Australian Government
Department of Defence

Defence Signals Directorate

Australasian Information Security Evaluation Program

Formal Correspondence Demonstration (ADV_RCR.3) - CC V2.2

Common Evaluation Methodology

15 February 2006

Version 1.1

Amendment Record

Version	Date	Description
1.0	12 September 2005	Released
1.1	15 February 2006	Releasable to AISEFs

Table of Contents

1 Formal Correspondence Demonstration (ADV_RCR.3)

1.1 Objectives

1 The objective of this sub-activity is to determine whether the developer has correctly and completely implemented the requirements of the ST, functional specification, high-level design and low-level design in the implementation representation.

1.2 Input

2 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) the high-level design;

d) the low-level design;

e) the implementation representation;

f) the correspondence analysis between the TOE summary specification and the functional specification;

g) the correspondence analysis between the functional specification and the high-level design;

h) the correspondence analysis between the high-level design and the low-level design;

i) the correspondence analysis between the low-level design and the implementation representation.

1.3 Evaluator Actions

1.3.1 ADV_RCR.3.1E

ADV_RCR.3.1C	**For those corresponding portions of representations that are formally specified, the developer shall prove that correspondence.**

ADV_RCR.3-1 The evaluator ***shall check*** that a formal proof of correspondence between those TSF representations that are formally specified has been provided.

3 A formal proof of correspondence uses well-established mathematical concepts to define the syntax and semantics of the formal notation and the proof rules that support logical reasoning.

4 A method for gaining assurance that the TOE implements its specification is to formally prove that a lower-level representation is both correct and complete in its representation of the TSF as presented by the higher-level representation.

5 Where the formal TSF representations are not specified in the same formal language, the evaluator should confirm that a translation between the formal notations has been provided. The evaluator should check that this translation is correct.

ADV_RCR.3.2C	**For each adjacent pair of provided TSF representations, the analysis shall prove or demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.**

ADV_RCR.3-2 The evaluator ***shall examine*** the correspondence analysis between the TOE summary specification and the functional specification to determine that the functional specification is a correct and complete representation of the TOE security functions.

6 The evaluator's goal in this work unit is to determine that all security functions identified in the TOE summary specification are represented in the functional specification and that they are represented accurately.

7 The evaluator reviews the correspondence between the TOE security functions of the TOE summary specification and the functional specification. The evaluator looks for consistency and accuracy in the correspondence. Where the correspondence analysis indicates a relationship between a security function of the TOE summary specification and an interface description in the functional specification, the evaluator verifies that the security functionality of both are the same. If the security functions of the TOE summary specification are correctly and completely present in the corresponding interface, this work unit will be satisfied.

8 This work unit may be done in conjunction with work units of the ADV_FSP.x sub activity.

ADV_RCR.3-3 The evaluator ***shall examine*** the correspondence analysis between the functional specification and the high-level design to determine that the high-level design is a correct and complete representation of the functional specification.

9 The evaluator uses the correspondence analysis (including any proof of correspondence), the functional specification, and the high-level design to ensure that it is possible to map each security function identified in the functional specification onto a TSF subsystem described in the high-level design. For each security function, the correspondence indicates which TSF subsystems are involved in the support of the function. The evaluator verifies that the high-level design includes a description of a correct realisation of each security function.

10 Where both the functional specification and high-level design have been specified at least in a semiformal style, this work unit should be completed in conjunction with ADV_RCR.3-6, ADV_RCR.3-7 and ADV_RCR.3-8.

11 Where both the functional specification and high-level design have been specified at least in a formal style, this work unit should be completed in conjunction with ADV_RCR.3-12. In this case the evaluator examines the proof of correspondence in addition to any semiformal and informal demonstration.

ADV_RCR.3-4 The evaluator ***shall examine*** the correspondence analysis between the high-level design and the low-level design to determine that the low-level design is a correct and complete representation of the high-level design.

12 The evaluator uses the correspondence analysis (including any proof of correspondence), the high-level design, and the low-level design to ensure that it is possible to map each TSF module identified in the low-level design onto a TSF subsystem described in the high-level design. For each TOE security function, the correspondence indicates which TSF modules are involved in the support of the function. The evaluator verifies that the low-level design includes a description of a correct realisation of each security function.

13 Where both the high-level design and the low-level design have been specified at least in a semiformal style, this work unit should be completed in conjunction with ADV_RCR.3-6, ADV_RCR.3-7 and ADV_RCR.3-8.

14 Where both the high-level design and low-level design have been specified at least in a formal style, this work unit should be completed in conjunction with ADV_RCR.3-12. In this case the evaluator examines the proof of correspondence in addition to any semiformal and informal demonstration.

ADV_RCR.3-5 The evaluator ***shall examine*** the correspondence analysis between the low-level design and the implementation representation to determine that it is a correct and complete representation of the low-level design.

15 The evaluator uses the correspondence analysis, the low-level design, and the implementation representation to ensure that it is possible to map each TSF module identified in the low-level design onto a source-code, firmware and hardware within the implementation representation. TSF subsystem described in the high-level design. For each TSF Module, the correspondence indicates which portions of the implementation representation implement the module. The evaluator verifies that the correspondence analysis includes a description of a correct realisation of each TSF module.

ADV_RCR.3.3C For each adjacent pair of provided TSF representations, where portions of both representations are at least semiformally specified, the demonstration of correspondence between those portions of the representations shall be semiformal.

ADV_RCR.3-6 The evaluator ***shall check*** that the demonstration of correspondence between provided TSF representations, where both representations are at least semi-formal, is presented using a semiformal style.

16 A semiformal correspondence demonstration requires the use of a notation which is explicitly defined. It may be based on a restricted subset of the natural language. Alternatively, it may be based on accepted methodologies or diagrams, eg data flow diagrams, computer-aided design drawings, state transition diagrams or flow charts.

ADV_RCR.3-7 The evaluator ***shall examine*** the semi-formal correspondence demonstration to determine that the semiformal notation(s) used is capable of completely expressing the correspondence between semiformal representations, where each representation is at least semi-formal.

17 For this work unit, the evaluator reviews the semiformal notations used within each representation and looks for a structured method for tracing portions of the TSF in the more abstract representation to portions of the TSF in the less abstract representation. However, a single notation, may not be capable of completely expressing correspondence between TSF representations. In this case, a combination of notations could be used to provide a complete correspondence demonstration.

ADV_RCR.3-8 The evaluator ***shall examine*** the semiformal correspondence demonstration to determine that it contains all necessary informal explanatory text.

18 Supporting narrative descriptions are necessary for those portions of the correspondence analysis that are difficult to understand only from the semiformal or formal description.

ADV_RCR.3.4C For each adjacent pair of provided TSF representations, where portions of both representations are formally specified, the proof of correspondence between those portions of the representations shall be formal.

ADV_RCR.3-9 The evaluator ***shall check*** that the demonstration of correspondence between provided TSF representations, where both representations are formal, is presented using a formal style.

19 The purpose of the formal security specification is to define security (in the context of the TOE and its environment) in a clear, unambiguous way. Appropriate formal notations used for the correspondence demonstration are to be agreed by the Scheme.

ADV_RCR.3-10 The evaluator ***shall examine*** the formal portions of the correspondence demonstration to determine that it is supported by syntactic and semantic rules.

20 The notation should possess well defined syntax and semantics, with both being themselves expressed in formal notations which satisfy either of the first two criteria. Syntactic and Semantic rules define how to recognise constructs unambiguously and determine their meaning.

ADV_RCR.3-11 The evaluator ***shall examine*** the formal portions of the correspondence demonstration determine that it contains all necessary informal explanatory text.

21 Supporting narrative descriptions are necessary for those portions of the model that are difficult to understand only from formal description.

1.3.2 ADV_RCR.3.2E

ADV_RCR.3.2E The evaluator shall determine the accuracy of the proofs of correspondence by selectively verifying the formal analysis.

ADV_RCR.3-12 The evaluator ***shall examine*** the proof of the correspondence between each formally specified TSF representation to determine that the lower level representation is correct and complete for a selected subset of the higher level TSF representation.

22 The evaluator completes this work unit on a selected sample of the formal TSF representations. For guidance on sampling refer to Annex B.2.

23 For the selected sample, the evaluator confirms that the formal specification of the lower level representation is shown to be deducible from the axioms and specifications of the higher level representation, and that the formal specification of the lower level representation does not conflict with or contradict the formal specification of the higher level representation.

24 Evaluators should consider the following questions when examining the formal proof of correspondence:

a) Is the objective of the proof appropriate?

b) Is the formalisation of the objective of the proof correct?

c) Are the assumptions or axioms from which the proof begins correct?

d) Are the deductive steps correct?

25 Ideally all proofs should be verifiable by the use of automated proof tools if appropriate tools are available and adequate.