(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 105471824 A
(43)申请公布日 2016.04.06

(21)申请号 201410446702.2

(22)申请日 2014.09.03

(71)申请人 阿里巴巴集团控股有限公司
地址 英属开曼群岛大开曼

(72)发明人 王耀耀

(74)专利代理机构 北京三友知识产权代理有限公司 11127
代理人 党晓林 李永强

(51)Int.Cl.
H04L 29/06(2006.01)
G06F 3/048(2013.01)

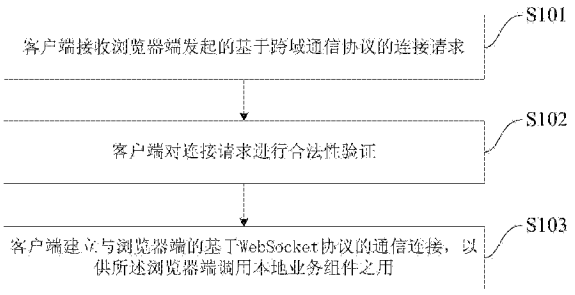
权利要求书2页 说明书6页 附图2页

(54)发明名称

实现浏览器调用本地业务组件的方法、装置及系统

(57)摘要

本申请实施例提供了一种实现浏览器调用本地业务组件的方法、装置及系统。该方法包括客户端接收浏览器端发起的基于跨域通信协议的连接请求；所述客户端对所述连接请求进行合法性验证；如果所述连接请求通过所述合法性验证，则所述客户端建立与所述浏览器端的基于所述跨域通信协议的通信连接，以供所述浏览器端调用本地业务组件之用。本申请实施例可实现多数主流浏览器调用本地业务组件且通信安全性更好。



CN 105471824 A

1. 一种实现浏览器调用本地业务组件的方法，其特征在于，包括以下步骤：

客户端接收浏览器端发起的基于跨域通信协议的连接请求；

所述客户端对所述连接请求进行合法性验证；

如果所述连接请求通过所述合法性验证，则所述客户端建立与所述浏览器端的基于所述跨域通信协议的通信连接，以供所述浏览器端调用本地业务组件之用。

2. 根据权利要求 1 所述的实现浏览器调用本地业务组件的方法，其特征在于，

所述客户端在接收所述浏览器端发起的基于跨域通信协议的连接请求时，按照设定的端口绑定优先顺序从所述端口列表中选择一个端口进行绑定；且所述端口绑定优先顺序与所述浏览器端的端口选择优先顺序一致。

3. 根据权利要求 1 所述的实现浏览器调用本地业务组件的方法，其特征在于，

当所述客户端崩溃时，所述客户端将关于自身的崩溃报告发送给服务器。

4. 根据权利要求 3 所述的实现浏览器调用本地业务组件的方法，其特征在于，所述崩溃报告包括错误代码、出错位置和当前状态。

5. 根据权利要求 1 所述的实现浏览器调用本地业务组件的方法，其特征在于，所述跨域通信协议为 WebSocket 协议、JSONP 协议或 CORS 协议。

6. 根据权利要求 1 所述的实现浏览器调用本地业务组件的方法，其特征在于，所述合法性验证具体为：

所述客户端判断发起所述连接请求的浏览器端所对应的网站是否为网站白名单中的网站。

7. 根据权利要求 6 所述的实现浏览器调用本地业务组件的方法，其特征在于，所述客户端通过定期向服务器查询的方式更新所述网站白名单。

8. 根据权利要求 3 所述的实现浏览器调用本地业务组件的方法，其特征在于，所述崩溃报告可由所述客户端通过操作系统的应用程序编程接口调用该操作系统自带的故障诊断程序来获取。

9. 根据权利要求 1 所述的实现浏览器调用本地业务组件的方法，其特征在于，在建立所述通信连接后，所述浏览器端调用本地业务组件的过程如下：

1)、所述客户端接收浏览器端发送的命令；

2)、所述客户端解析所述命令并分发给对应的本地业务组件；

3)、所述客户端接收所述本地业务组件处理该命令并返回的处理结果；

4)、所述客户端将该处理结果转发给所述浏览器端。

10. 一种实现浏览器调用本地业务组件的装置，其特征在于，该装置为一客户端，其包括：

连接请求接收模块，用于接收浏览器端发起的基于跨域通信协议的连接请求；

合法性验证模块，用于对所述连接请求进行合法性验证；

连接控制模块，用于当所述连接请求通过所述合法性验证时，建立与所述浏览器端的基于所述跨域通信协议的通信连接，以供所述浏览器端调用本地业务组件之用。

11. 根据权利要求 10 所述的实现浏览器调用本地业务组件的装置，其特征在于，该客户端还包括：

端口绑定模块，用于在所述连接请求接收模块接收所述浏览器端发起的基于跨域通信

协议的连接请求时，按照设定的端口绑定优先顺序从所述端口列表中选择一个端口进行绑定；且所述端口绑定优先顺序与所述浏览器端的端口选择优先顺序一致。

12. 根据权利要求 10 所述的实现浏览器调用本地业务组件的装置，其特征在于，该客户端还包括：

崩溃报告处理模块，用于当判断所述客户端崩溃时，将关于自身的崩溃报告发送给服务器。

13. 根据权利要求 12 所述的实现浏览器调用本地业务组件的装置，其特征在于，所述崩溃报告包括错误代码、出错位置和当前状态。

14. 根据权利要求 10 所述的实现浏览器调用本地业务组件的装置，其特征在于，所述跨域通信协议为 WebSocket 协议、JSONP 协议或 CORS 协议。

15. 根据权利要求 10 所述的实现浏览器调用本地业务组件的装置，其特征在于，所述合法性验证具体为：

所述合法性验证模块判断发起所述连接请求的浏览器端所对应的网站是否为网站白名单中的网站。

16. 根据权利要求 15 所述的实现浏览器调用本地业务组件的装置，其特征在于，该客户端还包括：

更新模块，用于通过定期向服务器查询的方式更新所述网站白名单。

17. 根据权利要求 12 所述的实现浏览器调用本地业务组件的装置，所述崩溃报告处理模块通过操作系统的应用程序编程接口调用该操作系统自带的故障诊断程序来获取所述崩溃报告。

18. 一种实现浏览器调用本地业务组件的系统，其特征在于，包括：

浏览器端，用于向客户端发起基于跨域通信协议的连接请求；

客户端，用于对所述连接请求进行合法性验证，如果所述连接请求通过所述合法性验证，则所述客户端建立与所述浏览器端的基于所述跨域通信协议的通信连接，以供所述浏览器端调用本地业务组件之用。

19. 根据权利要求 18 所述的实现浏览器调用本地业务组件的系统，其特征在于，该系统还包括：

服务器，用于接收并保存所述客户端发送的崩溃报告。

实现浏览器调用本地业务组件的方法、装置及系统

技术领域

[0001] 本申请涉及通信技术领域，尤其是涉及一种实现浏览器调用本地业务组件的方法、装置及系统。

背景技术

[0002] 诸如NPAPI(Netscape Plugin Application Programming Interface，网景插件应用程序接口）等浏览器插件是用于在浏览器中执行外部应用程序的通用接口。其中，执行外部应用程序也可以称之为调用本地业务组件，所谓的调用本地业务组件可以是安装数字证书、删除数字证书、校验数字证书、获取网卡mac地址等本机环境信息等等。除NPAPI外，目前常见的类似功能的浏览器插件还包括BHO(Browser Helper Object，浏览器辅助对象）、Native Messaging（本地通讯）和JS-Ctypes等等。

[0003] 然而上述浏览器插件也存在一些问题，具体如下：

[0004] 虽然几乎全部桌面端的图形界面浏览器（除IE外）都支持NPAPI，但是，由于NPAPI设计之初没有考虑安全性，NPAPI插件编写不当会导致浏览器崩溃，甚至造成系统被恶意软件攻击；而且NPAPI插件不支持移动设备，目前已濒临被淘汰。目前最新的浏览器（例如chrome，Firefox等）大都不再支持NPAPI，届时将无法使用浏览器插件去获取本地信息及校验网站数字证书。BHO是微软推出的作为浏览器对第三方程序员开放交互接口的业界标准，通过简单的代码就可以进入浏览器领域的交互接口(Interactive Interface)，而且现在很多IE浏览器个性化工具都是利用BHO的来实现，但是，BHO只支持IE浏览器，应用受限。此外，谷歌与在其新的chrome浏览器中加入了一个新的本机通讯(Native Messaging)接口，以替换掉原来的NPAPI。但是，目前Native Messaging只支持chrome浏览器，而且需要在浏览器中安装扩展。类似的，JS-Ctypes只支持Firefox浏览器，而且同样需要在浏览器中安装扩展。

[0005] 因此，在NPAPI濒临淘汰而其他浏览器插件又不具备通用性的情况下，目前亟需一种通用性强且安全性好的可实现多数主流浏览器调用本地业务组件的方案。

发明内容

[0006] 本申请实施例的目的在于提供一种实现浏览器调用本地业务组件的方法、装置及系统，以保证多数主流浏览器可调用本地业务组件的同时兼顾其安全性。

[0007] 为达到上述目的，一方面，本申请实施例提供了一种实现浏览器调用本地业务组件的方法，包括以下步骤：

[0008] 客户端接收浏览器端发起的基于跨域通信协议的连接请求；

[0009] 所述客户端对所述连接请求进行合法性验证；

[0010] 如果所述连接请求通过所述合法性验证，则所述客户端建立与所述浏览器端的基于所述跨域通信协议的通信连接，以供所述浏览器端调用本地业务组件之用。

[0011] 另一方面，本申请实施例还提供了一种实现浏览器调用本地业务组件的装置，该

装置为一客户端，其包括：

[0012] 连接请求接收模块，用于接收浏览器端发起的基于跨域通信协议的连接请求；

[0013] 合法性验证模块，用于对所述连接请求进行合法性验证；

[0014] 连接控制模块，用于当所述连接请求通过所述合法性验证时，建立与所述浏览器端的基于所述跨域通信协议的通信连接，以供所述浏览器端调用本地业务组件之用。

[0015] 再一方面，本申请实施例还提供了一种实现浏览器调用本地业务组件的系统，包括：

[0016] 浏览器端，用于向客户端发起基于跨域通信协议的连接请求；

[0017] 客户端，用于对所述连接请求进行合法性验证，如果所述连接请求通过所述合法性验证，则所述客户端建立与所述浏览器端的基于所述跨域通信协议的通信连接，以供所述浏览器端调用本地业务组件之用。

[0018] 本申请实施例中，由于客户端支持跨域通信协议，而现有主流浏览器也大都支持跨域通信协议，因此客户端可与现有主流浏览器建立通信通道，从而在当前 NPAPI 濒临淘汰而其他浏览器插件又不具备通用性的情况下，提供了一种可使多数主流浏览器调用本地业务组件的通信方案，同时本申请实施例中，在与浏览器建立通信连接前，客户端对于浏览器的连接请求进行合法性验证，只有通过合法性验证的请求才能与其建立用于调用本地业务组件的通信连接，因此，本申请实施例同时还具备防止恶意网站利用客户端的功能，从而提高了浏览器调用本地业务组件的通信安全。

附图说明

[0019] 此处所说明的附图用来提供对本申请实施例的进一步理解，构成本申请的一部分，并不构成对本申请实施例的限定。在附图中：

[0020] 图 1 为本申请实施例的实现浏览器调用本地业务组件的方法的一个实施例的方法流程图；

[0021] 图 2 为本申请实施例的实现浏览器调用本地业务组件的装置的一个实施例的逻辑结构图；

[0022] 图 3 为本申请实施例的实现浏览器调用本地业务组件的系统的一个实施例的逻辑结构图。

具体实施方式

[0023] 为使本申请实施例的目的、技术方案和优点更加清楚明白，下面结合实施例和附图，对本申请实施例做进一步详细说明。在此，本申请实施例的示意性实施例及其说明用于解释本申请实施例，但并不作为对本申请实施例的限定。

[0024] 下面结合附图，对本申请实施例的具体实施方式作进一步的详细说明。

[0025] 参考图 1 所示，本申请实施例的实现浏览器调用本地业务组件的方法包括以下步骤：

[0026] 步骤 S101、客户端接收浏览器端发起的基于跨域通信协议的连接请求。当浏览器端需要调用本地业务组件时，其向客户端发起的基于 WebSocket 协议连接请求，客户端则对应接收该请求。其中，WebSocket 协议是一种全双工通讯的网络技术，在 WebSocket 协

议中，通信双方只需要做一个握手的动作，然后就可快速在通信双方之间形成了一条通信通道，通信双方之间就直接可以数据互相传送。而当前的主流浏览器（例如 Chrome16+、IE10+、Firefox11+、Safari6.0+、Opera12.10+、360 安全浏览器、360 极速浏览器、淘宝浏览器、搜狗浏览器、UC 浏览器、猎豹浏览器、傲游浏览器等）均支持 WebSocket 协议。

[0027] 需要明确的是，为了使多数主流浏览器可调用本地业务组件，本申请实施例的客户端与浏览器之间需要采用跨域通信协议，这里的跨域是指跨系统安全域。WebSocket 协议是本申请实施例的优选，但本申请实施例也可以采用其他跨域通信协议（例如 JSONP 协议、CORS 协议等）替代。但是，JSONP 协议对发送的数据包有大小限制，如果发送的数据长度超过限制，余下的数据会被截断。而采用 CORS 协议时，客户端返回给服务器的数据头(Response header) 中需加上一个特殊字段，但客户端无法主动给服务器发送消息，服务器必须不停向客户端查询是否有数据发送给自己才行。WebSocket 协议则不存在上述两种协议的局限，从而可快速为通信双方建立双向通信通道。而且，使用 WebSocket 协议这种通信方式，即使在客户端崩溃时，网站也是无感知的，因为客户端可以在后台重启并继续服务，从而可以提升网站体验。

[0028] 步骤 S102、客户端对连接请求进行合法性验证；如果通过合法性验证，则执行步骤 S103；否则，客户端断开与浏览器端的连接，以防止恶意网站利用客户端。本步骤中，合法性验证的目的在于防止恶意网站利用客户端，这样恶意网站就无法利用客户端获取本机的信息了，因而提高了通信安全性。本步骤中，合法性验证优选通过判断发起连接请求的浏览器端所对应的网站是否为网站白名单中的网站的验证方式。这是因为通常需要调用本地业务组件的网站相对较少，因此使用网站白名单验证可加快客户端的合法性验证的处理速度。当然，本步骤中，合法性验证也可采用其他方式，比如令牌 (token) 验证。令牌验证是用非对称加密算法加密的，只有合法的网站才能生成那个令牌，而令牌被修改后可以被客户端识别，从而拒绝连接。

[0029] 步骤 S103、当该连接请求通过合法性验证时，客户端建立与浏览器端的基于 WebSocket 协议的通信连接，以供浏览器端调用本地业务组件之用。建立通信连接后，浏览器端调用本地业务组件过程大致如下：

[0030] 1)、浏览器端向客户端发送命令；

[0031] 2)、客户端解析命令并分发给对应的本地业务组件；

[0032] 3)、本地业务组件处理该命令并返回处理结果至客户端；

[0033] 4)、客户端将返回处理结果转发给浏览器端。

[0034] 本申请实施例中，客户端在接收浏览器端发起的基于 WebSocket 协议的连接请求时，按照设定的端口绑定优先顺序从端口列表中选择一个端口进行绑定；且端口绑定优先顺序与浏览器端的端口选择优先顺序一致，比如，端口列表中有 A、B、C、D 四个可选端口，客户端的端口绑定优先顺序为 A ＞ B ＞ C ＞ D；同样，浏览器端的端口选择优先顺序也为 A ＞ B ＞ C ＞ D。这样，当最优选的端口 A 被其他软件占用时，客户端和浏览器端都会首先尝试选择 B 端口，这样就有利于提高建立连接的成功概率，从而有利于缩短建立连接的时间。

[0035] 此外，本申请实施例中，当客户端崩溃时，客户端将关于自身的崩溃报告发送给服务器，以供后续根据该崩溃报告对客户端进行相应的改进和升级。其中，崩溃报告可以包括错误代码、出错位置和当前状态等信息。而崩溃报告的获取，可由客户端通过操作系统的应

用程序编程接口(API,Application Programming Interface)调用操作系统自带的故障诊断程序来实现。

[0036] 本申请实施例中,客户端侧的网站白名单的更新可采用如下方式实现:

[0037] 客户端定期向服务器查询是否有网站白名单的更新,如果有,则请求获取该更新。

[0038] 本申请实施例中,由于客户端支持 WebSocket 协议,而现有主流浏览器也大都支持 WebSocket 协议,因此客户端可与现有主流浏览器建立通信通道,从而在当前 NPAPI 濒临淘汰而其他浏览器插件又不具备通用性的情况下,提供了一种可使多数主流浏览器调用本地业务组件的通信方式,且同时本申请实施例中,在与浏览器建立通信连接前,客户端对于浏览器的连接请求进行合法性验证,只有通过合法性验证的请求才能与其建立通信连接,因此,本申请实施例同时还具备防止恶意网站利用客户端的功能,从而提高了浏览器调用本地业务组件的通信安全。

[0039] 结合图 2 所示,本申请实施例的实现浏览器调用本地业务组件的装置为一客户端,与上述实现浏览器调用本地业务组件的方法对应,该客户端包括连接请求接收模块 21、合法性验证模块 22 和连接控制模块 23。其中:

[0040] 连接请求接收模块 21,用于接收浏览器端发起的基于跨域通信协议的连接请求。当浏览器端需要调用本地业务组件时,其向客户端发起的基于 WebSocket 协议连接请求,连接请求接收模块 21 则对应接收该请求。其中,WebSocket 协议是一种浏览器与服务器间进行全双工通讯的网络技术,在 WebSocket 协议中,通信双方只需要做一个握手的动作,然后就可快速在通信双方之间形成了一条通信通道,通信双方之间就直接可以数据互相传送。而当前的主流浏览器(例如 Chrome16+、IE10+、Firefox11+、Safari6.0+、Opera12.10+、360 安全浏览器、360 极速浏览器、淘宝浏览器、搜狗浏览器、UC 浏览器、猎豹浏览器、傲游浏览器等)均支持 WebSocket 协议。

[0041] 需要明确的是,为了使多数主流浏览器可调用本地业务组件,本申请实施例的客户端与浏览器之间需要采用跨域通信协议,这里的跨域是指跨系统安全域。WebSocket 协议是本申请实施例的优选,但本申请实施例也可以采用其他跨域通信协议(例如 JSONP 协议、CORS 协议等)替代。但是,JSONP 协议对发送的数据包有大小限制,如果发送的数据长度超过限制,余下的数据会被截断。而采用 CORS 协议时,客户端返回给服务器的数据头中需加上一个特殊字段,但客户端无法主动给服务器发送消息,服务器必须不停向客户端查询是否有数据发送给自己才行。WebSocket 协议则不存在上述两种协议的局限,从而可快速为通信双方建立双向通信通道。而且,使用 WebSocket 协议这种通信方式,即使在客户端崩溃时,网站也是无感知的,因为客户端可以在后台重启并继续服务,从而可以提升网站体验。

[0042] 合法性验证模块 22,用于对连接请求进行合法性验证。合法性验证的目的在于防止恶意网站利用客户端,从而提高了通信安全性。本合法性验证模块 22 中,合法性验证优选通过判断发起连接请求的浏览器端所对应的网站是否为网站白名单中的网站的验证方式。这是因为通常需要调用本地业务组件的网站不多,一般十几个,因此使用网站白名单验证可加快客户端的合法性验证的处理速度。当然,本步骤中,合法性验证也可采用其他方式,比如令牌验证。令牌验证是用非对称加密算法加密的,只有合法的网站才能生成那个令牌,而令牌被修改后可以被客户端识别,从而拒绝连接。

[0043] 连接控制模块 23,用于当连接请求通过合法性验证时,建立与浏览器端的基于

WebSocket 协议的通信连接，以供浏览器端调用本地业务组件之用。否则，断开与浏览器端的连接，以防止恶意网站利用客户端。建立通信连接后，浏览器端调用本地业务组件过程大致如下：

[0044] 1)、浏览器端向客户端发送命令；

[0045] 2)、客户端解析命令并分发给对应的本地业务组件；

[0046] 3)、本地业务组件处理该命令并返回处理结果至客户端；

[0047] 4)、客户端将该处理结果转发给浏览器端。

[0048] 本申请实施例中，客户端还可以包括：

[0049] 端口绑定模块 24，用于在连接请求接收模块 21 接收浏览器端发起的基于跨域通信协议的连接请求时，按照设定的端口绑定优先顺序从端口列表中选择一个端口进行绑定；且端口绑定优先顺序与浏览器端的端口选择优先顺序一致。比如，端口列表中有 A、B、C、D 四个可选端口，客户端的端口绑定优先顺序为 A ＞ B ＞ C ＞ D；同样，浏览器端的端口选择优先顺序也为 A ＞ B ＞ C ＞ D。这样，当最优选的端口 A 被其他软件占用时，端口绑定模块 24 和浏览器端都会首先尝试选择 B 端口，这样就有利于提高建立连接的成功概率，从而有利于缩短建立连接的时间。

[0050] 本申请实施例中，客户端还可以包括：

[0051] 崩溃报告处理模块 25，用于当判断客户端崩溃时，将关于自身的崩溃报告发送给服务器，以供后续根据该崩溃报告对客户端进行相应的改进和升级。其中，崩溃报告可以包括错误代码、出错位置和当前状态等信息。而崩溃报告的获取，可由崩溃报告处理模块 25 通过操作系统的应用程序编程接口调用操作系统自带的故障诊断程序来实现。

[0052] 此外，本申请实施例中，该客户端还可以包括：

[0053] 更新模块 26，用于通过定期向服务器查询的方式更新客户端侧的网站白名单。

[0054] 本申请实施例中，由于客户端支持 WebSocket 协议，而现有主流浏览器也大都支持 WebSocket 协议，因此客户端可与现有主流浏览器建立通信通道，从而在当前 NPAPI 濒临淘汰而其他浏览器插件又不具备通用性的情况下，提供了一种可使多数主流浏览器调用本地业务组件的通信方式，且同时本申请实施例中，在与浏览器建立通信连接前，客户端对于浏览器的连接请求进行合法性验证，只有通过合法性验证的请求才能与其建立通信连接，因此，本申请实施例同时还具备防止恶意网站利用客户端的功能，从而提高了浏览器调用本地业务组件的通信安全。

[0055] 结合图 3 所示，本申请实施例的实现浏览器调用本地业务组件的系统包括：浏览器端 31 和客户端 32，其中：

[0056] 浏览器端 31，用于向客户端 32 发起基于跨域通信协议的连接请求；

[0057] 客户端 32，用于对浏览器端 31 发送的连接请求进行合法性验证，如果连接请求通过合法性验证，则客户端 32 建立与浏览器端 31 的基于 WebSocket 协议的通信连接，以供浏览器端调用本地业务组件之用。其中，客户端 32 参见上述本申请实施例，在此不再赘述。

[0058] 本申请实施例中，该系统还包括：

[0059] 服务器 33，用于接收并保存客户端 31 发送的崩溃报告。

[0060] 本申请实施例中，由于客户端支持 WebSocket 协议，而现有主流浏览器也大都支持 WebSocket 协议，因此客户端可与现有主流浏览器建立通信通道，从而在当前 NPAPI 濒临

淘汰而其他浏览器插件又不具备通用性的情况下，实现了多数主流浏览器可调用本地业务组件。并且，本申请实施例中，在与浏览器建立通信连接前，客户端对于浏览器的连接请求进行合法性验证，只有通过合法性验证的请求才能与其建立通信连接，因此，本申请实施例同时还具备防止恶意网站利用客户端的功能，从而提高了通信安全性。

[0061] 本领域技术人员还可以了解到本申请实施例列出的各种说明性逻辑块、单元和步骤可以通过硬件、软件或两者的结合来实现。至于是通过硬件还是软件来实现取决于特定的应用和整个系统的设计要求。本领域技术人员可以对于每种特定的应用，可以使用各种方法实现所述的功能，但这种实现不应被理解为超出本申请实施例保护的范围。

[0062] 本申请实施例中所描述的各种说明性的逻辑块，或单元都可以通过通用处理器，数字信号处理器，专用集成电路 (ASIC)，现场可编程门阵列或其它可编程逻辑装置，离散门或晶体管逻辑，离散硬件部件，或上述任何组合的设计来实现或操作所描述的功能。通用处理器可以为微处理器，可选地，该通用处理器也可以为任何传统的处理器、控制器、微控制器或状态机。处理器也可以通过计算装置的组合来实现，例如数字信号处理器和微处理器，多个微处理器，一个或多个微处理器联合一个数字信号处理器核，或任何其它类似的配置来实现。

[0063] 本申请实施例中所描述的方法或算法的步骤可以直接嵌入硬件、处理器执行的软件模块、或者这两者的结合。软件模块可以存储于 RAM 存储器、闪存、ROM 存储器、EPROM 存储器、EEPROM 存储器、寄存器、硬盘、可移动磁盘、CD-ROM 或本领域中其它任意形式的存储媒介中。示例性地，存储媒介可以与处理器连接，以使得处理器可以从存储媒介中读取信息，并可以向存储媒介存写信息。可选地，存储媒介还可以集成到处理器中。处理器和存储媒介可以设置于 ASIC 中，ASIC 可以设置于用户终端中。可选地，处理器和存储媒介也可以设置于用户终端中的不同的部件中。

[0064] 在一个或多个示例性的设计中，本申请实施例所描述的上述功能可以在硬件、软件、固件或这三者的任意组合来实现。如果在软件中实现，这些功能可以存储与电脑可读的媒介上，或以一个或多个指令或代码形式传输于电脑可读的媒介上。电脑可读媒介包括电脑存储媒介和便于使得让电脑程序从一个地方转移到其它地方的通信媒介。存储媒介可以是任何通用或特殊电脑可以接入访问的可用媒体。例如，这样的电脑可读媒体可以包括但不限于 RAM、ROM、EEPROM、CD-ROM 或其它光盘存储、磁盘存储或其它磁性存储装置，或其它任何可以用于承载或存储以指令或数据结构和其它可被通用或特殊电脑、或通用或特殊处理器读取形式的程序代码的媒介。此外，任何连接都可以被适当地定义为电脑可读媒介，例如，如果软件是从一个网站站点、服务器或其它远程资源通过一个同轴电缆、光纤电缆、双绞线、数字用户线 (DSL) 或以例如红外、无线和微波等无线方式传输的也被包含在所定义的电脑可读媒介中。所述的碟片 (disk) 和磁盘 (disc) 包括压缩磁盘、镭射盘、光盘、DVD、软盘和蓝光光盘，磁盘通常以磁性复制数据，而碟片通常以激光进行光学复制数据。上述的组合也可以包含在电脑可读媒介中。

[0065] 以上所述的具体实施例，对本申请实施例的目的、技术方案和有益效果进行了进一步详细说明，所应理解的是，以上所述仅为本申请实施例的具体实施例而已，并不用于限定本申请实施例的保护范围，凡在本申请实施例的精神和原则之内，所做的任何修改、等同替换、改进等，均应包含在本申请实施例的保护范围之内。

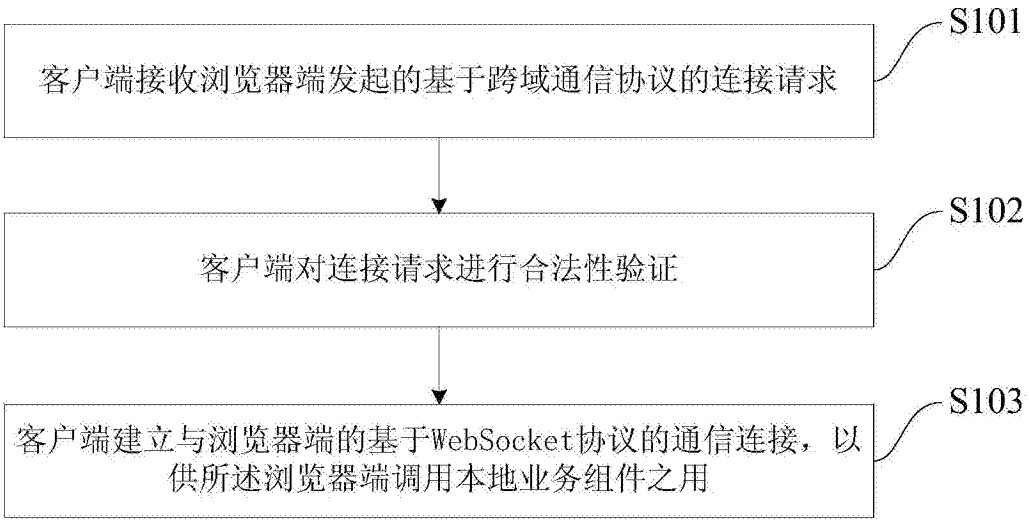

客户端接收浏览器端发起的基于跨域通信协议的连接请求
S101
客户端对连接请求进行合法性验证
S102
客户端建立与浏览器端的基于WebSocket协议的通信连接，以供所述浏览器端调用本地业务组件之用
S103

图 1

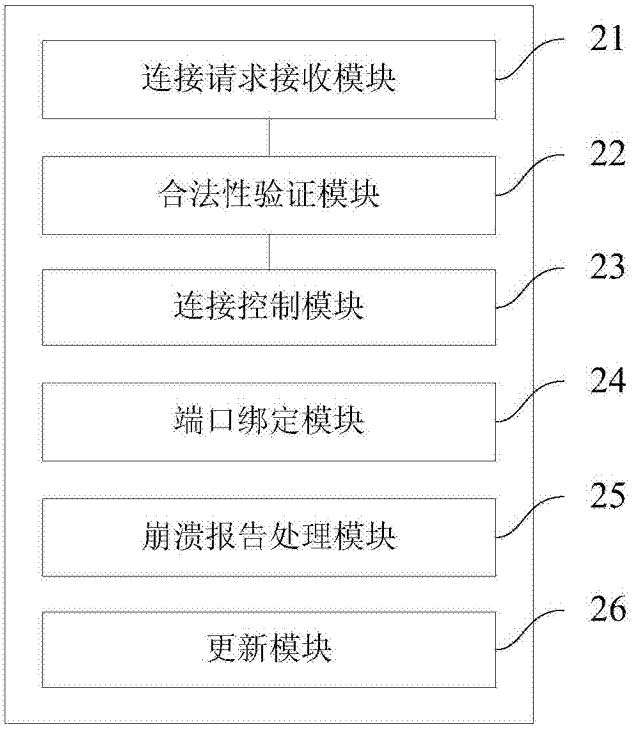

连接请求接收模块
21
合法性验证模块
22
连接控制模块
23
端口绑定模块
24
崩溃报告处理模块
25
更新模块
26

图 2

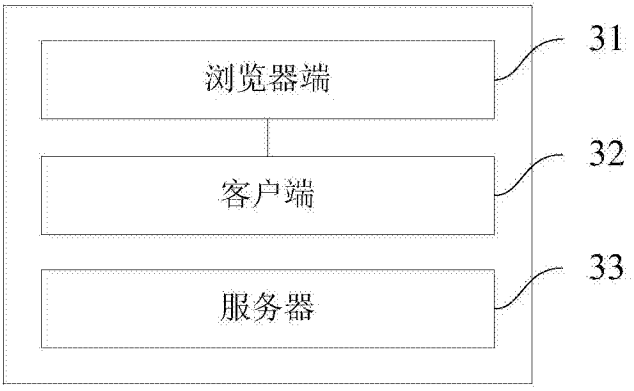


图 3