# Mobile Device Fundamentals PP Equivalency Considerations

*Version 1.0*
*3 June 2015*

## 1. Introduction

The purpose of equivalence in PP-based evaluations is to find a balance between evaluation rigor and commercial practicability—to ensure that assurance is achieved across differences in product model and platform, while recognizing that there might be little to be gained from requiring that every variation in a product or platform be fully tested.

In general, if one product model is found to be compliant with a PP on one platform, then all equivalent product models on equivalent platforms are also considered to be compliant with the PP. But product models and platforms may also be partially equivalent. In cases of partial equivalence, only the differences that affect PP-specified security functionality need be tested. Of course, in the case of non-equivalent product models and platforms, all PP-specified tests must be run.

This document provides guidance for determining equivalency of mobile devices for purposes of evaluation against the Mobile Device Fundamentals Protection Profile (MDF PP).

The MDF PP defines the TOE thus:

> A Mobile Device in the context of this assurance standard is a device which is composed of a hardware platform and its system software.

In this document, we use Device to refer to the Mobile Device as a whole. For purposes of this guidance, the Device consists of System Software and a Hardware Platform. “System Software” can be considered synonymous with “product model.”

Given these definitions, equivalency has two aspects in the context of the MDF PP:

1. ***System Software Equivalence*****:** Different feature sets or versions of System Software may be considered equivalent if there are no differences that affect PP-specified security functionality.
2. ***Hardware Platform Equivalence*****:** Platforms may be considered equivalent if there are no differences in the services they provide to the System Software—or in the way the platforms provide the services—that affect PP-specified security functionality.

Mobile Devices are considered equivalent in accordance with these guidelines if the System Software and Hardware Platforms of the Devices are equivalent. If Devices are found to be equivalent, then it is necessary to test only one of the equivalent Devices against the MDF PP. Alternatively, some requirements may be tested on one device, and some requirements tested on others of the equivalent devices, as long as all requirements are covered by testing.

If Devices are found to be partially equivalent, then only those differences that affect PP-specified security functionality need be tested.

All tests must be run for Devices that are non-equivalent.

The determination of System Software and Hardware Platform equivalence for purposes of evaluating against the MDF PP is made by the Evaluator/Vendor and Validator in accordance with these guidelines.

## 2. Approach to Equivalency Analysis

The intent of these guidelines is to ensure that each different System Software implementation of each PP-specified security function is tested on each Hardware Platform that implements that security function differently.

When performing equivalency analysis, the Evaluator/Vendor should first use the factors and guidelines for System Software equivalence to determine the variant feature sets for each System Software instance that must be evaluated. System Software instances that are identical are considered equivalent. More generally, System Software instances that do not differ in any PP-specified security functionality are considered equivalent. Software instances that differ in some PP-specified security functionality, but not all, are considered partially equivalent.

Having determined the set of System Software features that must be evaluated, the next step is to determine the set of Hardware Platforms on which each System Software instance must be tested.

The Evaluator/Vendor and Validator should use the factors and guidelines for Hardware Platform equivalence to determine equivalent hardware platforms. In general, platforms that implement the same architecture, instruction set, chipset, and device architectures are equivalent absent evidence to the contrary affecting PP-specified security functionality. Like System Software, Hardware Platforms can be equivalent, partially equivalent, or non-equivalent.

Equivalent System Software instances must be tested on each non-equivalent Hardware Platform for which compliance is claimed.

In the end, each non-equivalent System Software implementation of each PP-specified security function must be tested on each non-equivalent Hardware Platform implementation of that function.

## 3. Specific Guidance for Determining System Software Equivalence

System Software equivalence attempts to determine whether different feature levels or revision levels of System Software across a product line are equivalent for purposes of MDF PP testing. For example, if a Device has a “basic” edition and an “enterprise” edition, is it necessary to test both models? Or does testing one model provide sufficient assurance that both models are compliant? Or does each model require that some functions be tested?

If equivalence is to be claimed between different feature levels or revision levels, the Vendor must present an argument detailing the differences between the feature/revision levels and justifies that the differences in features/revisions do not affect any of the MDF PP requirements as exercised by the “test” assurance activities. The evaluator will examine this argument to determine that it is accurate. The information in Table 1 provides the factors and guidance to be considered in determining System Software equivalence that go into the argument.

|  | Factor | Guidance |
|---|---|---|
| 1. | **Lapse of Time** | If a System Software instance is submitted for evaluation more than two years after the last evaluation activity on that instance or an equivalent instance, then any such instances are considered to be not equivalent and all tests must be run. |
| 2. | **Application Processor--Architecture** | System Software instances that run on different application processor architectures (e.g. x86 vs. ARM) are not equivalent and all tests must be run for each instance. |
| 3. | **Application Processor—Instruction Set** | System Software instances that run on application processors with the same architecture but different instruction sets are not equivalent if the instruction set differences affect PP-specified security functionality. It is necessary to test only the PP-specified security functionality affected by the instruction set differences. If only differences are tested, then the differences must be enumerated, and for each difference the Vendor must provide an explanation of why each difference does or does not affect PP-specified functionality. If the differences (e.g. between different ARM processor versions) do not affect PP-specified functionality, then the instances can be considered equivalent. |
| 4. | **TOE Software Binaries** | If System Software binaries are identical, the instances can be considered equivalent. |
| 5. | **PP-Specified Security Functionality Not Affected by Differences** | If the differences between System Software instances do not affect PP-specified security functionality, then the instances can be considered equivalent. The Vendor must provide an explanation of why the differences do not affect PP-specified security functionality. |
| 6. | **PP-Specified Security Functionality Affected by Differences** | If PP-specified security functionality is affected by the differences between instances of System Software, then the instances are not equivalent and must be tested separately. It is necessary only to test the functionality affected by the software differences. If only differences are tested, then the differences must be enumerated, and for each difference the Vendor must provide an explanation of why each difference does or does not affect PP-specified functionality. If different System Software instances are fully tested separately, then there is no need to document the differences. |

Table 1. Factors for Determining System Software Equivalence

## 4. Specific Guidance for Determining Hardware Platform Equivalence

Platform equivalence is used to determine the Hardware Platforms that must be tested for each System Software instance.

Identical application processor architectures, instruction sets, chipsets, and device architectures are strong factors in favor of Hardware Platform equivalence for purposes of MDF PP testing. In such cases, Hardware Platforms should be considered equivalent absent evidence that MDF PP-specified security functionality is implemented in a way that requires the System Software to interface differently between the platforms.

If a Vendor claims equivalence between Hardware Platforms, the Vendor must document a comparison of the features of the two platforms—to include processor architectures, instruction sets, memory controllers, baseband/application communication architectures, network hardware, and I/O devices—and make the argument that the differences do not affect any of the MDF PP requirements as exercised by the "test" assurance activities. The Evaluator examines this argument to ensure that it is accurate.

| | Factor | Guidance |
|---|---|---|
| **1.** | **Platform Hardware -- Application Processor** | Hardware platforms that implement different application processor architectures are not equivalent.<br>Hardware Platforms that implement the same application processor architecture, but not the same application processor model may be considered equivalent if no PP-specified security functionality is affected by the differences, and subject to the additional constraints in (2) below.<br>If some, but not all, PP-specified security functionality is affected by processor model differences, then the Platforms can be considered partially equivalent. The differences must be documented and tested for all models, still subject to the additional constraints in (2), below.<br>Hardware platforms that implement the same model application processor may be considered equivalent, subject to the additional constraints in (2) below. |
| **2.** | **Platform Hardware – Other hardware** | Given that (1) allows for possible equivalence, Hardware Platforms that implement the same RF part, Bluetooth part, GPS component, camera, microphone, and memory types can be considered equivalent. If a Hardware Platform model does not implement all of these features, then it can be considered equivalent to a model that implements additional features up to the full set.<br>Given that (1) allows for possible equivalence, hardware platforms that implement different RF parts, Bluetooth parts, GPS component, camera, microphone, and memory types would have to have the requirements relating to those parts tested. Likewise, if baseband processor and USB controllers are different, then requirements relating to those devices must be tested if they affect PP-specified security functionality. |

**Table 2. Factors for Determining Hardware Platform Equivalence**