

一种装置代码风险预警方法、软件系统和运算装置

技术领域

本发明涉及代码风险预警领域，尤其涉及一种装置代码风险预警方法、软件系统和运算装置。

背景技术

针对一在线运行的网站服务器进行漏洞检测时，现有技术需建立网络联机至在线服务器，取得在线服务器上网页前端系统的源代码后，以补丁代码的源代码来和所取得的源代码进行比对，来判断是否存在已知或新发现的漏洞。其中，在线运行的网站服务器（在线服务器）本身也是一种硬件装置，其作业环境亦执行了大量的二进制代码（例如：Linux Kernel、Web Server 等）。

针对在线服务器上的网站后端系统和在线运行的硬件装置进行漏洞检测时。由于服务器上的中间码、二进制代码，以及硬件装置里的二进制代码皆是人类无法识别的代码，且该服务器和硬件装置已经在线运行，因此目前的现有技术是经由网络，依据一特定漏洞的行为描述，对该服务器和硬件装置发出仿真攻击的一网络封包，在该服务器和硬件装置收到该网络封包后，监测该服务器和硬件装置的反应是否符合该特定漏洞的行为描述，若是相符，即表示该在线运行的服务器和硬件装置中存有该特定漏洞，进而为服务器和在线硬件装置的运营者提供安全预警的功效。

针对在线服务器及/或硬件装置的漏洞进行分析和风险的检测与防控时，由于运行于服务器及/或硬件装置中的代码是以中间码或二进制代码为主，在缺乏源代码的情况下，无法以补丁的源代码来跟在线系统的源代

码进行比对；源代码的比对，在没有源代码的情况下将无法实施。而仿真封包的漏洞分析，则需网络环境的支持，在脱机或者是网络不能正常运作的情况下，模拟封包无法送到待测的在线的服务器及/或硬件装置，仿真封包的漏洞分析和风险监控亦无法实施。且当特定漏洞的驱动并非是能够以模拟封包来触发，则该特定漏洞亦无法以模拟封包的方式来检测。

综合以上的说明可知，在信息安全的风险防控技术领域里，如何对缺乏源代码及/或在线运行的硬件装置进行漏洞分析和安全预警是一重要的技术课题。目前尚未有一种装置代码风险防控方法、软件系统和运算装置，能够在缺乏装置系统的源代码及/或是网络脱机的状态，为装置提供漏洞分析和安全预警的风险防控方法与运用该方法的系统，故仍亟待开发。

发明内容

本发明提供一种装置代码风险预警方法、软件系统和运算装置，是通过以下技术方案来实现的：

方案一：

一种装置代码风险预警方法，包括以下步骤：

收集一装置代码的一更新请求，依据所述更新请求的内容，取得所述装置代码，并将所述装置代码传输至一装置；所述装置代码包含至少一软件代码；

当相对应于一风险代码的一代码分析事件触发时，取得所述风险代码，对所述风险代码和至少一所述软件代码进行成份内容的比对；

当所述比对的结果显示出所述至少一软件代码包含至少一安全风险时，发出风险预警通知。

进一步的，所述发出风险预警通知，是指透过至少一网路通信渠道，向所述装置代码的提供者、所述装置的运营者、管理者中至少一人，发出一风

险预警的通知。

进一步的，所述装置代码是由一第一用户提供，所述更新请求的收集是针对所述装置代码的发布及/或更新而接收；所述装置是由一第二用户所运营。

进一步的，所述至少一软件代码包含至少一二进制代码。

进一步的，所述取得装置代码具体为，从一储存媒体取得所述装置代码，包括：

所述装置代码是经由一代码版本信息所指定，依据所述代码版本信息从所述储存媒体取得所述装置代码。

进一步的，所述代码版本信息是由所述更新请求的内容中直接指定，或者是依据所述更新请求的内容中所包含的一产品规格、一硬体规格，及/或一软韧体规格，而来决定出所述装置所需更新的所述代码版本信息。

进一步的，所述更新请求是经由网路传输的线上更新请求。

进一步的，所述更新请求的内容包含所述装置的网路位置信息，依据所述网路位置信息，经由网路将所述装置代码传输至所述装置。

进一步的，所述风险代码和所述至少一软件代码进行成份内容的比对，包括：

撷取相对应于所述风险代码的至少一第一成份特征；

撷取相对应于所述至少一软件代码的至少一第二成份特征；

比对所述至少一第一成份特征和所述至少一第二成份特征。

进一步的，所述第一成份特征是藉由对相对应于所述风险代码的代码组成内容及/或代码执行结果进行剖析后所得。

进一步的，所述代码组成内容包括所述风险代码及/或所述风险代码编译后的第一中间代码，所述代码执行结果则包括对所述风险代码及/或所述第一中间代码做模拟执行时，所述风险代码及/或所述第一中间代码在模拟执行过程中所显露出的语义信息。

进一步的，所述至少一第二成份特征是藉由对相对应于所述至少一软件代码的至少一代码组成内容及/或至少一代码执行结果进行剖析后所得。

进一步的，所述至少一代码组成内容包括至少一二进制代码及/或所述至少一二进制代码反组译后的至少一第二中间代码，所述至少一代码执行结果则包括对所述至少一二进制代码及/或所述至少一第二中间代码做模拟执行时，所述至少一二进制代码及/或所述至少一第二中间代码在模拟执行过程中所显露出的语义信息。

进一步的，所述中间代码的程序语言结构是由一中间语言、一中间表示、一结构化语言，及/或一结构化协议所宣告而成。

进一步的，所述风险代码的种类是包括一补丁代码、恶意程序代码，及/或任一种用于修正或产生一软件漏洞或软件病毒的代码。

进一步的，相对应于所述风险代码的所述代码分析事件是在所述风险代码被新增或被更新时所触发。

进一步的，所述代码分析事件触发前，依据所述至少一软件代码的一组清单，对至少一代码分析事件进行侦测，相对应于所述风险代码的所述代码分析事件是所述至少一代码分析事件的其中之一。

进一步的，所述组成清单是在所述装置代码被收集时，对所述装置代码的成份组成进行扫描及/或剖析后所得。

进一步的，所述组成清单包括所述装置代码本身的元数据及/或所述装置代码成份组成中所含的所述至少一软件代码的元数据。

进一步的，所述至少一软件代码是以包括系统核心、驱动程序、应用程序、组件，及/或类别库的型态存在于所述装置代码的成份组成中。

进一步的，所述装置代码是一系统映像档，对系统映像档中所含的成份组成进行扫描及/或剖析，辨识出存在于所述系统映像档中的所述至少一软件代码，进而得到所述至少一软件代码的所述组成清单；

或者所述装置代码是一软件代码包裹，对所述软件代码包裹中所含的成

份组成进行扫描及/或剖析，辨识出存在于所述软件代码包裹中的所述至少一软件代码，进而得到所述至少一软件代码的所述组成清单。

进一步的，所述风险预警的通知夹带一指令，藉由所述指令来指示所述装置及/或操作所述装置的用户，于所述装置上执行所述风险代码的更新工作；或者所述风险预警的通知夹带一提示信息，藉由所述提示信息为所述装置代码的提供者、所述装置的运营者及/或管理者提供对所述至少一安全风险的预警。

方案二：

一种装置代码风险预警的软件系统，包括收集单元、取码单元、比对单元、传输单元；

所述收集单元用于接收装置代码的更新请求；

所述取码单元用于取得所述装置代码，所述装置代码包含至少一软件代码；

所述传输单元用于将所述装置代码传输至一装置和发出风险预警通知；

所述比对单元用于，相对应于一风险代码的一代码分析事件触发时，取码单元取得所述风险代码，对所述风险代码和所述装置代码进行比对。

进一步的，所述比对单元包括第一特征提取单元、第二特征提取单元和特征比对单元；

所述第一特征提取单元，用于撷取相对应于所述风险代码的至少一第一成份特征；

所述第二特征提取单元，用于撷取自相对应于所述至少一软件代码的至少一第二成份特征；

所述特征比对单元，用于对至少一第一成份特征和至少一第二成份特征进行比对。

进一步的，所述第一特征提取单元或第二特征提取单元包括至少一剖

析单元和至少一撮取单元；

所述至少一剖析单元，用于对相对应于所述风险代码的代码组成内容及/或代码执行结果进行剖析后得到第一成份特征，及/或对相对应于所述至少一软件代码的代码组成内容及/或代码执行结果进行剖析后得到第二成份特征；

所述至少一撮取单元，用于撮取剖析后所得到的第一成份特征及/或第二成份特征。

进一步的，还包括至少一侦测单元，

所述至少一侦测单元，用于依据所述至少一软件代码的组成清单，对所述组成清单中所含各个软件代码所相对应的风险代码登录列表进行监控，藉由监控所述风险代码登录列表的异动来侦测代码分析事件。

方案三

一种装置代码风险预警的运算装置，包括收集单元、取码单元、比对单元、传输单元；

所述收集单元用于接收装置代码的更新请求；

所述取码单元用于从一储存媒体取得所述装置代码，所述装置代码包含至少一软件代码；

所述传输单元用于将所述装置代码传输至一装置和发出风险预警通知；

所述比对单元用于，相对应于一风险代码的一代码分析事件触发时，取码单元从所述储存媒体及/或另一储存媒体取得所述风险代码，对所述风险代码和所述装置代码进行比对。

如此设计具有如下优点：本发明通过对一在线运行的装置进行包含至少一软件代码的装置代码的更新、对风险代码和所述装置代码成份内容的比对，以及为所述装置发出风险预警，无论所述装置是否离线，网路状态是否正常，也能对所述装置进行风险预警。

附图说明

图 1 是本发明一种装置代码风险预警方法的流程图；

图 2 是本发明一种装置代码风险预警的软件系统的结构框图；

图 3 是本发明风险代码和至少一软件代码进行成份内容的比对的流程图。

图 4 是代码分析事件触发前软件系统依据至少一软件代码的一组成清单的装置代码风险预警方法的流程图。

图 5 是本发明一种装置代码风险预警的运算装置的结构框图；

图中，1、软件系统；2、收集单元；3、取码单元；4、比对单元；5、传输单元；6、第一用户；7、第二用户；8、装置；9、储存媒体；80、装置代码；11、运算装置；21、收集单元；31、取码单元；41、比对单元；51、传输单元。

具体实施方式

本发明得藉由以下描述，包括以下的术语汇编以及结论性实例，而更充分地理解。为简洁起见，本说明书中引用之出版物，包括专利的公开内容，系并入本文以供参考。

以下系提出实施例进行详细说明，本发明的实施例是非限制性的，并且仅代表本发明之各种态样及特征，实施例仅用以作为范例说明，并不会限缩本发明欲保护之范围。此外，实施例中之图式系省略不必要或以通常技术即可完成之组件，以清楚显示本发明之技术特点。

就最受限制的技术意义而言，本文所述的「软件代码」的代码组成内容包括任一种供加载和执行用的二进制代码（Binary Code）、中间码及/或任一种由计算机程序语言编写的计算机程序代码。本文所述的「装置代码」是用于在一装置上运行，并于该装置上为该装置提供逻辑运算及服务的代码，所述装置代码由包括二进制代码、中间码，及/或是任一种由计算机程序语言编写的计算机程序代码所组成，亦即所述装置代码包含至少一软件代码，所述至少一软件代码是以包括系统核心、驱动程序、应用程序、组件，及/或类别库的组件型态存在于所述装置代码的成份组成中。本文所述的「二进制代码」可包括任一种以二进制作作为表示格式的代码。本文所述的「中间代码」可包括任一种以一中间语言、中间表示、结构化语言，及/或结构化协议所宣告的结构所表示的代码；例如，遵循一中间表示所定义的格式而表示的 Java Bytecode，以及从二进制代码反组译而得的 LLVM bitcode，皆为一种中间代码，也可视为中间码。本文所述的「风险代码」的代码组成内容包括任一种供加载和执行用的二进制代码（Binary Code）、中间码，及/或任一种由计算机程序语言编写的计算机程序代码，所述风险代码的种类包括一补丁代码、恶意程序代码，及/或任一种用于修正或产生一软件漏洞（Vulnerabilities）或软件病毒的代码。本文所述的「元数据」又称诠释数据、中介数据或元数据，是为描述一对象或该对象相关信息之数据。本文所述的「风险代码登录列表」可包括任一种依据一结构化语言及/或结构化协议所宣告的结构所登录、注册及/或表示的一列表，所述列表可以包括一文件、资料库及/或网页的形式储存和发布，所述列表的内容包含本文所述风险代码的登录信息，其中所述登录信息中，包括风险代码本身的内容、风险代码的元数据、风险代码所对应的软件漏洞信息或软件病毒信息；其中 CVE（Common Vulnerabilities and Exposures）列表（CVE Repository）是本文所述风险代码登录列表的一种实施态样，但不以此为限。本文所述的「语义信息」可包括在模拟执行

一代码时，在执行过程中从内存及/或中间变量取出的值、执行过程中各指令的执行历程、函数的输出/输入值，及/或任一种可被撷取的数值化信息。两份代码的语义信息的全部或一部份片段，可作为两相比对用的成份特征，依比对的相符程度，决定出两份代码的相似度。

又，本文所述的「收集单元」、「取码单元」、「传输单元」、「比对单元」、「第一特征提取单元」、「第二特征提取单元」、「特征比对单元」、「剖析单元」、「撷取单元」和「侦测单元」是一种提供逻辑运算或服务的处理单元，所述处理单元的实施态样可包括一类别库、软件组件、软件代理程序、应用程序、任一种可被内建、安装、引入或扩充至本文所述「提供装置代码风险预警的软件系统或运算装置」中运行的程序区块，或者是任一种以硬件描述语言所设计而可被嵌入本文所述「提供装置代码风险预警的运算装置」的集成电路单元。

本文所述的「提供装置代码风险预警的软件系统」，其实施态样可包括一系统韧体、应用程序、软件引擎、操作系统，或者是任一种可被程序扩充的软件平台。于一较佳实施例中，本文所述「提供装置代码风险预警的软件系统」可以部署在同一机器上运行，亦即将本文所述收集单元、取码单元、传输单元、比对单元、第一特征提取单元、第二特征提取单元、特征比对单元、剖析单元、撷取单元、侦测单元，全部集成在同一软件系统中，并将该软件系统部署至一机器上运行。于另一较佳实施例中，本文所述「提供装置代码风险预警的软件系统」可以透过多部机器的协同来加以实施，亦即将本文所述收集单元、取码单元、传输单元、比对单元、第一特征提取单元、第二特征提取单元、特征比对单元、剖析单元、撷取单元、侦测单元等处理单元，提供至不同的机器中，不同机器里的处理单元，彼此之间经由包括 internet、WiFi、蓝牙等联机方式来协同交互，进而通

过多部机器来共同集成为一提供装置代码风险预警的软件系统。

本文所述的「提供装置代码风险预警的运算装置」，其实施态样可包括一网络通讯设备、服务器，或者是任一种具连网、数据处理或控制能力的资通讯设备。于一较佳实施例中，本文所述「提供装置代码风险预警的运算装置」可以是单一机器所实施，亦即将本文所述收集单元、取码单元、传输单元、比对单元、第一特征提取单元、第二特征提取单元、特征比对单元、剖析单元、撷取单元、侦测单元，全部集成在同一机器中，作为提供装置代码风险预警的运算装置。于另一较佳实施例中，本文所述「提供装置代码风险预警的运算装置」可以由多部机器的协同来加以实施，亦即将本文所述收集单元、取码单元、传输单元、比对单元、第一特征提取单元、第二特征提取单元、特征比对单元、剖析单元、撷取单元、侦测单元等处理单元，提供至不同的机器中，不同机器里的处理单元，彼此之间可以经由包括 internet、WiFi、蓝牙等联机方式来协同交互，进而通过多部机器来共同集成为一提供装置代码风险预警的运算装置。

承上，于一较佳实施例中，本文所述的「收集单元」、「取码单元」、「传输单元」、「比对单元」、「第一特征提取单元」、「第二特征提取单元」、「特征比对单元」、「剖析单元」、「撷取单元」和「侦测单元」可以个别实作于不同的处理单元中。于另一较佳实施例中，亦可于同一处理单元里，对本文所述的「收集单元」、「取码单元」、「传输单元」、「比对单元」、「第一特征提取单元」、「第二特征提取单元」、「特征比对单元」、「剖析单元」、「撷取单元」和「侦测单元」中的至少二者提供实作。

下面结合附图和具体实施方式对本发明的内容做进一步详细说明。

如图 1 所示为本发明一种装置代码风险预警运算的较佳流程图。装置代码 80 风险预警的方法包括：

S1、收集一装置代码 80 的一更新请求，依据所述更新请求的内容，从一储存媒体 9 取得所述装置代码 80，并将所述装置代码 80 传输至一装置；所述装置代码 80 包含至少一软件代码；

S2、当相对应于一风险代码的一代码分析事件触发时，软件系统 1 从所述储存媒体 9 及/或另一储存媒体 9 取得所述风险代码，对所述风险代码和所述至少一软件代码进行成份内容的比对；

S3、当比对结果显示所述至少一软件代码包含至少一安全风险时，发出风险预警通知。

如图 2 所示为本发明一种装置代码 80 风险预警的软件系统 1 的结构框图。提供本案装置代码风险预警运算于一较佳实施例的说明。其中，第一用户 6 是软件系统 1 的用户，第二用户 7 是装置 8 的运营者或管理者；收集单元 2、取码单元 3、比对单元 4 和传输单元 5 是集成至软件系统 1 中的处理单元；储存媒体 9 是一内存、外接式储存装置、数据库，或者是任一种可以储存装置代码 80 或风险代码的储存装置；装置代码 80 是用于在装置 8 上运行，并于装置 8 上为装置 8 提供逻辑运算及服务的代码。

于一较佳实施例中，第二用户 7 是软件系统 1 的用户，藉由将装置 8 的元数据注册至软件系统 1，通过软件系统 1 来对装置 8 进行管理。其中，装置 8 的元数据包括装置 8 的网路位置信息、产品规格、硬体规格，及/或软体规格。于另一较佳实施例中，第二用户 7 仅为装置 8 的运营者，装置 8

的元数据是经由软件系统 1 的系统管理者注册至软件系统 1，由软件系统 1 的系统管理者，作为装置 8 的管理者。

于一较佳实施例中，第一用户 6 是装置 8 的开发商，第一用户 6 藉由软件系统 1 所提供的一人机操作接口，将装置代码 80 提供给软件系统 1。在第一用户 6 提供装置代码 80 至软件系统 1 时，第一用户 6 可于该人机操作接口上，输入装置代码 80 的相关资料，并将该些相关资料藉由该人机操作接口，连同装置代码 80 一并提供予软件系统 1；其中该些相关数据包括装置代码 80 的代码版本信息、以及装置代码 80 所支持更新的装置 8 之产品规格、硬件规格，及/或软韧体规格。

于一较佳实施例中，软件系统 1 将装置代码 80 及其相关数据（包括所述装置代码 80 的代码版本信息、以及装置代码 80 所支持更新的装置 8 之产品规格、硬件规格，及/或软韧体规格），储存至储存媒体 9 中，完成装置代码 80 于软件系统 1 中的登录。

于一较佳实施例中，收集单元 2、取码单元 3、比对单元 4 和传输单元 5 作为集成至软件系统 1 中的处理单元，每个处理单元可以提供至少一个函数来实现与其它处理单元之间的交互，当一处理单元需要对另一处理单元发出请求时，得以将请求的内容作为参数，以函数调用的方式，将请求的内容输入至另一处理单元，该另一处理单元收到所述请求的内容后，即可依据所述请求的内容，为所述请求提供服务。于另一较佳实施例中，每个处理单元也可以藉由网络联机或行程间通讯（Inter-process communication）等方式来实现与其它处理单元之间的交互，当一处理单元需要对另一处理单元发出请求时，该处理单元可以对该另一处理单元建立一 TCP/UDP Socket 连线，通过彼此约定好的通讯协议，将请求的内容传送至该另一处理单元，或者是

通过以部分内存共享所实作的行程间通讯机制，将请求的内容发送至该另一处理单元，使得该另一处理单元得以收到所述请求的内容，依据所述请求的内容，为所述请求提供服务；其中，如何将一请求提供至一处理单元，以及处理单元如何接收该请求的实施方式，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

于软件系统 1 完成装置代码 80 的登录后。于一较佳实施例中，软件系统 1 可以以装置代码 80 所支持产品规格、硬件规格，及/或软韧体规格，对注册于软件系统 1 中的装置 8 进行搜寻，如果装置 8 的产品规格、硬件规格，及/或软韧体规格与装置代码 80 所支持的规格相符，软件系统 1 即可为装置 8 生成出一更新请求，该更新请求是一个用于更新装置代码 80 的更新请求，并将该更新请求提供予收集单元 2。于另一较佳实施例中，当第一用户 6 透过软件系统 1 的人机操作接口，将装置代码 80 提供至软件系统 1 时，可以直接将针对装置 8 的更新请求，输入人机操作接口，将所述更新请求直接提供予软件系统 1，软件系统 1 收到所述更新请求后，即可将所述更新请求提供予收集单元 2。于另一较佳实施例中，收集单元 2 可以藉由网络联机来接收请求，于此情况下，第一用户 6 得以通过任意的其他软件系统 1，建立一 TCP/UDP Socket 连线至收集单元 2，并将针对装置 8 的更新请求，以在线更新请求的态样，透过网络联机传输并提供至收集单元 2。于另一较佳实施例中，软件系统 1 通过一网络连线发出通知至装置 8，通知装置 8 有一新的装置代码 80 已登录至软件系统 1，并将装置代码 80 的代码版本信息夹带在通知的内容中，装置 8 在收到通知后，即可依据通知的内容，生成出一含有所述代码版本信息的更新请求，以在线更新请求的态样，建立网络连线至收集单元 2，并通过网络传输来将所述在线更新请求提供至收集单元 2。于另一较佳实施例中，软件系统 1 不需要将所述代码版本信息夹带于对装置 8 的通知之内容中，装置 8 在收到通知后，可生成出一更新请求，所述更新请求的

内容包含装置 8 的产品规格、硬体规格，及/或软韧体规格，并将所述更新请求经过网络连线传输至软件系统 1，软件系统 1 收到所述更新请求后，得以依据所述更新请求的内容，决定出与所述产品规格、硬体规格，及/或软韧体规格相符合的装置代码 80 之代码版本信息，并将所述代码版本信息加入至所述更新请求的内容后，将所述更新请求提供予收集单元 2。

于一较佳实施例中，软件系统 1 在完成装置代码 80 的登录后，完全不需要对装置 8 发出任何通知。由第二用户 7 所运营的装置 8，周期性地经由网络联机向软件系统 1 发出线上更新请求，所述更新请求的内容包含装置 8 的产品规格、硬体规格，及/或软韧体规格，经过网络连线传输至软件系统 1，软件系统 1 收到所述更新请求后，即得以依据所述更新请求的内容，藉由比对所述产品规格、硬体规格，及/或软韧体规格的方式，决定出与所述产品规格、硬体规格，及/或软韧体规格相符合的装置代码 80 之代码版本信息，当装置代码 80 是装置 8 所需更新的装置代码 80，软件系统 1 即可将所述代码版本信息加入至所述更新请求的内容后，并将所述更新请求提供予收集单元 2。于另一较佳实施例中，第二用户 7 作为装置 8 的运营者，亦可经由另一软件系统 1 来为装置 8 生成更新请求，所述更新请求的内容包含装置 8 的产品规格、硬体规格，及/或软韧体规格，并经过网络连线将所述更新请求传输至软件系统 1。

综合以上所述，更新请求内容的生成、传输，以及收集单元 2 对更新请求的收集方法，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

进一步的，所述装置代码 80 是由第一用户 6 所提供，装置代码 80 于软件系统 1 中的登录，可以是针对一新装置 8 所发布的装置代码 80，亦可以是

针对一旧装置 8 所更新的装置代码 80，亦或者是针对刚注册于软件系统 1 的既有装置 8 所更新和发布的装置代码 80。所述装置代码 80 包含的至少一软件代码中，包含至少一二进制代码。

于一较佳实施例中，所述装置 8 是由第二用户 7 所运营。所述装置 8 是一嵌入式系统装置、服务器装置、物联网感测设备、网络通讯设备，或任一具联网能力的计算机装置。

于一较佳实施例中，所述更新请求的内容中，包含装置 8 的网络位置信息。所述网络位置信息是依据注册于软件系统 1 中的装置 8 的元数据，由软件系统 1 所填入更新请求中；或者所述更新请求是由装置 8 所生成，在装置 8 生成所述更新请求时，由装置 8 将本身的网络位置信息所填入。

于一较佳实施例中，收集单元 2 收集到装置代码 80 的一更新请求后，即可对取码单元 3 发出请求，请求取码单元 3 依据更新请求内容中的代码版本信息，从储存媒体 9 中取得与所述代码版本信息相符合的装置代码 80，即所述代码版本信息所指定的装置代码 80。取得装置代码 80 后，即可对传输单元 5 发出请求，请求传输单元 5 依据更新请求内容中所含的装置 8 的网络位置信息，对装置 8 建立一 TCP/UDP Socket 连线，通过所述的 Socket 连线和彼此约定的通讯协议，将装置代码 80 传输至装置 8，在装置 8 上完成装置代码 80 的更新，令装置代码 80 成为装置 8 在线运行时所执行的代码。

所述一种装置代码风险预警运算实现了第一用户 6 通过软件系统 1 能对第二用户 7 所运营的装置 8 进行装置代码 80 更新，使装置代码 80 成为装置 8 在线运行时所执行的代码；在装置 8 是离线的环境下，实施装置代码风险预警运算的软件系统 1，能为装置 8 进行装置代码 80 的风险预警。在上述方

法下，装置 8 是在线的情况下，软件系统 1 也能对为装置 8 进行装置代码 80 的风险预警，也即在线和离线的情况均包含于装置代码风险预警运算之中。

S1 中所述软件系统 1 其中所述软件系统 1 是依据对应于所述装置代码 80 的代码版本信息，从所述储存媒体 9 取得所述装置代码 80；

进一步的，所述代码版本信息是由所述更新请求的内容中直接指定，或者所述软件系统 1 是依据所述更新请求的内容中所包含的一产品规格、一硬体规格，及/或一软韧体规格，而来决定出所述装置 8 所需更新的所述代码版本信息。

如图 2，针对所述更新请求若由所述第一用户 6 发出，显然第一用户 6 是能在更新请求中直接填入该装置代码 80 的版本信息，该软件系统 1 在收到该更新请求后，就可以直接依据更新请求内容中的版本信息，从储存媒体 9 中取得该装置代码 80。但所述更新请求若由第二用户 7 或者是装置 8 所发出，那么第二用户 7 和装置 8 将不会知道第一用户 6 的所提供的装置代码 80 的最新版本信息是什么，第二用户 7 只掌握得到其本身所运营的装置 8，以及所运营的装置 8 本身的产品规格、硬体规格，及/或软韧体规格。

因此，在第二用户 7 和装置 8 所发出的更新请求包括装置 8 本身的产品规格、硬体规格，及/或软韧体规格，但不包括装置代码 80 的最新版本信息的情况下，当第二用户 7 和装置 8 的更新请求送达软件系统 1 时，软件系统 1 可以以第二用户 7 和装置 8 传来的产品规格、硬体规格，及/或软韧体规格来决定出该装置 8 使用的是哪一个版本的装置代码 80，在装置代码 80 所支持的产品规格、硬体规格，及/或软韧体规格与所述更新请求中的产品规格、硬体规格，及/或软韧体规格相符，即可决定出装置代码 80 是第二用户 7 和装置 8 所需要的装置代码 80。

于一较佳实施例中，为实现软件系统 1 对不同版本的装置代码 80 的比对和决定采用哪一个，不同版本的装置代码 80 所支持的产品规格、硬体规格，及/或软韧体规格均能在登录装置代码 80 时，记录于所述储存媒体 9 或该软件系统 1 所能存取的其他数据库或媒体中。这样就可实现软件系统 1 对不同版本的装置代码 80 的比对，并决定出适合第二用户 7 装置 8 的版本信息，进而取出所述版本信息指定的装置代码 80。

S1 中更新请求是经由网路传输的线上更新请求。

于一较佳实施例中，所述经由网路传输的线上更新请求在传统意义上也称为代码的 OTA 线上更新，基本是由装置 8 连线至服务器，向服务器取得最新的代码后进行更新，服务器以及服务器上执行的软件系统 1 并不需要知道装置 8 的网路位置信息。亦即，装置 8 依据软件系统 1 的网络位置信息，建立网络连线至提供网络联机接入功能的软件系统 1，向软件系统 1 发出更新请求后。随后，经由同一条网络连线从软件系统 1 直接取得装置代码 80。或者是装置 8 依据软件系统 1 的网络位置信息，建立网络连线至提供网络联机接入功能的收集单元 2，向收集单元 2 发出更新请求后，随后，建立另一条网络连线至传输单元 5，经由另一条网络连线从传输单元 5 取得装置代码 80。其中，以装置 8 端所建立的网络连线来实施代码的 OTA 在线更新之联机和取得代码之方法，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

就上述代码的 OTA 在线更新方法，装置 8 必须知道服务器的网址位置信息。因此本案软件系统 1 就所述经由网路传输的线上更新请求为代码的 OTA 更新，亦提供包括以下改进方法：

所述更新请求的内容包含所述第二用户 7 所运营的装置 8 的网路位置信息，所述软件系统 1 是依据所述网路位置信息，经由网路将所述装置代码 80

传输至所述第二用户 7 所运营的装置 8。

这种改进方法能够对以下情景进行使用：

装置 8 无需知道软件系统 1 以及其服务器的网路位置信息。软件系统 1 以推送的方式为装置 8 完成装置代码 80 的更新，于一较佳实施例中，装置 8 的网路位置信息是由软件系统 1 收到第一用户 6 或者是第二用户 7 所发出的更新请求后，亦或者更新请求是在新的装置代码 80 登录后，由软件系统 1 主动生成的，软件系统 1 皆得以依据注册于软件系统 1 里的装置 8 的元数据而将其网络位置信息写入更新请求；并由传输单元 5 依据更新请求里的网络位置信息，建立网络连线至装置 8，将装置代码 80 传送给该装置 8 做更新。这种以异步的方式完成装置代码 80 的更新的方式，更新请求的发出和装置代码 80 的接收是经由两条连接对象不同的连线来完成的，有别于传统意义上的装置 8 建立连线至软件系统 1，然后透过装置 8 与软件系统 1 的连线来下载。

事实上，上述情景使得装置代码 80 的更新不需要装置 8 本身下载。第一用户 6、第二用户 7 或其他用户在权限允许的情况下，均可以透过上述改进方法，发出更新请求至软件系统 1，随后，软件系统 1 依据装置 8 的网路位置信息，对该网路位置信息所指的装置 8 建立连线，然后将装置代码 80 主动地传输给它完成更新。于一较佳实施例中，更新请求的内容中，并不必要包含装置 8 的网络位置信息，软件系统 1 经由查询数据库的方式，查询到注册于软件系统 1 里的装置 8 的元数据，从所述元数据中，获取装置 8 的网络位置信息后，即可对传输单元 5 发出请求，请求传输单元 5 依据从装置 8 的元数据中获取的网络位置信息，将装置代码 80 传输至装置 8。

所述网路位置信息可以单纯是一个 IP Address，所述软件系统 1 和装置 8 之间有一个默认用于传送装置代码 80 的通讯协定（装置 8 所默认支持的 Port），在这样的情况下，软件系统 1 只需要知道装置 8 的 IP Address，就可以主动对其连线，然后传送更新的装置代码 80。另外，网路位置信息也可以

是 IP Address 加上 Port，也即该网路位置信息里包含了网路通讯协定的指定信息，这样软件系统 1 和装置 8 之间不需要有任何一个默认的网络的通讯协定，软件系统 1 也可以根据 IP Address（位置）加上 Port（通讯协定）来建立装置和装置的连线，并传送更新的装置代码 80。其中，以异步的方式完成装置代码 80 的更新之方法，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

于一较佳实施例中，第一用户 6 所提供的装置代码 80 是以软件代码包裹的型态，被提供至软件系统 1；所述软件代码包裹的型态，可以包括一压缩文件，该压缩文件中包含装置代码 80 所包含的所有软件代码。于另一较佳实施例中，第一用户 6 所提供的装置代码 80 是以系统映像文件的型态，被提供至软件系统 1，所述系统映像文件中包含装置代码 80 所包含的所有软件代码。

于一较佳实施例中，不论是以系统映像文件或软件代码包裹的型态，当软件系统 1 收到装置代码 80 时，即可对装置代码 80 所包含的所有软件代码进行扫描。首先，将装有装置代码 80 的系统映像文件挂载至内存或是由硬盘内存分割出的扇区（如果是 zImage，可将以解压缩的方式，解压缩其内容至一文件夹中），或者是将装有装置代码 80 的软件代码包裹解压缩至硬盘内存的一指定文件夹中，接着即可以文件系统对挂载的扇区或是所述文件夹进行档案和目录的遍历，亦即对系统映像文件和软件代码包裹所含的成份组成进行扫描。依据遍历的结果，掌握到装置代码 80 的档案和目录结构后，即可与已知操作系统核心、驱动程序、应用程序、常用软件组件及/或类别库的档案和目录结构进行比对，比对出档案和目录结构相符合的操作系统核心、驱动程序、应用程序、常用软件组件及/或类别库。依上述程序，软件系统 1 即可在收集到装置代码 80 后，扫描和剖析出装置代码 80 中所含的软件代码

的一组成清单。所述组成清单可由包括以操作系统核心、驱动程序、应用程序、常用软件组件及/或类别库等型态存在的各个软件代码之元数据（所述元数据为描述软件代码相关信息的数据，例如各软件代码的名称等描述信息，皆可为一种元数据的态样）所组成，其中该组成清单的内容里亦可包含装置代码 80 本身的元数据（所述元数据为描述装置代码 80 相关信息的数据）。

于一较佳实施例中，依据所述装置代码 80 所包括的所述至少一软件代码的组成清单，软件系统 1 即可对所述组成清单中所含各个软件代码所相对应的 CVE 列表进行监控（例如针对 Linux Kernel，即对 Linux Kernel 所对应的 CVE 列表监控），藉由监控所述 CVE 列表的异动来侦测代码分析事件。当有一个漏洞被新增至列表、相对于该漏洞的补丁代码被提供并新增至该漏洞纪录，或者是该漏洞的补丁代码有所更新，导致该列表的内容出现异动时，即可触发出一代码分析事件；其中，补丁代码是风险代码所包括的一种态样，但并不以此为限，针对风险代码其中一种态样的恶意软件代码，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。相对应于所述风险代码的所述代码分析事件是在所述风险代码被新增或被更新时所触发，所触发的事件得以由比对单元 4 所接收。其中，CVE 列表是本文所述风险代码登录列表的一种实施态样，并不以此为限，熟知本技艺人士可依据实际应用需求而对不同形式的风险代码登录列表提供任何均等的变更设计。

于一较佳实施例中，在装置 8 完成装置代码 80 的更新后，当一补丁代码被新增或更新，因而导致相对应于所述补丁代码的一代码分析事件被触发，软件系统 1 将所述代码分析事件传递至比对单元 4，比对单元 4 接受到所述代码分析事件时，比对单元 4 得以请求取码单元 3 从储存着所述补丁代码的储存媒体 9 中，将所述补丁代码取出；其中，所述储存媒体 9 可以和储存装置代码 80 的储存媒体 9 为同一储存媒体 9，或者是另一储存媒体 9。于另一

较佳实施例中，当软件系统 1 监控到一补丁代码被新增或更新时，即可先将储存于另一储存媒体 9 中的所述补丁代码的源代码先行取出并予以编译，并将编译后的结果（二进制代码或中间码）储存于储存媒体 9，在此情况下，先行编译后的结果，可视为补丁代码整体的一部分，则比对单元 4 是从储存媒体 9 及所述另一储存媒体 9 中，将整体的补丁代码（包含源代码、二进制代码及/或中间码）取出。上述对风险代码登录列表之监控和取码之方法，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

于一较佳实施例中，在比对单元 4 取出所述补丁代码后，即可对所述补丁代码和装置代码 80 中相对应的至少一软件代码进行成份内容的比对。当所述比对的结果显示出所述至少一软件代码包含至少一安全风险时，即可请求传输单元 5，就所述安全风险发出风险预警通知。

于一较佳实施例中，所述风险预警通知的发出，是指透过至少一网路通信渠道，例如包括电子邮件、手机推播通知、手机简讯通知等网络通信渠道，向所述装置代码 80 的提供者（第一用户 6）、所述装置 8 的运营者（第二用户 7）、管理者（第二用户 7 或软件系统 1 的管理者）中至少一人，发出一风险预警的通知。其中，风险预警之作用在于补丁代码是风险代码所包含的其中一种型态，补丁代码本身之所以存在，其所表示的就是装置代码 80 所含的软件代码中，有着一相对应的已知或新发现的漏洞，而所述补丁代码即为针对该漏洞做修正的代码，若是所述补丁代码和所述软件代码的成份内容比对结果，显示出所述软件代码的成份内容并未包含所述的补丁代码，即表示装置代码 80 中仍存在所述的漏洞，不肖人士即可利用所述漏洞对执行所述装置代码 80，且正在线运行中的装置 8 进行攻击，亦即所述漏洞的存在是包括第一用户 6、第二用户 7 或任意其他第三人的风险，风险预警通知即为针

对此风险，而对相关人士的预先警告通知，据以实现出对装置代码 80 的风险防控。

于一较佳实施例中，当风险代码的型态是一恶意软件代码，所述恶意软件代码所表示的则是一已知漏洞的代码。亦即，若是所述恶意软件代码和所述软件代码的成份内容比对结果，显示出所述软件代码的成份内容仍然包含所述的恶意软件代码，即表示装置代码 80 中仍存在所述的漏洞，而需发出风险预警通知。

针对本文所述的「软件代码的成份内容」及「风险代码的成份内容」，其中所述「成份内容」所指的是包括“所述代码的代码组成内容里所含的程序指令”、“所述代码于内存中的储存结构”、“所述执行代码所需的环境设置及组态资料”；其中，所述代码的代码组成内容包括任一种供加载和执行用的二进制代码 (Binary Code)、中间码，及/或任一种由计算机程序语言编写的计算机程序代码。另，本文所述的「安全风险」包括所述代码中含有漏洞、病毒及/或恶意软件的风险，及/或是任一种让所述代码执行不可预期逻辑运算的风险。上述以风险代码和软件代码的比对来扫描一安全风险的方法，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

如图 3 所示，S2 的所述风险代码和所述至少一软件代码进行成份内容的比对，包括以下步骤：

- S21、撷取相对应于所述风险代码的至少一第一成份特征；
- S22、撷取自相对应于所述至少一软件代码的至少一第二成份特征；
- S23、对至少一第一成份特征和至少一第二成份特征进行比对。

进一步的，S21 中所述第一成份特征是藉由对相对应于所述风险代码的代码组成内容及/或代码执行结果进行剖析后所得；其中，所述代码组成内容包括所述风险代码及/或所述风险代码编译后的第一中间代码，所述代码执行结果则包括对所述风险代码及/或所述第一中间代码做模拟执行时，所述风险代码及/或所述第一中间代码在模拟执行过程中所显露出的语义信息。

进一步的，S22 中所述至少一第二成份特征是藉由对相对应于所述至少一软件代码的至少一代码组成内容及/或至少一代码执行结果进行剖析后所得；其中，所述至少一代码组成内容包括至少一二进制代码及/或所述至少一二进制代码反组译后的至少一第二中间代码，所述至少一代码执行结果则包括对所述至少一二进制代码及/或所述至少一第二中间代码做模拟执行时，所述至少一二进制代码及/或所述至少一第二中间代码在模拟执行过程中所显露出的语义信息。

进一步的，所述其中所述第一中间代码及/或所述至少一第二中间代码的程序语言结构是由一中间语言（intermediate language）、一中间表示（intermediate presentation）、一结构化语言（structured language），及/或一结构化协议（structured protocol）所宣告而成。

S2 中所述风险代码的种类是包括与所述至少一软件代码所关联的一补丁代码、恶意程序代码，及/或任一种用于修正或产生一软件漏洞或软件病毒的代码。

S2 中的所述风险代码的所述代码分析事件是在所述风险代码被新增或被更新至一风险代码登录列表时所触发。

如图 4 所示，S2 所述代码分析事件触发前，进一步包括：所述软件系统 1 依据所述至少一软件代码的一组成清单，对至少一代码分析事件进行侦测。在 S2 中，相对应于所述风险代码的所述代码分析事件是所述至少一代码分析事件的其中之一。

如图 2，装置 8 的装置代码 80 的成份组成可以由包括系统核心、驱动

程序、应用程序、组件，及/或类别库等不同型态存在的软件代码所组成。各软件代码的代码组成内容则可包括一些 Script、二进制代码及/或是中间码。所以至少一软件代码，可以视其为装置代码 80 中的一部分作理解。而至少一代码组成内容，可以视其为所述至少一软件代码中的一部分作理解。

一个装置 8 里所含的该至少一软件代码，一般来说是由一个清单所组成，例如：其中可能包含 Linux Kernel、OpenSSL、sqlite 等组件。这些组件随着不断地被检验，都有漏洞被发现以及相对应的补丁代码被发布，然后新增或更新至网路上的 CVE 列表（CVE repository）和 GIT Hub。

这些漏洞和补丁代码由所属单位或组织新增或更新至网路上的 CVE 列表和 GIT Hub 时，并不会通知本文所述的软件系统 1。因此代码分析事件要能够被触发，即表示在触发前，软件系统 1 应依据装置代码 80 的组成清单，向 Linux Kernel、OpenSSL、sqlite 等 CVE 列表和 GIT Hub 进行监控，以侦测这些代码分析事件。

这样带来的技术效果是，该软件系统 1 可以在一漏洞和补丁代码被新增或更新至网络上的风险代码登录列表（例如：CVE 列表）时，即时地完成装置代码 80 的代码安全检测工作。

进一步的，所述组成清单是在所述装置代码 80 被提供至所述软件系统 1 时，由所述软件系统 1 对所述装置代码 80 的成份组成进行扫描及/或剖析后所得。

装置代码 80 可以是单独的一个 .o 档（目的档）、.img 档（系统映像档）、.so 档（程序库），或者是一个 .zip or .gz 压缩档（内容包含整个装置所需的 .o 和 .so 档，并且压缩档中也保有该些档案的路径

结构)。

以一个装置 8 的整体装置代码 80 来说，其中所包含的软件代码的档案是很多的，各软件代码以不同组件（例如：Linux Kernel、OpenSSL、sqlite 等）的型态，组合在装置代码 80 中。各软件代码的档案会存在不同的路径位置。有不同方式可以取得装置代码 80 所包含的软件代码的组成清单，一种是针对档案路径的信息去扫描整个装置代码 80，作为不同组件的软件代码，其相关档案会放在不同的位置，如此可基本确定出装置代码 80 中含有哪些组件。另外，也可以去读取各软件代码的档案的内容，比对看看该些档案内容中是否存在特定组建的一些特征来掌握组件清单，例如在读取一软件代码所含的代码档案中的代码组成内容时，可以和 Linux Kernel、OpenSSL、sqlite 等该些已知软件的已知代码片段做比对，依据比对结果，即可确定出装置代码 80 中含有那些组件。扫描及/或剖析装置代码 80 的成份组成以取得装置代码 80 中各软件代码的组成清单之方法，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

于一较佳实施例中，补丁代码的代码组成内容包含一 Javascript 所编写，内含 Javascript 代码的档案。该补丁代码的作用在于修正装置代码 80 所包含的一软件代码的一漏洞。软件系统 1 中的比对单元，首先可藉由所述 Javascript 档案的档名，定位出该档名所对应的 Javascript 档案是在该软件代码的代码组成内容中的哪个位置。分别对该补丁代码的 Javascript 档案和该软件代码的相对应 Javascript 档案的内容进行读取后，就该两个档案所含的成份内容进行比对，亦即，比对两个档案的内容中，是否包含修正后的代码片段，以修正后的代码片段作为比对的成份特征，比对结果相符表示该软件代码不含该漏洞，不相符表示该软件代码含有该漏洞。

于一较佳实施例中，比对用的成份特征，除了是以代码中所含的片段作为特征外，亦可以补丁代码和软件代码在内存中的储存结构及/或补丁代码和软件代码的环境设置及组态信息来作为比对用的成份特征。例如：一补丁代码的代码组成内容中，新增了一个档案，而这个档案的档名，并不存在于相对应软件代码的代码组成内容中，及/或补丁代码在环境设置和组态上加入了一新的参数，而此参数不存在于相对应软件代码的环境设置和组态里，即可谓所述软件代码的成份内容里，并不包含该补丁代码，因而所述软件代码中仍存在着该补丁代码所对应的漏洞。

依上例，用于比对的补丁代码的代码片段、文件名结构及/或组态内容是成份内容比对里的第一成份特征。用于比对的软件代码的代码片段、文件名结构及/或组态内容是成份内容比对里的第二成份特征。通过剖析代码组成内容而得出的成份特征进行比对的方法，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

于一较佳实施例中，一软件代码的代码组成内容包含一内含二进制代码的第一档案，而所述第一档案的档名及其于储存结构中的相对位置可对应到一补丁代码所含的内含以 C/C++ 源代码编写的第二档案。所述第二档案包含用于修正一漏洞的代码。在此情况下，先将所述第一档案反汇编成第一中间代码（以 LLVM 的 IR 为中间代码表示形式的代码），亦将第二档案进行编译后，再反汇编成第二中间代码（以 LLVM 的 IR 为中间代码表示形式的代码）。进而即可针对所述第一中间代码和所述第二中间代码进行模拟执行，亦即通过读取所述第一中间代码和所述第二中间代码中的程序指令，喂入仿真的参数（parameter）后，依据所述读取的程序指令进行逻辑运算。当所有程序指令都被执行过后，即可依据程序指令的执行过程

中各指令的执行历程、函数呼叫的输出、输入及/或执行顺序来得到模拟执行过程中所显露出的语义信息；其中，本文所述的「语义信息」包括程序运行时的内存及/或中间变量的取值，及/或模拟执行过程中调用的指令、函数及其输出、输入值。对第一中间代码语义信息和第二中间代码语义信息的进行比对而得出一相似度，即可依所述相似度决定出第一中间代码和第二中间代码的代码片段是否相符，进而判断所述第一档案是否含有第二档案中用于修正所述漏洞的代码。其中，第一中间代码语义信息和第二中间代码语义信息全部内容所构成的成份特征，完全包含于彼此的语义信息中，即可谓相似度为 100%，仅一部分内容所构成的成份特征的包含于彼此的语义信息中，即可为所述部分所占全部内容的比例，即为其相似度，据此可得到 0%至 100%的相似度结果，相似度超过所设定的一阈值即可表示第一档案包含第二档案的代码。所述阈值是用于判断第二档案的代码是否包含于第一档案的参考数值，例如：所述阈值设定为 90%，即表示当相似度高于 90%，则第一档案包含第二档案的代码，反之则不包含。

于一较佳实施例中，一软件代码的代码组成内容包含一内含中间码（Java bytecode）的第一档案，而所述第一档案的档名及其于储存结构中的相对位置可对应到一补丁代码所含的内含以 Java 源代码编写的第二档案。所述第二档案包含用于修正一漏洞的代码。在此情况下，将所述第二档案编译为中间码（Java bytecode）后，透过模拟执行的来取得其语义信息，进而得以透过语义信息里所含从内存及/或中间变量取出的值，及/或模拟执行过程中调用的指令、函数及其输出、输入值的成份特征来比对出所述第一档案是否含有所述第二档案中用于修正所述漏洞的代码。

以上所述以模拟执行得到的语义信息的成份特征进行比对的方法，并不以此限，熟知本技艺人士可依据实际需求而进行任何均等的变更设

计。

S3 中所述发出风险预警夹带一指令，藉由所述指令来指示所述装置 8 及/或操作所述装置 8 的一用户，于所述装置 8 上执行所述风险代码的更新工作；或者所述风险预警的通知夹带一提示信息，藉由所述提示信息为所述第一用户 6、所述第二用户 7，及/或所述系统管理者提供对所述至少一安全风险的预警。

这个技术效果在于，风险预警通知可夹带一指令，在装置 8 的安全风险出现时，直接透过该指令，让该装置 8 对该风险代码实施即时的更新，即时填补该漏洞。之所以我方技术方案可以实现这个效益，原因是在于我方所针对的是二进制码的成份分析，也就是我方在检测出装置代码 80 中出现漏洞时，就已经有了该漏洞的补丁的二进制代码（风险代码），所以可以通知装置 8 即时的更新该风险代码。

如图 5 所示，本实施例还提供一种装置代码风险预警运算装置 11，包括收集单元 21、取码单元 31、比对单元 41、传输单元 51；

所述收集单元 21 用于接收装置代码 80 的更新请求；

所述取码单元 31 用于从一储存媒体 9 取得所述装置代码 80，所述装置代码 80 包含至少一软件代码；

所述传输单元 51 用于将所述装置代码 80 传输至一装置 8 和发出风险预警通知。

所述比对单元 41 用于，相对应于一风险代码的一代码分析事件触发时，取码单元 31 从所述储存媒体 9 及/或另一储存媒体 9 取得所述风险代码，对所述风险代码和所述装置代码 80 进行比对。

在上述比对若比对结果显示所述至少一软件代码包含至少一安全风

险时，所述传输单元 51 还用于发出预警通知，传输单元 51 透过至少一网络通信渠道，向所述装置代码 80 的提供者、所述装置 8 的运营者、管理者中至少一人，发出一风险预警的通知。

于一较佳实施例中，所述运算装置 11 是一嵌入式装置。所述收集单元 21、取码单元 31、比对单元 41 及/或传输单元 51，是以硬件描述语言所设计而可被运算装置 11 的集成电路单元或者是以软件代码的形式集成至运算装置 11 的嵌入式系统中。通过运行于运算装置 11 里的嵌入式系统对收集单元 21、取码单元 31、比对单元 41 和传输单元 51 进行驱动。

于另一较佳实施例中，所述运算装置 11 是一伺服器，所述收集单元 21、取码单元 31、比对单元 41 及/或传输单元 51 是以软件代码的形式集成至一软件系统 1，所述软件系统 1 通过内建或安装的方式，提供至运算装置 11 中运行，通过运行于运算装置 11 里的软件系统 1 对收集单元 21、取码单元 31、比对单元 41 和传输单元 51 进行驱动。

于另一较佳实施例中，所述运算装置 11 是由分散的伺服器所构成，所述收集单元 21、取码单元 31、比对单元 41 及/或传输单元 51 是以软件代码的形式集成至一分散式软件系统 1，所述分散式软件系统 1 部署至所述分散的伺服器中运行，组成一运算装置 11。通过运行于运算装置 11 里的分散式软件系统 1 对收集单元 21、取码单元 31、比对单元 41 和传输单元 51 进行驱动。

上述以不同形式的硬件和软件结合组成运算装置 11 之方式，并不以此为限，熟知本技艺人士可依据实际应用需求而进行任何均等的变更设计。

进一步的，结合图 3，所述比对单元 41 包括第一特征提取单元、第二特征提取单元和特征比对单元；

所述第一特征提取单元，用于撷取相对应于所述风险代码的至少一第一成份特征；

所述第二特征提取单元，用于撷取自相对应于所述至少一软件代码的至少一第二成份特征；

所述特征比对单元，用于对至少一第一成份特征和至少一第二成份特征进行比对。

再进一步的，所述第一特征提取单元或第二特征提取单元包括至少一剖析单元和至少一撷取单元；

至少一剖析单元，用于对相对应于所述风险代码的代码组成内容及/或代码执行结果进行剖析后得到第一成份特征，及/或对相对应于所述至少一软件代码的代码组成内容及/或代码执行结果进行剖析后得到第二成份特征；

至少一撷取单元，用于撷取剖析后所得到的第一成份特征或第二成份特征。

结合图 4，所述一种装置代码风险预警运算装置还包括至少一侦测单元；

所述至少一侦测单元，用于依据所述装置代码 80 所包括的所述至少一软件代码的组成清单，对所述组成清单中所含各个软件代码所相对应的风险代码登录列表（例如：CVE 列表）进行监控，藉由监控所述风险代码登录列表的异动来侦测代码分析事件。

综上，本发明公开一种装置代码风险预警方法、软件系统 1 和运算装置 11，通过对一在线运行的装置 8 进行包含至少一软件代码的装置代码

80 的更新、对风险代码和所述装置代码 80 成份内容的比对，以及为所述装置 8 发出风险预警，无论所述装置 8 是否离线，网路状态是否正常，也能对所述装置 8 进行风险预警，据以实现对装置代码 80 的风险防控。

上列详细说明是针对本发明可行实施例的具体说明，所述实施例并非用以限制本发明的保护范围，凡未脱离本发明所为的等效实施或变更，均应包含于本发明的保护范围中。

权 利 要 求 书

1、一种装置代码风险预警方法，其特征在于：包括以下步骤：

收集一装置代码的一更新请求，依据所述更新请求的内容，取得所述装置代码，并将所述装置代码传输至一装置；所述装置代码包含至少一软件代码；

当相对应于一风险代码的一代码分析事件触发时，取得所述风险代码，对所述风险代码和至少一所述软件代码进行成份内容的比对；

当所述比对的结果显示出所述至少一软件代码包含至少一安全风险时，发出风险预警通知。

2、根据权利要求1所述的装置代码风险预警方法，其特征在于：所述发出风险预警通知，是指透过至少一网路通信渠道，向所述装置代码的提供者、所述装置的运营者、管理者中至少一人，发出一风险预警的通知。

3、根据权利要求1所述的装置代码风险预警方法，其特征在于：所述装置代码是由一第一用户所提供，所述更新请求的收集是针对所述装置代码的发布及/或更新而接收；所述装置是由一第二用户所运营。

4、根据权利要求1所述的装置代码风险预警方法，其特征在于：所述至少一软件代码包含至少一二进制代码。

5、根据权利要求1所述的装置代码风险预警方法，其特征在于：所述取得装置代码具体为，从一储存媒体取得所述装置代码，包括：

所述装置代码是经由一代码版本信息所指定，依据所述代码版本信息从所述储存媒体取得所述装置代码。

6、根据权利要求5所述的装置代码风险预警方法，其特征在于：

所述代码版本信息是由所述更新请求的内容中直接指定，或者是依据所述更新请求的内容中所包含的一产品规格、一硬体规格，及/或一软韧体规格，而来决定出所述装置所需更新的所述代码版本信息。

7、根据权利要求1所述的装置代码风险预警方法，其特征在于：所述更新请求是经由网路传输的线上更新请求。

8、根据权利要求 1 所述的装置代码风险预警方法，其特征在于：所述更新请求的内容包含所述装置的网路位置信息，依据所述网路位置信息，经由网路将所述装置代码传输至所述装置。

9、根据权利要求 1 所述的装置代码风险预警方法，其特征在于：所述风险代码和所述至少一软件代码进行成份内容的比对，包括：

撷取相对应于所述风险代码的至少一第一成份特征；

撷取相对应于所述至少一软件代码的至少一第二成份特征；

比对所述至少一第一成份特征和所述至少一第二成份特征。

10、根据权利要求 9 所述的装置代码风险预警方法，其特征在于：所述第一成份特征是藉由对相对应于所述风险代码的代码组成内容及/或代码执行结果进行剖析后所得。

11、根据权利要求 10 所述的装置代码风险预警方法，其特征在于：所述代码组成内容包括所述风险代码及/或所述风险代码编译后的第一中间代码，所述代码执行结果则包括对所述风险代码及/或所述第一中间代码做模拟执行时，所述风险代码及/或所述第一中间代码在模拟执行过程中所显露出的语义信息。

12、根据权利要求 9 所述的装置代码风险预警方法，其特征在于：所述至少一第二成份特征是藉由对相对应于所述至少一软件代码的至少一代码组成内容及/或至少一代码执行结果进行剖析后所得。

13、根据权利要求 12 所述的装置代码风险预警方法，其特征在于：所述至少一代码组成内容包括至少一二进制代码及/或所述至少一二进制代码反组译后的至少一第二中间代码，所述至少一代码执行结果则包括对所述至少一二进制代码及/或所述至少一第二中间代码做模拟执行时，所述至少一二进制代码及/或所述至少一第二中间代码在模拟执行过程中所显露出的语义信息。

14、根据权利要求 11 或 13 所述的装置代码风险预警方法，其特征在于：

所述中间代码的程序语言结构是由一中间语言、一中间表示、一结构化语言，及/或一结构化协议所宣告而成。

15、根据权利要求 1 所述的装置代码风险预警方法，其特征在于：所述风险代码的种类是包括一补丁代码、恶意程序代码，及/或任一种用于修正或产生一软件漏洞或软件病毒的代码。

16、根据权利要求 1 所述的装置代码风险预警方法，其特征在于：相对应于所述风险代码的所述代码分析事件是在所述风险代码被新增或被更新时所触发。

17、根据权利要求 1 所述的装置代码风险预警方法，其特征在于：所述代码分析事件触发前，依据所述至少一软件代码的一组成清单，对至少一代码分析事件进行侦测，相对应于所述风险代码的所述代码分析事件是所述至少一代码分析事件的其中之一。

18、根据权利要求 17 所述的装置代码风险预警方法，其特征在于：所述组成清单是在所述装置代码被收集时，对所述装置代码的成份组成进行扫描及/或剖析后所得。

19、根据权利要求 18 所述的装置代码风险预警方法，其特征在于：所述组成清单包括所述装置代码本身的元数据及/或所述装置代码成份组成中所含的所述至少一软件代码的元数据。

20、根据权利要求 19 所述的装置代码风险预警方法，其特征在于：所述至少一软件代码是以包括系统核心、驱动程序、应用程序、组件，及/或类别库的型态存在于所述装置代码的成份组成中。

21、根据权利要求 20 所述的装置代码风险预警方法，其特征在于：所述装置代码是一系统映像档，对系统映像档中所含的成份组成进行扫描及/或剖析，辨识出存在于所述系统映像档中的所述至少一软件代码，进而得到所述至少一软件代码的所述组成清单；

或者所述装置代码是一软件代码包裹，对所述软件代码包裹中所含的成

份组成进行扫描及/或剖析，辨识出存在于所述软件代码包裹中的所述至少一软件代码，进而得到所述至少一软件代码的所述组成清单。

22、根据权利要求 2 所述的装置代码风险预警方法，其特征在于：所述风险预警的通知夹带一指令，藉由所述指令来指示所述装置及/或操作所述装置的用户，于所述装置上执行所述风险代码的更新工作；或者所述风险预警的通知夹带一提示信息，藉由所述提示信息为所述装置代码的提供者、所述装置的运营者及/或管理者提供对所述至少一安全风险的预警。

23、一种装置代码风险预警的软件系统，其特征在于：包括收集单元、取码单元、比对单元、传输单元；

所述收集单元用于接收装置代码的更新请求；

所述取码单元用于取得所述装置代码，所述装置代码包含至少一软件代码；

所述传输单元用于将所述装置代码传输至一装置和发出风险预警通知；

所述比对单元用于，相对应于一风险代码的一代码分析事件触发时，取码单元取得所述风险代码，对所述风险代码和所述装置代码进行比对。

24、根据权利要求 23 所述的装置代码风险预警的软件系统，其特征在于：所述比对单元包括第一特征提取单元、第二特征提取单元和特征比对单元；

所述第一特征提取单元，用于撷取相对应于所述风险代码的至少一第一成份特征；

所述第二特征提取单元，用于撷取自相对应于所述至少一软件代码的至少一第二成份特征；

所述特征比对单元，用于对至少一第一成份特征和至少一第二成份特征进行比对。

25、根据权利要求 24 所述的装置代码风险预警的软件系统，其特征在于：所述第一特征提取单元或第二特征提取单元包括至少一剖析单元和

至少一撮取单元；

所述至少一剖析单元，用于对相对应于所述风险代码的代码组成内容及/或代码执行结果进行剖析后得到第一成份特征，及/或对相对应于所述至少一软件代码的代码组成内容及/或代码执行结果进行剖析后得到第二成份特征；

所述至少一撮取单元，用于撮取剖析后所得到的第一成份特征及/或第二成份特征。

26、根据权利要求 23 所述的装置代码风险预警的软件系统，其特征在于：还包括至少一侦测单元，

所述至少一侦测单元，用于依据所述至少一软件代码的组成清单，对所述组成清单中所含各个软件代码所相对应的风险代码登录列表进行监控，藉由监控所述风险代码登录列表的异动来侦测代码分析事件。

27、一种装置代码风险预警的运算装置，其特征在于：包括收集单元、取码单元、比对单元、传输单元；

所述收集单元用于接收装置代码的更新请求；

所述取码单元用于从一储存媒体取得所述装置代码，所述装置代码包含至少一软件代码；

所述传输单元用于将所述装置代码传输至一装置和发出风险预警通知；

所述比对单元用于，相对应于一风险代码的一代码分析事件触发时，取码单元从所述储存媒体及/或另一储存媒体取得所述风险代码，对所述风险代码和所述装置代码进行比对。