(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101616136 B

(45)授权公告日 2013.05.01

(21)申请号 200810126113.0

(22)申请日 2008.06.26

(73)专利权人 阿里巴巴集团控股有限公司

地址 英属开曼群岛大开曼岛

(72)发明人 岑文初 王霖 赵进 郑色树 曾义

(74)专利代理机构 北京同达信恒知识产权代理有限公司 11291

代理人 魏杉

(51)Int.Cl.

H04L 29/06(2006.01)

H04L 29/08(2006.01)

(56)对比文件

CN 101014958 A,2007.08.08, 全文.

CN 101079901 A,2007.11.28, 摘要、权利要求1、说明书第5页第7行到第7页第18行以及附图2-3.

US 2007/0061393 A1,2007.03.15, 全文.

审查员 肖丽华

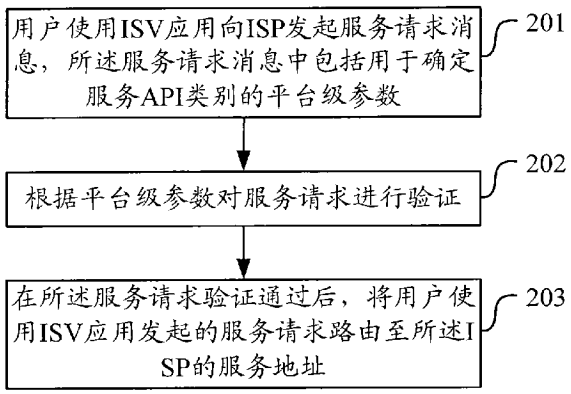
权利要求书3页 说明书13页 附图4页

(54)发明名称

一种提供互联网服务的方法及服务集成平台系统

(57)摘要

本发明公开了一种提供互联网服务的方法及服务集成平台系统,包括:用户使用独立软件供应商应用向互联网服务提供商发起服务请求消息,服务请求消息中包括用于确定服务应用编程接口类别的平台级参数;根据所述平台级参数对所述服务请求进行验证;在所述服务请求验证通过后,将用户使用独立软件供应商应用发起的服务请求路由至所述互联网服务提供商的服务地址。使用本发明可以实现维护一对多的安全认证结果,简化独立软件供应商在多互联网服务提供商模式下的安全认证流程;达到了简化独立软件供应商与互联网服务提供商的开发过程的效果。



CN 101616136 B

1. 一种提供互联网服务的方法，其特征在于，包括如下步骤：

用户使用独立软件供应商应用向互联网服务提供商发起服务请求消息，所述服务请求消息中包括用于确定服务应用编程接口类别的平台级参数，所述平台级参数包括：调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名、在独立软件供应商应用中用户会话唯一标识、返回值类型其中之一或者其组合；

根据所述平台级参数对所述服务请求进行验证；

在所述服务请求验证通过后，将用户使用独立软件供应商应用发起的服务请求路由至所述互联网服务提供商的服务地址；

确定所述平台级参数包括在独立软件供应商应用中用户会话唯一标识，在所述服务请求验证没有通过时，或者

确定所述平台级参数包括服务互联网服务提供商标识、调用独立软件供应商服务身份证明、在独立软件供应商应用中用户会话唯一标识时，获取缓存中的令牌列表，确定令牌不存在、或令牌无效时，

返回用户绑定出错信息以及所述互联网服务提供商登录地址；

所述独立软件供应商在收到用户绑定出错消息后，向所述互联网服务提供商登录地址发起服务请求消息，所述服务请求消息中的平台级参数包括调用独立软件供应商服务身份证明、服务名、在独立软件供应商应用中用户会话唯一标识、独立软件供应商应用通过用户绑定后需重定向的地址；

根据所述平台级参数中的服务名重定向到互联网服务提供商的登录地址，且将平台级参数传送至互联网服务提供商；

互联网服务提供商根据在独立软件供应商应用中用户会话唯一标识对用户登录进行校验；

互联网服务提供商在校验通过后，调用令牌颁发服务，传入调用独立软件供应商服务身份证明和在独立软件供应商应用中用户会话唯一标识，并重定向到独立软件供应商应用通过用户绑定后需重定向的地址；

互联网服务提供商通知独立软件供应商用户绑定成功。

2. 如权利要求 1 所述的方法，其特征在于，所述服务应用编程接口类型包括以下类别之一或者其组合：

不需要任何验证的匿名访问服务类型；

需要签名校验、时间戳校验的授权服务类型；

需要签名校验、时间戳校验和用户身份校验的用户授权服务类型；

需要签名校验、时间戳校验，用户身份校验可选的可选用户授权服务类型。

3. 如权利要求 1 所述的方法，其特征在于，所述验证包括签名校验、时间戳校验、用户身份校验其中之一或者其组合。

4. 如权利要求 1 所述的方法，其特征在于，所述根据所述平台级参数对所述服务请求进行验证，具体为：

所述平台级参数为调用独立软件供应商服务身份证明、服务名时，对所述服务请求不进行签名校验、用户身份校验；

所述平台级参数为调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名

时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

所述平台级参数为调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名、在独立软件供应商应用中用户会话唯一标识时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

所述平台级参数为调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名时，对所述服务请求进行签名校验、时间戳校验、用户身份校验可选。

5. 一种提供互联网服务的服务集成平台系统，其特征在于，包括：

接收模块，用于接收用户使用独立软件供应商应用向互联网服务提供商发起服务请求消息，所述服务请求消息中包括用于确定服务应用编程接口类别的平台级参数，所述平台级参数包括：调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名、在独立软件供应商应用中用户会话唯一标识、返回值类型其中之一或者其组合；

验证模块，用于根据所述平台级参数对所述服务请求进行验证；

所述验证模块进一步包括：第一绑定出错返回单元、和 / 或第二绑定出错返回单元，其中：

第一绑定出错返回单元，用于确定所述平台级参数包括在独立软件供应商应用中用户会话唯一标识，在所述服务请求验证没有通过时，返回用户绑定出错信息及所述互联网服务提供商登录地址；

第二绑定出错返回单元，用于确定所述平台级参数包括服务互联网服务提供商标识、调用独立软件供应商服务身份证明、在独立软件供应商应用中用户会话唯一标识时，获取缓存中的令牌列表；确定令牌不存在、或令牌无效时，返回用户绑定出错消息及所述互联网服务提供商登录地址；

路由模块，用于在所述服务请求验证通过后，将用户使用独立软件供应商应用发起的服务请求路由至所述互联网服务提供商的服务地址；

重定向模块，用于接收到独立软件供应商在收到用户绑定出错消息后向所述互联网服务提供商登录地址发起服务请求消息时，根据所述平台级参数中的服务名重定向到互联网服务提供商的登录地址，且将平台级参数传送至互联网服务提供商，所述服务请求消息中的平台级参数包括调用独立软件供应商服务身份证明、服务名、在独立软件供应商应用中用户会话唯一标识、独立软件供应商应用通过用户绑定后需重定向的地址；

令牌颁发服务模块，用于在互联网服务提供商根据在独立软件供应商应用中用户会话唯一标识对用户登录进行校验通过后，根据互联网服务提供商传入的调用独立软件供应商服务身份证明和在独立软件供应商应用中用户会话唯一标识颁发令牌。

6. 如权利要求 5 所述的服务集成平台系统，其特征在于，所述验证模块进一步用于进行包括签名校验、时间戳校验、用户身份校验其中之一或者其组合的验证。

7. 如权利要求 5 所述的服务集成平台系统，其特征在于，所述验证模块进一步用于在根据所述平台级参数对所述服务请求进行验证时，

当所述平台级参数为调用独立软件供应商服务身份证明、服务名时，对所述服务请求不进行签名校验、用户身份校验；

当所述平台级参数为调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

当所述平台级参数为调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名、在独立软件供应商应用中用户会话唯一标识时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

当所述平台级参数为调用独立软件供应商服务身份证明、服务名、服务请求时间戳、签名时，对所述服务请求进行签名校验、时间戳校验、用户身份校验可选。

一种提供互联网服务的方法及服务集成平台系统

技术领域

[0001] 本发明涉及网络运用，特别涉及一种提供互联网服务的方法及服务集成平台系统。

背景技术

[0002] 随着SOA(Service-Oriented Architecture，面向服务的架构）的不断成熟，REST(Representational State Transfer，表述性状态迁移）风格的深入人心，使得服务本身逐渐成为互联网新兴资源。传统的互联网软件企业也开始尝试新角色，作为服务提供商，ISP(Internet Service Provider，互联网服务提供商）开始开放自身的服务资源，将内部数据最大化的服务于社会化作用，为网站自身发展提供了新的开放模式。同时，Web2.0应用的丰富化，也造就了很多ISV(Independent Software Vendor，独立软件供应商)，利用网络有效的服务资源，针对客户的需求，设计出丰富多样的交互式应用，将不同服务提供商提供的服务组合在一起，产生聚合后的创新效应。

[0003] 这种以服务为中心的新型开发模式也是针对互联网应用的高速更新，创新第一的特点而产生的。在一些大网站成功的案例中，ISV开发者针对这些开放API(Application Program Interface，应用编程接口）的网站来构建特色应用，吸引用户。

[0004] 因Google在该领域最具有代表性，下面以Google的安全控制流程为例进行说明现有技术的安全控制处理方式。

[0005] 图1为Google的安全控制流程实施示意图，如图所示，对于Google的Open API(Open Application Program Interface，开放性应用编程接口）来说安全控制流程主要是处理通过Google API(Application Program Interface，应用编程接口）获取用户信息的ISV应用请求。具体的流程如下：

[0006] 步骤101、Web Application(Web应用）向Google Accounts Authentication(Google账户验证)发送AuthSub HTTP(Hyper Text Transport Protocol；超级文本传送协议)request(请求)；当应用需要访问涉及到用户信息的Google服务时，首先要产生AuthSub的请求到Google的验证代理服务。

[0007] 步骤102、Google Accounts Authentication向User(用户）发送Access Consentpage(确认接入页面)；验证代理服务将会把Google的确认接入页面呈现给应用使用者，即User，应用服务者会首先确认是否授权该应用通过Google API来获取用户信息，同时如果是第一次访问将会提示登录操作。

[0008] 步骤103、User返回Login data and accept/deny access(登录数据及接受/拒绝接入)；用户确定是否授权应用，并且是否登录，如果拒绝授权将会返回到Google的页面，而不是应用页面。

[0009] 步骤104、Google Accounts Authentication向Web Application返回Redirectwith token(携带令牌的重定向）指示；如果用户授权应用并且登录，那么验证服务将会重定向到应用页面，同时将会带上token作为用户身份标识。

[0010] 步骤105、Web Application向Google service Access(服务接入)发送Servicerequest with token(携带令牌的服务请求);应用带上用户身份标识请求服务,Google服务检查过Token以后将会接受请求。

[0011] 步骤106、Google service Access处理后向Web Application返回服务调用结果Response from service(服务响应)。

[0012] Google在Token的管理方面,在这里的Token有两方面的限制,一是使用范围,每一个Google服务都需要申请一个独立的Token;二是使用的有效期,Token分为one-time-use(一次性)和session(会议)的,前者只能使用一次,后者可以多次使用。

[0013] 由于Google是比较有代表性的,由Google的实施就可以看出现有技术中存在的不足有:

[0014] 当前已经存在的安全策略只是解决的单个ISP的情景模式而没有解决多个ISP的情景模式,如Google需要发送专用的AuthSub请求。也正是由于单ISP的情景模式,在安全控制方面策略固化,比较单一,无法满足服务集成平台的需求。

[0015] 对于一个ISP的每一个服务而言,都需要维护Token,而这会使ISV应用开发的复杂性增加,开发成本很高,同时还需要具体了解ISP一整套的安全策略,这显然也增加了ISV的负担。

发明内容

[0016] 本发明提供一种提供互联网服务的方法及服务集成平台系统,用以在ISV与ISP在开发及提供服务的的过程中,解决在业务以外的交互过程中出现的安全问题。

[0017] 本发明实施例提供了一种提供互联网服务的方法,包括如下步骤:

[0018] 用户使用ISV应用向ISP发起服务请求消息,所述服务请求消息中包括用于确定服务API类别的平台级参数;

[0019] 根据所述平台级参数对所述服务请求进行验证;

[0020] 在所述服务请求验证通过后,将用户使用ISV应用发起的服务请求路由至所述ISP的服务地址。

[0021] 本发明实施例还提供了一种提供互联网服务的服务集成平台系统,包括:

[0022] 接收模块,用于接收用户使用ISV应用向ISP发起服务请求消息,所述服务请求消息中包括用于确定服务API类别的平台级参数;

[0023] 验证模块,用于根据所述平台级参数对所述服务请求进行验证;

[0024] 路由模块,用于在所述服务请求验证通过后,将用户使用ISV应用发起的服务请求路由至所述ISP的服务地址。

[0025] 本发明有益效果如下:

[0026] 在本发明实施中,在用户使用ISV应用向ISP发起服务请求消息时,在服务请求消息中包括用于确定服务API类别的平台级参数;这些平台级参数是根据ISP的API类别进行确定的;在接收到ISV的服务请求以后,便可以根据平台级参数对服务请求进行验证,在服务请求验证通过后,将用户使用ISV应用发起的服务请求路由至ISP的服务地址即可。通过该方案,ISV仅需发送与业务内容无关的平台级参数便可以完成各种安全验证,使得ISV仅需关注与业务有关的内容;而对于ISP来说,其只需提供自身所需的安全级别,而无须自

身再去完成与该安全有关的管理，仅需关注 API 的开发便可获得相应的安全保障；除了从 ISV、ISP 看各种安全验证以外，本方案还可以实现维护一对多的安全认证结果，简化 ISV 在多 ISP 模式下的安全认证流程；达到了简化 ISV 与 ISP 的开发过程的效果。

附图说明

[0027] 图 1 为背景技术中所述 Google 的安全控制流程实施示意图；

[0028] 图 2 为本发明实施例中所述提供互联网服务的处理方法实施流程示意图；

[0029] 图 3 为本发明实施例中所述安全控制实施流程示意图；

[0030] 图 4 为本发明实施例中所述绑定用户实施流程示意图；

[0031] 图 5 为本发明实施例中所述提供互联网服务的服务集成平台系统结构示意图。

具体实施方式

[0032] 本发明实施中，设计了一套安全策略来降低 ISV 应用开发的复杂性，具体的，首先将 ISP 提供的 API 接口按安全级别进行分类；然后根据该分类设定相应的安全验证内容；最后根据平台级参数的内容来进行安全验证。这样，当 ISV 需要请求 ISP 服务时，其仅需携带相应的参数便可得到相应的安全服务，即对 ISV 来说，安全策略是透明的，使得 ISV 开发只需关注具体的业务服务；同时，对于 ISP 来说，ISP 仅需提供 API 及安全要求，便可以对 ISV 请求进行相应的安全验证，从而可以让 ISP 集中于业务服务接口开发，而无需关心业务服务的安全策略体系。下面结合附图对本发明的具体实施方式进行说明。

[0033] 图 2 为提供互联网服务的处理方法实施流程示意图，如图所示，包括如下步骤：

[0034] 步骤 201、用户使用 ISV 应用向 ISP 发起服务请求消息，所述服务请求消息中包括用于确定服务 API 类别的平台级参数；

[0035] 步骤 202、根据所述平台级参数对所述服务请求进行验证；

[0036] 步骤 203、在所述服务请求验证通过后，将用户使用 ISV 应用发起的服务请求路由至所述 ISP 的服务地址。

[0037] 下面对上述步骤的具体实施进行说明。在说明中结合本发明构建安全策略的思路进行说明，即对服务安全级别的划分、平台级参数的定义、安全的控制及它们之间的关系进行说明。

[0038] 一、服务安全级别划分

[0039] 安全策略控制流程的基础就是对于服务安全级别的划分，安全涉及到两个方面：服务访问控制，服务业务安全级别。

[0040] 服务访问控制是设定服务的使用是否需要 ISP 的审批，对于高级别的服务，ISP 可以设定此类服务需要审批才可以使用，平台对于此类服务的访问根据审批情况来确定是否路由服务。

[0041] 因此，根据安全性需求，可以将步骤 201 中的服务 API 类型可以包括以下类别之一或者其组合：

[0042] 匿名访问服务：不需要任何验证的匿名访问服务类型；

[0043] 授权服务：需要签名校验、时间戳校验的授权服务类型；

[0044] 用户授权服务：需要签名校验、时间戳校验和用户身份校验的用户授权服务类

型；

[0045] 可选用户授权服务：需要签名校验、时间戳校验，用户身份校验可选的可选用户授权服务类型。

[0046] 该服务 API 由 ISP 提供，具体内容可以如下表所示：

[0047]

服务级别	安全性	apitype	描述
匿名访问服务	极低	0	不需要任何验证（包括签名）
授权服务	低	1	需要签名校验（不需用户绑定）
用户授权服务	高	2	所有的服务调用，需要签名校验和用户绑定
可选用户授权服务	中	3	部份服务调用，需要签名校验（用户绑定可选）

[0048] 其中，可选用户授权服务指此类服务是否需要绑定用户信息可选，如果绑定用户信息，同样的服务接口将会返回用户相关的私有信息，如果没有绑定，将只返回公开信息。

[0049] 相应的，为了实现安全策略控制流程，在步骤 202 中的验证可以包括签名校验、时间戳校验、用户身份校验其中之一或者其组合。

[0050] 当结合上述 API 类型，可以提供相应的验证要求，各种类型服务及验证在安全策略中的对应关系可以如下表所示：

[0051]

服务级别	签名校验	时间戳校验	用户身份校验
匿名访问服务	×	×	×
授权服务	√	√	×
可选用户授权服务	√	√	○
用户授权服务	√	√	√

[0052] 表中：√代表需要校验，× 代表不需要校验，○代表可选。

[0053] 二、平台级参数定义

[0054] 由于在 ISV 开发过程中，对于某一个服务的调用需要传递两部分的数据：平台级参数以及业务参数，业务参数是各个 ISP 提供的服务接口参数，平台级参数则是用于定位 ISP 服务以及安全流程需要的参数配置。与本发明实施相关的是平台级参数，在实施中其可以包括：调用 ISV 服务身份证明、服务名、服务请求时间戳、签名、在 ISV 应用中用户会话唯一标识、返回值类型其中之一或者其组合。

[0055] 为便于实施及描述，用于验证、安全控制相关的平台级参数可以具体如下表：

[0056]

名称	类型	是否必选	描述
sip_appkey	Varchar(20)	必选	ISV 应用的 appkey
sip_apiname	Varchar(20)	必选	服务名
sip_timestamp	Varchar(19)	可选	服务请求时间戳 (yyyy-mm-dd hh:mi:ss)
sip_sign	Varchar(19)	可选	签名（包括了对前面三个必选参数和所有业务参数按签名规则做签名）
sip sessionid	Varchar(20)	可选	在 ISV 应用中用户会话唯一标识
sip_format	Varchar(4)	可选	规定返回值的类型。json/xml

[0057] 其中：在名称一栏的的符号将会用于以下实施例，并代表该参数；Varchar 表示字符变量长度；是否必选指该参数是否属于必选携带的参数。json/xml 为返回内容的格式。

[0058] 表中的 ISV 应用的 appkey 是指调用 ISV 服务的身份证明。

[0059] 则，结合上述的服务 API 类型，按安全级别分类便可以确定所需携带的平台级参数，即通过携带不同的平台级参数便可确定不同的安全级别服务，具体的可以如下表所示：

[0060]

名称	类型	是否必选	描述
匿名访问服务：不需要任何验证			
sip_appkey	Varchar（20）	必选	
sip_apiname	Varchar（20）	必选	
授权服务：需要签名校验（不需用户绑定）			
sip_appkey	Varchar（20）	必选	
sip_apiname	Varchar（20）	必选	
sip_timestamp	Varchar（19）	必选	
sip_sign	Varchar（19）	必选	
需要用户绑定			
sip_appkey	Varchar（20）	必选	
sip_apiname	Varchar（20）	必选	
sip_timestamp	Varchar（19）	必选	
sip_sign	Varchar（19）	必选	
sip_sessionid	Varchar（20）	必选	
可选用户绑定			
sip_appkey	Varchar（20）	必选	
sip_apiname	Varchar（20）	必选	
sip_timestamp	Varchar（19）	必选	
sip_sign	Varchar（19）	必选	

[0061] 结合以上表格可知，在实施中可以根据携带的平台级参数对服务请求进行相应的验证，从而达到安全的要求。具体的可以为：

[0062] 所述平台级参数为调用 ISV 服务身份证明、服务名时，对所述服务请求不进行签名校验、用户身份校验；

[0063] 所述平台级参数为调用 ISV 服务身份证明、服务名、服务请求时间戳、签名时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

[0064] 所述平台级参数为调用 ISV 服务身份证明、服务名、服务请求时间戳、签名、在 ISV

应用中用户会话唯一标识时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

[0065] 所述平台级参数为调用 ISV 服务身份证明、服务名、服务请求时间戳、签名时，对所述服务请求进行签名校验、时间戳校验、用户身份校验可选。

[0066] 三、安全控制

[0067] 实施中，可以专门设置一个服务集成平台系统用来进行实施安全策略控制的相关处理，实施中，称该实体为 SIP(Service Integration Platform，服务互联平台)。以下实施例中便以该实体为例进行说明。

[0068] 1、调用 ISV 服务身份证明 (Appkey) 的颁发

[0069] 具体实施中，可以在服务集成平台系统，即在SIP 中设置一个 ISV Portal(入口)，用于管理 ISV，负责颁发 AppKey 和 AppSecret。ISV 在 ISV Portal 中注册应用后，颁发给 ISV 一组 AppKey 和 AppSecret。AppKey 是 ISV 调用 ISP 服务的身份证明，不可更改；AppSecret 是应用访问 API 的密钥，ISV 可在 ISVPortal 中自助更换。两者关系类似于登录帐号和密码，两者都会在签名和业务参数中使用。

[0070] 实施中，针对 AppKey 进行说明，按上表用 sip_appkey 进行表示。AppSecret 则是以根据用户需要选择或更改，不进行另外说明。

[0071] 2、签名 (sign) 的实施：

[0072] 签名可以由服务集成平台系统完成验证，ISP 并不需要做签名认证。ISV 需要对提交的所有的参数签名 (sip_sign 参数除外)，签名的参数包括两部分：平台级参数和业务参数，通过算法可以起到数字签名的作用。

[0073] ISV 应用在向 SIP 调用 API 前，需要生成 sip_sign 参数。生成规则为：

[0074] sip_sign ＝ MD5(与 appsecret 连接成串 (

[0075] 连接数组生成字符串 (

[0076] 按字母顺序对数组排序 (

[0077] 未经 URL 编码的 <key><value> 形式的字符串数组 (所有参数，除 sip_sign 外)

[0078])

[0079])

[0080]))

[0081] 其中：

[0082] sip_sign ＝ MD5(sip_appsecret+[param 1+value1…paramn+valuen])，参数按字母顺序排列。如：

[0083] sip_appkey ＝ 001

[0084] sip_appsecret ＝ a312

[0085] 1)、接口方法定义：

[0086] getUserListByCondition(String company，boolean isdeleted，int count)

[0087] 2)、接口调用实例：

[0088] getUserListByCondition(“alisoft”，false，10)

[0089] 则，可以通过以下方式完成签名制作：

[0090] 1)、将接口方法参数值对根据参数名称的字母排列连接：

[0091] Company+alisoft+count+10+isdeleted+false

[0092] 2)、加入方法的 secretCode 获得最后的签名前的字符串：(此处假设 appSecret 为 a312)

[0093] sip_sign = MD5(a312companyalisoftcount 10isdeletedfalse)

[0094] 3、时间戳验证的实施

[0095] 基于签名算法，每一次请求都会将当前时间签入，服务集成平台系统收到签名以后就可以验证时间戳，判断请求是否超时，防止消息重放的产生。

[0096] 4、加密算法的实施

[0097] 这里的加密算法用在用户 Token 的生成过程中，由于 Token 只是 SIP 平台自身验证，因此采用效率较高的对称性加密，密钥定期更换。

[0098] sip_usertoken = DES(appkey = $sip_appkey, ispid = $ispid, userid = $username, timestamp = $timestamp, type = (0|1))

[0099] 上式中的参数说明如下：

[0100]

参数名称	值	说明
appkey	请求参数值 :sip_appkey	调用 ISV 服务的身份证明
ispid	请求参数值 :sip_apiname 只分析出的 isp 标识	如 sip_apiname = taobao.test.echo 则 :ispid = taobao
userid	请求参数值 :username	当用户在 isp 提供的登录页面成功登录之后，回调 sip 的 token 生成 url，传到 sip 的 isp 中的 username
timestamp	sip 中的系统当前时间格式为：yyyy-MM-dd HH:mm:ss	作为 Session 超时判断的依据
type		Token 的类型 type = 0 :session(一次登录多次有效) (在 session 及 timeout 内有效) type = 1 :request(一次登录仅一次有效)

[0101] 四、安全控制流程的实施

[0102] 图 3 为安全控制实施流程示意图，为更清楚地表述该流程的实施，图中不仅按时间进行了流程安排，还按实施的主体进行了安排，涉及的主体有用户、ISV 应用、SIP、ISP，其中 SIP 指服务集成平台系统，图中还涉及到了对 Token 的管理，实施中 Token 是一种身份令牌，当用户登录以后，就可以将用户在 ISV 的系统中的身份和 SIP 中的身份关联起来，每次请求中的 ISV 体系中的用户身份都可以对应到某一个已经产生的 SIP 的身份令牌，SIP 认为

有令牌的 ISV 应用使用者有权限去操作 ISP 的用户相关信息。则如图所示，具体可以为：

[0103] 步骤 301、用户使用 ISV 应用；

[0104] 步骤 302、利用 ISV 向 ISP 发起服务请求消息；

[0105] 本步骤中可以提供参数如：appkey、sign、timestamp、apiname、sessionID，还包含有其他相关的业务参数。

[0106] 用户使用 ISV 应用发起服务请求，ISV 应用可以根据服务级别判断是否签名，并将请求发送至 SIP。

[0107] 步骤 303、SIP 判断服务是否存在，是则转入步骤 304，否则转入步骤 316；

[0108] 判断服务是否存在时，可以通过消息中携带的参数 servicename 进行判断。

[0109] 步骤 304、判断服务访问控制类型，确定是否需要订购，是则转入步骤 305，否则转入步骤 306；

[0110] 实施中，如果 ISP 提供的服务不是开放给所有的 ISV 任意使用，那么此类服务使用之前就需要 ISV 向 ISP 发出申请使用请求，如果 ISP 同意请求，那么将会产生一条订购记录，此类服务在被调用前将会先检查订购关系才会继续交验其他安全策略。通过本步骤便可以对是否存在该类订购关系进行验证。

[0111] 步骤 305、判断是否有权访问服务，是则转入步骤 306，否则转入步骤 316；

[0112] 该判断可以通过所订购的 appkey 与其订购的 API 之间的关系进行判断。

[0113] 步骤 306、判断是否需要进行安全验证，是则转入步骤 307，否则转入步骤 312；

[0114] 步骤 307、判断签名校验是否通过，是则转入步骤 308，否则转入步骤 316；

[0115] 步骤 308、对是否超时进行校验，是则转入步骤 316，否则转入步骤 309；

[0116] 超时校验可以通过时间戳参数、Token 等来进行校验，超时的设置时长可以由系统来统一进行配置。

[0117] 步骤 309、判断服务是否需要绑定用户，可选或者必选绑定则转入步骤 310，否则转入步骤 312；

[0118] 本步骤中，由于有一些服务的调用是需要获取使用 ISV 应用的终端用户私有信息的，因此需要用户通过 ISP 网站进行登录操作，然后 ISP 会通知 SIP，由 SIP 将此用户信息和 SIP 颁发的 Token 绑定，而 ISV 应用后续服务请求只需要带有能够转换成为 SIP 的用户 Token 的 SessionId 即可，这类服务被调用时除了验证 ISV 身份以外还需要交验用户绑定的状况。

[0119] 步骤 310、判断 Token 是否有效，有效则转入步骤 312，无效或者不存在则转入步骤 311；

[0120] 本步骤中，可以根据 sessionID、服务所需的 Token 类型、以及 ISP 的 ID 来判断是否已经存在对应的 Token，而且 token 是否有效。

[0121] 进一步的，如果 Token 存在并有效，且该服务采用的是一次用户认证模式，那么还可以将该一次性 Token 移出，即该次运用已经验证用过。

[0122] 本步骤中，SIP 会维护 SessionId 和多个 ISP TokenList（令牌列表）的关系缓存，TokenList 虽然保存了 Token，但是每次做有效验证的时候不一定采取解密认证。每一个 ISP 都可以配置 Token 超时时间，从缓存中获取配置。

[0123] 步骤 311、判断服务是否需要绑定用户，是则转入步骤 316，否则转入步骤 312；

[0124] 与步骤 309 的判断绑定不同的是，步骤 309 是将可选用户绑定和必选用户绑定作为一条分支判断出来，而本步骤中是在判断用户是否绑定的情况下对于可选用户绑定和必选用户绑定作出进一步的不同安全处理。

[0125] 步骤 312、SIP 处理访问服务请求；

[0126] 本步骤中将请求服务路由至 ISP 进行服务处理。

[0127] 步骤 313、ISP 处理用户发起的服务请求；

[0128] 步骤 314、服务响应后处理；

[0129] 本步骤中的处理可以是记录日志等处理，并转发响应。

[0130] 步骤 315、获取处理结果，展示信息；

[0131] 步骤 316、错误处理；

[0132] 本步骤可以根据不同的错误信息作不同的响应，例如需要用户绑定就会回复对应的绑定用户地址以及相关信息，如果是其他的参数校验错误，则返回相关提示。

[0133] 在进行错误处理时，如果是需要绑定用户，那么可以根据返回的 login_url(login Uniform Resource Locator，登录统一资源定位符）来重定向到登录页面。

[0134] 步骤 317、返回页面。

[0135] 实施中，主要的部分有发起请求、根据请求的业务服务类别进行校验、对 Token 的校验等，下面再略作说明。

[0136] 用户使用 ISV 应用发起服务请求，ISV 应用根据服务级别判断是否签名，并将请求发送至 SIP。在接收到请求后，SIP 根据请求的服务名判断服务的有效性，再根据服务访问控制类型确定是否需要检查服务访问控制缓存。对所访问的服务安全级别进行判断后，若是匿名访问服务级别则直接路由到 ISP 的服务地址，而其他类型的服务级别则需根据其携带的参数作基本的安全认证。

[0137] SIP 校验签名信息和时间戳。并对根据服务级别判断是否需要用户认证，如果需要用户认证，则继续校验用户 Token。

[0138] 对于步骤 309 中涉及的用户绑定判断的后续处理中，由于有一些服务的调用需要将此用户信息和颁发的 Token 绑定，这类服务被调用时除了验证 ISV 身份以外还需要交验用户绑定的状况。因此，对于 Token 的校验再进行说明如下：

[0139] 检查参数中是否带有 ISV 标示用户唯一性的 SessionId，如果没有则认为未绑定用户身份；即当确定业务平台参数包括在 ISV 应用中用户会话唯一标识时，在服务请求验证没有通过时，可以返回用户绑定出错信息。

[0140] 确定业务平台参数包括服务 ISPID(ISP 标识）、调用 ISV 服务身份证明、在 ISV 应用中用户会话唯一标识时，获取缓存中的 Token 列表；当确定 Token 不存在、或 Token 无效时，可以返回用户绑定出错消息。具体实施中，可以简单的将服务 ISPId+ 应用的 Appkey+SessionId 的平台级参数组合成为用户 Token 缓存的 Key，然后获取缓存中的 Token 列表（同一个应用对同一个 ISP 的不同级别的服务会保存两类不同生命周期的 Token，one-time-use 和 session 的 Token)，判断 Token 是否存在，如果不存在表明未绑定用户身份。

[0141] 如果 Token 存在，判断 Token 的类型，如果是 Request Token（只能使用一次），则将从 Token 缓存中移出，并且返回用户身份校验成功，如果是 SessionToken，则需要再次校

验 Token 创建时间和当前时间，判断是否超期。

[0142] 如果用户认证未通过，那么返回给 ISV 应用需要用户绑定出错提示，并且返回对应的 ISP 的登录页面地址。

[0143] 如果用户认证通过，将用户名作为参数后传（ISP 可以根据此信息在作进一步的业务逻辑校验），再路由到 ISP 的服务地址。

[0144] 图 4 为绑定用户实施流程示意图，在步骤 316 的错误处理中，包含一种情况，即如果是需要绑定用户，则还需重定向到登录页面，如图所示，该绑定用户并进行重定向的实施流程可以如下：

[0145] 步骤 401、ISV 在访问服务时要求绑定用户；

[0146] 步骤 402、ISV 重定向到 SIP 回传的用户登录 url，需要将 appkey、apiname、redirect_url、sessionId 作为参数传递；

[0147] 用户登录的 url 即步骤 316 中所返回的地址。

[0148] 步骤 403、登录 SIP 的中转页面，作一定的处理后重定向到 ISP 的登录页面；

[0149] 步骤 404、用户进行服务访问确认，以及用户登录页面；

[0150] 本步骤中用户需要传入 AppKey、APIName、Session_id，其中 Session_id 是做为用户会话的唯一标识。

[0151] 步骤 405、ISP 校验用户身份，通过则转入步骤 407，否则转入步骤 406；

[0152] 步骤 406、返回出错信息；

[0153] 步骤 407、重定向到 redirect_url 参数地址，回调 SIP 登录成功通知 SIP；

[0154] 根据 redirect_url 参数中的地址，将登录后的界面转向到此地址，使得用户将会在看到登录页面后直接跳转到 ISV 的某一个地址上。

[0155] 步骤 408、SIP 判断 Session_id 是否存在，是则转入步骤 410，否则转入步骤 409；

[0156] 具体的实施中也可以将 IspId+appkey+session_id 作为系统内部真正的 Session_id 来进行校验，那么此时校验内容则不仅仅是 Session_id。

[0157] 步骤 409、SIP 将该 Session_id 存入 Session 池中对应的 ISP 部分，转入步骤 410；

[0158] 步骤 410、SIP 根据 API 的类型创建 Token 和附加验证信息并且和 Session_id 关联；并通知 ISV 绑定成功，在通知消息中包含 session_id 和 appkey 以及 apiname；

[0159] 步骤 411、ISV 根据自身情况选择是否处理。

[0160] 本步骤中，ISV 可以根据自身情况进行处理，例如在自己系统中可以标示某一个 sessionId 已经绑定了用户信息，在后续的服务请求中直接可以传递此 sessionId 请求服务，而无须再去请求绑定身份。

[0161] 从上述实施中可以看出，ISV 在收到需要用户绑定出错返回以后，重定向到 SIP 返回登录页面地址，同时需要带有 appkey，apiname（请求的服务名称），redirecturl（ISV 应用通过用户绑定后需要重定向的 url），sessionId（用户的唯一身份标示）。

[0162] 在 SIP 获取登录请求后，便可以根据 apiname 重定向到对应的 ISP 的登录地址，并且透传参数。

[0163] 用户在 ISP 的登录页面上可以选择是否登录，同时 ISP 的登录页面也可以根据类似于 cookie 等方式来确定用户是否登录过，并确定是否跳过登录步骤。该步骤主要是提供了一种免登录的方案，例如某用户已经在网站登录过，然后网站在客户端赋予了 cookie 作

为身份标识，如果在服务请求过程中再次转到登录页面，作为友好来说，登录页面检查一下是否存在这样的已登录的标识（cookie）来确定是否直接跳过登录的过程。

[0164] 在用户登录校验完毕后，ISP 便可以调用 SIP 平台系统的 Token 颁发服务，传入 Appkey 和 sessionId ；同时重定向到 ISV 应用传递过来的 redirect_url 地址。

[0165] SIP Token 颁发服务根据 Token 生成规则来产生 Token，并将 Token 存入缓存（以 ISPId+AppKey+SessionId）作为 Token 的 Key。

[0166] SIP 调用 ISV 应用的回调通知 URL，通知用户绑定成功，ISV 应用根据需求来确定是否处理响应。

[0167] 本发明还提供了互联网服务的服务集成平台系统，下面对本系统的具体实施方式进行说明。

[0168] 图 5 为提供互联网服务的服务集成平台系统结构示意图，为方便表述，将服务集成平台系统称为 SIP，则如图所示，图中包括了用户、ISV 应用服务器、SIP、ISP，其中 SIP 中可以包括：

[0169] 接收模块 501，用于接收用户使用 ISV 应用向 ISP 发起服务请求消息，所述服务请求消息中包括用于确定服务 API 类别的平台级参数；

[0170] 验证模块 502，用于根据所述平台级参数对所述服务请求进行验证；

[0171] 路由模块 503，用于在所述服务请求验证通过后，将用户使用 ISV 应用发起的服务请求路由至所述 ISP 的服务地址。

[0172] 具体的，验证模块可以用于进行包括签名校验、时间戳校验、用户身份校验其中之一或者其组合的验证。

[0173] 所述平台级参数中可以包括：调用 ISV 服务身份证明、服务名、服务请求时间戳、签名、在 ISV 应用中用户会话唯一标识、返回值类型其中之一或者其组合；

[0174] 则验证模块在根据所述平台级参数对所述服务请求进行验证时，可以按下面进行实施：

[0175] 当平台级参数为调用 ISV 服务身份证明、服务名时，对所述服务请求不进行签名校验、用户身份校验；

[0176] 当平台级参数为调用 ISV 服务身份证明、服务名、服务请求时间戳、签名时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

[0177] 当平台级参数为调用 ISV 服务身份证明、服务名、服务请求时间戳、签名、在 ISV 应用中用户会话唯一标识时，对所述服务请求进行签名校验、时间戳校验、用户身份校验；

[0178] 当平台级参数为调用 ISV 服务身份证明、服务名、服务请求时间戳、签名时，对所述服务请求进行签名校验、时间戳校验、用户身份校验可选。

[0179] 当存在需要对用户进行绑定的情况时，验证模块 502 还可以进一步包括：第一绑定出错返回单元 5021、和 / 或第二绑定出错返回单元 5022，其中：

[0180] 第一绑定出错返回单元 5021，用于确定业务平台参数包括在 ISV 应用中用户会话唯一标识，在服务请求验证没有通过时，返回用户绑定出错信息。

[0181] 第二绑定出错返回单元 5022，用于确定业务平台参数包括服务 ISPID、调用 ISV 服务身份证明、在 ISV 应用中用户会话唯一标识时，获取缓存中的 Token 列表；确定 Token 不存在、或 Token 无效时，返回用户绑定出错消息。

[0182] 进一步的，第一绑定出错返回单元、和 / 或第二绑定出错返回单元还可以在返回用户绑定出错信息时，返回所述 ISP 登录地址。

[0183] 当向用户返回绑定出错以及 ISP 登录地址后，用户可以进行再次的用户绑定，此时，服务集成平台系统中还可以进一步包括 ：

[0184] 重定向模块 504，用于接收到 ISV 在收到用户绑定出错消息后向所述 ISP 登录地址发起服务请求消息时，根据所述平台级参数中的服务名重定向到 ISP 的登录地址，且将平台级参数传送至 ISP，所述服务请求消息中的平台级参数包括调用 ISV 服务身份证明、服务名、在 ISV 应用中用户会话唯一标识、ISV 应用通过用户绑定后需重定向的地址 ；

[0185] 令牌颁发服务模块 505，用于在 ISP 根据在 ISV 应用中用户会话唯一标识对用户登录进行校验通过后，根据 ISP 传入的调用 ISV 服务身份证明和在 ISV 应用中用户会话唯一标识颁发令牌。

[0186] 由上述实施可以看出，在本发明实施中提供了一种可以实现将多服务提供商服务集成的安全控制策略，通过对签名、加密逻辑设计、结合服务级别的细分，通过平台级参数的整合，从而完成了整个安全流程设计，完善了服务分级体系，确保了用户信息安全以及 ISP 服务访问安全。克服了已有的单 ISP 服务安全机制的不足，对 ISV 而言，安全策略是透明的，从而能够降低 ISV 开发成本，使得 ISV 在开发中只需关注具体的业务服务 ；同时，也可以让 ISP 只需集中于业务服务接口开发，而无需关心业务服务的安全策略体系。

[0187] 显然，本领域的技术人员可以对本发明进行各种改动和变型而不脱离本发明的精神和范围。这样，倘若本发明的这些修改和变型属于本发明权利要求及其等同技术的范围之内，则本发明也意图包含这些改动和变型在内。

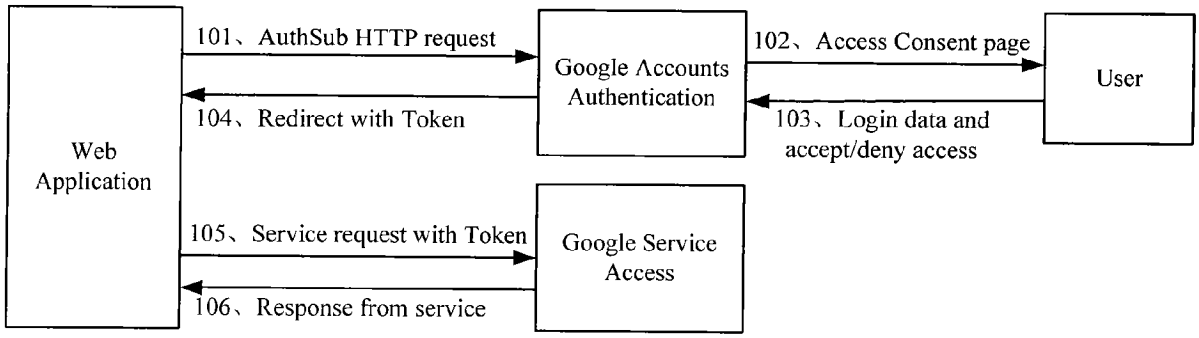

Web Application
101、AuthSub HTTP request
104、Redirect with Token
Google Accounts Authentication
102、Access Consent page
103、Login data and accept/deny access
User
105、Service request with Token
106、Response from service
Google Service Access

图 1

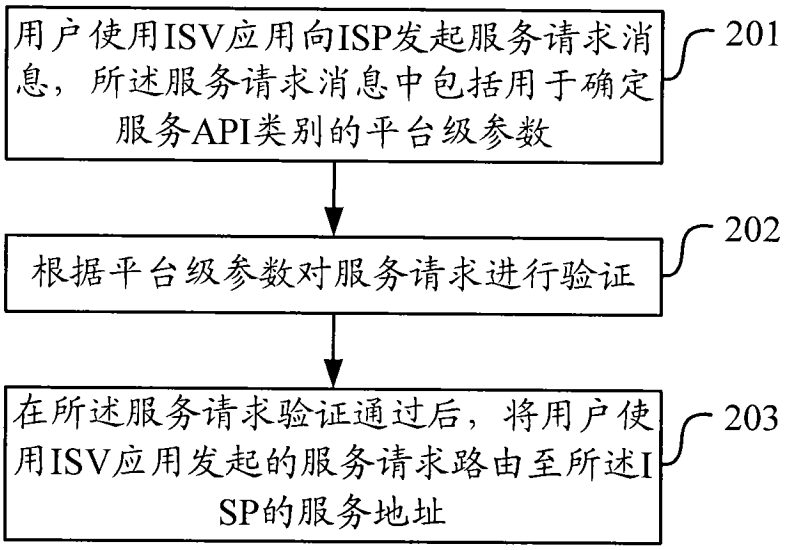

用户使用ISV应用向ISP发起服务请求消息，所述服务请求消息中包括用于确定服务API类别的平台级参数
201
根据平台级参数对服务请求进行验证
202
在所述服务请求验证通过后，将用户使用ISV应用发起的服务请求路由至所述ISP的服务地址
203

图 2

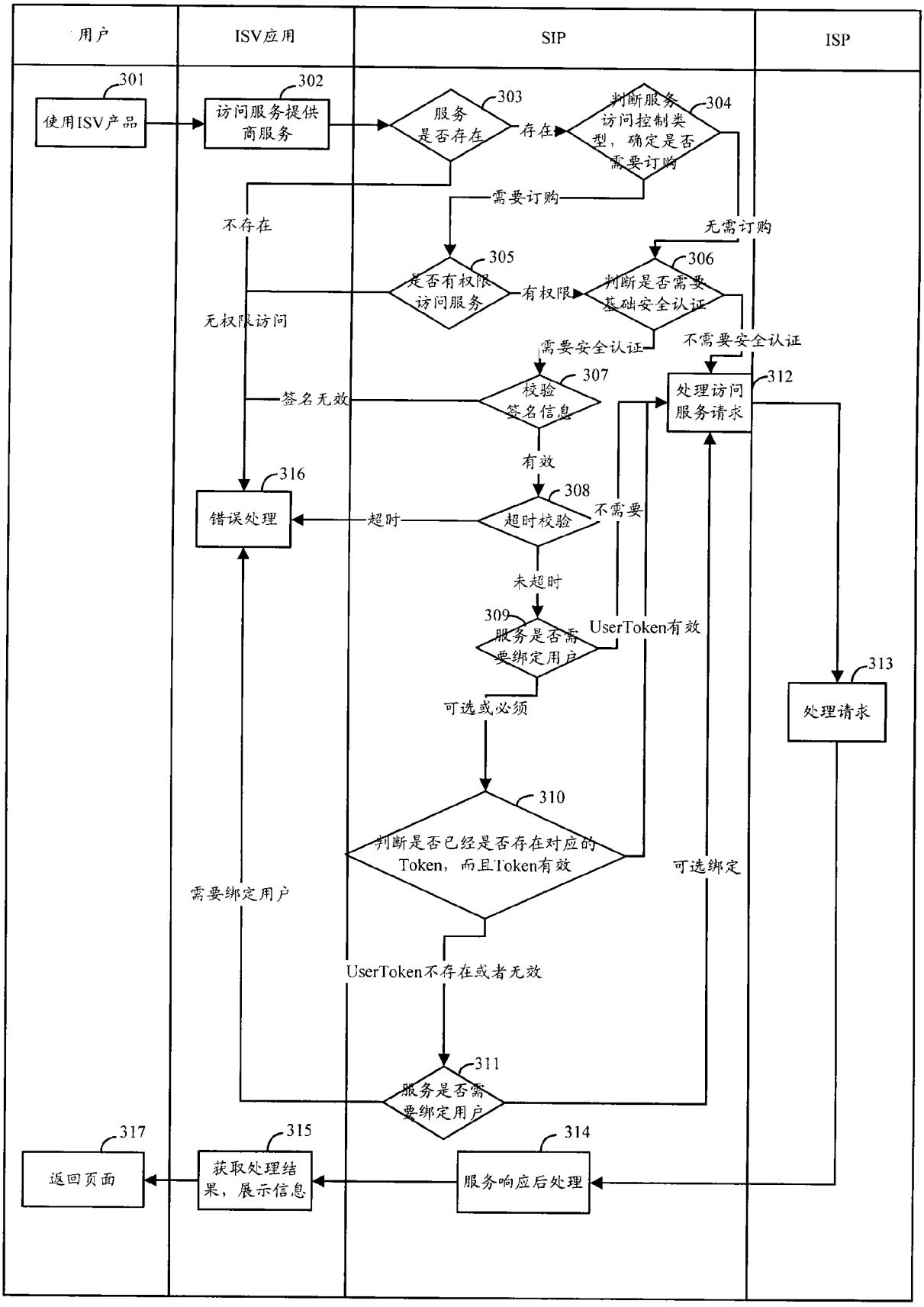

用户
ISV应用
SIP
ISP
301
使用ISV产品
302
访问服务提供商服务
303
服务是否存在
存在
304
判断服务访问控制类型，确定是否需要订购
需要订购
不存在
无需订购
305
是否有权限访问服务
有权限
306
判断是否需要基础安全认证
无权限访问
需要安全认证
不需要安全认证
307
校验签名信息
签名无效
312
处理访问服务请求
有效
316
错误处理
超时
308
超时校验
不需要
未超时
309
服务是否需要绑定用户
UserToken有效
313
处理请求
可选或必须
310
判断是否已经是否存在对应的Token，而且Token有效
可选绑定
需要绑定用户
UserToken不存在或者无效
311
服务是否需要绑定用户
317
返回页面
315
获取处理结果，展示信息
314
服务响应后处理

图 3

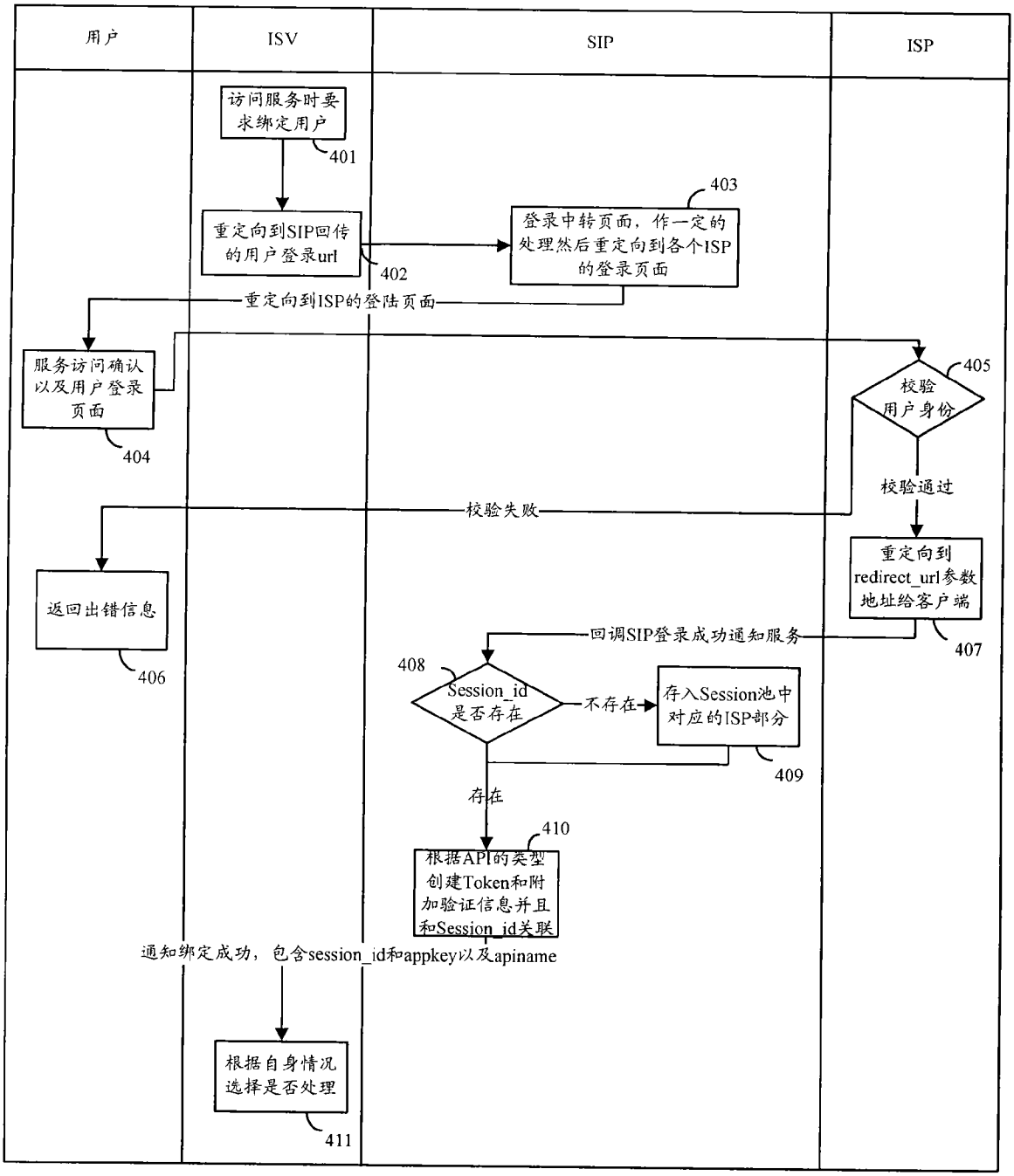

用户
ISV
SIP
ISP
访问服务时要求绑定用户
401
重定向到SIP回传的用户登录url
402
登录中转页面，作一定的处理然后重定向到各个ISP的登录页面
403
重定向到ISP的登陆页面
服务访问确认以及用户登录页面
404
校验用户身份
405
校验通过
校验失败
返回出错信息
406
重定向到redirect_url参数地址给客户端
407
回调SIP登录成功通知服务
408
Session_id是否存在
不存在
存入Session池中对应的ISP部分
409
存在
410
根据API的类型创建Token和附加验证信息并且和Session_id关联
通知绑定成功，包含session_id和appkey以及apiname
根据自身情况选择是否处理
411

图 4

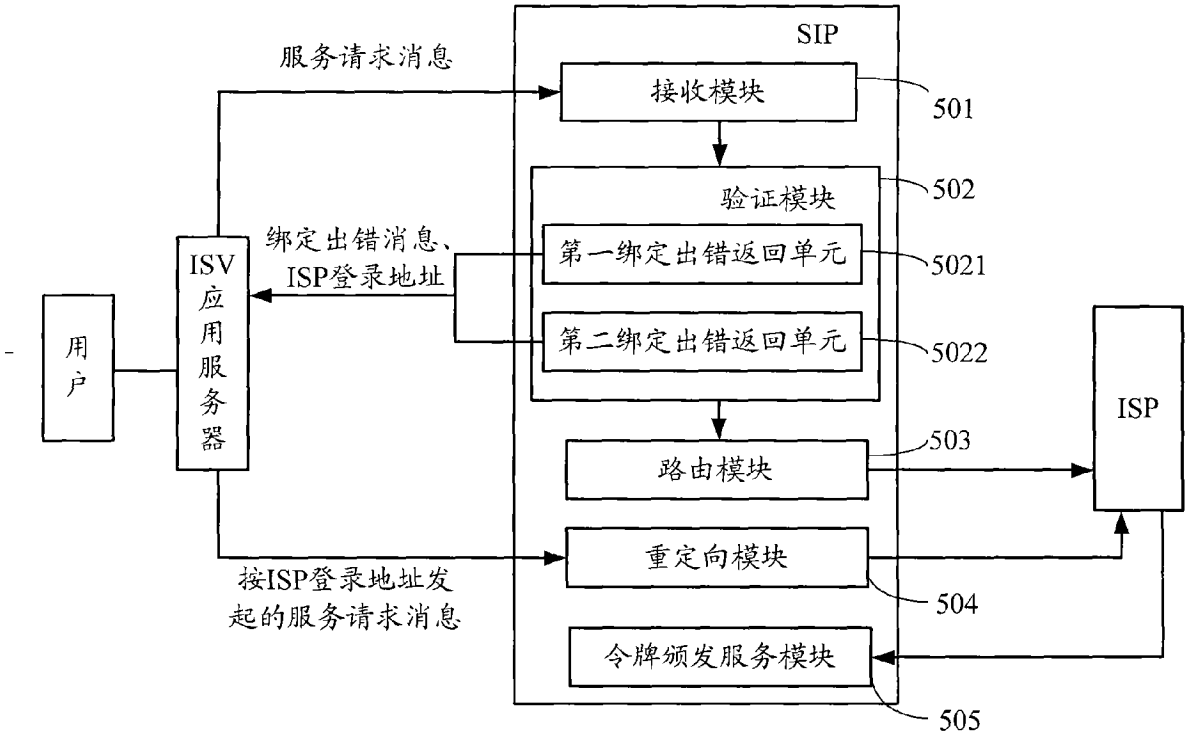

SIP
服务请求消息
接收模块
501
验证模块
502
绑定出错消息、
ISP登录地址
第一绑定出错返回单元
5021
第二绑定出错返回单元
5022
用户
ISV
应用服务器
ISP
路由模块
503
重定向模块
504
按ISP登录地址发
起的服务请求消息
令牌颁发服务模块
505

图 5