(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 106295342 A
(43)申请公布日 2017.01.04

(21)申请号 201610698138.2

(22)申请日 2016.08.19

(71)申请人 北京金山安全管理系统技术有限公司
地址 100041 北京市石景山区实兴大街30号院3号楼2层A-0003

(72)发明人 颜华甲 秦伟杰

(74)专利代理机构 北京康盛知识产权代理有限公司 11331
代理人 张宇峰

(51)Int.Cl.
G06F 21/56(2013.01)

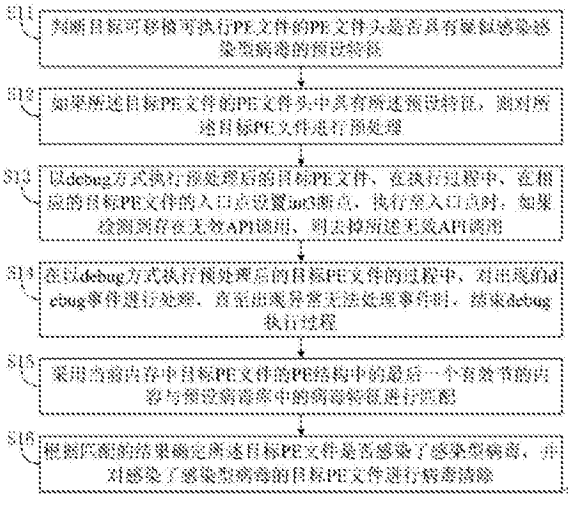
权利要求书3页 说明书7页 附图2页

(54)发明名称

检测和清除可移植可执行文件中感染型病毒的方法及装置

(57)摘要

一种检测和清除可移植可执行文件中感染型病毒的方法及装置，该方法包括：判断目标PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征；如果是，则对目标PE文件进行预处理；之后以debug方式执行相应文件，在相应文件的入口点设置int3断点，执行至入口点时，如果检测到存在无效API调用，则去掉该调用；对出现的debug事件进行处理，直至出现异常无法处理事件时，结束执行过程；采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配；根据匹配结果确定目标PE文件是否感染了感染型病毒，并对感染了感染型病毒的目标PE文件进行病毒清除。该方法能够提高病毒检测效率。



CN 106295342 A

1.一种检测和清除可移植可执行文件中感染型病毒的方法,其特征在于,包括:

判断目标可移植可执行PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征;

如果所述目标PE文件的PE文件头中具有所述预设特征,则对所述目标PE文件进行预处理;

以debug方式执行预处理后的目标PE文件,在执行过程中,在相应的目标PE文件的入口点设置int3断点,执行至入口点时,如果检测到存在无效API调用,则去掉所述无效API调用;

在以debug方式执行预处理后的目标PE文件的过程中,对出现的debug事件进行处理,直至出现异常无法处理事件时,结束debug执行过程;

采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配;

根据匹配的结果确定所述目标PE文件是否感染了感染型病毒,并对感染了感染型病毒的目标PE文件进行病毒清除。

2.根据权利要求1所述的方法,其特征在于,对所述目标PE文件进行预处理的过程,具体包括:

去除所述目标PE文件的PE文件头中的随机基址ASLR标志和动态链接库DLL标志;

将所述目标PE文件的PE文件头中的数据目录表清零;

去除所述目标PE文件的PE文件头中的导入表;

去除所述目标PE文件的PE文件头中的线程本地存储TLS字段。

3.根据权利要求2所述的方法,其特征在于,对出现的debug事件进行处理的过程,包括下述至少一个过程:

如果出现动态链接库DLL加载的debug事件,则再次判断相应的目标PE文件的入口点EP代码当前是否已经被执行,如果是,则将当前内存中的DLL代码置为0xCC后继续后续的debug执行过程;或,如果否,则保存DLL的加载信息后继续后续的debug执行过程;

如果出现断点异常的debug事件,则判断该断点异常的debug事件是否为入口点EP断点异常事件,如果是,则将当前内存中的DLL代码置为0xCC后继续后续的debug执行过程;或,如果否,则继续后续的debug执行过程。

4.根据权利要求3所述的方法,其特征在于,根据匹配的结果确定所述目标PE文件是否感染了感染型病毒的过程,具体包括:

如果所述预设病毒库中存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征,则确定所述目标PE文件感染了感染型病毒;或,

如果所述预设病毒库中不存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征,则确定所述目标PE文件未感染感染型病毒。

5.根据权利要求4所述的方法,其特征在于,对感染了感染型病毒的目标PE文件进行病毒清除的过程,具体包括:

根据与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征的特征串定位该目标PE文件被替换的程序入口点OEP代码长度、OEP位置、保存位置信息;

定位到所述目标PE文件被替换的OEP代码长度、OEP位置、保存位置信息后,修复所述目标PE文件,从而清除所述目标PE文件中的感染型病毒。

6.根据权利要求5所述的方法，其特征在于，所述预设特征包括下述至少一个特征：

所述目标PE文件的PE文件头的入口点EP所在节可写；

所述目标PE文件的PE文件头的节区数目大于一；

内存中目标PE文件的PE结构中的最后一个有效节包含可写和可执行属性。

7.一种检测和清除可移植可执行文件中感染型病毒的装置，其特征在于，包括：

疑似感染分析模块，用于判断目标可移植可执行PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征；

预处理模块，用于如果所述目标PE文件的PE文件头中具有所述预设特征，则对所述目标PE文件进行预处理；

执行模块，用于以debug方式执行预处理后的目标PE文件，在执行过程中，在相应的目标PE文件的入口点设置int3断点，执行至入口点时，如果检测到存在无效API调用，则去掉所述无效API调用；

事件处理模块，用于在以debug方式执行预处理后的目标PE文件的过程中，对出现的debug事件进行处理，直至出现异常无法处理事件时，结束debug执行过程；

特征匹配模块，用于采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配；

病毒清除模块，用于根据匹配的结果确定所述目标PE文件是否感染了感染型病毒，并对感染了感染型病毒的目标PE文件进行病毒清除。

8.根据权利要求7所述的装置，其特征在于，所述预处理模块具体用于：

去除所述目标PE文件的PE文件头中的随机基址ASLR标志和动态链接库DLL标志；

将所述目标PE文件的PE文件头中的数据目录表清零；

去除所述目标PE文件的PE文件头中的导入表；

去除所述目标PE文件的PE文件头中的线程本地存储TLS字段。

9.根据权利要求8所述的装置，其特征在于，所述事件处理模块包括下述至少一个单元：

第一事件处理单元，用于如果出现动态链接库DLL加载的debug事件，则再次判断相应的目标PE文件的入口点EP代码当前是否已经被执行，如果是，则将当前内存中的DLL代码置为0xCC后，触发所述执行模块继续后续的debug执行过程；或，如果否，则保存DLL的加载信息后，触发所述执行模块继续后续的debug执行过程；

第二事件处理单元，用于如果出现断点异常的debug事件，则判断该断点异常的debug事件是否为入口点EP断点异常事件，如果是，则将当前内存中的DLL代码置为0xCC后，触发所述执行模块继续后续的debug执行过程；或，如果否，则触发所述执行模块继续后续的debug执行过程。

10.根据权利要求9所述的装置，其特征在于，所述病毒清除模块具体包括：

病毒感染确定单元，用于如果所述预设病毒库中存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件感染了感染型病毒；或，如果所述预设病毒库中不存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件未感染感染型病毒；

病毒感染清除单元，用于根据与当前内存中目标PE文件的PE结构中的最后一个有效节

的内容相匹配的病毒特征的特征串定位该目标PE文件被替换的程序入口点OEP代码长度、OEP位置、保存位置信息；定位到所述目标PE文件被替换的OEP代码长度、OEP位置、保存位置信息后，修复所述目标PE文件，从而清除所述目标PE文件中的感染型病毒。

检测和清除可移植可执行文件中感染型病毒的方法及装置

技术领域

[0001] 本发明属于安全防御技术领域，尤其涉及一种检测和清除可移植可执行文件中感染型病毒的方法及装置。

背景技术

[0002] 通常，感染型病毒在感染正常可移植可执行PE(Portable Executable,PE)文件时，在不同的PE文件中植入的代码都可能不同，形态多变，而且感染型病毒在感染PE文件时，通常会将原PE文件入口点处的一段代码覆盖，并且将原PE文件入口点处的该段被覆盖的代码加密后保存在被感染后的PE文件中，这样，一方面，被感染型病毒感染的PE文件执行时需要先解密被加密的代码，使得传统的基于特征码的查杀技术，很难提取针对这类感染型病毒的通用特征码。另一方面，如果想要清除PE文件中的感染型病毒，必须还原该段被加密的代码。

[0003] 现有技术中，清除文件中感染型病毒采用的方式通常为：分析感染后的文件，逆向病毒的解密代码。这样，由于感染型病毒的变种颇多，不同种类的感染型病毒的加密和解密的逻辑都可能发生变化，一旦感染型病毒的加密和解密逻辑发生变化，反病毒工程师就需要花费大量的人力物力来对其进行分析，大大增加了这类病毒的检测和清除的难度，并且延长了检测和清除的时间周期。

[0004] 所以，现有的检测和清除PE文件中感染型病毒的方法，检测和清除病毒的难度较大，周期较长。

发明内容

[0005] 有鉴于此，本发明的一个目的是提出一种检测和清除可移植可执行文件中感染型病毒的方法，以解决现有的检测和清除可移植可执行文件中感染型病毒的方法中，检测和清除病毒的难度较大，周期较长的问题。为了对披露的实施例的一些方面有一个基本的理解，下面给出了简单的概括。该概括部分不是泛泛评述，也不是要确定关键/重要组成元素或描绘这些实施例的保护范围。其唯一目的是用简单的形式呈现一些概念，以此作为后面的详细说明的序言。

[0006] 在一些可选的实施例中，该方法包括：判断目标可移植可执行PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征；如果所述目标PE文件的PE文件头中具有所述预设特征，则对所述目标PE文件进行预处理；以debug方式执行预处理后的目标PE文件，在执行过程中，在相应的目标PE文件的入口点设置int3断点，执行至入口点时，如果检测到存在无效API调用，则去掉所述无效API调用；在以debug方式执行预处理后的目标PE文件的过程中，对出现的debug事件进行处理，直至出现异常无法处理事件时，结束debug执行过程；采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配；根据匹配的结果确定所述目标PE文件是否感染了感染型病毒，并对感染了感染型病毒的目标PE文件进行病毒清除。

[0007] 进一步，对所述目标PE文件进行预处理的过程，具体包括：去除所述目标PE文件的PE文件头中的随机基址ASLR标志和动态链接库DLL标志；将所述目标PE文件的PE文件头中的数据目录表清零；去除所述目标PE文件的PE文件头中的导入表；去除所述目标PE文件的PE文件头中的线程本地存储TLS字段。

[0008] 进一步，对出现的debug事件进行处理的过程，包括下述至少一个过程：如果出现动态链接库DLL加载的debug事件，则再次判断相应的目标PE文件的入口点EP代码当前是否已经被执行，如果是，则将当前内存中的DLL代码置为0xCC后继续后续的debug执行过程；或，如果否，则保存DLL的加载信息后继续后续的debug执行过程；如果出现断点异常的debug事件，则判断该断点异常的debug事件是否为入口点EP断点异常事件，如果是，则将当前内存中的DLL代码置为0xCC后继续后续的debug执行过程；或，如果否，则继续后续的debug执行过程。

[0009] 进一步，根据匹配的结果确定所述目标PE文件是否感染了感染型病毒的过程，具体包括：如果所述预设病毒库中存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件感染了感染型病毒；或，如果所述预设病毒库中不存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件未感染感染型病毒。

[0010] 进一步，对感染了感染型病毒的目标PE文件进行病毒清除的过程，具体包括：根据与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征的特征串定位该目标PE文件被替换的程序入口点OEP代码长度、OEP位置、保存位置信息；定位到所述目标PE文件被替换的OEP代码长度、OEP位置、保存位置信息后，修复所述目标PE文件，从而清除所述目标PE文件中的感染型病毒。

[0011] 进一步，预设特征包括下述至少一个特征：所述目标PE文件的PE文件头的入口点EP所在节可写；所述目标PE文件的PE文件头的节区数目大于一；内存中目标PE文件的PE结构中的最后一个有效节包含可写和可执行属性。

[0012] 本发明的另一个目的是提出一种检测和清除可移植可执行文件中感染型病毒的装置。

[0013] 在一些可选的实施例中，该装置包括：疑似感染分析模块，用于判断目标可移植可执行PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征；预处理模块，用于如果所述目标PE文件的PE文件头中具有所述预设特征，则对所述目标PE文件进行预处理；执行模块，用于以debug方式执行预处理后的目标PE文件，在执行过程中，在相应的目标PE文件的入口点设置int3断点，执行至入口点时，如果检测到存在无效API调用，则去掉所述无效API调用；事件处理模块，用于在以debug方式执行预处理后的目标PE文件的过程中，对出现的debug事件进行处理，直至出现异常无法处理事件时，结束debug执行过程；特征匹配模块，用于采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配；病毒清除模块，用于根据匹配的结果确定所述目标PE文件是否感染了感染型病毒，并对感染了感染型病毒的目标PE文件进行病毒清除。

[0014] 进一步，预处理模块具体用于：去除所述目标PE文件的PE文件头中的随机基址ASLR标志和动态链接库DLL标志；将所述目标PE文件的PE文件头中的数据目录表清零；去除所述目标PE文件的PE文件头中的导入表；去除所述目标PE文件的PE文件头中的线程本地存

储TLS字段。

[0015] 进一步，事件处理模块包括下述至少一个单元：第一事件处理单元，用于如果出现动态链接库DLL加载的debug事件，则再次判断相应的目标PE文件的入口点EP代码当前是否已经被执行，如果是，则将当前内存中的DLL代码置为0xCC后，触发所述执行模块继续后续的debug执行过程；或，如果否，则保存DLL的加载信息后，触发所述执行模块继续后续的debug执行过程；第二事件处理单元，用于如果出现断点异常的debug事件，则判断该断点异常的debug事件是否为入口点EP断点异常事件，如果是，则将当前内存中的DLL代码置为0xCC后，触发所述执行模块继续后续的debug执行过程；或，如果否，则触发所述执行模块继续后续的debug执行过程。

[0016] 进一步，病毒清除模块具体包括：病毒感染确定单元，用于如果所述预设病毒库中存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件感染了感染型病毒；或，如果所述预设病毒库中不存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件未感染感染型病毒；病毒感染清除单元，用于根据与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征的特征串定位该目标PE文件被替换的程序入口点OEP代码长度、OEP位置、保存位置信息；定位到所述目标PE文件被替换的OEP代码长度、OEP位置、保存位置信息后，修复所述目标PE文件，从而清除所述目标PE文件中的感染型病毒。

[0017] 与现有技术相比，本发明的有益效果为：

[0018] 本发明提供一种检测和清除可移植可执行文件中感染型病毒的方法及装置，该方法中，首先初步判断目标PE文件是否疑似感染感染型病毒，如果确定目标PE文件疑似感染感染型病毒，则对该目标PE文件进行预处理，之后以debug方式执行预处理后的目标PE文件，在执行过程中，对不同的debug事件进行处理，直至出现异常无法处理事件时，停止debug执行过程，此时，病毒已经被解密，采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒进行匹配，如果匹配成功，说明目标PE文件确实感染了感染型病毒，采用与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征的特征串定位目标PE文件被替换的程序入口点OEP(Original Entry Point，OEP)代码长度、OEP位置、保存位置信息，定位到之后，修复目标PE文件，即可完成对目标PE文件中的感染型病毒清除的目的，无需再花费大量时间与工程逆向病毒的解密代码，病毒的检测和清除过程更加简单，并且极大的缩短了检测和清除病毒的时间周期，效率更高。

[0019] 为了上述以及相关的目的，一个或多个实施例包括后面将详细说明并在权利要求中特别指出的特征。下面的说明以及附图详细说明某些示例性方面，并且其指示的仅仅是各个实施例的原则可以利用的各种方式中的一些方式。其它的益处和新颖性特征将随着下面的详细说明结合附图考虑而变得明显，所公开的实施例是要包括所有这些方面以及它们的等同。

附图说明

[0020] 图1是本发明实施例的一种检测和清除可移植可执行文件中感染型病毒的方法的流程示意图；

[0021] 图2是本发明实施例的一种检测和清除可移植可执行文件中感染型病毒的装置的结构框图。

具体实施方式

[0022] 以下描述和附图充分地示出本发明的具体实施方案，以使本领域的技术人员能够实践它们。实施例仅代表可能的变化。除非明确要求，否则单独的部件和功能是可选的，并且操作的顺序可以变化。一些实施方案的部分和特征可以被包括在或替换其他实施方案的部分和特征。本发明的实施方案的范围包括权利要求书的整个范围，以及权利要求书的所有可获得的等同物。在本文中，本发明的这些实施方案可以被单独地或总地用术语“发明”来表示，这仅仅是为了方便，并且如果事实上公开了超过一个的发明，不是要自动地限制该应用的范围为任何单个发明或发明构思。

[0023] 现在结合附图进行说明，图1示出的是一些可选的实施例中一种检测和清除可移植可执行文件中感染型病毒的方法的流程图；图2示出的是一些可选的实施例中一种检测和清除可移植可执行文件中感染型病毒的装置的结构框图。

[0024] 如图1所示，在一些可选的实施例中，公开了一种检测和清除可移植可执行文件中感染型病毒的方法，该方法包括：

[0025] S11、判断目标可移植可执行PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征；

[0026] 为了确保可移植可执行PE文件在被执行时，能够保证系统运行的安全可靠性，用户通常会对PE文件进行感染型病毒的检测和清除，本文中，将用户想要对其进行感染型病毒检测和清除的当前可移植可执行PE文件定义为目标可移植可执行PE文件，简称目标PE文件，任意一个用户想要对其进行感染型病毒检测和清除的PE文件都可以作为目标PE文件。

[0027] 预设特征预先存储于检测系统内，包括下述至少一个特征：所述目标PE文件的PE文件头的入口点EP所在节可写；所述目标PE文件的PE文件头的节区数目大于一；内存中目标PE文件的PE结构中的最后一个有效节包含可写和可执行属性。亦即，只要目标PE文件的PE文件头中出现上述一个或多个特征，均认为该目标PE文件疑似感染感染型病毒。并且，用户可以将上述未提及的其它与目标PE文件的PE文件头相关的、可以判断目标PE文件疑似感染感染型病毒的特征作为预设特征预先存储于检测系统内。

[0028] S12、如果所述目标PE文件的PE文件头中具有所述预设特征，则对所述目标PE文件进行预处理；

[0029] 对目标PE文件进行预处理的过程，具体包括：去除所述目标PE文件的PE文件头中的随机基址ASLR(Address space layout randomization，ASLR)标志和动态链接库DLL(Dynamic Link Library，DLL)标志；将所述目标PE文件的PE文件头中的数据目录表清零；去除所述目标PE文件的PE文件头中的导入表；去除所述目标PE文件的PE文件头中的线程本地存储TLS字段。

[0030] S13、以debug方式执行预处理后的目标PE文件，在执行过程中，在相应的目标PE文件的入口点设置int3断点，执行至入口点时，如果检测到存在无效API调用，则去掉所述无效API调用；

[0031] S14、在以debug方式执行预处理后的目标PE文件的过程中，对出现的debug事件进

行处理，直至出现异常无法处理事件时，结束debug执行过程；

[0032] 在以debug方式执行预处理后的目标PE文件的过程中，可能会出现debug事件，对出现的每一个debug事件需要进行相应的处理。对debug执行过程中出现的debug事件进行处理的过程，包括下述至少一个过程：

[0033] 如果出现动态链接库DLL加载的debug事件，则再次判断相应的目标PE文件的入口点EP代码当前是否已经被执行，如果是，则将当前内存中的DLL代码置为0xCC后继续后续的debug执行过程；或，如果否，则保存DLL的加载信息后继续后续的debug执行过程；

[0034] 如果出现断点异常的debug事件，则判断该断点异常的debug事件是否为入口点EP断点异常事件，如果是，则将当前内存中的DLL代码置为0xCC后继续后续的debug执行过程；或，如果否，则继续后续的debug执行过程。

[0035] S15、采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配；

[0036] debug执行过程结束后，病毒便已经被解密，之后采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配后，就可以根据匹配结果确定目标PE文件是否感染了感染型病毒，并根据匹配的病毒特征对目标PE文件进行修复，以清除目标PE文件中的感染型病毒。整个过程中，无需再花费大量时间与工程逆向病毒的解密代码，使得病毒的检测和清除更加简单，并且极大的减小了检测和清除病毒的时间周期，提高了检测效率。

[0037] S16、根据匹配的结果确定所述目标PE文件是否感染了感染型病毒，并对感染了感染型病毒的目标PE文件进行病毒清除。

[0038] 根据匹配的结果确定所述目标PE文件是否感染了感染型病毒的过程，具体包括：如果所述预设病毒库中存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件感染了感染型病毒；或，如果所述预设病毒库中不存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件未感染感染型病毒。

[0039] 对感染了感染型病毒的目标PE文件进行病毒清除的过程，具体包括：根据与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征的特征串定位该目标PE文件被替换的程序入口点OEP代码长度、OEP位置、保存位置信息；定位到所述目标PE文件被替换的OEP代码长度、OEP位置、保存位置信息后，修复所述目标PE文件，从而清除所述目标PE文件中的感染型病毒。

[0040] 该方法中，首先初步判断目标PE文件是否疑似感染感染型病毒，如果确定目标PE文件疑似感染感染型病毒，则对该目标PE文件进行预处理，之后以debug方式执行预处理后的目标PE文件，在执行过程中，对不同的debug事件进行处理，直至出现异常无法处理事件时，停止debug执行过程，此时，病毒已经被解密，采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒进行匹配，如果匹配成功，说明目标PE文件确实感染了感染型病毒，采用与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征的特征串定位目标PE文件被替换的程序入口点OEP(Original Entry Point，OEP)代码长度、OEP位置、保存位置信息，定位到之后，修复目标PE文件，即可完成对目标PE文件中的感染型病毒清除的目的，无需再花费大量时间与工程逆向病毒的解密代

码，病毒的检测和清除过程更加简单，并且极大的缩短了检测和清除病毒的时间周期，效率更高。

[0041] 如图2所示，在一些可选的实施例中，公开了一种检测和清除可移植可执行文件中感染型病毒的装置200，该装置200包括：

[0042] 疑似感染分析模块201，用于判断目标可移植可执行PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征；

[0043] 预处理模块202，用于如果所述目标PE文件的PE文件头中具有所述预设特征，则对所述目标PE文件进行预处理；

[0044] 执行模块203，用于以debug方式执行预处理后的目标PE文件，在执行过程中，在相应的目标PE文件的入口点设置int3断点，执行至入口点时，如果检测到存在无效API调用，则去掉所述无效API调用；

[0045] 事件处理模块204，用于在以debug方式执行预处理后的目标PE文件的过程中，对出现的debug事件进行处理，直至出现异常无法处理事件时，结束debug执行过程；

[0046] 特征匹配模块205，用于采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配；

[0047] 病毒清除模块206，用于根据匹配的结果确定所述目标PE文件是否感染了感染型病毒，并对感染了感染型病毒的目标PE文件进行病毒清除。

[0048] 进一步，预处理模块202具体用于：

[0049] 去除所述目标PE文件的PE文件头中的随机基址ASLR标志和动态链接库DLL标志；

[0050] 将所述目标PE文件的PE文件头中的数据目录表清零；

[0051] 去除所述目标PE文件的PE文件头中的导入表；

[0052] 去除所述目标PE文件的PE文件头中的线程本地存储TLS字段。

[0053] 进一步，事件处理模块204包括下述至少一个单元：

[0054] 第一事件处理单元2041，用于如果出现动态链接库DLL加载的debug事件，则再次判断相应的目标PE文件的入口点EP代码当前是否已经被执行，如果是，则将当前内存中的DLL代码置为0xCC后，触发所述执行模块203继续后续的debug执行过程；或，如果否，则保存DLL的加载信息后，触发所述执行模块203继续后续的debug执行过程；

[0055] 第二事件处理单元2042，用于如果出现断点异常的debug事件，则判断该断点异常的debug事件是否为入口点EP断点异常事件，如果是，则将当前内存中的DLL代码置为0xCC后，触发所述执行模块203继续后续的debug执行过程；或，如果否，则触发所述执行模块203继续后续的debug执行过程。

[0056] 进一步，病毒清除模块206具体包括：

[0057] 病毒感染确定单元2061，用于如果所述预设病毒库中存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件感染了感染型病毒；或，如果所述预设病毒库中不存在与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征，则确定所述目标PE文件未感染感染型病毒；

[0058] 病毒感染清除单元2062，用于根据与当前内存中目标PE文件的PE结构中的最后一个有效节的内容相匹配的病毒特征的特征串定位该目标PE文件被替换的程序入口点OEP代码长度、OEP位置、保存位置信息；定位到所述目标PE文件被替换的OEP代码长度、OEP位置、

保存位置信息后，修复所述目标PE文件，从而清除所述目标PE文件中的感染型病毒。

[0059] 采用该装置对目标PE文件进行感染型病毒的检测和清除时，无需花费大量时间与工程逆向病毒的解密代码，病毒的检测和清除过程更加简单，并且极大的缩短了检测和清除病毒的时间周期，效率更高。

[0060] 总之，以上所述仅为本发明的实施例，并非用于限定本发明的保护范围，而用于说明本发明。凡在本发明的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本发明的保护范围之内。

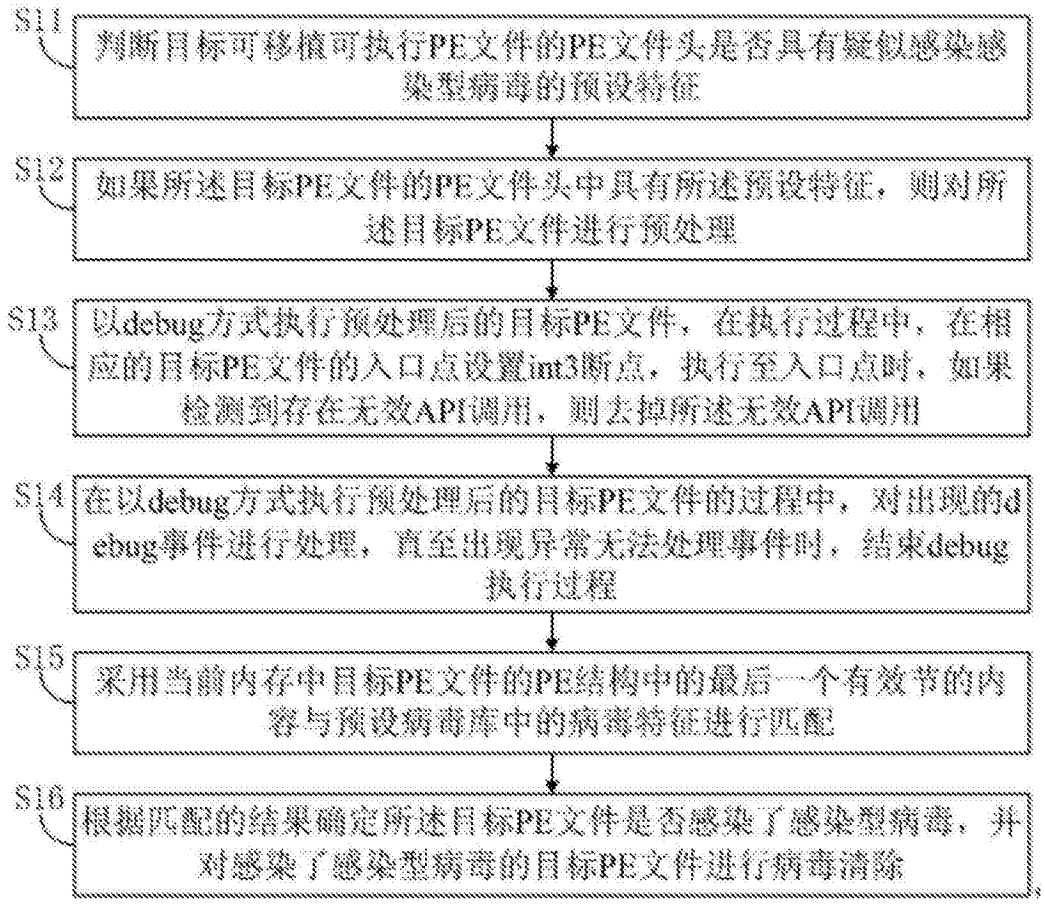

S11
判断目标可移植可执行PE文件的PE文件头是否具有疑似感染感染型病毒的预设特征
S12
如果所述目标PE文件的PE文件头中具有所述预设特征，则对所述目标PE文件进行预处理
S13
以debug方式执行预处理后的目标PE文件，在执行过程中，在相应的目标PE文件的入口点设置int3断点，执行至入口点时，如果检测到存在无效API调用，则去掉所述无效API调用
S14
在以debug方式执行预处理后的目标PE文件的过程中，对出现的debug事件进行处理，直至出现异常无法处理事件时，结束debug执行过程
S15
采用当前内存中目标PE文件的PE结构中的最后一个有效节的内容与预设病毒库中的病毒特征进行匹配
S16
根据匹配的结果确定所述目标PE文件是否感染了感染型病毒，并对感染了感染型病毒的目标PE文件进行病毒清除

图1

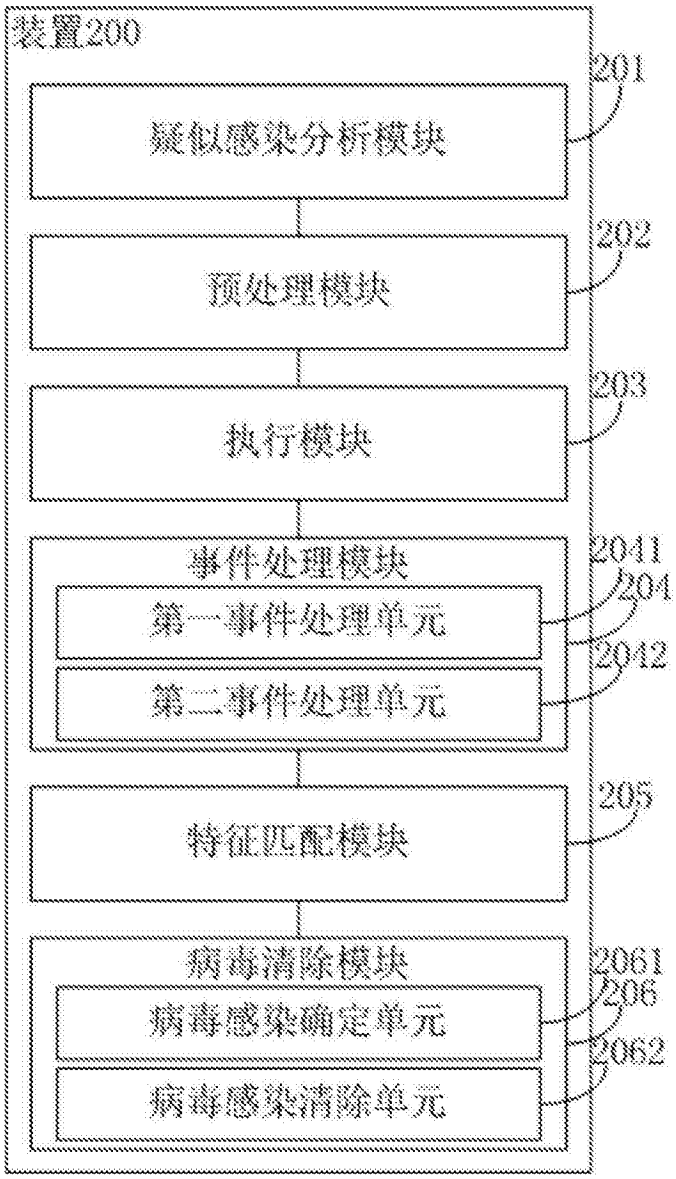

装置200
201
疑似感染分析模块
202
预处理模块
203
执行模块
2041
事件处理模块
204
第一事件处理单元
2042
第二事件处理单元
205
特征匹配模块
2061
病毒清除模块
206
病毒感染确定单元
2062
病毒感染清除单元

图2