

---

## UK IT SECURITY EVALUATION & CERTIFICATION SCHEME

UK CC Interpretation - UK/2.2/009

25 February 2005

Status: Endorsed for use in UK Scheme

Effective from: 25 February 2005

Subject: Abstract Machine Testing and the TOE Boundary

---

**REFERENCES** CC, version 2.2, January 2004

### 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 10 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.

### Interpretation

5. FPT\_AMT.1 can be included in a PP or ST regardless of whether the TOE includes the underlying firmware and hardware.
6. 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.

7. Where appropriate, a PP or ST author may choose to apply the refinement operation on FPT\_AMT.1 (consistent with CC Part 1 paragraph 176) to clarify the concern of FPT\_AMT.1 is with underlying firmware and hardware that is part of the TSF.
8. The UK Scheme recommendation to CCIMB is that the various statements relating to FPT\_AMT.1 in CC Part 2 are amended to make this point clear.

## Rationale

9. Readers who confine themselves to the description of FPT\_AMT.1 in CC Part 2 (section 10.1) would, quite reasonably, conclude that FPT\_AMT.1 only applies where the hardware and firmware platform is part of the IT environment. Statements which 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).
10. However, this impression is contradicted by statements elsewhere in Section 10 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.
11. 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.*

12. 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.
13. 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.