(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 107103239 A
(43)申请公布日 2017.08.29

(21)申请号 201710229677.6

(22)申请日 2017.04.10

(71)申请人 中国民生银行股份有限公司
地址 100031 北京市西城区复兴门内大街2号

(72)发明人 张磊 高晓梦 吕晓强 李吉慧

(74)专利代理机构 北京同立钧成知识产权代理有限公司 11205
代理人 宋扬 刘芳

(51)Int.Cl.
G06F 21/56(2013.01)
G06F 21/57(2013.01)

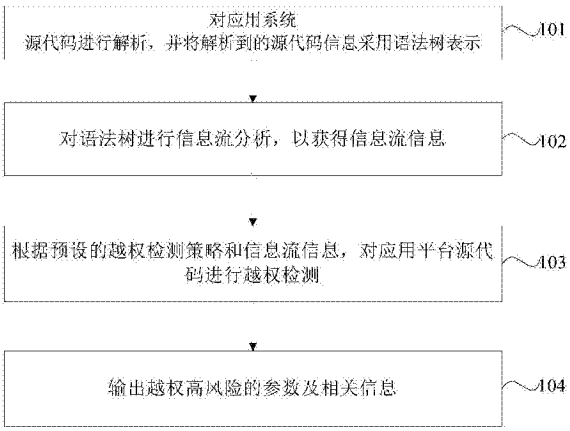
权利要求书2页 说明书9页 附图5页

(54)发明名称

基于应用系统业务处理逻辑的源代码越权检测方法及装置

(57)摘要

本发明提供了一种基于应用系统业务处理逻辑的源代码越权检测方法及装置。该方法包括：对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示；对语法树进行信息流分析，以获得信息流信息；根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测；输出越权高风险的参数及相关信息。由于采用信息流对参数进行越权检测，与应用系统业务逻辑紧密相关，能够对源代码的逻辑进行深入分析，所以能够减少误报率，提高检测的准确率，并且实现了基于业务逻辑的越权漏洞的检测，使应用系统的源代码在银行等业务场景广泛的行业能被安全使用。



CN 107103239 A

1.一种基于应用系统业务处理逻辑的源代码越权检测方法，其特征在于，包括：

对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示；

对所述语法树进行信息流分析，以获得信息流信息；

根据预设的越权检测策略和所述信息流信息，对所述应用系统源代码进行越权检测；

输出越权高风险的参数及相关信息。

2.根据权利要求1所述的方法，其特征在于，所述对所述语法树进行信息流分析，以获得信息流信息，具体包括：

对所述语法树进行控制流分析，以获得控制流信息，所述控制流信息至少包括：类间关系信息、方法间关系信息；

对所述语法树进行数据流分析，以获得数据流信息，所述数据流信息至少包括：类信息、方法信息、参数信息、常量信息、表达式信息；

其中，所述信息流信息包括：控制流信息和数据流信息。

3.根据权利要求2所述的方法，其特征在于，所述根据预设的越权检测策略和所述信息流信息，对所述应用系统源代码进行越权检测，具体包括：

根据配置文件，查看易发生越权参数列表中第一参数的样式，判断第一参数的样式是否存在于已配置越权处理的参数样式表中；

获取所述第一参数中第一参数的样式不存在于已配置越权处理的参数样式表中的第二参数及所述第二参数对应的交易标识码，判断所述第二参数及对应的交易标识码是否关联存在于越权处理表中；

获取所述第二参数中所述第二参数及对应的交易标识码不关联存在于所述越权处理表中的第三参数及所述第三参数对应的java处理类的信息流信息；

对所述第三参数对应的java处理类的信息流信息进行分析，判断是否对所述第三参数进行数据库查询核对校验操作；

获取所述第三参数中不进行数据库查询核对校验操作的第四参数及所述第四参数对应的java处理类的信息流信息；

对所述第四参数对应的java处理类的信息流信息进行分析，判断所述第四参数是否和会话相关；

获取所述第四参数中不与会话相关的第五参数及对应的前端页面，根据所述前端页面判断所述第五参数是否通过用户输入；

将所述第五参数中不通过用户输入的参数作为第六参数。

4.根据权利要求3所述的方法，其特征在于，所述输出越权高风险的参数及相关信息，具体包括：

输出越权高风险的参数名及越权高风险的参数对应的配置文件的XML文本的行号、交易标识码、java处理类、前端页面；

其中，所述越权高风险的参数为第六参数。

5.根据权利要求1-4任一项所述的方法，其特征在于，所述对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示之前，还包括：

获取应用系统源代码；

对所述应用系统源代码进行预处理。

6.一种基于应用系统业务处理逻辑的源代码越权检测装置，其特征在于，包括：

源代码解析模块，用于对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示；

信息流分析模块，用于对所述语法树进行信息流分析，以获得信息流信息；

越权检测模块，用于根据预设的越权检测策略和所述信息流信息，对所述应用系统源代码进行越权检测；

越权参数输出模块，用于输出越权高风险的参数及相关信息。

7.根据权利要求6所述的装置，其特征在于，所述信息流分析模块，具体用于：

对所述语法树进行控制流分析，以获得控制流信息，所述控制流信息至少包括：类间关系信息、方法间关系信息；数据流分析模块，用于对所述语法树进行数据流分析，以获得数据流信息，所述数据流信息至少包括：类信息、方法信息、参数信息、常量信息、表达式信息；其中，所述信息流信息包括：控制流信息和数据流信息。

8.根据权利要求7所述的装置，其特征在于，所述越权检测模块，具体用于：

根据配置文件，查看易发生越权参数列表中第一参数的样式，判断第一参数的样式是否存在于已配置越权处理的参数样式表中；获取所述第一参数中第一参数的样式不存在于已配置越权处理的参数样式表中的第二参数及所述第二参数对应的交易标识码，判断所述第二参数及对应的交易标识码是否关联存在于越权处理表中；获取所述第二参数中所述第二参数及对应的交易标识码不关联存在于所述越权处理表中的第三参数及所述第三参数对应的java处理类的信息流信息；对所述第三参数对应的java处理类的信息流信息进行分析，判断是否对所述第三参数进行数据库查询核对校验操作；获取所述第三参数中不进行数据库查询核对校验操作的第四参数及所述第四参数对应的java处理类的信息流信息；对所述第四参数对应的java处理类的信息流信息进行分析，判断所述第四参数是否和会话相关；获取所述第四参数中不与会话相关的第五参数及对应的前端页面，根据所述前端页面判断所述第五参数是否通过用户输入；将所述第五参数中不通过用户输入的参数作为第六参数。

9.根据权利要求8所述的装置，其特征在于，所述越权参数输出模块，具体用于：

输出越权高风险的参数名及越权高风险的参数对应的配置文件的XML文本的行号、交易标识码、java处理类、前端页面；其中，所述越权高风险的参数为第六参数。

10.根据权利要求6-9任一项所述的装置，其特征在于，还包括：

源代码获取模块，用于获取应用系统源代码；

源代码预处理模块，用于对所述应用系统源代码进行预处理。

基于应用系统业务处理逻辑的源代码越权检测方法及装置

技术领域

[0001] 本发明实施例涉及计算机技术领域，尤其涉及一种基于应用系统业务处理逻辑的源代码越权检测方法及装置。

背景技术

[0002] 随着网络技术和应用系统的飞速发展，信息安全正面临着前所未有的挑战。信息系统与互联网或其他网络的互连，使信息系统遭受攻击的概率增加。

[0003] 近年来，重大安全事件的频频发生揭示了当前信息系统安全形式的严峻性。软件的源代码是构建信息的基础组件，软件源代码中安全漏洞的存在是安全事件频繁发生的根源。因此多种软件源代码的安全检测软件应运而生。

[0004] 目前主流的源代码检测的开源软件包括：ITS4，RATS，BOON等。商业软件包括：Foritify，CheckMarx和CodeSecure等。ITS4代码检测是基于函数匹配的，其不关心上下文，只是搜索与漏洞数据库相匹配的函数或API接口，如果漏洞函数存在则发出警告。RATS结合了ITS4的静态检查技术和MOPS的深度语义分析技术检测缓冲区是否存在溢出漏洞，其能够对整个工程代码进行检测。BOON使用深度语义分析技术自动扫描源代码中存在的缓冲区溢出漏洞，可以对整数范围进行分析从而确定程序中的数组是否越界。Fortify是提供应用软件安全开发工具和管理方案的厂商，为应用软件开发组织、安全审计人员和应用安全管理人员提供工具并确立最佳的应用软件安全实践和策略。Checkmarx是以色列的一家高科技软件公司，其产品CheckmarxCxSuite可识别、跟踪和修复软件源代码上的技术和逻辑方面的安全风险，以查询语言定位代码安全问题，其采用独特的词汇分析技术和CxQL专利查询技术来扫描和分析源代码中的安全漏洞和弱点。CodeSecure内建语法剖析功能无需依赖编译环境，任何人员均可利用Web操作与集成开发环境双接口，找出存在信息安全问题的源代码，并提供修补建议进行调整。

[0005] 但目前主流的源代码检测的开源软件和商业软件，并没有严格的信息流机制，会产生极高的误报率，并且均主要关注常规的代码缺陷，只能检测如跨站脚本(简称:XSS)，结构化查询语言(简称:SQL)，密码管理，危险API接口等缺陷，对于应用系统源代码中的业务逻辑并不了解，不能检测基于业务逻辑而产生的越权的漏洞，使应用系统的源代码在银行等业务场景广泛的行业不能被安全使用。

发明内容

[0006] 本发明实施例提供一种基于应用系统业务处理逻辑的源代码越权检测方法及装置，该方法解决了现有技术中对源代码检测的开源软件和商业软件不能检测基于业务逻辑而产生的越权的漏洞，使应用系统的源代码在银行等业务场景广泛的行业不能被安全使用的技术问题。

[0007] 本发明实施例提供一种基于应用系统业务处理逻辑的源代码越权检测方法，包括：

[0008] 对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示；

[0009] 对所述语法树进行信息流分析，以获得信息流信息；

[0010] 根据预设的越权检测策略和所述信息流信息，对所述应用系统源代码进行越权检测；

[0011] 输出越权高风险的参数及相关信息。

[0012] 本发明实施例提供一种基于应用系统业务处理逻辑的源代码越权检测装置，包括：

[0013] 源代码解析模块，用于对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示；

[0014] 信息流分析模块，用于对所述语法树进行信息流分析，以获得信息流信息；

[0015] 越权检测模块，用于根据预设的越权检测策略和所述信息流信息，对所述应用系统源代码进行越权检测；

[0016] 越权参数输出模块，用于输出越权高风险的参数及相关信息。

[0017] 本发明实施例提供一种基于应用系统业务处理逻辑的源代码越权检测方法及装置，通过对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示；对语法树进行信息流分析，以获得信息流信息；根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测；输出越权高风险的参数及相关信息。由于采用信息流对参数进行越权检测，与应用系统业务逻辑紧密相关，能够对源代码的逻辑进行深入分析，所以能够减少误报率，提高检测的准确率，并且实现了基于业务逻辑的越权漏洞的检测，使应用系统的源代码在银行等业务场景广泛的行业能被安全使用。

附图说明

[0018] 为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图做一简单地介绍，显而易见地，下面描述中的附图是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动性的前提下，还可以根据这些附图获得其他的附图。

[0019] 图1为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例一的流程图；

[0020] 图2为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例二的流程图；

[0021] 图3为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例二中步骤204的流程图；

[0022] 图4为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例二中步骤205的流程图；

[0023] 图5为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例二中步骤205中各参数关系的示意图；

[0024] 图6为本发明基于应用系统业务处理逻辑的源代码越权检测装置实施例一的结构示意图；

[0025] 图7为本发明基于应用系统业务处理逻辑的源代码越权检测装置实施例二的结构

示意图。

具体实施方式

[0026] 为使本发明实施例的目的、技术方案和优点更加清楚，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0027] 应当理解，本文中使用的术语“和/或”仅仅是一种描述关联对象的关联关系，表示可以存在三种关系，例如，A和/或B，可以表示：单独存在A，同时存在A和B，单独存在B这三种情况。另外，本文中字符“/”，一般表示前后关联对象是一种“或”的关系。

[0028] 取决于语境，如在此所使用的词语“如果”可以被解释成为“在……时”或“当……时”或“响应于确定”或“响应于检测”。类似地，取决于语境，短语“如果确定”或“如果检测(陈述的条件或事件)”可以被解释成为“当确定时”或“响应于确定”或“当检测(陈述的条件或事件)时”或“响应于检测(陈述的条件或事件)”。

[0029] 图1为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例一的流程图，本实施例的执行主体为基于应用系统业务处理逻辑的源代码越权检测装置，该基于应用系统业务处理逻辑的源代码越权检测装置可以安装或集成在计算机或服务器上，如图1所示，则本实施例提供的基于应用系统业务处理逻辑的源代码越权检测方法包括以下几个步骤。

[0030] 步骤101，对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示。

[0031] 其中，应用系统可以为科蓝平台或其他应用系统，本实施例中对此不做限定。

[0032] 具体地，本实施例中，对应用系统的源代码进行词法、语法、语义解析，将得到的源代码信息用语法树表示，该语法树可以为抽象语法树或其他语法树，本实施例中对此不做限定。解析到的语法树中包括：包信息、类信息、方法信息、定义信息、表达式信息等。

[0033] 步骤102，对语法树进行信息流分析，以获得信息流信息。

[0034] 具体地，本实施例中，对语法树进行信息流分析包括对语法树进行控制流分析和信息流分析，信息流信息包括：控制流信息和数据流信息。

[0035] 其中，控制流信息包括：类间关系信息、方法间关系信息以及其他控制流信息。数据流信息包括：类信息、方法信息、参数信息、常量信息、表达式信息以及其他数据流信息。

[0036] 步骤103，根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测。

[0037] 本实施例中，在应用系统的源代码的配置文件中关联存储有参数及与参数相关的信息，与参数相关的信息可以包括：参数的样式、参数对应的交易标识码、参数对应的java处理类、参数对应的前端页面等。

[0038] 具体地，本实施例中，对预设的越权检测策略不做限定。如预设的越权检测策略可以为：获取易发生越权参数，对易发生越权参数进行样式检测，根据易发生越权参数的样式与应用系统已配置越权处理的参数样式表中存储的参数样式进行比对，若易发生越权参数的样式存在于应用系统已配置越权处理的参数样式表中，则说明该应用系统会对该易发生

越权参数进行越权检测，该易发生越权参数是安全参数，否则该易发生越权参数具有越权高风险。

[0039] 本实施例中，预设的越权检测策略还可以为：获取易发生越权参数对应的交易标识码，判断易发生越权参数及对应的交易标识码是否关联存在于越权处理表中。在越权处理表中存储有应用系统的每个交易及在该交易中会进行越权检测的参数，该越权处理表是根据实际应用预先配置的。若易发生越权参数及对应的交易标识码关联存在于越权处理表中，则说明该应用系统会对该参数进行越权检测，该易发生越权参数是安全参数，否则该易发生越权参数具有越权高风险。

[0040] 本实施例中，预设的越权检测策略还可以为：获取易发生越权参数及对应的java处理类的信息流信息，并对java处理类的信息流信息进行分析，判断是否对易发生越权参数进行数据库查询核对校验操作，在科蓝平台即是判断易发生越权参数是否调用了searchAccountById函数，若进行数据库查询核对校验操作，则说明应用系统会该易发生越权参数进行越权检测，该易发生越权参数是安全参数，否则该易发生越权参数具有越权高风险。

[0041] 本实施例中，预设的越权检测策略还可以为：获取易发生越权参数及对应的java处理类的信息流信息，并对java处理类的信息流信息进行分析，判断该易发生越权参数是否和会话相关，即是否被会话对应的内容覆盖或是否与会话中对应的内容进行比较，若该易发生越权参数被会话对应的内容覆盖或与会话中对应的内容进行了比较，则该易发生越权参数和会话相关，则说明该应用系统会该易发生越权参数进行越权检测，该易发生越权参数是安全参数，否则该参数具有越权高风险。

[0042] 本实施例中，预设的越权检测策略还可以为：获取易发生越权参数及对应的前端页面，根据前端页面判断该易发生越权参数是否通过用户输入，若该易发生越权参数为用户输入的，则是安全参数，否则该参数有越权高风险。

[0043] 综上可知，本实施例中，预设的越权检测策略还可以为：根据上述列举的预设的越权检测策略依次对易发生越权参数进行筛选，每一步中去除安全参数，筛选出有越权风险的参数，再将有越权风险的参数输入到下一步越权检测中，以对参数进行越权检测。其中的筛选顺序本实施例中不做限定。可以理解的是，本实施例中，预设的越权检测策略还可以为：对上述列举的预设的越权检测策略进行挑选，选出至少两个上述列举的预设的越权检测策略，并依次对易发生越权参数进行筛选，去除安全参数，筛选出有越权高风险的参数，以对易发生越权参数进行越权检测。

[0044] 本实施例中，易发生越权参数是对应用系统的源代码进行解析，获取到信息流信息，并获取信息系流信息中所有的参数，利用越权发生原理对参数进行筛选得到的，可将易发生越权参数写入列表中，形成易发生越权参数列表。

[0045] 步骤104，输出越权高风险的参数及相关信息。

[0046] 本实施例中，经过对应用系统源代码进行越权检测后，去除安全参数，输出具有越权高风险的参数及相关信息，其中，相关信息可以包括：的配置文件的XML文本的行号、交易标识码、java处理类、前端页面等，还可以包括其他信息，本实施例中对此不做限定。

[0047] 本实施例提供的基于应用系统业务处理逻辑的源代码越权检测方法，通过对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示；对语法树进行信息流分

析，以获得信息流信息；根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测；输出越权高风险的参数及相关信息。由于采用信息流对参数进行越权检测，与应用系统业务逻辑紧密相关，能够对源代码的逻辑进行深入分析，所以能够减少误报率，提高检测的准确率，并且实现了基于业务逻辑的越权漏洞的检测，使应用系统的源代码在银行等业务场景广泛的行业能被安全使用。

[0048] 图2为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例二的流程图，如图2所示，本实施例提供的基于应用系统业务处理逻辑的源代码越权检测方法，是在本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例一的基础上，对步骤102-步骤104的进一步细化，并且包括了对应用系统源代码进行预处理的步骤，则本实施例提供的基于应用系统业务处理逻辑的源代码越权检测方法包括以下步骤。

[0049] 步骤201，获取应用系统源代码。

[0050] 进一步地，本实施例中，可采用用户通过上传设备上传的方式获取待检测的应用系统源代码，也可从预设存储区域获取待检测的应用系统源代码，本实施例中对获取应用系统源代码的方式不做限定。

[0051] 步骤202，对应用系统源代码进行预处理。

[0052] 进一步地，本实施例中，解析应用系统的源代码中引入的外部文件、宏信息，将外部文件、宏信息替换到源代码中相应的位置。

[0053] 步骤203，对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示。

[0054] 本实施例中，步骤203的实现方式与本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例一中的步骤101的实现方式相同，在此不再一一赘述。

[0055] 步骤204，对语法树进行信息流分析，以获得信息流信息。

[0056] 进一步地，本实施例中，步骤204，对语法树进行信息流分析，以获得信息流信息具体包括以下步骤：

[0057] 步骤204a，对语法树进行控制流分析，以获得控制流信息。

[0058] 其中，控制流信息至少包括：类间关系信息、方法间关系信息。

[0059] 步骤204b，对语法树进行数据流分析，以获得数据流信息。

[0060] 其中，数据流信息至少包括：类信息、方法信息、参数信息、常量信息、表达式信息。

[0061] 可以理解的是，信息流信息包括：控制流信息和数据流信息。

[0062] 步骤205，根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测。

[0063] 进一步地，本实施例中，步骤205，根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测具体包括以下步骤。

[0064] 步骤205a，根据配置文件，查看易发生越权参数列表中第一参数的样式，判断第一参数的样式是否存在于已配置越权处理的参数样式表中，若是，则结束，否则，执行步骤205b。

[0065] 本实施例中，应用系统的配置文件中关联存储有源代码中所有参数的参数名、参数样式、参数所属交易标识码、参数对应的java处理类、参数对应的前端页面等信息。

[0066] 进一步地，本实施例中，在易发生越权参数表中存储有应用系统易发生越权的所

有参数名，在已配置越权处理的参数样式表中存储有应用系统在源代码中已经配置的会进行越权处理的参数的所有样式。如在科蓝平台中，在已配置越权处理的参数样式表中存储的会进行越权处理的参数样式为以“AcAcNoStyleWithBean”开头的参数样式。查看配置文件中所有的易发生越权的参数对应的参数样式，判断每个易发生越权的参数样式是否存在于已配置越权处理的参数样式表中，若存在于已配置越权处理的参数样式表中，则说明应用系统会对该易发生越权参数进行越权检测，该易发生越权参数为安全参数，否则该易发生越权参数具有越权风险，需要进一步进行越权检测。

[0067] 其中，图5为本发明基于应用系统业务处理逻辑的源代码越权检测方法实施例二中步骤205中各参数关系的示意图，如图5所示，易发生越权参数列表中的所有参数称为第一参数。

[0068] 步骤205b，获取第一参数中第一参数的样式不存在于已配置越权处理的参数样式表中的第二参数及第二参数对应的交易标识码。

[0069] 步骤205c，判断第二参数及对应的交易标识码是否关联存在于越权处理表中，若是，则结束，否则，执行步骤205d。

[0070] 其中，如图5所示，第二参数为第一参数中第一参数的样式不存在于已配置越权处理的参数样式表中的参数，第一参数中第一参数的样式存在于已配置越权处理的参数样式表中的参数为安全参数。

[0071] 本实施例中，越权处理表中关联存储有应用系统的每个交易及在该交易中会进行越权检测的参数，该越权处理表是根据实际应用预先配置的。关联存储的方式如可以为“A1.a”，其中“.”之前的文字为交易标识码，“.”之后的文字为会进行越权检测的参数。还可以为其他关联存储方式，本实施例中对此不做限定。

[0072] 具体地，本实施例中，从第一参数中获取具有越权风险的第二参数进一步进行越权检测，由于配置文件中关联存储有每个参数及对应交易标识码，所以通过配置文件，获取每个第二参数及对应的交易标识码，判断第二参数及对应的交易标识码是否关联存在于越权处理表中，若是，则说明应用系统会对该参数进行越权检测，该第二参数是安全参数，否则，说明该第二参数具有越权风险，需要进一步进行越权检测。

[0073] 步骤205d，获取第二参数中第二参数及对应的交易标识码不关联存在于越权处理表中的第三参数及第三参数对应的java处理类的信息流信息。

[0074] 其中，如图5所示，第三参数为第二参数中不与对应的交易标识码关联存在于越权处理表中的参数，第二参数中与对应的交易标识码关联存在于越权处理表中的参数为安全参数。

[0075] 步骤205e，对第三参数对应的java处理类的信息流信息进行分析，判断是否对第三参数进行数据库查询核对校验操作，若是，则结束，否则执行步骤205f。

[0076] 具体地，本实施例中，由于配置文件中关联存储有每个参数及对应的java处理类，所以通过配置文件，获取第三参数及第三参数对应的java处理类的信息流信息。对每个第三参数的java处理类的信息流信息进行分析，分析每个第三参数是否进行数据库查询核对校验操作，其可通过是否调取了对应的函数进行判断，如在科蓝平台中，通过判断第三参数是否调用了searchAccountById函数来判断第三参数是否进行数据库查询核对校验操作。若第三参数进行数据库查询核对校验操作，说明该第三参数会进行越权检测，为安全参数，

否则，说明该第三参数具有越权风险，需要进一步进行越权检测。

[0077] 步骤205f，获取第三参数中不进行数据库查询核对校验操作的第四参数及第四参数对应的java处理类的信息流信息。

[0078] 其中，如图5所示，第四参数为第三参数中不进行数据库查询核对校验操作的参数。第三参数中进行数据库查询核对校验操作的参数为安全参数。

[0079] 步骤205g，对第四参数对应的java处理类的信息流信息进行分析，判断第四参数是否和会话相关，若是，则结束，否则，执行步骤205h。

[0080] 进一步地，本实施例中，对第四参数对应的java处理类的信息流信息进一步进行分析，此次判断第四参数是否和会话相关。若第四参数来自于会话中对应内容的覆盖或者第四参数与会话中对应的内容进行比较，则说明第四参数与会话相关，与会话相关的第四参数为进行越权检测的参数，为安全参数，否则，该第四参数具有越权风险，需要进一步进行越权检测。

[0081] 步骤205h，获取第四参数中不与会话相关的第五参数及对应的前端页面。

[0082] 其中，如图5所示，第五参数为第四参数中不与会话相关的参数，第四参数中与会话相关的参数为安全参数。

[0083] 可以理解的是，根据配置文件获取第五参数对应的前端页面。前端页面可为JSP/HTML页面。

[0084] 步骤205i，根据前端页面判断第五参数是否通过用户输入，若是，则结束，否则，执行步骤205j。

[0085] 步骤205j，将第五参数中不通过用户输入的参数作为第六参数。

[0086] 其中，如图5所示，第六参数为第五参数中不通过用户输入的参数，第五参数中通过用户输入的参数为安全参数。

[0087] 进一步地，本实施例中，根据前端页面中参数的输入格式判断第五参数是否通过用户输入，若输入格式为.txt格式，则说明第五参数是通过用户输入，否则不是通过用户输入。通过用户输入的第五参数为安全参数，否则，将第五参数中不通过用户输入的参数作为第六参数，经过上述的一步步的越权检测后，第六参数对于每一步的越权检测均未通过，则将第六参数作为发生越权的参数。

[0088] 步骤206，输出越权高风险的参数及相关信息。

[0089] 进一步地，本实施例中，步骤206中输出越权高风险的参数及相关信息，具体包括：

[0090] 输出越权高风险的参数名及越权高风险的参数对应的配置文件的XML文本的行号、交易标识码、java处理类、前端页面；

[0091] 其中，越权高风险的参数为第六参数。

[0092] 本实施例提供的基于应用系统业务处理逻辑的源代码越权检测方法，根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测具体包括：根据配置文件，查看易发生越权参数列表中每个第一参数的样式，判断每个第一参数的样式是否存在于已配置越权处理的参数样式表中；获取第一参数中第一参数的样式不存在于已配置越权处理的参数样式表中的第二参数及第二参数对应的交易标识码，判断第二参数及对应的交易标识码是否关联存在于越权处理表中；获取第二参数中第二参数及对应的交易标识码不关联存在于越权处理表中的第三参数及第三参数对应的java处理类的信息流信息；对第三参数对

应的java处理类的信息流信息进行分析，判断是否对第三参数进行数据库查询核对校验操作；获取第三参数中不进行数据库查询核对校验操作的第四参数及第四参数对应的java处理类的信息流信息；对第四参数对应的java处理类的信息流信息进行分析，判断第四参数是否和会话相关；获取第四参数中不与会话相关的第五参数及对应的前端页面，根据前端页面判断第五参数是否通过用户输入；将第五参数中不通过用户输入的参数作为第六参数。经过五步的越权检测，能够使检测出的越权高风险参数更加准确，进一步减少误报率，提高检测的准确率，并且对易发生越权参数进行大范围到小范围的筛选，有效提高了越权检测的效率。

[0093] 本领域普通技术人员可以理解：实现上述各方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成。前述的程序可以存储于一可读取存储介质中。该程序在执行时，执行包括上述各方法实施例的步骤；而前述的存储介质包括：ROM、RAM、磁碟或者光盘等各种可以存储程序代码的介质。

[0094] 图6为本发明基于应用系统业务处理逻辑的源代码越权检测装置实施例一的结构示意图，如图6所示，本实施例提供的基于应用系统业务处理逻辑的源代码越权检测装置包括：源代码解析模块61，信息流分析模块62，越权检测模块63和越权参数输出模块64。

[0095] 其中，源代码解析模块61，用于对应用系统源代码进行解析，并将解析到的源代码信息采用语法树表示。信息流分析模块62，用于对语法树进行信息流分析，以获得信息流信息。越权检测模块63，用于根据预设的越权检测策略和信息流信息，对应用系统源代码进行越权检测。越权参数输出模块64，用于输出越权高风险的参数及相关信息。

[0096] 本实施例提供的基于应用系统业务处理逻辑的源代码越权检测装置可以执行图1所示方法实施例的技术方案，其实现原理和技术效果类似，此处不再赘述。

[0097] 图7为本发明基于应用系统业务处理逻辑的源代码越权检测装置实施例二的结构示意图，如图7所示，本实施例提供的基于应用系统业务处理逻辑的源代码越权检测装置在本发明基于应用系统业务处理逻辑的源代码越权检测装置实施例一的基础上，进一步地，还包括：源代码获取模块71和源代码预处理模块72。

[0098] 进一步地，信息流分析模块62，具体用于：对语法树进行控制流分析，以获得控制流信息，控制流信息至少包括：类间关系信息、方法间关系信息；数据流分析模块，用于对语法树进行数据流分析，以获得数据流信息，数据流信息至少包括：类信息、方法信息、参数信息、常量信息、表达式信息；其中，信息流信息包括：控制流信息和数据流信息。

[0099] 优选地，越权检测模块63，具体用于：根据配置文件，查看易发生越权参数列表中第一参数的样式，判断第一参数的样式是否存在于已配置越权处理的参数样式表中；获取第一参数中第一参数的样式不存在于已配置越权处理的参数样式表中的第二参数及第二参数对应的交易标识码，判断第二参数及对应的交易标识码是否关联存在于越权处理表中；获取第二参数中第二参数及对应的交易标识码不关联存在于越权处理表中的第三参数及第三参数对应的java处理类的信息流信息；对第三参数对应的java处理类的信息流信息进行分析，判断是否对第三参数进行数据库查询核对校验操作；获取第三参数中不进行数据库查询核对校验操作的第四参数及第四参数对应的java处理类的信息流信息；对第四参数对应的java处理类的信息流信息进行分析，判断第四参数是否和会话相关；获取第四参数中不与会话相关的第五参数及对应的前端页面，根据前端页面判断第五参数是否通过用

户输入;将第五参数中不通过用户输入的参数作为第六参数。

[0100] 进一步地,越权参数输出模块64,具体用于:输出越权高风险的参数名及越权高风险的参数对应的配置文件的XML文本的行号、交易标识码、java处理类、前端页面;其中,越权高风险的参数为第六参数。

[0101] 进一步地,源代码获取模块71,用于获取应用系统源代码。源代码预处理模块72,用于对应用系统源代码进行预处理。

[0102] 本实施例提供的基于应用系统业务处理逻辑的源代码越权检测装置可以执行图2、图3和图4所示方法实施例的技术方案,其实现原理和技术效果类似,此处不再赘述。

[0103] 最后应说明的是:以上各实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述各实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分或者全部技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的范围。

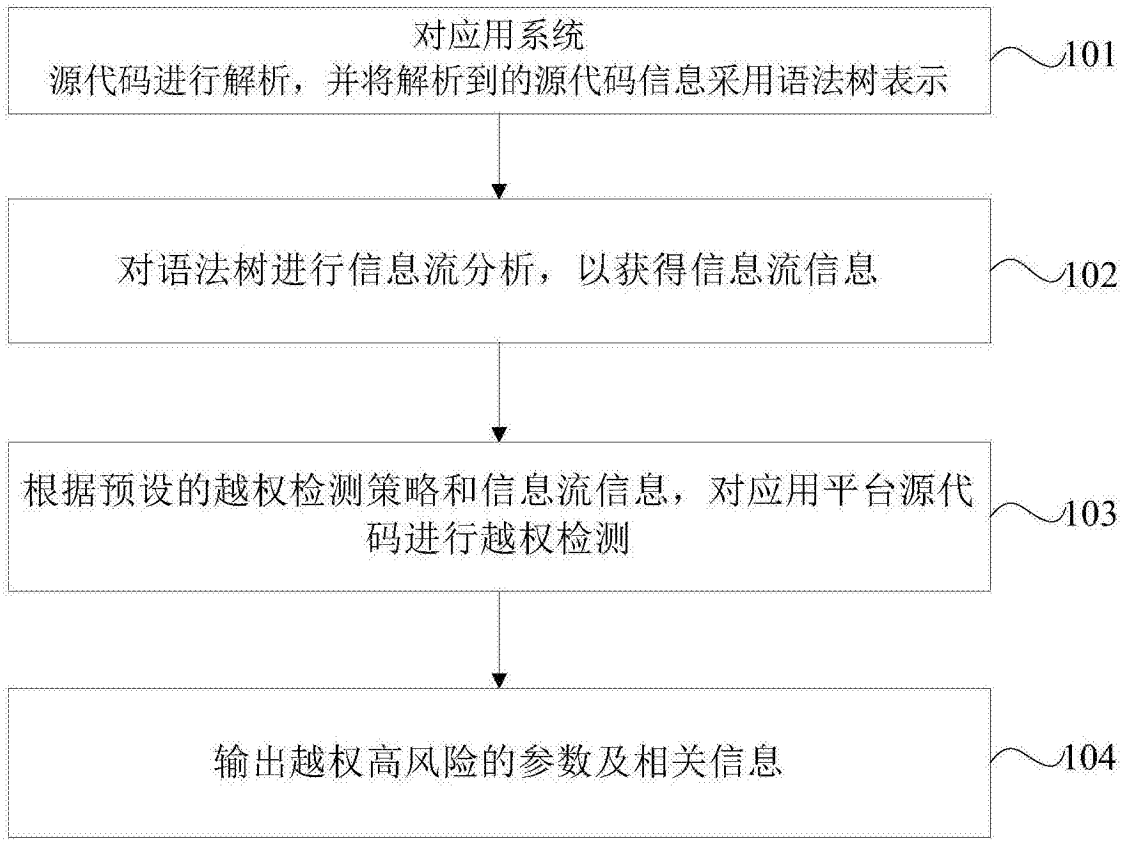

对应用系统
源代码进行解析，并将解析到的源代码信息采用语法树表示
101
对语法树进行信息流分析，以获得信息流信息
102
根据预设的越权检测策略和信息流信息，对应用平台源代码进行越权检测
103
输出越权高风险的参数及相关信息
104

图1

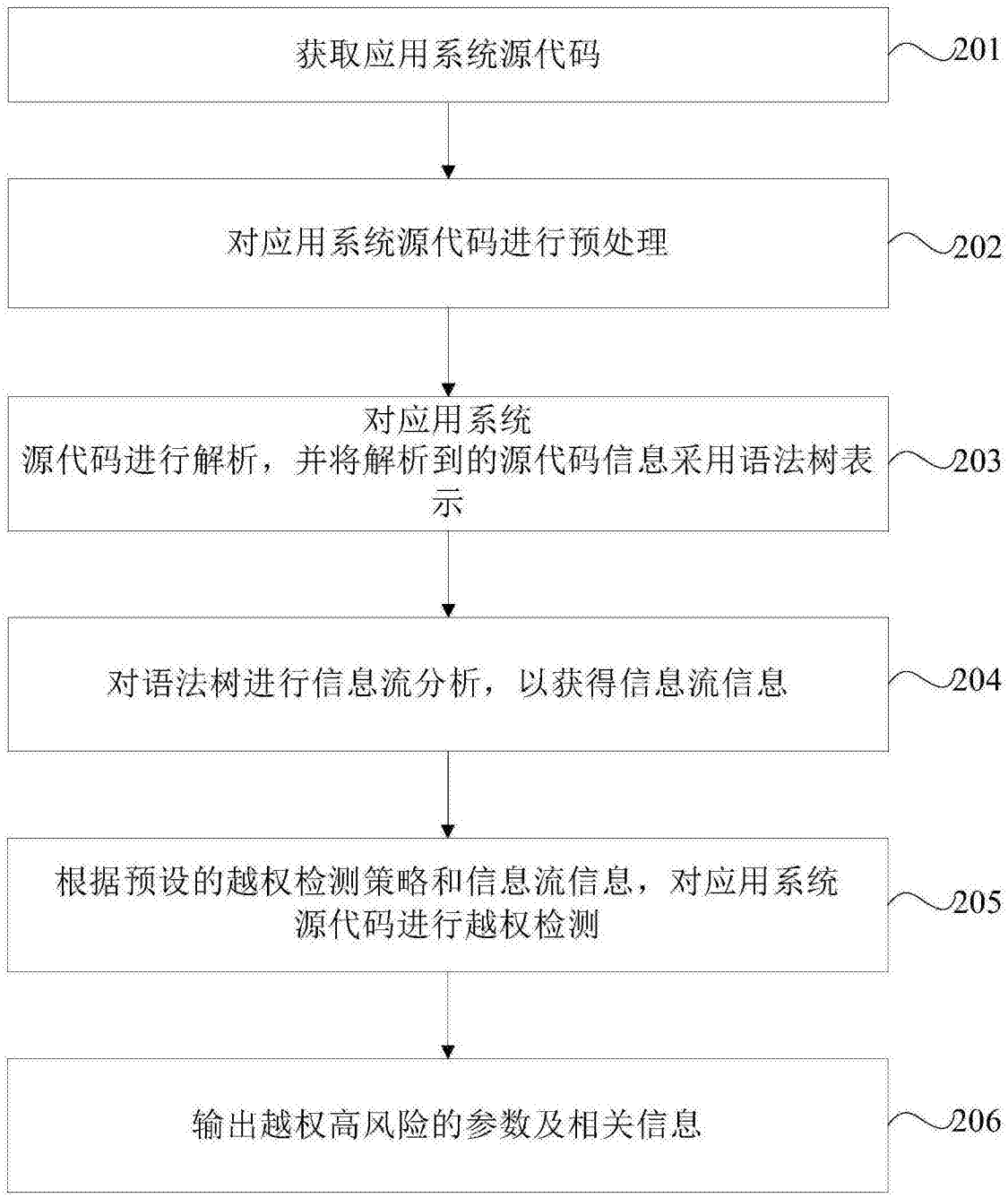

获取应用系统源代码
201
对应用系统源代码进行预处理
202
对应用系统
源代码进行解析，并将解析到的源代码信息采用语法树表示
203
对语法树进行信息流分析，以获得信息流信息
204
根据预设的越权检测策略和信息流信息，对应用系统
源代码进行越权检测
205
输出越权高风险的参数及相关信息
206

图2

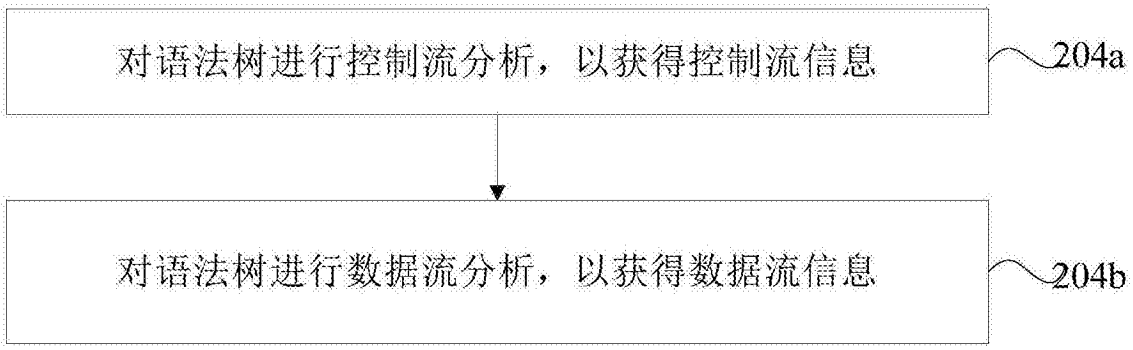


图3

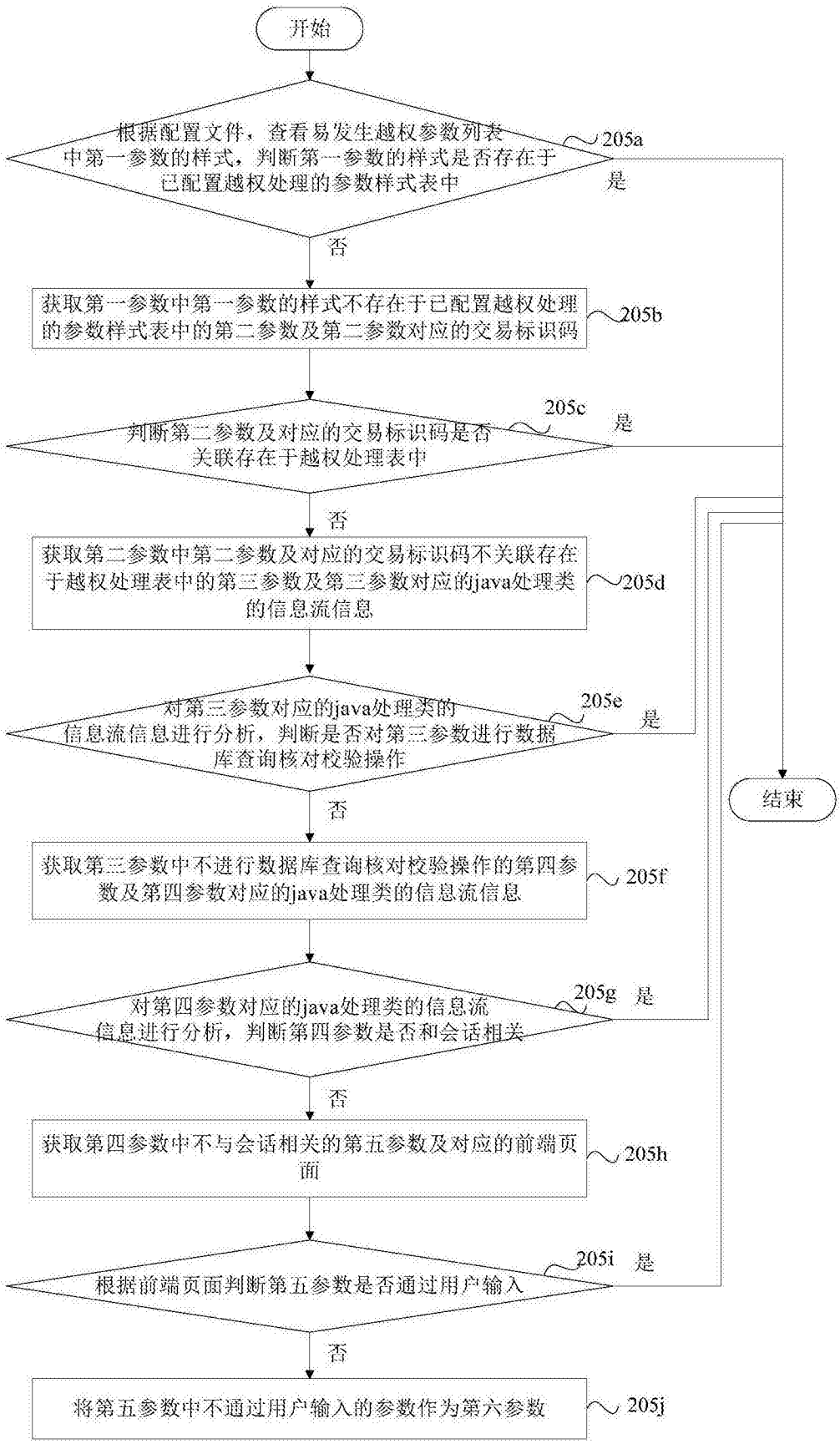

开始
根据配置文件，查看易发生越权参数列表中第一参数的样式，判断第一参数的样式是否存在于已配置越权处理的参数样式表中
205a
是
否
获取第一参数中第一参数的样式不存在于已配置越权处理的参数样式表中的第二参数及第二参数对应的交易标识码
205b
判断第二参数及对应的交易标识码是否关联存在于越权处理表中
205c
是
否
获取第二参数中第二参数及对应的交易标识码不关联存在于越权处理表中的第三参数及第三参数对应的java处理类的信息流信息
205d
对第三参数对应的java处理类的信息流信息进行分析，判断是否对第三参数进行数据库查询核对校验操作
205e
是
否
结束
获取第三参数中不进行数据库查询核对校验操作的第四参数及第四参数对应的java处理类的信息流信息
205f
对第四参数对应的java处理类的信息流信息进行分析，判断第四参数是否和会话相关
205g
是
否
获取第四参数中不与会话相关的第五参数及对应的前端页面
205h
根据前端页面判断第五参数是否通过用户输入
205i
是
否
将第五参数中不通过用户输入的参数作为第六参数
205j

图4

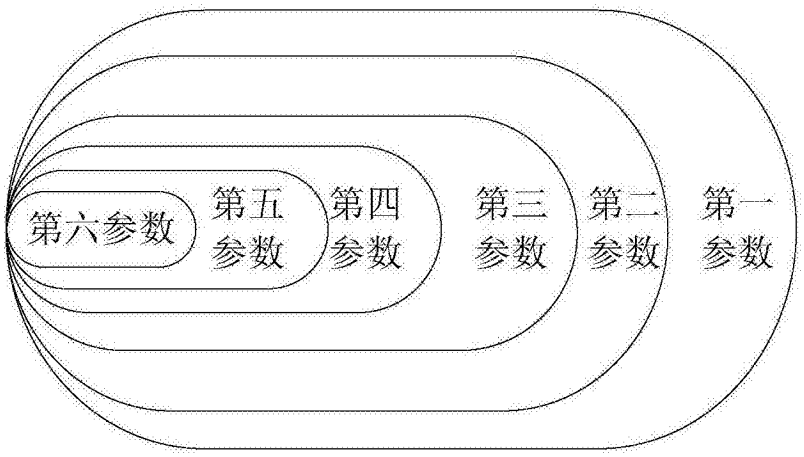

第六参数
第五参数
第四参数
第三参数
第二参数
第一参数

图5

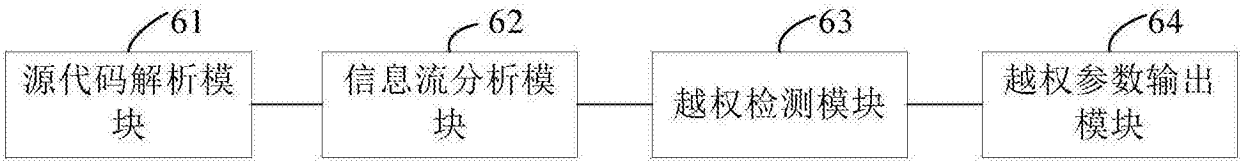

61
源代码解析模块
62
信息流分析模块
63
越权检测模块
64
越权参数输出模块

图6

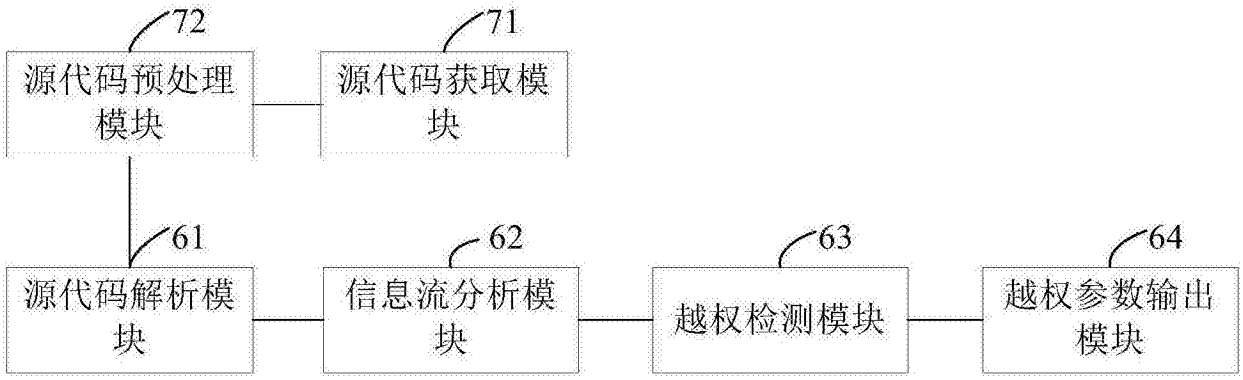

72
源代码预处理模块
71
源代码获取模块
61
源代码解析模块
62
信息流分析模块
63
越权检测模块
64
越权参数输出模块

图7