

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2012 年 11 月 15 日 (15.11.2012) WIPO | PCT



(10) 国际公布号
WO 2012/152210 A1

- (51) 国际专利分类号:
G06F 9/46 (2006.01)
- (21) 国际申请号: PCT/CN2012/075145
- (22) 国际申请日: 2012 年 5 月 7 日 (07.05.2012)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
201110121180.5 2011 年 5 月 11 日 (11.05.2011) CN
- (71) 申请人 (对除美国外的所有指定国): 北京奇虎科技有限公司 (BEIJING QIHOO TECHNOLOGY COMPANY LIMITED) [CN/CN]; 中国北京市西城区新街口外大街 28 号 D 座 112 室 (德胜园区), Beijing 100088 (CN)。
- (72) 发明人; 及
- (75) 发明人/申请人 (仅对美国): 王宇 (WANG, Yu) [CN/CN]; 中国北京市朝阳区建国路 71 号惠通时代广场 D 座, Beijing 100025 (CN)。 潘剑锋 (PAN, Jianfeng) [CN/CN]; 中国北京市朝阳区建国路 71 号惠通时代广场 D 座, Beijing 100025 (CN)。
- (74) 代理人: 北京润泽恒知识产权代理有限公司 (BEIJING RISEHIGH INTELLECTUAL PROPERTY LAW FIRM); 中国北京市海淀区中关村南大街 31 号神舟大厦 1116, Beijing 100081 (CN)。
- (81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。
- (84) 指定国 (除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)。

[见续页]

(54) Title: METHOD AND DEVICE FOR EXECUTING FILE OPERATION

(54) 发明名称: 一种文件操作的执行方法及装置

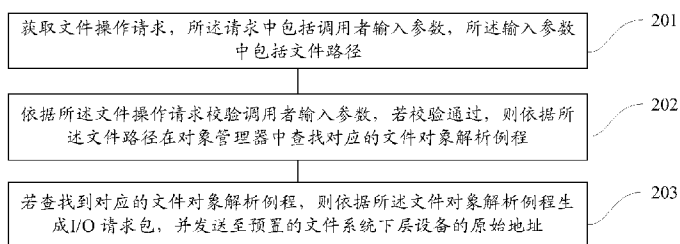


图 2 /Fig.2

201 ACQUISITION OF THE FILE OPERATION REQUEST, WHERE THE REQUEST COMPRISES A CALLER INPUT PARAMETER, AND THE INPUT PARAMETER COMPRISES THE FILE PATH
202 VERIFICATION OF THE CALLER INPUT PARAMETER ON THE BASIS OF THE FILE OPERATION REQUEST, IF THE VERIFICATION PASSES, THEN SEARCHING FOR THE CORRESPONDING FILE OBJECT PARSING ROUTINE IN THE FILE OBJECT MANAGER ON THE BASIS OF THE FILE PATH
203 IF THE CORRESPONDING FILE OBJECT PARSING ROUTINE IS FOUND, THEN GENERATION OF THE I/O REQUEST PACKET, AND TRANSMISSION TO THE ORIGINAL ADDRESS OF THE PRECONFIGURED FILE SYSTEM LOW-LEVEL DEVICE

(57) Abstract: Provided in the present application is a method for executing a file operation, comprising: acquiring a file operation request, where the request comprises a caller input parameter, and the input parameter comprises a file path; searching for a file object parsing routine in an object manager on the basis of the file path; if a corresponding file object parsing routine is found, then generating an I/O request packet on the basis of the file object parsing routine, and transmitting to an original address of a preconfigured file system low-level device. The present application improves the resistive capability against offence and defense of driver-level malicious programs, and prevents the potential possibility of incompatibility between security software caused by interference from file operation.

[见续页]



WO 2012/152210 A1



本国际公布:

— 包括国际检索报告(条约第 21 条(3))。

(57) 摘要:

本申请提供了一种文件操作的执行方法, 包括: 获取文件操作请求, 所述请求中包括调用者输入参数, 所述输入参数中包括文件路径; 依据所述文件路径在对象管理器中查找对应的文件对象解析例程; 若查找到对应的文件对象解析例程, 则依据所述文件对象解析例程生成 I/O 请求包, 并发送至预置的文件系统下层设备的原始地址。本申请可以增强和驱动级恶意程序攻防的对抗能力, 并避免因文件操作干扰而导致安全软件间产生不兼容的潜在可能性。

说明书

一种文件操作的执行方法及装置

技术领域

本申请涉及文件穿透的技术领域，特别是涉及一种文件操作的执行方法和一种文件操作的执行装置。

背景技术

在面对复杂问题的时候，人们往往会采用分而治之的方法分割、缩小问题的范围，这一点在操作系统的设计中亦是如此，Windows 操作系统借助分层的设计思想解决复杂的问题，这一做法带来了可移植性、可扩展性等优点。不过，由于在设计理论上存在着安全性缺陷（例如缺乏完整性校验机制等），高可扩展性的另一方面也意味着系统存在大量的被篡改可能。以文件系统的设计为例，文件调用栈的分层式结构决定了其调用链上存在数据流被篡改的风险。正因为如此，保证自身文件操作过程的真实、可信成了安全软件的一个必然要求。

参考图 1 所示的操作系统文件操作执行流的示意图，在执行文件操作采用如下分层调用的方式：

调用者 101 调用内核接口层 102，内核接口层 102 调用内核执行层 103，内核执行层 103 调用文件对象解析例程 104，文件对象解析例程 104 调用过滤驱动 a105，顶层过滤驱动逐层调用直至调用到底层过滤驱动 N106，底层过滤驱动 N106 调用文件系统下层设备 107。其中，调用者 101 和内核接口层 102 属于操作系统用户态，内核执行层 103、文件对象解析例程 104、顶层过滤驱动 a105、底层过滤驱动 N106 和文件系统下层设备 107 属于操作系统内核态。

从操作系统的角度看，文件操作执行流存在着如下潜在的篡改点：

1) 内核接口层的用户态 IAT Hook (Import Address Table Hook, 导入表钩挂) /EAT Hook (Export Address Table Hook, 导出表钩挂)；

- 2) 内核接口层的用户态 Inline Hook (内联钩挂);
- 3) 内核接口层调用内核执行层时的 Int 2E(中断)/SysEnter Hook(用户态进入内核态钩挂);
- 4) 内核执行层的 Native API SSDT Hook (System Service Dispatch Table Hook, 系统服务分派表钩挂);
- 5) 内核执行层的 Native API Inline Hook;
- 6) 文件对象解析例程的 Object Parse Routine Hook (对象解析例程钩挂);
- 7) 顶层过滤驱动 a 的顶层过滤驱动获取;
- 8) 底层过滤驱动 N 的底层过滤驱动获取。

传统安全软件厂商虽然意识到了种种文件操作调用被劫持的可能性,但是现有的解决方案大多只考虑了操作系统用户态的劫持风险,在操作系统内核态的攻防上往往体现出一定的能力缺失,驱动级恶意程序(Rootkit)对抗能力偏弱。

因此,目前需要本领域技术人员迫切解决的一个技术问题就是:提出一种文件穿透操作的处理机制,不仅提供操作系统用户态的攻防,还进一步提供操作系统内核态的攻防,以增强和驱动级恶意程序攻防的对抗能力,并避免因文件操作干扰而导致安全软件间产生不兼容的潜在可能性。

发明内容

本申请的目的之一是提供一种文件操作的执行方法,以增强和驱动级恶意程序攻防的对抗能力,并避免因文件操作干扰而导致安全软件间产生不兼容的潜在可能性。

本申请还提供了一种文件操作的执行装置,用以保证上述方法在实际中的应用及实现。

为了解决上述问题,本申请实施例公开了一种文件操作的执行方法,包括:

获取文件操作请求，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

依据所述文件路径在对象管理器中查找对应的文件对象解析例程；

若查找到对应的文件对象解析例程，则依据所述文件对象解析例程生成 I/O 请求包，并发送所述 I/O 请求包至预置的文件系统下层设备的原始地址。

优选地，所述依据文件路径在对象管理器中查找对应的文件对象解析例程的步骤具体包括以下子步骤；

子步骤 S1、判断文件路径是否已经拆解完毕，若否，则执行子步骤 S2；若是，则执行子步骤 S4；

子步骤 S2、按照路径分隔符拆解出文件路径中下一个待拆解的路径段；

子步骤 S3、采用当前拆解出的路径段在对象管理器中搜索，判断是否存在对应的文件对象例程；若是，则返回子步骤 S1；若否，则执行子步骤 S5；

子步骤 S4、获得所述文件路径对应的文件对象解析例程。

子步骤 S5、返回未找到对应文件对象解析例程的信息。

优选地，所述调用者输入参数具有用户态地址；在查找文件对象解析例程之前，所述的方法还包括：

重构所述用户态地址至内核态内存空间。

优选地，所述 I/O 请求包中包括从文件操作请求中提取的文件操作信息，在发送 I/O 请求包至预置的文件系统下层设备的原始地址之后，还包括：

由所述文件系统下层设备依据所述文件操作信息继续执行对应的文件操作。

优选地，在获取文件操作请求之前，还包括：

调用者发起文件操作请求，调用对应的文件操作接口例程；其中，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

依据系统平台的类型构建内核态结构参数，依据所述内核态结构参数生成相应的文件操作控制码，并发送所述文件操作控制码至操作系统内核态驱动。

优选地，在构建内核态结构参数之前，还包括：

文件操作接口例程将调用者输入参数中的 ANSI 相关参数转换为 UNICODE 类型，并调用对应的文件操作接口宽字符例程。

优选地，所述文件操作接口例程包括文件创建例程 FSCreateFile，在由所述文件系统下层设备继续执行所请求的文件操作之前，还包括：

插入新创建的文件对象至对象管理器，并得到返回的文件句柄。

优选地，所述的方法，还包括：

依据所述文件操作请求校验调用者输入参数，若校验通过，则执行查找文件对象解析例程的步骤。

优选地，所述文件操作接口例程与 WINDOWS 标准 API 一致，所述文件操作接口例程还包括：文件读取例程 FSReadFile、文件写入例程 FSWriteFile、文件属性设置例程 FSSetFileAttributes、文件属性获取例程 FSGetFileAttributes、文件指针设置例程 FSSetFilePointer、增强文件指针设置例程 FSSetFilePointerEx、文件大小获取例程 FSGetFileSize、文件删除例程 FSDeleteFile、目录移除例程 FSRemoveDirectory、句柄关闭例程 FSCloseHandle、第一个文件查找例程 FSFindFirstFile、下一个文件查找例程 FSFindNextFile、文件查找关闭例程 FSFindClose、增加文件属性获取例程 FSGetFileAttributesEx、判断路径是否为目录的例程 FSPathIsDirectory、判断目标文件是否存在的例程 FSPathFileExists、长路径获取例程 FSGetLongPathName、短路径获取例程 FSGetShortPathName、路径查找例程 FSSearchPath、增强文件大小获取例程 FSGetFileSizeEx、文件复制例程 FSCopyFile、文件移动例程 FSMoveFile 和/或增强文件移动例程 FSMoveFileEx。

优选地，所述控制码包括：文件创建操作控制码 FILE_IO_CREATE_FILE、文件读取操作控制码 FILE_IO_READ_FILE、

文件写入操作控制码 FILE_IO_WRITE_FILE、文件查询操作控制码 FILE_IO_QUERY_FILE、文件设置操作控制码 FILE_IO_SET_FILE 和/或文件关闭准备操作控制码 FILE_IO_PREPARE_CLOSE。

本申请实施例还公开了一种文件操作的执行装置，包括：

内核态请求获取模块，设置为获取文件操作请求，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

内核态对象解析模块，设置为依据所述文件路径在对象管理器中查找对应的文件对象解析例程；若查找到对应的文件对象解析例程，则调用内核 IRP 生成发送模块；

内核态 IRP 生成发送模块，设置为依据所述文件对象解析例程生成 I/O 请求包，并发送所述 I/O 请求包至预置的文件系统下层设备的原始地址。

优选地，所述内核对象解析模块具体包括以下模块；

文件路径拆解模块，设置为按照路径分隔符逐级拆解出文件路径中的路径段；

对象管理器搜索模块，设置为采用当前拆解出的路径段在对象管理器中搜索，查找对应的文件对象例程。

优选地，所述调用者输入参数具有用户态地址；所述的装置还包括：

内核地址重构模块，设置为重构所述用户态地址至内核态内存空间。

优选地，所述的装置，还包括：

用户态请求发送模块，设置为由调用者发起文件操作请求，调用对应的文件操作接口例程；其中，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

用户态控制码发送模块，设置为依据系统平台的类型构建内核态结构参数，依据所述内核态结构参数生成相应的文件操作控制码，并发送所述文件操作控制码至操作系统内核态驱动。

优选地，所述的装置，还包括：

宽字符例程调用模块, 设置为将调用者输入参数中的 ANSI 相关参数转换为 UNICODE 类型, 并调用对应的文件操作接口宽字符例程。

优选地, 所述文件操作接口例程包括文件创建例程 FSCreateFile, 所述的装置还包括:

句柄获取模块, 设置为插入新创建的文件对象至对象管理器, 并得到返回的文件句柄。

优选地, 所述的装置, 还包括:

内核态参数校验模块, 设置为依据所述文件操作请求校验调用者输入参数, 若校验通过, 则调用内核对象解析模块。

本申请实施例还公开了一种在其上记录有用于执行一种文件操作的执行方法的程序的计算机可读记录介质, 其中, 所述一种文件操作的执行方法, 包括:

获取文件操作请求, 所述请求中包括调用者输入参数, 所述输入参数中包括文件路径;

依据所述文件路径在对象管理器中查找对应的文件对象解析例程;

若查找到对应的文件对象解析例程, 则依据所述文件对象解析例程生成 I/O 请求包, 并发送至预置的文件系统下层设备的原始地址。

与现有技术相比, 本申请具有以下优点:

在操作系统用户态接口完整实现一套文件操作调用库, 调用者发起文件操作请求, 调用对应的文件操作接口例程, 操作系统内核态驱动获取并校验所述来自用户态的请求, 构建查询数据结构循环解析传入的文件路径, 最终查找到对象管理器中维护的对象类型, 这一过程有效对抗了内核态的劫持内险。此后, 操作系统内核态驱动构建并填充 IRP 请求数据包, 发送至预先确定的文件系统下层设备的原始地址处, 此时文件系统调用栈上的第三方过滤驱动, 包括其他安全软件和驱动级恶意程序可以被穿透, 从而既可以有效避免因文件操作干扰而导致和其他安全软

件间产生不兼容的潜在可能性；又可以增强和驱动级恶意程序攻防时的对抗能力。

本申请实施例中所采用的文件路径解析方法还能动态解析目标文件路径，例如，对于动态映射的网络磁盘驱动器，通过搜索对象管理器就能动态获得 DOS-Style 文件路径格式、盘符和文件系统下层设备对象之间的对应处理关系，因而本申请实施例还有适用范围广，适用场景多的优点。

附图说明

图 1 是操作系统文件操作执行流的示意图；

图 2 是本申请的一种文件操作的执行方法实施例 1 的步骤流程图；

图 3 是本申请中依据文件路径在对象管理器中查找对应的文件对象解析例程的步骤流程图；

图 4 是本申请的一种具体示例中在对象管理器中进行搜索的第一示意图；

图 5 是本申请的一种具体示例中在对象管理器中进行搜索的第二示意图；

图 6 是本申请的一种具体示例中在对象管理器中进行搜索的第三示意图；

图 7 是本申请的另一具体示例中在对象管理器中进行搜索的第四示意图；

图 8 是本申请的另一具体示例中在对象管理器中进行搜索的第五示意图；

图 9 是本申请的另一具体示例中在对象管理器中进行搜索的第六示意图；

图 10 是本申请的一种文件操作的执行方法实施例 2 的步骤流程图；

图 11 是应用本申请实施例实现的文件操作执行流示意图；

图 12 是本申请的一种文件操作的执行装置实施例的结构框图。

具体实施方式

为使本申请的上述目的、特征和优点能够更加明显易懂，下面结合附图和具体实施方式对本申请作进一步详细的说明。

本申请实施例的核心构思之一在于，在操作系统用户态接口完整实现一套文件操作调用库，调用者发起文件操作请求，调用对应的文件操作接口例程，操作系统内核态驱动获取并校验所述来自用户态的请求，构建查询数据结构循环解析传入的文件路径，最终查找到对象管理器中维护的对象类型，这一过程有效对抗了内核态的劫持内险。此后，操作系统内核态驱动构建并填充 IRP 请求数据包，发送至预先确定的文件系统下层设备的原始地址处，此时文件系统调用栈上的第三方过滤驱动（其他安全软件、驱动级恶意程序）被穿透（旁路，bypass）。简而言之，即本申请通过建立一条新的、可信的、可以穿过滤驱动的文件操作执行路径，有效避免了传统操作系统的文件执行路径上存在的风险。

参考图 2，示出了本申请的一种文件操作的执行方法实施例 1 的步骤流程图，具体可以包括以下步骤：

步骤 201、获取文件操作请求，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

需要说明的是，在本申请实施例中，所述文件包括 WINDOWS 操作系统中所支持类型的文件，所述文件操作是指针对文件的原子操作或原子操作的组合，所述原子操作包括：文件创建、文件读取、文件写入、文件属性设置、文件属性获取、文件指针设置、文件大小获取、文件删除、目录移除、句柄关闭、第一个文件查找、下一个文件查找、文件查找关闭、判断路径是否为目录、判断目标文件是否存在、长路径获取、短路径获取、路径查找、文件复制、文件移动等。例如，文件的杀毒操作即为文件读取、第一个文件查找、下一个文件查找、文件查找关闭、文件移动等原子操作的组合。

步骤 202、依据所述文件操作请求校验调用者输入参数，若校验通过，则依据所述文件路径在对象管理器中查找对应的文件对象解析例程；

参考图 3, 在本申请的一种优选实施例中, 所述依据文件路径在对象管理器中查找对应的文件对象解析例程的步骤具体可以包括以下子步骤;

子步骤 S1、判断文件路径是否已经拆解完毕, 若否, 则执行子步骤 S2; 若是, 则执行子步骤 S4;

子步骤 S2、按照路径分隔符拆解出文件路径中下一个待拆解的路径段;

子步骤 S3、采用当前拆解出的路径段在对象管理器中搜索, 判断是否存在对应的文件对象例程; 若是, 则返回子步骤 S1; 若否, 则执行子步骤 S5;

子步骤 S4、获得所述文件路径对应的文件对象解析例程。

子步骤 S5、返回未找到对应文件对象解析例程的信息。

在具体实现中, 可以预先构建对象管理器查询的 OpenPacket 结构, 基于路径分隔符“\”循环拆解文件路径, 例如, 文件路径为: c:\a\b.txt, 则第一次拆解出的路径段为 c:, 第二次拆解出的路径段为: c:\a, 第三次拆解出的路径段为: c:\a\b.txt, 即在本申请实施例, 是基于递归调用的方式拆解文件路径。

对象管理器 (Object Manager) 是 Windows NT 内核的一个基本组件。Windows NT 在设计的时候, “面向对象”的设计思想已大行其道, 即将原本散落在操作系统各处的资源集中抽象、封装起来, 进而为各种内部服务提供一致的访问途径。对象管理器主要用于实现以下功能: (1) 提供一种公共的、统一的机制来使用系统资源; (2) 将对象保护隔离到操作系统的统一区域中, 从而可以做到 C2 安全等级; (3) 提供一种机制来记录进程使用对象的数量, 从而可以对系统资源的使用加上限制; (4) 建立一套对象命名方案, 可以更方便地融合现有的对象。对象管理器共维护了几十种对象类型 (Windows 2000 是 27 种对象类型; Windows XP 是 29 种对象类型), 常见的如: 符号链接 (Symbolic Link)、进程 (Process)、线程 (Thread)、作业 (Job)、文件 (File)、事件 (Event)、定时器 (Timer)

等等。

对象管理器中维护有拉链式的对象 Hash 表，基于每次拆解出的路径段搜索对象管理器，若能查找到对应的对象解析例程 ParseProcedure，则继续下一次的文件路径拆解，并基于下一次拆解出的路径段和之前已拆解出的路径段搜索对象管理器，若当前的文件路径经过循环解析被完全拆解完毕，则经过搜索对象管理器找到的文件对象解析例程 Parse Routine 为当前文件路径对应的文件对象解析例程。

公知的是，例程是某个系统对外提供的功能接口或服务的集合。比如操作系统的 API、服务等就是例程。

为使本领域技术人员更好地理解本申请，以下结合图 4、图 5、图 6、图 7、图 8 和图 9 所示的对象管理器搜索示意图，通过一个具体例子详细说明本申请拆解文件路径搜索对象管理器的操作。

例如，文件路径为：c:\test\test.txt，c: 就是一个符号链接 (SymbolicLink)，对象管理器搜索实质上是一个符号链接的查询过程。

(1) 参考图 4，搜索从对象管理器的“根”开始 (即 “\”);

(2) 找到“\”下的 GLOBAL?? 目录;

(3) 根据 GLOBAL?? 目录找到“\”下的 GLOBAL?? 文件夹;

此后，继续在 GLOBAL?? 里寻找 c: (注意 C: 在图中属于 SymbolicLink)

(4) 参考图 5，从 GLOBAL?? 文件夹中找到 C:;

(5) 依据 C: 所对应的类型 (Type): SymbolicLink 和附加信息 (Additional Information): \Device\HarddiskVolume1 找到 Device 文件夹; 如有必要，会继续解析 \Device\harddiskVolume1 符号链接等等，直接无法拆解为止。

(6) 参考图 6，从 Device 文件夹下找到 HarddiskVolume1。

又如，文件路径为 “SystemRoot\System32\Drivers\ntfs.sys”，“SystemRoot”也是符号链接;

(1) 参考图 7，先找到 SystemRoot;

(2) 依据 SystemRoot 所对应的类型 (Type): SymbolicLink 和附加信息 (Additional Information): \Device\Harddisk0\Partition1\WINDOWS 找到 Device 文件夹;

(3) 参考图 8, 找到 Device 文件夹下的 \Harddisk0;

(4) 找到 \Harddisk0 下的 Partition1, 依据 Partition1 所对应的类型 (Type): SymbolicLink 和附加信息 (Additional Information): \Device\HarddiskVolume1 找到 Device 文件夹;

(5) 参考图 9, 找到 Device 文件夹下的 HarddiskVolume1。

在本例中, 路径拆解的依据是路径分隔符 “\”, 每找到一个分隔符, 就认为找到一段“因子”, 然后去搜索对象管理器, 如有必要再合并路径, 重新搜索对象管理器。在本实施例中, 搜索出的文件对象解析例程是一个函数, 这个函数在注册的时候就和某个因子对应, 它知道如何去正确地处理这些因子。

本申请实施例中所采用的文件路径解析方法能动态解析目标文件路径, 例如, 对于动态映射的网络磁盘驱动器, 通过搜索对象管理器就能动态获得 DOS-Style 文件路径格式、盘符和文件系统下层设备对象之间的对应处理关系, 因而本申请实施例还有适用范围广, 适用场景多的优点。

步骤 203、若查找到对应的文件对象解析例程, 则依据所述文件对象解析例程生成 I/O 请求包, 并发送至预置的文件系统下层设备的原始地址。

在具体实现中, 文件对象解析 Parse Routine 的实现类似于微软的 IoParseDevice 例程 (模拟实现), 其内部会构建并填充 I/O 请求包 (I/O Request Packet, IRP) 并发送至文件系统下层设备的原始地址处。在微软的 Windows 操作系统家族中, 都通过发送 IRP 来进行和驱动程序的通讯。用来封装 IRP 的数据结构不仅仅用来描述一个 I/O 操作的请求本身的内容, 还要用来维护这一请求在一系列驱动程序中传递的过程中的相关状态信息。也就是说 IRP 可以被定义为: 是 I/O 系统为了处理一个 I/O

请求而用来存储必要信息的地方。当一个线程调用一个 I/O 服务的时候，I/O 管理器构造 IRP，用于在 I/O 系统处理此请求的过程中代表此请求。

在具体实现中，所述文件系统下层设备的原始地址可以在系统初始化时设置。经过此步骤，文件系统调用栈上的第三方过滤驱动（其他安全软件、驱动级恶意程序）会被穿透，从而可以有效避免因文件操作干扰而导致和其他安全软件间产生不兼容的潜在可能性；还可以增强和驱动级恶意程序攻防时的对抗能力。

在实际中，所述 I/O 请求包中包括从文件操作请求中提取的文件操作信息，当 I/O 请求包（IPR）被发送至预置的文件系统下层设备的原始地址之后，则由所述文件系统下层设备依据所述文件操作信息继续执行对应的文件操作。具体而言，当 IRP 发送给文件系统的设备对象后，直到写入硬盘，还要经过一系列复杂的处理过程，一般而言，这个请求还要经过卷快照（Volsnap.sys）、卷管理器（Ftdisk.sys）、分区管理器（Partmgr.sys）、磁盘类驱动（disk.sys）、磁盘端口驱动（以 IDE 系统为例是 atapi.sys）、小端口驱动（以 Adaptec 1540 SCSI 为例是 Ahal54x.sys）等一系列的传递，最终由小端口驱动决定写入磁盘或磁带机的相应偏移。卷管理（包括快照）引入了动态的概念，使得 windows 可以创建多分区卷（比如镜像卷 mirrors，条带卷 stripes，RAID-5 等），该组件会根据实际情况将请求定位到目标卷的某个偏移。分区管理器负责通知即插即用管理器当前有哪些分区，以及他们的状态（创建、删除等）。磁盘类驱动实现了对于所有磁盘共有的功能，例如，SCSI（Small Computer System Interface，小型计算机系统接口）Port 是针对 SCSI 总线上的磁盘特性。最后小端口驱动针对某些厂商的特有产品特点，这类驱动往往由厂商自己提供。总的来说，就是每一层根据自己的接口，接收上层传递来的请求，以自己看到的“视图”，找到目标设备的某个扇区偏移，创建、写入、删除数据。

参考图 10，示出了本申请的一种文件操作的执行方法实施例 2 的步

骤流程图，具体可以包括：

步骤 401、加载文件操作接口例程，初始化文件系统下层设备的原始地址；

作为具体应用的一种示例，所述文件操作接口例程包括：文件创建例程 FSCreateFile、文件读取例程 FSReadFile、文件写入例程 FSWriteFile、文件属性设置例程 FSSetFileAttributes、文件属性获取例程 FSGetFileAttributes、文件指针设置例程 FSSetFilePointer、增强文件指针设置例程 FSSetFilePointerEx、文件大小获取例程 FSGetFileSize、文件删除例程 FSDeleteFile、目录移除例程 FSRemoveDirectory、句柄关闭例程 FSCloseHandle、第一个文件查找例程 FSFindFirstFile、下一个文件查找例程 FSFindNextFile、文件查找关闭例程 FSFindClose、增加文件属性获取例程 FSGetFileAttributesEx、判断路径是否为目录的例程 FSPathIsDirectory、判断目标文件是否存在的例程 FSPathFileExists、长路径获取例程 FSGetLongPathName、短路径获取例程 FSGetShortPathName、路径查找例程 FSSearchPath、增强文件大小获取例程 FSGetFileSizeEx、文件复制例程 FSCopyFile、文件移动例程 FSMoveFile 和/或增强文件移动例程 FSMoveFileEx。所述文件操作接口例程的设置，如调用约定、调用参数和对应的 WINDOWS 标准 API 一致。上述每个例程均包括窄字符例程和宽字符例程，例如对于 FSCreateFile 而言，包括窄字符例程 FSCreateFileA 和宽字符例程 FSCreateFileW。

步骤 402、调用者发起文件操作请求，调用对应的文件操作接口例程；其中，所述请求中包括调用者输入参数，所述输入参数中包括文件路径和用户态地址；

例如，调用者进程发起 FSCreateFileA 的文件创建请求。

步骤 403、所述文件操作接口例程的用户态部分将调用者输入参数中的 ANSI 相关参数转换为 UNICODE 类型，并调用对应的文件操作接口宽字符例程；

公知的是，ANSI 中的字符采用 8bit，而 UNICODE 中的字符采用

16bit。(对于字符来说 ANSI 以单字节存放英文字符,以双字节存放中文等字符,而 Unicode 下,英文和中文的字符都以双字节存放)。

以文件创建过程为例,为保证平台适用性, FSCreateFileA 会将调用者输入参数中的 ANSI 相关参数转换为 UNICODE 类型,并调用对应的文件操作接口宽字符例程 FSCreateFileW。

当然,若在实际中直接调用的是宽字符例程,则无需执行本步骤。

步骤 404、依据系统平台的类型构建内核态结构参数,依据所述内核态结构参数生成相应的文件操作控制码,并发送所述文件操作控制码至操作系统内核态驱动;

所述系统平台的类型包括 32 位、64 位以及 32 位兼容模式。作为本申请一种具体应用的示例,与所述文件操作接口例程对应的控制码包括:文件创建操作控制码 FILE_IO_CREATE_FILE、文件读取操作控制码 FILE_IO_READ_FILE、文件写入操作控制码 FILE_IO_WRITE_FILE、文件查询操作控制码 FILE_IO_QUERY_FILE、文件设置操作控制码 FILE_IO_SET_FILE 和 / 或 文件关闭准备操作控制码 FILE_IO_PREPARE_CLOSE。所述控制码定义了操作系统用户态和内核态驱动通信时的统一标识。

以文件创建过程为例, FSCreateFileW 通过判断系统平台类型(32 位、64 位或 32 位兼容模式)构建结构参数,发送控制码 FILE_IO_CREATE_FILE 并同步等待返回。

在实际中, FSCreateFileW 还可以处理畸形文件名、文件路径,并实际完成参数转换的操作。操作系统用户态和内核态驱动通信时输入、输出缓冲区可以采用 METHOD_BUFFERED 方式传输。METHOD_BUFFERED 方式为:先分配缓冲,再从这个缓冲 copy 数据,缓冲区大小是输入缓冲和输出缓冲间空间较大的那个。读缓冲被 copy 到新的缓冲。在返回前,只是 copy 返回值到相同的缓冲。返回值被放到 IO_STATUS_BLOCK, IO 管理器 copy 数据到输出缓冲。

步骤 405、操作系统内核态驱动获取文件操作请求,校验调用者输入

参数，并重构（Captured）所述用户态地址至内核态内存空间；

以文件创建过程为例，由 FSCreateFileW 例程内核部分将执行步骤，并在内核态内存空间处理相应的调用者输入参数。

步骤 406、若输入参数校验通过，则依据所述文件路径在对象管理器中查找对应的文件对象解析例程；

步骤 407、若查找到对应的文件对象解析例程，则依据所述文件对象解析例程生成 I/O 请求包，并发送至预置的文件系统下层设备的原始地址。

以文件创建过程为例，FSCreateFileW 的内核部分会校验用户态传入参数，构建 OpenPacket 结构，循环解析文件路径格式并搜索对象管理器维护的拉链式对象 Hash 表。具体可以采用路径分隔符“\”拆解输入的文件路径，拆解出的路径部分搜索对象管理器维护的拉链式对象 Hash 表，查找出对应的 ParseProcedure。在循环解析完毕时，便认为找到了对象的 Parse Routine 例程。Parse Routine 内部会构建并填充 IRP 请求数据包并发送至文件系统下层设备的原始地址处，完成文件穿透的创建过程。此时，文件系统调用栈上的第三方过滤驱动（其他安全软件、驱动级恶意程序）被旁路。

在具体实现中，对于新创建出的对象还可以通过 ObInsertObject 例程插入至对象管理器的 Hash 结构中，并得到返回的文件句柄。并且，内核还可以同步调用返回用户态句柄信息以及调用结果。如果调用失败，用户态接口可以设置相应的错误码，这样，调用者线程可以通过 GetLastError 例程得到详细的错误信息。

以下结合图 11 所示的应用本申请实施例实现的文件操作执行流示意图更进一步说明本申请的工作原理。

调用者 111 调用本实施例中实现的驱动接口层 112，而非调用现有技术中的内核接口层 113；

驱动接口层 112 调用模拟内核执行层校验调用者输入参数 114；

调用者输入参数验证通过后，执行搜索对象管理器循环解析文件路径 115 得到对象解析例程 116，以及，构建 IRP 发送至文件系统下层设备原始地址 117 的操作；

在此过程中，现有技术中的文件对象解析例程 118、顶层过滤驱动 a119、底层过滤驱动 N120 被穿透。

IRP 被发送至文件系统下层设备 121 后，由文件系统下层设备 121 执行 IRP 所请求的操作。

需要说明的是，对于方法实施例，为了简单描述，故将其都表述为一系列的动作组合，但是本领域技术人员应该知悉，本申请并不受所描述的动作顺序的限制，因为依据本申请，某些步骤可以采用其他顺序或者同时进行。其次，本领域技术人员也应该知悉，说明书中所描述的实施例均属于优选实施例，所涉及的动作和模块并不一定是本申请所必须的。

参考图 12，示出了本申请的一种文件操作的执行装置实施例的结构框图，具体可以包括以下模块：

内核态请求获取模块 121，设置为获取文件操作请求，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

内核态参数校验模块 122，设置为依据所述文件操作请求校验调用者输入参数，若校验通过，则调用内核对象解析模块 123；

内核态对象解析模块 123，设置为依据所述文件路径在对象管理器中查找对应的文件对象解析例程；若查找到对应的文件对象解析例程，则调用内核 IRP 生成发送模块 124；

内核态 IRP 生成发送模块 124，设置为依据所述文件对象解析例程生成 I/O 请求包，并发送所述 I/O 请求包至预置的文件系统下层设备的原始地址。

在本申请的一种优选实施例中，所述内核对象解析模块 123 具体可以包括以下模块；

文件路径拆解模块，设置为按照路径分隔符逐级拆解出文件路径中的路径段；

对象管理器搜索模块，设置为采用当前拆解出的路径段在对象管理器中搜索，查找对应的文件对象例程。

在具体实现中，所述调用者输入参数具有用户态地址；所述的装置还可以包括：

内核地址重构模块，设置为重构所述用户态地址至内核态内存空间。

在本申请的一种优选实施例中，还可以包括以下模块：

用户态请求发送模块，设置为由调用者发起文件操作请求，调用对应的文件操作接口例程；其中，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

用户态控制码发送模块，设置为依据系统平台的类型构建内核态结构参数，依据所述内核态结构参数生成相应的文件操作控制码，并发送所述文件操作控制码至操作系统内核态驱动。

在具体实现中，还可以包括以下模块：

宽字符例程调用模块，设置为将调用者输入参数中的 ANSI 相关参数转换为 UNICODE 类型，并调用对应的文件操作接口宽字符例程。

作为本申请实施例一种具体应用的示例，所述文件操作接口例程可以包括文件创建例程 FSCreateFile，所述的装置还可以包括以下模块：

句柄获取模块，设置为插入新创建的文件对象至对象管理器，并得到返回的文件句柄。

由于所述装置实施例基本相应于前述图 1、图 2 和图 3 所示的方法实施例，故本实施例的描述中未详尽之处，可以参见前述实施例中的相关说明，在此就不赘述了。

本申请实施例还公开了一种在其上记录有用于执行一种文件操作的执行方法的程序的计算机可读记录介质，其中，所述一种文件操作的执行方法可以包括如下步骤：

获取文件操作请求，所述请求中包括调用者输入参数，所述输入参

数中包括文件路径;

依据所述文件路径在对象管理器中查找对应的文件对象解析例程;

若查找到对应的文件对象解析例程, 则依据所述文件对象解析例程生成 I/O 请求包, 并发送至预置的文件系统下层设备的原始地址。

所述计算机可读记录介质包括用于以计算机(例如计算机)可读的形式存储或传送信息的任何机制。例如, 机器可读介质包括只读存储器(ROM)、随机存取存储器(RAM)、磁盘存储介质、光存储介质、闪速存储介质、电、光、声或其他形式的传播信号(例如, 载波、红外信号、数字信号等)等。

本申请可设置为众多通用或专用的计算系统环境或配置中。例如: 个人计算机、服务器计算机、手持设备或便携式设备、平板型设备、多处理器系统、基于微处理器的系统、置顶盒、可编程的消费电子设备、网络 PC、小型计算机、大型计算机、包括以上任何系统或设备的分布式计算环境等等。

本申请可以在由计算机执行的计算机可执行指令的一般上下文中描述, 例如程序模块。一般地, 程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本申请, 在这些分布式计算环境中, 由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中, 程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

以上对本申请所提供的一种文件操作的执行方法和一种文件操作的执行装置进行了详细介绍, 本文中应用了具体个例对本申请的原理及实施方式进行了阐述, 以上实施例的说明只是用于帮助理解本申请的方法及其核心思想; 同时, 对于本领域的一般技术人员, 依据本申请的思想, 在具体实施方式及应用范围上均会有改变之处, 综上所述, 本说明书内容不应理解为对本申请的限制。

权 利 要 求 书

1、一种文件操作的执行方法，其特征在于，包括：

获取文件操作请求，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

依据所述文件路径在对象管理器中查找对应的文件对象解析例程；

若查找到对应的文件对象解析例程，则依据所述文件对象解析例程生成 I/O 请求包，并发送所述 I/O 请求包至预置的文件系统下层设备的原始地址。

2、如权利要求 1 所述的方法，其特征在于，所述依据文件路径在对象管理器中查找对应的文件对象解析例程的步骤具体包括以下子步骤：

子步骤 S1、判断文件路径是否已经拆解完毕，若否，则执行子步骤 S2；若是，则执行子步骤 S4；

子步骤 S2、按照路径分隔符拆解出文件路径中下一个待拆解的路径段；

子步骤 S3、采用当前拆解出的路径段在对象管理器中搜索，判断是否存在对应的文件对象例程；若是，则返回子步骤 S1；若否，则执行子步骤 S5；

子步骤 S4、获得所述文件路径对应的文件对象解析例程。

子步骤 S5、返回未找到对应文件对象解析例程的信息。

3、如权利要求 1 或 2 所述的方法，其特征在于，所述调用者输入参数具有用户态地址；在查找文件对象解析例程之前，所述的方法还包括：

重构所述用户态地址至内核态内存空间。

4、如权利要求 1 或 2 所述的方法，其特征在于，所述 I/O 请求包中包括从文件操作请求中提取的文件操作信息，在发送 I/O 请求包至预置的文件系统下层设备的原始地址之后，还包括：

由所述文件系统下层设备依据所述文件操作信息继续执行对应的文件操作。

5、如权利要求 3 所述的方法，其特征在于，在获取文件操作请求之

前，还包括：

调用者发起文件操作请求，调用对应的文件操作接口例程；其中，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

依据系统平台的类型构建内核态结构参数，依据所述内核态结构参数生成相应的文件操作控制码，并发送所述文件操作控制码至操作系统内核态驱动。

6、如权利要求 5 所述的方法，其特征在于，在构建内核态结构参数之前，还包括：

文件操作接口例程将调用者输入参数中的 ANSI 相关参数转换为 UNICODE 类型，并调用对应的文件操作接口宽字符例程。

7、如权利要求 6 所述的方法，其特征在于，所述文件操作接口例程包括文件创建例程 FSCreateFile，在由所述文件系统下层设备继续执行所请求的文件操作之前，还包括：

插入新创建的文件对象至对象管理器，并得到返回的文件句柄。

8、如权利要求 1 所述的方法，其特征在于，还包括：

依据所述文件操作请求校验调用者输入参数，若校验通过，则执行查找文件对象解析例程的步骤。

9、如权利要求 7 所述的方法，其特征在于，所述文件操作接口例程与 WINDOWS 标准 API 一致，所述文件操作接口例程还包括：文件读取例程 FSReadFile、文件写入例程 FSWriteFile、文件属性设置例程 FSSetFileAttributes、文件属性获取例程 FSGetFileAttributes、文件指针设置例程 FSSetFilePointer、增强文件指针设置例程 FSSetFilePointerEx、文件大小获取例程 FSGetFileSize、文件删除例程 FSDeleteFile、目录移除例程 FSRemoveDirectory、句柄关闭例程 FSCloseHandle、第一个文件查找例程 FSFindFirstFile、下一个文件查找例程 FSFindNextFile、文件查找关闭例程 FSFindClose、增加文件属性获取例程 FSGetFileAttributesEx、判断路径是否为目录的例程 FSPathIsDirectory、判断目标文件是否存在的例程 FSPathFileExists、长路径获取例程 FSGetLongPathName、短路径获取

例程 FSGetShortPathName、路径查找例程 FSSearchPath、增强文件大小获取例程 FSGetFileSizeEx、文件复制例程 FSCopyFile、文件移动例程 FSMoveFile 和/或增强文件移动例程 FSMoveFileEx。

10、如权利要求 9 所述的方法，其特征在于，所述控制码包括：文件创建操作控制码 FILE_IO_CREATE_FILE、文件读取操作控制码 FILE_IO_READ_FILE、文件写入操作控制码 FILE_IO_WRITE_FILE、文件查询操作控制码 FILE_IO_QUERY_FILE、文件设置操作控制码 FILE_IO_SET_FILE 和 / 或 文件 关 闭 准 备 操 作 控 制 码 FILE_IO_PREPARE_CLOSE。

11、一种文件操作的执行装置，其特征在于，包括：

内核态请求获取模块，设置为获取文件操作请求，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

内核态对象解析模块，设置为依据所述文件路径在对象管理器中查找对应的文件对象解析例程；若查找到对应的文件对象解析例程，则调用内核 IRP 生成发送模块；

内核态 IRP 生成发送模块，设置为依据所述文件对象解析例程生成 I/O 请求包，并发送所述 I/O 请求包至预置的文件系统下层设备的原始地址。

12、如权利要求 11 所述的装置，其特征在于，所述内核对象解析模块具体包括：

文件路径拆解模块，设置为按照路径分隔符逐级拆解出文件路径中的路径段；

对象管理器搜索模块，设置为采用当前拆解出的路径段在对象管理器中搜索，查找对应的文件对象例程。

13、如权利要求 11 或 12 所述的装置，其特征在于，所述调用者输入参数具有用户态地址；所述的装置还包括：

内核地址重构模块，设置为重构所述用户态地址至内核态内存空间。

14、如权利要求 13 所述的装置，其特征在于，还包括：

用户态请求发送模块，设置为由调用者发起文件操作请求，调用对应的文件操作接口例程；其中，所述请求中包括调用者输入参数，所述输入参数中包括文件路径；

用户态控制码发送模块，设置为依据系统平台的类型构建内核态结构参数，依据所述内核态结构参数生成相应的文件操作控制码，并发送所述文件操作控制码至操作系统内核态驱动。

15、如权利要求 14 所述的装置，其特征在于，还包括：

宽字符例程调用模块，设置为将调用者输入参数中的 ANSI 相关参数转换为 UNICODE 类型，并调用对应的文件操作接口宽字符例程。

16、如权利要求 15 所述的装置，其特征在于，所述文件操作接口例程包括文件创建例程 FSCreateFile，所述的装置还包括：

句柄获取模块，设置为插入新创建的文件对象至对象管理器，并得到返回的文件句柄。

17、如权利要求 11 所述的装置，其特征在于，还包括：

内核态参数校验模块，设置为依据所述文件操作请求校验调用者输入参数，若校验通过，则调用内核对象解析模块。

18、一种在其上记录有用于执行权利要求 1 所述方法的程序的计算机可读记录介质。

说明书附图

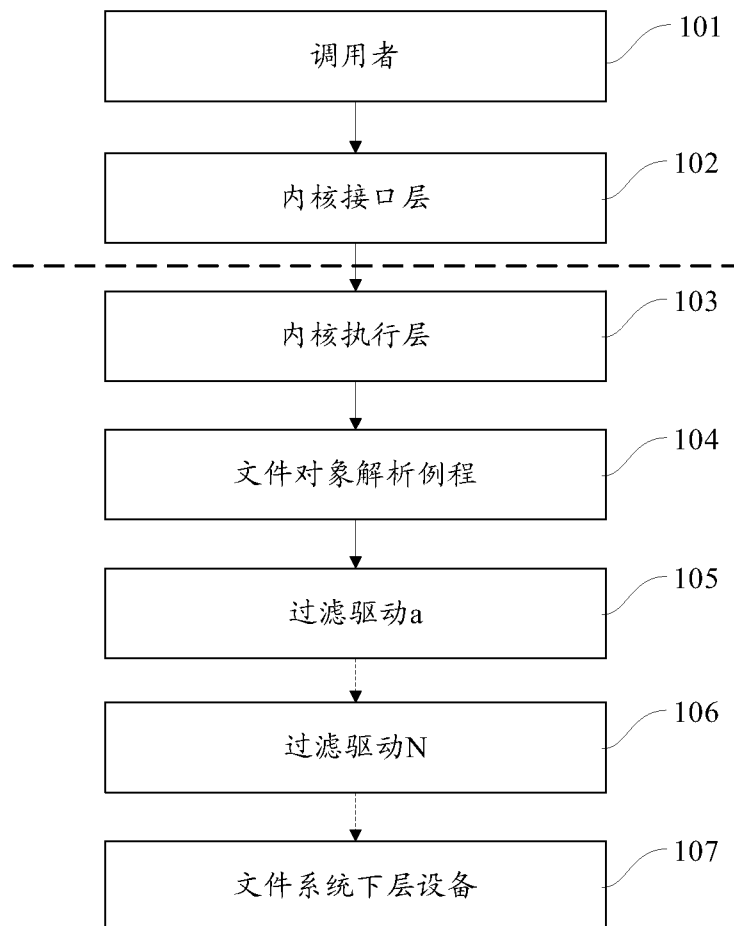


图 1

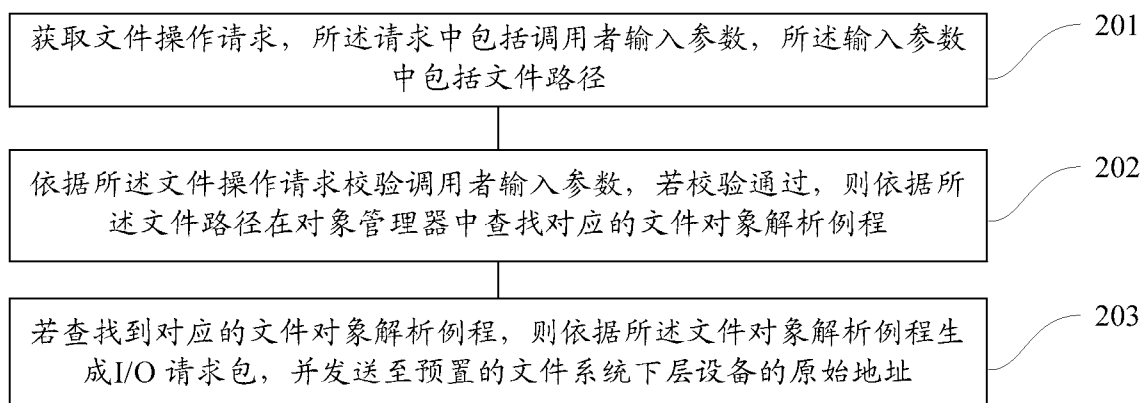


图 2

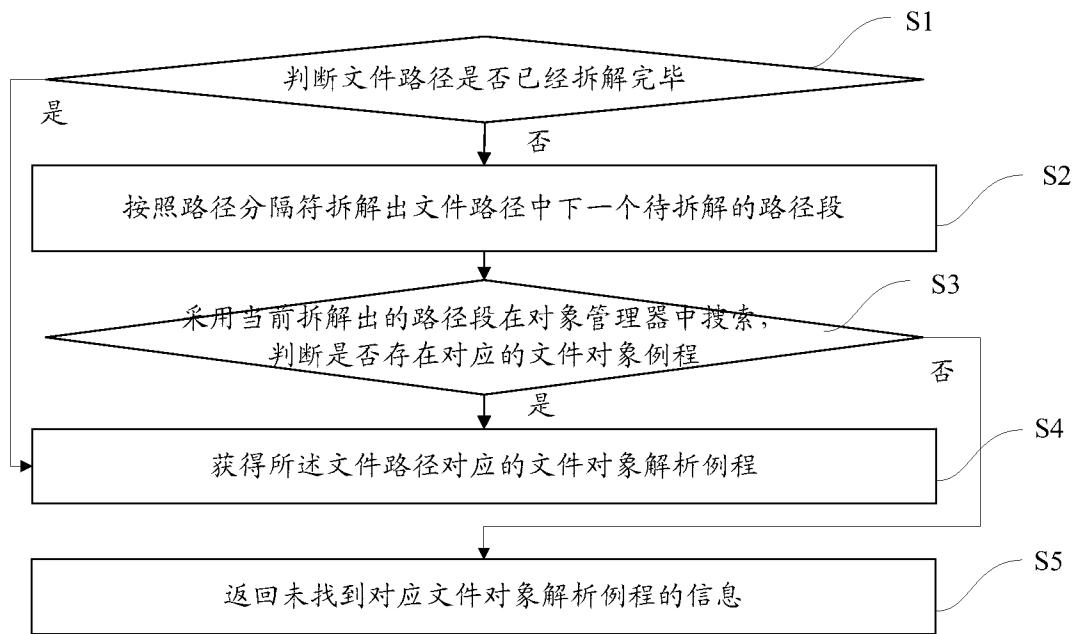


图 3

Figure 4 shows a screenshot of the Windows Object Manager. The left pane displays a tree view of object types, with **GLOBAL??** selected. The right pane displays a list of objects with columns: Name, Type, and Additional Information. The objects are listed in alphabetical order.

Name	Type	Additional Information
ArcName	Directory	
Ntfs	Device	
SeLsaCommandPort	Port	
FatCdrom	Device	
REGISTRY	Key	
ThemeApiPort	Port	
XactSrvLpcPort	Port	
NLS	Directory	
DosDevices	SymbolicLink	\\??
SeRmCommandPort	Port	
Dfs	Device	
LsaAuthenticationPort	Port	
LanmanServerAnnounceEvent	Event	
Driver	Directory	
FGPort	FilterCon...	
Device	Directory	
Windows	Directory	
Sessions	Directory	
SAM_SERVICE_STARTED	Event	
RPC Control	Directory	
Fat	Device	
SmApiPort	Port	
KernelObjects	Directory	
BaseNamedObjects	Directory	
Global??	Directory	
FileSystem	Directory	
NLAPublicPort	WaitablePort	
ObjectTypes	Directory	
SmSsWinStationApiPort	Port	
Security	Directory	
NETLOGON_SERVICE_STARTED	Event	
ErrorLogPort	Port	
SystemRoot	SymbolicLink	\\Device\\Harddisk0\\Partition1\\WINDOWS
Callback	Directory	

图 4

	Name	Type	Additional Information
ArcName	ACPI\FixedButton#2&daba3ff&1#{4afa3d53-...	SymbolicLink	\Device\00000059
NLS	ACPI\GenuineIntel_-_x86_Family_6_Model_...	SymbolicLink	\Device\00000054
Driver	ACPI\GenuineIntel_-_x86_Family_6_Model_...	SymbolicLink	\Device\00000055
Device (5)	ACPI\PNP0303#4&2c575acb&0#{884b96c3-56e...	SymbolicLink	\Device\00000071
Windows	ACPI\PNP0501#1#{4d36e978-e325-11ce-bfcl...	SymbolicLink	\Device\0000006f
Sessions	ACPI\PNP8501#1#{86e0d1e0-8089-11d0-9ce4...	SymbolicLink	\Device\0000006f
RPC Control	ACPI\PNP0501#2#{4d36e978-e325-11ce-bfcl...	SymbolicLink	\Device\00000070
BaseNamedObjects	ACPI\PNP0501#2#{86e0d1e0-8089-11d0-9ce4...	SymbolicLink	\Device\00000070
KernelObjects	ACPI\PNP0C0C#aa#{4afa3d53-74a7-11d0-b2e5...	SymbolicLink	\Device\00000058
FileSystem	ACPI\PNP0F03#4&2c575acb&0#{378de44c-56e...	SymbolicLink	\Device\00000072
GLOBAL?? (4)	AUX	SymbolicLink	\DosDevices\COM1
ObjectTypes	COM1	SymbolicLink	\Device\Serial0
Security	COM2	SymbolicLink	\Device\Serial1
Callback	CVPNDRVA	SymbolicLink	\Device\CVPNDRVA
KnownDlls	D:	SymbolicLink	\Device\HarddiskVolume2

图 5

	Name	Type	Additional Information
ArcName	DNE	Device	
NLS	DSFKSvcs	Device	
Driver	Fips	Device	
Device (6)	FSVideo	Device	
Windows	FsWrap	Device	
Sessions	FtControl	Device	
RPC Control	Gpc	Device	
BaseNamedObjects	Harddisk0	Directory	
KernelObjects	Harddisk1	Directory	
FileSystem	Harddisk0mVolumes	Directory	
GLOBAL??	HarddiskVolume1	Device	通用卷
ObjectTypes	HarddiskVolume2	Device	通用卷
Security	HarddiskVolume3	Device	通用卷
Callback	HarddiskVolume4	Device	通用卷
KnownDlls	hcomn	Device	
	Http	Directory	

图 6

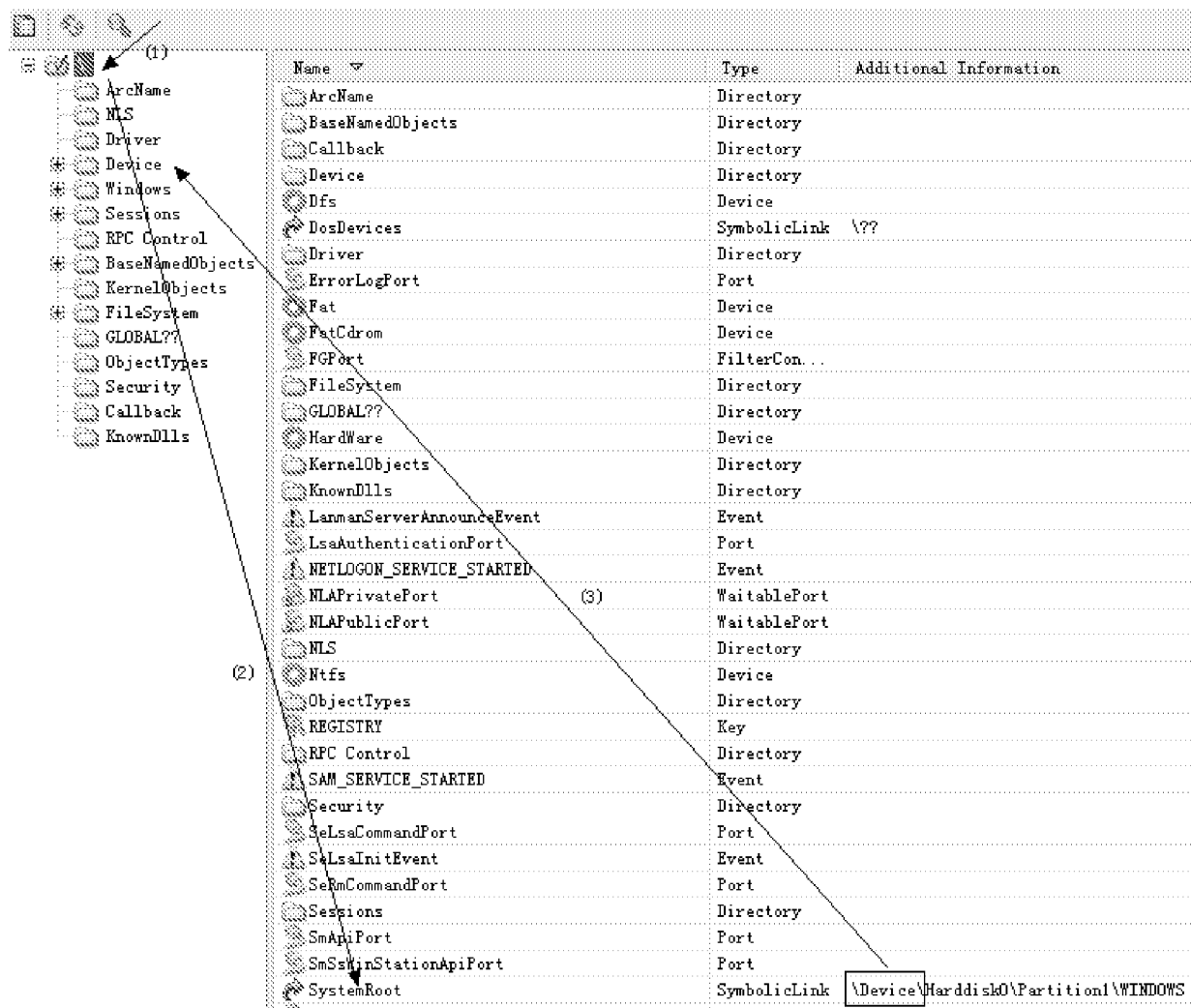


图 7

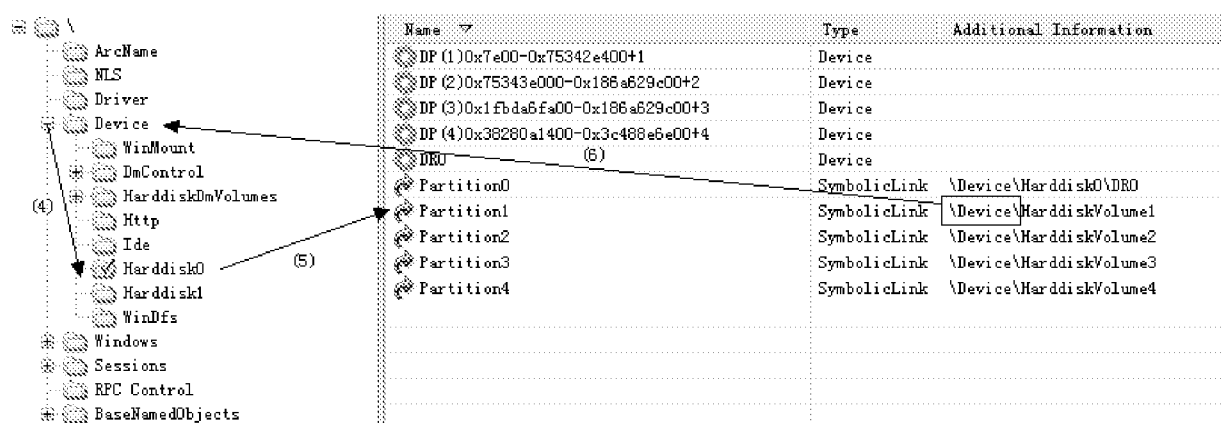


图 8

	Name	Type	Additional Information
ArcName	DNE	Device	
NLS	DSFKSvcs	Device	
Driver	Fips	Device	
Device	FSVideo	Device	
Windows	FsWrap	Device	
Sessions	FtControl	Device	
RPC Control	Gpc	Device	
BaseNamedObjects	Harddisk0	Directory	
KernelObjects	Harddisk1	Directory	
FileSystem	HarddiskDmVolumes	Directory	
GLOBAL??	HarddiskVolume1	Device	通用卷
ObjectTypes	HarddiskVolume2	Device	通用卷
Security	HarddiskVolume3	Device	通用卷
Callback	HarddiskVolume4	Device	通用卷
KnownDlls	hcomon	Device	
	Http	Directory	

图 9



图 10

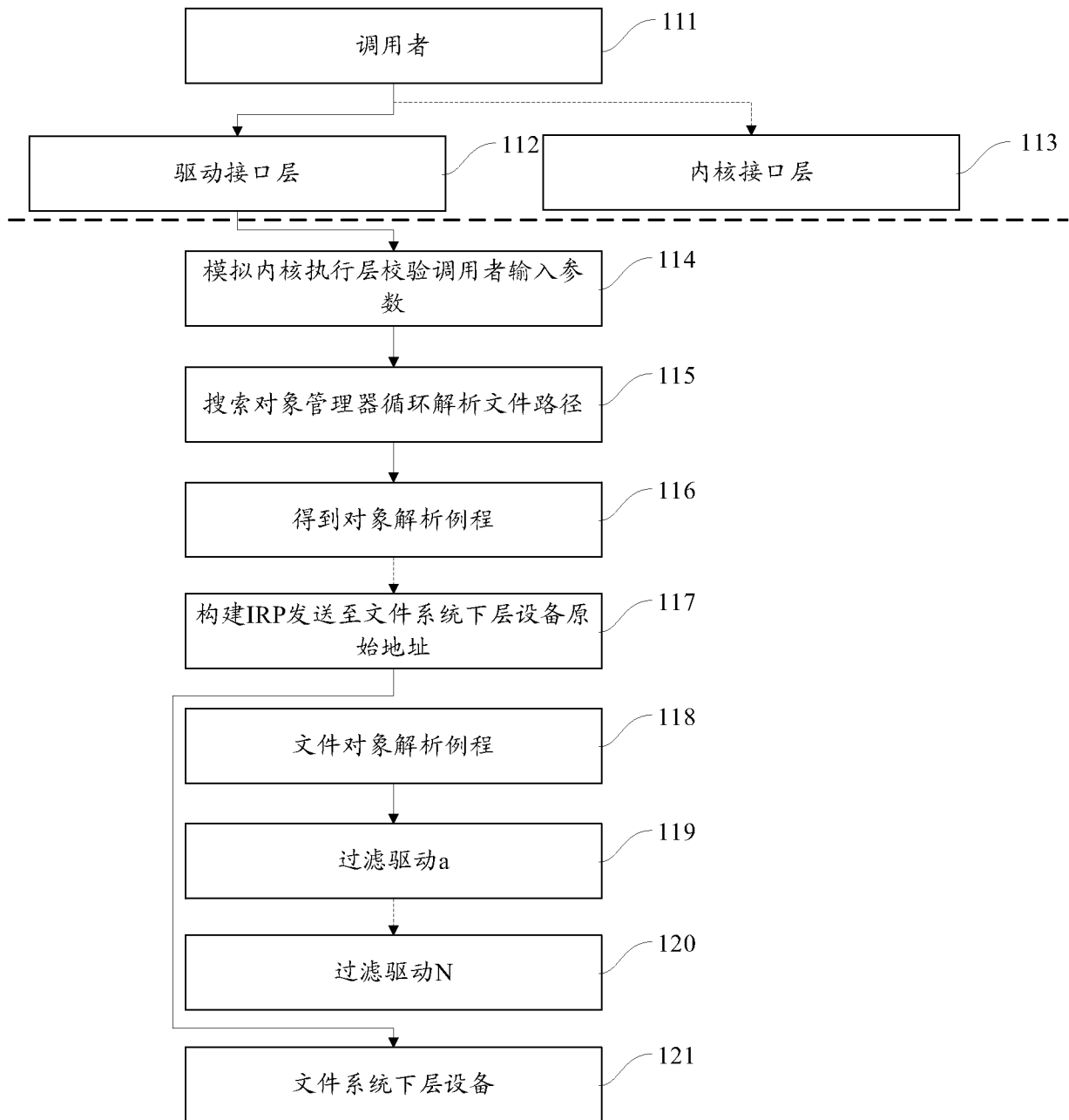


图 11

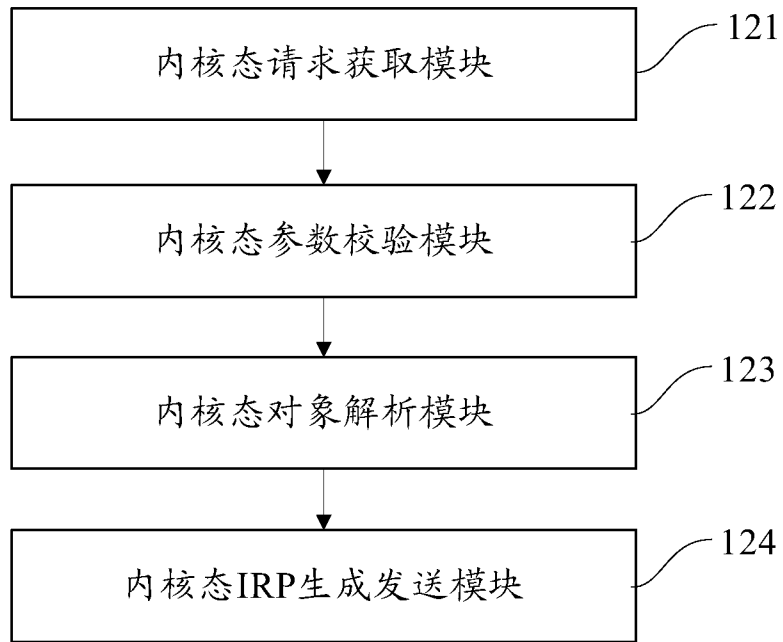


图 12

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2012/075145

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/46 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: G06F; H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS; CNTXT; CNPAT; WOTXT; VEN; CJFD: lower layer, bottom layer, kernel mode, control code, IRP, file, file folder, operation, execute, processing, obtain, receiving, request, parameter, path, object, analysis, analyze, routine, thread, search, retrieval, input/output, I/O, preset, bottom, sub-layer, address, reconstitution, redirection, write, control, code

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 101459697 A (TSINGHUA UNIVERSITY), 17 June 2009 (17.06.2009), see description, page 6, line 17 to page 7, line 13	1-18
A	CN 101291346 A (INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES), 22 October 2008 (22.10.2008), see the whole document	1-18
A	US 2004/0010565 A1 (ICUBE et al.), 15 January 2004 (15.01.2004), see the whole document	1-18
A	CN 101464900 A (SHANGHAI JIAO TONG UNIVERSITY), 24 June 2009 (24.06.2009), see the whole document	1-18

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

10 July 2012 (10.07.2012)

Date of mailing of the international search report

26 July 2012 (26.07.2012)

Name and mailing address of the ISA/CN:

State Intellectual Property Office of the P. R. China
No. 6, Xitucheng Road, Jimenqiao
Haidian District, Beijing 100088, China
Facsimile No.: (86-10) 62019451

Authorized officer

ZHAO, Weihua

Telephone No.: (86-10) 62411830

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2012/075145

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 101459697 A	17.06.2009	CN 101459697 B	04.01.2012
CN 101291346 A	22.10.2008	CN 101291346 B	15.02.2012
US 2004/0010565 A1	15.01.2004	US 7519686 B2	14.04.2009
		KR 20030092678 A	06.12.2003
		USRE 43210 E1	21.02.2012
CN 101464900 A	24.06.2009	CN 101464900 B	21.07.2010

国际检索报告

国际申请号

PCT/CN2012/075145

A. 主题的分类

G06F 9/46 (2006.01) i

按照国际专利分类(IPC)或者同时按照国家分类和 IPC 两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

IPC: G06F; H04L

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))

CNABS; CNTXT; CNPAT; WOTXT; VEN; CJFD: 文件, 文件夹, 操作, 执行, 处理, 获取, 接收, 请求, 参数, 路径, 对象, 解析, 分析, 例程, 线程, 查找, 检索, 输入/输出, 预置, 下层, 底层, 地址, 重构, 重定向, 内核态, 写入, 控制码, IRP, file, file folder, operation, execute, processing, obtain, receiving, request, parameter, path, object, analysis, analyze, routine, thread, search, retrieval, input/output, I/O, preset, bottom, sub-layer, address, reconstitution, redirection, write, control, code

C. 相关文件

类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
A	CN101459697A (清华大学) 17.6 月 2009 (17.06.2009) 参见说明书第 6 页第 17 行至第 7 页第 13 行	1-18
A	CN101291346A (中国科学院计算技术研究所) 22.10 月 2008 (22.10.2008) 参见全文	1-18
A	US2004/0010565A1 (ICUBE 等) 15.1 月 2004 (15.01.2004) 参见全文	1-18
A	CN101464900A (上海交通大学) 24.6 月 2009 (24.06.2009) 参见全文	1-18

☐ 其余文件在 C 栏的续页中列出。☒ 见同族专利附件。

* 引用文件的具体类型:

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的)

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期

10.7 月 2012 (10.07.2012)

国际检索报告邮寄日期

26.7 月 2012 (26.07.2012)

ISA/CN 的名称和邮寄地址:

中华人民共和国国家知识产权局

中国北京市海淀区蓟门桥西土城路 6 号 100088

传真号: (86-10)62019451

受权官员

赵伟华

电话号码: (86-10) 62411830

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2012/075145

检索报告中引用的 专利文件	公布日期	同族专利	公布日期
CN101459697A	17.06.2009	CN101459697B	04.01.2012
CN101291346A	22.10.2008	CN101291346B	15.02.2012
US2004/0010565A1	15.01.2004	US7519686B2	14.04.2009
		KR20030092678A	06.12.2003
		USRE43210E1	21.02.2012
CN101464900A	24.06.2009	CN101464900B	21.07.2010