(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 102195956 A
(43)申请公布日 2011.09.21

(21)申请号 201010131235.6

(22)申请日 2010.03.19

(71)申请人 富士通株式会社
地址 日本神奈川县

(72)发明人 张军 苏亮 李邵明 孟遥 于浩

(74)专利代理机构 北京集佳知识产权代理有限公司 11227
代理人 杜诚 陈炜

(51)Int.Cl.
H04L 29/06(2006.01)

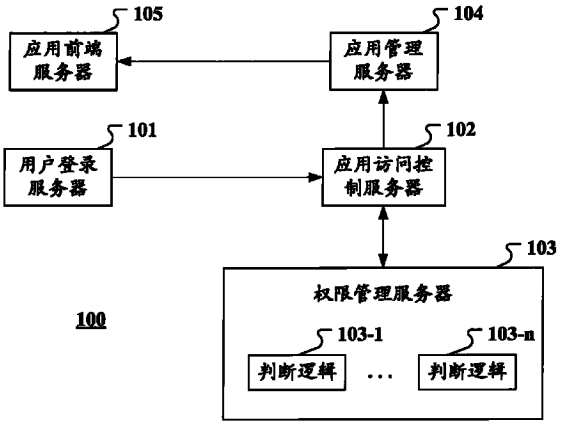
权利要求书 1 页 说明书 6 页 附图 4 页

(54)发明名称

云服务系统及其中管理用户权限的方法

(57)摘要

云服务系统及其中管理用户权限的方法。云服务系统包括：用户登录服务器，响应于用户登录确定用户的一或多个角色；权限管理服务器，包括一或多个判断逻辑，其每个对应于一个指定角色、应用、操作和判断逻辑的应用访问规则，对于每个应用访问规则，所指定的判断逻辑能够根据预定条件确定充当所指定的角色的用户是否被允许在所指定的应用中执行所指定的操作；应用访问控制服务器，响应于角色的确定而确定指定有角色的一或多个应用访问规则，并且请求权限管理服务器执行应用访问规则所指定的判断逻辑；应用管理服务器，响应于判断逻辑的确定结果生成有关允许用户使用的应用及操作的信息；及应用前端服务器，根据信息向用户呈现相应应用及操作。



CN 102195956 A

1. 一种云服务系统，包括：

用户登录服务器，其响应于用户登录确定用户的一或多个角色；

权限管理服务器，包括一或多个判断逻辑，每个所述判断逻辑对应于一个应用访问规则，所述应用访问规则中指定角色、应用、操作和所述判断逻辑，其中对于每个应用访问规则，所指定的判断逻辑能够根据预定条件确定充当所指定的角色的用户是否被允许在所指定的应用中执行所指定的操作；

应用访问控制服务器，其响应于所述角色的确定而确定指定有所述角色的一或多个应用访问规则，并且请求所述权限管理服务器执行所述应用访问规则所指定的判断逻辑；

应用管理服务器，其响应于所述判断逻辑的确定结果生成有关允许所述用户使用的应用及操作的信息；以及

应用前端服务器，其根据所述信息向用户呈现相应应用及操作。

2. 如权利要求 1 所述的云服务系统，其中基于 REST 架构来实现应用访问控制服务器对权限管理服务器的判断逻辑的执行的请求。

3. 如权利要求 1 或 2 所述的云服务系统，其中所述应用访问控制服务器包括应用访问规则的库。

4. 如权利要求 1 或 2 所述的云服务系统，其中所述应用前端服务器通过控制按钮的呈现来表示相应操作的允许。

5. 如权利要求 1 或 2 所述的云服务系统，其中所述应用前端服务器通过应用的呈现来表示查看操作的允许。

6. 一种云服务系统中管理用户权限的方法，包括：

响应于用户登录确定用户的一或多个角色；

响应于所述角色的确定而确定指定有所述角色的一或多个应用访问规则，并且请求执行每个所述应用访问规则所指定的判断逻辑，其中所述应用访问规则中指定有所述角色、应用、操作和所述判断逻辑，并且对于每个应用访问规则，所指定的判断逻辑能够根据预定条件确定充当所指定的角色的用户是否被允许在所指定的应用中执行所指定的操作；

响应于所述判断逻辑的确定结果生成有关允许所述用户使用的应用及操作的信息；以及

根据所述信息向用户呈现相应应用及操作。

7. 如权利要求 6 所述的方法，其中基于 REST 架构来实现对判断逻辑的执行的请求。

8. 如权利要求 6 或 7 所述的方法，其中从包括应用访问规则的库中确定指定有所述角色的一或多个应用访问规则。

9. 如权利要求 6 或 7 所述的方法，其中所述呈现包括通过控制按钮的呈现来表示相应操作的允许。

10. 如权利要求 6 或 7 所述的方法，其中所述呈现包括通过应用的呈现来表示查看操作的允许。

云服务系统及其中管理用户权限的方法

技术领域

[0001] 本发明涉及用户权限管理，尤其涉及云服务系统及其中管理用户权限的方法。

背景技术

[0002] 随着计算机以及网络技术的发展，云计算逐渐成为信息技术 (IT) 业界革命性的新技术。云计算改变的不仅仅是计算模式，同时也是商业模式上的改变。云计算技术通过网络以按需、易扩展的方式提供 IT 服务。这种服务的内容不仅仅包括软件即服务 (SaaS)，也包括平台即服务 (PaaS) 与更底层的基础设施即服务 (IaaS)。云计算通过服务的方式来提供计算与存储资源。提供这样的服务的系统也称为云服务系统。

[0003] 在云服务系统当中，针对不同的用户和不同的资源 (具体表现为云应用)，需要进行有效的权限控制，以保证各项资源能够限制在特定级别的用户那里。在云服务系统中，对云应用的访问控制决定了谁能够以何种方式访问某个应用，以及在对应用进行某项具体的操作时，决定了用户是否被授权进行该操作。

[0004] 已有的访问控制的方法主要有使用访问控制列表 (ACL) 和基于角色的访问控制 (RBAC) 两种。

发明内容

[0005] 在使用访问控制列表的方法中，针对需要限制权限的资源设置一个权限列表。这个权限列表称为 ACL。图 5a 示出了访问控制列表的例子。如图 5a 所示，访问控制列表 501 包含资源 ID、用户 ID 和访问方式 (即权限) 三个字段，以指定那个用户被授权访问哪个资源，以及可以以何种方式访问该资源。比如，如果一条记录定义了对应于文件 abc 的资源 ID、对应于用户 Tom 的用户 ID 和对应于“删除”的访问方式，则表示用户 Tom 可以删除文件 abc。当需要判断某个用户是否有权限在某项资源上执行某项操作时，需要先查询该资源的 ACL 的相关项，然后再决定该用户是否可以继续操作。虽然 ACL 的方法表述直观，易于理解，并且被广泛应用在文件系统和路由系统中，但是将其应用在云服务的权限控制时，却有下面的明显缺点 :

[0006] 1) 数据存储量过大，因为云服务系统中一般用户数量巨大，而且需要管理的数据也巨大，这会导致 ACL 过于庞大 ;

[0007] 2) 因为需要针对每个数据资源和用户的组合来维护 ACL，所以当云服务系统中用户和职能发生变化时，就需要对所有数据资源进行 ACL 的更新，而云服务系统中的数据资源通常数量庞大且复杂，这往往会导致权限管理系统维护困难 ;

[0008] 3) 缺少对同一类型资源的统一权限管理。

[0009] 基于角色的权限管理中定义了各种角色。图 5b 示出了定义用户角色的表的例子。如图 5b 所示，表 502 包含用户 ID 和角色两个字段。每种角色可以完成一定的职能，不同的用户根据其职能和责任被赋予了相应的角色。一旦某个用户成为某角色的成员，则此用户可以行使角色所具有的权利。角色和权限之间的关系，是可以预先定义的。图 5b 示出了定

义角色的权限的表的例子。如图 5b 所示，表 503 包含角色和访问方式两个字段。具体到权限的定义，一般是对同样类型的一组数据资源的操作权限，比如创建、删除、更新某一类型数据资源的操作权限。显然，基于角色的访问控制更便于实施整个组织或单位的网络信息系统的安全策略。但是因为基于角色的权限管理的粒度在于对同一类型的数据资源统一进行权限访问限制，从而缺少足够的灵活性。例如，缺少对某个具体的数据的编辑的权限。

[0010] 本发明的目的在于提供一种云服务系统及其中管理用户权限的方法，以至少部分地克服现有技术的上述缺陷。

[0011] 本发明的一个实施例是一种云服务系统，包括：用户登录服务器，其响应于用户登录确定用户的一或多个角色；权限管理服务器，包括一或多个判断逻辑，每个所述判断逻辑对应于一个应用访问规则，所述应用访问规则中指定角色、应用、操作和所述判断逻辑，其中对于每个应用访问规则，所指定的判断逻辑能够根据预定条件确定充当所指定的角色的用户是否被允许在所指定的应用中执行所指定的操作；应用访问控制服务器，其响应于所述角色的确定而确定指定有所述角色的一或多个应用访问规则，并且请求所述权限管理服务器执行所述应用访问规则所指定的判断逻辑；应用管理服务器，其响应于所述判断逻辑的确定结果生成有关允许所述用户使用的应用及操作的信息；以及应用前端服务器，其根据所述信息向用户呈现相应应用及操作。

[0012] 在一个进一步的实施例中，可以基于表形化状态转变 (REST) 架构来实现应用访问控制服务器对权限管理服务器的判断逻辑的执行的请求。

[0013] 在一个进一步的实施例中，应用访问控制服务器可以包括应用访问规则的库。

[0014] 在一个进一步的实施例中，应用前端服务器可以通过控制按钮的呈现来表示相应操作的允许。

[0015] 在一个进一步的实施例中，应用前端服务器可以通过应用的呈现来表示查看操作的允许。

[0016] 本发明的一个实施例是一种云服务系统中管理用户权限的方法，包括：响应于用户登录确定用户的一或多个角色；响应于所述角色的确定而确定指定有所述角色的一或多个应用访问规则，并且请求执行每个所述应用访问规则所指定的判断逻辑，其中所述应用访问规则中指定有所述角色、应用、操作和所述判断逻辑，并且对于每个应用访问规则，所指定的判断逻辑能够根据预定条件确定充当所指定的角色的用户是否被允许在所指定的应用中执行所指定的操作；响应于所述判断逻辑的确定结果生成有关允许所述用户使用的应用及操作的信息；以及根据所述信息向用户呈现相应应用及操作。

[0017] 在一个进一步的实施例中，可以基于 REST 架构来实现对判断逻辑的执行的请求。

[0018] 在一个进一步的实施例中，可以从包括应用访问规则的库中确定指定有所述角色的一或多个应用访问规则。

[0019] 在一个进一步的实施例中，呈现可以包括通过控制按钮的呈现来表示相应操作的允许。

[0020] 在一个进一步的实施例中，呈现可以包括通过应用的呈现来表示查看操作的允许。

[0021] 根据本发明的实施例，能够实现适用于云服务应用的基于灵活判断逻辑的权限控制。

附图说明

[0022] 参照下面结合附图对本发明实施例的说明，会更加容易地理解本发明的以上和其它目的、特点和优点。在附图中，相同的或对应的技术特征或部件将采用相同或对应的附图标记来表示。在附图中不必依照比例绘制出单元的尺寸和相对位置。

[0023] 图 1 是示出根据本发明实施例的云服务系统的结构的框图。

[0024] 图 2 是示出根据本发明实施例的云服务系统中的权限管理信息的示例的示意图。

[0025] 图 3a、3b 和 3c 分别示出判断逻辑的示例的伪码。

[0026] 图 4 是示出根据本发明实施例的云服务系统中管理用户权限的方法的流程图。

[0027] 图 5a 示出了访问控制列表的例子，图 5b 示出了定义用户角色的表和定义角色的权限的表的例子。

[0028] 图 6 是示出其中实现本发明的计算机的示例性结构的框图。

具体实施方式

[0029] 下面参照附图来说明本发明的实施例。应当注意，为了清楚的目的，附图和说明中省略了与本发明无关的、本领域普通技术人员已知的部件和处理的表示和描述。

[0030] 图 1 是示出根据本发明实施例的云服务系统 100 的结构的框图。

[0031] 如图 1 所示，云服务系统 100 包括用户登录服务器 101、权限管理服务器 103、应用访问控制服务器 102、应用管理服务器 104 和应用前端服务器 105。

[0032] 用户登录服务器 101 响应于用户登录确定用户的一或多个角色。

[0033] 图 2 是示出根据本发明实施例的云服务系统中的权限管理信息的示例的示意图。图 2 中的表 201 包含用户 ID 字段和相应的认证信息字段，用于在用户登录时验证其是否合法用户。在用户登录成功后，用户登录服务器 101 根据表 202 的内容确定用户的角色。表 202 包含用户 ID 字段和相应的角色字段，用于定义允许用户承担的角色。

[0034] 应用访问控制服务器 102 响应于角色的确定而确定指定有该角色的一或多个应用访问规则，并且请求权限管理服务器执行应用访问规则所指定的判断逻辑。

[0035] 每个判断逻辑对应于一个应用访问规则。每个应用访问规则中指定角色、应用、操作和判断逻辑。对于每个应用访问规则，所指定的判断逻辑能够根据预定条件确定充当所指定的角色的用户是否被允许在所指定的应用中执行所指定的操作。

[0036] 在一个例子中，应用是在云服务中具体提供的一组有逻辑关联的对数据资源进行操作的、拥有操作界面的 web 应用程序。比如日程管理应用，文章管理应用等等；操作是对一个或某一类型的数据资源的改变其属性和内容的行为，具体的是在应用的界面上通过点击和输入内容等方式来改变数据资源的内容或属性。

[0037] 图 2 中的表 203 提供了应用访问规则的例子。

[0038] 在图 2 所示的例子中，应用访问规则是四元组（角色，应用，操作，判断逻辑）。存在角色：新闻管理员、新闻发布员和新闻审阅员。存在用户：用户 A、用户 B、用户 C、用户 D、用户 E 和用户 F。存在应用：新闻管理应用。针对新闻管理的操作有：发布新闻、编辑新闻、浏览新闻和审阅新闻。存在的应用访问规则有：

[0039] 规则 1：(新闻发布员，新闻管理应用，发布新闻，always_true)

[0040] 规则 2 :(新闻发布员,新闻管理应用,编辑新闻,can_edit_own_article) ;

[0041] 规则 3 :(新闻管理员,新闻管理应用,发布新闻,always_true)

[0042] 规则 4 :(新闻管理员,新闻管理应用,编辑新闻,always_true)

[0043] 规则 5 :(新闻审阅员,新闻管理应用,浏览新闻,always_true)

[0044] 规则 6 :(新闻审阅员,新闻管理应用,审阅新闻,always_true)

[0045] 规则 7 :(新闻审阅员,新闻管理应用,编辑新闻,can_edit_older_article) 各个用户所赋予的角色如下 :

[0046] 用户 A :新闻发布员、新闻管理员、新闻审阅员

[0047] 用户 B :新闻发布员

[0048] 用户 C :新闻管理员

[0049] 用户 D :新闻审阅员

[0050] 用户 E :新闻审阅员

[0051] 用户 F :无

[0052] 当用户 A 登录时,用户登录服务器 101 根据表 201 确定登录成功,并根据表 202 确定用户 A 的三种角色 :新闻发布员、新闻管理员、新闻审阅员。根据所确定的用户 A 的三种角色,应用访问控制服务器 102 在表 203 中找到涉及这些角色的应用访问规则,即规则 1 至规则 7。应用访问控制服务器 102 相应请求权限管理服务器 103 执行规则 1 至 7 所指定的判断逻辑。

[0053] 当用户 B 登录时,用户登录服务器 101 根据表 201 确定登录成功,并根据表 202 确定用户 B 的角色 :新闻发布员。根据所确定的用户 B 的角色,应用访问控制服务器 102 在表 203 中找到涉及这些角色的应用访问规则,即规则 1 和规则 2。应用访问控制服务器 102 相应请求权限管理服务器 103 执行规则 1 和 2 所指定的判断逻辑。

[0054] 权限管理服务器 103 包括一或多个判断逻辑 103-1 至 103-n。

[0055] 在图 2 所示的例子中,判断逻辑 always_true 的返回结果永远为真 (即,有相应权限)。图 3a 示出判断逻辑 always_true 返回 XML 形式的永远为真的结果。

[0056] 判断逻辑 can_edit_own_article 在当前所登录的用户是由 id 标识的某篇文章的作者的情况下才允许该用户对该文章有相应权限。图 3b 以伪码的形式示出了逻辑判断 can_edit_own_article 的处理流程。

[0057] 判断逻辑 can_edit_older_article 在由 id 标识的某篇文章的创建时间大于某个时间的情况下才允许相应角色对该文章有相应权限。图 3c 以伪码的形式示出了逻辑判断 can_edit_older_article 的处理流程。

[0058] 在图 2 所示的例子中,根据规则 1 或 3 能够确定用户 A 有权执行发布新闻的操作,根据规则 4 能够确定用户 A 有权执行编辑新闻的操作,根据规则 5 能够确定用户 A 有权执行浏览新闻的操作,根据规则 6 能够确定用户 A 有权执行审阅新闻的操作。根据规则 1 能够确定用户 B 有权执行发布新闻的操作。但用户 B 无权编辑新闻。

[0059] 应用管理服务器 104 响应于判断逻辑的确定结果生成有关允许用户使用的应用及操作的信息。

[0060] 在图 2 所示的例子中,当用户 A 登录时,由于确定用户 A 有权执行发布新闻、编辑新闻、浏览新闻和审阅新闻的操作,应用管理服务器 104 生成的信息可以包含新闻管理应

用的显示以及发布新闻、编辑新闻、浏览新闻和审阅新闻的操作选项的显示。

[0061] 应用前端服务器 105 根据应用管理服务器 104 生成的信息向用户呈现相应应用及操作。在图 2 所示的例子中，当用户 A 登录时，向用户 A 呈现新闻管理应用以及发布新闻、编辑新闻、浏览新闻和审阅新闻的操作选项。

[0062] 应当明白，权限管理信息的内容和形式不限图 2 所示的例子。

[0063] 图 4 是示出根据本发明实施例的云服务系统中管理用户权限的方法的流程图。

[0064] 如图 4 所示，方法开始于步骤 401。

[0065] 在步骤 403，用户登录云服务系统。

[0066] 在步骤 405，响应于用户登录确定用户的一或多个角色。

[0067] 在步骤 407，响应于角色的确定而确定指定有该角色的一或多个应用访问规则，并且请求执行每个应用访问规则所指定的判断逻辑。应用访问规则中指定有角色、应用、操作和判断逻辑。对于每个应用访问规则，所指定的判断逻辑能够根据预定条件确定充当所指定的角色的用户是否被允许在所指定的应用中执行所指定的操作。

[0068] 在步骤 409，响应于判断逻辑的确定结果生成有关允许用户使用的应用及操作的信息。

[0069] 在步骤 411，根据步骤 409 生成的信息向用户呈现相应应用及操作。

[0070] 在步骤 413，方法结束。

[0071] 在上述云服务系统的一个具体实现中，可以基于 REST 架构来实现应用访问控制服务器对权限管理服务器的判断逻辑的执行的请求。

[0072] 在上述方法的一个具体实现中，可以基于 REST 架构来实现对判断逻辑的执行的请求。

[0073] 例如，可以在权限管理服务器上以 REST 方式定义具体的判断逻辑。每个判断逻辑可以是对应于一个 HTTP GET（或 POST）请求的 CGI。该 CGI 接受一个或多个参数，然后给出以 XML 格式描述的表示真（允许）或假（不允许）的结果。

[0074] 在上述云服务系统的一个具体实现中，应用访问控制服务器可以包括应用访问规则的库。

[0075] 在上述方法的一个具体实现中，可以从包括应用访问规则的库中确定指定有所述角色的一或多个应用访问规则。

[0076] 在上述云服务系统的一个具体实现中，应用前端服务器可以通过控制按钮的呈现来表示相应操作的允许。

[0077] 在上述方法的一个具体实现中，呈现可以包括通过控制按钮的呈现来表示相应操作的允许。

[0078] 在上述云服务系统的一个具体实现中，应用前端服务器可以通过应用的呈现来表示查看操作的允许。

[0079] 在上述方法的一个具体实现中，呈现可以包括通过应用的呈现来表示查看操作的允许。

[0080] 图 6 是示出其中实现本发明的计算机的示例性结构的框图。

[0081] 本发明的设备和方法实现环境如图 6 所示。

[0082] 在图 6 中，中央处理单元 (CPU)601 根据只读映射数据 (ROM)602 中存储的程序或

从存储部分 608 加载到随机存取映射数据 (RAM)603 的程序执行各种处理。在 RAM 603 中，也根据需要存储当 CPU 601 执行各种处理等等时所需的数据。

[0083] CPU 601、ROM 602 和 RAM 603 经由总线 604 彼此连接。输入 / 输出接口 605 也连接到总线 604。

[0084] 下述部件连接到输入 / 输出接口 605 :输入部分 606，包括键盘、鼠标等等 ;输出部分 607，包括显示器，比如阴极射线管 (CRT)、液晶显示器 (LCD) 等等，和扬声器等等 ;存储部分 608，包括硬盘等等 ;和通信部分 609，包括网络接口卡比如 LAN 卡、调制解调器等等。通信部分 609 经由网络比如因特网执行通信处理。

[0085] 根据需要，驱动器 610 也连接到输入 / 输出接口 605。可拆卸介质 611 比如磁盘、光盘、磁光盘、半导体映射数据等等根据需要被安装在驱动器 610 上，使得从中读出的计算机程序根据需要被安装到存储部分 608 中。

[0086] 在通过软件实现上述步骤和处理的情况下，从网络比如因特网或存储介质比如可拆卸介质 611 安装构成软件的程序。

[0087] 本领域的技术人员应当理解，这种存储介质不局限于图 6 所示的其中存储有程序、与方法相分离地分发以向用户提供程序的可拆卸介质 611。可拆卸介质 611 的例子包含磁盘、光盘 (包含光盘只读映射数据 (CD-ROM) 和数字通用盘 (DVD))、磁光盘 (包含迷你盘 (MD) 和半导体映射数据。或者，存储介质可以是 ROM 602、存储部分 608 中包含的硬盘等等，其中存有程序，并且与包含它们的方法一起被分发给用户。

[0088] 在前面的说明书中参照特定实施例描述了本发明。然而本领域的普通技术人员理解，在不偏离如权利要求书限定的本发明的范围的前提下可以进行各种修改和改变。

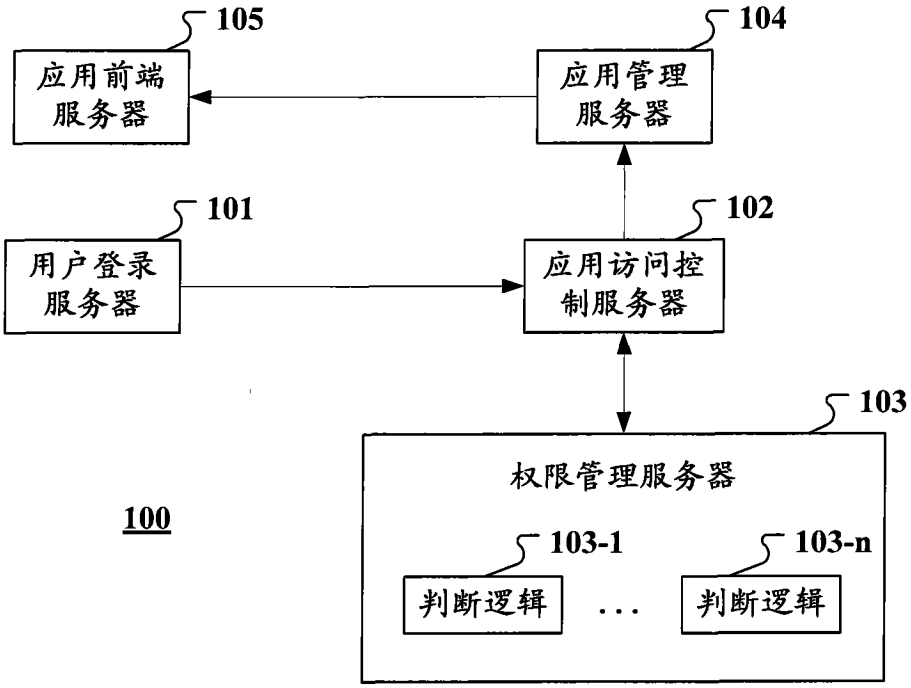


图 1

用户ID	认证信息
用户A	***
用户B	***
用户C	***
用户D	***
用户E	***
用户F	***

201

用户ID	角色
用户A	新闻发布员
用户A	新闻管理员
用户A	新闻审阅员
用户B	新闻发布员
用户C	新闻管理员
用户D	新闻审阅员
用户E	新闻审阅员

202

203

规则	角色	应用	操作	判断逻辑
1	新闻发布员	新闻管理应用	发布新闻	always_true
2	新闻发布员	新闻管理应用	编辑新闻	can_edit_own_article
3	新闻管理员	新闻管理应用	发布新闻	always_true
4	新闻管理员	新闻管理应用	编辑新闻	always_true
5	新闻审阅员	新闻管理应用	浏览新闻	always_true
6	新闻审阅员	新闻管理应用	审阅新闻	always_true
7	新闻审阅员	新闻管理应用	编辑新闻	can_edit_own_article

图 2

```
def always_true()
<xml>
<result>1</result>   //1表示真
</xml>
```

图 3a

```
def can_edit_own_article(id)
    article=Article.find(id) //找到文章
    if article.author == User.current //用户是作者？
      render true //是，则返回真
    else
      render false //否，则返回假
    end
end
```

图 3b

```
def can_edit_older_article(id)
    article=Article.find(id) //找到文章
    if article.created_at > a_time_stamp   //文章创建在时间a_time_stamp之前？
      render true //是，则返回真
    else
      render false //否，则返回假
    end
end
```

图 3c

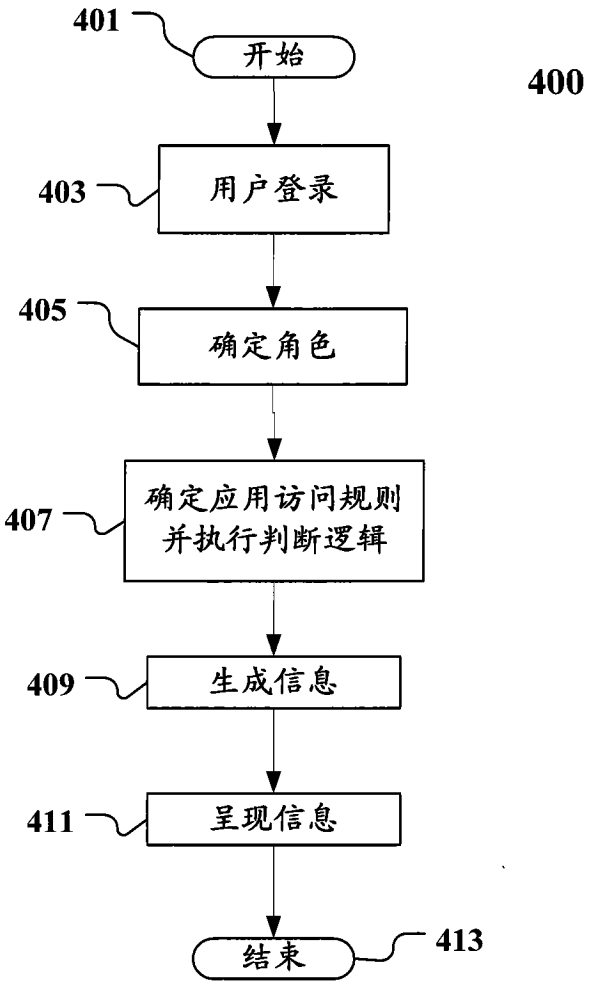


图 4

501

资源ID	用户ID	访问方式
资源1	用户A	读
资源2	用户A	写
...	...	...
...	...	...
...	...	...
...	...	...
...	...	...

图 5a

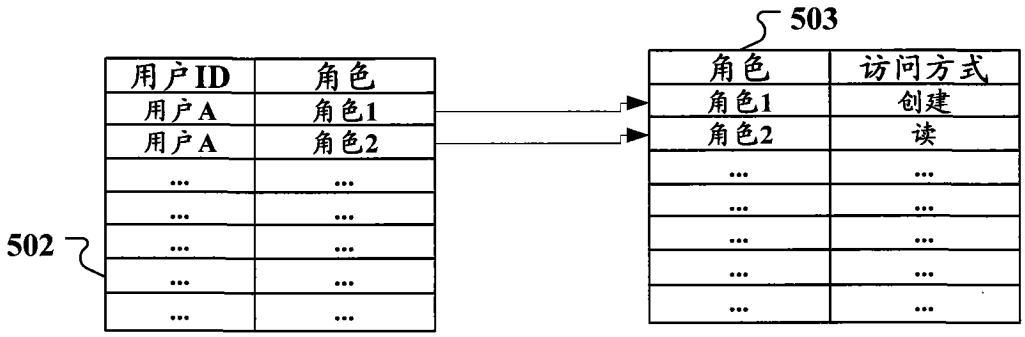

503
502
用户ID
角色
用户A
角色1
用户A
角色2
角色
访问方式
角色1
创建
角色2
读

图 5b

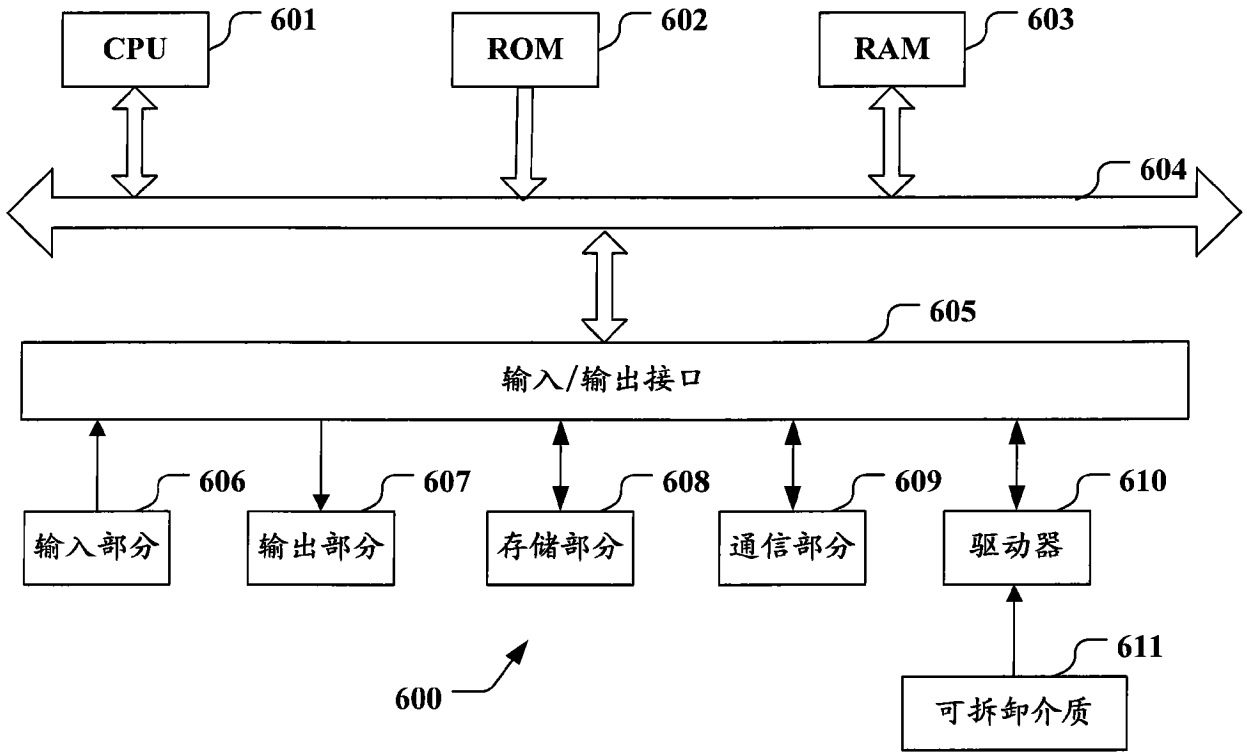

CPU
601
ROM
602
RAM
603
604
605
输入/输出接口
606
607
608
609
610
输入部分
输出部分
存储部分
通信部分
驱动器
611
可拆卸介质
600

图 6