(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2020 年 12 月 30 日 (30.12.2020)

WIPO | PCT

(10) 国际公布号
WO 2020/259390 A1

(51) 国际专利分类号:
G06F 21/57 (2013.01)

(21) 国际申请号: PCT/CN2020/096901

(22) 国际申请日: 2020 年 6 月 18 日 (18.06.2020)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910551537.X 2019年6月24日 (24.06.2019) CN

(71) 申请人: 深圳前海微众银行股份有限公司 (WEBANK CO., LTD.) [CN/CN]; 中国广东省深圳市前海深港合作区前湾一路1号A栋201室, Guangdong 518027 (CN)。

(72) 发明人: 郑祎 (ZHENG, Yi); 中国广东省深圳市前海深港合作区前湾一路1号A栋201室, Guangdong 518027 (CN)。

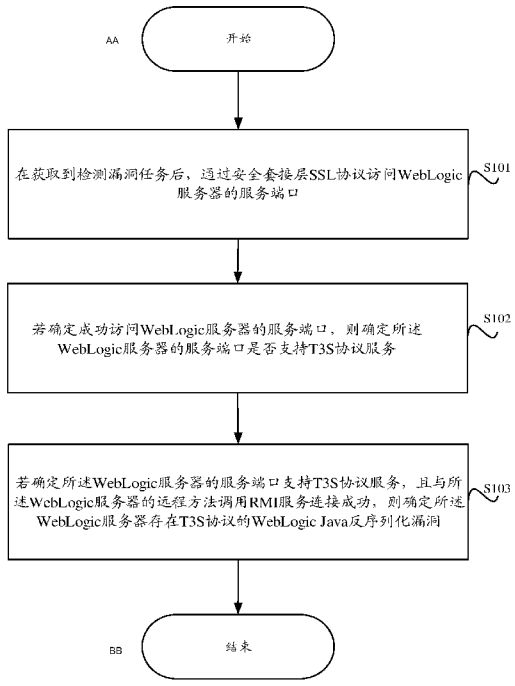
(74) 代理人: 北京同达信恒知识产权代理有限公司 (TDIP & PARTNERS); 中国北京市西城区裕民路18号北环中心A座2002, Beijing 100029 (CN)。

(81) 指定国(除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,

WO 2020/259390 A1

(54) **Title:** METHOD AND APPARATUS FOR DETECTING DESERIALIZATION VULNERABILITY

(54) 发明名称: 一种反序列化漏洞的检测方法及装置



S101 After a vulnerability detection task is acquired, access a service port of a WebLogic server by means of a secure sockets layer (SSL) protocol
S102 If it is determined that the service port of the WebLogic server is successfully accessed, determine whether the service port of the WebLogic server supports a T3S protocol service
S103 If it is determined that the service port of the WebLogic server supports the T3S protocol service, and is successfully connected to a remote method invocation (RMI) service of the WebLogic server, determine that there is a WebLogic Java deserialization vulnerability of a T3S protocol in the WebLogic server
AA Start
BB End

图 1

(57) **Abstract:** A method and apparatus for detecting a deserialization vulnerability, relating to the field of fintech, and used to solve the problem in the prior art of being unable to detect a deserialization vulnerability of a T3S protocol. The method comprises: after a vulnerability detection task is acquired, accessing a service port of a WebLogic server by means of a secure sockets layer (SSL) protocol (S101); if it is determined that the service port of the WebLogic server is successfully accessed, determining whether the service port of the WebLogic server supports a T3S protocol service (S102); and if it is determined that the service port of the WebLogic server

[见续页]

ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

(84) 指定国(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布：
— 包括国际检索报告(条约第21条(3))。

supports the T3S protocol service, and is successfully connected to a remote method invocation (RMI) service of the WebLogic server, determining that there is a WebLogic Java deserialization vulnerability of a T3S protocol in the WebLogic server (S103). By means of the method, the detection of the deserialization vulnerability of the T3S protocol can be realized, and same can be performed on the premise of determining that the service port can be accessed and that the service port supports the T3S protocol service, thereby improving the efficiency of vulnerability detection.

(57) 摘要：一种反序列化漏洞的检测方法及装置，涉及金融科技领域，用以解决现有技术中无法针对T3S协议进行反序列化漏洞检测的问题。其中，方法包括：在获取到检测漏洞任务后，通过安全套接层SSL协议访问WebLogic服务器的服务端口（S101）；若确定成功访问WebLogic服务器的服务端口，则确定WebLogic服务器的服务端口是否支持T3S协议服务（S102）；若确定WebLogic服务器的服务端口支持T3S协议服务，且与WebLogic服务器的远程方法调用RMI服务连接成功，则确定WebLogic服务器存在T3S协议的WebLogic Java反序列化漏洞（S103）。通过该方法能够实现针对T3S协议的反序列化漏洞检测，并能够在确定服务端口能够访问且服务端口支持T3S协议服务的前提下进行的，提高漏洞检测的效率。

一种反序列化漏洞的检测方法及装置

相关申请的交叉引用

本申请要求在2019年06月24日提交中国专利局、申请号为201910551537.X、申请名称为“一种反序列化漏洞的检测方法及装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本发明涉及金融科技（Fintech）技术领域，尤其涉及一种反序列化漏洞的检测方法及装置。

背景技术

WebLogic（一种应用服务器软件）是基于 Java EE（一种 Java 企业应用）架构的中间件，可作为用于开发、集成、部署和管理大型分布式 Web（World Wide Web，全球广域网）应用、网络应用和数据库应用的 Java 应用服务器。广泛用于政府、金融、医疗、交通、教育、科研等行业及领域。

当 WebLogic 使用 T3S 协议用于服务管理时，提供的 Web 服务的协议为 HTTPS；使用 T3 协议用于服务管理时，提供的 Web 服务的协议为 HTTP。因此使用 T3S 协议的 WebLogic 所提供的服务安全性要求更高，说明此类 WebLogic 服务中支持的业务功能更重要。一般来说，在政府、金融、医疗、交通、教育、科研等领域中，比如在基金、证券、保险、医院、交通局、教育主管部门、高校、研究所等行业或部门中，其部署的 WebLogic 大多提供 HTTPS 服务，使用 T3S 协议进行服务管理。

但是由于 WebLogic 存在 Java 反序列化漏洞，所以当攻击者向 WebLogic 服务器发送精心构造的反序列化数据时，可能会触发漏洞，进一步地，WebLogic 服务器会执行攻击者指定的操作，那么攻击者就可以控制该服务器，窃取数据库中的数据，造成严重影响。

现有技术中，只有针对 T3 协议的反序列化漏洞的检测方法，并没有针对 T3S 协议的反序列化漏洞的检测方法，所以亟需一种针对 T3S 协议的反序列化漏洞的检测方法。

发明内容

本申请提供一种反序列化漏洞的检测方法及装置，用以解决现有技术中没有针对 T3S 协议的反序列化漏洞的检测方法的问题。

第一方面，本发明实施例提供一种反序列化漏洞的检测方法，包括：在获取到检测漏洞任务后，通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口；若确定成功访问 WebLogic 服务器的服务端口，则确定所述 WebLogic 服务器的服务端口是否支持 T3S 协议服务；若确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3S 协议的 Java 反序列化漏洞。

本发明实施例中，在获取到检测漏洞任务后，开始进行针对 T3S 协议的反序列化漏洞进行检测，首先确定 WebLogic 服务器的服务端口是否能够访问，即是否能够通过安全套接层 SSL（Secure Sockets Layer）协议访问 WebLogic 服务器的服务端口，并在确定能够进

行访问后，则确定该 WebLogic 服务器的服务端口是否支持 T3S 协议服务，若支持 T3S 协议服务，则尝试与 WebLogic 服务器的远程方法调用 RMI 服务连接，若连接成功，则可以确定 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。通过上述方法能够实现针对 T3S 协议的反序列化漏洞检测，并且是在确定出 WebLogic 服务器的服务端口能够访问且并支持 T3S 协议服务的前提下进行的，避免了无用的检测步骤，提高了检测效率。

可选的，所述确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，包括:

向所述 WebLogic 服务器发送 T3S 协议数据;

若确定接收到的所述 WebLogic 服务器针对所述 T3S 协议返回的反馈数据与 T3S 协议规定的反馈数据匹配，则确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务。

可选的，所述确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功后，与所述 WebLogic 服务器的 RMI 服务连接成功前，还包括:

生成第一漏洞利用数据，并将所述第一漏洞利用数据发送给所述 WebLogic 服务器，所述第一漏洞利用数据用于指示所述 WebLogic 服务器在所述 WebLogic 服务器的当前目录下创建远程指令文件目录;

生成第二漏洞利用数据，所述第二漏洞利用数据中包括远程指令文件，所述远程指令文件用于启动 RMI 服务，并将所述第二漏洞利用数据发送给所述 WebLogic 服务器，所述第二漏洞利用数据用于指示所述 WebLogic 服务器在所述远程指令文件目录中保存所述远程指令文件。

生成第三漏洞利用数据，所述第三漏洞利用数据用于指示所述 WebLogic 服务器执行所述远程指令文件，启动所述 WebLogic 服务器的 RMI 服务。

可选的，所述与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，包括:

在获取所述 WebLogic 服务器的 SSL 证书后，若确定通过 T3S 协议与所述 WebLogic 服务器的 RMI 服务连接成功，则确定与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功。

可选的，所述方法还包括:

在通过传输控制协议 TCP 成功访问 WebLogic 服务器的服务端口后，确定所述 WebLogic 服务器的服务端口是否支持 T3 协议服务。

若确定所述 WebLogic 服务器的服务端口支持 T3 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3 协议的 WebLogic Java 反序列化漏洞。

可选的，所述获取检测漏洞任务，包括:

将所述检测漏洞任务加入到待执行任务队列中，并使用线程池中的空闲线程从所述执行任务队列获取所述检测漏洞任务。

第二方面，本发明实施例提供一种反序列化漏洞的检测装置，包括:

安全套接层 SSL 协议连接单元，用于在获取到检测漏洞任务后，通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口。

T3S 协议服务确定单元，用于若确定成功访问 WebLogic 服务器的服务端口，则确定所述 WebLogic 服务器的服务端口是否支持 T3S 协议服务。

漏洞确定单元，用于若确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。

可选的，所述 T3S 协议服务确定单元具体用于:

向所述 WebLogic 服务器发送 T3S 协议数据。

若确定接收到的所述 WebLogic 服务器针对所述 T3S 协议返回的反馈数据与 T3S 协议规定的反馈数据匹配，则确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务。

可选的，所述装置还包括:

数据生成单元，用于生成第一漏洞利用数据，并将所述第一漏洞利用数据发送给所述 WebLogic 服务器，所述第一漏洞利用数据用于指示所述 WebLogic 服务器在所述 WebLogic 服务器的当前目录下创建远程指令文件目录。

生成第二漏洞利用数据，所述第二漏洞利用数据中包括远程指令文件，所述远程指令文件用于启动 RMI 服务，并将所述第二漏洞利用数据发送给所述 WebLogic 服务器，所述第二漏洞利用数据用于指示所述 WebLogic 服务器在所述远程指令文件目录中保存所述远程指令文件。

生成第三漏洞利用数据，所述第三漏洞利用数据用于指示所述 WebLogic 服务器执行所述远程指令文件，启动所述 WebLogic 服务器的 RMI 服务。

可选的，所述漏洞确定单元具体用于:

在获取所述 WebLogic 服务器的 SSL 证书后，则确定通过 T3S 协议与所述 WebLogic 服务器的 RMI 服务连接成功。

可选的，所述装置还包括 T3 协议漏洞检测单元，所述 T3 协议漏洞检测单元用于:

在通过传输控制协议 TCP 成功访问 WebLogic 服务器的服务端口后，确定所述 WebLogic 服务器的服务端口是否支持 T3 协议服务。

若确定所述 WebLogic 服务器的服务端口支持 T3 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3 协议的 WebLogic Java 反序列化漏洞。

可选的，所述装置还包括获取单元，所述获取单元用于:

将所述检测漏洞任务加入到待执行任务队列中，并使用线程池中的空闲线程从所述执行任务队列获取所述检测漏洞任务。

第三方面，本发明提供一种计算机设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，所述处理器执行所述程序时实现反序列化漏洞的检测方法。

第四方面，本发明提供了一种计算机可读存储介质，其存储有可由计算机设备执行的计算机程序，当所述程序在计算机设备上运行时，使得所述计算机设备执行反序列化漏洞的检测方法。

在本发明实施例中，在获取到检测漏洞任务后，开始进行针对 T3S 协议的反序列化漏洞进行检测，首先确定 WebLogic 服务器的服务端口是否能够访问，即是否能够通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口，并在确定能够进行访问后，则进一步地确定该 WebLogic 服务器的服务端口是否支持 T3S 协议服务，若支持 T3S 协议服务，再进一步地尝试与 WebLogic 服务器的远程方法调用 RMI 服务连接，若连接成功，则可以确定 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。通过上述方法实现了针

对 T3S 协议的反序列化漏洞检测，并且是在确定出服务端口能够访问且服务端口支持 T3S 协议服务的前提下进行的，避免了无用的检测步骤，提高检测效率。并且本发明实施例中的检测方法及装置还可以针对 T3 协议的 WebLogic Java 反序列化漏洞，实现对多种漏洞的检测；还可以通过任务队列，针对多个 WebLogic 服务器进行漏洞检测，提高漏洞检测效率。

附图说明

图 1 为本发明实施例提供的一种反序列化漏洞的检测方法的流程示意图；

图 2 为本发明实施例提供的一种针对 T3S 协议的反序列化漏洞检测方法的流程示意图；

图 3 为本发明实施例提供的一种针对 T3S 协议的反序列化漏洞检测方法的时序流程示意图；

图 4 为本发明实施例提供的一种针对 T3 协议的反序列化漏洞检测方法的流程示意图；

图 5 为本发明实施例提供的一种针对 T3 协议的反序列化漏洞检测方法的时序流程示意图；

图 6 为本发明实施例提供的一种针对 T3 协议以及 T3S 协议的反序列化漏洞检测方法的流程示意图；

图 7 为本发明实施例提供的一种并行进行反序列化漏洞的检测方法的流程示意图；

图 8 为本发明实施例提供的一种反序列化漏洞的检测装置的结构示意图；

图 9 为本发明实施例提供的一种计算机设备的结构示意图。

具体实施方式

为了使本申请的目的、技术方案及有益效果更加清楚明白，以下结合附图及实施例，对本申请进行进一步详细说明。应当理解，此处所描述的具体实施例仅仅用以解释本申请，并不用于限定本申请。

为了方便理解，下面对本申请实施例中涉及的名词进行解释。

WebLogic，WebLogic 是美国 Oracle 公司出品的一个 application server，确切的说是一个基于 JAVAEE（一种 Java 平台企业版）架构的中间件，WebLogic 是用于开发、集成、部署和管理大型分布式 Web 应用、网络应用和数据库应用的 Java 应用服务器。将 Java 的动态功能和 Java Enterprise（一种 Java 应用程序）标准的安全性引入大型网络应用的开发、集成、部署和管理之中。

序列化与反序列化，序列化的过程就是把一个对象变成可以传输的数据，而反序列化的过程就是把序列化数据再变回对象的过程。

反序列化漏洞，如果 Java 应用对用户输入，即不可信数据做了反序列化处理，那么攻击者可以通过构造恶意输入，让反序列化产生非预期的对象，非预期的对象在产生过程中就有可能带来任意代码并被执行。在利用 WebLogic Java 反序列化漏洞时，需要向 WebLogic 提供服务的端口发送 T3/T3S 协议数据包。其中，T3 是一种优化协议，用于在 WebLogic Server 和其他 Java 程序（包括客户端和其他 WebLogic Server）之间传输数据。WebLogic Server 会跟踪与其连接的每个 Java 虚拟机（JVM），并创建单个 T3 连接用以承载每个 JVM 的所有流量。T3S 是在 SSL（Secure Sockets Layer，安全套接层）上的 WebLogic T3 协议。

在具体实践过程中，本发明的发明人发现，现有技术中的反序列化漏洞的检测方法都

是针对于 T3 协议的，所以无法检测基于 T3S 协议的 Java 反序列化漏洞。所以基于现有技术中的反序列漏洞检测方法的缺点，本申请的提出了一种反序列化漏洞的检测方法，图 1 为本申请实施例提供的一种反序列化漏洞的检测方法的流程示意图，该方法包括以下步骤:

步骤 S101，在获取到检测漏洞任务后，通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口。

本发明实施例中的方法适用于漏洞检测装置，该装置用于检测 WebLogic 服务器中是否至少存在 T3S 协议漏洞，具体的，在本发明实施例中，在获取到检测漏洞任务后，开始针对待检测的 WebLogic 服务器进行漏洞检测。

首先本发明实施例中，可以通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口，从而确定出该服务器端口是否可以提供服务。具体的，在本发明实施例中，通过安全套接层 SSL 协议向 WebLogic 服务器发送数据，若能够接收到 WebLogic 服务器返回的已连接成功消息，则确定成功访问 WebLogic 服务器的服务端口，则该服务器端口可以提供服务；否则认为该 WebLogic 服务器的服务端口不支持访问，也就是不可以提供服务。

步骤 S102，若确定成功访问 WebLogic 服务器的服务端口，则确定所述 WebLogic 服务器的服务端口是否支持 T3S 协议服务。

具体的，在本发明实施例中，当确定成功访问 WebLogic 服务器的服务端口后，还需要确定该服务器端口是否支持 T3S 协议服务。

在本发明实施例中，向 WebLogic 服务器发送 T3S 协议；若确定接收到的 WebLogic 服务器针对 T3S 协议返回的反馈数据与 T3S 协议规定的反馈数据匹配，则确定 WebLogic 服务器的服务端口支持 T3S 协议服务。

也就是说，在本发明实施例中，向 WebLogic 服务器发送 T3S 协议后，收到的反馈数据需要与 T3S 协议规定的反馈数据匹配，若匹配，则认为 WebLogic 服务器的服务端口支持 T3S 协议服务。

一种可选的实施例，对 WebLogic 服务器的 T3S 协议数据发送时进行抓包可获得 T3S 协议连接信息。示例性地，当使用 WebLogic 9.2.0 版本作为客户端时，发送的 T3S 协议连接信息为 “t3S 9.2.0\nAS:255\nHL:19\n\n”，其中 T3S 协议连接信息包含三行有效数据，第一行的内容为 “t3S” 与 WebLogic 客户端的版本号。以上 T3S 协议连接信息可在连接不同版本的 WebLogic 服务器时使用。

示例性地，当使用 WebLogic 10.0.2 版本作为服务器时，也就是说，作为 T3S 漏洞检测的服务器时，在接收到 T3S 协议后，返回的信息为 “HELO:10.0.2.0.false\nAS:2048\nHL:19\n\n”，也就是说，当连接服务器端口后，返回的数据满足以 “HELO:” 开头，且同一行的后续内容为 WebLogic 版本号信息时，则可以认定该服务器端口提供了 WebLogic 的 T3S 协议服务，进一步地，可以认为 WebLogic 服务器的服务端口支持 T3S 协议服务。

步骤 S103，若确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。

具体的，在本发明实施例中，现有技术中针对 T3 协议的反序列漏洞的检测是在进行漏洞检测时会尝试利用 Java 反序列化漏洞在 WebLogic 服务启动新的 RMI(Remote Method Invocation，远程方法调用) 服务，扫描程序在连接 RMI 服务成功后，在执行完毕远程命

令成功时才认定存在漏洞，实际上当连接了 RMI 服务时，就可以认定存在漏洞。所以在本发明实施例中，在确定 WebLogic 服务器的服务端口支持 T3S 协议服务后，就可以尝试与 WebLogic 服务器的远程方法调用 RMI 服务连接，若连接成功，则可以确定 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。

可选的，在步骤 S101 中，为了能够进行多台 WebLogic 服务器的漏洞检测，可以在获取到检测漏洞任务后，将该检测漏洞任务加入到待执行任务序列中，调用线程池中的空闲线程，从待执行任务序列中获取检测漏洞任务后，开始进行漏洞检测。

具体的，在本发明实施例中，在对 WebLogic 服务器进行反序列化漏洞检测时，每执行一次命令，都需要向 WebLogic 服务器发送两次数据，一次数据为 T3S 协议连接信息，另一次为负载数据，该负载数据用于触发 WebLogic 服务器中的漏洞。

具体的，在本发明实施例中，在确定接收到的 WebLogic 服务器针对 T3S 协议返回的反馈数据与 T3S 协议规定的反馈数据匹配之后，确定与 WebLogic 服务器的 RMI 服务连接成功之前，首先需要生成第一漏洞利用数据，并将第一漏洞利用数据发送给 WebLogic 服务器，其中，第一漏洞利用数据用于指示 WebLogic 服务器在 WebLogic 服务器的当前目录下创建远程指令文件目录。进一步地，生成第二漏洞利用数据，其中第二漏洞利用数据中包括远程指令文件，该远程指令文件用于启动 RMI 服务，并将第二漏洞利用数据发送给 WebLogic 服务器，第二漏洞利用数据用于指示 WebLogic 服务器在远程指令文件目录中保存远程指令文件。再进一步地，生成第三漏洞利用数据，其中第三漏洞利用数据用于指示 WebLogic 服务器执行远程指令文件，启动 WebLogic 服务器的 RMI 服务。

在本发明实施例中，第一漏洞利用数据、第二漏洞利用数据、第三漏洞利用数据为 payload 负载数据，也就是说，生成第一漏洞利用数据 payload1 并发送，若 WebLogic 服务器存在 Java 反序列化漏洞，WebLogic 服务器在接收到该 payload1 后，会在当前目录下创建远程指令文件目录。然后生成第二漏洞利用数据 payload2 并发送给 WebLogic 服务器，payload2 中包括远程指令文件，WebLogic 服务器会在远程指令文件目录中保存远程指令，即 jar 包。进一步地，生成第三漏洞利用数据 payload3 并发送 WebLogic 服务器，使 WebLogic 加载远程指令执行 jar（一种软件包文件格式）包，并启动 RMI 服务方法。

当然，在本发明实施例中，还包括其它数据，这些数据用于停止 WebLogic 服务器中的 RMI 服务以及清除 WebLogic 服务器中的 jar 包，具体的，生成 payload4 并发送，使 WebLogic 服务器加载远程指令执行 jar 包并执行停止 RMI 服务方法。生成 payload5 并发送，若 WebLogic 服务器存在 Java 反序列化漏洞，在接收到该 payload5 后，会清空 jar 包文件内容。

在本发明实施例中，发送的 payload 数据包含了三部分，第一部分为数据总长度的十六进制形式，长度为四个字节。第二部分可使用固定数据，对 WebLogic 的 T3 协议数据发送时进行抓包可获得该部分固定数据，并可直接使用。第三部分为 payload 本身。以数据总长度为 1711 为例，第一部分内容应为“00 00 06 af”（1711=0x6AF）。第二部分固定数据中包含可见字符与不可见字符，其中包含了 WebLogic 版本信息等。进一步地，第三部分 payload 为 Java 序列化数据，其中 Java 序列化数据以“ac ed 00 05”开头，其中 0xaced 为 Java 序列化数据的魔数，0x0005 为 Java 序列化的版本。

可选的，在获取所述 WebLogic 服务器的 SSL 证书后，在正常情况下都会对 WebLogic 服务器的 SSL 证书进行验证。但是，由于 WebLogic 服务器的 SSL 证书可能不在检测装置

的受信任证书列表中，所以会导致SSL连接时握手失败，那么就无法连接WebLogic的RMI服务，也就导致无法进行漏洞检测，所以在本发明实施例中，为了能够进行漏洞检测，需要提前将检测装置对WebLogic服务器SSL证书验证的操作屏蔽，使得在连接WebLogic服务器T3S协议的RMI服务时不再出现握手失败问题，继而保证后续的漏洞检测可以正常进行。

进一步地，现有技术中，检测装置会执行主机名验证检查。具体来说，检测装置将从WebLogic服务器接收的数字证书中的CN字段与客户端用于连接WebLogic服务器的URL中的WebLogic服务器名称进行比较。进一步地，CN字段和WebLogic服务器名称必须匹配才能通过主机名验证检查，用于防止中间人攻击。

但是检测装置在进行漏洞检测时，很可能出现SSL证书中的CN字段与WebLogic服务器URL的服务器名称不符的情况，例如通过网际协议IP而不是域名连接WebLogic服务器，在这种情况下检测装置会终止SSL连接操作，导致无法进行漏洞检测。所以在本发明实施例中，可以取消对SSL服务器主机名的验证。

可选的，在本发明实施例中，可以通过在检测装置的JVM启动参数中增加“-Dweblogic.security.SSL.ignoreHostnameVerification=true”，取消检测装置对SSL服务器的主机名验证，使得当SSL证书中的地址与服务器URL的服务器名称不符时也能够正常建立SSL连接，从而正常进行后续的漏洞检测。

也就是说，在本发明实施例中，在获取WebLogic服务器的SSL证书后，在取消对WebLogic服务器的SSL证书验证，以及取消对SSL服务器的主机名验证的过程后，若确定通过T3S协议与所述WebLogic服务器的RMI服务连接成功，则确定与所述WebLogic服务器的远程方法调用RMI服务连接成功。

在本发明实施例中，检测装置除了能够对T3S协议的反序列化漏洞进行检测外，还可以对T3协议的反序列化漏洞进行检测。具体的，在通过传输控制协议TCP成功访问WebLogic服务器的服务端口后，确定WebLogic服务器的服务端口是否支持T3协议服务；进一步地，若确定WebLogic服务器的服务端口支持T3协议服务，且与WebLogic服务器的RMI服务连接成功，则确定所述WebLogic服务器存在T3协议的WebLogic Java反序列化漏洞。

也就是说，在通过传输控制协议TCP成功访问WebLogic服务器的服务端口，确定该服务端口可以访问之后，再确定该服务器端口是否支持T3协议，再进一步的，若支持，则尝试与WebLogic服务器的RMI服务进行连接，若连接成功，则确定存在T3协议的WebLogic Java反序列化漏洞。

示例性地，在检测过程中，同检测T3S漏洞一样，可以通过向WebLogic服务器发送5个payload数据，实现检测漏洞的过程。

可选的，在本发明实施例中，当确定WebLogic服务器存在T3或者T3S协议的反序列化漏洞后，还需要将漏洞信息进行记录，本发明实施例中，记录的漏洞信息至少包括WebLogic服务器IP、IP对应的地区、WebLogic端口信息、WebLogic版本信息、WebLogic域路径、操作系统名称、操作系统主机名、操作系统用户、Web服务协议（例如HTTP、HTTPS）、网站标题、网站URL、证书信息（包括域名、机构信息等）。

在本发明实施例中，能够通过上述实施例中的方法实现针对T3S协议的反序列化漏洞检测，并首先确定了所连接的WebLogic服务器服务端口是否提供T3/T3S协议服务，若所

连接端口不提供 T3/T3S 协议服务，则不会存在 Java 反序列化漏洞，也不需要进行反序列化漏洞检测，所以可以对不需要检测的端口跳过，以提高检测效率。进一步地，由于在检测过程中建立的文件目录是在 WebLogic 服务器的当前目录中的，所以使用的 payload 可以支持多种操作系统，不限于 Windows 与 Linux，还支持 AIX、SunOS、HP-UX 等操作系统，兼容性更好。本发明实施例中，在检测时不需要提前设置目标机器的操作系统，对于不同的操作系统均能够支持检测，提高检测效率；并且在尝试利用漏洞在 WebLogic 服务启动 RMI 服务，当检测装置连接 RMI 服务成功时，就认定存在漏洞，精简了检测步骤，且不会影响检测效果，能够提高漏洞检测效率。本发明实施例中的检测方法还可以支持并行对不同的服务器进行漏洞检测，提高检测效率。

为了更好的解释本申请实施例，下面结合具体的实施场景描述本申请实施例提供的一种反序列化漏洞的检测方法，该方法应用在检测装置中，该检测方法的具体流程针对 T3S 协议的反序列化漏洞进行检测，具体过程如图 2 所示：

步骤 S201，接收检测漏洞任务。

步骤 S202，向 WebLogic 服务器发送安全套接层 SSL 协议。

步骤 S203，确定 WebLogic 服务器的服务端口是否可以访问，若是，则执行步骤 S204；否则，执行步骤 S205。

步骤 S204，向 WebLogic 服务器发送 T3S 协议数据，执行步骤 S206。

步骤 S205，退出漏洞检测流程。

步骤 S206，确定 WebLogic 服务器的服务端口是否支持 T3S 协议，若支持，则执行步骤 S207，否则执行步骤 S205。

步骤 S207，生成 payload1 并发送，以使 WebLogic 服务器在当前目录下创建远程指令文件保存目录。

步骤 S208，生成 payload2 并发送，以使 WebLogic 服务器在创建的远程指令文件保存目录下保存 jar 包。

步骤 S209，生成 payload3 并发送，以使 WebLogic 服务器执行 jar 包启动 RMI 服务。

步骤 S210，连接 WebLogic 服务器的 RMI 服务。

步骤 S211，确定是否能够连接 WebLogic 服务器的 RMI 服务，若能连接，则执行步骤 S212；否则执行步骤 S213。

步骤 S212，确定 WebLogic 服务器的漏洞信息并记录，执行步骤 S215。

步骤 S213，确定 WebLogic 服务器中不存在 T3S 协议的反序列化漏洞，执行步骤 S205。

步骤 S214，生成 payload4 并发送，以使 WebLogic 服务器停止 RMI 服务。

步骤 S215，生成 payload5 并发送，以使 WebLogic 服务器清空 jar 包，执行步骤 S205。

本发明实施例提供另一种实施场景，来描述本申请实施例提供的另一种反序列化漏洞的检测方法，该方法应用在检测装置中，该检测方法的具体时序图针对 T3S 协议的反序列化漏洞进行检测，具体过程如图 3 所示。

本发明实施例还提供一种针对 T3 协议的反序列化漏洞检测方法，该方法与针对 T3S 协议的反序列化漏洞检测方法的过程相似，只是使用传输控制协议 TCP 与 WebLogic 服务器进行连接，并向 WebLogic 服务器发送的是 T3 协议，具体过程如图 4 所示。

同样的，本发明实施例还提供一种针对 T3 协议的反序列化漏洞的检测的时序图，如图 5 所示。

本发明实施例还提供一种实施场景描述本申请实施例提供的一种反序列化漏洞的检测方法，该方法应用在检测装置中，该检测方法适用于 T3 协议以及 T3S 协议的反序列化漏洞检测，具体过程如图 6 所示，包括:

步骤 S601，接受检测漏洞任务。

步骤 S602，使用安全套接层 SSL 协议连接 WebLogic 服务器端口。

步骤 S603，确定是否能够连接 WebLogic 服务器端口，若能，则执行步骤 S604；否则执行步骤 S605。

步骤 S604，向 WebLogic 服务器发送 T3S 协议，执行步骤 S606。

步骤 S605，使用传输控制协议 TCP 连接 WebLogic 服务器端口，执行步骤 S607。

步骤 S606，确定 WebLogic 服务器的端口是否支持 T3S 协议，若支持，则执行步骤 S608；否则，执行步骤 S609。

步骤 S607，确定 WebLogic 服务器的端口是否支持 T3 协议，若支持，则执行步骤 S610；否则执行步骤 S609。

步骤 S608，确定 WebLogic 服务器是否存在 T3S 协议的反序列化漏洞，若存在，则执行步骤 S611；否则执行步骤 S609。

步骤 S609，退出漏洞检测任务。

步骤 S611，记录 WebLogic 服务器的 T3S 协议的反序列化漏洞的漏洞信息，执行步骤 S609。

步骤 S612，记录 WebLogic 服务器的 T3 协议的反序列化漏洞的漏洞信息，执行步骤 S609。

上述实施例只是已先确定 WebLogic 服务器的服务端口是否提供 SSL 服务进行阐述的，也可以首先确定 WebLogic 服务器的服务端口是否提供传输控制协议 TCP 服务。上述检测 T3S 协议的反序列化漏洞以及检测 T3 协议的反序列化漏洞的过程没有顺序的规定。

本发明实施例还提供一种实施场景，用于描述本申请实施例提供的一种反序列化漏洞的检测方法，该方法应用在检测装置中，该检测方法适用于检测多个 WebLogic 服务器中是否存在 T3 协议以及 T3S 协议的反序列化漏洞，具体过程如图 7 所示。在图 7 中，一个检测装置可以检测多个 WebLogic 服务器中是否存在 T3 协议以及 T3S 协议的反序列化漏洞，当接收到检测漏洞任务后，首先将检测漏洞加入到待执行任务序列中，然后检测装置中的空闲线程会从待执行任务序列中获取检测漏洞任务，进一步地，针对一个 WebLogic 服务器中是否存在 T3 协议以及 T3S 协议的反序列化漏洞进行检测，若存在漏洞，则返回检测到的漏洞信息，并且可以向用户进行漏洞信息展示。

基于相同的技术构思，本申请实施例提供了一种反序列化漏洞的检测装置，如图 8 所示，该装置 800 包括:

安全套接层 SSL 协议连接单元 801，用于在获取到检测漏洞任务后，通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口。

T3S 协议服务确定单元 802，用于在确定成功访问 WebLogic 服务器的服务端口时，进一步确定所述 WebLogic 服务器的服务端口是否支持 T3S 协议服务。

漏洞确定单元 803，用于在确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功时，进一步确定所述 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。

进一步地，可选的，所述 T3S 协议服务确定单元 802 具体用于:

向所述 WebLogic 服务器发送 T3S 协议数据。

若确定接收到的所述 WebLogic 服务器针对所述 T3S 协议返回的反馈数据与 T3S 协议规定的反馈数据匹配，则确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务。

可选的，所述装置还包括:

数据生成单元 804，用于生成第一漏洞利用数据，并将所述第一漏洞利用数据发送给所述 WebLogic 服务器，所述第一漏洞利用数据用于指示所述 WebLogic 服务器在所述 WebLogic 服务器的当前目录下创建远程指令文件目录。

在确定所述 WebLogic 服务器在当前目录下已创建远程指令文件目录后，生成第二漏洞利用数据，所述第二漏洞利用数据中包括远程指令文件，所述远程指令文件用于启动 RMI 服务，并将所述第二漏洞利用数据发送给所述 WebLogic 服务器，其中，所述第二漏洞利用数据用于指示所述 WebLogic 服务器在所述远程指令文件目录中保存所述远程指令文件。

进一步地，在确定所述 WebLogic 服务器已在所述远程指令文件目录中保存所述远程指令文件后，生成第三漏洞利用数据，其中，所述第三漏洞利用数据用于指示所述 WebLogic 服务器执行所述远程指令文件，启动所述 WebLogic 服务器的 RMI 服务。

可选的，所述漏洞确定单元 803 具体用于:

在获取所述 WebLogic 服务器的 SSL 证书后，则确定通过 T3S 协议与所述 WebLogic 服务器的 RMI 服务连接成功。

可选的，所述装置还包括 T3 协议漏洞检测单元 805，所述 T3 协议漏洞检测单元 805 用于:

在通过传输控制协议 TCP 成功访问 WebLogic 服务器的服务端口后，进一步地确定所述 WebLogic 服务器的服务端口是否支持 T3 协议服务。

若确定所述 WebLogic 服务器的服务端口支持 T3 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3 协议的 WebLogic Java 反序列化漏洞。

可选的，所述装置还包括获取单元 808，所述获取单元 808 用于:

将所述检测漏洞任务加入到待执行任务队列中，并使用线程池中的空闲线程从所述执行任务队列获取所述检测漏洞任务。

基于相同的技术构思，本申请实施例提供了一种计算机设备，如图 9 所示，包括至少一个处理器 901，以及与至少一个处理器连接的存储器 902，本申请实施例中不限定处理器 901 与存储器 902 之间的具体连接介质，图 9 中处理器 901 和存储器 902 之间通过总线连接为例。总线可以分为地址总线、数据总线、控制总线等。

在本申请实施例中，存储器 902 存储有可被至少一个处理器 901 执行的指令，至少一个处理器 901 通过执行存储器 902 存储的指令，可以执行前述的反序列化漏洞的检测方法中所包括的各个步骤。

其中，处理器 901 是计算机设备的控制中心，可以利用各种接口和线路连接终端设备的各个部分，通过运行或执行存储在存储器 902 内的指令以及调用存储在存储器 902 内的数据，从而获得客户端地址。可选的，处理器 901 可包括一个或多个处理单元，处理器 901 可集成应用处理器和调制解调处理器，其中，应用处理器主要处理操作系统、用户界面和

应用程序等，调制解调处理器主要处理无线通信。可以理解的是，上述调制解调处理器也可以不集成到处理器 901 中。在一些实施例中，处理器 901 和存储器 902 可以在同一芯片上实现，在一些实施例中，它们也可以在独立的芯片上分别实现。

处理器 901 可以是通用处理器，例如中央处理器（CPU）、数字信号处理器、专用集成电路（Application Specific Integrated Circuit，ASIC）、现场可编程门阵列或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件，可以实现或者执行本申请实施例中公开的各方法、步骤及逻辑框图。通用处理器可以是微处理器或者任何常规的处理器等。结合本申请实施例所公开的方法的步骤可以直接体现为硬件处理器执行完成，或者用处理器中的硬件及软件模块组合执行完成。

存储器 902 作为一种非易失性计算机可读存储介质，可用于存储非易失性软件程序、非易失性计算机可执行程序以及模块。存储器 902 可以包括至少一种类型的存储介质，例如可以包括闪存、硬盘、多媒体卡、卡型存储器、随机访问存储器（Random Access Memory，RAM）、静态随机访问存储器（Static Random Access Memory，SRAM）、可编程只读存储器（Programmable Read Only Memory，PROM）、只读存储器（Read Only Memory，ROM）、带电可擦除可编程只读存储器（Electrically Erasable Programmable Read-Only Memory，EEPROM）、磁性存储器、磁盘、光盘等等。存储器 902 是能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其他介质，但不限于此。本申请实施例中的存储器 902 还可以是电路或者其它任意能够实现存储功能的装置，用于存储程序指令和/或数据。

基于相同的技术构思，本申请实施例提供了一种计算机可读存储介质，其存储有可由计算机设备执行的计算机程序，当所述程序在计算机设备上运行时，使得所述计算机设备执行反序列化漏洞的检测方法的各个步骤。

本领域普通技术人员可以理解：实现上述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计算机可读取存储介质中，该程序在执行时，执行包括上述方法实施例的步骤；而前述的存储介质包括：移动存储设备、只读存储器（ROM，Read-Only Memory）、随机存取存储器（RAM，Random Access Memory）、磁碟或者光盘等各种可以存储程序代码的介质。

或者，本申请上述集成的单元如果以软件功能模块的形式实现并作为独立的产品销售或使用时，也可以存储在一个计算机可读取存储介质中。基于这样的理解，本申请实施例的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质中，包括若干指令用以使得一台计算机设备（可以是个人计算机、服务器、或者网络设备等）执行本申请各个实施例所述方法的全部或部分。而前述的存储介质包括：移动存储设备、ROM、RAM、磁碟或者光盘等各种可以存储程序代码的介质。

以上所述，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到变化或替换，都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以所述权利要求的保护范围为准。

权 利 要 求

1、一种反序列化漏洞的检测方法，其特征在于，所述方法包括:

在获取到检测漏洞任务后，通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口;

若确定成功访问所述 WebLogic 服务器的服务端口，则确定所述 WebLogic 服务器的服务端口是否支持 T3S 协议服务;

若确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。

2、根据权利要求 1 所述的方法，其特征在于，所述确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，包括:

向所述 WebLogic 服务器发送 T3S 协议数据;

若确定接收到的所述 WebLogic 服务器针对所述 T3S 协议返回的反馈数据与 T3S 协议规定的反馈数据匹配，则确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务。

3、根据权利要求 1 所述的方法，其特征在于，所述确定所述 WebLogic 服务器的服务端口支持T3S协议服务后，与所述WebLogic服务器的远程方法调用RMI服务连接成功前，还包括:

生成第一漏洞利用数据，并将所述第一漏洞利用数据发送给所述 WebLogic 服务器，所述第一漏洞利用数据用于指示所述WebLogic服务器在所述WebLogic服务器的当前目录下创建远程指令文件目录;

生成第二漏洞利用数据，所述第二漏洞利用数据中包括远程指令文件，所述远程指令文件用于启动 RMI 服务，并将所述第二漏洞利用数据发送给所述 WebLogic 服务器，所述第二漏洞利用数据用于指示所述 WebLogic 服务器在所述远程指令文件目录中保存所述远程指令文件;

生成第三漏洞利用数据，所述第三漏洞利用数据用于指示所述 WebLogic 服务器执行所述远程指令文件，启动所述 WebLogic 服务器的 RMI 服务。

4、根据权利要求 1~3 中任意一项所述的方法，其特征在于，所述与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，包括:

在获取所述 WebLogic 服务器的 SSL 证书后，若确定通过 T3S 协议与所述 WebLogic 服务器的 RMI 服务连接成功，则确定与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功。

5、根据权利要求 1 所述的方法，其特征在于，所述方法还包括:

在通过传输控制协议 TCP 成功访问 WebLogic 服务器的服务端口后，确定所述 WebLogic 服务器的服务端口是否支持 T3 协议服务;

若确定所述 WebLogic 服务器的服务端口支持 T3 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3 协议的 WebLogic Java 反序列化漏洞。

6、根据权利要求 1 所述的方法，其特征在于，所述获取检测漏洞任务，包括:

将所述检测漏洞任务加入到待执行任务队列中，并使用线程池中的空闲线程从所述执行任务队列获取所述检测漏洞任务。

7、一种反序列化漏洞检测装置，其特征在于，所述装置包括:

安全套接层 SSL 协议连接单元，用于在获取到检测漏洞任务后，通过安全套接层 SSL 协议访问 WebLogic 服务器的服务端口;

T3S 协议服务确定单元，用于若确定成功访问所述 WebLogic 服务器的服务端口，则确定所述 WebLogic 服务器的服务端口是否支持 T3S 协议服务;

漏洞确定单元，用于若确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3S 协议的 WebLogic Java 反序列化漏洞。

8、根据权利要求 7 所述的装置，其特征在于，所述 T3S 协议服务确定单元具体用于:

向所述 WebLogic 服务器发送 T3S 协议数据;

若确定接收到的所述 WebLogic 服务器针对所述 T3S 协议返回的反馈数据与 T3S 协议规定的反馈数据匹配，则确定所述 WebLogic 服务器的服务端口支持 T3S 协议服务。

9、根据权利要求 7 所述的装置，其特征在于，所述装置还包括:

数据生成单元，用于生成第一漏洞利用数据，并将所述第一漏洞利用数据发送给所述 WebLogic 服务器，所述第一漏洞利用数据用于指示所述 WebLogic 服务器在所述 WebLogic 服务器的当前目录下创建远程指令文件目录;

生成第二漏洞利用数据，所述第二漏洞利用数据中包括远程指令文件，所述远程指令文件用于启动 RMI 服务，并将所述第二漏洞利用数据发送给所述 WebLogic 服务器，所述第二漏洞利用数据用于指示所述 WebLogic 服务器在所述远程指令文件目录中保存所述远程指令文件;

生成第三漏洞利用数据，所述第三漏洞利用数据用于指示所述 WebLogic 服务器执行所述远程指令文件，启动所述 WebLogic 服务器的 RMI 服务。

10、根据权利要求 7~9 中任意一项所述的装置，其特征在于，所述漏洞确定单元具体用于:

在获取所述 WebLogic 服务器的 SSL 证书后，若确定通过 T3S 协议与所述 WebLogic 服务器的 RMI 服务连接成功，则确定与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功。

11、根据权利要求 7 所述的装置，其特征在于，所述装置还包括 T3 协议漏洞检测单元，所述 T3 协议漏洞检测单元用于:

在通过传输控制协议 TCP 成功访问 WebLogic 服务器的服务端口后，确定所述 WebLogic 服务器的服务端口是否支持 T3 协议服务;

若确定所述 WebLogic 服务器的服务端口支持 T3 协议服务，且与所述 WebLogic 服务器的远程方法调用 RMI 服务连接成功，则确定所述 WebLogic 服务器存在 T3 协议的 WebLogic Java 反序列化漏洞。

12、根据权利要求 11 所述的装置，其特征在于，所述装置还包括获取单元，所述获取单元用于:

将所述检测漏洞任务加入到待执行任务队列中，并使用线程池中的空闲线程从所述执

行任务队列获取所述检测漏洞任务。

13、一种计算机设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，其特征在于，所述处理器执行所述计算机程序时实现权利要求 1 至 6 任一项所述方法的步骤。

14、一种计算机可读存储介质，其特征在于，其存储有可由计算机设备执行的计算机程序，当所述程序在计算机设备上运行时，使得计算机执行如权利要求 1 至 6 中任一项所述的方法。

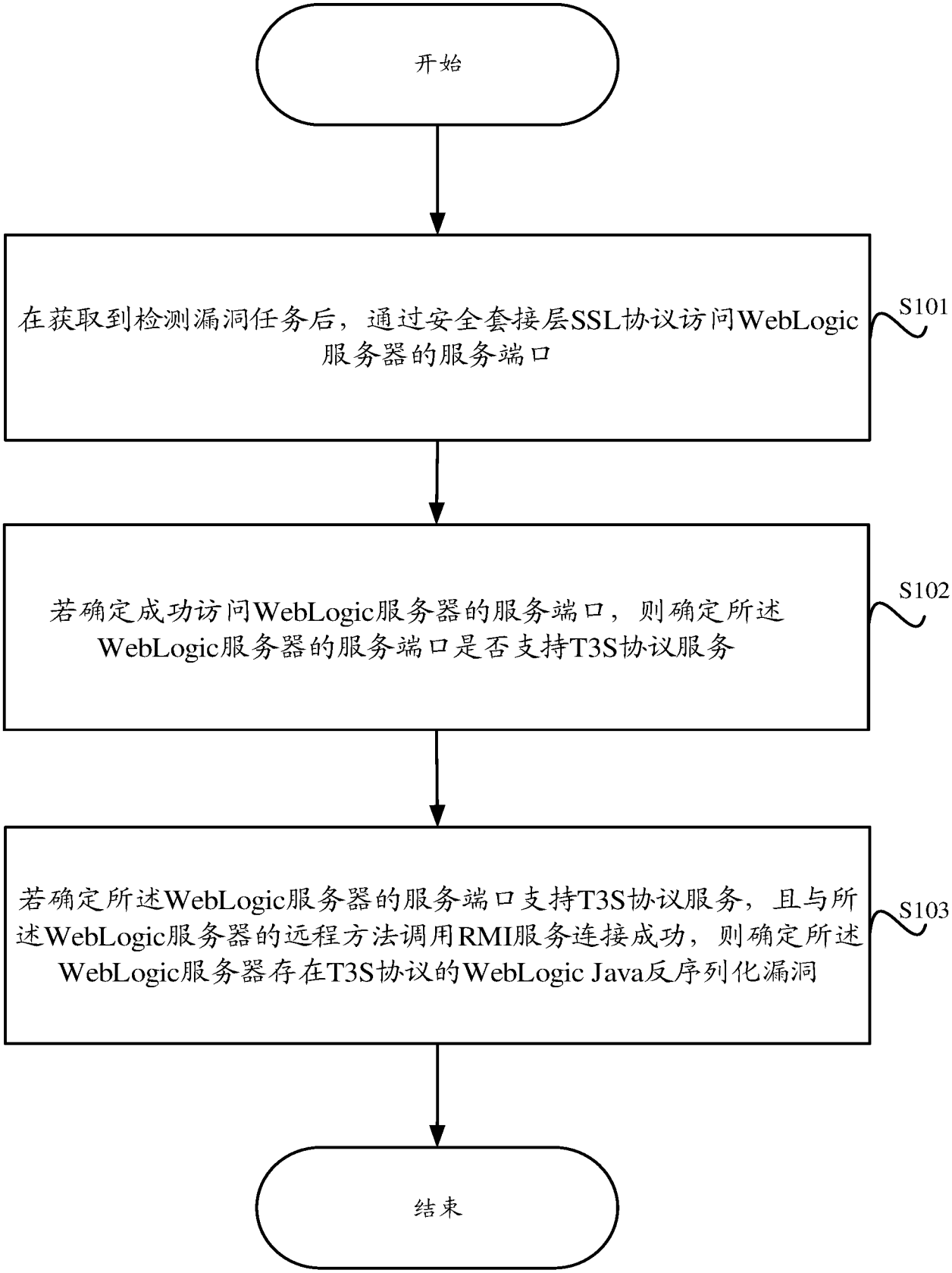

开始
在获取到检测漏洞任务后，通过安全套接层SSL协议访问WebLogic服务器的服务端口
S101
若确定成功访问WebLogic服务器的服务端口，则确定所述WebLogic服务器的服务端口是否支持T3S协议服务
S102
若确定所述WebLogic服务器的服务端口支持T3S协议服务，且与所述WebLogic服务器的远程方法调用RMI服务连接成功，则确定所述WebLogic服务器存在T3S协议的WebLogic Java反序列化漏洞
S103
结束

图 1

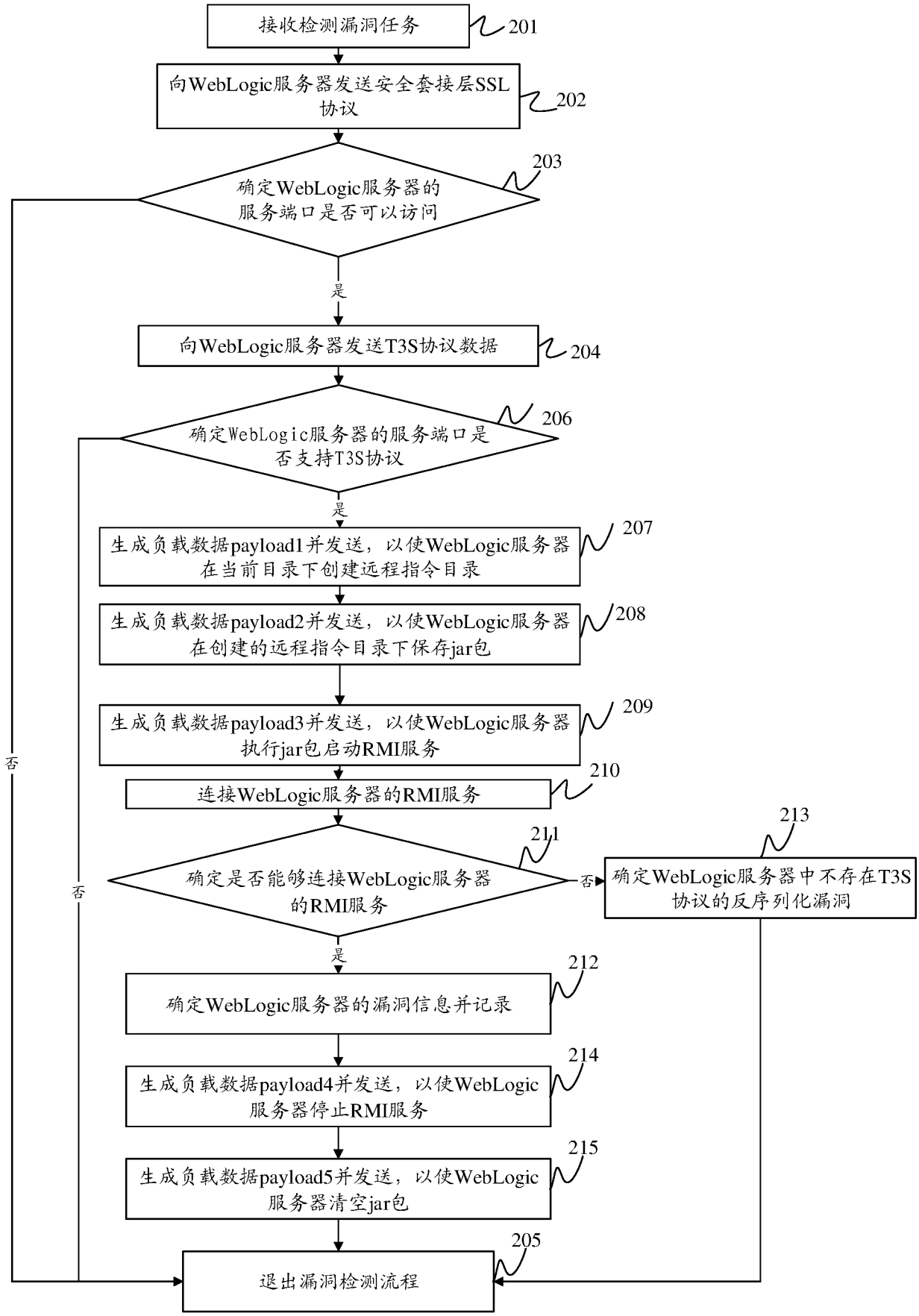

接收检测漏洞任务
201
向WebLogic服务器发送安全套接层SSL协议
202
确定WebLogic服务器的服务端口是否可以访问
203
是
否
向WebLogic服务器发送T3S协议数据
204
确定WebLogic服务器的服务端口是否支持T3S协议
206
是
否
生成负载数据payload1并发送，以使WebLogic服务器在当前目录下创建远程指令目录
207
生成负载数据payload2并发送，以使WebLogic服务器在创建的远程指令目录下保存jar包
208
生成负载数据payload3并发送，以使WebLogic服务器执行jar包启动RMI服务
209
连接WebLogic服务器的RMI服务
210
确定是否能够连接WebLogic服务器的RMI服务
211
否
确定WebLogic服务器中不存在T3S协议的反序列化漏洞
213
是
确定WebLogic服务器的漏洞信息并记录
212
生成负载数据payload4并发送，以使WebLogic服务器停止RMI服务
214
生成负载数据payload5并发送，以使WebLogic服务器清空jar包
215
退出漏洞检测流程
205

图 2

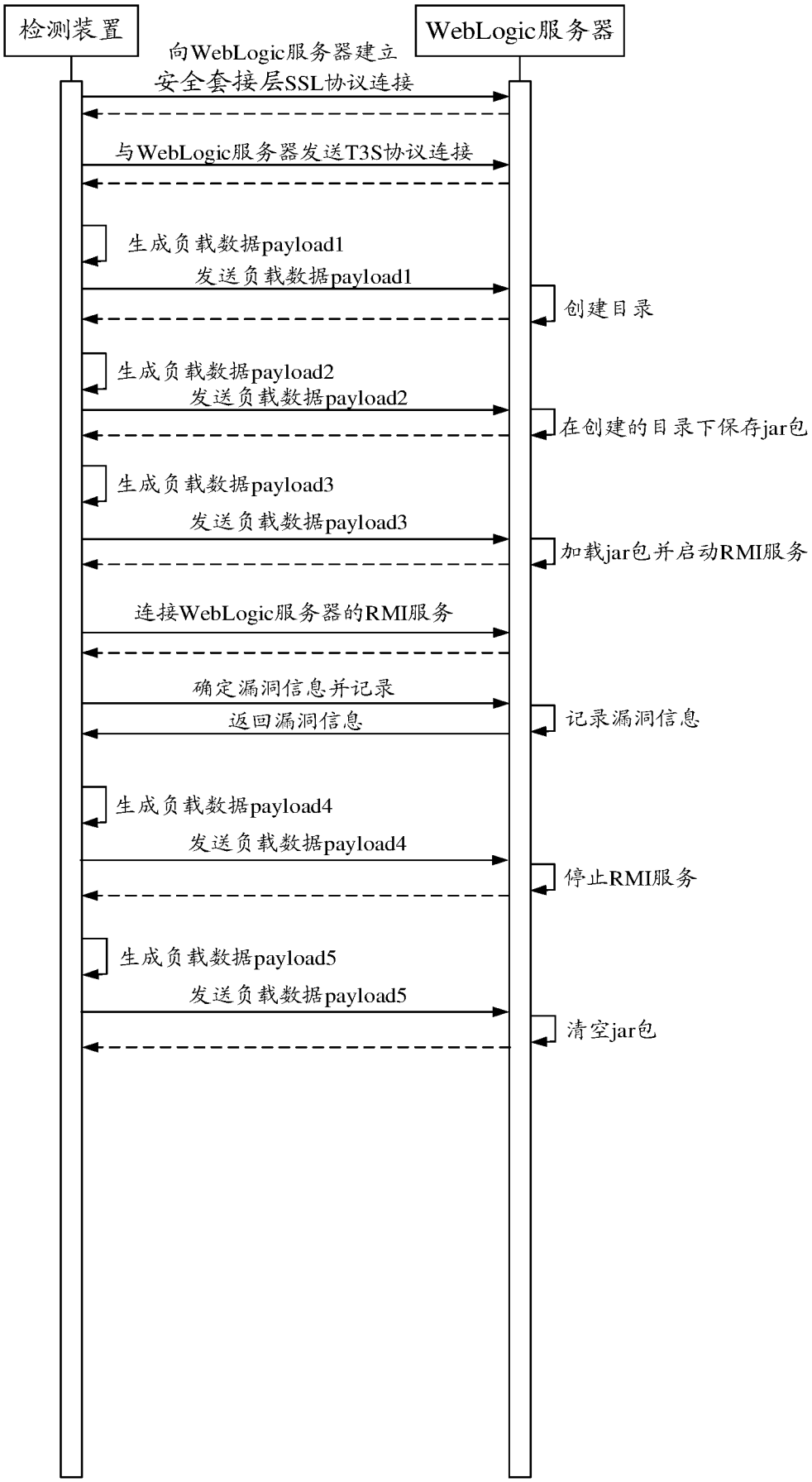

检测装置
WebLogic服务器
向WebLogic服务器建立
安全套接层SSL协议连接
与WebLogic服务器发送T3S协议连接
生成负载数据payload1
发送负载数据payload1
创建目录
生成负载数据payload2
发送负载数据payload2
在创建的目录下保存jar包
生成负载数据payload3
发送负载数据payload3
加载jar包并启动RMI服务
连接WebLogic服务器的RMI服务
确定漏洞信息并记录
返回漏洞信息
记录漏洞信息
生成负载数据payload4
发送负载数据payload4
停止RMI服务
生成负载数据payload5
发送负载数据payload5
清空jar包

图 3

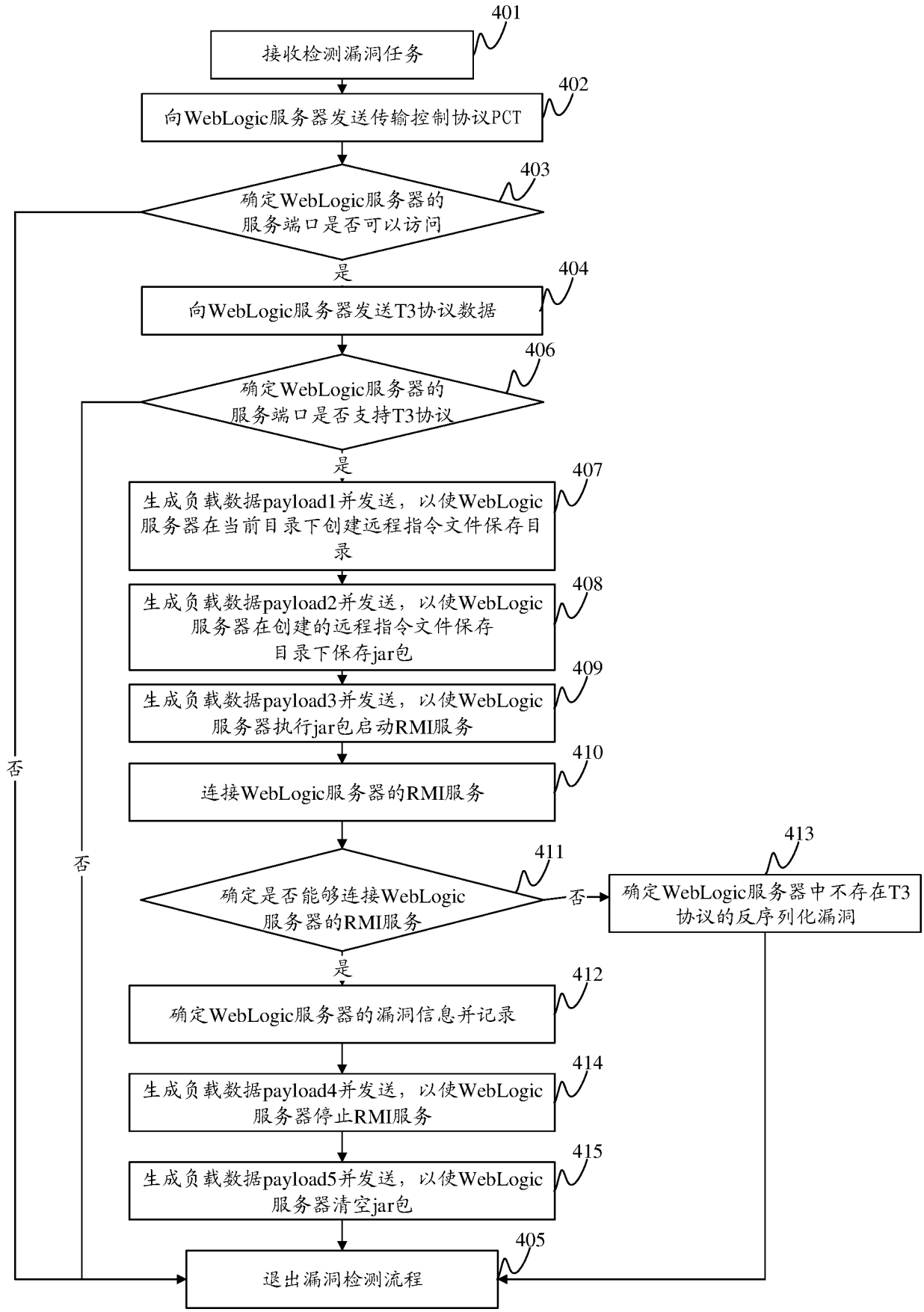

401
接收检测漏洞任务
402
向WebLogic服务器发送传输控制协议PCT
403
确定WebLogic服务器的服务端口是否可以访问
是
否
404
向WebLogic服务器发送T3协议数据
406
确定WebLogic服务器的服务端口是否支持T3协议
是
否
407
生成负载数据payload1并发送，以使WebLogic服务器在当前目录下创建远程指令文件保存目录
408
生成负载数据payload2并发送，以使WebLogic服务器在创建的远程指令文件保存目录下保存jar包
409
生成负载数据payload3并发送，以使WebLogic服务器执行jar包启动RMI服务
410
连接WebLogic服务器的RMI服务
411
确定是否能够连接WebLogic服务器的RMI服务
否
413
确定WebLogic服务器中不存在T3协议的反序列化漏洞
是
412
确定WebLogic服务器的漏洞信息并记录
414
生成负载数据payload4并发送，以使WebLogic服务器停止RMI服务
415
生成负载数据payload5并发送，以使WebLogic服务器清空jar包
405
退出漏洞检测流程

图 4

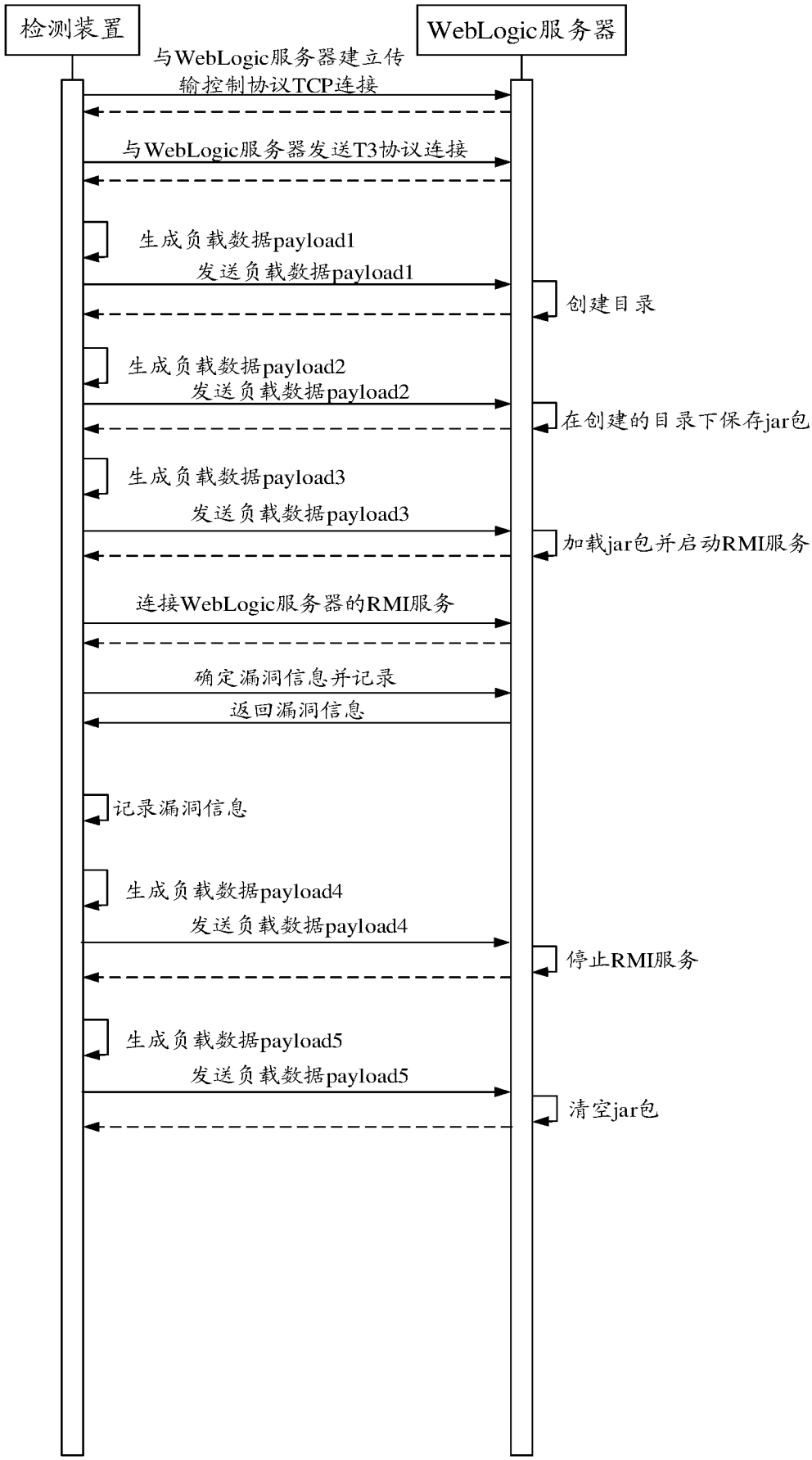

检测装置
WebLogic服务器
与WebLogic服务器建立传输控制协议TCP连接
与WebLogic服务器发送T3协议连接
生成负载数据payload1
发送负载数据payload1
创建目录
生成负载数据payload2
发送负载数据payload2
在创建的目录下保存jar包
生成负载数据payload3
发送负载数据payload3
加载jar包并启动RMI服务
连接WebLogic服务器的RMI服务
确定漏洞信息并记录
返回漏洞信息
记录漏洞信息
生成负载数据payload4
发送负载数据payload4
停止RMI服务
生成负载数据payload5
发送负载数据payload5
清空jar包

图 5

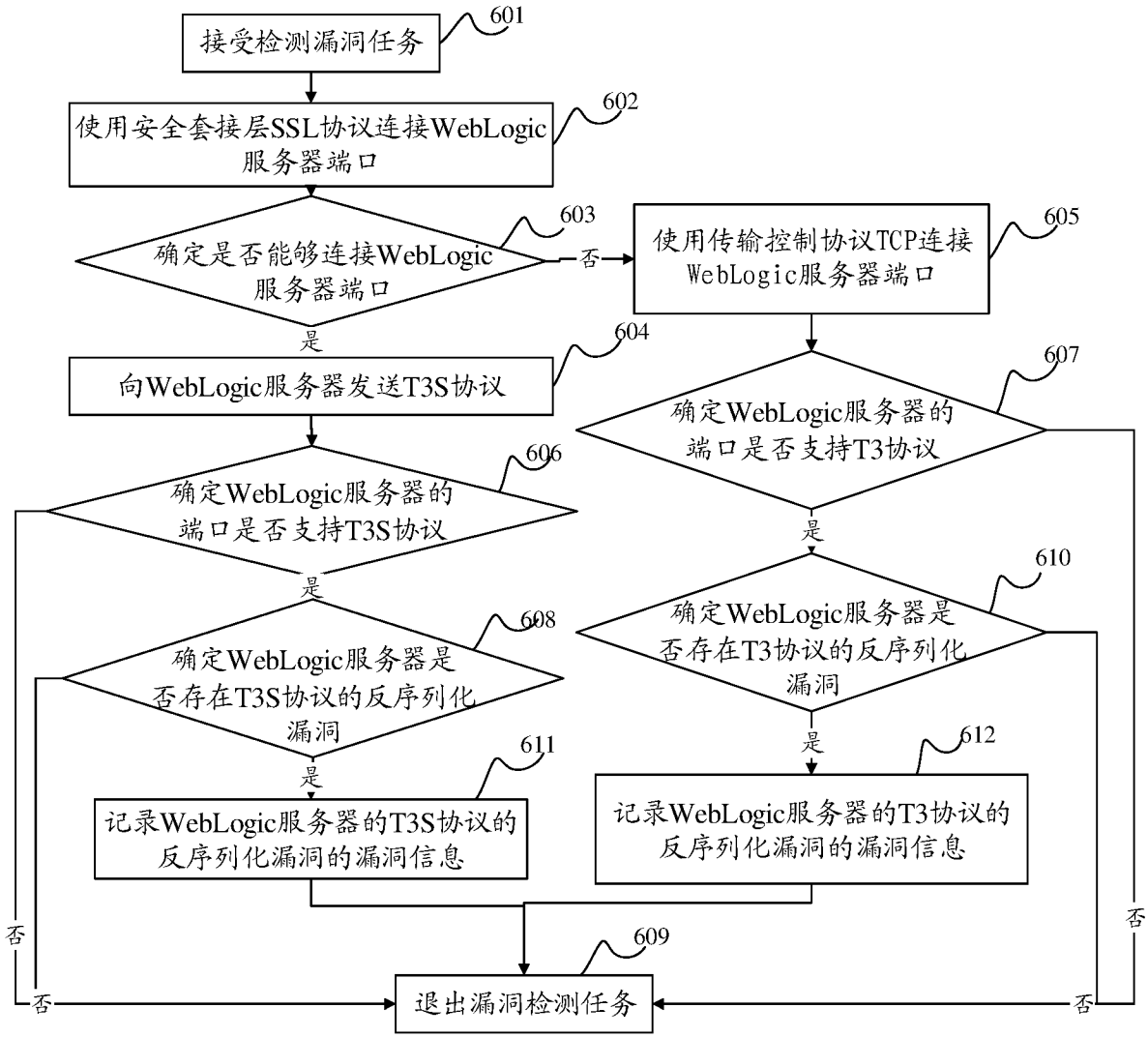

接受检测漏洞任务
601
使用安全套接层SSL协议连接WebLogic服务器端口
602
确定是否能够连接WebLogic服务器端口
603
否
使用传输控制协议TCP连接WebLogic服务器端口
605
是
向WebLogic服务器发送T3S协议
604
确定WebLogic服务器的端口是否支持T3协议
607
确定WebLogic服务器的端口是否支持T3S协议
606
是
是
确定WebLogic服务器是否存在T3协议的反序列化漏洞
610
确定WebLogic服务器是否存在T3S协议的反序列化漏洞
608
是
是
记录WebLogic服务器的T3协议的反序列化漏洞的漏洞信息
612
记录WebLogic服务器的T3S协议的反序列化漏洞的漏洞信息
611
否
否
否
退出漏洞检测任务
609

图 6

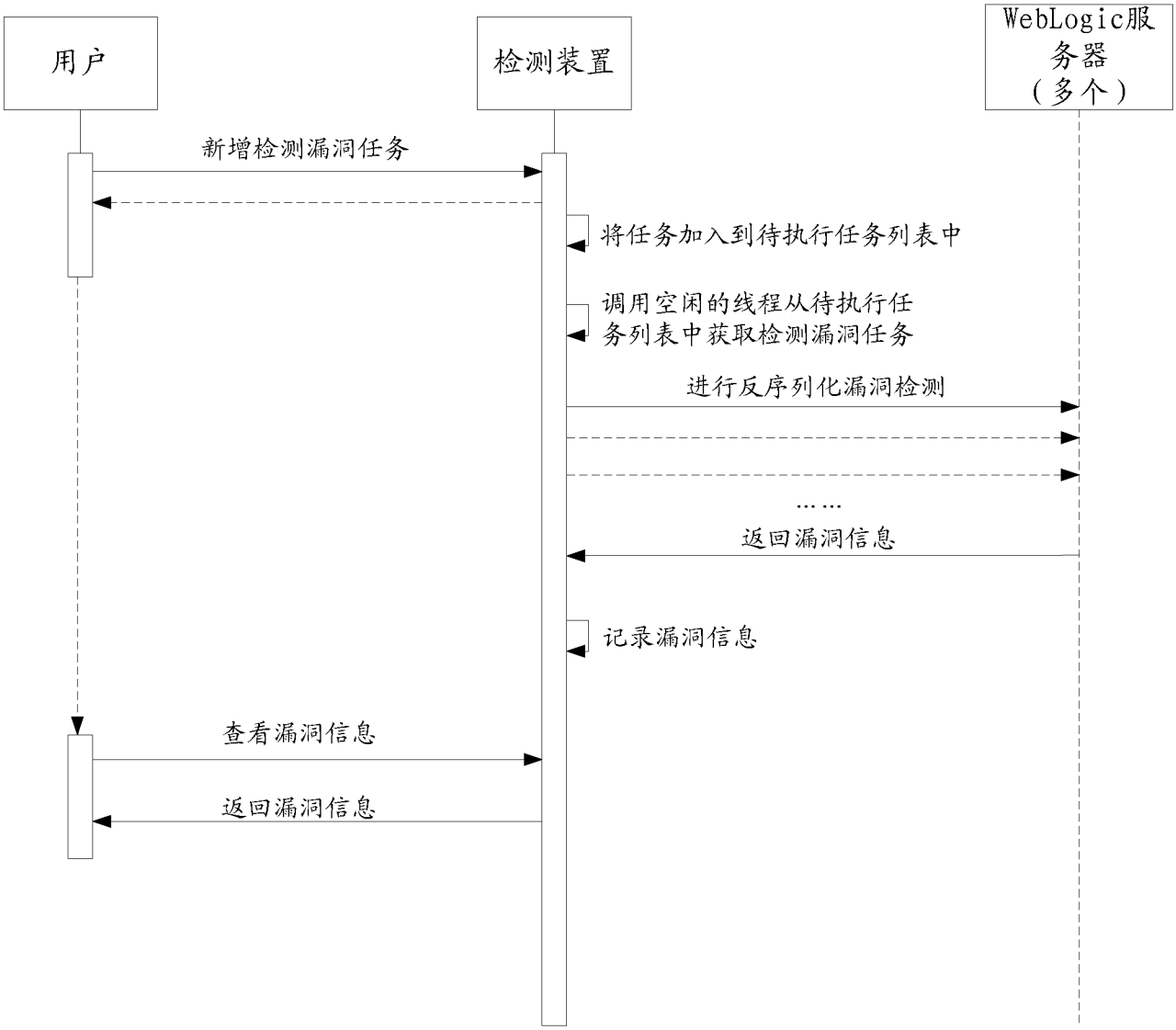

用户
检测装置
WebLogic服务器（多个）
新增检测漏洞任务
将任务加入到待执行任务列表中
调用空闲的线程从待执行任务列表中获取检测漏洞任务
进行反序列化漏洞检测
……
返回漏洞信息
记录漏洞信息
查看漏洞信息
返回漏洞信息

图 7

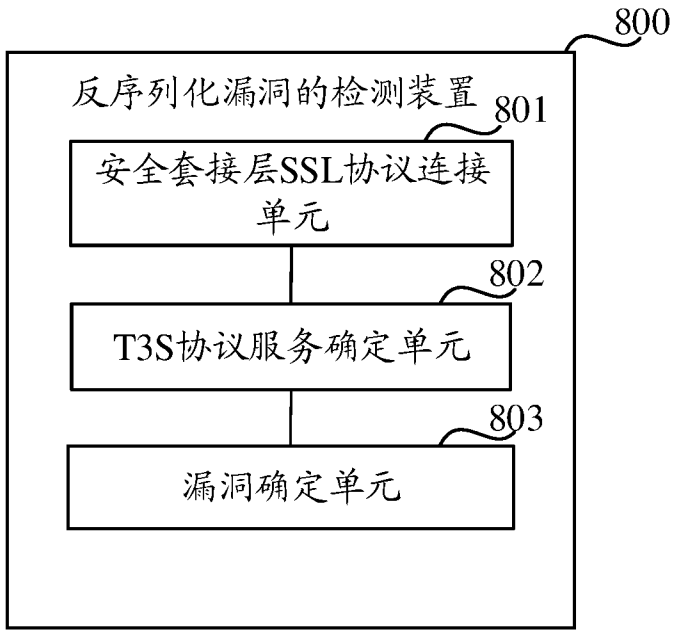

800
反序列化漏洞的检测装置
801
安全套接层SSL协议连接单元
802
T3S协议服务确定单元
803
漏洞确定单元

图 8

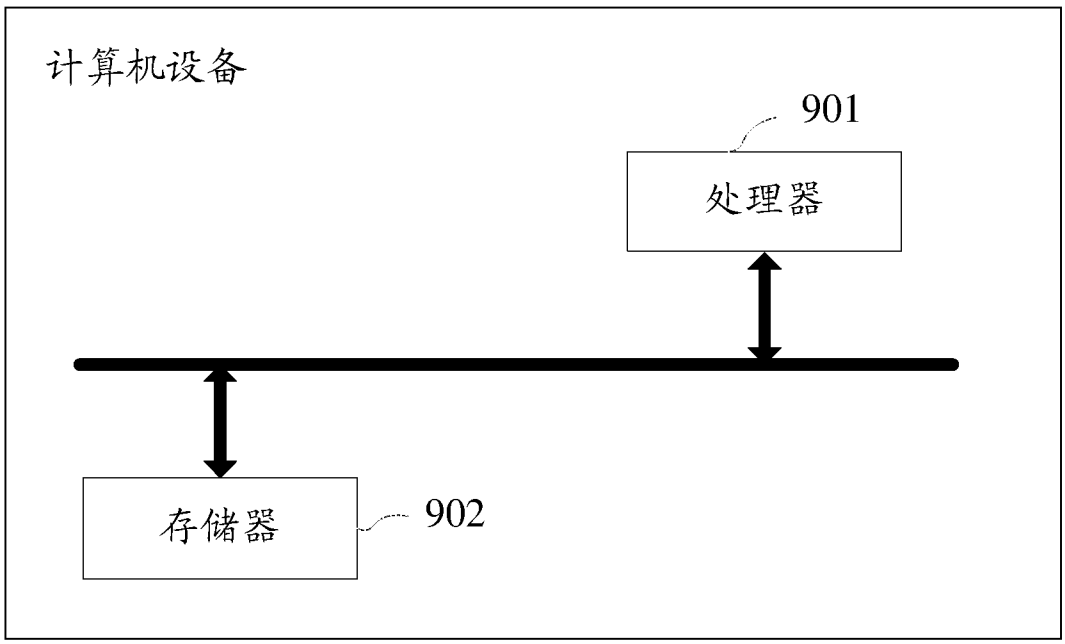

计算机设备
901
处理器
存储器
902

图 9

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2020/096901

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/57(2013.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT, WPI, EPODOC, CNKI, IEEE: 反序列, 漏洞, T3S, 安全套接层, 端口, 协议, 远程方法调用, deserialization, bug, SSL, port, interface, protocol, RMI

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 110276202 A (SHENZHEN QIANHAI WEBANK CO., LTD.) 24 September 2019 (2019-09-24) claims 1-14	1-14
X	IT资讯 (IT Index. net). "如何控制开放HTTPS服务的weblogic服务器 (Non-official translation: How to Enable HTTPS Service on WebLogic Server)" *https://itindex2.rssing.com/chan-27974975/all_p224.html,* 16 March 2016 (2016-03-16), pp. 1-5	1-14
A	CN 106096417 A (INSTITUTE OF ELECTRIC POWER SCIENCES, STATE GRID CHONGQING ELECTRIC POWER COMPANY et al.) 09 November 2016 (2016-11-09) entire document	1-14
A	CN 109344622 A (HANGZHOU DPTECH TECHNOLOGIES CO., LTD.) 15 February 2019 (2019-02-15) entire document	1-14
A	CN 109857669 A (HANGZHOU XIAODAO TECHNOLOGY CO., LTD.) 07 June 2019 (2019-06-07) entire document	1-14

☑ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
26 August 2020	**25 September 2020**
Name and mailing address of the ISA/CN **China National Intellectual Property Administration (ISA/CN)** **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China** Facsimile No. **(86-10)62019451**	Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2020/096901

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2014067780 A1 (NETSUITE INC.) 06 March 2014 (2014-03-06) entire document	1-14

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2020/096901

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 110276202 A	24 September 2019	None	
CN 106096417 A	09 November 2016	None	
CN 109344622 A	15 February 2019	None	
CN 109857669 A	07 June 2019	None	
US 2014067780 A1	06 March 2014	None	

国际检索报告

国际申请号
PCT/CN2020/096901

A. 主题的分类

G06F 21/57(2013.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

CNPAT, WPI, EPODOC, CNKI, IEEE: 反序列，漏洞，T3S，安全套接层，端口，协议，远程方法调用，deserialization, bug, SSL, port, interface, protocol, RMI

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
PX	CN 110276202 A (深圳前海微众银行股份有限公司) 2019年 9月 24日 (2019-09-24) 权利要求1-14	1-14
X	IT资讯,. "如何控制开放HTTPS服务的weblogic服务器," https://itindex2.rssing.com/chan-27974975/all_p224.html, 2016年 3月 16日 (2016-03-16), 第1-5页	1-14
A	CN 106096417 A (国网重庆市电力公司电力科学研究院 等) 2016年 11月 9日 (2016-11-09) 全文	1-14
A	CN 109344622 A (杭州迪普科技股份有限公司) 2019年 2月 15日 (2019-02-15) 全文	1-14
A	CN 109857669 A (杭州孝道科技有限公司) 2019年 6月 7日 (2019-06-07) 全文	1-14
A	US 2014067780 A1 (NETSUITE INC.) 2014年 3月 6日 (2014-03-06) 全文	1-14

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2020年 8月 26日	2020年 9月 25日
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 杨继爽 电话号码 86-(10)-53961357

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2020/096901

检索报告引用的专利文件	公布日 (年/月/日)	同族专利	公布日 (年/月/日)
CN 110276202 A	2019年 9月 24日	无	
CN 106096417 A	2016年 11月 9日	无	
CN 109344622 A	2019年 2月 15日	无	
CN 109857669 A	2019年 6月 7日	无	
US 2014067780 A1	2014年 3月 6日	无	