

MAILED
FEB 05 2002
Technology Center 2100

BW

THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re PATENT APPLICATION of

Applicants	:	Miki MULLOR et al.)	Customer No.
)	*26694*
Appln. No.	:	09/164,777)	26694
)	PATENT TRADEMARK
Filed	:	October 1, 1998)	OFFICE
)	
For	:	METHOD OF RESTRICTING SOFTWARE OPERATION WITHIN A LICENSED LIMITATION)	
)	
Group Art Unit	:	2161)	
Examiner	:	C. Hewitt)	
)	
Atty. Dkt.	:	39636-176166)	

Assistant Commissioner for Patents
Washington, D.C. 22031

AMENDMENT

Sir:

Responsive to the Office Action dated January 15, 2002, please amend the application as follows:

IN THE CLAIMS:

Please cancel claims 11, 12, 14 and 15 without prejudice to their re-entry at a later date.

Please amended the claims as follows:

C1 12/16. (Amended) The method of Claim 1, wherein a pseudo-unique key is stored in the non-volatile memory of the BIOS.

C2 18/20. (Amended) A method for accessing an application software program using a

Received from <202.962.8300> at 2/5/02 3:38:28 PM [Eastern Standard Time]

23

Q

pseudo-unique key stored in a first non-erasable non-volatile memory area of a computer, the first non-volatile memory area being unable to be programmatically changed, the method, comprising:

- CA
insp
D
- loading the application software program residing in a non-volatile memory area of the computer;
 - extracting license information from the software program;
 - encrypting license information using the pseudo-unique key stored in the first non-volatile memory area;
 - storing the encrypting license information in a second erasable, writable, non-volatile memory area of the BIOS of the computer;
 - subsequently verifying the application software program based on the encrypted license information stored in the second erasable, writable, non-volatile memory area of the BIOS; and
 - acting on the application software program based on the verification.

REMARKS

Claims 1-10, 13 and 16-23 are now pending in this application. Each of the pending claims is believed to define an invention which is novel and unobvious over the cited references. Favorable reconsideration of this case is respectfully requested.

Claims 16 and 20 have been amended to correct the informalities noted by the Examiner. Claims 11, 12, 14 and 15 have been canceled. In view of these amendments, it is respectfully submitted that all pending claims are now in all aspects in compliance with 35 U.S.C. 112, second paragraph. Therefore, the withdrawal of this rejection is respectfully requested.

24

C

Claims 1-23 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Misra et al. in view of U.S. Patent No. 5,684,951 to Goldman et al. and U.S. Patent No. 5,479,639 Ewertz et al.

The cited references do not render the present invention obvious as they do not teach or suggest, among other things, storing a verification structure, such as a software license information, in the BIOS of a computer as is recited in the present claims.

Misra et al. is cited as the primary reference against the present claims. Misra relates to a system and method for enforcing software licenses. The system of Misra generates unique identifiers for servers and clients, col 12, lines 41-42. The client system ID 142 is a unique identifier for the client computer, col 12, lines 50-51. The client system IDs can be based on information collected from a computer's hardware and installed software. For example, hard disk volume numbers, registered software, video cards, and some microprocessors contain unique identifiers. This information can be combined to uniquely identify a particular PC. Thus, the client system ID of Misra, is similar to the pseudo-unique key recited in claims 1 and 20.

Misra also describes a license ID, which is a unique identifier assigned to a software license when the software license is issued to a client device, col. 11, lines 9-12. The license ID may be a digital certificate indicating the right to use the particular software at issue, col. 10, lines 60-67. The license ID of Misra is similar to the verification structure and license information recited in claims 1 and 20, respectively.

Misra fails to teach using the BIOS of a computer to store the license ID, as noted in Section 7, Page 6 of the Office Action. Ewertz is cited as supplementing Misra to teach this feature. However, the license information described in Ewertz has a different meaning and a different function from the license information described in Misra. Therefore, a combination of these references would not result in the claimed invention, as is discussed in detail below.

In Ewertz, a “software license number” is described as one type of identification information, col. 3, lines 20-22. This identification information may also include an Ethernet address or system serial numbers, col 3, lines 20-22. The identification information is a unique identification value stored in a non-writable, non-erasable area of the BIOS during manufacture. The identification information uniquely identifies a particular computer. Therefore, according to Ewertz a “software license number” is one of a type of static data structures identifying a specific computer and the static data structure is stored such that it cannot be modified. Accordingly, the software license number of Ewertz is simply identification for the operating system of a particular computer.

For example, col. 2, lines 47-49 of Ewertz disclose that the memory storing the identification information may be electronically locked to prevent erasure or modification of its contents once installed. Moreover, in teaching a preferred embodiment, col. 11, line 23 - col. 12, line 14 of Ewertz describe that several types of identification information must be retained for individual computer systems. One type of identification number, as mentioned above, is an Ethernet address. The Ethernet address is stored in a protected area 306 in static page 2 of the flash memory of Ewertz and cannot be erased or altered once the device is installed. Thus the identification number cannot be destroyed. Ewertz also teaches other computer system identification numbers, such as unique serial number, printed board assembly (PBA) numbers or operating system license numbers may be stored in the locked memory.

Consequently, Ewertz teaches storing identification information for the computer in a non-writable, non-erasable non-volatile memory. This identification information of Ewertz corresponds to the pseudo-unique key stored in the first non-erasable, non-volatile memory as recited in claims 1 and 20 and does not correspond to the license information recited in these claims. The identification information of Ewertz is a static data structure, like the system ID of

C

Misra, that uniquely identifies a computer and simply does not correspond the license ID of Misra or the license information of the present invention as defined by claims 1 and 20.

From the above discussion, it is clear that the "software license number" according to Ewertz is equivalent in definition and function to Misra's system ID. Therefore, even if Misra is combined with Ewertz, this combination does not result in the present invention. The proposed combination results in the system ID of Misra being stored in the BIOS, not the verification structure or license information being stored in the BIOS as is required by the present claims.

Furthermore, there is no suggestion or motivation to combine Misra and Ewertz in the manner suggested in the Office Action. BIOS is a configuration utility. Software license management applications, such as the one of the present invention, are operating system (OS) level programs. Therefore, BIOS programs and software licensing management applications do not ordinarily interact or communicate because when BIOS is running, the computer is in a configuration mode, hence OS is not running. Thus, BIOS and OS level programs are normally mutually exclusive.

Ewertz teaches that writing to the BIOS area is performed by the BIOS routines:

"Referring to Fig. 8, processing logic for updating the flash memory device with configuration data, such as EISA information, is illustrated... The processing logic shown in Fig. 8 resides in the system BIOS of the preferred embodiment" Col 10, lines 20-28

Misra teaches a licensing system that is OS level based:

"The license generator 26, license server 28 and intermediate server 32 are preferably implemented as computer servers, such as Windows NT servers that run Windows NT server operating systems from Microsoft corporation or UNIX-based servers" Col 5, lines 3-7

Thus, the systems described in Misra and Ewertz are an OS program and a BIOS program, respectively, that cannot run at the same time. Therefore, there is no teaching or suggestion to combine these programs. In fact such a combination would change the operation

C

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.