BESCHWERDEKAMMERN DES EUROPÄISCHEN PATENTAMTS

BOARDS OF APPEAL OF THE EUROPEAN PATENT OFFICE

CHAMBRES DE RECOURS DE L'OFFICE EUROPEEN DES BREVETS

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

Datasheet for the decision of 8 April 2010

Case Number: T 1568/05 - 3.5.01

Application Number: 02005454.0

Publication Number: 1278113

IPC: G06F 17/30, G06F 1/00

Language of the proceedings: EN

Title of invention:
On-disk file format for a serverless distributed file system

Applicant:
MICROSOFT CORPORATION

Headword:
Convergent encryption / MICROSOFT

Relevant legal provisions:
EPC Art. 52(1)(2)(3), 54(3), 123(2)

Relevant legal provisions (EPC 1973):
EPC Art. 54(1)(2)(4), 56, 84, 87(1)

Keyword:
"Claimed method obvious from prior art cited by examining division - no (after amendment)"
"Remittal for further examination - yes"

Decisions cited:
T 1173/97, T 0641/00, T 0258/03, T 0424/03

Catchword:
-

Europäisches Patentamt | European Patent Office | Office européen des brevets

Beschwerdekammern | Boards of Appeal | Chambres de recours

Case Number: T 1568/05 - 3.5.01

D E C I S I O N
of the Technical Board of Appeal 3.5.01
of 8 April 2010

Appellant: MICROSOFT CORPORATION
One Microsoft Way
Redmond, Washington 98052-6399 (US)

Representative: Grünecker, Kinkeldey,
Stockmair & Schwanhäusser
Anwaltssozietät
Leopoldstrasse 4
80802 München (DE)

Decision under appeal: **Decision of the Examining Division of the European Patent Office posted 25 July 2005 refusing European patent application No. 02005454.0 pursuant to Article 97(1) EPC 1973.**

Composition of the Board:

Chairman: S. Wibergh
Members: K. Bumes
G. Weiss

Summary of Facts and Submissions

I. This appeal is against the decision of the examining division to refuse European patent application No. 02005454.0, published as

A2: EP-A2-1 278 113,

for lack of inventive step (Article 56 EPC 1973) over

D1: Bruce Schneier, "Applied Cryptography", second edition, John Wiley & Sons Inc., New York 1996, pages 351 to 354, and

D2: US-A-5 940 507.

The examining division introduced those prior art documents without considering a complete search to be necessary (decision under appeal, point 2.2, last paragraph).

II. The statement setting out the grounds of appeal was accompanied by two sets of claims 1 to 78 entitled "Main Request" and "Auxiliary Request", respectively. Further amendments to the main request were suggested on an auxiliary basis.

III. The Board summoned the appellant to oral proceedings, as requested on an auxiliary basis in the notice of appeal. In an annex to the summons, the Board expressed its preliminary opinion that claim 1 (both requests) was too broad to be supported by the description (Article 84 EPC 1973) and to involve an inventive contribution (Article 56 EPC 1973). In addition to D1 and D2, the Board considered the following documents,

D0: WO-A-01/86396,

D10: US-A-5 202 982, and

D11: US-A-5 742 807.

IV. By letter dated 19 February 2010 the appellant submitted a new main request comprising an amended set of claims 1 to 70. The appellant requested that the decision of the examining division be set aside and a patent be granted on the basis of that main request. On an auxiliary basis, the appellant requested that the decision be set aside and the case be remitted to the department of first instance "for further examination of the requirements of Article 52 EPC". On a further auxiliary basis, the appellant requested that the decision be set aside and the case be remitted to the department of first instance "for further examination".

V. In a second letter, dated 1 March 2010, the appellant clarified its requests as follows:

As a new main request, an amended set of claims 1 to 69 was submitted (the amendment consisting in the removal of previous claim 38, with previous claims 39 to 70 renumbered as 38 to 69).

A new description page 2a was filed to replace the corresponding page underlying the decision under appeal.

The request for oral proceedings was withdrawn in the case that the claims of the new main request allowed the Board to set the decision aside and to remit the case to the department of first instance.

VI. On 8 March 2010 the Board cancelled the summons to oral proceedings.

VII. *The claims*

(a) Claim 1 reads:

"1. A method performed by a component (204) of a computing device (200) in a distributed file system (100), the method comprising:

segmenting (806) a file into multiple blocks;

computing (808) a hash of each of the blocks to produce a corresponding block hash value for each block; and

encrypting (810) the blocks using for each block its corresponding block hash value as an encryption key to produce encrypted blocks;

creating an indexing structure (408) to index individual encrypted blocks, the indexing structure containing a leaf node for each corresponding encrypted block, the leaf node containing an access value formed by encrypting the block hash value for the corresponding encrypted block using an access key and a verification value formed by hashing the corresponding encrypted block, wherein the access key is encrypted using a key of a user who is granted access to the file."

(b) Claim 38 reads:

"38. One or more computer readable media comprising computer-executable instructions that, when executed, perform the method as recited in claim 1."

(c) Claim 39 reads:

"39. A method performed by a component (228) of a computing device (200) in a distributed file system (100), the method comprising:

accessing a file (400) composed of a data stream

(402) and a metadata stream (404), the data stream containing multiple encrypted blocks that are each encrypted using a hash of a plaintext version of the encrypted block as an encryption key to produce the corresponding encrypted block, the metadata stream (404) containing an indexing structure (408) to index to the individual encrypted blocks, the indexing structure (408) having a leaf node for each corresponding encrypted block that contains a verification value used to verify the corresponding encrypted block, wherein the verification value is formed by hashing the corresponding encrypted block, and the leaf node for each corresponding encrypted block contains an access value formed by encrypting the hash for the corresponding encrypted block using an access key, wherein the access key is encrypted using a key of a user who is granted access to the file;

traversing (910) the indexing structure (408) to a leaf node associated with a target encrypted block; and

verifying (900) an authenticity of the target encrypted block independently of other encrypted blocks by using the verification value in the leaf node associated with the target encrypted block."

(d) Claim 48 reads:

"48. A component (204) in a distributed file system (100) in which files are stored across multiple distributed computers, the component (204) comprising:

a hash module (226) adapted to hash each of the blocks to produce for each block a corresponding block hash value; and

a cryptographic engine (224) adapted to encrypt the blocks using for each block its corresponding block hash value as an encryption key to produce encrypted

blocks;

a segmenting module (222) adapted to divide a file into multiple blocks;

an index builder (230) adapted to create an indexing structure (408) for indexing individual encrypted blocks, the indexing structure (408) containing a leaf node for each corresponding encrypted block, the leaf node containing an access value formed by encrypting the block hash value for the corresponding encrypted block using an access key and a verification value formed by hashing the corresponding encrypted block, wherein the access key is encrypted using a key of a user who is granted access to the file; and

wherein the cryptographic engine is further configured to encrypt the access key using the key of the user who is granted access to the file."

(e) Claim 60 reads:

"60. A distributed file system (100; 200) comprising a client component (204) according to claim 48 resident at a first computer to facilitate creation of a file (400)."

Reasons for the decision

1. *The invention*

The application is entitled "On-disk file format for a serverless distributed file system". However, the disclosed file format relates to any distributed file system (see paragraph 0006 of A2). Moreover, the term "file" is intended to include "data objects or essentially any other storage subject matter" (paragraph 0028).

1.1 *Problems to be solved as set out by the application*

It is well-known in the field of distributed file systems that files should be stored in a secure way preventing access by non-authorised users (A2, paragraph 0005); encryption is a usual way to achieve that goal. At the same time, file duplication (i.e. redundant files) should be minimised to reduce the amount of wasted storage space (A2, paragraphs 0003/0004).

1.1.1 Hence, there is a need for a method which allows files to be encrypted in such a way that files having identical contents can be recognised as such without revealing their encryption keys (A2, paragraph 0041).

This problem is also addressed by the appellant's earlier application D0. A corresponding US application,

D0': US serial No. 09/565 821 (5 May 2000),

is "incorporated by reference" in the present application (A2, paragraph 0042).

1.1.2 A second problem addressed by the present application (and by the earlier application D0) is that it is inefficient to read or update a part of a large file if the whole file has to be decrypted or encrypted (A2, paragraph 0069).

1.1.3 A third object of the present application (but not of the earlier application D0) is that the user should be enabled to quickly access a file and verify that it is indeed the requested file (A2, paragraph 0005).

1.2 *Solution*

1.2.1 The application teaches a rule for choosing a key with which to encrypt a user's file. The rule is simple and the same for every user but does not allow any user to know other users' keys. The rule is: generate a hash value of the file and use the hash value as a key to encrypt the file.

Identical plain text files produce identical hash values (i.e. identical keys) and, thus, identical cipher texts (see D0, page 15, lines 6 to 9). In other words, when two encrypted files are identical, their plain text contents are most likely to be identical (see D0, page 16, lines 6 to 9; page 18, lines 16 to 23). Thus, the file system can recognise redundancies from the encrypted versions of the files (without knowing the plain text contents) and can replace redundant files by a short link to a single storage location where a complete version of the encrypted file is stored in the distributed file system.

1.2.2 Further, the application teaches dividing, or segmenting, a large file into multiple blocks (A2, paragraph 0007). Individual blocks can be encrypted and decrypted more quickly than the whole file (A2, paragraphs 0011, 0070, 0072; see also D0, pages 28/29, chapter "File Segmentation", and pages 50/51, claims 79 to 83).

In addition, the distributed file system can detect duplicated file portions even if other portions of the file do not have matching counterparts in the system.

1.2.3 Moreover, the application teaches an indexing structure which enables the blocks of the segmented file to be managed individually. An access value is associated with each encrypted block to decrypt the block, and a verification value is associated with each encrypted block to verify the encrypted block independently of other blocks (A2, paragraphs 0008, 0010). The access value is an encrypted form of the hash value of the cleartext file block (see also paragraph 0104), whereas the verification value is a hash value of the encrypted block (paragraphs 0008 and 0091).

Only an authorised user is able to decrypt the access value to recover the hash value with which the block was encrypted. Thus, he can decrypt and read the block (A2, paragraphs 0036/0037, 0104, 0110/0111).

As the verification value is a hash value of the *encrypted* block, the integrity of each encrypted block can be verified directly, without decryption and without any knowledge of the keys used to encrypt the file (A2, paragraph 0088).

Above all, when the distributed file system is checked for duplicate files (or file portions), the system does not have to compare complete ciphertexts but it is sufficient to compare their hash values (i.e. verification values). If the hash values differ, it is clear that the ciphertexts differ (which, in turn, implies that the plaintexts differ). If the hash values match, the ciphertexts normally also match (this can be confirmed by comparing the full ciphertexts).

2. *Article 123(2) EPC - Amendment within content of the application as filed*

2.1 As compared to original claim 1 (A2, page 19), the amended claim 1 adds the step of creating an indexing structure to index individual encrypted blocks.

That step is specified by original claim 8 (dependent on original claim 1) and detailed by original claim 9 which introduces an "access value" and a "verification value", those values being defined e.g. in original claim 22: the access value of a block is formed by encrypting the block hash value using an access key, and the verification value is formed by hashing the encrypted block (see also paragraph 0008 of A2).

As present claim 1 reflects this teaching, the Board is satisfied that the subject-matter of the claim does not extend beyond the content of the application as filed.

2.2 The other amended claims cited above (point VII(b) to (e)) rely on the concept of claim 1 and do not add any matter beyond the content of the application as filed.

3. *Article 84 EPC 1973 - Clarity and conciseness of the claims; support by the description*

3.1 The Board is satisfied that amended claim 1 defines a clear encryption and indexing method based on a block-by-block handling of segmented files.

At the same time, the definition is precise enough to exclude undisclosed or speculative embodiments. Essential features of the method (e.g. encryption of the access key; use of a computing device) are included in the claim. Hence, the claim is also adequately supported by the description.

3.2 The other amended claims cited above (point VII(b) to (e)) rely on the concept of claim 1. They are also clear and supported by the description.

3.3 Despite the presence of five formally independent claims (claims 1, 38, 39, 48, 60), the claim set as a whole is concise since those claims cover complementary aspects of cryptographic file handling in a distributed file system: Claim 1 specifies a method for encrypting a file and forming a verification value. Claim 38 relates to a computer program for performing the method of claim 1. Claim 39 specifies a method for using the verification value formed by the method of claim 1. Claim 48 defines a component for performing the method of claim 1. Claim 60 relates to a distributed file system comprising such a component.

4. *Article 52(1)(2)(3) EPC - Eligibility for patent protection*

4.1 The Board acknowledges the technical character of the method according to claim 1 because the method uses technical means (a computing device) for a technical purpose in a distributed file system, see decision T 258/03-*Auction method/HITACHI* (OJ EPO 2004, 575). A technical effect consists not only in the encryption of files but also in an efficient verification and identification of encrypted files in the distributed file system.

4.2 The computer-executable instructions according to claim 38 have the potential for achieving the aforementioned technical effect which goes beyond the elementary general interaction between software and hardware. Already for that reason, the computer readable medium according to claim 38 constitutes an invention within the meaning of Article 52(1) EPC, see decision T 1173/97-*Computer program product/IBM* (OJ EPO 1999, 609). In addition, a computer readable data carrier is a technical object irrespective of the data stored on it, see decision T 424/03-*Clipboard formats I/MICROSOFT* (point 5.3 of the reasons).

4.3 Claim 39 also relates to a technical method because the method uses technical means (a computing device) for a technical purpose in a distributed file system. A technical effect consists not only in the way the files have been encrypted and provided with verification values but also in the use of the verification values to identify encrypted blocks in the distributed file system.

4.4 Claim 48 defines a technical component adapted to perform the method of claim 1.

4.5 Claim 60 defines a distributed file system comprising the technical component of claim 48.

5. *Article 87(1) EPC 1973 - Priority claim*

The present application claims priority rights from A0: US patent application serial number 09/814 259 filed on 21 March 2001.

The present application is identical to A0 except for the last drawing sheet (A2, Figures 12/13); that sheet of the application is missing in the copy of A0 available from the EPO's public file inspection database.

On the other hand, the claims on file do not relate to the subject-matter disclosed in Figure 12 or 13 (producing and signing a manifest). Hence, the Board has no doubt that the present claims are entitled to the filing date of A0, i.e. 21 March 2001.

6. *Article 54(3) EPC - Post-published prior art*

6.1 D0 claims an earlier priority date (5 May 2000) than the present application (21 March 2001). The priority claim of D0 is *prima facie* justified since D0 reproduces D0'.

6.2 D0 was published (15 November 2001) after the priority date of the claims now considered (21 March 2001).

6.3 Hence, D0 forms part of the prior art according to Article 54(3) EPC and Article 54(4) EPC 1973 (which are applicable to applications pending at the time of entry into force of the EPC 2000).

7. *Article 54(1) EPC 1973 - Novelty*

7.1 *Novelty over D0*

7.1.1 D0 anticipates the concept of convergent encryption (i.e. a hash value of a file is used to encrypt the file) and the concept of file segmentation where each file portion is separately encrypted using its own hash value (D0, pages 28/29; claims 79 to 83).

Further, without using the term, D0 provides an indexing structure for the file portions: "for each file portion, there is a corresponding ordered tuple that contains a cipher object and its associated information (i.e. list)" (D0, page 29, paragraph 1). The contents of such a list ("or other data structure") are described in the paragraph bridging pages 15/16 of D0. In particular, the list contains information that identifies the keys that were used to encrypt the representations (i.e. hash values) of files (or file portions, respectively). Those keys constitute access keys within the meaning of the present application, and the encrypted hash values constitute access values.

The list ("or other data structure") may be implemented as a tree (D0, page 18, lines 6 to 11). This is also the preferred embodiment of the indexing structure described by the present application (A2,

paragraphs 0088 to 0103).

7.1.2 On the other hand, D0 does not mention any distributed file system, and it does not disclose any step of hashing an encrypted block or file to form a verification value.

7.2 *Novelty over D1*

7.2.1 The *Karn* algorithm described in section 14.11 of D1 (pages 351/352) is said to operate on blocks of a plaintext but it splits each block of plaintext into two halves and encrypts each half using a hash value of the other half. Hence, the *Karn* algorithm does not encrypt a piece of plaintext by its own hash value.

7.2.2 Moreover, D1 does not disclose any indexing structure to index individual encrypted blocks. Nor is D1 concerned with access or verification values for managing blocks of a file in a distributed file system.

7.3 *Novelty over D2*

D2 relates to a file archive secured by encryption and deals with ways of managing encryption keys. D2 (column 4, paragraph 1) suggests authentication and verification techniques including hash values (in the form of checksums and message digests MD4/MD5), but the document does not address the problem of undesired file duplication and it does not feature any type of convergent encryption.

7.4 *Novelty over technology referred to in A2*

The application itself presents convergent encryption as a "known" technology (A2, paragraph 0038). However, that remark seems to relate to the inventors' knowledge as documented by reference D0' (A2, paragraph 0042).

7.5 *Novelty over D10*

D10 describes a concept of content hash naming. Each file in a database is given a name based on a hash of its contents. Thus, duplicate files can be recognised efficiently by their (short) names.

However, the citation does not deal with encryption, let alone convergent encryption. That is, it does not contemplate using the hash value of a content as a key for encrypting the content (content hash keying).

8. *Closest prior art*

8.1 The Board concurs with the appellant in considering D2 to be closer to the invention than D1 because D2 deals with a file archiving system using encryption. The problem of file duplication obviously arises in such a system (even though D2 does not address it) whether the system is implemented as a centralised or a distributed file system. The *Karn* algorithm of D1 represents a mere encryption algorithm; D1 does not disclose why and how that algorithm would be expanded into a file system.

8.2 The post-published application D0 cannot be considered in deciding whether there has been an inventive step (Article 56 EPC 1973, second sentence).

9. *Article 56 EPC 1973 - Inventive step*

9.1 Setting out from an encrypted file archive according to D2, the skilled person would envisage using any encryption algorithm known at that time. This includes the *Karn* algorithm which is known from D1. However, the *Karn* algorithm does not represent a convergent encryption proper (see point 7.2.1 *supra*).

9.2 Segmenting a file into blocks constitutes a notorious programming technique to facilitate the processing of large files at the price of additional file management overhead. Calling the overhead an indexing structure, tree, list or table does not provide any non-obvious technical contribution.

9.3 Where the security of files in the system relies on encryption, it is self-evident that the indexing structure has to include the management of access keys.

9.4 The use of hash values for file verification and/or file handling purposes is also well-known in distributed file systems, see e.g. D10 (title and abstract) or D11 (the paragraph bridging columns 2/3).

9.5 In summary, the method of claim 1 involves only one candidate for an inventive contribution, namely the use of convergent encryption, i.e. the encryption of each block of plaintext by its own hash value, which provides the advantageous effects mentioned at point 1.2.1 *supra*.

9.6 The pre-published prior art discussed above does not reveal the concept of convergent encryption. On the other hand, the decision under appeal points out that a complete search has not yet been carried out. Therefore, before the presence of an inventive step can be acknowledged and the grant of a patent can be envisaged, the search for relevant prior art will have to be completed with respect to the claim set filed on 1 March 2010. This should be the next step of the proceedings.

10. Since the case must be remitted to the examining division for a search to be carried out followed by continued examination, oral proceedings before the Board need not be held (cf. point V above).

Order

For these reasons it is decided that:

1. The decision under appeal is set aside.

2. The case is remitted to the department of first instance for further examination on the basis of claims 1 to 69 filed on 1 March 2010.

The Registrar: The Chairman:

T. Buschek S. Wibergh