求することにより、CAP-A からの有意義な保証の増加を表す。**

保証クラス	保証コンポーネント
ACO: 統合	ACO_COR.1 統合の根拠
	ACO_CTT.2 厳格なインタフェーステスト
	ACO_DEV.2 設計の基本証拠
	ACO_REL.1 基本依存情報
	ACO_VUL.2 統合の脆弱性分析
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクルサポート	ALC_CMC.1 TOE のラベル付け
	ALC_CMS.2 TOE の一部の CM 範囲
ASE: セキュリティターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様

表 11 CAP-B

9.5 統合保証レベル C (CAP-C) - 方式的統合、テスト、及びレビュー

目的

153 CAP-C は、統合 TOE のコンポーネント間の相互作用の有効な分析（それは厳格ではあるが、基本コンポーネントのすべての評価証拠への完全なアクセスを必要としない）から、開発者が最大の保証を得られるようにする。

154 そこで、CAP-C は、開発者または利用者が従来の商品としての統合 TOE に独立して保証された中レベルから高レベルのセキュリティを必要とし、セキュリティ特有のエンジニアリングコストを追加で負担する用意ができている状況で適用される。

保証コンポーネント

155 **CAP-C** は、統合 TOE の完全なセキュリティターゲットの分析によって保証を提供する。統合 TOE の ST 内の SFR は、セキュリティのふるまいを理解するために、コンポーネント TOE の評価からの出力(例えば、ST、ガイダンス証拠資料)、コンポーネント TOE 間のインタフェースの仕様、及び統合開発情報に含まれている TOE 設計(TSF **モジュール**の記述)を使って分析される。

156 分析は、依存情報(TOE 設計を含む)に記述されているように依存コンポーネントが信頼する基本コンポーネントのインタフェースの独立テスト、依存情報、開発情報、及び統合の根拠に基づく開発者テストの証拠、及び開発者テスト結果の選択的で独立した確認によってサポートされる。分析は、**拡張された基本的な**攻撃能力を持つ攻撃者に対する抵抗力を実証する評価者による統合 TOE の脆弱性分析によってもサポートされる。

157 この CAP は、より多くの**設計記述及びより高い攻撃能力に対する抵抗力の実証**を要求することにより、**CAP-B** からの有意義な保証の増加を表す。

保証クラス	保証コンポーネント
ACO: 統合	ACO_COR.1 統合の根拠
	ACO_CTT.2 厳格なインタフェーステスト
	ACO_DEV.3 設計の詳細証拠
	ACO_REL.2 依存情報
	ACO_VUL.3 拡張された基本的な統合の脆弱性分析
AGD: ガイダンス文書	AGD_OPE.1 利用者操作ガイダンス
	AGD_PRE.1 準備手続き
ALC: ライフサイクルサポート	ALC_CMC.1 TOE のラベル付け
	ALC_CMS.2 TOE の一部の CM 範囲
ASE: セキュリティターゲット評価	ASE_CCL.1 適合主張
	ASE_ECD.1 拡張コンポーネント定義
	ASE_INT.1 ST 概説
	ASE_OBJ.2 セキュリティ対策方針
	ASE_REQ.2 派生したセキュリティ要件
	ASE_SPD.1 セキュリティ課題定義
	ASE_TSS.1 TOE 要約仕様

表 12 CAP-C

10 APE クラス: プロテクションプロファイル評価

158 PP の評価は、PP が信頼でき内部的に一貫していること、及び PP が 1 つまたは複数の PP またはパッケージに基づいている場合に、それらの PP やパッケージを PP が正しく具体化していることを実証するために必要である。これらの特性は、ST または別の PP を記述するための基礎として PP を使用できるようにするために必要である。

159 CC パート 1 の附属書 A、B、及び C は、ここでの概念を明確にし、多くの例を提供するため、この章はこれらの附属書とともに使用されるべきである。

160 図 8 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

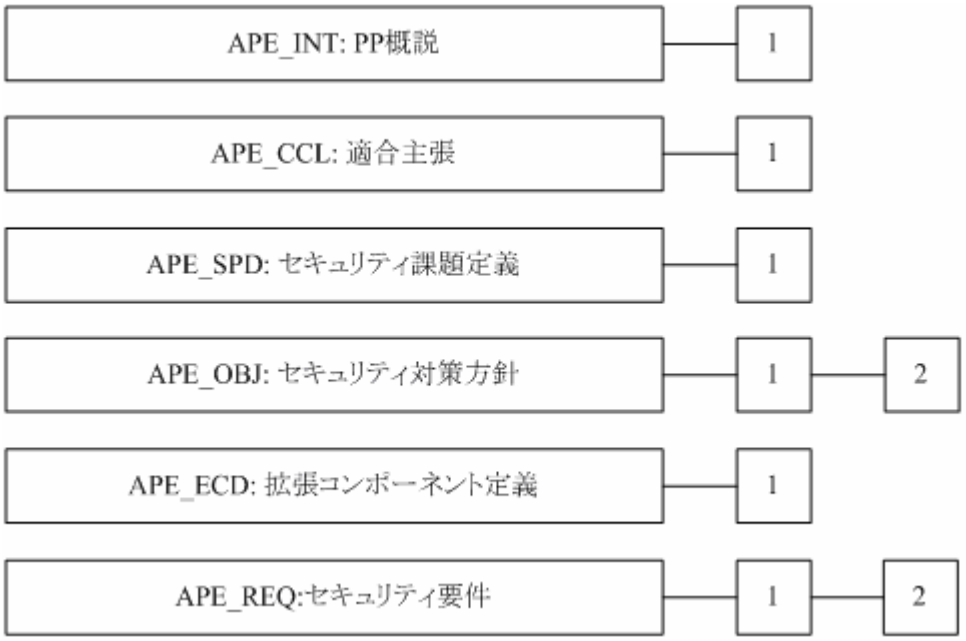

図 8 APE: プロテクションプロファイル評価クラスのコンポーネント構成

10.1 PP 概説(APE_INT)

目的

161 このファミリの目的は、TOE を順序立てて記述することである。

162 PP 概説の評価は、PP が正しく識別されていること、及び PP 参照と TOE 概要が相互に一貫していることを実証するために必要である。

APE_INT.1 PP 概説

依存性: なし

開発者アクションエレメント:

APE_INT.1.1D **開発者は、PP 概説を提供しなければならない。**

内容・提示エレメント:

APE_INT.1.1C **PP 概説は、PP 参照と TOE 概要を含めなければならない。**

APE_INT.1.2C **PP 参照は、PP を一意に識別しなければならない。**

APE_INT.1.3C **TOE 概要は、TOE の使用法及び主要なセキュリティ機能の特徴を要約しなければならない。**

APE_INT.1.4C **TOE 概要は、TOE 種別を識別しなければならない。**

APE_INT.1.5C **TOE 概要は、TOE が利用できる TOE 以外のハードウェア/ソフトウェア/ファームウェアを識別しなければならない。**

評価者アクションエレメント:

APE_INT.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

10.2 適合主張(APE_CCL)

目的

163 このファミリの目的は、適合主張の有効性を決定することである。さらに、このファミリは、ST 及び他の PP が PP に対する適合を主張する方法を特定する。

APE_CCL.1 適合主張

依存性: APE_INT.1 PP 概説
APE_ECD.1 拡張コンポーネント定義
APE_REQ.1 主張されたセキュリティ要件

開発者アクションエレメント:

APE_CCL.1.1D 開発者は、適合主張を提供しなければならない。

APE_CCL.1.2D 開発者は、適合主張根拠を提供しなければならない。

APE_CCL.1.3D 開発者は、適合ステートメントを提供しなければならない。

内容・提示エレメント:

APE_CCL.1.1C 適合主張は、**PP** が適合を主張する **CC** のバージョンを識別する **CC** 適合主張を含めなければならない。

APE_CCL.1.2C **CC** 適合主張は、**CC** パート **2** に対する **PP** の適合を **CC** パート **2** 適合または **CC** パート **2** 拡張として記述しなければならない。

APE_CCL.1.3C **CC** 適合主張は、**CC** パート **3** に対する **PP** の適合を **CC** パート **3** 適合または **CC** パート **3** 拡張として記述しなければならない。

APE_CCL.1.4C **CC** 適合主張は、拡張コンポーネント定義と一貫していなければならない。

APE_CCL.1.5C 適合主張は、**PP** が適合を主張する **PP** 及びセキュリティ要件パッケージをすべて識別しなければならない。

APE_CCL.1.6C 適合主張は、パッケージに対する **PP** の適合をパッケージ適合またはパッケージ追加として記述しなければならない。

APE_CCL.1.7C 適合主張根拠は、**TOE** 種別が、適合が主張されている **PP** 内の **TOE** 種別と一貫していることを実証しなければならない。

APE_CCL.1.8C 適合主張根拠は、セキュリティ課題定義のステートメントが、適合が主張されている **PP** 内のセキュリティ課題定義のステートメントと一貫していることを実証しなければならない。

APE_CCL.1.9C 適合主張根拠は、セキュリティ対策方針のステートメントが、適合が主張されている **PP** 内のセキュリティ対策方針のステートメントと一貫していることを実証しなければならない。

APE_CCL.1.10C 適合主張根拠は、セキュリティ要件のステートメントが、適合が主張されている **PP** 内のセキュリティ要件のステートメントと一貫していることを実証しなければならない。

APE_CCL.1.11C 適合ステートメントは、**PP** に対する任意の **PP/ST** に必要とされる適合を、正確 **PP** 適合または論証 **PP** 適合として記述しなければならない。

評価者アクションエレメント:

APE_CCL.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

10.3 セキュリティ課題定義(APE_SPD)

目的

164 PP のこの部分は、TOE 及び TOE の運用環境によって対処されるセキュリティ課題を定義する。

165 セキュリティ課題定義の評価は、TOE 及び TOE の運用環境によって対処されることが意図されているセキュリティ課題が明確に定義されていることを実証するために必要である。

APE_SPD.1 セキュリティ課題定義

依存性: なし

開発者アクションエレメント:

APE_SPD.1.1D 開発者は、セキュリティ課題定義を提供しなければならない。

内容・提示エレメント:

APE_SPD.1.1C セキュリティ課題定義は、脅威を記述しなければならない。

APE_SPD.1.2C すべての脅威は、脅威エージェント、資産、及び有害なアクションの観点から記述しなければならない。

APE_SPD.1.3C セキュリティ課題定義は、OSP を記述しなければならない。

APE_SPD.1.4C セキュリティ課題定義は、TOE の運用環境についての前提条件を記述しなければならない。

評価者アクションエレメント:

APE_SPD.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

10.4 セキュリティ対策方針(APE_OBJ)

目的

166 セキュリティ対策方針は、セキュリティ課題定義(APE_SPD)ファミリを通して定義されるセキュリティ課題に対して意図される対応の簡潔なステートメントである。

167 セキュリティ対策方針の評価は、セキュリティ対策方針が適切かつ完全にセキュリティ課題定義を扱うこと、及び TOE とその運用環境の間でのこの課題に対する分担が明確に定義されていることを実証するために必要である。

コンポーネントのレベル付け

168 このファミリのコンポーネントは、運用環境のセキュリティ対策方針のみを記述しているのか、または TOE のセキュリティ対策方針も含めて記述しているのかによって、レベル付けされている。

APE_OBJ.1 運用環境のセキュリティ対策方針

依存性: なし

開発者アクションエレメント:

APE_OBJ.1.1D **開発者は、セキュリティ対策方針のステートメントを提供しなければならない。**

内容・提示エレメント:

APE_OBJ.1.1C **セキュリティ対策方針のステートメントは、運用環境のセキュリティ対策方針を記述しなければならない。**

評価者アクションエレメント:

APE_OBJ.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

APE_OBJ.2 セキュリティ対策方針

依存性: APE_SPD.1 セキュリティ課題定義

開発者アクションエレメント:

APE_OBJ.2.1D 開発者は、セキュリティ対策方針のステートメントを提供しなければならない。

APE_OBJ.2.2D **開発者は、セキュリティ対策方針根拠を提供しなければならない。**

内容・提示エレメント:

APE_OBJ.2.1C セキュリティ対策方針のステートメントは、**TOE** のセキュリティ対策方針及び運用環境の**セキュリティ対策方針**を記述しなければならない。

APE_OBJ.2.2C **セキュリティ対策方針根拠は、TOE の各セキュリティ対策方針をそのセキュリティ対策方針によって対抗される脅威及びそのセキュリティ対策方針によって実施される OSP までさかのぼらなければならない。**

APE_OBJ.2.3C セキュリティ対策方針根拠は、運用環境の各セキュリティ対策方針をそのセキュリティ対策方針によって対抗される脅威、そのセキュリティ対策方針によって実施される **OSP**、及びそのセキュリティ対策方針によって充足される前提条件にまでさかのぼらなければならない。

APE_OBJ.2.4C セキュリティ対策方針根拠は、セキュリティ対策方針がすべての脅威に対抗することを実証しなければならない。

APE_OBJ.2.5C セキュリティ対策方針根拠は、セキュリティ対策方針がすべての **OSP** を実施することを実証しなければならない。

APE_OBJ.2.6C セキュリティ対策方針根拠は、運用環境のセキュリティ対策方針がすべての前提条件を充足することを実証しなければならない。

評価者アクションエレメント:

APE_OBJ.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

10.5 拡張コンポーネント定義(APE_ECD)

目的

169 拡張セキュリティ要件は、CC パート 2 または CC パート 3 のコンポーネントではなく、拡張コンポーネント、つまり PP の作成者によって定義されるコンポーネントに基づく要件である。

170 拡張コンポーネント定義の評価は、拡張コンポーネントが明確で曖昧さがないこと、及びそれらが必要であること、つまり既存の CC パート 2 または CC パート 3 のコンポーネントを使用して明白に表現できる可能性はなかったことを決定するために必要である。

APE_ECD.1 拡張コンポーネント定義

依存性: なし

開発者アクションエレメント:

APE_ECD.1.1D 開発者は、セキュリティ要件のステートメントを提供しなければならない。

APE_ECD.1.2D 開発者は、拡張コンポーネント定義を提供しなければならない。

内容・提示エレメント:

APE_ECD.1.1C セキュリティ要件のステートメントは、すべての拡張セキュリティ要件を識別しなければならない。

APE_ECD.1.2C 拡張コンポーネント定義は、各拡張セキュリティ要件に対応する拡張コンポーネントを定義しなければならない。

APE_ECD.1.3C 拡張コンポーネント定義は、各拡張コンポーネントが既存の CC コンポーネント、ファミリ、及びクラスにどのように関連するかを記述しなければならない。

APE_ECD.1.4C 拡張コンポーネント定義は、提示モデルとして既存の CC コンポーネント、ファミリ、クラス、及び方法を使用しなければならない。

APE_ECD.1.5C 拡張コンポーネントは、エレメントに対する適合または非適合を実証できるように、評価可能で客観的なエレメントで構成されていなければならない。

評価者アクションエレメント:

APE_ECD.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

APE_ECD.1.2E 評価者は、拡張コンポーネントが既存のコンポーネントを使用して明確には表現できないことを*確認しなければならない*。

10.6 セキュリティ要件(APE_REQ)

目的

171 SFR は、TOE に期待されるセキュリティのふるまいについての、明確で曖昧さがなく十分に定義された記述となる。SAR は、TOE で保証を得るために採用される期待されるアクティビティについての、明確で曖昧さがなく十分に定義された記述となる。

172 セキュリティ要件の評価は、それらの要件が明確で曖昧さがなく十分に定義されていることを保証するために必要である。

コンポーネントのレベル付け

173 このファミリのコンポーネントは、そのままで主張されているのか、または SFR が TOE のセキュリティ対策方針から導き出されているのかによって、レベル付けされている。

APE_REQ.1 主張されたセキュリティ要件

依存性: APE_ECD.1 拡張コンポーネント定義

開発者アクションエレメント:

APE_REQ.1.1D 開発者は、セキュリティ要件のステートメントを提供しなければならない。

APE_REQ.1.2D 開発者は、セキュリティ要件根拠を提供しなければならない。

内容・提示エレメント:

APE_REQ.1.1C セキュリティ要件のステートメントは **SFR** 及び **SAR** を記述しなければならない。

APE_REQ.1.2C **SFR** と **SAR** で使用されるすべてのサブジェクト、オブジェクト、操作、セキュリティ属性、外部のエンティティ及びその他の用語が定義されなければならない。

APE_REQ.1.3C セキュリティ要件のステートメントは、セキュリティ要件のすべての操作を識別しなければならない。

APE_REQ.1.4C すべての操作は正しく実行しなければならない。

APE_REQ.1.5C セキュリティ要件の各依存性が満たされていなければならない。また、満たされない依存性がある場合は、セキュリティ要件根拠によってそのことが正当化されなければならない。

APE_REQ.1.6C セキュリティ要件のステートメントは、内部的に一貫していなければならない。

評価者アクションエレメント:

APE_REQ.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

APE_REQ.2 導き出されたセキュリティ要件

依存性: ASE_OBJ.2 セキュリティ対策方針
APE_ECD.1 拡張コンポーネント定義

開発者アクションエレメント:

APE_REQ.2.1D 開発者は、セキュリティ要件のステートメントを提供しなければならない。

APE_REQ.2.2C 開発者は、セキュリティ要件根拠を提供しなければならない。

内容・提示エレメント:

APE_REQ.2.1C セキュリティ要件のステートメントは SFR 及び SAR を記述しなければならない。

APE_REQ.2.2C SFR と SAR で使用されるすべてのサブジェクト、オブジェクト、操作、セキュリティ属性、外部のエンティティ及びその他の用語が定義されなければならない。

APE_REQ.2.3C セキュリティ要件のステートメントは、セキュリティ要件のすべての操作を識別しなければならない。

APE_REQ.2.4C すべての操作は正しく実行しなければならない。

APE_REQ.2.5C セキュリティ要件の各依存性が満たされていなければならない。また、満たされない依存性がある場合は、セキュリティ要件根拠によってそのことが正当化されなければならない。

APE_REQ.2.6C **セキュリティ要件根拠は、各 SFR を TOE のセキュリティ対策方針にまでさかのぼらなければならない。**

APE_REQ.2.7C **セキュリティ要件根拠は、SFR が TOE のすべてのセキュリティ対策方針を満たすことを実証しなければならない。**

APE_REQ.2.8C **セキュリティ要件根拠は、なぜ SAR が選ばれたかを説明しなければならない。**

APE_REQ.2.9C セキュリティ要件のステートメントは、内部的に一貫していなければならない。

評価者アクションエレメント:

APE_REQ.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

11 ASE クラス: セキュリティターゲット評価

174 ST の評価は、ST が信頼でき内部的に一貫していること、及び ST が 1 つまたは複数の PP またはパッケージに基づいている場合に、それらの PP やパッケージを ST が正しく具体化していることを実証するために必要である。これらの特性は、ST を TOE 評価の基礎として使用できるようにするために必要である。

175 CC パート 1 の附属書 A、B、及び C は、ここでの概念を明確にし、多くの例を提供するため、この章はこれらの附属書とともに使用されるべきである。

176 図 9 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

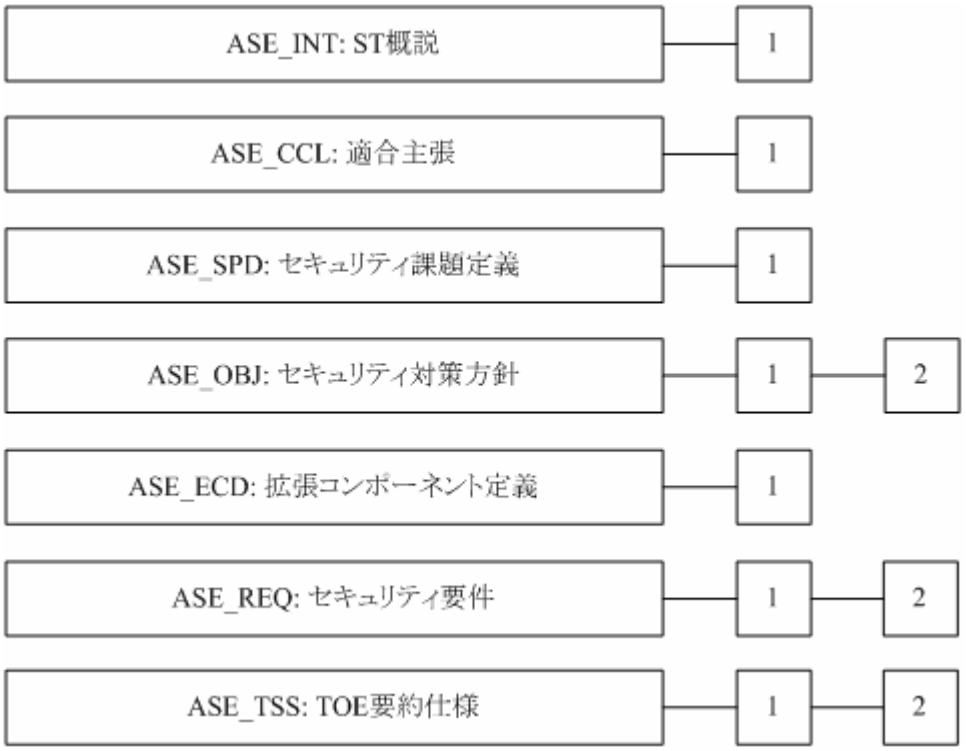

図 9 ASE: セキュリティターゲット評価クラスのコンポーネント構成

11.1 ST 概説(ASE_INT)

目的

177 このファミリの目的は、TOE を、TOE 参照、TOE 概要、及び TOE 記述の 3 つの抽象レベルで順序立てて記述することである。

178 ST 概説の評価は、ST と TOE が正しく識別されていること、TOE が 3 つの抽象レベルで正しく記述されていること、及びその 3 つの記述が互いに一貫していることを実証するために必要である。

ASE_INT.1 ST 概説

依存性: なし

開発者アクションエレメント:

ASE_INT.1.1D **開発者は、ST 概説を提供しなければならない。**

内容・提示エレメント:

ASE_INT.1.1C **ST 概説は、ST 参照、TOE 参照、TOE 概要、及び TOE 記述を含めなければならない。**

ASE_INT.1.2C **ST 参照は、ST を一意に識別しなければならない。**

ASE_INT.1.3C **TOE 参照は、TOE を識別しなければならない。**

ASE_INT.1.4C **TOE 概要は、TOE の使用法及び主要なセキュリティ機能の特徴を要約しなければならない。**

ASE_INT.1.5C **TOE 概要は、TOE 種別を識別しなければならない。**

ASE_INT.1.6C **TOE 概要は、TOE に必要な TOE 以外のハードウェア/ソフトウェア/ファームウェアを識別しなければならない。**

ASE_INT.1.7C **TOE 記述は、TOE の物理的範囲を記述しなければならない。**

ASE_INT.1.8C **TOE 記述は、TOE の論理的範囲を記述しなければならない。**

評価者アクションエレメント:

ASE_INT.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ASE_INT.1.2E **評価者は、TOE 参照、TOE 概要、及び TOE 記述が相互に一貫していることを*確認しなければならない*。**

11.2 適合主張(ASE_CCL)

目的

179 このファミリの目的は、適合主張の有効性を決定することである。さらに、このファミリは、ST が PP に対する適合を主張する方法を特定する。

ASE_CCL.1 適合主張

依存性: ASE_INT.1 ST 概説
ASE_ECD.1 拡張コンポーネント定義
ASE_REQ.1 主張されたセキュリティ要件

開発者アクションエレメント:

ASE_CCL.1.1D 開発者は、適合主張を提供しなければならない。

ASE_CCL.1.2D 開発者は、適合主張根拠を提供しなければならない。

内容・提示エレメント:

ASE_CCL.1.1C 適合主張は、**ST** と **TOE** が適合を主張する **CC** のバージョンを識別する **CC** 適合主張を含めなければならない。

ASE_CCL.1.2C **CC** 適合主張は、**CC** パート **2** に対する **ST** の適合を **CC** パート **2** 適合または **CC** パート **2** 拡張として記述しなければならない。

ASE_CCL.1.3C **CC** 適合主張は、**CC** パート **3** に対する **ST** の適合を **CC** パート **3** 適合または **CC** パート **3** 拡張として記述しなければならない。

ASE_CCL.1.4C **CC** 適合主張は、拡張コンポーネント定義と一貫していなければならない。

ASE_CCL.1.5C 適合主張は、**ST** が適合を主張する **PP** 及びセキュリティ要件パッケージをすべて識別しなければならない。

ASE_CCL.1.6C 適合主張は、パッケージへの **ST** の適合をパッケージ適合またはパッケージ追加として記述しなければならない。

ASE_CCL.1.7C 適合主張根拠は、**TOE** 種別が、適合が主張されている **PP** 内の **TOE** 種別と一貫していることを実証しなければならない。

ASE_CCL.1.8C 適合主張根拠は、セキュリティ課題定義のステートメントが、適合が主張されている **PP** 内のセキュリティ課題定義のステートメントと一貫していることを実証しなければならない。

ASE_CCL.1.9C 適合主張根拠は、セキュリティ対策方針のステートメントが、適合が主張されている **PP** 内のセキュリティ対策方針のステートメントと一貫していることを実証しなければならない。

ASE_CCL.1.10C 適合主張根拠は、セキュリティ要件のステートメントが、適合が主張されている **PP** 内のセキュリティ要件のステートメントと一貫していることを実証しなければならない。

評価者アクションエレメント:

ASE_CCL.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

11.3 セキュリティ課題定義(ASE_SPD)

目的

180 ST のこの部分は、TOE 及び TOE の運用環境によって対処されるセキュリティ課題を定義する。

181 セキュリティ課題定義の評価は、TOE 及び TOE の運用環境によって対処されることが意図されているセキュリティ課題が明確に定義されていることを実証するために必要である。

ASE_SPD.1 セキュリティ課題定義

依存性: なし

開発者アクションエレメント:

ASE_SPD.1.1D 開発者は、セキュリティ課題定義を提供しなければならない。

内容・提示エレメント:

ASE_SPD.1.1C セキュリティ課題定義は、脅威を記述しなければならない。

ASE_SPD.1.2C すべての脅威は、脅威エージェント、資産、及び有害なアクションの観点から記述しなければならない。

ASE_SPD.1.3C セキュリティ課題定義は、**OSP** を記述しなければならない。

ASE_SPD.1.4C セキュリティ課題定義は、**TOE** の運用環境についての前提条件を記述しなければならない。

評価者アクションエレメント:

ASE_SPD.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

11.4 セキュリティ対策方針(ASE_OBJ)

目的

182 セキュリティ対策方針は、セキュリティ課題定義(ASE_SPD)ファミリを通して定義されるセキュリティ課題に対して意図される対応の簡潔なステートメントである。

183 セキュリティ対策方針の評価は、セキュリティ対策方針が適切かつ完全にセキュリティ課題定義を扱うこと、及び TOE とその運用環境の間でのこの課題に対する分担が明確に定義されていることを実証するために必要である。

コンポーネントのレベル付け

184 このファミリのコンポーネントは、運用環境のセキュリティ対策方針のみを記述しているのか、または TOE のセキュリティ対策方針も含めて記述しているのかによって、レベル付けされている。

ASE_OBJ.1 運用環境のセキュリティ対策方針

依存性: なし

開発者アクションエレメント:

ASE_OBJ.1.1D 開発者は、セキュリティ対策方針のステートメントを提供しなければならない。

内容・提示エレメント:

ASE_OBJ.1.1C セキュリティ対策方針のステートメントは、運用環境のセキュリティ対策方針を記述しなければならない。

評価者アクションエレメント:

ASE_OBJ.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ASE_OBJ.2 セキュリティ対策方針

依存性: ASE_SPD.1 セキュリティ課題定義

開発者アクションエレメント:

ASE_OBJ.2.1D 開発者は、セキュリティ対策方針のステートメントを提供しなければならない。

ASE_OBJ.2.2D 開発者は、セキュリティ対策方針根拠を提供しなければならない。

内容・提示エレメント:

ASE_OBJ.2.1C セキュリティ対策方針のステートメントは、**TOE** のセキュリティ対策方針及び運用環境のセキュリティ対策方針を記述しなければならない。

ASE_OBJ.2.2C セキュリティ対策方針根拠は、**TOE** の各セキュリティ対策方針をそのセキュリティ対策方針によって対抗される脅威及びそのセキュリティ対策方針によって実施される **OSP** までさかのぼらなければならない。

ASE_OBJ.2.3C **セキュリティ対策方針根拠は、運用環境の各セキュリティ対策方針をそのセキュリティ対策方針によって対抗される脅威、そのセキュリティ対策方針によって実施される OSP、及びそのセキュリティ対策方針によって充足される前提条件にまでさかのぼらなければならない。**

ASE_OBJ.2.4C セキュリティ対策方針**根拠**は、セキュリティ対策方針が**すべての脅威に対抗することを実証**しなければならない。

ASE_OBJ.2.5C **セキュリティ対策方針根拠は、セキュリティ対策方針がすべての OSP を実施することを実証しなければならない。**

ASE_OBJ.2.6C **セキュリティ対策方針根拠は、運用環境のセキュリティ対策方針がすべての前提条件を充足することを実証しなければならない。**

評価者アクションエレメント:

ASE_OBJ.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

11.5 拡張コンポーネント定義(ASE_ECD)

目的

185 拡張セキュリティ要件は、CC パート 2 または CC パート 3 のコンポーネントではなく、拡張コンポーネント、つまり ST の作成者によって定義されるコンポーネントに基づく要件である。

186 拡張コンポーネント定義の評価は、拡張コンポーネントが明確で曖昧さがないこと、及びそれらが必要であること、つまり既存の CC パート 2 または CC パート 3 のコンポーネントを使用して明白に表現できないことを決定するために必要である。

ASE_ECD.1 拡張コンポーネント定義

依存性: なし

開発者アクションエレメント:

ASE_ECD.1.1D 開発者は、セキュリティ要件のステートメントを提供しなければならない。

ASE_ECD.1.2D 開発者は、拡張コンポーネント定義を提供しなければならない。

内容・提示エレメント:

ASE_ECD.1.1C セキュリティ要件のステートメントは、すべての拡張セキュリティ要件を識別しなければならない。

ASE_ECD.1.2C 拡張コンポーネント定義は、各拡張セキュリティ要件に対応する拡張コンポーネントを定義しなければならない。

ASE_ECD.1.3C 拡張コンポーネント定義は、各拡張コンポーネントが既存の CC コンポーネント、ファミリ、及びクラスにどのように関連するかを記述しなければならない。

ASE_ECD.1.4C 拡張コンポーネント定義は、提示モデルとして既存の CC コンポーネント、ファミリ、クラス、及び方法を使用しなければならない。

ASE_ECD.1.5C 拡張コンポーネントは、エレメントに対する適合または非適合を実証できるように、評価可能で客観的なエレメントで構成されていなければならない。

評価者アクションエレメント:

ASE_ECD.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ASE_ECD.1.2E 評価者は、拡張コンポーネントが既存のコンポーネントを使用して明確には表現できないことを*確認しなければならない*。

11.6 セキュリティ要件(ASE_REQ)

目的

187 SFR は、TOE に期待されるセキュリティのふるまいについての、明確で曖昧さがなく十分に定義された記述となる。SAR は、TOE で保証を得るために採用される期待されるアクティビティについての、明確で曖昧さのない標準的な記述となる。

188 セキュリティ要件の評価は、それらの要件が明確で曖昧さがなく十分に定義されていることを保証するために必要である。

コンポーネントのレベル付け

189 このファミリのコンポーネントは、それらがどこからも派生せずに主張されているかどうかによってレベル付けされている。

ASE_REQ.1 主張されたセキュリティ要件

依存性: ASE_ECD.1 拡張コンポーネント定義

開発者アクションエレメント:

ASE_REQ.1.1D 開発者は、セキュリティ要件のステートメントを提供しなければならない。

ASE_REQ.1.2D 開発者は、セキュリティ要件根拠を提供しなければならない。

内容・提示エレメント:

ASE_REQ.1.1C セキュリティ要件のステートメントは、**SFR** 及び **SAR** を記述しなければならない。

ASE_REQ.1.2C **SFR** と **SAR** で使用されるすべてのサブジェクト、オブジェクト、操作、セキュリティ属性、外部のエンティティ及びその他の用語は、定義されなければならない。

ASE_REQ.1.3C セキュリティ要件のステートメントは、セキュリティ要件のすべての操作を識別しなければならない。

ASE_REQ.1.4C すべての操作は、正しく実行されなければならない。

ASE_REQ.1.5C セキュリティ要件の各依存性は、満たされていなければならない。また、満たされない場合は、セキュリティ要件根拠によってそのことが正当化されなければならない。

ASE_REQ.1.6C セキュリティ要件のステートメントは、内部的に一貫していなければならない。

評価者アクションエレメント:

ASE_REQ.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ASE_REQ.2 導き出されたセキュリティ要件

依存性: ASE_OBJ.2 セキュリティ対策方針
ASE_ECD.1 拡張コンポーネント定義

開発者アクションエレメント:

ASE_REQ.2.1D 開発者は、セキュリティ要件のステートメントを提供しなければならない。

ASE_REQ.2.2D 開発者は、セキュリティ要件根拠を提供しなければならない。

内容・提示エレメント:

ASE_REQ.2.1C セキュリティ要件のステートメントは、SFR 及び SAR を記述しなければならない。

ASE_REQ.2.2C SFR と SAR で使用されるすべてのサブジェクト、オブジェクト、操作、セキュリティ属性、外部のエンティティ及びその他の用語は、定義されなければならない。

ASE_REQ.2.3C セキュリティ要件のステートメントは、セキュリティ要件のすべての操作を識別しなければならない。

ASE_REQ.2.4C すべての操作は、正しく実行されなければならない。

ASE_REQ.2.5C セキュリティ要件の各依存性は、満たされていなければならない。また、満たされない場合は、セキュリティ要件根拠によってそのことが正当化されなければならない。

ASE_REQ.2.6C **セキュリティ要件根拠は、各 SFR を TOE のセキュリティ対策方針にまでさかのぼらなければならない。**

ASE_REQ.2.7C **セキュリティ要件根拠は、SFR が TOE のすべてのセキュリティ対策方針を満たすことを実証しなければならない。**

ASE_REQ.2.8C **セキュリティ要件根拠は、なぜ SAR が選ばれたかを説明しなければならない。**

ASE_REQ.2.9C セキュリティ要件のステートメントは、内部的に一貫していなければならない。

評価者アクションエレメント:

ASE_REQ.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

11.7 TOE 要約仕様(ASE_TSS)

目的

190 TOE 要約仕様は、TOE がどのように実装されているかについて、評価者と潜在的消費者が概要を理解できるようにする。

191 TOE 要約仕様の評価は、以下の点について適切に記述されているかどうかを決定するために必要である:

- TOE が SFR をどのように満たすか;
- TOE が干渉、論理的な改ざん及びバイパスに対して自身をどのように保護するか。

また、TOE 要約仕様が TOE の他の叙述的記述と一貫しているかどうかを確認するためにも必要である。

コンポーネントのレベル付け

192 このファミリのコンポーネントは、TOE 要約仕様が、TOE がどのように SFR を満たすかのみを記述するために必要かどうか、または TOE が論理的な改ざんやバイパスから自身をどのように保護するか記述するためにも必要かどうかによってレベル付けされる。この追加の記述は、TOE セキュリティアーキテクチャに関する特定の問題が発生する可能性がある特殊な状況で使用することができる。

ASE_TSS.1 TOE 要約仕様

依存性: ASE_INT.1 ST 概説
ASE_REQ.1 主張されたセキュリティ要件

開発者アクションエレメント:

ASE_TSS.1.1D **開発者は、TOE 要約仕様を提供しなければならない。**

内容・提示エレメント:

ASE_TSS.1.1C **TOE 要約仕様は、TOE がどのように各 SFR を満たすかを記述しなければならない。**

評価者アクションエレメント:

ASE_TSS.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ASE_TSS.1.2E **評価者は、TOE 要約仕様が TOE 概要及び TOE 記述と一貫していることを*確認しなければならない*。**

ASE_TSS.2 アーキテクチャ設計要約を伴う TOE 要約仕様

依存性: ASE_INT.1 ST 概説
ASE_REQ.1 主張されたセキュリティ要件

開発者アクションエレメント:

ASE_TSS.2.1D 開発者は、TOE 要約仕様を提供しなければならない。

内容・提示エレメント:

ASE_TSS.2.1C TOE 要約仕様は、TOE がどのように各 SFR を満たすかを記述しなければならない。

ASE_TSS.2.2C **TOE 要約仕様は、TOE がどのように干渉や論理的な改ざんから自身を保護するかを記述しなければならない。**

ASE_TSS.2.3C **TOE 要約仕様は、TOE がどのようにバイパスから自身を保護するかを記述しなければならない。**

評価者アクションエレメント:

ASE_TSS.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ASE_TSS.2.2E 評価者は、TOE 要約仕様が TOE 概要及び TOE 記述と一貫していることを***確認しなければならない。***

12 ADV クラス: 開発

193 開発クラスの要件は、TOE に関する情報を提供する。この情報から得られた知識は、AVA クラス及び ATE クラスで記述されている TOE に対する脆弱性分析とテストを実施するための基礎として使用される。

194 開発クラスは、抽象化の様々なレベル及び形式で TSF を構造化し、表現するための要件で構成される 6 つのファミリを含んでいる。これらのファミリには、次のものが含まれる:

- SFR の設計及び実装の(様々な抽象レベルでの)記述に関する要件(ADV_FSP、ADV_TDS、ADV_IMP)
- ドメイン分離、TSF の自己保護、及びセキュリティ機能性の非バイパス性というアーキテクチャ指向の特徴の記述に関する要件(ADV_ARC)
- セキュリティ方針モデルに関する要件、及びセキュリティ方針モデルと機能仕様の間の対応付けに関する要件(ADV_SPM)
- モジュール化、階層化、複雑さの最小化などの側面に対応する TSF の内部構造に関する要件(ADV_INT)

195 TOE のセキュリティ機能性について証拠資料を提出する際に、2 つの特性を実証する必要がある。第 1 の特性は、セキュリティ機能性が正しく機能すること、つまり仕様どおりに動作することである。第 2 の特性は、おそらく簡単には実証できないが、セキュリティ機能性が破壊またはバイパスされかねない方法では TOE を使用できないようになっていることである。この 2 つの特性の分析にはやや異なるアプローチが必要になるため、ADV のファミリはそれらのアプローチをサポートするように構成されている。機能仕様(ADV_FSP)、TOE 設計(ADV_TDS)、実装表現(ADV_IMP)、セキュリティ方針モデル化(ADV_SPM)の各ファミリは、第 1 の特性つまりセキュリティ機能性の仕様を扱う。セキュリティアーキテクチャ(ADV_ARC)及び TSF 内部構造(ADV_INT)の各ファミリは、第 2 の特性、つまりセキュリティ機能性が破壊またはバイパスされないことを実証する TOE 設計の仕様を扱う。どちらの特性も実現される必要があることに注意するべきである。つまり、特性が満たされているという信頼が高いほど、TOE の信頼も高くなる。ファミリ内のコンポーネントは、コンポーネントの階層が上がるにつれて、より多くの保証が得られるように設計されている。

196 第 1 の特性を対象とするファミリのパラダイムは、設計分解の 1 つである。最上位レベルには、TSF のインタフェースに関する TSF の機能仕様がある(TSF に対するサービスの要求及びその結果の応答に関して TSF が*何を*行うかが記述される)。TSF が(求められる保証と TOE の複雑さに応じて)より小さな単位に分解され、TSF がその機能を*どのように*果たすかが(保証レベルに合った詳細レベルまで)記述され、TSF の実装が示される。セキュリティのふるまいの形式的なモデルが示される場合もある。分解のすべてのレベルが、その他すべてのレベルの完全性と正確さの決定に使用され、それによってレベルの相互サポートが保証される。種々の TSF 表現に関する要件は、複数のファミリに分けられて、PP/ST の作成者が必要な TSF 表現を特定できるようになっている。選択されたレベルによって、必要な保証/得られる保証が決まる。

197 図 10 は、ADV クラスの各種 TSF 表現、及びそれらと他のクラスとの関係を示している。この図が示すように、APE クラスと ASE クラスは、SFR と TOE セキュリティ対策方針の対応に関する要件を定義する。また、ASE クラスは、セキュリティ対策方針と SFR、及びどのように TOE が SFR を満たすか説明する TOE 要約仕様間の対応に関する要件についても定義する。ALC_CMC.5.2E のアクティビティには、ATE クラス及び AVA クラスでテストされる TSF が実際にすべての ADV 分解レベルで記述されているかどうかの検証が含まれる。

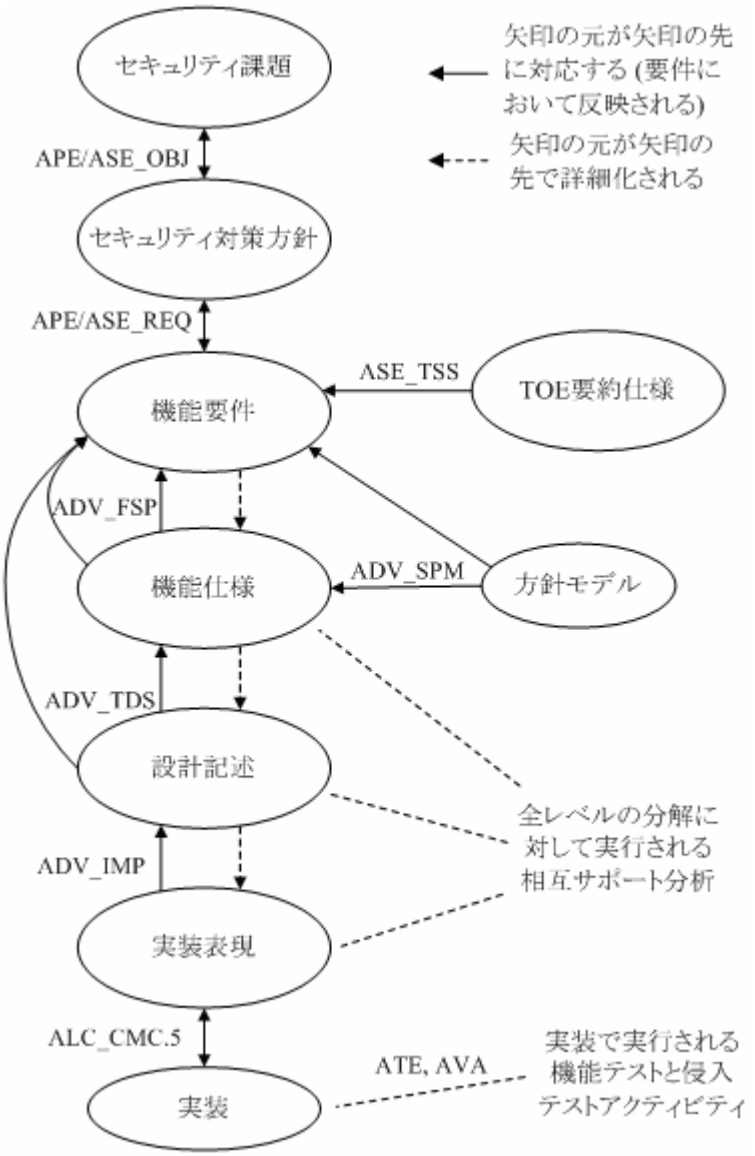

図 10 ADV 構造間及びそれらと他のファミリとの関係

198 これ以外の図 10 で示される対応関係は、すべて ADV クラスで定義される。セキュリティ方針モデル化(ADV_SPM)ファミリは、選択された SFR セキュリティ機能を形式的にモデル化するための要件、及び機能仕様と形式的モデルの間の対応を提供するための要件を定義する。TSF 表現に焦点を当てた各保証ファミリ(つまり、機能仕様(ADV_FSP)、TOE 設計(ADV_TDS)、及び実装表現(ADV_IMP))は、その TSF 表現を SFR に関係付ける要件を定義する。すべての分解は他のすべての分解を正確に反映する(つまり相互サポートする)必要がある。開発者はコンポーネントの最後の.C エレメントでの追跡を提供する。この要因に関連する保証は、特定レベルの分解が分析される間に、他のレベルの分解を(再帰的に)参照する方法で各レベルの分解を分析することによって得られる。評価者は 2 つ目の E エレメントの一部として対応を検証する。これらのレベルの分解から得られた理解は、機能テスト及び侵入テスト成果の基礎となる。

199 ADV_INT ファミリは、TSF の内部構造に関連し、TSF 表現の詳細化のプロセスには間接的にしか関連していないため、この図には示されていない。同様に、ADV_ARC ファミリは、TSF の表現ではなくそのアーキテクチャへの信頼に関連しているため、この図に示されていない。ADV_INT と ADV_ARC はどちらも、TOE がそのセキュリティ機能性を回避または破壊できないようになっているという特性の分析に関連する。

200 TOE セキュリティ機能(TSF)は、SFR の実施に必要となる TOE のすべての部分で構成される。TSF は、SFR を直接実施する機能性、及び、SFR を侵害する可能性のある機能性も含めた、SFR を直接実施しないが間接的に SFR の実施に寄与する機能性を含んでいる。これには、立ち上げ時に呼び出され、TSF をそのセキュアな初期状態にする TOE の各部分が含まれる。

201 ADV ファミリのコンポーネントの開発には、いくつかの重要な概念が採用された。これらの概念については、この節で簡単に紹介し、ファミリの適用上の注釈で詳しく説明する。

202 最も重要な概念の 1 つは、提供される情報が多いほど、セキュリティ機能性が 1)正しく実装され、2)損なわれず、3)バイパスされないことの保証が高まることである。これは、証拠資料が正確で他の証拠資料と一貫していることを検証し、テストアクティビティ(機能テストと侵入テストの両方)が包括的であることを保証するための情報を提供することで実現される。これは、ファミリのコンポーネントのレベル付けで反映される。一般に、コンポーネントは、提供される(さらにその後分析される)情報の量に基づいてレベル付けされる。

203 すべての TOE には当てはまらないが、TSF が複雑であるために、TSF のある部分に他の部分よりも厳しい検査が必要になることは一般的である。このような部分の判別はあいにくやや主観的であるため、保証レベルの増大に伴って、詳細な検査の必要な TSF の部分を判別する責任が開発者から評価者にシフトするように、用語及びコンポーネントが定義されている。この概念を表すのに役立つように、次の用語が導入されている。このクラスのファミリでは、TOE の SFR に関連する部分を表す際に、(つまり、機能仕様(ADV_FSP)、TOE 設計(ADV_TDS)、実装表現(ADV_IMP)の各ファミリに統合されているエレメントとワークユニットで)この用語が使用されることに注意する必要がある。他のファミリには一般的な概念(TOE の一部分に対する関心が他の部分より高いという概念)が適用されるが、その基準は、必要な保証を得るために異なる方法で表される。

204 TSF のすべての部分は *セキュリティに関連している*。これは、それらの部分が、SFR 及びドメイン分離と非バイパス性に関する要件で表されているとおりに TOE のセキュリティを保持しなければならないことを意味する。セキュリティ関連性の側面の 1 つに、TSF の一部分がセキュリティ要件を実施する程度がある。TOE の各部分が、セキュリティ要件の実施においてそれぞれに異なる役割を果たす(あるいは明らかな役割がない)ため、このことによって SFR の関連性の連続体が作成される。この連続体の一方の終端は、*SFR 実施*と呼ばれる TOE の部分である。このような各部分は、あらゆる SFR を TOE に実装するために直接的な役割を果たす。ここで言う SFR とは、ST に含まれている SFR のいずれかによって提供される機能性を指す。SFR 実施機能性 *での役割を果たす*という表現の定義は、定量的に表すことが不可能であることに注意するべきである。例えば、任意アクセス制御(DAC)メカニズムの実装の場合、ごく狭い視野で見た *SFR 実施*は、オブジェクトの属性に対してサブジェクトの属性を実際にチェックする数行のコードである。より広い視野で見ると、その数行のコードを含んだソフトウェアエンティティ(例えば C 関数)が含まれる。より広い視野には、C 関数のコール元も含まれる。これは、属性チェックによって返された決定を実施する責任をそれらのコール元が負うためである。さらに広い視野には、その C 関数のコールツリー(または使用される実装言語における同等のプログラミング構造)内のコードが含まれる(例えば、ファーストマッチアルゴリズムの実装でアクセス制御リストのエントリをソートするソート機能)。ある点で、コンポーネントはセキュリティ方針の実施にそれほど *関与*しなくなり、*サポート*の役割を果たすようになる。このようなコンポーネントは *SFR 支援*と呼ばれる。

205 SFR 支援機能性の特性の 1 つは、誤りなく動作することで正しい SFR 実装を保つと信頼されている点である。この機能性に SFR 実施機能性が依存している場合もあるが、一般にその依存性は、メモリ管理やバッファ管理などの機能レベルである。セキュリティ関連性の連続体における次の機能性は、*SFR 非干渉*と呼ばれる。この機能性は、SFR の実装では役割がなく、その環境のために TSF の一部分である場合が多い。例えば、オペレーティングシステム上で特権的なハードウェアモードで実行されるコードが挙げられる。このコードが損なわれると(あるいは悪意のあるコードで置き換えられると)、それが特権的なハードウェアモードで動作することによって SFR の正しい動作が損なわれる可能性があるため、このコードは TSF の一部とみなす必要がある。SFR 非干渉機能性の例には、処理速度を考慮してカーネルモードに実装される一連の浮動小数点演算が挙げられる。

206 アーキテクチャファミリ(セキュリティアーキテクチャ(ADV_ARC))は、TSF の信頼できる初期化、自己保護、及び非バイパス性の特性に基づいた、TOE の要件と分析を提供する。これらの特性は、存在しない場合、SFR を実装しているメカニズムの障害につながる可能性があるという点で、SFR に関連している。これらの特性に関連する機能性と設計は、その性質と分析要件が根本的に異なっていることから、上で説明した連続体の一部とは*みなされず*、別個に扱われる。

207 SFR の実装(SFR 実施機能性及び SFR 支援機能性)と、初期化、自己保護、非バイパス性に関係する、どちらかというと基本的な TOE のセキュリティ特性の実装との分析における違いは、SFR 関連機能性がある程度直接的に見え、比較的テストが容易であるのに対し、上で説明した特性には、はるかに広い範囲の機能性セットに対して様々な程度の分析が必要であるという点である。さらに、このような特性の分析の深さは、TOE の設計によって異なる。ADV ファミリは、初期化、自己保護、及び非バイパス性の各要件の分析のみを行う別のファミリ(セキュリティアーキテクチャ(ADV_ARC))でこれに対応し、その他のファミリで SFR を支援する機能性の分析を行うように構成されている。

208 複数の抽象レベルに別々の記述が必要な場合であっても、個々の TSF 表現を別個の文書として記述する必要はまったくない。実際、要求されているのはこれらの TSF 表現についての情報であって、結果としての文書構造ではないために、1 つの文書が複数の TSF 表現に対する証拠資料要件に合致する場合もある。複数の TSF 表現が 1 つの文書中に混在している場合、開発者は、文書のどの部分が、どの要件に合致しているかを示さなければならない。

209 3 種の仕様の様式(非形式的、準形式的、及び形式的)がこのクラスによって指定される。機能仕様及び TOE 設計証拠資料は、常に非形式的または準形式的のいずれかの様式で記述される。準形式的な様式は、非形式的表現に比べて、これらの文書での曖昧さが少ない。準形式的表現*に加えて*形式的仕様が必要となる場合がある。その意義は、TSF が複数の方法で記述されることにより、TSF が完全かつ正確に特定されているというさらなる保証が得られるという点にある。

210 非形式的仕様とは、自然言語によって普通に書かれる。ここで言う自然言語とは、(スペイン語や、ドイツ語、フランス語、英語、オランダ語など)通常の会話で用いられる言葉を示している。非形式的仕様は、その言語で通常用いられている(例: 文法や構文)規則として要求されること以外、表記や特別な制約は課されない。表記に関する制約は課されないものの、非形式的仕様では、文脈上、通常用いられる意味と異なる場合には、定められている用語の意味が定義されていなければならない。

211 準形式的文書と非形式的文書の違いは、形式/表現の点のみである。準形式的表記には、明示的な用語集や標準化された表現形式などが含まれる。準形式的仕様は、標準の表現テンプレートに書き込まれる。自然言語で記述される場合は、表現で用語が矛盾なく使用されるべきである。表現では、より構造化された言語/図が使用される場合もある(例えば、データフロー図、状態遷移図、E-R 図、データ構造図、プロセスやプログラムの構造図)。図または自然言語のどちらに基づいている場合でも、表現では一連の規則を使用しなければならない。用語集は、正確かつ一定して使用される単語を明示的に識別する。同様に、標準化された形式は、できる限り明確になるように文書を方式的に準備することに、最大限の注意が払われたことを示す。TSF の基本的に異なっている部分は、その準形式的表記規則及び表現様式が異なっていることがあるので注意する必要がある(ただし、異なっている「準形式的表記」の数が少ない場合)。この点はまだ*準形式的表現*の概念に適合している。

212 形式的仕様とは、数学的概念に基づいた表記によって書かれ、これに(非形式的な)補足説明が加わっているようなものをいう。これらの数学的概念は、表記の構文と意味、及び論理的な推論を助ける証明規則を定義するために用いられる。形式的表記をサポートする構文意味規則は、どのようにして曖昧さなくその構造を認識し、その意味を決定付けるかを定義するべきである。矛盾を引き出すのが不可能であることの証拠が必要であり、表記をサポートするすべての規則を定義または参照付けする必要がある。

213 図 11 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

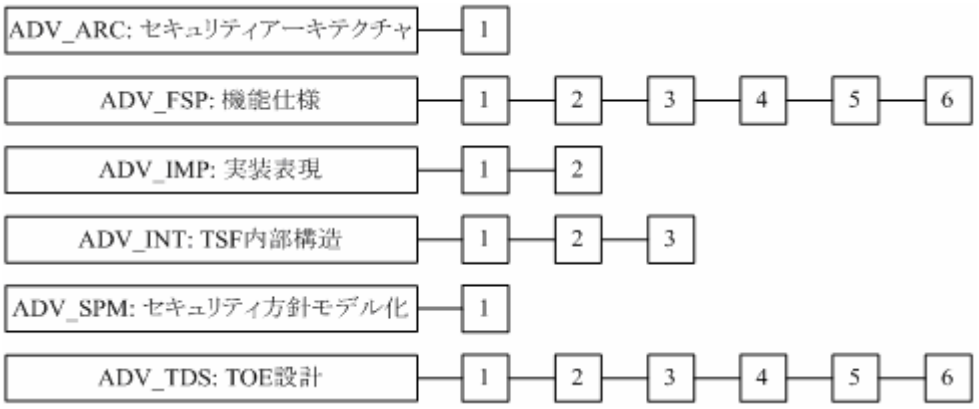

図 11 ADV: 開発クラスのコンポーネント構成

12.1 セキュリティアーキテクチャ(ADV_ARC)

目的

214 このファミリの目的は、TSF のセキュリティアーキテクチャの記述を、開発者が提供することである。この証拠に TSF に対して提示されるその他の証拠が加味されると、TSF が目的の特性を達成したことを確認する情報の分析が可能となる。セキュリティアーキテクチャの記述は、TOE のセキュリティ分析は TSF の検査により達成できるという暗黙的な主張をサポートする。適切なアーキテクチャがない場合は、TOE の機能性全体を調査する必要がある。

コンポーネントのレベル付け

215 このファミリは、ただ 1 つのコンポーネントからなる。

適用上の注釈

216 自己保護、ドメイン分離、及び非バイパス性の特性は、パート 2 の SFR で表現されているセキュリティ機能性とは区別される。これは、一般に自己保護や非バイパス性は、TSF に直接観察可能なインタフェースを持たないからである。これらはむしろ TOE 及び TSF の設計によって達成される TSF の特性であり、その設計の正しい実装によって実施される。

217 このファミリでのアプローチとしては、まず開発者が、上述の特性を示す TSF を設計し提供することと、TSF のこれらの特性を説明する証拠を(証拠資料の形で)提供する。この説明は、TOE 設計文書における TOE の SFR 実施エレメントの記述と同じ詳細レベルで提供される。評価者には、その証拠を調べ、TOE 及び TSF のために配付されるその他の証拠と組み合わせて、特性が達成されていることを決定する責任がある。

218 SFR を実装するセキュリティ機能性の仕様(機能仕様(ADV_FSP)及び TOE 設計(ADV_TDS)内)が、自己保護及び非バイパス性を実装する際に採用されるメカニズム(メモリ管理メカニズムなど)を必ずしも記述するとは限らない。このため、これらの要件が達成されていることの保証を提供するために必要な資料には、ADV_FSP 及び ADV_TDS に組み込まれた TSF の設計コンポーネント構成からは切り離された表現が適している。これは、このコンポーネントで要求されるセキュリティアーキテクチャ記述が、設計コンポーネント構成資料を参照または利用できないことを意味するものではないが、コンポーネント構成証拠資料内の詳細情報の大半は、セキュリティアーキテクチャ記述文書に提供されている論証とは無関係である。

219 アーキテクチャへの信頼の記述は、TSF が信頼される理由についての正当性を提供し、その SFR をすべて実施するという点で、開発者の脆弱性分析と考えることができる。信頼が特定のセキュリティメカニズムを通して達成される場合、これらは深さ(ATE_DTP)要件の一部としてテストされる。信頼がアーキテクチャのみを通して達成される場合、そのふるまいは AVA: 脆弱性評定要件の一部としてテストされる。

220 このファミリは、自己保護、ドメイン分離、非バイパス性の各原則を記述するセキュリティアーキテクチャ記述に関する要件で構成される。これには、これらの原則が、TSF の初期化に使用される TOE の部分によってどのようにサポートされるかについての記述が含まれる。

221 自己保護、ドメイン分離、及び非バイパス性のセキュリティアーキテクチャ特性に関する追加情報が、附属書 A.1「ADV_ARC: セキュリティアーキテクチャに関する補足資料」に記載されている。

ADV_ARC.1 セキュリティアーキテクチャ記述

依存性: ADV_FSP.1 基本機能仕様
ADV_TDS.1 基本設計

開発者アクションエレメント:

ADV_ARC.1.1D 開発者は、**TSF** のセキュリティ特性がバイパスされないように **TOE** を設計及び実装しなければならない。

ADV_ARC.1.2D 開発者は、**TSF** が信頼できない能動的なエンティティによって改ざんされるのを防ぐことができるように **TSF** を設計及び実装しなければならない。

ADV_ARC.1.3D 開発者は、**TSF** のセキュリティアーキテクチャ記述を提供しなければならない。

内容・提示エレメント:

ADV_ARC.1.1C セキュリティアーキテクチャ記述は、**TOE** 設計文書に記述されている **SFR** 実施抽象概念の記述に見合った詳細レベルでなければならない。

ADV_ARC.1.2C セキュリティアーキテクチャ記述は、**TSF** によって維持されるセキュリティドメインを、**SFR** と一貫する形で記述しなければならない。

ADV_ARC.1.3C セキュリティアーキテクチャ記述は、**TSF** の初期化プロセスのセキュリティがどのようにして確保されるのかを記述しなければならない。

ADV_ARC.1.4C セキュリティアーキテクチャ記述は、**TSF** が改ざんから自分自身を保護することを実証しなければならない。

ADV_ARC.1.5C セキュリティアーキテクチャ記述は、**TSF** が **SFR** 実施機能性のバイパスを防ぐことを実証しなければならない。

評価者アクションエレメント:

ADV_ARC.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

12.2 機能仕様(ADV_FSP)

目的

222 このファミリは、TSF インタフェース(TSFI)を記述する機能仕様に対して要件を課す。TSFI は、利用者が TSF からサービスを呼び出す(TSF によって処理されるデータを提供する方法で)ためのすべての手段、及びこれに対応するそれらのサービスの呼び出しへの応答で構成される。ただし、TSF がそれらのサービス要求を処理する方法や、TSF がその運用環境からサービスを呼び出す際の通信については記述*されない*。この情報は、TOE 設計(ADV_TDS)ファミリ及び依存コンポーネントの依存(ACO_REL)ファミリでそれぞれ扱われる。

223 このファミリは、TSF が主張された SFR をどのように満たしているかを評価者が把握できるようにすることで、直接的に保証を提供する。また、次の保証ファミリ及びクラスへの入力として、間接的にも保証を提供する:

- ADV_ARC。このファミリでは、TSF が不正行為(自己保護やドメイン分離の破壊)及び/またはバイパスからどのように保護されるかを的確に把握するために TSFI の記述が使用される;
- ATE。このクラスでは、TSFI の記述が、開発者テストと評価者テストの両方の重要な入力として使用される;
- AVA。このクラスでは、TSFI の記述が脆弱性の探索に使用される。

コンポーネントのレベル付け

224 このファミリのコンポーネントは、TSFI の記述で要求される詳細の程度、及び TSFI 記述で要求される形式化の程度に基づいて、レベル付けされている。

適用上の注釈

225 TSFI が決定されると(TSFI の決定付けのガイダンス及び例については、A.2.1 の「TSFI の決定」を参照のこと)、それらは記述される。下位レベルのコンポーネントでは、開発者は証拠資料を(評価者は分析を)、より TOE のセキュリティに関連する側面に焦点を合わせる。TSFI の 3 つのカテゴリは、それらを通して利用できるサービスと、主張されている SFR との関連性に基づいて定義される:

- インタフェースを通して実行されるサービスを、TSF に課せられた SFR の 1 つにまでたどることができる場合、そのインタフェースは *SFR 実施*と呼ばれる。場合によっては、インタフェースに各種のサービスと結果があり、その中に SFR 実施とそうでないものが含まれる可能性があるので注意する必要がある。
- SFR 実施機能性が依存しているが、TOE のセキュリティ方針を保持するために正しく機能することだけが要求されるサービスへのインタフェース(またはそのサービスに関連するインタフェースを通じて利用可能なサービス)は、*SFR 支援*と呼ばれる。
- SFR 実施機能性が一切依存していないサービスへのインタフェースは、*SFR 非干渉*と呼ばれる。

226 インタフェースを SFR 支援または SFR 非干渉とする場合、そのインタフェースには SFR 実施のサービスや結果が *含まれてはならない* という点に注意するべきである。一方、SFR 実施インタフェースは、SFR 支援サービスを含むこともできる(例えば、システムの時刻を設定する操作は SFR 実施サービスで、それと同じインタフェースを使用するシステムの日付を表示するサービスは SFR 支援という場合もある)。純粋な SFR 支援インタフェースの例としては、利用者及び利用者の代わりに実行される TSF の一部の両方が使用するシステムコールインタフェースなどがある。

227 TSFI に関する情報がより多く提供されるほど、インタフェースが正しく分類/分析される保証も高くなる。評価者がこの判断を効果的に行えるように、要件は、最も低いレベルで SFR 非干渉インタフェースに必要な情報が必要最小限となるように構造化される。レベルが上位になるほど、利用可能な情報が増え、評価者がより強い自信を持って指示を行うことができる。

228 3 つのラベル(SFR 実施、SFR 支援、及び SFR 非干渉)を定義し、それぞれに(より下位の保証コンポーネントで)異なる要件を課す目的は、分析及びその分析の実行対象である証拠の最初のおおよその焦点を絞ることである。開発者が提供した TSF インタフェースの証拠資料が、すべてのインタフェースを、SFR 実施インタフェースの要件で特定された程度まで記述している(つまり証拠資料が要件を上回っている)場合、開発者は要件と一致する新たな証拠を作成する必要はない。同様に、ラベルは要件内でインタフェースのタイプを区別する手段に過ぎないため、開発者はインタフェースを SFR 実施、SFR 支援、または SFR 非干渉と分類するためだけに証拠を更新する必要はない。このラベル付けの主な目的は、成熟した開発方法(及び詳細なインタフェース及び設計証拠資料などの関連する資料)を確立していない開発者が、過度なコストをかけずに必要な証拠のみを提供できるようにすることである。

229 このファミリ内の各コンポーネントの最後の C エレメントは、SFR と機能仕様間の直接的な対応を提供する。つまり、主張されている各 SFR を呼び出すために使用されるインタフェースを示す。ST に、残存情報保護(FDP_RIP)などの、TSFI ではその機能性が発現しない機能要件が含まれている場合は、機能仕様及び/または追跡が、これらの SFR を識別することが期待される。機能仕様にそれらの SFR を含めると、下位レベルの分解でそれらが失われず、関連性を持つことを保証するのに役立つ。

12.2.1 インタフェースに関する詳細

230 要件は提供される TSFI に関する詳細の集合を定義する。要件上、インタフェースはその目的、使用方法、パラメタ、パラメタ記述、及び誤りメッセージの観点から(様々な詳細の程度で)特定される。

231 インタフェースの *目的* は、インタフェースの全般的な目標を上位レベルで記述することである(例えば、GUI コマンドの処理、ネットワークパケットの受信、プリンタ出力の提供など)。

232 インタフェースの *使用方法* は、インタフェースがどのように使用されることが期待されているかを記述する。この記述は、そのインタフェースで利用可能な各種の相互作用を中心に作成されるべきである。例えば、インタフェースが UNIX コマンドシェルである場合は、ls、mv、cp が、そのインタフェースの相互作用である。使用方法は、そのインタフェースでのふるまい(例えば、プログラマによる API の呼び出しや Windows 利用者によるレジストリ設定の変更など)及び他のインタフェースでのふるまい(例えば、監査レコードの生成)の両方に対して相互作用が何を行うかを、相互作用ごとに記述する。

233 *パラメタ*は、インタフェースへの明示的な入力、及びインタフェースからの明示的な出力であり、そのインタフェースのふるまいを制御する。例えば、API に渡される引数、特定のネットワークプロトコルのパケットの様々なフィールド、Windows レジストリの個々のキーの値、チップの一連のピンでやり取りされる信号、*ls* に設定可能なフラグなどがパラメタである。パラメタはそれらの内容の単純なリストで「識別」される。

234 *パラメタの記述*は、そのパラメタが何であるかを意味のある形で伝える。例えば、インタフェース *foo(i)*に対して受け入れられるパラメタ記述は、「パラメタ i は、現在システムにログインしている利用者の数を示す整数である」という記述になる。「パラメタ i は整数である」などの記述は受け入れられない。

235 インタフェースの *アクション*の記述は、インタフェースが何を実行するのかを記述する。これは、「目的」がそれを使用する理由を表すのに対し、「アクション」はそれが実行するすべてのものを表すという点で、目的よりも詳細になる。これらのアクションは、SFR と関連する場合と関連しない場合がある。インタフェースのアクションが SFR に関連しない場合、その記述は *概要*、つまり、記述は SFR がまったく関連しないことを単に明らかにするものとなる。

236 *誤りメッセージ記述*は、そのメッセージが生成された条件、メッセージの内容、及び誤りコードの意味を識別する。誤りメッセージは、問題やある程度の異常が検出されたことを示すために TSF によって生成される。このファミリの要件は、次のようなさまざまな種類の誤りメッセージを表す:

- 「直接的」誤りメッセージは、特定の TSFI 呼び出しに関連付けることができるセキュリティ関連の応答である。
- 「間接的」誤りは、システム全体の条件(資源の消耗、接続の中断など)が原因で発生するため、特定の TSFI 呼び出しに関連付けることはできない。セキュリティに関連しない誤りメッセージも「間接的」とみなされる。
- 「残り」の誤りは、コード内で参照可能な誤りなど、他のすべての誤りである。例えば、論理的には発生しない条件(「case」ステートメントのリストの後に最後の「else」が存在する場合など)をチェックする条件チェックコードの使用が、catch-all 誤りメッセージを生成するために提供される。運用 TOE において、これらの誤りメッセージは表示されてはならない。

237 機能仕様の例は A.2.3 に示される。

12.2.2 このファミリのコンポーネント

238 このファミリの様々な階層コンポーネントで詳述されているように、インタフェース仕様での完全性と正確さが増すことで高まる保証は、開発者に要求される証拠資料に反映される。

239 ADV_FSP.1 基本機能仕様では、唯一要求される証拠資料として、すべての TSFI の特性及び SFR 実施 TSFI と SFR 支援 TSFI の上位レベルの記述がある。TSF の「重要な」側面が TSFI で正しく特徴付けされていることの保証を提供するために、開発者には、SFR 実施及び SFR 支援 TSFI の目的、使用方法、パラメタ、及びパラメタ記述を提供することが要求される。

240 ADV_FSP.2 セキュリティ実施機能仕様では、開発者に対し、すべての TSFI の目的、使用方法、パラメタ、及びパラメタ記述を提供することが要求される。さらに、SFR 実施 TSFI について、開発者は SFR 実施アクション及び直接的メッセージを記述しなければならない。

241 ADV_FSP.3 完全な要約を伴う機能仕様では、開発者は、ADV_FSP.2 で要求される情報に加えて、SFR 支援アクションと SFR 非干渉アクション、及び誤りメッセージに関し、それらが SFR 実施ではないことを示す十分な情報を提供しなければならない。さらに、開発者は SFR 実施 TSFI の呼び出しに起因するすべての直接的誤りメッセージを、証拠資料として提出しなければならない。

242 ADV_FSP.4 完全な機能仕様では、すべての TSFI(SFR 実施、SFR 支援、及び SFR 非干渉)を、直接的誤りメッセージをすべて含め同じ程度に記述する必要がある。

243 ADV_FSP.5 追加の誤り情報を伴う完全な準形式的機能仕様では、TSFI 記述には間接的誤りメッセージも含まれる。

244 ADV_FSP.6 追加の形式的仕様を伴う完全な準形式的機能仕様では、ADV_FSP.5 に必要な情報に加え、すべての残りの誤りメッセージが含まれる。開発者は、TSFI の形式的な記述も提供しなければならない。これにより、TSFI を新たな観点から見て、不一致あるいは不完全な仕様を明らかにすることができる。

ADV_FSP.1 基本機能仕様

依存性: なし

開発者アクションエレメント:

ADV_FSP.1.1D **開発者は、機能仕様を提供しなければならない。**

ADV_FSP.1.2D **開発者は、機能仕様から SFR への追跡を提供しなければならない。**

内容・提示エレメント:

ADV_FSP.1.1C **機能仕様は、SFR 実施及び SFR 支援の各 TSFI の目的と使用方法を記述しなければならない。**

ADV_FSP.1.2C **機能仕様は、SFR 実施及び SFR 支援の各 TSFI に関連するすべてのパラメタを識別しなければならない。**

ADV_FSP.1.3C **機能仕様は、暗黙的に SFR 非干渉として分類されているインタフェースについて、その分類の根拠を示さなければならない。**

ADV_FSP.1.4C **追跡は、機能仕様での TSFI に対する SFR の追跡を実証するものでなければならない。**

評価者アクションエレメント:

ADV_FSP.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ADV_FSP.1.2E **評価者は、機能仕様が、SFR の正確かつ完全な具体化であることを*決定しなければならない*。**

ADV_FSP.2 セキュリティ実施機能仕様

依存性: ADV_TDS.1 基本設計

開発者アクションエレメント:

ADV_FSP.2.1D 開発者は、機能仕様を提供しなければならない。

ADV_FSP.2.2D 開発者は、機能仕様から SFR への追跡を提供しなければならない。

内容・提示エレメント:

ADV_FSP.2.1C **機能仕様は、完全に TSF を表現しなければならない。**

ADV_FSP.2.2C 機能仕様は、**すべての** TSFI の目的と使用方法を記述しなければならない。

ADV_FSP.2.3C 機能仕様は、各 TSFI に関連するすべてのパラメタを識別**及び記述**しなければならない。

ADV_FSP.2.4C **SFR 実施 TSFI について、**機能仕様は、その TSFI に関連する **SFR 実施アクションを記述しなければならない。**

ADV_FSP.2.5C **SFR 実施 TSFI について、機能仕様は、SFR 実施アクションに関連する処理によって発生する誤りメッセージを記述しなければならない。**

ADV_FSP.2.6C 追跡は、機能仕様での TSFI に対する SFR の追跡を実証するものでなければならない。

評価者アクションエレメント:

ADV_FSP.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_FSP.2.2E 評価者は、機能仕様が、SFR の正確かつ完全な具体化であることを***決定しなければならない。***

ADV_FSP.3 完全な要約を伴う機能仕様

依存性: ADV_TDS.1 基本設計

開発者アクションエレメント:

ADV_FSP.3.1D 開発者は、機能仕様を提供しなければならない。

ADV_FSP.3.2D 開発者は、機能仕様から SFR への追跡を提供しなければならない。

内容・提示エレメント:

ADV_FSP.3.1C 機能仕様は、完全に TSF を表現しなければならない。

ADV_FSP.3.2C 機能仕様は、すべての TSFI の目的と使用方法を記述しなければならない。

ADV_FSP.3.3C 機能仕様は、各 TSFI に関連するすべてのパラメタを識別及び記述しなければならない。

ADV_FSP.3.4C SFR 実施 TSFI について、機能仕様は、その TSFI に関連する SFR 実施アクションを記述しなければならない。

ADV_FSP.3.5C **SFR 実施 TSFI について、機能仕様は、その TSFI の呼び出しに関連するセキュリティ実施効果及び例外によって発生する直接的誤りメッセージを記述しなければならない。**

ADV_FSP.3.6C 機能仕様は、**各 TSFI** に関連する**非 SFR 実施アクションを要約**しなければならない。

ADV_FSP.3.7C 追跡は、機能仕様での TSFI に対する SFR の追跡を実証するものでなければならない。

評価者アクションエレメント:

ADV_FSP.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_FSP.3.2E 評価者は、機能仕様が、SFR の正確かつ完全な具体化であることを***決定しなければならない。***

ADV_FSP.4 完全な機能仕様

依存性: ADV_TDS.1 基本設計

開発者アクションエレメント:

ADV_FSP.4.1D 開発者は、機能仕様を提供しなければならない。

ADV_FSP.4.2D 開発者は、機能仕様から SFR への追跡を提供しなければならない。

内容・提示エレメント:

ADV_FSP.4.1C 機能仕様は、完全に TSF を表現しなければならない。

ADV_FSP.4.2C 機能仕様は、すべての TSFI の目的と使用方法を記述しなければならない。

ADV_FSP.4.3C 機能仕様は、各 TSFI に関連するすべてのパラメタを識別及び記述しなければならない。

ADV_FSP.4.4C 機能仕様は、各 TSFI に関連する**すべての**アクションを**記述**しなければならない。

ADV_FSP.4.5C **機能仕様は、各 TSFI の呼び出しに関連する、セキュリティ実施効果及び例外によって発生する可能性があるすべての直接的誤りメッセージを記述しなければならない。**

ADV_FSP.4.6C 追跡は、機能仕様での TSFI に対する SFR の追跡を実証するものでなければならない。

評価者アクションエレメント:

ADV_FSP.4.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_FSP.4.2E 評価者は、機能仕様が、SFR の正確かつ完全な具体化であることを***決定しなければならない。***

ADV_FSP.5　追加の誤り情報を伴う完全な準形式的機能仕様

依存性:　ADV_TDS.1 基本設計
ADV_IMP.1 TSF の実装表現

開発者アクションエレメント:

ADV_FSP.5.1D　開発者は、機能仕様を提供しなければならない。

ADV_FSP.5.2D　開発者は、機能仕様から SFR への追跡を提供しなければならない。

内容・提示エレメント:

ADV_FSP.5.1C　機能仕様は、完全に TSF を表現しなければならない。

ADV_FSP.5.2C　**機能仕様は、準形式的スタイルを使用して TSFI を記述しなければならない。**

ADV_FSP.5.3C　機能仕様は、すべての TSFI の目的と使用方法を記述しなければならない。

ADV_FSP.5.4C　機能仕様は、各 TSFI に関連するすべてのパラメタを識別及び記述しなければならない。

ADV_FSP.5.5C　機能仕様は、各 TSFI に関連するすべてのアクションを記述しなければならない。

ADV_FSP.5.6C　機能仕様は、各 TSFI の呼び出しによって発生する可能性があるすべての直接的誤りメッセージを記述しなければならない。

ADV_FSP.5.7C　**機能仕様は、TSFI の呼び出しによって発生しないすべての誤りメッセージを記述しなければならない。**

ADV_FSP.5.8C　**機能仕様は、TSF の実装に含まれているが TSFI の呼び出しによって発生しない各誤りメッセージについて、その根拠を示さなければならない。**

ADV_FSP.5.9C　追跡は、機能仕様での TSFI に対する SFR の追跡を実証するものでなければならない。

評価者アクションエレメント:

ADV_FSP.5.1E　評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_FSP.5.2E　評価者は、機能仕様が、SFR の正確かつ完全な具体化であることを***決定しなければならない。***

ADV_FSP.6　追加の形式的仕様を伴う完全な準形式的機能仕様

依存性:　ADV_TDS.1 基本設計

開発者アクションエレメント:

ADV_FSP.6.1D　開発者は、機能仕様を提供しなければならない。

ADV_FSP.6.2D　**開発者は、TSF の機能仕様の形式的表現を提供しなければならない。**

ADV_FSP.6.3D　開発者は、機能仕様から SFR への追跡を提供しなければならない。

内容・提示エレメント:

ADV_FSP.6.1C 機能仕様は、完全に TSF を表現しなければならない。

ADV_FSP.6.2C 機能仕様は、**形式的**スタイルを使用して TSFI を記述しなければならない。

ADV_FSP.6.3C 機能仕様は、すべての TSFI の目的と使用方法を記述しなければならない。

ADV_FSP.6.4 C 機能仕様は、各 TSFI に関連するすべてのパラメタを識別及び記述しなければならない。

ADV_FSP.6.5C 機能仕様は、各 TSFI に関連するすべてのアクションを記述しなければならない。

ADV_FSP.6.6C 機能仕様は、各 TSFI の呼び出しによって発生する可能性があるすべての直接的誤りメッセージを記述しなければならない。

ADV_FSP.6.7C 機能仕様は、TSF 実装に含まれているが**機能仕様には記述**されない、**すべての**誤り**メッセージを記述**しなければならない。

ADV_FSP.6.8C **機能仕様は、TSF 実装に含まれているが機能仕様には記述されない、各誤りメッセージについて、TSFI に関連しない理由を正当化する根拠を提供しなければならない。**

ADV_FSP.6.9C **TSF の**機能仕様の**形式的表現は、適切な個所に対して非形式的で説明的なテキストで補足される形式的スタイルを使用して、TSFI を記述しなければならない。**

ADV_FSP.6.10C 追跡は、機能仕様での TSFI に対する SFR の追跡を実証するものでなければならない。

評価者アクションエレメント:

ADV_FSP.6.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_FSP.6.2E 評価者は、機能仕様が、SFR の正確かつ完全な具体化であることを***決定しなければならない。***

12.3 実装表現(ADV_IMP)

目的

245 実装表現(ADV_IMP)ファミリの機能は、TOE の実装表現(及び上位レベルでは実装自体)を、評価者が分析できる形式で開発者が提供できるようにする。実装表現は、他のファミリの分析アクティビティ(TOE 設計の分析など)で、TOE がその設計に適合していることを実証し、その他の評価領域(脆弱性の探索など)での分析の基礎を提供するために使用される。実装表現は、TSF の詳細な内部動作を示す形式であることが期待される。これには、ソフトウェアのソースコード、ファームウェアのソースコード、ハードウェア図、及び/または IC ハードウェア設計言語コードまたはレイアウトデータが含まれる。

コンポーネントのレベル付け

246 このファミリのコンポーネントは、TOE 設計記述にマッピングされる実装の量に基づいてレベル付けされている。

適用上の注釈

247 ソースコードや、実際のハードウェアの製造に用いられるハードウェア図及び/または IC ハードウェア設計言語コードやレイアウトデータは、実装表現の一部の例である。実装表現は、評価者に提供しなければならないが、これは評価者がその表現を所有する必要があることを意味するものではないということが重要である。例えば、開発者の選んだサイトで評価者が実装表現をレビューすることを、開発者が要求する場合がある。

248 情報不足によって分析アクティビティが制限されることのないように、実装表現全体が提供される。とはいえ、分析アクティビティが行われる際にすべての表現が検査されるわけではない。そのようなことは、ほとんどすべての場合に現実的でないうえ、たいていは、実装表現のターゲットサンプリングに比べて TOE の保証が高くなるわけでもない。実装表現は、他の TOE 設計コンポーネント構成(機能仕様や TOE 設計など)の分析を可能にするため、及び設計の上位レベルで記述されているセキュリティ機能性が実際に TOE に実装されるという確信を得るために提供される。ある種の実装表現には、コンパイルや実行時の解釈の実際の結果を、実装表現のみから決定するのを困難にしたり不可能にしたりするような規則がある。例えば C 言語コンパイラでは、コンパイラディレクティブによって、コードの特定の部分全体が除外されたり含まれたりする。このため、このような「付加的な」情報または関連ツール(スクリプト、コンパイラなど)を提供して、実装表現が正確に決定されるようにすることが重要である。

249 実装表現と TOE 設計記述間のマッピングの目的は、評価者の分析を支援することである。TOE の内部動作は、TOE 設計が実装表現の対応する部分で分析されたときに理解が深まる可能性がある。マッピングは、実装表現への索引として使用される。下位のコンポーネントでは、実装表現のサブセットだけが TOE 設計記述にマッピングされる。実装表現の中でそのようなマッピングを必要とする部分が不確定であるため、開発者は実装表現全体を事前にマップするか、実装表現の中で評価者がマッピングを必要とする部分が確認できるまで待つかを選択できる。

250 実装表現は、開発者によって、実際の実装への変換に適した形式で操作される。例えば開発者は、最終的にコンパイルされて TSF の一部となるソースコードを含むファイルを使用することができる。開発者は、評価者が分析において自動化の技法を使用できるように、開発者が使用する形式で実装表現を提供する。これにより、検査される実装表現が、実際に TSF の作成に使用されるものであるという信頼も高まる(ワードプロセッサ文書などの別の表現形式で提供される場合とは対照的)。ただし、開発者は他の形式の実装表現も使用できるという点に注意するべきである。それらの形式の実装表現も一緒に提供される。全体的な目標は、評価者の分析の効果を最大限に高める情報を提供することである。

251 ある種の実装表現では、理解や分析に対する重大な障害が持ち込まれるために、追加の情報が必要になることがある。例えば、「隠蔽されている」ソースコードや、理解や分析を妨げるその他の形で分かりにくくされているコードがこれに該当する。一般に、このような形式の実装表現は、TOE 開発者によって使用されているバージョンの実装表現に対して、コードを隠蔽したり分かりにくくしたりするプログラムが実行された結果である。隠蔽されている表現はコンパイルの対象であり、元の隠蔽されていない表現より(構造の観点からは)実装に近いと言えるが、そのように分かりにくくされているコードを提供すると、その表現に関連する分析作業にかかる時間が大幅に増加する可能性がある。このような形式の表現が作成される場合は、隠蔽されていない表現を提供できるように、使用されている隠蔽ツール/アルゴリズムについての詳細がコンポーネントで必要とされる。この追加の情報は、隠蔽のプロセスによって弱体化しているセキュリティ機能性がないという確信を得るために使用できる。

ADV_IMP.1 TSF の実装表現

依存性: ADV_TDS.3 基本モジュール設計
ALC_TAT.1 明確に定義された開発ツール

開発者アクションエレメント:

ADV_IMP.1.1D 開発者は、TSF 全体の実装表現を提供しなければならない。

ADV_IMP.1.2D 開発者は、TOE 設計記述と実装表現のサンプルの間のマッピングを提供しなければならない。

内容・提示エレメント:

ADV_IMP.1.1C 実装表現は、それ以上の設計上の決定を必要とせずに、TSF を生成できるような詳細レベルまで TSF を定義しなければならない。

ADV_IMP.1.2C 実装表現の形式は、開発要員が使用する形式でなければならない。

ADV_IMP.1.3C TOE 設計記述と実装表現のサンプルの間のマッピングは、両者の対応を実証しなければならない。

評価者アクションエレメント:

ADV_IMP.1.1E 評価者は、選択された実装表現のサンプルについて、提供された情報が証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ADV_IMP.2 TSF の実装

依存性: ADV_TDS.3 基本モジュール設計
ALC_TAT.1 明確に定義された開発ツール
ALC_CMC.5 高度なサポート

開発者アクションエレメント:

ADV_IMP.2.1D 開発者は、TSF 全体の実装表現を提供しなければならない。

ADV_IMP.2.2D 開発者は、TOE 設計記述と実装表現**全体**の間のマッピングを提供しなければならない。

内容・提示エレメント:

ADV_IMP.2.1C 実装表現は、それ以上の設計上の決定を必要とせずに、TSF を生成できるような詳細レベルまで TSF を定義しなければならない。

ADV_IMP.2.2C 実装表現の形式は、開発要員が使用する形式でなければならない。

ADV_IMP.2.3C TOE 設計記述と実装表現**全体**の間のマッピングは、両者の対応を実証しなければならない。

評価者アクションエレメント:

ADV_IMP.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

12.4 TSF 内部構造(ADV_INT)

目的

252 このファミリは、TSF の内部構造の評定を扱う。内部構造が適切に構成された TSF は、実装が容易になり、脆弱性の原因となる欠陥を含む可能性が低くなる。また、欠陥をもたらさない保守も容易になる。

コンポーネントのレベル付け

253 このファミリのコンポーネントは、必要となる構造と複雑さの最小化の量に基づいて、レベル付けされている。ADV_INT.1 TSF 内部構造の適切に構成されたサブセットが要件を課す対象は、TSF の選択された部分のみでの適切に構成された内部構造である。このコンポーネントは EAL に含まれない。これは、このコンポーネントが特別な状況(例えば、スポンサーが、TSF の他の部分からは孤立している暗号モジュールに特別な関心を持っている場合)で使用されると見られ、広く適用されることがないからである。

254 次のレベルでは、適切に構成された内部構造に対する要件が TSF 全体に課せられる。最後に、複雑さの最小化が最上位のコンポーネントに導入される。

適用上の注釈

255 一般的にこれらの要件は、TSF の内部構造に適用されることで、開発者と評価者の両方が TSF を理解しやすくなるという改良をもたらし、またテストスイートを設計及び評価するための基礎を提供する。さらに、TSF の理解のしやすさが向上することは、開発者による TSF の保守性の単純化に役立つ。

256 このファミリの要件は、かなり抽象的なレベルで表される。TOE の多様性から、「適切に構成された」または「最小の複雑さ」よりも具体的なものを体系化することは不可能である。構造と複雑さに関する判断は、TOE で使用される特定の技術から導き出されることが期待される。例えば、ソフトウェアエンジニアリングの分野で挙げられる特性を示す場合、ソフトウェアは適切に構成されたものとみなされる可能性が高い。このファミリ内のコンポーネントは、適切に構成され複雑すぎない特性を測定するための標準を識別することを要求する。

ADV_INT.1 適切に構成された TSF 内部構造のサブセット

依存性: ADV_IMP.1 TSF の実装表現
ADV_TDS.3 基本モジュール設計
ALC_TAT.1 明確に定義された開発ツール

目的

257 このコンポーネントの目的は、TSF の特定の部分を適切に構成することを要求する手段を提供することである。その意図は、TSF 全体は適切なエンジニアリングの原則を使用して設計及び実装されるが、分析は特定のサブセットにのみ基づいて実行されるということである。

適用上の注釈

258 このコンポーネントは、PP または ST の作成者に対し、TSF のサブセットを割付に記入することを要求する。このサブセットは、抽象化のいずれかの層で TSF の内部構造という観点から識別される場合がある。たとえば:

- TOE 設計で識別される TSF の構造エレメント(例:「開発者は、適切に構成された内部構造を持つように*監査サブシステム*を設計及び実行しなければならない」)。
- 実装(例:「開発者は *encrypt.c ファイルと decrypt.c ファイル*を、適切に構成された内部構造を持つように設計及び実装しなければならない」または「開発者は、適切に構成された内部構造を持つように *6227 IC チップ*を設計及び実装しなければならない」)。

259 分析の焦点となる場所を示さないため、主張されている SFR を参照しても確実に実現できない可能性がある(例:「開発者は、適切に構成された内部構造を持つように、*FPR_ANO.2 で定義されているとおりに匿名性を提供する TSF の部分*を設計及び実装しなければならない」)。

260 このコンポーネントは値が制限されており、悪意を持っている可能性がある利用者/サブジェクトによる TSFI へのアクセスが制限または厳しく制御されている場合や、TSF の選択されたサブセットが TSF の他の部分から悪影響を受けない(例えば暗号化機能性は適切に構成され、TSF の他の部分から分離される)ことを保証する別の保護手段(例えばドメイン分離)が存在する場合に適している。

開発者アクションエレメント:

ADV_INT.1.1D **開発者は、適切に構成された内部構造を持つように[割付: *TSF のサブセット*]を設計及び実装しなければならない。**

ADV_INT.1.2D **開発者は、内部構造の記述及び正当化を提供しなければならない。**

内容・提示エレメント:

ADV_INT.1.1C **正当化は、「適切に構成された」の意味を判断するために使用される特性を説明しなければならない。**

ADV_INT.1.2C **TSF 内部構造の記述は、割り付けられた TSF のサブセットが適切に構成されていることを実証しなければならない。**

評価者アクションエレメント:

ADV_INT.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ADV_INT.1.2E **評価者は、割り付けられた TSF のサブセットに関する内部構造の分析を*実行しなければならない*。**

ADV_INT.2 適切に構成された内部構造

依存性: ADV_IMP.1 TSF の実装表現
ADV_TDS.3 基本モジュール設計
ALC_TAT.1 明確に定義された開発ツール

目的

261 このコンポーネントの目的は、TSFを適切に構成することを要求する手段を提供することである。その意図は、TSF 全体が適切なエンジニアリングの原則を使用して設計及び実装されていることである。

適用上の注釈

262 構造の適切性に関する判断は、TOE で使用される特定の技術から導き出されることが期待される。このコンポーネントは、適切に構成されているという特性を測定するための標準を識別することを要求する。

開発者アクションエレメント:

ADV_INT.2.1D 開発者は、適切に構成された内部構造を持つように **TSF 全体**を設計及び実装しなければならない。

ADV_INT.2.2D 開発者は、内部構造の記述及び正当化を提供しなければならない。

内容・提示エレメント:

ADV_INT.2.1C 正当化は、「適切に構成された」の意味を判断するために使用される特性を説明しなければならない。

ADV_INT.2.2C TSF 内部構造の記述は、TSF **全体**が適切に構成されていることを実証しなければならない。

評価者アクションエレメント:

ADV_INT.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_INT.2.2E 評価者は、TSF に関する内部構造の分析を***実行しなければならない。***

ADV_INT.3 最小限複雑な内部構造

依存性: ADV_IMP.1 TSF の実装表現
ADV_TDS.3 基本モジュール設計
ALC_TAT.1 明確に定義された開発ツール

目的

263 このコンポーネントの目的は、TSF が適切に構成され最小の複雑さであることを要求する手段を提供することである。その意図は、TSF 全体が適切なエンジニアリングの原則を使用して設計及び実装されていることである。

適用上の注釈

264 構造及び複雑さの適切性に関する判断は、TOE で使用される特定の技術から導き出されることが期待される。このコンポーネントは、構造及び複雑さを測定するための標準を識別することを要求する。

開発者アクションエレメント:

ADV_INT.3.1D 開発者は、適切に構成された内部構造を持つようにTSF 全体を設計及び実装しなければならない。

ADV_INT.3.2D 開発者は、内部構造の記述及び正当化を提供しなければならない。

内容・提示エレメント:

ADV_INT.3.1C 正当化は、「適切に構成された」**及び「複雑」**の意味を判断するために使用される特性を説明しなければならない。

ADV_INT.3.2C TSF 内部構造の記述は、TSF 全体が適切に構成されていることを実証しなければならない。

ADV_INT.3.3C **TSF 内部構造の記述は、TSF 全体が適切に構成され、複雑すぎないことを実証しなければならない。**

評価者アクションエレメント:

ADV_INT.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_INT.3.2E 評価者は、TSF **全体**に関する内部構造の分析を***実行しなければならない。***

12.5 セキュリティ方針モデル化(ADV_SPM)

目的

265 このファミリの目的は、TSF の形式的な*セキュリティ方針モデル*を開発し、機能仕様とセキュリティ方針モデルの間に対応を確立することにより追加の保証を提供することである。内部的一貫性を維持することで、セキュリティ方針モデルは、数学的証明によりその特性に基づいて形式的にセキュリティ原則を確立することが期待される。

コンポーネントのレベル付け

266 このファミリは、ただ 1 つのコンポーネントより成る。

適用上の注釈

267 TOE の不十分性は、セキュリティ要件の理解不足、またはそれらのセキュリティ要件の実装に欠陥があることに起因している可能性がある。セキュリティ要件を適切に定義してそれらが確実に理解されるようにすることは難しい。これは、TOE の実装中に、望まない結果、つまりわずかでも欠陥が生じないようにするには、定義が十分に正確でなければならないからである。設計、実装、レビューの各プロセス全体で、モデル化されたセキュリティ要件が正確な設計及び実装のガイダンスとして使用されることで、モデル化されたセキュリティ要件を TOE が満たしていることの保証が高まる。モデル及びその結果であるガイダンスの正確さは、モデルを形式的な言語で作成すること及び形式的な証明によりセキュリティ要件を検証することで大幅に向上する。

268 形式的なセキュリティ方針モデルの作成は、セキュリティ方針の曖昧なエレメント、一貫性のないエレメント、矛盾するエレメント、あるいは実施不可能なエレメントを識別し、排除するのに役立つ。TOE が構築されると、形式的モデルは、実装されているセキュリティ機能性を開発者がどの程度十分に理解しているか、及びセキュリティ要件と TOE 設計の間に一貫しない点がないかどうかという評価者の判定に寄与することになり、評価成果に役立つ。このモデルの信頼は、不一致が含まれないという証明により達成される。

269 形式的セキュリティモデルは、セキュリティの重要な側面、及びそれらの側面と TOE のふるまいとの関係を正確かつ形式的に表したものであり、TSF がシステム資源をどのように管理し、保護し、制御するかを規定する一連の規則及び慣行を識別する。モデルには、情報と計算資源が、SFR を侵害する目的で使用されないようにする方法を特定する一連の制約と特性が含まれ、それらの制約と特性が SFR の実施に主要な役割を果たすことを示す説得力ある一連の工学的論証がこれに付随している。モデルは、セキュリティ機能性を表す形式化と、モデルについて説明し、モデルの枠組みを示す補助的な説明文の両方で構成される。TSF のセキュリティのふるまいは、外部的なふるまい(つまり TSF が TOE の残りの部分及びその運用環境とどのように相互作用するか)及びその内部的なふるまいの両方の点からモデル化される。

270 TOE のセキュリティ方針モデルは、ST の提案するセキュリティ要件を考慮することで、その実現から非形式的に抽象化される。非形式的な抽象は、TOE の原則(「不変式」とも呼ばれる)がその特性により実施されることになる場合は、正しいとみなせる。形式的な方法の目的は、実施の厳格性の強化の範囲内にある。非形式的な論証は、特にサブジェクト、オブジェクト、及び操作間の関係がより入り組むと、常に誤りが発生しやすくなる。安全でない状態になるリスクを最小限に抑えるために、セキュリティ方針モデルの規則と特性は一部の形式的なシステム内の個々の特性や機能にマップされ、それらの厳格性や強度を後から使用して、定理及び形式的な証明によりセキュリティの特性を得ることができる。

271 「形式的なセキュリティ方針モデル」という用語は、学術的に使用されるが、CC のアプローチでは「セキュリティ」の固定的な定義がない。つまり、主張されているどの SFR とも同一になる。したがって、形式的なセキュリティ方針モデルは、主張されている一連の SFR の単なる形式的な表現にすぎない。

272 *セキュリティ方針*という用語は、従来、ラベルベース(強制アクセス制御)または利用者ベース(任意アクセス制御)のアクセス制御方針のみに関連付けられていた。ところが、セキュリティ方針はアクセス制御に限定されるわけではなく、PP/ST で記述されているように、他にも監査方針、識別方針、認証方針、暗号化方針、管理方針、あるいは TOE によって実施されるその他のセキュリティ方針が存在する。**ADV_SPM.1.1D** には、形式的にモデル化されている方針を識別するための割付が含まれている。

ADV_SPM.1 形式的な TOE セキュリティ方針モデル

依存性: ADV_FSP.4 完全な機能仕様

開発者アクションエレメント:

ADV_SPM.1.1D 開発者は、**[**割付**:** *形式的にモデル化され、モデル化された各方針で構成される **SFR** のステートメントの各部分を識別する方針のリスト***]**の形式的なセキュリティ方針モデルを提供しなければならない。

ADV_SPM.1.2D 開発者は、モデルと任意の形式的な機能仕様との間の対応の形式的な証明を提供しなければならない。

ADV_SPM.1.3D 開発者は、モデルと機能仕様との間の対応の実証を提供しなければならない。

内容・提示エレメント:

ADV_SPM.1.1C モデルは、必要に応じて説明文で補足される形式的スタイルで記述され、モデル化する **TSF** のセキュリティ方針を識別しなければならない。

ADV_SPM.1.2C モデル化されるすべての方針について、モデルは **TOE** のセキュリティを定義し、**TOE** が安全ではない状態にならないことの形式的な証明を提供しなければならない。

ADV_SPM.1.3C モデルと機能仕様との対応は、正しい形式化のレベルでなければならない。

ADV_SPM.1.4C 対応は、機能仕様がモデルに対して一貫し完全であることを示さなければならない。

ADV_SPM.1.5C 対応の実証は、機能仕様の*インタフェース*が **ADV_SPM.1.1D** 割付の方針に対して一貫し完全であることを示さなければならない。

評価者アクションエレメント:

ADV_SPM.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

12.6 TOE 設計(ADV_TDS)

目的

273 TOE の設計記述は、TSF の記述の枠組み、及び TSF の綿密な記述の両方を提供する。より高い保証が必要になると、記述で提供される詳細のレベルも高くなる。TSF の大きさと複雑さが増大するにつれて、複数レベルの分解が適切となる。設計要件の意図するところは、指定の保証レベルに見合った情報を提供して、セキュリティ機能要件が実現されていることを判断できるようにすることである。

コンポーネントのレベル付け

274 このファミリのコンポーネントは、TSF に関して提示する必要がある情報の量、及び設計記述に要求される形式化の程度に基づいて、レベル付けされている。

適用上の注釈

275 設計証拠資料の目標は、TSF の境界を決定するための十分な情報を提供し、TSF がセキュリティ機能要件を*どのように*実装するかを記述することである。設計証拠資料の量と構造は、TOE の複雑さと SFR の数によって決まる。一般に、多数の SFR を実装する非常に複雑な TOE は、わずかな数の SFR のみを実装する非常に単純な TOE に比べて、より多くの設計証拠資料を必要とする。非常に複雑な TOE では、設計を記述する際に様々なレベルの分解を生成することが(提供される保証の点で)有用であるが、非常に単純な TOE は、その実装の上位レベルの記述と下位レベルの記述をどちらも必要としない。

276 このファミリは、*サブシステム*及び*モジュール*の 2 つのレベルの分解を使用する。モジュールは、最も具体的な機能性の記述、つまり実装の記述である。開発者は、モジュールで記述された TOE の部分を、それ以上の設計上の決定を行うことなく実装できるべきである。サブシステムは TOE の設計の記述である。つまり、TOE の部分が何をどのように行うかを上位レベルで記述するのに役立つ。したがって、サブシステムは下位レベルのサブシステムまたはモジュールに分割することができる。非常に複雑な TOE で、TOE の動作内容に関する有用な記述を十分に伝えるためには、複数レベルのサブシステムが必要となる。これとは対照的に、非常に単純な TOE はサブシステムレベルの記述を必要としない。つまり、TOE の動作内容はモジュールで明確に記述される。

277 設計証拠資料に採用される一般的な手法では、保証レベルの上昇に伴って、記述の重点が概略(サブシステムレベル)から詳細(モジュールレベル)にシフトする。モジュールレベルで十分に記述できるほど TOE が単純であるために、モジュールレベルの抽象化が適切である場合は、保証レベルでサブシステムレベルの記述が要求されても、モジュールレベルの記述で間に合う。しかしながら、複雑な TOE の場合はこれが当てはまらない。膨大な量の(モジュールレベルの)詳細は、サブシステムレベルの記述が付随していないと理解できない。

278 この手法は、TSF の実装に関する追加の詳細を提供することで、SFR が正しく実装されることの保証が高まり、さらにテスト(ATE: テスト)でこれを実証するための情報が提供されるという、一般的なパラダイムに従っている。

279 このファミリの要件では、*インタフェース*という用語が(2 つのサブシステムまたはモジュール間の)通信手段として使用される。インタフェースは、通信が起動される方法を記述する点で TSFI の詳細に似ている(機能仕様(ADV_FSP)を参照のこと)。*相互作用*という用語は、通信の目的を識別するために使用される。つまりこの用語は、2 つのサブシステムまたはモジュールが通信する理由を識別する。

12.6.1 サブシステム及びモジュールに関する詳細

280 要件は提供されるサブシステム及びモジュールに関する詳細の集合を定義する:

- サブシステムとモジュールは、それらの内容を示す単純なリストで*識別される*。
- サブシステムは、「SFR 実施」、「SFR 支援」、または「SFR 非干渉」として(暗黙的または明示的に)*分類できる*。これらの用語は機能仕様(ADV_FSP)で使用されているものと同様に使用される。
- サブシステムの*ふるまい*は、そのサブシステムが実施する内容である。ふるまいも、SFR 実施、SFR 支援、または SFR 非干渉として分類できる。サブシステムのふるまいは、サブシステムそのものの分類よりもより SFR に関連するものとして分類されない。例えば、SFR 実施サブシステムは、SFR 実施のふるまい及び非 SFR 実施のふるまいを持つことができる。
- サブシステムの*ふるまいの要約*は、そのサブシステムが実行するアクションの概要である(例:「TCP サブシステムは IP データグラムを信頼できるバイトストリームに集合させる」)。
- サブシステムの*ふるまいの記述*は、サブシステムが行うすべてのアクションの説明である。この記述は、ふるまいが SFR の実施と何らかの関連性を持つかどうかを確実に決定できる 1 つの詳細レベルにあるべきである。
- サブシステム間での*相互作用の記述*は、サブシステムが通信する理由、及び渡される情報の特性を識別する。インタフェース仕様と同じレベルの詳細まで情報を定義する必要はない。例えば、「サブシステム X はメモリマネージャにメモリのブロックを要求し、メモリマネージャは割り当てられたメモリの場所で応答する」というような記述で十分である。
- モジュールの*目的*には、それ以上の設計上の決定が必要ない十分な詳細を提供する。モジュールを実装するソースコードとモジュールの目的との間の対応は確実に明らかにすべきである。
- そうしない場合、モジュールはエレメントで識別されるものすべての観点から*記述される*。

サブシステムとモジュール、及び「SFR 実施」などについては、附属書 A の A.4「ADV_TDS: サブシステム及びモジュール」で詳しく説明されている。

ADV_TDS.1 基本設計

依存性: ADV_FSP.2 セキュリティ実施機能仕様

開発者アクションエレメント:

ADV_TDS.1.1D **開発者は、TOE の設計を提供しなければならない。**

ADV_TDS.1.2D **開発者は、機能仕様の TSFI から TOE 設計で利用可能な分解の最下位レベルへのマッピングを提供しなければならない。**

内容・提示エレメント:

ADV_TDS.1.1C 設計は、サブシステムの観点から TOE の構造を記述しなければならない。

ADV_TDS.1.2C 設計は、TSF のすべてのサブシステムを識別しなければならない。

ADV_TDS.1.3C 設計は、非 SFR 実施であることを決定するために、TSF の各 SFR 支援または SFR 非干渉サブシステムのふるまいを十分に詳細に記述しなければならない。

ADV_TDS.1.4C 設計は、SFR 実施サブシステムの SFR 実施のふるまいを要約しなければならない。

ADV_TDS.1.5C 設計は、TSF の SFR 実施サブシステム間、及び TSF の SFR 実施サブシステムと TSF のその他のサブシステム間の相互作用の記述を提供しなければならない。

ADV_TDS.1.6C マッピングは、TOE 設計で記述されているすべてのふるまいが、そのふるまいを呼び出す TSFI にマッピングされていることを実証しなければならない。

評価者アクションエレメント:

ADV_TDS.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ADV_TDS.1.2E 評価者は、設計が、すべてのセキュリティ機能要件の正確かつ完全な具体化であることを*決定しなければならない*。

ADV_TDS.2 アーキテクチャ設計

依存性: ADV_FSP.3 完全な要約を伴う機能仕様

開発者アクションエレメント:

ADV_TDS.2.1D 開発者は、TOE の設計を提供しなければならない。

ADV_TDS.2.2D 開発者は、機能仕様の TSFI から TOE 設計で利用可能な分解の最下位レベルへのマッピングを提供しなければならない。

内容・提示エレメント:

ADV_TDS.2.1C 設計は、サブシステムの観点から TOE の構造を記述しなければならない。

ADV_TDS.2.2C 設計は、TSF のすべてのサブシステムを識別しなければならない。

ADV_TDS.2.3C 設計は、SFR 非干渉であることを決定するために、TSF の各 SFR 非干渉サブシステムのふるまいを十分に詳細に記述しなければならない。

ADV_TDS.2.4C 設計は、SFR 実施サブシステムの SFR 実施のふるまいを記述しなければならない。

ADV_TDS.2.5C 設計は、SFR 実施サブシステムの非 SFR 実施のふるまいを要約しなければならない。

ADV_TDS.2.6C 設計は、SFR 支援サブシステムのふるまいを要約しなければならない。

ADV_TDS.2.7C 設計は、TSF のすべてのサブシステム間の相互作用の記述を提供しなければならない。

ADV_TDS.2.8C マッピングは、TOE 設計で記述されているすべてのふるまいが、そのふるまいを呼び出す TSFI にマッピングされていることを実証しなければならない。

評価者アクションエレメント:

ADV_TDS.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_TDS.2.2E 評価者は、設計が、すべてのセキュリティ機能要件の正確かつ完全な具体化であることを***決定しなければならない。***

ADV_TDS.3 基本モジュール設計

依存性 ADV_FSP.4 完全な機能仕様

開発者アクションエレメント:

ADV_TDS.3.1D 開発者は、TOE の設計を提供しなければならない。

ADV_TDS.3.2D 開発者は、機能仕様の TSFI から TOE 設計で利用可能な分解の最下位レベルへのマッピングを提供しなければならない。

内容・提示エレメント:

ADV_TDS.3.1C 設計は、サブシステムの観点から TOE の構造を記述しなければならない。

ADV_TDS.3.2C **設計は、モジュールの観点から TSF を記述しなければならない。**

ADV_TDS.3.3C 設計は、TSF のすべてのサブシステムを識別しなければならない。

ADV_TDS.3.4C 設計は、**TSF の各サブシステムの記述を提供しなければならない。**

ADV_TDS.3.5C 設計は、TSF のすべてのサブシステム間の相互作用の記述を提供しなければならない。

ADV_TDS.3.6C **設計は、TSF のサブシステムから TSF のモジュールへのマッピングを提供しなければならない。**

ADV_TDS.3.7C 設計は、**目的の観点から**各 **SFR 実施モジュール**を記述しなければならない。

ADV_TDS.3.8C **設計は、各 SFR 実施モジュールの SFR 関連インタフェース、それらのインタフェースからの戻り値、及びその他のモジュールに対して呼び出されるインタフェースの観点から各 SFR 実施モジュールを記述しなければならない。**

ADV_TDS.3.9C 設計は、**目的及びその他のモジュールとの相互作用の観点から**各 **SFR 支援モジュールまたは SFR 非干渉モジュール**を記述しなければならない。

ADV_TDS.3.10C マッピングは、TOE 設計で記述されているすべてのふるまいが、そのふるまいを呼び出す TSFI にマッピングされていることを実証しなければならない。

評価者アクションエレメント:

ADV_TDS.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_TDS.3.2E 評価者は、設計が、すべてのセキュリティ機能要件の正確かつ完全な具体化であることを*決定しなければならない。*

ADV_TDS.4 準形式的なモジュール設計

依存性: ADV_FSP.5 追加の誤り情報を伴う完全な準形式的機能仕様

開発者アクションエレメント:

ADV_TDS.4.1D 開発者は、TOE の設計を提供しなければならない。

ADV_TDS.4.2D 開発者は、機能仕様の TSFI から TOE 設計で利用可能な分解の最下位レベルへのマッピングを提供しなければならない。

内容・提示エレメント:

ADV_TDS.4.1C 設計は、サブシステムの観点から TOE の構造を記述しなければならない。

ADV_TDS.4.2C 設計は、**各モジュールを SFR 実施、SFR 支援、または SFR 非干渉として指示し、**モジュールの観点から TSF を記述しなければならない。

ADV_TDS.4.3C 設計は、TSF のすべてのサブシステムを識別しなければならない。

ADV_TDS.4.4C 設計は、TSF の各サブシステムの記述を提供しなければならない。

ADV_TDS.4.5C 設計は、TSF のすべてのサブシステム間の相互作用の記述を提供しなければならない。

ADV_TDS.4.6C 設計は、TSF のサブシステムから TSF のモジュールへのマッピングを提供しなければならない。

ADV_TDS.4.7C 設計は、目的の観点から各 SFR 実施及び**SFR 支援**モジュールを記述しなければならない。

ADV_TDS.4.8C 設計は、各 SFR 実施モジュール及び **SFR 支援**モジュールの SFR 関連インタフェース、それらのインタフェースからの戻り値、及びその他のモジュールに対して呼び出されるインタフェースの観点から各 SFR 実施モジュール及び SFR 支援モジュールを記述しなければならない。

ADV_TDS.4.9C 設計は、目的及びその他のモジュールとの相互作用の観点から各 SFR 非干渉モジュールを記述しなければならない。

ADV_TDS.4.10C マッピングは、TOE 設計で記述されているすべてのふるまいが、そのふるまいを呼び出す TSFI にマッピングされていることを実証しなければならない。

評価者アクションエレメント:

ADV_TDS.4.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない。*

ADV_TDS.4.2E 評価者は、設計が、すべてのセキュリティ機能要件の正確かつ完全な具体化であることを*決定しなければならない。*

ADV_TDS.5 完全な準形式的モジュール設計

依存性: ADV_FSP.5 追加の誤り情報を伴う完全な準形式的機能仕様

開発者アクションエレメント:

ADV_TDS.5.1D 開発者は、TOE の設計を提供しなければならない。

ADV_TDS.5.2D 開発者は、機能仕様の TSFI から TOE 設計で利用可能な分解の最下位レベルへのマッピングを提供しなければならない。

内容・提示エレメント:

ADV_TDS.5.1C 設計は、サブシステムの観点から TOE の構造を記述しなければならない。

ADV_TDS.5.2C 設計は、各モジュールを SFR 実施、SFR 支援、または SFR 非干渉として指示し、モジュールの観点から TSF を記述しなければならない。

ADV_TDS.5.3C 設計は、TSF のすべてのサブシステムを識別しなければならない。

ADV_TDS.5.4C 設計は、TSF の各サブシステムの記述を提供しなければならない。

ADV_TDS.5.5C 設計は、TSF のサブシステム間の相互作用の記述を提供しなければならない。

ADV_TDS.5.6C 設計は、TSF のサブシステムから TSF のモジュールへのマッピングを提供しなければならない。

ADV_TDS.5.7C 設計は、各モジュールを、その**目的**、インタフェース、インタフェースからの戻り値、及び他のモジュールに対して呼び出されるインタフェースの観点から記述しなければならない。

ADV_TDS.5.8C マッピングは、TOE 設計で記述されているすべてのふるまいが、そのふるまいを呼び出す TSFI にマッピングされていることを実証しなければならない。

評価者アクションエレメント:

ADV_TDS.5.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_TDS.5.2E 評価者は、設計が、すべてのセキュリティ機能要件の正確かつ完全な具体化であることを***決定しなければならない。***

ADV_TDS.6 形式的な上位レベルの設計提示を伴う完全な準形式的モジュール設計

依存性: ADV_FSP.6 追加の形式的仕様を伴う完全な準形式的機能仕様

開発者アクションエレメント:

ADV_TDS.6.1D 開発者は、TOE の設計を提供しなければならない。

ADV_TDS.6.2D 開発者は、機能仕様の TSFI から TOE 設計で利用可能な分解の最下位レベルへのマッピングを提供しなければならない。

ADV_TDS.6.3D **開発者は、TSF サブシステムの形式的仕様を提供しなければならない。**

ADV_TDS.6.4D **開発者は、TSF サブシステムの形式的な仕様と機能仕様の形式的な仕様との間の対応の証明を提供しなければならない。**

内容・提示エレメント:

ADV_TDS.6.1C 設計は、サブシステムの観点から TOE の構造を記述しなければならない。

ADV_TDS.6.2C 設計は、各モジュールを SFR 実施、SFR 支援、または SFR 非干渉として指示し、モジュールの観点から TSF を記述しなければならない。

ADV_TDS.6.3C 設計は、TSF のすべてのサブシステムを識別しなければならない。

ADV_TDS.6.4C 設計は、TSF の各サブシステムの記述を提供しなければならない。

ADV_TDS.6.5C 設計は、TSF の**すべての**サブシステム間の相互作用の記述を提供しなければならない。

ADV_TDS.6.6C 設計は、TSF のサブシステムから TSF のモジュールへのマッピングを提供しなければならない。

ADV_TDS.6.7C 設計は、各モジュールを、その目的、インタフェース、インタフェースからの戻り値、及び他のモジュールに対して呼び出されるインタフェースの観点から記述しなければならない。

ADV_TDS.6.8C **TSF サブシステムの形式的な仕様は、適切な個所に対して非形式的で説明的なテキストで補足される形式的スタイルを使用して、TSF を記述しなければならない。**

ADV_TDS.6.9C マッピングは、TOE 設計で記述されているすべてのふるまいが、そのふるまいを呼び出す TSFI にマッピングされていることを実証しなければならない。

ADV_TDS.6.10C **TSF サブシステムの形式的仕様と機能仕様の形式的仕様間の対応の証明は、TOE 設計に記述されているすべてのふるまいがそれを呼び出している TSFI の正確かつ完全な詳細化であることを実証しなければならない。**

評価者アクションエレメント:

ADV_TDS.6.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ADV_TDS.6.2E 評価者は、設計が、すべてのセキュリティ機能要件の正確かつ完全な具体化であることを***決定しなければならない。***

13 AGD クラス: ガイダンス文書

281 ガイダンス文書クラスは、すべての利用者の役割に対するガイダンス証拠資料に関する要件を提供する。TOE をセキュアに準備して操作するためには、TOE のセキュアな取り扱いに関連するすべての側面を記述する必要がある。このクラスでは、TOE の意図しない間違った構成や、取り扱いについての可能性についても扱う。

282 多くの場合、ガイダンスは TOE の準備と操作で別々の文書として提供するか、あるいは、ソフトウェアインタフェースやハードウェアインタフェースなどを使用するエンド利用者、管理者、アプリケーションプログラマなどの様々な利用者の役割ごとに別々の文書として提供するのが適切である。

283 ガイダンス文書クラスは、利用者準備ガイダンス(ST で記述されたように、配付された TOE を運用環境においてその評価構成に移行するために行うべきこと)及び利用者操作ガイダンス(その評価構成において TOE の操作中に行うべきこと)に関係する 2 つのファミリに分割される。

284 図 12 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

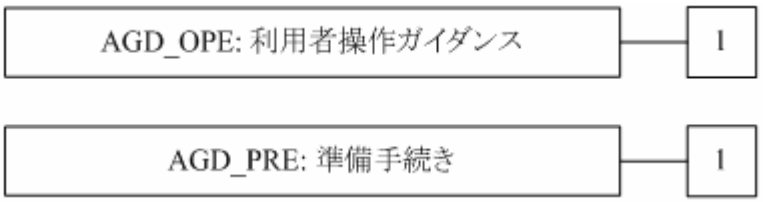

図 12 AGD: ガイダンス文書クラスのコンポーネント構成

13.1 利用者操作ガイダンス(AGD_OPE)

目的

285 利用者操作ガイダンスは、TOE の評価構成におけるすべてのタイプの TOE 利用者、つまりエンド利用者と、最大のセキュリティを得るための適正な方法で TOE を保守、管理する責任を負う者、及び TOE の外部インタフェースを使用するその他の人々(プログラマなど)に使用されることを意図して書かれた文書である。利用者操作ガイダンスは、TSF により提供されるセキュリティ機能性を記述し、(警告を含む)指示やガイドラインを提供し、TSF の理解を助け、セキュアな使用のためにセキュリティ上重要な情報と要求されるアクションを含むものである。ガイダンス証拠資料には、誤解を招く不合理なガイダンスが含まれるべきではなく、また、すべての操作モードにおけるセキュアな手続きが示されるべきである。セキュアでない状態は、容易に検出されるべきである。

286 利用者操作ガイダンスは、悪意のない利用者、管理者、アプリケーションの提供者、その他 TOE の外部インタフェースを使用する者が、TOE のセキュアな操作を理解し、意図されたとおりに使用することについて、信頼の指標を提供する。利用者ガイダンスの評価には、セキュアでないにもかかわらず、TOE の利用者が合理的にセキュアであると判断した方法で TOE が使用され得るかどうかの調査も含まれる。目的は、操作中の人的な誤りやそれ以外の誤りが、セキュリティ機能性の非活性化、無効化、または活性化の失敗を招き、それによって検出されずにセキュアでない状態に陥るリスクを最小化することである。

コンポーネントのレベル付け

287 このファミリは、ただ 1 つのコンポーネントより成る。

適用上の注釈

288 TOE によって認識され、TSF と相互作用を行うことができる各種の利用者の役割とグループが存在することができる。これらの利用者の役割とグループは、利用者操作ガイダンスで考慮されるべきである。これらは管理者と管理者以外の利用者に大きく分類され、さらに TOE の受領、受入れ、設置、保守の担当者、アプリケーションプログラマ、修正者、監査者、日常管理、エンド利用者として責任を負うものへとより明確に分類される。各役割は、広範な一連の能力を含むか、または単一の能力であることができる。

289 **AGD_OPE.1.1C** の要件には、PP/ST に記述されている TOE セキュリティ環境と運用環境のセキュリティ対策方針に関して、TOE を操作中の利用者に通知されるあらゆる警告が、利用者ガイダンスに適切に扱われているという側面が含まれている。

290 **AGD_OPE.1.3C** で採用されているセキュアな値という概念は、利用者がセキュリティパラメタを管理している場合に関連する。ガイダンスには、このようなパラメタについて、セキュアな及びセキュアでない設定が記述される必要がある。

291 **AGD_OPE.1.4C** は、利用者ガイダンスが、すべてのセキュリティ関連事象への適切な対応を記述することを要求する。セキュリティ関連事象の多くは機能を実行した結果であるが、必ずしもそうであるとは限らない(例えば、監査ログが満杯になった場合や侵入が検知された場合)。さらに、セキュリティ関連事象は、機能の特定の連鎖の結果として起こる場合や、逆に、複数のセキュリティ関連事象が 1 つの機能によって誘発される場合もある。

292 **AGD_OPE.1.7C** では、利用者ガイダンスが明確で、合理的なものであることが要求される。誤解を招くガイダンスや不合理なガイダンスは、TOE の利用者にセキュアでない TOE をセキュアであると信じさせるおそれがある。

293 誤解を招くガイダンスの例として、1つのガイダンスの指示が何通りにも解釈でき、そのうちの 1 つで、セキュアでない状態が生じるおそれのある記述が挙げられる。

294 不合理なガイダンスの例として、利用者がこのガイダンスに従うことが当然のこととして期待できないほど複雑な手順を要求するものが挙げられる。

AGD_OPE.1 利用者操作ガイダンス

依存性: ADV_FSP.1 基本機能仕様

開発者アクションエレメント:

AGD_OPE.1.1D 開発者は、利用者操作*ガイダンス*を提供しなければならない。

内容・提示エレメント:

AGD_OPE.1.1C 利用者操作*ガイダンス*は、適切な警告を含め、セキュアな処理環境で管理するべき、利用者がアクセス可能な機能と権限について、利用者の役割ごとに記述しなければならない。

AGD_OPE.1.2C 利用者操作*ガイダンス*は、**TOE** により提供された利用可能なインタフェースをセキュアな方法でどのように使用するかを利用者の役割ごとに記述しなければならない。

AGD_OPE.1.3C 利用者操作*ガイダンス*は、利用可能な機能とインタフェース、特に利用者の管理下にあるすべてのセキュリティパラメタを、必要に応じてセキュアな値を示し、利用者の役割ごとに記述しなければならない。

AGD_OPE.1.4C 利用者操作*ガイダンス*は、**TSF** の制御下にあるエンティティのセキュリティ特性の変更を含む、利用者がアクセス可能で実行が必要な機能に関連するセキュリティ関連事象の各タイプについて、利用者の役割ごとに明確に提示しなければならない。

AGD_OPE.1.5C 利用者操作*ガイダンス*は、**TOE** の操作のすべての可能なモード(障害や操作誤りの後の操作を含む)、それらの結果、及びセキュアな運用を維持するために必要なことを識別しなければならない。

AGD_OPE.1.6C 利用者操作*ガイダンス*は、**ST** に記述された運用環境のセキュリティ対策方針を満たすために従うべきセキュリティ手段を、利用者の役割ごとに記述しなければならない。

AGD_OPE.1.7C 利用者操作*ガイダンス*は、明確で、合理的なものでなければならない。

評価者アクションエレメント:

AGD_OPE.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

13.2 準備手続き(AGD_PRE)

目的

295 準備手続きは、開発者の意図したセキュアな方法で TOE が受領され、設置されたことを保証するのに有用である。準備の要件は、配付された TOE から TOE の最初の運用環境へのセキュアな移行を要求する。これには、セキュアでないにもかかわらず TOE の利用者が合理的にセキュアであると判断した方法で、TOE を構成及び設置できるかどうかの調査も含まれる。

コンポーネントのレベル付け

296 このファミリは、ただ 1 つのコンポーネントより成る。

適用上の注釈

297 これらの要件の適用は、TOE が、運用可能な状態で配付されるか、TOE 所有者のサイトで設置しなければならないかなどの側面に応じて変動することが認知されている。

298 準備手続きで扱われる最初のプロセスは、開発者の配付手続きに従って受領した TOE の消費者のセキュアな受入れである。開発者が配付手続きを定義していない場合、受入れのセキュリティが別の方法で保証されなければならない。

299 TOE の設置には、ST で提供されている運用環境のセキュリティ対策方針に準拠した状態に、TOE の運用環境を移行することが含まれる。

300 スマートカードなど設置が不要の場合もある。このような場合には、設置手順に対する要求や解析は不適当となりうる。

301 この保証ファミリの要件は、利用者操作ガイダンス(AGD_OPE)ファミリの要件とは別に提示される。これは、準備手続きがまれにしか(おそらく 1 回限り)使用されないからである。

AGD_PRE.1 準備手続き

依存性: なし

開発者アクションエレメント:

AGD_PRE.1.1D **開発者は、準備手続きを含めて TOE を提供しなければならない。**

内容・提示エレメント:

AGD_PRE.1.1C **準備手続きは、開発者の配付手続きに従って配付された TOE のセキュアな受入れに必要なすべてのステップを記述しなければならない。**

AGD_PRE.1.2C **準備手続きには、TOE のセキュアな設置、及び ST に記述された運用環境のセキュリティ対策方針に従った運用環境のセキュアな準備に必要なすべてのステップを記述しなければならない。**

評価者アクションエレメント:

AGD_PRE.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

AGD_PRE.1.2E 評価者は、**TOE** が運用に向けてセキュアに準備されることを確認するために、準備手続きを適用しなければならない。

14 ALC クラス: ライフサイクルサポート

302 ライフサイクルサポートは、TOE の開発及び保守中に、TOE を改良するプロセスに統制と管理を確立するための要件である。セキュリティ分析と証拠の作成が、開発と保守アクティビティの必須部分として標準的に行われるならば、TOE のセキュリティ要件と TOE との対応の信頼度はより大きくなる。

303 製品のライフサイクルでは、TOE が開発環境と利用者環境のどちらに置かれているかということよりも、開発者と利用者のどちらの責任の下に置かれているかが区別される。移行の時点は、TOE が利用者に引き渡されるときである。また、これは ALC クラスから AGD クラスへの移行の時点でもある。

304 ALC クラスは 7 つのファミリで構成される。ライフサイクル定義(ALC_LCD)は、TOE ライフサイクルの上位レベルの記述であり、CM 能力(ALC_CMC)は構成要素の管理に関するより詳細な記述である。CM 範囲(ALC_CMS)は、構成要素の最小限のセットが、規定された方法で管理されることを要求する。開発セキュリティ(ALC_DVS)は開発者の物理的、手続き的、人的、及びその他のセキュリティ手段に関係し、ツールと技法(ALC_TAT)は開発者が使用する開発ツールと実装標準に関係する。欠陥修正(ALC_FLR)はセキュリティ欠陥の取り扱いに関係する。配付(ALC_DEL)は、消費者への TOE の配付に使用される手続きを定義する。TOE の開発中に発生する配付プロセスは転送と呼ばれ、このクラスの他のファミリでの統合及び受入れ手続きで扱われる。

305 このクラスを通して、開発及び関連用語(開発者、開発する)が、より一般的な意味で開発と*製造*を含むように意図されている一方で、製造は、実装表現を最終的な TOE に変換するプロセスのみを意味する。

306 図 13 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

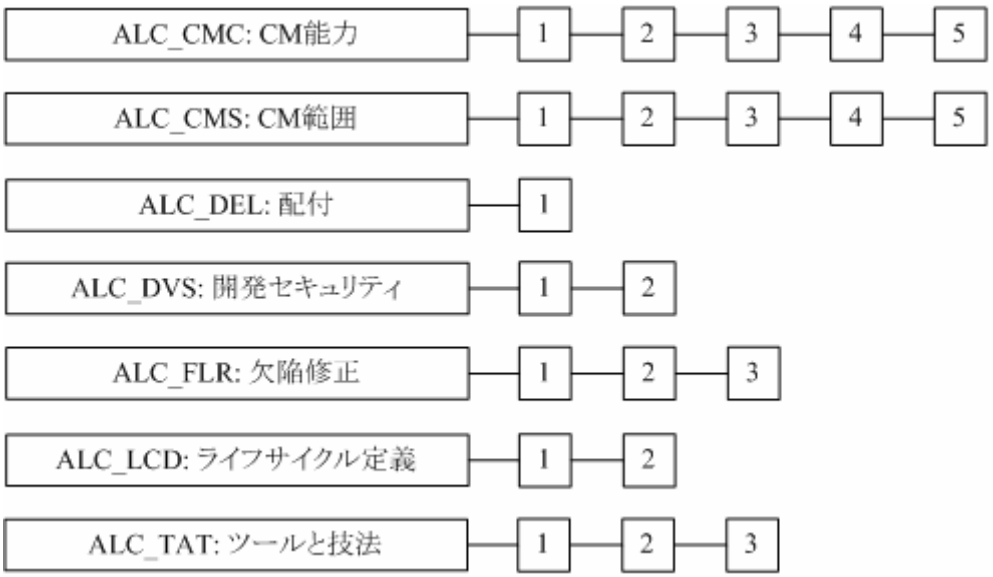

図 13 ALC: ライフサイクルサポートクラスのコンポーネント構成

14.1 CM 能力(ALC_CMC)

目的

307 構成管理(CM)は、TOE が SFR を満たしていることの保証を高めるための手段である。CM は、TOE 及びその関連情報の改良と修正のプロセスにおいて、統制と管理を要求することにより、これを確立する。CM システムは、あらゆる変更を追跡する手段を提供し、すべての変更が許可されたものであることを保証することにより、TOE の部分の完全性を保証する。

308 このファミリの目的は、開発者の CM システムに一定の能力を要求することである。これらは、構成要素の事故による、または許可されない修正が起きる可能性を削減しなければならない。CM システムは、設計の初期段階から後続する保守作業までを通して、TOE の完全性を保証するべきである。

309 自動化された CM ツール導入の目的は、CM システムの効率化である。自動化された CM システムも手作業の CM システムもバイパスされたり、無視されたり、許可されていない修正を防止するには不十分であるが、自動化されたシステムの方が、人の誤りや不注意に対し影響を受けにくい。

310 このファミリは、以下の目的を含んでいる:

- 消費者に送る前に、TOE が正確で完全であることを保証する;
- 評価中に、構成要素の漏れがないことを保証する;
- TOE 構成要素の許可されない修正、追加、削除を防止する。

コンポーネントのレベル付け

311 このファミリのコンポーネントは、CM システムの能力、CM 証拠資料の適用範囲、及び開発者により提供された証拠に基づいて、レベル付けされている。

適用上の注釈

312 CM は、設計段階の初期から適用され、将来に渡って継続することが望まれるが、一方このファミリは、CM が評価の終了までの期間、適切であり、使用されていることを要求する。

313 TOE がある製品のサブセットである場合、このファミリの要件は、製品全体に適用されるのではなく、TOE 構成要素のみに適用される。

314 開発者が、異なるライフサイクルフェーズ(例えば、開発、製造、及び/または最終製品)に対して別々の CM システムを使用する場合は、それらすべてについて証拠資料が要求される。評価上の目的から、個々の CM システムは、基準で扱われる全体的な CM システムの部分とみなされるべきである。

315 同様に、TOE の各部分が異なる開発者または異なるサイトで製造される場合は、それぞれの場所で使用されている CM システムは、基準で扱われる全体的な CM システムの部分とみなされるべきである。この状況では、統合上の側面も考慮されなければならない。

316 このファミリのエレメントのいくつかは、構成要素を参照する。これらのエレメントは、構成リストで識別されるすべての要素に課せられた CM 要件を識別するが、リストの内容は開発者の裁量に任せている。CM 範囲(ALC_CMS)は、構成リストに含まれ CM によってカバーされなければならない特定の要素を識別することでこの裁量を制限するために使用されることができる。

317 **ALC_CMC.2.3C** は、CM システムが、すべての構成要素を一意に識別することへの要件である。この要件は、構成要素への修正に対し、構成要素に新たな一意の識別情報を割り当てることを含む。

318 **ALC_CMC.3.8C** は、CM システムが CM 計画に従って機能していることを実証する証拠への要件である。このような証拠の例は、CM システムが出力する画面のスナップショットや監査証跡のような証拠資料、または開発者による CM システムの詳細な実証である。評価者は、CM システムが CM 計画に従って機能していることを示すのにこの証拠が十分であるかを決定する責任がある。

319 **ALC_CMC.4.5C** は、TOE の生成をサポートするための自動化された手段を CM システムが提供することへの要件である。これは、正しい構成要素が TOE の生成に使用されているかを決定することを助けるための自動化された手段を CM システムが提供することを要求する。

320 **ALC_CMC.5.10C** は、TOE とその前のバージョンとの間の変更を明確にするための自動化された手段を CM システムが提供することへの要件である。TOE の以前のバージョンが存在しない場合でも、TOE と TOE の将来のバージョンとの間の変更を明確にするための自動化された手段を開発者が提供する必要がある。

ALC_CMC.1 TOE のラベル付け

依存性: ALC_CMS.1 TOE の CM 範囲

目的

321 TOE のどの段階のもの(instance)が評価されているかの観点から、曖昧さのないことを保証するために、一意の参照が要求される。TOE をその参照でラベル付けすることは、TOE の利用者が TOE のどの段階のものを使用しているかを知ることができることを保証する。

開発者アクションエレメント:

ALC_CMC.1.1D **開発者は、TOE 及び TOE の参照を提供しなければならない。**

内容・提示エレメント:

ALC_CMC.1.1C **TOE は、その一意の参照でラベル付けされなければならない。**

評価者アクションエレメント:

ALC_CMC.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ALC_CMC.2 CM システムの使用

依存性: ALC_CMS.1 TOE の CM 範囲

目的

322 TOE のどの段階のもの(instance)が評価されているかの観点から、曖昧さのないことを保証するために、一意の参照が要求される。TOE をその参照でラベル付けすることは、TOE の利用者が TOE のどの段階のものを使用しているかを知ることができることを保証する。

323 構成要素の一意の識別は、TOE の構成をより明快に理解することにつながり、その結果、TOE のための評価要件の対象になる要素を決定することを助ける。

324 CM システムの使用は、構成要素が管理された方法で維持されることの保証を高める。

開発者アクションエレメント:

ALC_CMC.2.1D 開発者は、TOE 及び TOE の参照を提供しなければならない。

ALC_CMC.2.2D **開発者は、CM 証拠資料を提供しなければならない。**

ALC_CMC.2.3D **開発者は、CM システムを使用しなければならない。**

内容・提示エレメント:

ALC_CMC.2.1C TOE は、その一意の参照でラベル付けされなければならない。

ALC_CMC.2.2C **CM 証拠資料は、構成要素を一意に識別する方法を記述しなければならない。**

ALC_CMC.2.3C **CM システムは、すべての構成要素を一意に識別しなければならない。**

評価者アクションエレメント:

ALC_CMC.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ALC_CMC.3 許可の管理

依存性: ALC_CMS.1 TOE の CM 範囲
ALC_DVS.1 セキュリティ手段の識別

目的

325 TOE のどの段階のもの(instance)が評価されているかの観点から、曖昧さのないことを保証するために、一意の参照が要求される。TOE をその参照でラベル付けすることは、TOE の利用者が TOE のどの段階のものを使用しているかを知ることができることを保証する。

326 構成要素の一意の識別は、TOE の構成をより明快に理解することにつながり、その結果、TOE のための評価要件の対象になる要素を決定することを助ける。

327 CM システムの使用は、構成要素が管理された方法で維持されることの保証を高める。

328 許可されていない修正が TOE に対して行われないことを保証する管理(「CM アクセス制御」)を提供すること、及び CM システムの適切な機能性と使用を保証することは、TOE の完全性を維持することを助ける。

開発者アクションエレメント:

ALC_CMC.3.1D 開発者は、TOE 及び TOE の参照を提供しなければならない。

ALC_CMC.3.2D 開発者は、CM 証拠資料を提供しなければならない。

ALC_CMC.3.3D 開発者は、CM システムを使用しなければならない。

内容・提示エレメント:

ALC_CMC.3.1C TOE は、その一意の参照でラベル付けされなければならない。

ALC_CMC.3.2C CM 証拠資料は、構成要素を一意に識別する方法を記述しなければならない。

ALC_CMC.3.3C CM システムは、すべての構成要素を一意に識別しなければならない。

ALC_CMC.3.4C **CM システムは、許可された変更のみが構成要素に対して行われる手段を提供しなければならない。**

ALC_CMC.3.5C **CM 証拠資料は、CM 計画を含まなければならない。**

ALC_CMC.3.6C **CM 計画は、TOE の開発に対して CM システムがどのように使用されるかを記述しなければならない。**

ALC_CMC.3.7C **証拠は、すべての構成要素が CM システム下で維持されていることを実証しなければならない。**

ALC_CMC.3.8C **CM システムが、CM 計画に従って機能していることを実証しなければならない。**

評価者アクションエレメント:

ALC_CMC.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ALC_CMC.4 製造サポート、受入れ手続き、及び自動化

依存性: ALC_CMS.1 TOE の CM 範囲
ALC_DVS.1 セキュリティ手段の識別
ALC_LCD.1 開発者によるライフサイクルモデルの定義

目的

329 TOE のどの段階のもの(instance)が評価されているかの観点から、曖昧さのないことを保証するために、一意の参照が要求される。TOE をその参照でラベル付けすることは、TOE の利用者が TOE のどの段階のものを使用しているかを知ることができることを保証する。

330 構成要素の一意の識別は、TOE の構成をより明快に理解することにつながり、その結果、TOE のための評価要件の対象になる要素を決定することを助ける。

331 CM システムの使用は、構成要素が管理された方法で維持されることの保証を高める。

332 許可されていない修正が TOE に対して行われないことを保証する管理(「CM アクセス制御」)を提供すること、及び CM システムの適切な機能性と使用を保証することは、TOE の完全性を維持することを助ける。

333 受入れ手続きの目的は、TOE の部分が適切な品質であることを保証すること、及び構成要素のいかなる生成や修正も許可されていることを確認することである。受入れ手続きは、統合プロセス及び TOE のライフサイクル管理における不可欠な要素である。

334 構成要素が複雑な開発環境では、自動化ツールなしでの変更の管理は困難である。特に、このような自動化ツールでは、開発中発生する多数の変更をサポートし、これらの変更が許可されたものであることを保証できることが必要とされる。このコンポーネントの目的は、構成要素が、自動化された手段で管理されることを保証することである。TOE が複数の開発者によって開発される場合、つまり統合を行う必要がある場合は、自動化ツールの使用が適切である。

335 製造サポート手続きは、特に複数の開発者が関与し、統合プロセスの実行が必要な状況で、管理された構成要素のセットからの TOE の生成が、許可された方法で正しく実行されることを保証する助けとなる。

開発者アクションエレメント:

ALC_CMC.4.1D 開発者は、TOE 及び TOE の参照を提供しなければならない。

ALC_CMC.4.2D 開発者は、CM 証拠資料を提供しなければならない。

ALC_CMC.4.3D 開発者は、CM システムを使用しなければならない。

内容・提示エレメント:

ALC_CMC.4.1C TOE は、その一意の参照でラベル付けされなければならない。

ALC_CMC.4.2C CM 証拠資料は、構成要素を一意に識別する方法を記述しなければならない。

ALC_CMC.4.3C CM システムは、すべての構成要素を一意に識別しなければならない。

ALC_CMC.4.4C CM システムは、許可された変更のみが構成要素に対して行われる**自動化された**手段を提供しなければならない。

ALC_CMC.4.5C **CM システムは、自動化された手段によって TOE の製造をサポートしなければならない。**

ALC_CMC.4.6C CM 証拠資料は、CM 計画を含まなければならない。

ALC_CMC.4.7C CM 計画は、TOE の開発に対して CM システムがどのように使用されるかを記述しなければならない。

ALC_CMC.4.8C **CM 計画は、改変もしくは新規に生成された構成要素を TOE の一部として受け入れるための手続きを記述しなければならない。**

ALC_CMC.4.9C 証拠は、すべての構成要素が CM システム下で維持されていることを実証しなければならない。

ALC_CMC.4.10C CM システムが、CM 計画に従って機能していることを証拠により実証しなければならない。

評価者アクションエレメント:

ALC_CMC.4.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ALC_CMC.5 進んだサポート

依存性: ALC_CMS.1 TOE の CM 範囲
ALC_DVS.2 セキュリティ手段の十分性
ALC_LCD.1 開発者によるライフサイクルモデルの定義

目的

336 TOE のどの段階のもの(instance)が評価されているかの観点から、曖昧さのないことを保証するために、一意の参照が要求される。TOE をその参照でラベル付けすることは、TOE の利用者が TOE のどの段階のものを使用しているかを知ることができることを保証する。

337 構成要素の一意の識別は、TOE の構成をより明快に理解することにつながり、その結果、TOE のための評価要件の対象になる要素を決定することを助ける。

338 CM システムの使用は、構成要素が管理された方法で維持されることの保証を高める。

339 許可されていない修正が TOE に対して行われないことを保証する管理(「CM アクセス制御」)を提供すること、及び CM システムの適切な機能性と使用を保証することは、TOE の完全性を維持することを助ける。

340 受入れ手続きの目的は、TOE の部分が適切な品質であることを保証すること、及び構成要素のいかなる生成や修正も許可されていることを確認することである。受入れ手続きは、統合プロセス及び TOE のライフサイクル管理における不可欠な要素である。

341 構成要素が複雑な開発環境では、自動化ツールなしでの変更の管理は困難である。特に、このような自動化ツールでは、開発中発生する多数の変更をサポートし、これらの変更が許可されたものであることを保証できることが必要とされる。このコンポーネントの目的は、構成要素が、自動化された手段で管理されることを保証することである。TOE が複数の開発者によって開発される場合、つまり統合を行う必要がある場合は、自動化ツールの使用が適切である。

342 製造サポート手続きは、特に複数の開発者が関与し、統合プロセスの実行が必要な状況で、管理された構成要素のセットからの TOE の生成が、許可された方法で正しく実行されることを保証する助けとなる。

343 CM システムが、TOE の生成元である実装表現のバージョンを識別できることを要求することは、その資材の完全性が、適切な技術的、物理的、及び手続き的保護手段により保護されることを保証する助けとなる。

344 TOE のバージョン間の変更を確認する自動化された手段を提供すること、及び他の構成要素の修正によって影響を受ける構成要素を識別することは、TOE の連続するバージョン間の変更による影響を決定する際に役立つ。その結果、このことは、TOE に対する変更の結果がすべての構成要素で相互に矛盾がないかどうかを決定するために、有益な情報を提供できる。

開発者アクションエレメント:

ALC_CMC.5.1D 開発者は、TOE 及び TOE の参照を提供しなければならない。

ALC_CMC.5.2D 開発者は、CM 証拠資料を提供しなければならない。

ALC_CMC.5.3D 開発者は、CM システムを使用しなければならない。

内容・提示エレメント:

ALC_CMC.5.1C TOE は、その一意の参照でラベル付けされなければならない。

ALC_CMC.5.2C CM 証拠資料は、構成要素を一意に識別する方法を記述しなければならない。

ALC_CMC.5.3C **CM 証拠資料は、受入れ手続きが、すべての構成要素に対する十分で適切な変更のレビューを提供することを正当化しなければならない。**

ALC_CMC.5.4C CM システムは、すべての構成要素を一意に識別しなければならない。

ALC_CMC.5.5C CM システムは、許可された変更のみが構成要素に対して行われる自動化された手段を提供しなければならない。

ALC_CMC.5.6C CM システムは、自動化された手段によって TOE の製造をサポートしなければならない。

ALC_CMC.5.7C **CM システムは、構成要素を CM に受け入れる責任のある人はその開発者でないことを保証しなければならない。**

ALC_CMC.5.8C **CM システムは、TSF を構成する構成要素を識別しなければならない。**

ALC_CMC.5.9C **CM システムは、監査証跡に発信者、日時を含んでいる自動化された手段により、TOE のすべての変更についての監査をサポートしなければならない。**

ALC_CMC.5.10C **CM システムは、ある構成要素の変更により影響を受けるすべての他の構成要素を特定するための、自動化された手段を提供しなければならない。**

ALC_CMC.5.11C **CM システムは、TOE の生成元である実装表現のバージョンを識別できなければならない。**

ALC_CMC.5.12C CM 証拠資料は、CM 計画を含まなければならない。

ALC_CMC.5.13C CM 計画は、TOE の開発に対して CM システムがどのように使用されるかを記述しなければならない。

ALC_CMC.5.14C CM 計画は、改変もしくは新規に生成された構成要素を TOE の一部として受け入れるための手続きを記述しなければならない。

ALC_CMC.5.15C 証拠は、すべての構成要素が CM システム下で維持されていることを実証しなければならない。

ALC_CMC.5.16C CM システムが、CM 計画に従って機能していることを証拠により実証しなければならない。

評価者アクションエレメント:

ALC_CMC.5.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ALC_CMC.5.2E 評価者は、製造サポート手続きを適用するとテストアクティビティのために開発者によって提供された **TOE** になることを*決定しなければならない*。

14.2 CM 範囲(ALC_CMS)

目的

345 このファミリの目的は、構成要素として含まれる要素を識別することであり、それ故に CM 能力(ALC_CMC)の CM 要件下に置かれる。これらの追加要素に対して構成管理を適用すれば、TOE の完全性が維持されるという追加の保証を提供する。

コンポーネントのレベル付け

346 このファミリのコンポーネントは、以下のどれが構成要素として含まれることを要求されるかに基づいてレベル付けされている。TOE 及び SAR が要求する評価証拠;TOE の部分;実装表現;セキュリティ欠陥;開発ツール及び関連情報。

適用上の注釈

347 CM 範囲(ALC_CMS)は、構成要素のリスト及びこのリスト上の各要素が CM 下に置かれることを要求するが、CM 能力(ALC_CMC)は、構成リストの内容を開発者の裁量に任せる。CM 範囲(ALC_CMS)は、構成リストに組み込まれ CM 能力(ALC_CMC)の CM 要件下に置かれなければならない要素を識別することで、この裁量を制限する。

ALC_CMS.1 TOE の CM 範囲

依存性: なし

目的

348 CM システムは、CM 下に置かれている要素に対してのみ変更を管理することができる(すなわち構成リストで識別されている構成要素)。TOE 自体及び ST の他の SAR が要求する評価証拠を CM 下に置くことは、これらが適切な許可を伴う管理された方法で修正がなされることの保証を提供する。

適用上の注釈

349 **ALC_CMS.1.1C** は、TOE 自体及び ST の他の SAR が要求する評価証拠が、構成リストに組み込まれ、それによって CM 能力(ALC_CMC)の CM 要件の影響下に置かれることへの要件である。

開発者アクションエレメント:

ALC_CMS.1.1D **開発者は、TOE の構成リストを提供しなければならない。**

内容・提示エレメント:

ALC_CMS.1.1C **構成リストは、TOE 自体、及び SAR が要求する評価証拠を含まなければならない。**

ALC_CMS.1.2C **構成リストは、構成要素を一意に識別しなければならない。**

評価者アクションエレメント:

ALC_CMS.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ALC_CMS.2 TOE の一部の CM 範囲

依存性: なし

目的

350 CM システムは、CM 下に置かれている要素に対してのみ変更を管理することができる(すなわち構成リストで識別されている構成要素)。TOE 自体、TOE を構成する部分、及び他の SAR が要求する評価証拠を CM 下に置くことは、これらが適切な許可を伴う管理された方法で修正がなされることの保証を提供する。

適用上の注釈

351 **ALC_CMS.2.1C** は、TOE(消費者に配付されるすべての部分、例えばハードウェア部品や実行可能ファイル)を構成する部分が構成リストに組み込まれ、それによって CM 能力(ALC_CMC)の CM 要件の影響下に置かれることへの要件である。

352 **ALC_CMS.2.3C** は、構成リストが各 TSF 関連構成要素の開発者を示すことへの要件である。ここでの「開発者」は人を表すのではなく、要素の開発に責任がある組織を表す。

開発者アクションエレメント:

ALC_CMS.2.1D 開発者は、TOE の構成リストを提供しなければならない。

内容・提示エレメント:

ALC_CMS.2.1C 構成リストは、TOE 自体、SAR が要求する評価証拠、**及び TOE を構成する部分**を含まなければならない。

ALC_CMS.2.2C 構成リストは、構成要素を一意に識別しなければならない。

ALC_CMS.2.3C **各 TSF 関連の構成要素に対して、構成リストはその要素の開発者を示さなければならない。**

評価者アクションエレメント:

ALC_CMS.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ALC_CMS.3 実装表現の CM 範囲

依存性: なし

目的

353 CM システムは、CM 下に置かれている要素に対してのみ変更を管理することができる(すなわち構成リストで識別されている構成要素)。TOE 自体、TOE を構成する部分、TOE 実装表現、及び他の SAR が要求する評価証拠を CM 下に置くことは、これらが適切な許可を伴う管理された方法で修正がなされることの保証を提供する。

適用上の注釈

354 **ALC_CMS.3.1C** は、TOE 実装表現が構成要素のリストに組み込まれ、それによって CM 能力(ALC_CMC)の CM 要件の影響下に置かれることへの要件である。

開発者アクションエレメント:

ALC_CMS.3.1D 開発者は、TOE の構成リストを提供しなければならない。

内容・提示エレメント:

ALC_CMS.3.1C 構成リストは、TOE 自体、SAR が要求する評価証拠、TOE を構成する部分、**及び実装表現**を含まなければならない。

ALC_CMS.3.2C 構成リストは、構成要素を一意に識別しなければならない。

ALC_CMS.3.3C 各 TSF 関連の構成要素に対して、構成リストはその要素の開発者を示さなければならない。

評価者アクションエレメント:

ALC_CMS.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ALC_CMS.4 問題追跡の CM 範囲

依存性: なし

目的

355 CM システムは、CM 下に置かれていた要素に対してのみ変更を管理することができる(すなわち構成リストで識別されている構成要素)。TOE 自体、TOE を構成する部分、TOE 実装表現、及び他の SAR が要求する評価証拠を CM 下に置くことは、これらが適切な許可を伴う管理された方法で修正がなされることの保証を提供する。

356 CM 下にセキュリティ欠陥を置くことは、セキュリティ欠陥報告が紛失したり、忘れられたりすることがなく、開発者がセキュリティ欠陥をその解決まで追跡することを保証する。

適用上の注釈

357 **ALC_CMS.4.1C** は、セキュリティ欠陥が構成リストに組み込まれ、それによって CM 能力(ALC_CMC)の CM 要件の影響下に置かれることへの要件である。これは、現状のセキュリティ欠陥の詳細だけでなく、以前のセキュリティ欠陥とその解決の情報が維持されることを要求する。

開発者アクションエレメント:

ALC_CMS.4.1D 開発者は、TOE の構成リストを提供しなければならない。

内容・提示エレメント:

ALC_CMS.4.1C 構成リストは、TOE 自体、SAR が要求する評価証拠、TOE を構成する部分、実装表現、**及びセキュリティ欠陥報告及び解決ステータス**を含まなければならない。

ALC_CMS.4.2C 構成リストは、構成要素を一意に識別しなければならない。

ALC_CMS.4.3C 各 TSF 関連の構成要素に対して、構成リストはその要素の開発者を示さなければならない。

評価者アクションエレメント:

ALC_CMS.4.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない。*

ALC_CMS.5 開発ツールの CM 範囲

依存性: なし

目的

358 CM システムは、CM 下に置かれていた要素に対してのみ変更を管理することができる(すなわち構成リストで識別されている構成要素)。TOE 自体、TOE を構成する部分、TOE 実装表現、及び他の SAR が要求する評価証拠を CM 下に置くことは、これらが適切な許可を伴う管理された方法で修正がなされることの保証を提供する。

359 CM 下にセキュリティ欠陥を置くことは、セキュリティ欠陥報告が紛失したり、忘れられたりすることがなく、また、開発者がセキュリティ欠陥をその解決まで追跡することを保証する。

360 開発ツールは、品質の高いバージョンの TOE の生成を保証するのに重要な役割を持つ。そのため、これらのツールに対する修正を管理することは重要である。

適用上の注釈

361 **ALC_CMS.5.1C** は、開発ツール及びその他の関連情報が構成要素のリストに組み込まれ、それによって CM 能力(ALC_CMC)の CM 要件の影響下に置かれることへの要件である。開発ツールの例として、プログラミング言語とコンパイラが挙げられる。TOE の生成に付随する情報(コンパイラオプション、生成オプション、及び構築オプションなど)が、開発ツールに関連する情報の例である。

開発者アクションエレメント:

ALC_CMS.5.1D 開発者は、TOE の構成リストを提供しなければならない。

内容・提示エレメント:

ALC_CMS.5.1C 構成リストは、TOE 自体、SAR が要求する評価証拠、TOE を構成する部分、実装表現、セキュリティ欠陥報告及び解決状況、**及び開発ツールと関連情報**を含まなければならない。

ALC_CMS.5.2C 構成リストは、構成要素を一意に識別しなければならない。

ALC_CMS.5.3C 各 TSF 関連の構成要素に対して、構成リストはその要素の開発者を示さなければならない。

評価者アクションエレメント:

ALC_CMS.5.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない。*

14.3 配付(ALC_DEL)

目的

362 このファミリの関心事は、完成した TOE の開発環境から利用者の責任の下へのセキュアな転送である。

363 配付要件は、システム管理及び配送設備並びに TOE が利用者に配送される際に TOE のセキュリティが維持されるという保証を提供するのに必要な手段を詳述する手続きを要求する。TOE の確実な配送のため、TOE の配送に用いられる手続きは、配付中の TOE のセキュリティに関係して PP/ST で識別された対策方針を記述する。

コンポーネントのレベル付け

364 このファミリは、ただ 1 つのコンポーネントより成る。保護レベルの上昇は、脆弱性分析(AVA_VAN)ファミリで想定される攻撃能力に相応する配付手続きを要求することにより確立される。

適用上の注釈

365 下請業者から開発者への転送、または異なる開発サイト間の転送は、ここでは考慮されず、開発セキュリティ(ALC_DVS)ファミリで考慮される。

366 配付フェーズは、利用者の責任下への TOE の転送をもって終了する。これは、TOE の利用者の場所への到着と必ずしも一致しない。

367 配付手続きは、適用できる場合、以下のような論点を考慮するべきである:

- 消費者の受け取った TOE が評価済みバージョンの TOE と正確に一致することを保証する;
- 現行のバージョンの TOE に対するあらゆる改ざんを避けるまたは検出する;
- 誤ったバージョンの TOE の送付を防止する;
- 消費者に対し、TOE の配送に関する不要な知識を与えない。潜在的な攻撃者に、配付のタイミングと方法を知られてはならない場合がある;
- 配付中に TOE が横取りされるのを避けるまたは検出する;
- TOE の配送が遅らされるまたは止められるのを避ける。

368 配付手続きは、これらの論点によって暗示されている受信者のアクションを含むべきである。これらの暗黙のアクションの一貫した記述は、存在する場合は、準備手続き(AGD_PRE)ファミリで検査される。

ALC_DEL.1 配付手続き

依存性: なし

開発者アクションエレメント:

ALC_DEL.1.1D 開発者は、**TOE** またはその一部を消費者に配付するための手続きに関する証拠資料を提出しなければならない。

ALC_DEL.1.2D 開発者は、配付手続きを使用しなければならない。

内容・提示エレメント:

ALC_DEL.1.1C 配付証拠資料は、**TOE** のバージョンを消費者に配送するときにセキュリティを維持するために必要なすべての手続きを記述しなければならない。

評価者アクションエレメント:

ALC_DEL.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

14.4 開発セキュリティ(ALC_DVS)

目的

369 開発セキュリティは、TOE 及びその構成部分を保護するために開発環境で使用される、物理的、手続き的、人的、及びその他のセキュリティ手段に関係する。開発セキュリティは、開発場所の物理的セキュリティや開発要員の選定手続きを含む。

コンポーネントのレベル付け

370 このファミリのコンポーネントは、セキュリティ手段が十分であることの正当化が要求されるかどうかに基づいて、レベル付けされている。

適用上の注釈

371 このファミリは、開発者サイトに存在する脅威を除去するまたは減少させるための手段を扱う。

372 評価者は、開発セキュリティの証拠を評定するためにサイトを訪問するべきである。これには、TOE の開発及び製造に関わる下請け業者のサイトも含まれる場合がある。訪問を行わないという決定は評価監督機関と合意されなければならない。

373 開発セキュリティは TOE の保守を扱っており、そのため評価の完了後に関係する内容もあるが、開発セキュリティ(ALC_DVS)の要件は、開発セキュリティ手段が評価時点で適切であることのみを特定する。さらに、開発セキュリティ(ALC_DVS)は、評価完了後に、開発セキュリティ手段を将来的に適用するというスポンサーの意図に関連する要件を含んでいない。

374 機密性は、開発環境において TOE を保護するための論点となるとは限らない。用語「必要がある」(necessary)を使用している場合は、適切な保護手段の選択ができる。

ALC_DVS.1 セキュリティ手段の識別

依存性: なし

開発者アクションエレメント:

ALC_DVS.1.1D **開発者は、開発セキュリティ証拠資料を作成しなければならない。**

内容・提示エレメント:

ALC_DVS.1.1C **開発セキュリティ証拠資料は、開発環境での TOE の設計及び実装の機密性と完全性を保護するために必要となる、物理的、手続き的、人的、及びその他のセキュリティ手段をすべて記述しなければならない。**

評価者アクションエレメント:

ALC_DVS.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ALC_DVS.1.2E **評価者は、セキュリティ手段が適用されていることを*確認しなければならない*。**

ALC_DVS.2 セキュリティ手段の十分性

依存性: なし

開発者アクションエレメント:

ALC_DVS.2.1D 開発者は、開発セキュリティ証拠資料を作成しなければならない。

内容・提示エレメント:

ALC_DVS.2.1C 開発セキュリティ証拠資料は、開発環境での TOE の設計及び実装の機密性と完全性を保護するために必要となる、物理的、手続き的、人的、及びその他のセキュリティ手段をすべて記述しなければならない。

ALC_DVS.2.2C **開発セキュリティ証拠資料は、セキュリティ手段が、TOE の機密性と完全性を維持するうえで、必要な保護レベルを提供することを正当化しなければならない。**

評価者アクションエレメント:

ALC_DVS.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ALC_DVS.2.2E 評価者は、セキュリティ手段が適用されていることを***確認しなければならない。***

14.5 欠陥修正(ALC_FLR)

目的

375 欠陥修正は、発見されたセキュリティの欠陥が開発者により追跡され訂正されることを要求する。TOE 評価時に将来も欠陥修正手続きが遵守されることを決定することはできないが、開発者が適切に、欠陥を追跡及び訂正し、欠陥情報と訂正を配付するための方針と手続きを評価することは可能である。

コンポーネントのレベル付け

376 このファミリのコンポーネントは、欠陥修正手続きの対象範囲の拡大と、欠陥修正方針の厳密さに基づいて、レベル付けされている。

適用上の注釈

377 このファミリは、TOE の開発者に TOE の欠陥を追跡及び訂正することを要求することにより、TOE が将来に渡って維持継続されることの保証を提供する。さらに、欠陥の訂正を配付するための要件も含んでいる。ただし、このファミリは、現在の評価を超えた評価要求を課すものではない。

378 TOE 利用者は、セキュリティ欠陥に対する処置を受け取る及び実装することに責任を負う利用者組織において中心であると考えられる。これは必ずしも個々の利用者ではなく、セキュリティ欠陥の取扱いに責任を負う、組織的な代表者であってもよい。用語「TOE 利用者」の使用は、異なる組織が個々の利用者でもよいしあるいは中央管理機関によって行われてもよい欠陥報告を扱うための異なる手続きを持っていることを認識する。

379 欠陥修正手続きは、可能性のあるすべてのタイプの欠陥についての対処方法を記述しなければならない。これらの欠陥は、開発者、TOE の利用者、あるいは TOE について熟知している他の機関によって報告されるかもしれない。欠陥によっては、直ちに修正できない場合がある。欠陥が修正できず、他の(例えば、手続き的な)手段が取られなければならない場合もありうる。提供された証拠資料は、運用サイトに修正を提供したり、修正が遅れている(その間何をすればよいか)または修正ができない欠陥に関する情報を提供したりする手続きを含まなければならない。

380 TOE のリリース後に適用される変更は、評価されずに示されるが、元の評価の一部の情報が適用される場合もある。したがって、このファミリの中で使用される語句「TOE のリリース」は、変更が適用され認証が済んだ TOE のリリースである製品のバージョンのことをいう。

ALC_FLR.1 基本的な欠陥修正

依存性: なし

開発者アクションエレメント:

ALC_FLR.1.1D **開発者は、TOE 開発者に対する欠陥修正手続きの証拠資料を提出しなければならない。**

内容・提示エレメント:

ALC_FLR.1.1C **欠陥修正手続き証拠資料は、TOE のリリースごとに報告されたすべてのセキュリティ欠陥を追跡するのに使用する手続きを記述しなければならない。**

ALC_FLR.1.2C **欠陥修正手続きは、欠陥の訂正の調査状況の記述とともに、各々のセキュリティ欠陥の性質と影響の記述が提供されることを要求しなければならない。**

ALC_FLR.1.3C **欠陥修正手続きは、各々のセキュリティ欠陥の訂正アクションが識別されることを要求しなければならない。**

ALC_FLR.1.4C **欠陥修正手続き証拠資料は、TOE 利用者に、欠陥情報、訂正、及び訂正アクションについてのガイダンスを提供するために使用する方法を記述しなければならない。**

評価者アクションエレメント:

ALC_FLR.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ALC_FLR.2 欠陥報告手続き

依存性: なし

目的

381 開発者が TOE 利用者からのセキュリティ欠陥報告に基づいて適切に行動することができ、かつ誰に訂正処置を送るかを知るために、TOE 利用者は、開発者へセキュリティ欠陥報告を提出する方法を理解する必要がある。開発者から TOE 利用者への欠陥修正ガイダンスは、TOE 利用者がこの重要な情報に気づくことを保証する。

開発者アクションエレメント:

ALC_FLR.2.1D 開発者は、TOE 開発者に対する欠陥修正手続きの証拠資料を提出しなければならない。

ALC_FLR.2.2D **開発者は、すべてのセキュリティ欠陥の報告とそれらの欠陥の訂正要求を受け付け、処理する手続きを確立しなければならない。**

ALC_FLR.2.3D **開発者は、TOE 利用者に対する欠陥修正ガイダンスを提供しなければならない。**

内容・提示エレメント:

ALC_FLR.2.1C 欠陥修正手続き証拠資料は、TOE のリリースごとに報告されたすべてのセキュリティ欠陥を追跡するのに使用する手続きを記述しなければならない。

ALC_FLR.2.2C 欠陥修正手続きは、欠陥の訂正の調査状況の記述とともに、各々のセキュリティ欠陥の性質と影響の記述が提供されることを要求しなければならない。

ALC_FLR.2.3C 欠陥修正手続きは、各々のセキュリティ欠陥の訂正アクションが識別されることを要求しなければならない。

ALC_FLR.2.4C 欠陥修正手続き証拠資料は、TOE 利用者に、欠陥情報、訂正、及び訂正アクションについてのガイダンスを提供するために使用する方法を記述しなければならない。

ALC_FLR.2.5C **欠陥修正手続きは、開発者が TOE 利用者からの報告及び TOE の疑わしいセキュリティ欠陥に関する問合せを受け取る手段を記述しなければならない。**

ALC_FLR.2.6C **報告されたセキュリティ欠陥を処理する手続きは、報告されたすべての欠陥が修正され、TOE 利用者に修正手続きが発行されることを保証しなければならない。**

ALC_FLR.2.7C **報告されたセキュリティ欠陥を処理する手続きは、これらのセキュリティ欠陥のいかなる訂正も、新規の欠陥を引き起こすことのないよう、保護手段を提供しなければならない。**

ALC_FLR.2.8C **欠陥修正ガイダンスは、TOE 利用者が開発者へ TOE の疑わしいセキュリティ欠陥を報告する手段を記述しなければならない。**

評価者アクションエレメント:

ALC_FLR.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない。*

ALC_FLR.3 系統的な欠陥修正

依存性: なし

目的

382 開発者が TOE 利用者からのセキュリティ欠陥報告に基づいて適切に行動することができ、かつ誰に訂正処置を送るかを知るために、TOE 利用者は、開発者にセキュリティ欠陥報告を提出する方法及び開発者がこれらの訂正処置を受け取ることができるように、開発者に対して TOE 利用者自身を登録する方法を理解する必要がある。開発者から TOE 利用者への欠陥修正ガイダンスは、TOE 利用者がこの重要な情報に気づくことを保証する。

開発者アクションエレメント:

ALC_FLR.3.1D 開発者は、TOE 開発者に対する欠陥修正手続きの証拠資料を提出しなければならない。

ALC_FLR.3.2D 開発者は、すべてのセキュリティ欠陥の報告とそれらの欠陥の訂正要求を受け付け、処理する手続きを確立しなければならない。

ALC_FLR.3.3D 開発者は、TOE 利用者に対する欠陥修正ガイダンスを提供しなければならない。

内容・提示エレメント:

ALC_FLR.3.1C 欠陥修正手続き証拠資料は、TOE のリリースごとに報告されたすべてのセキュリティ欠陥を追跡するのに使用する手続きを記述しなければならない。

ALC_FLR.3.2C 欠陥修正手続きは、欠陥の訂正の調査状況の記述とともに、各々のセキュリティ欠陥の性質と影響の記述が提供されることを要求しなければならない。

ALC_FLR.3.3C 欠陥修正手続きは、各々のセキュリティ欠陥の訂正アクションが識別されることを要求しなければならない。

ALC_FLR.3.4C 欠陥修正手続き証拠資料は、TOE 利用者に、欠陥情報、訂正、及び訂正アクションについてのガイダンスを提供するために使用する方法を記述しなければならない。

ALC_FLR.3.5C 欠陥修正手続きは、開発者が TOE 利用者からの報告及び TOE の疑わしいセキュリティ欠陥に関する問合せを受け取る手段を記述しなければならない。

ALC_FLR.3.6C **欠陥修正手続きは、セキュリティ欠陥により影響を受ける可能性がある登録された利用者に対する、タイムリな応答、セキュリティ欠陥報告及び関連する訂正の自動配付を要求する手続きを含まなければならない。**

ALC_FLR.3.7C 報告されたセキュリティ欠陥を処理する手続きは、報告されたすべての欠陥が修正され、TOE 利用者に修正手続きが発行されることを保証しなければならない。

ALC_FLR.3.8C 報告されたセキュリティ欠陥を処理する手続きは、これらのセキュリティ欠陥のいかなる訂正も、新規の欠陥を引き起こすことのないよう、保護手段を提供しなければならない。

ALC_FLR.3.9C 欠陥修正ガイダンスは、TOE 利用者が開発者へ TOE の疑わしいセキュリティ欠陥を報告する手段を記述しなければならない。

ALC_FLR.3.10C **欠陥修正ガイダンスは、TOE 利用者がセキュリティ欠陥報告及び訂正を受け取る資格を得るために開発者へ登録する手段を記述しなければならない。**

ALC_FLR.3.11C **欠陥修正ガイダンスは、TOE に関係するセキュリティ問題に関するすべての報告及び問合せのための特定の連絡先を識別しなければならない。**

評価者アクションエレメント:

ALC_FLR.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

14.6 ライフサイクル定義(ALC_LCD)

目的

383 TOE の開発及び保守の管理が貧弱な場合、そのすべての SFR を満たさない TOE になってしまう可能性がある。したがって TOE のライフサイクルにおいて、できるだけ早い時期に、TOE の開発及び保守のモデルを確立することが重要である。

384 TOE の開発及び保守のモデルを使用することは、TOE がそのすべての SFR を満たすことを保証するものではない。採用したモデルが、不十分または不適当で、TOE の品質に何の利点も生じないことが分かるのみである。専門家のグループ(例えば、学術専門家や標準化組織)で認められたライフサイクルモデルを使用することは、開発及び保守のモデルが SFR を満たす TOE に寄与する可能性を高める。ある程度の定量的評価を含んだライフサイクルモデルを使用すると、TOE 開発プロセスの全体的な品質に、さらなる保証が得られる。

コンポーネントのレベル付け

385 このファミリのコンポーネントは、ライフサイクルモデルの計測可能性、及びそのモデルに準拠するための要件の増加に基づいて、レベル付けされている。

適用上の注釈

386 ライフサイクルモデルは、TOE を開発及び保守するために使用される手順、ツール、及び技法を含んでいる。このようなモデルは、設計方法、レビュー手順、プロジェクト管理の統制手段、変更管理手続き、テスト方法、及び受入れ手続きなどをカバーしている。効果的なライフサイクルモデルは、このような開発及び保守のプロセスの側面を、責任や工程の監視を割り当てる全体の管理機構の中で取り組んでいる。

387 受入れ状況には、基準内の様々な場所で扱われる様々なタイプがある。下請け業者から配付される部分の受入れ(「統合」)は、このファミリのライフサイクル定義(ALC_LCD)で、内部転送に続く受入れは開発セキュリティ(ALC_DVS)で、CM システムへの部分の受入れは CM 能力(ALC_CMC)で、消費者による配付された TOE の受入れは配付(ALD_DEL)で扱われるべきである。最初の 3 タイプは重複する可能性がある。

388 ライフサイクルの定義は、TOE の保守も扱っており、そのため評価完了後に関係する内容もあるが、評価時に提供された TOE のライフサイクル情報の分析を通じて、それらも保証される。

389 ライフサイクルモデルが、TOE がそのセキュリティ要件を満たさないという危険を十分に最小化できるならば、このモデルは、TOE の開発及び保守で必要な管理方法を提供する。

390 測定可能なライフサイクルモデルとは、製品の開発特性を測定するために、管理された製品の何らかの定量的評価(数値パラメタ及び/または数値的尺度)を使用するライフサイクルモデルである。代表的な尺度には、ソースコードの複雑性尺度、欠陥密度(コードのサイズあたりのエラー数)、または平均故障間隔がある。セキュリティ評価の場合、それらのすべての尺度が関係を持ち、失敗の可能性を減らし、それに伴って TOE のセキュリティの保証を増加向上させることによって、品質を上げるために使用される。

391 一方には標準化されたライフサイクルモデル(ウォータフォールモデルなど)、もう一方には標準化された尺度(誤り密度など)が存在し、それらが組み合わせられる可能性があることを考慮するべきである。CC は、両方の側面を定義している一つの標準に正確に従うためのライフサイクルを要求しない。

ALC_LCD.1　開発者によるライフサイクルモデルの定義

依存性:　なし

開発者アクションエレメント:

ALC_LCD.1.1D　開発者は、**TOE** の開発及び保守で使用されるライフサイクルモデルを確立しなければならない。

ALC_LCD.1.2D　開発者は、ライフサイクル定義証拠資料を提供しなければならない。

内容・提示エレメント:

ALC_LCD.1.1C　ライフサイクル定義証拠資料は、**TOE** の開発及び保守で使用されるモデルを記述しなければならない。

ALC_LCD.1.2C　ライフサイクルモデルは、**TOE** の開発及び保守で必要な管理方法を提供しなければならない。

評価者アクションエレメント:

ALC_LCD.1.1E　評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ALC_LCD.2　測定可能なライフサイクルモデル

依存性:　なし

開発者アクションエレメント:

ALC_LCD.2.1D　開発者は、TOE の開発及び保守で使用される、**測定可能なライフサイクルモデルに基づいた**ライフサイクルモデルを確立しなければならない。

ALC_LCD.2.2D　開発者は、ライフサイクル定義証拠資料を提供しなければならない。

ALC_LCD.2.3D　開発者は、測定可能なライフサイクルモデルを使用して **TOE** の開発を測定しなければならない。

ALC_LCD.2.4D　開発者は、ライフサイクル出力証拠資料を提供しなければならない。

内容・提示エレメント:

ALC_LCD.2.1C　ライフサイクル定義証拠資料は、**TOE 及び/または TOE の開発の品質を測定するために使用された数値パラメタ及び/または数値的尺度の詳細を含む**、TOE の開発及び保守で使用されるモデルを記述しなければならない。

ALC_LCD.2.2C　ライフサイクルモデルは、TOE の開発及び保守で必要な管理方法を提供しなければならない。

ALC_LCD.2.3C　ライフサイクル出力証拠資料は、測定可能なライフサイクルモデルを使用して **TOE** の開発の測定結果を提供しなければならない。

評価者アクションエレメント:

ALC_LCD.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない。*

14.7 ツールと技法(ALC_TAT)

目的

392 ツールと技法は、TOE の開発、分析、及び実装に使用されるツールの選択に関連する。これは、TOE の開発時に不明確な、一貫性がない、もしくは不正確な開発ツールが使用されるのを防止する要件を含む。これは、プログラミング言語、証拠資料、実装標準、及びサポートするランタイムライブラリのような TOE の部分も含むが、これらに限定されない。

コンポーネントのレベル付け

393 このファミリのコンポーネントは、記述と実装標準、及び実装に依存するオプションの証拠資料の範囲に関する要件の増加に基づいて、レベル付けされている。

適用上の注釈

394 明確に定義された開発ツールが要求される。これらは、明確かつ完全に記述されたツールである。例えば、標準化組織などにより発行された標準に基づいているプログラム言語や CAD システムは、明確に定義されていると考えられる。自己製らのツールは、それらが明確に定義されているかどうかを明確化するために、さらに調査が必要である。

395 **ALC_TAT.1.2C** の要件は、ソースコードのすべてのステートメントが、曖昧でない意味を持つことを保証するために、特にプログラミング言語に適用可能である。

396 ALC_TAT.2 及び ALC_TAT.3 では、実装ガイドラインが専門家のグループ(例えば、学術専門家や標準化組織)で認められている場合に、それらが実装標準として受け入れられる。実装標準は、通常は特定の業界で公然と十分に受け入れられている一般的な実践であるが、開発者固有の実装ガイドラインも標準として受け入れられる場合があり、専門性に重点が置かれている。

397 ツールと技法は、開発者が適用する実装標準(**ALC_TAT.2.3D**)と、サードパーティのソフトウェア、ハードウェア、またはファームウェアを含む「TOE のすべての部分」についての実装標準(**ALC_TAT.3.3D**)とを区別している。CM 範囲(ALC_CMS)で導入されている構成リストでは、各 TSF 関連の構成要素の生成元が TOE 開発者なのかサードパーティの開発者なのかを示す必要がある。

ALC_TAT.1 明確に定義された開発ツール

依存性: ADV_IMP.1 TSF の実装表現

開発者アクションエレメント:

ALC_TAT.1.1D **開発者は、TOE に対して使用される各開発ツールを識別しなければならない。**

ALC_TAT.1.2D **開発者は、各開発ツールのオプションの中で実装に依存するものについて証拠資料を提出しなければならない。**

内容・提示エレメント:

ALC_TAT.1.1C **実装に使用される各開発ツールは、明確に定義されていなければならない。**

ALC_TAT.1.2C **各開発ツールの証拠資料は、実装に使用されるすべてのステートメントの意味、及び規則と指示文を曖昧さなく定義しなければならない。**

ALC_TAT.1.3C 各開発ツールの証拠資料は、実装に依存するすべてのオプションの意味を、曖昧さなく定義しなければならない。

評価者アクションエレメント:

ALC_TAT.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ALC_TAT.2 実装標準への準拠

依存性: ADV_IMP.1 TSF の実装表現

開発者アクションエレメント:

ALC_TAT.2.1D 開発者は、TOE に対して使用される各開発ツールを識別しなければならない。

ALC_TAT.2.2D 開発者は、各開発ツールのオプションの中で実装に依存するものについて証拠資料を提出しなければならない。

ALC_TAT.2.3D 開発者は、開発者が適用している実装標準を記述しなければならない。

内容・提示エレメント:

ALC_TAT.2.1C 実装に使用される各開発ツールは、明確に定義されていなければならない。

ALC_TAT.2.2C 各開発ツールの証拠資料は、実装に使用されるすべてのステートメントの意味、及び規則と指示文を曖昧さなく定義しなければならない。

ALC_TAT.2.3C 各開発ツールの証拠資料は、実装に依存するすべてのオプションの意味を、曖昧さなく定義しなければならない。

評価者アクションエレメント:

ALC_TAT.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ALC_TAT.2.2E 評価者は、実装標準が適用されていることを*確認しなければならない*。

ALC_TAT.3 実装標準への準拠 - すべての部分

依存性: ADV_IMP.1 TSF の実装表現

開発者アクションエレメント:

ALC_TAT.3.1D 開発者は、TOE に対して使用される各開発ツールを識別しなければならない。

ALC_TAT.3.2D 開発者は、各開発ツールのオプションの中で実装に依存するものについて証拠資料を提出しなければならない。

ALC_TAT.3.3D 開発者は、**TOE のすべての部分に対して**開発者**及びサードパーティプロバイダ**が適用している実装標準を記述しなければならない。

内容・提示エレメント:

ALC_TAT.3.1C 実装に使用される各開発ツールは、明確に定義されていなければならない。

ALC_TAT.3.2C 各開発ツールの証拠資料は、実装に使用されるすべてのステートメントの意味、及び規則と指示文を曖昧さなく定義しなければならない。

ALC_TAT.3.3C 各開発ツールの証拠資料は、実装に依存するすべてのオプションの意味を、曖昧さなく定義しなければならない。

評価者アクションエレメント:

ALC_TAT.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ALC_TAT.3.2E 評価者は、実装標準が適用されていることを***確認しなければならない。***

15 ATE クラス: テスト

398 「テスト」クラスは、4 つのファミリを含む。つまり、カバレージ(ATE_COV)、深さ(ATE_DPT)、独立テスト(即ち、評価者によって実行される機能テスト)(ATE_IND)、及び機能テスト(ATE_FUN)である。テストは、TSF が記述(機能仕様、TOE 設計、及び実装表現)に従ってふるまうことの保証を提供する。

399 このクラスは、TSF がその設計記述に従って動作することの確認に重点を置いている。このクラスでは、TSF の設計及び実装での脆弱性の識別を特に求める TSF の分析に基づく侵入テストを扱わない。侵入テストは、AVA: 脆弱性評定クラスで、脆弱性評定の 1 つの側面として別に述べられている。

400 ATE: テストクラスでは、テストが開発者テストと評価者テストに分けられる。カバレージ(ATE_COV)ファミリと深さ(ATE_DPT)ファミリは、開発者テストの完全性を扱う。カバレージ(ATE_COV)は機能仕様がテストされるときの厳格性を扱い、深さ(ATE_DPT)は他の設計記述(セキュリティアーキテクチャ、TOE 設計、実装表現)に対するテストが必要かどうかを扱う。

401 機能テスト(ATE_FUN)は、開発者によるこれらのテストの実行、及びこのテストについてどのように証拠資料を提出するべきかを扱う。最後に、独立テスト(ATE_IND)は、評価者テスト、つまり評価者が開発者テストの一部または全部を繰り返すべきであるか、及び評価者がどの程度の量の独立テストを実行するべきであるかを扱う。

402 図 14 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

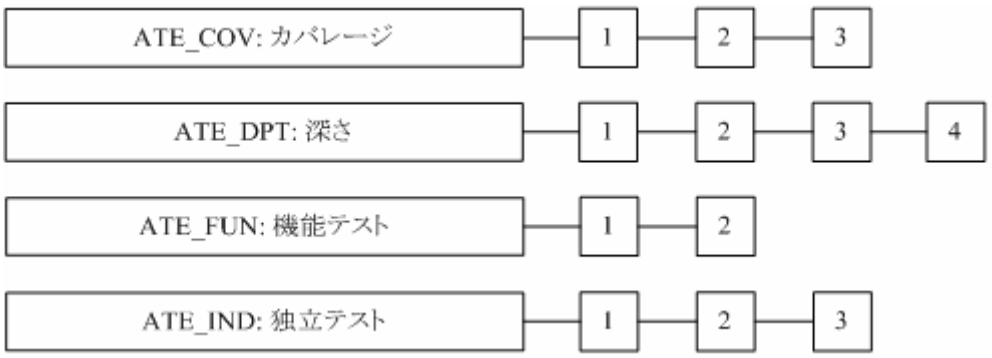

図 14 ATE: テストクラスのコンポーネント構成

15.1 カバレージ(ATE_COV)

目的

403 このファミリは TSF がその機能仕様に対してテストされていることを確認する。これは、開発者の対応証拠の検査を通して達成される。

コンポーネントのレベル付け

404 このファミリのコンポーネントは、仕様に基づいてレベル付けされている。

適用上の注釈

ATE_COV.1 カバレージの証拠

依存性: ADV_FSP.2 セキュリティ実施機能仕様
ATE_FUN.1 機能テスト

目的

405 このコンポーネントの目的は、TSFI の一部がテストされていることを確認することである。

適用上の注釈

406 このコンポーネントでは、開発者は、テスト証拠資料内のテストが機能仕様内の TSFI とどのように対応するかを示す。これは、対応のステートメント(多分、表を使うこと)により達成可能である。

開発者アクションエレメント:

ATE_COV.1.1D **開発者は、テストカバレージの証拠を提供しなければならない。**

内容・提示エレメント:

ATE_COV.1.1C **テストカバレージの証拠は、テスト証拠資料におけるテストと機能仕様における TSFI との間の対応を提示しなければならない。**

評価者アクションエレメント:

ATE_COV.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ATE_COV.2 カバレージの分析

依存性: ADV_FSP.2 セキュリティ実施機能仕様
ATE_FUN.1 機能テスト

目的

407 このコンポーネントの目的は、TSFI のすべてがテストされていることを確認することである。

適用上の注釈

408 このコンポーネントでは、開発者は、テスト証拠資料内のテストが機能仕様内のすべての TSFI と対応していることを確認する。これは、対応のステートメント(多分表を使うこと)によって達成できるが、開発者はテストカバレージの分析も提供する。

開発者アクションエレメント:

ATE_COV.2.1D 開発者は、テストカバレージの**分析**を提供しなければならない。

内容・提示エレメント:

ATE_COV.2.1C テストカバレージの**分析**は、テスト証拠資料におけるテストと機能仕様における TSFI との間の対応を**実証**しなければならない。

ATE_COV.2.2C **テストカバレージの分析は、機能仕様におけるすべての TSFI がテストされていることを実証しなければならない。**

評価者アクションエレメント:

ATE_COV.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ATE_COV.3 カバレージの厳格な分析

依存性: ADV_FSP.2 セキュリティ実施機能仕様
ATE_FUN.1 機能テスト

目的

409 このコンポーネントの目的は、機能仕様内のすべてのインタフェースについて開発者が徹底的なテストを実行したことを確認することである。

410 このコンポーネントの目的は、すべての TSFI のすべてのパラメタがテストされていることを確認することである。

適用上の注釈

411 このコンポーネントでは、開発者に対し、テスト証拠資料内のテストが機能仕様内のすべての TSFI とどのように対応するかを示すことが要求される。これは、対応のステートメント(多分表を使うこと)によって達成できるが、さらに開発者には、すべての TSFI のすべてのパラメタに対してテストが実行されることを実証することが要求される。この追加の要件には、境界テスト(つまり、指定された限界を超えたときに誤りが生成されることの検証)、及び否定テスト(例えば、利用者 A にアクセス権が与えられるときに、利用者 A がアクセスできるようになったことだけでなく、利用者 B が突然アクセスできなくなったことを検証する)が含まれる。この種のテストは厳密には徹底的ではない。なぜなら、パラメタのすべての可能な値がチェックされるようにはなっていないからである。

開発者アクションエレメント:

ATE_COV.3.1D 開発者は、テストカバレージの分析を提供しなければならない。

内容・提示エレメント:

ATE_COV.3.1C テストカバレージの分析は、テスト証拠資料におけるテストと機能仕様における TSFI との間の対応を実証しなければならない。

ATE_COV.3.2C テストカバレージの分析は、機能仕様におけるすべての TSFI が**完全に**テストされていることを実証しなければならない。

評価者アクションエレメント:

ATE_COV.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

15.2 深さ(ATE_DPT)

目的

412 このファミリのコンポーネントは、開発者による TSF テストの詳細レベルを扱う。TSF のテストは、追加の設計表現及び記述(TOE 設計、実装表現、及びセキュリティアーキテクチャ記述)から引き出される情報の深さに基づいている。

413 目的は、TOE の開発中の誤りを見逃すリスクに対抗することである。特定の内部インタフェースを使用するテストは、TSF が望ましい外部的なセキュリティのふるまいを示すことの保証だけではなく、このふるまいが内部機能性の正常な動作に起因していることの保証も提供することができる。

コンポーネントのレベル付け

414 このファミリのコンポーネントは、TOE 設計から実装表現までの TSF 表現で提供される詳細の量に基づいて、レベル付けされている。このレベルは、ADV クラスで提供された TSF 表現を反映する。

適用上の注釈

415 TOE 設計は、内部コンポーネント(例えばサブシステム)、及び場合によっては TSF のモジュールを、それらのコンポーネント及びモジュール間のインタフェースとともに記述する。この TOE 設計のテストの証拠は、内部インタフェースが使用され、記述どおりに動作することが確認されたことを示さなければならない。これは、TSF の外部インタフェースを介したテストを通じて、またはテストハーネスを使用するなどして TOE コンポーネントインタフェースを単独でテストすることによって達成される。内部インタフェースのある側面が外部インタフェースを介してテストできない場合は、それらの側面のテストが不要であること、または内部インタフェースを直接テストする必要があることが正当化されるべきである。後者の場合は、直接テストを容易にするために、TOE 設計が十分に詳細化されている必要がある。

416 TSF のアーキテクチャへの信頼の記述(セキュリティアーキテクチャ(ADV_ARC)での)で特定のメカニズムが挙げられている場合、開発者が実行するテストは、記述されているとおりにメカニズムが実行され、動作していることを示さなければならない。

417 このファミリの最上位のコンポーネントでは、TOE 設計に対してだけではなく、実装表現に対してもテストが実行される。

ATE_DPT.1 テスト: 基本設計

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_TDS.2 アーキテクチャ設計
ATE_FUN.1 機能テスト

目的

418 TSF のサブシステム記述は、TSF の内部動作に関する上位レベルの記述を提供する。TOE サブシステムのレベルでのテストは、TSF サブシステムが、TOE 設計及びセキュリティアーキテクチャ記述で記述されたとおり動作及び相互作用することへの保証を提供する。

開発者アクションエレメント:

ATE_DPT.1.1D 開発者は、テストの深さの分析を提供しなければならない。

内容・提示エレメント:

ATE_DPT.1.1C テストの深さの分析は、テスト証拠資料におけるテストと **TOE** 設計における **TSF** サブシステムの間の対応を実証しなければならない。

ATE_DPT.1.2C テストの深さの分析は、**TOE** 設計内のすべての **TSF** サブシステムがテストされていることを実証しなければならない。

評価者アクションエレメント:

ATE_DPT.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ATE_DPT.2 テスト: セキュリティ実施モジュール

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_TDS.3 基本モジュール設計
ATE_FUN.1 機能テスト

目的

419 TSF のサブシステム及びモジュール記述は、TSF の内部動作に関する上位レベルの記述、及び SFR 実施モジュールのインタフェースの記述を提供する。TOE 記述のこのレベルでのテストは、TSF サブシステム及び SFR 実施モジュールが、TOE 設計及びセキュリティアーキテクチャ記述で記述されたとおり動作及び相互作用することへの保証を提供する。

開発者アクションエレメント:

ATE_DPT.2.1D 開発者は、テストの深さの分析を提供しなければならない。

内容・提示エレメント:

ATE_DPT.2.1C テストの深さの分析は、テスト証拠資料におけるテストと TOE 設計における TSF サブシステム**及びモジュール**の間の対応を実証しなければならない。

ATE_DPT.2.2C テストの深さの分析は、TOE 設計内のすべての TSF サブシステムがテストされていることを実証しなければならない。

ATE_DPT.2.3C テストの深さの分析は、**TOE** 設計内の **SFR** 実施モジュールがテストされていることを実証しなければならない。

評価者アクションエレメント:

ATE_DPT.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ATE_DPT.3 テスト: モジュール設計

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_TDS.4 準形式的モジュール設計
ATE_FUN.1 機能テスト

目的

420 TSF のサブシステム及びモジュール記述は、TSF の内部動作に関する上位レベルの記述、及びモジュールのインタフェースの記述を提供する。TOE 記述のこのレベルでのテストは、TSF サブシステム及びモジュールが、TOE 設計及びセキュリティアーキテクチャ記述で記述されたとおり動作及び相互作用することへの保証を提供する。

開発者アクションエレメント:

ATE_DPT.3.1D 開発者は、テストの深さの分析を提供しなければならない。

内容・提示エレメント:

ATE_DPT.3.1C テストの深さの分析は、テスト証拠資料におけるテストと TOE 設計における TSF サブシステム及びモジュールの間の対応を実証しなければならない。

ATE_DPT.3.2C テストの深さの分析は、TOE 設計内のすべての TSF サブシステムがテストされていることを実証しなければならない。

ATE_DPT.3.3C テストの深さの分析は、TOE 設計内の**すべての**モジュールがテストされていることを実証しなければならない。

評価者アクションエレメント:

ATE_DPT.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ATE_DPT.4 テスト: 実装表現

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_TDS.4 準形式的モジュール設計
ADV_IMP.1 TSF の実装表現
ATE_FUN.1 機能テスト

目的

421 TSF のサブシステム及びモジュール記述は、TSF の内部動作に関する上位レベルの記述、及びモジュールのインタフェースの記述を提供する。TOE 記述のこのレベルでのテストは、TSF サブシステム及びモジュールが、TOE 設計及びセキュリティアーキテクチャ記述で記述されたとおり、及び実装表現に従って動作及び相互作用することへの保証を提供する。

開発者アクションエレメント:

ATE_DPT.4.1D 開発者は、テストの深さの分析を提供しなければならない。

内容・提示エレメント:

ATE_DPT.4.1C テストの深さの分析は、テスト証拠資料におけるテストと TOE 設計における TSF サブシステム及びモジュールの間の対応を実証しなければならない。

ATE_DPT.4.2C テストの深さの分析は、TOE 設計内のすべての TSF サブシステムがテストされていることを実証しなければならない。

ATE_DPT.4.3C テストの深さの分析は、TOE 設計内のすべてのモジュールがテストされていることを実証しなければならない。

ATE_DPT.4.4C **テストの深さの分析は、TSF がその実装表現に従って動作することを実証しなければならない。**

評価者アクションエレメント:

ATE_DPT.4.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

15.3 機能テスト(ATE_FUN)

目的

422 開発者によって実行される機能テストは、テスト証拠資料におけるテストが正しく実行されて証拠資料として提出されることの保証を提供する。TSF の設計記述に対するこれらのテストの対応は、カバレージ(ATE_COV)ファミリ及び深さ(ATE_DPT)ファミリを通じて達成される。

423 このファミリは、未発見の欠点の公算が比較的少ないという保証を提供するのに寄与する。

424 カバレージ(ATE_COV)、深さ(ATE_DPT)、機能テスト(ATE_FUN)のファミリは、開発者により提供されるべきテストの証拠を定義するのに組み合わせて使用される。評価者による独立機能テストは、独立テスト(ATE_IND)で特定される。

コンポーネントのレベル付け

425 このファミリは、2 つのコンポーネントを含み、上位は順序依存性を分析することを要求する。

適用上の注釈

426 テスト遂行の手順は、テスト環境、テスト条件、テストデータのパラメタと値を含むテストプログラムとテストスイートを使うための指示を提供することを期待されている。テスト手順は、またテスト入力からテスト結果がどのように引き出されるかを示さなければならない。

427 順序依存性は、特定のテストの実行がうまくいくかどうかが、特定の状態の存在に依存する場合に関係する。例えば、テスト A の実行の成功から生じる状態がテスト B の実行の成功に必須であるため、順序依存性は、テスト A がテスト B の直前に実行されることを要求する。このようにして、テスト B の失敗が順序依存性の問題に関係しているかも知れない。前述の例で、テスト B は、テスト A ではなくてテスト C がそれの直前に実行されたために失敗するかも知れない、またはテスト B の失敗はテスト A の失敗に関係しているかも知れない。

ATE_FUN.1 機能テスト

依存性: ATE_COV.1 カバレージの証拠

目的

428 目的は、テスト証拠資料におけるテストが正しく実行されて証拠資料として提出されることを開発者が実証することである。

開発者アクションエレメント:

ATE_FUN.1.1D **開発者は、TSF をテストし、結果を証拠資料で提出しなければならない。**

ATE_FUN.1.2D **開発者は、テスト証拠資料を提供しなければならない。**

内容・提示エレメント:

ATE_FUN.1.1C **テスト証拠資料は、テスト計画、期待されるテスト結果、及び実際のテスト結果から構成されなければならない。**

ATE_FUN.1.2C テスト計画は、実行されるべきテストを識別し、各テストを実行するシナリオを記述しなければならない。これらのシナリオは、他のテストの結果へのすべての順序依存性を含んでいなければならない。

ATE_FUN.1.3C 期待されるテスト結果は、テストの実行が成功したときの予期される出力を示さなければならない。

ATE_FUN.1.4C 実際のテスト結果は、期待されたテスト結果と一貫していなければならない。

評価者アクションエレメント:

ATE_FUN.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ATE_FUN.2 順序付けられた機能テスト

依存性: ATE_COV.1 カバレージの証拠

目的

429 目的は、開発者が、テスト証拠資料におけるテストが正しく実行され、証拠資料として提出されることを実証すること、及びテスト対象のインタフェースの正しさに関する論証の堂々巡りを回避するようにテストが構成されることを保証することである。

適用上の注釈

430 テスト手順は、テストの順序に関して必須の初期テスト条件を記述できるかもしれないが、順序の根拠が提供されていない場合がある。テスト順序の分析は、テスト順序に障害が隠れている可能性があることから、テストの妥当性を決定する重要な要因である。

開発者アクションエレメント:

ATE_FUN.2.1D 開発者は、TSF をテストし、結果を証拠資料で提出しなければならない。

ATE_FUN.2.2D 開発者は、テスト証拠資料を提供しなければならない。

内容・提示エレメント:

ATE_FUN.2.1C テスト証拠資料は、テスト計画、期待されるテスト結果、及び実際のテスト結果から構成されなければならない。

ATE_FUN.2.2C テスト計画は、実行されるべきテストを識別し、各テストを実行するシナリオを記述しなければならない。これらのシナリオは、他のテストの結果へのすべての順序依存性を含んでいなければならない。

ATE_FUN.2.3C 期待されるテスト結果は、テストの実行が成功したときの予期される出力を示さなければならない。

ATE_FUN.2.4C 実際のテスト結果は、期待されたテスト結果と一貫していなければならない。

ATE_FUN.2.5C テスト証拠資料は、テスト手順の順序依存性の分析を含まなければならない。

評価者アクションエレメント:

ATE_FUN.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

15.4 独立テスト(ATE_IND)

目的

431 このファミリの目的は、評価者が開発者テストを検証し追加のテストを実行することで、ATE_FUN、ATE_COV、及び ATE_DPT ファミリで達成される保証に基づいている。

コンポーネントのレベル付け

432 レベル付けは、開発者テスト証拠資料の量、テスト支援、及び評価者テストの量に基づいている。

適用上の注釈

433 このファミリは、TSF 独立機能テストの程度を扱う。独立機能テストは、全体、または一部において、開発者機能テストを繰り返す形式、または開発者のテストの範囲または深さを拡張する形式でも良い。これらのアクティビティは補完的であり、テスト結果の可用性とカバレージ、及び TSF の機能の複雑性を考慮して、TOE ごとに適切な組み合わせが計画されなければならない。

434 開発者テストのサンプリングは、開発者が計画した TSF に対するテスト計画を実行し、結果を正しく記録していることの確証を提供することを意図している。選択されるべきサンプルの量は、開発者による機能テスト結果の詳細さと品質によって影響される。評価者は、また追加テストを考え出す範囲と、これらの 2 つの領域での労力から得られる相対的利益を考察する必要がある。すべての開発者テストを繰り返すことは、可能であり、望ましい場合もあるが、それ以外の場合では非常に困難で生産性が低いことが認識されている。したがって、このファミリの最上位のコンポーネントは注意して使用すべきである。サンプリングは、カバレージ(ATE_COV)と深さ(ATE_DPT)の両方の要件を満たすために提供されるテスト結果を含む、利用可能なテスト結果全体から行われる。

435 評価に含まれる TOE の異なる構成を考慮することもまた必要である。評価者は、提供された結果の有効性を評価し、それに応じて自らのテストを計画する必要がある。

436 テストに対する TOE の適合は、TOE へのアクセス、テストの実行に必要な支援の証拠資料、及び情報(あらゆるテストソフトウェアまたはツールを含む)に基づく。そのような支援の必要性は別の保証ファミリへの依存によって述べられている。

437 加えて、テストに対する TOE の適合は、別の考えに基づいている。例えば、開発者から提供された TOE のバージョンが最終バージョンではない可能性がある。

438 *インタフェース*という用語は、機能仕様及び TOE 設計で記述されているインタフェースと、実装表現で識別される呼び出しを通じて渡されるパラメタを指している。使用される一連のインタフェースは、カバレージ(ATE_COV)及び深さ(ATE_DPT)コンポーネントを通じて選択される。

439 インタフェースのサブセットに対するリファレンスは、実施する評価の目的に一致する一連の適切なテストを、評価者が設計できるようにすることを意図する。

ATE_IND.1 独立テスト - 準拠

依存性: ADV_FSP.1 基本機能仕様
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き

目的

440 このコンポーネントの目的は、TOE がその設計表現とガイダンス文書に従って動作することを実証することである。

適用上の注釈

441 このコンポーネントは、開発者テスト結果の使用について述べていない。そのような結果が利用できない場合や開発者テストが確認なく承認されている場合に有効である。評価者は、機能仕様を含んでいるがそれには限定されない TOE の設計表現に従って TOE が動作することを確認する目的で、テストを考案し、実施することを要求される。近道は、すべての可能なテストを実施するよりも、代表的なテストを通じて正常動作の自信を得ることである。この目的のために計画されるテスト範囲は方法の成果であり、特定の TOE の背景と他の評価アクティビティとのバランスを考慮する必要がある。

開発者アクションエレメント:

ATE_IND.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:

ATE_IND.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:

ATE_IND.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ATE_IND.1.2E **評価者は、TSF が仕様どおりに動作することを確認するために、TSF インタフェースのサブセットを*テストしなければならない*。**

ATE_IND.2 独立テスト - サンプル

依存性: ADV_FSP.2 セキュリティ実施機能仕様
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き
ATE_COV.1 カバレージの証拠
ATE_FUN.1 機能テスト

目的

442 このコンポーネントの目的は、TOE がその設計表現とガイダンス文書に従って動作することを実証することである。評価者テストは、開発者が、機能仕様内の一部のインタフェースについて何らかのテストを実行したことを確認する。

適用上の注釈

443 意図するところは、開発者テストの効果的な再現に必要な資料を、開発者が評価者に提供することである。これには、マシンが読み取ることのできるテスト証拠資料やテストプログラムなどが含まれる。

444 このコンポーネントは、テストの計画を補うために、評価者が開発者からの利用可能なテスト結果を入手する要件を含んでいる。評価者は、得られた結果に対する確信を得るために、開発者テストのサンプルを繰り返す。確信が得られたら、評価者は開発者テストに基づいて、別の方法で TOE を使用する追加テストを実施する。正当性が確認された開発者テスト結果の基盤を使うことにより、評価者は単に開発者自身の労力や与えられた固定レベルの資源を使うことで可能となる以上に、より広い範囲の条件で、TOE が正常に動作することの確信が得られる。開発者が TOE のテストを完了しているとの確信を得ることで、評価者は、また証拠資料の調査や専門家の知識で特別に関心がある領域のテストに適切に集中するより多くの自由が得られる。

開発者アクションエレメント:

ATE_IND.2.1D 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:

ATE_IND.2.1C TOE は、テストに適していなければならない。

ATE_IND.2.2C **開発者は、TSF の開発者機能テストで使用されたものと同等の一連の資源を提供しなければならない。**

評価者アクションエレメント:

ATE_IND.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ATE_IND.2.2E **評価者は、開発者テスト結果を検証するために、テスト証拠資料内のテストのサンプルを*実行しなければならない。***

ATE_IND.2.3E 評価者は、TSF が仕様どおりに動作することを確認するために、TSF インタフェースのサブセットを***テストしなければならない。***

ATE_IND.3 独立テスト - 完全

依存性: ADV_FSP.4 完全な機能仕様
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き
ATE_COV.1 カバレージの証拠
ATE_FUN.1 機能テスト

目的

445 このコンポーネントの目的は、TOE がその設計表現とガイダンス文書に従って動作することを実証することである。評価者テストは、開発者テストをすべて繰り返すことを含む。

適用上の注釈

446 意図するところは、開発者テストの効果的な再現に必要な資料を、開発者が評価者に提供すべきことである。これには、マシンが読み取ることのできるテスト証拠資料やテストプログラムなどが含まれる。

447 このコンポーネントでは、評価者はテスト計画の一部として、開発者テストのすべてを繰り返さなければならない。前のコンポーネントと同様に、評価者はまた、開発者が行ったのとは異なる方法で、TSF を実行させることを目的とするテストを実施する。開発者テストが徹底的に行われている場合には、これを行う余地は殆ど残っていないであろう。

開発者アクションエレメント:

ATE_IND.3.1D 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:

ATE_IND.3.1C TOE は、テストに適していなければならない。

ATE_IND.3.2C 開発者は、TSF の開発者機能テストで使用されたものと同等の一連の資源を提供しなければならない。

評価者アクションエレメント:

ATE_IND.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ATE_IND.3.2E 評価者は、開発者テスト結果を検証するために、テスト証拠資料内の**すべての**テストを***実行しなければならない。***

ATE_IND.3.3E 評価者は、TSF **全体**が仕様どおりに動作することを確認するために、TSF を***テストしなければならない。***

16 AVA クラス: 脆弱性評定

448 AVA: 脆弱性評定クラスは、TOE の開発または運用で生じる悪用可能な脆弱性の可能性を扱う。

449 図 15 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

図 15 AVA: 脆弱性評定クラスのコンポーネント構成

適用上の注釈

450 一般的に、脆弱性評定アクティビティは、TOE の開発及び運用における様々な脆弱性をカバーする。開発上の脆弱性は、TOE の開発中に入り込むいくつか特性(TSF の改ざん、直接攻撃または監視により TSF 自己保護が破られる、TSF の監視または直接攻撃により TSF ドメイン分離が破られる、あるいは TSF の回避(バイパス)により非バイパス性が破られるなど)を悪用する。運用上の脆弱性は、TOE SFR を侵害する非技術的な対抗策(誤使用や不正な構成など)における弱点を悪用する。誤使用は、セキュアでないにもかかわらず TOE の管理者または利用者が合理的にセキュアであると判断した方法で、TOE が構成または使用され得るかどうかを調査する。

451 開発上の脆弱性の評定は、保証ファミリ AVA_VAN によってカバーされる。基本的に、すべての開発上の脆弱性は AVA_VAN の範囲で考慮することができる。これは、このファミリがある種の攻撃シナリオに特定されない幅広い評定方法を適用できるという事実に基づいている。これらの不特定の評定方法には、隠れチャネルが考慮される TSF(チャネル容量の見積もりは、実際のテスト測定と同様、非形式的な工学的な測定によってなされる)、あるいは直接攻撃の形式で十分な資源を利用することで打開することができる TSF(これらの TSF の基になっている技術的な概念は、確率的または順列的メカニズムに基づいている。つまり、それらを破るために必要なセキュリティ上のふるまいや労力の評価付けは、定量的または統計的分析結果を用いて行うことができる)に対応するような評定方法などが含まれる。

452 TOE の 1 人の利用者が TOE の別の利用者に関連付けられているアクティビティを観察するのを防止する、または情報フローを使用して不正なデータ信号を得ることができないことを保証するというセキュリティ対策方針が ST で特定されている場合、脆弱性分析の実施中に隠れチャネル分析を考慮するべきである。これは、観察不能性(FPR_UNO)及び情報フロー方針(アクセス制御方針(FDP_ACC)要件のマルチレベルアクセス制御方針を通じて表される)を ST に含めることで反映されることが多い。

16.1 脆弱性分析(AVA_VAN)

目的

453 脆弱性分析とは、TOE の開発及び予期される運用の評価を通して、または他の方法(例えば、欠陥仮説法や、基礎となるセキュリティメカニズムのセキュリティのふるまいを定量的または統計的に分析する方法)によって識別された潜在的脆弱性が、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。

454 脆弱性分析は、データと機能性に許可されないアクセスを許す、TSF を妨害または変更できることを許す、または他の利用者の許可された能力を妨害することができる、といった欠陥を攻撃者が見つけられるような脅威を扱う。

コンポーネントのレベル付け

455 レベル付けは、評価者による脆弱性分析の厳格さ、及び攻撃者が潜在的脆弱性を見つけて悪用するために必要とする攻撃能力のレベルに基づいている。

AVA_VAN.1 脆弱性調査

依存性: ADV_FSP.1 基本機能仕様
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き

目的

456 公知の情報の脆弱性調査は、攻撃者が容易に発見す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。

457 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。

開発者アクションエレメント:

AVA_VAN.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_VAN.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_VAN.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

AVA_VAN.1.2E **評価者は、TOE の潜在的脆弱性を識別するために、公知の情報源の探索を*実行しなければならない*。**

AVA_VAN.1.3E **評価者は、基本的な攻撃能力を持つ攻撃者からの攻撃に TOE が耐えられることを決定するために、識別された潜在的脆弱性に基づいて侵入テストを*実施しなければならない*。**

AVA_VAN.2 脆弱性分析

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_FSP.1 基本機能仕様
ADV_TDS.1 基本設計
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き

目的

458 脆弱性分析は、評価者が潜在的脆弱性の存在を確認するために実行する。

459 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。

開発者アクションエレメント:

AVA_VAN.2.1D 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_VAN.2.1C 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_VAN.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

AVA_VAN.2.2E 評価者は、TOE の潜在的脆弱性を識別するために、公知の情報源の探索を***実行しなければならない。***

AVA_VAN.2.3E **評価者は、TOE の潜在的脆弱性を識別するために、ガイダンス証拠資料、機能仕様、TOE 設計、及びセキュリティアーキテクチャ記述を使用して、TOE の独立脆弱性分析を*実行しなければならない。***

AVA_VAN.2.4E 評価者は、基本的な攻撃能力を持つ攻撃者からの攻撃に TOE が耐えられることを決定するために、識別された潜在的脆弱性に基づいて侵入テストを***実施しなければならない。***

AVA_VAN.3 焦点を置いた脆弱性分析

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_FSP.2 セキュリティ実施機能仕様
ADV_TDS.3 基本モジュール設計
ADV_IMP.1 TSF の実装表現
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き

目的

460 脆弱性分析は、評価者が潜在的脆弱性の存在を確認するために実行する。

461 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、拡張された基本的な攻撃能力を想定して実行する。

開発者アクションエレメント:

AVA_VAN.3.1D 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_VAN.3.1C 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_VAN.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

AVA_VAN.3.2E 評価者は、TOE の潜在的脆弱性を識別するために、公知の情報源の探索を***実行しなければならない。***

AVA_VAN.3.3E 評価者は、TOE の潜在的脆弱性を識別するために、ガイダンス証拠資料、機能仕様、TOE 設計、セキュリティアーキテクチャ記述、**及び実装表現**を使用して、TOE の独立脆弱性分析を***実行しなければならない。***

AVA_VAN.3.4E 評価者は、**拡張された基本的な**攻撃能力を持つ攻撃者からの攻撃に TOE が耐えられることを決定するために、識別された潜在的脆弱性に基づいて侵入テストを***実施しなければならない。***

AVA_VAN.4 系統的脆弱性分析

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_FSP.2 セキュリティ実施機能仕様
ADV_TDS.3 基本モジュール設計
ADV_IMP.1 TSF の実装表現
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き

目的

462 系統的脆弱性分析は、評価者が潜在的脆弱性の存在を確認するために実行する。

463 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、中程度の攻撃能力を想定して実行する。

開発者アクションエレメント:

AVA_VAN.4.1D 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_VAN.4.1C 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_VAN.4.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

AVA_VAN.4.2E 評価者は、TOE の潜在的脆弱性を識別するために、公知の情報源の探索を***実行しなければならない。***

AVA_VAN.4.3E 評価者は、TOE の潜在的脆弱性を識別するために、ガイダンス証拠資料、機能仕様、TOE 設計、セキュリティアーキテクチャ記述、及び実装表現を使用して、TOE の独立した、**系統的**脆弱性分析を***実行しなければならない。***

AVA_VAN.4.4E 評価者は、**中程度の**攻撃能力を持つ攻撃者からの攻撃に TOE が耐えられることを決定するために、識別された潜在的脆弱性に基づいて侵入テストを***実施しなければならない。***

AVA_VAN.5 高度な系統的脆弱性分析

依存性: ADV_ARC.1 セキュリティアーキテクチャ記述
ADV_FSP.2 セキュリティ実施機能仕様
ADV_TDS.3 基本モジュール設計
ADV_IMP.1 TSF の実装表現
AGD_OPE.1 利用者操作ガイダンス
AGD_PRE.1 準備手続き

目的

464 系統的脆弱性分析は、評価者が潜在的脆弱性の存在を確認するために実行する。

465 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、高い攻撃能力を想定して実行する。

開発者アクションエレメント:

AVA_VAN.5.1D 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_VAN.5.1C 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_VAN.5.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

AVA_VAN.5.2E 評価者は、TOE の潜在的脆弱性を識別するために、公知の情報源の探索を***実行しなければならない。***

AVA_VAN.5.3E 評価者は、TOE の潜在的脆弱性を識別するために、ガイダンス証拠資料、機能仕様、TOE 設計、セキュリティアーキテクチャ記述、及び実装表現を使用して、TOE の独立した、系統的脆弱性分析を***実行しなければならない。***

AVA_VAN.5.4E 評価者は、**高い**攻撃能力を持つ攻撃者からの攻撃に TOE が耐えられることを決定するために、識別された潜在的脆弱性に基づいて侵入テストを***実施しなければならない。***

17 ACO クラス: 統合

466 ACO: 統合クラスは、5 つのファミリを含む。これらのファミリでは、統合 TOE が、すでに評価されたソフトウェア、ファームウェア、またはハードウェアコンポーネントが提供するセキュリティ機能性に依存する場合にセキュアに動作するという信頼を提供するために策定された保証要件を特定する。

467 統合では、CC セキュリティ保証要件パッケージに対して正常に評価された複数の IT エンティティ(基本コンポーネント及び依存コンポーネント、附属書 B を参照)を、そのいずれのエンティティもそれ以上開発することなく、結合して使用できるようにする。追加の IT エンティティ(以前コンポーネント評価の対象になっていなかったエンティティ)の開発は含まれない。統合 TOE は、環境に関する対策方針を満たす特定の環境事例すべてに設置及び統合できる新しい製品を形成する。

468 このアプローチは、コンポーネントを評価するための代替アプローチとはならない。ACO 下での統合は、統合 TOE のインテグレータに 1 つの方法を提供する。その方法は、統合 TSF を再評価することなく、評価が完了した複数のコンポーネントの組み合わせである TOE で信頼を得るために、CC で特定された他の保証レベルの代替として使用できる。(統合 TOE のインテグレータは、そのように分類された基本コンポーネントまたは依存コンポーネントの開発者に関連して、ACO クラス全体で「開発者」と呼ばれる)。

469 9 章及び 7.3 で定義されている統合保証パッケージは、統合 TOE の保証尺度である。EAL に対して評価されたコンポーネントを組み合わせ、結果として EAL の保証を得るためには、EAL 内のすべての SAR を統合 TOE に適用する必要があるため、EAL に加えてこの保証尺度が必要である。再使用はコンポーネント TOE の評価結果で構成できるが、附属書 B.3 に記述されているように、統合 TOE ではコンポーネントの追加側面を考慮しなければならないことがよくある。統合 TOE の評価アクティビティには様々な当事者が関わるため、通常は適切な EAL を適用するために、コンポーネントのこれらの追加側面に関して必要なすべての証拠を得ることは不可能である。このため、評価されたコンポーネントを組み合わせ、有意な結果を得るための問題に対処するために CAP が定義されている。これについては、附属書 B で詳しく説明する。

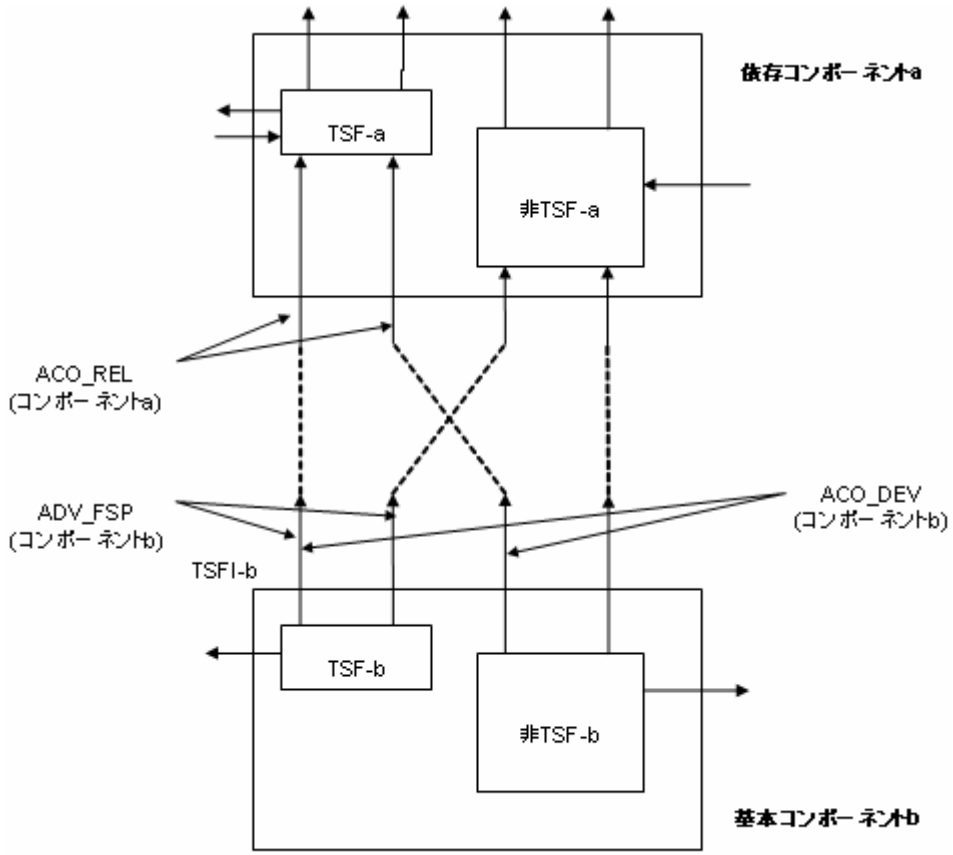

図 16　ACO ファミリ間の関係とコンポーネント間の相互作用

470 統合 TOE では、通常 1 つのコンポーネントは別のコンポーネントが提供したサービスに依存する。サービスを要求するコンポーネントは依存コンポーネントと呼ばれ、サービスを提供するコンポーネントは基本コンポーネントと呼ばれる。この相互作用と区別については、附属書 B で詳しく説明する。これは、依存コンポーネントの開発者が統合 TOE 評価を何らかの方法(開発者、スポンサーとして、または単に協調して依存コンポーネント評価から必要な評価証拠を提供する)でサポートしている場合を想定している。CAP 保証パッケージに含まれる ACO コンポーネントは、コンポーネント TOE 評価の要件追加として使用すべきではない。要件追加として使用した場合、コンポーネントに対する意味のある保証は提供されない。

471 ACO クラス内のファミリは、コンポーネント TOE 評価内の ADV クラス、ATE クラス及び AVA クラスと同様の方法で相互作用するため、これらのクラスの要件の仕様を適宜利用する。ただし、統合 TOE 評価に固有な要素がいくつか存在する。コンポーネント同士がどのように相互作用するかを判別し、コンポーネントの評価からのすべての不足を識別するために、依存コンポーネントの下層の基本コンポーネントに対する依存性が識別される(ACO_REL)。この基本コンポーネントへの依存は、依存コンポーネント SFR の支援をするサービスに対して依存コンポーネントがコールを行う際に使用されるインタフェースの観点から特定される。それらのサービス要求に応答して基本コンポーネントが提供するインタフェース、及び上位レベルにおける支援のふるまいは、ACO_DEV で分析される。ACO_DEV ファミリは ADV_TDS ファミリに基づいており、各コンポーネントの最も単純なレベルにおける TSF は統合 TOE のサブシステムとみなすことができ、各コンポーネントの追加の部分は追加のサブシステムとみなされる。したがって、コンポーネント間のインタフェースは、統合 TOE 評価内のサブシステム間の相互作用とみなされる。

472 ACO_DEV に対して提供されるインタフェース、及び支援のふるまいの記述は不完全である可能性が存在する。これは、ACO_COR の実施中に決定される。ACO_COR ファミリは、ACO_REL 及び ACO_DEV の出力を取得して、コンポーネントが評価構成で使用されているかどうかを判断し、仕様が不完全な個所を識別する。これは、統合 TOE のテスト(ACO_CTT)及び脆弱性分析(ACO_VUL)アクティビティへの入力として識別される。

473 統合 TOE のテストは、統合 TOE が、統合 TOE SFR によって決定された期待されるふるまいを示していることを判断するために実行され、より上位レベルでは、統合 TOE のコンポーネント間のインタフェースの互換性を実証する。

474 統合 TOE の脆弱性分析は、コンポーネント評価の脆弱性分析の出力を利用する。統合 TOE の脆弱性分析は、コンポーネント評価のすべての残存脆弱性を考慮して、残存脆弱性が統合 TOE に適用可能でないことを決定する。コンポーネントに関連する公知の情報の探索もまた、それぞれの評価の完了以降にコンポーネントで報告されたすべての問題を識別するために実行される。

475 ACO ファミリ間の相互作用を図 17 に示す。この図では、1 つのファミリ内で得られる証拠と理解が次のアクティビティに提供される場所は実線の矢印で示され、破線の矢印は、上記のとおり、アクティビティが明示的に統合 TOE SFR にまでさかのぼるところを識別している。

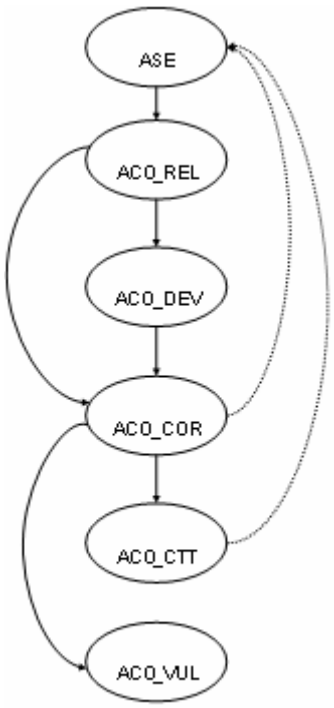

図 17 ACO ファミリ間の関係

476 統合 TOE における定義と相互作用の詳細は、附属書 B に記載されている。

477 図 18 は、このクラスのファミリと、各ファミリのコンポーネントの階層を示す。

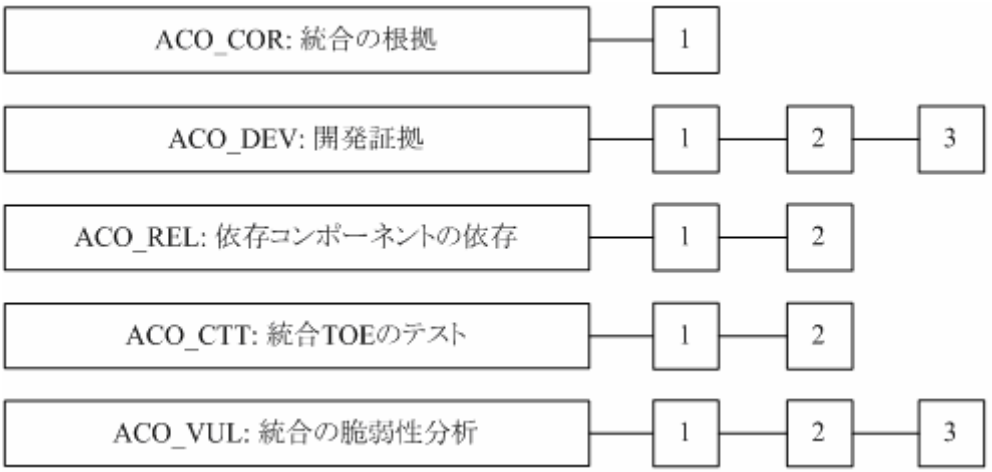

図 18 ACO: 統合クラスのコンポーネント構成

17.1 統合の根拠(ACO_COR)

目的

478 このファミリは、統合で使用する適切なレベルの保証を基本コンポーネントが提供できることを実証するための要件を扱う。

コンポーネントのレベル付け

479 このファミリに含まれるコンポーネントは 1 つのみである。

ACO_COR.1 統合の根拠

依存性: ACO_DEV.1 機能記述
ALC_CMC.1 TOE のラベル付け
ACO_REL.1 基本依存情報

開発者アクションエレメント:

ACO_COR.1.1D 開発者は、基本コンポーネントの統合の根拠を提供しなければならない。

内容・提示エレメント:

ACO_COR.1.1C 統合の根拠は、基本コンポーネントが、依存コンポーネントの TSF を支援する要求に従い構成された場合、依存コンポーネントのものと少なくとも同じ保証のレベルが、基本コンポーネントの支援機能性に対して得られることを実証しなければならない。

評価者アクションエレメント:

ACO_COR.1.1E 評価者は、情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

17.2 開発証拠(ACO_DEV)

目的

480 このファミリは、基本コンポーネントの仕様に対する要件を、詳細レベルを上げながら設定する。このような情報は、(依存情報で識別されている)依存コンポーネントの要件を支援するために適切なセキュリティ機能性が提供されているという確信を得るために必要である。

コンポーネントのレベル付け

481 コンポーネントは、提供されているインタフェースに関する詳細の量、及びそれらの実装状況に基づいて、レベル付けされている。

適用上の注釈

482 基本コンポーネントの TSF は、その統合での可能な適用における依存性に関する知識なしに定義されることが少なくない。この基本コンポーネントの TSF は、基本コンポーネント SFR を実施するために依存しなければならない基本コンポーネントのすべての部分を含むように定義される。これには、基本コンポーネントの SFR の実装に必要な基本コンポーネントのすべての部分が含まれる。

483 この基本コンポーネントの機能仕様は、TSF の操作の呼び出しを外部エンティティに許可するために基本コンポーネントが提供するインタフェースの観点から TSFI を記述する。これには、SFR を呼び出す TSF の操作との相互作用を可能にする人間の利用者とのインタフェース、及び外部 IT エンティティに TSF へのコールを許可するインタフェースが含まれる。

484 機能仕様は、TSF がそのインタフェースで何を提供するか、及び TSF の機能性が呼び出される手段についての記述のみを提供する。したがって、機能仕様は、必ずしも外部エンティティと基本コンポーネントの間で利用可能なすべてのインタフェースの完全なインタフェース仕様を提供するわけではない。TSF が運用環境に何を期待/要求するかは含まれない。依存コンポーネント TSF が基本コンポーネントに依存する内容の記述は、依存コンポーネントの依存(ACO_REL)で考慮され、開発情報の証拠は、特定されているインタフェースへの応答を提供する。

485 開発情報の証拠には、基本コンポーネントの仕様が含まれる。これは、基本コンポーネントの評価中に、ADV 要件を満たすために使用される証拠の場合もあれば、基本コンポーネントの開発者または統合 TOE の開発者によって生成される別の形式の証拠の場合もある。基本コンポーネントのこの仕様は、開発証拠(ACO_DEV)の中で、依存コンポーネントの要件を支援するために適切なセキュリティ機能性が提供されているという確信を得るために使用される。この証拠に要求される詳細レベルは、統合 TOE で要求される保証のレベルを反映して上昇する。これは、コンポーネントに対する保証パッケージの適用によって得られる確信の増加を広く反映することが期待される。評価者は、基本コンポーネントのこの記述が、依存コンポーネントに提供される依存情報と一致していることを決定する。

ACO_DEV.1 機能記述

依存性: ACO_REL.1 基本依存情報

目的

486 依存コンポーネントが依存する基本コンポーネントのインタフェースの記述が要求される。この記述は、依存情報での依存コンポーネントが依存するインタフェースの記述と一貫しているかどうかを決定するために検査される。

開発者アクションエレメント:

ACO_DEV.1.1D **開発者は、基本コンポーネントの開発情報を提供しなければならない。**

内容・提示エレメント:

ACO_DEV.1.1C **開発情報は、統合 TOE で使用される基本コンポーネントの各インタフェースの目的を記述しなければならない。**

ACO_DEV.1.2C **開発情報は、依存コンポーネントの TSF を支援するために、基本コンポーネントと依存コンポーネントの、統合 TOE で使用されるインタフェース間の対応を示さなければならない。**

評価者アクションエレメント:

ACO_DEV.1.1E **評価者は、情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ACO_DEV.1.2E **評価者は、提供されたインタフェース記述が、依存コンポーネントに提供される依存情報と一貫していることを*決定しなければならない*。**

ACO_DEV.2 設計の基本証拠

依存性: ACO_REL.1 基本依存情報

目的

487 依存コンポーネントが依存する基本コンポーネントのインタフェースの記述が要求される。この記述は、依存情報での依存コンポーネントが依存するインタフェースの記述と一貫しているかどうかを決定するために検査される。

488 さらに、依存コンポーネントの TSF を支援する基本コンポーネントのセキュリティのふるまいが記述される。

開発者アクションエレメント:

ACO_DEV.2.1D 開発者は、基本コンポーネントの開発情報を提供しなければならない。

内容・提示エレメント:

ACO_DEV.2.1C 開発情報は、統合 TOE で使用される基本コンポーネントの各インタフェースの目的及び**使用方法**を記述しなければならない。

ACO_DEV.2.2C **開発情報は、依存コンポーネントの SFR の実施を支援する、基本コンポーネントのふるまいの上位レベルの記述を提供しなければならない。**

ACO_DEV.2.3C 開発情報は、依存コンポーネントの TSF を支援するために、基本コンポーネントと依存コンポーネントの、統合 TOE で使用されるインタフェース間の対応を示さなければならない。

評価者アクションエレメント:

ACO_DEV.2.1E 評価者は、情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ACO_DEV.2.2E 評価者は、提供されたインタフェース記述が、依存コンポーネントに提供される依存情報と一貫していることを***決定しなければならない。***

ACO_DEV.3 設計の詳細証拠

依存性: ACO_REL.2 依存情報

目的

489 依存コンポーネントが依存する基本コンポーネントのインタフェースの記述が要求される。この記述は、依存情報での依存コンポーネントが依存するインタフェースの記述と一貫しているかどうかを決定するために検査される。

490 基本コンポーネントのアーキテクチャのインタフェース記述は、そのインタフェースが基本コンポーネントの TSF の一部であるかどうかを評価者が決定できるようにするために提供される。

開発者アクションエレメント:

ACO_DEV.3.1D 開発者は、基本コンポーネントの開発情報を提供しなければならない。

内容・提示エレメント:

ACO_DEV.3.1C 開発情報は、統合 TOE で使用される基本コンポーネントの各インタフェースの目的及び使用方法を記述しなければならない。

ACO_DEV.3.2C **開発情報は、統合 TOE で使用される基本コンポーネントのインタフェースを提供する基本コンポーネントのサブシステムを識別しなければならない。**

ACO_DEV.3.3C 開発情報は、依存コンポーネントの SFR の実施を**支援する**、基本コンポーネントの**サブシステム**のふるまいの上位レベルの記述を提供しなければならない。

ACO_DEV.3.4C **開発情報は、インタフェースから基本コンポーネントのサブシステムへのマッピングを提供しなければならない。**

ACO_DEV.3.5C 開発情報は、依存コンポーネントの TSF を支援するために、基本コンポーネントと依存コンポーネントの、統合 TOE で使用されるインタフェース間の対応を示さなければならない。

評価者アクションエレメント:

ACO_DEV.3.1E 評価者は、情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ACO_DEV.3.2E 評価者は、提供されたインタフェース記述が、依存コンポーネントに提供される依存情報と一貫していることを***決定しなければならない。***

17.3 依存コンポーネントの依存(ACO_REL)

目的

491 このファミリの目的は、基本コンポーネントに対する依存コンポーネントの依存を記述する証拠を提供することである。この情報は、コンポーネントを他の評価された IT コンポーネントに統合して統合 TOE を形成する担当者、及び結果としての統合のセキュリティ特性に関する洞察を提供する担当者にとって役立つ。

492 このファミリは、インタフェースが個々のコンポーネント TOE の TSFI でなかったために、個々のコンポーネントの評価中に分析されなかった可能性のある、統合 TOE の依存コンポーネントと基本コンポーネント間のインタフェースの記述を提供する。

コンポーネントのレベル付け

493 このファミリのコンポーネントは、基本コンポーネントに対する依存コンポーネントの依存の記述に示されている詳細の量に従ってレベル付けされている。

適用上の注釈

494 依存コンポーネントの依存(ACO_REL)ファミリでは、依存コンポーネントがそのセキュリティ機能性の操作を支援するために基本コンポーネントのサービスを信頼している状況での、コンポーネント間の相互作用が考慮される。基本コンポーネントのサービスはコンポーネント評価の中でセキュリティに関連性があるとみなされていなかったため、基本コンポーネントのそれらサービスに対するインタフェースは基本コンポーネント評価の中で考慮されていなかった可能性がある。これは、サービス特有の目的(例えば、タイプフォントの調整)が原因であるか、または関連する CC SFR が基本コンポーネントの ST で要求されていない(例えば、FIA:識別認証 SFR が主張されていない場合のログインインタフェース)ことが原因である。基本コンポーネントに対するこれらのインタフェースは、基本コンポーネントの評価で機能インタフェースとみなされることが多く、機能仕様で考慮されるセキュリティインタフェース(TSFI)に追加されている。

495 要約すると、機能仕様で記述されている TSFI には、外部エンティティが TSF に対して行うコールと、それらのコールに対する応答のみが含まれる。TSF によって行われるコールは、コンポーネントの評価では明示的に考慮されず、依存コンポーネントの依存(ACO_REL)を満たすために提供される依存情報で記述される。

ACO_REL.1 基本依存情報

依存性: なし

開発者アクションエレメント:

ACO_REL.1.1D **開発者は、依存コンポーネントの依存情報を提供しなければならない。**

内容・提示エレメント:

ACO_REL.1.1C **依存情報は、依存コンポーネント TSF が依存する基本コンポーネントハードウェア、ファームウェア及び/またはソフトウェアの機能性を記述しなければならない。**

ACO_REL.1.2C **依存情報は、依存コンポーネント TSF が基本コンポーネントからサービスを要求するために使用するすべての相互作用を記述しなければならない。**

ACO_REL.1.3C **依存情報は、依存 TSF が、基本コンポーネントによる干渉及び改ざんから自分自身をどのように保護するかを記述しなければならない。**

評価者アクションエレメント:

ACO_REL.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ACO_REL.2 依存情報

依存性: なし

開発者アクションエレメント:

ACO_REL.2.1D 開発者は、依存コンポーネントの依存情報を提供しなければならない。

内容・提示エレメント:

ACO_REL.2.1C 依存情報は、依存コンポーネント TSF が依存する基本コンポーネントハードウェア、ファームウェア及び/またはソフトウェアの機能性を記述しなければならない。

ACO_REL.2.2C 依存情報は、依存コンポーネント TSF が基本コンポーネントからサービスを要求するために使用するすべての相互作用を記述しなければならない。

ACO_REL.2.3C **依存情報は、使用されるインタフェース及びそれらのインタフェースからの戻り値の観点から各相互作用を記述しなければならない。**

ACO_REL.2.4C 依存情報は、依存 TSF が、基本コンポーネントによる干渉及び改ざんから自分自身をどのように保護するかを記述しなければならない。

評価者アクションエレメント:

ACO_REL.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない*。**

17.4 統合 TOE のテスト(ACO_CTT)

目的

496 このファミリは、統合 TOE で使用されるように、統合 TOE のテスト及び基本コンポーネントのテストが実行されることを要求する。

コンポーネントのレベル付け

497 このファミリのコンポーネントは、インタフェーステストの厳格さ、及び統合 TSF が依存情報及び統合 TOE SFR に従って動作することを実証するテストの十分性の分析の厳格さに基づいて、レベル付けされている。

適用上の注釈

498 このファミリに関連するテストには、次の 2 つの異なる側面が存在する:

- 基本コンポーネント及び依存コンポーネント間の、セキュリティ機能性の実施のために依存コンポーネントが依存するインタフェースの互換性を実証するためのテスト;
- 統合 TOE の SFR に従って TOE が動作することを実証するための統合 TOE のテスト。

499 依存コンポーネントの評価中に使用されるテスト構成に「プラットフォーム」としての基本コンポーネントの使用が含まれ、TSF が SFR に従って動作することをテストの分析が十分に実証する場合、開発者は統合 TOE の機能性のテストをさらに実行する必要はない。ただし、基本コンポーネントが依存コンポーネントのテストで使用されなかった場合、またはどちらかのコンポーネントの構成が変更された場合、開発者は、統合 TOE のテストを実行する。これが、SFR に従って統合 TOE の TSF が動作していることを十分に実証していれば、依存コンポーネントの依存コンポーネント開発者テストを繰り返す形式でもよい。

500 開発者は、統合で使用される基本コンポーネントインタフェースをテストすることの証拠を提供しなければならない。基本コンポーネントの TSFI の操作は、基本コンポーネントの評価中に ATE: テストのアクティビティの一部としてテストされている可能性がある。このため、依存コンポーネントが必要とするすべてのセキュリティ機能性が TSF に含まれている状態で、適切なインタフェースが基本コンポーネント評価のテストサンプル内に含まれ、基本コンポーネントが評価構成に従って動作していることが、統合の根拠(ACO_COR)で決定された場合は、基本コンポーネント ATE: テスト判定の再利用によって評価者アクション **ACO_CTT.1.1E** を満たすことができる。

501 これに該当しない場合は、統合に関連して使用され、評価構成に対する変更及び追加のセキュリティ機能によって影響を受ける基本コンポーネントインタフェースが、期待されたふるまいを示すことを保証するためにテストされる。テストの対象である期待されるふるまいは、依存情報(依存コンポーネントの依存(ACO_REL)の証拠)で記述される。

ACO_CTT.1 インタフェーステスト

依存性: ACO_REL.1 基本依存情報
ACO_DEV.1 機能記述

目的

502 このコンポーネントの目的は、依存コンポーネントが依存する基本コンポーネントの各インタフェースがテストされることを保証することである。

開発者アクションエレメント:

ACO_CTT.1.1D 開発者は、統合 **TOE** のテスト証拠資料を提供しなければならない。

ACO_CTT.1.2D 開発者は、基本コンポーネントのインタフェーステスト証拠資料を提供しなければならない。

ACO_CTT.1.3D 開発者は、テストのために統合 **TOE** を提供しなければならない。

ACO_CTT.1.4D 開発者は、基本コンポーネントの基本コンポーネント開発者機能テストで使用されたものと同等の一連の資源を提供しなければならない。

内容・提示エレメント:

ACO_CTT.1.1C 統合 **TOE** 及び基本コンポーネントインタフェーステスト証拠資料は、テスト計画、期待されるテスト結果、及び実際のテスト結果から構成されなければならない。

ACO_CTT.1.2C 開発者が実行した統合 **TOE** のテストによるテスト証拠資料は、**TSF** が仕様どおりに動作することを実証しなければならない。

ACO_CTT.1.3C 開発者が実行した基本コンポーネントインタフェーステストのテスト証拠資料は、依存コンポーネントが依存する基本コンポーネントインタフェースが仕様どおりに動作することを実証しなければならない。

ACO_CTT.1.4C 基本コンポーネントは、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:

ACO_CTT.1.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。

ACO_CTT.1.2E 評価者は、開発者テスト結果を検証するために、テスト証拠資料内のテストのサンプルを*実行しなければならない*。

ACO_CTT.1.3E 評価者は、統合 **TSF** が仕様どおりに動作することを確認するために、統合 **TOE** の **TSF** インタフェースのサブセットを*テストしなければならない*。

ACO_CTT.2 厳格なインタフェーステスト

依存性: ACO_REL.2 依存情報
ACO_DEV.2 設計の基本証拠

目的

503 このコンポーネントの目的は、依存コンポーネントが依存する基本コンポーネントの各インタフェースがテストされることを保証することである。

開発者アクションエレメント:

ACO_CTT.2.1D 開発者は、統合 TOE のテスト証拠資料を提供しなければならない。

ACO_CTT.2.2D 開発者は、基本コンポーネントのインタフェーステスト証拠資料を提供しなければならない。

ACO_CTT.2.3D 開発者は、テストのために統合 TOE を提供しなければならない。

ACO_CTT.2.4D 開発者は、基本コンポーネントの基本コンポーネント開発者機能テストで使用されたものと同等の一連の資源を提供しなければならない。

内容・提示エレメント:

ACO_CTT.2.1C 統合 TOE 及び基本コンポーネントインタフェーステスト証拠資料は、テスト計画、期待されるテスト結果、及び実際のテスト結果から構成されなければならない。

ACO_CTT.2.2C 開発者が実行した統合 TOE のテストによるテスト証拠資料は、TSF が仕様どおりに動作し、**完全である**ことを実証しなければならない。

ACO_CTT.2.3C 開発者が実行した基本コンポーネントインタフェーステストのテスト証拠資料は、依存コンポーネントが依存する基本コンポーネントインタフェースが仕様どおりに動作し、**完全である**ことを実証しなければならない。

ACO_CTT.2.4C 基本コンポーネントは、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:

ACO_CTT.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ACO_CTT.2.2E 評価者は、開発者テスト結果を検証するために、テスト証拠資料内のテストのサンプルを***実行しなければならない。***

ACO_CTT.2.3E 評価者は、統合 TSF が仕様どおりに動作することを確認するために、統合 TOE の TSF インタフェースのサブセットを***テストしなければならない。***

17.5 統合の脆弱性分析(ACO_VUL)

目的

504 このファミリは、公知に利用できる脆弱性情報、及び統合の結果として生じる可能性がある脆弱性の分析を要求する。

コンポーネントのレベル付け

505 このファミリのコンポーネントは、公知の脆弱性情報及び独立脆弱性分析の検査の厳格さに基づいて、レベル付けされている。

適用上の注釈

506 開発者は、コンポーネントの評価で報告された残存脆弱性の詳細を提供する。これらの詳細は、コンポーネント開発者またはコンポーネントの評価報告書から得ることができる。これらは、運用環境における統合 TOE の評価者による脆弱性分析への入力として使用される。

507 統合 TOE の運用環境は、コンポーネント運用環境の(各コンポーネント ST で特定された)前提条件及び目的が統合 TOE で満たされることを保証するために検査される。コンポーネント及び統合 TOE ST 間の前提条件と目的の一貫性の初期分析は、統合 TOE の ASE アクティビティの実施中に実行される。ただし、この分析は、例えば、依存コンポーネント ST 内の環境で扱われた依存コンポーネントの前提条件が、統合の結果として再度生じない(つまり、基本コンポーネントが、統合 TOE 内の依存コンポーネント ST の前提条件を適切に扱っている)ことを保証するために、ACO_REL、ACO_DEV 及び ACO_COR アクティビティ中に得た知識に基づいて再び使用される。

508 評価者による各コンポーネントの問題の探索は、コンポーネントの評価の完了以降に公知に報告された潜在的脆弱性を識別する。その後、潜在的脆弱性はテストのサブジェクトとなる。

509 統合 TOE で使用される基本コンポーネントが認証以後の保証継続性アクティビティの対象となった場合、評価者は、統合 TOE 脆弱性分析アクティビティ中に基本コンポーネントに加えられた変更を考慮する。

ACO_VUL.1 統合の脆弱性レビュー

依存性: ACO_DEV.1 機能記述

開発者アクションエレメント:

ACO_VUL.1.1D **開発者は、テストのために統合 TOE を提供しなければならない。**

内容・提示エレメント:

ACO_VUL.1.1C **統合 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:

ACO_VUL.1.1E **評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを*確認しなければならない*。**

ACO_VUL.1.2E **評価者は、基本コンポーネント及び依存コンポーネントで識別された残存脆弱性が、その運用環境の統合 TOE で悪用不能であることを決定する分析を*実行しなければならない*。**

ACO_VUL.1.3E **評価者は、統合 TOE の運用環境で基本コンポーネントと依存コンポーネントを使用することで生じる可能性がある脆弱性を識別するために、公知の情報源の探索を*実行しなければならない*。**

ACO_VUL.1.4E **評価者は、基本的な攻撃能力を持つ攻撃者からの攻撃に統合 TOE が耐えられることを実証するために、識別された脆弱性に基づいて侵入テストを*実施しなければならない*。**

ACO_VUL.2 統合の脆弱性分析

依存性: ACO_DEV.2 設計の基本証拠

開発者アクションエレメント:

ACO_VUL.2.1D 開発者は、テストのために統合 TOE を提供しなければならない。

内容・提示エレメント:

ACO_VUL.2.1C 統合 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:

ACO_VUL.2.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ACO_VUL.2.2E 評価者は、基本コンポーネント及び依存コンポーネントで識別された残存脆弱性が、その運用環境の統合 TOE で悪用不能であることを決定する分析を***実行しなければならない。***

ACO_VUL.2.3E 評価者は、統合 TOE の運用環境で基本コンポーネントと依存コンポーネントを使用することで生じる可能性がある脆弱性を識別するために、公知の情報源の探索を***実行しなければならない。***

ACO_VUL.2.4E **評価者は、統合 TOE での潜在的脆弱性を識別するために、*ガイダンス*証拠資料、依存情報、及び統合の根拠を使用して、統合 TOE の独立脆弱性分析を*実行しなければならない*。**

ACO_VUL.2.5E 評価者は、基本的な攻撃能力を持つ攻撃者からの攻撃に統合 TOE が耐えられることを実証するために、識別された脆弱性に基づいて侵入テストを***実施しなければならない。***

ACO_VUL.3 拡張された基本的な統合の脆弱性分析

依存性: ACO_DEV.3 設計の詳細証拠

開発者アクションエレメント:

ACO_VUL.3.1D 開発者は、テストのために統合 TOE を提供しなければならない。

内容・提示エレメント:

ACO_VUL.3.1C 統合 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:

ACO_VUL.3.1E 評価者は、提供された情報が、証拠の内容・提示に対するすべての要件を満たしていることを***確認しなければならない。***

ACO_VUL.3.2E 評価者は、基本コンポーネント及び依存コンポーネントで識別された残存脆弱性が、その運用環境の統合 TOE で悪用不能であることを決定する分析を***実行しなければならない。***

ACO_VUL.3.3E 評価者は、統合 TOE の運用環境で基本コンポーネントと依存コンポーネントを使用することで生じる可能性がある脆弱性を識別するために、公知の情報源の探索を***実行しなければならない。***

ACO_VUL.3.4E 評価者は、統合 TOE での潜在的脆弱性を識別するために、ガイダンス証拠資料、依存情報、及び統合の根拠を使用して、統合 TOE の独立脆弱性分析を***実行しなければならない。***

ACO_VUL.3.5E 評価者は、**拡張された基本的な**攻撃能力を持つ攻撃者からの攻撃に統合 TOE が耐えられることを実証するために、識別された脆弱性に基づいて侵入テストを***実施しなければならない。***

附属書A　開発(ADV) (規定)

510　この附属書は、ADV: 開発クラスのファミリで提示されたトピックをさらに説明して追加の例を提供するための補助資料を記載する。

A.1　ADV_ARC: セキュリティアーキテクチャに関する補足資料

511　セキュリティアーキテクチャは、TSF が示す特性のセットである。これらの特性には、自己保護、ドメイン分離、及び非バイパス性が含まれる。これらの特性を持つことで、TSF がセキュリティサービスを提供することへの信頼の基礎が提供される。この附属書では、セキュリティアーキテクチャ記述の内容について説明し、これらの特性に関する追加の資料も提供する。

512　この節の残りの部分では、最初にこれらの特性について説明し、その後 TSF がそれらの特性をどのように示すかを記述するために必要な情報の種類について説明する。

A.1.1　セキュリティアーキテクチャの特性

513　*自己保護*とは、結果として TSF が変更される場合もあるような外部のエンティティによる操作から自分自身を保護する TSF の能力を指す。これらの特性がないと、TSF はセキュリティサービスを実行できない場合がある。

514　TOE が、その機能を実行するために他の IT エンティティから提供されるサービスまたは資源を使用することはよくある(下層のオペレーティングシステムに依存するアプリケーションなど)。このような場合、TSF は、自ら使用するサービスの保護を他の IT エンティティに依存するため、自らを完全には自力で保護しない。

515　*ドメイン分離*とは、TSF が、信頼できない能動的な各エンティティに対して与えられた資源で動作するために個別の *セキュリティドメイン*を作成し、エンティティが他のドメインで実行されないようにするために、ドメインを互いに分離した状態にする特性を指す。例えば、あるオペレーティングシステム TOE は、信頼できないエンティティに関連するプロセスごとに、1 つのドメイン(アドレス空間、プロセスごとの環境変数)を提供する。

516　信頼できないエンティティのアクションはすべて TSF によって仲介されるため、このようなドメインが存在しない TOE もある。パケットフィルタリングファイアウォールはこのような TOE の一例である。この TOE には、信頼できないエンティティドメインが存在せず、TSF によって維持されるデータ構造のみが存在する。このように、ドメインの存在は、1)TOE のタイプ及び 2)TOE に課せられる SFR に依存する。TOE が信頼できないエンティティにドメインを提供する場合、このファミリは、あるドメインの信頼できないエンティティが別の信頼できないエンティティのドメインからの(TSF による仲介なしに影響を受ける)改ざんを回避するように、それらのドメインが互いに分離されることを要求する。

517　*非バイパス性*は、TSF のセキュリティ機能性(SFR で特定されたもの)が、その特定のメカニズムにとって適切なタイミングで常に呼び出され、回避できないという特性である。例えば、ファイルへのアクセス制御が、SFR を通じて TSF の能力として特定されている場合は、TSF のアクセス制御メカニズムを呼び出さずにファイルへのアクセスを可能にするインタフェースが存在してはならない(ローディスクアクセスに使用されるインタフェースは、このようなインタフェースの一例であるかもしれない)。

518 自己保護と同様に、一部のTOEの本質が、TSFの非バイパス性において役割を果たすためにそれらの環境に依存する場合がある。例えば、セキュリティアプリケーション TOE は、下層のオペレーティングシステムに呼び出される必要がある。同様に、ファイアウォールは、内部及び外部のネットワーク間に直接の接続がないこと、及びそれらのネットワーク間のすべてのトラフィックがファイアウォールを通る必要があるという事実に依存する。

A.1.2 セキュリティアーキテクチャ記述

519 セキュリティアーキテクチャ記述は、上記の特性がTSFでどのように示されるかについて説明する。ドメインがどのように定義され、TSF がそれらのドメインをどのように分離するかについて記述する。信頼できないプロセスが TSF にアクセスして変更することをどのようなことによって回避するかについて記述する。TSF の制御下にあるすべての資源が適切に保護され、SFR に関連するすべてのアクションがTSFによって仲介されることをどのようなことによって保証するかについて記述する。環境がこれらのいずれかにおいて果たす役割(例えば、下層環境によって正しく呼び出されることを想定した場合、セキュリティ機能がどのように呼び出されるか)を説明する。

520 セキュリティアーキテクチャ記述は、分解された記述の観点でTSFの自己保護、ドメイン分離、及び非バイパス性という TSF の特性を示す。この記述のレベルは、主張されているADV_FSP、ADV_TDS 及び ADV_IMP 要件に必要な TSF 記述に対応するものである。例えば、ADV_FSP が使用可能な唯一の TSF 記述である場合は、TSF のあらゆる内部動作の詳細を得られないため、有意義なアーキテクチャ設計を提供することが難しくなるであろう。

521 ただし、TOE 設計が提供されていれば、最も基本的なレベル(ADV_TDS.1)であっても、TSF を構成するサブシステムに関するある程度の情報が示され、それらがどのように動作して自己保護、ドメイン分離、及び非バイパス性を実装するかが記述されているであろう。例えば、TOE に対するおそらくすべての利用者相互作用は、その利用者のすべてのセキュリティ属性を使用して利用者に代わって作動するプロセスを通じて行われるものに制約される。アーキテクチャ設計は、このようなプロセスがどのように発生し、そのプロセスのふるまいが TSF によってどのように(TSF を破壊できないように)制約され、そのプロセスのすべてのアクションが TSF によってどのように調停されるか(それによって TSF をバイパスできない理由を説明)などを記述するであろう。

522 提供される TOE 設計がより詳細である(例えばモジュールレベル)場合、あるいは実装表現も提供される場合は、それに応じてアーキテクチャ設計の記述もより詳細になり、利用者のプロセスがTSFプロセスとどのように通信するか、複数の異なる要求をTSFがどのように処理するか、どのパラメタが渡されるか、どのようなプログラムによる保護が行われるか(バッファオーバーフロー防止、パラメタ境界チェック、チェック時/使用時についてのチェック)などが説明されるであろう。同様に、ST が ADV_IMP コンポーネントを主張しているTOE は、実装固有の詳細が示されるであろう。

523 セキュリティアーキテクチャ記述で提供される説明は、それらの正確性をテストできるための十分な詳細さであることが期待される。つまり、単純な主張(「TSF はドメインを分離する」など)は、TSF が実際にドメインを作成して分離することを読者に納得させるために役立つ情報を提供しない。

A.1.2.1 ドメイン分離

524 TOE が完全に自力でドメイン分離を示す場合、これをどのように達成するかについて直接的に記述されるであろう。このセキュリティアーキテクチャ記述は、TSF で定義される各種のドメイン、それらの定義方法(各ドメインにどの資源が割り当てられるか)、保護されない資源をなくす方法、及びあるドメイン内の能動的なエンティティが、別のドメインの資源を改ざんできないようにドメインを分離する方法を説明するであろう。

525 TOE がドメイン分離で役割を果たすために他の IT エンティティに依存する場合、その役割の共有は明確にしなければならない。例えば、単なるアプリケーションソフトウェアである TOE は、TOE が定義するドメインを正しく具体化するために下層のオペレーティングシステムに依存する。つまり、TOE がドメインごとに個別の処理空間、メモリ空間などを定義すれば、TOE は下層のオペレーティングシステムに依存して正しくかつ悪意なく動作する(例えば、プロセスは TOE ソフトウェアが要求した実行空間でのみ実行できる)。

526 例えば、ドメイン分離を実装するメカニズム(例えば、メモリ管理、ハードウェアが提供する保護された処理モードなど)は、識別され、記述される。または、TSF はソフトウェアドメインの分離の実装に寄与する(利用者のアドレス空間とシステムのアドレス空間を明確に区別するなど)ソフトウェア保護構造やコーディング規則を実装する場合がある。

527 脆弱性分析及びテスト(AVA_VAN を参照)アクティビティには、TSF の監視または直接攻撃を利用することで、記述された TSF ドメイン分離を打ち負かす試みが含まれる可能性が高い。

A.1.2.2 TSF 自己保護

528 TOE が完全に自力で自己保護を示す場合、この自己保護をどのように達成するかについて直接的に記述されるであろう。他の(利用者)ドメインから保護される TSF ドメインを定義するためにドメイン分離を使用するメカニズムが識別及び記述されるであろう。

529 TOE が自己保護で役割を果たすために他の IT エンティティに依存する場合、その役割の共有は明確にしなければならない。例えば、単なるアプリケーションソフトウェアである TOE は、正しくかつ悪意なく動作するために下層のオペレーティングシステムに依存する。つまり、アプリケーションはそれ自身を破壊する(例えば、実行可能コードまたは TSF データを上書きする)悪意のあるオペレーティングシステムから自分自身を保護できない。

530 セキュリティアーキテクチャ記述は、また TSF が利用者入力によって自分自身を破壊しないように、TSF が利用者入力をどのように処理するかもカバーする。例えば TSF は、特権の概念を実装し、特権モードのルーチンを使用して利用者データを処理することによって、自分自身を保護することができる。TSF は、TSF コード及びデータを利用者コードやデータから分離するためにプロセッサベースの分離メカニズム(例えば、特権レベルやリング)を使用する場合がある。TSF は(おそらくは利用者のアドレス空間とシステムのアドレス空間を明確に区別することにより)ソフトウェアの分離の実装に寄与するソフトウェア保護構造やコーディング規則を実装する場合がある。

531 低機能モード(例えば、設置者または管理者にのみアクセスできる単一利用者モード)で立ち上げてから、評価されたセキュアな構成へ移行する(信頼できない利用者がログインして、TOE のサービスや資源を利用できるモード)TOE の場合、セキュリティアーキテクチャ記述には、評価構成で実行されないこの初期化コードから TSF をどのように保護するかについての説明も含まれる。このような TOE の場合、セキュリティアーキテクチャ記述では、初期化中にのみ使用可能であるべきサービス(資源への直接アクセスなど)が評価構成でアクセス可能になるのを回避するための説明がなされるであろう。また、TOE が評価構成であるときに初期化コードが実行されるのを回避する方法についても説明するであろう。

532 TSF が初期のセキュアな状態であると信じ込ませるような結果を招く改変を初期化プロセスが検出できるように、信頼できる初期化コードがどのように TSF(及びその初期化プロセス)の完全性を維持するかについても説明しなければならない。

533 脆弱性分析及びテスト(AVA_VAN を参照)アクティビティには、TSF の改ざん、直接攻撃、または監視を利用することで、記述された TSF 自己保護を打ち負かす試みが含まれる可能性が高い。

A.1.2.3 TSF の非バイパス性

534 非バイパス性の特性は、実施メカニズムのバイパスを許可するインタフェースに関係する。ほとんどの場合、この特性は実装によってもたらされる。つまりその実装では、オブジェクトをアクセスまたは操作するインタフェースをプログラマが作成する場合に、そのプログラマが、オブジェクトに対する SFR 実施メカニズムの一部であるインタフェースを使用し、それらのインタフェースを回避しないようにする責任を負う。そのため、非バイパス性に関する記述では、2 つの広範な領域を扱う必要がある。

535 第1の領域は、SFR 実施に対するインタフェースで構成される。これらのインタフェースの特性は、それらを使用して TSF をバイパスできる操作やモードを含んでいないという点である。この決定を行うために、ADV_FSP 及び ADV_TDS の証拠を大いに使用できることは十分に考えられる。非バイパス性は重要な問題であるため、TSFI を通じて利用できる操作の一部のみが(SFR 実施操作であるために)証拠資料として提出され、それ以外の操作は証拠資料として提出されない場合は、TSFI の非 SFR 実施操作が、実施されている方針をバイパスする能力を信頼できないエンティティに与えないことを決定するために ADV_FSP 及び ADV_TDS で提示された情報に対する追加情報が必要かどうかを、開発者は考慮するべきである。このような情報が必要である場合は、アーキテクチャ設計文書にその情報が組み込まれる。

536 非バイパス性の第 2 の領域は、SFR 実施に関連しない相互作用を持つインタフェースに関係している。主張された ADV_FSP 及び ADV_TDS コンポーネントによっては、これらのインタフェースに関する情報が、機能仕様及び TOE 設計証拠資料に存在する場合と存在しない場合がある。このようなインタフェース(またはインタフェースのグループ)に対して提示される情報は、実施メカニズムのバイパスが不可能であることを読者が(ADV: 開発クラスで提供されるその他の証拠と同等の詳細レベルで)決定できるほど十分な内容であるべきである。

537 セキュリティ機能性をバイパスできないという特性は、すべてのセキュリティ機能性に等しく適用される。つまり、設計記述は、SFR(FDP_*コンポーネントなど)で保護されるオブジェクト及びTSFが提供する機能性(監査など)を扱うべきである。また、この記述は、セキュリティ機能性に関連するインタフェースを識別するべきである。これには、機能仕様の情報が使用される場合がある。この記述は、オブジェクトマネージャなどのあらゆる設計の構成要素、及びそれらの使用方法も記述しているべきである。例えば、監査レコードを生成する標準マクロをルーチンが使用することになっている場合は、この規則が監査メカニズムの非バイパス性に寄与する設計の一部となる。この文脈での*非バイパス性*とは、「TSF 実装の一部が、不当にセキュリティ機能性をバイパスできるか」という質問に回答する試みではなく、実装がいかにしてセキュリティ機能性をバイパスしないかを証拠資料として提供する試みであるという点に注意が必要である。

538 脆弱性分析及びテスト(AVA_VAN を参照)アクティビティには、TSF の回避によって、記述された非バイパス性を打ち負かす試みが含まれる可能性が高い。

A.2 ADV_FSP: TSFI に関する補足資料

539 TSFI の仕様を特定する目的は、テストの実施に必要な情報を提供することである。つまり、TSFとの相互作用を行うために可能な手段がわからなければ、TSFのふるまいを適切にテストすることはできない。

540 TSFIの仕様の特定には、識別及び記述の2つの部分がある。考えられるTOE、及びそれらの中の TSF が多様であるため、「TSFI」を構成する標準のインタフェースセットは存在しない。この附属書では、どのインタフェースが TSFI であるかを決定する要因についてのガイダンスを提供する。

A.2.1 TSFI の決定

541 TSF に対するインタフェースを識別するには、まず TSF を構成する TOE の部分を識別しなければならない。この識別は、実際にはTOE設計(ADV_TDS)分析の一部であるが、保証パッケージにTOE設計(ADV_TDS)が含まれていない場合は、開発者によるTSFIの識別と記述を通じて暗黙に実行される。この分析では、ST の SFR を(全面的または部分的に)満たすことに寄与している TOE の一部分が、TSF にあるとみなさなければならない。これには、例えば、TSF の実行時の初期化に寄与する TOE のすべての部分が含まれる。このような部分には、SFR の実施がまだ開始されていないために(起動中など)、TSF が自己を保護できるようになる前に実行されるソフトウェアなどがある。また、アーキテクチャの原則である TSF の自己保護、ドメイン分離、及び非バイパス性(セキュリティアーキテクチャ(ADV_ARC)を参照)に寄与する TOE のすべての部分も、TSF に組み込まれる。

542 TSF が定義されると、TSFI が識別される。TSFI は、利用者が(TSF によって処理されるデータを供給することによって)TSF からサービスを呼び出すためのすべての手段、及びこれに対応するそれらのサービスの呼び出しへの応答で構成される。これらのサービス呼び出しと応答は、TSF 境界を越えるための手段である。これらの多くはすぐに明らかになるが、明確にはわからないものもある。TSFIを決定する場合、「潜在的な攻撃者はSFRの破壊を試みる際にどのようにTSFと相互作用する可能性があるか」という点を考えるべきである。様々な状況での TSFI 定義の適用について以下で説明する。

A.2.1.1 電気インタフェース

543 スマートカードなどの TOE では、相手側が TOE に論理的にアクセスできるだけでなく、物理的にも完全にアクセスできるため、TSF 境界は物理的な境界である。したがって、接触可能な電気インタフェースは、その操作が TSF のふるまいに影響する可能性があるため、TSFI とみなされる。このため、これらすべてのインタフェース(電気的な接点)について、適用される様々な電圧などを記述する必要がある。

A.2.1.2 ネットワークプロトコルスタック

544 プロトコル処理を実行する TOE の TSFI は、攻撃者が直接アクセスするプロトコル層であろう。これは、プロトコルスタック全体である必要はないが、プロトコルスタック全体になる場合もある。

545 例えば、TOE が、潜在的な攻撃者がプロトコルスタックのすべてのレベルに影響を与えることができる(つまり、任意の信号、任意の電圧、任意のパケット、任意のデータグラムを送信する)ある種のネットワーク装置である場合、TSF 境界はスタックの各層に存在する。したがって、機能仕様は、スタックのすべての層におけるすべてのプロトコルを扱う必要があるであろう。

546 ただし、TOE が、内部のネットワークをインターネットから保護するファイアウォールである場合、潜在的な攻撃者には、TOE に入る電圧を直接操作する手段がないであろう(一切の過剰電圧はインターネットを通じて簡単には渡せないだろうから)。つまり、攻撃者はインターネット層またはさらに上位の層のプロトコルにのみアクセスできるであろう。TSF 境界はスタックの各階層に存在する。したがって、機能仕様は、インターネット層と上位の層のプロトコルのみを扱う必要があるであろう(つまり、ファイアウォールへの接触が可能な各通信層を、境界上に現れる適格な入力の構成、及び適格な入力と不適格な入力の結果という点から記述するであろう)。例えば、インターネットプロトコル層の記述では、適格な IP パケットの構成内容、及び適格なパケットと不適格なパケットを受信したときの結果が記述されるであろう。同様に、TCP 層の記述では、正常な TCP 接続、及び正常な接続が確立されたときの結果と、接続を確立できなかったとき、または意図せずに接続を切断したときの結果が記述されるであろう。ファイアウォールの目的がアプリケーションレベルのコマンド(FTP や telnet など)を選別することであると仮定すると、アプリケーション層の記述では、ファイアウォールによって認識され選別されるアプリケーションレベルのコマンド、及び未知のコマンドに遭遇した場合の結果が記述されるであろう。

547 これらの層の記述では、使用される公開通信標準(telnet、FTP、TCP など)の参照と、どの利用者定義のオプションが選択されたかが記述されるであろう。

A.2.1.3 ラッパー

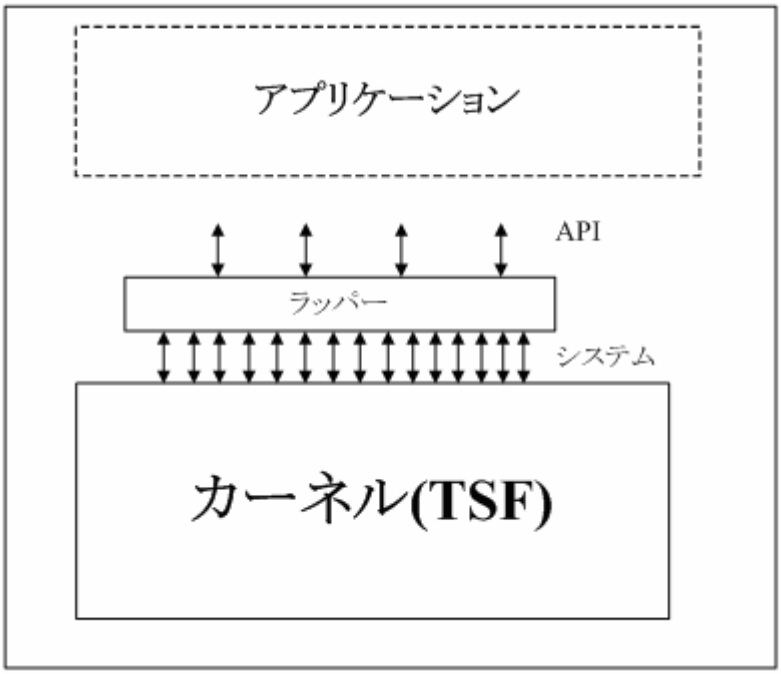

図 19 ラッパー

548 「ラッパー」は、複雑な一連の相互作用を単純化された共通のサービスに変換する。例えば、オペレーティングシステムがアプリケーションで使用できる API を作成する場合がこれにあたる(図 19 を参照)。TSFI がシステムコールであるか API であるかは、何がアプリケーションから使用可能かに依存するであろう。アプリケーションがシステムコールを直接使用できる場合、システムコールが TSFI となる。一方、システムコールを直接使用することが何らかの方法で禁止され、すべての通信を API を通じて行う必要がある場合、API が TSFI になるであろう。

549 グラフィカルユーザインタフェースも同様に、マシンで理解可能なコマンドを利用者が理解しやすいグラフィックに変換する。同様に、TSFI は、利用者がコマンドにアクセスできる場合はコマンドとなり、利用者がコマンドの使用について制約されている場合はグラフィック(プルダウンメニュー、チェックボックス、テキストフィールド)になるであろう。

550 これらの例はどちらも、利用者がよりプリミティブなインタフェース(システムコールまたはコマンド)の使用を禁じられている場合、この制約及び実施の記述が、セキュリティアーキテクチャ記述に含まれるであろうという点で注目に値する(A.1 を参照)。また、ラッパーは TSF の一部になるであろう。

A.2.1.4 アクセス不可能なインタフェース

551 ある特定の TOE では、すべてのインタフェースが*アクセス可能*とは限らない場合がある。つまり、(セキュリティターゲットにおける)運用環境のセキュリティ対策方針によって、それらのインタフェースへのアクセスが阻止される場合や、それらを事実上アクセス不可にする方法でアクセスが制限される場合がある。このようなインタフェースは、TSFI とはみなされないであろう。以下に例を挙げる。

- スタンドアロンファイアウォールに対する運用環境のセキュリティ対策方針で、「ファイアウォールは、訓練を受けた信頼できる人員のみがアクセスできるサーバルーム環境で稼働し、かつ(停電に備えて)無停電電源装置が搭載される」と記述されている場合、物理的なインタフェースと電源インタフェースはアクセス不可能となる。これは、訓練を受けた信頼できる人員が、ファイアウォールを分解したり、その電源装置の機能を無効にしたりすることがないからである。

- ソフトウェアファイアウォール(アプリケーション)に対する運用環境のセキュリティ対策方針で、「OS 及びハードウェアは、他のプログラムから改ざんされることのないようにアプリケーションにセキュリティドメインを提供する」と記述されている場合、OS 上の他のアプリケーションを介してファイアウォールにアクセスできる(例えば、ファイアウォール実行可能ファイルの削除や修正、ファイアウォールのメモリ空間に対する直接的な読み書き)インタフェースはアクセス不可能になる。これは、運用環境の OS/ハードウェアの部分が、このインタフェースをアクセス不可能にするからである。

- ソフトウェアファイアウォールに対する運用環境のセキュリティ対策方針で、OS とハードウェアが TOE のコマンドを忠実に実行し、TOE をいかなる方法によっても改ざんしないことが追加で記述されている場合、ファイアウォールが OS 及びハードウェアからプリミティブな機能性を取得する(マシンコード命令、ファイルの作成、読み取り、書き込み、削除などを行う OS API、グラフィカル API などを実行する)ようなインタフェースはアクセス不可能になる。これは、OS/ハードウェアがそのインタフェースにアクセスできる唯一のエンティティであり、完全に信頼されているからである。

上記のすべての例で、これらのアクセス不可能なインタフェースは TSFI ではないであろう。

A.2.2 例: 複雑な DBMS

552 図 20 は、複雑な TOE の例として、TOE の境界の外部にあるハードウェアとソフトウェア(これ以降は *IT 環境*と呼ぶ)に依存するデータベース管理システムを示している。この例を単純化するために、TOE は TSF と同一になっている。陰影の付いたボックスは TSF を、陰影の付いていないボックスは環境の IT エンティティを表している。TSF は、データベースエンジン及び管理 GUI(図の *DB* というラベルが付いたボックス)と、OS の一部として実行されてセキュリティ機能を実行するカーネルモジュール(図の *PLG* というラベルが付いたボックス)で構成されている。TSF のカーネルモジュールには、OS の仕様によって定義される入口点がある。OS は、この入口点を呼び出すことによって、特定の機能(デバイスドライバや認証モジュールなど)を呼び出す。ここで重要なのは、この着脱可能なカーネルモジュールが、ST の機能要件によって特定されているセキュリティサービスを提供するという点である。

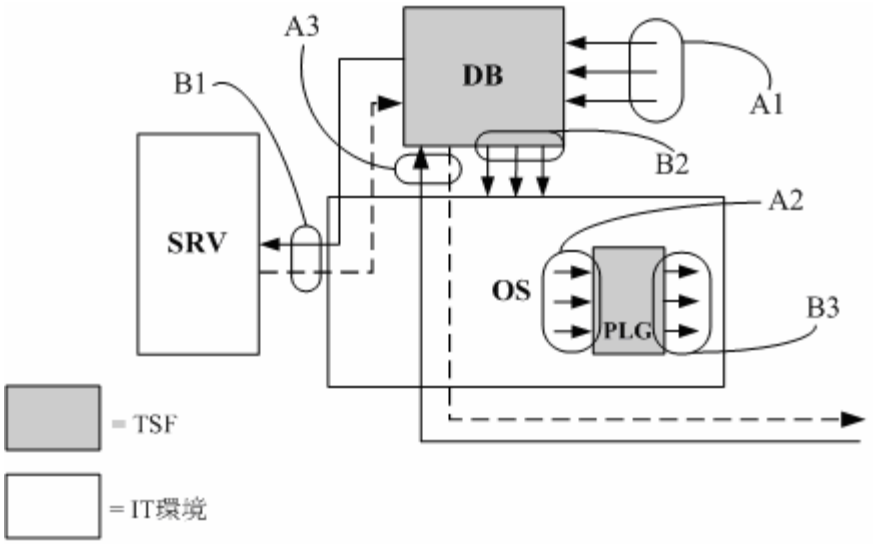

図 20　DBMS システムのインタフェース

553 IT 環境は、オペレーティングシステム自体(*OS* というラベルのボックス)及び外部サーバ(*SRV* というラベルのボックス)で構成される。この外部サーバは、OS と同様に、TSF が依存しているサービスを提供するため、IT 環境に含まれている必要がある。この図では、TSFI が Ax というラベルで示され、ACO: 統合で証拠資料として提出されるであろうその他のインタフェースが Bx というラベルで示されている。次に、これらのインタフェースのグループについてそれぞれ説明する。

554 インタフェースグループ A1 は、最も明白な TSFI のセットを表す。これらは、利用者がデータベース及びそのセキュリティ機能性と資源に直接アクセスするために使用するインタフェースである。

555 インタフェースグループ A2 は、着脱可能なモジュールによって提供される機能性を取得するために OS が呼び出す TSFI を表す。これと対照的なのがインタフェースグループ B3 で、これは、IT 環境からサービスを取得するために着脱可能なモジュールが行う呼び出しを表す。

556 インタフェースグループ A3 は、IT 環境を通過する TSFI を表している。この場合、DBMS は、アプリケーションレベルの専用プロトコルを使用してネットワーク通信を行う。様々なサポートプロトコル(例えば、イーサネット、IP、TCP)を提供する責任は IT 環境にあるが、DBMS からサービスを取得するために使用されるアプリケーション層のプロトコルは TSFI であり、そのように証拠資料に記述されなければならない。図の点線は、ネットワーク接続を通じて TSF から返される戻り値やサービスを示している。

557 *Bx* というラベルのインタフェースは、IT 環境内の機能性に対するインタフェースを表す。これらのインタフェースは TSFI ではなく、ACO クラスに関連するアクティビティの一部としての統合評価で TOE が使用されるときに、説明または分析される必要がある。

A.2.3 機能仕様の例

558 ファイアウォールの例は、内部及び外部のネットワーク間で使用される。この例では、受信データの発信元アドレスを検証する(外部データが、内部データから発信しているかのようになりすまそうとしていないことを保証するため)。なりすましの試みが検出された場合、監査ログにその違反の試みが保存される。管理者は、内部ネットワークからファイアウォールに対する telnet 接続を確立してファイアウォールに接続する。管理者のアクションは、認証、パスワードの変更、監査ログのレビュー、及び内部ネットワークと外部ネットワークのアドレスの設定や変更で構成される。

559 ファイアウォールの例では、内部ネットワークへの以下のインタフェースがある:

- IP データグラム
- 管理者コマンド

また、外部ネットワークへの以下のインタフェースがある:

- IP データグラム

560 **インタフェース記述: IP データグラム**

561 データグラムは、RFC 791 で特定されている形式に従う。

- 目的 - 発信元ホストから宛先ホスト(ともに固定長のアドレスで識別される)にデータのブロック(「データグラム」)を送信する。また、長いデータグラムについては、小さいパケットネットワークを通じて送信できるように、必要に応じて断片化及び再組み立ても提供する。
- 使用方法 - これらは下位レベル(データリンクなど)のプロトコルから送られる。
- パラメタ - IP データグラムヘッダーのフィールド(発信元アドレス、宛先アドレス、断片化禁止フラグ)
- パラメタ記述 - [RFC 791 の 3.1 節(「インターネットヘッダーフォーマット」)の定義に従う]
- アクション - なりすまされていないデータグラムを送信する。必要に応じて大きいデータグラムを断片化する。複数の断片をデータグラムに再組み立てする。
- 誤りメッセージ - (なし)。保証された信頼性がない(上位レベルのプロトコルで提供される信頼性)、配信不可能なデータグラム(送信するために断片化する必要があるが、断片化禁止フラグが設定されている)は破棄。

562 **インタフェース記述: 管理者コマンド**

563 管理者コマンドは、ファイアウォールとの相互作用の手段を管理者に提供する。これらのコマンド及び応答は、内部ネットワーク上の任意のホストから確立された telnet (RFC 854) 接続上で実行される。使用可能なコマンドは以下のとおりである:

- **Passwd**
  - 目的 - 管理者パスワードを設定する

- 使用方法 - **Passwd** <*パスワード*>
- パラメタ - パスワード
- パラメタ記述 - 新しいパスワードの値
- アクション - パスワードを新しく指定された値に変更する。制限はない。
- 誤りメッセージ - なし。

- **Readaudit**
  - 目的 - 管理者に監査ログを提示する
  - 使用方法 - **Readaudit**
  - パラメタ - なし
  - パラメタ記述 - なし
  - アクション - 監査ログのテキストを提供する
  - 誤りメッセージ - なし。

- **Setintaddr**
  - 目的 - 内部ネットワークのアドレスを設定する。
  - 使用方法 - **Setintaddr** <*アドレス*>
  - パラメタ - アドレス
  - パラメタ記述 - IP アドレスの先頭から 3 つ目までのフィールド(RFC 791 の定義に従う)。例: 123.123.123。
  - アクション - 内部ネットワークを定義する変数の内部値、つまり、なりすましが試行されているかどうかを判断するために使用される値を変更する。
  - 誤りメッセージ -「アドレスは使用中」: 識別された内部ネットワークが、外部ネットワークと同一であることを示す。

- **Setextaddr**
  - 目的 - 外部ネットワークのアドレスを設定する
  - 使用方法 - **Setextaddr** <*アドレス*>
  - パラメタ - アドレス
  - パラメタ記述 - IP アドレスの先頭から 3 つ目までのフィールド(RFC 791 の定義に従う)。例: 123.123.123。
  - アクション - 外部ネットワークを定義する変数の内部値を変更する。

- 誤りメッセージ -「アドレスは使用中」: 識別された外部ネットワークが、内部ネットワークと同一であることを示す。

A.3 ADV_INT: TSF 内部構造に関する補足資料

564 TOE の多様性から、「適切に構造化された」または「最小の複雑さ」よりも具体的なものを体系化することは不可能である。構造と複雑さに関する判断は、TOE で使用される特定の技術から導き出されることが期待される。例えば、ソフトウェアエンジニアリングの分野で挙げられる特性を示す場合、ソフトウェアは適切に構造化されたものとみなされる可能性が高い。

565 この附属書では、TSF の手続きベースのソフトウェア部分の構造及び複雑さの評定に関する補足資料を提供する。この資料はソフトウェアエンジニアリングの文献で容易に入手できる情報に基づいている。その他の種類の内部構造(例えば、ハードウェア、オブジェクト指向のコードなどの非手続き型ソフトウェア)については、規範に関する対応する文献を参照するべきである。

A.3.1 手続き型ソフトウェアの構造

566 手続き型ソフトウェアの構造は、従来、その*モジュール性*に従って評定される。モジュール設計を使用して作成されたソフトウェアでは、理解のしやすさの実現を支援するために、モジュール間の依存性を明確化し(*結合度*)、互いに強い関連を持つタスクのみをモジュールに組み込む(*凝集度*)。モジュール設計の使用によって、TSF に含まれる要素間の相互依存性が減り、それによって1つのモジュールでの変更または誤りが、TOE 全体に影響を及ぼすリスクが軽減される。その使用によって、設計がより明解になり、予期しない結果が起きないことの保証が高まる。モジュール分解のもう 1 つの望ましい特性は、冗長なコードや不要なコードの量が減ることである。

567 TSF における機能性の量を最小化することで、評価者及び開発者が SFR の実施に必要な機能性のみに集中できるようになり、理解のしやすさのさらなる向上及び設計または実装の誤りが発生する可能性のさらなる軽減に役立つ。

568 モジュール分解、階層化、及び最小化を、設計と実装のプロセスに統合する際には、適切なソフトウェアエンジニアリングについて考慮しなければならない。実用的で有用なソフトウェアシステムは、たいてい、モジュール間の望ましくない結び付き、関連性の弱い機能を含んだモジュール、及びモジュール設計の難解さと複雑さを内包する。理想的なモジュール分解からのこれらの逸脱は、パフォーマンス、互換性、将来予定されている機能性、またはその他の要因に関連する目標や制約を達成するために必要と判断されることが多く、それらの逸脱に対して開発者が提供する正当化に基づいて受け入れられる場合がある。このクラスの要件を適用する際には、適切なソフトウェアエンジニアリングの原則について十分に検討しなければならないが、理解のしやすさを達成するという全体的な目的も達成されなければならない。

A.3.1.1 凝集度

569 凝集度とは、単一のソフトウェアモジュールによって実行されるタスクが相互に関連する方法とその度合いである。凝集度には、偶発的、通信的、機能的、論理的、連続的、時間的の各タイプがある。以下に、これらの凝集度のタイプを望ましいものから順にリストし、その特徴を示す。

- *機能的*凝集度 - 機能的凝集度を持つモジュールは、単一の目的に関連するアクティビティを実行する。機能的に凝集するモジュールは、スタックマネージャやキューマネージャのように、単一タイプの入力を単一タイプの出力に変換する。

- *連続的*凝集度 - 連続的凝集度を持つモジュールでは、モジュールに含まれる各機能の出力がモジュールに含まれるその次の機能の入力となる。連続的に凝集するモジュールの例としては、監査レコードを書き出す機能及び特定タイプの監査違反の累積数を動的にカウントする機能を含んだモジュールが挙げられる。

- *通信的*凝集性 - 通信的凝集性を持つモジュールでは、ある機能が同じモジュール内の他の機能に対して出力を生成するか、または他の機能からの出力を使用する。通信的に凝集するモジュールの例としては、強制チェック、任意チェック、及び権限(capability)チェックを含んだアクセスチェックモジュールが挙げられる。

- *時間的*凝集度 - 時間的凝集度を持つモジュールでは、大体同時に実行する必要がある複数の機能を含む。時間的に凝集するモジュールの例としては、初期化、回復、シャットダウンなどのモジュールが挙げられる。

- *論理的*(または *手続き的*)凝集度 - 論理的凝集度を持つモジュールは、類似するアクティビティを異なるデータ構造に対して実行する。モジュールの機能が、別々の入力に対して、関連しているが異なっている操作を実行する場合、そのモジュールは論理的凝集度を示す。

- *偶発的*凝集度 - 偶発的凝集度を持つモジュールは、関連がまったくない、またはほとんどない複数のアクティビティを実行する。

A.3.1.2 結合度

570 結合度は、ソフトウェアモジュール間の相互依存の方法とその度合いである。結合には、コール、共通、内容の各タイプがある。以下に、これらの結合度のタイプを望ましいものから順にリストし、その特徴を示す。

- *コール*: 2 つのモジュールが、厳密にそれぞれの証拠資料に記述された機能コールの使用を通して通信する場合、これらのモジュールはコール結合されている。コール結合の例としては、次に定義するデータ、スタンプ、制御がある。

    1. *データ*: 2 つのモジュールが、厳密に単一のデータ項目を表すコールパラメタの使用を通して通信する場合、それらのモジュールはデータ結合されている。

    2. *スタンプ*: 2 つのモジュールが、複数のフィールドからなるコールパラメタ、または意味のある内部構造を持つコールパラメタの使用を通して通信する場合、それらのモジュールはスタンプ結合されている。

    3. *制御*: 2 つのモジュールの一方が、他方の内部ロジックに影響するように意図された情報を渡す場合、それらのモジュールは制御結合されている。

- *共通*: 2 つのモジュールが共通データ領域または共通システム資源を共有する場合、それらのモジュールは共通結合されている。グローバル変数は、それを使用するモジュールが共通結合されていることを示す。グローバル変数による共通結合は、一般に許可されているが、その程度は限定される。例えば、グローバル領域に置かれているが、単一のモジュールのみが使用する変数は、配置が不適切であり、削除するべきである。このほかに、グローバル変数の適切性を評定する際には、次の要因を検討する必要がある:

  1. グローバル変数を改変するモジュールの数: 一般に、グローバル変数の内容を制御する責任は 1 つのモジュールのみに割り当てるべきであるが、第 2 のモジュールと責任を共有する状況も発生する。このような場合は、十分な正当性を提示する必要がある。2 つより多いモジュール間でこの責任を共有することは受け入れられない(この評定を行う際には、変数の内容について実際に責任を負うモジュールを注意して決定するべきである。例えば、単一のルーチンを使用して変数を改変するが、そのルーチンが単にその呼び出し側から要求された改変を実行する場合、呼び出し側モジュールが責任を負うことになり、責任を負うモジュールが複数になる可能性がある)。さらに、複雑さ決定の一環として、2 つのモジュールがグローバル変数の内容について責任を負う場合は、それらのモジュール間で改変がどのように調整されるかが明確に示されるべきである。

  2. グローバル変数を参照するモジュールの数: 一般に、グローバル変数を参照するモジュールの数に制限はないが、多数のモジュールが参照する場合は、妥当性と必要性を調査すべきである。

- *内容*: 2 つのモジュールの一方が他方の内部を直接参照できる場合、それらのモジュールは内容結合されている(例えば、他方のモジュールのコードを改変する場合やその内部ラベルを参照する場合)。その結果、一方のモジュールの内容の一部または全部が、他方のモジュールに実質的に取り込まれる。内容結合は非通知型モジュールインタフェースを使用しているとみなすことができる。これは、通知型モジュールインタフェースを使用するコール結合とは対照的である。

A.3.2 手続き型ソフトウェアの複雑さ

571 複雑さとは、コードが実行される際の決定ポイント及び論理パスの尺度である。ソフトウェアエンジニアリングの文献では、複雑さは、コードのロジックと流れの理解を妨げるため、ソフトウェアの否定的な特性として挙げられる。コードの理解を妨げるもう 1 つのものとして、使用されないまたは冗長という点で不要なコードな存在が挙げられる。

572 階層化の使用によって抽象化のレベルを分離し、循環的な依存性を最小化することで、TSF がさらに理解しやすくなり、TOE セキュリティ機能要件が実装で正確かつ完全に実現されることの保証が高まる。

573 複雑さの軽減には、相互依存性の軽減/排除という概念も含まれる。これは、同じ層にあるモジュールと別々の層にあるモジュールの両方に関係する。相互に依存するモジュールは互いに依存して 1 つの結果を導き出すが、それがデッドロック状態を招いたり、さらに悪い場合は、最終的な結論が決まらずに、ある瞬間の計算環境の影響を受ける競合状態(例えば、チェック時に対する使用時の問題)に陥る可能性がある。

574 設計の複雑さの最小化は、リファレンス確認メカニズムの主要な特質であり、その目的は、容易に理解できる TSF を実現して、TSF を完全に分析できるようにすることである(リファレンス確認メカニズムには、他にも TSF の自己保護や非バイパス性などの重要な特性がある。これらの特性は、ADV_ARC ファミリの要件で扱われる)。

A.4 ADV_TDS: サブシステム及びモジュール

575 この節では、TDS ファミリに関する追加のガイダンスを提供し、このファミリでの「サブシステム」と「モジュール」という用語の用法を示す。その後に、提供される詳細レベルが高くなると、低い詳細レベルに対する要件が縮小されるしくみについて説明する。

A.4.1 サブシステム

576 図 21 は、TSF の複雑さに応じて、設計がサブシステムとモジュールの観点から記述される場合(サブシステムの抽象レベルがモジュールより高い場合)と、単にある抽象レベル(例えば、下位の保証レベルでの *サブシステム*、上位の保証レベルでの *モジュール*)の観点から記述される場合があることを示している。下位レベルの抽象(モジュール)が提示される場合、上位レベルの抽象(サブシステム)に課せられる要件は、基本的に何もしなくても満たされる。この概念については、この後のサブシステムとモジュールに関する説明でさらに詳しく述べる。

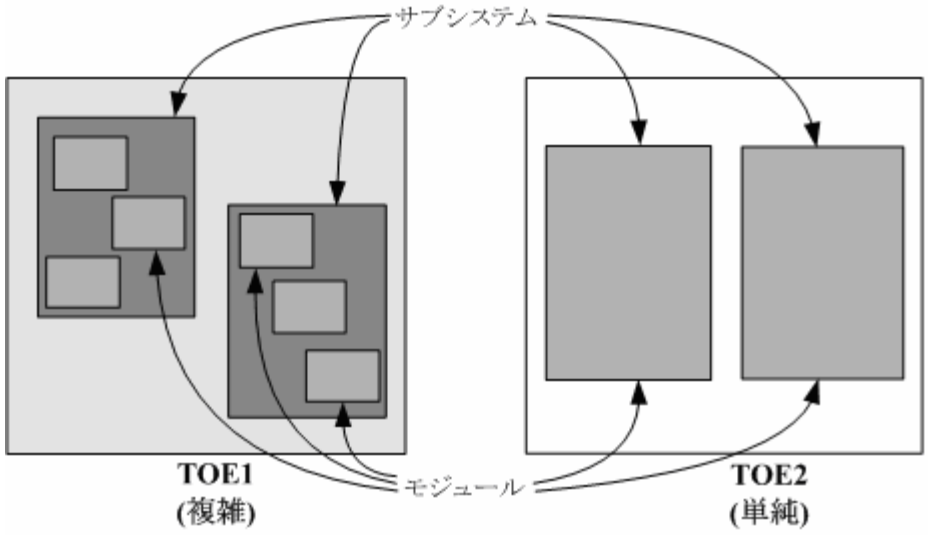

図 21 サブシステム及びモジュール

577 開発者は、*サブシステム*の観点から TOE の設計を記述することが期待される。「サブシステム」は、特に曖昧になるように選択された用語であるため、TOE 固有の単位(例えば、サブシステムやモジュール)を指す可能性もある。サブシステムは、サブシステムの記述に関する要件が満たされている限り、範囲さえも均一でなくてよい。

578 サブシステムの第 1 の用法は、TSF の境界、つまり TSF を構成する TOE の部分を区別することである。一般に、サブシステムがいずれかの SFR の正しい動作に影響する(設計または実装によって)能力を持つ場合、そのサブシステムは TSF の一部である。例えば、ソフトウェアが様々なハードウェア実行モードに応じてドメイン分離(A.1 を参照)を提供し、その 1 つのドメインで SFR 実施コードが実行される場合、そのドメインで実行されるすべてのサブシステムは TSF の一部とみなされるであろう。同様に、そのドメインの外部にあるサーバが SFR を実装する場合(例えば、そのサーバが管理するオブジェクト上でアクセス制御方針を実施する場合)は、そのサーバも TSF の一部とみなされるであろう。

579 サブシステムの第 2 の用法は、TSF を記述するための構造を提供することで、その記述レベルは、TSF の作用を記述する一方で、モジュール記述(後で説明)に見られる下位レベルの実装詳細は必ずしも含んでいない。サブシステムは、上位レベル(実装の詳細がほとんど記述されない)または詳細レベル(実装に対するより深い洞察を提供)のいずれかで記述される。サブシステムに提供される記述のレベルは、サブシステムが SFR の実装にどの程度責任を持っているかによって決まる。

580 *SFR 実施*サブシステムは、任意の SFR のエレメントを実施するためのメカニズムを提供するサブシステム、または SFR の実施に責任を持つサブシステムを直接支援するサブシステムである。サブシステムが SFR 実施 TSFI を提供(実装)する場合、そのサブシステムは SFR 実施である。

581 サブシステムには、*SFR 支援*または *SFR 非干渉*と識別されるものもある。SFR 支援サブシステムは、SFR を実装するために SFR 実施サブシステムが依存しているサブシステムであるが、SFR 支援要件ほど直接的な役割を果たさない。SFR 非干渉サブシステムは、支援の役割においても実施の役割においても、SFR を実装するために依存されないサブシステムである。

A.4.2 モジュール

582 モジュールは、一般的に、TSF 内部構造(ADV_INT)で説明される特性の観点から特徴を表すことができる比較的小さいアーキテクチャの単位である。ADV_TDS.3 基本モジュール設計(またはそれ以上)の要件及び TSF 内部構造(ADV_INT)の要件の両方が PP または ST に存在する場合、TOE 設計(ADV_TDS)の要件の観点での「モジュール」は、TSF 内部構造(ADV_INT)の要件に対する「モジュール」と、同じものを指す。サブシステムとは異なり、モジュールは、実装表現のレビューに対するガイドとしての役割を果たすことができる詳細レベルで実装を記述する。

583 TOE によっては、モジュールとサブシステムが同じ抽象概念を指す場合があるので注意が必要である。ADV_TDS.1 基本設計と ADV_TDS.2 アーキテクチャ設計(モジュールレベルでの記述を要求しない)の場合、サブシステム記述は TSF に関する最下位レベルの詳細を提供する。ADV_TDS.3 基本モジュール設計(モジュール記述を要求する)の場合、これらの記述は最下位レベルの詳細を提供する一方で、サブシステム記述(別のものとして存在する場合)は、単にモジュール記述の枠組みを示すために使用される。つまり、モジュール記述が存在する場合は、詳細なサブシステム記述を提供する必要がない。きわめて単純な TOE では、独立した「サブシステム記述」が必要ない。つまり、モジュールによって提供される証拠資料で要件を満たすことができる。複雑な TOE での(TSF に関する)サブシステム記述の目的は、読者がそれぞれの分析の焦点を適切に絞り込むことができるように枠組みを提供することである。図 21 には、この違いが示されている。

584 SFR 実施モジュールは、ST のセキュリティ機能要件(SFR)を直接実装するモジュールである。このようなモジュールは一般に SFR 実施 TSFI を実装するが、SFR で表現されている一部の機能性(例えば、監査機能やオブジェクト再使用機能性)が単一の TSFI に直接結び付いていない場合がある。サブシステムの場合と同様に、SFR 支援モジュールは、SFR 実施モジュールから依存されているが、SFR を直接実装する責任を負わないモジュールである。SFR 非干渉モジュールは、SFR の実施を直接的にも間接的にも扱わないモジュールである。

585 「直接実装する」が何を意味するかの判断は、やや主観的になるので注意が必要である。最も狭義には、要件を実装する比較やゼロ化操作などを実際に実行する 1、2 行のコードを意味すると解釈できる。解釈を広げると、SFR 実施 TSFI に応答して呼び出されるモジュール、及びそのモジュールによって呼び出されることがあるすべてのモジュール(呼び出しが完了するまで続く)までが含まれるかもしれない。どちらの解釈も特に十分ではない。なぜなら、最初の解釈は意味が狭いために、重要なモジュールが間違って SFR 支援と分類される可能性があり、2 番目の解釈では、実際には SFR 実施でないモジュールが SFR 実施と分類されてしまうからである。

586 モジュールの記述は、その記述からモジュールの実装を作成できるものであるべきであり、その結果の実装は 1)モジュールによって提示及び使用されるインタフェースの観点から実際の TSF 実装と同一であり、2)TSF モジュールとアルゴリズムが同一である。例えば、RFC 793 は TCP プロトコルの上位レベル記述を提供する。これは、必ずしも実装には依存していない。これは、豊富な詳細を提供するが、実装の特定ではないため、適切な設計記述では***ない***。実際の実装は RFC で指定されているプロトコルを追加でき、実装の選択(例えば、実装の様々な部分で、グローバルデータとローカルデータのどちらを使用するか)は実行される分析に対して影響を与える可能性がある。TCP モジュールの設計記述は、(RFC 793 で定義されたインタフェースだけではなく)実装によって提示されるインタフェース、及び TCP を(TSF の一部であったと想定して)実装しているモジュールに関連する処理のアルゴリズム記述をリストするであろう。

587 設計では、モジュールが提供する機能(目的)、モジュールが提示するインタフェース、それらのインタフェースからの戻り値、モジュールが使用するインタフェース(他のモジュールが提示)、及びモジュールがその機能性を提供する方法を示すアルゴリズム記述の観点から、モジュールが詳細に記述される。

588 モジュールの目的は、モジュールが提供する機能を示しながら記述されるべきである。また、アーキテクチャにおけるモジュールの機能を読者が全般的に把握できるように十分に記述されるべきである。

589 モジュールが提示するインタフェースは、提供されている機能性を呼び出すために他のモジュールが使用するインタフェースである。インタフェースには、*明示的な*インタフェース(例えば、他のモジュールによって呼び出されるコーリングシーケンス)及び*暗黙の*インタフェース(例えば、モジュールによって操作されるグローバルデータ)の両方が含まれる。インタフェースは、どのように呼び出されるかという観点から、及び戻されるすべての値の観点から記述される。この記述には、パラメタのリスト、及びこれらのパラメタの記述が含まれるであろう。あるパラメタが値のセット(例えば「フラグ」パラメタ)であることを期待されていた場合、処理しているモジュールに影響を与えるパラメタがとり得る値の完全なセットが特定されるであろう。同様に、データ構造を表すパラメタは、データ構造の各フィールドが識別及び記述されるように記述される。グローバルデータについては、モジュールがそのデータの読み取りと書き込みのどちら(または両方)を行うかが記述されるべきである。

590 プログラミング言語によっては、明白とはならない追加の「インタフェース」を持つ可能性がある。この例として挙げられるのは、C++における演算子/関数のオーバーロードがあるだろう。クラス記述におけるこの「暗黙のインタフェース」は、モジュール設計の一部としても記述されるであろう。モジュールは 1 つのインタフェースのみを提示する可能性があるが、関連するインタフェースの小規模なセットをモジュールが提示することのほうがより一般的である。

591 それとは対照的に、あるモジュールが使用するインタフェースは、その記述されているモジュールによってどのモジュールが呼び出されているかを決定できるように識別されなければならない。設計記述から、呼び出されるモジュールがコールされるアルゴリズム上の理由も明確でなければならない。例えば、モジュール A が記述されており、それがモジュール B のバブルソートルーチンを使用する場合、「モジュール A は、モジュール B 内の *double_bubble()*インタフェースを呼び出してバブルソートを実行する」は不適切なアルゴリズム記述であろう。適切なアルゴリズム記述は、「モジュール A は、アクセス制御エントリのリストを渡して *double_bubble* ルーチンを呼び出し、*double_bubble()*は、最初に利用者名でソートされたエントリを戻してから、次の規則に従って access_allowed フィールドでソートされたエントリを戻し...」となるであろう。設計におけるモジュールの詳細な記述は、モジュール A がバブルソートインタフェースから期待する効果が明確となるように十分な詳細を提供しなければならない。これらコールしたインタフェースを提示する 1 つの方法はコールツリーを介した方法であり、それにより、アルゴリズム記述はコールされたモジュールのアルゴリズム記述に含めることができる。

592 すでに説明したように、モジュールのアルゴリズム記述は、アルゴリズム方式でモジュールの実装を記述するべきである。これは、擬似コード、フローチャート、または(ADV_TDS.3 基本モジュール設計での)非形式的説明文で実現することができる。この記述では、モジュール入力と呼び出される関数をどのように使用してモジュールの機能が達成されるかが説明される。また、グローバルデータ、システム状態、モジュールによって生成される戻り値に対する影響について言及する。この記述は、TOE の実際の実装にきわめて似た実装を導き出せる程度の詳細レベルにある。

593 ソースコードは、モジュール証拠資料の要件を満たさないことに注意するべきである。モジュール設計は実装を記述するが、実装そのものではない。ソースコードの周囲にあるコメントがソースコードの意図を説明している場合は、それらが十分な証拠資料となることがある。単に各コード行の処理内容を記述するインラインコメントは、モジュールが達成するべき内容を説明しないので役に立たない。

594 以下のエレメントでは、サブシステムとモジュールについて検討されたラベル(SFR 実施、SFR 支援、及び SFR 非干渉)が、開発者が提供する必要がある情報の量とタイプを記述するために使用される。エレメントは、開発者が特定された情報*のみ*を提供することを期待しないように構成されている。つまり、開発者が提供する TSF の証拠資料が以下の要件の情報を提供する場合、開発者が自らの証拠資料を更新し、サブシステムとモジュールを SFR 実施、SFR 支援、または SFR 非干渉とラベル付けすることは期待されない。このラベル付けの主な目的は、成熟した開発方法(及び詳細なインタフェース及び設計証拠資料などの関連する資料)を確立していない開発者が、過度なコストをかけずに必要な証拠を提供できるようにすることである。

A.4.3 レベル付けアプローチ

595 何が SFR 実施で何が SFR 支援かを決定(さらに場合によっては何が SFR 非干渉かも決定)する際には主観が影響するため、このファミリでは次のパラダイムが採用されている。このファミリの前半のコンポーネントでは、開発者がサブシステムを SFR 実施などに分類するための決定を行い、適切な情報を提供する。また、評価者がこの主張を支援するために検査する追加の証拠はほとんどない。望まれる保証のレベルが高くなると、開発者はやはり分類に関する決定を行うが、評価者には、開発者の分類を確認するために使用する証拠がより多く提供されるようになる。

596 評価者の分析をTOEのSFR関連部分に集中させるために、特に低いレベルの保証では、最初にSFR実施アーキテクチャエンティティについてのみ詳細な情報が要求されるように、ファミリのサブシステムがレベル付けされる。保証のレベルが高まるにつれて、SFR 支援エンティティ及び(最終的には)SFR非干渉エンティティについて、より多くの情報が要求されるようになる。完全な情報が要求されている場合でも、この情報すべてを同じ詳細レベルで分析することは要求されないので注意する必要がある。いずれの場合も、*必要な*情報が提供され、分析されるかどうかに焦点を置くべきである。

597 表 13 は、記述されるアーキテクチャエンティティについて各ファミリサブシステムで要求される情報を要約したものである。

	TSF サブシステム			TSF モジュール		
	SFR 実施	SFR 支援	SFR 非干渉	SFR 実施	SFR 支援	SFR 非干渉
ADV_TDS.1 基本設計 (非形式的表現)	アーキテクチャ、 SFR 実施の ふるまいの上位 レベルの記述、 相互作用	指定支援(1)	指定支援			
ADV_TDS.2 アーキテクチャ 設計 (非形式的表現)	アーキテクチャ、 SFR 実施の ふるまいの 詳細な記述、 その他の ふるまいの上位 レベルの記述、 相互作用	アーキテクチャ、 ふるまいに 関する上位 レベルの記述、 相互作用	指定支援、 相互作用			
ADV_TDS.3 基本モジュール 設計 (非形式的表現)	記述、 相互作用	記述、 相互作用	記述、 相互作用	共通データ、 インタ フェース(2)、 アルゴリズム(3)	相互作用、 目的	相互作用、 目的
ADV_TDS.4 準形式的 モジュール設計 (準形式的表現)	記述、 相互作用	記述、 相互作用	記述、 相互作用	共通データ、 インタ フェース、 アルゴリズム	共通データ、 インタ フェース、 アルゴリズム	相互作用、 目的
ADV_TDS.5 完全な 準形式的 モジュール設計 (準形式的表現)	記述、 相互作用	記述、 相互作用	記述、 相互作用	共通データ、 インタ フェース、 アルゴリズム	共通データ、 インタ フェース、 アルゴリズム	共通データ、 インタ フェース、 アルゴリズム
ADV_TDS.6 形式的な 上位レベルの 設計提示を伴う 完全な 準形式的 モジュール設計 (準形式的 表現、追加の 形式的表現)	記述、 相互作用	記述、 相互作用	記述、 相互作用	共通データ、 インタ フェース、 アルゴリズム	共通データ、 インタ フェース、 アルゴリズム	共通データ、 インタ フェース、 アルゴリズム

(1)指定支援とは、サブシステム/モジュールの分類を支援するのに十分な証拠資料のみが必要であることを意味する。

(2)インタフェースとは、モジュール記述に目的、提示されるインタフェース、及び使用されるインタフェースが含まれていることを意味する。

(3)アルゴリズムとは、モジュール全体のアルゴリズム記述が提供されることを意味する。

表 13　記述の詳細に関するレベル付け

A.5 形式的な方法に関する補足資料

598 形式的な方法は、TSF 及びそのふるまいの数学的表現を提供し、ADV_FSP.6 追加の形式的仕様を伴う完全な準形式的機能仕様、ADV_SPM.1 形式的 TOE セキュリティ方針モデル、及び ADV_TDS.6 形式的な上位レベルの設計提示を伴う完全な準形式的モジュール設計の各コンポーネントで必要とされる。形式的な方法には 2 つの側面がある。1 つは形式的な表現に使用される*仕様言語*で、もう 1 つは形式的仕様の完全性と正確さを数学的に証明する*定理証明系*である。

599 形式的仕様は、確立された数学上の概念に基づいて、形式的体系内で表現される。これらの数学的概念は、明確に定義された意味、構文及び推論則を定義するために使用される。形式的体系は、形式的なアルファベット、形式的構文に基づいた、そのアルファベットによる形式的な言語、及び形式的な言語で文の導出を構成する形式的な推論則のセットを特定することで記述できる識別情報及び関係の抽象体系である。

600 評価者は、以下の点を確認するために識別された形式的体系を検査するべきである:

- 形式的体系の意味、構文、及び推論則が定義されている、または定義が参照されていること。
- 各形式的体系に、以下のことができるように定義されている**意味**を提供する説明文が併記されていること:
    1. 説明文は、通常の使用では受け入れられない文脈で使用される用語、省略語及び頭文語の定義された意味を提供する、
    2. 形式的体系の使用及び準形式的表記の使用には、曖昧さのないように、適切に支援する非形式的スタイルの説明文が併記される、
    3. 形式的体系は、適用可能な SFP、表記が使用される保証ファミリに対して特定される TSF やそのサブシステムまたはモジュールのセキュリティ機能性や(影響、例外及び誤りメッセージの詳細を提供する)インタフェースの規則及び特性を表現することができる。
    4. 表記は、構文上有効な構造の意味を決定するための規則を提供する。
- 各形式的体系は、構造を曖昧さなく認識するための規則を提供する 1 つの形式的構文を使用する。
- 各形式的体系は、次のことを行う証明規則を提供する
    1. 確立された数学的な概念の論理的な推論の支援、
    2. 矛盾の導出を回避するための支援

601 開発者が認証機関ですでに受け入れられている形式的体系を使用する場合、評価者は、その体系の形式性及び強度の程度を信頼し、TOE 仕様及び対応の証明への形式的体系の具体化に焦点を絞ることができる。

602 形式的スタイルは、セキュリティ上の特徴、詳細化の一貫性、及び表現の対応に基づいてセキュリティ特性の数学的な証明を支援する。形式的なツールの支援は、手動による導出が冗長で理解不可能になるような場合は常に適切と思われる。形式的ツールは、手動による導出に固有の誤りの確率を下げる傾向にもある。

603 形式的システムの例を以下に示す:

- **Z 言語**は表現力の高い言語であり、形式的仕様の様々な方法やスタイルを支援する。Z の使用は、形式的に操作を特定するための*スキーム*を使用する、モデル指向の仕様で広く普及している。詳細については、http://vl.zuser.org/を参照のこと。

- **ACL2** は、LISP ベースの仕様言語及び定理証明系で構成されるオープンソースの形式的体系である。詳細については、http://www.cs.utexas.edu/users/moore/acl2/を参照のこと。

- **Isabelle** は、一般に広く普及している定理証明系環境であり、形式的な言語で数式を表すことができ、論理計算の範囲で式を証明するためのツールを提供する(詳細については、http://www.cl.cam.ac.uk/Research/HVG/Isabelle/を参照のこと)。

- **B 方法**は、命題計算、推論則による第一階の述語計算、及び集合論に基づく形式的体系である(詳細については、http://vl.fmnet.info/b/を参照のこと)。

附属書B　統合(ACO) (参考)

604　この附属書の目標は、構成評価と ACO 基準の背景にある概念を説明することである。本附属書では ASE 基準の定義は行わない。定義は、11 章にある。

B.1　統合 TOE 評価の必要性

605　IT 市場は全体として、特定のタイプの製品/技術を提供するベンダで構成されている。PC のハードウェアベンダがアプリケーションソフトウェア及び/またはオペレーティングシステムも提供する場合や、チップメーカーが自社のチップセット専用のオペレーティングシステムを開発する場合のように、部分的な重複も見られるが、1 つの IT ソリューションが様々なベンダによって実装されることは少なくない。

606　個々のコンポーネントの保証に加えてコンポーネントの組み合わせ(統合)における保証が必要な場合がある。コンポーネントの技術的な統合に必要な一定の資料を配布する際にはベンダ間で協力が行われるが、詳細な設計情報及び開発プロセス/手続きの証拠を提供するところまで合意が拡大することはまれである。あるコンポーネントが依存しているコンポーネントの開発者からこの情報を入手できないことは、依存コンポーネントの開発者が EAL2 以上の依存コンポーネントと基本コンポーネント両方の評価を実行するために必要な情報にアクセスできないことを意味する。したがって、依存コンポーネントの評価はどの保証レベルでも実行できるが、EAL2 以上のレベルの保証でコンポーネントを構成するには、評価証拠及びコンポーネント開発者向けに行なわれた評価の結果を再使用する必要がある。

607　ACO 基準は、ある IT エンティティが別の IT エンティティにセキュリティサービスの提供を依存している状況で適用できるように意図されている。サービスを提供するエンティティは「基本コンポーネント」と呼ばれ、そのサービスを受けるエンティティは「依存コンポーネント」と呼ばれる。この関係が存在する状況はいくつかある。例えば、アプリケーション(依存コンポーネント)が、オペレーティングシステム(基本コンポーネント)の提供するサービスを使用する場合がある。あるいは、共通のオペレーティングシステム環境内または別々のハードウェアプラットフォーム上で稼働する 2 つのリンクされたアプリケーションという意味では、この関係はピアツーピアの可能性もある。比較的重要でないピアにサービスを提供する主要なピアが存在する場合は、その主要なピアが基本コンポーネント、比較的重要でないピアが依存コンポーネントとみなされる。ピアが互いにサービスを提供し合う場合は、それぞれのピアが、提供されるサービスについては基本コンポーネントとみなされ、要求されるサービスについては依存コンポーネントとみなされる。この場合、ACO コンポーネントの繰返しによって、各タイプのコンポーネントピアにすべての要件を適用する必要がある。

608　また、この基準は、より複雑な関係で段階的により広く適用されるように意図されているが(この場合、依存コンポーネントと基本コンポーネントそのもので構成される統合 TOE は、別の統合 TOE の基本コンポーネントになる)、これにはさらなる解釈が必要となる場合がある。

609　統合評価は、個々のコンポーネント評価の結果に基づいているため、統合 TOE 評価では、個々のコンポーネントがそれぞれ単独で評価される必要もある。統合 TOE 評価が開始されるときに、依存コンポーネントの評価が進行中でも構わない。ただし、統合 TOE 評価が完了する前に依存コンポーネント評価が完了する必要がある。

610　統合評価アクティビティは、依存コンポーネント評価と同時に行われる場合がある。これは次の 2 つの要因による:

- 経済/ビジネス上の要因 - 統合評価アクティビティには、依存コンポーネント評価からの評価用提供物件が必要であるため、依存コンポーネントの開発者は、統合評価アクティビティのスポンサーとなるか、またはそれらのアクティビティを支援することになる。

- 技術的な要因 - コンポーネントは、依存コンポーネントが最近コンポーネント評価を受けており(評価中であり)、かつ評価に関連するすべての評価用提供物件が入手可能であることを了解したうえで、基本コンポーネントから必要な保証が提供されているかどうかを考慮する(例えば、コンポーネント評価完了後の基本コンポーネントに対する変更を考慮)。したがって、統合中に依存コンポーネント評価アクティビティの再検証を要求するアクティビティは発生しない。また、依存コンポーネントの評価中に基本コンポーネントによって依存コンポーネントのテスト構成(の 1 つ)が作成され、ACO_CTT によってこの構成で基本コンポーネントが考慮されているか検証される。

611 依存コンポーネントの評価から得られた評価証拠は、統合 TOE 評価アクティビティに対して必要な入力である。統合 TOE 評価アクティビティに対して必要な入力である基本コンポーネントの評価から得られる唯一の評価資料は以下のとおりである:

- 基本コンポーネント評価で報告される、基本コンポーネントにおける残存脆弱性。これは、ACO_VUL アクティビティに必要となる。

612 基本コンポーネントのコンポーネント評価で得た評価結果は再使用されるべきであるため、統合 TOE 評価には、基本コンポーネントアクティビティで得られた他の評価証拠は必要ないようにすべきである。統合 TOE の TSF に、基本コンポーネントのコンポーネント評価中に TSF とみなされた部分よりも多くのものを含んでいる場合、基本コンポーネントの追加情報が必要になる場合がある。

613 基本コンポーネント及び依存コンポーネントのコンポーネント評価は、ACO コンポーネントに対して最終的な判定が行われる時点までに完了していると想定される。

614 ACO_VUL コンポーネントでは、最高でも拡張された基本的な攻撃能力を持つ攻撃者に対する抵抗力のみが考慮される。これは、基本コンポーネントが、依存コンポーネントが依存するサービスを ACO_DEV アクティビティの適用を通してどのように提供するかに関して提供できる設計情報のレベルに起因している。したがって、CAP を使用して統合 TOE 評価から得られる信頼は、EAL4 コンポーネント TOE 評価で得られる信頼と同様のレベルに制限される。ただし、統合 TOE を構成するコンポーネント内の保証は、EAL4 よりも高いレベルになる場合がある。

B.2 統合 TOE に対するセキュリティターゲット評価の実行

615 統合 TOE(評価コンポーネント+依存コンポーネント)の評価のために、開発者から ST が提出される。この ST は、統合 TOE に適用される保証パッケージを識別し、コンポーネント評価で得られた保証を利用することで統合エンティティでの保証を提供する。

616 ST 内でコンポーネントの統合を考慮する目的は、環境と要件の両方の観点からコンポーネントの両立性の有効性を確認することと、統合 TOE の ST がコンポーネント ST 及びそれらの ST で表現されているセキュリティ方針と一貫していることを評価することである。これには、コンポーネント ST 及びそれらの ST で表現されているセキュリティ方針が両立できることの判断も含まれる。

617 コンポーネントSTが統合TOEのSTでどのように表現されているかの根拠を提供しつつ、統合 TOE の ST はコンポーネント ST の内容を参照してもよいし、ST 作成者は統合 TOE の ST 内でコンポーネント ST の記述を繰り返してもより。

618 統合TOEのSTに対するASE_CCL評価アクティビティの実施中に、評価者は、統合TOEのSTでコンポーネントSTが正確に表現されていることを判断する。これは、統合 TOE の ST がコンポーネント TOE の ST と適合することが論証可能であることを決定することで達成される。また、評価者は、運用環境に対する依存コンポーネントの依存性が、統合 TOE で十分に満たされていることを決定する必要がある。

619 統合TOEの記述では、統合ソリューションが記述される。統合ソリューションの論理的及び物理的な範囲と境界が記述され、コンポーネント間の論理的な境界も識別される。この記述は、各コンポーネントから提供されるセキュリティ機能性を識別する。

620 統合TOEに対するSFRのステートメントは、SFRを満たすコンポーネントを識別する。SFRが両方のコンポーネントによって満たされる場合は、SFR の様々な側面を満たすコンポーネントが識別できるよう記述される。同様に、統合 TOE の要約仕様は、記述されているセキュリティ機能性を提供するコンポーネントを識別する。

621 統合 TOE の ST に適用される ASE: セキュリティターゲット評価要件のパッケージは、コンポーネント評価で使用される ASE: セキュリティターゲット評価要件のパッケージと一致しているべきである。

622 統合 TOE の ST がコンポーネント ST を直接参照している場合は、コンポーネント ST の評価で得た評価結果を再利用できる。例えば、統合 TOE の ST が SFR のそのステートメントの一部についてコンポーネント ST を参照している場合、評価者は、すべての割付操作と選択操作(ASE_REQ.*.3C で述べられているもの)の完了に対する要件が、コンポーネント評価で満たされていると推察できる。

B.3 統合 IT エンティティ間の相互作用

623 基本コンポーネントの TSF は、その統合での可能な適用における依存性に関する知識なしに定義されることが少なくない。この基本コンポーネントの TSF は、基本コンポーネント SFR を実施するために依存しなければならない基本コンポーネントのすべての部分を含むように定義される。これには、基本コンポーネントの SFR の実装に必要な基本コンポーネントのすべての部分が含まれる。

624 この基本コンポーネントのTSFIは、TSFのサービスを呼び出すためにSFRのステートメントで定義された外部エンティティに TSF が提供するインタフェースを表す。これには、人間の利用者とのインタフェースに加え、外部 IT エンティティとのインタフェースも含まれる。ただし、TSFIにはTSFに対するインタフェースのみが含まれるため、TSFIは必ずしも外部エンティティと基本コンポーネント間で利用可能なすべてのインタフェースを網羅するインタフェース仕様ではない。基本コンポーネントは、セキュリティ関連とみなされていなかったサービスに対するインタフェースを提示する場合がある。その理由は、サービス特有の目的(例えば、タイプフォントの調整)によるか、または関連する CC SFR が基本コンポーネントの ST で主張されていない(例えば、FIA: 識別認証 SFR が主張されていない場合のログインインタフェース)ことによる。

625 基本コンポーネントが提供する機能インタフェースは、セキュリティインタフェース(TSFI)に追加されるもので、基本コンポーネント評価で考慮することは要求されない。機能インタフェースには、基本コンポーネントが提供するサービスを依存コンポーネントが呼び出す場合に使用されるインタフェースが含まれることが多い。

626 基本コンポーネントには、TSFI をコールすることができる間接的なインタフェースも含まれるかもしれない。例えば、TSF のサービスの呼び出しに使用できる API があり、これは基本コンポーネントの評価中に考慮されていない。

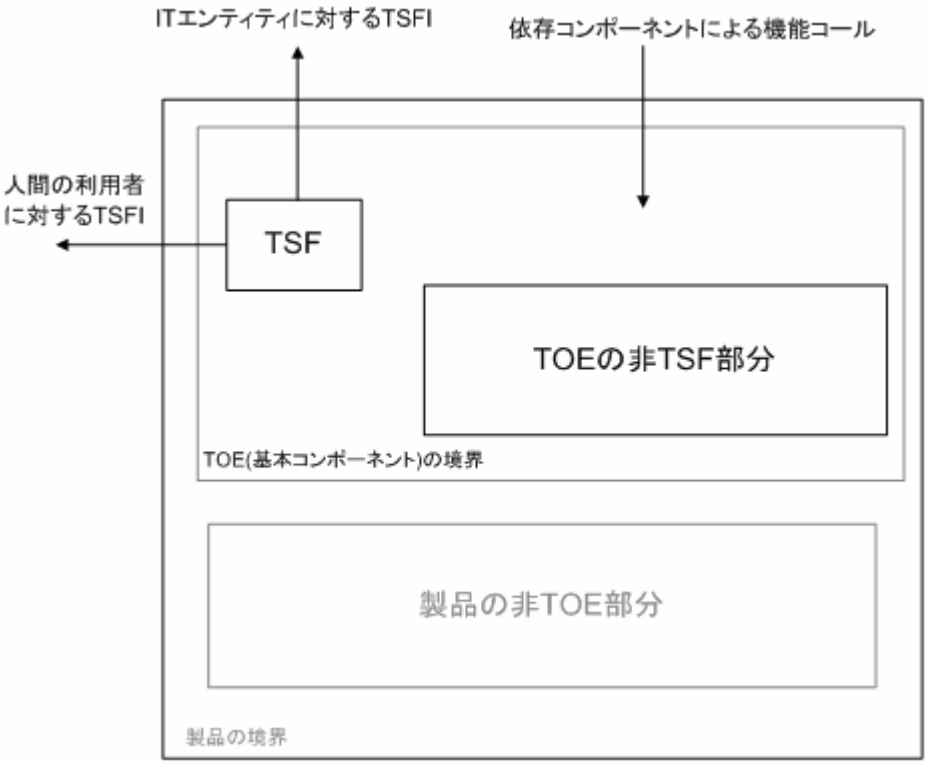

図 22　基本コンポーネントの抽象概念

627 基本コンポーネントに依存する依存コンポーネントも同様に定義される。コンポーネント ST の SFR で定義される外部エンティティへのインタフェースは、TSFI として分類され、ADV_FSP で検査される。

628 依存 TSF から SFR を支援する環境に対して行われるすべてのコールは、宣言された依存コンポーネント SFR の実施を満たすために依存 TSF が環境に対して何らかのサービスを要求することを示す。これらのサービスは、依存コンポーネントの境界外にあり、基本コンポーネントは、依存コンポーネント ST で外部エンティティとして定義される可能性はあまりない。このため、依存 TSF から下層のプラットフォーム(基本コンポーネント)に対して行われるコールは、機能仕様(ADV_FSP)アクティビティの一環として分析されることはない。基本コンポーネントにおけるこれらの依存性は依存コンポーネント ST において環境のセキュリティ対策方針として表現される。

629 この依存コンポーネントとインタフェースの抽象概念を、次の図 23 に示す。

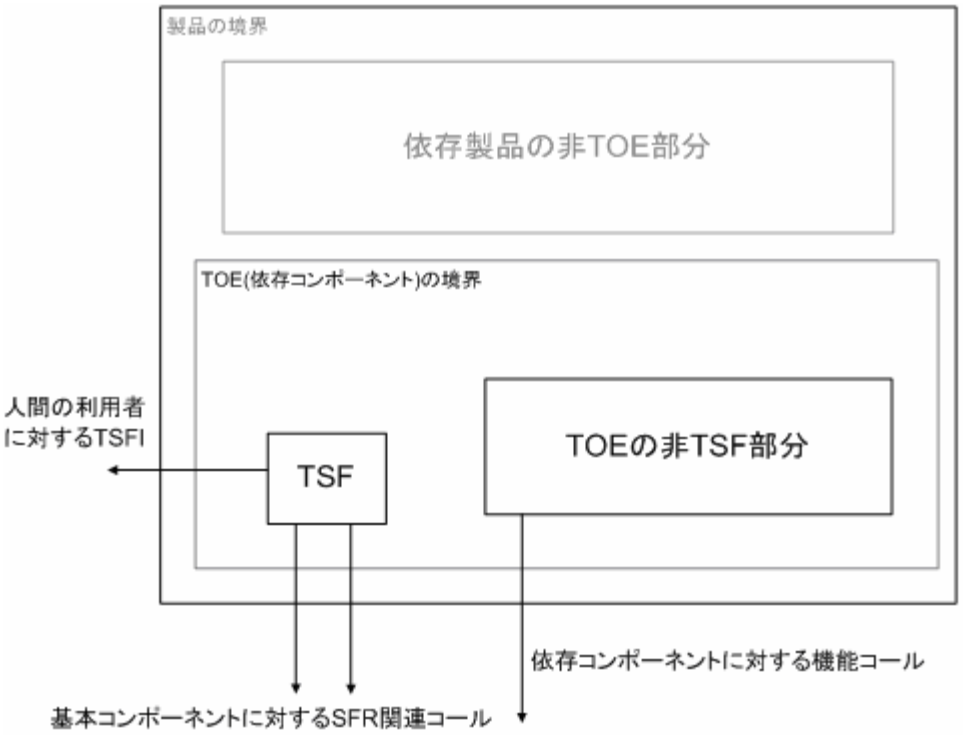

図 23　依存コンポーネントの抽象概念

630　基本コンポーネントと依存コンポーネントの統合を考慮する際に、依存コンポーネントの TSF が SFR の実装をサポートするために基本コンポーネントのサービスを要求する場合は、サービスに対するインタフェースを定義する必要がある。このサービスが基本コンポーネントの TSF によって提供される場合、そのインタフェースは基本コンポーネントの TSFI であるべきであるため、基本コンポーネントの機能仕様で既に定義されているであろう。

631　ただし、依存コンポーネントの TSF によって呼び出されるサービスが基本コンポーネントの TSF から提供されない(つまり、そのサービスが基本コンポーネントの非 TSF 部分で実装されるか、場合によっては基本コンポーネントの非 TOE 部分(図 24 には示されていない)で実装される)場合は、そのサービスが基本コンポーネントの TSF によって仲介されない限り、そのサービスに関連する基本コンポーネントの TSFI である可能性は低い。依存コンポーネントから運用環境へのこれらのサービスに対するインタフェースは、依存コンポーネントの依存(ACO_REL)ファミリで考慮される。

632　基本コンポーネントの非 TSF 部分は、依存コンポーネントの SFR を支援するために依存コンポーネントが基本コンポーネントに対して持つ依存性のために、統合 TOE の TSF に取り込まれることもある。したがってこのような場合は、統合 TOE の TSF が、コンポーネントの TSF を単純に合計したものよりも大きくなる。

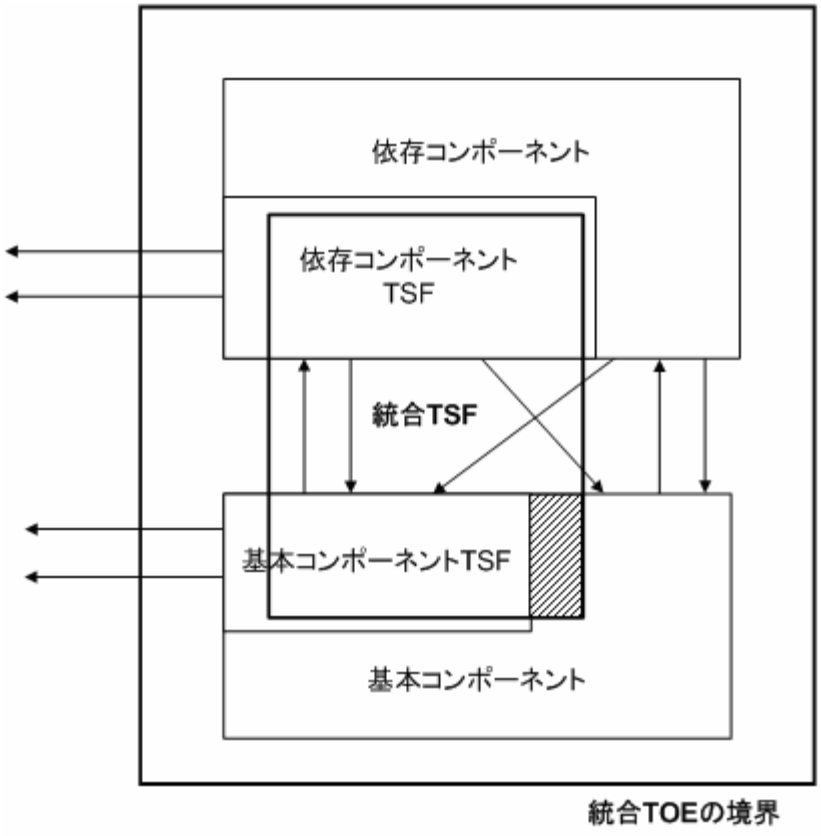

図 24　統合 TOE の抽象概念

633　基本コンポーネントの TSFI が、基本コンポーネント評価では予期されなかった方法で呼び出される場合がある。このため、基本コンポーネントの TSFI を追加でテストする必要が生じる。

634　想定されるインタフェースについては、次の図(図 25)及び補足説明で詳しく記述されている。

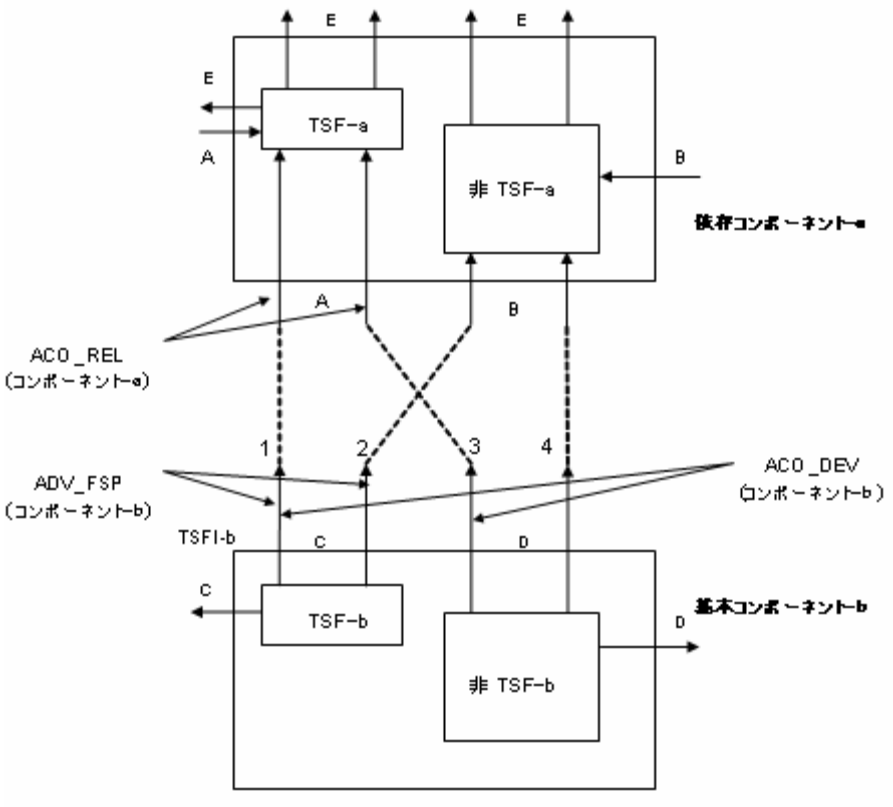

図 25　統合コンポーネントのインタフェース

- 「依存コンポーネント-a」に*向かう*矢印(A 及び B) = コンポーネントは環境がサービス要求に応じることを期待する(依存コンポーネントから環境に対するコールに応答する);
- 「基本コンポーネント-b」から*出ている*矢印(C 及び D) = 基本コンポーネントから環境に対して提供されるサービスのインタフェース;
- コンポーネント間の破線 = 対を成すインタフェース間の通信のタイプ;
- その他の(グレーの)矢印 = 特定の基準で記述されているインタフェース。

635　次に、単純化した例で考慮の必要な事項を説明する。

636　a (「依存コンポーネント-a」)及び b (「依存コンポーネント-b」)というコンポーネントがある。TSF-a から*出ている*矢印は、TSF-a が提供するサービスであるため、TSFI(a)とする。同様に、TSF-b から出ている矢印(「C」)は TSFI(b)とする。これらは、それぞれの機能仕様で詳しく説明されている。コンポーネント-a は、その環境からサービスを要求する。TSF(a)が必要とするサービスは「A」というラベルで示され、それ以外のサービス(TSF-a に関連しないもの)は「B」というラベルで示される。

637 コンポーネント-aがコンポーネント-bと結合されると、破線(対を成すインタフェース間の通信のタイプ)で示されるように、{コンポーネント-aが必要とするサービス}と{コンポーネント-bが提供するサービス}の組み合わせが 4 通り発生する。このいずれのセットも、特定の統合に存在する可能性がある:

- TSF-a が TSF-b の提供するサービスを必要とする(「A」が「C」に連結される)場合: これは直接的であり、「C」に関する詳細がコンポーネント-bのFSP内にある。この場合は、すべてのインタフェースがコンポーネント-a 及びコンポーネント-b の機能仕様で定義されているべきである。

- 非TSF-aがTSF-bの提供するサービスを必要とする(「B」が「C」に連結される)場合: これは直接的である(同じく「C」に関する詳細がコンポーネント-b の FSP 内にある)が、セキュリティ上は重要でない。

- 非 TSF-a が非 TSF-b の提供するサービスを必要とする(「B」が「D」に連結される)場合: D に関する詳細はわからないが、これらのインタフェースの使用はセキュリティに影響しないため、開発者にとってはこれらのインタフェースが統合上の課題になる可能性があるが、これらを評価で考慮する必要はない。

- TSF-aが非TSF-bの提供するサービスを必要とする(「A」が「D」に連結される)場合: これは、コンポーネント-a とコンポーネント-bで「セキュリティサービス」の意味するところが異なっている場合に生じる。例えば、コンポーネント-b は I&A に関する主張を行っていない(その ST に FIA SFR がない)が、コンポーネント-a はその環境が提供する認証を必要とする。「D」インタフェースに関する詳細はない(これらはTSFI(b)ではないため、コンポーネント-b の FSP 内にない)。

638 注意: 上述のケース c で説明されている種類の相互作用が存在する場合、統合 TOE の TSF は、TSF-a + TSF-b + 非TSF-Bとなる。存在しない場合、統合TOEのTSFはTSF-a + TSF-b となる

639 図 25 のインタフェースタイプ 2 及び 4 は、統合 TOE の評価に直接関連しない。インタフェース 1 及び 3 は、各種ファミリを適用する際に考慮される:

- 機能仕様(ADV_FSP) (コンポーネント-b 用)は、C インタフェースを記述する。

- 依存コンポーネントの依存(ACO_REL)は、A インタフェースを記述する。

- 開発証拠(ACO_DEV)は、連結タイプ 1 の C インタフェースと、連結タイプ 3 の D インタフェースを記述する。

640 統合が適用される典型的な例は、下層のオペレーティングシステム(OS)に依存しているデータベース管理システム(DBMS)である。DBMS コンポーネントの評価中は、(評価で使用されている保証コンポーネントによって指示されている厳格さの度合いまで)その DBMS のセキュリティ特性についての評定が実施される。例えば、TSF 境界が識別され、機能仕様が TSF によって提供されるセキュリティサービスに対するインタフェースを記述するかどうかが評定される。TSF に関する追加情報(設計、アーキテクチャ、内部構造)が提供されたり、TSF がテストされたり、そのライフサイクルの側面とガイダンス証拠資料が評定されたりする可能性もある。

641 ただし、DBMS 評価は、DBMS が OS に対して持つ依存性に関する証拠を要求しない。DBMS の ST は、前提条件の節で OS に関する前提条件を述べたり、環境の節で OS に対するセキュリティ対策方針を述べるケースが多い。DBMS の ST は、OS に対する SFR の観点から環境に対するそれらの対策方針の具体化までを行う場合もある。ただし、機能仕様、アーキテクチャ記述、またはDBMSのその他のADV証拠における詳細を反映するOS の仕様は存在しない。依存コンポーネントの依存(ACO_REL)は、その要件を満たす。

642 依存コンポーネントの依存(ACO_REL)は、サービスの提供のために基本コンポーネントにコールを行う依存 TOE のインタフェースを記述する。これらは、基本コンポーネントが応答するインタフェースである。インタフェース記述は、依存コンポーネントの観点から提供される。

643 開発証拠(ACO_DEV)は、基本コンポーネントより提供される依存コンポーネントのサービス要求に応答するインタフェースを記述する。これらのインタフェースは、依存情報で識別される関連する依存コンポーネントのインタフェースにマッピングされる(このマッピングの完全性(記述された基本コンポーネントのインタフェースがすべての依存コンポーネントインタフェースを表すかどうか)は、ここでは検証されないが統合の根拠(ACO_COR)で検証される)。ACO_DEV の上位レベルで、インタフェースを提供するサブシステムが記述される。

644 基本コンポーネントで記述されない依存コンポーネントが必要とするインタフェースはすべて、統合の根拠(ACO_COR)の根拠で報告される。根拠は、依存コンポーネントが依存している基本コンポーネントのインタフェースが、基本コンポーネント評価内で考慮されているかどうかについても報告する。基本コンポーネント評価で考慮されなかったすべてのインタフェースについて、基本コンポーネント TSF でそのインタフェースを使用した場合の影響についての根拠が提供される。

附属書C　保証コンポーネントの依存性の相互参照(参考)

645 10 章及び 11 章から 17 章のコンポーネントに記述されている依存性は、保証コンポーネント間の直接的な依存性である。

646 次の保証コンポーネントに対する依存性の表は、それぞれの直接的、間接的、あるいは自由選択の依存性を示す。ある保証コンポーネントが依存する個々のコンポーネントは、列に配置される。各保証コンポーネントは、行に配置される。表のセルにおける値は、列に書かれたコンポーネントが、行に書かれたコンポーネントによって、直接的に要求されるか(クロス「X」で表示)、間接的に要求されるか(ダッシュ「-」で表示)を示す。文字が表示されていない場合、そのコンポーネントは他のコンポーネントに依存しない。

	ACO_DEV.1	ACO_DEV.2	ACO_DEV.3	ACO_REL.1	ACO_REL.2	ALC_CMC.1	ALC_CMS.1
ACO_COR.1	X			X		X	-
ACO_CTT.1	X			X			
ACO_CTT.2		X		-	X		
ACO_DEV.1				X			
ACO_DEV.2				X			
ACO_DEV.3					X		
ACO_REL.1							
ACO_REL.2							
ACO_VUL.1	X			-			
ACO_VUL.2		X		-			
ACO_VUL.3			X		-		

表 14　ACO: 統合クラスの依存性の表

	ADV_FSP.1	ADV_FSP.2	ADV_FSP.3	ADV_FSP.4	ADV_FSP.5	ADV_FSP.6	ADV_IMP.1	ADV_TDS.1	ADV_TDS.3	ALC_CMC.5	ALC_CMS.1	ALC_DVS.2	ALC_LCD.1	ALC_TAT.1
ADV_ARC.1	X	-						X						
ADV_FSP.1														
ADV_FSP.2		-						X						
ADV_FSP.3		-						X						
ADV_FSP.4		-						X						
ADV_FSP.5		-		-			X	X	-					-
ADV_FSP.6		-						X						
ADV_IMP.1		-		-			-	-	X					X
ADV_IMP.2		-		-			-	-	X	X	-	-	-	X
ADV_INT.1		-		-			X	-	X					X
ADV_INT.2		-		-			X	-	X					X
ADV_INT.3		-		-			X	-	X					X
ADV_SPM.1		-		X				-						
ADV_TDS.1		X						-						
ADV_TDS.2		-	X					-						
ADV_TDS.3		-		X				-						
ADV_TDS.4		-		-	X		-	-	-					-
ADV_TDS.5		-		-	X		-	-	-					-
ADV_TDS.6		-				X		-						

表 15　ADV: 開発クラスの依存性の表

	ADV_FSP.1
AGD_OPE.1	X
AGD_PRE.1	

表 16　AGD: ガイダンス文書クラスの依存性の表

	ADV_FSP.2	ADV_FSP.4	ADV_IMP.1	ADV_TDS.1	ADV_TDS.3	ALC_CMS.1	ALC_DVS.1	ALC_DVS.2	ALC_LCD.1	ALC_TAT.1
ALC_CMC.1						X				
ALC_CMC.2						X				
ALC_CMC.3						X	X			
ALC_CMC.4						X	X		X	
ALC_CMC.5						X		X	X	
ALC_CMS.1										
ALC_CMS.2										
ALC_CMS.3										
ALC_CMS.4										
ALC_CMS.5										
ALC_DEL.1										
ALC_DVS.1										
ALC_DVS.2										
ALC_FLR.1										
ALC_FLR.2										
ALC_FLR.3										
ALC_LCD.1										
ALC_LCD.2										
ALC_TAT.1	-	-	X	-	-					-
ALC_TAT.2	-	-	X	-	-					-
ALC_TAT.3	-	-	X	-	-					-

表 17　ALC: ライフサイクルサポートクラスの依存性の表

	APE_ECD.1	APE_INT.1	APE_OBJ.2	APE_REQ.1	APE_SPD.1
APE_CCL.1	X	X		X	
APE_ECD.1					
APE_INT.1					
APE_OBJ.1					
APE_OBJ.2					X
APE_REQ.1	X				
APE_REQ.2	X		X		-
APE_SPD.1					

表 18　APE: プロテクションプロファイル評価クラスの依存性の表

	ASE_ECD.1	ASE_INT.1	ASE_OBJ.2	ASE_REQ.1	ASE_SPD.1
ASE_CCL.1	X	X		X	
ASE_ECD.1					
ASE_INT.1					
ASE_OBJ.1					
ASE_OBJ.2					X
ASE_REQ.1	X				
ASE_REQ.2	X		X		-
ASE_SPD.1					
ASE_TSS.1	-	X		X	
ASE_TSS.2	-	X		X	

表 19　ASE: セキュリティターゲット評価クラスの依存性の表

	ADV_ARC.1	ADV_FSP.1	ADV_FSP.2	ADV_FSP.3	ADV_FSP.4	ADV_FSP.5	ADV_IMP.1	ADV_TDS.1	ADV_TDS.2	ADV_TDS.3	ADV_TDS.4	AGD_OPE.1	AGD_PRE.1	ALC_TAT.1	ATE_COV.1	ATE_FUN.1
ATE_COV.1			X					-							-	X
ATE_COV.2			X					-							-	X
ATE_COV.3			X					-							-	X
ATE_DPT.1	X	-	-	-				-	X						-	X
ATE_DPT.2	X	-	-		-			-		X					-	X
ATE_DPT.3	X	-	-		-	-	-	-		-	X			-	-	X
ATE_DPT.4	X	-	-		-	-	X	-		-	X			-	-	X
ATE_FUN.1			-					-							X	-
ATE_FUN.2			-					-							X	-
ATE_IND.1		X										X	X			
ATE_IND.2		-	X					-				X	X		X	X
ATE_IND.3		-	-		X			-				X	X		X	X

表 20　ATE: テストクラスの依存性の表

	ADV_ARC.1	ADV_FSP.1	ADV_FSP.2	ADV_FSP.4	ADV_IMP.1	ADV_TDS.1	ADV_TDS.3	AGD_OPE.1	AGD_PRE.1	ALC_TAT.1
AVA_VAN.1		X						X	X	
AVA_VAN.2	X	X	-			X		X	X	
AVA_VAN.3	X	-	X	-	X	-	X	X	X	-
AVA_VAN.4	X	-	X	-	X	-	X	X	X	-
AVA_VAN.5	X	-	X	-	X	-	X	X	X	-

表 21　AVA: 脆弱性評定クラスの依存性の表

附属書D　PP と保証コンポーネントの相互参照(規定)

647　表 22 は、PP と、APE クラスのファミリとコンポーネントの間の関係を示している。

保証クラス	保証ファミリ	保証コンポーネント	
		低保証 PP	PP
プロテクション プロファイル評価	APE_CCL	**1**	1
	APE_ECD	**1**	1
	APE_INT	**1**	1
	APE_OBJ	**1**	**2**
	APE_REQ	**1**	**2**
	APE_SPD		**1**

表 22　PP 保証レベルの要約

附属書E　EAL と保証コンポーネントの相互参照(規定)

648　表 23 は、評価保証レベルと、保証クラス、ファミリ、及びコンポーネントとの関係を示す。

保証クラス	保証ファミリ	評価保証レベル別の保証コンポーネント						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
開発	ADV_ARC		**1**	1	1	1	1	1
	ADV_FSP	**1**	**2**	**3**	**4**	**5**	5	**6**
	ADV_IMP				**1**	1	**2**	2
	ADV_INT					**2**	**3**	3
	ADV_SPM						**1**	1
	ADV_TDS		**1**	**2**	**3**	**4**	**5**	**6**
ガイダンス文書	AGD_OPE	**1**	1	1	1	1	1	1
	AGD_PRE	**1**	1	1	1	1	1	1
ライフサイクルサポート	ALC_CMC	**1**	**2**	**3**	**4**	4	**5**	5
	ALC_CMS	**1**	**2**	**3**	**4**	**5**	5	5
	ALC_DEL		**1**	1	1	1	1	1
	ALC_DVS			**1**	1	1	**2**	2
	ALC_FLR							
	ALC_LCD			**1**	1	1	1	**2**
	ALC_TAT				**1**	**2**	**3**	3
セキュリティターゲット評価	ASE_CCL	**1**	1	1	1	1	1	1
	ASE_ECD	**1**	1	1	1	1	1	1
	ASE_INT	**1**	1	1	1	1	1	1
	ASE_OBJ	**1**	**2**	2	2	2	2	2
	ASE_REQ	**1**	**2**	2	2	2	2	2
	ASE_SPD		**1**	1	1	1	1	1
	ASE_TSS	**1**	1	1	1	1	1	1
テスト	ATE_COV		**1**	**2**	2	2	**3**	3
	ATE_DPT			**1**	**2**	**3**	3	**4**
	ATE_FUN		**1**	1	1	1	**2**	2
	ATE_IND	**1**	**2**	2	2	2	2	**3**
脆弱性評定	AVA_VAN	**1**	**2**	2	**3**	**4**	**5**	5

表 23　評価保証レベルの要約

附属書F　CAP と保証コンポーネントの相互参照(規定)

649　表 24 は、統合保証レベルと、保証クラス、ファミリ、及びコンポーネントとの関係を示す。

保証クラス	保証ファミリ	統合保証パッケージ別の保証コンポーネント		
		CAP-A	CAP-B	CAP-C
統合	ACO_COR	**1**	1	1
	ACO_CTT	**1**	**2**	2
	ACO_DEV	**1**	**2**	**3**
	ACO_REL	**1**	1	**2**
	ACO_VUL	**1**	**2**	**3**
ガイダンス文書	AGD_OPE	**1**	1	1
	AGD_PRE	**1**	1	1
ライフサイクルサポート	ALC_CMC	**1**	1	1
	ALC_CMS	**2**	2	2
	ALC_DEL			
	ALC_DVS			
	ALC_FLR			
	ALC_LCD			
	ALC_TAT			
セキュリティターゲット評価	ASE_CCL	**1**	1	1
	ASE_ECD	**1**	1	1
	ASE_INT	**1**	1	1
	ASE_OBJ	**1**	**2**	2
	ASE_REQ	**1**	**2**	2
	ASE_SPD		**1**	1
	ASE_TSS	**1**	1	1

表 24　統合保証レベルの要約