# RELATING FUNCTIONALITY CLASS AND SECURITY SUB-PROFILE SPECIFICATIONS


## ABSTRACT

This document is a deliverable item from Joint Task 1 (JT01) whose terms of reference are defined in the Joint Workplan for cooperation on Security of Information Systems [1]. The objectives of JT01 are to:

(a) Establish a common set of security functionality classes, representative of international and regional market-driven needs.

(b) Develop a common approach to the creation of profiles from these security functionality classes consistent with current regional and international activities.

(c) Create guidelines to support the prototyping of such profiles and their interpretability.

This document describes methods for relating security functionality classes being proposed by various security evaluation criteria standardization efforts and security sub-profile specifications stemming from profiling of Open Systems standards. It is intended to fulfil item "b" of the JT01 workplan, "Description of the approach for expressing functionality classes with regard to OSE standards."


## 1. INTRODUCTION

This document contains a discussion of various alternatives for associating functionality class and security sub-profile specifications. The underlying problem to be addressed is that both forms of specification describe security functionality albeit in different ways. Functionality class specifications arise from recent efforts to improve the security evaluation criteria for computer systems [2, 3,4]. As such, they are broad specifications fashioned to be independent of specific mechanisms, and tend to be oriented toward stand alone (i.e., end-system) operations.

Security sub-profile specifications, on the other hand, stem from Open Systems standardization efforts. The best example of these specifications come from International Workshops attempts to assemble the Open Systems Interconnection (OSI) standards into meaningful profiles. The profile specifications tend to be fairly detailed, to embrace specific mechanisms, and, because of the interconnection aspects, to be concerned with distributed operations. Security sub-profiles refer to those parts of the profile that describe security functionality.

Unfortunately, coordination between the communities doing each form of specification has been limited. Work has proceeded on the premise that neither form of security functionality specification interferes with the other. While in general this may be true, it is not clear that it is true for every case. For example, the Defense Intelligence Agency's Compartmented Mode Workstation Evaluation Criteria [5] can be considered the result of requirements for a B1 level workstation exhibiting Open Systems standards such as the X Window System [6].

Therefore, the following aspects must be considered when formulating or selecting a method of association:

(a) While functionality class specifications tend to be at a high level of abstraction, security sub-profiles, although abstract in nature (algorithm independent), include much greater detail (e.g., protocol mechanisms).

(b) Functionality class specifications tend to be oriented toward stand alone operating system environments, while security sub-profiles are more often concerned with distributed system environments.

(c) Functionality class specifications and security sub-profiles may be developed independently, by different groups, at different points in time.

(d) There is limited experience with developing either security sub-profiles of Open System standards, or functionality class specifications following recent evaluation criteria, on which to base decisions.

The remainder of the document considers methods to express associations between functionality class and security sub-profiles specifications. The intent is to explore the range of possible alternatives and discuss their pros and cons, but not to make a final choice of method. That judgement is the subject of another JT01 deliverable.

## 2. ASSOCIATION BY DIRECT REFERENCE

This alternative for associating function class and security sub-profile specifications is the simplest. It maintains an independence between the two forms of specification, and merely relies on a reference to indicate the intended binding between them. This alternative is further divided based on the specificity and direction of the reference. Figure 1 illustrates the range of possibilities described below.

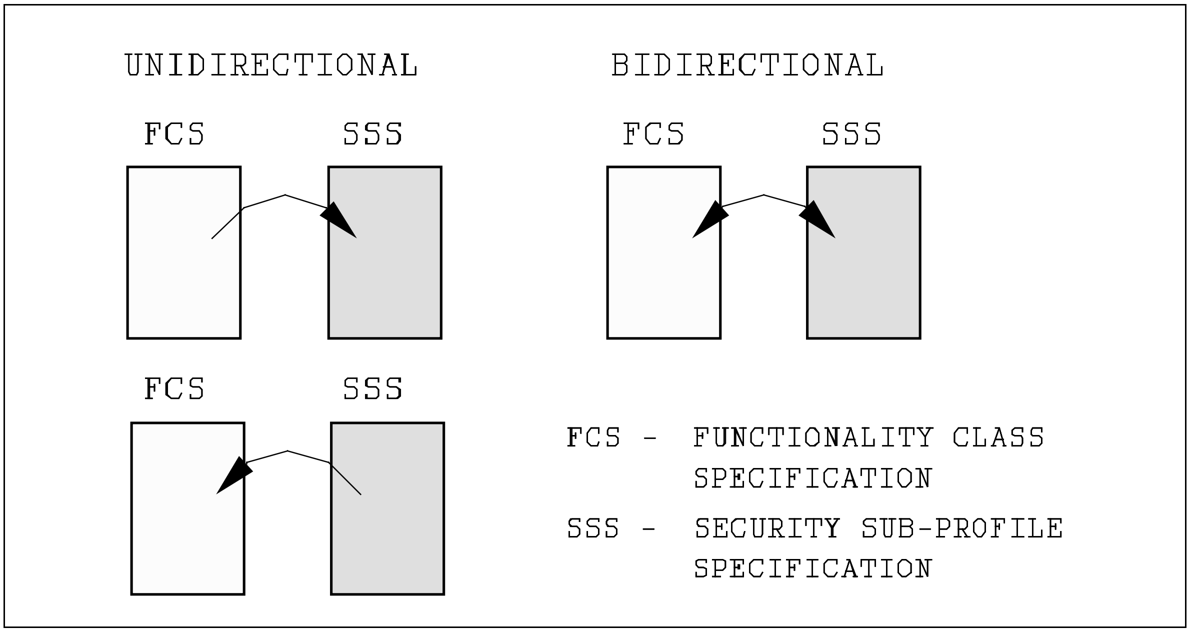


**Figure 1**: Direct Reference Possibilities

## 2.1 From a Functionality Class

This alternative employs a reference from a functionality class specification to one or more security sub-profiles that may be applicable in either a mutually exclusive or collective manner. Since the direction of the reference is conceptually from the general to the specific, the alternative is a viable one. The proposed "Mandatory Characteristics" functionality class component would be a possible choice for locating the reference. One drawback to using this component is that the relationship is quite vague. That is, the particular set of functionality "building blocks" within the specification, to which this external reference applies, is not identified.

To express a more explicit relationship an external reference to a sub-profile should occur from a "grouping" of security enforcing functions. For example, the ITSEC [2] recommends eight generic headings as natural groupings. A "Mandatory Characteristics" component could be added immediately after each heading to indicate an external reference. The recommended "Data Exchange" heading would be appropriate for references to the current set of security sub-profiles developed for OSI. However, it is not clear that the recommended generic headings would be appropriate in all situations.

## 2.2 From a Functionality Package

This alternative is similar to that immediately above. The difference here is the supposition that functionality class descriptions are composed of functional packages, and at least one package matches the functionality of a security sub-profile. A functional package is defined in the U.S. Federal Criteria [4] as a "Grouping of functional components assembled to ease specification and common understanding of what an IT product is capable of doing." A component of the package could contain a reference to the matching security profile. This approach allows security enforcing functions to be grouped meaningfully and external references to be made explicit. However, it requires further development and application of the packaging concept, and the ability to determine areas of security sub-profile functionality to package.

## 2.3 From a Security Sub-Profile

This alternative transfers the burden of reference from the functionality class specification to the security sub-profile. The approach requires Open Systems profile developers to be aware of functionality classes that are registered or under development, and to reference one or more that are applicable to the profile. Because the direction of the reference is from the specific to the general, the alternative may be more appropriate for certain types of functionality class specifications. For example, functionality class specifications that stipulate incremental security functionality with regard to an existing functionality class may be a likely candidate.

## 2.4 Reciprocal

This alternative employs bi-directional references as opposed to the uni-directional references used in the previous alternatives. This approach implies a cooperative development of the two forms of functionality specifications, as well as reciprocating references to one another. A new paradigm for security functionality specification, organized around a common model of distributed secure computing, underpins the approach.

# 3. ASSOCIATION BY THIRD-PARTY REFERENCE

Because of the potentially large number of both functionality class and security sub-profile specifications direct references from one to the other may reach a point of diminishing return. This is in part due to the dual nature of functionality class specifications. Some specifications, such as the counterparts to the TCSEC [7] classes (C2, B1, B2, B3, A1, A2), are intended to provide a foundation for secure computing. Others, such as that for a Database Management System, are more enhancement oriented, intended to build upon and extend the foundation functionality class specifications. For the remainder of the document, these different types of functionality class specifications are referred to as foundation and delta functionality class specifications (FCS).

For delta functionality class specifications, association by direct reference seems suitable. The special purpose of a delta FCS may be relevant to a specific well-known set of Open Systems standards, or developed with one in mind. Foundation FCS, on the other hand, may be too general purpose to directly reference or be directly referenced by all relevant security sub-profiles of Open Systems standards. This is especially true when considering the concurrent development of both forms of specifications. The alternative suggested for these situations is to reference pertinent functionality class and security sub-profiles specifications in a third type of specification whose purpose is requirements integration.

The security target is an existing form of specification intended to define the security enforcing functions against which a product or system will be evaluated and to describe the operating environment. It is therefore an appropriate choice for integrating security functionality specifications. One advantage of using the security target is that it already accommodates the differing levels of abstraction through the categories of security enforcing functions and security mechanisms. That is, functionality class specifications pertain to security enforcing functions, while security sub-profiles pertain to security mechanisms. Figure 2 illustrates this alternative.

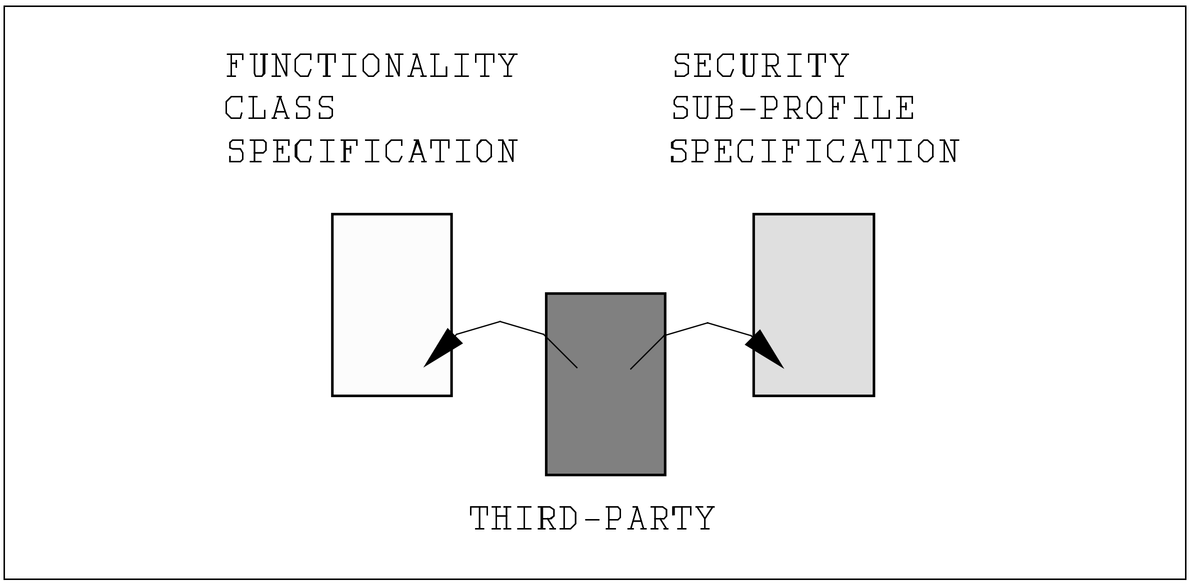


**Figure 2**: Third-Party Reference

## 4. ASSOCIATION BY SPECIALIZATION

As opposed to inter-specification references, this alternative involves incorporating the details of the security sub-profile directly into the functionality class description. The approach requires refinement of security functionality building blocks contained in a functionality class description to include the details of the associated security sub-profile, security services and mechanisms. Figure 3 illustrates this alternative. The concept of refinement is described in further detail in the U.S. Federal Criteria document [4].

One obvious drawback to this alternative is that the functionality class becomes tailored toward a specific Open Systems standard and looses its general applicability. However, the approach may prove useful to security sub-profile developers that desire to promote a particular profile with regard to a commercially accepted foundation FCS.

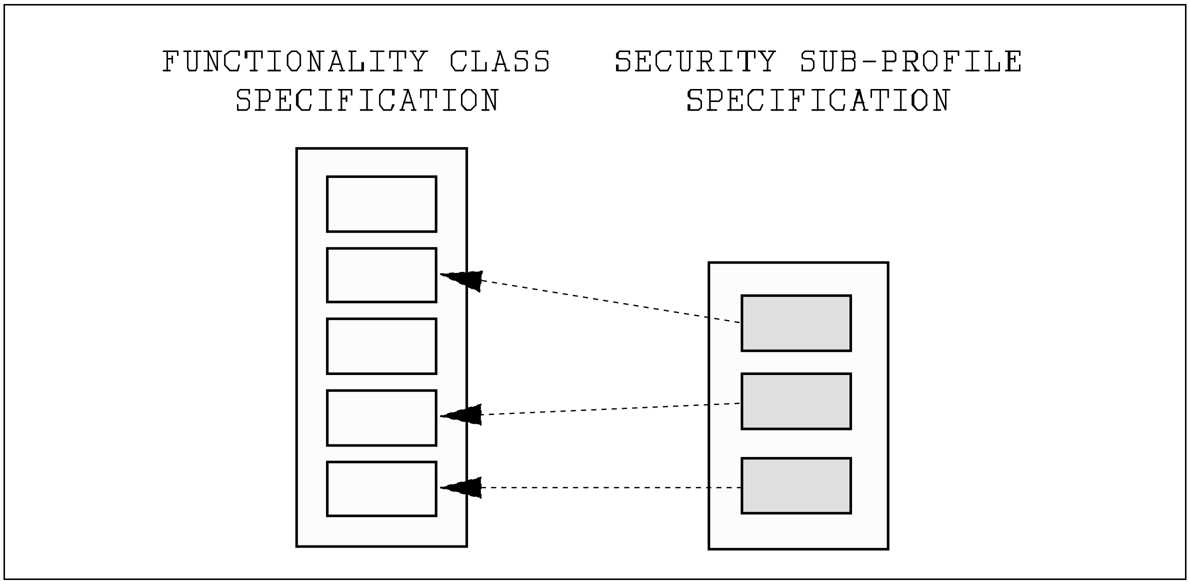


**Figure 3**: Specialization

## 5. ASSOCIATION BY GENERALIZATION

This alternative is the reverse of specialization described immediately above. The approach requires summarization of security sub-profile services and mechanisms into broad functionality class statements, including security enforcing functions suitable for a functionality building block. Figure 4 illustrates this alternative. As the figure suggests, the derived functionality class specification serves as a kind of textual icon for the associated security sub-profile.

The concept of synthesizing and abstracting functionality to a more general level is outside the scope of recent criteria documents. Still, some comments are possible. An apparent drawback to this alternative is that a generalization of a security sub-profile may be applicable to other security sub-profiles as well, because of the high level of abstraction possible. The intended binding between security sub-profile and FCS could be lost unless the sub-profile is included as a refinement of the functionality building block. The process of generalization may be beneficial to criteria standards for deriving complete sets of functionality building blocks. At a minimum, the generalization of security mechanisms known and deployed provide a measure of coverage by a criteria development standard.

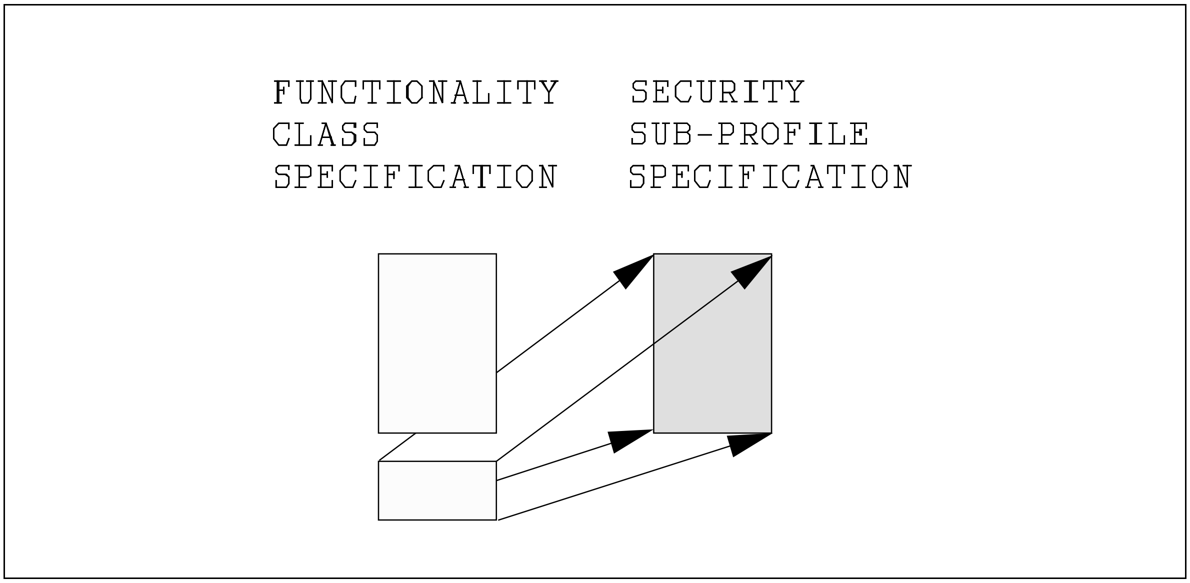

**Figure 4**: Generalization

## 6. SUMMARY

This document describes several methods for relating security functionality classes and security sub-profile specifications. They are as follows:

(a) Direct Reference

(b) Third-Party Reference

(c) Specialization

(d) Generalization

Each method has its shortcomings, but may be suitable in a specific situation. The use of a third-party reference is the most general scheme. The ITSEC and other recent criteria standards incorporate the notion of a security target, which is considered an appropriate vehicle for third-party reference. The document also makes an important distinction between two common uses of functionality class specifications as either foundation or delta type of specifications.

# REFERENCES

[1] Joint Workplan for EC/US Cooperation on Security of Information Systems, Edition 1, Brussels, February 1992.

[2] Information Technology Security Evaluation Criteria (ITSEC), Version 1.2, June 1991.

[3] The Canadian Trusted Computer Product Evaluation Criteria (CTPEC), Version 3.0e, April 1992.

[4] Federal Criteria for Information Technology Security, Version 1.0, December 1992.

[5] Compartmented Mode Workstation Evaluation Criteria, Version 1 (Final), Defense Intelligence Agency, DDS-2600-6243-91, November 1991

[6] Rosenthal, D. S., X Window System, Version 11, Inter-Client Communication Conventions Manual, Version 1.0, 1989.

[7] Department of Defense (DoD) Trusted Computer System Evaluation Criteria (TCSEC), DoD 5200.28-STD, December 1985.