

MARKING OF ELECTRONIC DOCUMENTS IN ORDER TO EXPOSE UNAUTHORIZED PUBLICATION

BACKGROUND OF INVENTION

The invention relates to a technique by which unauthorized publication and copying of electronic information can be exposed. The electronic information means particularly documents and programs distributed in an Internet type network.

The copying of electronically distributed information is technically very simple. For example, illegal copying of computer programs causes annual losses of millions of dollars to software industry. Pirate copies are also a major problem in video and recording industry.

The common feature of all such information is that any protection of the information is based on copyright (or closely related forms of protection, such as data base protection), an agreement, or non-disclosure obligation. As previously known, the essential difference between copyright and industrial protection - such as patent and design protection - is that in industrial protection infringement is sufficiently shown if products can be shown to be similar, whereas to show infringement of copyright it must be shown that a product has been copied rather than created independently. The same applies not only to copyright, but also to infringement of commercial secrets and other non-disclosure obligations. Such infringement is particularly difficult to prove in respect of databases where information is, in principle, available to anyone, and any additional value produced by the information provider is based on advantageous selection or presentation of information. In the present application, the term 'copyright' should thus be understood in the wide sense to comprise commercial secrets, non-disclosure obligations, etc. as well as the actual author's rights. Correspondingly, a 'copyright proprietor' here refers to anyone who wants to prevent and/or expose unauthorized copying and/or publication of information.

Even if the copyright proprietor is able to show that the product has been copied without permission, he must also be able to indicate who is responsible for the copying. If an unauthorized copy is found in the possession of an end user who has bought the product in good faith, no compensation can usually be required from the end user. Further, the end user cannot be required to be able to or willing to expose who has sold the product.

For example, software suppliers use technology in which the software asks the user for his name and contact information in connection with

the first installation. The information can be encoded in protected form and stored on an installation diskette. On the basis of this information the installation program produces a client identifier, which the client needs as he phones the supplier's telephone support. By monitoring incoming calls and
5 their client identifiers the supplier may be able to expose unauthorized copying. The technology, however, has many defects and restrictions. For example, the technology is based on the assumption that the user will at some point need telephone support. This, however, is not always a correct assumption. Most information can be used even without obtaining subsequent
10 support from the supplier. Since upon installing the software, the dishonest user can supply any contact information whatsoever, it is not always possible to conclude from the unauthorized copy where the copy has been made. By comparing installations conducted with different user information, the unauthorized user can conclude where on the disk the user-specific
15 information is located, and change the information. The technique cannot be applied to protecting text and image files, since the added information is easy to delete.

It is also known to slightly modify e.g. the character spacing, font size, etc. in text documents. The idea is that the user will not notice if there
20 are, for example, a few 11-point characters amidst 12-point text. This solution is previously known e.g. from U.S. Patent 5,467,447. The technology, however, cannot be used for protecting electronically distributed text, since the unauthorized user can simply impose the same layout on all text. Even if a so protected text is delivered on paper, the dishonest user can supply the text to
25 a text scanner, which removes all extra formatting.

BRIEF DESCRIPTION OF INVENTION

The object of the invention is thus to develop a technique by which unauthorized copying of electronically distributed information and the person responsible for the copying can be exposed. The objects of the invention are
30 achieved with a technique that is characterized by what is stated in the independent claims. The preferred embodiments of the invention are claimed in the dependent claims.

The invention is based on the idea that the electronically distributed information is marked, i.e. modified to be different for each user. The
35 modifications are not based on formatting, which can be easily changed, but are buried deep in the bit stream that carries the information. Differences are

introduced in so many places in the versions delivered to different users that the unauthorized user cannot detect all the differences, or that it is not economically sensible to attempt to detect all the differences.

A known technique in which the information to be produced is
5 marked in order to expose unauthorized publication comprises the steps of

- defining a user identifier,
- creating a source file containing a substantial part of the information produced, and
- creating a different target file for each user on the basis of the
10 source file and the user identifier.

The method of the invention further comprises the steps of

- storing, in a file, several predefined modification rules for modifying the information contained in the source file, and
by means of a digital processor:
- 15 - generating a sequence of modification keys on the basis of the user identifier,
- defining the locations in the source file that are to be modified and making modifications in these locations, determining the nature and/or position of the modifications on the basis of the predefined modification rules and
20 modification keys.

An advantage of the technique of the invention is that a user-specific identifier and thereby the person responsible for the unauthorized copy can be concluded from the information delivered to the user. The advantage obtained by keeping the modifications small is that the information
25 content remains intact and that the unauthorized user will not detect the modifications. When modifications are made in a large number of various locations, the copying and the copier can be exposed even if the information is copied only in part.

BRIEF DESCRIPTION OF FIGURES

30 In the following the invention will be described in greater detail by means of preferred embodiments illustrated in the attached drawings, in which

fig. 1 is a block diagram illustrating a technique of the invention,

fig. 2 illustrates generating a modification key on the basis of a user identifier,

35 figs. 3 to 5 illustrate modifying a source file on the basis of modification keys, and

fig. 6 illustrates delivery of documents through the Internet.

DETAILED DESCRIPTION OF INVENTION

With reference to fig. 1, we shall now describe different ways of implementing the technique of the invention in greater detail. Let us first define
5 some of the terms used in the present application.

A 'document' is a general term referring to material at least some version of which is to be delivered to the PC user. The document can comprise text or numerical information, audio or video information, a computer program, or a combination of any of these.

10 A 'source' in phrases like 'source material' and 'source document' refers to material that is not protected and that would be delivered to the user in prior art solutions.

A 'target' in e.g. 'target document' refers to material that is protected by modifying it specifically for each user.

15 A 'file' means that the material is in a computer-readable form.

In fig. 1 the source material provider (copyright proprietor) forms a source file 10 from the product. The function of a user identification block 12 is to define a user identifier. A sequence generator 14 generates sequences of modification keys on the basis of the user identifier, generating a different
20 sequence for each user. A modification block 16, which receives an input of a source file 10 and a user-specific sequence, forms from them a user-specific target file 17. The file is delivered through a distribution channel 18. In the following the blocks and functions will be described in greater detail.

The user identification block 12 generates a user identifier 12a. In
25 the present invention, any technique whatsoever known to the person skilled in the art can be used for identifying the user and generating the identifier 12a. The reliability of the identification technique should correspond to the importance of the information to be protected. In the field of electronic payment transfers, for example, it is common to use techniques that are based
30 on client identifiers and passwords that are valid only once. The structure and operation of the identification block 12 depends on the transmission path on which the communication takes place, i.e. the user identifier 12a is received and the target document 17 is delivered. If the document is delivered by mail, e.g. on a diskette, the user identifier 12a can be based, for example, on the
35 user's name or the number of his credit card. In the Internet, in particular, much information is distributed free of charge, whereby the information

provider profits e.g. by gaining publicity. In a case where no actual user identification takes place, information about the user's identity can be obtained on the basis of the electronic address, such as the TCP/IP address or e-mail address. Where particularly valuable information is concerned, the user
5 identifier can be checked e.g. by a call-back procedure.

The sequence generator 14 receives, as an input, a user identifier 12a, and generates a sequence 15 that is different for each user. The sequence 15 can be, for example, a series of pseudo random numbers. A simple way of forming pseudo random numbers is to raise a 2N-digit seed
10 number to the power of two, to take N middle digits of the number obtained, and to raise them to the power of two, and so on. The process is illustrated in fig. 2, where the arrow pointing downward indicates where the raising to the power of two is conducted. For example, when the user identifier 12a, which has the value 12345678, is raised to the power of two, and the four middle
15 digits of the number are taken, the first pseudo random number 15a - having the value 5765 - is obtained. When this number is raised to the power of two and the four middle numbers are taken, the second pseudo random number 15b - having the value 2352 - is obtained, and so on. If in such a sequence is obtained a number in which all the four middle digits are zeros, then the
20 sequence degenerates to zero. This can be avoided e.g. by interpreting the N-digit number as a bit string, which is then divided into parts and between the parts are placed '1' bits.

As an alternative to the embodiment of fig. 2, the user identifier 12a can be interpreted directly as a bit string from which a certain number of bits
25 are used at each modification. When all the bits have been used, one starts to use the bit string all over again. If, for example, the user identifier has 40 bits and three bits are used for forming each modification key, then the 14th modification key can consist of the last bit, after which the first two bits can be used again, and so on.

30 The source file 10 created contains or is associated with irrelevant information, or redundancy, which can be modified individually for each user. Image and voice information always contains such redundancy. For example, in the loudest moments of a piece of music, the human ear cannot distinguish whether the least significant bit is modified or not. In a movie, the position of
35 two frames can be changed without that the audience notices anything, and so on. Further, the elements of drawings can be in a different order in a file, and

any markings in the drawings can be in slightly different positions or be of different size.

A text file comprises notably less redundancy. Likewise, the information rate is much lower. The information content of a typical type-written page is 2 kilobytes, and it takes several minutes to read it, whereas the information rate of CD-quality music is 150 kilobytes per second, and that of a TV signal is several megabytes per second. Redundancy, however, can be added to a text file by utilizing the fact that many words have synonyms, that the word order can sometimes be changed, etc. Small and capital letters can be varied, e.g. GSM, Gsm or gsm. Some words also have alternative spellings, such as 'disc' or 'disk'. Small letter l can sometimes be replaced by number 1, and capital O by 0, or vice versa. The order of blocks in computer programs can vary in a file. Even the instructions of the computer programs have several alternative forms: for example, the addition of 1 or subtraction of -1 give the same result.

Alternative bit strings can thus be added to a text file or to a source file of a computer program, for example as follows: {alternative 1 / alternative 2 /.../ alternative N}.

The modification block 16 receives, as inputs, a source file 10 and a user-specific sequence 15. On the basis of the source file 10 and the sequence 15, the block creates a different target file 17 for each user. At each alternative bit string, the modification block 16 converts an element 15a-15e of the sequence 15 to a number range corresponding to the number N of the alternatives. The processing is described in greater detail in fig. 3. The modification block 16 reads a sentence 30 from the source file 10. The sentence, which serves as an example, contains two lists of alternative expressions 31 and 32, the former being here examined in greater detail. Reference 31a indicates where alternative list 31 begins. In this example, lists 31 and 32 begin with the sign {. Reference 31b indicates the first number of list 31, which shows how many of the alternatives of list 31 are used. If the alternative list 31 or 32 does not indicate the number of bit strings to be used, then one of the alternative bit strings is used. Here the list contains three alternatives 31c-31e, all of which are used. Reference 31f indicates a delimiter (here the sign /) between alternative bit strings, and reference 31g indicates the end of the list (here the sign }).

At each alternative list 31 and 32, the modification block 16 reads the random numbers 15a-15e, etc. and converts them to a number range corresponding to the number of alternatives. In the example of fig. 2, the random numbers 15a-15e vary within the range 0 to 9999, and the first list of alternatives 31 comprises three alternatives. The range of the random numbers can simply be divided into three parts, the limits being 3333 and 6666. The first random number 15a is within the range 3333 to 6666, so the elements of the first list of alternatives 31 can be used in the order second, third, first. In the second alternative list 32 there are two alternatives, of which one is used. The value of the second random number 15b is 2352, which is less than half of the range (there are two alternatives). From the second list of alternatives 32, the first element is used. The alternative bit strings need not all be equally probable. In particular, when an alternative bit string is based on a misspelling made on purpose, the correct form can be selected with a high - e.g. 90 to 99% - probability. In alternative lists 31 and 32, for example, it is possible to use an indication (e.g. a different delimiter 31f) showing that some of the alternatives are misspellings that are highly unlikely to be selected.

The number of possible permutations can be increased e.g. in such a way that when the random number 15a, 15b, etc. meets a certain additional condition, the list of alternatives is read from right to left. The additional condition can be, for example, that the random number is an even number or that it is closer to the upper limit of the part than to the lower limit. With only three alternatives, six permutations are thus obtained. In fig. 3 reference 33 indicates that even one simple sentence generates 12 different permutations.

Fig. 4 shows an embodiment in which the lists of alternative bit strings are not combined with the source file 10, as in the embodiment of fig. 3. In fig. 4 source file 10a comprises only substantial information, exemplified by sentence 40. The list of alternative bit strings is stored in its own file 10b, which contains files 10b1, 10b2, 10b3, etc. In this embodiment, the modification block 16 processes source file 10a. At sentence 40, which serves as an example, the modification block 16 recognizes bit strings 40a, 40b and 40c, which are also stored in alternative bit string file 10b. As the recognition takes place, the modification block 16 modifies the information read from source file 10b, and creates a target file 17 in almost the same way as described in connection with fig. 3. Even here 12 permutations can be

generated from a simple sentence 40 by three alternative bit strings 10b1-10b3.

The embodiments of figs. 3 and 4 can also be used together, as shown in fig. 5. In this embodiment, block 51 automatically identifies those bits strings (here 40a-40c) of the sentences (which are exemplified by sentence 40) of source file 10a for which an alternative bit string is defined in records 10b1-10b3 of file 10b. Block 51 automatically creates a source file 10 whose content is of the same form as sentence 30 in fig. 3. Such an automatically generated, combined source file can be supplemented manually before it is supplied to the modification block 16 according to the invention.

In figs. 3 and 4 the sentences 30 and 40 to be modified are simple sentences written in English. The sole reason for this is to make the operation of the invention more readily understandable. It is thus not essential to the invention what kind of information the files 10, 10a and 10b contain. The modification block 16 operates by simple mechanic rules of conclusion, and it does not understand the content of the source file 10 or target file 17. With respect to the modification block 16, the lists of alternatives 31, 32, etc. can be any bit strings whatsoever. In fact, the different permutations of sentence 30 in fig. 3 are not exact synonyms of one another; their information content, however, differs so little that the difference is not significant.

An alternative for storing possible modifications beforehand either in the source file 10 or in a separate modification file 10b is that the modifications are made on the basis of a suitable algorithm upon creating each separate target file 17, whereby the modifications can be stored in the modification block, or modification program 16. For example, let us assume that the supplier wants to protect a product catalogue intended for retailers, the catalogue possibly containing confidential information. The product catalogue contains product numbers, and to the corresponding bit strings extra redundancy can be added by interleaving in them bits formed from modification keys. With a 10-bit code, 1024 retailers can be separated from one another. For example, let the product number be 98765, the 20-bit binary presentation of which is 00011000000111001101. For example, one bit derived from the modification key can thus be interleaved between two bits of the product number. Each retailer would then receive a product list with individual product numbers. When a retailer makes an order, the extra bits are deleted automatically. If such a list of products is found in the possession of an

unauthorized user, the origin of the confidential information can be concluded from the bits added.

In the above embodiments the nature of the modification depends on the modification keys 15a-15e generated on the basis of the user identifier
5 12a. Alternatively, the modification keys can define the position of the modification rather than the nature of the modification. For example, a repetitive modification can be generated in a video signal, the distance between the modifications being defined on the basis of the modification keys 15a-15e. This kind of modification means, for example, that the mutual order
10 of two video frames is changed, or that the bits of a video frame are modified so that the checksum calculated from the bits is a predefined number, such as zero.

The structure of the distribution channel 17 is usually dependent on the user identification block 12. If the user orders a product by mail, the
15 distribution channel 17 refers, for example, to the delivery of a diskette by mail. Particular attention should be paid to distribution of electronic documents in an electric network, such as the Internet. The syntax of the lists of alternatives 31 and 32 described in fig. 3 can be implemented in an Internet server, for example as described below.

20 Let us now study fig. 6. In the prior art, information suppliers distribute information via servers, which are commonly called Web servers and are represented by a computer indicated by reference 63 in fig. 6. The PC user communicates with the Internet via a browser. The connection can be established, for example, via a modem 60 to the network operator's
25 communications server 61. Let us assume that the PC user wants to establish a connection with a Web server 63 having an identifier of the form 'http://www.xxx.yy', where xxx is a firm identifier, and yy is a specifier, such as a country code. The network forwards the request to a Domain Name Server (DNS) 62, which tells the TCP/IP address of server 63, on the basis of which
30 the network establishes a connection from the user's PC to server 63. The Web server 63 sends information to the user's PC, the information usually being a HyperText Markup Language (HTML) document. The service provider (copyright proprietor) can publish information in the Internet, for example, such that the Web server 63 is connected through a local area network 65 to other
35 computers 64 of the same firm, which produce the information. Alternatively, the server 63 can be the network operator's computer, whereby the service

provider can maintain the information contained in the server 63 either by sending diskettes to the operator or via a connection 60-61 like the one on which the PC user communicates with the Internet. The documents transmitted through the Internet can contain simple text or entire multimedia
5 programs.

When the user has found interesting information, he can give the browser the command 'print' or 'save', whereby the printer prints the information or the information is stored in the memory. The storing is problematic for the information provider, for the browsers usually store the
10 information with the HTML commands. The lists of alternative bit strings described in connection with figs. 3 to 5 must not be forwarded to the user under any circumstances whatsoever. In the invention, the operation of the Web server is expanded as follows. The HTML language is expanded by an additional command that can be e.g. the verb 'pick'. Sentence 30 of fig. 3,
15 expanded with the HTML in accordance with the present invention, would thus read as follows:

You are in a <pick 3, 'little', 'maze of', 'twisty'>
<pick 1, 'corridors', 'passages'>, all different.

As regards the verb 'pick', the Web server expanded in accordance
20 with the invention generates a different sentence for each user, as described in connection with fig. 3. The Web server forwards only the final sentence to the user via the Internet, the final sentence in the example of fig. 3 being one of sentences 33, from which any occurrences of the verb 'pick' and any unused alternatives have been deleted.

25 The Internet also comprises proxy servers, which are not shown separately in fig. 6. The proxy servers reduce the load on international connections by storing the last-read pages of information in the memory. If the same page is read several times in succession, the page is read once from the information provider's server, but after that from the proxy server. There is a
30 risk that if there are two users A and B and the latter copies information without authorization, the information content may refer to user A. In fact, the information copied without authorization does not include anything that would refer to the dishonest user B. However, user B can probably be exposed on the basis of the log file maintained by the operator. It is possible to conclude
35 from the information copied without authorization who the information has been delivered to. Proxy servers usually have a time limit, e.g. 24 hours, and

no information that is older than that is kept in the memory. It is then possible to conclude from the log who have requested information substantially simultaneously, so the number of suspected users is limited to a very small group of users.

5 It is a time-consuming process to prove an alleged copyright infringement in court. Technology can be used for preventing new infringements such that the Web server is provided with a list of blocked users that prevents (e.g. on the basis of the electronic address) creation of a service to a user that is suspected of having published information without
10 authorization.

 The copyright can be later proved more easily if it is possible to show - without any doubt - that the source file has been in existence on a certain day. This can be proved, for example, by calculating from the source file a multibyte cyclic checksum using a known algorithm, and publishing the
15 checksum in a means of communication.

 When the material provider later detects a document that he suspects to have been copied illegally, he can find out who the information in question has been delivered to, e.g. by testing which user identifier gives a document that is identical to the one that is suspected to be a pirate copy.
20 Since it is also possible that the user has modified the document he has 'borrowed', it is possible to define, for example, a certain coefficient of correlation or some other corresponding threshold value. To expose an unauthorized copy, it is not necessary that a whole document has been copied: copying of even part of a document is sufficient. As shown above,
25 even one simple sentence can yield 12 permutations. Two such sentences yield 144 permutations, and six sentences yield about three million. Since the copyright allows reasonable borrowing, it is not worthwhile to even try and expose small-scale borrowing.

 Since modifications have to be made in several locations of the
30 source file, it is evident that the only reasonable way to make the modifications is to use a computer or some other digital processor. The number of modifications needed must be estimated specifically for each case. As a rule, there must be so many modifications that even large-scale modification of a copied document can be exposed or at least made uneconomic. In the long
35 run, the information provider will have a large library of alternative expressions, which can be utilized in various documents. If all the alternative expressions

and other modifications are made by the same author as the actual text to be protected, the author owns the copyright on all the computer-made modifications. The computer and software do thus not add any contributions of their own that could interfere with the idea of the copyright.

- 5 It is naturally advantageous to use all the above techniques in different combinations. In multimedia documents, which comprise voice and/or video information as well as text, the different parts can be protected in different ways. Even in other respects it will be obvious to the person skilled in the art that as technology advances, the basic idea of the invention can be
- 10 implemented in many different ways. The invention and its embodiments are thus not limited to the above examples but can vary within the scope of the claims.



CLAIMS

1. A method of marking information to be produced in order to expose unauthorized publication of said information, the method comprising the steps of
- 5 - defining a user identifier (12a),
 - creating a source file (10) containing a substantial part of said information to be produced, and
 - creating a different target file (17) for each user on the basis of the source file (10) and the user identifier (12a),
- 10 **characterized** by further comprising the steps of
 - storing, in a file (10, 10b, 16), several predefined modification rules (10b1-10b3, 31, 32) for modifying the information contained in the source file (10),
 and, by means of a digital processor,
- 15 - generating a sequence of modification keys (15a, 15b,...) on the basis of the user identifier (12a),
 - defining the locations in the source file (10) to be modified and making modifications in these locations, determining the nature and/or position of the modifications on the basis of the predefined modification rules
- 20 (10b1-10b3, 31, 32) and modification keys (15a, 15b,...).
2. A method according to claim 1, **characterized** in that a number of predefined modifications (31, 32) are stored in the source file (10).
3. A method according to claim 1 or 2, **characterized** in that a number of predefined modifications (10b1-10b3) are stored in a separate
- 25 modification file (10b).
4. A method according to any one of claims 1 to 3, **characterized** in that a number of predefined modification rules are stored in a modification program (16).
5. A method according to any one of claims 1 to 4, **characterized**
- 30 **terized** in that the modifications are stored in the source file (10) as bit strings (31, 32) comprising at least a start character (31a), a stop character (31g), and a number of alternative bit strings (31c-31e), from which at least one is written into the source file (17) at the modification concerned, and that
- 35 the modifications are determined on the basis of the positions of the bit strings (31, 32) in the information contained in the source file (10).



6. A method according to claim 5, **characterized** in that the modifications are stored as bit strings (31, 32), which also define a number (31b) indicating how many alternative bit strings (31c-31e) are written into the target file (17) at the modification concerned.

5 7. A method according to claim 3, **characterized** by storing the modification rules (10b1-10b3) in a separate modification file (10b) as lists of alternative bit strings (10b1-10b3), and, when the source file (10) is modified to create a target file (17):

10 - recognizing, from the information (40) contained in the source file (10), bit strings (40a-40c) for which a list (10b1-10b3) of alternative bit strings has been defined, and

15 - in response to the recognition, replacing the bit string (40a-40c) of the source file with at least one alternative bit string obtained from the list (10b1-10b3) of bit strings that are alternative to the bit string (40a-40c) of the source file.

8. A method according to claim 4, **characterized** in that as the information passes from the information provider to the user, bits derived from the modification keys (15a-15e) are interleaved with bits of information contained in the source file (10), and in the reverse direction the bits added by
20 interleaving are ignored.

9. A method according to any one of the preceding claims, **characterized** in that the source file comprises a first part containing the actual information, and a second part containing formatting instructions, and that modifications are made at least in said first part.

25 10. A system for marking information to be produced in order to expose unauthorized publication of information, the system comprising

- means (12) for defining a user identifier (12a),
- a source file (10) containing a substantial part of the information to be produced, and

30 - processing means (16) for creating a different target file (17) for each user on the basis of the source file (10) and the user identifier (12a),

characterized in that the system comprises

- a file (10, 10b, 16) storing several predefined modification rules (10b1-10b3, 31, 32) in order to modify the information contained in the source
35 file (10),

This Sheet Has Been Replaced



RECTIFIED SHEET (RULE 91)

S BEEN
D

- processing means (14) for generating a sequence of modification keys (15a, 15b) on the basis of the user identifier (12a), and that

- the processing means (16) are arranged to define the locations of the source file (10) to be modified, and to make modifications in these
5 locations, the nature and/or position of the modifications being determined on the basis of the modification rules (10b1-10b3, 31, 32) and modification keys (15a, 15b,...).

This Sheet Has Been Replaced



THIS SHEET HAS BEEN
REPLACED