

移动端安全切面的按需开启方法及装置

技术领域

[01] 本发明涉及移动端安全切面领域，更具体地涉及一种移动端安全切面的按需开启方法、装置、可读存储介质及电子设备。

背景技术

[02] 安全切面（即安全平行切面）是指通过采用面向切面编程（Aspect-oriented Programming, AOP）的方式，在不修改业务应用程序的情况下，在业务应用程序的运行逻辑中动态的添加或修改用于实现安全切面业务的切面程序的方法，使得在实现安全切面业务的同时，使实现安全切面业务的程序与业务应用程序解耦，从而避免高耦合导致的开发迭代问题。

[03] 移动端安全切面是指在移动端运行的安全切面，其需要与切面服务器进行数据通信。在移动端的应用场景下，一个重量级业务应用程序可能会有上亿级别的客户端日活。如果这些移动设备上全部开启安全切面，对切面服务器带来的流量和存储压力是非常巨大的。

[04] 实际上，在业务应用程序运行时，很多移动端并没有被攻击的风险，若在这些移动设备上开启安全切面，会加重切面服务器的负担，造成不必要的资源浪费。

发明内容

[05] 本发明的目的之一在于提供一种移动端安全切面的按需开启方法，在业务应用程序开启后，先判断移动端是否存在被攻击的风险，并根据判断结果决定移动端安全切面继续运行或退出，从而实现按需开启，避免不必要的资源浪费。

[06] 基于上述目的，本发明提供一种移动端安全切面的按需开启方法，其包括以下步骤：在移动端安全切面随移动端的业务应用程序开启后，判断所述移动端是否存在被攻击的风险；若是，则根据所述移动端被攻击的方式获取所述业务应用程序中被攻击的函数作为切点，并通过所述移动端安全切面的切面底座在所述切点注入切面程序，以通过注入的所述切面程序执行安全切面业务；否则，使所述移动端安全切面退出。

[07] 进一步地，在一些实施方式中，判断当前移动端是否存在被攻击的风险，具体包括：判断所述移动端是否越狱/ROOT；和/或，判断所述业务应用程序是否被注入/HOOK；

和/或，判断所述业务应用程序是否被调试。

[08]进一步地，在一些实施方式中，判断所述移动端是否越狱/ROOT，具体包括：检测所述移动端的内存中是否存在用于存储配置信息的高风险文件，若是，则所述移动端有越狱/ROOT；否则所述移动端没有越狱/ROOT。

[09]进一步地，在一些实施方式中，判断所述业务应用程序是否被注入/HOOK，具体包括：遍历所述业务应用程序中的函数的入口地址，然后检测函数的入口地址是否与所述业务应用程序在同一个内存地址空间中，若是，则所述业务应用程序没有被注入/HOOK；否则，所述业务应用程序被注入/HOOK。

[10]进一步地，在一些实施方式中，判断所述业务应用程序是否被调试，具体包括：读取内存中控制调试开关的对象或文件的值，若是 true 或 1，则所述业务应用程序被调试，若是 false 或 0，则所述业务应用程序没有被调试。

[11]进一步地，在一些实施方式中，根据所述移动端被攻击的方式获取被攻击的函数作为切点，具体包括：获取所述业务应用程序中的用于操作所述移动端越狱/ROOT 后留下的高风险文件的函数作为所述切点；和/或，获取所述业务应用程序中被注入/HOOK 的函数作为所述切点；和/或，获取所述业务应用程序中被调试的函数作为所述切点。

[12]本发明的另一目的在于提供一种移动端安全切面的按需开启装置，在业务应用程序开启后，先判断移动端是否存在被攻击的风险，并根据判断结果决定移动端安全切面开启或退出，从而实现按需开启，避免不必要的资源浪费。

[13]基于上述目的，本发明提供一种移动端安全切面的按需开启装置，包括：判断模块，设置为在移动端安全切面随移动端的业务应用程序开启后，判断所述移动端是否存在被攻击的风险；注入模块，设置为在所述判断模块的判断结果为是时，根据所述移动端被攻击的方式获取被攻击的函数作为切点，并通过所述移动端安全切面的切面底座在所述切点注入切面程序，以通过注入的所述切面程序执行安全切面业务；退出模块，设置为在所述判断模块的判断结果否时，使所述移动端安全切面退出。

[14]进一步地，在一些实施方式中，判断当前移动端是否存在被攻击的风险，具体包括：判断所述移动端是否越狱/ROOT；和/或，判断所述业务应用程序是否被注入/HOOK；和/或，判断所述业务应用程序是否被调试。

[15]进一步地，在一些实施方式中，判断所述移动端是否越狱/ROOT，具体包括：检测所述移动端的内存中是否存在用于存储配置信息的高风险文件，若是，则所述移动端有

越狱/ROOT；否则所述移动端没有越狱/ROOT。

[16]进一步地，在一些实施方式中，判断所述业务应用程序是否被注入/HOOK，具体包括：遍历所述业务应用程序中的函数的入口地址，然后检测函数的入口地址是否与所述业务应用程序在同一个内存地址空间中，若是，则所述业务应用程序没有被注入/HOOK；否则，所述业务应用程序被注入/HOOK。

[17]进一步地，在一些实施方式中，判断所述业务应用程序是否被调试，具体包括：读取内存中控制调试开关的对象或文件的值，若是 true 或 1，则所述业务应用程序被调试，若是 false 或 0，则所述业务应用程序没有被调试。

[18]进一步地，在一些实施方式中，根据所述移动端被攻击的方式获取被攻击的函数作为切点，具体包括：获取所述业务应用程序中的用于操作所述移动端越狱/ROOT 后留下的高风险文件的函数作为所述切点；和/或，获取所述业务应用程序中被注入/HOOK 的函数作为所述切点；和/或，获取所述业务应用程序中被调试的函数作为所述切点。

[19]本发明又一目的在于提供一种可读存储介质，其上存储有计算机程序，当计算机程序在计算机中执行时，令计算机执行如上所述的移动端安全切面的按需开启方法的步骤。

[20]本发明又一目的在于提供一种电子设备，其包括存储器和处理器，存储器中存储有可执行代码，当处理器执行可执行代码时，其执行如上所述的移动端安全切面的按需开启方法的步骤。

附图说明

[21]图 1 为根据本发明实施例的移动端安全切面的按需开启方法的流程图；

[22]图 2 为根据本发明另一实施例的移动端安全切面的按需开启装置的结构框图。

具体实施方式

[23]下面结合附图，给出本发明的较佳实施例，并予以详细描述。

[24]安全切面是指通过采用面向切面编程 (Aspect-oriented Programming, AOP) 的方式，在不修改业务应用程序的情况下，在业务应用程序的运行逻辑中动态的添加或修改用于实现安全切面业务的切面程序的方法。使得在实现安全切面业务的同时，使实现安全切面业务的程序与业务应用程序解耦，从而避免高耦合导致的开发迭代问题。

[25]这里说的切面程序，即基于业务运行逻辑来实现安全切面业务的增强程序。可通过采用面向切面编程的方式，将切面程序注入至业务应用程序的相应切点处，该切面程序在执行业务应用程序的过程中被触发执行，实现所需的安全切面业务功能。

[26]而业务应用程序在执行业务时，业务应用程序通常通过方法之间的调用进行业务执行。因此，可将业务应用程序中的任意方法作为切面程序的切入点，即上述所说的切点，并将切面程序注入对应切点。使得业务应用程序在执行至切点，即调用该切点对应的业务应用程序的方法时，执行在该切点注入的切面程序。

[27]通常，负责将切面程序注入至切点的过程的代码存在较高的可复用性，因此，通常将实现这一过程的程序抽象为一个服务模块，即切面底座。切面底座可从提供安全切面业务的服务器获取需要部署的切面程序以及获取切面程序在业务应用程序中注入的切点，在应用容器启动后被唤醒并在业务应用程序的切点处注入对应的切面程序。

[28]业务应用程序可以是业务平台的服务器中提供业务服务的业务应用程序。其中，业务服务可以是业务平台的服务器向用户提供的业务服务，如，查询业务、支付业务等。业务服务还可以是业务平台的某服务器向其他服务器提供的业务服务，如，结算业务等。

[29]当然了，由上述描述可知，为了使安全切面业务的程序与业务应用程序解耦，本说明书通过采用面向切面编程的方式使得安全切面业务的程序与业务应用程序在业务执行时相互交织，但又互相平行可各自独立维护。因此，区别于业务应用程序的业务提供方，提供安全切面业务的第三方可通过服务器管理安全切面业务所涉及的内容，例如，安全切面业务管控策略的配置、切面程序的版本迭代、切面程序的部署规则配置等等。当然，提供安全切面业务的可以是第三方，也可以是业务提供方。

[30]服务器在管理安全切面业务所涉及的内容时，可通过配置文件记录各种配置信息，例如各种策略的配置、切面程序的部署规则配置等。使得切面底座可根据配置文件完成对切面程序的部署，或者服务器可根据配置文件实现安全切面业务。

[31]在实际应用中，业务提供方通常设置有包括若干物理机或物理服务器的机房，通过物理机提供业务应用程序所需的物理资源。当然，一个业务应用程序可能无需整个物理机的所有物理资源，于是，一般还会通过虚拟化技术在一个物理机上运行多个虚拟主机（virtual hosting）。各个虚拟主机之间是互相独立的，各自享有物理机的部分物理资源。然后，可在虚拟主机中部署应用容器，并通过应用容器运行业务应用程序。应用容器通常包含分配给该应用容器的物理资源，如 CPU、内存等，以及提供给该应用容器的运行

环境,如操作系统(Operating System, OS)或其他运行环境数据,如,容器的序列(Serial Number, SN)号、分配的IP(Influential Property)、应用名、租户、环境变量等)。业务应用程序可部署在应用容器中以执行业务。

[32]而在基于安全切面执行业务的场景中,业务提供方或者提供安全切面业务的第三方的服务器可提供服务器,通过服务器管理安全切面业务所涉及的内容,并将切面底座部署在应用容器中,通过切面底座将切面程序注入业务应用容器中的业务应用程序中,以为业务提供方的应用容器提供安全切面业务的支持。

[33]于是,可预先在业务提供方的应用容器中部署切面底座。一般情况下,在启动应用容器时,可唤起提供给应用容器的操作系统,以及运行预先部署的切面底座,通过切面底座从服务器中得到切面程序和业务应用程序的切点,并将切面程序注入应用容器中的业务应用程序的切点处。此外,切面底座也可在业务应用程序执行过程中,从服务器中得到切面程序和业务应用程序的切点,并将切面程序注入应用容器中的业务应用程序的切点处。

[34]当然,切面底座如何从服务器中得到部署切面程序所需的信息,可根据需要设置。例如,可以是根据配置文件主动从服务器中拉取所需的信息,或者服务器可主动下发使切面底座接收所需的信息。

[35]将切面程序注入业务应用程序的切点后,业务应用程序在执行过程中,即可触发该切面程序从而实现相应的安全切面业务功能。

[36]当业务应用程序被用户开启后,移动端安全切面也会跟随业务应用程序一起开启,移动端安全切面开启后,常规做法是切面服务器将默认的切面配置下发至移动端安全切面的切面底座,然后由切面底座根据默认的切面配置完成切点注入,即在业务应用程序的切点注入切面程序,在业务应用程序运行过程中,通过切面程序实现相应的安全切面业务功能。但是,对一个日活量很大的移动端的业务应用程序来说,若每个移动端的安全切面均在业务应用程序运行时保持运行,那对切面服务器带来的流量和存储压力是非常巨大的。实际上,很多移动端可能并没有处于风险状态,若所有移动端安全切面均运行,会造成不必要的资源浪费。因此,对于一些日活量比较大的业务应用程序,相关技术通常通过黑白名单的机制来随机确定某个移动端的安全切面是否在业务应用程序运行时保持运行。例如,黑名单中的移动端是没有风险的,其安全切面不必运行,而白名单的移动端可能是存在风险的,其安全切面需要保持运行。但是,黑白名单是预先确定并存储在切面服务器中的静态名单,一些移动端可能在确定黑白名单时不处于风险状态

(例如没有越狱/ROOT 等), 因此被归入黑名单中, 而在确定黑白名单后, 其又越狱/ROOT 了, 因此在业务应用程序运行时, 其安全切面应当运行, 但是由于在黑名单中, 并不会运行, 导致该移动端可能出现安全问题; 同理, 在确定白名单时, 某移动端可能是处于风险状态的, 但是在业务应用程序运行时, 其风险已消除, 因此并不需要安全切面运行, 这样就会导致资源的浪费。

[37]为此, 本发明实施例提供一种移动端安全切面的按需开启方法, 在业务应用程序开启后, 先对移动端进行风险检测, 当其存在被攻击的风险时, 再根据攻击方式进行切点注入, 从而通过移动端安全切面对攻击者的行为进行监测并实现相应的安全切面业务功能; 当不存在被攻击的风险时, 则退出移动端安全切面, 从而减小切面服务器的压力, 避免造成不必要的资源浪费。

[38]如图 1 所示, 本发明实施例的移动端安全切面的按需开启方法包括以下步骤 100 至步骤 300。

[39]步骤 100: 在移动端安全切面随移动端的业务应用程序开启后, 判断移动端是否存在被攻击的风险, 若是则执行步骤 200, 否则执行步骤 300。

[40]移动端安全切面会随着业务应用程序的开启而开启, 在开启后, 先判断移动端是否存在被攻击的风险, 从而确定业务应用程序的运行环境是否存在风险。在一些实施例中, 移动端被攻击的风险包括移动端越狱/ROOT 和/或业务应用程序被注入/HOOK 和/或业务应用程序被调试。若移动端越狱/ROOT 和/或业务应用程序被注入/HOOK 和/或业务应用程序被调试, 则说明移动端存在被攻击的风险。反之, 若移动端没有越狱/ROOT, 且业务应用程序既没有被注入/HOOK, 也没有被调试, 则说明移动端不存在被攻击的风险。因此, 判断移动端是否存在被攻击的风险主要就是判断移动端是否越狱/ROOT 和/或判断业务应用程序是否被注入/HOOK 和/或判断业务应用程序是否被调试。

[41]越狱/ROOT 是指获得移动端的系统的最高权限。移动端越狱/ROOT 后, 攻击者可以获得用户的任何信息, 或者对移动端进行任何操作, 造成隐私泄露等风险。移动端越狱/ROOT 后, 攻击者会在移动端的内存中留下很多高风险的文件用于存储相关配置信息, 因此, 可以通过检测这些高风险文件是否存在来判断移动端是否越狱/ROOT。具体地, 当检测到移动端的内存中存在高风险文件, 则说明移动端有越狱/ROOT, 反之, 则说明移动端没有越狱/ROOT。

[42]HOOK 又被称为钩子, 是一种可以实现替换函数调用入口地址的技术。注入是指攻

击者将自己实现的三方库注入到业务应用程序的进程中。攻击者通常会结合使用注入和HOOK技术将目标函数的入口地址替换到攻击者注入的三方库的函数地址上，从而改变业务应用程序中原有的函数功能。为了检测业务应用程序是否存在被注入/HOOK的风险，可以遍历业务应用程序中的所有函数（或关键函数）的入口地址，然后检测入口地址是否与业务应用程序在同一个内存地址空间，如果某个函数的入口地址不与业务应用程序在同一个内存地址空间，而是被修改到了攻击者的三方库的内存地址空间，那么说明该函数被注入/HOOK了，即业务应用程序被注入/HOOK。若所有遍历的函数的入口地址均与业务应用程序在同一个内存地址空间，说明没有函数被注入/HOOK，即业务应用程序没有被注入/HOOK。

[43]业务应用程序在运行过程中，还可能存在被调试的风险，若业务应用程序被调试，攻击者可以利用GDB、IDA、Ptrace等调试器跟踪运行的目标程序，查看、修改内存中的代码和数据，甚至分析篡改程序的逻辑，对用户关键数据或者服务器进行恶意攻击，例如修改用户端业务操作的数据，比如转账账号、金额等，导致用户的损失。因此，业务应用程序开启后，需要判断业务应用程序是否被调试，以决定移动端安全切面是否保持运行。具体地，可通过读取移动端的内存中控制调试开关的对象或文件的值来确定业务应用程序是否被调试。若控制调试开关的对象或文件的值为ture（或1），说明业务应用程序被调试了，若控制调试开关的对象或文件的值为false（或0），说明业务应用程序没有被调试。

[44]可以理解的是，业务应用程序在运行时还可能存在其他被攻击的风险，本发明的方法并不仅限于上述几种风险的检测，而是可以根据需要进行扩展，即针对业务应用程序其他任意可能存在的风险进行检测，若存在被攻击的风险，则执行步骤200，否则，执行步骤300。

[45]步骤200：根据移动端被攻击的方式获取业务应用程序中被攻击的函数作为切点，并通过移动端安全切面的切面底座在切点注入切面程序，以通过注入的切面程序执行安全切面业务。

[46]当判断出移动端存在被攻击的风险时，说明业务应用程序处于风险状态，需要使移动端安全切面保持运行以进行监测并提供相应的安全切面业务功能。在一些实施例中，判断出移动端存在被攻击的风险后，移动端可以将该信息上传至切面服务器，切面服务器收到该信息后可将默认的切面配置文件下发至移动端安全切面的切面底座，然后由切面底座根据默认的切面配置文件注入切面程序，从而在业务应用程序运行过程中通过切

面程序实现安全切面业务功能。但是，由于切面配置文件是默认的，其包含的切点可能与业务应用程序被攻击的函数并不相同，例如切面配置文件中并不是以被攻击的函数作为切点，或者切面配置中除了以被攻击的函数作为切点，还有其他没有被攻击的函数作为切点，这样增加了切点的数量，从而增加服务器的压力。因此，在另外一些实施例中，可以直接根据移动端被攻击的方式获取业务应用程序中被攻击的函数作为切点，然后通过切面底座在这些切点注入切面程序，以监测攻击者的行为，并提供相应的安全切面业务功能。

[47]具体地，若移动端越狱/ROOT了，那么攻击者会在内存中留下用户存储配置信息的高风险文件，当业务应用程序在运行过程中对这些高风险件进行时，会被攻击者攻击。因此，可获取业务应用程序中用于操作这些高风险文件的函数（例如 `fopen`、`fread` 函数等）作为切点，然后通过切面底座在这些切点处注入切面程序。例如，可将这些切点上传至切面服务器，切面服务器再根据切点下放切面配置文件至切面底座，切面底座再根据切面配置在切点注入切面程序，从而对攻击者的行为进行细粒度的数据收集和上报。

[48]若业务应用程序被注入/HOOK，则可以获取业务应用程序中被注入/HOOK 的函数作为切点，然后通过切面底座将这些切点处注入切面程序。例如，可将这些切点上传至切面服务器，切面服务器再根据切点下放切面配置文件至切面底座，切面底座再根据切面配置在切点注入切面程序，从而对攻击者的行为进行细粒度的数据收集和上报。

[49]若业务应用程序被调试，则可以获取业务应用程序中被调试的函数作为切点，然后通过切面底座将这些切点处注入切面程序。例如，可将这些切点上传至切面服务器，切面服务器再根据切点下放切面配置文件至切面底座，切面底座再根据切面配置在切点注入切面程序，从而对攻击者的行为进行细粒度的数据收集和上报。

[50]步骤 300：使移动端安全切面退出。

[51]若判断出移动端没有被攻击的风险，说明业务应用程序不处于风险状态，在业务应用程序运行时，移动端安全切面不用保持运行。因此，当移动端没有被攻击的风险时，可以使移动端安全切面直接退出，避免占用切面服务器的资源。例如，当判断出移动端没有被攻击的风险时，可以将该信息上传至切面服务器，当切面服务器接收该信息时，会下发一个空的切面配置文件至切面底座，切面底座接收到空的切面配置文件后，移动端安全会直接退出。或者，也可以直接将该消息发送至移动端安全切面，移动端安全切面接收到该消息后，直接退出。

[52]本发明实施例的移动端安全切面的按需开启方法，在业务应用程序和移动端安全切面开启后先判断移动端是否存在被攻击的风险，在确定移动端存在被攻击的风险后才继续运行移动端安全切面，并根据被攻击的方式注入相关切点，否则使移动端安全切面退出，从而实现移动端安全切面的按需开启，避免造成不必要的资源浪费。

[53]如图 2 所示，本发明另一实施例提供一种移动端安全切面的按需开启装置，其包括判断模块 10、注入模块 20 和退出模块 30。

[54]判断模块 10 用于在移动端安全切面随移动端的业务应用程序开启后，判断所述移动端是否存在被攻击的风险。

[55]移动端安全切面会随着业务应用程序的开启而开启，在开启后，判断模块 10 可以先判断移动端是否存在被攻击的风险，从而确定业务应用程序的运行环境是否存在风险。在一些实施例中，移动端被攻击的风险包括移动端越狱/ROOT 和/或业务应用程序被注入/HOOK 和/或业务应用程序被调试。若移动端越狱/ROOT 和/或业务应用程序被注入/HOOK 和/或业务应用程序被调试，则说明移动端存在被攻击的风险。反之，若移动端没有越狱/ROOT，且业务应用程序既没有被注入/HOOK，也没有被调试，则说明移动端不存在被攻击的风险。因此，判断移动端是否存在被攻击的风险主要就是判断移动端是否越狱/ROOT 和/或判断业务应用程序是否被注入/HOOK 和/或判断业务应用程序是否被调试。

[56]移动端越狱/ROOT 后，攻击者会在移动端的内存中留下很多高风险的文件用于存储相关配置信息，因此，判断模块 10 可以通过检测这些高风险文件是否存在来判断移动端是否越狱/ROOT。具体地，当检测到移动端的内存中存在高风险文件，则说明移动端有越狱/ROOT，反之，则说明移动端没有越狱/ROOT。

[57]为了检测业务应用程序是否存在被注入/HOOK 的风险，判断模块 10 可以遍历业务应用程序中的所有函数（或关键函数）的入口地址，然后检测入口地址是否与业务应用程序在同一个内存地址空间，如果某个函数的入口地址不与业务应用程序在同一个内存地址空间，而是被修改到了攻击者的三方库的内存地址空间，那么说明该函数被注入/HOOK 了，即业务应用程序被注入/HOOK。若所有遍历的函数的入口地址均与业务应用程序在同一个内存地址空间，说明没有函数被注入/HOOK，即业务应用程序没有被注入/HOOK。

[58]业务应用程序开启后，判断模块 10 可以判断业务应用程序是否被调试，以决定移动

端安全切面是否保持运行。具体地，判断模块 10 可通过读取移动端的内存中控制调试开关的对象或文件的值来确定业务应用程序是否被调试。若控制调试开关的对象或文件的值为 true（或 1），说明业务应用程序被调试了，若控制调试开关的对象或文件的值为 false（或 0），说明业务应用程序没有被调试。

[59]可以理解的是，业务应用程序在运行时还可能不存在其他被攻击的风险，本发明的装置并不仅限于上述几种风险的检测，而是可以根据需要进行扩展，即判断模块 10 可针对业务应用程序其他任意可能存在的风险进行检测，判断其是否存在其他被攻击的风险。

[60]注入模块 20 用于在判断模块 10 的判断结果为是时，根据移动端被攻击的方式获取业务应用程序中被攻击的函数作为切点，并通过移动端安全切面的切面底座在切点注入切面程序，以通过注入的切面程序执行安全切面业务。

[61]当判断出移动端存在被攻击的风险时，说明业务应用程序处于风险状态，需要使移动端安全切面保持运行以进行监测并提供相应的安全切面业务功能。在一些实施例中，判断出移动端存在被攻击的风险后，判断模块 10 可以将判断结果发送至注入模块 20，注入模块 20 再将其上传至切面服务器，切面服务器收到该信息后可将默认的切面配置文件下发至移动端安全切面的切面底座，然后由切面底座根据默认的切面配置文件注入切面程序，从而在业务应用程序运行过程中通过切面程序实现安全切面业务功能。但是，由于切面配置文件是默认的，其包含的切点可能与业务应用程序被攻击的函数并不相同，例如切面配置文件中并不是以被攻击的函数作为切点，或者切面配置中除了以被攻击的函数作为切点，还有其他没有被攻击的函数作为切点，这样增加了切点的数量，从而增加服务器的压力。因此，在另外一些实施例中，注入模块 20 在接收到判断模块 10 的判断结果为是时，可以根据移动端被攻击的方式获取业务应用程序中被攻击的函数作为切点，然后将这些切点发送至切面底座或者先上传至切面服务器，再下发至切面底座，然后通过切面底座在这些切点注入切面程序，以监测攻击者的行为，并提供相应的安全切面业务功能。

[62]具体地，若移动端越狱/ROOT 了，那么攻击者会在内存中留下用户存储配置信息的高风险文件，当业务应用程序在运行过程中对这些高风险文件进行时，会被攻击者攻击。因此，注入模块 20 可获取业务应用程序中用于操作这些高风险文件的函数（例如 fopen、fread 函数等）作为切点，然后通过切面底座在这些切点处注入切面程序。例如，注入模块 20 可将这些切点直接发送至切面底座，由切面底座在切点注入切面程序；或者先上传至切面服务器，切面服务器再根据切点下发切面配置文件至切面底座，切面底座再

根据切面配置在切点注入切面程序，从而对攻击者的行为进行细粒度的数据收集和上报。

[63]若业务应用程序被注入/HOOK，注入模块 20 可以获取业务应用程序中被注入/HOOK 的函数作为切点，然后通过切面底座将这些切点处注入切面程序。例如，注入模块 20 可将这些切点直接发送至切面底座，由切面底座在切点注入切面程序；或者先上传至切面服务器，切面服务器再根据切点下发切面配置文件至切面底座，切面底座再根据切面配置在切点注入切面程序，从而对攻击者的行为进行细粒度的数据收集和上报。

[64]若业务应用程序被调试，注入模块 20 可以获取业务应用程序中被调试的函数作为切点，然后通过切面底座将这些切点处注入切面程序。例如，注入模块 20 可将这些切点直接发送至切面底座，由切面底座在切点注入切面程序；或者先上传至切面服务器，切面服务器再根据切点下发切面配置文件至切面底座，切面底座再根据切面配置在切点注入切面程序，从而对攻击者的行为进行细粒度的数据收集和上报。

[65]退出模块 30 在判断模块的判断结果是否为否时，使移动端安全切面退出。

[66]若判断出移动端没有被攻击的风险，说明业务应用程序不处于风险状态，在业务应用程序运行时，移动端安全切面不用保持运行。因此，当移动端没有被攻击的风险时，可以使移动端安全切面直接退出，避免占用切面服务器的资源。例如，当判断模块 10 判断出移动端没有被攻击的风险时，可以将为否的判断结果发送至退出模块 30，退出模块 30 再将其上传至切面服务器，当切面服务器接收到判断结果是否为否时，会下发一个空的切面配置文件至切面底座，切面底座接收到空的切面配置文件后，移动端安全会直接退出。或者，退出模块 30 也可以在接收到为否的判断结果时，直接使移动端安全切面退出。

[67]本发明实施例的移动端安全切面的按需开启装置，判断模块 10 在业务应用程序和移动端安全切面开启后先判断移动端是否存在被攻击的风险，在确定移动端存在被攻击的风险后注入模块 20 才使移动端安全切面继续运行，并根据被攻击的方式注入相关切点，否则退出模块 30 使移动端安全切面退出，从而实现移动端安全切面的按需开启，避免造成不必要的资源浪费。

[68]本发明的又一实施例提供一种可读存储介质，其上存储有计算机程序，当计算机程序在计算机中执行时，令计算机执行本发明上述实施例中的方法的步骤。

[69]本发明又一实施例提供一种电子设备，其包括存储器和处理器，存储器中存储有可执行代码，当处理器执行可执行代码时，其执行本发明上述实施例中的方法的步骤。

[70]上述实施例阐明的系统、装置、模块或单元，具体可以由计算机芯片或实体实现，或者由具有某种功能的产品来实现。一种典型的实现设备为计算机。具体的，计算机例如可以为个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任何设备的组合。

[71]为了描述的方便，描述以上装置时以功能分为各种单元分别描述。当然，在实施本发明时可以把各单元的功能在同一个或多个软件和/或硬件中实现。

[72]本领域内的技术人员应明白，本发明的实施例可提供为方法、系统、或计算机程序产品。因此，本发明可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且，本发明可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质（包括但不限于磁盘存储器、CD-ROM、光学存储器等）上实施的计算机程序产品的形式。

[73]本发明是参照根据本发明实施例的方法、设备（系统）、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器，使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[74]这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中，使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品，该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[75]这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上，使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理，从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[76]在一个典型的配置中，计算设备包括一个或多个处理器(CPU)、输入/输出接口、网络接口和内存。

[77]内存可能包括计算机可读介质中的非永久性存储器,随机存取存储器(RAM)和/或非易失性内存等形式,如只读存储器(ROM)或闪存(flash RAM)。内存是计算机可读介质的示例。

[78]计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括,但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带,磁带磁磁盘存储或其他磁性存储设备或任何其他非传输介质,可用于存储可以被计算设备访问的信息。按照本文中的界定,计算机可读介质不包括暂存电脑可读媒体(transitory media),如调制的数据信号和载波。

[79]还需要说明的是,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

[80]本领域技术人员应明白,本发明的实施例可提供为方法、系统或计算机程序产品。因此,本发明可采用完全硬件实施例、完全软件实施例或结合软件和硬件方面的实施例的形式。而且,本发明可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[81]本发明可以在由计算机执行的计算机可执行指令的一般上下文中描述,例如程序模块。一般地,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本发明,在这些分布式计算环境中,由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中,程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

[82]本发明中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实

实施例的部分说明即可。

[83]以上所述的，仅为本发明的较佳实施例，并非用以限定本发明的范围，本发明的上述实施例还可以做出各种变化。即凡是依据本发明申请的权利要求书及说明书内容所作的简单、等效变化与修饰，皆落入本发明专利的权利要求保护范围。本发明未详尽描述的均为常规技术内容。

权利要求书

1、一种移动端安全切面的按需开启方法，包括：

在移动端安全切面随移动端的业务应用程序开启后，判断所述移动端是否存在被攻击的风险；

若是，则根据所述移动端被攻击的方式获取所述业务应用程序中被攻击的函数作为切点，并通过所述移动端安全切面的切面底座在所述切点注入切面程序，以通过注入的所述切面程序执行安全切面业务；

否则，使所述移动端安全切面退出。

2、根据权利要求 1 所述的移动端安全切面的按需开启方法，判断当前移动端是否存在被攻击的风险，包括：

判断所述移动端是否越狱/ROOT；和/或，

判断所述业务应用程序是否被注入/HOOK；和/或，

判断所述业务应用程序是否被调试。

3、根据权利要求 2 所述的移动端安全切面的按需开启方法，判断所述移动端是否越狱/ROOT，包括：

检测所述移动端的内存中是否存在用于存储配置信息的高风险文件，若是，则所述移动端有越狱/ROOT；否则所述移动端没有越狱/ROOT。

4、根据权利要求 2 所述的移动端安全切面的按需开启方法，判断所述业务应用程序是否被注入/HOOK，包括：

遍历所述业务应用程序中的函数的入口地址，然后检测函数的入口地址是否与所述业务应用程序在同一个内存地址空间中，若是，则所述业务应用程序没有被注入/HOOK；否则，所述业务应用程序被注入/HOOK。

5、根据权利要求 2 所述的移动端安全切面的按需开启方法，判断所述业务应用程序是否被调试，包括：

读取内存中控制调试开关的对象或文件的值，若是 true 或 1，则所述业务应用程序被调试，若是 false 或 0，则所述业务应用程序没有被调试。

6、根据权利要求 2 所述的移动端安全切面的按需开启方法，根据所述移动端被攻击的方式获取被攻击的函数作为切点，包括：

获取所述业务应用程序中的用于操作所述移动端越狱/ROOT 后留下的高风险文件的函数作为所述切点；和/或，

获取所述业务应用程序中被注入/HOOK 的函数作为所述切点；和/或，

获取所述业务应用程序中被调试的函数作为所述切点。

7、一种移动端安全切面的按需开启装置，包括：

判断模块，设置为在移动端安全切面随移动端的业务应用程序开启后，判断所述移动端是否存在被攻击的风险；

注入模块，设置为在所述判断模块的判断结果为是时，根据所述移动端被攻击的方式获取被攻击的函数作为切点，并通过所述移动端安全切面的切面底座在所述切点注入切面程序，以通过注入的所述切面程序执行安全切面业务；

退出模块，设置为在所述判断模块的判断结果是否为否时，使所述移动端安全切面退出。

8、根据权利要求 7 所述的移动端安全切面的按需开启装置，判断当前移动端是否存在被攻击的风险，包括：

判断所述移动端是否越狱/ROOT；和/或，

判断所述业务应用程序是否被注入/HOOK；和/或，

判断所述业务应用程序是否被调试。

9、根据权利要求 8 所述的移动端安全切面的按需开启装置，判断所述移动端是否越狱/ROOT，包括：

检测所述移动端的内存中是否存在用于存储配置信息的高风险文件，若是，则所述移动端有越狱/ROOT；否则所述移动端没有越狱/ROOT。

10、根据权利要求 8 所述的移动端安全切面的按需开启装置，判断所述业务应用程序是否被注入/HOOK，包括：

遍历所述业务应用程序中的函数的入口地址，然后检测函数的入口地址是否与所述业务应用程序在同一个内存地址空间中，若是，则所述业务应用程序没有被注入/HOOK；否则，所述业务应用程序被注入/HOOK。

11、根据权利要求 8 所述的移动端安全切面的按需开启装置，判断所述业务应用程序是否被调试，包括：

读取内存中控制调试开关的对象或文件的值，若是 true 或 1，则所述业务应用程序被调试，若是 false 或 0，则所述业务应用程序没有被调试。

12、根据权利要求 8 所述的移动端安全切面的按需开启装置，根据所述移动端被攻击的方式获取被攻击的函数作为切点，包括：

获取所述业务应用程序中的用于操作所述移动端越狱/ROOT 后留下的高风险文件的函数作为所述切点；和/或，

获取所述业务应用程序中被注入/HOOK 的函数作为所述切点；和/或，

获取所述业务应用程序中被调试的函数作为所述切点。

13、一种可读存储介质，其上存储有计算机程序，当计算机程序在计算机中执行时，令计算机执行如权利要求 1-6 中任一项所述的移动端安全切面的按需开启方法。

14、一种电子设备，其包括存储器和处理器，存储器中存储有可执行代码，当处理器执行可执行代码时，其执行如权利要求 1-6 中任一项所述的移动端安全切面的按需开启方法。