(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 106295402 B
(45)授权公告日 2020.03.31

(21)申请号 201610671287.X

(22)申请日 2016.08.16

(65)同一申请的已公布的文献号
申请公布号 CN 106295402 A

(43)申请公布日 2017.01.04

(73)专利权人 武汉斗鱼网络科技有限公司
地址 430000 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

(72)发明人 周志刚

(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 陈建

(51)Int.Cl.
G06F 21/62(2013.01)
G06F 21/12(2013.01)
G06F 21/60(2013.01)

(56)对比文件
CN 102930005 A,2013.02.13,
CN 101908119 A,2010.12.08,
CN 102982073 A,2013.03.20,
CN 102799815 A,2012.11.28,
CN 101719077 A,2010.06.02,

审查员 王青

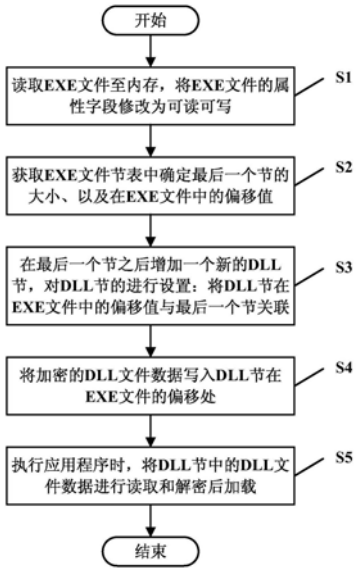
权利要求书2页 说明书4页 附图1页

(54)发明名称

一种DLL文件的隐藏方法及系统

(57)摘要

本发明公开了一种DLL文件的隐藏方法及系统，涉及DLL文件的设置领域。该方法的步骤为：S1：读取EXE文件至内存，将EXE文件的属性字段修改为可读可写；S2：根据EXE文件的内存地址，确定最后一个节的大小、以及在EXE文件中的偏移值；S3：在最后一个节之后增加一个新的DLL节，对DLL节进行设置：将DLL节在EXE文件中的偏移值赋值为：最后一个节在EXE文件中的偏移值，加上最后一个节的大小；S4：将DLL文件数据写入DLL节在EXE文件中的偏移处。本发明能够将DLL文件隐藏在EXE文件之中，DLL文件非常难以被盗用者找到，进而显著的增大了盗用者自行任意使用应用程序难度，最大化保证了应用程序的使用安全。



CN 106295402 B

1.一种DLL文件的隐藏方法,其特征在于,该方法包括以下步骤:

S1:读取EXE文件至内存,将EXE文件的属性字段修改为可读可写;

S2:根据EXE文件的内存地址,读取EXE文件的DOS头部内存地址;根据DOS头部内存地址,读取EXE文件的NT头部内存地址;根据NT头部内存地址,获取EXE文件节表中第一个节的内存地址;根据第一个节的内存地址,获取EXE文件节表中最后一个节的内存地址;根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值;

S3:在EXE文件节表中,在最后一个节之后增加一个新的DLL节,对DLL节进行设置:将DLL节在EXE文件中的偏移值赋值为:最后一个节在EXE文件中的偏移值,加上最后一个节的大小;

S4:将DLL文件数据写入DLL节在EXE文件中的偏移处。

2.如权利要求1所述的DLL文件的隐藏方法,其特征在于,S3中所述对DLL节进行设置还包括以下流程:

将DLL节的内存地址设置为所述最后一个节的内存地址加1;

将DLL节的节名称与EXE文件节表中的已有的节名称进行区分;

将DLL节的属性字段设置为可读可写;

将DLL节的大小赋值为需要存入的DLL文件的大小。

3.如权利要求1所述的DLL文件的隐藏方法,其特征在于,S4的具体流程为:读取所述DLL文件至内存,采用加密算法对DLL文件数据进行加密后,将加密的DLL文件数据写入DLL节在EXE文件中的偏移处。

4.如权利要求1至3任一项所述的DLL文件的隐藏方法,其特征在于,S4之后还包括以下步骤:

S5:执行包含有所述DLL文件的应用程序时,将所述DLL节中的DLL文件数据进行读取,应用程序加载读取的DLL文件数据。

5.一种实现权利要求1所述方法的DLL文件的隐藏系统,其特征在于,该系统包括EXE文件读取模块、EXE文件节表查找模块、DLL节生成模块和DLL文件数据写入模块;

EXE文件读取模块用于:读取EXE文件至内存,将EXE文件的属性字段修改为可读可写;

EXE文件节表查找模块用于:根据EXE文件的内存地址,读取EXE文件的DOS头部内存地址;根据DOS头部内存地址,读取EXE文件的NT头部内存地址;根据NT头部内存地址,获取EXE文件节表中第一个节的内存地址;根据第一个节的内存地址,获取EXE文件节表中最后一个节的内存地址;根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值;

DLL节生成模块用于:在EXE文件节表中,在最后一个节之后增加一个新的DLL节,对DLL节进行设置:将DLL节在EXE文件中的偏移值赋值为:最后一个节在EXE文件中的偏移值,加上最后一个节的大小;

DLL文件数据写入模块用于:将DLL文件数据写入DLL节在EXE文件中的偏移处。

6.如权利要求5所述的DLL文件的隐藏系统,其特征在于,所述DLL节生成模块中对DLL节进行设置时,还包括以下工作流程:

将DLL节的内存地址设置为所述最后一个节的内存地址加1;

将DLL节的节名称与EXE文件节表中的已有的节名称进行区分;

将DLL节的属性字段设置为可读可写；

将DLL节的大小赋值为需要存入的DLL文件的大小。

7.如权利要求5所述的DLL文件的隐藏系统，其特征在于，所述DLL文件数据写入模块的具体工作流程为：读取所述DLL文件至内存，采用加密算法对DLL文件数据进行加密后，将加密的DLL文件数据写入DLL节在EXE文件中的偏移处。

8.如权利要求5至7任一项所述的DLL文件的隐藏系统，其特征在于，该系统还包括应用程序执行模块，其用于：执行包含有所述DLL文件的应用程序时，将所述DLL节中的DLL文件数据进行读取，加载读取的DLL文件数据。

一种DLL文件的隐藏方法及系统

技术领域

[0001] 本发明涉及DLL文件(Dynamic Link Library,动态链接库文件)的设置领域,具体涉及一种DLL文件的隐藏方法及系统。

背景技术

[0002] 目前,应用程序大多为模块化开发,应用程序的文件组织架构一般包括一个EXE文件(executable program,可执行文件)和多个DLL文件。应用程序发布时,会将EXE文件和多个DLL文件打包形成安装文件;应用程序安装时,会将安装文件中的所有文件存放至安装目录。

[0003] 但是,EXE文件在安装过程中,盗用者会在安装文件列表中得知DLL文件的位置,进而实现自行使用或修改DLL文件的目的。当DLL文件被不正当使用或修改时,DLL文件对应的应用程序可能无法使用、或者在未授权的情况下使用,进而使得应用程序开发商的利益严重受损。

发明内容

[0004] 针对现有技术中存在的缺陷,本发明解决的技术问题为:将DLL文件隐藏在EXE文件之中,本发明隐藏的DLL文件非常难以被盗用者找到,进而显著的增大了盗用者自行任意使用应用程序难度,最大化保证了应用程序的使用安全、以及应用程序开发商的利益。

[0005] 为达到以上目的,本发明提供的DLL文件的隐藏方法,包括以下步骤:

[0006] S1:读取EXE文件至内存,将EXE文件的属性字段修改为可读可写;

[0007] S2:根据EXE文件的内存地址,获取EXE文件节表中第一个节的内存地址;根据第一个节的内存地址,获取EXE文件节表中最后一个节的内存地址;根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值;

[0008] S3:在EXE文件节表中,在最后一个节之后增加一个新的DLL节,对DLL节进行设置:将DLL节在EXE文件中的偏移值赋值为:最后一个节在EXE文件中的偏移值,加上最后一个节的大小;

[0009] S4:将DLL文件数据写入DLL节在EXE文件中的偏移处。

[0010] 本发明提供的实现上述方法的DLL文件的隐藏系统,包括EXE文件读取模块、EXE文件节表查找模块、DLL节生成模块和DLL文件数据写入模块;

[0011] EXE文件读取模块用于:读取EXE文件至内存,将EXE文件的属性字段修改为可读可写;

[0012] EXE文件节表查找模块用于:根据EXE文件的内存地址,获取EXE文件节表中第一个节的内存地址;根据第一个节的内存地址,获取EXE文件节表中最后一个节的内存地址;根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值;

[0013] DLL节生成模块用于:在EXE文件节表中,在最后一个节之后增加一个新的DLL节,

对DLL节进行设置:将DLL节在EXE文件中的偏移值赋值为:最后一个节在EXE文件中的偏移值,加上最后一个节的大小;

[0014] DLL文件数据写入模块用于:将DLL文件数据写入DLL节在EXE文件中的偏移处。

[0015] 与现有技术相比,本发明的优点在于:

[0016] 本发明在EXE文件节表中新增了一个用于储存DLL文件的DLL节,进而将DLL文件以写入DLL节的方式隐藏在EXE文件中。与现有技术中盗用者会在安装文件列表中得知DLL文件的位置相比,本发明发布和安装应用程序时,因为DLL文件隐藏在EXE文件中,所以安装文件列表中没有DLL文件。因此,盗用者非常难以在安装文件列表中找到本发明隐藏的DLL文件,进而显著的增大了盗用者自行任意使用应用程序难度,最大化保证了应用程序的使用安全、以及应用程序开发商的利益。

[0017] 进一步,本发明将DLL文件以写入DLL节时,对DLL文件数据进行加密,即使盗用者在EXE文件中找到DLL文件,也需要得知与加密算法对应的解密算法才能读取DLL文件,进一步增大了盗用者自行任意使用应用程序难度。

附图说明

[0018] 图1为本发明实施例中DLL文件的隐藏方法的流程图。

具体实施方式

[0019] 以下结合附图及实施例对本发明作进一步详细说明。

[0020] 参见图1所示,本发明实施例中的DLL文件的隐藏方法,包括以下步骤:

[0021] S1:通过windows的读取文件函数(CreateFile),读取EXE文件至内存,将EXE文件的属性字段修改为可读可写后打开EXE文件。

[0022] S2:根据EXE文件的内存地址,获取EXE文件节表中第一个节的内存地址,根据第一个节的内存地址,获取EXE文件节表中最后一个节的内存地址;根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值。

[0023] S2中EXE文件节表的解释为:EXE文件由多个节形组成,每个节存储着不同的数据,一般一个EXE文件包括.text节(代码节)、.data节(数据节)、.reloc节(重定位节)、.idata节(导入表节)等。其中所有的节信息存储在一个节表中(即EXE文件节表),每一个节信息用来告诉本节数据所在EXE文件中的偏移和本节的大小以及本节的名称和本节的属性信息,属性信息包括只读、可读可写、可执行等。

[0024] 在此基础上S2的具体流程为:

[0025] S201:根据EXE文件的内存地址,读取EXE文件的DOS头部内存地址,具体流程为:PIMAGE_DOS_HEADER pDosHeader＝(PIMAGE_DOS_HEADER)ExeAddr;其ExeAddr为EXE文件的内存地址,pDosHeader为DOS头部内存地址。

[0026] S202:根据DOS头部内存地址(pDosHeader)读取EXE文件的NT头部内存地址,具体流程为:PIMAGE_NT_HEADERS pNtHeader＝(PIMAGE_NT_HEADERS)(ExeAddr+pDosHeader->e_lfanew);其中pNtHeader为NT头部内存地址。

[0027] S203:根据NT头部内存地址(pNtHeader)获取EXE文件节表中第一个节的内存地址,具体流程为:

[0028] Int nSize＝sizeof(pNtHeader->FileHeader)+

[0029] sizeof(pNtHeader->Signature)+

[0030] pNtHeader->FileHeader.SizeOfOptionalHeader;

[0031] PIMAGE_SECTION_HEADER pSecHeader＝(PIMAGE_SECTION_HEADER)((BYTE)pNtHeader+nSize);

[0032] 其中pSecHeader为EXE文件节表的第一个节的内存地址。

[0033] S204:在EXE文件节表中,根据第一个节的内存地址,获取最后一个节的内存地址,具体流程为:

[0034] pLastSectHeader＝pSecHeader+(NumberOfSections-1);其中pLastSectHeader为最后一个节的内存地址,NumberOfSections为EXE文件节表中节的总数目。

[0035] S205:在EXE文件节表中,根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值。

[0036] S3:在EXE文件节表中,在最后一个节之后增加一个新的DLL节,对DLL节进行设置:

[0037] S301:将DLL节在EXE文件中的偏移值赋值为:最后一个节在EXE文件中的偏移值,加上最后一个节的大小。

[0038] S302:将DLL节的内存地址pNewSection设置为:pNewSection＝pLastSectHeader+1;

[0039] S303:将DLL节的节名称设置为“.dlldata”,实际应用中也可以是其他名称,但需要和EXE文件节表中的已有的节名称进行区分,即不能和已有的节名称相同。

[0040] S304:将DLL节的属性字段设置为可读可写;

[0041] S305:将DLL节的大小赋值为需要存入的DLL文件的大小。

[0042] S4:将DLL文件数据写入DLL节在EXE文件中的偏移处后,将新增有DLL节并写入DLL文件的EXE文件保存至磁盘。

[0043] S4中将DLL文件写入DLL节在EXE文件中的偏移处的具体流程为:通过windows的读取文件函数(CreateFile),读取DLL文件至内存;采用加密算法对DLL文件数据进行加密后,将加密的DLL文件数据写入DLL节在EXE文件中的偏移处。本实施例中的加密算法可以为DES加密算法(Data Encryption Standard,对称加密算法)。

[0044] S5:在电脑上执行包含有本发明的DLL文件的应用程序时,将DLL节中的DLL文件数据进行读取和解密,将解密后的DLL文件数据进行保存;应用程序加载保存的DLL文件数据,以此完成对DLL文件的加载。

[0045] 本发明提供的实现上述方法的DLL文件的隐藏系统,包括EXE文件读取模块、EXE文件节表查找模块、DLL节生成模块、DLL文件数据写入模块和应用程序执行模块。

[0046] EXE文件读取模块用于:读取EXE文件至内存,将EXE文件的属性字段修改为可读可写。

[0047] EXE文件节表查找模块用于:根据EXE文件的内存地址,获取EXE文件节表中第一个节的内存地址;根据第一个节的内存地址,获取EXE文件节表中最后一个节的内存地址;根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值。

[0048] EXE文件节表查找模块的具体工作流程为:

[0049] (1)根据所述EXE文件的内存地址,读取EXE文件的DOS头部内存地址;

[0050] (2)根据DOS头部内存地址,读取EXE文件的NT头部内存地址;

[0051] (3)根据NT头部内存地址,获取EXE文件节表中第一个节的内存地址;

[0052] (4)在EXE文件节表中,根据第一个节的内存地址,获取最后一个节的内存地址;

[0053] (5)在EXE文件节表中,根据最后一个节的内存地址,确定最后一个节的大小、以及最后一个节在EXE文件中的偏移值。

[0054] DLL节生成模块用于:在EXE文件节表中,在最后一个节之后增加一个新的DLL节,对DLL节进行设置:

[0055] (1)将DLL节在EXE文件中的偏移值赋值为:最后一个节在EXE文件中的偏移值,加上最后一个节的大小;

[0056] (2)将DLL节的内存地址设置为所述最后一个节的内存地址加1;

[0057] (3)将DLL节的节名称与EXE文件节表中的已有的节名称进行区分;

[0058] (4)将DLL节的属性字段设置为可读可写;

[0059] (5)将DLL节的大小赋值为需要存入的DLL文件的大小。

[0060] DLL文件数据写入模块用于:将DLL文件数据写入DLL节在EXE文件中的偏移处,具体工作流程为:读取所述DLL文件至内存,采用加密算法对DLL文件数据进行加密后,将加密的DLL文件数据写入DLL节在EXE文件中的偏移处。

[0061] 应用程序执行模块用于:执行包含有所述DLL文件的应用程序时,将所述DLL节中的DLL文件数据进行读取,加载读取的DLL文件数据。

[0062] 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。

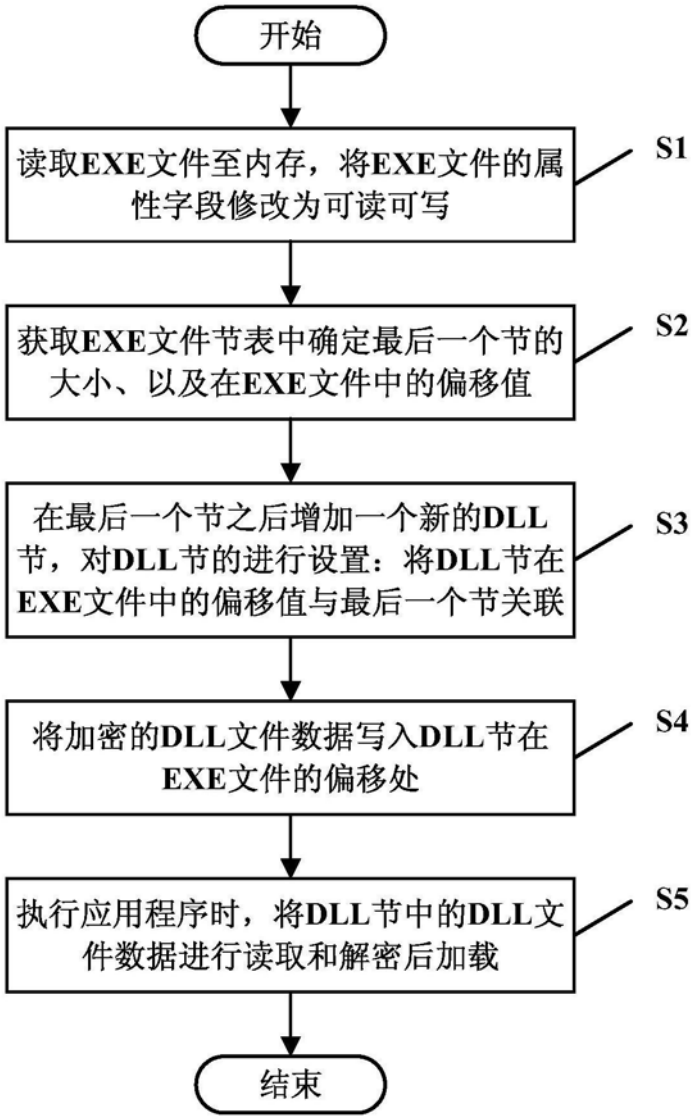

开始
读取EXE文件至内存，将EXE文件的属性字段修改为可读可写
S1
获取EXE文件节表中确定最后一个节的大小、以及在EXE文件中的偏移值
S2
在最后一个节之后增加一个新的DLL节，对DLL节的进行设置：将DLL节在EXE文件中的偏移值与最后一个节关联
S3
将加密的DLL文件数据写入DLL节在EXE文件的偏移处
S4
执行应用程序时，将DLL节中的DLL文件数据进行读取和解密后加载
S5
结束

图1