(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 104991774 A
(43)申请公布日 2015.10.21

(21)申请号 201510388329.4

(22)申请日 2015.07.03

(71)申请人 武汉噢易云计算有限公司
地址 430074 湖北省武汉市东湖新技术开发区关山大道465号光谷创意大厦17层1701室

(72)发明人 曾丽星

(74)专利代理机构 武汉智权专利代理事务所（特殊普通合伙） 42225
代理人 张凯

(51)Int.Cl.
G06F 9/44(2006.01)
G06F 9/445(2006.01)

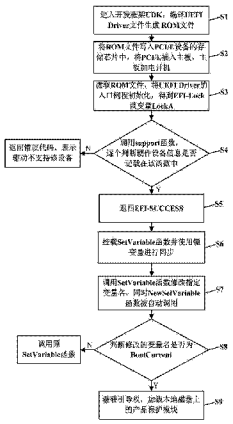
权利要求书2页 说明书6页 附图1页

(54)发明名称

一种UEFI平台下截获系统引导的系统和方法

(57)摘要

本发明公开了一种UEFI平台下截获系统引导的系统和方法，涉及计算机系统维护领域，该方法包括启动模块挂载变量配置模块，建立变量配置模块和判断模块的联系，使每次调用变量配置模块时判断模块都会被自动调用；UEFI平台调用变量配置模块修改指定变量名的变量值的同时，判定模块判定变量配置模块修改的变量名为当前信息记录模块中记载的当前引导设备的变量名，则截获当前引导设备的引导权，并加载本地磁盘上已安装的产品保护模块。本发明不依赖于传统的CSM支持模块，能够长期使用，能够提高计算机运行的稳定性和兼容性，可维护性较高。



CN 104991774 A

1.一种UEFI平台下截获系统引导的系统，其特征在于：该系统在UEFI平台下运行，该系统包括变量配置模块、当前信息记录模块、判断模块、ROM生成模块和启动模块；

变量配置模块用于修改UEFI平台指定变量名的变量值；当前信息记录模块用于记录当前引导设备的变量名；判断模块用于判断变量配置模块修改的变量名是否为当前信息记录模块中记载的变量名；启动模块用于挂载变量配置模块，并建立变量配置模块和判断模块的联系，使每次系统调用变量配置模块时判断模块都会被自动调用；

ROM生成模块用于将非ROM文件生成能够被UEFI平台识别的ROM文件；系统还包括带有存储芯片的PCI/E设备，该存储芯片用于存储ROM文件。

2.一种UEFI平台下截获系统引导的方法，其特征在于：该方法包括以下步骤：

A、启动模块挂载变量配置模块，建立变量配置模块和判断模块的联系，使每次调用变量配置模块时判断模块都会被自动调用；

B、UEFI平台调用变量配置模块修改指定变量名的变量值的同时，判定模块判定变量配置模块修改的变量名为当前信息记录模块中记载的当前引导设备的变量名，则截获当前引导设备的引导权，并加载本地磁盘上已安装的产品保护模块。

3.如权利要求2所述的UEFI平台下截获系统引导的方法，其特征在于：所述UEFI平台的开发框架为UDK开发包，所述UDK开发包包括根目录，所述步骤A之前还包括以下步骤：

进入开发包UDK的根目录并配置其编译环境，在该环境下编译生成UEFI Driver文件，所述UEFI Driver文件包括入口例程函数和Support函数；将ROM文件写入到PCI/E设备的存储芯片中，将带有存储芯片的PCI/E设备插入主板的插槽中，对主板加电开机；

UEFI平台主动读取存储芯片中ROM文件，并执行UEFI Driver的入口例程函数，UEFI Driver的入口例程函数进行初始化，并得到一个EFI_LOCK锁变量，该锁变量用于进行同步，防止在进行函数挂载时，被挂载的函数被UEFI平台调用；

UEFI平台调用Support函数，逐个判定计算机中的硬件设备信息在Support函数中有记载，Support函数返回成功代码EFI_SUCCESS。

4.如权利要求3所述的UEFI平台下截获系统引导的方法，其特征在于：所述启动模块通过Start函数来实现，判断模块通过NewSetVariable函数来实现，变量配置模块通过SetVariable函数来实现，当前信息记录模块记录的变量名为BootCurrent；

所述Support函数返回成功代码EFI_SUCCESS之后，还包括以下步骤，UEFI平台对返回EFI_SUCCESS的设备调用Start函数，Start函数挂载SetVariable函数，挂载过程中使用EFI_LOCK锁变量进行同步。

5.如权利要求4所述的UEFI平台下截获系统引导的方法，其特征在于，所述步骤B包括以下步骤：调用SetVariable函数修改UEFI平台指定变量名的变量值，同时，NewSetVariable函数被自动调用；

NewSetVariable函数判断UEFI平台调用SetVariable函数修改的变量名是否为BootCurrent，若是，截获当前引导设备的引导权，加载本地磁盘上已安装的产品保护模块。

6.如权利要求3所述的UEFI平台下截获系统引导的方法，其特征在于：所述驱动文件遵循UEFI平台驱动模型的规范。

7.如权利要求5所述的UEFI平台下截获系统引导的方法，其特征在于：所述NewSetVariable函数判断UEFI平台调用SetVariable函数修改的变量名是否为

BootCurrent 之后还包括以下步骤：NewSetVariable 函数判定当前修改的不是针对 BootCurrent 进行的修改，调用原 SetVariable 完成变量修改。

一种 UEFI 平台下截获系统引导的系统和方法

技术领域

[0001] 本发明涉及计算机系统维护领域，具体涉及一种 UEFI 平台下截获系统引导的系统和方法。

背景技术

[0002] 传统的 (Legacy)BIOS(Basic Input/Output System，基本输入输出系统）是一种固件，BIOS 保存着计算机最重要的基本输入输出的程序、开机后自检程序和系统自启动程序，其主要功能是为计算机提供最底层的、最直接的硬件设置和控制，并且担任操作系统控制硬件时的中介角色，是计算机系统硬件与上层软件之间的桥梁。

[0003] 随着计算机硬件和集成电路技术的飞速发展，BIOS 并没有随之发展，同时，由于传统的 BIOS 没有统一的标准和规范，不同品牌的使用环境不相同，不仅难以与不同的硬件兼容，而且当BIOS运行于 16 位实模式时，存在以下缺陷 ：a、主机的启动速度较慢，启动后计算机的硬件初始化和自检时间比较长 ；b、BIOS 需要使用大量的汇编语言代码，开发和维护成本较高 ；c、BIOS 向操作系统提供的服务需要通过有限的 16 位软中断来实现，BIOS 和 16 位软中断的耦合性较高，16 位软中断的的开发代价较高。因此，传统的 BIOS 很大程度上制约了计算机技术的发展。

[0004] 为了解决传统 BIOS 面临的问题，现有的 BIOS 标准和框架 UEFI(Unified Extensible Firmware Interface，统一可扩展固件接口）被提出，UEFI 是一种开放的用于定义平台固件与操作系统之间的接口规范，UEFI 不依赖于特定的 BIOS 和平台实现 ；UEFI 是为操作系统以及启动前的运行状态提供一个标准环境，系统地规定了计算机系统的控制权如何从启动前的环境传递给操作系统。

[0005] UEFI 平台框架本身不提供对传统操作系统以及传统 16 位代码支持，但为了更好的从传统 BIOS 平台过渡到 UEFI 平台，UEFI 平台提供 CSM(Compatibility Support Module 即兼容性支持模块）兼容支持模块对遗留的操作系统和其它 16 位代码进行支持。CSM 作为一个过渡方案，由于方案本身的限制，以及对启动性能的影响，在未来随着 WIN8 等新型操作系统的逐步普及，必将最终被移除。

[0006] 系统保护还原产品实现系统保护还原的基本原理是分析系统有效数据，通过对磁盘读写相关服务下钩子（即产品保护模块，用于防止系统的有效数据被破坏），对磁盘读写进行过滤映射，防止系统有效数据被破坏，以达到保护还原的目的。

[0007] 当非法用户从 U 盘设备或者网络设备引导非当前硬盘的操作系统后，产品将无法继续对非当前硬盘进行读写过滤，从而导致磁盘上的系统有效数据会被破坏，导致保护还原失效。因此防止非法用户从 U 盘或者网络设备引导是保护还原产品需要解决的一个重要问题。

[0008] 当前，依托 CSM 模块的支持，保护还原类产品在解决引导截获问题时，采用的一般方案为 ：在 PCI/E 设备中写入传统扩展 ROM(Read-Only Memory，只读内存），扩展 ROM 的初始化例程被加载执行后，挂载 19H 号（传统 BIOS 初始化完成后，调用该中断将控制权移交

给启动设备，让启动设备完成系统引导工作）中断来截获系统引导。这种方案由于需要与中断例程交互，通常代码需要使用汇编语言来完成，可维护性较差。而且不同 BIOS 产商实现标准不一致，导致代码在实现时需处理各种兼容。

发明内容

[0009] 针对现有技术中存在的缺陷，本发明的目的在于提供一种 UEFI 平台下截获系统引导的系统和方法，能够有效降低维护难度。

[0010] 为达到以上目的，本发明采取的技术方案是：一种 UEFI 平台下截获系统引导的系统，该系统在 UEFI 平台下运行，该系统包括变量配置模块、当前信息记录模块、判断模块、ROM 生成模块和启动模块；

[0011] 变量配置模块用于修改 UEFI 平台指定变量名的变量值；当前信息记录模块用于记录当前引导设备的变量名；判断模块用于判断变量配置模块修改的变量名是否为当前信息记录模块中记载的变量名；启动模块用于挂载变量配置模块，并建立变量配置模块和判断模块的联系，使每次系统调用变量配置模块时判断模块都会被自动调用；

[0012] ROM 生成模块用于将非 ROM 文件生成能够被 UEFI 平台识别的 ROM 文件；系统还包括带有存储芯片的 PCI/E 设备，该存储芯片用于存储 ROM 文件。

[0013] 一种 UEFI 平台下截获系统引导的方法，该方法包括以下步骤：

[0014] A、启动模块挂载变量配置模块，建立变量配置模块和判断模块的联系，使每次调用变量配置模块时判断模块都会被自动调用；

[0015] B、UEFI 平台调用变量配置模块修改指定变量名的变量值的同时，判定模块判定变量配置模块修改的变量名为当前信息记录模块中记载的当前引导设备的变量名，则截获当前引导设备的引导权，并加载本地磁盘上已安装的产品保护模块。

[0016] 在上述技术方案的基础上，所述 UEFI 平台的开发框架为 UDK 开发包，所述 UDK 开发包包括根目录，所述步骤 A 之前还包括以下步骤：

[0017] 进入开发包 UDK 的根目录并配置其编译环境，在该环境下编译生成 UEFI Driver 文件，所述 UEFI Driver 文件包括入口例程函数和 Support 函数；将 ROM 文件写入到 PCI/E 设备的存储芯片中，将带有存储芯片的 PCI/E 设备插入主板的插槽中，对主板加电开机；

[0018] UEFI 平台主动读取存储芯片中 ROM 文件，并执行 UEFI Driver 的入口例程函数，UEFI Driver 的入口例程函数进行初始化，并得到一个 EFI_LOCK 锁变量，该锁变量用于进行同步，防止在进行函数挂载时，被挂载的函数被 UEFI 平台调用；

[0019] UEFI 平台调用 Support 函数，逐个判定计算机中的硬件设备信息在 Support 函数中有记载，Support 函数返回成功代码 EFI_SUCCESS。

[0020] 在上述技术方案的基础上，所述启动模块通过 Start 函数来实现，判断模块通过 NewSetVariable 函数来实现，变量配置模块通过 SetVariable 函数来实现，当前信息记录模块记录的变量名为 BootCurrent；

[0021] 所述 Support 函数返回成功代码 EFI_SUCCESS 之后，还包括以下步骤，UEFI 平台对返回 EFI_SUCCESS 的设备调用 Start 函数，Start 函数挂载 SetVariable 函数，挂载过程中使用 EFI_LOCK 锁变量进行同步。

[0022] 在上述技术方案的基础上，所述步骤 B 包括以下步骤：调用 SetVariable 函数修改

UEFI 平台指定变量名的变量值，同时，NewSetVariable 函数被自动调用；

[0023] NewSetVariable 函数判断 UEFI 平台调用 SetVariable 函数修改的变量名是否为 BootCurrent，若是，截获当前引导设备的引导权，加载本地磁盘上已安装的产品保护模块。

[0024] 在上述技术方案的基础上，所述驱动文件遵循 UEFI 平台驱动模型的规范。

[0025] 在上述技术方案的基础上，所述 NewSetVariable 函数判断 UEFI 平台调用 SetVariable 函数修改的变量名是否为 BootCurrent 之后还包括以下步骤：NewSetVariable 函数判定当前修改的不是针对 BootCurrent 进行的修改，调用原 SetVariable 完成变量修改。

[0026] 与现有技术相比，本发明的优点在于：

[0027] (1) 本发明的 UEFI 平台下截获系统引导的方法，包括调启动模块挂载变量配置模块，建立变量配置模块和判断模块的联系，使每次调用变量配置模块时都会自动调用判断模块；UEFI 平台调用变量配置模块修改指定变量名的变量值的同时，判定模块判定变量配置模块修改的变量名为当前信息记录模块中记载的当前引导设备的变量名，并截获当前引导设备的引导权，加载本地磁盘上已安装的产品保护模块。该方法不依赖于传统的 CSM 支持模块，能够长期使用。

[0028] (2) 本发明的 UEFI 平台下截获系统引导的方法，是基于 UEFI 标准框架，能够提高稳定性和兼容性；同时，与传统的方法相比，能够避免挂接中断例程的方案，所有的流程在实现过程中，能够通过 C 语言完成，可维护性较高。

附图说明

[0029] 图 1 为本发明实施例中 UEFI 平台下截获系统引导的方法的流程图。

具体实施方式

[0030] 以下结合附图及实施例对本发明作进一步详细说明。

[0031] 参见图 1 所示，本发明实施例提供一种 UEFI(Unified Extensible Firmware Interface，统一的可扩展固件接口）平台下截获系统引导的系统和方法。

[0032] 该系统在 UEFI 平台下运行，该系统包括变量配置模块、当前信息记录模块、判断模块、ROM 生成模块和启动模块。

[0033] 变量配置模块用于修改 UEFI 平台指定变量名的变量值；判断模块用于判断变量配置模块修改的变量名是否为当前信息记录模块中记载的变量名；当前信息记录模块用于记录当前引导设备的变量名，启动模块用于挂载变量配置模块，并建立变量配置模块和判断模块的联系，使每次系统调用变量配置模块判断模块都会自动被调用。

[0034] ROM 生成模块用于将非 ROM 文件生成能够被 UEFI 平台识别的 ROM 文件；系统还包括带有存储芯片的 PCI/E 设备，该存储芯片用于存储 ROM 文件。

[0035] 在本实施例中，当前信息记录模块记录的变量名为 BootCurrent；ROM 生成模块通过 efirom 工具来实现，启动模块通过 Start 函数来实现，判断模块通过 NewSetVariable 函数来实现，变量配置模块通过 SetVariable 函数来实现。

[0036] 系统包括带有存储芯片的 PCI/E 设备，UEFI 平台的开发框架为 UDK(UEFI Development kit：UEFI 开发包)，开发包 UDK 包括根目录，进入其该根目录后才能运行 UDK。

[0037] UEFI 平台下截获系统引导的方法为：

[0038] A、启动模块挂载变量配置模块，建立变量配置模块和判断模块的联系，使每次调用变量配置模块时判断模块都会自动被调用。

[0039] B、UEFI 平台调用变量配置模块修改 UEFI 平台指定变量名的变量值的同时（UEFI 平台指定变量名可能为当前引导设备的变量名，也可能不是），判断模块判定变量配置模块修改的变量名是否为当前信息记录模块中记载的当前引导设备的变量名，若是，则截获当前引导设备的引导权，并加载本地磁盘上已安装的产品保护模块；否则，调用原变量配置模块完成变量修改。

[0040] 即：UEFI 平台在引导任何设备前都会调用 UEFI RUNTIME SERVICE 中的 SetVariable 函数，同时 NewSetVariable 函数会被自动调用。

[0041] 调用 SetVariable 函数修改 UEFI 平台指定变量名的变量值，同时，NewSetVariable 函数判断 SetVariable 函数修改的变量名是否为 BootCurrent，若是，截获当前引导设备的引导权，并加载本地磁盘上已安装的产品保护模块；否则，调用原 SetVariable 函数完成变量修改。

[0042] 参见图 1 所示，该方法具体包括以下步骤：

[0043] S1：进入开发包 UDK 的根目录并配置其编译环境，在该环境下编译生成 UEFI Driver 文件（驱动文件）。

[0044] UEFI Driver 文件遵守 UEFI Driver Model(UEFI 平台驱动模型）的规范，UEFI Driver 文件用于提供设备间接口协议，为每个设备独立运行提供设备版本号和相应的参数以及设备间关联，使得设备独立运行时不需要基于操作系统的支持。

[0045] UEFI Driver 文件包括文件目录、入口模块、信息存储比对模块。

[0046] 文件目录用于其它程序需要调用 UEFI Driver 文件时，直接调用该目录文件就可以调用 UEFI Driver 文件。

[0047] 入口模块：UEFI 平台调用该模块时才能进入 UEFI Driver 文件入口模块通过调用入口例程函数实现其功能。

[0048] 信息存储比对模块，该模块用于记录驱动支持的硬件设备的信息，并用于比对待引导的硬件设备是否为信息存储比对模块中记载的硬件设备。信息存储比对模块能够通过调用 Support 函数（支持函数）实现其功能。

[0049] UEFI Driver 文件遵循 UEFI Driver Model(UEFI 平台驱动模型）的规范，使用 efirom 工具，将 UEFI Driver 文件进行运算生成 ROM 文件，该 ROM 文件能够被 UEFI 平台识别。

[0050] S2：将 ROM 文件写入到 PCI/E 设备的存储芯片中，将带有存储芯片的 PCI/E 设备插入主板的插槽中，对主板加电开机。

[0051] S3：UEFI 平台主动读取存储芯片中 ROM 文件，并执行 UEFI Driver 的入口例程函数，UEFI Driver 的入口例程函数进行初始化，并得到一个 EFI_LOCK 锁变量，该锁变量用于进行同步，防止在进行函数挂载时，被挂载的函数被 UEFI 平台调用。

[0052] S4：UEFI 平台调用 Support 函数，逐个判断计算机中的硬件设备信息在 Support 函数中是否有记载，若是，转入步骤 S5；否则，返回错误码，表示该设备为不支持的设备。

[0053] S5：Support 函数返回成功代码 EFI_SUCCESS（比对成功的代码），转入步骤 S6。

[0054] S6：UEFI 平台对返回 EFI_SUCCESS 的设备调用 Start 函数，Start 函数挂载 SetVariable 函数，挂载过程中使用 EFI_LOCK 锁变量进行同步，转入步骤 S7；

[0055] S7：调用 SetVariable 函数修改 UEFI 平台指定变量名的变量值，同时，NewSetVariable 函数被自动调用，转入步骤 S8；

[0056] S8：NewSetVariable 函数判断 UEFI 平台调用 SetVariable 函数修改的变量名是否为 BootCurrent，若是，转入步骤 S9；否则，调用原 SetVariable 完成变量修改；

[0057] S9：截获当前引导设备的引导权，加载本地磁盘上已安装的产品保护模块。

[0058] 下面，通过一个具体实施例对本发明进行详细说明。

[0059] 参见图 1 所示，定义使用的 UEFI 的开发包为 UDK2014（一种开发包），PCI/PCIE 设备为型号为 realtek 8139 的网卡 CardA，该网卡包括存储芯片。UDK2014 包含 efirom 工具，用于将文件生成能够被 UEFI 平台识别的 ROM 文件：UEFI Option ROM 文件。

[0060] 在上述条件下实现 UEFI 平台下截获系统引导的方法包括以下步骤：

[0061] UEFI 平台包括 BootCurrent、SetVariable 函数和 NewSetVariable 函数，BootCurrent 为当前引导设备的信息。

[0062] 步骤一、进入 UDK2014 的根目录并配置其编译环境，在编译环境下编译生成 UEFI Driver 文件 FileA.efi 及其独立的文件目录，FileA.efi 在文件目录中。

[0063] FileA.efi 包括 EntryPointA 函数、SupportA 函数和 StartA 函数。

[0064] EntryPointA 函数为入口例程函数，UEFI 平台调用该函数 FileA.efi 驱动进行初始化。

[0065] SupportA 函数用于记载 FileA.efi 驱动支持的硬件设备的信息，同时，SupportA 函数还用于比对计算机中的硬件设备是否为 SupportA 函数中记载的硬件设备。

[0066] 步骤一、进入文件 FileA.efi 的目录，使用 UDK2014 中的 efirom 工具将 FileA.efi 生成 UEFI OptionROM 文件 FileB.rom。

[0067] 步骤二、将 FileB.rom 写入到 PCI/E 设备 CardA 的存储芯片中，将 CardA 插入主板插槽上，主板加电开机。

[0068] 步骤三、UEFI 平台自动加载 CardA 存储芯片中的 FileB.rom，并执行文件 FileA.efi 的 EntryPointA 函数，EntryPointA 函数进行初始化，并得到 EFI_LOCK 锁变量 LockA，该锁变量 LockA 用于进行同步，防止挂载 SetVariable 函数时，SetVariable 函数被 UEFI 平台调用。

[0069] 同时，UEFI 平台调用 SupportA 函数，逐个判断计算机中的硬件设备信息在中是否有记载，若是，返回 EFI_SUCCESS（能够比对成功的代码），转入步骤四；否则，返回错误码，表示驱动不支持该设备。

[0070] 步骤四、UEFI 平台对返回 EFI_SUCCESS 的设备调用 StartA 函数，StartA 函数挂载 SetVariable 函数，使用 LockA 对挂载操作进行同步。

[0071] 步骤五、UEFI 平台调用 SetVariable 函数修改 UEFI 平台指定变量名的变量值时，FileA.efi 中的 NewSetVariable 函数会被自动调用，转入步骤六。

[0072] 步骤六、NewSetVariable 函数判断 UEFI 平台调用 SetVariable 函数修改的变量名是否为 BootCurrent，若是，截获当前引导设备的引导权，并加载本地磁盘上已安装的产品保护模块；否则，调用原 SetVariable 完成变量修改。

[0073] 通过上述方法，能够有效保护还原产品保护模块在本地磁盘上安装后，无论非法用户选择从何种设备引导，都将会被截获而无法跳过产品保护模块，进而始终保证本地磁盘有效数据不会被非法破坏。

[0074] 本发明不局限于上述实施方式，对于本技术领域的普通技术人员来说，在不脱离本发明原理的前提下，还可以做出若干改进和润饰，这些改进和润饰也视为本发明的保护范围之内。本说明书中未作详细描述的内容属于本领域专业技术人员公知的现有技术。

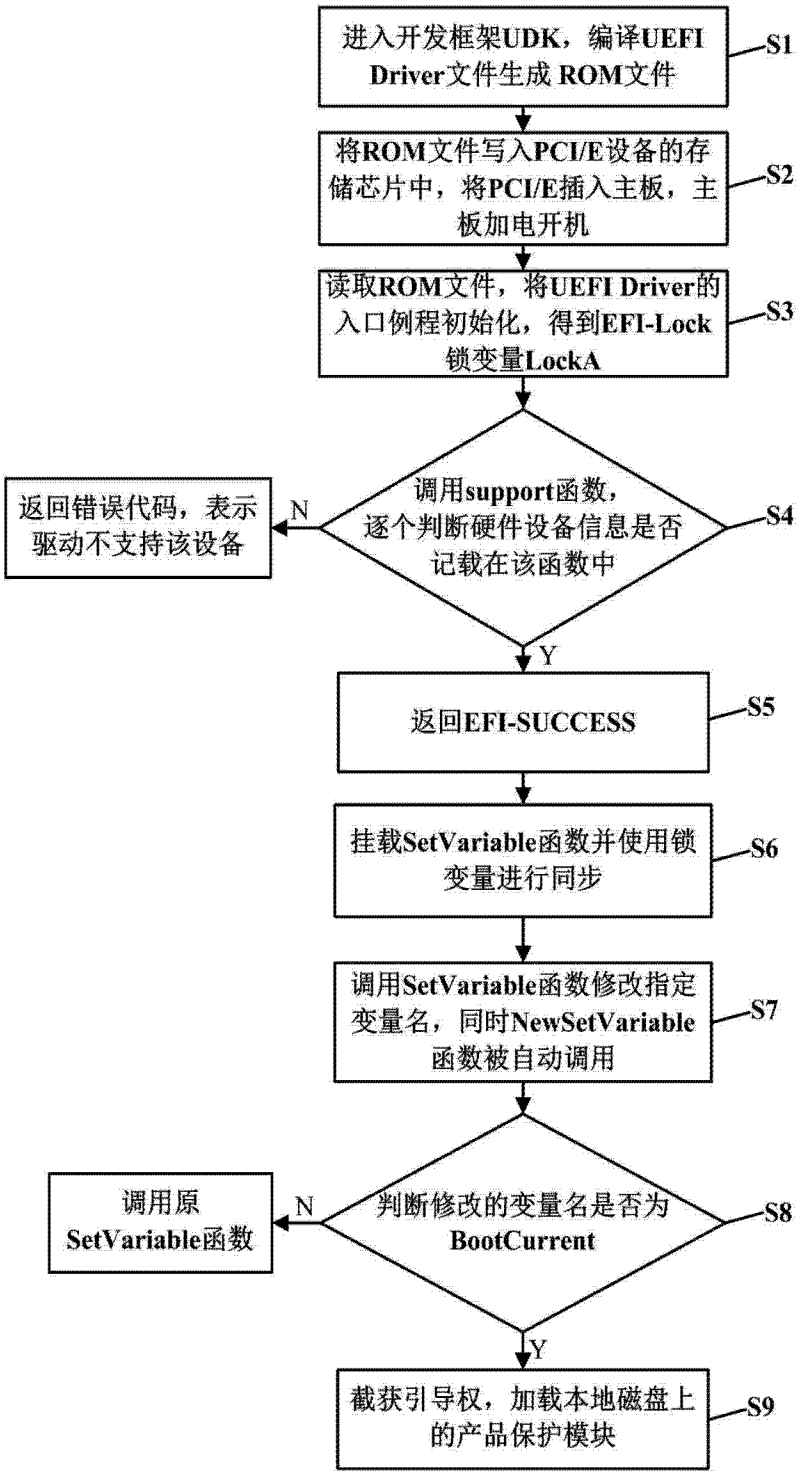

进入开发框架UDK，编译UEFI Driver文件生成 ROM文件
S1
将ROM文件写入PCI/E设备的存储芯片中，将PCI/E插入主板，主板加电开机
S2
读取ROM文件，将UEFI Driver的入口例程初始化，得到EFI-Lock锁变量LockA
S3
调用support函数，逐个判断硬件设备信息是否记载在该函数中
S4
N
返回错误代码，表示驱动不支持该设备
Y
返回EFI-SUCCESS
S5
挂载SetVariable函数并使用锁变量进行同步
S6
调用SetVariable函数修改指定变量名，同时NewSetVariable函数被自动调用
S7
判断修改的变量名是否为BootCurrent
S8
N
调用原SetVariable函数
Y
截获引导权，加载本地磁盘上的产品保护模块
S9

图 1