Common Criteria for Information Technology Security Evaluation

Part 1: Introduction and general model

June 2005

Version 3.0
Revision 2

CCMB-2005-07-001

Foreword

This version of the Common Criteria for Information Technology Security Evaluation (CC v3.0) is the first major revision since being published as CC v2.1 in 1999 and CC v2.2 in 2004.

CC v3.0 is released for public comment and aims to eliminate redundant evaluation activities; reduce/eliminate those activities that contributed little to the final assurance of a product; clarify CC terminology to reduce misunderstandings; restructure and refocus the evaluation activities to those areas where security assurance would truly be gained; and add new CC requirements if needed.

This revision 2 of the CC v3.0 includes all editorial updates as of the release date.

CC version 3.0 consists of the following parts:

– Part 1: Introduction and general model

– Part 2: Security functional components

– Part 3: Security assurance components

Trademarks:

– Microsoft is a registered trademark of Microsoft Corporation

– POSIX is a registered trademark of the IEEE

– UNIX is a registered trademark of The Open Group in the United States and other countries

– Windows is a registered trademark of Microsoft Corporation in the United States and other countries

Legal Notice:

The governmental organisations listed below contributed to the development of this version of the Common Criteria for Information Technology Security Evaluation. As the joint holders of the copyright in the Common Criteria for Information Technology Security Evaluation, version 3.0 *Parts 1 through 3 (called "CC* 3.0*"), they hereby grant non-exclusive license to ISO/IEC to use CC* 3.0 *in the continued development/maintenance of the ISO/IEC 15408 international standard. However, these governmental organisations retain the right to use, copy, distribute, translate or modify CC* 3.0 *as they see fit.*

Australia/New Zealand:	*The Defence Signals Directorate and the Government Communications Security Bureau respectively;*
Canada:	*Communications Security Establishment;*
France:	*Direction Centrale de la Sécurité des Systèmes d'Information;*
Germany:	*Bundesamt für Sicherheit in der Informationstechnik;*
Japan:	*Information Technology Promotion Agency*
Netherlands:	*Netherlands National Communications Security Agency;*
Spain:	*Ministerio de Administraciones Públicas and Centro Criptológico Nacional;*
United Kingdom:	*Communications-Electronics Security Group;*
United States:	*The National Security Agency and the National Institute of Standards and Technology.*

Table of Contents

Table of contents

List of figures

List of tables

1 Introduction

1 The CC permits comparability between the results of independent security evaluations. The CC does so by providing a common set of requirements for the security functionality of (collections of) IT products and for assurance measures applied to these IT products during a security evaluation. The evaluation process establishes a level of confidence that the security functionality of these products and the assurance measures applied to these IT products meet these requirements. The evaluation results may help consumers to determine whether these IT products fulfil their security needs.

2 The CC is useful as a guide for the development, evaluation and/or procurement of (collections of) products with IT security functionality.

3 The CC addresses protection of information from unauthorised disclosure, modification, or loss of use. The categories of protection relating to these three types of failure of security are commonly called confidentiality, integrity, and availability, respectively. The CC may also be applicable to aspects of IT security outside of these three. The CC is applicable to risks arising from human activities (malicious or otherwise) and to risks arising from non-human activities. Apart from IT security, the CC may be applied in other areas of IT, but makes no claim of competence in these areas.

4 The CC is applicable to IT security functionality implemented in hardware, firmware or software.

2 Scope

5 This multipart standard, the Common Criteria (CC), is meant to be used as the basis for evaluation of security properties of (collections of) IT products. By establishing such a common criteria base, the results of an IT security evaluation will be meaningful to a wider audience.

6 Certain topics, because they involve specialised techniques or because they are somewhat peripheral to IT security, are considered to be outside the scope of the CC. Some of these are identified below.

a) The CC does not contain security evaluation criteria pertaining to administrative security measures not related directly to the IT security functionality. However, it is recognised that significant security can often be achieved through or supported by administrative measures such as organisational, personnel, physical, and procedural controls.

b) The evaluation of technical physical aspects of IT security such as electromagnetic emanation control is not specifically covered, although many of the concepts addressed will be applicable to that area. In particular, the CC addresses some aspects of physical protection.

c) The CC addresses neither the evaluation methodology nor the administrative and legal framework under which the criteria may be applied by evaluation authorities. However, it is expected that the CC will be used for evaluation purposes in the context of such a framework and such a methodology.

d) The procedures for use of evaluation results in accreditation are outside the scope of the CC. Accreditation is the administrative process whereby authority is granted for the operation of an IT product (or collection thereof) in its full operational environment including all of its non-IT parts. The results of the evaluation process are an input to the accreditation process. However, as other techniques are more appropriate for the assessments of non-IT related properties and their relationship to the IT security parts, accreditors should make separate provisions for those aspects.

e) The subject of criteria for the assessment of the inherent qualities of cryptographic algorithms is not covered in the CC. Should independent assessment of mathematical properties of cryptography be required, the evaluation scheme under which the CC is applied must make provision for such assessments.

3 Normative references

7 The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

CEM Common Methodology for Information Technology Security Evaluation, Version 3.0, revision 2, June 2005.

ISO/IEC ISO/IEC Directives - Part 2: Rules for the structure and drafting of International Standards.

4 Terms and definitions

8 For the purposes of this document, the following terms and definitions apply.

9 This chapter 4 contains only those terms which are used in a specialised way throughout the CC. Some combinations of common terms used in the CC, while not meriting inclusion in this chapter 4, are explained for clarity in the context where they are used. Explanations of the use of terms and concepts used in a specialised way in CC Part 2 and CC Part 3 can be found in their respective “paradigm” sections.

10 **assets (in the development environment). —** entities that the developer of the Product places value upon.

11 **assets (in the operational environment) —** entities that the owner of the TOE places value upon.

12 **assignment —** the specification of an identified parameter in a component or requirement.

13 **assurance —** grounds for confidence that a TOE meets the TSP.

14 **attack potential —** a measure of the effort expended (or to be expended) in attacking a TOE, expressed in terms of an attacker's expertise, resources and motivation.

15 **augmentation —** the addition of one or more requirement(s) to a package.

16 **can —** within normative text, “can” indicates “statements of possibility and capability, whether material, physical or causal” (ISO/IEC).

17 **class —** a grouping of CC families that share a common focus.

18 **component (of the CC) —** the smallest selectable set of elements that may be used to base requirements on.

19 **component (of a TOE) —** a subset of a TOE that has a well-defined purpose. For simple TOEs, a component would be the same as a module; for more complex TOEs, a component would be a collection of modules, analogous to a subsystem; for very complex TOEs, a component would be a collection of subsystems.

20 **component TOE —** a certified TOE that is part of another TOE.

21 **dependency —** a relationship between components such that if a requirement based on the depending component is included in a PP, ST or package, a requirement based on the component that is depended upon must normally also be included in the PP, ST or package.

22 **development environment —** the environment in which the TOE is developed.

23 **element —** an indivisible statement of security need.

24 **evaluation —** assessment of a PP, an ST or a TOE, against defined criteria.

25 **evaluation assurance level (EAL) —** an assurance package, consisting of assurance requirements drawn from CC Part 3, representing a point on the CC predefined assurance scale.

26 **evaluation authority —** a body that implements the CC for a specific community by means of an evaluation scheme and thereby sets the standards and monitors the quality of evaluations conducted by bodies within that community.

27 **evaluation scheme —** the administrative and regulatory framework under which the CC is applied by an evaluation authority within a specific community.

28 **family —** a grouping of components that share a similar goal but may differ in emphasis or rigour.

29 **formal —** expressed in a restricted syntax language with defined semantics based on well-established mathematical concepts.

30 **guidance documentation —** guidance documentation describes the delivery and installation of the Product, and the operation, management and use of the TOE as these activities apply to the users, administrators, and integrators.

31 **informal —** expressed in natural language.

32 **informative —** informative text "provides additional information intended to assist the understanding or use of the document." (ISO/IEC).

33 **iteration —** the use of more than one requirement based on the same component.

34 **may —** within normative text, may indicates "a course of action permissible within the limits of the document" (ISO/IEC).

35 **normative —** normative text "describes the scope of the document, and sets out provisions." (ISO/IEC). Within normative text, the verbs "shall", "should", "may", and "can" have the ISO standard meanings described in this glossary and the verb "must" is not used. Unless explicitly labelled "informative", all CC text is normative.

36 **object —** a passive entity in the TOE upon which subjects perform operations.

37 **operation (on a subject) —** a specific type of action from a subject to an object.

38 **operation (on a component of the CC) —** modifying or repeating that component. Allowed operations on components are assignment, iteration, refinement and selection.

39 **operational environment —** the environment in which the TOE is operated.

40 **operation (of the TOE) —** usage of the TOE after delivery and preparation.

41 **organisational security policy (OSP) —** a set of security rules, procedures, practises, or guidelines imposed by an organisation.

42 **package —** a named set of either functional or assurance requirements (e.g. EAL 3).

43 **Product —** a set of software, firmware, hardware and/or guidance.

44 **protection profile (PP) —** an implementation-independent statement of security needs for a Product type.

45 **refinement —** the addition of details to a component.

46 **security attribute —** characteristics of subjects and/or objects that are used for the enforcement of the TSP.

47 **security objective —** a statement of intent to counter identified threats and/or satisfy identified organisation security policies and/or assumptions.

48 **security Target (ST) —** an implementation-dependent statement of security needs for a specific identified TOE.

49 **selection —** the specification of one or more items from a list in a component.

50 **semiformal —** expressed in a restricted syntax language with defined semantics.

51 **shall —** within normative text, “shall” indicates “requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.” (ISO/IEC).

52 **should —** within normative text, “should” indicates “that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required.” (ISO/IEC) The CC interprets 'not necessarily required' to mean that the choice of another possibility requires a justification of why the preferred option was not chosen.

53 **subject —** an active entity in the TOE that performs operations on objects.

54 **target of evaluation (TOE) —** a product that has been installed and is being operated according to its guidance.

55 **TOE Security Functionality (TSF) —** a set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.

56 **TSF Interface (TSFI) —** a means by which users supply data to and/or receive data from the TSF.

57 **TOE Security Policy (TSP) —** a description of the security properties of a TOE in the form of a set of SFRs in a PP or ST.

58 **user —** any entity (human user or machine user) outside the TOE that interacts with the TOE.

5 Symbols and abbreviated terms

59 The following abbreviations are used in one or more parts of the CC:

AES	Advanced Encryption Standard
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BIOS	Basic Input/Output System
CC	Common Criteria
CCRA	Arrangement on the Recognition of Common Criteria Certificates in the field of IT Security
DAC	Discretionary Access Control
EAL	Evaluation Assurance Level
GHz	Gigahertz
GUI	Graphical User Interface
IC	Integrated Circuit
IOCTL	Input Output Control
IP	Internet Protocol
IT	Information Technology
MB	Mega Byte
OS	Operating System
OSP	Organisational Security Policy
PC	Personal Computer
PCI	Peripheral Component Interconnect
PKI	Public Key Infrastructure
PP	Protection Profile
RAM	Random Access Memory
RPC	Remote Procedure Call
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
ST	Security Target
TCP	Transport Control Protocol
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
TSP	TOE Security Policy

UDP	User Datagram Protocol
VPN	Virtual Private Network

6 Overview

60 This chapter introduces the main concepts of the CC. It identifies the concepts "Product" and "TOE", the target audience of the CC, and the approach taken to present the material in the remainder of the CC.

6.1 Introduction

61 Information held by IT products is a critical resource that enables organisations to succeed in their mission. Additionally, individuals have a reasonable expectation that their personal information contained in IT products remains confidential, is available to them as needed, and is not modified. IT products should perform their functionality while exercising proper control of the information to ensure it is protected against hazards such as unwanted or unwarranted dissemination, alteration, or loss. The term IT security is used to cover prevention and mitigation of these and similar hazards.

62 Many consumers of IT products lack the knowledge, expertise or resources necessary to judge whether their confidence in the security of their IT products is appropriate, and they may not wish to rely solely on the assertions of the developers. These consumers may therefore choose to increase their confidence in the security measures of an IT product or collection of IT products by ordering an analysis of its security (i.e. a security evaluation).

63 The CC can be used to select the appropriate security functional requirements and it contains criteria for the evaluation of whether these security functional requirements have been met.

6.2 Products and TOEs

64 The previous sections used the term "IT product" or "collection of IT products" in many places. The CC uses the term Product (capitalised) to refer to an IT product, a part of an IT product, a set of IT products etc. Examples of Products include:

- A software application;
- An operating system;
- A software application in combination with an operating system;
- A software application in combination with an operating system and a workstation;
- An operating system in combination with a workstation;
- A smartcard integrated circuit;

– The cryptographic co-processor of a smartcard integrated circuit;

– A Local Area Network including all terminals, servers, network equipment and software;

– A database application excluding the remote client software normally associated with that database application;

65 Earlier versions of the CC used the term "System" to describe a specific instance of a Product installed in a specific environment. An example of this is the MinuteGap v18.5 Firewall at the Directorate of National Defence. As the same criteria that apply to Products apply to Systems as well, this version of the CC uses the term "Product" to denote both.

66 As far as the evaluation is concerned, the Product is viewed as a collection of software, firmware and/or hardware accompanied by guidance. The precise relation between the Product and any products is only important in one aspect: the evaluation of a Product containing only part of a product should not be misrepresented as the evaluation of the entire product.

67 Many Products can be installed and configured in many ways, leading to vastly different IT security behaviour of the Product. A typical CC evaluation will only look at a single configuration of the Product. This is called the Target of Evaluation (TOE). Any evaluation results pertain normally (see section 6.2.1 for exceptions) only to the TOE and not to other configurations of the Product.

68 For example: a typical software application Product consists of a CD-ROM and a manual (either paper or electronic) and is essentially passive: i.e. it does nothing with respect to IT security. By following the steps in the manual to install the Product, the Product is brought into a specific configuration and becomes the TOE. The TOE is active and does provide IT security. Note that in this case the workstation and/or the OS on which the Product is installed while becoming the TOE do not belong to the TOE: only those parts originally in the Product can become (part of) the TOE.

69 If, on the other hand, the Product consists of the application, the OS and the workstation, the TOE would also have consisted of the installed application and the installed OS and the workstation.

70 For some cases, such as a typical ATM card, no installation and configuration by the end-user is required, and in this case the Product and the TOE are identical.

6.2.1 Multiple TOE configurations

71 As the guidance provides the instructions for creating the TOE from the Product, if the guidance allows options in installing the Product, multiple TOEs might result. An example is where the guidance instructs that the Product should be installed in any new directory under */usr/exec* but does not specify the name of that new directory.

72 If the guidance allows too much leeway in this, this may lead to problems in the evaluation, as some of the TOE configurations may no longer meet the requirements. For this reason, the guidance normally does not allow much choice in the security-relevant installation options.

6.3 Target audience of the CC

73 There are three groups with a general interest in evaluation of the security properties of TOEs: consumers; developers; and evaluators. The criteria presented in this document have been structured to support the needs of all three groups. They are all considered to be the principal users of the CC. The three groups can benefit from the criteria as explained in the following paragraphs.

6.3.1 Consumers

74 The CC is written to ensure that evaluation fulfils the needs of the consumers as this is the fundamental purpose and justification for the evaluation process.

75 Consumers can use the results of evaluations to help decide whether a TOE fulfils their security needs. These security needs are typically identified as a result of both risk analysis and policy direction. Consumers can also use the evaluation results to compare different TOEs.

76 The CC gives consumers, especially in consumer groups and communities of interest, an implementation-independent structure termed the Protection Profile (PP) in which to express their special security requirements.

6.3.2 Developers

77 The CC is intended to support developers in preparing for and assisting in the evaluation of their TOEs and in identifying security requirements to be satisfied by those TOEs. These requirements are contained in an implementation-dependent construct termed the Security Target (ST). This ST may be based on one or more Protection Profiles (the security requirements from consumers as discussed earlier.

78 The CC can then be used to determine the responsibilities and actions to support evidence that is necessary to support the evaluation of the TOE against these requirements. It also defines the content and presentation of that evidence.

6.3.3 Evaluators

79 The CC contains criteria to be used by evaluators when forming judgements about the conformance of TOEs to their security requirements. The CC describes the set of general actions the evaluator is to carry out and the SFRs on which to perform these actions. Note that the CC does not specify procedures to be followed in carrying out those actions, but more information may be found in chapter 6.4 .

6.3.4 Others

80 While the CC is oriented towards specification and evaluation of the IT security properties of TOEs, it may also be useful as reference material to all parties with an interest in or responsibility for IT security. Some of the additional interest groups that can benefit from information contained in the CC are:

a) system custodians and system security officers responsible for determining and meeting organisational IT security policies and requirements;

b) auditors, both internal and external, responsible for assessing the adequacy of the security of an IT solution (which may consist of or contain a TOE);

c) security architects and designers responsible for the specification of security properties of Products;

d) accreditors responsible for accepting an IT solution for use within a particular environment;

e) sponsors of evaluation responsible for requesting and supporting an evaluation; and

f) evaluation authorities responsible for the management and oversight of IT security evaluation programmes.

6.3.5 The different parts of the CC

81 The CC is presented as a set of distinct but related parts as identified below. Terms used in the description of the parts are explained in chapter 7.

a) **Part 1, Introduction and general model** is the introduction to the CC. It defines the general concepts and principles of IT security evaluation and presents a general model of evaluation.

b) **Part 2, Security functional components** establishes a set of functional components that serve as standard templates upon which to base functional requirements for TOEs. CC Part 2 catalogues the set of functional components and organises them in families and classes.

c) **Part 3, Security assurance components** establishes a set of assurance components that serve as standard templates upon which to base assurance requirements for TOEs. CC Part 3 catalogues the set of assurance components and organises them into families and classes. CC Part 3 also defines evaluation criteria for PPs and STs and presents seven pre-defined assurance packages which are called the Evaluation Assurance Levels (EALs).

82 In support of the three parts of the CC listed above, other documents have been published, most notably the CEM. It is anticipated that other documents

will be published, including technical rationale material and guidance documents.

83 The following table presents, for the three key target audience groupings, how the parts of the CC will be of interest.

	Consumers	Developers	Evaluators
Part 1	Use for background information and reference purposes. Guidance structure for PPs.	Use for background information and reference purposes. Development of security specifications for TOEs.	Use for background information and reference purposes. Guidance structure for PPs and STs.
Part 2	Use for guidance and reference when formulating statements of requirements for a TOE.	Use for reference when interpreting statements of functional requirements and formulating functional specifications for TOEs.	Use as mandatory statement of evaluation criteria when determining whether a TOE meets claimed security functional requirements.
Part 3	Use for guidance when determining required levels of assurance.	Use for reference when interpreting statements of assurance requirements and determining assurance approaches of TOEs.	Use as mandatory statement of evaluation criteria when determining the assurance of TOEs and when evaluating PPs and STs.

Table 1 Road map to the Common Criteria

6.4 Evaluation context

84 In order to achieve greater comparability between evaluation results, evaluations should be performed within the framework of an authoritative evaluation scheme that sets the standards, monitors the quality of the evaluations and administers the regulations to which the evaluation facilities and evaluators must conform.

85 The CC does not state requirements for the regulatory framework. However, consistency between the regulatory frameworks of different evaluation authorities will be necessary to achieve the goal of mutual recognition of the results of such evaluations.

86 An example of such a regulatory framework is the CCRA (Arrangement on the Recognition of the CC Certificates in the field of IT Security). This arrangement has been executed among a number of evaluation authorities in

different countries and provides the conditions for mutual recognition of CC certificates between these evaluation authorities.

87 Use of a common evaluation methodology contributes to the repeatability and objectivity of the results but is not by itself sufficient. Many of the evaluation criteria require the application of expert judgement and background knowledge for which consistency is more difficult to achieve. In order to enhance the consistency of the evaluation findings, the final evaluation results could be submitted to a certification process. An example of such a methodology is the CEM.

88 The certification process is the independent inspection of the results of the evaluation leading to the production of the final certificate or approval. The certificate is normally publicly available. It is noted that the certification process is a means of gaining greater consistency in the application of IT security criteria.

89 The evaluation scheme, methodology, and certification processes are the responsibility of the evaluation authorities that run evaluation schemes and are outside the scope of the CC.

7 General model

90 This chapter presents the general concepts used throughout the CC, including the context in which the concepts are to be used and the CC approach for applying the concepts. CC Part 2 and CC Part 3 expand on the use of these concepts and assume that the approach described is used. This chapter assumes some knowledge of IT security and does not propose to act as a tutorial in this area.

91 The CC discusses security using a set of security concepts and terminology. An understanding of these concepts and the terminology is a prerequisite to the effective use of the CC. However, the concepts themselves are quite general and are not intended to restrict the class of IT security problems to which the CC is applicable.

7.1 Security in the operational environment

92 Security is concerned with the protection of assets. Examples of assets include:

– contents of a file or a server;

– number of votes cast (in an election);

– an electronic commerce process.

93 The environment(s) in which these assets are located is called the operational environment. Examples of (aspects of) operational environments are:

– the computer room of a bank;

– connected to the Internet;

– a LAN;

– a general office environment.

94 Many assets in the operational environment are in the form of information that is stored, processed and transmitted by IT products to meet requirements laid down by owners of the information. Information owners may require that dissemination and modification of any such information is strictly controlled and that the assets are protected from threats by countermeasures. Figure 1 illustrates these high level concepts and relationships.

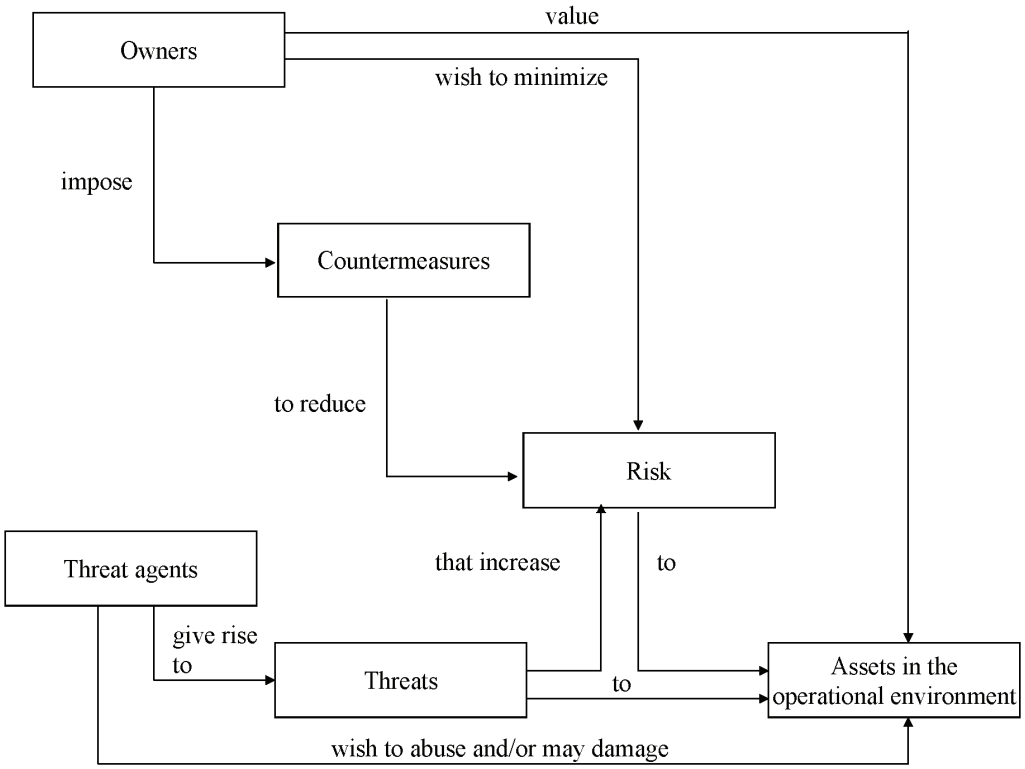

Figure 1 - Security concepts and relationships

95 Safeguarding assets of interest is the responsibility of owners who place value on those assets. Actual or presumed threat agents may also place value on the assets and seek to abuse assets in a manner contrary to the interests of the owner. Examples of threat agents include hackers, regular human users, computer processes, and accidents.

96 The owners of the assets will perceive such threats as potential for impairment of the assets such that the value of the assets to the owners would be reduced. Security specific impairment commonly includes, but is not limited to, damaging disclosure of the asset to unauthorised recipients (loss of confidentiality), damage to the asset through unauthorised modification (loss of integrity), or unauthorised deprivation of access to the asset (loss of availability).

97 The owners of the assets will analyse the possible threats to determine which ones apply to their operational environment. The results are known as risks.

98 Subsequently countermeasures are imposed to reduce risks to assets and to meet operational security policies of the owners of the assets in the operational environment (either directly or indirectly by providing direction to other parties). These countermeasures consist of IT countermeasures and non-IT countermeasures.

7.1.1 Evaluation concepts

99 Owners of assets may be (held) responsible for those assets and therefore should be able to defend the decision to accept the risks of exposing the assets to the threats. One of the methods of defending this decision is using an evaluated TOE to implement a part of the IT-countermeasures.

100 For this approach to be useful, the asset owner must be able to demonstrate that his TOE is:

a) sufficient: the TOE does its assigned part (in conjunction with the other countermeasures in the operational environment) in countering the threats to assets in the operational environment;

b) correct: the TOE contains no exploitable vulnerabilities whose exploitation might affect its operation, and thereby prohibit it from countering the threats.

101 Owners of assets may not themselves possess the capability to judge sufficiency and correctness of the TOE, and may therefore seek evaluation of the TOE.

102 The outcome of an evaluation is a statement about the extent to which assurance is gained that the TOE can be trusted to reduce the risks to the protected assets and does not itself possess exploitable vulnerabilities. The statement assigns an assurance rating to the TOE, assurance being that property of a TOE that gives grounds for confidence in its proper operation. This statement can be used by the owner of the assets in deciding whether to accept the risk of exposing the assets to the threats. Figure 2 illustrates these relationships. This mandates that evaluation leads to objective and repeatable results that are defensible and can be cited as evidence.

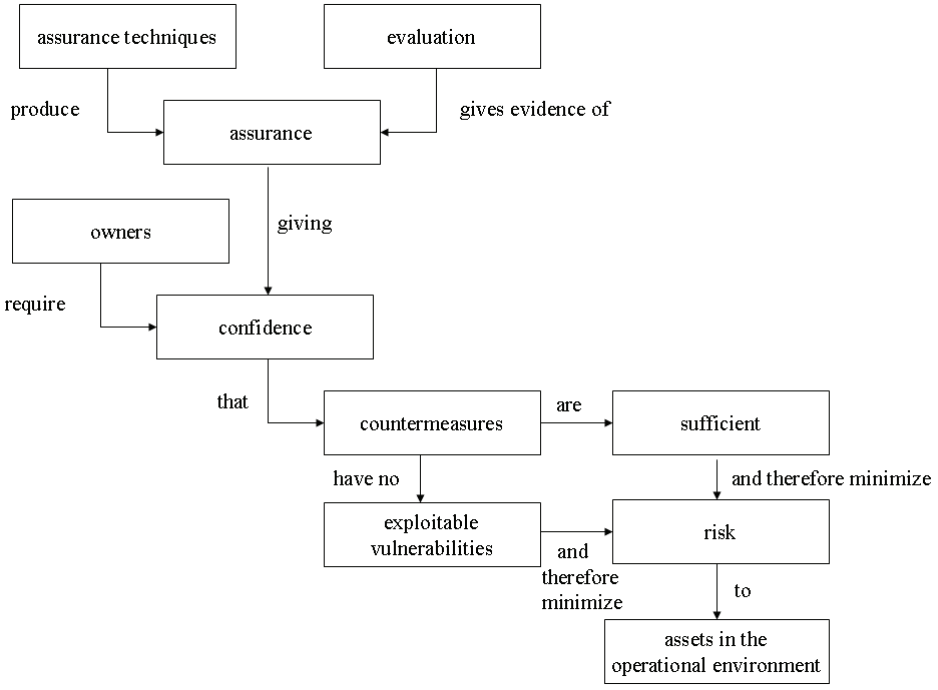

Figure 2 - Evaluation concepts and relationships

7.1.2 Sufficiency of the TOE

103 Sufficiency of the TOE is analysed through a construct called the Security Target. In this section a simplified view on this construct is provided: a more detailed and complete description may be found in Annex A.

104 In the Security Target the security problem to be solved is defined in terms of:

– the assets and the threats to those assets;

– any organisational security policies that are applicable;

– any assumptions that can be made on the operational environment.

105 The Security Target then describes a solution to this problem that is divided into two partwise solutions, and demonstrates that the combination of these partwise solutions is sufficient to solve the security problem:

– The partwise solution to be provided by the TOE;

– The partwise solution to be provided by the operational environment.

106 Finally, the Security Target provides a structured description of the partwise solution to be provided by the TOE in the form of the Security Functional Requirements (SFRs). These SFRs must be presented in a structured form to ensure exactness and facilitate comparability. The collection of SFRs is

called the TOE Security Policy (TSP). The concept of SFRs is described in more detail in Annex C.

107 A TOE meeting the TSP (in combination with the partwise solution to be provided by the operational environment) is sufficient, if it is correctly designed and implemented. This correctness arises from the development environment and is discusses in the next chapter.

7.2 Security in the development environment

108 A TOE is generally not correctly designed and implemented: it may contain errors that lead to vulnerabilities. By exploiting these vulnerabilities, attackers may still damage and/or abuse the assets, despite the TOE being theoretically sufficient.

109 These vulnerabilities arise from the development environment of the TOE: the environment or environments in which the TOE is designed, developed, produced, and delivered. Problems in the development environment, such as accidental errors made during development, or the intentional addition of malicious code, may lead to TOEs with exploitable vulnerabilities. The development environment therefore also has assets, such as design documents and source code. Similarly, the development environment has threat agents, such as cleaners, accidents, and development staff. These threats can cause vulnerabilities to appear in the TOE. This is depicted in Figure 3.

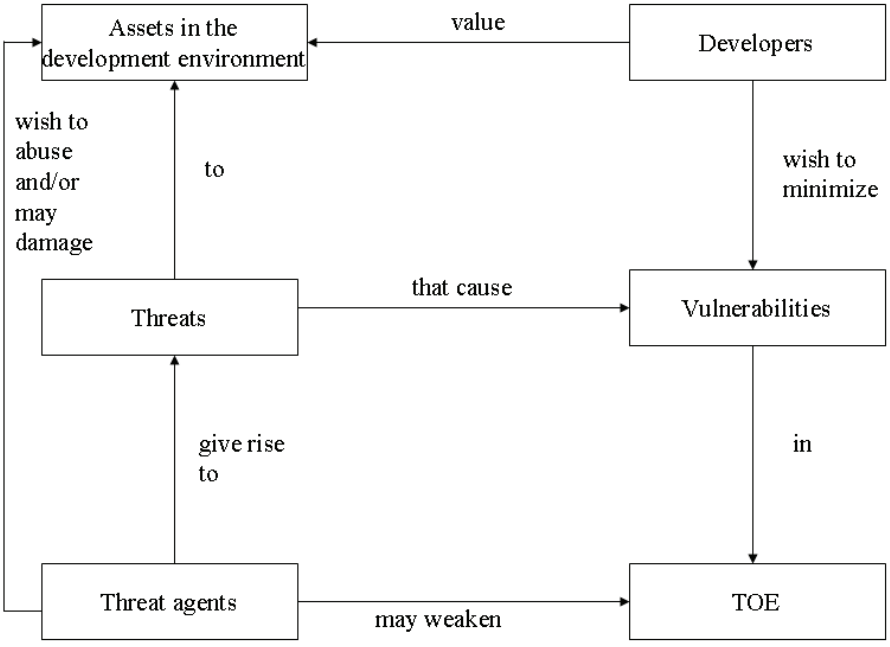

Figure 3 - Developer concepts and relationships

110 The CC uses a similar approach for correctness as it uses for sufficiency (once again, for details and completeness see Annex A).

111 In a Security Target a security problem to be solved is defined by listing:

- the assets in the development environment and the threats to those assets;
- any OSPs that apply to the development environment.

112 The Security Target also provides a structured description of the solution to this problem in the form of Security Assurance Requirements (SARs). These SARs must be presented in a structured form to ensure exactness and facilitate comparability.

113 If the SARs are all met, there exists sufficient assurance in the correct implementation of the TOE, and that any vulnerabilities that it still might have (residual vulnerabilities) are too difficult, too time-consuming and/or too expensive to exploit for an attacker.

7.3 Evaluation

114 In the CC an evaluation proceeds in two steps:

a) An ST evaluation: where it is determined whether the ST is complete, consistent, and technically sound and hence suitable for use as the basis for the corresponding TOE evaluation;

b) A TOE evaluation: where it is determined whether the TOE meets the TSP in the ST and whether the development environment of the TOE meets the SARs as specified in the ST.

115 An exception to this is a PP evaluation, which is described in more detail in Annex B

116 The ST evaluation is carried out by applying the ASE criteria (which are defined in CC Part 3) to the Security Target.

117 The TOE evaluation is more complex. The principal inputs to a TOE evaluation are:

a) the set of evaluation evidence, which includes the TOE and ST, but will usually also include input from the development environment, such as design documents or developer test results;

b) the evaluation methodology and scheme.

118 The TOE evaluation then consists of applying the SARs (from the Security Target) to the evaluation evidence (which includes the TOE). The precise method to apply a specific SAR is determined by the Evaluation Methodology that is used.

119 The TOE evaluation may be carried out after TOE development has finished, or in parallel with TOE development.

120 How the results of applying the SARs are documented, and what reports need to be generated and in what detail is determined by both the Evaluation Methodology that is used and the Evaluation Scheme under which the evaluation is carried out. The reports may be useful to actual and potential consumers as well as to the developer.

121 The result of the TOE evaluation process is either:

– A statement that not all SARs have been met and that therefore there is not enough assurance that the TOE meets the TSP as stated in the ST;

– A statement that all SARs have been met, and that therefore there is enough assurance that the TOE meets the TSP as stated in the ST.

122 The degree of assurance gained through an evaluation therefore depends on the SARs that were used. The formal method of stating evaluation results is described in Chapter 9. These results are also stated in terms of Protection Profiles and packages, and these constructs are described in the next chapter.

8 Protection Profiles and Packages

8.1 Introduction

123 To allow consumer groups and communities of interest to express their security needs, and to facilitate writing Security Targets, the CC provides two special constructs: packages and Protection Profiles. In the following two sections these constructs are described in more detail, followed by a section on how these constructs can be used.

8.2 Packages

124 A package is a named set of security requirements. A package can be a functional package and contain only SFRs. A package can be an assurance package and contain only SARs. Mixed packages containing both SFRs and SARs are not allowed.

125 A package can be defined by any party and is intended to be reusable. To this goal it should contain requirements that are useful and effective in combination.

126 Examples of assurance packages are the evaluation assurance levels (EALs) that are defined in CC Part 3.

127 Packages are used in the construction of larger packages, PPs and STs.

8.3 Protection Profiles

128 A Protection Profile is a template for a Security Target. Whereas a Security Target always describes a specific TOE (e.g. the MinuteGap v18.5 Firewall) a Protection Profile is intended to describe a TOE type (e.g. firewalls). A detailed description of Protection Profiles is given in Annex B.

129 In general a Security Target describes requirements for a TOE and is written by the developer of that TOE, while a Protection Profile describes the general requirement for a TOE type. A PP is therefore typically written by:

- A user community seeking to come to a consensus on the requirements for a given TOE type;
- A group of developers of similar TOEs wishing to establish a minimum baseline for that type of TOE;
- A government or large corporation specifying its requirements as part of its acquisition process.

130 Protection Profiles can be evaluated (by applying the APE criteria to them as listed in CC Part 3). The goal of such an evaluation is to demonstrate that the PP is complete, consistent, and technically sound and suitable for use as a template to build an ST on.

131 If an ST claims to be compliant with one or more packages and/or Protection Profiles, the evaluation of that ST will (among other properties of that ST) demonstrate that the ST actually complies with these packages and/or Protection Profiles that they claim compliance to. Details of this determination of compliance can be found in Annex A.

132 This allows the following process:

a) An organisation seeking to acquire a particular type of IT security product develops their security needs into a Protection Profile, then has this evaluated and publishes it;

b) A developer takes this Protection Profile, writes a Security Target that claims compliance with it and has this Security Target evaluated;

c) The developer then builds a TOE (or uses an existing one) and has this evaluated against the Security Target.

133 The result is that the developer can prove that his TOE is compliant with the security needs of the organisation: the organisation can therefore buy his TOE. A similar line of reasoning applies to packages.

8.4 Using Multiple Protection Profiles

134 The CC also allows PPs to comply with other PPs, allowing chains of PPs to be constructed, each based on the previous one(s).

135 For instance, one could take a PP for an Integrated Circuit and a PP for a Smart Card OS, and use these to construct a Smart Card PP (IC and OS). One could then combine this with a PP on Applet Loading and use this to write a PP on Smart Cards for Public Transport. Finally, a developer could then construct a ST based on this Public Transport PP.

9 Evaluation results

9.1 Introduction

136 This chapter presents the expected results from PP and ST/TOE evaluations.

– PP evaluations lead to catalogues of evaluated PPs.

– ST/TOE evaluations lead to catalogues of evaluated TOEs. In many cases these catalogues will refer to the Products that the TOEs are derived from rather than the specific TOE. Therefore, the existence of a Product in a catalogue should not be construed as that the whole Product has been evaluated: the extent of the ST/TOE evaluation is defined by the ST.

– An ST evaluation leads to intermediate results that are used in the frame of a TOE evaluation.

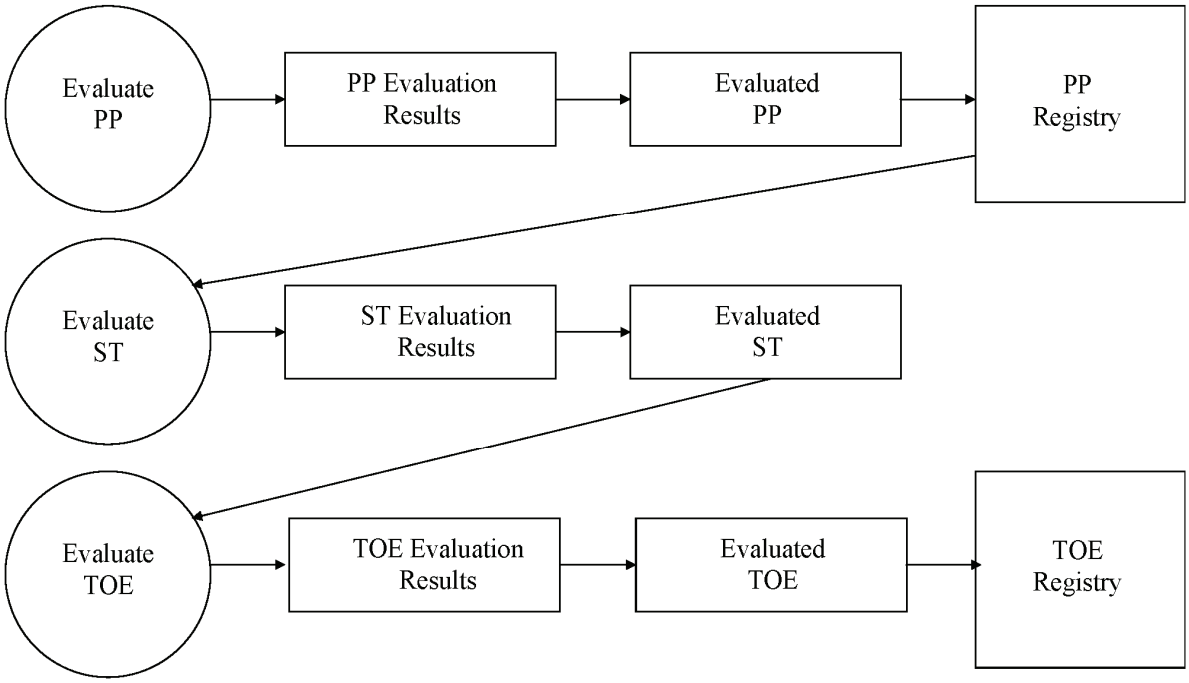

Figure 4 - Evaluation results

137 Evaluation should lead to objective and repeatable results that can be cited as evidence, even if there is no totally objective scale for representing the results of an security evaluation. The existence of a set of evaluation criteria is a necessary pre-condition for evaluation to lead to a meaningful result and provides a technical basis for mutual recognition of evaluation results between evaluation authorities. As the application of criteria contains both objective and subjective elements, precise and universal ratings for IT security are infeasible.

138 A rating made relative to the CC represents the findings of a specific type of investigation of the security properties of a TOE. Such a rating does not automatically guarantee fitness for use in any particular application environment. The decision to accept a TOE for use in a specific application environment is based on consideration of many security issues including the evaluation findings.

9.2 Results of a PP evaluation

139 The CC contains the evaluation criteria that permit an evaluator to state whether a PP is complete, consistent, and technically sound and hence suitable for use as a template for an ST.

140 Evaluation of the PP shall result in a pass/fail statement. A PP for which the evaluation results in a pass statement shall be eligible for inclusion within a registry. The results of the evaluation shall also include a "Conformance Claim" (see chapter 9.4).

9.3 Results of an ST/TOE evaluation

141 The CC contains the evaluation criteria that permit an evaluator to determine whether sufficient assurance exists that the TOE satisfies the TSP expressed in the ST. The result of the TOE evaluation shall be a statement that describes the extent to which the TOE can be trusted to conform to the requirements.

142 Evaluation of the TOE shall result in a pass/fail statement for a given ST. If both the ST and the TOE evaluation have resulted in a pass statement, the underlying Product shall be eligible for inclusion in a registry. The results of evaluation shall also include a "Conformance Claim" as defined in the next section.

9.4 Conformance claim

143 The conformance claim indicates the source of the collection of requirements that is met by a TOE or PP that passes its evaluation. This conformance claim contains a CC conformance claim that:

a) describes to which version of the CC the TOE or PP claims conformance.

b) describes the conformance to CC Part 2 (security functional requirements) as either:

- **CC Part 2 conformant** - A PP or TOE is CC Part 2 conformant if all SFRs are based only upon functional components in CC Part 2, or

- **CC Part 2 extended** - A PP or TOE is CC Part 2 extended if at least one SFR is not based upon functional components in CC Part 2.

c) describes the conformance to CC Part 3 (security assurance requirements) as either:

– **CC Part 3 conformant** - A PP or TOE is CC Part 3 conformant if all SARs are based only upon assurance components in CC Part 3, or

– **CC Part 3 extended** - A PP or TOE is CC Part 3 extended if at least one SAR is not based upon assurance components in CC Part 3.

144 Additionally, the conformance claim may include a statement made with respect to sets of defined requirements, in which case it consists of one of the following:

– *Package name Conformant* - A PP or TOE is conformant to a pre-defined package (e.g. EAL) if the SFRs and SARs of the PP or the ST of that TOE include all components in the packages listed as part of the conformance result.

– *Package name Augmented* - A PP or TOE is an augmentation of a pre-defined named functional and/or assurance package (e.g. EAL) if the SFRs or SARs are a proper superset of all components in the packages listed as part of the conformance result.

145 Finally, the conformance claim may also include two statements with respect to Protection Profiles:

a) *PP Conformant* - A PP or TOE meets specific PP(s), which are listed as part of the conformance result.

b) *Conformance Statement* (Only for PPs) - This statement describes the manner in which PPs or STs must conform to this PP: exact, strict or demonstrable. For more information on this Conformance Statement, see Annex A.

9.5 Use of TOE evaluation results

146 Once a TOE has been evaluated, consumers can have the assurance as defined in the ST that the TOE will meet the SFRs as defined in the ST.

147 However, the consumer should carefully check whether:

– the Security Problem Definition in the Security Target matches the security problem of the consumer;

– the Operational Environment of the consumer complies (or can be made to comply) with the Operational Environment described in the Security Target.

148 If either of these is not the case, it may not be possible to use the TOE. A possible approach is to adapt the Security Target and perform a re-evaluation.

149 Additionally, once a TOE is in operation, it is possible that previously unknown errors or vulnerabilities may surface. As a result of operation, feedback could be given that would require the developer to correct the TOE or redefine the Security Target. Such changes require the TOE and/or ST to be re-evaluated. In some instances this may only require that the needed updates are evaluated in order to regain confidence in the TOE.

150 The CC can be used for re-evaluation, including reuse of evaluation results, but discussion of the detailed procedures for doing so are outside the scope of this document.

A Specification of Security Targets (normative)

A.1 Goal and structure of this Annex

151 The goal of this annex is to explain the ASE criteria and provide examples of their application. This annex does not define the ASE criteria; this definition can be found in CC Part 3.

152 This annex consists of three major parts:

a) *What an ST must contain*. This is summarised in Section A.2, and described in more detail in Sections A.4 - A.10. These sections describe the mandatory contents of the ST, the interrelationships between these contents, and provide examples.

b) *How an ST should be used*. This is summarised in Section A.3, and described in more detail in Section A.11. These sections describe how an ST should be used, and some of the questions that can be answered with an ST.

c) *Low Assurance STs*. Low Assurance STs are STs with reduced content. They are described in detail in Section A.12.

A.2 Mandatory contents of an ST

153 Figure 5 portrays the mandatory contents of an ST. Figure 5 may also be used as a structural outline of the ST, though alternative structures are possible. For instance, if the security requirements rationale is particularly bulky, it could be included in an appendix of the ST instead of in the security requirements section. The separate sections of an ST and the contents of those sections are briefly summarised below and described in much more detail in Sections A.4 to A.10. An ST normally must contain:

a) an *ST introduction* containing three narrative descriptions of the TOE on different levels of abstraction;

b) a *conformance claim*, showing whether the ST claims conformance to any PPs and/or packages, and if so, to which PPs and/or packages;

c) a *security problem definition*, showing the threats, OSPs and assumptions that must be countered, enforced and upheld by the TOE and its operational environment;

d) *security objectives*, showing how the solution to the security problem is divided between:

– the TOE;

- the development environment of the TOE;
- the operational environment of the TOE;

e) *extended components definition*, where new components (i.e. not included in CC Part 2 or CC Part 3) may be defined. These new components can then be used to define extended functional and extended assurance requirements.

f) *security requirements*, where a well-defined translation of the security objectives for the TOE and the security objectives for the development environment is provided. This well-defined translation is in the form of SFRs (CC Part 2 requirements and extended functional requirements) and SARs (CC Part 3 requirements and extended assurance requirements);

g) a *TOE summary specification*, showing how the SFRs are implemented in the TOE.

154 There also exists low assurance STs which have reduced contents; these are described in detail in Section A.12. The rest of this Annex assumes that an ST with full contents is used.

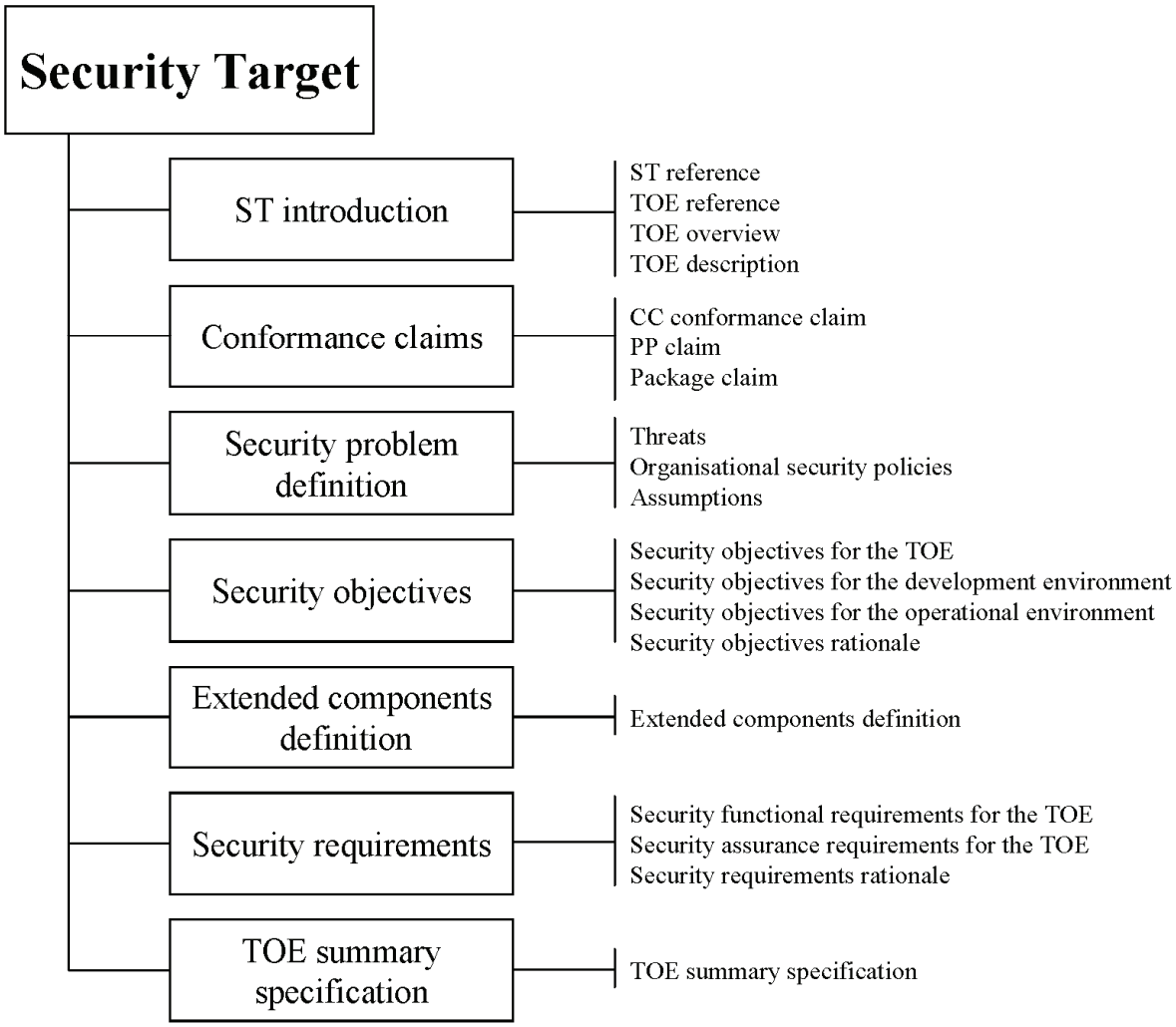

Figure 5 - Security Target contents

A.3 Using the ST (informative)

A.3.1 How an ST should be used

155 A typical ST fulfils two roles:

– Before and during the evaluation, the ST specifies "what is to be evaluated". In this role, the ST serves as a basis for agreement between the developer and the evaluator on the exact security properties of the TOE and the exact scope of the evaluation. Technical correctness and completeness are major issues for this role. Section A.7 describes how the ST should be used in this role.

– After the evaluation, the ST specifies "what was evaluated". In this role, the ST serves as a basis for agreement between the developer or re-seller of the TOE and the potential consumer of the TOE. The ST describes the exact security properties of the TOE in an abstract manner, and the potential consumer can rely on this description because the TOE has been evaluated to meet the ST. Ease of use and

understandability are major issues for this role. Section A.11 describes how the ST should be used in this role.

A.3.2 How an ST should not be used

156 Two roles (among many) that an ST should not fulfil are:

– *a detailed specification*: An ST is designed to be a security specification on a relatively high level of abstraction. An ST should, in general, not contain detailed protocol specifications, detailed descriptions of algorithms and/or mechanisms, long description of detailed operations etc.

– *a complete specification*: An ST is designed to be a security specification and not a general specification. Unless security-relevant, properties such as interoperability, physical size and weight, required voltage etc. should not be part of an ST. This means that in general an ST may be a part of a complete specification, but is not a complete specification in itself.

A.4 ST Introduction (ASE_INT)

157 The ST introduction describes the TOE in a narrative way on three levels of abstraction:

a) the ST reference and the TOE reference, which provide identification material for the ST and the TOE that the ST refers to;

b) the TOE overview, which briefly describes the TOE;

c) the TOE description, which describes the TOE in more detail.

A.4.1 ST reference and TOE reference

158 An ST contains a clear ST reference that identifies that particular ST. A typical ST reference consists of title, version, authors and publication date. An example of an ST reference is “MauveRAM Database ST, version 1.3, MauveCorp Specification Team, 11 October 2002”. The reference must be unique so that it is possible to tell different STs and different versions of the same ST apart.

159 An ST also contains a TOE reference that identifies the TOE that claims conformance to the ST. A typical TOE reference consists of developer name, TOE name and TOE version number. An example of a TOE reference is “MauveCorp MauveRAM Database v2.11”. As a single TOE may be evaluated multiple times, for instance by different consumers of that TOE, and therefore have multiple STs, this reference is not necessarily unique.

160 If the TOE is constructed from one or more well-known Products, it is allowed to reflect this in the TOE reference, by referring to the Product name(s). However, this should not be used to mislead consumers: situations

where major parts or security functionalities were not considered in the evaluation, yet the TOE reference does not reflect this are not allowed.

161 The ST reference and the TOE reference facilitate indexing and referencing the ST and TOE and their inclusion in summaries of lists of evaluated TOEs/Products.

A.4.2 TOE overview

162 The TOE overview is aimed at potential consumers of a TOE who are looking through lists of evaluated TOEs/Products to find TOEs that may meet their security needs, and are supported by their hardware, software and firmware. The typical length of a TOE overview is several paragraphs.

163 To this end, the TOE overview briefly describes the usage of the TOE and its major security features, identifies the TOE type and identifies any major non-TOE hardware/software/firmware required by the TOE.

A.4.2.1 Usage and major security features of a TOE

164 The description of the usage and major security features of the TOE is intended to give a very general idea of what the TOE is capable of in terms of security, and what it can be used for in a security context.

165 An example of this is "The MauveCorp MauveRAM Database v2.11 is a multi-user database intended to be used in a networked environment. It allows 1024 users to be active simultaneously. It allows password/token and biometric authentication, protects against accidental data corruption, and can roll-back 10.000 transactions. Its audit features are highly configurable, so as to allow detailed audit to be performed for some users and transactions, while protecting the privacy of other users and transactions."

A.4.2.2 TOE type

166 The TOE overview identifies the general type of TOE, such as: firewall, VPN-firewall, smartcard, crypto-modem, intranet, web server, database, web server and database, LAN, LAN with web server and database, etc.

167 In some cases, a TOE type can mislead consumers. Examples include:

- certain functionality can be expected of the TOE because of its TOE type, but the TOE does not have this functionality. Examples include:
  - an ATM-card type TOE, which does not have any identification/authentication functionality;
  - a firewall type TOE, which does not support protocols that are almost universally used;
  - a PKI-type TOE, which has no certificate revocation functionality.

- the TOE can be expected to operate in certain operational environments because of its TOE type, but it cannot do so. Examples include:
  - a PC-operating system type TOE, which is unable to function securely unless the PC has no network connection, floppy drive, and CD/DVD-player;
  - a firewall, which is unable to function securely unless all users that can connect through that firewall are benign.

168 In these cases, the TOE overview must contain additional information to ensure that potential consumers are not misled.

A.4.2.3 Required non-TOE hardware/software/firmware

169 While some TOEs do not rely upon other IT, many TOEs (notably software TOEs) rely on additional, non-TOE, hardware, software and/or firmware. In the latter case, the TOE overview is required to identify this non-TOE hardware/software/firmware.

170 It is not required to provide a complete and fully detailed identification of all this hardware/software/firmware, but the identification should be complete and detailed enough for potential consumers to determine the major hardware/software/firmware components needed to use the TOE.

171 Example hardware/software/firmware identifications are:

- a standard PC with a 1GHz or higher processor and 512MB or more RAM, running version 3.0 Update 6b, c, or 7, or version 4.0 of the Yaiza operating system;
- a standard PC with a 1GHz or higher processor and 512MB or more RAM, running version 3.0 Update 6d of the Yaiza operating system and the WonderMagic 1.0 Graphics card with the 1.0 WM Driver Set;
- a standard PC with version 3.0 of the Yaiza OS (or higher);
- a CleverCard SB2067 integrated circuit;
- a CleverCard SB2067 integrated circuit running v2.0 of the QuickOS smartcard operating system;
- the December 2002 installation of the LAN of the Director-General's Office of the Department of Traffic.

A.4.3 TOE description

172 A TOE description is a narrative description of the TOE, likely to run to several pages. The TOE description should provide evaluators and potential consumers with a general understanding of the security capabilities of the

TOE, in more detail than was provided in the TOE overview. The TOE description may also be used to describe the wider application context into which the TOE will fit.

173 The TOE description discusses the physical scope and boundaries of the TOE: the hardware, firmware and software parts that constitute the TOE at a level of detail that is sufficient to give the reader a general understanding of those parts. The TOE description should also list all guidance that is part of the TOE.

174 The TOE description should also discuss the logical scope and boundaries of the TOE: the logical security features offered by the TOE at a level of detail that is sufficient to give the reader a general understanding of those features.

175 An important property of the physical and logical descriptions is that they describe the boundaries of the TOE in such a way that there remains no doubt on whether a certain part or feature is in the TOE or whether this is outside the TOE. This is especially important when the TOE is intertwined with and cannot be easily separated from non-TOE entities.

176 Examples where the TOE is intertwined with non-TOE entities are:

- the TOE is a cryptographic co-processor of a smart card IC, instead of the entire IC;
- the TOE is a smartcard IC, except for the cryptographic processor;
- the TOE is the Network Address Translation part of the MinuteGap Firewall v18.5.

A.5 Conformance claims (ASE_CCL)

177 This section of an ST describes how the TOE conforms with:

- the Common Criteria itself
- Protection Profiles (if any)
- Packages (if any)

178 This conformance claim is described in detail in Section 9.4.

179 If the conformance claim refers to one or more PPs and/or packages, the ST must also be actually conformant to those PPs and/or packages. In some cases, this means that the ST must contain additional material in the form of a conformance rationale.

A.5.1 Conforming to a Protection Profile

180 The CC allows three types in which a ST can claim conformance to a PP: exact, strict and demonstrable. The type of conformance is specified in the conformance statement of the PP that is being claimed conformance to.

181 In other words, this PP effectively states “Any ST claiming conformance to me, must do so in an [exact, strict, demonstrable] manner”. The ST claiming conformance to that PP simply states that it claims conformance to that PP without stating the nature of that compliance [exact, strict, demonstrable].

182 The three types of conformance are summarised below, and described more extensively in Sections A.5.2, A.5.3 and A.5.4.

183 **Exact conformance** is expected to be used by those PP authors with the most stringent requirements that are to be expressed in a single manner. This approach to PP specification will limit the ST able to claim conformance to the PP purely on the basis of the wording used in the PP, rather than a technical ability to meet the security requirements. This may be used in a Request For Quotation in a product acquisition process.

184 **Strict conformance** is expected to be used by those PP authors with vast experience of developing PPs, who again have requirements that must be adhered to in the manner specified. However, this completion permits the ST author claiming compliance to the PP to add to those requirements, provided it is in a restrictive manner. i.e. the additional requirements cannot weaken the existing requirements.

185 **Demonstrable conformance** allows a PP author to describe a common security problem to be solved and generic guidelines to the requirements necessary for its resolution, in the knowledge that there is likely to be more than some way of specifying a resolution.

186 Note that conformance is a binary property of a ST; either the ST conforms to the PP in question or it does not. The CC does not recognise "partial" conformance. As partial conformance is not permissible, it is the responsibility of the PP author to ensure the PP is not overly onerous, prohibiting PP/ST authors in claiming conformance to the PP.

A.5.2 Exact conformance

187 Exact conformance is oriented to the PP-author who requires evidence that the requirements in the PP are met precisely and that any ST claiming conformance is an instantiation of the PP; there are to be no additions or modifications from the specification of the PP.

- The security problem definition and objectives specified in the PP are to be duplicated in the ST either by copying or by reference.
- Alternative security requirement claims to those in the PP cannot be used in the ST.
- No additional (functional or assurance) security requirement claims can be made in the ST.
- All remaining assignment and selection operations are to be completed.

188 The conformance rationale will be a trivial statement that the security problem definition, statement of security objectives and statement of security requirements have been included in the ST.

A.5.3 Strict conformance

189 Strict conformance is oriented to the PP-author who requires evidence that the requirements in the PP are met, that the ST is an instantiation of the PP, though the ST could be broader than the PP:

- The statements of the security problem definition and the security objectives in the ST are to be consistent with those in the PP. These statements can be re-worded using terminology with which the ST consumer will be conversant. However, the conformance rationale is to demonstrate that each aspect of the statements specified in the PP has been provided in the ST.

- The objectives for the operational environment can be modified providing the statement of security objectives in the ST is more restrictive that than that of the PP. This can include reassigning an security objective for the operational environment in the PP to be a TOE objective in the ST.

- The SFRs specified in the ST must be a non-strict superset of the SFRs specified in the PP; i.e. the ST must claim the SFRs specified in the PP as a minimum but could claim more (or hierarchically stronger SFRs).

- The SARs specified in the ST must be a non-strict superset of the SARs specified in the PP; i.e. the ST must claim SARs specified in the PP as a minimum, but could claim more (or hierarchically stronger SARs).

- The completion of operations in the ST must be consistent with that in the PP; either the same completion will be used in the ST as that in the PP or one that makes the requirement more restrictive (the rules of refinement apply).

190 If the PP author does not wish objectives for the environment to be re-assigned as objectives of the TOE, he should

a) consider whether it would be more appropriate to require exact conformance;

b) express the objective for the environment in such a way that it cannot be reworded as a security objective for the TOE, whilst remaining consistent with that specified in the PP.

191 The conformance rationale in an ST conforming to a PP requiring strict conformance will be a simple tracing between the statement of security requirements in the PP and the ST, and a discussion of:

– how the restatement of the security problem definition and objectives in the ST is consistent with that specified in the ST. All aspects of the statements will be considered and traced.

– the security requirements included in the ST in addition to those specified in the PP. This will include tracing these requirements to the additional aspects of the statements of security problem definition and objectives included in the ST.

A.5.4 Demonstrable conformance

192 Demonstrable conformance is orientated to the PP-author who requires evidence that the ST is a suitable solution to the generic security problem described in the PP. Demonstrable conformance is also suitable for the ST author wishing to claim conformance to multiple PPs.

– The SARs specified in the ST must be a non-strict superset of the SARs specified in the PP; i.e. the ST must claim SARs specified in the PP as a minimum, but could claim more (or hierarchically stronger SARs).

– The ST, although ensuring all requirements specified in the PP are expressed in the ST, is able to use alternative SFRs taken from CC Part 2 where applicable. A rationale will be provided to explain how the SFRs specified in the ST achieves at least the same as the SFRs specified in the PP.

– Any changes to the security objectives for the operational environment will make the description more restrictive (in the sense of refinement), or be as a result of moving an objective specified for the operational environment in the PP to become an objective for the TOE in the ST. A rationale will be provided to explain how the operational environment described in the ST is consistent with that described in the PP.

– The completion of operations will be consistent with those in the PP; i.e. the same completion is used in the ST as that in the PP or a completion that makes the requirement more restrictive (the rules of refinement apply).

  For example, if the PP author restricts the selection of four items in the component FAU_GEN.1.1b to two items in the PP. The ST can then only choose from the two in the PP, and not the other two. Nevertheless, the ST author may also add some audit events within the assignment in FAU_GEN.1.1c.

193 The conformance rationale is to demonstrate the following:

a) How each requirement in the PP is represented in the ST. If alternative requirements are expressed in the ST, the rationale is to contain the ST authors understanding of the relevant PP objective(s)

and how the alternative requirement(s) still result in achievement of the objective(s).

b) That the statement of objectives for the operational environment in the PP is fully expressed in the ST. This may be either:

– through equivalent or more restrictive objectives than those in the PP; or

– through expression of a TOE requirement that has been introduced in the ST to meet an objective stated for the environment in the PP.

c) The source of each additional security requirement; how it is necessary to meet the expanded set of security objectives for the TOE, resulting from the expanded security problem definition in the ST.

A.5.5 Conformance to a package

194 A package is defined as a set of functional or assurance requirements that meet an identifiable subset of security objectives. It is intended to be re-usable, to be used in the construction of larger packages, PPs and STs. At present there are no criteria for the evaluation of packages, to confirm their content or to place requirements upon packages, e.g. that a package must include a statement of the type of conformance. Therefore, only the security requirements specified in a package are considered when conformance to a package is claimed.

195 The package conformance claims are <package name> conformant and <package name> augmented. These are comparable to exact conformance and strict conformance respectively. The ST author specifies the type of conformance to a package.

196 The completions of operations in the ST are to be consistent with that specified in the requirements package. Therefore, the same completion is used in the ST as that in the package or a completion that makes the requirement more restrictive (the rules of refinement apply).

A.5.5.1 <package name> conformant

197 A conformance claim that an ST is "<package name> conformant" is considered to fall under the categorisation of "exact" conformance used for PP conformance claims. Therefore, all requirements in the package must be included in the ST, with no substitution and no additions.

A.5.5.2 <package name> augmented

198 A conformance claim that a ST is "<package name> augmented" is considered to fall under the categorisation of "strict" conformance used for PP conformance claims. Therefore, all requirements in the package must be

included in the ST, with no substitution. However, requirements in addition to those specified in the package may be included in the ST.

A.6 Security problem definition (ASE_SPD)

A.6.1 Introduction

199 The security problem definition defines the security problem that is to be addressed. The security problem definition is, as far as the CC is concerned, axiomatic. That is, the process of deriving the security problem definition falls outside the scope of the CC.

200 However, it should be noted that the usefulness of the results of an evaluation strongly depends on the ST, and the usefulness of the ST strongly depends on the quality of the security problem definition. It is therefore often worthwhile to spend significant resources and use well-defined processes and analyses to derive a good security problem definition.

201 Note that it is not mandatory to have statements in all sections, an ST can have no threats, or no OSPs, or no assumptions. However, if an ST has no threats, it must have OSPs and vice versa.

202 Also note that where the TOE is physically distributed, it may be better to discuss the relevant threats, OSPs and assumptions separately for distinct domains of the TOE operational environment. Similarly, where the development environment of the TOE consists of multiple sites or stages, it may be better to discuss the relevant threats, OSPs and assumptions separately for each distinct site or stage.

A.6.2 Threats

203 This section of the security problem definition shows the threats that are to be countered by the TOE, its development environment, its operational environment, or a combination of these three.

204 A threat consists of a threat agent, an asset (either in the operational or in the development environment) and an adverse action of that threat agent on that asset.

205 *Threat agents* are entities that can adversely act on assets. Examples of threat agents are hackers, users, computer processes, TOE development personnel, and accidents. Threat agents may be further described by aspects such as expertise, resources, opportunity and motivation.

206 Examples of *assets* can be found in Section 7.1.

207 *Adverse actions* are actions performed by a threat agent on an asset. These actions influence one or more properties of an asset from which that asset derives its value.

208 Examples of threats are:

- a hacker (with substantial expertise, standard equipment, and being paid to do so) remotely copying confidential files from a company network;
- a worm seriously degrading the performance of a wide-area network;
- a virus sending out stored confidential email to random recipients;
- a TOE developer employee making an accidental error affecting the correctness of the low-level design of the TOE;
- a system administrator violating user privacy;
- a malicious TOE developer employee (with very substantial expertise on the source code, but not many other IT security skills) modifying the source code;
- a cleaner stealing confidential design information and/or source code.

A.6.3 Organisational security policies (OSPs)

209 This section of the security problem definition shows the OSPs that are to be enforced by the TOE, its development environment, its operational environment, or a combination of these three.

210 OSPs are rules, practises, or guidelines. These may be laid down by the organisation controlling the operational environment of the TOE, or they may be laid down by legislative or regulatory bodies. OSPs can apply to the TOE, the operational environment of the TOE, and/or the development environment of the TOE.

211 Examples of OSPs are:

- All products that are used by the Government must conform to the National Standard for password generation and encryption;
- All products that are used by the Bank, must be CC-certified with the EAL 4 + ADV_IMP.2 assurance package;
- All system administrators that have access to the Department File Servers must be vetted to the level of Department Secret.

A.6.4 Assumptions

212 This section of the security problem definition shows the assumptions that the TOE makes on its operational environment in order to be able to provide security functionality. If the TOE is placed in an operational environment that does not meet these assumptions, the TOE may not be able to provide all of its security functionality anymore. Assumptions can be on physical, personnel and connectivity of the operational environment.

213 Examples of assumptions are:

- Assumptions on physical aspects of the operational environment:
  - the TOE assumes that it will be placed in a room that is designed to minimise electro-magnetic emanations;
  - the TOE assumes that its administrator consoles will be placed in a restricted access area.
- Assumptions on personnel aspects of the operational environment:
  - the TOE assumes that its users will be trained sufficiently in order to operate the TOE;
  - the TOE assumes that its users are vetted for information that is classified as National Secret;
  - the TOE assumes that its users will not write down their passwords.
- Assumptions on connectivity aspects of the operational environment:
  - the TOE assumes that it will run on a PC workstation with at least 10GB of disk space;
  - the TOE assumes that it is the only non-OS application running on this workstation;
  - the TOE assumes that it will not be connected to an untrusted network.

214 Note that assumptions can only apply to the operational environment. Assumptions can never apply to the TOE and/or the development environment, as the TOE cannot assume anything about itself, or on how it is developed.

A.7 Security objectives (ASE_OBJ)

215 The security objectives are a concise and abstract statement of the intended solution to the problem defined by the security problem definition. The role of the security objectives is threefold:

- provide a high-level, natural language solution of the problem;
- divide this solution into three partwise solutions, that reflect that different entities each have to address a part of the problem;
- demonstrate that these partwise solutions form a complete solution to the problem.

A.7.1 High-level solution

216 The security objectives consist of a set of short and clear statements without overly much detail that together form a high-level solution to the security problem. The level of abstraction of the security objectives aims at being clear and understandable to knowledgeable potential consumers of the TOE. The security objectives are in natural language, as a more exact, well-defined description of some of the security objectives will be provided as part of the security requirements, which are described later on in this chapter.

A.7.2 Partwise solutions

217 In an ST the high-level security solution, as described by the security objectives, is divided into three partwise solutions. These partwise solutions are called the security objectives for the TOE, the security objectives for the development environment, and the security objectives for the operational environment. This reflects that these partwise solutions are to be provided by three different entities: the TOE, the development environment and the operational environment.

A.7.2.1 Security objectives for the TOE

218 The TOE provides security functionality to solve a certain part of the problem defined by the security problem definition. This partwise solution is called the security objectives for the TOE and consists of a set of statements describing the security goals that the TOE should achieve in order to solve its part of the problem.

219 Examples of security objectives for the TOE are:

- The TOE shall keep confidential the content of all files transmitted between it and a Server;
- The TOE shall identify and authenticate all users before allowing them access to the Transmission Service provided by the TOE;
- The TOE shall restrict user access to data according to the Data Access policy described in Annex 3 of the ST.

220 If the TOE is physically distributed, it may be better to subdivide the security objectives for the TOE into several sections to reflect this.

A.7.2.2 Security objectives for the development environment

221 The development environment of the TOE contains technical and procedural measures to provide assurance that the TOE will correctly provide its security functionality (which is defined by the security objectives for the TOE). This partwise solution is called the security objectives for the development environment and consists of a set of statements describing the security goals that should be achieved in the development environment.

222 Examples of security objectives for the development environment are:

- The development environment shall ensure that the TOE is delivered to the consumer without compromising the integrity of the TOE;
- The development environment shall ensure that the integrity of the source code of the TOE is protected;
- The development environment shall ensure that complete and clear guidance to the TOE is developed, thus minimising the probability that users will use the TOE in manner that it was not intended;
- The development environment shall conform with EAL 4 augmented with ADV_IMP.2.

223 If the development environment of the TOE consists of multiple sites or stages, it may be better to subdivide the security objectives for the development environment into several sections to reflect this.

A.7.2.3 Security objectives for the operational environment

224 The operational environment of the TOE implements technical and procedural measures to assist the TOE in correctly providing its security functionality (which is defined by the security objectives for the TOE). This partwise solution is called the security objectives for the operational environment and consists of a set of statements describing the goals that the operational environment should achieve.

225 Examples of security objectives for the operational environment are:

- The operational environment shall provide a workstation with the OS Inux version 3.01b to execute the TOE on;
- The operational environment shall ensure that all human TOE users receive appropriate training before allowing them to work with the TOE;
- The operational environment of the TOE shall restrict physical access to the TOE to administrative personnel and maintenance personnel accompanied by administrative personnel;
- The operational environment shall ensure the confidentiality of the audit logs generated by the TOE before sending them to the central Audit Server.

226 If the operational environment of the TOE consists of multiple sites, each with different properties, it may be better to subdivide the security objectives for the operational environment into several sections to reflect this.

A.7.3 Relation between security objectives and the security problem definition

227 The ST also contains a security objectives rationale containing two sections:

– a tracing that shows which security objectives address which threats, OSPs and assumptions;

– a set of justifications that shows that all threats, OSPs, and assumptions are effectively addressed by the security objectives.

A.7.3.1 Tracing between security objectives and the security problem definition

228 The tracing shows how the security objectives trace back to the threats, OSPs and assumptions as described in the security problem definition. This tracing must obey three rules:

a) *No spurious objectives*: Each security objective traces to at least one threat, OSP or assumption.

b) *Complete with respect to the security problem definition*: Each threat, OSP and assumption has at least one security objective tracing to it.

c) *Correct tracing*: Since assumptions are always made by the TOE on the operational environment, security objectives for the TOE and for the development environment do not trace back to assumptions. The allowed tracings are depicted in Figure 6.

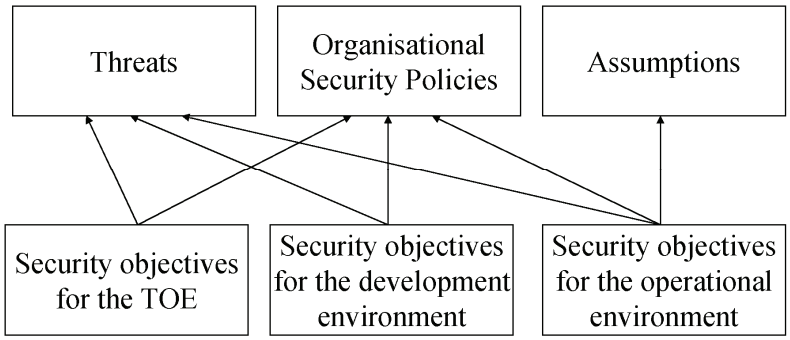

Figure 6 - Allowed tracings between security objectives and security problem definition

229 Multiple security objectives may trace to the same threat, indicating that the combination of those security objectives counters that threat. A similar argument holds for OSPs and assumptions.

A.7.3.2 Providing a justification for the tracing

230 The security objectives rationale also demonstrates that the tracing is effective: if all security objectives tracing to a particular threat/OSP/assumption are achieved, that threat/OSP/assumption is countered/enforced/upheld.

231 This demonstration analyses the effect of achieving the relevant security objectives on countering the threats, enforcing the OSPs and upholding the assumptions and leads to the conclusion that this is indeed the case.

232 In some cases, where parts of the security problem definition very closely resemble some security objectives, the demonstration can be very simple. An example is: a threat "T17: Threat agent X reads the Confidential Information in transit between A and B", a security objective for the TOE: "OT12: The TOE shall ensure that all information transmitted between A and B is kept confidential", and a demonstration "T17 is directly countered by OT12".

A.7.3.3 On countering threats

233 Countering a threat does not necessarily mean removing that threat, it can also mean sufficiently diminishing that threat or sufficiently mitigating that threat.

234 Examples of removing a threat are:

- removing the ability to execute the adverse action from the threat agent;
- moving, changing or protecting the asset in such a way that the adverse action is no longer applicable to it;
- removing the threat agent (e.g. removing machines from a network that frequently crash that network).

235 Examples of diminishing a threat are:

- restricting the ability of a threat agent to perform adverse actions;
- restricting the opportunity to execute an adverse action of a threat agent;
- reducing the likelihood of an executed adverse action being successful;
- reducing the motivation to execute an adverse action of a threat agent by deterrence;
- requiring greater expertise or greater resources from the threat agent.

236 Examples of mitigating the effects of a threat are:

- making frequent back-ups of the asset;
- obtaining spare copies of an asset;
- insuring an asset;
- ensuring that successful adverse actions are always timely detected, so that appropriate action can be taken.

A.7.4 Security objectives: conclusion

237 Based on the security objectives and the security objectives rationale, the following conclusion can be drawn: if all security objectives are achieved then the security problem as defined in ASE_SPD is solved: all threats are countered, all OSPs are enforced, and all assumptions are upheld.

A.8 Extended Components Definition (ASE_ECD)

238 In this section of the ST all additional components needed in the ST, but not present in CC Part 2 or CC Part 3, are defined. For more information on this, see Annex C.5

A.9 Security requirements (ASE_REQ)

A.9.1 Well-defined translation

239 The security requirements are a well-defined translation of the security objectives for the TOE and the security objectives for the development environment. They are usually at a more detailed level of abstraction, but they have to be a complete translation (the security objectives must be completely addressed). The CC requires this well-defined translation for several reasons:

- to provide an exact description of what is to be evaluated: the security functional requirements (SFRs). These are a well-defined translation of the security objectives for the TOE.
- to provide an exact description of how the TOE is to be evaluated: the security assurance requirements (SARs). These are a well-defined translation of the security objectives for the development environment.
- to allow comparison between two STs. As different ST authors may use different terminology in describing their security objectives, the well-defined translations must use the same terminology and concepts. This allows easy comparison.

240 There is no well-defined translation required in the CC for the security objectives for the operational environment, because the operational environment is not evaluated and does therefore not require a more exact description. It may be the case that parts of the operational environment are evaluated in another evaluation, but this is out of scope for the current evaluation.

A.9.2 How the CC supports this well-defined translation

241 The CC supports this well-defined translation in four ways:

a) by providing a predefined well-defined “language” designed to describe exactly what is to be evaluated. This language is defined as a

set of components defined in CC Part 2. The use of this language as a well-defined translation of the security objectives for the TOE to SFRs is mandatory, though some exceptions exist (see C.5).

b) by providing a predefined well-defined "language" designed to describe exactly how the TOE is to be evaluated. This language is defined as a set of components defined in CC Part 3. The use of this language as a well-defined translation of the security objectives for the development environment to SARs is mandatory, though some exceptions exist (see Annex C.5).

c) by providing operations: mechanisms that allow the ST writer to modify the SFRs and SARs to provide a more accurate translation of the security objectives for the TOE and the development environment. The CC has four operations: assignment, selection, iteration, and refinement. These are described further in Section C.4.4.

d) by providing dependencies: a mechanism that supports a more complete translation to SFRs and SARs. In the CC Part 2 and CC Part 3 languages, a security requirement can have a dependency on other security requirements. This signifies that if an ST uses that requirement, it generally needs to use those other security requirements as well. This makes it much harder for the ST writer to overlook including necessary requirements and thereby improves the completeness of PPs and STs. Dependencies are described further in Section C.3.

A.9.3 Relation between security requirements and security objectives

242 The ST also contains a security requirements rationale, consisting of two sections:

- a tracing that shows which security requirements address which security objectives;
- a set of justifications that shows that all security objectives for the TOE and for the development environment are effectively addressed by the security requirements.

A.9.3.1 Tracing between security requirements and the security objectives

243 The tracing shows how the SFRs and SARs trace back to the security objectives for the TOE and the security objectives for the development environment. This tracing must obey three rules:

a) *No spurious SFRs/SARs*: Each SFR/SAR traces back to at least one security objective.

b) *Complete with respect to the security objectives for the TOE and the development environment*: Each security objective for the TOE and

each security objective for the development environment has at least one security requirement tracing to it.

c) *Correct tracing*:

– SFRs define measurable functional properties of the TOE, and can therefore trace only to security objectives for the TOE;

– SARs define that the TOE and certain documents with a certain content and presentation must be available and that the developer and evaluator must undertake certain actions on the TOE and on these documents. As all these actions take place in the development environment, SARs can only trace to security objectives for the development environment.

244 The allowed tracings are depicted in Figure 7.

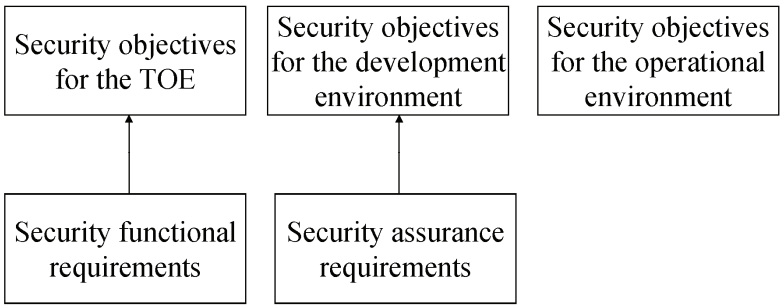

Figure 7 - Allowed tracings between security requirements and security objectives

245 Multiple security requirements may trace to the same security objective, indicating that the combination of those security requirements meets that objective.

A.9.3.1.1 Providing a justification for the tracing

246 The security requirements rationale must also demonstrate that the tracing is effective: if all security requirements tracing to a particular security objective are satisfied, that security objective is achieved.

247 This demonstration should analyse the effect of satisfying the relevant security requirement on achieving the security objective and lead to the conclusion that this is indeed the case.

248 In some cases, where some security requirements very closely resemble some security objectives, the demonstration can be very simple. An example is:

– A security objective for the development environment “OD14: The development environment shall conform to EAL3 + ADV_FSP.2”, a set of SARs consisting of EAL3 and ADV_FSP.2 and a rationale “OD14 is directly achieved by the SARs”.

A.9.3.2 Security requirements: conclusion

249 In the security problem definition of the ST, the security problem is defined as consisting of threats, OSPs and assumptions. In the security objectives section of the ST, the solution is provided in the form of three sub-solutions:

- security objectives for the TOE;
- security objectives for the development; environment
- security objectives for the operational environment.

250 Additionally, a security objectives rationale is provided showing that if all security objectives are achieved, the security problem is solved: all threats are countered, all OSPs are enforced, and all assumptions are upheld.

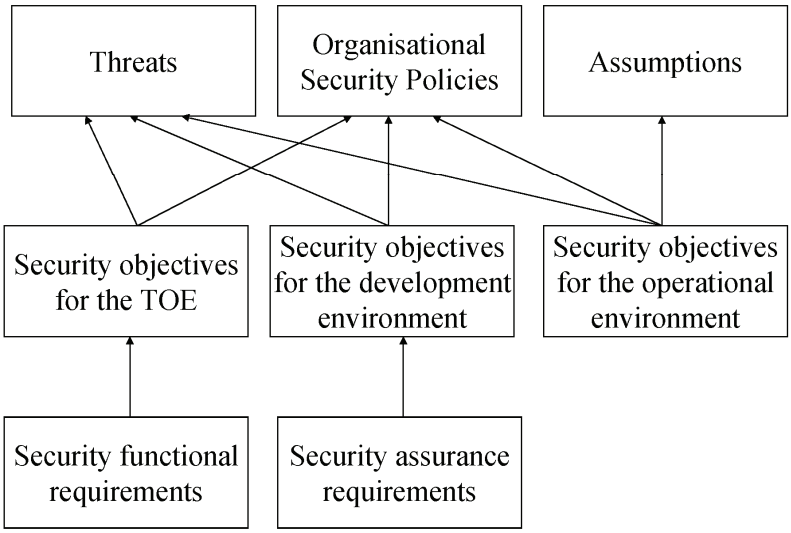

Figure 8 - Relations between the security problem definition, the security objectives and the security requirements

251 In the security requirements section of the ST a well-defined translation is provided of two of the sets of security objectives:

- the security objectives for the TOE are translated to SFRs
- the security objectives for the development environment are translated to SARs

252 Additionally, a security requirements rationale is provided showing that if all SFRs are satisfied, all security objectives for the TOE are achieved and if all SARs are satisfied, all security objectives for the development environment are achieved.

253 This can be combined into a single statement: If all SFRs and SARs are satisfied and all security objectives for the operational environment are achieved, then the security problem as defined in ASE_SPD is solved: all

threats are countered, all OSPs are enforced, and all assumptions are upheld. This is illustrated in Figure 8.

A.9.3.3 Notes on tracing and rationales

254 Figure 8 shows that (through the security objectives) every SFR and SAR must be traced back through the security objectives into individual statements in the security problem definition. This tracing can be coarse or detailed depending on the chosen level of granularity in the security problem definition and the security objectives.

255 For example, if the SARs consist of EAL 4 + ADV_IMP.2, some possible options are:

- A single OSP "The TOE shall be evaluated at EAL 4 + ADV_IMP.2" leading to a single security objective for the development environment "The development environment shall comply with EAL4 + ADV_IMP.2" and trace all SARs back to that single security objective.

- An OSP "The TOE shall be developed according to good commercial development practises applied rigorously", a threat "Threat Agent X obtains the source code by theft or reverse engineering, subverts the TOE and thereby is able to read the Confidential Data Asset", leading to two security objectives for the development environment "The development shall comply with EAL 4" and "The development environment shall have a thorough and complete source code level analysis performed" and tracing the EAL4 SARs to the EAL4 security objective and the ADV_IMP.2 SAR to the source code security objective.

- An extensive set of OSPs and threats relating to the development environment, leading to an extensive set of security objectives for the development environment and a detailed tracing of the SARs to these security objectives.

256 Similar examples apply to tracing of SFRs.

257 The choice of granularity is made by the ST author.

A.10 TOE summary specification (ASE_TSS)

258 The objective for the TOE summary specification is to provide potential consumers of the TOE with a description of how the TOE satisfies all the SFRs. The TOE summary specification should provide the general technical mechanisms that the TOE uses for this purpose. The level of detail of this description should be enough to enable potential consumers to understand the general form and implementation of the TOE.

259 For instance if the TOE is an Internet PC and the SFRs contain FIA_UAU.2 to specify authentication, the TOE summary specification should indicate

how this authentication is done: password, token, iris scanning etc. More information, like applicable standards that the TOE uses to meet SFRs, or more detailed descriptions may also be provided.

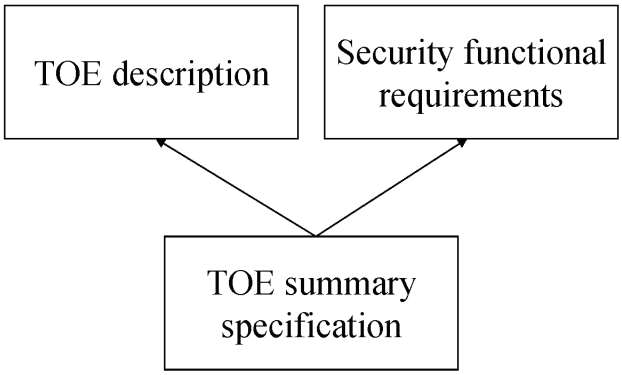

Figure 9 - Relations between the TOE description, the SFRs and the TOE summary specification

A.11 Questions that can be answered with an ST (informative)

260 After the evaluation, the ST specifies "what was evaluated". In this role, the ST serves as a basis for agreement between the developer or re-seller of the TOE and the potential consumer of the TOE. The ST can therefore answer the following questions (and more):

a) *How can I find the ST/TOE that I need given the multitude of existing STs/TOEs?* This question is addressed by the TOE overview, which gives a brief (several paragraphs) summary of the TOE;

b) *Does this TOE fit in with my existing IT-infrastructure?* This question is addressed by the TOE overview, which identifies the major hardware/firmware/software elements needed to run the TOE;

c) *Does this TOE fit in with my existing operational environment?* This question is addressed by the security objectives for the operational environment, which identifies all constraints the TOE places on the operational environment in order to function;

d) *What does the TOE do (interested reader)?* This question is addressed by the TOE overview, which gives a brief (several paragraphs) summary of the TOE;

e) *What does the TOE do (potential consumer)?* This question is addressed by the TOE description, which gives a less brief (several pages) summary of the TOE;

f) *What does the TOE do (technical)?* This question is addressed by the TOE summary specification which provides a high-level description of the mechanisms the TOE uses;

g) *What does the TOE do (expert)?* This question is addressed by the SFRs which provide an abstract highly technical description, and the TOE summary specification which provide additional detail;

h) *Does the TOE address the problem as defined by my government/organisation?* If your government/organisation has defined packages and/or PPs to define this solution, then the answer can be found in the Conformance Claims section of the ST, which lists all packages and PPs that the ST conforms to.

i) *Does the TOE address my security problem (expert)?* What are the threats countered by the TOE? What organisational security policies does it enforce? What assumptions does it make about the operational environment? These questions are addressed by the security problem definition;

j) *How much trust can I place in the TOE?* This can be found in the SARs in the security requirements section, which provide the assurance level that was used to evaluate the TOE, and hence the trust that the evaluation provides in the correct functioning of the TOE.

A.12 Low assurance Security Targets

261 Writing an ST is not a trivial task, and may, especially in low assurance evaluations, be a major part of the total effort expended by the developer and the evaluator in the whole of the evaluation. For this reason, it is also possible to write a low assurance ST.

262 The CC allows the use of a low assurance ST for an EAL 1 evaluation, but not for EAL 2 and up. Additionally, if the ST is based on a PP, this is only allowed if this PP is a low assurance PP (see Annex B): low assurance STs shall not be used in conjunction with non-low assurance PPs.

263 There are two important differences between a “full” ST and a low assurance ST:

– Reduced content: a low assurance ST does not have to contain a security problem definition, a security objectives rationale and a security requirements rationale. The content of the statement of security objectives in a low assurance ST is also reduced.

– Reduced completeness: the SFRs and SARs in a low assurance ST do not have to meet their dependencies.

A.12.1 Reduced content

264 A low assurance ST has a significantly reduced content:

– there is no need to describe the security problem definition (threats, OSPs and assumptions that the TOE must counter, enforce and uphold);

- there is no need to describe the security objectives for the TOE and the security objectives for the development environment. The security objectives for the operational environment shall still be described;
- there is no need to describe the security objectives rationale as there is no security problem definition in the ST;
- there is no need to describe the security requirements rationale as there are no security objectives for the TOE or security objectives for the development environment in the ST.

265 All that remains are:

a) the references to TOE and ST

b) the conformance claim

c) the various narrative descriptions

   1) the TOE overview

   2) the TOE description

   3) the TOE summary specification

d) the security objectives for the operational environment

e) the security requirements (including the extended components definition).

266 The reduced content of a low assurance ST is shown in Figure 10.

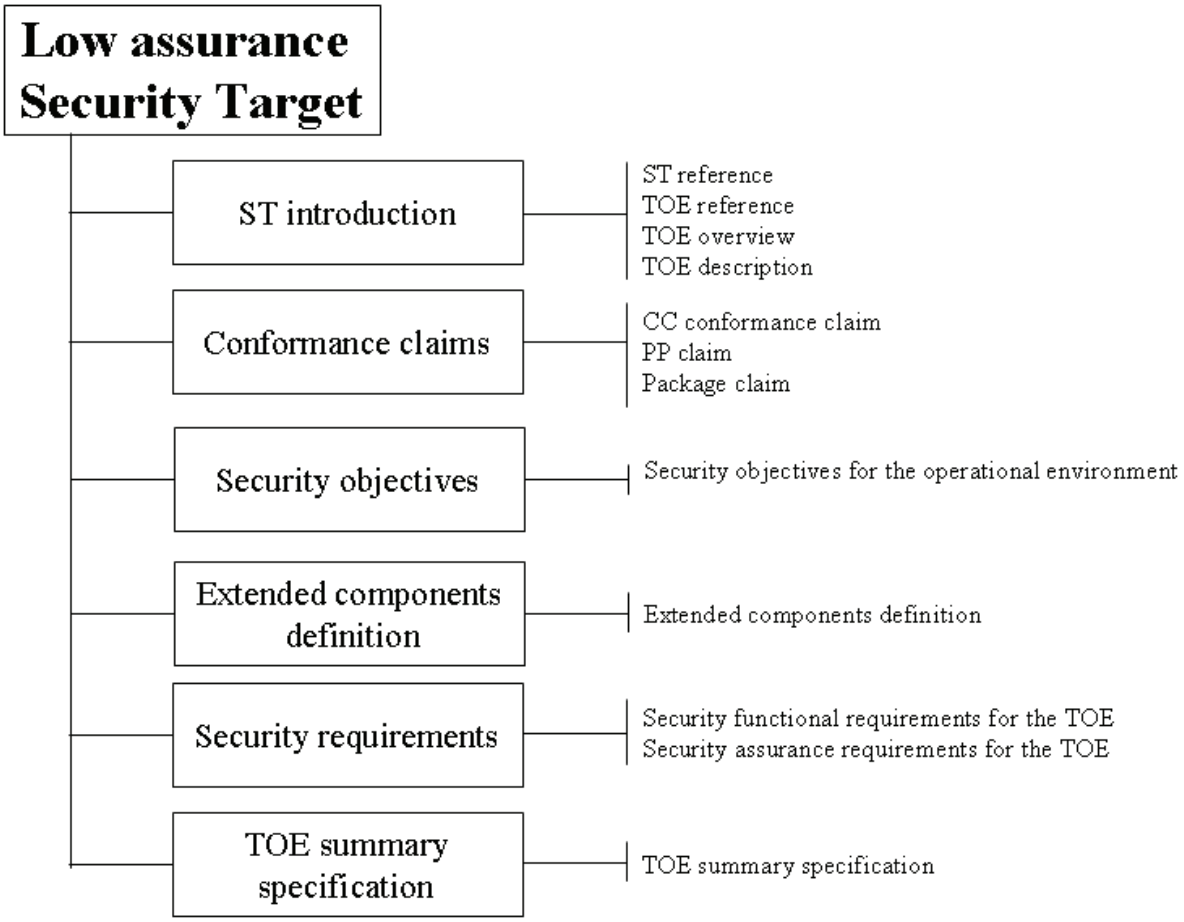

Figure 10 - Contents of a Low Assurance Security Target

A.12.2 Reduced completeness

267 A low assurance ST has reduced requirements for completeness: it is no longer required to provide a rationale for not meeting a dependency. However, a low assurance ST writer may consider the dependencies while writing the ST to prevent incoherent and/or obviously incomplete sets of SFRs and SARs.

B Specification of Protection Profiles (normative)

B.1 Goal and structure of this Annex

268 The goal of this Annex is to explain the APE criteria and provide examples of their application. This Annex does not define the APE criteria, this definition can be found in CC Part 3.

269 As Protection Profiles and Security Targets have a significant overlap, this Annex focuses on the differences between Protection Profiles and Security Targets. The material that is identical between Security Targets and Protection Profiles is described in Annex A.

270 This annex consists of three major parts:

a) *What a PP must contain*. This is summarised in Section B.2, and described in more detail in Sections B.4-B.9. These sections describe the mandatory contents of the PP, the interrelationships between these contents, and provide examples.

b) *How a PP should be used*. This is summarised in Section B.3.

c) *Low Assurance PPs*. Low Assurance PPs are PPs with reduced content. They are described in detail in Section B.11.

B.2 Mandatory contents of a PP

271 Figure 11 portrays the mandatory content for a PP. Figure 11 may also be used as a structural outline of the PP, though alternative structures are allowed. For instance, if the security requirements rationale is particularly bulky, it could be included in an appendix of the PP instead of in the security requirements section. The separate sections of a PP and the contents of those sections are briefly summarised below and described in much more detail in Sections B.4 - B.9. A PP must contain:

a) a PP *introduction* containing a narrative description of the TOE type;

b) a *conformance claim*, showing whether the PP claims conformance to any PPs and/or packages, and if so, to which PPs and/or packages;

c) a *security problem definition*, showing the threats, OSPs and assumptions that must be countered, enforced and upheld by the TOE and its operational environment;

d) *security objectives*, showing how the solution to the security problem is divided between:

– the TOE;

– the development environment of the TOE;

– the operational environment of the TOE;

e) *extended components definition*, where new components (i.e. not included in CC Part 2 or CC Part 3) may be defined. These new components can then be used to define extended functional and extended assurance requirements with.

f) *security requirements*, where a well-defined translation of the security objectives for the TOE and the security objectives for the development environment is provided. This well-defined translation is in the form of CC Part 2 requirements, CC Part 3 requirements and extended security requirements.

272 There also exist low assurance PPs, which have reduced contents, these are described in detail in Section B.11. The rest of this Annex assumes that a PP with full contents is used.

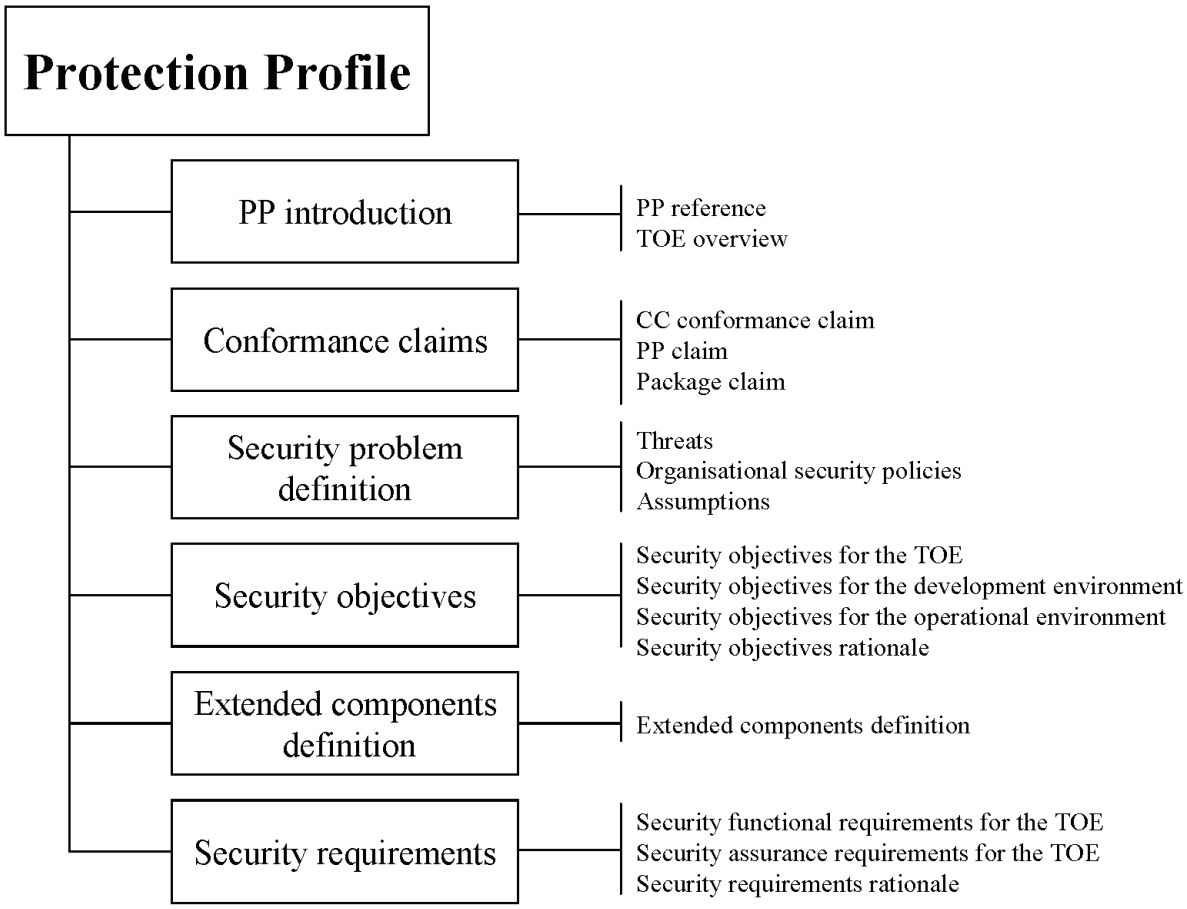

Figure 11 - Protection Profile contents

B.3 Using the PP (informative)

B.3.1 How a PP should be used

273 A PP is typically a statement of need where a user community, a regulatory entity, or a group of developers define a common set of security needs. A PP gives consumers a means of referring to this set, and facilitates future evaluation against these needs.

274 A PP is therefore typically used as:

- part of a requirement specification for a specific consumer or group of consumers, who will only consider buying a specific type of IT if it meets the PP;
- part of a regulation from a specific regulatory entity, who will only allow a specific type of IT to be used if it meets the PP;
- a baseline defined by a group of IT developers, who then agree that all IT that they produce of this type will meet this baseline.

though this does not preclude other uses.

B.3.2 How a PP should not be used

275 Three roles (among many) that a PP should not fulfil are:

- *a detailed specification*: A PP is designed to be a security specification on a relatively high level of abstraction. A PP should, in general, not contain detailed protocol specifications, detailed descriptions of algorithms and/or mechanisms, long description of detailed operations etc.
- *a complete specification*: A PP is designed to be a security specification and not a general specification. Unless security-relevant, properties such as interoperability, physical size and weight, required voltage etc. should not be part of a PP. This means that in general a PP is a part of a complete specification, but not a complete specification itself.
- *a specification of a single product*: Unlike an ST, a PP is designed to describe a certain type of IT, and not a single product. When only a single product is described, it is better to use a Security Target for this purpose.

B.4 PP introduction (APE_INT)

276 A PP introduction describes the TOE on two levels of abstraction:

a) the PP reference;

b) the TOE overview.

B.4.1 PP reference

277 A PP contains a clear PP reference that identifies that particular PP. A typical PP reference consists of title, version, authors and publication date. An example of a PP reference is "Atlantean Navy CablePhone Encryptor PP, version 2b, Atlantean Navy Procurement Office, April 7, 2003". The reference must be unique so that it is possible to tell different PPs and different versions of the same PP apart.

278 The PP reference facilitates indexing and referencing the PP and its inclusion in lists of Protection Profile.

B.4.2 TOE overview

279 The TOE overview is aimed at potential consumers of a TOE who are looking through lists of evaluated products to find TOEs that may meet their security needs, and are supported by their hardware, software and firmware. The typical length of a TOE overview is several paragraphs.

280 To this end, the TOE overview briefly describes the usage of the TOE and its major security features, identifies the TOE type and identifies any major non-TOE hardware/software/firmware available to the TOE.

B.4.2.1 Usage and major security features of a TOE

281 The description of the usage and major security features of the TOE is intended to give a very general idea of what the TOE should be capable of, and what it can be used for.

282 An example of this is "The Atlantean Navy CablePhone Encryptor is an encryption device that should allow confidential communication between ships across the Atlantean Navy CablePhone system. To this end it should allow at least 32 different users and support at least 100Mb encryption speed. It should allow both bilateral communication between ships and broadcast across the entire network."

B.4.2.2 TOE Type

283 The TOE overview identifies the general type of TOE, such as: firewall, VPN-firewall, smartcard, crypto-modem, intranet, web server, database, web server and database, LAN, LAN with web server and database, etc.

B.4.2.3 Available non-TOE hardware/software/firmware

284 While some TOEs do not rely upon other IT, many TOEs (notably software TOEs) rely on additional, non-TOE, hardware, software and/or firmware. In the latter case, the TOE overview is required to identify this non-TOE hardware/software/firmware.

285 It is not required to provide a complete and fully detailed identification of all this hardware/software/firmware, but the identification should be complete and detailed enough for potential consumers to determine the major hardware/software/firmware components needed to use the TOE.

286 Example hardware/software/firmware identifications are:

- a standard PC with a 1GHz or higher processor and 512MB or more RAM, running version 3.0 Update 6b, c, or 7, or version 4.0 of the Yaiza operating system;
- a standard PC with a 1GHz or higher processor and 512MB or more RAM, running version 3.0 Update 6d of the Yaiza operating system and the WonderMagic 1.0 Graphics card with the 1.0 WM Driver Set;
- a standard PC with the Yaiza OS version 3.0 or higher;
- a CleverCard SB2067 IC;
- a CleverCard SB2067 IC running v2.0 of the QuickOS smartcard operating system;
- the December 2002 installation of the LAN of the Director-General's Office of the Department of Traffic.

B.5 Conformance claims (APE_CCL)

287 This section of a PP describes how the PP conforms with other PPs and with packages. It is identical to the conformance claims section for an ST (see Section A.5), with one exception: the conformance statement.

288 The conformance statement in the PP states how STs and/or other PPs must conform to that PP. The PP author can select whether "exact", "strict" or "demonstrable" conformance is required.

289 The authors of PP/STs that subsequently claim conformance must then comply with the PP according to that conformance statement.

B.6 Security problem definition (APE_SPD)

290 This section is identical to the security problem definition section of an ST as described in Section A.6.

B.7 Security objectives (APE_OBJ)

291 This section is identical to the security objectives section of an ST as described in Section A.7.

B.8 Extended components definition (APE_ECD)

292 This section is identical to the extended components section of an ST as described in Section A.8.

B.9 Security requirements (APE_REQ)

293 This section is identical to the security requirements section of an ST as described in Section A.9. Note however that the rules for completing operations in a PP are slightly different from the rules for completing operations in an ST. This is described in more detail in Section C.4.

B.10 TOE summary specification

294 A PP has no TOE summary specification.

B.11 Low assurance Protection Profiles

295 A low assurance PP has the same relationship to a regular PP, as a low assurance ST has to a regular ST. This means that a low-assurance PP consists of

a) a PP introduction, consisting of a PP reference and a TOE overview;

b) a conformance claim;

c) security objectives for the operational environment;

d) the SFRs and the SARs (including the extended components definition).

296 A low assurance PP has similar reduced requirements for completeness as a low assurance ST (see Section A.12.2).

297 The reduced content of a low assurance PP is shown in Figure 12.

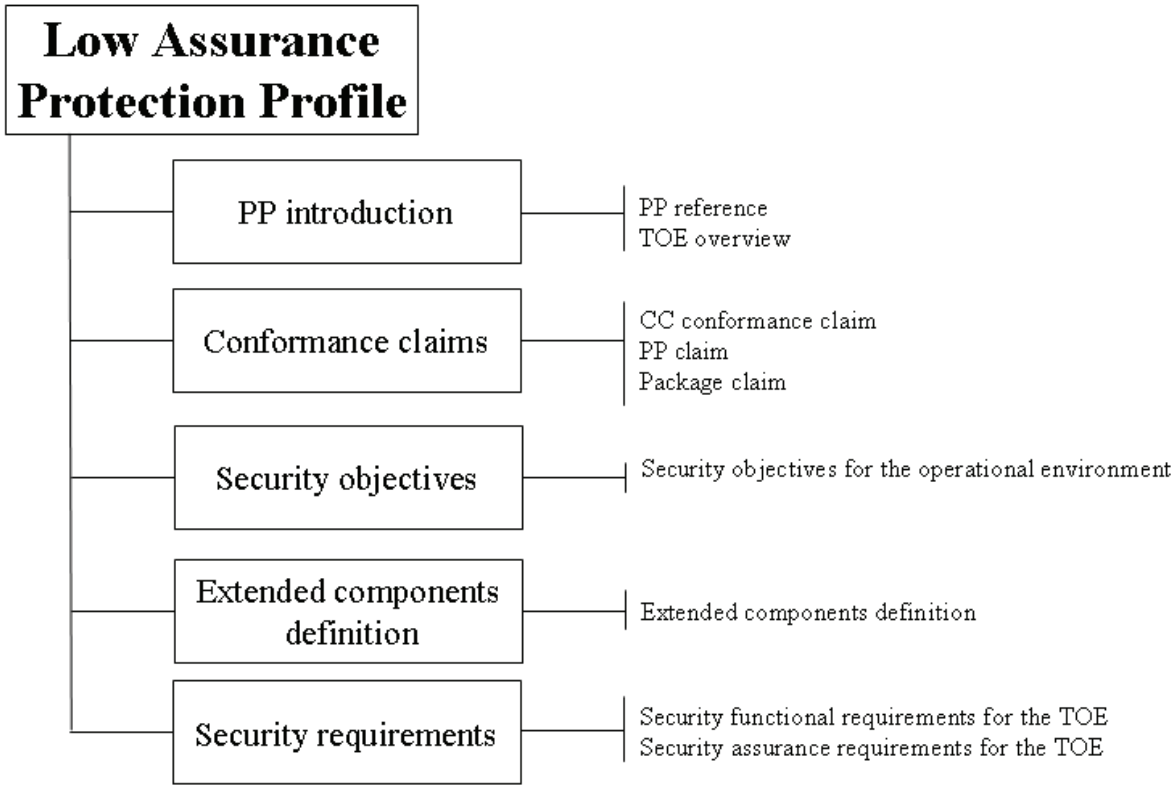

Figure 12 - Contents of a Low Assurance Protection Profile

C Security Requirements (normative)

C.1 Introduction

298 In the CC, packages, Protection Profiles and Security Targets contain security requirements. The CC has been developed around the central notion that these requirements are derived from:

– pre-defined security functional components that are listed in CC Part 2, and

– pre-defined security assurance components that are listed in CC Part 3.

299 These predefined components represent the preferred expression of security requirements as they are based on experience and represent a well-known and understood domain.

300 The components in CC Part 2 and CC Part 3 should be considered as pre-defined templates for SFRs and SARs, to be filled in and modified by operations in a PP, ST or package.

C.2 Organisation of components

301 The CC has organised the components in CC Part 2 and CC Part 3 into hierarchical structures:

– Classes, consisting of

– Families, consisting of

– Components, consisting of

– Elements.

302 This organisation into a hierarchy of class - family - component - element is provided to assist consumers, developers and evaluators in locating specific components.

303 The CC presents functional and assurance components in the same general hierarchical style and uses the same organisation and terminology for each.

C.2.1 Class

304 The term class is used for the most general grouping of security components. All the members of a class share a common general focus. An example of a class is the FIA class that is focused at identification of users, authentication

of users and binding of users and subjects. The members of a class are termed families.

C.2.2 Family

305 A family is a grouping of components that share a more specific focus but may differ in emphasis or rigour. An example of a family is the FIA_UAU family which is part of the FIA class. The FIA_UAU family concentrates on the authentication of users. The members of a family are termed components.

C.2.3 Component

306 A component is the smallest selectable unit in the CC. The set of components within a family may be ordered to represent increasing strength or capability. They may also be partially ordered to represent related non-hierarchical sets. In some instances, there is only one component in a family so ordering is not applicable. An example of a component is FIA_UAU.5 which concentrates on the conditions under which re-authentication of a user is required.

C.2.4 Element

307 The components are constructed from individual elements. The element is the lowest level expression of a security need that is verified by the evaluation. An example of an element is FAU_STG.3.1.

C.3 Dependencies between components

308 Dependencies may exist between components. Dependencies arise when a component is not self sufficient and relies upon the presence of another component.

309 The functional components in CC Part 2 have only dependencies on other functional components and the assurance components in CC Part 3 have only dependencies on other assurance components. However, this does not preclude extended functional components having dependencies on assurance components or vice versa.

310 Component dependency descriptions are part of the CC component definitions. In order to ensure completeness of the TOE security requirements, dependencies should be satisfied when requirements based on components with dependencies are incorporated into PPs and STs. Dependencies should also be considered when constructing packages.

311 In other words: if component A has a dependency on component B, this means that whenever a PP/ST contains a security requirement based on component A, the PP/ST shall also contain one of :

a) a security requirement based on component B, or

b) a security requirement based on a component that is hierarchical to B, or

c) a justification why the PP/ST does not contain a security requirement based on component B.

312 In cases a) and b), when a security requirement is included because of a dependency, it may be necessary to use operations on that security requirement to make sure that it actually satisfies the dependency.

313 In case c), the justification that a security requirement is not included should address either:

– why the dependency is not necessary or useful, or

– that the dependency has been addressed by the operational environment of the TOE, in which case the justification should describe how the security objectives for the operational environment address this dependency, or

– that the dependency has been addressed by the other SFRs in some other manner (extended SFRs, combinations of SFRs etc.)

314 An example of a valid justification that a dependency is not necessary is an ST that contains an SFR based on FDP_ACC.1 to specify access control based on attributes of subjects and objects. The ST author indicates that the dependency on FDP_ISA.1 is unnecessary, because in this particular TOE new objects and new subjects are never created. Specifying rules for the values of attributes for new subjects and objects is therefore unnecessary.

C.4 Operations

315 CC functional and assurance components may be used exactly as defined in the CC, or they may be tailored through the use of permitted operations in order to meet a security objective. When using operations, the PP/ST author must also be careful that the dependency needs of other requirements that depend on this requirement are satisfied. The permitted operations are selected from the following set:

– Iteration: allows a component to be used more than once with varying operations;

– Assignment: allows the specification of parameters;

– Selection: allows the specification of one or more items from a list; and

– Refinement: allows the addition of details.

316 The assignment and selection operations are permitted only where specifically indicated in a component. Iteration and refinement are permitted for all components. The operations are described in more detail below.

C.4.1 The iteration operation

317 The iteration operation can be performed on every component. The PP/ST author performs an iteration operation by including multiple requirements based on the same component. Each iteration of a component must be different from all other iterations of that component, which is realised by completing assignments and selections in a different way, or by applying refinements to it in a different way. An example of an iteration is FAU_ARP.1 being iterated twice to:

a) The TSF shall **backup the Database** upon detection of **the first** potential violation of the TSP.

b) The TSF shall **shutdown** upon detection of **the second** potential violation of the TSP.

318 Different iterations should be uniquely identified to allow clear rationales and tracings to and from these requirements.

C.4.2 The assignment operation

319 An assignment operation occurs where a given component contains an element with a parameter that may be set by the PP/ST author. The parameter may be an unrestricted variable, or a rule that narrows the variable to a specific range of values. An example of an element with an assignment is: FIA_AFL.1.2 “When the defined number of authentication attempts has been met or surpassed, the TSF shall **[assignment: list of actions]**.”

320 Whenever an element in a PP contains an assignment, a PP author may do one of three things:

a) leave the assignment uncompleted. The PP author could include FIA_AFL.1.2 “When the defined number of authentication attempts has been met or surpassed, the TSF shall **[assignment: list of actions]**.” in the PP.

b) complete the assignment. As an example, the PP author could include FIA_AFL.1.2 “When the defined number of authentication attempts has been met or surpassed, the TSF shall **prevent that user from binding to any subject in the future**.” in the PP

c) transform the assignment to a selection, thereby narrowing the assignment. As an example, the PP author could include FIA_AFL.1.2 “When the defined number of authentication attempts has been met or surpassed, the TSF shall **[selection: prevent that user from binding to any subject in the future, notify the administrator]**.” in the PP.

321 Whenever an element in an ST contains an assignment, an ST author must complete that assignment, as indicated in b) above. Options a) and c) are not allowed for STs.

322 The values chosen to in b) and c) above must comply with the indicated type required by the assignment.

C.4.3 The selection operation

323 The selection operation occurs where a given component contains an element where a choice from several items has to be made by the PP/ST author. An example of an element with a selection is: FPT_TST.1.1 "The TSF shall run a suite of self tests **[selection: immediately after installation, during each start-up, periodically during normal operation, at the request of a subject, [assignment: other conditions]]** to demonstrate the correct operations of **[selection: [assignment: parts of the TSF], the TSF]**."

324 Whenever an element in a PP contains a selection, the PP author may do one of three things:

a) leave the selection uncompleted. As an example, the PP author could include FPT_TST.1.1 "The TSF shall run a suite of self tests **immediately after installation, during each start-up, periodically during normal operation, at the request of a subject, [assignment: other conditions]]** to ...." in the PP.

b) complete the selection by choosing one or more items. As an example, the PP author could include FPT_TST.1.1 "The TSF shall run a suite of self tests **during each start-up and periodically during normal operation** to ...." in the PP.

c) restrict the selection by removing some of the choices, but leaving two or more. As an example, the PP author could include FPT_TST.1.1 "The TSF shall run a suite of self tests **[selection: during each start-up, periodically during normal operation]** to ...." in the PP.

325 Whenever an element in an ST contains a selection, an ST author must complete that selection, as indicated in b) above. Options a) and c) are not allowed for STs.

326 The item or items chosen in b) and c) must be taken from the items provided in the selection.

C.4.4 The refinement operation

327 The refinement operation can be performed on every requirement. The PP/ST author performs a refinement by altering that requirement. The main rule for a refinement is that it must not "weaken" the original requirement: a TOE meeting the refined requirement must also meet the unrefined requirement in the context of the PP/ST. If a requirement exceeds this boundary it is considered to be an extended requirement and must be treated as such.

328 In addition, a refinement should be related to the original component. For example, refining an audit component with an extra element on prevention of electromagnetic radiation is not allowed.

329 An example of a refinement is FIA_UAU.1.1 "The TSF shall authenticate a user before the user can bind to **FTP-handler**." being refined to "The TSF shall authenticate an **Internet** user before the user can bind to **FTP-handler**". If all users are Internet users, this is a valid refinement. If there are also users coming e.g. from the LAN, this would not be a valid refinement.

330 A special case of refinement is an editorial refinement, where a small change is made in a requirement, i.e. rephrasing a sentence due to adherence to proper English grammar. This change is not allowed to modify the meaning of the requirement in any way. An example of an editorial refinement is the requirement FAU_ARP.1 with a single action: "The TSF shall take **inform the operator** upon detection of a potential security violation" could be refined to: "The TSF shall **inform the operator** upon detection of a potential security violation".

331 Another special case of refinement is where multiple iterations of the same requirement are used, each with different refinements, where some of the refined iterations do not meet the full scope of the original requirement. This is acceptable, provided that all iterations of the refined requirement taken collectively, meet the entire scope of the original requirement.

332 An example of this is the requirement FIA_UAU.1.1 "The TSF shall authenticate a user before the user can bind to **FTP-handler**." being iterated and refined to "The TSF shall authenticate **by X.509v3 certificates** an **Internet** user before the user can bind to **FTP-handler**" and "The TSF shall authenticate **by username/password** a **LAN** user before the user can bind to **FTP-handler**"

C.5 Extended components

333 In the CC it is mandatory to base requirements on components from CC Part 2 or CC Part 3 with two exceptions:

a) there are security objectives for the TOE that can not be translated to Part 2 SFRs, or there are security objectives for the development environment that can not be translated to Part 3 SARs (e.g. strength of cryptographic algorithms);

b) a security objective can be translated, but only with great difficulty and/or complexity based on components in CC Part 2 and/or CC Part 3.

334 An example of this second case is already present in the CC in the form of FIA_AFL.1. This component can also be expressed with FAU_GEN.1, FAU_SAA.1 and FAU_ARP.1. FIA_AFL.1 could therefore be considered redundant, but was nevertheless included into CC Part 2 because it very clearly expresses a specific instance of the use of requirements that is often

used and provides a much clearer description than the combination of the three other components.

335 In both cases the PP/ST author is required to define his own components: new templates to base SFRs and SARs on. These newly defined components are called extended components. A precisely defined extended component is needed to provide context and meaning to the extended SFRs and SARs based on that component.

336 After the new components have been defined correctly, the PP/ST author can then base one or more SFRs or SARs on these newly defined extended components and use them in the same way as the other SFRs and SARs. From this point on, there is no further distinction between SARs and SFRs based on the CC and SARs and SFRs based on extended components.

C.5.1 How to define extended components

337 Whenever a PP/ST author defines an extended component, this has to be done in a similar manner to the existing CC components: clear, unambiguous and evaluable (it is possible to systematically demonstrate whether a requirement based on that component holds for a TOE). Extended components must use similar labelling, manner of expression, and level of detail as the existing CC components.

338 The PP/ST author also has to make to sure that all applicable dependencies of a extended component are included. Examples of possible dependencies are:

a) if an extended component refers to auditing, dependencies to components of the FAU class may have to be included;

b) if an extended component modifies or accesses data, dependencies to components of the FDP_ACC family may have to be included;

c) if an extended component uses a particular design description a dependency to the appropriate ADV family (e.g. Functional Specification) may have to be included.

339 In the case of an extended functional component, the PP/ST author also has to include any applicable audit and associated operations information, similar to existing CC Part 2 components. In the case of an extended assurance component, the PP/ST author also has to provide suitable methodology to "perform" the component, similar to the methodology provided in the CEM.

340 Extended components may be placed in existing families, in which case the PP/ST writer has to show how these families change. If they do not fit into an existing family, they shall be placed in a new family. New families have to be defined similarly to the CC.

341 New families may be placed in existing classes in which case the PP/ST writer has to show how these classes change. If they do not fit into an

existing class, they shall be placed in a new class. New classes have to be defined similarly to the CC.

D Bibliography (informative)

BL Bell, D. E. and LaPadula, L. J., Secure Computer Systems: Unified Exposition and MULTICS Interpretation, Revision 1, US Air Force ESD-TR-75-306, MITRE Corporation MTR-2997, Bedford MA, March 1976.

BIBA Biba, K. J., Integrity Considerations for Secure Computer Systems, ESD-TR-372, ESD/AFSC, Hanscom AFB, Bedford MA., April 1977.

BREW Brewer, D.F.C and Nash, M.J., The Chinese Wall Security Policy, IEEE Symposium on Research in Security and Privacy, 1989.

CTCPEC Canadian Trusted Computer Product Evaluation Criteria, Version 3.0, Canadian System Security Centre, Communications Security Establishment, Government of Canada, January 1993.

FC Federal Criteria for Information Technology Security, Draft Version 1.0, (Volumes I and II), jointly published by the National Institute of Standards and Technology and the National Security Agency, US Government, January 1993.

GOGU1 Goguen, J. A. and Meseguer, J., "Security Policies and Security Models," 1982 Symposium on Security and Privacy, pp.11-20, IEEE, April 1982.

GOGU2 Goguen, J. A. and Meseguer, J., "Unwinding and Inference Control," 1984 Symposium on Security and Privacy, pp.75-85, IEEE, May 1984.

ITSEC Information Technology Security Evaluation Criteria, Version 1.2, Office for Official Publications of the European Communities, June 1991.

OSI ISO/IEC 7498-2:1989 Information processing systems - Open Systems Interconnection - Basic Reference Model, Part 2: Security Architecture.

PPRP ISO/IEC 15292:2001 Information technology - Security techniques - Protection Profile registration procedures.

TCSEC Trusted Computer Systems Evaluation Criteria, US DoD 5200.28-STD, December 1985.