

EXHIBIT 1



(12) **United States Patent**
Mullor et al.

(10) **Patent No.:** **US 6,411,941 B1**
(45) **Date of Patent:** **Jun. 25, 2002**

(54) **METHOD OF RESTRICTING SOFTWARE OPERATION WITHIN A LICENSE LIMITATION**

(75) Inventors: **Miki Mullor; Julian Valiko**, both of Ramat Hasharon (IL)

(73) Assignee: **Beeble, Inc.**, Newport Beach, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/164,777**

(22) Filed: **Oct. 1, 1998**

(30) **Foreign Application Priority Data**

May 21, 1998 (IL) 124571

(51) **Int. Cl.**⁷ **G06F 17/60**

(52) **U.S. Cl.** **705/59; 705/50; 705/51; 705/53; 705/57**

(58) **Field of Search** **705/51, 54, 56, 705/57, 58, 59, 1, 50, 52, 53; 713/187, 189, 200**

(56) **References Cited**

U.S. PATENT DOCUMENTS

- 4,866,769 A 9/1989 Karp
- 4,903,296 A 2/1990 Chandra et al.
- 4,924,378 A 5/1990 Hershey et al.
- 5,386,369 A 1/1995 Christiano
- 5,390,297 A 2/1995 Barber et al.
- 5,479,639 A * 12/1995 Ewertz et al. 395/430
- 5,490,216 A * 2/1996 Richadson, III 380/4
- 5,671,412 A 9/1997 Christiano
- 5,684,951 A * 11/1997 Goodman et al. 395/188.01
- 5,754,763 A 5/1998 Bereiter
- 5,758,068 A 5/1998 Brandt et al.
- 5,758,069 A 5/1998 Olsen
- 5,790,664 A 8/1998 Coley et al.
- 5,826,011 A 10/1998 Chou et al.
- 5,892,900 A * 4/1999 Ginter et al. 395/186
- 5,905,860 A 5/1999 Olsen et al.

- 6,000,030 A * 12/1999 Steinberg et al. 713/200
- 6,006,190 A 12/1999 Baena-Arnaiz et al.
- 6,021,438 A 2/2000 Duvvoori et al.
- 6,023,763 A 2/2000 Grumpstrup et al.
- 6,052,600 A * 4/2000 Fette et al. 455/509
- 6,055,503 A 4/2000 Horstmann
- 6,067,582 A * 5/2000 Smith et al. 710/5
- 6,073,256 A 6/2000 Sesma
- 6,078,909 A 6/2000 Knutson
- 6,128,741 A 10/2000 Goetz et al.
- 6,173,446 B1 1/2001 Khan et al.
- 6,189,146 B1 * 2/2001 Misra et al. 717/11
- 6,192,475 B1 2/2001 Wallace
- 6,198,875 B1 * 3/2001 Edenson et al. 386/94
- 6,226,747 B1 5/2001 Larsson et al.
- 6,233,567 B1 5/2001 Cohen
- 6,243,468 B1 6/2001 Pearce et al.
- 6,272,636 B1 8/2001 Neville et al.
- 6,298,138 B1 10/2001 Gotoh et al.

FOREIGN PATENT DOCUMENTS

JP 408286906 A * 11/1996 G06F/9/06

OTHER PUBLICATIONS

Dornbusch et al., Destop management software: no need to adjust your set., Infoworld, v17, n37, p60.*

* cited by examiner

Primary Examiner—Hyung-Sub Sough

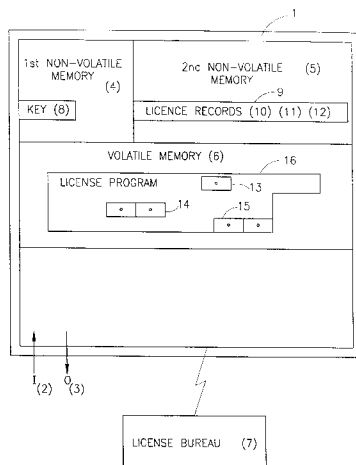
Assistant Examiner—Calvin L Hewitt

(74) *Attorney, Agent, or Firm*—Venable; Robert Kinberg; Jeffri A. Kaminski

(57) **ABSTRACT**

A method of restricting software operation within a license limitation that is applicable for a computer having a first non-volatile memory area, a second non-volatile memory area, and a volatile memory area. The method includes the steps of selecting a program residing in the volatile memory, setting up a verification structure in the non-volatile memories, verifying the program using the structure, and acting on the program according to the verification.

19 Claims, 2 Drawing Sheets



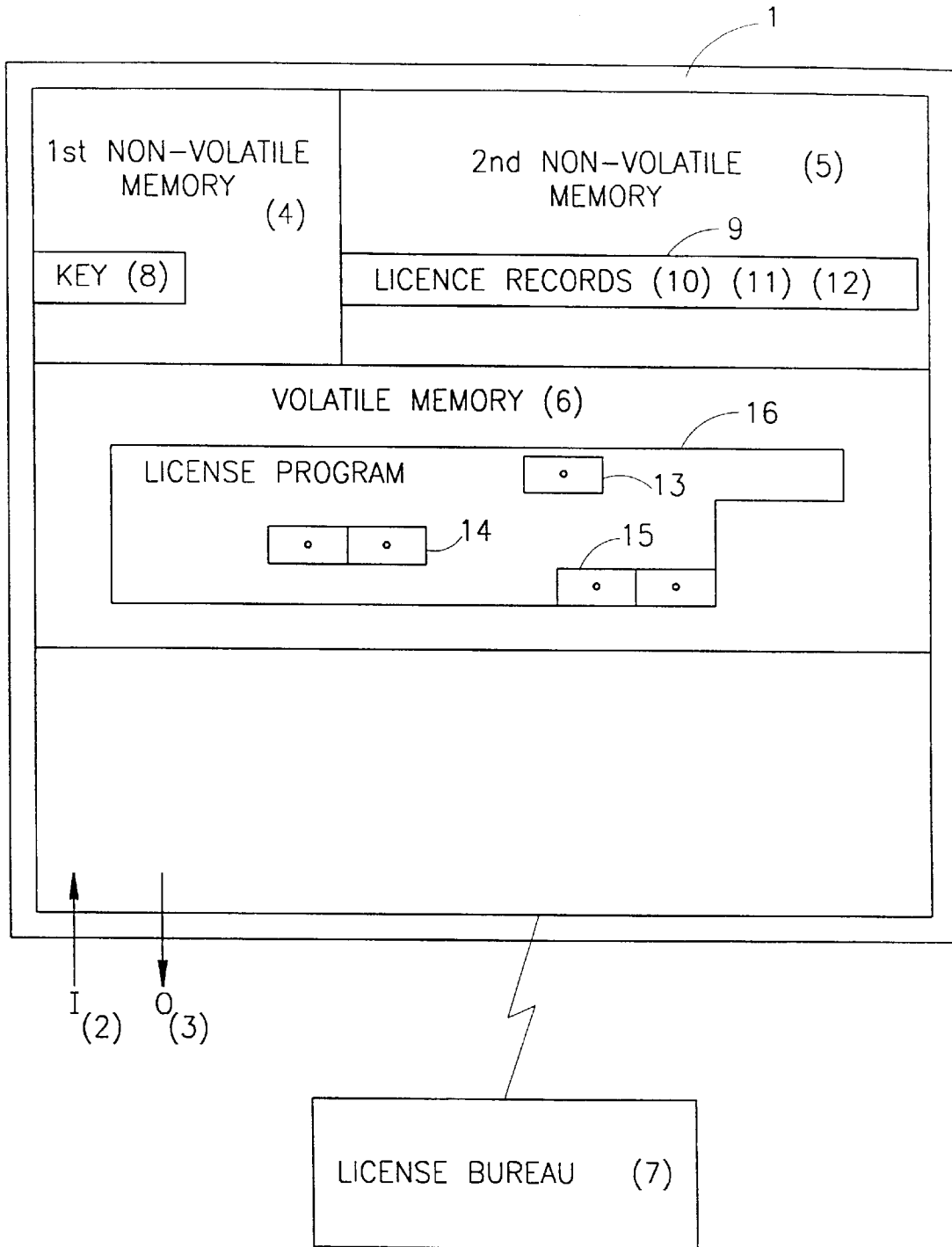


FIG. 1

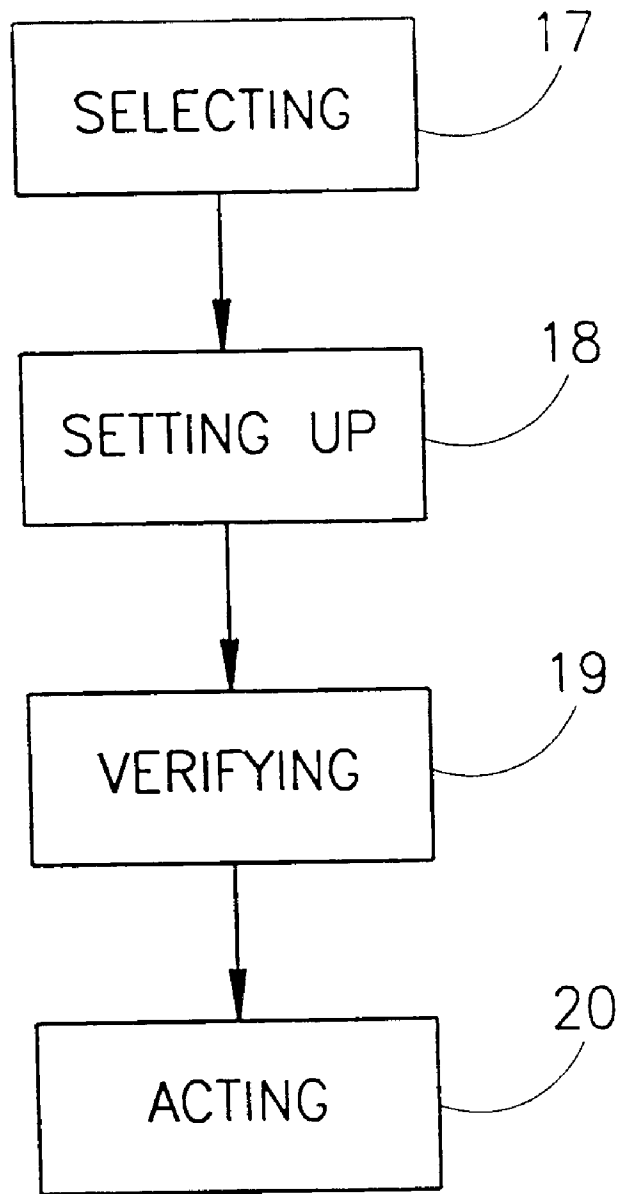


FIG.2

METHOD OF RESTRICTING SOFTWARE OPERATION WITHIN A LICENSE LIMITATION

FIELD OF THE INVENTION

This invention relates to a method and system of identifying and restricting an unauthorized software program's operation.

BACKGROUND OF THE INVENTION

Numerous methods have been devised for the identifying and restricting of an unauthorized software program's operation. These methods have been primarily motivated by the grand proliferation of illegally copied software, which is engulfing the marketplace. This illegal copying represents billions of dollars in lost profits to commercial software developers.

Software based products have been developed to validate authorized software usage by writing a license signature onto the computer's volatile memory (e.g. hard disk). These products may be appropriate for restricting honest software users, but they are very vulnerable to attack at the hands of skilled system's programmers (e.g. "hackers"). These license signatures are also subject to the physical instabilities of their volatile memory media.

Hardware based products have also been developed to validate authorized software usage by accessing a dongle that is coupled e.g. to the parallel port of the P.C. These units are expensive, inconvenient, and not particularly suitable for software that may be sold by downloading (e.g. over the internet).

There is accordingly a need in the art to provide for a system and method that substantially reduce or overcome the drawbacks of hitherto known solutions.

SUMMARY OF THE INVENTION

The present invention relates to a method of restricting software operation within a license limitation. This method strongly relies on the use of a key and of a record, which have been written into the non-volatile memory of a computer.

For a better understanding of the underlying concept of the invention, there follows a specific non-limiting example. Thus, consider a conventional computer having a conventional BIOS module in which a key was embedded at the ROM section thereof, during manufacture. The key constitutes, effectively, a unique identification code for the host computer. It is important to note that the key is stored in a non-volatile portion of the BIOS, i.e. it cannot be removed or modified.

Further, according to the invention, each application program that is to be licensed to run on the specified computer, is associated with a license record; that consists of author name, program name and number of licensed users (for network). The license record may be held in either encrypted or explicit form.

Now, there commences an initial license establishment procedure, where a verification structure is set in the BIOS so as to indicate that the specified program is licensed to run on the specified computer. This is implemented by encrypting the license record (or portion thereof) using said key (or portion thereof) exclusively or in conjunction with other identification information) as an encryption key. The resulting encrypted license record is stored in another (second) non-volatile section of the BIOS, e.g. E²PROM (or the

ROM). It should be noted that unlike the first non-volatile section, the data in the second non-volatile memory may optionally be erased or modified (using E²PROM manipulation commands), so as to enable to add, modify or remove licenses. The actual format of the license may include a string of terms that correspond to a license registration entry (e.g. lookup table entry or entries) at a license registration bureau (which will be further described as part of the preferred embodiment of the present invention).

Having placed the encrypted license record in the second non-volatile memory (e.g. the E²PROM), the process of verifying a license may be commenced. Thus, when a program is loaded into the memory of the computer, a so called license verifier application, that is a priori running in the computer, accesses the program under question, retrieves therefrom the license record, encrypts the record utilizing the specified unique key (as retrieved from the ROM section of the BIOS) and compares the so encrypted record to the encrypted records that reside in the E²PROM. In the case of match, the program is verified to run on the computer. If on the other hand the sought encrypted data record is not found in the E²PROM database, this means that the program under question is not properly licensed and appropriate application define action is invoked (e.g. informing to the user on the unlicensed status, halting the operation of the program under question etc.)

Those versed in the art will readily appreciate that any attempt to run a program at an unlicensed site will be immediately detected. Consider, for example, that a given application, say Lotus 123, is verified to run on a given computer having a first identification code (k1) stored in the ROM portion of the BIOS thereof. This obviously requires that the license record (LR) of the application after having been encrypted using k1 giving rise to (LR)_{k1} is stored in the E²PROM of the first computer.

Suppose now that a hacker attempts to run the specified application in a second computer having a second identification code (k2) stored in the ROM portion of the BIOS thereof. All or a portion the database contents (including of course (LR)_{k1}) that reside in the E²PROM portion in the first computer may be copied in a known per se means to the second computer. It is important to note that the hacker is unable to modify the key in the ROM of the second computer to K1, since, as recalled, the contents of the ROM is established during manufacture and is practically invariable.

Now, when the application under question is executed in the second computer, the license verifier retrieves said LR from the application and, as explained above, encrypts it using the key as retrieved from the ROM of the second computer, i.e. k2 giving rise to encrypted license record (LR)_{k2}. Obviously, the value (LR)_{k2} does not reside in the E²PROM database section of the second computer (since it was not legitimately licensed) and therefore the specified application is invalidated. It goes without saying that the data copied from the first (legitimate) computer is rendered useless, since comparing (LR)_{k2} with the copied value (LR)_{k1} results, of course, in mismatch.

The example above is given for clarity of explanation only and is by no means binding.

In its broadest aspect, the invention provides for a method of restricting software operation within a license limitation including; for a computer having a first non-volatile memory area, a second non-volatile memory area, and a volatile memory area; the steps of: selecting a program residing in the volatile memory, setting up a verification structure in the

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.