Internal distribution code:

(A) [-] Publication in OJ
(B) [-] To Chairmen and Members
(C) [-] To Chairmen
(D) [X] No distribution

Datasheet for the decision of 7 May 2019

Case Number: T 1563/17 - 3.5.06

Application Number: 11154503.4

Publication Number: 2487618

IPC: G06F21/57

Language of the proceedings: EN

Title of invention:
Managing booting of secure devices with untrusted software

Applicant:
BlackBerry Limited

Headword:
Booting untrusted software

Relevant legal provisions:
EPC Art. 56

Keyword:
Inventive step - after amendment (yes)

Decisions cited:
T 0641/00, T 1325/17

Catchword:

Europäisches Patentamt
European Patent Office
Office européen des brevets

Beschwerdekammern

Boards of Appeal

Chambres de recours

Boards of Appeal of the
European Patent Office
Richard-Reitzner-Allee 8
85540 Haar
GERMANY
Tel. +49 (0)89 2399-0
Fax +49 (0)89 2399-4465

Case Number: T 1563/17 - 3.5.06

D E C I S I O N
of Technical Board of Appeal 3.5.06
of 7 May 2019

Appellant: (Applicant)	BlackBerry Limited 2200 University Avenue East Waterloo, ON N2K 0A7 (CA)
Representative:	Hanna Moore + Curley Garryard House 25/26 Earlsfort Terrace Dublin 2, D02 PX51 (IE)
Decision under appeal:	**Decision of the Examining Division of the European Patent Office posted on 17 February 2017 refusing European patent application No. 11154503.4 pursuant to Article 97(2) EPC.**

Composition of the Board:

Chairman	G. Zucka
Members:	M. Müller
	A. Jimenez

Summary of Facts and Submissions

I. The appeal is against the decision of the examining division, with reasons dispatched on 17 February 2017, to refuse European patent application No. 11 154 503 for lack of inventive step over:

D1: US 2005/033969 A1
D2: WO 2010/121020 A1

During the examination procedure, two further documents were also referred to:

D3: US 2009/295461 A1
D4: US 2010/017659 A1

II. Notice of appeal was filed on 13 April 2017, the appeal being paid on the same day. A statement of grounds of appeal was received on 16 June 2017. The appellant requested that the decision be set aside and that a patent be granted on the basis of claims 1-5 according to a main or one of two auxiliary requests, all as filed with the grounds of appeal, the other application documents being description pages 1 and 3-12, and drawing sheets 1-4 as originally filed, and description page 2 as filed on 28 January 2016.

III. In an annex to a summons to oral proceedings, the board informed the appellant of its preliminary opinion that the claimed invention lacked inventive step over D1, Article 56 EPC.

IV. In response to the summons, by letter dated 4 April 2019, the appellant filed amended claims 1-5 according to a main and three auxiliary requests.

V. Oral proceedings were held on 7 May 2019, during which the appellant filed amended claims 1-5 according to an auxiliary request 4 and withdrew all previous requests.

VI. Claim 1 of the sole request reads as follows:

"A method of executing either an unsigned operating system or a signed operating system on a device (100) having a processor (128) and a collection of hardware resources designated as a security block, the security block comprising user data and device specific cryptography keys, the device being operable in a factory mode (220) where access to the security block is not allowed and in a product mode (210) where access to the security block is allowed, said method comprising the processor (128) loading a boot loader and under the instructions from the boot loader:

loading (302) an operating system and subsequent to loading the operating system:

determining (304) whether said device is in the factory mode (220) or in the product mode (210),

on determining (304) that the device (100) is in factory mode:

determining (310) whether the operating system is signed by a trusted entity:

if the operating system is signed by a trusted entity allowing (308) execution of the operating system on authentication (306) of [the] operating system and denying execution of [the] operating system on failure to authenticate the operating system;

if the operating system has not been signed by a trusted entity and represents an untrusted operating system, determining (312) whether a counter of allowed insecure boots exceeds zero and:

upon determining (312) that the counter of allowed insecure boots exceeds zero, decrementing (314) the counter of allowed insecure boots and disabling (316) access to the security block to maintain security for user data and cryptographic keys saved in or protected by the security block, the disabling further comprising erasing all user data not stored [in] or protected by the security block and, once access to the security block has been disabled, allowing execution (318) of the untrusted operating system; or

upon determining (step 312) that the counter of allowed insecure boots has reached zero, deactivating (320) factory mode and restarting (322) the device to exit factory mode (220);

on determining (304) that said device is in product mode determining (306) whether the operating system is authenticated and allowing execution (308) of the operating system on authentication of [the] operating system and denying execution of [the] operating system on failure to authenticate the operating system."

VII. At the end of the oral proceedings, the chairman announced the decision of the board.

Reasons for the Decision

The invention

1. The application considers the situation in which a "third party" or "contract manufacturer" manufactures a computing device on behalf of an "Original Equipment Manufacturer" (OEM) (see paragraphs 2 and 26). In the

eventually delivered product, only trusted operating systems should be executed; trust may be established with cryptography (see paragraph 3). During manufacture, however, it may be convenient to allow the execution of operating systems which are unsigned or only signed by an entity not trusted by the OEM (see paragraph 28).

1.1 To deal with this situation, the invention proposes to provide the device with two modes: "factory mode" and "product mode" (see figure 2).

1.2 In product mode, only "secure boots" are allowed (figure 3, nos. 304, 306 and 308). In factory mode, however, a limited number of "insecure boots" is allowed (see figure 3, nos. 310 and 312) - i.e. the booting of an operating system not signed by a trusted entity (see figure 3, no. 306, and paragraphs 28 and 38) - during which the access to a "security block" is disabled (see figure 3, no. 316; paragraphs 6 and 42-44; and original claim 2). At the latest when this number has been reached, factory mode is left (see figure 3, no. 320, and paragraph 29).

1.3 The security block is disclosed as containing, *inter alia*, device-specific cryptographic keys (see paragraph 42) or "user-specific data" (or "user data", see paragraphs 47 and 48). Where user data "is not stored in or protected by the security block", it may be deleted when access to the security block is disabled (paragraph 48).

The prior art

2. D1 addresses the problem of enabling a "third party to perform testing, debugging and servicing of [an]

electronic device and its software without risking that the third party is given access to [sensitive] information" (see paragraph 6). As a solution, D1 discloses a system which has two operating modes, a secure one for standard usage and an "insecure" one for testing or debugging. The secure mode is said to be the "normal" mode (see paragraph 12 and figure 2, box 2 from the top). In the secure mode, access to certain "security related data" is allowed, in the insecure mode it is inhibited (*loc. cit.* and abstract). During the boot procedure, the system of D1 checks the signatures "for the first protected applications and operating system" (see paragraph 31, and figure 2, box 3 from the top). When the signatures are correct, the application and the operating system are downloaded in the secure environment RAM (paragraph 31; figure 2, box 4 from the top, left branch; and figure 1) and both execute in secure mode. When the signature check fails or no signature is present, unsecure mode is activated and the non-verified application is loaded into a RAM outside the secure environment (paragraph 32, and figure 2, right branch). What happens with a non-verified operating system in this case is not specifically disclosed. At the latest after a fixed time period, the system returns to secure mode (see paragraphs 31 and 32, and figure 2).

3. D2 discloses a device with a TCG ("Trusted Computing Group") compliant trusted component (abstract; see also figures 6 and 7, no. 120) and a "secure boot process". Depending on whether, during that process, integrity of the trusted component can be verified, the device will "operate in accordance with" one of two "security policies". The "first security policy", used in case of successful verification, allows access to certain

confidential data; the "second" one, used in case of failure, does not (see paragraphs 35, 71 and 76-79, figure 4 and 9).

4. D3 discloses a device operating in one of two modes called "secure mode" and "production mode", the latter of which allows debugging and testing by "bypassing trust evaluation" (see figure 5, nos. 502A and 502B, and paragraphs 43-45).

5. D4 discloses another circuit with two modes of operation, a secure mode and a non-secure mode. The circuit stores "circuit identification information" indicating whether it is a "test circuit" or a "production circuit" (see paragraphs 12 and 28). For production circuits the secure mode is entered in which security checks are enforced. For test circuits, security checks may be bypassed and non-secure access to certain information or registers is allowed (see paragraphs 13 to 15, 25 and 29, figure 2).

Inventive step

6. The decision used D1 to assess inventive step.

6.1 D1 discloses that the operating system is checked in place but not, as claimed, that:

(i) the operating system is loaded before being tested (see also the decision, point 2.2 a) of the reasons).

6.2 The operating system in D1 is held in local ROM and thus need not be loaded to be validated (see paragraphs 22 and 31). However, it would have been an obvious alternative to load the operating system from

an external source, for instance, so as to allow for new and corrected versions of the operating system and so that the latest version of the operating system would always be used (see the decision, point 2.2.1 of the reasons). In this situation, the board agrees with the examining division that it would have been obvious that the loading would have to precede the validation. Therefore, feature (i) cannot represent an inventive step over D1.

6.3 In D1, the decision whether execution is to take place in the secure or the insecure mode is made depending on the result of the signature check (see figure 2). In contrast, D1 does not disclose that:

(ii) the two claimed modes are determined before - and thus independent of - the authentication of the operating system.

Accordingly, the insecure and secure modes of D1 must be compared to the two alternatives within the "factory mode" rather than with the factory and product modes, respectively. In passing, the board notes that the names of the two modes, "factory mode" and "product mode", do not, *per se*, imply any specific limitation on the claimed subject-matter and thus do not constitute a separate difference.

6.4 D1 discloses a method of determining whether a given operating system is signed and of executing at least the signed operating system (see paragraph 31). It further discloses that "if the signature check fails or if no signature is present", the non-verified application is executed "outside the secure environment" (paragraph 32). The appellant argued that this

paragraph only referred to the execution of applications with incorrect or absent signatures but did not allow the conclusion that "unsigned operating systems" were executed. Rather, it allowed for the option that an unsigned operating system was not executed at all.

6.4.1 The board agrees that D1 does not expressly disclose that or how:

(iii) an unsigned operating system may be executed,

let alone that:

(iii') the unsigned operating system may be executed a limited number of times.

6.4.2 D1 discloses that, in secure mode, access to a particular storage area, holding security data such as cryptographic keys (see paragraph 10), may be allowed in secure mode but not in insecure mode. That is, D1 discloses a "security block" storing cryptographic keys, but does not disclose that:

(iv) the keys are device-specific or that user data is also stored in the security block.

Moreover, D1 does not disclose that:

(v) "user data not stored in or protected by the security block" is erased along with the disabling of the access to the security block.

7. Having two "modes" of operation (see feature (ii)) in which two security policies apply is the direct consequence of a non-technical requirement. Starting

from D1, the invention would have to be viewed as the provision of a "product mode" as opposed to a (suitably modified) "factory mode" as disclosed in D1.

7.1.1 At the same time, a fair application of the COMVIK-style problem/solution approach (see T641/00) requires an argument as to why the non-technical requirement would have arisen in a particular technical context (see also T 1325/17, reasons 10.2).

7.1.2 The board takes the view that the objective technical problem to add a stricter "product mode" to the system of D1 while modifying the given security policy (claimed as the "factory mode") cannot realistically be assumed to have arisen and therefore starts its assessment from the prior art disclosed in the application.

8. The application starts from the situation in which a device manufacturer has to execute software provided by an OEM during manufacture (see paragraph 2) and, for security reasons, to validate a cryptographic signature of the OEM before execution (see paragraph 4). In this scenario, the operating system is first loaded and then verified (see paragraph 4). This is feature (i).

8.1 Claim 1 differs from this scenario at least in features (ii) to (v).

8.2 A device manufacturer would naturally experience the inconvenience of having to obtain an OEM signature for any software to be executed and would want to have this requirement relieved at least during manufacture. The OEM would then address the objective technical problem of providing a more lenient security policy to the

manufacturer without unnecessarily compromising security.

8.3 It follows from general security considerations that any lenient security policy should only be provided for as short a period as possible. In the given instance, this means: only during manufacture. It thus would have been obvious to provide two modes as claimed and to allow the execution of an unsigned operating system during but not after manufacture. This renders features (ii) and (iii) obvious.

8.4 The idea of allowing the execution of an unsigned operating system ("insecure boots") only a limited number of times (feature (iii')) is a generally obvious alternative to allowing unsigned software to run during a limited period of time as proposed in D1. Moreover, no detail is given of the manufacturing process so that no specific technical advantages of the claimed counter can be determined or invoked to establish an inventive step. Any particular manufacturing process would probably have suggested the claimed counter in view of the particular manufacturing process which might, for instance, require the execution of an unsigned operating system up to, say, 3 times during substantially different periods of times.

8.5 With regard to feature (iv), D1 discloses the use of a "security environment" storing and controlling access to all kinds of security related information, i.e. a "security block" as claimed. D1 explicitly discloses that the security data might contain cryptographic keys (see paragraph 10). That the claims require the storage of "device-specific" keys is not inventive because being "device-specific" is not a property of the key but, at best, its use which is not claimed. Moreover,

it would have been obvious to place "user data" into the "security block" of D1 if it happened to be security-relevant.

8.6 D1 does not disclose an explicit step of erasing user data that happens not to be stored in the security block - or rather, the security environment of D1, for that matter (see feature (v)).

8.7 Erasing user data outside of the security block before allowing the execution of an unsigned operating system contributes to increasing the security of the claimed device in a way not suggested by D1 or any of documents D2 to D4.

8.8 Therefore, the subject-matter of claim 1 is non-obvious over the prior art described in the application in view of D1, Article 56 EPC.

Order

For these reasons it is decided that:

1. The decision is set aside.
2. The case is remitted to the department of first instance with the order to grant a patent on the basis of the claims of the 4th auxiliary request filed on 7 May 2019 during oral proceedings with a description to be adapted.

The Registrar: The Chairman:

N. Schneider G. Zucka

Decision electronically authenticated