

# (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  
2019年3月14日 (14.03.2019)



(10) 国际公布号  
**WO 2019/047510 A1**

- (51) 国际专利分类号:  
**G06F 9/44** (2018.01)
- (21) 国际申请号: PCT/CN2018/081303
- (22) 国际申请日: 2018年3月30日 (30.03.2018)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:  
201710805670.4 2017年9月8日 (08.09.2017) CN
- (71) 申请人: 武汉斗鱼网络科技有限公司  
(**WUHAN DOUYU NETWORK TECHNOLOGY CO., LTD.**) [CN/CN]; 中国湖北省武汉市东湖开发区软件园东路1号软件产业4.1期B1栋11楼, Hubei 430000 (CN)。
- (72) 发明人: 周志刚(**ZHOU, Zhigang**); 中国湖北省武汉市东湖开发区软件园东路1号软件产业4.1期B1栋11楼, Hubei 430000 (CN)。 陈少杰(**CHEN, Shaojie**); 中国湖北省武汉市东湖开发区软件园东路1号软件产业4.1期B1栋11楼, Hubei 430000 (CN)。 张文明(**ZHANG, Wenming**); 中国湖北省武汉市东湖开发区软件园东路1号软件产业4.1期B1栋11楼, Hubei 430000 (CN)。
- (74) 代理人: 武汉智权专利代理事务所(特殊普通合伙) (**WUHAN ZHI QUAN PATENT AGENCY**); 中国湖北省武汉市东湖新技术开发区珞瑜路727号星光无限4栋21层2103室, Hubei 430000 (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, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB,

(54) **Title:** IOS PLATFORM DYLIB FILE HIDING METHOD, STORAGE MEDIUM, ELECTRONIC APPARATUS AND SYSTEM

(54) 发明名称: IOS平台隐藏dylib文件的方法、存储介质、电子设备及系统

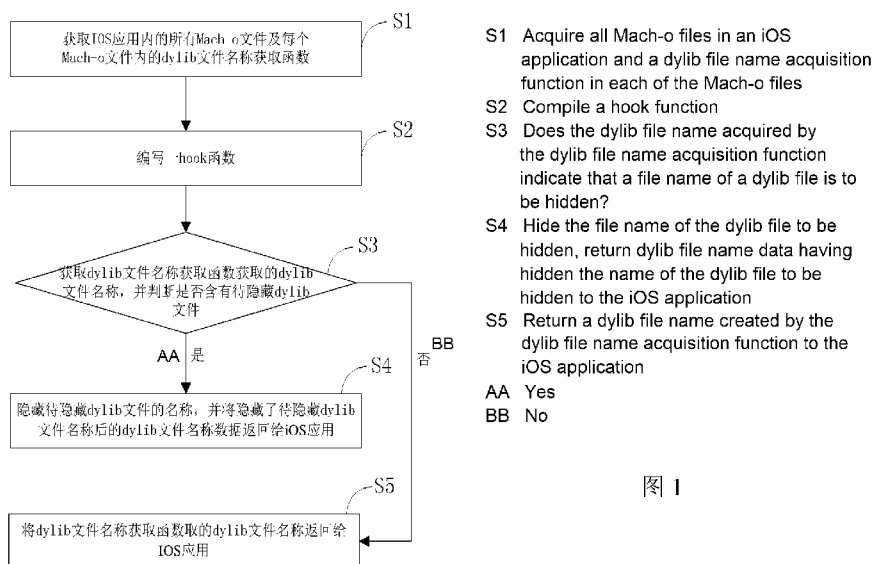


图 1

(57) **Abstract:** The present invention relates to the field of information processing, and discloses an iOS platform dylib file hiding method. The method comprises: S1: acquiring all Mach-o files in an iOS application and a dylib file name acquisition function in each of the Mach-o files; S2: compiling a hook function, wherein the hook function is used to acquire a name of a dylib file acquired by the dylib file name acquisition function; S3: upon the iOS application calling the dylib file name acquisition function to acquire all dylib file names in the iOS application, acquiring, by means of the hook function, a dylib file name acquired by the dylib file name acquisition function, and determining whether there is a dylib file to be hidden according to the file name, If so, proceeding to S4, if not, proceeding to S5; S4: hiding the name of the dylib file to be hidden, returning dylib file name data having hidden the name of the

WO 2019/047510 A1

GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, 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, ST, 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, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

---

dylib file to be hidden to the iOS application; and S5: returning a dylib file name created by the dylib file name acquisition function to the iOS application. The present invention effectively guarantees normal operation of late period debugging of an application.

**(57) 摘要:** 本发明涉及信息处理领域, 公开了一种IOS平台隐藏dylib文件的方法, 该方法包括S1: 获取IOS应用内的所有Mach-o文件及每个Mach-o文件内的dylib文件名称获取函数; S2: 编写一hook函数, 所述hook函数用于获取dylib文件名称获取函数获取的dylib文件的名称; S3: 当IOS应用调用dylib文件名称获取函数进行IOS应用内所有dylib文件名称的获取时, 通过hook函数获取dylib文件名称获取函数获取的dylib文件名称, 根据名称判断是否含有待隐藏dylib文件, 若是, 转到S4, 若否, 转到S5; S4: 隐藏待隐藏dylib文件的名称, 并将隐藏了待隐藏dylib文件名称后的dylib文件名称数据返回给IOS应用; S5: 将dylib文件名称获取函数取的dylib文件名称返回给IOS应用。本发明能够有效保证应用后期调试工作的正常进行。

## IOS 平台隐藏 dylib 文件的方法、存储介质、电子设备及系统

### 技术领域

本发明涉及信息处理领域，具体涉及一种 IOS 平台隐藏 dylib 文件的方法、存储介质、电子设备及系统。

### 背景技术

当前，随着移动设备的日益普及，移动应用产业尤其是 IOS 应用随之得到飞速发展，且 IOS 应用的开发模式和代码框架也随之发生了巨大变化。对于 IOS 平台，其应用程序的构成分为主程序和模块文件，模块文件即为 dylib 文件，dylib 文件是 IOS 平台上的动态链接库文件。在 IOS 应用开发的过程中，若需要编写某些功能模块注入到对应的进程中运行，则均是编写一个 dylib 文件，然后将该 dylib 文件注入到对应的进程中运行。

在 IOS 应用开发完成并进行运行后，通常会编写一个 dylib 文件注入到运行的 IOS 应用中，进行 IOS 应用的运行状态等相关数据的获取，以便于后期开发人员对 IOS 应用的调试和维护，由于 IOS 系统资源机制的原因，IOS 应用的进程经常会关闭一些 dylib 文件的运行，以节省系统资源，保证系统资源的合理利用，故注入到 IOS 应用中用于获取 IOS 应用运行状态的 dylib 文件会被误禁止，进行影响开发人员对 IOS 应用后期调试工作的正常进行。

### 发明内容

针对现有技术中存在的缺陷，本发明的目的在于提供一种 IOS 平台隐藏 dylib 文件的方法，能够有效保证应用后期调试工作的正常

进行。

为达到以上目的，本发明采取的技术方案是，包括：

S1：获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件内的 dylib 文件名称获取函数；

S2：编写一 hook 函数，所述 hook 函数用于获取 dylib 文件名称获取函数获取的 dylib 文件的名称；

S3：当 IOS 应用调用 dylib 文件名称获取函数进行 IOS 应用内所有 dylib 文件名称的获取时，通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称，根据 dylib 文件名称判断是否含有待隐藏 dylib 文件，若是，转到 S4，若否，转到 S5；

S4：隐藏待隐藏 dylib 文件名称，并将隐藏了待隐藏 dylib 文件名称后的 dylib 文件名称数据返回给 IOS 应用；

S5：将 dylib 文件名称获取函数取的 dylib 文件名称返回给 IOS 应用。

在上述技术方案的基础上，对于 Mach-o 文件内 dylib 文件名称获取函数的获取，具体为：

S101：获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件所对应的内存起始地址，并根据内存起始地址获取 Mach-o 文件在内存中的虚拟内存地址；

S102：根据 Mach-o 文件在内存中的虚拟内存地址，解析 Mach-o 文件，得到每个 Mach-o 文件内每个函数的名称和指针；

S103：遍历得到的每个 Mach-o 文件内每个函数的名称，找到每个 Mach-o 文件内的 dylib 文件名称获取函数。

在上述技术方案的基础上，

所述 hook 函数用于将目标函数的内存地址进行替换成设定地址；

当目标函数执行时，跳转至设定地址以进行 dylib 文件名称获取函数获取的 dylib 文件名称的获取；

所述目标函数为 dylib 文件名称获取函数。

在上述技术方案的基础上，

所述 IOS 应用还用于调用 Mach-o 文件内的 dylib 文件个数获取函数获取 IOS 应用内的 dylib 文件个数；

当 dylib 文件名称获取函数获取的 dylib 文件名称中含有待隐藏 dylib 文件名称，则返回给 IOS 应用的 dylib 文件个数为获取的 dylib 文件总个数减去待隐藏 dylib 文件的个数；

当 dylib 文件名称获取函数获取的 dylib 文件名称中不含有待隐藏 dylib 文件名称，则返回给 IOS 应用的 dylib 文件个数即为获取的 dylib 文件总个数。

在上述技术方案的基础上，通过 hook 方式对返回给 IOS 应用的 dylib 文件的个数进行修改。

本发明还提供一种存储介质，该存储介质上存储有计算机程序，所述计算机程序被处理器执行时实现上述所述的方法。

本发明还提供一种电子设备，包括存储器和处理器，存储器上储存有在处理器上运行的计算机程序，所述处理器执行所述计算机程序时实现上述所述的方法。

本发明还提供一种 IOS 平台隐藏 dylib 文件的系统，包括：

获取模块，其用于获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件内的 dylib 文件名称获取函数；

编写模块，其用于编写一 hook 函数，所述 hook 函数用于获取 dylib 文件名称获取函数获取的 dylib 文件的名称；

判断与执行模块，其用于当 IOS 应用调用 dylib 文件名称获取函

数进行 IOS 应用内 dylib 文件名称的获取时,通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称, 根据名称判断是否含有待隐藏 dylib 文件, 若是, 则隐藏待隐藏 dylib 文件的名称, 并将隐藏了待隐藏 dylib 文件名称后的 dylib 文件名称数据返回给 IOS 应用, 若否, 则将 dylib 文件名称获取函数取的 dylib 文件名称返回给 IOS 应用。

在上述技术方案的基础上, 所述获取模块对于 Mach-o 文件内 dylib 文件名称获取函数的获取, 具体为:

获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件所对应的内存起始地址, 并根据内存起始地址获取 Mach-o 文件在内存中的虚拟内存地址;

根据 Mach-o 文件在内存中的虚拟内存地址, 解析 Mach-o 文件, 得到每个 Mach-o 文件内每个函数的名称和指针;

遍历得到的每个 Mach-o 文件内每个函数的名称, 找到每个 Mach-o 文件内的 dylib 文件名称获取函数。

在上述技术方案的基础上, 所述 hook 函数用于将目标函数的内存地址进行替换成设定地址; 当目标函数执行时, 跳转至设定地址以进行 dylib 文件名称获取函数获取的 dylib 文件名称的获取; 所述目标函数为 dylib 文件名称获取函数。

与现有技术相比, 本发明的优点在于: 通过 hook 的方式, 当 IOS 应用调用 dylib 文件名称获取函数进行 dylib 文件名称的获取时, 通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称, 隐藏待隐藏的 dylib 文件, 即相当于隐藏了应用内的用于获取 IOS 应用运行状态的 dylib 文件, 避免 IOS 应用对该类 dylib 文件的禁止运行操作, 保证开发人员能够正常获取到 IOS 应用运行状态的数据, 保证

IOS 应用调试工作的正常进行。

## 附图说明

图 1 为本发明实施例中 IOS 平台隐藏 dylib 文件的方法的流程图；  
图 2 为本发明实施例一种电子设备的结构示意图。

## 具体实施方式

以下结合附图及实施例对本发明作进一步详细说明。

参见图 1 所示，本发明实施例提供一种 IOS 平台隐藏 dylib 文件的方法，用于当 IOS 应用对应用内的 dylib 文件获取时，隐藏应用内的用于获取 IOS 应用运行状态的 dylib 文件，避免 IOS 应用对该类 dylib 文件获取而禁止运行，保证开发人员对 IOS 应用调试工作的正常进行。

本发明实施例中，dylib 文件名称获取函数为 IOS 的系统函数 `_dyld_get_image_name`，通过该函数可以获取 IOS 应用中加载的每个 dylib 文件的名称，该函数的原型为：

```
const char* _dyld_get_image_name(uint32_t image_index);
```

该函数中，参数 `uint32_t image_index` 用来标示获取的 dylib 文件的索引，该函数的返回值 `const char*` 用来标示获取到的 dylib 文件的名称。

本发明实施例的 IOS 平台隐藏 dylib 文件的方法具体包括：

S1：获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件内的 dylib 文件名称获取函数。对于 Mach-o 文件内 dylib 文件名称获取函数的获取，具体为：

S101：获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件所对应的内存起始地址，并根据内存起始地址获取 Mach-o 文件在内

存中的虚拟内存地址；

dylib 文件名称获取函数位于 Mach-o 文件中，Mach-o 是 IOS 系统中可执行文件的格式，对于 Mach-o 文件的结构，具体包括 header structure（头部）、load command（加载命令）和 segment（段），一个 Mach-o 文件可以拥有多个段，每个段可以拥有零个或多个区域（section），每一个段都拥有一段虚拟地址映射到进程的地址空间，同时，一个完整的 Mach-o 文件的末端是链接信息，其中包含了动态加载器用来链接可执行文件或者依赖库所需使用的符号表、字符串等等。dylib 文件名称获取函数位于 Mach-o 文件，且一个 IOS 应用在运行时加载多个 Mach-o 文件，且每一个 Mach-o 文件都有可能调用到 dylib 文件名称获取函数进行应用内 dylib 文件名称的获取操作，因此为保证判断的准确性，我们需要获取 IOS 系统中所有的 Mach-o 文件，以便于进行后续的操作。

对于系统中 Mach-o 文件的获取，首先可以通过系统函数获取到系统中 Mach-o 文件的个数以及 Mach-o 文件的内存起始地址，具体实现为：

执行 `uint32_t c = _dyld_image_count();`即通过 `_dyld_image_count` 来获取 IOS 系统中所加载的 Mach-o 文件的个数，然后

执行 `for (uint32_t i = 0; i < c; i++)`，即通过 for 循环来遍历 Mach-o 文件，接着

执行 `onst struct mach_header *header = _dyld_get_image_header(i)`，即通过函数 `_dyld_get_image_header` 来获取 Mach-o 文件的内存起始地址，接着

执行 `intptr_t slide = _dyld_get_image_vmaddr_slide(i)`；即通过函数 `_dyld_get_image_vmaddr_slide` 来获取 Mach-o 文件在内存中的虚拟内



存地址。

S102: 根据 Mach-o 文件在内存中的虚拟内存地址, 解析 Mach-o 文件, 得到每个 Mach-o 文件内每个函数的名称和指针;

S103: 遍历得到的每个 Mach-o 文件内每个函数的名称, 找到每个 Mach-o 文件内的 dylib 文件名称获取函数。

即为了找出 Mach-o 文件中的目标函数, 目标函数即为 dylib 文件名称获取函数。具体的:

对 Mach-o 文件进行解析, 在 load command (加载命令) 内找到 linkedit\_segment、symtab\_cmd、dysymtab\_cmd, 通过 linkedit\_segment 则可以找到 symtab\_cmd 中的 symtab 和 strtab, 其中, strtab 内存储着 Mach-o 文件内每个函数的名称, symtab 内存储着 Mach-o 文件内每个函数的索引, 而 Mach-o 文件内每个函数的函数指针存储在 section 中, 因此需要进一步地从 Mach-o 文件中解析出 section, 其过程为: 在 Mach-o 文件中获取到对应的 cmd, 因为 Mach-o 文件的解析是开源的, 故可以按照开源的代码来解析需要查找的 linkedit\_segment、symtab\_cmd、dysymtab\_cmd 等 cmd 地址, 故遍历 Mach-o 文件的 Load command 来查找, 最终可以查找到 section、symtab、strtab、indirect\_symtab。得到每个 Mach-o 文件内每个函数的名称和指针后, 便可遍历每个函数的名称, 找到每个 Mach-o 文件内的 dylib 文件名称获取函数。

S2: 编写一 hook (钩子) 函数, 所述 hook 函数用于获取 dylib 文件名称获取函数获取的 dylib 文件的名称; 即 hook 函数用于将目标函数的内存地址进行替换成设定地址; 当目标函数执行时, 跳转至设定地址以进行 dylib 文件名称获取函数获取的 dylib 文件名称的获取; 目标函数为 dylib 文件名称获取函数。其中, 设定地址即为人为编写

的函数地址，用于当 dylib 文件名称获取函数执行时跳转，从而先进行预先编写的功能函数的执行，进而获取到获取 dylib 文件名称获取函数获取的 dylib 文件的名称。

本发明实施例中，hook 函数的原型如下：

```
HOOK_Function(char* pFuncName, void* pNew, void  
**pSaveOrg);
```

该函数中，HOOK\_Function 为 hook 函数的名称；

参数 char\* pFuncName 标示需要 hook 的函数名称，这里需要 hook 的函数名称为 dylib 文件名称获取函数；

参数 void\* pNew 标示替换的函数的内存地址；

参数 void \*\*pSaveOrg 标示存储系统原本的函数的内存地址。

S3：当 IOS 应用调用 dylib 文件名称获取函数进行 IOS 应用内所有 dylib 文件名称的获取时，通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称，根据 dylib 文件名称判断是否含有待隐藏 dylib 文件，若是，转到 S4，若否，转到 S5。本发明实施例中待隐藏 dylib 文件即为用于获取 IOS 应用运行状态的 dylib 文件。通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称的具体过程为：

1、定义一个函数指针：`static const char* (*orig_dyld_get_image_name)(uint32_t image_index)`；使其用来存储 dylib 文件名称获取函数 `_dyld_get_image_name` 的内存地址；

2、编写一功能函数 `my_dyld_get_image_name`，当 `_dyld_get_image_name` 执行时，通过 hook 函数将目标函数的内存地址进行替换成设定地址，进而执行编写的功能函数 `my_dyld_get_image_name`，进而获取 dylib 文件名称获取函数获取的 dylib 文件名称。具体的：

调用 hook 函数对 `_dyld_get_image_name` 进行 hook，

具体实现代码为：`HOOK_Function("_dyld_get_image_name", my_dyld_get_image_name, (void *)&orig_dyld_get_image_name)`

此时，上述的 `char* pFuncName` 即标示需要 hook 的函数为 `_dyld_get_image_name`，`void* pNew` 标示替换的函数为 `my_dyld_get_image_name`，`oid **pSaveOrg` 标示将原本的函数 `_dyld_get_image_name` 地址存储到 `orig_dyld_get_image_name` 函数指针处。

在 `my_dyld_get_image_name` 中依次判断当前获取的 dylib 文件名是否是待隐藏的 dylib 文件的名称，相应代码为：

```
const char* my_dyld_get_image_name(uint32_t image_index)
{
    const char* dyName = orig_dyld_get_image_name(image_index);
    if (strstr(dyName, "需要隐藏的 dylib 文件名称"))
    {
        dyName = orig_dyld_get_image_name(image_index + 1);
    }
    return dyName;
}
```

如果是需要隐藏的，则调用系统的 `orig_dyld_get_image_name` 函数得到下一个 dylib 文件名称，并进行判断，通过此方法来对 dylib 文件名称获取函数获取的所有 dylib 文件名称进行判断。

**S4:** 隐藏待隐藏 dylib 文件名称，并将隐藏了待隐藏 dylib 文件名称后的 dylib 文件名称数据返回给 IOS 应用，即返回给 IOS 应用的所有 dylib 文件名称中是不含有待隐藏 dylib 文件的名称；

**S5:** 将 dylib 文件名称获取函数取的 dylib 文件名称返回给 IOS 应用。

通过编写的 hook 函数, 获取 dylib 文件名称获取函数获取的 dylib 文件的名称, 并根据名称判断是否含有待隐藏 dylib 文件, 如果含有, 则隐藏, 即待隐藏 dylib 文件的名称是不会返回给 IOS 应用的。

在一种实施方式中, 为进一步提高待隐藏 dylib 文件隐藏的准确性, 避免被 IOS 应用所获取, IOS 应用还用于调用 Mach-o 文件内的 dylib 文件个数获取函数获取 IOS 应用内的 dylib 文件个数; 当 dylib 文件名称获取函数获取的 dylib 文件名称中含有待隐藏 dylib 文件名称, 则返回给 IOS 应用的 dylib 文件个数为获取的 dylib 文件总个数减去待隐藏 dylib 文件的个数; 当 dylib 文件名称获取函数获取的 dylib 文件名称中不含有待隐藏 dylib 文件名称, 则返回给 IOS 应用的 dylib 文件个数即为获取的 dylib 文件总个数。具体通过 hook 方式对返回给 IOS 应用的 dylib 文件的个数进行修改。

dylib 文件个数获取函数为 `_dyld_image_count`, 该函数为 IOS 的系统函数, 通过该函数可以获取 IOS 应用中加载的 dylib 文件的个数, 该函数的原型为:

`uint32_t _dyld_image_count(void);` 该函数不需要参数, 调用时直接返回 IOS 应用中的 dylib 文件个数, dylib 文件个数获取函数也位于 Mach-o 文件中, 其获取方式与 dylib 文件名称获取函数相同。

对于 dylib 文件个数获取函数获取的 Mach-o 文件个数的修改过程, 具体为:

定义一个函数指针: `static uint32_t (*orig_dyld_image_count)();` 用来存储 `_dyld_image_count` 函数的地址; 同样的编写一功能函数 `my_dyld_image_count()`, 当 `_dyld_image_count` 执行时, 通过 hook 方式将目标函数的内存地址进行替换成设定地址, 进而执行编写的功能函数 `my_dyld_image_count()`, 进而获取 dylib 文件个数获取函数获取

的 dylib 文件个数，当 dylib 文件名称获取函数获取的 dylib 文件名称中含有待隐藏 dylib 文件名称，则在 my\_dyld\_image\_count() 中修改 dylib 文件的个数，该过程相应的代码为：

```
uint32_t my_dyld_image_count()
{
    uint32_t nCount = orig_dyld_image_count();
    --nCount;
    return nCount;
}
```

若待隐藏 dylib 文件的个数为 1，则返回给 IOS 应用的 dylib 文件个数即为获取的 dylib 文件总个数减 1。

对于 hook 的方式，与 dylib 文件名称获取函数被 hook 的方式类似，此时相应 hook 函数为

```
HOOK_Function("_dyld_image_count", my_dyld_image_count,
(void *)&orig_dyld_image_count);
```

此处该 hook 函数中，char\* pFuncName 即标示需要 hook 的函数为 \_dyld\_image\_count，void\* pNew 标示替换的函数为 my\_dyld\_image\_count，oid \*\*pSaveOrg 标示将原本的函数 \_dyld\_image\_count 地址存储到 orig\_dyld\_image\_count 函数指针处。

本发明的 IOS 平台隐藏 dylib 文件的方法的原理在于：针对 IOS 应用中当进行应用内 dylib 文件名称获取时必须调用 dylib 文件名称获取函数的特性，通过 hook 的方式，当 IOS 应用调用 dylib 文件名称获取函数进行 IOS 应用内 dylib 文件名称的获取时，通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称，并根据名称判断是否含有待隐藏的 dylib 文件，若含有，则隐藏待隐藏 dylib 文件的名称，使最终返回给应用的 dylib 文件的名称中不含有待隐藏 dylib 文件

的名称,即相当于隐藏了应用内的用于获取 IOS 应用运行状态的 dylib 文件,避免 IOS 应用对该类 dylib 文件的禁止运行操作,保证开发人员能够正常获取到 IOS 应用运行状态的数据,保证 IOS 应用调试工作的正常进行。

另外,对应上述 IOS 平台隐藏 dylib 文件的方法,本发明还提供一种存储介质,存储介质上存储有计算机程序,计算机程序被处理器执行时实现上述各实施例所述的 IOS 平台隐藏 dylib 文件的方法的步骤。需要说明的是,所述存储介质包括 U 盘、移动硬盘、ROM (Read-Only Memory,只读存储器)、RAM(Random Access Memory,随机存取存储器)、磁碟或者光盘等各种可以存储程序代码的介质。

参见图 2 所示,对应上述 IOS 平台隐藏 dylib 文件的方法,本发明还提供一种电子设备,包括存储器和处理器,存储器上储存有在处理器上运行的计算机程序,处理器执行计算机程序时实现上述各实施例的 IOS 平台隐藏 dylib 文件的方法。

本发明实施例还提供一种基于上述 IOS 平台隐藏 dylib 文件的方法的 IOS 平台隐藏 dylib 文件的系统,包括获取模块、编写模块和判断与执行模块。

获取模块用于获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件内的 dylib 文件名称获取函数;编写模块用于编写一 hook 函数,所述 hook 函数用于获取 dylib 文件名称获取函数获取的 dylib 文件的名称;判断与执行模块用于当 IOS 应用调用 dylib 文件名称获取函数进行 IOS 应用内 dylib 文件名称的获取时,通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称,根据 dylib 文件名称判断是否含有待隐藏 dylib 文件,若是,则隐藏待隐藏 dylib 文件名称,并将隐藏了待隐藏 dylib 文件名称后的 dylib 文件名称数据返回给 IOS 应

用，若否，则将 dylib 文件名称获取函数取的 dylib 文件名称返回给 IOS 应用。

本发明实施例的 IOS 平台隐藏 dylib 文件的系统的原理在于：针对 IOS 应用中当进行应用内 dylib 文件名称获取时必须调用 dylib 文件名称获取函数的特性，通过编写模块，编写一 hook 函数，当 IOS 应用调用 dylib 文件名称获取函数进行 IOS 应用内 dylib 文件名称的获取时，通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称，通过判断与执行模块，根据名称判断是否含有待隐藏的 dylib 文件，若含有，则隐藏待隐藏 dylib 文件的名称，使最终返回给应用的 dylib 文件的名称中不含有待隐藏 dylib 文件的名称，即相当于隐藏了应用内的用于获取 IOS 应用运行状态的 dylib 文件，避免 IOS 应用对该类 dylib 文件的禁止运行操作，保证开发人员能够正常获取到 IOS 应用运行状态的数据，保证 IOS 应用调试工作的正常进行。

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。

## 权 利 要 求 书

1、一种 IOS 平台隐藏 dylib 文件的方法，其特征在于，包括：

S1：获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件内的 dylib 文件名称获取函数；

S2：编写一 hook 函数，所述 hook 函数用于获取 dylib 文件名称获取函数获取的 dylib 文件的名称；

S3：当 IOS 应用调用 dylib 文件名称获取函数进行 IOS 应用内所有 dylib 文件名称的获取时，通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称，根据 dylib 文件名称判断是否含有待隐藏 dylib 文件，若是，转到 S4，若否，转到 S5；

S4：隐藏待隐藏 dylib 文件名称，并将隐藏了待隐藏 dylib 文件名称后的 dylib 文件名称数据返回给 IOS 应用；

S5：将 dylib 文件名称获取函数取的 dylib 文件名称返回给 IOS 应用。

2、如权利要求 1 所述的一种 IOS 平台隐藏 dylib 文件的方法，其特征在于，对于 Mach-o 文件内 dylib 文件名称获取函数的获取，具体为：

S101：获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件所对应的内存起始地址，并根据内存起始地址获取 Mach-o 文件在内存中的虚拟内存地址；

S102：根据 Mach-o 文件在内存中的虚拟内存地址，解析 Mach-o 文件，得到每个 Mach-o 文件内每个函数的名称和指针；

S103：遍历得到的每个 Mach-o 文件内每个函数的名称，找到每个 Mach-o 文件内的 dylib 文件名称获取函数。

3、如权利要求 1 所述的一种 IOS 平台隐藏 dylib 文件的方法，



其特征在于：

所述 hook 函数用于将目标函数的内存地址进行替换成设定地址；

当目标函数执行时，跳转至设定地址以进行 dylib 文件名称获取函数获取的 dylib 文件名称的获取；

所述目标函数为 dylib 文件名称获取函数。

4、如权利要求 1 所述的一种 IOS 平台隐藏 dylib 文件的方法，其特征在于：

所述 IOS 应用还用于调用 Mach-o 文件内的 dylib 文件个数获取函数获取 IOS 应用内的 dylib 文件个数；

当 dylib 文件名称获取函数获取的 dylib 文件名称中含有待隐藏 dylib 文件名称，则返回给 IOS 应用的 dylib 文件个数为获取的 dylib 文件总个数减去待隐藏 dylib 文件的个数；

当 dylib 文件名称获取函数获取的 dylib 文件名称中不含有待隐藏 dylib 文件名称，则返回给 IOS 应用的 dylib 文件个数即为获取的 dylib 文件总个数。

5、如权利要求 4 所述的一种 IOS 平台隐藏 dylib 文件的方法，其特征在于：通过 hook 方式对返回给 IOS 应用的 dylib 文件的个数进行修改。

6、一种存储介质，该存储介质上存储有计算机程序，其特征在于：所述计算机程序被处理器执行时实现权利要求 1 至 5 任一项所述的方法。

7、一种电子设备，包括存储器和处理器，存储器上储存有在处理器上运行的计算机程序，其特征在于：所述处理器执行所述计算机程序时实现权利要求 1 至 5 任一项所述的方法。

8、一种 IOS 平台隐藏 dylib 文件的系统，其特征在于，包括：

获取模块，其用于获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件内的 dylib 文件名称获取函数；

编写模块，其用于编写一 hook 函数，所述 hook 函数用于获取 dylib 文件名称获取函数获取的 dylib 文件的名称；

判断与执行模块，其用于当 IOS 应用调用 dylib 文件名称获取函数进行 IOS 应用内 dylib 文件名称的获取时，通过 hook 函数获取 dylib 文件名称获取函数获取的 dylib 文件名称，根据 dylib 文件名称判断是否含有待隐藏 dylib 文件，若是，则隐藏待隐藏 dylib 文件名称，并将隐藏了待隐藏 dylib 文件名称后的 dylib 文件名称数据返回给 IOS 应用，若否，则将 dylib 文件名称获取函数取的 dylib 文件名称返回给 IOS 应用。

9、如权利要求 8 所述的一种 IOS 平台隐藏 dylib 文件的系统，其特征在于：所述获取模块对于 Mach-o 文件内 dylib 文件名称获取函数的获取，具体为：

获取 IOS 应用内的所有 Mach-o 文件及每个 Mach-o 文件所对应的内存起始地址，并根据内存起始地址获取 Mach-o 文件在内存中的虚拟内存地址；

根据 Mach-o 文件在内存中的虚拟内存地址，解析 Mach-o 文件，得到每个 Mach-o 文件内每个函数的名称和指针；

遍历得到的每个 Mach-o 文件内每个函数的名称，找到每个 Mach-o 文件内的 dylib 文件名称获取函数。

10、如权利要求 9 所述的一种 IOS 平台隐藏 dylib 文件的系统，其特征在于：所述 hook 函数用于将目标函数的内存地址进行替换成设定地址；当目标函数执行时，跳转至设定地址以进行 dylib 文件名称获

取函数获取的 dylib 文件名称的获取;所述目标函数为 dylib 文件名称  
获取函数。

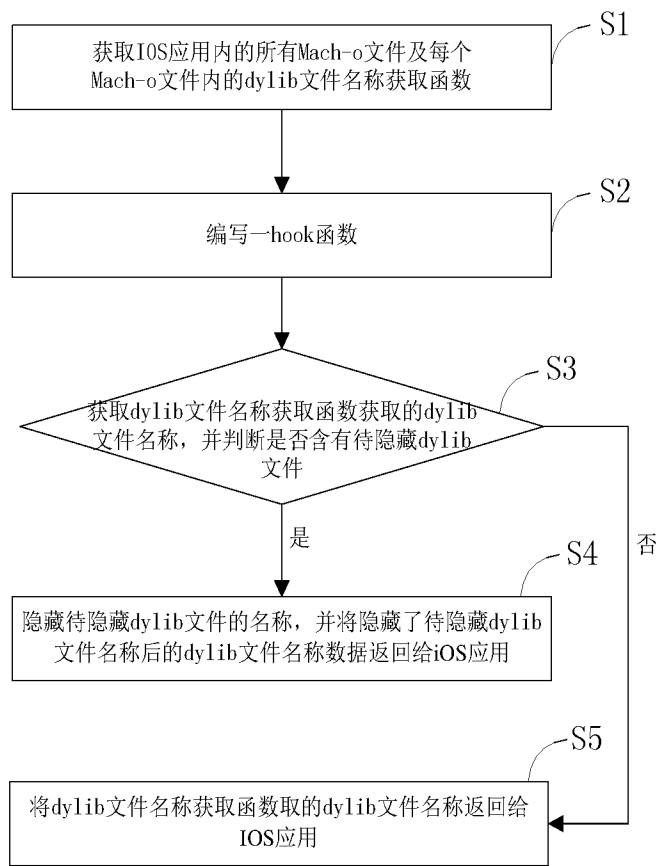


图 1

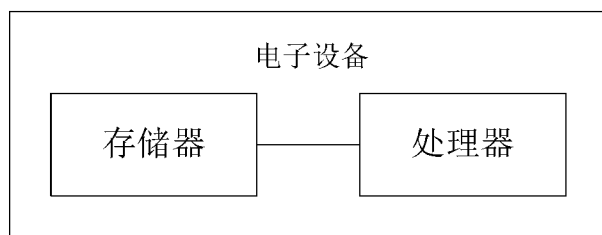


图 2

# INTERNATIONAL SEARCH REPORT

International application No.  
PCT/CN2018/081303

## A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/44 (201&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)

G06F

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, DWPI, SIPOABS, CNKI: 名称, 名字, 隐藏, 钩子, 函数, 过滤, name, hide, hook, function, filter

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 107544811 A (WUHAN DOUYU NETWORK TECHNOLOGY CO., LTD.) 05 January 2018 (05.01.2018), claims 1-10	1-10
X	CN 102194079 A (BEIJING STRONGUNION TECHNOLOGY CO., LTD.) 21 September 2011 (21.09.2011), description, paragraphs [0016]-[0020], and figure 1	1-10
X	CN 102142069 A (BEIJING STRONGUNION TECHNOLOGY CO., LTD.) 03 August 2011 (03.08.2011), description, paragraphs [0028]-[0037]	1-10
A	EP 1684151 A1 (WILLIAM GRANT ROTHWELL) 26 July 2006 (26.07.2006), see entire document	1-10
A	CN 101916349 A (SUN YAT-SEN UNIVERSITY) 15 December 2010 (15.12.2010), see entire document	1-10

Further documents are listed in the continuation of Box C.

See patent family annex.

<p>* Special categories of cited documents:</p> <p>“A” document defining the general state of the art which is not considered to be of particular relevance</p> <p>“E” earlier application or patent but published on or after the international filing date</p> <p>“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)</p> <p>“O” document referring to an oral disclosure, use, exhibition or other means</p> <p>“P” document published prior to the international filing date but later than the priority date claimed</p>	<p>“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</p> <p>“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</p> <p>“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</p> <p>“&amp;” document member of the same patent family</p>
---	---

Date of the actual completion of the international search  
21 June 2018

Date of mailing of the international search report  
28 June 2018

Name and mailing address of the ISA  
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  
WANG, Yan  
Telephone No. (86-10) 62411625

**INTERNATIONAL SEARCH REPORT**  
**Information on patent family members**

International application No.  
PCT/CN2018/081303

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 107544811 A	05 January 2018	None	
CN 102194079 A	21 September 2011	CN 102194079 B	11 September 2013
CN 102142069 A	03 August 2011	CN 102142069 B	12 September 2012
EP 1684151 A1	26 July 2006	US 9129111 B2	08 September 2015
		EP 1842148 A2	10 October 2007
		US 9760715 B2	12 September 2017
		WO 2006077443 A2	27 July 2006
		WO 2006077443 A3	28 September 2006
		US 2010043072 A1	18 February 2010
		US 2015332047 A1	19 November 2015
CN 101916349 A	15 December 2010	None	

国际检索报告

国际申请号

PCT/CN2018/081303

<p><b>A. 主题的分类</b> G06F 9/44 (2018.01) i</p> <p>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</p>																				
<p><b>B. 检索领域</b></p> <p>检索的最低限度文献(标明分类系统和分类号) G06F</p> <p>包含在检索领域中的除最低限度文献以外的检索文献</p> <p>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, DWPI, SIPOABS, CNKI: 名称, 名字, 隐藏, 钩子, 函数, 过滤, name, hide, hook, function, filter</p>																				
<p><b>C. 相关文件</b></p> <table border="1"> <thead> <tr> <th>类型*</th> <th>引用文件, 必要时, 指明相关段落</th> <th>相关的权利要求</th> </tr> </thead> <tbody> <tr> <td>PX</td> <td>CN 107544811 A (武汉斗鱼网络科技有限公司) 2018年 1月 5日 (2018 - 01 - 05) 权利要求1-10</td> <td>1-10</td> </tr> <tr> <td>X</td> <td>CN 102194079 A (北京思创银联科技股份有限公司) 2011年 9月 21日 (2011 - 09 - 21) 说明书第[0016]-[0020]段, 图1</td> <td>1-10</td> </tr> <tr> <td>X</td> <td>CN 102142069 A (北京思创银联科技股份有限公司) 2011年 8月 3日 (2011 - 08 - 03) 说明书第[0028]-[0037]段</td> <td>1-10</td> </tr> <tr> <td>A</td> <td>EP 1684151 A1 (WILLIAM GRANT ROTHWELL) 2006年 7月 26日 (2006 - 07 - 26) 参见全文</td> <td>1-10</td> </tr> <tr> <td>A</td> <td>CN 101916349 A (中山大学) 2010年 12月 15日 (2010 - 12 - 15) 参见全文</td> <td>1-10</td> </tr> </tbody> </table>			类型*	引用文件, 必要时, 指明相关段落	相关的权利要求	PX	CN 107544811 A (武汉斗鱼网络科技有限公司) 2018年 1月 5日 (2018 - 01 - 05) 权利要求1-10	1-10	X	CN 102194079 A (北京思创银联科技股份有限公司) 2011年 9月 21日 (2011 - 09 - 21) 说明书第[0016]-[0020]段, 图1	1-10	X	CN 102142069 A (北京思创银联科技股份有限公司) 2011年 8月 3日 (2011 - 08 - 03) 说明书第[0028]-[0037]段	1-10	A	EP 1684151 A1 (WILLIAM GRANT ROTHWELL) 2006年 7月 26日 (2006 - 07 - 26) 参见全文	1-10	A	CN 101916349 A (中山大学) 2010年 12月 15日 (2010 - 12 - 15) 参见全文	1-10
类型*	引用文件, 必要时, 指明相关段落	相关的权利要求																		
PX	CN 107544811 A (武汉斗鱼网络科技有限公司) 2018年 1月 5日 (2018 - 01 - 05) 权利要求1-10	1-10																		
X	CN 102194079 A (北京思创银联科技股份有限公司) 2011年 9月 21日 (2011 - 09 - 21) 说明书第[0016]-[0020]段, 图1	1-10																		
X	CN 102142069 A (北京思创银联科技股份有限公司) 2011年 8月 3日 (2011 - 08 - 03) 说明书第[0028]-[0037]段	1-10																		
A	EP 1684151 A1 (WILLIAM GRANT ROTHWELL) 2006年 7月 26日 (2006 - 07 - 26) 参见全文	1-10																		
A	CN 101916349 A (中山大学) 2010年 12月 15日 (2010 - 12 - 15) 参见全文	1-10																		
<input type="checkbox"/> 其余文件在C栏的续页中列出。		<input checked="" type="checkbox"/> 见同族专利附件。																		
<p>* 引用文件的具体类型:</p> <p>“A” 认为不特别相关的表示了现有技术一般状态的文件</p> <p>“E” 在国际申请日的当天或之后公布的在先申请或专利</p> <p>“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)</p> <p>“O” 涉及口头公开、使用、展览或其他方式公开的文件</p> <p>“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</p>		<p>“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了解发明之理论或原理的在后文件</p> <p>“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</p> <p>“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</p> <p>“&amp;” 同族专利的文件</p>																		
<p>国际检索实际完成的日期</p> <p>2018年 6月 21日</p>		<p>国际检索报告邮寄日期</p> <p>2018年 6月 28日</p>																		
<p>ISA/CN的名称和邮寄地址</p> <p>中华人民共和国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088</p> <p>传真号 (86-10)62019451</p>		<p>受权官员</p> <p>王艳</p> <p>电话号码 62411625</p>																		

国际检索报告  
关于同族专利的信息

国际申请号

PCT/CN2018/081303

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	107544811	A	2018年 1月 5日	无			
CN	102194079	A	2011年 9月 21日	CN	102194079	B	2013年 9月 11日
CN	102142069	A	2011年 8月 3日	CN	102142069	B	2012年 9月 12日
EP	1684151	A1	2006年 7月 26日	US	9129111	B2	2015年 9月 8日
				EP	1842148	A2	2007年 10月 10日
				US	9760715	B2	2017年 9月 12日
				WO	2006077443	A2	2006年 7月 27日
				WO	2006077443	A3	2006年 9月 28日
				US	2010043072	A1	2010年 2月 18日
				US	2015332047	A1	2015年 11月 19日
CN	101916349	A	2010年 12月 15日	无			