

Information technology — Security techniques — Evaluation criteria for IT Security

Part 1: Introduction and general model

ICS 35.040

National foreword

This British Standard is the UK implementation of ISO/IEC 15408-1:2009. It supersedes BS ISO/IEC 15408-1: 2005 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee IST/33, IT - Security techniques.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 January 2010

© BSI 2010

ISBN 978 0 580 59228 7

Amendments/corrigenda issued since publication

Date	Comments

Information technology — Security techniques — Evaluation criteria for IT security —

Part 1: Introduction and general model

Technologies de l'information — Techniques de sécurité — Critères d'évaluation pour la sécurité TI —

Partie 1: Introduction et modèle général



Reference number
ISO/IEC 15408-1:2009(E)

AVA **Documents**
Document Sharing Hub
© ISO/IEC 2009

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2009

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword	iv
Introduction.....	v
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
3.1 Terms and definitions common in ISO/IEC 15408	2
3.2 Terms and definitions related to the ADV class.....	9
3.3 Terms and definitions related to the AGD class	13
3.4 Terms and definitions related to the ALC class	13
3.5 Terms and definitions related to the AVA class.....	17
3.6 Terms and definitions related to the ACO class	17
4 Abbreviated terms	18
5 Overview.....	19
5.1 General	19
5.2 The TOE	19
5.3 Target audience of ISO/IEC 15408	20
5.4 The different parts of ISO/IEC 15408	21
5.5 Evaluation context.....	22
6 General model.....	23
6.1 Introduction to the general model	23
6.2 Assets and countermeasures	23
6.3 Evaluation.....	27
7 Tailoring Security Requirements	27
7.1 Operations.....	27
7.2 Dependencies between components	30
7.3 Extended components	30
8 Protection Profiles and Packages	31
8.1 Introduction.....	31
8.2 Packages	31
8.3 Protection Profiles.....	31
8.4 Using PPs and packages	34
8.5 Using Multiple Protection Profiles	35
9 Evaluation results.....	35
9.1 Introduction.....	35
9.2 Results of a PP evaluation.....	36
9.3 Results of an ST/TOE evaluation	36
9.4 Conformance claim	36
9.5 Use of ST/TOE evaluation results	37
Annex A (informative) Specification of Security Targets.....	38
Annex B (informative) Specification of Protection Profiles	54
Annex C (informative) Guidance for Operations.....	60
Annex D (informative) PP conformance	63
Bibliography.....	65

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this part of ISO/IEC 15408 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 15408-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *IT Security techniques*. The identical text of ISO/IEC 15408 is published by the Common Criteria Project Sponsoring Organisations as Common Criteria for Information Technology Security Evaluation. The common XML source for both publications can be found at <http://www.oc.ccn.cni.es/xml>

This third edition cancels and replaces the second edition (ISO/IEC 15408-1:2005), which has been technically revised.

ISO/IEC 15408 consists of the following parts, under the general title *Information technology — Security techniques — Evaluation criteria for IT security*:

- *Part 1: Introduction and general model*
- *Part 2: Security functional components*
- *Part 3: Security assurance components*

Introduction

This part of ISO/IEC 15408 permits comparability between the results of independent security evaluations. ISO/IEC 15408 does so by providing a common set of requirements for the security functionality of IT products and for assurance measures applied to these IT products during a security evaluation. These IT products may be implemented in hardware, firmware or software.

The evaluation process establishes a level of confidence that the security functionality of these IT 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.

ISO/IEC 15408 is useful as a guide for the development, evaluation and/or procurement of IT products with security functionality.

ISO/IEC 15408 is intentionally flexible, enabling a range of evaluation methods to be applied to a range of security properties of a range of IT products. Therefore users of this International Standard are cautioned to exercise care that this flexibility is not misused. For example, using ISO/IEC 15408 in conjunction with unsuitable evaluation methods, irrelevant security properties, or inappropriate IT products, may result in meaningless evaluation results.

Consequently, the fact that an IT product has been evaluated has meaning only in the context of the security properties that were evaluated and the evaluation methods that were used. Evaluation authorities are advised to carefully check the products, properties and methods to determine that an evaluation will provide meaningful results. Additionally, purchasers of evaluated products are advised to carefully consider this context to determine whether the evaluated product is useful and applicable to their specific situation and needs.

ISO/IEC 15408 addresses protection of assets 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. ISO/IEC 15408 may also be applicable to aspects of IT security outside of these three. ISO/IEC 15408 is applicable to risks arising from human activities (malicious or otherwise) and to risks arising from non-human activities. Apart from IT security, ISO/IEC 15408 may be applied in other areas of IT, but makes no claim of applicability in these areas.

Certain topics, because they involve specialized techniques or because they are somewhat peripheral to IT security, are considered to be outside the scope of ISO/IEC 15408. Some of these are identified below.

- a) ISO/IEC 15408 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 organizational, personnel, physical, and procedural controls.
- b) The evaluation of some 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.
- c) ISO/IEC 15408 does not address the evaluation methodology under which the criteria should be applied. This methodology is given in ISO/IEC 18045.
- d) ISO/IEC 15408 does not address the administrative and legal framework under which the criteria may be applied by evaluation authorities. However, it is expected that ISO/IEC 15408 will be used for evaluation purposes in the context of such a framework.
- e) The procedures for use of evaluation results in accreditation are outside the scope of ISO/IEC 15408. 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.

- f) The subject of criteria for the assessment of the inherent qualities of cryptographic algorithms is not covered in ISO/IEC 15408. Should independent assessment of mathematical properties of cryptography be required, the evaluation scheme under which ISO/IEC 15408 is applied must make provision for such assessments.

ISO terminology, such as “can”, “informative”, “may”, “normative”, “shall” and “should” used throughout the document are defined in the ISO/IEC Directives, Part 2. Note that the term “should” has an additional meaning applicable when using this International Standard. See the note below. The following definition is given for the use of “should” in ISO/IEC 15408.

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 Directives, Part 2)

NOTE ISO/IEC 15408 interprets “not necessarily required” to mean that the choice of another possibility requires a justification of why the preferred option was not chosen.

Information technology — Security techniques — Evaluation criteria for IT security —

Part 1: Introduction and general model

1 Scope

This part of ISO/IEC 15408 establishes the general concepts and principles of IT security evaluation and specifies the general model of evaluation given by various parts of the International Standard which in its entirety is meant to be used as the basis for evaluation of security properties of IT products.

It provides an overview of all parts of ISO/IEC 15408. It describes the various parts of the standard; defines the terms and abbreviations to be used in all parts of the International Standard; establishes the core concept of a Target of Evaluation (TOE); the evaluation context; and describes the audience to which the evaluation criteria are addressed. An introduction to the basic security concepts necessary for evaluation of IT products is given.

It defines the various operations by which the functional and assurance components given in ISO/IEC 15408-2 and ISO/IEC 15408-3 may be tailored through the use of permitted operations.

The key concepts of protection profiles (PP), packages of security requirements and the topic of conformance are specified and the consequences of evaluation and evaluation results are described. This part of ISO/IEC 15408 gives guidelines for the specification of Security Targets (ST) and provides a description of the organization of components throughout the model. General information about the evaluation methodology is given in ISO/IEC 18045 and the scope of evaluation schemes is provided.

2 Normative references

The following referenced documents are indispensable for the application of this part of ISO/IEC 15408. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 15408-2, *Information technology — Security techniques — Evaluation criteria for IT security — Part 2: Security functional components*

ISO/IEC 15408-3, *Information technology — Security techniques — Evaluation criteria for IT security — Part 3: Security assurance components*

ISO/IEC 18045, *Information technology — Security techniques — Methodology for IT security evaluation*

3 Terms and definitions

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

NOTE This clause contains only those terms which are used in a specialized way throughout ISO/IEC 15408. Some combinations of common terms used in ISO/IEC 15408, while not meriting inclusion in this clause, are explained for clarity in the context where they are used.

3.1 Terms and definitions common in ISO/IEC 15408

3.1.1

adverse actions

actions performed by a threat agent on an asset

3.1.2

assets

entities that the owner of the TOE presumably places value upon

3.1.3

assignment

specification of an identified parameter in a component (of ISO/IEC 15408) or requirement

3.1.4

assurance

grounds for confidence that a TOE meets the SFRs

3.1.5

attack potential

measure of the effort to be expended in attacking a TOE, expressed in terms of an attacker's expertise, resources and motivation

3.1.6

augmentation

addition of one or more requirement(s) to a package

3.1.7

authentication data

information used to verify the claimed identity of a user

3.1.8

authorised user

TOE user who may, in accordance with the SFRs, perform an operation

3.1.9

class

set of ISO/IEC 15408 families that share a common focus

3.1.10

coherent

logically ordered and having discernible meaning

NOTE For documentation, this addresses both the actual text and the structure of the document, in terms of whether it is understandable by its target audience.

3.1.11

complete

property where all necessary parts of an entity have been provided

NOTE In terms of documentation, this means that all relevant information is covered in the documentation, at such a level of detail that no further explanation is required at that level of abstraction.

3.1.12

component

smallest selectable set of elements on which requirements may be based

3.1.13

composed assurance package

assurance package consisting of requirements drawn from ISO/IEC 15408-3 (predominately from the ACO class), representing a point on ISO/IEC 15408 predefined composition assurance scale

3.1.14

confirm

declare that something has been reviewed in detail with an independent determination of sufficiency

NOTE The level of rigour required depends on the nature of the subject matter. This term is only applied to evaluator actions.

3.1.15

connectivity

property of the TOE allowing interaction with IT entities external to the TOE

NOTE This includes exchange of data by wire or by wireless means, over any distance in any environment or configuration.

3.1.16

consistent

relationship between two or more entities such that there are no apparent contradictions between these entities

3.1.17

counter, verb

meet an attack where the impact of a particular threat is mitigated but not necessarily eradicated

3.1.18

demonstrable conformance

relation between an ST and a PP, where the ST provides a solution which solves the generic security problem in the PP

NOTE The PP and the ST may contain entirely different statements that discuss different entities, use different concepts etc. Demonstrable conformance is also suitable for a TOE type where several similar PPs already exist, thus allowing the ST author to claim conformance to these PPs simultaneously, thereby saving work.

3.1.19

demonstrate

provide a conclusion gained by an analysis which is less rigorous than a "proof"

3.1.20

dependency

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

3.1.21

describe

provide specific details of an entity

3.1.22

determine

affirm a particular conclusion based on independent analysis with the objective of reaching a particular conclusion

NOTE The usage of this term implies a truly independent analysis, usually in the absence of any previous analysis having been performed. Compare with the terms "confirm" or "verify" which imply that an analysis has already been performed which needs to be reviewed

3.1.23

development environment

environment in which the TOE is developed

3.1.24

element

indivisible statement of a security need

3.1.25

ensure

guarantee a strong causal relationship between an action and its consequences

NOTE When this term is preceded by the word “help” it indicates that the consequence is not fully certain, on the basis of that action alone.

3.1.26

evaluation

assessment of a PP, an ST or a TOE, against defined criteria

3.1.27

evaluation assurance level

set of assurance requirements drawn from ISO/IEC 15408-3, representing a point on the ISO/IEC 15408 predefined assurance scale, that form an assurance package

3.1.28

evaluation authority

body that sets the standards and monitors the quality of evaluations conducted by bodies within a specific community and implements ISO/IEC 15408 for that community by means of an evaluation scheme

3.1.29

evaluation scheme

administrative and regulatory framework under which ISO/IEC 15408 is applied by an evaluation authority within a specific community

3.1.30

exhaustive

characteristic of a methodical approach taken to perform an analysis or activity according to an unambiguous plan

NOTE This term is used in ISO/IEC 15408 with respect to conducting an analysis or other activity. It is related to “systematic” but is considerably stronger, in that it indicates not only that a methodical approach has been taken to perform the analysis or activity according to an unambiguous plan, but that the plan that was followed is sufficient to ensure that all possible avenues have been exercised.

3.1.31

explain

give argument accounting for the reason for taking a course of action

NOTE This term differs from both “describe” and “demonstrate”. It is intended to answer the question “Why?” without actually attempting to argue that the course of action that was taken was necessarily optimal.

3.1.32

extension

addition to an ST or PP of functional requirements not contained in ISO/IEC 15408-2 and/or assurance requirements not contained in ISO/IEC 15408-3

3.1.33

external entity

human or IT entity possibly interacting with the TOE from outside of the TOE boundary

NOTE An external entity can also be referred to as a user.

3.1.34

family

set of components that share a similar goal but differ in emphasis or rigour

3.1.35

formal

expressed in a restricted syntax language with defined semantics based on well-established mathematical concepts

3.1.36

guidance documentation

documentation that describes the delivery, preparation, operation, management and/or use of the TOE

3.1.37

identity

representation uniquely identifying entities (e.g. a user, a process or a disk) within the context of the TOE

NOTE An example of such a representation is a string. For a human user, the representation can be the full or abbreviated name or a (still unique) pseudonym.

3.1.38

informal

expressed in natural language

3.1.39

inter TSF transfers

communicating data between the TOE and the security functionality of other trusted IT products

3.1.40

internal communication channel

communication channel between separated parts of the TOE

3.1.41

internal TOE transfer

communicating data between separated parts of the TOE

3.1.42

internally consistent

no apparent contradictions exist between any aspects of an entity

NOTE In terms of documentation, this means that there can be no statements within the documentation that can be taken to contradict each other.

3.1.43

iteration

use of the same component to express two or more distinct requirements

3.1.44

justification

analysis leading to a conclusion

NOTE "Justification" is more rigorous than a demonstration. This term requires significant rigour in terms of very carefully and thoroughly explaining every step of a logical argument.

3.1.45

object

passive entity in the TOE, that contains or receives information, and upon which subjects perform operations

3.1.46

operation

⟨on a component of ISO/IEC 15408⟩ modification or repetition of a component

NOTE Allowed operations on components are assignment, iteration, refinement and selection.

3.1.47

operation

⟨on an object⟩ specific type of action performed by a subject on an object

3.1.48

operational environment

environment in which the TOE is operated

3.1.49

organizational security policy

set of security rules, procedures, or guidelines for an organization

NOTE A policy may pertain to a specific operational environment.

3.1.50

package

named set of either security functional or security assurance requirements

NOTE An example of a package is “EAL 3”.

3.1.51

Protection Profile evaluation

assessment of a PP against defined criteria

3.1.52

Protection Profile

implementation-independent statement of security needs for a TOE type

3.1.53

prove

show correspondence by formal analysis in its mathematical sense

NOTE It is completely rigorous in all ways. Typically, “prove” is used when there is a desire to show correspondence between two TSF representations at a high level of rigour.

3.1.54

refinement

addition of details to a component

3.1.55

role

predefined set of rules establishing the allowed interactions between a user and the TOE

3.1.56

secret

information that must be known only to authorised users and/or the TSF in order to enforce a specific SFP

3.1.57

secure state

state in which the TSF data are consistent and the TSF continues correct enforcement of the SFRs

3.1.58

security attribute

property of subjects, users (including external IT products), objects, information, sessions and/or resources that is used in defining the SFRs and whose values are used in enforcing the SFRs

3.1.59

security function policy

set of rules describing specific security behaviour enforced by the TSF and expressible as a set of SFRs

3.1.60

security objective

statement of an intent to counter identified threats and/or satisfy identified organization security policies and/or assumptions

3.1.61

security problem

statement which in a formal manner defines the nature and scope of the security that the TOE is intended to address

NOTE This statement consists of a combination of:

- threats to be countered by the TOE,
- the OSPs enforced by the TOE, and
- the assumptions that are upheld for the TOE and its operational environment.

3.1.62

security requirement

requirement, stated in a standardized language, which is meant to contribute to achieving the security objectives for a TOE

3.1.63

Security Target

ST

implementation-dependent statement of security needs for a specific identified TOE

3.1.64

selection

specification of one or more items from a list in a component

3.1.65

semiformal

expressed in a restricted syntax language with defined semantics

3.1.66

specify

provide specific details about an entity in a rigorous and precise manner

3.1.67

strict conformance

hierarchical relationship between a PP and an ST where all the requirements in the PP also exist in the ST

NOTE This relation can be roughly defined as “the ST shall contain all statements that are in the PP, but may contain more”. Strict conformance is expected to be used for stringent requirements that are to be adhered to in a single manner.

3.1.68

ST evaluation

assessment of an ST against defined criteria

3.1.69

subject

active entity in the TOE that performs operations on objects

3.1.70

TOE

target of evaluation

set of software, firmware and/or hardware possibly accompanied by guidance

3.1.71

threat agent

entity that can adversely act on assets

3.1.72

TOE evaluation

assessment of a TOE against defined criteria

3.1.73

TOE resource

anything useable or consumable in the TOE

3.1.74

TOE security functionality

combined functionality of all hardware, software, and firmware of a TOE that must be relied upon for the correct enforcement of the SFRs

3.1.75

trace, verb

perform an informal correspondence analysis between two entities with only a minimal level of rigour

3.1.76

transfers outside of the TOE

TSF mediated communication of data to entities not under the control of the TSF

3.1.77

translation

describes the process of describing security requirements in a standardized language

NOTE Use of the term translation in this context is not literal and does not imply that every SFR expressed in standardized language can also be translated back to the security objectives.

3.1.78

trusted channel

a means by which a TSF and another trusted IT product can communicate with necessary confidence

3.1.79

trusted IT product

IT product, other than the TOE, which has its security functional requirements administratively coordinated with the TOE and which is assumed to enforce its security functional requirements correctly

NOTE An example of a trusted IT product would be one that has been separately evaluated.

3.1.80

trusted path

means by which a user and a TSF can communicate with the necessary confidence

3.1.81

TSF data

data for the operation of the TOE upon which the enforcement of the SFR relies

3.1.82

TSF interface

means by which external entities (or subjects in the TOE but outside of the TSF) supply data to the TSF, receive data from the TSF and invoke services from the TSF

3.1.83

user data

data for the user, that does not affect the operation of the TSF

3.1.84

verify

rigorously review in detail with an independent determination of sufficiency

NOTE Also see “confirm” (3.1.14). The term “verify” has more rigorous connotations. It is used in the context of evaluator actions where an independent effort is required of the evaluator.

3.2 Terms and definitions related to the ADV class

NOTE The following terms are used in the requirements for software internal structuring. Some of these are derived from the IEEE Std 610.12-1990, *Standard glossary of software engineering terminology*, Institute of Electrical and Electronics Engineers.

3.2.1

administrator

entity that has a level of trust with respect to all policies implemented by the TSF

NOTE Not all PPs or STs assume the same level of trust for administrators. Typically administrators are assumed to adhere at all times to the policies in the ST of the TOE. Some of these policies may be related to the functionality of the TOE, others may be related to the operational environment.

3.2.2

call tree

identifies the modules in a system in diagrammatic form showing which modules call one another

NOTE Adapted from IEEE Std 610.12-1990.

3.2.3

cohesion

module strength

manner and degree to which the tasks performed by a single software module are related to one another

[IEEE Std 610.12-1990]

NOTE Types of cohesion include coincidental, communicational, functional, logical, sequential, and temporal. These types of cohesion are described by the relevant term entry.

3.2.4

coincidental cohesion

module with the characteristic of performing unrelated, or loosely related, activities

[IEEE Std 610.12-1990]

NOTE See also cohesion (3.2.3).

3.2.5

communicational cohesion

module containing functions that produce output for, or use output from, other functions within the module

[IEEE Std 610.12-1990]

NOTE 1 See also cohesion (3.2.3).

NOTE 2 An example of a communicationally cohesive module is an access check module that includes mandatory, discretionary, and capability checks.

3.2.6

complexity

measure of how difficult software is to understand, and thus to analyse, test, and maintain

[IEEE Std 610.12-1990]

NOTE Reducing complexity is the ultimate goal for using modular decomposition, layering and minimization. Controlling coupling and cohesion contributes significantly to this goal.

A good deal of effort in the software engineering field has been expended in attempting to develop metrics to measure the complexity of source code. Most of these metrics use easily computed properties of the source code, such as the number of operators and operands, the complexity of the control flow graph (cyclomatic complexity), the number of lines of source code, the ratio of comments to executable code, and similar measures. Coding standards have been found to be a useful tool in generating code that is more readily understood.

The TSF internals (ADV_INT) family calls for a complexity analysis in all components. It is expected that the developer will provide support for the claims that there has been a sufficient reduction in complexity. This support could include the developer's programming standards, and an indication that all modules meet the standard (or that there are some exceptions that are justified by software engineering arguments). It could include the results of tools used to measure some of the properties of the source code, or it could include other support that the developer finds appropriate.

3.2.7

coupling

manner and degree of interdependence between software modules

[IEEE Std 610.12-1990]

NOTE Types of coupling include call, common and content coupling. These are characterised below.

3.2.8

call coupling

relationship between two modules communicating strictly through their documented function calls

NOTE Examples of call coupling are data, stamp and control.

3.2.9

call coupling

⟨data⟩ relationship between two modules communicating strictly through the use of call parameters that represent single data items

NOTE See also call coupling (3.2.8).

3.2.10

call coupling

⟨stamp⟩ relationship between two modules communicating through the use of call parameters that comprise multiple fields or that have meaningful internal structures

NOTE See also call coupling (3.2.8).

3.2.11

call coupling

(control) relationship between two modules if one passes information that is intended to influence the internal logic of the other

NOTE See also call coupling (3.2.8).

3.2.12

common coupling

relationship between two modules sharing a common data area or other common system resource

NOTE Global variables indicate that modules using those global variables are common coupled. Common coupling through global variables is generally allowed, but only to a limited degree.

For example, variables that are placed into a global area, but are used by only a single module, are inappropriately placed, and should be removed. Other factors that need to be considered in assessing the suitability of global variables are:

The number of modules that modify a global variable: In general, only a single module should be allocated the responsibility for controlling the contents of a global variable, but there may be situations in which a second module may share that responsibility; in such a case, sufficient justification must be provided. It is unacceptable for this responsibility to be shared by more than two modules. (In making this assessment, care should be given to determining the module actually responsible for the contents of the variable; for example, if a single routine is used to modify the variable, but that routine simply performs the modification requested by its caller, it is the calling module that is responsible, and there may be more than one such module). Further, as part of the complexity determination, if two modules are responsible for the contents of a global variable, there should be clear indications of how the modifications are coordinated between them.

The number of modules that reference a global variable: Although there is generally no limit on the number of modules that reference a global variable, cases in which many modules make such a reference should be examined for validity and necessity.

3.2.13

content coupling

relationship between two modules where one makes direct reference to the internals of the other

NOTE Examples include modifying code of, or referencing labels internal to, the other module. The result is that some or all of the content of one module are effectively included in the other. Content coupling can be thought of as using unadvertised module interfaces; this is in contrast to call coupling, which uses only advertised module interfaces.

3.2.14

domain separation

security architecture property whereby the TSF defines separate security domains for each user and for the TSF and ensures that no user process can affect the contents of a security domain of another user or of the TSF

3.2.15

functional cohesion

functional property of a module which performs activities related to a single purpose

[IEEE Std 610.12-1990]

NOTE A functionally cohesive module transforms a single type of input into a single type of output, such as a stack manager or a queue manager. See also cohesion (3.2.3).

3.2.16

interaction

general communication-based activity between entities

3.2.17

interface

means of interaction with a component or module

3.2.18
layering

design technique where separate groups of modules (the layers) are hierarchically organized to have separate responsibilities such that one layer depends only on layers below it in the hierarchy for services, and provides its services only to the layers above it

NOTE Strict layering adds the constraint that each layer receives services only from the layer immediately beneath it, and provides services only to the layer immediately above it.

3.2.19
logical cohesion
procedural cohesion

characteristics of a module performing similar activities on different data structures

NOTE A module exhibits logical cohesion if its functions perform related, but different, operations on different inputs. See also cohesion (3.2.3).

3.2.20
modular decomposition

process of breaking a system into components to facilitate design, development and evaluation

[IEEE Std 610.12-1990]

3.2.21
non-bypassability

(of the TSF) security architecture property whereby all SFR-related actions are mediated by the TSF

3.2.22
security domain

collection of resources to which an active entity has access privileges

3.2.23
sequential cohesion

module containing functions each of whose output is input for the following function in the module

[IEEE Std 610.12-1990]

NOTE An example of a sequentially cohesive module is one that contains the functions to write audit records and to maintain a running count of the accumulated number of audit violations of a specified type.

3.2.24
software engineering

application of a systematic, disciplined, quantifiable approach to the development and maintenance of software; that is, the application of engineering to software

[IEEE Std 610.12-1990]

NOTE As with engineering practices in general, some amount of judgement must be used in applying engineering principles. Many factors affect choices, not just the application of measures of modular decomposition, layering, and minimization. For example, a developer may design a system with future applications in mind that will not be implemented initially. The developer may choose to include some logic to handle these future applications without fully implementing them; further, the developer may include some calls to as-yet unimplemented modules, leaving call stubs. The developer's justification for such deviations from well-structured programs will have to be assessed using judgement, as well as the application of good software engineering discipline.

3.2.25

temporal cohesion

characteristics of a module containing functions that need to be executed at about the same time

NOTE 1 Adapted from [IEEE Std 610.12-1990].

NOTE 2 Examples of temporally cohesive modules include initialization, recovery, and shutdown modules.

3.2.26

TSF self-protection

security architecture property whereby the TSF cannot be corrupted by non-TSF code or entities

3.3 Terms and definitions related to the AGD class

3.3.1

installation

procedure performed by a human user embedding the TOE in its operational environment and putting it into an operational state

NOTE This operation is normally performed only once, after receipt and acceptance of the TOE. The TOE is expected to be progressed to a configuration allowed by the ST. If similar processes have to be performed by the developer they are denoted as “generation” throughout ALC: Life-cycle support. If the TOE requires an initial start-up that does not need to be repeated regularly, this process would be classified as installation.

3.3.2

operation

usage phase of the TOE including “normal usage”, administration and maintenance of the TOE after delivery and preparation

3.3.3

preparation

activity in the life-cycle phase of a product, comprising the customer's acceptance of the delivered TOE and its installation which may include such things as booting, initialization, start-up and progressing the TOE to a state ready for operation

3.4 Terms and definitions related to the ALC class

3.4.1

acceptance criteria

criteria to be applied when performing the acceptance procedures (e.g. successful document review, or successful testing in the case of software, firmware or hardware)

3.4.2

acceptance procedures

procedures followed in order to accept newly created or modified configuration items as part of the TOE, or to move them to the next step of the life-cycle

NOTE These procedures identify the roles or individuals responsible for the acceptance and the criteria to be applied in order to decide on the acceptance.

There are several types of acceptance situations some of which may overlap:

- a) acceptance of an item into the configuration management system for the first time, in particular inclusion of software, firmware and hardware components from other manufacturers into the TOE (“integration”);
- b) progression of configuration items to the next life-cycle phase at each stage of the construction of the TOE (e.g. module, subsystem, quality control of the finished TOE);
- c) subsequent to transports of configuration items (for example parts of the TOE or preliminary products) between different development sites;
- d) subsequent to the delivery of the TOE to the consumer.

3.4.3 **configuration management** **CM**

discipline applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status and verify compliance with specified requirements

NOTE Adapted from IEEE Std 610.12.

3.4.4 **CM documentation**

all CM documentation including CM output, CM list (configuration list), CM system records, CM plan and CM usage documentation

3.4.5 **configuration management evidence**

everything that may be used to establish confidence in the correct operation of the CM system

NOTE For example, CM output, rationales provided by the developer, observations, experiments or interviews made by the evaluator during a site visit.

3.4.6 **configuration item**

object managed by the CM system during the TOE development

NOTE These may be either parts of the TOE or objects related to the development of the TOE like evaluation documents or development tools. CM items may be stored in the CM system directly (for example files) or by reference (for example hardware parts) together with their version.

3.4.7 **configuration list**

configuration management output document listing all configuration items for a specific product together with the exact version of each configuration management item relevant for a specific version of the complete product

NOTE This list allows distinguishing the items belonging to the evaluated version of the product from other versions of these items belonging to other versions of the product. The final configuration management list is a specific document for a specific version of a specific product. (Of course the list can be an electronic document inside of a configuration management tool. In that case it can be seen as a specific view into the system or a part of the system rather than an output of the system. However, for the practical use in an evaluation the configuration list will probably be delivered as a part of the evaluation documentation.) The configuration list defines the items that are under the configuration management requirements of ALC_CMC.

3.4.8 **configuration management output**

results, related to configuration management, produced or enforced by the configuration management system

NOTE These configuration management related results could occur as documents (for example filled paper forms, configuration management system records, logging data, hard-copies and electronic output data) as well as actions (for example manual measures to fulfil configuration management instructions). Examples of such configuration management outputs are configuration lists, configuration management plans and/or behaviours during the product life-cycle.

3.4.9 **configuration management plan**

description of how the configuration management system is used for the TOE

NOTE The objective of issuing a configuration management plan is that staff members can see clearly what they have to do. From the point of view of the overall configuration management system this can be seen as an output document (because it may be produced as part of the application of the configuration management system). From the point of view of the concrete project it is a usage document because members of the project team use it in order to understand the steps that they have to perform during the project. The configuration management plan defines the usage

of the system for the specific product; the same system may be used to a different extent for other products. That means the configuration management plan defines and describes the output of the configuration management system of a company which is used during the TOE development.

3.4.10

configuration management system

set of procedures and tools (including their documentation) used by a developer to develop and maintain configurations of their products during their life-cycles

NOTE Configuration management systems may have varying degrees of rigour and function. At higher levels, configuration management systems may be automated, with flaw remediation, change controls, and other tracking mechanisms.

3.4.11

configuration management system records

output produced during the operation of the configuration management system documenting important configuration management activities

NOTE Examples of configuration management system records are configuration management item change control forms or configuration management item access approval forms.

3.4.12

configuration management tools

manually operated or automated tools realising or supporting a configuration management system

NOTE For example tools for the version management of the parts of the TOE.

3.4.13

configuration management usage documentation

part of the configuration management system which describes how the configuration management system is defined and applied by using, for example, handbooks, regulations and/or documentation of tools and procedures

3.4.14

delivery

transmission of the finished TOE from the production environment into the hands of the customer

NOTE This product life-cycle phase may include packaging and storage at the development site, but does not include transportations of the unfinished TOE or parts of the TOE between different developers or different development sites.

3.4.15

developer

organization responsible for the development of the TOE

3.4.16

development

product life-cycle phase which is concerned with generating the implementation representation of the TOE

NOTE Throughout the ALC requirements, development and related terms (developer, develop) are meant in the more general sense to comprise development and production.

3.4.17

development tools

tools (including test software, if applicable) supporting the development and production of the TOE

NOTE For example, for a software TOE, development tools are usually programming languages, compilers, linkers and generating tools.