
UK IT SECURITY EVALUATION & CERTIFICATION SCHEME

UK CC Interpretation - UK/3.1/009

12 March 2007

Status: Endorsed for use in UK Scheme

Subject: Abstract Machine Testing and the TOE Boundary

REFERENCES CC, Version 3.1, Revision 1, September 2006
Controlled Access PP (CAPP) v1.d, 8 Oct 1999

Issue

1. Does the inclusion of the CC Part 2 component FPT_AMT.1 imply that the TOE excludes the underlying firmware/hardware?
2. Closely related to this issue is the question of when it is appropriate to use FPT_TST.1 in a PP or ST, and when it is appropriate to use FPT_AMT.1 (noting that the former has a dependency on the latter).
3. The wording of FPT_AMT.1 suggests that it is only referring to the case where the underlying abstract machine is part of the IT environment, i.e. it is not relevant if the underlying platform is declared to be within the TOE boundary. However, this is contradicted by various statements made elsewhere in CC Part 2 (Section 15 and Annex J in particular – see the Rationale below for specific details). Furthermore, if FPT_AMT.1 is **not** relevant where the TOE includes the underlying platform, this would mean that (in such cases):
 - a. There is no appropriate CC Part 2 component for testing the correct operation of the underlying hardware and firmware within the TSF (given that the concern of FPT_TST.1 is with the TSF software and data).
 - b. The dependency of FPT_TST.1 on FPT_AMT.1 would be unresolved.
4. Note: in this interpretation, the term “underlying abstract machine” and “underlying platform” typically refers to underlying hardware/firmware. However (as recognised by CC), in some cases, the underlying platform or abstract machine may also include software.
5. In a related issue, CAPP states in an application note FPT_AMT.1 “In general this component refers to the proper operation of the hardware platform on which a TOE is running...”. That is to say, CAPP includes this requirement to test the underlying hardware of the TOE. This requirement is mapped to O.ENFORCEMENT stating, “TOE must provide the capability to demonstrate correct operation of the TSF’s underlying abstract machine”. When claiming compliance with this PP, is it possible to exclude the component FPT_AMT.1 as it is related to the IT environment rather than the TOE?

Interpretation

6. FPT_AMT.1 can be included in a PP or ST regardless of whether the TOE includes the underlying firmware and hardware.
7. If a PP or ST author wishes to specify requirements to test the correct operation of the TSF software, FPT_TST.1 should be used. If, however, there is a desire to test the correct operation of the underlying platform, then FPT_AMT.1 should be used – whether or not the underlying platform is within the TOE boundary.
8. Where appropriate, a PP or ST author may choose to apply the refinement operation on FPT_AMT.1 (consistent with CC Part 1 Annex C.4.4) to clarify the concern of FPT_AMT.1 is with underlying firmware and hardware that is part of the TSF.
9. The UK Scheme recommendation to the CCMB is that the various statements relating to FPT_AMT.1 in CC Part 2 are amended to make this point clear. These include the following changes to CC Part 2:
 - Chapter 15.1 rename this section “Underlying abstract machine test” as “Abstract machine test” and update the component levelling diagram accordingly.
 - Chapter 15.1, delete “underlying” from paragraphs 363, 364 and 366.
 - Chapter 15.1, revise the wording of the element FPT_AMT.1.1 as follows to remove the hardcoded reference to security assumption, making this a selection operation depending on whether the abstract machine is within the TOE or in the IT environment:

FPT_AMT.1.1 The TSF shall run a suite of tests [selection: *during initial start-up, periodically during normal operation, at the request of an authorised user, [assignment: other conditions]*] to demonstrate the correct operation of [selection: *chose one of: that underlies the TSF, that is included in the TSF*] provided by the abstract machine that acts as a virtual machine for the TSF.
 - Annex J, delete “underlying” from paragraph 1165.
 - Annex J.1 rename this section “Underlying abstract machine test (FPT_AMT)” as “Abstract machine test (FPT_AMT)”.
 - Annex J.1, paragraph 1167 replace “This family defines the requirements for the TSF's testing of security assumptions made about the underlying abstract machine upon which the TSF relies” with “This family defines the requirements for the TSF's testing of the abstract machine upon which the TSF relies. This abstract machine may form part of the TOE or may be provided by the IT environment.
 - Annex J.1, paragraph 1168 replace with “The term “abstract machine” typically refers to hardware components upon which the TSF has been implemented. However, the phrase can also be used to refer to a hardware and software combination behaving as a virtual machine upon which the TSF relies.”
 - Annex J.1, delete “underlying” from 1st bullet point of paragraph 1169 and from paragraph 1170 (2 instances).
 - Revise Annex J.1, paragraph 1170 to read, “The tests of the abstract machine should be sufficient to test all security relevant characteristics of the abstract machine upon which the TSF relies”.

- Annex J.1, paragraph 1171, replace with “This component provides support for the periodic testing of the abstract machine upon which the TSF's operation depends, by requiring the ability to periodically invoke testing functions. Depending on whether the abstract machine is implemented in the TOE or forms part of the IT environment, this testing will demonstrate that the security objectives for the TOE or security assumptions (respectively) relating to the abstract machine are being satisfied.”
- Annex J.1, add Selection: operation after para 1170 with the guidance “In FPT_AMT.1.1, the PP/ST author should specify whether the security assumptions for or the security objectives for the TOE relating to the abstract machine are to be satisfied by the tests. The selection made by the PP/ST author will depend on whether the abstract machine forms part of the TOE in which case the selection should be “the security objectives for the TOE”, or the abstract machine forms part of the IT environment in which case “the security assumptions” should be selected”.
- Chapter 15.4, remove the dependency from FPT_TST.1 on FPT_AMT.1. Replace dependencies with “None”, and replace Annex J.14, paragraph 1281 final sentence with “The abstract machine upon which the TSF software is implemented may be tested via Abstract machine test (FPT_AMT)”.
- Replace Chapter 15.1 paragraph 362 “there are three significant portions for the TSF” and Annex J paragraph 1164 “there are three significant portions that make up the TSF” with “there are three significant aspects relating to the TSF”.

Rationale

10. Readers who confine themselves to the description of FPT_AMT.1 in CC Part 2 (section 15.1 and Annex J.1) would, quite reasonably, conclude that FPT_AMT.1 only applies where the hardware and firmware platform is part of the IT environment. Statements that refer to an “underlying” abstract machine, which “could be some known and assessed combination of hardware and software”, and which provide “security assumptions ... upon which the TSF's operation depends” are all strongly suggestive that FPT_AMT.1 refers to hardware and firmware (and, in some cases, software) that is part of the IT environment, i.e. an underlying platform that is not part of the TSF. Use of the phrase “security assumptions” in the text of the FPT_AMT.1 requirement appears to be a particularly strong pointer, given that (in the context of evaluation), “assumption” is normally used to refer to some property or expectation that will not be directly validated during the evaluation, but is taken to be axiomatic for the evaluation (and thus out of its scope).
11. However, this impression is contradicted by statements elsewhere in Section 15 of CC Part 2:
 - a. In the introduction to Section 10, the TSF’s “abstract machine” is declared to be one of three “significant portions for the TSF”. This is confirmed in CC Part 2 Annex J which again includes the “abstract machine” as one of the three “significant portions” that “make up the TSF”. By definition, these statements firmly place the “abstract machine” within the TOE boundary.
 - b. In CC Part 2 Annexes for FPT_TST.1 (Section J.16), the User Notes state that, in terms of ensuring the correct operation of the TSF, the concern of FPT_TST.1 is with the TSF *software*. The User Notes then go on to explain the dependency of

FPT_TST.1 on FPT_AMT.1 in terms of the latter being concerned with the correct operation of the “abstract machine upon which the TSF software is implemented”. This makes eminent sense: if there is a requirement to confirm the correct operation of the TSF software, there is also a need to confirm the correct operation of the abstract machine upon which the software runs. This need is independent of where the abstract machine lies in relation to the TOE boundary.

12. Taking all these (not entirely consistent) statements in context, it would therefore be reasonable to conclude from this that FPT_AMT.1, whilst concerned with ensuring the correct operation of the underlying abstract machine, can be used in either case - where the TOE includes the hardware/firmware, and where it excludes it. This can be assumed to be the thrust of the following statements taken from the User Notes for FPT_AMT.1 in the CC Part 2 Annexes (Section J.1):

The term “underlying abstract machine” typically refers to the hardware components upon which the TSF has been implemented. However, the phrase can also be used to refer to an underlying, previously evaluated hardware and software combination behaving as a virtual machine upon which the TSF relies.

13. Whilst the first sentence can be seen as broadly consistent with the earlier statements, the second sentence (significantly) permits FPT_AMT.1 to be used to test the assumptions provided by the underlying abstract machine if this is provided by the IT environment.
14. In arriving at this interpretation, we have had to carry out a detailed analysis of the various statements in CC Part 2 that relate to the notion of an “abstract machine”. It should not be necessary for the reader to carry out such an analysis of CC Part 2 in order to discern its intended meaning. It is highly desirable, therefore, that all the apparent inconsistencies in CC Part 2 that are highlighted in this interpretation are resolved.
15. A ST that does not include the component FPT_AMT.1 can only claim conformance to a PP that includes FPT_AMT.1 if the following are satisfied:
 - The PP states that demonstrable conformance is permitted.
 - The ST includes the component FPT_TST.1 and provides a suitable rationale demonstrating that the abstract machine is implemented within the boundary of the TOE, and that the testing performed to meet the requirement FPT_TST.1 satisfies the security objectives for the TOE relating to the abstract machine.