(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 101753570 A
(43)申请公布日 2010.06.23

(21)申请号 200910254092.5

(22)申请日 2009.12.18

(30)优先权数据
12/338,877 2008.12.18 US

(71)申请人 赛门铁克公司
地址 美国加利福尼亚州

(72)发明人 马克·肯尼迪

(74)专利代理机构 中原信达知识产权代理有限责任公司 11219
代理人 张焕生 安翔

(51)Int.Cl.
H04L 29/06(2006.01)

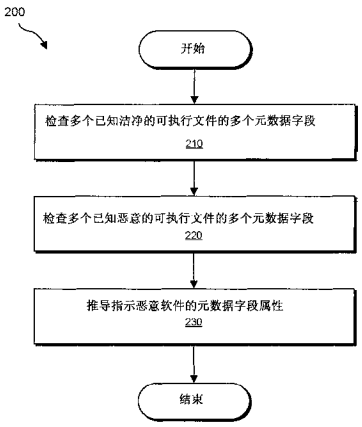
权利要求书 5 页 说明书 12 页 附图 6 页

(54)发明名称

用于检测恶意软件的方法和系统

(57)摘要

公开了一种用于检测恶意软件的方法。该方法可以包括对多个已知洁净的可执行文件的多个元数据字段进行检查。该方法还可以包括对多个已知恶意的可执行文件的多个元数据字段进行检查。该方法可以进一步包括:基于从对多个已知洁净和已知恶意的可执行文件的多个元数据字段进行检查所获取的信息,推导指示恶意软件的元数据字段属性。还公开了对应的系统和计算机可读介质。



CN 101753570 A

1. 一种用于检测恶意软件的计算机实现的方法，所述计算机实现的方法包括：

检查多个已知洁净的可执行文件的多个元数据字段；

检查多个已知恶意的可执行文件的多个元数据字段；

基于从对所述多个已知洁净和已知恶意的可执行文件的所述多个元数据字段进行检查所获取的信息，推导指示恶意软件的元数据字段属性。

2. 如权利要求 1 所述的计算机实现的方法，进一步包括：

接收未知的可执行文件；

通过确定所述未知的可执行文件是否包含指示恶意软件的所述元数据字段属性，来确定所述未知的可执行文件是否包含恶意软件。

3. 如权利要求 2 所述的计算机实现的方法，进一步包括：

如果所述未知的可执行文件包括恶意软件，则执行安全动作。

4. 如权利要求 1 所述的计算机实现的方法，其中：

所述多个已知洁净的可执行文件包括可移植可执行文件；

所述多个已知恶意的可执行文件包括可移植可执行文件；

所述元数据字段属性包括头字段属性。

5. 如权利要求 1 所述的计算机实现方法，其中指示恶意软件的所述元数据字段属性包括以下中的至少一个：

调试段属性；

导入属性；

符号表属性；

可选头属性；

特征属性；

图像子系统属性；

链接器版本属性；

大小属性；

真实虚拟地址属性；

入口点属性；

代码段基础属性；

对准属性；

操作系统版本属性；

图像版本属性；

最低子系统版本属性；

动态链接库特征属性；

堆栈大小属性；

堆大小属性；

段数属性；

无进出属性；

线程级前瞻属性；

图像基础属性。

6. 如权利要求 1 所述的计算机实现的方法，其中指示恶意软件的所述元数据字段属性包括：在所述多个已知洁净的可执行文件和已知恶意的可执行文件中检查到的所述多个元数据字段的属性的子集。

7. 如权利要求 1 所述的计算机实现的方法，其中指示恶意软件的所述元数据字段属性包括静态属性。

8. 如权利要求 1 所述的计算机实现的方法，其中推导指示恶意软件的元数据字段属性包括：确定指示恶意软件的元数据字段属性的至少一个组合。

9. 如权利要求 1 所述的计算机实现的方法，其中：

推导指示恶意软件的元数据字段属性包括：确定指示恶意软件的第一元数据字段属性是比指示恶意软件的第二元数据字段属性更强的恶意软件指示符。

10. 如权利要求 1 所述的计算机实现的方法，有形地具体化为在至少一个计算机可读介质上的计算机可执行指令。

11. 一种用于检测恶意软件的计算机实现的方法，所述计算机实现的方法包括：

检查多个已知洁净的可移植可执行文件的多个元数据字段；

检查多个已知恶意的可移植可执行文件的多个元数据字段；

基于从对所述多个已知洁净和已知恶意的可移植可执行文件的所述多个元数据字段进行检查所获取的信息，推导指示恶意软件的元数据字段属性；

接收未知的可执行文件；

通过确定所述未知的可执行文件是否包含指示恶意软件的所述元数据字段属性，来确定所述未知的可执行文件是否包含恶意软件。

12. 如权利要求 11 所述的计算机实现的方法，其中指示恶意软件的所述元数据字段属性包括以下中的至少一个：

调试段属性；

符号表属性；

图像子系统属性；

链接器版本属性；

真实虚拟地址属性；

操作系统版本属性；

图像版本属性；

最低子系统版本属性；

动态链接库特征属性；

堆栈大小属性；

堆大小属性；

段数属性。

13. 如权利要求 11 所述的计算机实现的方法，其中指示恶意软件的所述元数据字段属性包括以下中的至少两个：

段对准整数；

文件对准整数；

主要操作系统版本整数；

次要操作系统版本整数；
主要图像版本整数；
次要图像版本整数；
主要子系统版本整数；
次要子系统版本整数；
图像大小整数；
头大小整数；
图形用户界面整数；
基于字符的用户界面整数；
堆栈保留大小整数；
堆栈提交大小整数；
堆保留大小整数；
堆提交大小整数；
符号表指针整数；
符号数整数；
调试段整数；
主要链接器版本整数；
次要链接器版本整数；
代码段大小整数；
已初始化数据大小整数；
未初始化数据大小整数；
真实虚拟地址入口点整数；
代码段的真实虚拟地址开始整数；
代码段的真实虚拟地址基础整数；
数据段的真实虚拟地址开始整数；
图像基础整数；
具有导入整数；
具有延迟的导入整数；
外部绑定设施整数；
无进出整数；
urlmon 导入整数；
线程级前瞻整数；
msvcrt 导入整数；
oleaut32 导入整数；
setupapi 导入整数；
user32 导入整数；
advapi32 导入整数；
shell32 导入整数；
gdi32 导入整数；

comdlg32 导入整数；

imm32 导入整数；

具有证书整数；

节点整数；

段数整数。

14. 如权利要求 11 所述的计算机实现的方法，进一步包括：

如果所述未知的可执行文件包括恶意软件，则执行安全动作，其中所述安全动作包括以下中的至少一个：

隔离所述未知的可执行文件；

将所述未知的可执行文件报告给安全软件销售商；

将所述未知的可执行文件添加到恶意软件文件列表。

15. 如权利要求 11 所述的计算机实现的方法，有形地具体化为在至少一个计算机可读介质上的计算机可执行指令。

16. 一种系统，包括：

检查模块，被编程为：

检查多个已知洁净的可执行文件的多个头字段；

检查多个已知恶意的可执行文件的多个头字段；

数据库，与所述检查模块进行通信，并被配置为存储所述检查模块获取的信息；

推导模块，被编程为：基于从所述检查模块获取的信息，推导指示恶意软件的头字段属性。

17. 如权利要求 16 所述的系统，进一步包括：

安全系统，被编程为：

接收未知的可执行文件；

通过确定所述未知的可执行文件是否包含指示恶意软件的所述头字段属性，来确定所述未知的可执行文件是否包含恶意软件。

18. 如权利要求 17 所述的系统，其中所述安全模块进一步被编程为：如果所述未知的可执行文件包括恶意软件，则执行安全动作。

19. 如权利要求 16 所述的系统，其中所述头字段属性包括以下中的至少一个：

调试段属性；

导入属性；

符号表属性；

可选头属性；

特征属性；

图像子系统属性；

链接器版本属性；

大小属性；

真实虚拟地址属性；

入口点属性；

代码段基础属性；

对准属性；
操作系统版本属性；
图像版本属性；
最低子系统版本属性；
动态链接库特征属性；
堆栈大小属性；
堆大小属性；
段数属性；
无进出属性；
线程级前瞻属性；
图像基础属性。

20. 如权利要求 16 所述的系统，其中指示恶意软件的所述头字段属性包括静态属性。

用于检测恶意软件的方法和系统

背景技术

[0001] 用户和企业越来越依赖于用计算机存储敏感数据。结果，恶意程序员似乎不断地增加其获取对其他计算机非法控制和访问的努力。具有恶意动机的计算机程序员已经并将继续创造病毒、特洛伊木马、蠕虫、和其他意图损害计算机系统和他人数据的程序。通常将这些恶意程序称为恶意软件 (malware)。

[0002] 为抵抗日益增多的恶意软件，安全软件公司为其用户定期创建和部署恶意软件签名（例如，识别恶意软件的散列函数）。然而，还有相当数量的恶意软件尚未被识别。所以，需要一种用于检测未识别的恶意软件的过程。

发明内容

[0003] 本公开的实施例涉及基于恶意文件的一个或多个元数据字段属性对恶意文件进行检测。例如，检查模块可以对多个已知洁净的可执行文件的多个元数据字段进行检查。检查模块还可以对多个已知恶意的可执行文件的多个元数据字段进行检查。推导模块可以基于通过对多个已知洁净和已知恶意的可执行文件的多个元数据字段进行检查所收集的信息来推导指示恶意软件的元数据字段属性。

[0004] 在一些实施例中，安全模块可以使用从对多个已知洁净和已知恶意的可执行文件的元数据字段的检查所推导出的信息，来确定未知的可执行文件是否包括恶意软件。例如，安全模块可以接收未知的可执行文件。然后，安全模块可以通过确定该未知的可执行文件是否包含指示恶意软件的元数据字段属性，来确定该未知的可执行文件是否包含恶意软件。

[0005] 在至少一个实施例中，如果未知的可执行文件包括恶意软件，安全模块可以执行安全动作。安全模块可以通过隔离该未知的可执文件、把该未知的可执行文件报告给安全软件销售商、将该未知的执行文件添加到恶意软件文件列表和 / 或通过执行任何其他适当的安动作来执行安全动作。根据此处描述的一般原理，取自任何上述实例的特征可以相互结合使用。通过结合附图阅读以下详细描述和权要求，将会更加透彻理解这些和其他实施例、特征、和优点。

附图说明

[0006] 附图说明了多个示例性实施例，并且是本说明的一部分。与以下描述一起，这些附图展示并解释了本公开的各种原理。

[0007] 图 1 是根据某些实施例的用于检测恶意软件的示例性系统的框图。

[0008] 图 2 是根据某些实施例的用于检测恶意软件的示例性方法的流程图。

[0009] 图 3 是根据某些实施例的示例性可执行文件的框图。

[0010] 图 4 是根据某些实施例的用于检测恶意软件的示例性方法的流程图。

[0011] 图 5 是能够实现此处描述和 / 或说明的一个或多个实施例的示例性计算系统的框图。

[0012] 图 6 是能够实现此处描述和 / 或说明的一个或多个实施例的示例性计算网络的框图。

[0013] 在所有附图中，相同的附图标记和描述指示相似但不一定是相同的元素。尽管此处描述的示例性实施例可以采用各种修改和替代形式，但是，特定实施例已通过附图中的示例示出，并将在此处详细描述。然而，此处描述的示例性实施例并非意欲限于公开的特定形式。更确切地说，本公开覆盖落于所附权利要求范围内的所有修改、等效物和替选。

具体实施方式

[0014] 如以下将要详细描述的，本公开一般地涉及通过对已知洁净和已知恶意的可执行文件进行检查，并使用在检查期间获取的信息来确定未知文件是否包含恶意软件，从而检测恶意软件的方法和系统。图 1 是用于检查已知洁净和已知恶意的可执行文件并确定未知的可执行文件是否包含恶意软件的示例性系统的图。图 2 示出用于检查已知洁净和已知恶意的可执行文件并推导指示恶意软件的元数据属性的示例性过程。图 3 示出示例性可移植可执行 (PE，Portable Executable) 文件，而图 4 示出用于确定 PE 文件是否包含恶意软件的示例性过程。在图 5 和 6 中示出用于实现本公开的实施例的示例性计算系统和示例性网络。

[0015] 图 1 是示例性系统 100 的框图。系统 100 可以包括用于执行一个或多个任务的一个或多个模块 110。例如，模块 110 可以包括检查模块 112，用于检查已知洁净和已知恶意的可执行文件的元数据字段。可执行文件的元数据字段可以包括头字段和 / 或可执行文件的其他字段。模块 110 还可以包括推导模块 114，用于基于从对已知洁净和已知恶意的可执行软件进行检查所获取的信息，推导指示恶意软件的元数据字段属性。模块 110 可以包括安全模块 116，其可以使用指示恶意软件的元数据字段属性，来确定未知的可执行文件是否包含恶意软件。

[0016] 在某些实施例中，图 1 中的模块 110 中的一个或多个可以代表一个或多个软件应用或程序，当其被计算系统执行时，可以使计算系统执行此处公开的一个或多个步骤。例如，如以下将要详细描述的，模块 110 中的一个或多个可以代表被配置来在一个或多个计算设备上运行的软件模块，所述计算设备诸如图 5 中的计算系统 510 和 / 或图 6 中的示例性网络架构 600 的一部分。图 1 中的一个或多个模块 110 还可以代表全部或部分被配置来执行与此处公开的步骤相关联的一个或多个任务的一个或多个专用计算机。

[0017] 系统 100 可以包括数据库 120。数据库 120 可以包括元数据字段数据库 122，其可以存储待在已知洁净和已知恶意的可执行文件中检查的一个或多个元数据字段的列表。数据库 120 还可以包括恶意软件元数据字段信息数据库 124。恶意软件元数据字段信息数据库 124 可以包括 :识别指示恶意软件的元数据字段属性和 / 或元数据字段属性可以如何指示恶意软件的任何信息。指示恶意软件的元数据字段属性可以包括以下的一个或多个元数据字段属性，即当在可执行文件中发现其时，可以指示该可执行文件包括恶意软件。

[0018] 在一些实施例中，恶意软件元数据字段信息数据库 124 可以包括以下的信息 :其指示如何使用恶意软件的元数据字段属性来确定未知的可执行文件是否包含恶意软件。根据某些实施例，恶意软件元数据字段信息数据库 124 可以包括从将任何适当的机器学习算法应用于已知洁净和已知恶意的可执行文件推导出的任何信息。例如，恶意软件元数据字

段信息数据库 124 可以包括用于一个或多个元数据字段属性的权重信息，其指示元数据属性在检测恶意软件中的有用性。

[0019] 在一些实施例中，恶意软件元数据字段信息数据库 124 可以包括指示阈值数的阈值信息。如果可执行文件的元数据字段属性与恶意软件元数据字段信息数据库 124 中的元数据字段属性匹配的数量大于或等于阈值数，则安全模块 116 可以确定可执行文件包括恶意软件。恶意软件元数据字段信息数据库 124 可以附加地或替选地包括以下信息，该信息指示对恶意软件进行指示的元数据字段属性的一个或多个组合。在一些实施例中，恶意软件元数据字段信息数据库 124 可以包括以下信息，该信息指示对恶意软件进行指示的一个元数据字段属性是比指示恶意软件的其他元数据字段属性更强的恶意软件指示符。

[0020] 图 1 中的数据库 120 中的一个或多个可以代表一个或多个计算设备的一部分。数据库 120 中的一个或多个可以代表图 5 中的计算系统 510 的一部分，和 / 或图 6 中的示例性网络架构 600 的一部分。替选地，图 1 中的数据库 120 中的一个或多个可以代表一个或多个物理上分离的能够由计算设备访问的设备。

[0021] 图 2 示出可以由诸如检查模块 112 的检查模块、诸如推导模块 114 的推导模块、和 / 或诸如安全模块 116 的安全模块实现的过程。检查模块可以对多个已知洁净的可执行文件的多个元数据字段进行检查（步骤 210）。例如，检查模块可以对两个或更多个已知洁净的可执行文件的两个或更多个元数据字段进行检查。

[0022] 已知洁净的可执行文件可以包括已经被识别为是洁净的任何文件（即不包括恶意软件的文件）。可执行文件的元数据字段可以是携带关于可执行文件和 / 或与其相关联的信息的任何字段。例如，可执行文件的元数据字段中的条目可以解释该可执行文件的属性。

[0023] 在一些实施例中，元数据字段可以包括用于可执行文件的静态属性。如此处所用，短语“静态属性”可以指：当文件没有正在被执行时，可以观察到的文件的任何属性。换言之，静态属性可以是基于对可执行文件中的信息进行检查所能够确定的属性。相反，动态属性可以是基于对文件的执行而观察到的属性。

[0024] 可执行文件可以是包括可由计算机执行的代码（即指令）的任何文件。还可以将可执行文件称为可执行或二进制。可以根据任何适当的可执行文件格式对可执行文件进行格式化。可执行文件格式的一个示例是可移植可执行 (PE) 文件格式。PE 文件格式可以是在 32 位或 64 位版本的视窗操作系统中使用的文件格式。可执行文件还可以包括用于 LINUX 操作系统、MAC 操作系统、UNIX 操作系统、和 / 或任何其他操作系统的可执行文件。

[0025] 除了对已知洁净的可执行文件进行检查以外，检查模块可以对多个已知恶意的可执行文件的多个元数据字段进行检查（步骤 220）。例如，检查模块可以对两个或更多个已知恶意的可执行文件的两个或更多个元数据字段进行检查。已知恶意的可执行文件可以包括已知包括诸如病毒、特洛伊木马、蠕虫和 / 或意图损害计算机系统和数据的其他程序之类的恶意软件的可执行文件。在一些实施例之中，检查模块可以对在已知洁净的可执行文件和已知恶意的可执行文件中的相同元数据字段进行检查。在其他实施例中，检查模块可以对在已知洁净的可执行文件中与在已知恶意的可执行文件中不同的一个或多个元数据字段进行检查。

[0026] 检查模块可以通过确定可执行文件的元数据字段中的属性值，来对可执行文件的

元数据字段进行检查。检查模块可以将在检查期间收集的信息（例如，属性值）存储在数据库中，诸如恶意软件元数据字段信息数据库 124。

[0027] 在检查模块对已知洁净和已知恶意的可执行文件进行检查之后，推导模块可以基于在检查期间获取的信息推导指示恶意软件的元数据字段属性。如前所述，在检查期间获取的信息可以包括属性信息。推导模块可以对属性信息进行处理，以确定哪些属性可以指示恶意软件。推导模块可以使用任何适当的机器学习算法来确定哪些属性可以指示恶意软件。推导模块还可以使用任何适当的机器学习算法确定如何使用元数据属性来确定未知的可执行文件是否包含恶意软件。例如，推导模块可以确定元数据属性的哪些组合是恶意可执行文件的指示。推导模块可以将关于如何使用元数据属性来识别恶意软件的任何信息存储于数据库中，诸如恶意软件元数据字段信息数据库 124。

[0028] 可执行文件可以包括各种类型的元数据字段和对应的属性。例如，可执行文件可以包括：用于一个或多个调试段属性的一个或多个调试段字段、用于一个或多个导入属性的一个或多个导入字段、用于一个或多个符号表属性的一个或多个符号表字段、用于一个或多个可选头属性的一个或多个可选头字段、用于一个或多个特征属性的一个或多个特征字段、用于一个或多个图像子系统属性的一个或多个图像子系统字段、用于一个或多个图像基础属性的一个多个图像基础字段、用于一个或多个链接器版本属性的一个或多个链接器版本字段、用于一个或多个大小属性的一个或多个大小字段和 / 或用于一个或多个真实虚拟地址 (RVA) 属性的一个或多个 RVA 字段。

[0029] 可执行文件还可以包括：用于一个或多个入口点属性的一个或多个入口点字段、用于一个或多个代码段基础属性的一个或多个代码段基础字段、用于一个或多个无进出属性的一个或多个无进出字段、用于一个或多个线程级前瞻 (TLS) 属性的一个或多个 TLS 字段、用于一个或多个具有证书属性的一个或多个具有证书字段、用于一个或多个节点属性的一个或多个节点字段、用于一个或多个对准属性的一个或多个对准字段、用于一个或多个操作系统版本属性的一个或多个操作系统版本字段、用于一个或多个图像版本属性的一个或多个图像版本字段、用于一个或多个最低子系统版本属性的一个或多个最低子系统版本字段、用于一个或多个动态链接库 (DLL) 特征属性的一个或多个 DLL 特征字段、用于一个或多个外部绑定设施 (EBF) 属性的一个或多个 EBF 字段、用于一个或多个堆栈大小属性的一个或多个堆栈大小字段、用于一个或多个堆大小属性的一个或多个堆大小字段、和 / 或用于一个或多个段数属性的一个或多个段数字段。

[0030] 可以由任何适当的数据类型来表示此处描述的属性。例如，调试段属性可以包括调试段整数，符号表属性可以包括符号表指针整数和 / 或符号数整数，图像子系统属性可以包括图形用户界面整数和 / 或基于字符的用户界面整数，而链接器版本属性可以包括主要链接器版本整数和 / 或次要链接器版本整数。RVA 属性可以包括：RVA 入口点整数、代码段的 RVA 开始整数、代码段的 RVA 基础整数、和 / 或数据段的 RVA 开始整数。

[0031] 操作系统版本属性可以包括主要操作系统版本整数和 / 或次要操作系统版本整数。图像版本属性可以包括主要图像版本整数和 / 或次要图像版本整数。最低子系统版本属性可以包括主要子系统版本整数和 / 或次要子系统版本属性。大小属性可以包括：图像大小整数、代码段大小整数、已初始化数据大小整数、未初始化数据大小整数、和 / 或头大小整数。可选头属性可以包括在可执行文件的可选头中的任何属性。

[0032] 入口点属性可以包括入口点整数。代码段基础属性可以包括代码段基础整数。对准属性可以包括段对准整数和/或文件对准整数。DLL特征属性可以包括DLL特征整数，堆栈大小属性可以包括堆栈保留大小整数和/或堆栈提交大小整数，而堆大小属性可以包括堆保留大小整数和/或堆提交大小整数。图像基础属性可以包括图像基础整数，而EBF属性可以包括EBF整数。无进出属性可以包括无进出整数，具有证书属性可以包括具有证书整数，段数属性可以包括段数整数，节点属性可以包括节点整数，而TLS属性可以包括TLS整数。

[0033] 导入属性可以包括：具有导入整数、具有延迟的导入整数、urlmon导入整数、msvcrt导入整数、oleaut32导入整数、setupapi导入整数、user32导入整数、advapi32导入整数、shell32导入整数、gdi32导入整数、comdlg32导入整数、和/或imm32导入整数。特征属性可以包括用于各种可执行文件特征的一个或多个特征标记。

[0034] 如上所述，元数据字段属性可以表示可执行文件的各种特征。例如，多个段对准整数可以用于指示段需要被加载的位置。多个文件对准整数可以是用于开始段的偏移量。主要和次要操作系统版本整数可以指示需要执行可执行文件的最低操作系统版本。主要和次要图像版本整数可以指示可执行文件的版本。主要和次要子系统版本整数可以指示执行可执行文件需要的最低子系统版本。图像大小整数可以指示考虑段对准之后的图像大小。头大小整数可以指示可执行文件的头的全部大小。图形用户界面整数可以包括指示可执行文件是否使用图形用户界面的标记。基于字符的用户界面整数可以包括指示可执行文件是否使用基于字符的用户界面的标记。

[0035] 堆栈保留大小整数可以指示可能需要为堆栈保留的地址空间的数量。堆栈提交大小整数可以指示为堆栈提交的实际存储器的数量。堆保留大小整数可以指示可能需要为堆保留的地址空间的数量。堆提交大小整数可以指示为堆提交的实际存储器的数量。符号表指针整数可以是到符号表的偏移量。符号数整数可以指示符号表中的符号数。调试段整数可以指示可执行文件是否具有调试段。主要和次要链接器版本整数可以指示产生可执行文件的链接器的版本。代码段大小整数可以指示可执行文件中的代码段的大小。已初始化数据大小整数可以指示可执行文件中的已初始化数据段的大小。未初始化数据大小整数可以指示可执行文件中的未初始化数据段的大小。

[0036] urlmon导入整数可以指示可执行文件是否链接到urlmon.dll文件。msvcrt导入整数可以指示可执行文件是否链接到msvcrt.dll文件。oleaut32导入整数可以指示可执行文件是否链接到oleaut32.dll文件。setupapi导入整数可以指示可执行文件是否链接到setupapi.dll文件。user32导入整数可以指示可执行文件是否链接到user32-import.dll文件。advapi32导入整数可以指示可执行文件是否链接到advapi.exe文件。shell32导入整数可以指示可执行文件是否链接到shell32-imports.dll文件。gdi32导入整数可以指示可执行文件是否链接到gdi32.dll文件。comdlg32导入整数可以指示可执行文件是否链接到comdlg32.dll文件。imm32导入整数可以指示可执行文件是否链接到imm32.dll文件。

[0037] 如前所述，可执行文件可以为PE文件格式。图3示出PE文件格式的可执行文件300的示例。如图3所示，可执行文件300可以包括盘操作系统(DOS)驻留程序(stub)310。可执行文件300还可以包括文件头320。文件头320可以包括一个或多个元数据字段。例

如，文件头 320 可以包括 :指示意图在其上运行二进制的系统的机器字段、段数字段、时间戳字段、符号表指针字段、符号数字段、调试信息字段、可选头大小字段、图像文件重分配剥离字段、图像文件可执行图像字段、图像文件线数剥离字段、文件本地符号剥离字段、图像文件积极工作集调整 (image-file-aggressive-working-set-trim) 字段、图像文件字节保留低字段、图像文件 32 位机器字段、图像文件调试剥离字段、从交换运行的图像文件可去除字段、从交换运行的图像文件网字段、图像文件系统字段、图像文件动态链接库 (DLL) 字段、和 / 或仅有图像文件出现的系统 (image-file-up-system-only) 字段。

[0038] 可执行文件 300 还可以包括可选头 330。可选头 330 可以包括一个或多个元数据字段。例如，可选头 330 可以包括 :主要链接器版本字段、次要链接器版本字段、代码大小字段、已初始化数据大小字段、未初始化数据大小字段、到代码入口点的偏移量字段、入口点地址字段、到代码基础的偏移量字段、数据基础字段、段对准字段、文件对准字段、主要操作系统版本字段、次要操作系统版本字段、主要图像版本字段、次要图像版本字段、主要子系统版本字段、次要子系统版本字段、32 位视窗图形用户界面 (GUI) 应用字段、32 位视窗版本值字段、图像大小字段、头大小字段、校验和字段、和本原图像子系统字段。

[0039] 可选头 330 还可以包括 :图像子系统视窗 GUI 字段、图像子系统视窗字符用户界面 (CUI) 字段、图像子系统 OS/2-CUI 字段、图像子系统 POSIX-CUI 字段、DLL 特征字段、进程附接字段、线程分离字段、线程附接字段、进程分离字段、堆栈保留大小字段、堆栈提交大小字段、堆保留大小字段、堆提交大小字段、负载标记、和图像目录标记。

[0040] 可执行文件 300 还可以包括数据目录 340 和段头 350。段头 350 可以包括一个或多个元数据字段。例如，段头 350 可以包括 :图像大小缩短字段、图像段头字段、虚拟地址字段、原始数据大小字段、到原始数据的指针字段、到重分配的指针字段、和特征字段，其可以包括一个或多个指示可执行文件 300 的一个或多个属性的标记。可执行文件 300 还可以包括 :段 1 360(1) 到段 N 360(n)。

[0041] 图 4 示出用于在可移植可执行文件中检测恶意软件的方法。检查模块可以对多个已知洁净的可移植可执行文件的多个元数据字段进行检查 (步骤 410)。检查模块还可以对多个已知恶意的可移植可执行文件的多个元数据字段进行检查 (步骤 420)。检查模块可以对任意数量的已知洁净的可移植可执行文件和 / 或已知恶意的可移植可执行文件进行检查。例如，检查模块可以对数十、数百、数千、数十万、和 / 或数百万个可执行文件进行检查。检查模块还可以对在已知洁净的可移植可执行文件和 / 或已知恶意的可移植可执行文件中的任意数量元数据字段进行检查。在可执行文件已被检查之后，推导模块可以基于该检查推导指示恶意软件的元数据字段属性 (步骤 430)。

[0042] 安全模块可以使用由推导模块推导出的信息来确定未知的可执行文件是否包含恶意软件。例如，安全模块可以接收未知的可执行文件 (步骤 440)。安全模块可以通过确定未知的可执行文件是否包含先前已识别的、指示恶意软件的元数据字段属性，来确定该未知的可执行文件是否包含恶意软件。

[0043] 在一些实施例中，安全模块可以包括防病毒安全软件程序或是其一部分。根据至少一个实施例，客户端计算设备可以包括安全模块，并且该安全模块可以通过确定客户端计算设备上的未知文件是否包括恶意软件来对客户端计算设备进行保护。安全模块还可以确定即将下载到客户端计算设备的文件是否包括恶意软件。在其他实施例中，服务器或任

何其他计算设备可以包括安全模块。

[0044] 在一个示例中，检查模块检查了 850,000 个已知洁净的可执行文件和 500,000 个已知恶意的可执行文件。推导模块对在检查期间收集的信息进行处理，以确定指示恶意软件的头字段属性。基于推导模块推导出的信息，安全模块可以将大约 50％ -60％的恶意可执行文件识别为恶意，而返回低于 0.5％的误肯定 (false-positive) 确定。本领域的普通技术人员不会期望这样的结果。

[0045] 图 5 是能够实现此处描述和 / 或说明的一个或多个实施例的示例性计算系统 510 的框图。计算系统 510 宽泛地表示能够执行计算机可读指令的任何单或多处理器计算设备或系统。计算系统 510 的示例包括但不限于 :工作站、笔记本电脑、客户端侧终端、服务器、分布式计算系统、手持设备、或任何其他的计算系统或设备。在其最基本的配置中，计算系统 510 可以包括至少一个处理器 514 和系统存储器 516。

[0046] 处理器 514 一般代表任何类型或形式的能够处理数据或解释和执行指令的处理单元。在某些实施例中，处理器 514 可以从软件应用或模块接收指令。这些指令可以使得处理器 514 来执行此处描述和 / 或说明的一个或多个示例性实施例的功能。例如，处理器 514 可以单独或与其他元件组合来执行检查、推导、接收、确定中的一个或多个和 / 或执行此处描述的步骤和 / 或作为执行检查、推导、接收、确定中的一个或多个和 / 或执行此处描述的步骤的装置。处理器 514 还可以执行此处描述和 / 或说明的任何其他步骤、方法、或过程和 / 或作为执行此处描述和 / 或说明的任何其他步骤、方法、或过程的装置。

[0047] 系统存储器 516 一般代表任何类型或形式的能够存储数据和 / 或其他计算机可读指令的易失或非易失存储设备或介质。系统存储器 516 的示例包括但不限于 :随机存取存储器 (RAM)、只读存储器 (ROM)、闪存、或任何其他适当的存储器设备。尽管不是必要，但在某些实施例中，计算系统 510 既可以包括易失存储器单元 (例如，系统存储器 516)，也可以包括非易失存储设备 (例如，如以下将要详细描述的主存储设备 532)。

[0048] 在某些实施例中，示例性计算系统 510 还可以包括除处理器 514 和系统存储器 516 之外的一个或多个组件或元件。例如，如图 5 所说明的，计算系统 510 可以包括 :存储控制器 518、输入 / 输出 (I/O) 控制器 520、和通信接口 522，其均可以经由通信基础设施 512 互连。通信基础设施 512 一般代表任何类型或形式的能够促成在计算设备的一个或多个组件之间通信的基础结构。通信基础设施 512 的示例包括但不限于通信总线 (诸如 ISA、PCI、PCIe 或类似总线) 和网络。

[0049] 存储控制器 518 一般代表任何类型或形式的能够处理存储器或数据或控制在计算系统 510 的一个或多个组件之间的通信的设备。例如，在某些实施例中，存储控制器 518 可以经由通信基础设施 512 控制处理器 514、系统存储器 516、和 I/O 控制器 520 之间的通信。在某些实施例中，存储器控制器可以单独或与其他元件来执行此处描述和 / 或说明的一个或多个步骤或特征，诸如检查、推导、接收、确定和 / 或执行，和 / 或作为执行此处描述和 / 或说明的一个或多个步骤或特征的装置。

[0050] I/O 控制器 520 一般代表任何类型或形式的能够协调和 / 或控制计算设备的输入和输出功能的模块。例如，在某些实施例中，I/O 控制器可以控制或促进在计算系统 510 的一个或多个元件之间的数据传递，所述元件诸如处理器 514、系统存储器 516、通信接口 522、显示适配器 526、输入接口 530、和存储器接口 534。例如，I/O 控制器 520 可以用来单

独或与其他元件组合来执行检查、推导、接收、确定中的一个或多个和/或执行此处描述的步骤，和/或作为执行检查、推导、接收、确定中的一个或多个和/或执行此处描述的步骤的装置。I/O 控制器 520 还可以用于执行本公开中阐述的其他步骤和特征和/或作为执行本公开中阐述的其他步骤和特征的装置。

[0051] 通信接口 522 宽泛地表示任何形式或类型的能够促成在示例性计算系统 510 和一个或多个附加设备之间的通信的通信设备或适配器。例如，在某些实施例中，通信接口 522 可以促成在计算系统 510 和包括附加计算系统的专用或公共网络之间的通信。通信接口 522 的示例包括但不限于：有线网络接口（诸如网络接口卡）、无线网络接口（诸如无线网络接口卡）、调制解调器、和任何其他适当的接口。在至少一个实施例中，通信接口 522 可以经由到诸如因特网的网络的直接链路提供到远程服务器的直接连接。通信接口 522 还可以通过例如局域网（诸如以太网）、个人区域网、电话或有线电视网、蜂窝电话连接、卫星数据连接、或任何其他适当的连接，间接提供这样的连接。

[0052] 在某些实施例中，通信接口 522 还可以表示主机适配器，其被配置为经由外部总线或通信信道促成在计算系统 510 和一个或多个附加网络或存储设备之间的通信。主机适配器的示例包括但不限于：SCSI 主机适配器、USB 主机适配器、IEEE 594 主机适配器、SATA 和 eSATA 主机适配器、ATA 和 PATA 主机适配器、光纤信道接口适配器、以太网适配器等。通信接口 522 还可以允许计算系统 510 进行分布式或远程计算。例如，通信接口 522 可以从远程设备接收指令，或将指令发送给远程设备来执行。在某些实施例中，通信接口 522 可以单独或与其他元件组合来执行检查、推导、接收、确定中的一个或多个和/或执行此处公开的步骤，和/或作为执行检查、推导、接收、确定中的一个或多个和/或执行此处公开的步骤的装置。通信接口 522 还可以用于执行本公开中阐述的其他步骤和特征和/或作为执行本公开中阐述的其他步骤和特征的装置。

[0053] 如图 5 说明的，计算系统 510 还可以包括：经由显示适配器 526 连接到通信基础设施 512 的至少一个显示设备 524。显示设备 524 一般代表任何类型或形式的能够视觉显示由显示适配器 526 转发的信息的设备。类似地，显示适配器 526 一般代表任何类型或形式的被配置为转发来自通信基础设施 512（或如本领域所知的，来自帧缓存）的图形、文本、和其他数据以用于在显示设备 524 上显示的设备。

[0054] 如图 5 说明的，示例性计算系统 510 还可以包括：经由输入接口 530 连接到通信基础设施 512 的至少一个输入设备 528。输入设备 528 一般代表任何类型或形式的能够将计算机或人工生成的输入提供给示例性计算系统 510 的输入设备。输入设备 528 的示例包括但不限于：键盘、指示设备、语音识别设备、或任何其他输入设备。在至少一个实施例中，输入设备 528 可以单独或与其他元件组合来执行检查、推导、接收、确定中的一个或多个和/或执行此处公开的步骤和/或作为执行检查、推导、接收、确定中的一个或多个和/或执行此处公开的步骤的装置。输入设备 528 还可以用于执行本公开中阐述的其他步骤和特征和/或作为执行本公开中阐述的其他步骤和特征的装置。

[0055] 如图 5 所说明的，示例性计算系统 510 还可以包括：经由存储接口 534 连接到通信基础设施 512 的主存储设备 532 和备份存储设备 533。存储设备 532 和 533 一般代表任何类型或形式的能够存储数据和/或其他计算机可读指令的存储设备或介质。例如，存储设备 532 和 533 可以是磁盘驱动（例如，所谓的硬盘驱动）、软盘驱动、磁带驱动、光盘驱动、闪

存驱动等。存储接口 534 一般代表任何类型或形式的用于在存储设备 532 和 533 以及计算系统 510 的其他组件之间传递数据的接口或设备。

[0056] 在某些实施例中，可以将存储设备 532 和 533 配置为：从被配置为存储计算机软件、数据、或其他计算机可读信息的可移动存储单元进行读取或对其进行写入。适当可移动存储单元的示例包括但不限于：软盘、磁带、光盘、闪存设备等。存储设备 532 和 533 还可以包括其他类似的结构或设备，用于允许将计算机软件、数据、或其他计算机可读指令加载到计算系统 510 中。例如，可以将存储设备 532 和 533 配置为对软件、数据、或其他计算机可读信息进行读写。存储设备 532 和 533 还可以是计算系统 510 的一部分，或可以是经由其他接口系统访问的分立设备。

[0057] 在某些实施例中，可以将此处公开的示例性文件系统存储在主存储设备 532 上，而将此处公开的示例性文件系统备份存储在备份存储设备 533 上。例如，存储设备 532 和 533 还可以单独或与其他元件组合来执行检查、推导、接收、确定中的一个或多个和 / 或执行此处公开的步骤和 / 或作为执行检查、推导、接收、确定中的一个或多个和 / 或执行此处公开的步骤的装置。存储设备 532 和 533 还可以用于执行本公开中阐述的其他步骤和特征和 / 或作为执行本公开中阐述的其他步骤和特征的装置。

[0058] 很多其他设备或子系统可以连接到计算系统 510。相反，为实践此处描述和 / 或说明的实施例，图 5 中说明的组件和设备不必都呈现。也可以用与图 5 所示不同的方式将以上引用的设备和子系统进行互连。计算系统 510 还可以采用任意数量的软件、固件、和 / 或硬件配置。例如，可以将此处公开的一个或多个示例性实施例作为计算机程序（也称为计算机软件、软件应用、计算机可读指令、或计算机控制逻辑）编码于计算机可读介质上。短语“计算机可读介质”一般指任何形式的能够存储或承载计算机可读指令的设备、载体、或介质。计算机可读介质的示例包括但不限于：诸如载波的传输型介质，以及物理介质，诸如磁存储介质（例如，硬盘驱动和软盘）、光存储介质（例如，CD 或 DVD-ROM）、电存储介质（例如，固态驱动和闪存介质）、以及其他分发系统。

[0059] 包含计算机程序的计算机可读介质可以加载到计算系统 510 中。然后，可以将存储在计算机可读介质上的全部或部分计算机程序存储于系统存储器 516 和 / 或存储设备 532 和 533 的各部分中。当由处理器 514 执行时，加载到计算系统 510 中的计算机程序可以使得处理器 514 执行此处描述和 / 或说明的一个或多个示例性实施例的功能和 / 或作为执行此处描述和 / 或说明的一个或多个示例性实施例的功能的装置。附加地或替选地，可以用固件和 / 或硬件实现此处描述和 / 或说明的一个或多个示例性实施例。例如，可以将计算系统 510 配置为适合实现此处公开的一个或多个示例性实施例的专用集成电路 (ASIC)。

[0060] 图 6 是示例性网络架构 600 的框图，其中客户端系统 610、620、和 630 和服务器 640 和 645 可以连接到网络 650。客户端系统 610、620、和 630 一般代表任何类型或形式的计算设备或系统，诸如图 5 中的示例性计算系统 510。类似地，服务器 640 和 645 一般代表被配置为提供各种数据库服务和 / 或运行某些软件应用的计算设备或系统，诸如应用服务器或数据库服务器。网络 650 一般代表任何电信或计算机网络，例如包括：企业内联网、广域网 (WAN)、局域网 (LAN)、个人区域网 (PAN)、或因特网。

[0061] 如图 6 所说明的，一个或多个存储设备 660(1)-(N) 可以直接附接到服务器 640。类似地，一个或多个存储设备 670(1)-(N) 可以直接附接到服务器 645。存储设备 660(1)-(N)

和存储设备 670(1)-(N) 一般代表任何类型或形式的能够存储数据和 / 或其他计算机可读指令的存储设备或介质。在某些实施例中，存储设备 660(1)-(N) 和存储设备 670(1)-(N) 可以代表被配置为使用诸如 NFS、SMB、或 CIFS 的各种协议与服务器 640 和 645 进行通信的网络附接存储 (NAS) 设备。

[0062] 服务器 640 和 645 还可以连接到存储区域网络 (SAN) 构造 680。SAN 构造 680 一般代表任何类型或形式的能够促成在多个存储设备之间通信的计算机网络或结构。SAN 构造 680 可以促成服务器 640 和 645 与多个存储设备 690(1)-(N) 和 / 或智能存储阵列 695 之间的通信。SAN 构造 680 还可以用诸如设备 690(1)-(N) 和阵列 695 表现为到客户端系统 610、620 和 630 的本地附接设备的方式经由网络 650 和服务器 640 和 645 促成在客户端系统 610、620 和 630 以及存储设备 690(1)-(N) 和 / 或智能存储阵列 695 之间的通信。与存储设备 660(1)-(N) 和存储设备 670(1)-(N) 一样，存储设备 690(1)-(N) 和智能存储阵列 695 一般代表任何类型或形式的能够存储数据和 / 或其他计算机可读指令的存储设备或介质。

[0063] 在某些实施例中，并参照图 5 的示例性计算系统 510，诸如图 5 中通信接口 522 的通信接口可以用于提供每个客户端系统 610、620、和 630 和网络 650 之间的互联性。客户端系统 610、620、和 630 可以能够使用例如 web 浏览器或其他客户端软件来访问服务器 640 或 645 上的信息。这样的软件可以允许客户端系统 610、620、和 630 访问服务器 640、服务器 645、存储设备 660(1)-(N)、存储设备 670(1)-(N)、存储设备 690(1)-(N)、或智能存储阵列 695 所托管的数据。尽管图 6 描述了使用网络（诸如因特网）进行交换数据，但是此处描述和 / 或说明的实施例不限于因特网或任何特定的基于网络的环境。

[0064] 在至少一个实施例中，可以将此处公开的一个或多个示例性实施例的全部或部分编码为计算机程序，并加载到服务器 640、服务器 645、存储设备 660(1)-(N)、存储设备 670(1)-(N)、存储设备 690(1)-(N)、智能存储阵列 695、或其任何组合上并由其执行。可以将此处公开的一个或多个示例性实施例的全部或部分编码为计算机程序，存储在服务器 640 中，由服务器 645 来运行，并通过网络 650 分布到客户端系统 610、620、和 630。因此，网络架构 600 可以单独或与其他元件结合来执行检查、推导、接收、确定中的一个或多个和 / 或执行此处公开的步骤和 / 或作为执行检查、推导、接收、确定中的一个或多个和 / 或执行此处公开的步骤的装置。网络架构 600 还可以用于执行本公开中阐述的其他步骤和特征和 / 或作为执行本公开中阐述的其他步骤和特征的装置。

[0065] 如以上详细描述的，计算系统 410 和 / 或网络结构 500 的一个或多个组件可以单独或与其他元件组合来执行此处描述和 / 或说明的示例性方法的一个或多个步骤和 / 或作为执行此处描述和 / 或说明的示例性方法的一个或多个步骤的装置。例如，计算系统可以对多个已知洁净的可执行文件的多个元数据字段进行检查。计算系统还可以对多个已知恶意的可执行文件的多个元数据字段进行检查。计算系统可以基于从对多个已知洁净和已知恶意的可执行文件的多个元数据字段进行检查中获取的信息，推导出指示恶意软件的元数据字段属性。

[0066] 在一些实施例中，计算系统可以接收未知的可执行文件。计算系统可以通过确定该未知的可执行文件是否包含指示恶意软件的元数据字段属性，来确定该未知的可执行文件是否包含恶意软件。在至少一个实施例中，如果未知的可执行文件包括恶意软件，则计算

系统可以执行安全动作。

[0067] 根据各种实施例，多个已知洁净的可执行文件可以包括可移植可执行文件，并且多个已知恶意的可执行文件可以包括可移植可执行文件。在至少一个实施例中，元数据字段属性可以包括头字段属性。根据某些实施例，头字段属性包括以下中的至少一个：调试段属性、导入属性、符号表属性、可选头属性、特征属性、图像子系统属性、链接器版本属性、大小属性、真实虚拟地址属性、入口点属性、代码段基础属性、对准属性、操作系统版本属性、图像版本属性、最低子系统版本属性、动态链接库特征属性、堆栈大小属性、堆大小属性、段数属性、无进出属性、线程级前瞻属性、和 / 或图像基础属性。

[0068] 在一些实施例中，指示恶意软件的元数据字段属性包括在多个已知洁净的可执行文件和已知恶意的可执行文件中检查到的多个元数据字段的子集。在至少一个实施例中，指示恶意软件的元数据字段属性可以包括静态属性。在各种实施例中，推导指示恶意软件的元数据字段属性包括确定指示恶意软件的元数据字段属性的至少一个组合。根据至少一个实施例，推导指示恶意软件的元数据字段属性包括确定指示恶意软件的第一元数据字段属性是比指示恶意软件的第二元数据字段属性更强的恶意软件指示符。

[0069] 在一些实施例中，指示恶意软件的元数据字段属性包括以下中的至少一个：调试段属性、符号表属性、图像子系统属性、链接器版本属性、真实虚拟地址属性、操作系统版本属性、图像版本属性、最低子系统属性、动态链接库特征属性、堆栈大小属性、堆大小属性、和 / 或段数属性。

[0070] 在一些实施例中，指示恶意软件的元数据字段属性包括以下中的至少两个：段对准整数、文件对准整数、主要操作系统版本整数、次要操作系统版本整数、主要图像版本整数、次要图像版本整数、主要子系统版本整数、次要子系统版本整数、图像大小整数、头大小整数、图形用户界面整数、基于字符的用户界面整数、堆栈保留大小整数、堆栈提交大小整数、堆保留大小整数、堆提交大小整数、符号表指针整数、符号数整数、调试段整数、主要链接器版本整数、次要链接器版本整数、代码段大小整数、已初始化数据大小整数、未初始化数据大小整数、真实虚拟地址 (RVA) 入口点整数、代码段的 RVA 开始整数、代码段的 RVA 基础整数、数据段的 RVA 开始整数、图像基础整数、具有导入整数、具有延迟的导入整数、外部绑定设施整数、无进出整数、urlmon 导入整数、线程级前瞻整数、msvcrt 导入整数、oleaut32 导入整数、setupapi 导入整数、user32 导入整数、advapi32 导入整数、shell32 导入整数、gdi32 导入整数、comdlg32 导入整数、imm32 导入整数、具有证书整数、节点整数、和 / 或段数整数。

[0071] 根据某些实施例，系统可以包括：被编程为对多个已知洁净的可执行文件的多个头字段进行检查和对多个已知恶意的可执行文件的多个头字段进行检查的检查模块。系统还可以包括：被配置为存储由检查模块获取的信息的数据库，以及被编程为基于从检查模块获取的信息推导出指示恶意软件的头字段属性的推导模块。

[0072] 在某些实施例中，系统可以包括安全系统，其被编程为接收未知的可执行文件，和 / 或通过确定该未知可执行文件是否包括指示恶意软件的头字段属性来确定该未知的可执行文件是否包含恶意软件。在至少一个实施例中，安全模块可以进一步被编程为：如果该未知的可执行文件包括恶意软件，则执行安全动作。

[0073] 尽管以上公开使用了特定的框图、流程图和示例阐述了各种实施例，但是可以使

用很宽范围的硬件、软件、或固件（或其任何组合）配置单独和/或共同地实现此处描述和/或说明的每个框图组件、流程图步骤、操作、和/或组件。另外，应当将对包含在其他组件内的组件的任何公开认为本质上是示例性的，因为可以实施很多其他结构来实现同样的功能。

[0074] 此处描述和/或说明的过程参数和步骤的顺序仅是示例性的，并且可以根据需要变化。例如，尽管可以用特定顺序显示或讨论此处说明和/或描述的步骤，但是不必用已说明或讨论的顺序来执行这些步骤。此处描述和/或说明的各种示例性方法还可以省略此处描述或说明的一个或多个步骤，或包括公开内容以外的附加步骤。

[0075] 另外，尽管已在全功能计算系统的背景下描述和/或说明了各种实施例，但是可以将这些示例性实施例中的一个或多个分布为各种形式的程序产品，而不论实际用于执行该分布的特定类型的计算机可读介质。还可以使用执行特定任务的软件模块实现此处公开的实施例。这些软件模块可以包括：可以存储在计算机可读存储介质上或计算系统中的脚本、批处理、或其他可执行文件。在一些实施例中，这些软件模块可以对计算系统进行配置，以执行此处公开的一个或多个示例性实施例。

[0076] 已提供了以上描述，以使得本领域的技术人员最佳地利用此处描述的示例性实施例的各个方面。该示例性描述不是意图穷尽或受限于公开的任何特定形式。在不偏离本公开的精神和范围的前提下，很多修改和变化都是可能的。期望此处描述的实施例在所有方面被认为是说明性而非限制性的，并且参考所附权利要求及其等效来确定本公开的范围。

[0077] 除非另外注明，否则如在说明书和权利要求中使用的，不加数量词限定的项应被解释为意指“至少一个”项。另外，为了使用上的简便，如在说明和权利要求中使用的词“包括”和“具有”可互换，并具有与词“包含”相同的含义。

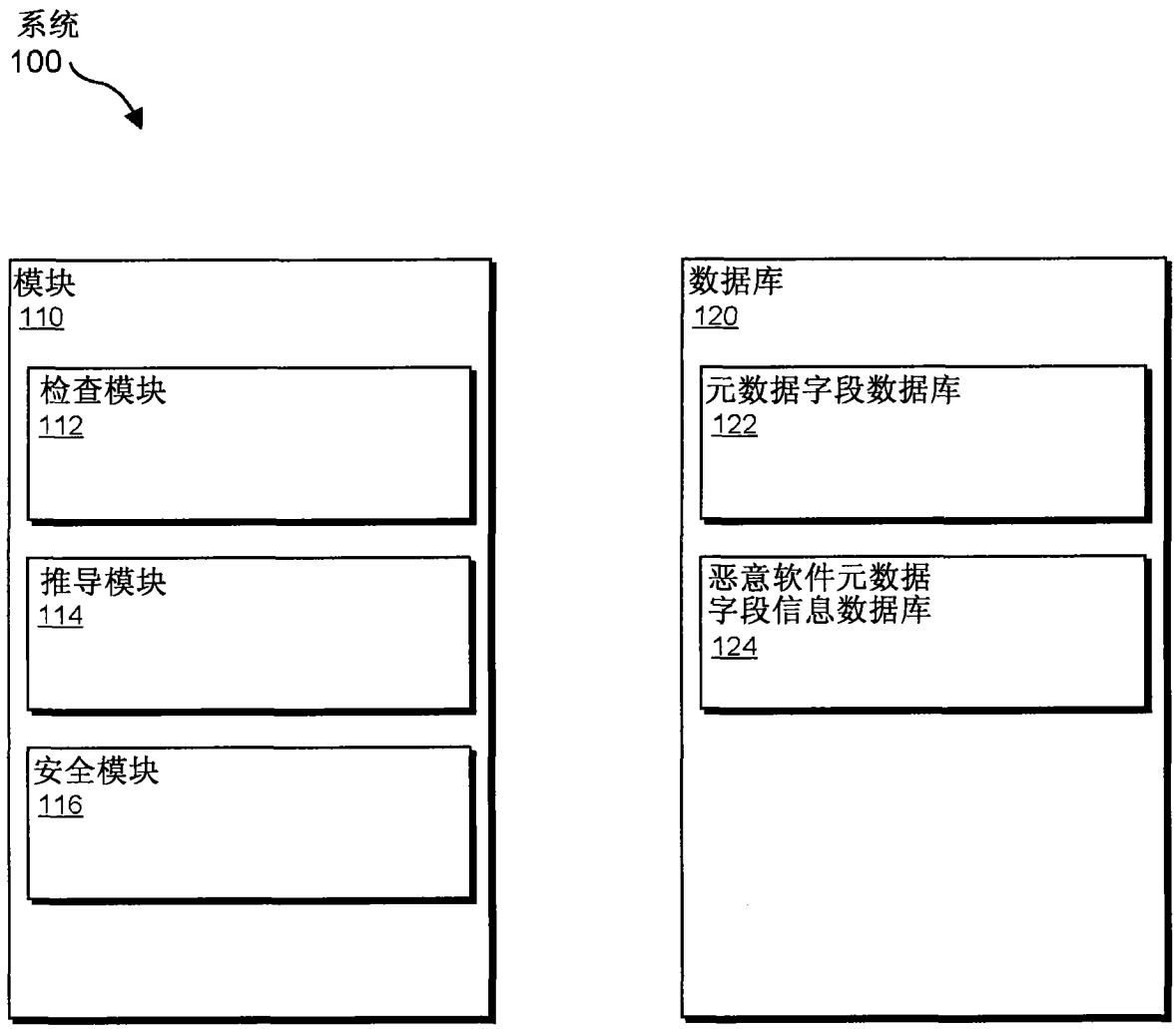

系统
100
模块
110
检查模块
112
推导模块
114
安全模块
116
数据库
120
元数据字段数据库
122
恶意软件元数据
字段信息数据库
124

图 1

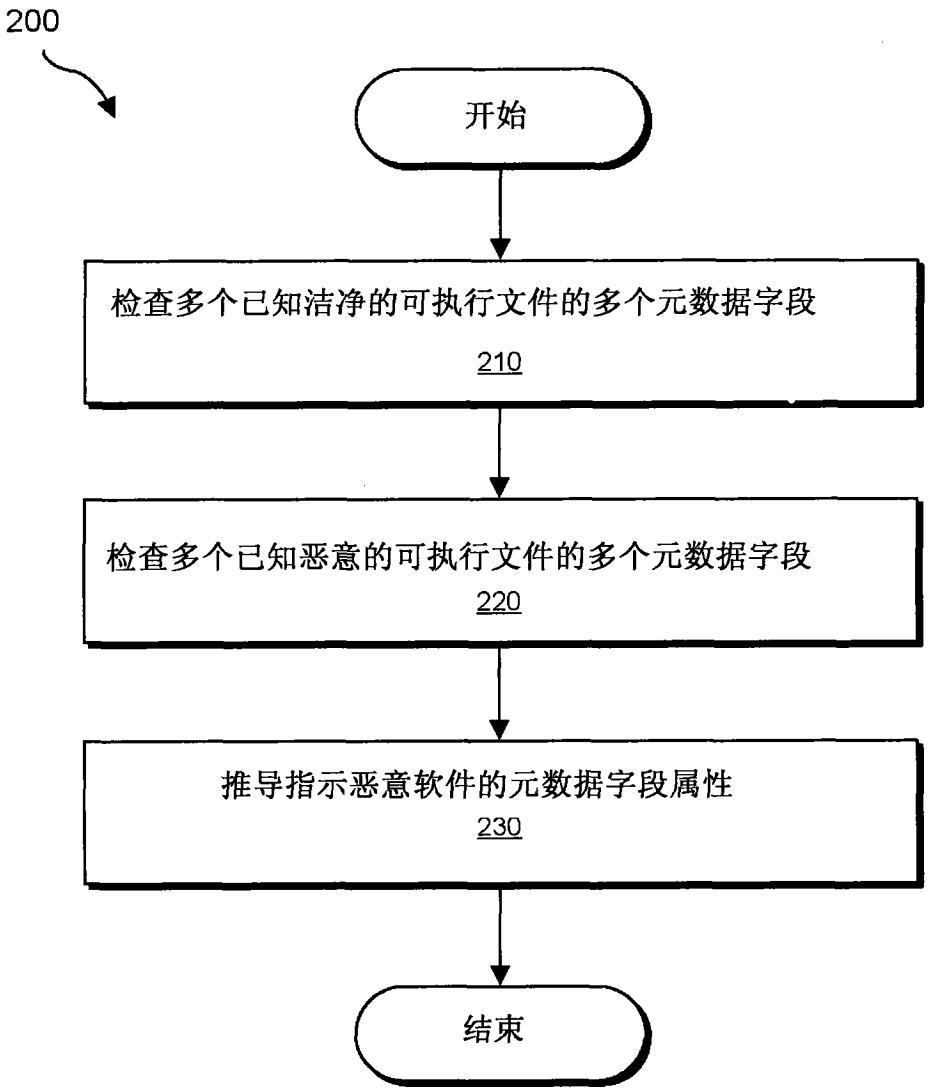

200
开始
检查多个已知洁净的可执行文件的多个元数据字段
210
检查多个已知恶意的可执行文件的多个元数据字段
220
推导指示恶意软件的元数据字段属性
230
结束

图 2

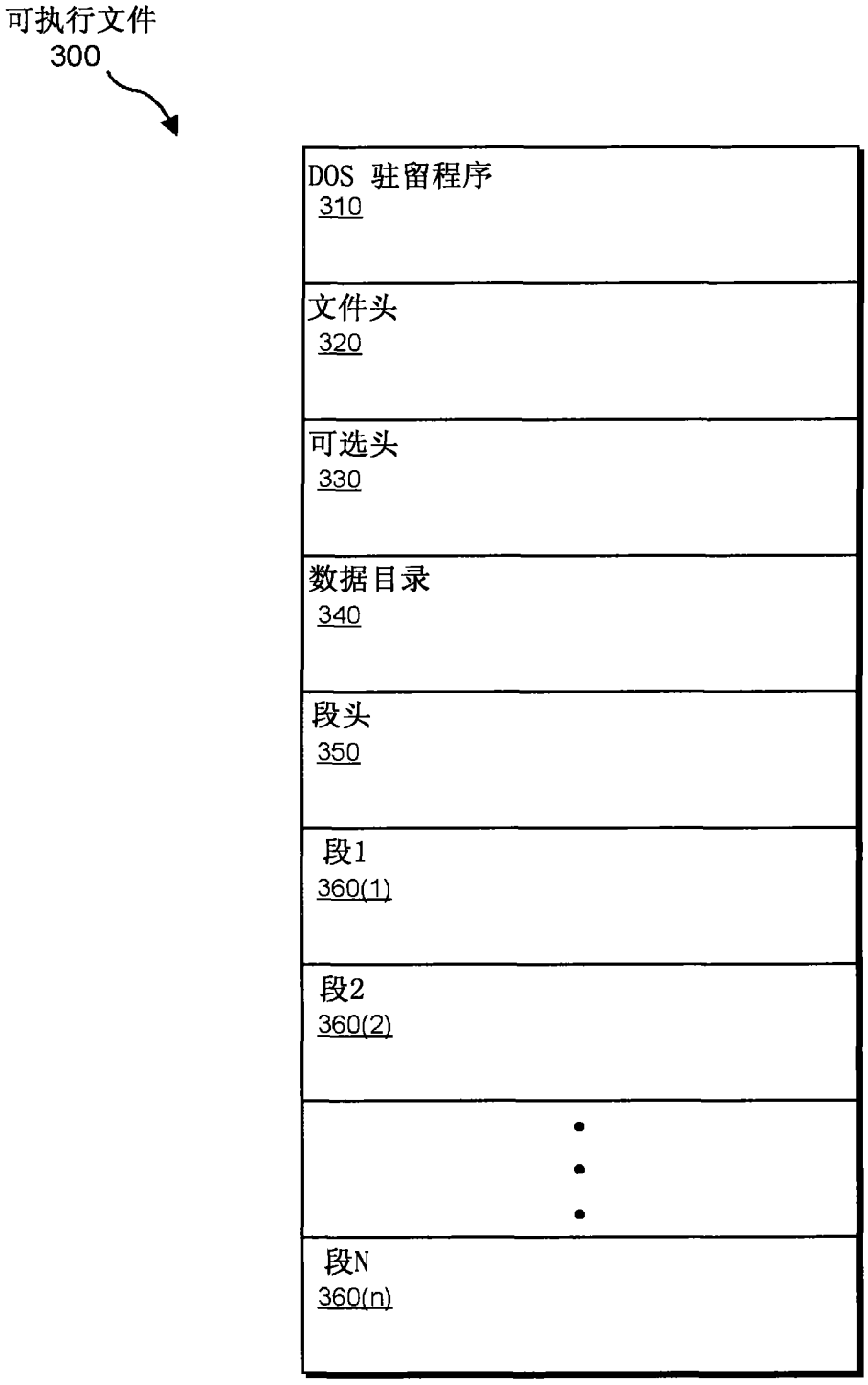

可执行文件
300
DOS 驻留程序
310
文件头
320
可选头
330
数据目录
340
段头
350
段1
360(1)
段2
360(2)
段N
360(n)

图 3

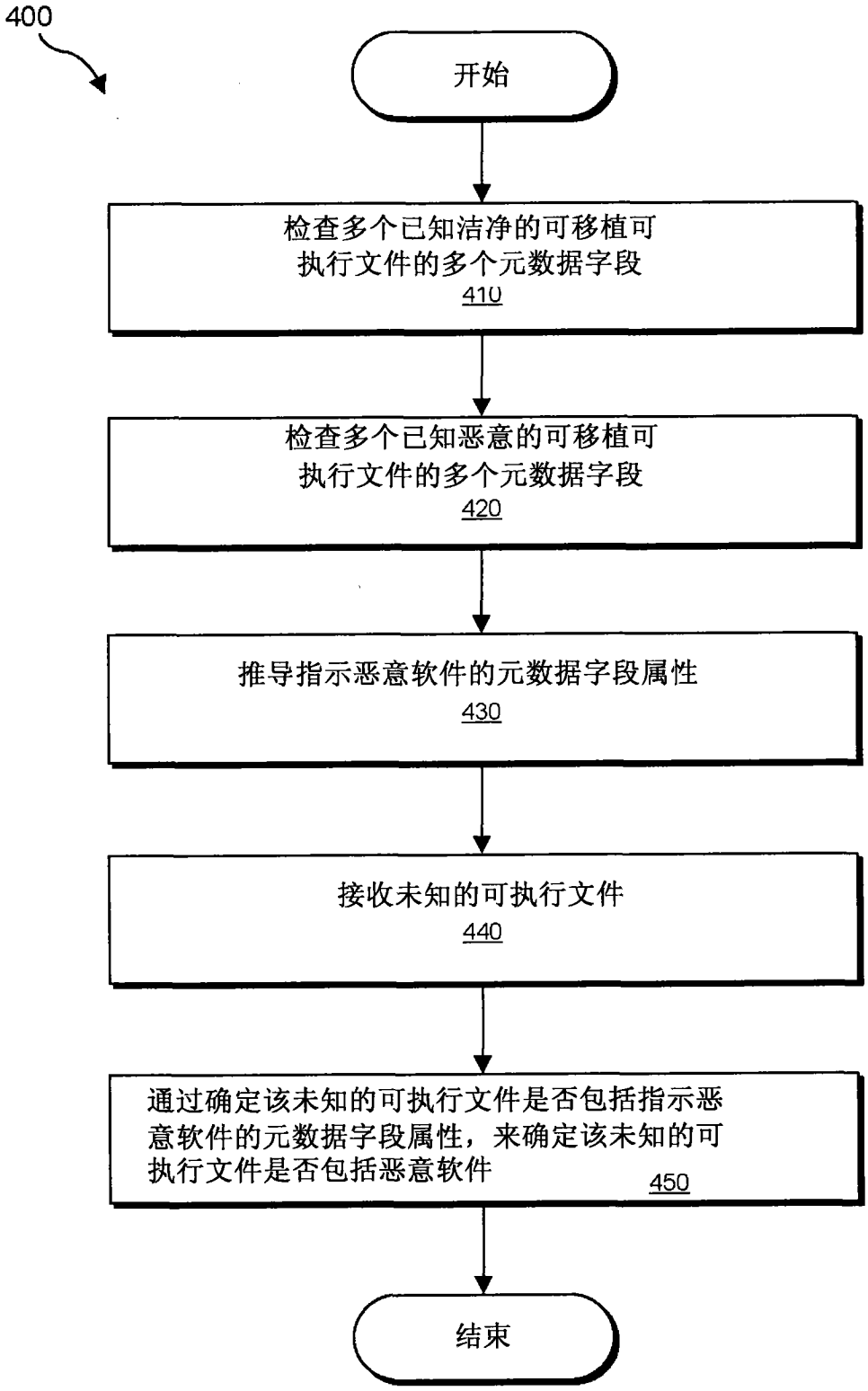

400
开始
检查多个已知洁净的可移植可执行文件的多个元数据字段
410
检查多个已知恶意的可移植可执行文件的多个元数据字段
420
推导指示恶意软件的元数据字段属性
430
接收未知的可执行文件
440
通过确定该未知的可执行文件是否包括指示恶意软件的元数据字段属性，来确定该未知的可执行文件是否包括恶意软件
450
结束

图 4

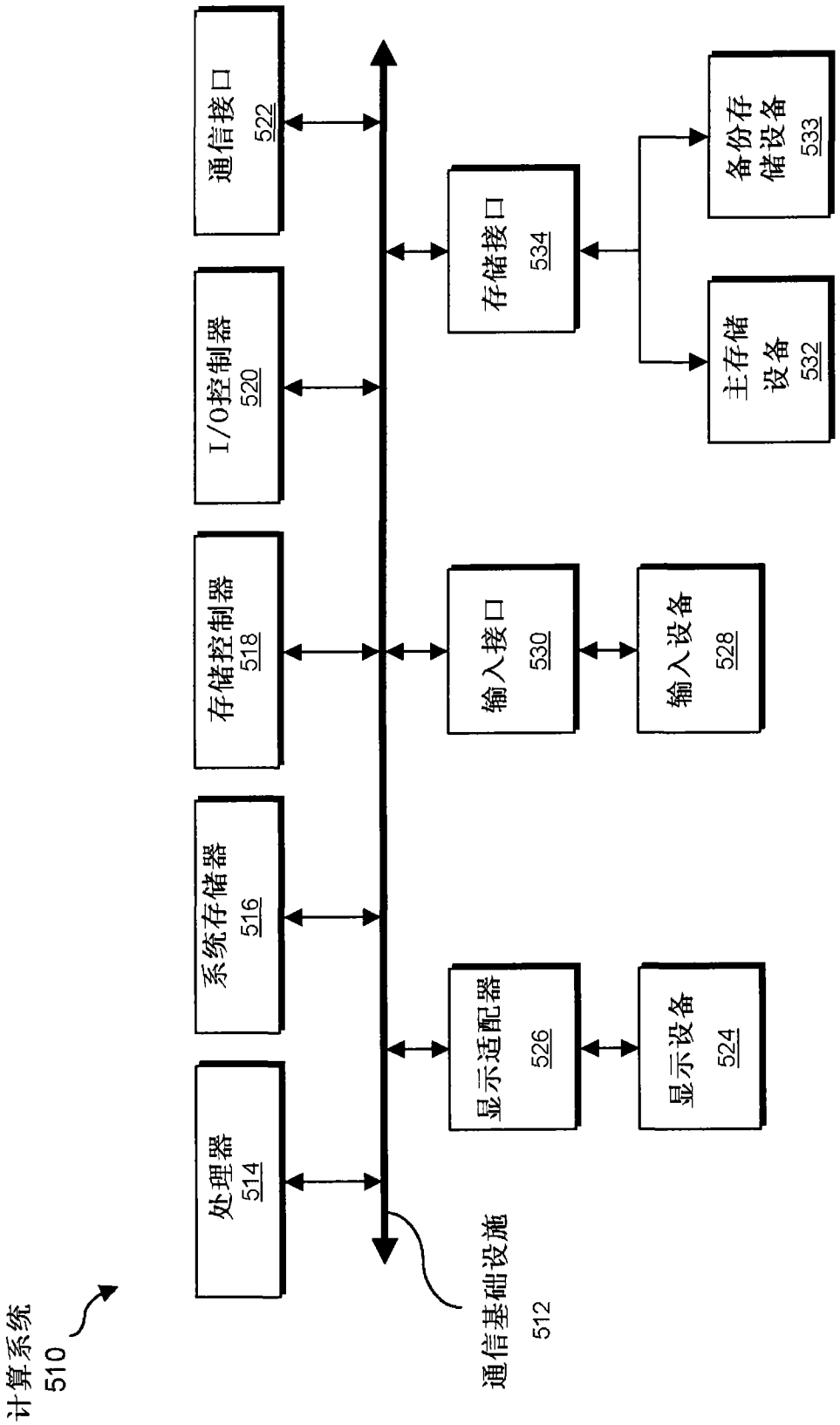

计算系统
510
处理器
514
系统存储器
516
存储控制器
518
I/O控制器
520
通信接口
522
通信基础设施
512
显示适配器
526
显示设备
524
输入接口
530
输入设备
528
存储接口
534
主存储设备
532
备份存储设备
533

图 5

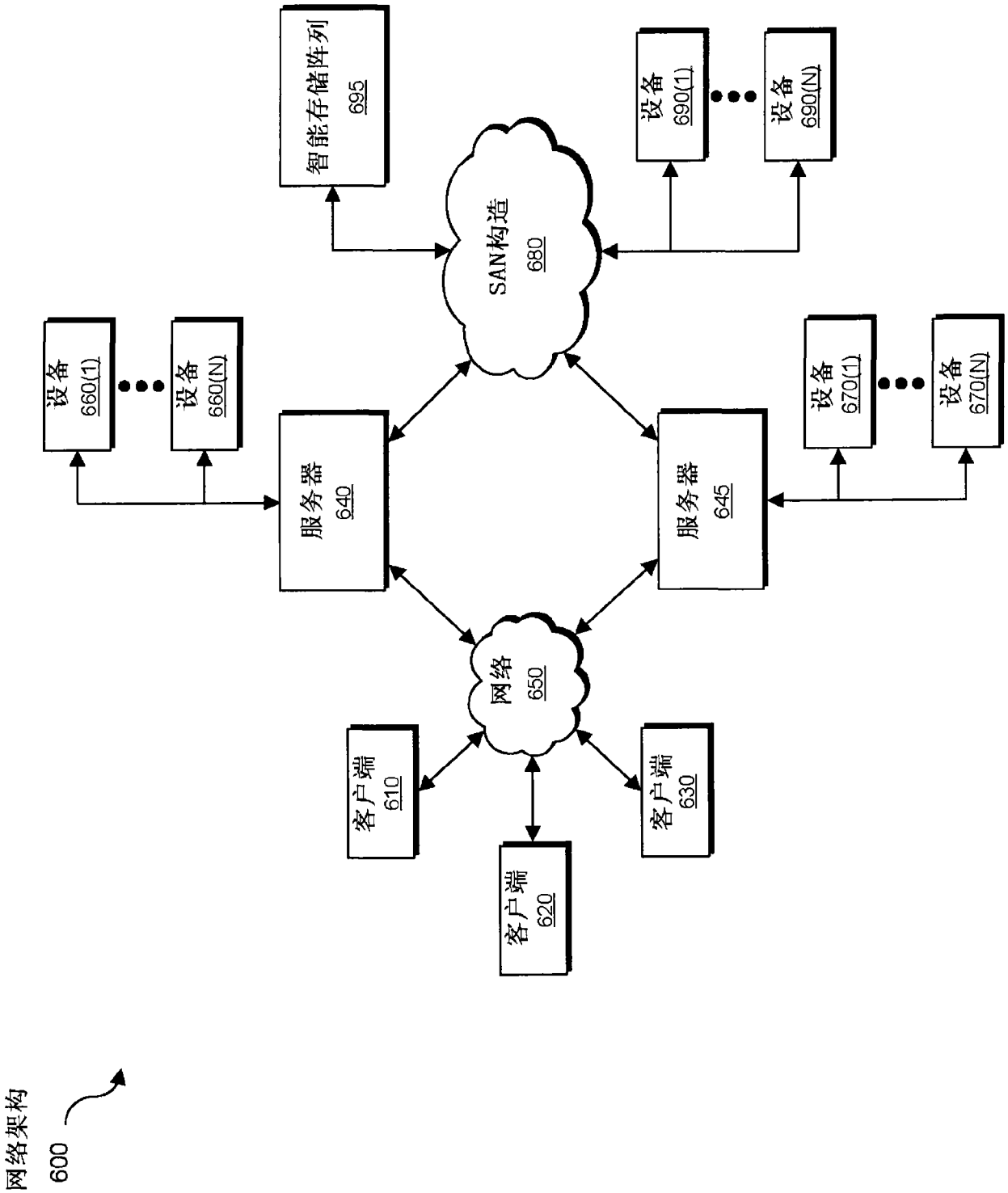

网络架构
600
智能存储阵列
695
SAN构造
680
设备
690(1)
设备
690(N)
设备
660(1)
设备
660(N)
服务器
640
服务器
645
设备
670(1)
设备
670(N)
网络
650
客户端
610
客户端
620
客户端
630

图 6