阿里云
访问控制

用户指南

文档版本：20190819

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **禁止:** 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **警告:** 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 **说明:** 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定。
`courier` 字体	命令。	执行 `cd /d C:/windows` 命令，进入Windows系统文件夹。
*## *	表示参数、变量。	`bae log list --instanceid` *`Instance_ID`*
[]或者[a|b]	表示可选项，至多选择一个。	`ipconfig` *`[-all|-t]`*
{}或者{a|b}	表示必选项，至多选择一个。	`swich` *`{stand | slave}`*

目录

1 用户

1.1 RAM 用户概述

RAM 用户是 RAM 的一种实体身份类型，有确定的身份 ID 和身份凭证，它通常与某个确定的人或应用程序一一对应。

RAM 用户基本概念

- 一个云账号下可以创建多个 RAM 用户，对应企业内的员工、系统或应用程序。
- RAM 用户不拥有资源，没有独立的计量计费，这些用户由所属云账号统一控制和付费。
- RAM 用户归属于云账号，只能在所属云账号的空间下可见，而不是独立的云账号。
- RAM 用户必须在获得云账号的授权后才能登录控制台或使用 API 操作云账号下的资源。

RAM 用户的应用场景

当企业有多种云资源时，使用 RAM 的授权管理功能，可以实现用户分权及资源统一管理。详情请参考：#unique_5。

1.2 创建 RAM 用户

RAM 用户是 RAM 中的一种身份。 RAM 用户对应某一个操作实体，如运维操作人员或应用程序。通过创建新的 RAM 用户并授权，RAM 用户便可以访问相关资源。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 单击新建用户，输入登录名称和显示名称。

> 说明:
> 单击添加用户，可一次性创建多个 RAM 用户。

4. 在访问方式区域下，选择控制台密码登录或编程访问。
   - 控制台密码登录：可以完成对登录安全的基本设置，包括自动生成或自定义登录密码、是否要求下次登录时重置密码以及是否要求开启多因素认证。
   - 编程访问：将会自动为 RAM 用户创建访问密钥（AccessKey）。RAM 用户可以通过 API 或其他开发工具访问阿里云。

   
   说明:
   为了保障账号安全，建议仅为 RAM 用户选择一种登录方式。避免 RAM 用户离开组织后仍可以通过访问密钥访问阿里云资源。
5. 单击确认。

后续步骤

- 可以选择为用户添加到一个或多个组，对 RAM 用户进行分类并授权。详情请参考：#unique_7。
- 可以为用户添加一个或多个权限策略，使 RAM 用户具有资源的访问能力。详情请参考：为 RAM 用户授权。

1.3 查看 RAM 用户基本信息

本文为您介绍如何查看 RAM 用户基本信息，包括用户名、显示名称和 UID 等信息。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在用户基本信息区域，可以查看用户基本信息。

1.4 修改 RAM 用户基本信息

本文为您介绍如何修改 RAM 用户基本信息，包括用户名和显示名称等信息。

操作步骤

1. 登录 RAM 控制台。
2. 单击人员管理 > 用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在用户基本信息区域，单击编辑基本信息。

5. 修改完成后，单击确认。

1.5 为 RAM 用户授权

为 RAM 用户授权后，用户可以访问相应的阿里云资源。本文为您介绍如何为 RAM 用户授权。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击授权。
3. 单击新增授权。
4. 在被授权主体区域下，输入 RAM 用户名称或 ID 后，单击需要授权的 RAM 用户。

> 说明:
> 可以输入用户的 ID 或名称进行模糊搜索。

5. 在左侧权限策略名称列表下，单击需要授予 RAM 用户的权限策略。

> 说明:
> 在右侧区域框，选择某条策略并单击 ×，可撤销该策略。

6. 单击确定。

1.6 为RAM用户移除权限

当RAM用户不再需要某些权限或离开组织时，可以将这些权限移除。本文为您介绍移除RAM用户的权限的几种方式。

方式一

您可以在授权页面下为RAM用户移除权限。

1. 云账号登录RAM控制台。
2. 在左侧导航栏的权限管理菜单下，单击授权。
3. 找到目标RAM用户，单击移除授权。
4. 单击确认。

方式二

您可以在用户页面下的权限策略页签为RAM用户移除权限。

1. 云账号登录RAM控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。

3. 在用户登录名称/显示名称列表下，单击目标RAM用户名称。
4. 在权限管理页签下，找到目标权限策略，单击移除权限。
5. 单击确认。

1.7 RAM用户登录控制台

本文为您介绍RAM用户如何登录RAM控制台，包括登录地址和登录方式。

操作步骤

1. RAM用户登录RAM控制台。

   
   说明:
   云账号登录RAM控制台，在概览页可以快速查询登录RAM用户登录地址。

2. 输入RAM用户登录名称，单击下一步。

   - 方式一：使用默认域名登录。RAM用户登录格式为`<$username>@<$AccountAlias>.onaliyun.com`，例如：username@company-alias.onaliyun.com。

     
     说明:
     RAM用户登录账号为UPN（User Principal Name）格式，即RAM控制台用户列表中所见的用户登录名称。<$username> 为RAM用户名称，<$AccountAlias>.onaliyun.com 为默认域名。

   - 方式二：使用账号别名登录。RAM用户登录格式为`<$username>@<$AccountAlias>`，例如：username@company-alias。

     
     说明:
     <$username> 为RAM用户名称，<$AccountAlias> 为账号别名。

   - 方式三：如果创建了域别名，也可以使用域别名登录。RAM用户登录格式为`<$username>@<$DomainAlias>`，例如：username@example.com。

     
     说明:
     <$username>为RAM用户名称，<$DomainAlias>为域别名。

3. 输入RAM用户登录密码，单击登录。

1.8 删除 RAM 用户

当不再需要某个 RAM 用户或 RAM 用户离开组织时，可以删除该 RAM 用户。删除 RAM 用户会删除对应的访问密钥，解绑多因素认证设备并撤销 RAM 用户拥有的权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，找到目标 RAM 用户，单击删除。
4. 单击确认。

> **说明:**
> 删除 RAM 用户需要谨慎操作。如果有业务系统正在以此用户身份运行，那么可能会导致客户的业务故障。

2 用户组

2.1 用户组概述

访问控制（RAM）通过用户组对职责相同的 RAM 用户进行分类并授权，可以更加高效地管理 RAM 用户及其权限。

- 在 RAM 用户职责发生变化时，只需将其移动到相应职责的用户组下，不会对其他 RAM 用户产生影响。

  关于如何创建用户组，请参考：#unique_16。
- 当用户组的权限发生变化时，只需修改用户组的权限策略，即可应用到所有 RAM 用户。

  关于如何为用户组授权，请参考：#unique_17。

2.2 创建用户组

若云账号下有多个 RAM 用户，通过创建用户组对职责相同的 RAM 用户进行分类并授权，从而更好的管理用户及其权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户组。
3. 单击新建用户组，输入登录名称、显示名称和备注。
4. 单击确认。

后续步骤

可以为用户组添加一个或多个权限策略，详情请参考：#unique_19。

2.3 添加用户组成员

您可以为用户组添加一个或多个 RAM 用户。当 RAM 用户加入到用户组后，将拥有该用户组的所有权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户组。
3. 在用户组名称/显示名称列表下，找到目标用户组。

4. 单击添加组成员，用户组名称会自动填入。
5. 在左侧名称列表下，勾选需要添加到当前用户组的 RAM 用户名称。

> 说明:
> 在右侧区域框，选择某个 RAM 用户名称并单击 ×，可撤销该操作。

6. 单击确定。

2.4 移出用户组成员

当某个 RAM 用户离开组织或权限发生变化时，您需要将该 RAM 用户从用户组中移出。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户组。
3. 在用户组名称/显示名称列表下，单击目标用户组名称。
4. 在组成员管理页签下，找到目标 RAM 用户，单击移出用户组。
5. 单击确认。

2.5 查看用户组基本信息

本文为您介绍如何查看用户组基本信息，包括用户组名称、显示名称和备注。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户组。
3. 在用户组名称/显示名称列表下，单击目标用户组名称。
4. 在组基本信息区域，可以查看用户组基本信息。

2.6 修改用户组基本信息

本文为您介绍如何修改用户组基本信息，包括用户组名称、显示名称和备注。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户组。
3. 在用户组名称/显示名称列表下，单击目标用户组名称。
4. 在组基本信息区域，单击编辑基本信息。

5. 修改完成后，单击确认。

2.7 为用户组授权

本文为您介绍如何为用户组授权。为用户组授权后，用户组中的所有 RAM 用户将拥有该用户组的所有权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击授权。
3. 单击新增授权。
4. 在被授权主体区域下，输入用户组名称后，单击需要授权的用户组。
5. 在左侧权限策略名称列表下，单击需要授予用户组的权限策略。

   说明:
   在右侧区域框，选择某条策略并单击 ×，可撤销该策略。
6. 单击确定。

2.8 为用户组移除权限

当用户组权限发生变化时，可以将这些权限移除。本文为您介绍如何移除用户组的权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击授权。
3. 找到目标用户组，单击移除授权。
4. 单击确认。

2.9 删除用户组

当不再需要某个用户组时，可以删除该用户组。删除用户组会将所有用户从用户组中移除并撤销用户组拥有的权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户组。
3. 在用户组名称/显示名称列表下，找到目标用户组，单击删除。

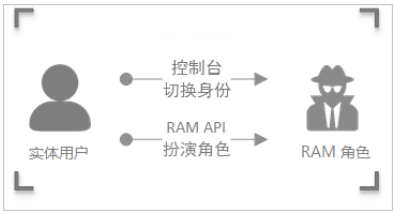
4. **单击确认**。

3 角色

3.1 RAM 角色概览

RAM角色（RAM role）与RAM用户一样，都是RAM身份类型的一种。RAM角色是一种虚拟用户，没有确定的身份认证密钥，需要被一个受信的实体用户扮演才能正常使用。

RAM角色基本概念



RAM角色（RAM role）	RAM角色是一种虚拟用户，与实体用户（云账号、RAM用户和云服务）和教科书式角色（Textbook role）不同。 · 实体用户：拥有确定的登录密码或访问密钥。 · 教科书式角色：教科书式角色或传统意义上的角色是指一组权限集合，类似于RAM里的权限策略。如果一个用户被赋予了这种角色，也就意味着该用户被赋予了一组权限，可以访问被授权的资源。 · RAM角色：RAM角色有确定的身份，可以被赋予一组权限策略，但没有确定的登录密码或访问密钥。RAM角色需要被一个受信的实体用户扮演，扮演成功后实体用户将获得RAM角色的安全令牌，使用这个安全令牌就能以角色身份访问被授权的资源。
角色ARN（Role ARN）	ARN是角色的全局资源描述符，用来指定具体角色。ARN遵循阿里云ARN的命名规范。例如，某个云账号下的devops角色的ARN为：`acs:ram::1234567890 12****:role/samplerole`。创建角色后，单击角色名后，可在基本信息页查看其ARN。
可信实体（Trusted entity）	角色的可信实体是指可以扮演角色的实体用户身份。创建角色时必须指定可信实体，角色只能被受信的实体扮演。可信实体可以是受信的阿里云账号、受信的阿里云服务或身份提供商。
权限策略（Policy）	一个角色可以绑定一组权限策略。没有绑定权限策略的角色也可以存在，但不能访问资源。

扮演角色（Assume role）	扮演角色是实体用户获取角色身份的安全令牌的方法。一个实体用户调用STS API AssumeRole可以获得角色的安全令牌，使用安全令牌可以访问云服务API。
切换身份（Switch role）	切换身份是在控制台中实体用户从当前登录身份切换到角色身份的方法。一个实体用户登录到控制台之后，可以切换到被许可扮演的某一种角色身份，然后以角色身份操作云资源。当用户不需要使用角色身份时，可以从角色身份切换回原来的登录身份。
角色令牌（Role token）	角色令牌是角色身份的一种临时访问密钥。角色身份没有确定的访问密钥，当一个实体用户要使用角色时，必须通过扮演角色来获取对应的角色令牌，然后使用角色令牌来调用阿里云服务API。

RAM角色的使用方法

1. RAM角色指定可信实体，即指定可以扮演角色的实体用户身份。
2. 可信实体通过控制台或调用API扮演角色并获取角色令牌。
   - 通过控制台扮演角色：切换身份是在控制台中实体用户从当前登录身份切换到RAM角色身份的方法，详情请参见#unique_29。
   - 通过调用API扮演角色：一个实体用户通过调用AssumeRole可以获得角色令牌，使用角色令牌可以访问云服务API。

> 说明:

扮演角色是实体用户获取RAM角色令牌的方法，角色令牌是角色身份的一种临时访问凭证，使用角色令牌可以访问阿里云资源。

3. 为RAM角色绑定权限策略，详情请参见#unique_30。

说明:

一个RAM角色可以绑定一组权限策略，没有绑定权限策略的角色也可以存在，但不能访问资源。

4. 受信实体通过扮演角色，使用角色令牌访问阿里云资源。

RAM角色类型

根据RAM可信实体的不同，RAM支持以下三种类型的角色：

- 阿里云账号：允许RAM用户所扮演的角色。扮演角色的RAM用户可以属于自己的云账号，也可以属于其他云账号。此类角色主要用来解决跨账号访问和临时授权问题。
- 阿里云服务：允许云服务所扮演的角色。此类角色主要用于授权云服务代理您进行资源操作。
- 身份提供商：允许受信身份提供商下的用户所扮演的角色。此类角色主要用于实现与阿里云的SSO。

RAM角色的应用场景

- #unique_31
- #unique_32
- #unique_33

3.2 创建 RAM 角色

3.2.1 创建可信实体为阿里云账号的 RAM 角色

阿里云支持三种不同类型的 RAM 角色，本文介绍如何创建可信实体为阿里云账号的 RAM 角色。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 RAM 角色管理。
3. 单击新建 RAM 角色，选择可信实体类型为阿里云账号，单击下一步。
4. 输入角色名称和备注。
5. 选择云账号后，单击完成。

说明:

若选择其他云账号，需要填写其他云账号的 ID。

后续步骤

成功创建角色后，角色没有任何权限，单击为角色授权可直接为该角色授权。详情请参考：#unique_36。

3.2.2 创建可信实体为阿里云服务的 RAM 角色

阿里云支持三种不同类型的 RAM 角色，本文介绍如何创建可信实体为阿里云服务的 RAM 角色。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 RAM 角色管理。
3. 单击新建 RAM 角色，选择可信实体类型为阿里云服务，单击下一步。
4. 输入角色名称和备注。
5. 选择受信服务后，单击完成。

> 说明:
> 更多受信服务请以实际界面为准。

后续步骤

成功创建角色后，角色没有任何权限，单击为角色授权可直接为该角色授权。详情请参考：#unique_38。

3.2.3 创建可信实体为身份提供商的 RAM 角色

阿里云支持三种不同类型的 RAM 角色，本文介绍如何创建可信实体为身份提供商的 RAM 角色。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 RAM 角色管理。
3. 单击新建 RAM 角色，选择可信实体类型为身份提供商，单击下一步。
4. 输入角色名称和备注。
5. 选择身份提供商并查看限制条件后，单击完成。

> 说明:
> 目前只支持一个条件关键字`saml:recipient`，必选且不能修改。

后续步骤

成功创建角色后，角色没有任何权限，单击为角色授权可直接为该角色授权。详情请参考：#unique_38。

3.3 查看 RAM 角色基本信息

本文为您介绍如何查看 RAM 角色基本信息，包括 RAM 角色名称、创建时间和 ARN 等信息。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 RAM 角色管理。
3. 在RAM 角色名称列表下，单击目标 RAM 角色名称。
4. 在基本信息区域，可以查看 RAM 角色基本信息。

> 说明:
> RAM 角色信息只能查看，不能修改。

3.4 为 RAM 角色授权

本文为您介绍如何为 RAM 角色授权。您可以为可信实体为阿里云账号、阿里云服务或身份提供商的 RAM 角色进行授权。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击授权。
3. 单击新增授权。
4. 在被授权主体区域下，输入 RAM 角色名称后，单击需要授权的 RAM 角色。
5. 在左侧权限策略名称列表下，单击需要授予 RAM 角色的权限策略。

> 说明:
> 在右侧区域框，选择某条策略并单击 ×，可撤销该策略。

6. 单击确定。

3.5 为 RAM 角色移除权限

当 RAM 角色不再需要某些权限时，可以将这些权限移除。本文为您介绍如何移除 RAM 角色的权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击授权。
3. 找到目标 RAM 角色，单击移除授权。
4. 单击确认。

3.6 编辑 RAM 角色策略内容

本文为您介绍如何通过修改角色的策略内容来改变允许扮演该角色的可信实体。

背景信息

策略中的 `Principal` 部分决定了允许扮演该角色的可信实体，通过修改 `Principal` 的内容可以改变该可信实体。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 RAM 角色管理。
3. 在RAM 角色名称列表下，单击目标 RAM 角色名称。
4. 在信任策略管理页签下，单击修改信任策略。

> 说明:
> 可以参考#unique_44编辑策略内容。

5. 单击确认。

3.7 修改 RAM 角色的可信实体

通过修改 RAM 角色的策略内容，可以修改 RAM 角色的可信实体。本文通过示例为您介绍如何修改 RAM 角色的可信实体为阿里云账号、阿里云服务或身份提供商。

修改 RAM 角色的可信实体为阿里云账号

若`Principal`中有 RAM 字段，表示该 RAM 角色的可信实体为阿里云账号，即可以被受信云账号下授权的 RAM 用户扮演。

以下述策略为例：该 RAM 角色可以被阿里云账号（AccountID=123456789012****）下授权的任何 RAM 用户扮演。

```
{
    "Statement": [
        {
```

```
            "Action": "sts:AssumeRole",
            "Effect": "Allow",
            "Principal": {
                "RAM": [
                    "acs:ram::123456789012****:root"
                ]
            }
        }
    ],
    "Version": "1"
}
```

若您将Principal中的内容更改如下，则表示该 RAM 角色可以被阿里云账号（AccountID=123456789012****）下的用户testuser扮演。

```
            "Principal": {
                "RAM": [
                    "acs:ram::123456789012****:user/testuser"
```

修改 RAM 角色的可信实体为阿里云服务

若Principal中有 Service 字段，表示该 RAM 角色的可信实体为阿里云服务，即可以被受信云服务扮演。

以下述策略为例：该 RAM 角色可以被当前云账号下的 ECS 服务扮演。

```
{
    "Statement": [
        {
            "Action": "sts:AssumeRole",
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "ecs.aliyuncs.com"
                ]
            }
        }
    ],
    "Version": "1"
}
```

修改 RAM 角色的可信实体为身份提供商

若Principal中有 Federated 字段，表示该 RAM 角色的可信实体为身份提供商，即可以被受信身份提供商下的用户扮演。

以下述策略为例：该 RAM 角色可以被当前云账号（AccountID=123456789012****）中的身份提供商testprovider下的用户扮演。

```
{
    "Statement": [
```

```
        {
            "Action": "sts:AssumeRole",
            "Effect": "Allow",
            "Principal": {
                "Federated": [
                    "acs:ram::123456789012****:saml-provider/
testprovider"
                ]
            },
            "Condition":{
                "StringEquals":{
                    "saml:recipient":"https://signin.aliyun.com/saml-
role/sso"
                }
            }
        }
    ],
    "Version": "1"
}
```

3.8 使用 RAM 角色

本文针对受信实体为阿里云账号的 RAM 角色为您介绍 RAM 用户如何扮演 RAM 角色登录控制台。

前提条件

> **说明:**
> 为了安全起见，阿里云不允许受信云账号以自己的身份扮演角色，如果一个实体用户想扮演某个 RAM 角色，该实体用户必须先以自己身份登录，然后将自己从实体身份切换到 RAM 角色身份。

使用 RAM 角色前，请先创建一个 RAM 用户。为该 RAM 用户创建访问密钥或设置登录密码并进行相应授权。

1. #unique_47。
2. 为该 RAM 用户创建访问密钥或设置登录密码。
   - 关于如何创建访问密钥，请参考：#unique_48。
   - 关于如何设置登录密码，请参考：#unique_49。
3. 为该 RAM 用户授权，授权时添加系统权限策略：`AliyunSTSAssumeRoleAccess`。

   关于如何为 RAM 用户授权，请参考：为 RAM 用户授权。

操作步骤

1. RAM 用户登录 RAM 控制台。
2. 将鼠标悬停在右上角头像的位置，单击切换身份。

3. 在角色切换页面，输入相应账号别名或默认域名以及角色名，单击切换。

> 说明:
> 切换成功后，用户将以 RAM 角色身份登录控制台，控制台右上角头像位置将显示角色身份（即当前身份）和登录身份，此时用户只能执行该角色身份被授权的所有操作。

4. 在扮演角色身份时，将鼠标悬停在右上角头像的位置，单击返回登录身份可以切换回登录身份。

后续步骤

RAM 用户也可以使用 API 扮演 RAM 角色。

当 RAM 用户被授予 `AliyunSTSAssumeRoleAccess` 权限策略之后，可以使用其访问密钥调用 STS API#unique_50 接口，以获取某个角色的安全令牌，从而使用安全令牌访问阿里云。

3.9 删除 RAM 角色

当不再需要某个 RAM 角色时，可以删除该 RAM 角色。

前提条件

> 说明:
> 删除角色前，角色不能有任何权限策略。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 RAM 角色管理。
3. 在RAM 角色名称列表下，找到目标 RAM 角色，单击删除。
4. 单击确认。

4 权限策略

4.1 权限策略概述

权限指在某种条件下允许或拒绝对某些资源执行某些操作，权限策略是一组访问权限的集合。

权限（Permission）

阿里云使用权限来描述用户、用户组、角色对具体资源的访问能力，下面为您介绍云账号、RAM 用户、资源创建者所拥有的权限：

- 云账号（资源属主）控制所有权限。
  - 每个资源有且仅有一个资源属主，该资源属主必须是云账号，对资源拥有完全控制权限。
  - 资源属主不一定是资源创建者。例如：一个 RAM 用户被授予创建资源的权限，该用户创建的资源归属于云账号，该用户是资源创建者但不是资源属主。
- RAM 用户（操作员）默认无任何权限。
  - RAM 用户代表的是操作员，其所有操作都需被云账号显式授权。
  - 新建的 RAM 用户默认没有任何操作权限，只有在被授权之后，才能通过控制台和 API 操作资源。
- 资源创建者（RAM 用户）默认对所创建资源的没有任何权限。
  - RAM 用户被授予创建资源的权限，用户将可以创建资源。
  - RAM 用户默认对所创建资源的没有任何权限，除非资源属主对 RAM 用户有显式的授权。

权限策略（Policy）

权限策略是用语法结构描述的一组权限的集合，可以精确地描述被授权的资源集、操作集以及授权条件。权限策略是描述权限集的一种简单语言规范，RAM 支持的语言规范请参考：#unique_54。

在 RAM 中，权限策略是一种资源实体，RAM 支持以下两种权限策略：

- 阿里云管理的系统策略：统一由阿里云创建，用户只能使用不能修改，策略的版本更新由阿里云维护。
- 客户管理的自定义策略：用户可以自主创建、更新和删除，策略的版本更新由客户自己维护。

通过为 RAM 用户、用户组或 RAM 角色绑定权限策略，可以获得权限策略中指定的访问权限。详情请参考：#unique_55、#unique_17和#unique_36。

为 RAM 主体绑定权限策略

为 RAM 主体授权，指为用户、用户组或角色绑定一个或多个权限策略。

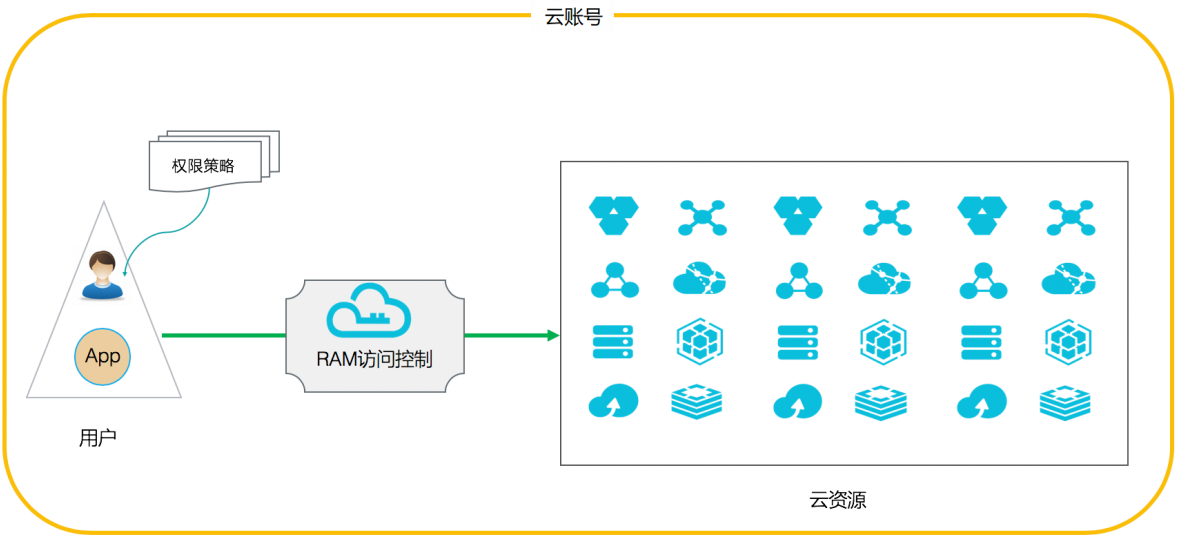
- 绑定的权限策略可以是系统策略也可以是自定义策略。
- 如果绑定的权限策略被更新，更新后的权限策略自动生效，无需重新绑定权限策略。

4.2 权限策略模型

阿里云提供了云账号内授权和资源组内授权两级授权能力，您可以根据需要选择合理的授权模型。

云账号内授权模型

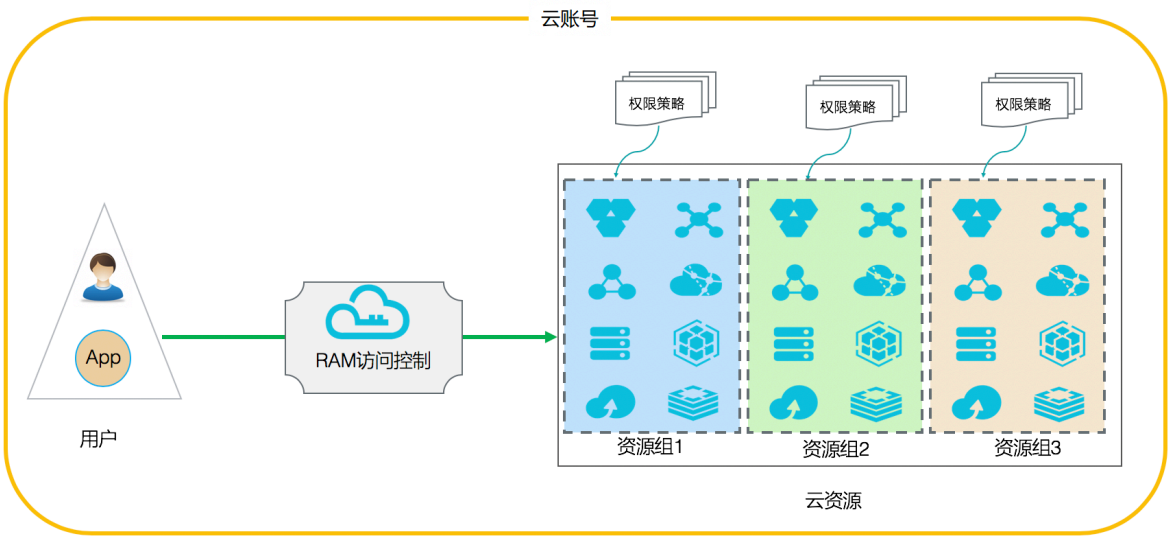
云账号内授权：对一个 RAM 身份主体添加权限策略时，该策略的可授权范围是云账号内的所有资源，这是最常见的一种权限模型。



资源组内授权模型

资源组内授权：在某个资源组内对一个 RAM 身份主体添加权限策略时，该策略的可授权范围仅仅是该资源组内的资源。

管理员：在资源组内拥有`AdministratorAccess`系统策略的用户，资源组创建者默认为管理员。资源组管理员可以在资源组的成员管理中添加其他的 RAM 用户并在资源组内进行授权。



4.3 查看权限策略基本信息

本文为您介绍如何查看权限策略基本信息，包括权限策略名称、备注、策略类型和被引用次数等信息。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击权限策略管理。
3. 在搜索框中，输入策略名称或备注。
4. 策略类型选择系统策略或自定义策略，可以查看权限策略。

> **说明:**
> 系统策略用户只能查看不能修改，自定义策略用户可以自行创建、查看和修改。

4.4 自定义策略

4.4.1 创建自定义策略

自定义策略可以更大程度的满足您的细粒度的要求，从而实现更灵活的权限管理。

前提条件

创建自定义策略前，需要先了解权限策略语言的基本结构和语法，请参考：#unique_60。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击权限策略管理。

3. 单击新建权限策略。
4. 填写策略名称和备注。
5. 配置模式选择可视化配置或脚本配置。
   - 若选择可视化配置：单击添加授权语句，根据界面提示，对权限效力、操作名称和资源等进行配置。
   - 若选择脚本配置，请参考#unique_60编辑策略内容。
6. 单击确认。

4.4.2 修改自定义策略内容

当用户的权限发生变更时，您可以根据需要修改策略内容。

背景信息

当需要新增或撤销权限时，可能存在以下需求：

- 希望一段时间后，老的权限策略还能继续使用。
- 修改策略内容后，如果权限策略修改有误，需要使用修改前的权限策略。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击权限策略管理。
3. 在权限策略名称列表下，单击目标权限策略名称。

   说明：
   RAM 支持两种权限策略，其中系统策略只能查看不支持修改，自定义策略支持创建、查看和修改。
4. 在策略内容页签下，单击修改策略内容。

   说明：
   可以参考#unique_60编辑策略内容。
5. 单击确认。

   说明：
   修改完成后，系统会自动生成一个新的版本，此版本将变为默认版本。

4.4.3 管理自定义策略版本

本文为您介绍如何管理自定义策略版本，包括查看权限版本、设置当前版本和删除权限版本。

背景信息

权限策略具备版本管理机制：

- 可以为一个权限策略保留多个版本。
- 如果版本数量超出限制，需要手动删除不需要的版本。
- 对于一个存在多版本的权限策略，只有一个版本是活跃的，即当前版本（默认版本）。
- 当前版本（默认版本）只能查看，不能删除。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击权限策略管理。
3. 在权限策略名称列表下，单击目标权限策略名称。
4. 在版本管理页签下，您可以查看、设置和删除权限策略版本。
   - 查看权限版本：单击查看可以查看权限策略的版本号和策略内容。
   - 设置默认版本：找到目标版本，单击操作列表下的设为当前版本，可以将选定版本设为默认版本。
   - 删除权限版本：找到不需要的非默认版本，单击操作列表下的删除，单击确认，可以删除不需要的版本。

4.4.4 删除自定义策略

当权限发生变化或不再需要某个自定义策略时，可以删除自定义策略。

前提条件

- 删除权限策略前，应保证当前权限策略不存在多版本，只有一个默认版本。若该权限策略存在多个版本，您需要先删除除默认版本之外的所有版本。
- 删除权限策略前，应保证当前权限策略未被引用（即授予 RAM 用户、用户组或 RAM 角色）。若该权限策略已被引用，您需要在该权限策略的引用记录中移除授权。请参考：#unique_64。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击权限策略管理。
3. 在权限策略名称列表下，找到目标权限策略，单击删除。
4. 单击确认。

4.5 管理权限策略引用记录

本文为您介绍如何管理权限策略引用记录，包括查看权限策略引用记录和删除权限策略引用记录。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的权限管理菜单下，单击权限策略管理。
3. 在权限策略名称列表下，单击目标权限策略名称。
4. 在引用记录页签下，您可以查看或删除引用记录。
   - 查看引用记录：您可以查看被授权主体和主体类型等信息。
   - 删除引用记录（移除权限）：单击操作列表下的移除授权，单击确认可以移除引用记录。

4.6 权限策略语言

4.6.1 权限策略基本元素

权限策略基本元素是权限策略的基本组成部分，RAM 中使用权限策略来描述授权的具体内容，掌握权限策略基本元素的基本知识可以更好的使用权限策略。

基本元素

元素名称	描述
效力（Effect）	授权效力包括两种：允许（Allow）和拒绝（Deny）。
操作（Action）	操作是指对具体资源的操作。
资源（Resource）	资源是指被授权的具体对象。
限制条件（Condition）	限制条件是指授权生效的限制条件。

使用规则

- 效力（Effect）

  取值为：允许（Allow）或拒绝（Deny）。

  > 说明:
  > 当权限策略中既有允许（Allow）又有拒绝（Deny）的授权语句时，遵循 Deny 优先的原则。

  样例：`"Effect": "Allow"`。

- 操作（Action）

  操作支持多值，取值为：云服务所定义的 API 操作名称。

  > 说明:
  > 多数情况下 操作与云产品的 API 一一对应，但也有例外。各产品支持的操作列表请参考：#unique_68。

  格式：`<service-name>:<action-name>`。

  - `service-name`：阿里云产品名称。
  - `action-name: service`：相关的 API 操作接口名称。

  样例：`"Action": ["oss:ListBuckets", "ecs:Describe*", "rds:Describe*"]`

- 资源（Resource）

  资源是指被授权的具体对象。

  格式：`acs:<service-name>:<region>:<account-id>:<relative-id>`。

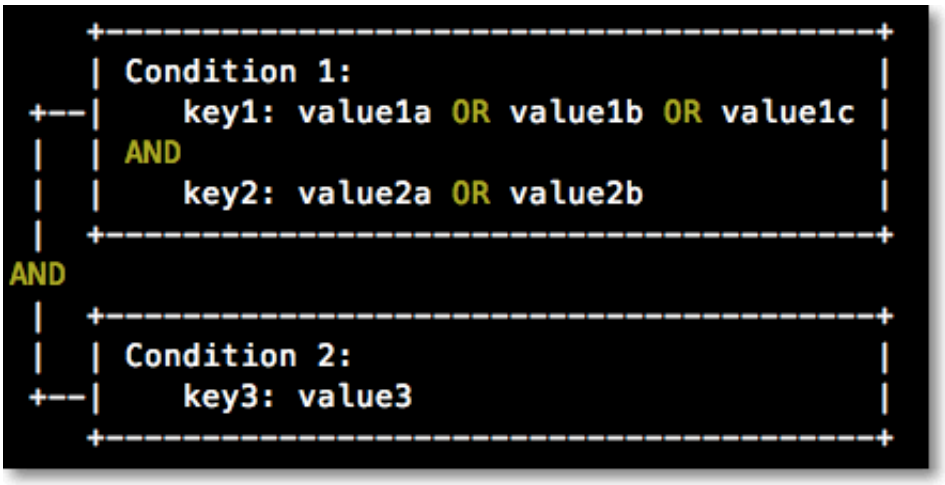
  - `acs`：Alibaba Cloud Service 的首字母缩写，表示阿里云的公有云平台。
  - `service-name`：阿里云产品名称。
  - `region`：地域信息。如果不支持该项，可以使用通配符`*`来代替。
  - `account-id`：账号 ID。例如：`123456789012****`，可以用`*`代替。
  - `relative-id`：与服务相关的资源描述部分，其语义由具体服务指定。这部分的格式支持树状结构（类似文件路径）。以 OSS 为例，表示一个 OSS 对象的格式为：`relative-id = “mybucket/dir1/object1.jpg”`。

  样例：`"Resource": ["acs:ecs:*:*:instance/inst-001", "acs:ecs:*:*:instance/inst-002", "acs:oss:*:*:mybucket", "acs:oss:*:*:mybucket/*"]`。

- 限制条件（Condition）

  条件块（Condition Block）由一个或多个条件子句构成。一个条件子句由条件操作类型、条件关键字和条件值组成。

  图 4-1: 条件块判断逻辑



  逻辑说明

  - 条件满足：一个条件关键字可以指定一个或多个值，在条件检查时，如果条件关键字的值与指定值中的某一个相同，即可判定条件满足。
  - 条件子句满足：同一条件操作类型的条件子句下，若有多个条件关键字，所有条件关键字必须同时满足，才能判定该条件子句满足。
  - 条件块满足：条件块下的所有条件子句同时满足的情况下，才能判定该条件块满足。

  条件操作类型

  条件操作类型包括：字符串类型（String）、数字类型（Numeric）、日期类型（Date and time）、布尔类型（Boolean）和 IP 地址类型（IP address）。

条件操作类型	支持类型
字符串类型（String）	- StringEquals - StringNotEquals - StringEqualsIgnoreCase - StringNotEqualsIgnoreCase - StringLike - StringNotLike

条件操作类型	支持类型
数字类型（Numeric）	- NumericEquals - NumericNotEquals - NumericLessThan - NumericLessThanEquals - NumericGreaterThan - NumericGreaterThanEquals
日期类型（Date and time）	- DateEquals - DateNotEquals - DateLessThan - DateLessThanEquals - DateGreaterThan - DateGreaterThanEquals
布尔类型（Boolean）	Bool
IP 地址类型（IP address）	- IpAddress - NotIpAddress

条件关键字

- **阿里云通用条件关键字命名格式：**

```
acs:<condition-key>
```

通用条件关键字	类型	描述
`acs:CurrentTime`	Date and time	Web Server 接收到请求的时间。以 ISO 8601 格式表示，例如：`2012-11-11T23:59:59Z`。
`acs:SecureTransport`	Boolean	发送请求是否使用了安全信道。例如：HTTPS。
`acs:SourceIp`	IP address	发送请求时的客户端 IP 地址。

通用条件关键字	类型	描述
acs:MFAPresent	Boolean	用户登录时是否使用了多因素认证。

- 阿里云产品级别条件关键字命名格式：

```
<service-name>:<condition-key>
```

产品级别条件关键字	产品名称	类型	描述
ecs:tag/<tag-key>	ECS	String	ECS 资源的标签关键字，可自定义。
rds:ResourceTag/<tag-key>	RDS	String	RDS 资源的标签关键字，可自定义。
oss:Delimiter	OSS	String	OSS 对 Object 名字进行分组的分隔符。
oss:Prefix	OSS	String	OSS Object 名称的前缀。

4.6.2 权限策略语法和结构

本文介绍 RAM 中权限策略的语法和结构，帮助您正确理解权限策略语法，以完成创建或更新权限策略。

运用权限策略语法的前提条件

运用权限策略语法前，首先应了解权限策略字符及其使用规则。

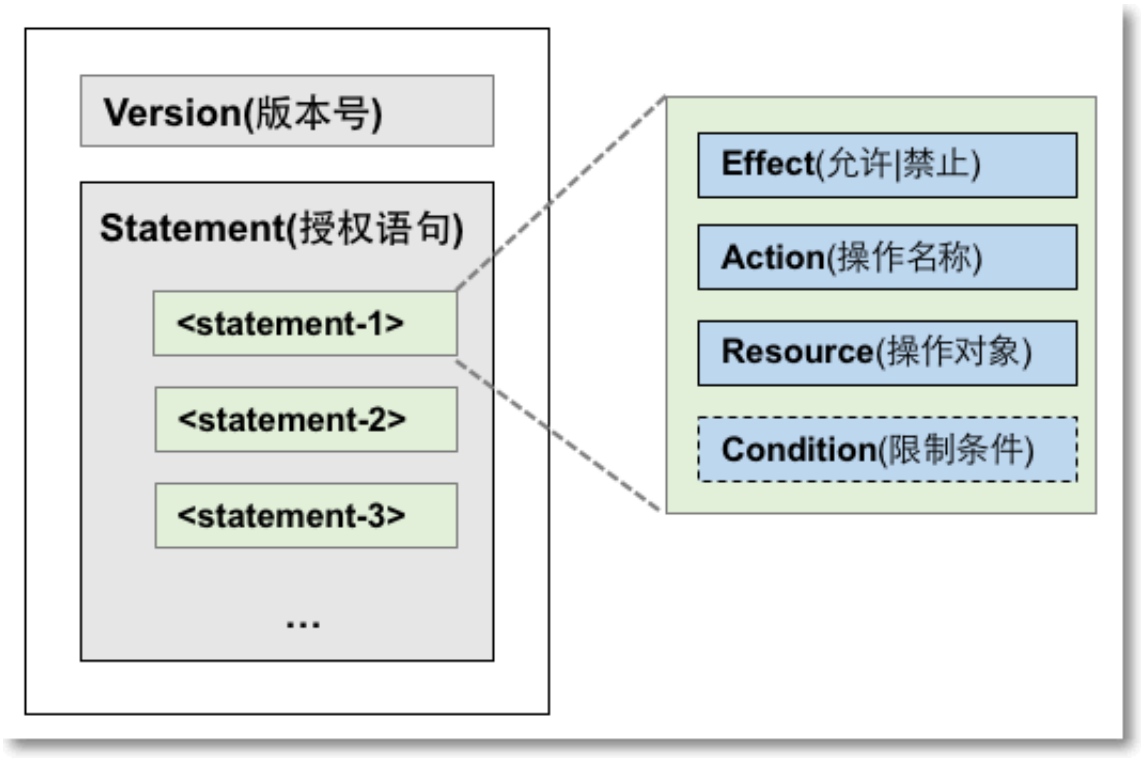
- 权限策略字符
  - 权限策略中所包含的 JSON 字符：`{ } [ ] " , :`。
  - 描述语法使用的特殊字符：`= < > ( ) |`。

- 字符使用规则
  - 当一个元素允许多值时，可以使用下述两种方式表达，效果相同。
    - 使用逗号和省略号进行表达。例如：`[ <action_string>, <action_string>, ...]`。
    - 使用单值进行表达。例如：`"Action": [<action_string>]` 和 `"Action": <action_string>`。
  - 元素带有问号表示此元素是一个可选元素。例如：`<condition_block?>`。
  - 多值之间用竖线 `|` 隔开，表示取值只能选取这些值中的某一个。例如：`("Allow" | "Deny")`。
  - 使用双引号的元素，表示此元素是文本串。例如：`<version_block> = "Version" : ("1")`。

权限策略结构

权限策略结构包括：

- 版本号。
- 授权语句列表。每条授权语句包括授权效力（Effect）、操作（Action）、资源（Resource）以及限制条件（Condition，可选项）。



权限策略语法

```
policy  = {
     <version_block>,
     <statement_block>
}
<version_block> = "Version" : ("1")
<statement_block> = "Statement" : [ <statement>, <statement>, ... ]
<statement> = {
    <effect_block>,
    <action_block>,
    <resource_block>,
    <condition_block?>
}
<effect_block> = "Effect" : ("Allow" | "Deny")
<action_block> = ("Action" | "NotAction") :
    ("*" | [<action_string>, <action_string>, ...])
<resource_block> = ("Resource" | "NotResource") :
    ("*" | [<resource_string>, <resource_string>, ...])
<condition_block> = "Condition" : <condition_map>
<condition_map> = {
  <condition_type_string> : {
      <condition_key_string> : <condition_value_list>,
      <condition_key_string> : <condition_value_list>,
      ...
  },
  <condition_type_string> : {
      <condition_key_string> : <condition_value_list>,
      <condition_key_string> : <condition_value_list>,
      ...
  }, ...
}
<condition_value_list> = [<condition_value>, <condition_value>, ...]
<condition_value> = ("String" | "Number" | "Boolean")
```

权限策略语法说明：

- **版本：当前支持的权限策略版本为 1。**
- **授权语句：一个权限策略可以有多条授权语句。**
  - **每条授权语句的效力为：`Allow`或`Deny`。**

    说明：

    一条授权语句中，操作（Action）和资源（Resource）都支持多值。
  - **每条授权语句都支持独立的限制条件（Condition）。**

    说明：

    一个条件块可以支持多种条件操作类型，以及多种条件的逻辑组合。
- **Deny 优先原则：一个用户可以被授予多个权限策略，当这些权限策略同时包含`Allow`和`Deny`时，遵循 Deny 优先原则。**

- 元素取值：
  - 当元素取值为数字（Number）或布尔值（Boolean）时，与字符串类似，需要使用双引号。
  - 当元素取值为字符串值（String）时，支持使用*和?进行模糊匹配。
    - *代表 0 个或多个任意的英文字母。 例如：ecs:Describe* 表示 ECS 的所有以Describe开头的操作。
    - ?代表 1 个任意的英文字母。

权限策略格式检查

RAM 仅支持 JSON 格式。当创建或更新权限策略时，RAM 会首先检查 JSON 格式的正确性。

- 关于 JSON 的语法标准请参考：RFC 7159。
- 您也可以使用一些在线的 JSON 格式验证器和编辑器来校验 JSON 文本的有效性。

4.6.3 权限策略检查规则

本文为您介绍了几种不同的权限策略检查规则，掌握权限策略检查规则可以更好的理解权限策略。

权限策略检查规则

在 RAM 中访问阿里云资源分为三种类型：以主账号身份访问、以 RAM 用户身份访问、以 RAM 角色身份访问。

针对上述不同的访问类型，系统的权限检查规则如下表所示。

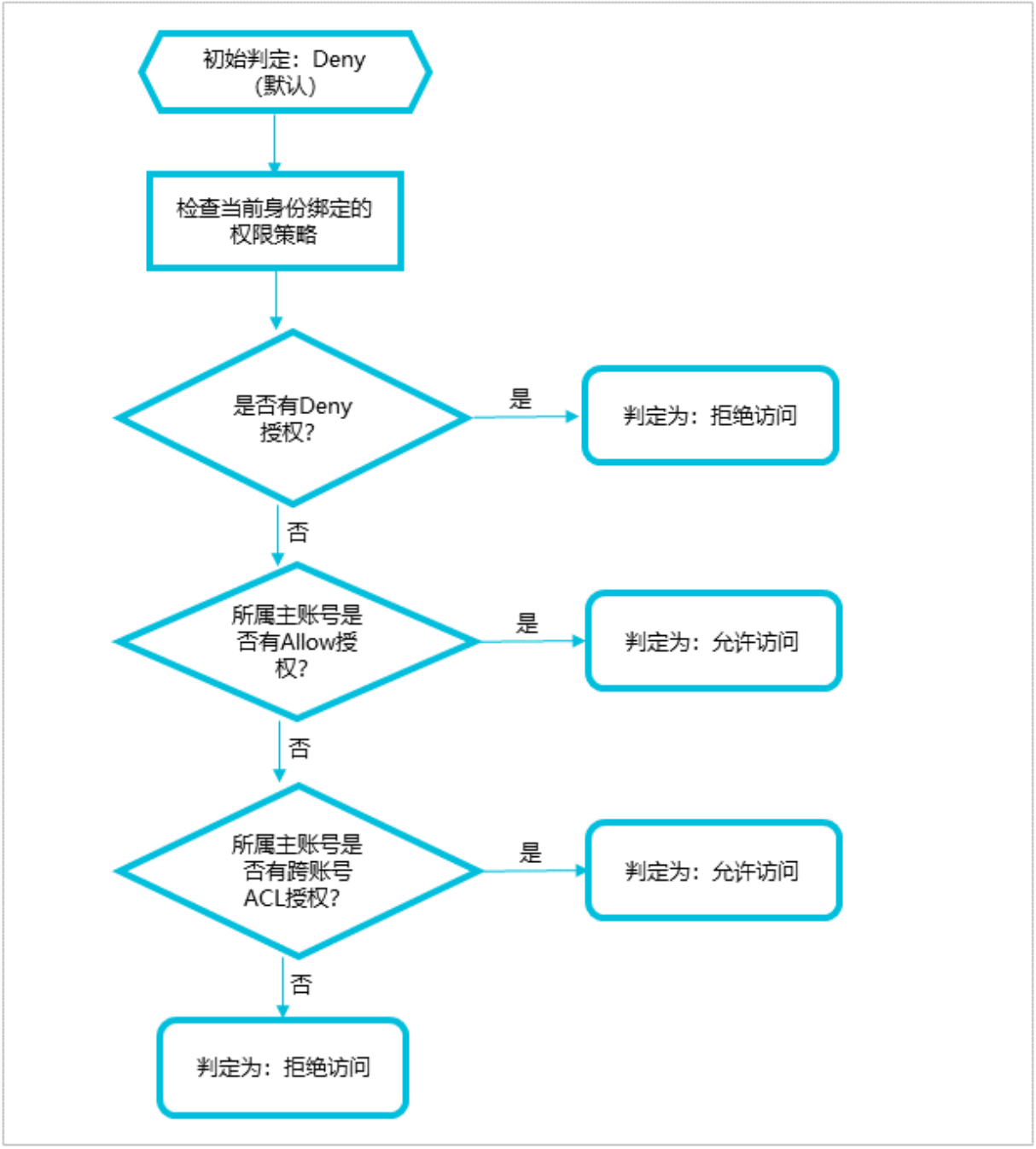
访问类型	权限检查规则
以主账号身份访问	主账号是资源所有者，默认可以访问该账号下的所有资源。  说明: 少数阿里云产品（例如：日志服务）支持跨云账号进行访问控制列表（ACL）授权，如果通过 ACL 授权检查，则允许访问相应资源。

访问类型	权限检查规则
以 RAM 用户身份访问	· 主账号对 RAM 用户有显式的授权。 · RAM 用户所属的主账号对资源有访问权限。 说明: RAM 用户访问资源时，默认没有任何权限，以上条件需同时满足 RAM 用户才能访问相应资源。 具体权限检查规则请参考：RAM 用户的权限策略检查规则。
以 RAM 角色身份访问	· RAM 角色令牌有相应的权限策略。 RAM 角色令牌相关信息，请参考：STS 简介。 · 主账号对 RAM 角色有显式的授权。 · RAM 角色所属的主账号对资源有访问权限。 说明: RAM 角色访问资源时，默认没有任何权限，以上条件需同时满足 RAM 角色才能访问相应资源。 具体权限检查规则请参考：RAM 角色的权限策略检查规则。

RAM 用户的权限策略检查规则

RAM 用户默认没有任何权限，主账号对 RAM 用户进行显示授权后，RAM 用户可以访问相应的资源。

说明:
权限策略支持 Allow（允许）和 Deny（禁止）两种授权类型，当同时出现 Allow 和 Deny 授权时，遵循 Deny 优先原则。



1. 检查 RAM 用户所绑定权限策略是否有授权：
   - 如果有 Deny 授权，判定为：拒绝访问。
   - 否则，需要进行下一步检查。
2. 检查 RAM 用户所属的主账号是否有访问权限：
   - 如果有 Allow 授权，判定为：允许访问。
   - 否则，需要进行下一步检查。

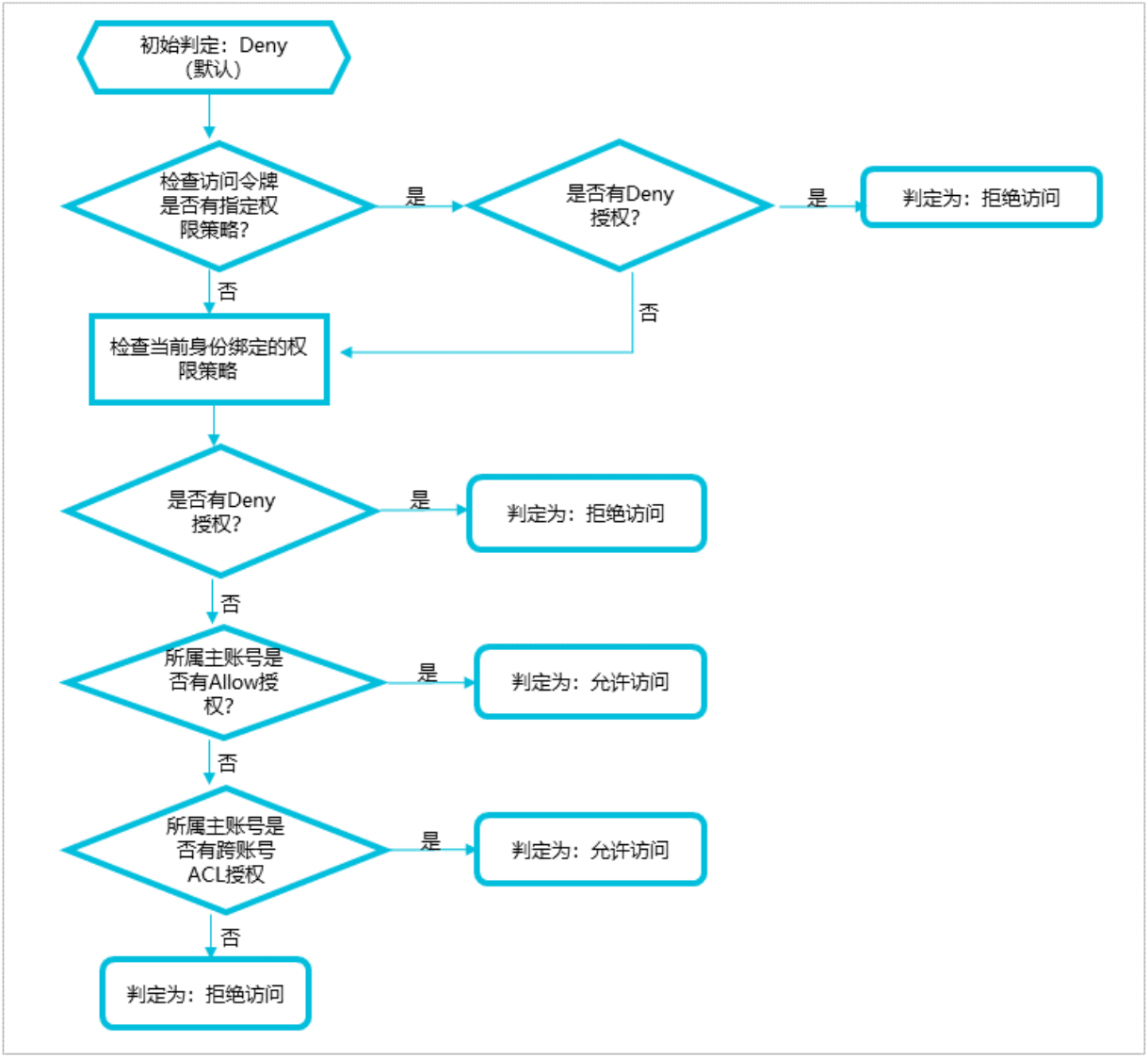
3. 检查 RAM 用户所属的主账号是否有跨账号 ACL 授权：

   · 如果有 ACL 授权，判定为：允许访问。

   · 否则，判定为：拒绝访问。

RAM 角色的权限策略检查规则

RAM 角色可以使用角色访问令牌访问阿里云资源，调用 #unique_72，请求参数Policy可以控制访问阿里云资源的权限。



1. 检查访问令牌是否有指定权限策略：

   · 如果有指定权限策略，需要查看是否有 Deny 授权：

     - 如果有 Deny 授权，判定为：拒绝访问。
     - 否则，需要检查 RAM 角色所绑定的权限策略。

   · 如果没有指定权限策略，需要检查 RAM 角色所绑定的权限策略。

2. 检查 RAM 角色所绑定的权限策略是否有授权：
   - 如果有 Deny 授权，判定为：拒绝访问。
   - 否则，需要进行下一步检查。
3. 检查 RAM 角色所属的主账号是否有访问权限：
   - 如果有 Allow 授权，判定为：允许访问。
   - 否则，需要进行下一步检查。
4. 检查 RAM 角色所属的主账号是否有跨账号 ACL 授权：
   - 如果有 ACL 授权，判定为：允许访问。
   - 否则，判定为：拒绝访问。

5 安全设置

5.1 安全设置概述

本文介绍了访问控制涉及的一些安全设置基本概念，这些安全设置可以更有效的保护账号安全。

登录密码（Password）

登录密码是登录阿里云的身份凭证，用于证明用户真实身份的凭证。

说明:

请妥善保管您的登录密码并定期更换。

关于如何设置登录密码，请参考：修改云账号登录密码和#unique_49。

默认域名（Default domain name）

阿里云为每个云账号分配了一个默认域名，格式为：<AccountAlias>.onaliyun.com。默认域名可作为 RAM 用户登录、单点登录（SSO）等场景下该云账号的唯一标识符。

关于如何设置默认域名，请参考：#unique_76。

域别名（Domain alias）

如果您持有公网上可以解析的域名，那么您可以使用该域名替代您的默认域名，该域名称为域别名。域别名就是指默认域名的别名。

说明:

您创建的域别名必须经过域名归属验证，才能使用。验证通过后，您可以使用域别名替代所有需要使用默认域名的场景。

关于如何设置域别名，请参考：#unique_77。

访问密钥（AccessKey）

访问密钥指的是访问身份验证中用到的 AccessKey ID 和 AccessKeySecret。您可以使用访问密钥（或阿里云服务 SDK）创建一个 API 请求，RAM 通过使用 AccessKey ID 和 AccessKeySecret 对称加密的方法来验证某个请求的发送者身份，身份验证成功后将可以操作相应资源。

AccessKey ID 和 AccessKeySecret 一起使用，AccessKey ID 用于标识用户，AccessKeySecret 用于加密签名字符串和 RAM 用来验证签名字符串的密钥。

> 说明:
> AccessKeySecret 只在创建时显示，不支持查询，请妥善保管。

关于如何创建访问密钥，请参考：#unique_48。

多因素认证（MFA）

多因素认证是一种简单有效的最佳安全实践，在用户名和密码之外再增加一层安全保护。这些多重要素结合起来将为您的账号提供更高的安全保护。启用多因素认证后，再次登录阿里云时，系统将要求输入两层安全要素：

1. 第一安全要素：用户名和密码
2. 第二安全要素：多因素认证设备生成的验证码

关于如何设置多因素认证，请参考：#unique_78和#unique_79。

5.2 密码

5.2.1 修改云账号登录密码

为了提高账号的安全性，您可以定期修改密码，设置一个字母、符号或数字至少两项元素且长度超过 6 位的密码。

操作步骤

1. 登录阿里云控制台。
2. 将鼠标悬停在右上角头像的位置，单击安全设置。
3. 在安全设置页面下的登录密码区域，单击修改。
4. 在验证身份页面，根据页面提示选择合适的方式进行身份验证。
5. 验证成功后，输入新的登录密码和确认新的登录密码。
6. 单击确定。

5.2.2 设置 RAM 用户密码强度

为了保护账号安全，您可以编辑密码规则，包括密码长度、密码有效期和历史密码检查策略等。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击设置。

3. 在安全设置页签下，单击编辑密码规则，根据配置面板，配置相关参数。

   - 密码长度：密码长度范围为 8~32位。

     
     说明:
     为了保护账号安全，建议至少设置 8 位以上密码长度。

   - 密码中必须包含元素：可以根据需要勾选大写字母、小写字母、数字和符号。

     
     说明:
     此设置表示登录密码必须包含勾选项。 为了提高账号安全强度，上述元素中，建议至少勾选 2 项以上。

   - 密码有效期：单位为天，取值范围为 0~1095 天，默认为 0，表示永不过期。

     
     说明:
     重置密码将重置密码过期时间。

   - 密码过期后：表示密码过期后是否仍可以登录，根据需要勾选不可登录或不限制登录。
     - 不可登录：表示密码过期后必须由主账号重置密码，RAM 用户才能正常登录。
     - 不限制登录：表示 RAM 用户可以在密码过期后自行更改密码，并继续以 RAM 用户身份登录。
   - 历史密码检查策略：表示禁止使用前 *N* 次密码，取值范围为 0~24。默认取值为 0，表示不启用历史密码检查策略。
   - 密码重试约束：一小时内使用错误密码最多尝试登录 *N* 次，取值范围为 0~32。默认取值为 0，表示不启用密码重试约束。

     
     说明:
     重置密码可清除尝试登录次数。

4. 单击确认。

   
   说明:
   设置成功后，此密码规则适用于所有 RAM 用户。

5.2.3 修改 RAM 用户登录密码

云账号可以定期为 RAM 用户修改登录密码以提高账号安全性。

操作步骤

1. 登录 RAM 控制台。

2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在认证管理页签下，单击修改登录设置。
5. 在设置登录密码区域下，选择重新设置自定义密码。
6. 输入新的密码后，单击确认。

> 说明:
> 如果云账号允许 RAM 用户自主管理密码，RAM 用户也可以登录 RAM 控制台，单击安全管理，在密码管理菜单下，单击修改密码进行修改。

5.3 基本安全设置

5.3.1 进行账号安全检查

通过 RAM 安全报告可以评估账号的安全性，定期进行账号安全检查并进行相应设置可以更有效的保护账号安全。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的概览菜单下，可以检查账号安全。
3. 单击目标安全项的名称，单击前往设置，可以快速完成相应设置。

后续步骤

单击下载安全报告可以获取一份安全实践报告，其中列出了您账号中安全实践相关的状态。

- SubUser：表示云账号中 RAM 用户的个数。
- SubUserBindMfa：表示 RAM 用户是否绑定了多因素认证设备（MFA）。
- SubUserWithUnusedAccessKey：表示 RAM 用户未使用的访问密钥（AccessKey）的个数。
- RootWithAccessKey：表示云账号创建的访问密钥（AccessKey）的个数。
- SubUserWithOldAccessKey：表示 RAM 用户中旧的访问密钥（AccessKey）的个数。
- SubUserPwdLevel：表示 RAM 用户密码强度的等级。
- UnusedAkNum：表示云账号中未使用的访问密钥（AccessKey）的个数。
- OldAkNum：表示云账号中旧的访问密钥（AccessKey）的个数。
- BindMfa：表示云账号是否绑定多因素认证设备（MFA）。

- Score：表示账号安全最终得分。

> 说明：
> - 若您的得分较低，请您及时进行相应的安全设置。
> - 遵循最佳安全实践原则，可以更有效的保护账号及资产的安全。详情请参考：#unique_85。

5.3.2 修改 RAM 用户登录设置

如果创建 RAM 用户时没有设置控制台登录相关信息，您可以再次为 RAM 用户修改登录设置。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在认证管理页签下的控制台登录管理区域，单击修改登录设置，配置相关参数。
   - 控制台密码登录：表示是否允许 RAM 用户通过密码登录控制台。
   - 设置登录密码：表示是否允许 RAM 用户保留当前密码、重新自动生成默认密码或重新设置自定义密码。

     > 说明：
     > 如果勾选重新自动生成默认密码，设置完成后，新密码会自动弹出，请妥善保管。
   - 是否要求重置密码：表示是否要求 RAM 用户下次登录时重置密码。
   - 是否开启多因素认证：表示是否要求 RAM 用户开启多因素认证。

     > 说明：
     > 如果云账号要求 RAM 用户开启多因素认证，RAM 用户在登录时会直接进入多因素认证绑定流程。
5. 单击确认。

5.3.3 设置 RAM 用户安全策略

云账号可以通过修改 RAM 用户安全设置更好的管理 RAM 用户的权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击设置。

3. 在安全设置页签下，单击修改 RAM 用户安全设置，配置相关参数。
   - 保存 MFA 登录状态 7 天：表示是否允许 RAM 用户登录时保存多因素认证设备登录状态，有效期为 7 天，默认为不允许。
   - 自主管理密码：表示是否允许 RAM 用户修改密码。
   - 自主管理 AccessKey：表示是否允许 RAM 用户管理访问密钥。
   - 自主管理多因素设备：表示是否允许 RAM 用户绑定或解绑多因素认证设备。
   - 登录 session 过期时间：表示 RAM 用户登录有效期，单位为小时。
   - 登录掩码设置：登录掩码决定哪些 IP 地址会受到登录控制台的影响。默认为空字符串，不限制登录 IP。如果设置了登录掩码，使用密码登录或单点登录（SSO）时会受到影响，但使用访问密钥发起的 API 访问不受影响。
4. 单击确认。

> 说明:
> 设置成功后，此规则适用于所有 RAM 用户。

5.3.4 为云账号设置操作保护

通过设置操作保护二次验证您的身份，在控制台关键操作（释放或修改密码等）时，进一步提高账号安全性，有效的保护您安全使用阿里云产品。

操作步骤

1. 登录阿里云控制台。
2. 将鼠标悬停在右上角头像的位置，单击安全设置。
3. 在安全设置页面下的操作保护区域，单击设置。

> 说明:
> 目前操作保护设置仅对 ECS、RDS、OSS、DirectMail 控制台生效。

4. 在操作保护设置页面，根据页面提示配置保护强度和验证方式。
   - 保护强度：可以根据需要选择强制二次验证或系统默认规则。

     > 说明:
     > 如果选择系统默认规则，系统会自行判断是否需要二次验证。
   - 验证方式：可以根据需要选择 MFA、手机或邮箱。
5. 单击提交。

5.3.5 为云账号设置登录掩码

通过设置登录掩码，云账号只能从指定的 IP 地址进行登录，进一步提高了账号的安全性。

操作步骤

1. 登录阿里云控制台。
2. 将鼠标悬停在右上角头像的位置，单击安全设置。
3. 在安全设置页面下的登录掩码区域，单击设置。
4. 在登录掩码页面，输入正确的登录掩码。

> 说明:
> 当需要配置多个登录掩码时，请使用分号来分隔登录掩码，例如：192.168.0.0/16;10.0.0.0/8。

5. 单击保存。

> 说明:
> 设置完成后，云账号使用密码登录或单点登录（SSO）时会受到影响，但使用访问密钥发起的 API 访问不受影响。

5.4 高级设置

5.4.1 管理默认域名

每个云账号都有一个默认域名，RAM用户可以通过默认域名登录RAM控制台。本文为您介绍如何修改登录名后缀，便于用户记忆登录名称。

操作步骤

1. 云账号登录RAM控制台。
2. 在左侧导航栏的人员管理菜单下，单击设置。
3. 在高级设置页签下，可以查看或更新默认域名。
   - 查看默认域名：默认域名格式为 `<$AccountAlias>.onaliyun.com`。账号别名（AccountAlias）的默认值为`AccountID`，如果未设置过账号别名，此时默认域名的格式为 `<$AccountID>.onaliyun.com`。
   - 更新默认域名：单击更新，输入新的账号别名，单击确认。

后续步骤

RAM用户登录RAM控制台时可以使用默认域名登录。

此时RAM用户登录名称为：<$username>@<$AccountAlias>.onaliyun.com，即RAM用户登录名称@默认域名。详情请参见RAM用户登录控制台。

另外，使用默认域名可以简化SAML SSO的配置流程，详情请参见#unique_92。

5.4.2 创建域别名

域别名就是指默认域名的别名。如果您持有公网上可以解析的域别名，RAM用户便可以使用该域别名登录RAM控制台。

操作步骤

1. 云账号登录RAM控制台。
2. 在左侧导航栏的人员管理菜单下，单击设置。
3. 在高级设置页签下，单击创建域别名。
4. 填写域名称。
5. 单击确认。
6. 进行域名归属验证。

> 说明:
> 创建域别名成功后，复制弹出的随机验证码，到域名购买平台进行域名解析TXT记录设置；设置完毕后，再进行域名归属验证。

后续步骤

创建域别名后，RAM用户登录RAM控制台时可以使用域别名登录。

此时RAM用户登录名称为：<$username>@<$DomainAlias>，即RAM用户登录名称@域别名。详情请参见RAM用户登录控制台。

另外，使用域别名可以简化SAML SSO的配置流程，详情请参见#unique_92。

5.5 访问密钥

5.5.1 为 RAM 用户创建访问密钥

访问密钥（AccessKey）是 RAM 用户的长期凭证。如果为 RAM 用户创建了访问密钥，RAM 用户可以通过 API 或其他开发工具访问阿里云资源。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。

3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在用户 AccessKey 区域下，单击创建新的 AccessKey。

> 说明:
> 首次创建时需填写手机验证码。

5. 单击确认。

> 说明:
> - AccessKeySecret 只在创建时显示，不提供查询，请妥善保管。
> - 若 AccessKey 泄露或丢失，则需要创建新的 AccessKey，最多可以创建 2 个 AccessKey。

5.5.2 查看访问密钥基本信息

本文为您介绍如何查看访问密钥基本信息，目前可以查询的信息包括 AccessKey ID、状态、最后使用时间和创建时间等信息。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在用户 AccessKey区域下，可以查看访问密钥基本信息。

> 说明:
> AccessKeySecret 只在创建时显示，不提供查询。

5.5.3 禁用访问密钥

当 RAM 用户权限发生变化时或不再需要通过 API 访问阿里云资源，可以禁用其访问密钥。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在用户 AccessKey区域下，单击禁用。

> 说明:
> 单击激活可以重新激活访问密钥。

5. 单击确认。

5.5.4 删除访问密钥

当您不再需要通过 API 或其他开发工具访问阿里云资源时，可以删除访问密钥。

前提条件

删除访问密钥前，可以通过访问密钥的最后使用时间确认访问密钥的使用情况。

> **说明:**
> 删除访问密钥需慎重，在使用中的访问密钥一旦删除，可能会造成客户应用系统故障。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在用户 AccessKey区域下，单击删除。
5. 单击确认。

5.6 多因素认证

5.6.1 为云账号设置多因素认证

本文以阿里云应用为例介绍如何为云账号开启多因素认证（Multi-factor authentication，MFA）。开启多因素认证后，可以为您的账号提供更高的安全保护。

操作步骤

1. 登录阿里云控制台。
2. 将鼠标悬停在右上角头像的位置，单击安全设置。
3. 在安全设置页面下的虚拟 MFA 区域，单击设置。
4. 在验证身份页面，选择合适的方式根据页面提示进行身份验证。
5. 在移动设备端，下载并安装阿里云 应用，安装完成后单击下一步。
   - iOS：在 App Store 中搜索阿里云。
   - Android：在应用市场中搜索阿里云。
6. 在移动设备端，登录阿里云应用。
7. 在屏幕底部单击控制台页签。
8. 在常用工具区域下，单击虚拟 MFA。

9. 选择合适的方式添加多因素认证设备。

   - 扫码添加（推荐）：在移动设备端，单击扫码添加，扫描阿里云控制台绑定 MFA 步骤页面出现的二维码，单击确定。
   - 手动添加：在移动设备端，单击手动输入，填写用户名和密钥，单击确定。

     说明:
     用户名和密钥可以通过阿里云控制台获取，在安全设置页面下的绑定 MFA 步骤，鼠标悬停在扫描失败处，可以查看用户名和密钥。

10.在阿里云控制台，输入移动设备端显示的动态验证码，单击下一步，完成绑定。

说明:
移动设备端的阿里云应用会显示您当前账号的动态验证码，每 30 秒更新一次。

后续步骤

绑定多因素认证设备后，云账号再次登录阿里云时，系统将要求输入两层安全要素：

1. 第一安全要素：用户名和密码
2. 第二安全要素：多因素认证设备生成的验证码

说明:

- 为云账号绑定多因素认证设备后，不影响 RAM 用户的登录。
- 卸载 MFA 应用或删除绑定好的 MFA 前，请先前往阿里云停用 MFA，否则可能无法正常登录阿里云。

5.6.2 为云账号解绑多因素认证

如果您不再需要为云账号绑定多因素认证设备或需要更换多因素认证设备时，可以将绑定的多因素认证设备进行解绑。本文以阿里云应用为例介绍如何为云账号解绑多因素认证设备。

操作步骤

1. 登录阿里云控制台。
2. 将鼠标悬停在右上角头像的位置，单击安全设置。
3. 在安全设置页面下的虚拟 MFA 区域，单击解绑。

4. 在解绑 MFA 页面，选择合适的方式根据页面提示进行身份验证。
   - 通过拍摄脸部：单击立即验证，通过阿里云应用扫描控制台页面出现的二维码，按照提示完成解绑。

     说明:
     如果云账号没有进行实名认证，将没有人脸识别解绑多因素认证设备的选项。
   - 通过安全令牌：单击立即验证，输入阿里云应用生成的动态验证码，单击确定，完成解绑。
   - 通过联系客服：单击立即验证，输入相关申诉信息，按照提示完成解绑。

5.6.3 为 RAM 用户设置多因素认证

本文以阿里云应用为例介绍如何为RAM 用户开启多因素认证（Multi-factor authentication，MFA）。开启多因素认证后，可以为您的账号提供更高的安全保护。

操作步骤

1. 云账号登录 RAM 控制台。

   说明:
   - 如果云账号要求 RAM 用户开启多因素认证，那么 RAM 用户登录时会直接进入多因素认证绑定流程，请直接从第 5 步开始操作。
   - 如果云账号允许 RAM 用户自主管理多因素认证设备，RAM 用户也可以登录 RAM 控制台，单击安全管理，在 MFA 设备管理菜单下，单击启用 MFA 设备进行绑定。
2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在认证管理页签下，单击启用虚拟 MFA 设备。
5. 在移动设备端，下载并安装阿里云 应用。
   - iOS：在 App Store 中搜索阿里云。
   - Android：在应用市场中搜索阿里云。
6. 在移动设备端，登录阿里云应用。
7. 在屏幕底部单击控制台页签。
8. 在常用工具区域下，单击虚拟 MFA。

9. 选择合适的方式添加多因素认证设备。

   · 扫码添加（推荐）：在移动设备端，单击扫码添加，扫描 RAM 控制台扫码获取页签下的二维码，单击确定。

   · 手动添加：在移动设备端，单击手动输入，填写用户名和密钥，单击确定。

   > 说明:
   > 用户名和密钥可以通过 RAM 控制台获取，在启用虚拟 MFA 设备页面，单击手输信息获取，可以查看用户名和密钥。

10.在 RAM 控制台，输入移动设备端显示的两组连续的动态验证码，单击确定启用，完成绑定。

> 说明:
> 移动设备端的阿里云应用会显示您当前账号的动态验证码，每 30 秒更新一次。

后续步骤

绑定多因素认证设备后，RAM 用户再次登录阿里云时，系统将要求输入两层安全要素：

1. 第一安全要素：用户名和密码
2. 第二安全要素：多因素认证设备生成的两组连续的动态验证码

> 说明:
> 卸载 MFA 应用或删除绑定好的 MFA 前，请先前往阿里云停用 MFA，否则可能无法正常登录阿里云。

5.6.4 为 RAM 用户解绑多因素认证

如果您不再需要为 RAM 用户绑定多因素认证设备或需要更换多因素认证设备时，云账号可以为 RAM 用户解绑多因素认证设备。

操作步骤

1. 云账号登录 RAM 控制台。

   > 说明:
   > 如果云账号允许 RAM 用户自主管理多因素认证设备，RAM 用户也可以登录 RAM 控制台，单击安全管理，在 MFA 设备管理菜单下，单击停用 MFA 设备进行解绑。

2. 在左侧导航栏的人员管理菜单下，单击用户。
3. 在用户登录名称/显示名称列表下，单击目标 RAM 用户名称。
4. 在认证管理页签下，单击停用虚拟 MFA 设备。

5. 单击确认，完成解绑。

说明:

首次解绑需要输入手机验证码。

6 单点登录管理（SSO）

6.1 SSO 概述

阿里云支持基于 SAML 2.0 的 SSO（Single Sign On，单点登录），也称为身份联合登录。本文为您介绍企业如何使用自有的身份系统实现与阿里云的 SSO。

SSO 基本概念

阿里云提供基于 SAML 2.0 协议的 SSO。为了更好的理解 SSO，下面简要介绍与 SAML / SSO 相关的一些基本概念：

身份提供商（IdP）	一个包含有关外部身份提供商元数据的 RAM 实体，身份提供商可以提供身份管理服务。例如： · 企业本地 IdP： Microsoft Active Directory Federation Service （AD FS）、Shibboleth 等。 · Cloud IdP： Azure AD、Google G Suite、Okta、OneLogin 等。
服务提供商（SP）	利用 IdP 的身份管理功能，为用户提供具体服务的应用，SP 会使用 IdP 提供的用户信息。一些非 SAML 协议的身份系统（例如：OpenID Connect），也把服务提供商称作 IdP 的信赖方。
安全断言标记语言（SAML 2.0）	实现企业级用户身份认证的标准协议，它是 SP 和 IdP 之间实现沟通的技术实现方式之一。SAML 2.0 已经是目前实现企业级 SSO 的一种事实标准。
SAML 断言（SAML assertion）	SAML 协议中用来描述认证请求和认证响应的核心元素。例如：用户的具体属性就包含在认证响应的断言里。
信赖（Trust）	建立在 SP 和 IdP 之间的互信机制，通常由公钥和私钥来实现。SP 通过可信的方式获取 IdP 的 SAML 元数据，元数据中包含 IdP 签发 SAML 断言的签名验证公钥，SP 则使用公钥来验证断言的完整性。

SSO 的方式

企业根据自身需要，使用支持 SAML 2.0 的企业 IdP（例如：AD FS）与阿里云进行 SSO。阿里云提供以下两种基于 SAML 2.0 协议的 SSO 方式：

- 用户 SSO：阿里云通过 IdP 颁发的 SAML 断言确定企业用户与阿里云 RAM 用户的对应关系 。企业用户登录后，使用该 RAM 用户访问阿里云。详情请参考：#unique_106。
- 角色 SSO：阿里云通过 IdP 颁发的 SAML 断言确定企业用户在阿里云上可以使用的 RAM 角色。企业用户登录后，使用 SAML 断言中指定的 RAM 角色访问阿里云。详情请参考：#unique_107。

SSO 方式的比较

SSO 方式	SP 发起的 SSO	IdP 发起的 SSO	使用 RAM 用户账号和密码登录	一个 IdP 关联多个阿里云账号	多个 IdP
用户 SSO	√	√	×	×	×
角色 SSO	×	√	√	√	√

说明:

- “√”支持，“×”表示不支持。
- 关于用户 SSO 与角色 SSO 的更多比较，请参考：#unique_108。

6.2 SSO 方式的适用场景

阿里云目前支持两种 SSO 方式：角色 SSO 和用户 SSO。本文为您介绍这两种方式的适用场景和选择依据，帮助您根据整体业务需求选择合适的 SSO 方式。

角色 SSO

角色 SSO 适用于以下场景：

- 出于管理成本考虑，您不希望在云端创建和管理用户，从而避免用户同步带来的工作量。
- 您希望在使用 SSO 的同时，仍然保留一部分云上本地用户，可以在阿里云直接登录。云上本地用户的用途可以是新功能测试、网络或企业IdP出现问题时的备用登录方式等。
- 您希望根据用户在本地IdP中加入的组，或者用户的某个特殊属性，来区分他们在云上的权限。当进行权限调整时，只需要在本地进行分组或属性的更改。
- 您拥有多个阿里云账号，但使用统一的企业 IdP，希望在企业 IdP 配置一次，就可以实现到多个阿里云账号的 SSO。
- 您的各个分支机构存在多个 IdP，都需要访问同一个阿里云账号，您需要在一个阿里云账号内配置多个 IdP 进行 SSO。
- 除了控制台，您也希望使用程序访问的方式来进行 SSO。

用户 SSO

用户 SSO 适用于以下场景：

- 您希望从阿里云的登录页面开始发起登录，而非直接访问您 IdP 的登录页面。
- 您需要使用的云产品中有部分暂时不支持角色访问。支持角色访问（即通过 STS 访问）的云产品请参见 #unique_68。
- 您的 IdP 不支持复杂的自定义属性配置。
- 您没有上述需要使用角色 SSO 的业务需求，而又希望尽量简化 IdP 配置。

6.3 用户 SSO

6.3.1 用户 SSO 概述

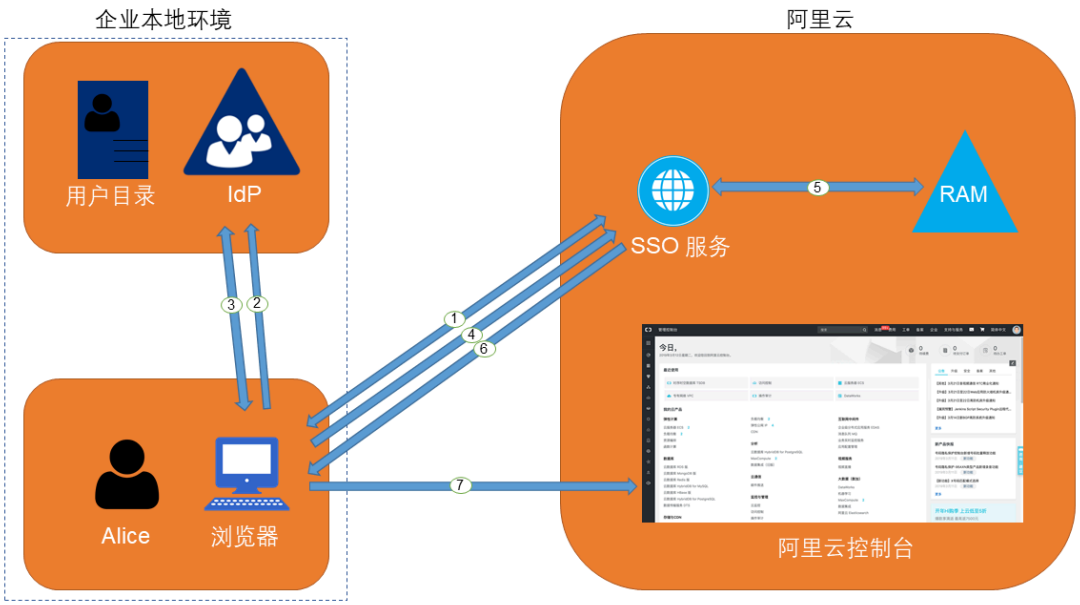
本文为您介绍用户 SSO 的背景、基本流程以及配置步骤。

背景信息

阿里云与企业进行用户 SSO 时，阿里云是服务提供商（SP），而企业自有的身份管理系统则是身份提供商（IdP）。通过用户 SSO，企业员工在登录后，将以 RAM 用户身份访问阿里云。

用户 SSO 基本流程

图 6-1: 基本流程



当管理员在完成用户 SSO 的相关配置后，企业员工 Alice 可以通过如图所示的方法登录到阿里云：

1. Alice 使用浏览器登录阿里云，阿里云将 SAML 认证请求返回给浏览器。
2. 浏览器向 IdP 转发 SAML 认证请求。
3. IdP 提示 Alice 登录，并在 Alice 登录成功后生成 SAML 响应返回给浏览器。
4. 浏览器将 SAML 响应转发给 SSO 服务。
5. SSO 服务通过 SAML 互信配置，验证 SAML 响应的数字签名来判断 SAML 断言的真伪，并通过 SAML 断言的`NameID`元素值，匹配到对应阿里云账号中的 RAM 用户身份。
6. SSO 服务向浏览器返回控制台的 URL。
7. 浏览器重定向到阿里云控制台。

> 说明:
> 在第 1 步中，企业员工从阿里云发起登录并不是必须的。企业员工也可以在企业自有 IdP 的登录页直接点击登录到阿里云的链接，向企业 IdP 发出登录到阿里云的 SAML 认证请求。

用户 SSO 的配置步骤

为了建立阿里云与企业 IdP 之间的互信关系，需要进行阿里云作为 SP 的 SAML 配置和企业 IdP 的 SAML 配置，配置完成后才能进行用户 SSO。

1. 为了建立阿里云对企业 IdP 的信任，需要将企业 IdP 配置到阿里云。

   详情请参考：云账号的 SAML配置。
2. 为了建立企业 IdP 对阿里云的信任，需要在企业 IdP 中配置阿里云为可信 SAML SP 并进行 SAML 断言属性的配置。

   详情请参考：#unique_113。
3. 企业 IdP 和阿里云均配置完成后，企业需要使用 SDK、CLI 或登录到 RAM 控制台创建与企业 IdP 匹配的 RAM 用户。

   详情请参考：创建 RAM 用户。

用户 SSO 示例

由于不同 IdP 的系统差异，关于 SAML SP 配置和断言属性配置的操作流程都有些差异。我们会提供一个以 AD FS（Microsoft Active Directory Federation Service）与阿里云进行用户 SSO 的示例，用于帮助理解企业 IdP 与阿里云的端到端配置流程。

详情请参考：#unique_114。

6.3.2 阿里云用户 SSO 的 SAML 配置

本文介绍通过基于 SAML 2.0 的用户 SSO，配置相应元数据来建立阿里云对企业身份提供商（IdP）的信任，实现企业 IdP 通过用户 SSO 登录阿里云。

前提条件

设置默认域名、域别名或辅助域名可以简化 SAML SSO 的配置流程。关于如何设置阿里云账号的默认域名或域别名，请参考：#unique_116和#unique_77。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 SSO 管理。
3. 在用户 SSO页签下，可查看当前 SSO 登录设置相关信息。
4. 单击编辑，可以配置 SSO 登录设置相关信息，包括选择 SSO 功能状态、上传元数据文档和设置辅助域名。
   - SSO 功能状态：可以选择开启或关闭。

     说明:
     该功能只对云账号下的所有 RAM 用户生效，不会影响云账号的登录。

     - 此功能默认为关闭，此时 RAM 用户可以使用密码登录，所有 SSO 设置不生效。
     - 如果选择开启此功能，此时 RAM 用户密码登录方式将会被关闭，统一跳转到企业 IdP 登录服务进行身份认证。如果再次关闭，用户密码登录方式自动恢复。
   - 元数据文档：单击上传文件，上传企业 IdP 提供的元数据文档。

     说明:
     元数据文档由企业 IdP 提供，一般为 XML 格式，包含 IdP 的登录服务地址以及 X.509 公钥证书（用于验证 IdP 所颁发的 SAML 断言的有效性）。
   - 辅助域名（可选）：开启辅助域名开关，可以设置一个辅助域名。
     - 如果设置了辅助域名，SAML 断言中的 `NameID` 元素将可以使用此辅助域名作为后缀。
     - 如果没有设置辅助域名，SAML 断言中的 `NameID` 元素将只能使用当前账号的默认域名或域别名作为后缀。

     关于 `NameID` 元素的取值，请参考：#unique_117。

     说明:

> 如果您同时设置了域别名和辅助域名，辅助域名将不会生效。此时，`NameID` 元素只能使用域别名或默认域名作为后缀。

后续步骤

完成 SAML 配置后，选择以下一种方法，将企业 IdP 中的用户数据迁移或同步到阿里云 RAM：

- 登录 RAM 控制台手动创建与企业 IdP 匹配的 RAM 用户。
- 使用 RAM SDK 编写程序或基于阿里云 CLI 来定制解决方案。

6.3.3 进行用户 SSO 时企业 IdP 的 SAML 配置

本文主要介绍企业在使用用户 SSO 时，如何在企业身份提供商（IdP）中配置阿里云为可信 SAML 服务提供商（SP）。

操作步骤

1. 从阿里云获取 SAML 服务提供商元数据 URL。
   a) 登录 RAM 控制台。
   b) 在左侧导航栏，单击 SSO 管理。
   c) 在用户 SSO 页签下的 SSO 登录设置区域，可以查看当前云账号的 SAML 服务提供商元数据 URL。
2. 在企业 IdP 中创建一个 SAML SP，并根据实际情况选择下面任意一种方式配置阿里云为信赖方：
   - 直接使用上述阿里云的元数据 URL 进行配置。
   - 如果您的 IdP 不支持 URL 配置，您可以根据上述 URL 下载元数据文件并上传至您的 IdP。
   - 如果您的 IdP 不支持元数据文件上传，则需要手动配置以下参数：
     - `Entity ID`：下载的元数据 XML 中，`md:EntityDescriptor` 元素的 `entityID` 属性值。
     - `ACS URL`：下载的元数据XML中，`md:AssertionConsumerService`元素的 `Location` 属性值。
     - `RelayState`（可选）：如果您的 IdP 支持设置`RelayState`参数，您可以将其配置成 SSO 登录成功后希望跳转到的页面 URL。如果不进行配置，SSO 登录成功后，将会跳转到阿里云控制台首页。

       > **说明：**
       > 您只能填写 `*.console.aliyun.com` 域名下的 URL 作为 `RelayState` 的值。

后续步骤

在企业 IdP 中配置阿里云为可信 SAML SP 后，需要在企业 IdP 中配置 SAML 断言属性。

阿里云需要通过 UPN （User Principal Name）来定位一个 RAM 用户，所以要求企业 IdP 生成的 SAML 断言包含用户的 UPN。阿里云通过解析 SAML 断言中的NameID元素，来匹配 RAM 用户的 UPN 从而实现用户 SSO。

因此，在配置 IdP 颁发的 SAML 断言时，需要将对应于 RAM 用户 UPN 的字段映射为 SAML 断言中的 NameID 元素。NameID 元素必须是以下几种：

- 使用域别名作为 NameID 元素的后缀，即 <username>@<domain_alias>。其中 <username> 为 RAM 用户的用户名，<domain_alias> 为域别名。关于如何设置域别名，请参考：#unique_77。
- 使用辅助域名作为 NameID 元素的后缀，即 <username>@<auxiliary_domain>。其中 <username> 为 RAM 用户的用户名，<auxiliary_alias> 为辅助域名。关于如何设置辅助域名，请参考：设置辅助域名。

  说明：

  如果您同时设置了域别名和辅助域名，辅助域名将不会生效。此时，NameID 元素只能使用域别名作为后缀。
- 使用默认域名作为 NameID 元素的后缀，即 <username>@<default_domain>。其中 <username> 为 RAM 用户的用户名，<default_domain> 为默认域名。关于如何设置默认域名，请参考：#unique_76。

  说明：

  即使设置了域别名或辅助域名，仍可以使用默认域名作为 NameID 的后缀。

示例：RAM 用户名为：Alice，默认域名为：example.onaliyun.com。

- 如果设置了域别名为：example.com，SAML 断言中的 NameID 取值为：Alice@example.onaliyun.com 或 Alice@example.com。
- 如果没有设置域别名，设置了辅助域名为：example2.com，SAML 断言中的 NameID 取值为：Alice@example.onaliyun.com 或 Alice@example2.com。
- 如果设置了域别名为：example.com后，又设置了辅助域名为：example2.com，SAML 断言中的 NameID 取值为：Alice@example.onaliyun.com 或 Alice@example.com。

6.3.4 使用 AD FS 进行用户 SSO 的示例

本文提供一个以 AD FS 与阿里云进行 SSO 的示例，帮助您理解企业 IdP 与阿里云进行 SSO 的端到端配置流程。

注意事项

本文以 Windows Server 2012 R2 为例，介绍如何配置 AD FS 与阿里云，从而实现 SSO。

> **注意:**
> 本文中涉及到 Microsoft Active Directory 配置的部分属于建议，仅用于帮助理解阿里云 SSO 的端到端配置，阿里云不提供 Microsoft Active Directory 配置官方咨询服务。

前提条件

用户对 Microsoft Active Directory (AD) 需进行合理正确的配置，在 Windows Server 2012 R2 上配置以下服务器角色：

- DNS 服务器：将身份认证请求解析到正确的 Federation Service 上。
- Active Directory 域服务 （AD DS）：提供对域用户和域设备等对象的创建、查询和修改等功能。
- Active Directory Federation Service （AD FS）：提供配置 SSO 信赖方的功能，并对配置好的信赖方提供 SSO 认证。

示例配置

示例中用到的相关配置如下：

- 云账号的默认域名为：`secloud.onaliyun.com`。
- 云账号下包含 RAM 用户：`alice`，其完整的 UPN（User Principal Name）为：`alice@secloud.onaliyun.com`。
- 创建的 Microsoft AD 中的 AD FS 服务名称为：`adfs.secloud.club`。
- 创建的 Microsoft AD 的域名为：`secloud.club`，NETBIOS 名为：`secloud`。
- RAM 用户 `alice` 在 AD 中的 UPN 为：`alice@secloud.club`，域内登录也可以使用：`secloud\alice`。

在 RAM 中将 AD FS 配置为可信 SAML IdP

1. 在浏览器中输入如下地址：

```
https://adfs.secloud.club/FederationMetadata/2007-06/FederationMetadata.xml
```

2. 将元数据 XML 文件下载到本地。

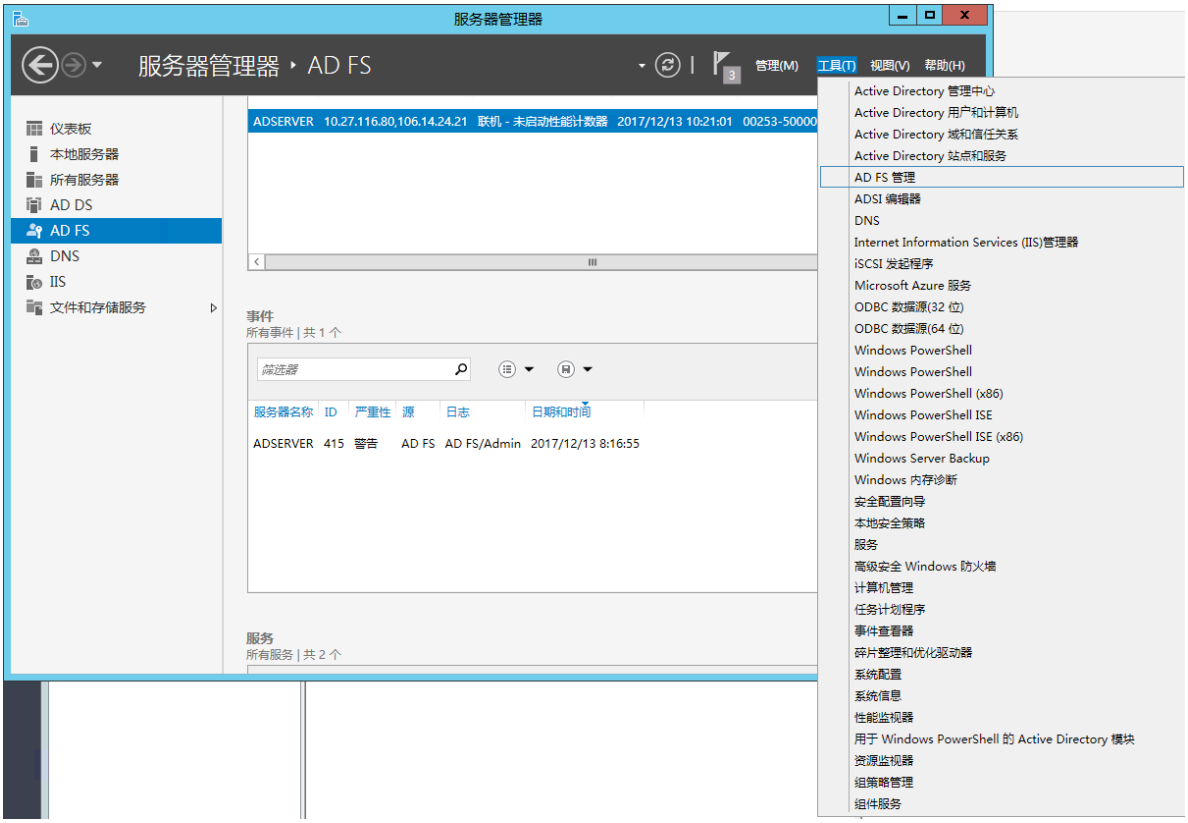
3. 在 RAM 控制台的 SSO 配置时使用下载好的元数据文件。

具体配置请参考：#unique_119。

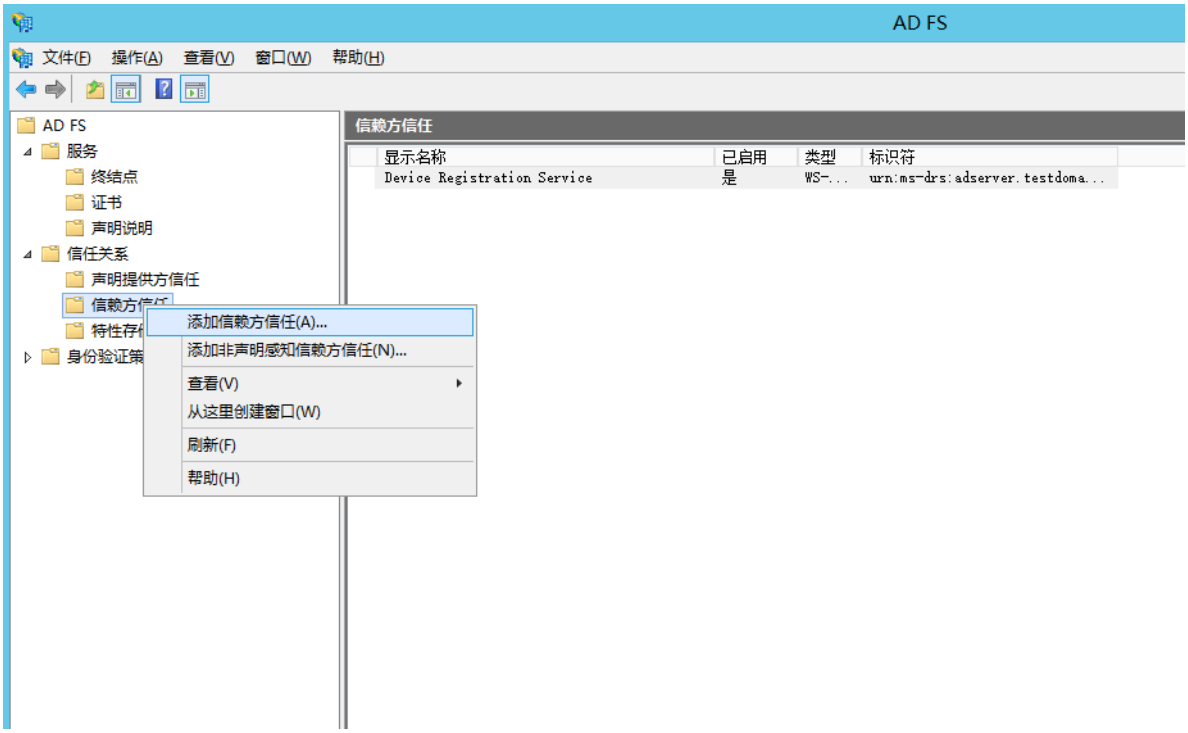
在 AD FS 中将阿里云配置为可信 SAML SP

在 AD FS 中，SAML SP 被称作 信赖方。

1. 在服务器管理器的工具菜单中选择AD FS 管理 。

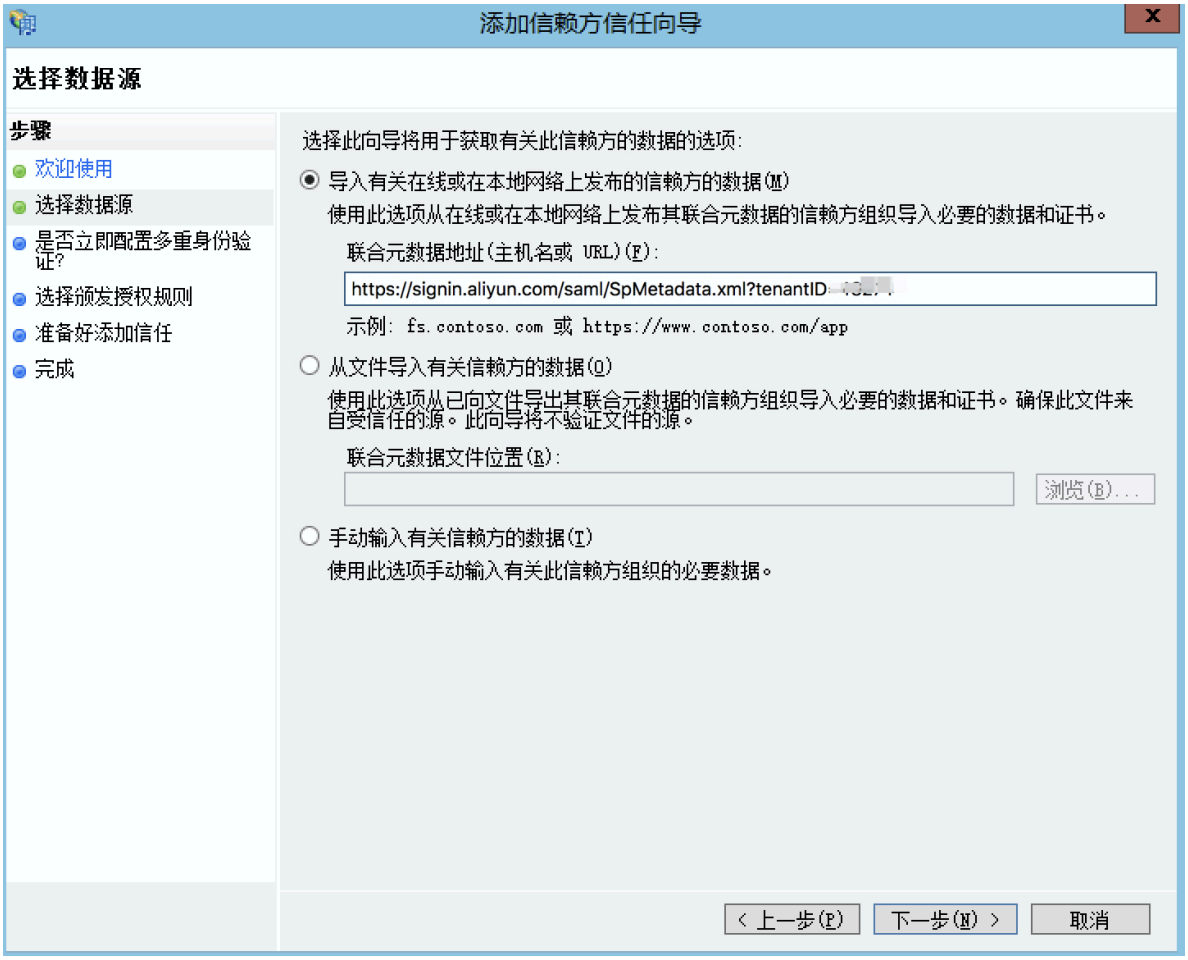


2. 在 AD FS 管理工具中添加信赖方信任。



3. 为新创建的信赖方设置阿里云的 SAML 元数据。

   阿里云账号的 SAML 服务提供商元数据 URL 可以登录 RAM 控制台，在左侧菜单栏，单击 SSO 管理，在 用户 SSO页签下的 SSO 登录设置区域下查看。AD FS 信赖方可以直接配置元数据的 URL。



   完成配置信赖方之后，阿里云和 AD FS 就产生了互信，阿里云会将默认域名为`secloud.onaliyun.com`的云账号下所有 RAM 用户的认证请求转发到 AD FS：`adfs.secloud.club`，AD FS 也会接受来自于阿里云的认证请求并向阿里云转发认证响应。

为阿里云 SP 配置 SAML 断言属性

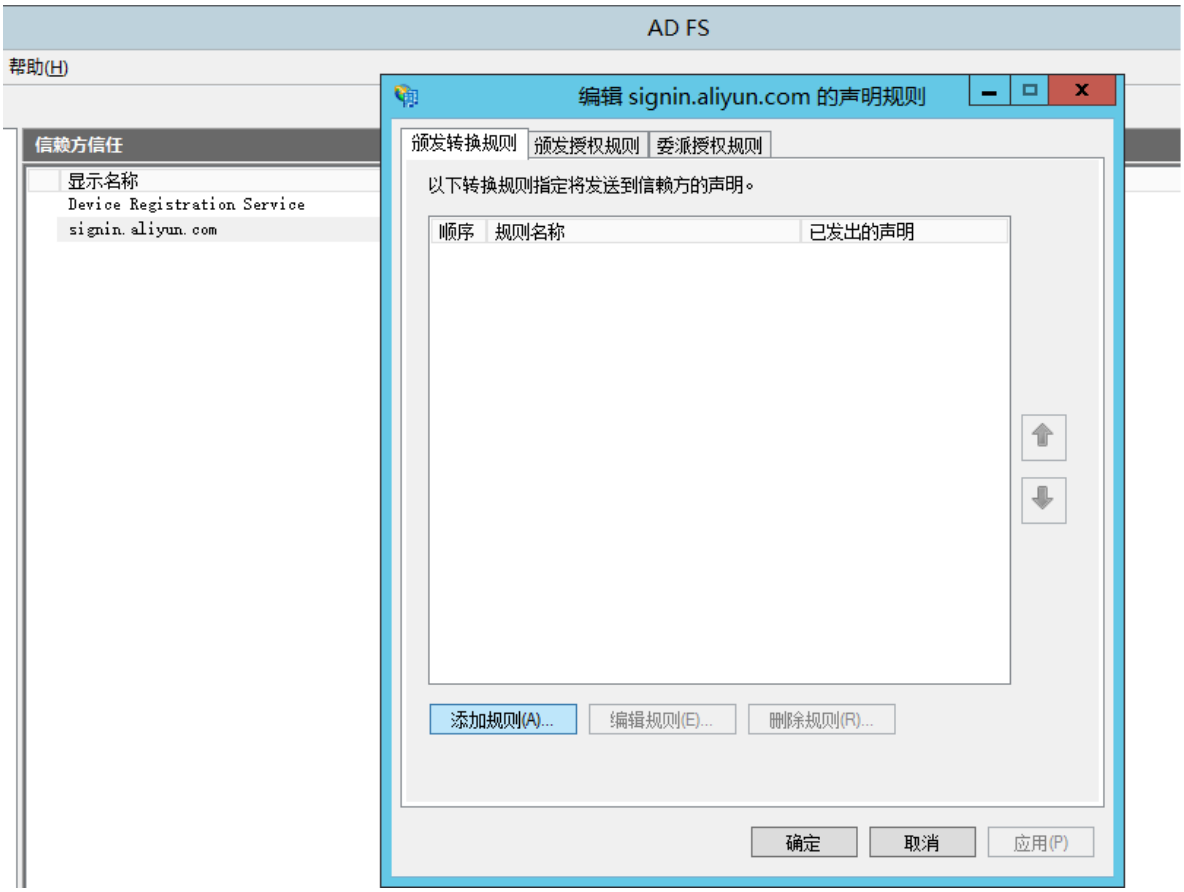
为了让阿里云能使用 SAML 响应定位到正确的 RAM 用户，SAML 断言中的 `NameID` 字段取值应为 RAM 用户的 UPN。

我们需要配置 AD 中的 UPN 为 SAML 断言中的 `NameID`。

1. 为信赖方编辑声明规则。

2. 添加颁发转换规则。

> **说明:**
>
> 颁发转换规则（Issuance Transform Rules）：指如何将一个已知的用户属性，经过转换后颁发为 SAML 断言中的属性。由于我们要将用户在 AD 中的 UPN 颁发为`NameID`，因此需要添加一个新的规则。



3. 声明规则模版选择转换传入声明。

4. 编辑规则。

说明:

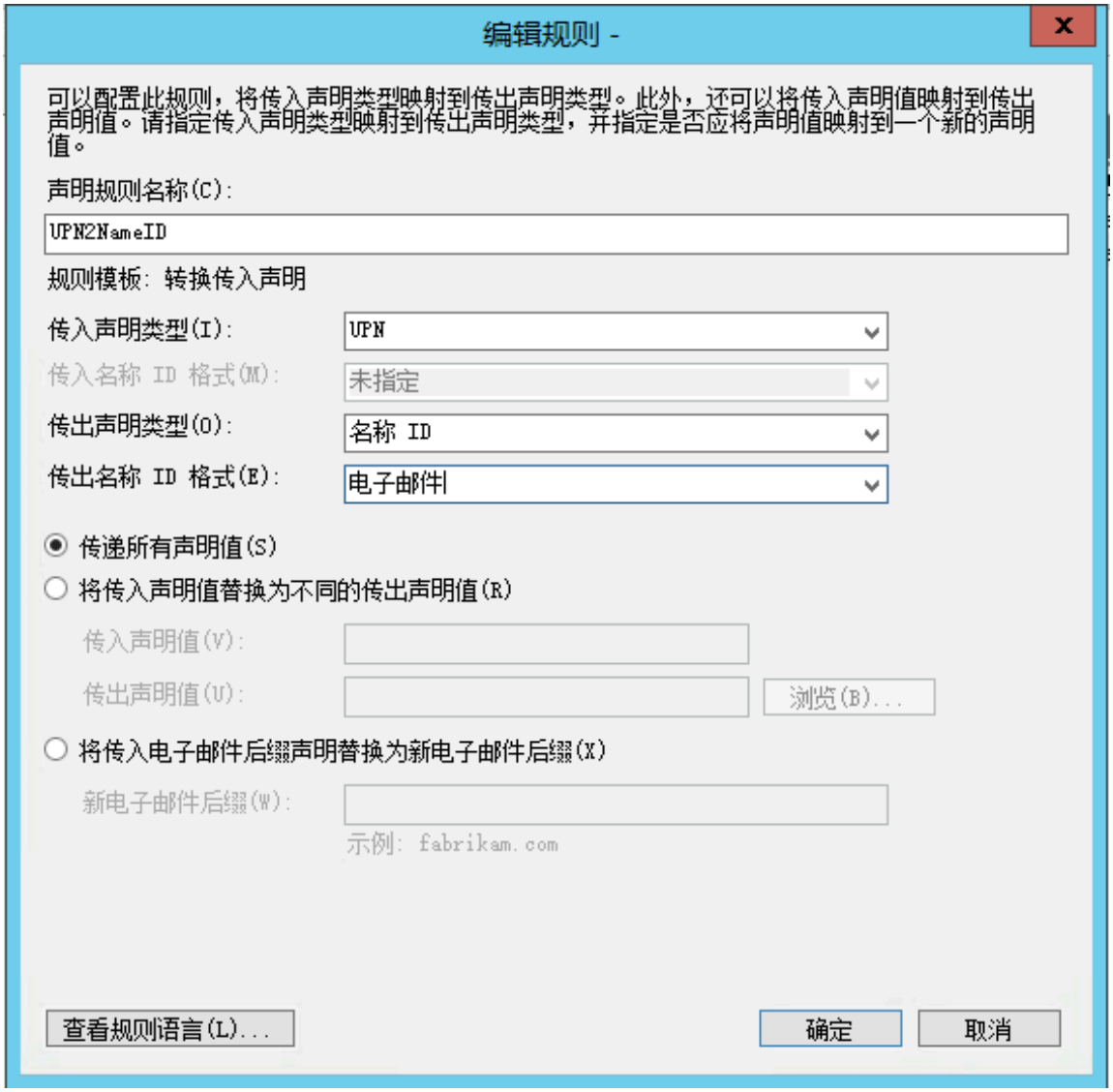
由于示例中的云账号里的 UPN 域名为secloud.onaliyun.com，而 AD 中的 UPN 域名为secloud.club，如果直接将 AD 中的 UPN 映射为NameID，阿里云将无法匹配到正确的 RAM 用户。

下面提供几种设置 RAM 用户的 UPN 与 AD 用户的 UPN 保持一致的方法：

a. 方法一：将 AD 域名设置为 RAM 的域别名。

如果 AD 域名secloud.club是一个在公网 DNS 中注册的域名。用户可以将secloud.club设置为 RAM 的域别名。关于如何设置域别名，请参考：#unique_77。

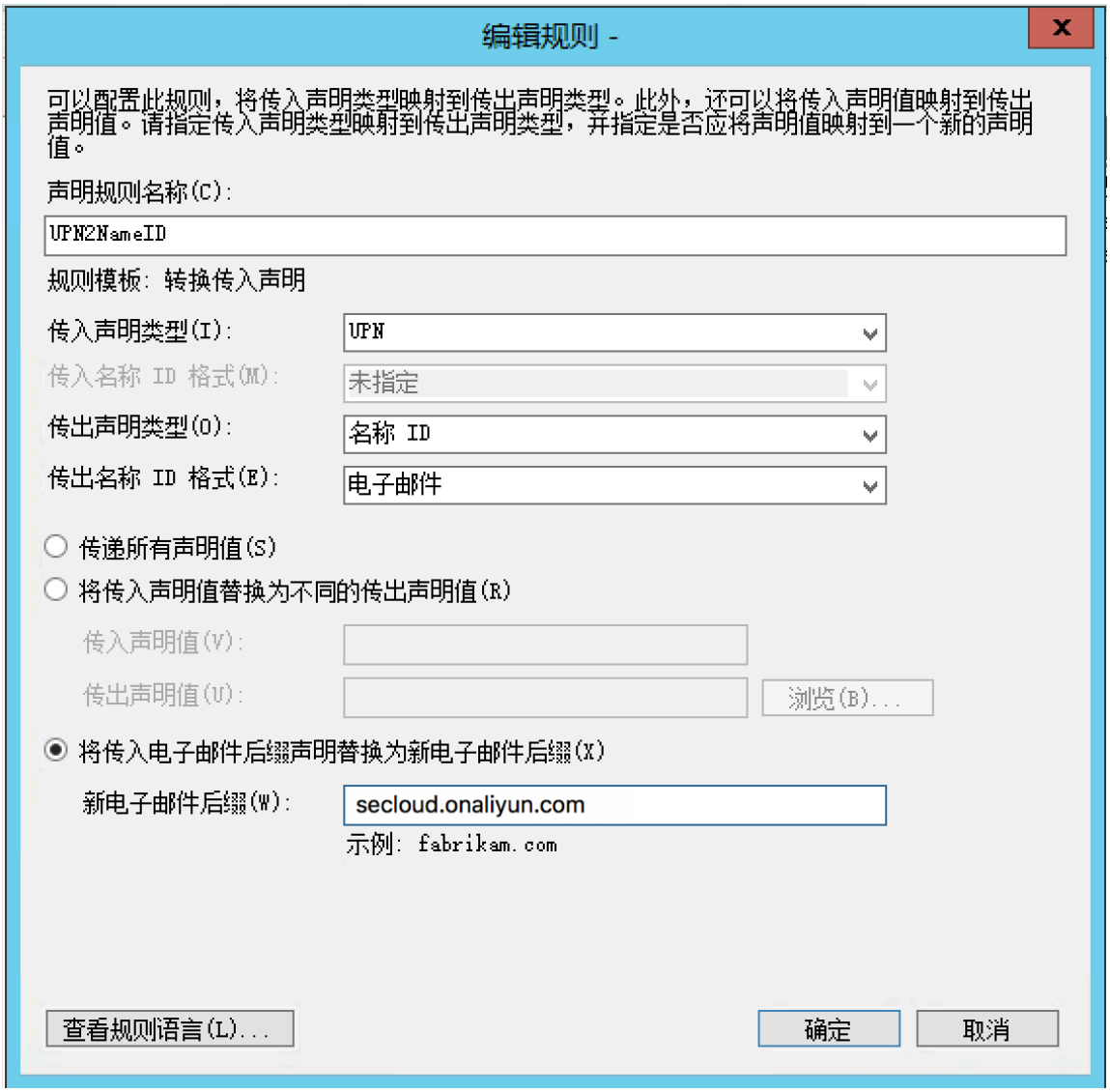
完成设置后，在编辑规则窗口，将 UPN 映射为名称 ID（NameID）。



b. 方法二：在 AD FS 中设置域名转换。

如果域名secloud.club是企业的内网域名，那么阿里云将无法验证企业对域名的所有权。RAM 就只能使用默认域名secloud.onaliyun.com。

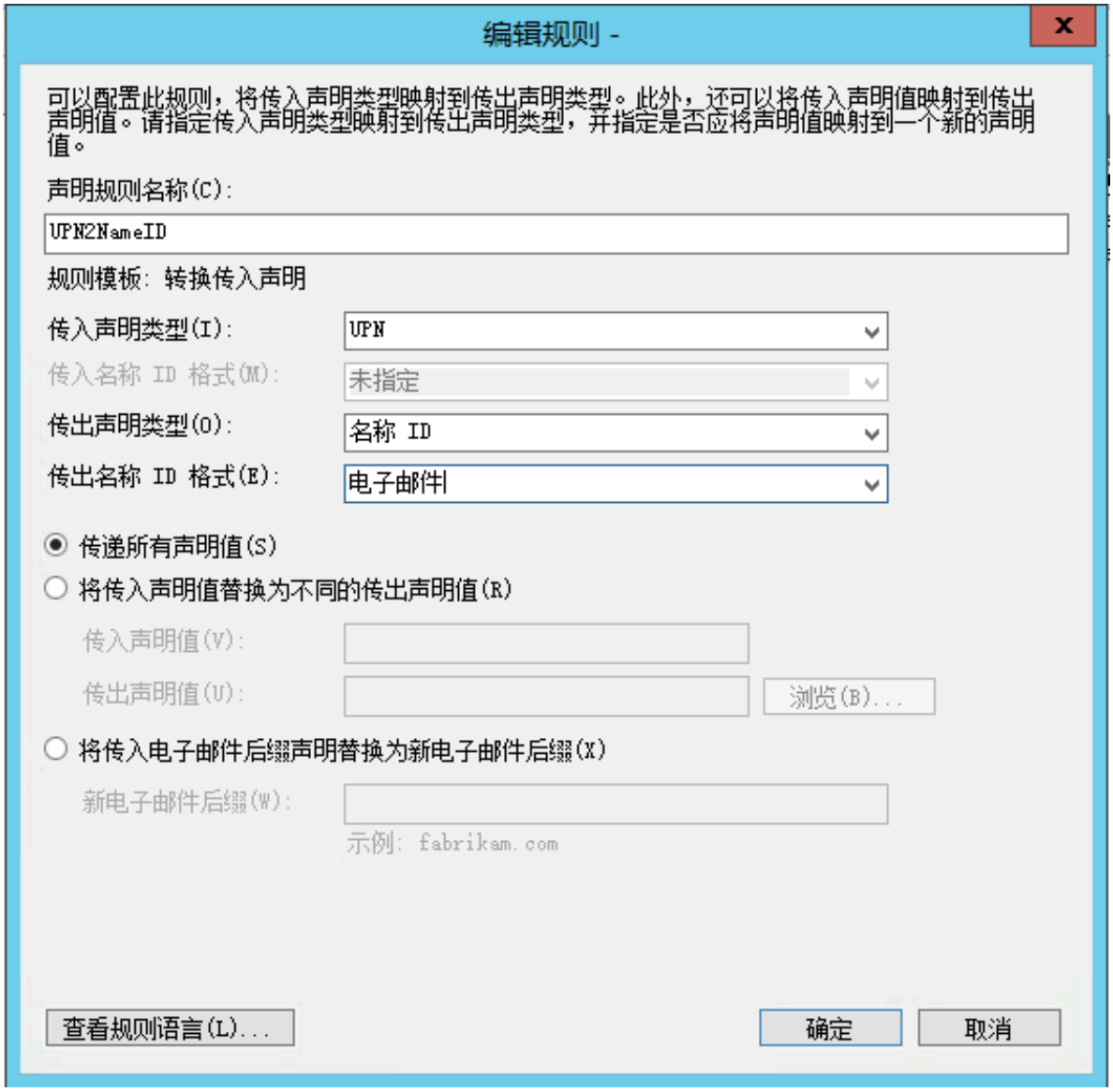
在 AD FS 给阿里云颁发的 SAML 断言中必须将 UPN 的域名后缀从 `secloud.club` 替换为：`secloud.onaliyun.com`。



c. 方法三：将 AD 域名设置为用户 SSO 的辅助域名。

如果域名`secloud.club`是企业的内网域名，那么阿里云将无法验证企业对域名的所有权。您可以将 `secloud.club` 设置为用户 SSO 的辅助域名，无需进行域名转换。关于如何设置辅助域名，请参考：设置辅助域名。

完成设置后，在编辑规则窗口，将 UPN 映射为名称 ID（`NameID`）。



6.4 身份提供商

6.4.1 创建身份提供商

在使用角色 SSO 时，需要创建身份提供商。

操作步骤

1. **登录 RAM 控制台。**
2. **在左侧导航栏，单击 SSO 管理。**
3. **在角色 SSO页签下，单击新建身份提供商。**
4. **输入提供商名称和备注。**

5. 在元数据文档处，单击上传文件。

> 说明:
> 元数据文档由企业 IdP 提供，一般为 XML 格式，包含 IdP 的登录服务地址、用于验证签名的公钥及断言格式等信息。

6. 单击确定。

6.4.2 查看身份提供商基本信息

本文为您介绍如何查看身份提供商基本信息，包括身份提供商名称、ARN 和备注等信息。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 SSO 管理。
3. 在角色 SSO页签下，单击目标身份提供商名称。
4. 在身份提供商信息页面下，可以查看身份提供商基本信息。

6.4.3 修改身份提供商基本信息

本文为您介绍如何修改身份提供商基本信息，包括备注和元数据文档。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 SSO 管理。
3. 在角色 SSO页签下，单击目标身份提供商名称。
4. 在身份提供商信息页面下，单击编辑。

> 说明:
> 身份提供商名称不允许修改。修改身份提供商名称会导致与其受信实体信息不一致，无法正常进行单点登录（SSO）。

5. 修改完成后，单击确认。

6.4.4 删除身份提供商

如果不再需要身份提供商，可以将其删除。删除身份提供商后，企业将无法与阿里云 RAM 进行 SSO。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 SSO 管理。

3. 在角色 SSO页签下，找到目标身份提供商，单击删除。

4. 单击确认。

6.5 角色 SSO

6.5.1 角色 SSO 概述

本文为您介绍角色 SSO 的背景、基本流程以及配置步骤。

背景信息

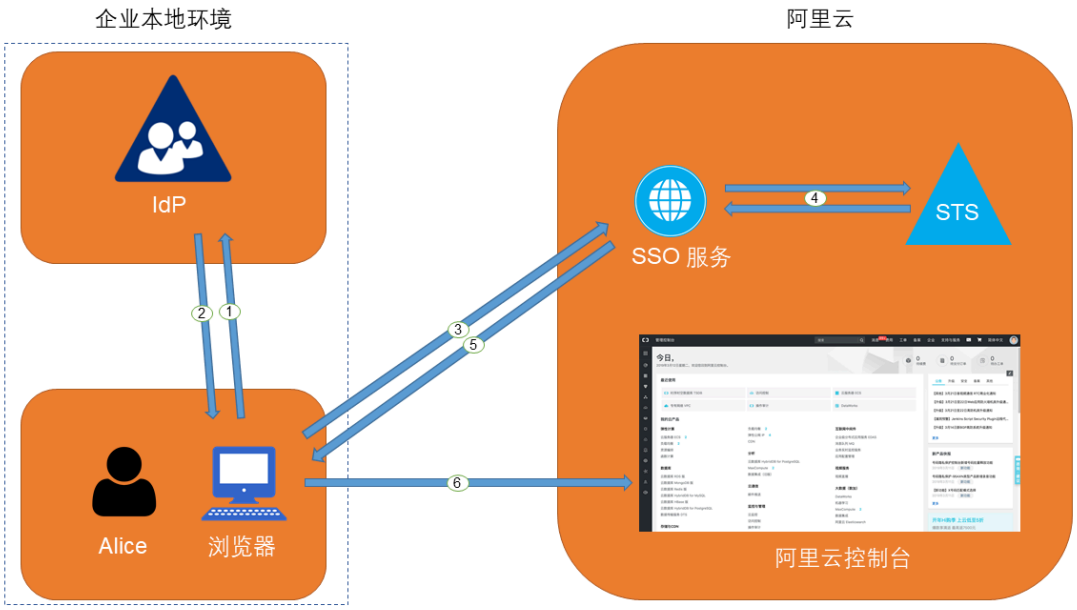
阿里云与企业进行角色 SSO 时，阿里云是服务提供商（SP），而企业自有的身份管理系统则是身份提供商（IdP）。通过角色 SSO，企业可以在本地 IdP 中管理员工信息，无需进行阿里云和企业 IdP 间的用户同步，企业员工将使用指定的 RAM 角色来登录阿里云。

角色 SSO 流程

通过角色 SSO，企业员工既可以通过控制台也可以使用程序访问阿里云。

通过控制台访问阿里云

图 6-2: 基本流程



当管理员在完成角色 SSO 的相关配置后，企业员工 Alice 可以通过如图所示的方法登录到阿里云：

1. Alice 使用浏览器在 IdP 的登录页面中选择阿里云作为目标服务。

   例如：如果企业 IdP 使用 AD FS (Microsoft Active Directory Federation Service)，则登录 URL 为：`https://ADFSServiceName/adfs/ls/IdpInitiatedSignOn.aspx`。

   > **说明:**
   > 有些 IdP 会要求用户先登录，再选择代表阿里云的 SSO 应用。

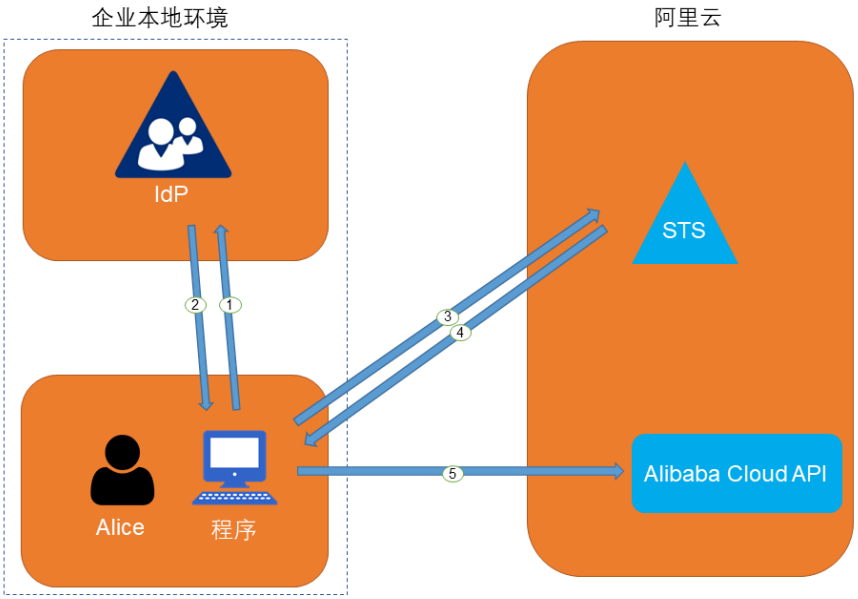
2. IdP 生成一个 SAML 响应并返回给浏览器。
3. 浏览器重定向到 SSO 服务页面，并转发 SAML 响应给 SSO 服务。
4. SSO 服务使用 SAML 响应向阿里云 STS 服务请求临时安全凭证，并生成一个可以使用临时安全凭证登录阿里云控制台的 URL。

   > **说明:**
   > 如果 SAML 响应中包含映射到多个 RAM 角色的属性，系统将会首先提示用户选择一个用于访问阿里云的角色。

5. SSO 服务将 URL 返回给浏览器。
6. 浏览器重定向到该 URL，以指定角色身份登录到阿里云控制台。

使用程序访问阿里云

图 6-3: 基本流程



企业员工 Alice 可以通过编写程序来访问阿里云，基本流程如图所示：

1. Alice 使用程序向企业 IdP 发起登录请求。
2. IdP 生成一个 SAML 响应，其中包含关于登录用户的 SAML 断言，并将此响应返回给程序。
3. 程序调用阿里云 STS 服务提供的 API #unique_128，并传递以下信息：阿里云中身份提供商的 ARN、要扮演的角色的 ARN 以及来自企业 IdP 的 SAML 断言。
4. STS 服务将校验 SAML 断言并返回临时安全凭证给程序。
5. 程序可以开始使用临时安全凭证来调用阿里云 API。

角色 SSO 的配置步骤

为了建立阿里云与企业 IdP 之间的互信关系，需要进行阿里云作为 SP 的 SAML 配置和企业 IdP 的 SAML 配置，配置完成后才能进行角色 SSO。

1. 为了建立阿里云对企业 IdP 的信任，需要将企业 IdP 配置到阿里云。

   详情请参考：阿里云角色 SSO 的 SAML 配置。
2. 企业需要使用程序或登录 RAM 控制台创建用于 SSO 的角色，并授予相关权限。

   详情请参考：创建可信实体为身份提供商的 RAM 角色。
3. 为了建立企业 IdP 对阿里云的信任，需要在企业 IdP 中配置阿里云为可信 SAML SP 并进行 SAML 断言属性的配置。

   详情请参考：#unique_130。

角色 SSO 配置示例

由于不同 IdP 的系统差异，关于 SAML SP 配置和断言属性配置的操作流程都有些差异。我们会提供一个以 AD FS 与阿里云进行角色 SSO 的示例，用于帮助理解企业 IdP 与阿里云的端到端配置流程。

详情请参考：#unique_131。

6.5.2 阿里云角色 SSO 的 SAML 配置

本文介绍如何通过基于 SAML 2.0 的角色 SSO，配置相应元数据来建立阿里云对企业身份提供商（IdP） 的信任，实现企业 IdP 通过角色 SSO 登录阿里云。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 SSO 管理。
3. 在角色 SSO页签下，单击新建身份提供商。
4. 输入提供商名称和备注。

5. 在元数据文档处，单击上传文件。

> 说明:
> 元数据文档由企业 IdP 提供，一般为 XML 格式，包含 IdP 的登录服务地址、用于验证签名的公钥及断言格式等信息。

6. 单击确定。

后续步骤

创建身份提供商后，必须创建一个或多个 RAM 角色，该 RAM 角色的可信实体类型为身份提供商，从而建立企业 IdP 与阿里云的关联。

单击前往新建 RAM 角色可直接跳转到创建 RAM 角色的界面。关于 RAM 角色的创建，请参考：创建可信实体为身份提供商的 RAM 角色。

6.5.3 进行角色 SSO 时企业 IdP 的 SAML 配置

本文主要介绍企业在使用角色 SSO 时，如何在企业身份提供商（IdP）中配置阿里云为可信 SAML 服务提供商（SP）。

操作步骤

1. 企业 IdP 的 SAML SP 配置需要使用阿里云的 SAML 服务提供商元数据 URL：`https://signin.aliyun.com/saml-role/sp-metadata.xml`。
   a) 登录 RAM 控制台。
   b) 在左侧导航栏，单击 SSO 管理。
   c) 在角色 SSO 页签下查看 SAML 服务提供商元数据 URL。

2. 在企业 IdP 中创建一个 SAML SP，并根据实际情况选择下面任意一种方式配置阿里云为信赖方：

   - 直接使用上述阿里云的元数据 URL 进行配置。
   - 如果您的 IdP 不支持 URL 配置，您可以根据上述 URL 下载元数据文件并上传至您的 IdP。
   - 如果您的 IdP 不支持元数据文件上传，则需要手动配置以下参数：
     - `Entity ID`：`urn:alibaba:cloudcomputing`
     - `ACS URL`：`https://signin.aliyun.com/saml-role/sso`
     - `RelayState`（可选）：如果您的 IdP 支持设置`RelayState`参数，您可以将其配置成 SSO 登录成功后希望跳转到的页面 URL。如果不进行配置，SSO 登录成功后，将会跳转到阿里云控制台首页。

       说明：

       您只能填写`*.console.aliyun.com`域名下的 URL 作为`RelayState`的值。

后续步骤

在企业 IdP 中配置阿里云为可信 SAML SP 后，需要在企业 IdP 中配置 SAML 断言属性。

阿里云要求企业 IdP 生成的 SAML 断言里包含一些必要的信息以确定企业用户的登录身份，因此企业 IdP 必须进行属性配置来匹配 RAM 角色，从而实现企业用户与阿里云的 SSO。

具体需要配置的 SAML 断言属性请参考：c72e1357747c8019b1a5aa7a4f4e46c376dfa9df.dita。

6.5.4 支持角色 SSO 的 SAML 断言

本文介绍在进行角色 SSO 时，您的 IdP 颁发的 SAML 断言必须具备的属性元素。

背景信息

在基于 SAML 2.0 的 SSO 流程中，当企业用户在 IdP 登录后，IdP 将根据 SAML 2.0 HTTP-POST 绑定的要求生成包含 SAML 断言的认证响应，并由浏览器（或程序）自动转发给阿里云。

这个 SAML 断言会被用来确认用户登录状态并从中解析出登录的主体。因此，断言中必须包含阿里云要求的元素，否则登录用户的身份将无法被确认，导致 SSO 失败。

SAML 2.0 协议的通用元素

- `Issuer`

  `Issuer`的值必须与您在阿里云创建的身份提供商实体中上传的 IdP 元数据文件中的`EntityID`匹配。

- Signature

  阿里云要求 SAML 断言必须被签名以确保没有篡改，Signature及其包含的元素必须包含签名值、签名算法等信息。

- Subject

  Subject必须包含以下元素：

  - 有且仅有一个NameID元素。您必须按照 SAML 2.0 协议的定义来给出NameID的值，但阿里云不会依赖该元素的值来确认登录主体。
  - 有且仅有一个SubjectConfirmation元素，其中包含一个SubjectConfirmationData元素。SubjectConfirmationData必须有如下两个属性：
    - NotOnOrAfter：规定 SAML 断言的有效期。
    - Recipient：阿里云通过检查该元素的值来确保阿里云是该断言的目标接收方，其取值必须为 https://signin.aliyun.com/saml-role/sso。

  如下是一个Subject元素的示例：

```
<Subject>
  <NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-format:
persistent">administrator</NameID>
  <SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:
bearer">
    <SubjectConfirmationData NotOnOrAfter="2019-01-01T00:01:00.
000Z" Recipient="https://signin.aliyun.com/saml-role/sso"/>
  </SubjectConfirmation>
</Subject>
```

- Conditions

  在Conditions元素中，必须包含一个AudienceRestriction元素，其中可包含一至多个Audience元素，但必须有一个Audience元素的取值为 urn:alibaba:cloudcomputing。

  如下是一个Conditions元素的示例：

```
<Conditions>
  <AudienceRestriction>
    <Audience>urn:alibaba:cloudcomputing</Audience>
  </AudienceRestriction>
</Conditions>
```

阿里云要求的自定义属性

在 SAML 断言的AttributeStatement元素中，必须包含如下阿里云要求的Attribute元素：

- Name属性值为https://www.aliyun.com/SAML-Role/Attributes/Role的Attribute元素

  该元素为必选，可以有多个。其包含的AttributeValue元素取值代表允许当前用户扮演的角色，取值的格式是由角色 ARN 与身份提供商 ARN 组合而成的，中间用英文逗号（,）隔开。这两个 ARN 您可以在控制台获取：

  - 角色 ARN：在 RAM 角色管理页面，单击 RAM 角色名称后，基本信息页面可以查看对应的 ARN。
  - 身份提供商 ARN：在 SSO 管理页面的角色 SSO 页签下，单击身份提供商名称后，身份提供商信息页面可以查看对应的 ARN。

  如果是多个，则当使用控制台登录时，将会在界面上列出所有角色供用户选择。

  如下是一个 Role Attribute 元素示例：

```
<Attribute Name="https://www.aliyun.com/SAML-Role/Attributes/Role">

  <AttributeValue>acs:ram::$account_id:role/role1,acs:ram::$
account_id:saml-provider/provider1</AttributeValue>
  <AttributeValue>acs:ram::$account_id:role/role2,acs:ram::$
account_id:saml-provider/provider1</AttributeValue>
</Attribute>
```

  说明：

  $account_id是定义角色和身份提供商的阿里云账号ID。

- Name属性值为https://www.aliyun.com/SAML-Role/Attributes/RoleSessionName的Attribute元素

  该元素为必选且只能有一个。其包含的AttributeValue元素取值将被用来作为登录用户信息的一部分显示在控制台上和操作审计日志中。如果您有多个用户使用同一个角色，请确保使用可以唯一标识用户的RoleSessionName值，以区分不同的用户，如员工 ID、email 地址等。

  其 AttributeValue元素取值要求：长度不少于2个字符且不超过32个字符，只能是英文字母、数字和以下特殊字符：-_.@=,+。

  如下是一个 RoleSessionName Attribute 元素示例：

```
<Attribute Name="https://www.aliyun.com/SAML-Role/Attributes/
RoleSessionName">
  <AttributeValue>user_id</AttributeValue>
</Attribute>
```

- Name属性值为https://www.aliyun.com/SAML-Role/Attributes/SessionDuration的Attribute元素

  该元素为可选，且最多只能有一个。在通过控制台登录的情况下，其包含的AttributeValue元素取值将会被作为用户会话的有效时长。在通过程序登录的情况下，其包含的AttributeValue元素取值无效。

  其AttributeValue元素取值要求：整数，单位为秒，最小900秒（15分钟），最大3600秒（1小时）。若此元素不存在，则取默认值3600秒。

  如下是一个 SessionDuration Attribute 元素示例：

```
<Attribute Name="https://www.aliyun.com/SAML-Role/Attributes/
SessionDuration">
  <AttributeValue>1800</AttributeValue>
</Attribute>
```

6.5.5 使用 AD FS 进行角色 SSO 的示例

本文提供一个以 AD FS 与阿里云进行 SSO 的示例，帮助用户理解企业 IdP 与阿里云进行 SSO 的端到端配置流程。

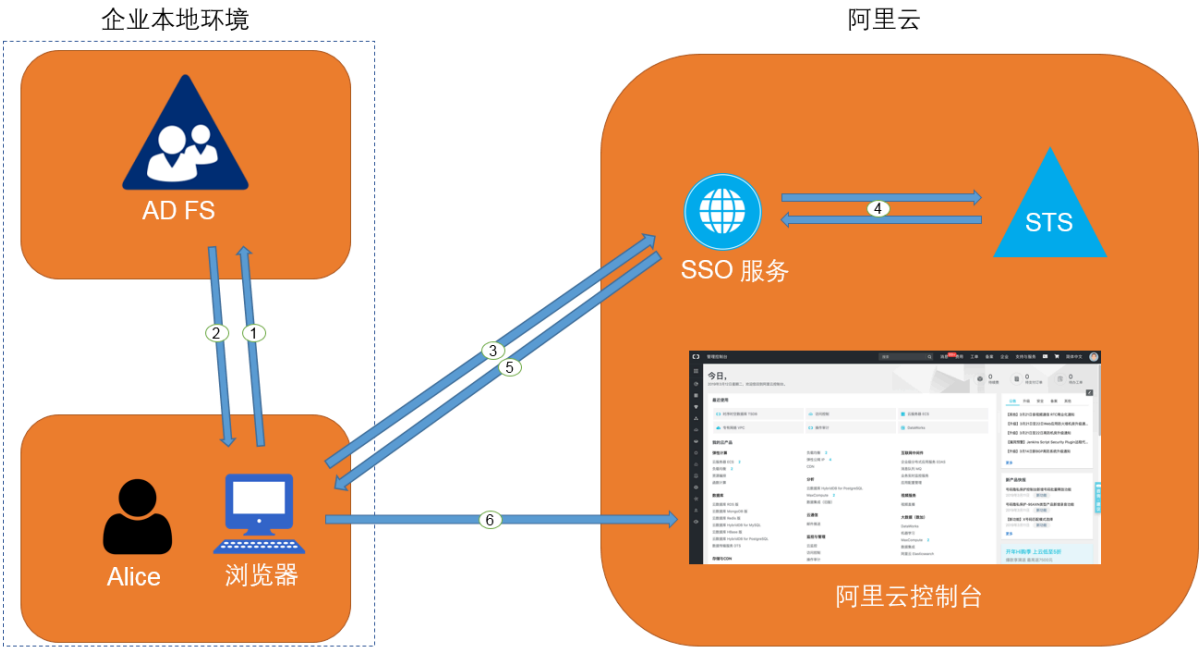
场景

企业使用 Active Directory （AD）进行员工管理，并通过 AD FS 配置包括阿里云在内的企业应用。AD 管理员通过员工的用户组来管理员工对阿里云账号的访问权限。在本例中，企业拥有两个阿里云账号（Account1 和 Account2），要管理的权限为 Admin 和 Reader，企业员工用户名为 Alice，该用户所在的 AD 用户组为 Aliyun-<account-id>-ADFS-Admin 和 Aliyun-<account-id>-ADFS-Reader，企业想要实现从 AD FS 到 Account1 和 Account2 的 SSO。

说明:

<account-id>为云账号 Account1 或 Account2 的账号 ID，因此用户 Alice 所在的 AD 用户组共4个，分别对应两个云账号中的 Admin 和 Reader 权限。

员工进行控制台登录的基本流程如下图所示：



AD 管理员在完成角色联合登录的配置后，企业员工（Alice）可以通过如图所示的方法登录到阿里云控制台。详情请参见#unique_135。

上述过程表示，用户登录时，企业会进行统一登录认证，无需用户提供在阿里云上的任何用户名和密码。

配置步骤

为了实现上述登录过程，管理员需要在阿里云和 AD FS 上进行以下配置：

- 在阿里云中将 AD FS 配置为可信 SAML IdP。
  1. 在阿里云 RAM 控制台创建一个名为`ADFS`的身份提供商，并配置相应的元数据。AD FS 的元数据 URL 为：`https://<ADFS-server>/federationmetadata/2007-06/federationmetadata.xml`。

     > **说明:**
     > <ADFS-server>是您的 AD FS 服务器域名或 IP 地址。

     详情请参见#unique_136。
  2. 在阿里云账号 Account1 中创建两个可信实体类型为身份提供商的 RAM 角色（ADFS-Admin 和 ADFS-Reader），选择刚刚创建的`ADFS`作为可信身份提供商，并对两个角色分别赋予`AdministratorAccess`和`ReadOnlyAccess`权限。详情请参见#unique_137。
  3. 使用同样的方法在 Account2 中创建同样的身份提供商和角色。

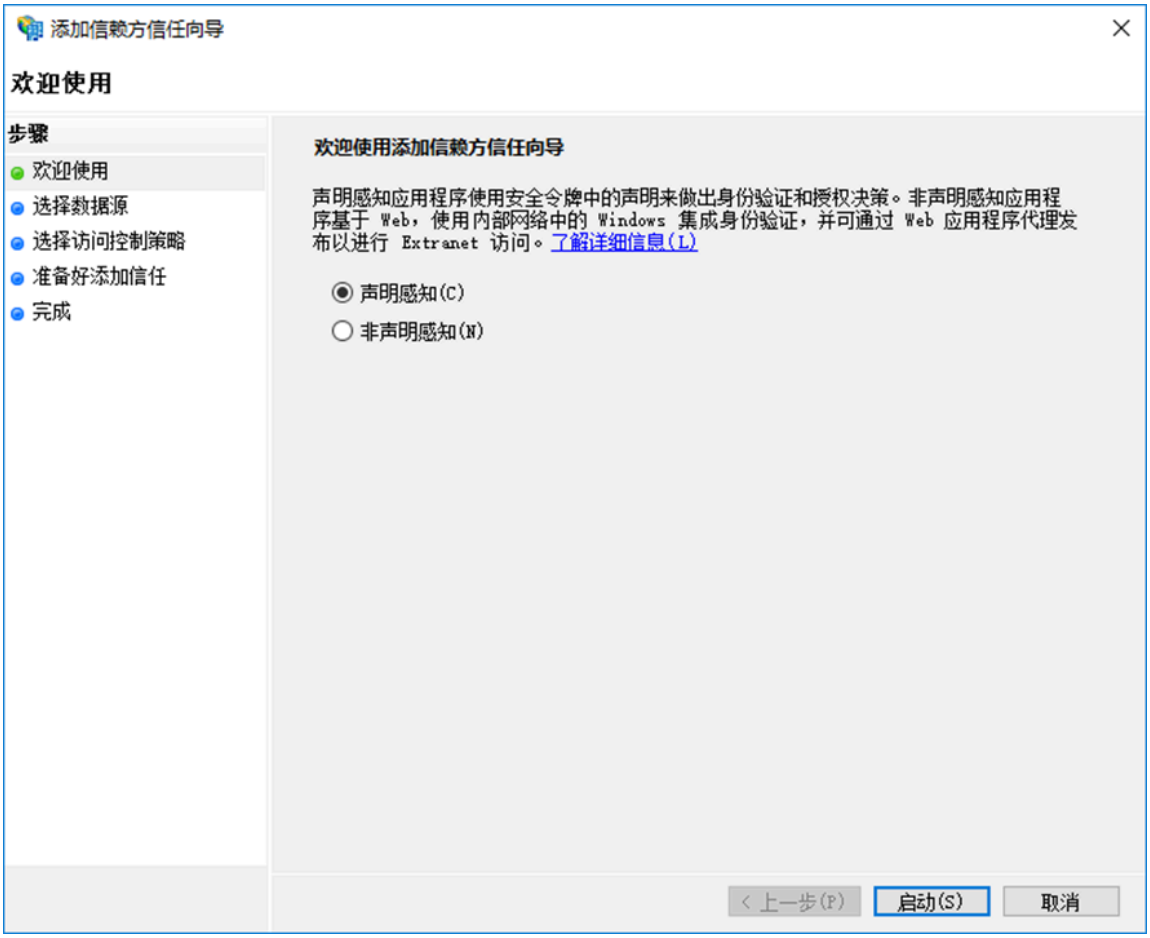
     > **说明:**

配置完成后，企业的阿里云账号将信任企业 AD FS 发来的 SAML 请求中的用户身份和角色信息。

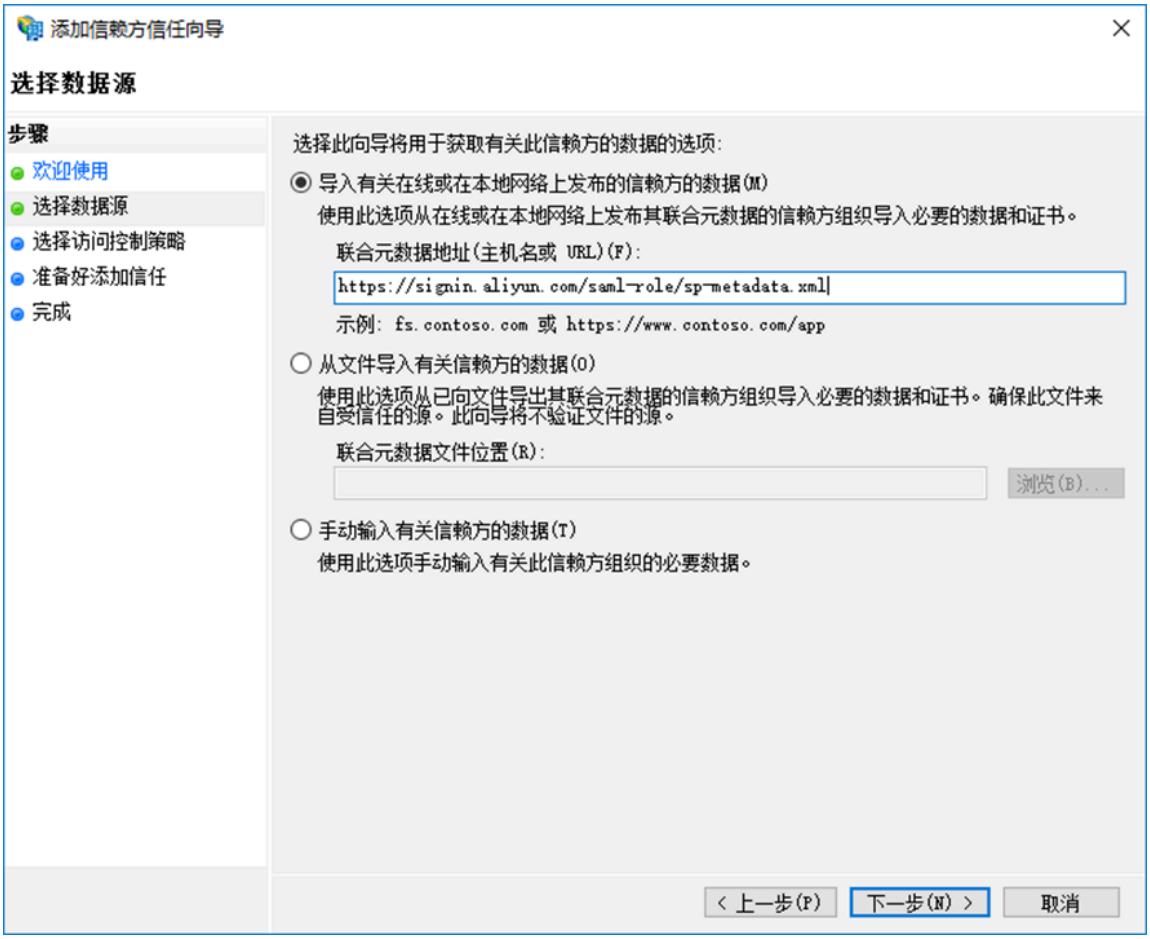
- 在 AD FS 中将阿里云配置为可信 SAML SP。

  在 AD FS 中，SAML SP 被称作信赖方（Relying Party）。设置阿里云作为 AD FS 的可信 SP 的操作步骤如下：

  1. 在服务器管理器的工具菜单中选择AD FS 管理。
  2. 在 AD FS 管理工具中添加信赖方信任。



  3. 为新创建的信赖方设置阿里云的角色 SSO 的 SAML SP 元数据，元数据 URL为`https://signin.aliyun.com/saml-role/sp-metadata.xml`。



4. 按照向导完成配置。

- 为阿里云 SP 配置 SAML 断言属性。

  阿里云需要 AD FS 在 SAML 断言中提供`NameID`，`Role`，`RoleSessionName`属性。AD FS 中通过颁发转换规则来实现这一功能。

  - `NameID`

    配置 Active Directory 中的 Windows 账户名为 SAML 断言中的`NameID`，其操作步骤如下：

    1. 为信赖方编辑声明规则。
    2. 添加颁发转换规则。

    说明:

颁发转换规则（Issuance Transform Rules）：指如何将一个已知的用户属性，经过转换后，颁发为 SAML 断言中的属性。由于我们要将用户在 AD 中的 Windows 账户名颁发为`NameID`，因此需要添加一个新的规则。

3. 声明规则模版选择转换传入声明。

4. 使用如下配置规则，并点击完成。

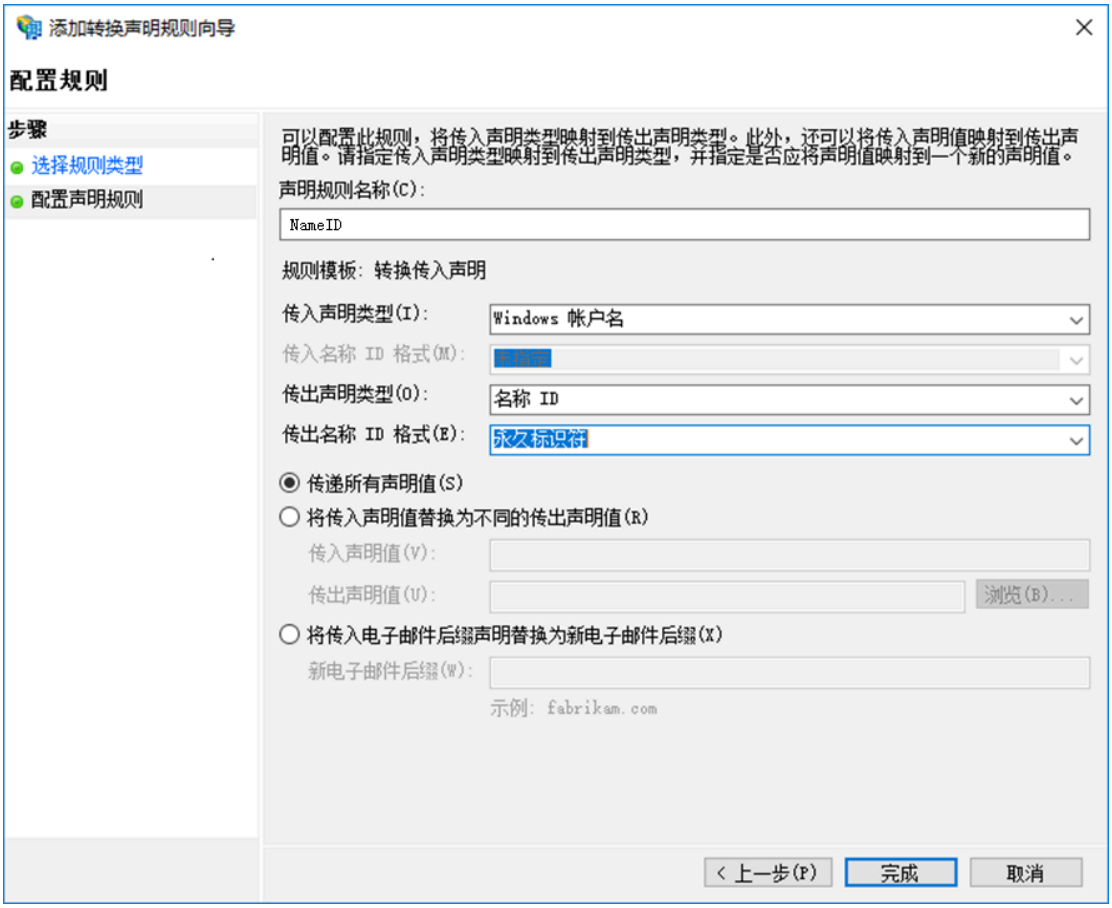
   ■ 声明规则名称：NameID

   ■ 传入声明类型：Windows 账户名

   ■ 传出声明类型：名称 ID

   ■ 传出名称 ID 格式：永久标识符

   ■ 传递所有声明值：勾选



配置完成后，AD FS 将发送阿里云需要的`NameID`格式，如下：

```
<NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-format:
persistent">
    YourDomain\rolessouser
</NameID>
```

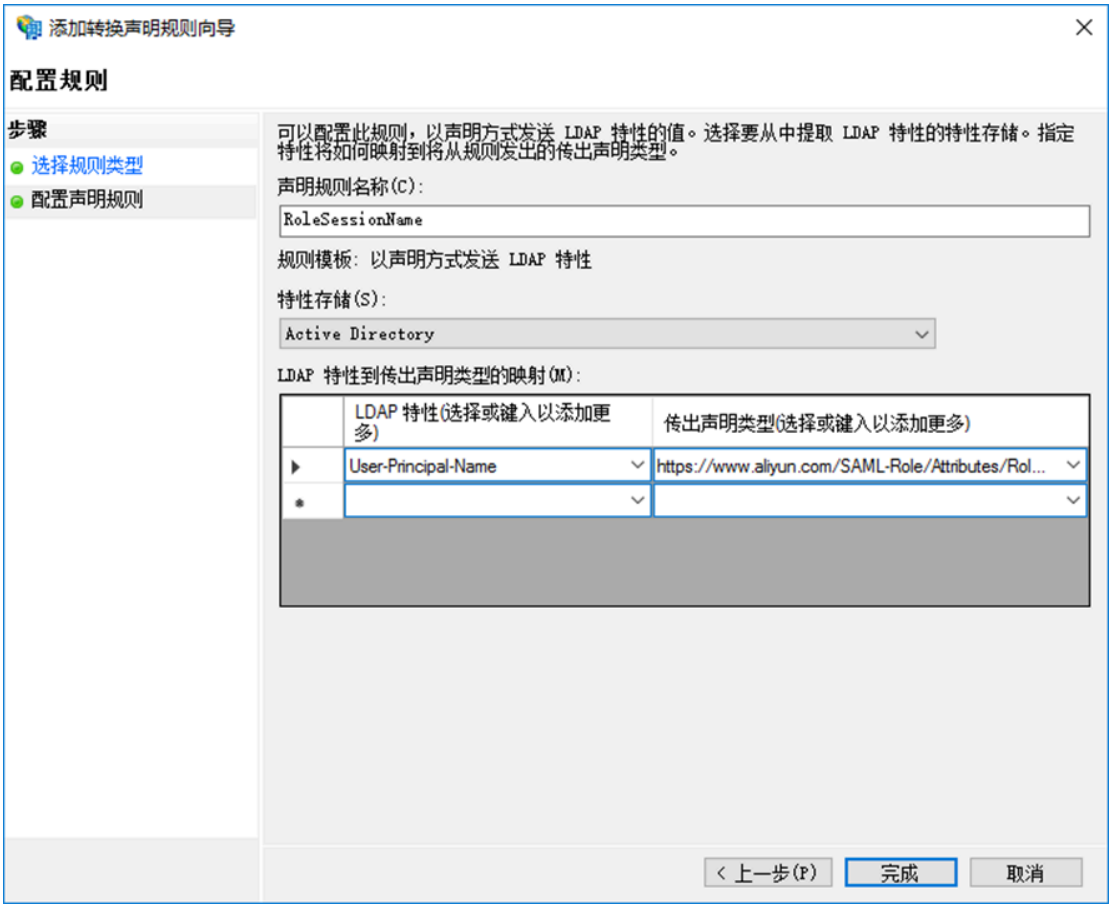
- `RoleSessionName`

  配置 Active Directory 中的 UPN 为 SAML 断言中的`RoleSessionName`，其操作步骤如下：

  1. 单击添加转换声明规则。
  2. 从声明规则模板中选择以声明方式发送 LDAP 特性。

  3. 使用如下配置规则，并点击完成。

     - 声明规则名称：RoleSessionName
     - 特性存储：Active Directory
     - LDAP 特性列：User-Principal-Name（您也可以根据具体需求选择其他属性，如 Email。）
     - 传出声明类型：`https://www.aliyun.com/SAML-Role/Attributes/RoleSessionName`



配置完成后，AD FS 将发送阿里云需要的`RoleSessionName`格式，如下：

```
<Attribute Name="https://www.aliyun.com/SAML-Role/Attributes/
RoleSessionName">
    <AttributeValue>rolessouser@example.com<AttributeValue>
</Attribute>
```

- `Role`

通过自定义规则将特定的用户所属组的信息转化成阿里云上的角色名称。其操作步骤如下：

1. 单击添加转换声明规则。

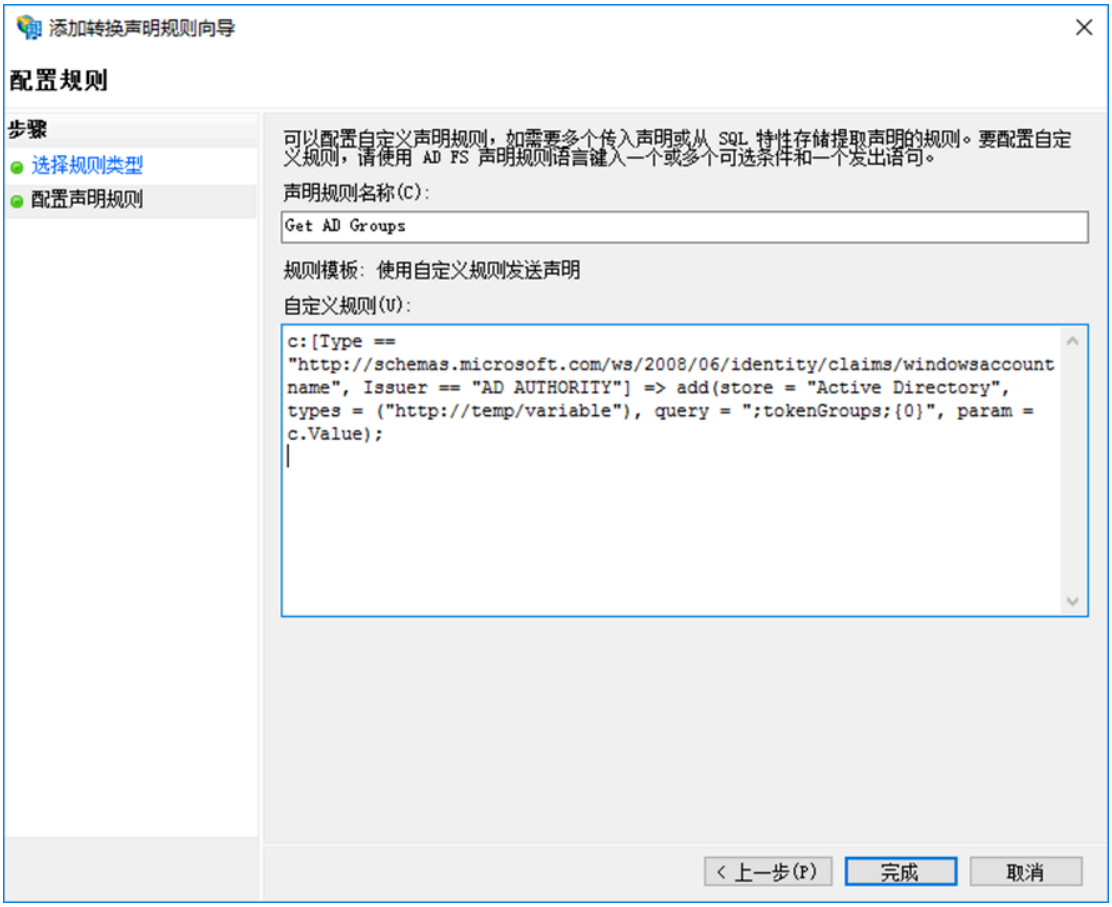
2. 从声明规则模板中选择使用自定义规则发送声明，点击下一步。

3. 使用如下配置规则，并点击完成。

- **声明规则名称：Get AD Groups。**
- **自定义规则：**

```
c:[Type ==
"http://schemas.microsoft.com/ws/2008/06/identity/claims/
windowsaccount
name", Issuer == "AD AUTHORITY"] => add(store = "Active
Directory",
types = ("http://temp/variable"), query = ";tokenGroups;{0
}", param =
```

```
c.Value);
```



说明:

这个规则获取用户在 AD 中所属组的信息，保存在中间变量*http://temp/variable*中。

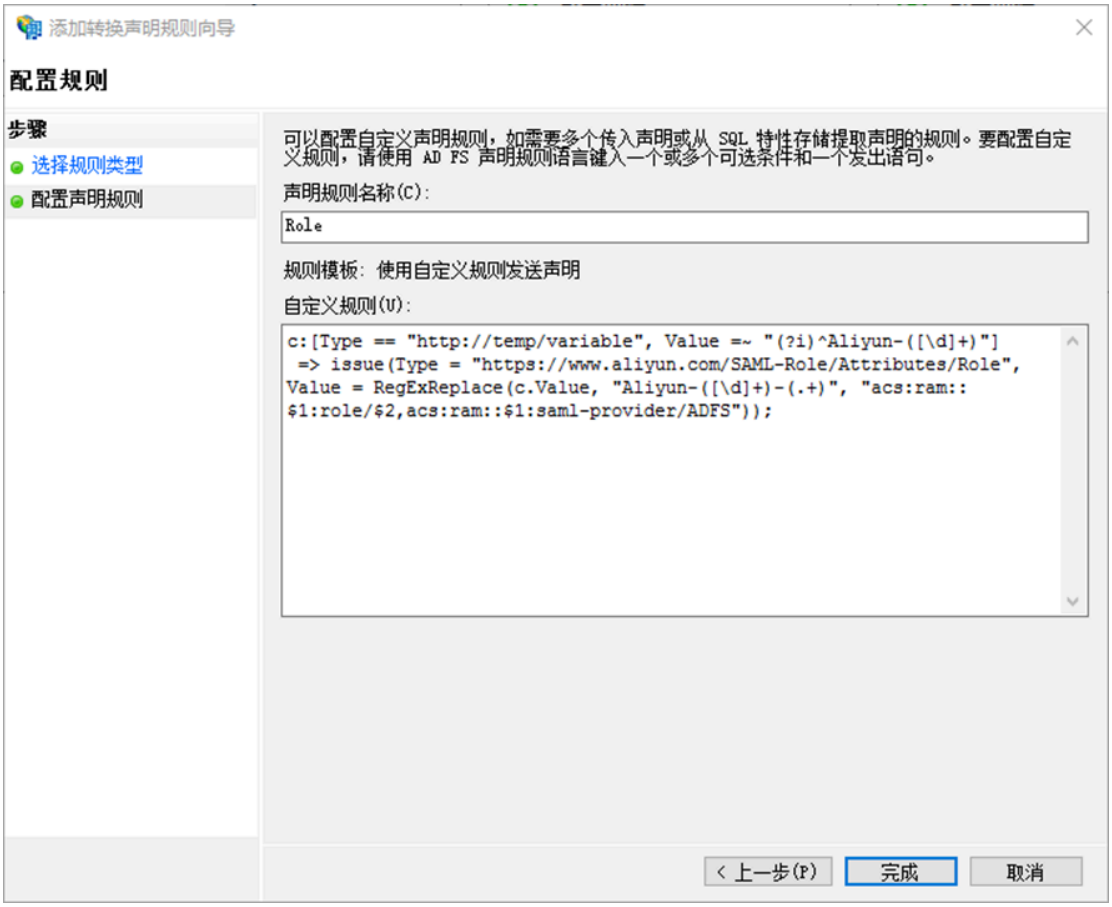
4. **单击添加转换声明规则。**
5. **重复以上步骤，并点击完成。**

   ■ **声明规则名称：Role。**

   ■ **自定义规则：**

```
c:[Type == "http://temp/variable", Value =~ "(?i)^Aliyun-([\d]+)"]
 => issue(Type = "https://www.aliyun.com/SAML-Role/Attributes/Role",
Value = RegExReplace(c.Value, "Aliyun-([\d]+)-(.+)", "acs:ram::
```

```
$1:role/$2,acs:ram::$1:saml-provider/ADFS"));
```



说明:

根据这个规则，如果用户所属的 AD 组中包含 Aliyun-<account-id>-ADFS-Admin 或 Aliyun-<account-id>-ADFS-Reader，则将生成一个 SAML 属性，映射到阿里云上的角色 ADFS-Admin 或 ADFS-Reader。

配置完成后 IdP 将返回阿里云所需要的 SAML 断言，片段如下：

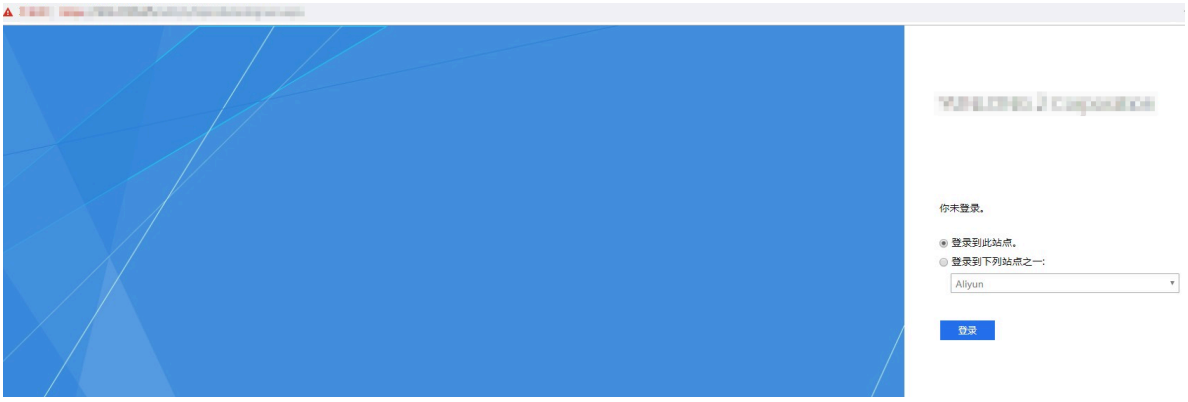
```
<Attribute Name="https://www.aliyun.com/SAML-Role/Attributes/Role
">
    <AttributeValue>acs:ram::<account-id>:role/ADFS-Admin,acs:ram
::<account-id>:saml-provider/ADFS</AttributeValue>
</Attribute>
```

配置验证

1. **登录 AD FS SSO 门户（URL：`https://<ADFS-server>/adfs/ls/IdpInitiatedSignOn.aspx`），选择阿里云应用，输入用户名密码。**

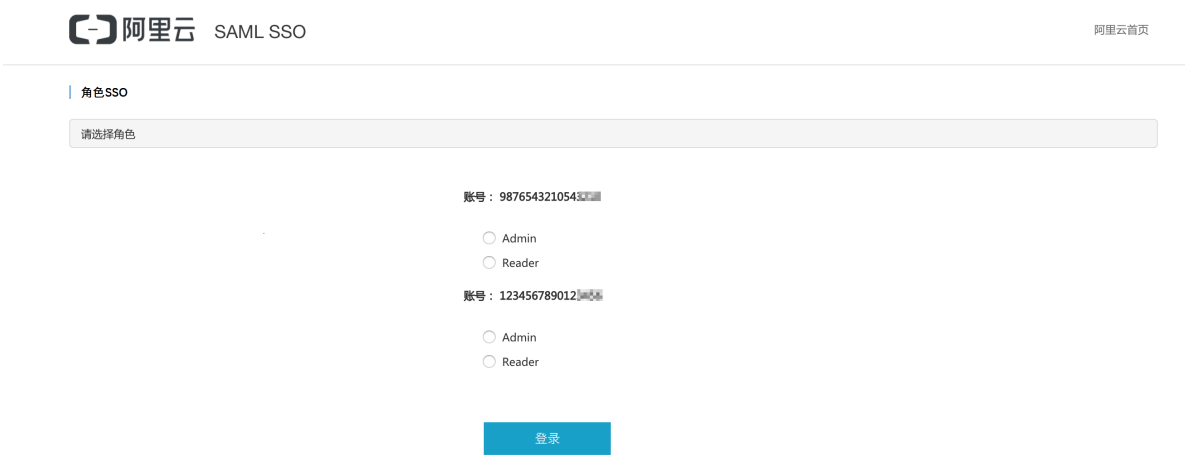
说明:

<ADFS-server>是您的 AD FS 服务器域名或 IP 地址。如果网页不可用，可以通过 PowerShell 开启：`Set-AdfsProperties -EnableIdpInitiatedSignonPage $True`。



2. 在阿里云角色 SSO 页面，选择一个您要登录的角色，单击登录。

> **说明:**
> 如果您的用户在 AD 中只加入了一个组，则在阿里云上只会对应一个角色，该用户将直接登录，无需选择角色。

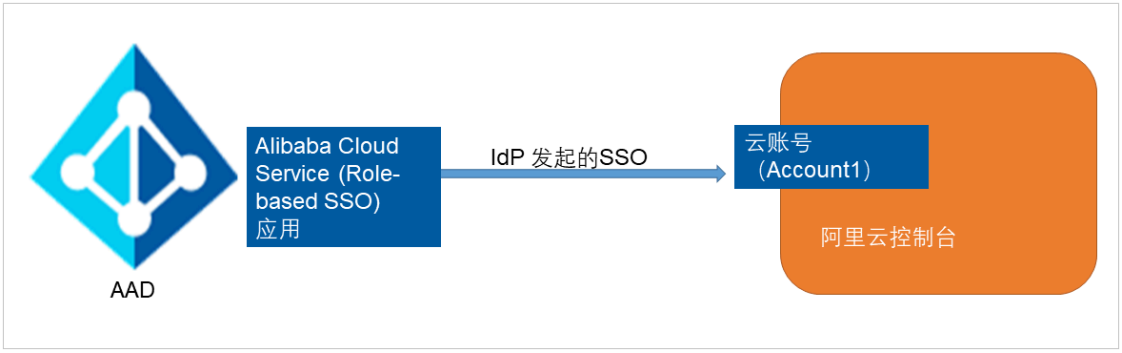


6.5.6 使用Azure AD进行角色SSO的示例

本文提供一个以Azure AD（Azure Active Directory，以下简称 AAD）与阿里云进行角色SSO的示例，帮助用户理解企业IdP与阿里云进行SSO的端到端配置流程。

背景信息

在本示例中，企业拥有一个阿里云账号（Account1）和一个企业员工用户（u2）。企业可以使用AAD进行员工管理，并通过AAD配置包括阿里云在内的企业应用。配置完成后，您可以更好的管理企业员工，企业员工也可以实现从AAD到阿里云的角色SSO。

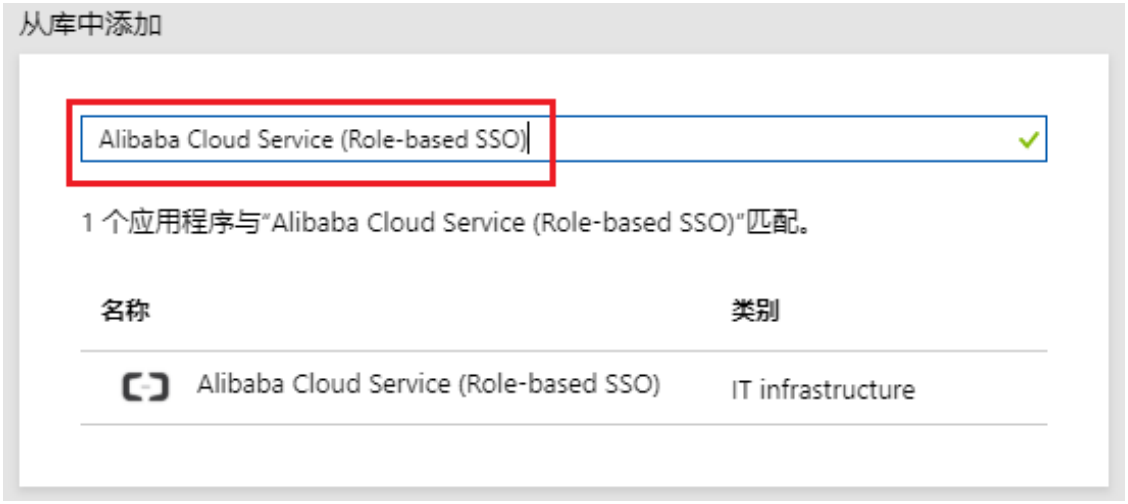


在AAD库中添加应用程序

1. 管理员登录Azure门户。
2. 在左侧导航栏，单击Azure Active Directory > 企业应用程序 > 所有应用程序。

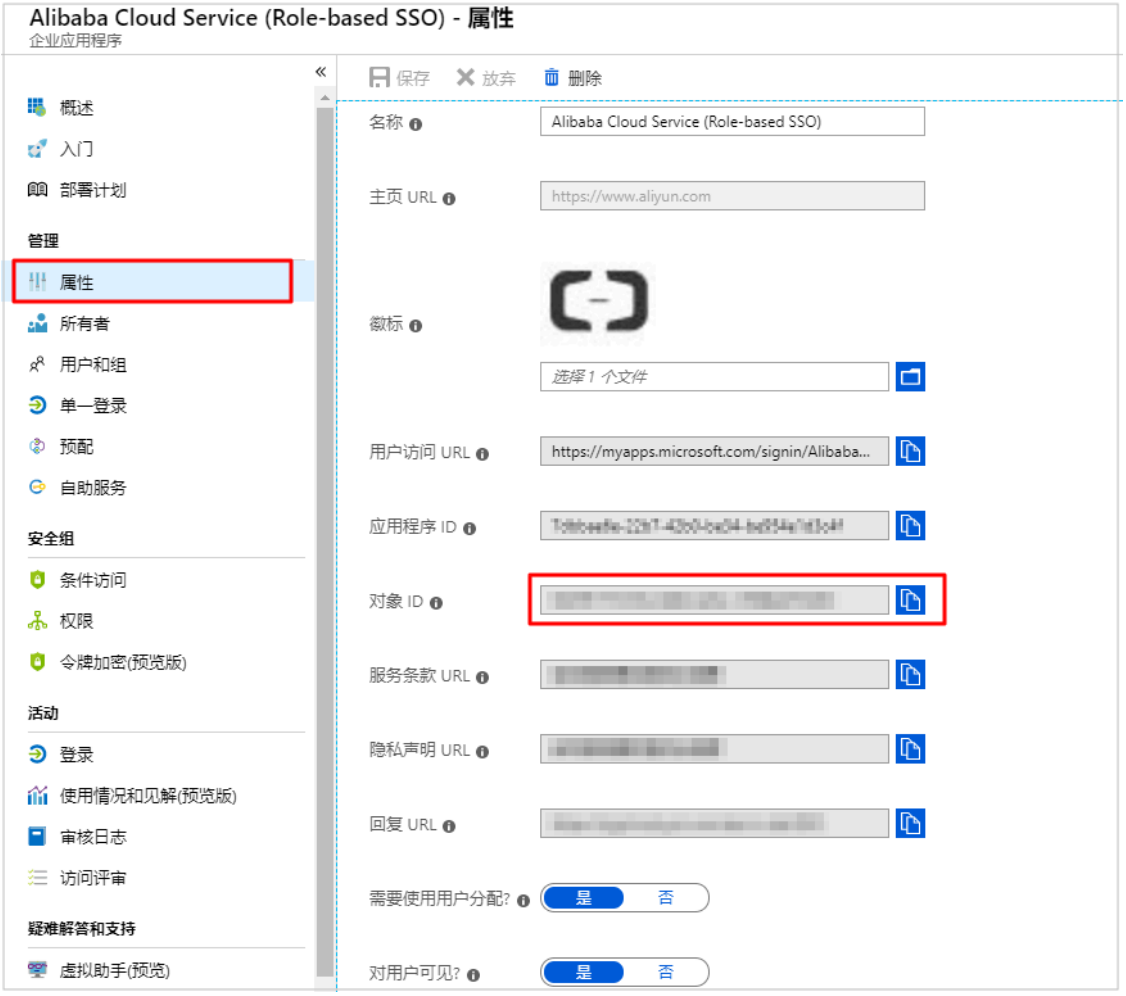
3. 单击新建应用程序。

4. 在添加应用程序页面下的从库中添加搜索区域，输入Alibaba Cloud Service（Role-based SSO）并单击选择。



5. 单击添加。

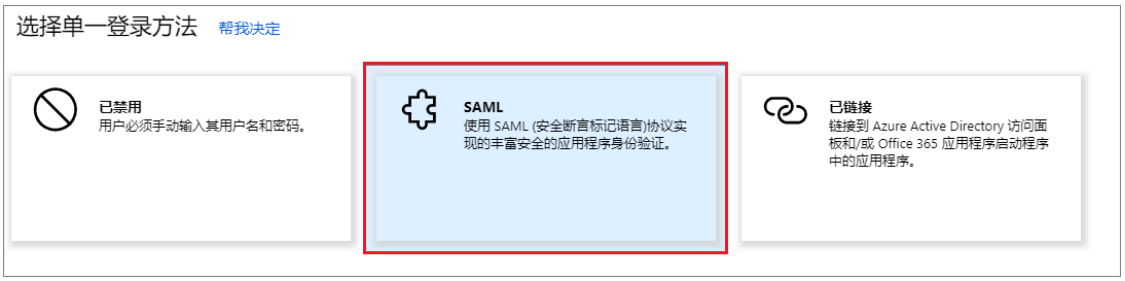
6. 在Alibaba Cloud Service（Role-based SSO）页面下，单击左侧导航栏下的属性，复制并保存对象ID。



配置AAD SSO

1. 管理员登录Azure门户。
2. 在左侧导航栏，单击Azure Active Directory > 企业应用程序 > 所有应用程序。
3. 在名称列表下，单击Alibaba Cloud Service（Role-based SSO）。
4. 在左侧导航栏，选择单一登录。

5. 在更改单一登录模式页面下，单击SAML。



6. 在设置SAML单一登录页面进行配置。

a) 在页面左上角，单击上传元数据文件，选择文件后，单击添加。

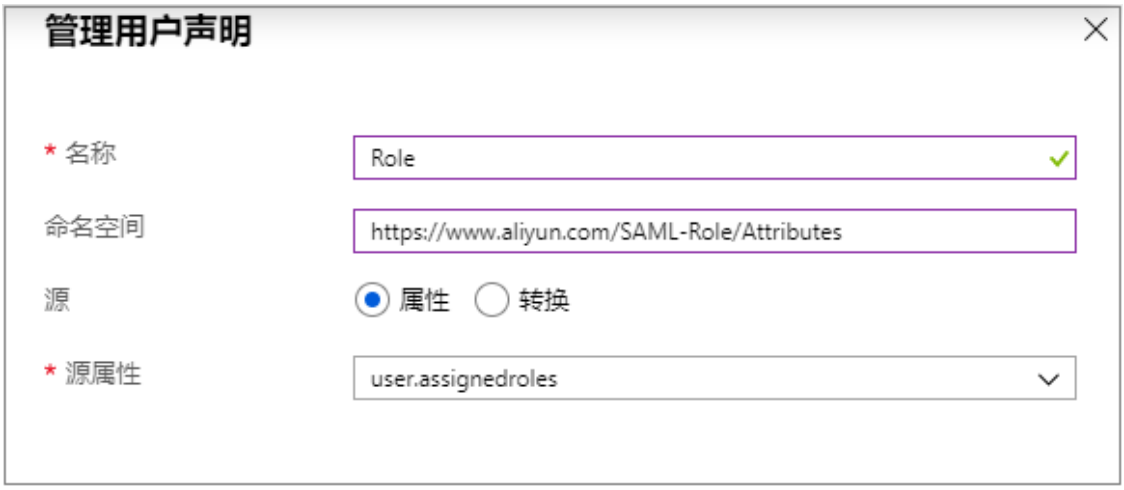
说明:

您可以通过以下URL获取元数据文件： `https://signin.aliyun.com/saml-role/sp-metadata.xml`。

b) 在用户属性和声明区域，单击编辑图标。

c) 单击添加新的声明，设置以下配置后，单击保存。

- 在名称区域下，输入`Role`。
- 在命名空间区域下，输入`https://www.aliyun.com/SAML-Role/Attributes`。
- 在源区域下，选择属性。
- 在源属性区域下，从下拉列表中选择`user.assignedroles`。

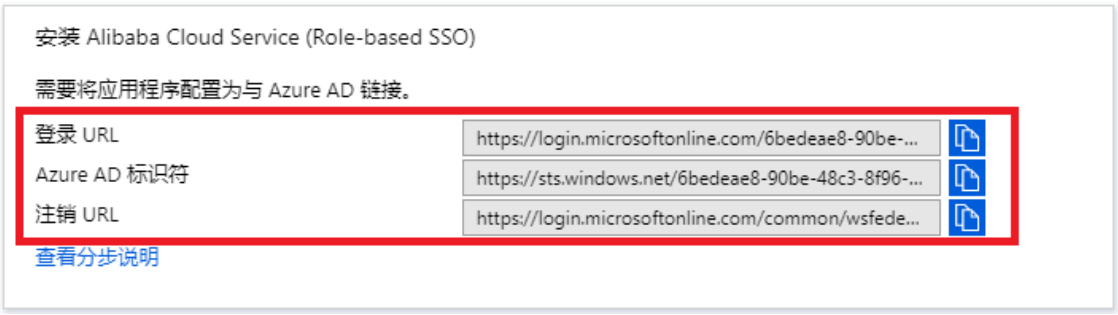


d) 重复上述步骤，添加一个新的声明。

- 在名称区域下，输入`RoleSessionName`。
- 在命名空间区域下，输入`https://www.aliyun.com/SAML-Role/Attributes`。
- 在源区域下，选择属性。
- 在源属性区域下，从下拉列表中选择`user.userprincipalname`。

e) 在SAML签名证书区域下的联合元数据XML，单击下载。

f) 在安装Alibaba Cloud Service（Role-based SSO）区域下，复制登录URL、Azure AD标识符和注销URL。



在阿里云配置角色SSO

1. 云账号（Account1）登录RAM控制台。
2. 在左侧导航栏，单击SSO管理。
3. 在角色SSO页签下，单击新建身份提供商。
4. 输入提供商名称`AAD`和备注。
5. 在元数据文档处，单击上传文件。

> **说明:**
> 上传上述步骤中在SAML 签名证书区域下载的联合元数据XML。

6. 单击确定。
7. 创建身份提供商后，单击前往新建RAM角色。
8. 输入角色名称`AADrole`和备注。
9. 在下拉列表中选择身份提供商`AAD`，单击完成。

> **说明:**
>
> - 您可以根据需要为RAM角色添加权限。关于如何为RAM角色添加权限，请参见#unique_30。

- 当身份提供商和对应的RAM角色后，请保存好对应的ARN。关于如何查看ARN，请参见#unique_139。

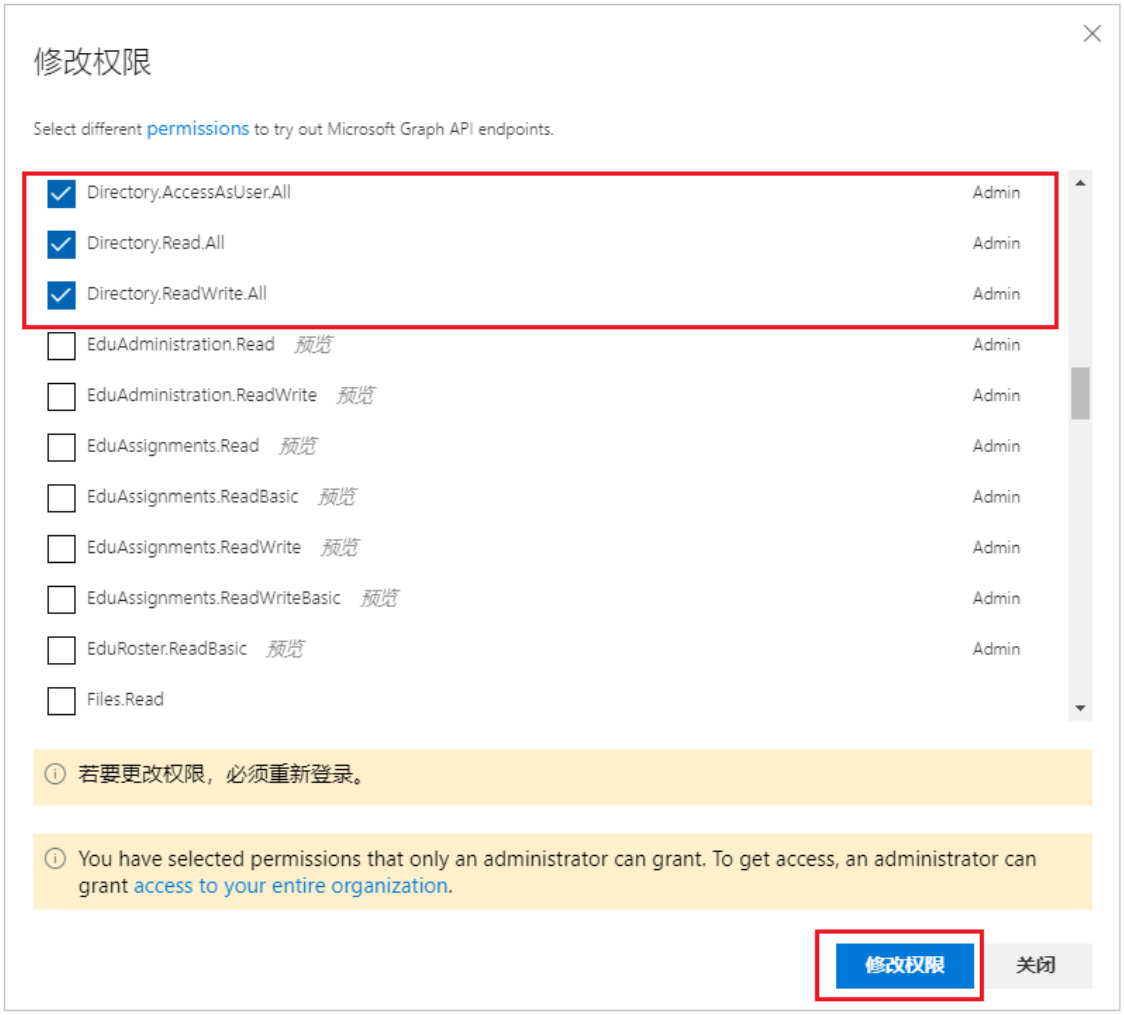
将阿里云RAM角色与AAD用户进行关联

1. **在AAD中创建角色。**

   a) **用户（u2）登录AAD Graph浏览器。**

   b) **单击修改权限。**

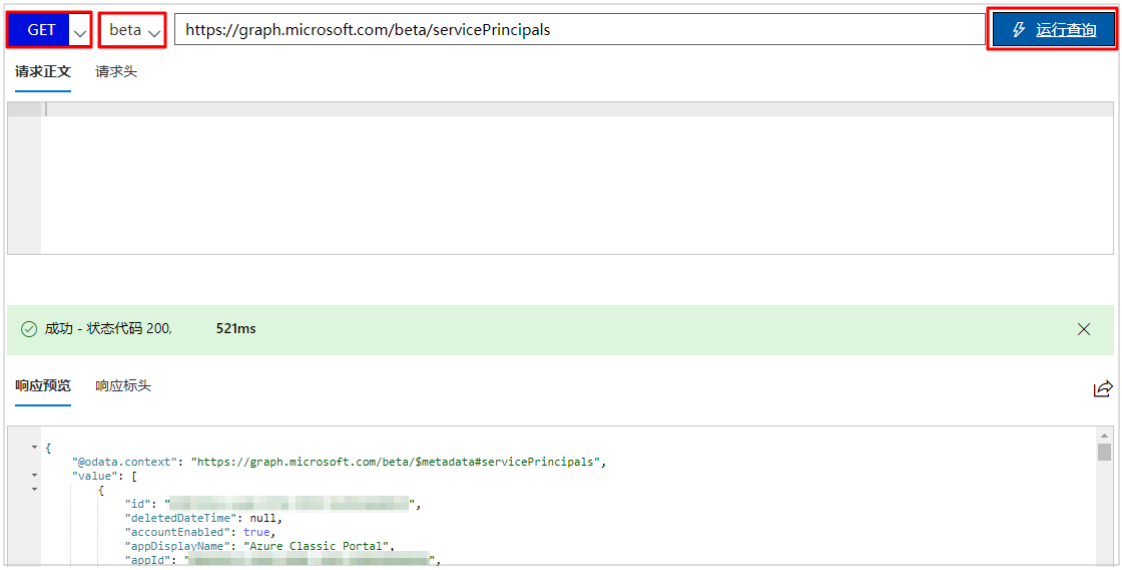
   c) **从下拉列表中选择以下权限并单击修改权限。**



说明:

修改权限后，系统会重定向到Graph浏览器。

d) 在Graph浏览器页面，第一个下拉列表中选择GET，第二个下拉列表中选择beta。在搜索框中输入`https://graph.microsoft.com/beta/servicePrincipals`并单击运行查询。



说明:

如果您有多个目录，您可以在查询区域输入`https://graph.microsoft.com/beta/contoso.com/servicePrincipals`。

e) 在响应预览页签下，从`Service Principal`中提取出`appRoles`属性并保存。

```
"appRoles": [
                {
                    "allowedMemberTypes": [
                        "User"
                    ],
                    "description": "msiam_access",
                    "displayName": "msiam_access",
                    "id": "7dfd756e-8c27-4472-b2b7-38c17fc5****",
                    "isEnabled": true,
                    "origin": "Application",
                    "value": null
                }
            ],
```

说明:

您可以在查询字段中输入https://graph.microsoft.com/beta/servicePrincipals/<objectID>来定位appRoles属性，其中objectID是您在AAD属性页面保存的。

f）返回Graph浏览器，将第一个下拉列表改为PATCH，第二个下拉列表中选择beta。在搜索框中输入https://graph.microsoft.com/beta/servicePrincipals/<objectID>，将以下内容复制到请求正文中并选择运行查询。

```
{
  "appRoles": [
    {
      "allowedMemberTypes":[
        "User"
      ],
      "description": "msiam_access",
      "displayName": "msiam_access",
      "id": "41be2db8-48d9-4277-8e86-f6d22d35****",//appRoles的ID
      "isEnabled": true,
      "origin": "Application",
      "value": null
    },
    { "allowedMemberTypes": [
        "User"
    ],
    "description": "Admin,AzureADProd",
    "displayName": "Admin,AzureADProd",
    "id": "68adae10-8b6b-47e6-9142-6476078c****",//ID生成器（例如:
GUID生成器）实时生成的ID
    "isEnabled": true,
    "origin": "ServicePrincipal",
    "value": "acs:ram::187125022722****:role/aadrole,acs:ram::
187125022722****:saml-provider/AAD"//身份提供商和RAM角色的ARN
    }
  ]
}
```

说明:

您可以根据需要创建多个RAM角色，AAD将在SAML中将这些角色作为声明值进行传送，但是您只能在msiam_access后添加新的角色。

2. 将RAM角色添加到用户（u2）中。

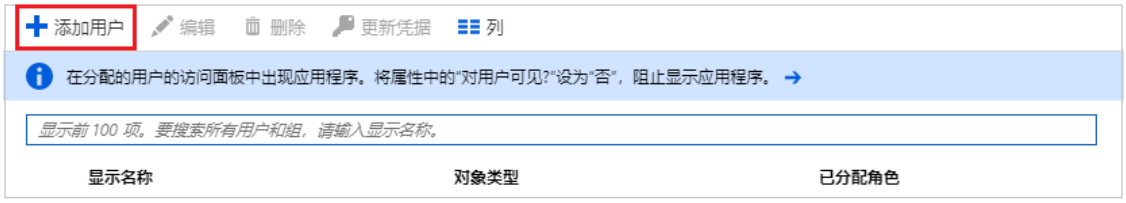
a) 用户（u2）登录Azure门户。

b) 在左侧导航栏，单击Azure Active Directory > 企业应用程序 > 所有应用程序。

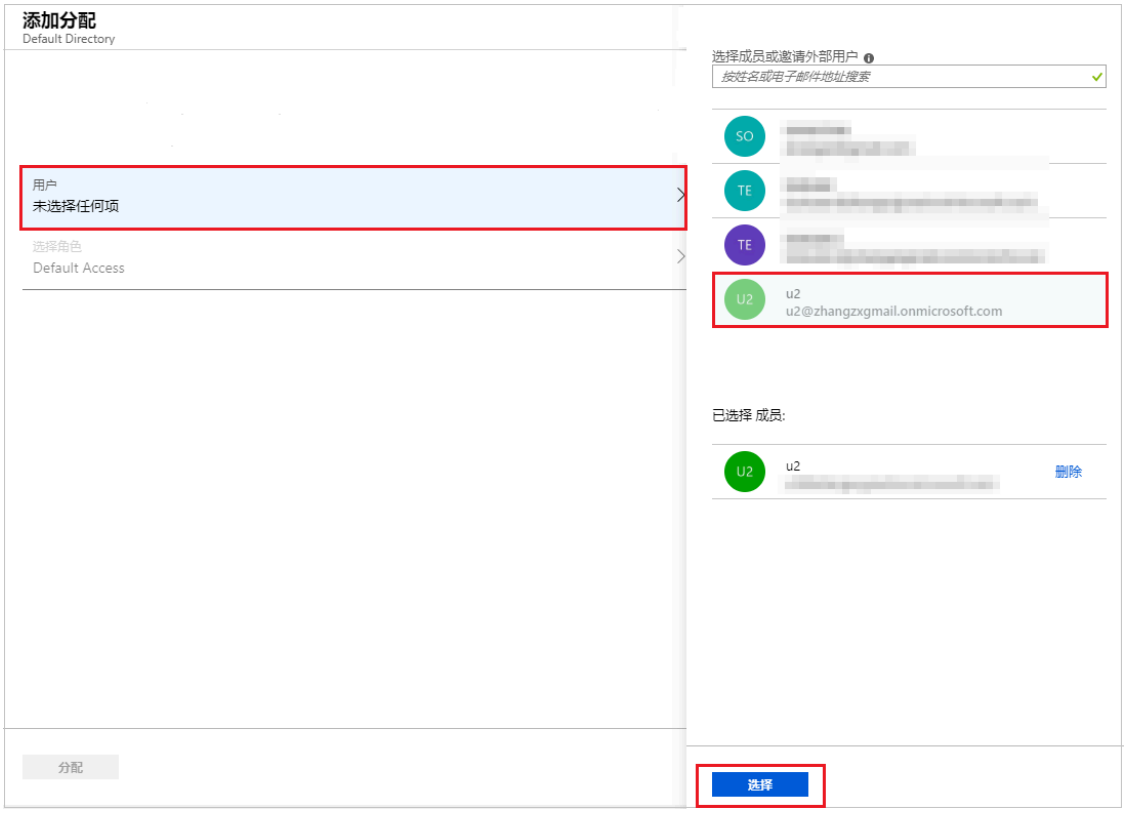
c) 在名称列表下，单击Alibaba Cloud Service（Role-based SSO）。

d) 在左侧导航栏，单击用户和组。

e) 单击左上角的添加用户。

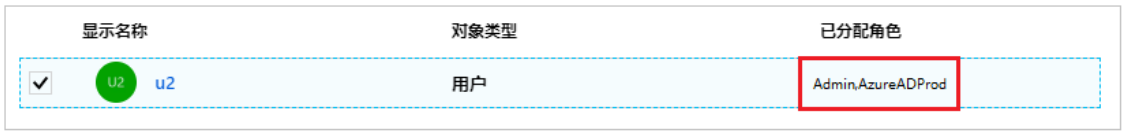


f) 在左侧导航栏，单击用户和组，从用户列表中选择用户（u2），单击选择。



g) 单击分配。

h) 查看分配的角色。

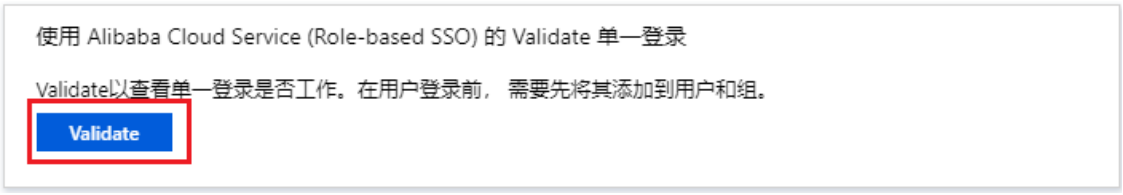


说明:

如果您分配了用户（u2），创建好的RAM角色会自动附加给用户。如果您创建了多个角色，您需要根据需要合理分配角色。如果您需要完成AAD与多个阿里云账号的角色SSO，请重复上述配置步骤。

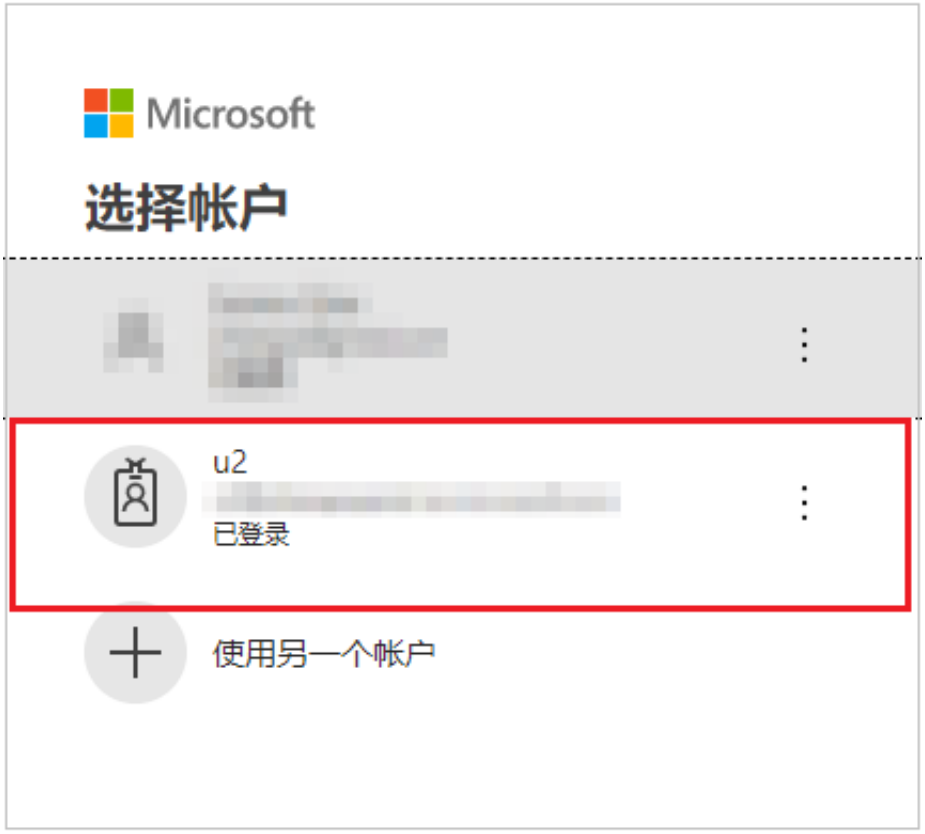
测试角色SSO

1. 登录Azure门户。
2. 在左侧导航栏，单击Azure Active Directory > 企业应用程序 > 所有应用程序。
3. 在名称列表下，单击Alibaba Cloud Service（Role-based SSO）。
4. 在左侧导航栏，选择单一登录。
5. 在使用Alibaba Cloud Service（Role-based SSO）的Validate单一登录区域下，单击Validate。



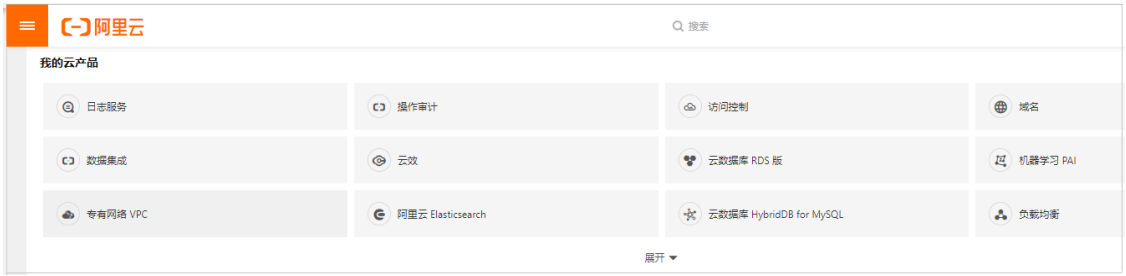
6. 选择作为当前用户登录。

7. 在选择账户页面下，选择用户（u2）。



预期结果

以下界面出现，表示角色SSO成功。



7 开放授权管理（OAuth）

7.1 OAuth 应用概述

RAM 支持使用 OAuth 2.0 协议进行用户认证和应用授权。本文介绍 OAuth 2.0 的基本概念和应用场景。

OAuth 基本概念

为了更好的理解 OAuth 2.0 协议，下面简要介绍与 OAuth 2.0 相关的一些基本概念：

用户	此处用户可以是主账号也可以是 RAM 用户，用户需要登录到阿里云并授权应用访问阿里云资源。
阿里云 OAuth 2.0 服务	对用户进行认证，并接受用户对应用的授权，生成代表用户身份的令牌并返回给被授权的应用。
OAuth 应用	获取用户授权，并获取代表用户身份的令牌，从而可以访问阿里云。 OAuth 2.0 服务目前支持的应用类型包括： · WebApp：指基于浏览器交互的网络应用。 · NativeApp：指操作系统中运行的本地应用，主要为运行在桌面操作系统或移动操作系统中的应用。
OAuth 范围	OAuth 2.0 服务通过 OAuth 范围来限定应用扮演用户登录阿里云后可以访问的范围。 目前 OAuth 支持的范围如下： · `openid`：获取登录用户的基本信息（默认授权域，不可移除）。 说明： OpenID Connect（OIDC） 协议的默认范围，所有的应用默认具有这一范围，不需要额外添加。 · `aliuid`：阿里云颁发的唯一用户标志符。 · `profile`：用户名称等个人信息。 · `/acs/ccc`：阿里云呼叫中心服务 API。 · `/acs/alidns`：阿里云解析 API。

> 说明:
> openid、aliuid、profile 这几个范围与身份令牌相关，其他范围都与访问令牌相关。

令牌	OAuth 2.0 服务可以给应用下发代表登录用户的令牌。 · 身份令牌：身份令牌只包含用户的身份信息，不能用于访问阿里云资源。 · 访问令牌：访问令牌包含了用户的身份信息以及应用的 OAuth 范围，可以用于访问 OAuth 范围内的阿里云资源。 · 刷新令牌：刷新令牌可以用于换取新的访问令牌。
阿里云 API	应用通过调用 API 可以访问相应资源。

OAuth 的应用场景

- Web 应用登录阿里云
- Native 应用登录阿里云
- 通过 OIDC 获取用户信息

7.2 OAuth 应用典型场景

7.2.1 Web 应用登录阿里云

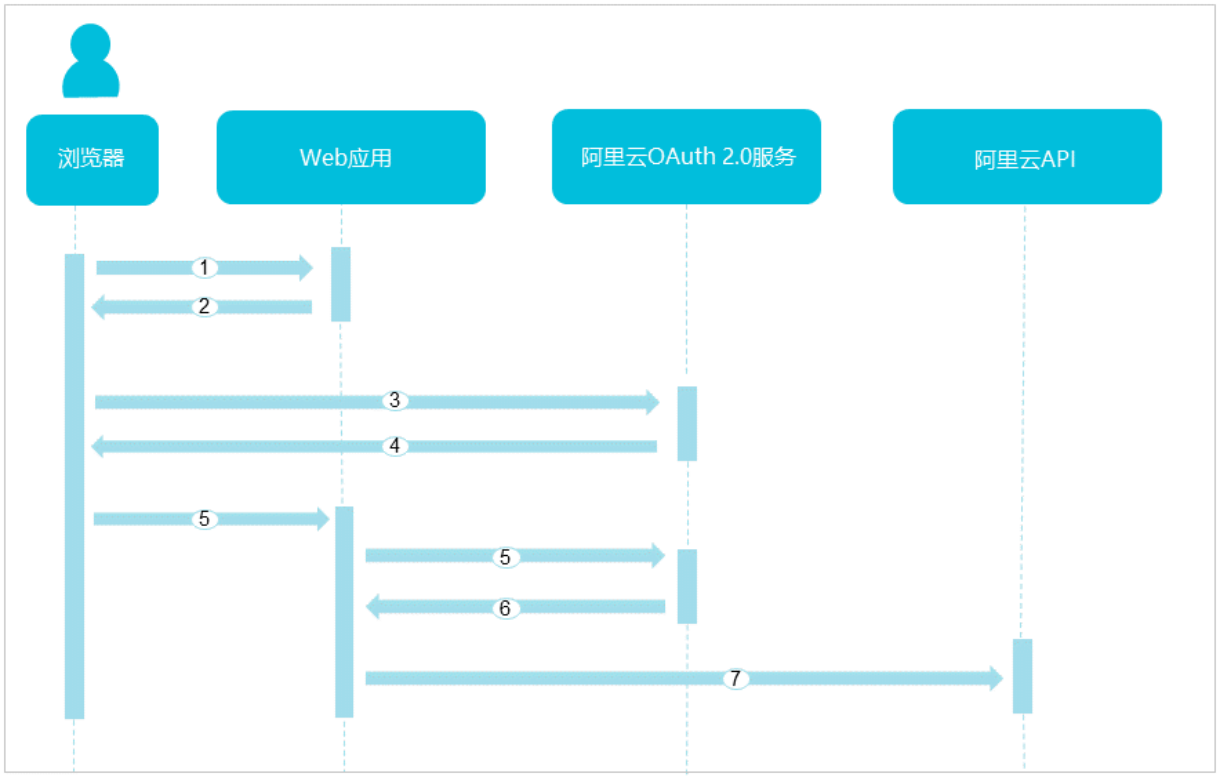
本文介绍 Web 应用如何通过 OAuth 2.0 扮演登录用户访问阿里云 API。

前提条件

Web 应用扮演登录用户访问阿里云首先需要创建应用，为应用提供恰当的名称、OAuth 范围、回调地址等关键信息，并为应用生成应用密钥。详情请参考：#unique_146。

> 说明:
> 应用创建成功之后，可以在云账号内直接扮演用户。如果要扮演其他云账号的用户，需要获得其他云账号的授权。

基本流程



1. 用户通过浏览器登录 Web 应用。
2. Web 应用重定向到阿里云 OAuth 2.0 服务并将 URL 返回给浏览器。

> 说明:
> 如果用户还未登录，则会进一步重定向到阿里云登录服务。

3. 用户通过浏览器登录阿里云 OAuth 2.0 服务并申请授权码。
4. 阿里云 OAuth 2.0 服务重定向到 Web 应用并返回授权码。
5. Web 应用使用授权码向阿里云 OAuth 2.0 服务申请代表用户身份的令牌。
   - 如何获取访问令牌，请参考：获取访问令牌。
   - 如何获取新的访问令牌，请参考：获取新的访问令牌。
   - 如何撤销刷新令牌，请参考：撤销刷新令牌。
6. 阿里云 OAuth 2.0 服务向 Web 应用返回令牌。
7. Web 应用通过获取的令牌向阿里云发起访问 API 的请求。

> 说明:
> 由于令牌可以代表用户身份，因此应用可以访问当前用户的资源。

获取访问令牌

1. Web 应用通过浏览器将用户重定向到阿里云 OAuth 2.0 服务从而获取授权码。

   授权码的请求地址：`https://signin.aliyun.com/oauth2/v1/auth`。

   表 7-1: 请求参数

参数名称	是否必选	描述
client_id	是	应用的身份 ID。
redirect_uri	是	创建应用的重定向 URI 之一。
response_type	是	返回类型。根据 OAuth 2.0 协议，目前支持设置此参数的取值为：code。
scope	否	空格分隔的 OAuth 范围列表。如不指定此参数取值，则默认为应用的全部 OAuth 范围。
access_type	否	应用的访问类型。取值分为两种类型： · online：应用不需要离线刷新访问令牌。 · offline：针对离线访问类型的请求，会发放刷新令牌，应用可以根据需求持续刷新访问令牌。 默认取值为：online。
state	否	应用通过 state 参数实现多种目的，例如：状态保持、作为 nonce 使用从而减少 CSRF 威胁等。state 如果设置为任意字符串，阿里云 OAuth2.0 服务会将请求中的 state 参数及取值原样放到返回参数中以供后续使用。

   请求示例

```
https://signin.aliyun.com/oauth2/v1/auth?
client_id=123****
redirect_uri=https%3A%2F%2Fyourwebapp.com%2Fauthcallback%2F&
```

```
response_type=code&
scope=openid%20%2Facs%2Fccc&
access_type=offline&
state=123456****
```

返回示例

```
GET HTTP/1.1 302 Found
Location: https://yourwebapp.com/authcallback/?code=ABAFDGDFXYZW888&
state=123456****
```

2. **Web 应用使用授权码向阿里云 OAuth 2.0 服务申请代表用户身份的令牌。**

   换取访问令牌的请求地址：`https://oauth.aliyun.com/v1/token`。

   表 7-2: 请求参数

参数名称	是否必选	描述
code	是	初始请求中获取的授权码。
client_id	是	应用的身份 ID。
redirect_uri	是	初始请求中设置的参数。
grant_type	是	根据 OAuth 2.0 协议， 取值为：authorization_code。
client_secret	否	应用的密钥，用作换取访问令牌时鉴定应用身份的密码。

   请求示例

```
POST /v1/token HTTP/1.1
Host: oauth.aliyun.com
Content-Type: application/x-www-form-urlencoded
code=ABAFDGDFXYZW888&
client_id=123****
client_secret=`your_client_secret`&
redirect_uri=https://yourwebapp.com/authcallback/&
grant_type=authorization_code
```

   表 7-3: 返回参数

参数名称	描述
access_token	访问令牌。访问令牌可以代表用户身份，应用使用此访问令牌来访问阿里云 API。应用不需要理解访问令牌的含义，直接使用即可。
expires_in	访问令牌的剩余有效时间，单位为秒。

参数名称	描述
token_type	访问令牌的类型。取值为：Bearer。
id_token	身份令牌。身份令牌为 OAuth 签名的 JWT（JSON Web Token）。如果初始请求的 scope 参数包含了 openid，则返回身份令牌。
refresh_token	刷新令牌。如果初始请求时应用的访问类型为：offline，则返回刷新令牌。

返回示例

```
{
  "access_token": "eyJraWQiOiJrMTIzNCIsImVu****",
  "token_type": "Bearer",
  "expires_in": 3600,
  "refresh_token": "Ccx63VVeTn2dxV7ovXXfLtAqLLERA****",
  "id_token": "eyJhbGciOiJIUzI1****"
}
```

获取新的访问令牌

换取访问令牌的请求地址：`https://oauth.aliyun.com/v1/token`。

表 7-4: 请求参数

参数名称	是否必选	描述
refresh_token	是	用授权码换取访问令牌时获得的刷新令牌。
client_id	是	应用的身份 ID。
grant_type	是	根据 OAuth 2.0 协议， 取值为：refresh_token。
client_secret	否	应用的密钥，用作换取访问令牌时鉴定应用身份的密码。

请求示例

```
POST /v1/token HTTP/1.1
Host: oauth.aliyun.com
Content-Type: application/x-www-form-urlencoded
refresh_token=Ccx63VVeTn2dxV7ovXXfLtAqLLERAH1Bc&
client_id=123****
client_secret=`your_client_secret`&
```

```
grant_type=refresh_token
```

表 7-5: 返回参数

参数名称	描述
access_token	新的访问令牌。应用可以使用新的访问令牌来访问阿里云 API。
expires_in	访问令牌的剩余有效时间，单位为秒。
token_type	访问令牌的类型。取值为：Bearer。

返回示例

```
{
  "access_token": "eyJraWQiOiJrMTIzNCIsImVu****",
  "token_type": "Bearer",
  "expires_in": 3600,
}
```

说明:

本次请求的返回值与用授权码换取访问令牌的返回值一致，但不包含 refresh_token 和 id_token。

撤销刷新令牌

如果应用获取了刷新令牌，在特定的场景下也需要撤销刷新令牌，例如：用户退出登录应用或用户将自己的账号从应用中移除等。

撤销刷新令牌的请求地址：`https://oauth.aliyun.com/v1/revoke`。

表 7-6: 请求参数

参数名称	是否必选	描述
token	是	需要撤销的刷新令牌。
client_id	是	应用的身份 ID。
client_secret	否	应用的密钥。

7.2.2 Native 应用登录阿里云

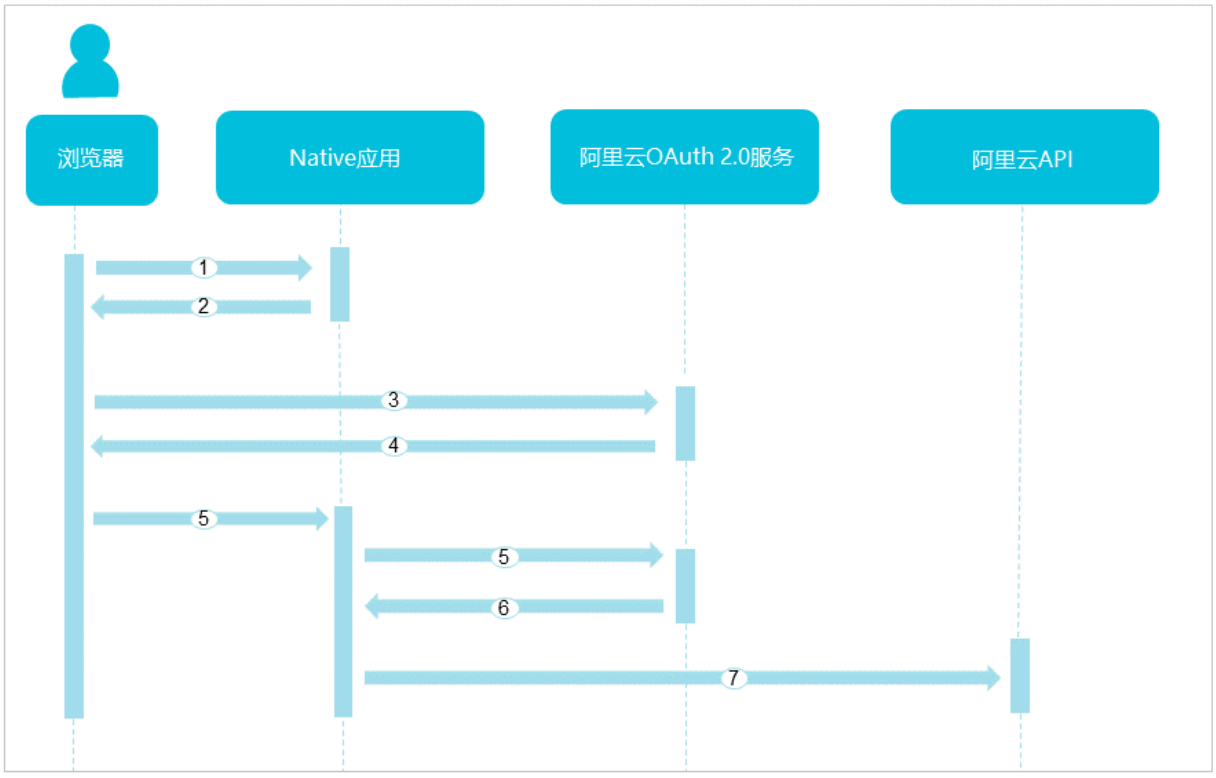
本文介绍桌面和移动端的 Native 应用如何通过 OAuth 2.0 扮演登录用户访问阿里云 API。

前提条件

Native 应用扮演登录用户访问阿里云首先需要创建应用，为应用提供恰当的名称、OAuth 范围、回调地址等关键信息。详情请参考：#unique_146。由于 Native 应用运行在非可信环境，无法有效保护应用密钥，因此 Native 应用不使用应用密钥。

> **说明:**
> 应用创建成功之后，可以在云账号内直接扮演用户。

基本流程



1. 用户通过浏览器登录 Native 应用。
2. Native 应用重定向到阿里云 OAuth 2.0 服务并将 URL 返回给浏览器。

> **说明:**
> 如果用户还未登录，则会进一步重定向到阿里云登录服务。

3. 用户通过浏览器登录阿里云 OAuth 2.0 服务并申请授权码。
4. 阿里云 OAuth 2.0 服务重定向到 Native 应用并返回授权码。

5. Native 应用使用授权码向阿里云 OAuth 2.0 服务申请代表用户身份的令牌。
   - 如何获取访问令牌，请参考：获取访问令牌。
   - 如何获取新的访问令牌，请参考：获取新的访问令牌。
   - 如何撤销刷新令牌，请参考：撤销刷新令牌。
6. 阿里云 OAuth 2.0 服务向 Native 应用返回令牌。
7. Native 应用通过获取的令牌向阿里云发起访问 API 的请求。

说明:

由于令牌可以代表用户身份，因此应用可以访问当前用户的资源。

Proof Key 机制的原理

Native 应用支持 Proof Key 机制，用于每次获取授权码以及用授权码换取访问令牌。

说明:

这一机制可以减轻针对授权码截获的攻击。

1. Native 应用应用生成：`code_verifier`，并保存好这个随机字符串。

说明:

`code_verifier`是一个高熵值的随机字符串，其取值为：`[A-Z] / [a-z] / [0-9] / "-" / "." / "_" / "~"`，长度限制为：43 ~ 128 个字符。

2. 应用根据`transform`的方式选择生成`code_challenge`。应用在申请授权码的同时提交：`code_challenge` 以及生成`code_challenge`的方式。

```
code_challenge = transform(code_verifier, [Plain|S256])
```

方式	取值
`plain`	如果 `transform`的方式选择为：`plain`，那么`code_challenge`与`code_verifier`的值相同。

方式	取值
S256	如果transform的方式选择为：S256，那么code_challenge等于code verifier的SHA256 哈希值。 code_challenge=BASE64URL-Encode(SHA256(ASCII(code_verifier))) 说明: 哈希算法的输入为：code_verifier 的ASCII 编码串，哈希算法的输出字符串需要进行 BASE64-URL 编码。

code_challenge选择方式计算示例：

如果应用采用方式为 S256，生成code_verifier的值为：dBjftJeZ4CVP-mB92K27uhbUJU1p1r_wW1gFWFOEjXk，那么code_challenge为：E9Melhoa2OwvFrEMTJguCHaoeK1t8URWbuGJSstw-cM。

3. 应用获取授权码后，在使用授权码换取访问令牌时，服务端通过计算判断是否颁发令牌。

   授权码需包括code_verifier，服务端按照应用选择的transform方式对code_verifier进行计算，将结果与code_challenge进行对比，如果一致则颁发访问令牌。

获取访问令牌

1. Native 应用通过浏览器将用户重定向到阿里云 OAuth 2.0 服务从而获取授权码。

   授权码的请求地址：https://signin.aliyun.com/oauth2/v1/auth 。

   表 7-7: 请求参数

参数名称	是否必选	描述
client_id	是	应用的身份 ID。
redirect_uri	是	创建应用的重定向 URI 之一。
response_type	是	返回类型。根据 OAuth 2.0 协议，目前支持设置此参数的取值为：code。

参数名称	是否必选	描述
scope	否	空格分隔的 OAuth 范围列表。如不指定此参数取值，则默认为应用的全部 OAuth 范围。
state	否	应用通过 state 参数实现多种目的，例如：状态保持、作为 nonce 使用从而减少 CSRF 威胁等。state 如果设置为任意字符串，阿里云 OAuth 2.0 服务会将请求中的 state 参数以及取值原样放到返回参数中以供后续使用。
code_challenge_method	否	如果不指定此参数，则默认取值为：plain。
code_challenge	否	根据客户端指定的 code_challenge_method，对随机生成的 code_verifier 进行转换和编码，转换的结果作为 code_challenge 参数的值在授权码请求中使用。 **说明:** 如果不指定此参数，则无法使用 Proof Key 机制，在使用授权码换取令牌时也不需要指定相应的 code_verifier，若授权码被截获可以直接被用于换取访问令牌。

请求示例

```
https://signin.aliyun.com/oauth2/v1/authorize?
client_id=98989****
redirect_uri=meeting%3A%2F%2Fauthorize%2F
&response_type=code
&scope=openid%20%2Fworksuite%2Fuseraccess
&state=123456****
&code_challenge=E9Melhoa2OwvFrEMTJguCHaoeK1t8URWbuGJSst****
```

```
&code_challenge_method=S256
```

返回示例

```
GET HTTP/1.1 302 Found
Location: meeting://authorize/?code=ABAFDGDFXYZW888&state=123456****
```

2. **Native 应用使用授权码向阿里云 OAuth 2.0 服务申请代表用户身份的令牌。**

   换取访问令牌的请求地址：`https://oauth.aliyun.com/v1/token`。

   表 7-8: 请求参数

参数名称	是否必选	描述
code	是	初始请求中获取的授权码。
client_id	是	应用的身份 ID。
redirect_uri	是	初始请求中设置的参数。
grant_type	是	根据 OAuth 2.0 协议， 取值为：authorization_code。
code_verifier	否	初始请求中生成的 code verifier，对应于授权码请求中的 code_challenge 参数。

   请求示例

```
POST /v1/token HTTP/1.1
Host: oauth.aliyun.com
Content-Type: application/x-www-form-urlencoded
code=ABAFDGDFXYZW888&
client_id=98989****
redirect_uri=meeting://authorize/&
grant_type=authorization_code&
code_verifier=dBjftJeZ4CVP-mB92K27uhbUJU1p1r_wW1gFWFOEjXk
```

   表 7-9: 返回参数

参数名称	描述
access_token	访问令牌。访问令牌可以代表用户身份，应用使用此访问令牌来访问阿里云 API。应用不需要理解访问令牌的含义，直接使用即可。
expires_in	访问令牌的剩余有效时间，单位为秒。

参数名称	描述
token_type	访问令牌的类型。取值为：Bearer。
id_token	身份令牌。id_token 为 OAuth 签名的 JWT（JSON Web Token）。如果初始请求的 scope 参数包含了 openid，则返回身份令牌。
refresh_token	刷新令牌。此参数可以直接使用，Native 应用不需要指定访问令牌的值，可以直接获得刷新令牌。

返回示例

```
{
  "access_token": "eyJraWQiOiJrMTIzNCIsImVuYyI6****",
  "token_type": "Bearer",
  "expires_in": 3600,
  "refresh_token": "Ccx63VVeTn2dxV7ovXXfLtAqLLERA****",
  "id_token": "eyJhbGciOiJIUzI1****"
}
```

获取新的访问令牌

换取访问令牌的请求地址：`https://oauth.aliyun.com/v1/token`。

表 7-10: 请求参数

参数名称	是否必选	描述
refresh_token	是	用授权码换取访问令牌时获得的刷新令牌。
client_id	是	应用的身份 ID。
grant_type	是	根据 OAuth 2.0 协议， 取值为：refresh_token。

请求示例

```
POST /v1/token HTTP/1.1
Host: oauth.aliyun.com
Content-Type: application/x-www-form-urlencoded
refresh_token=Ccx63VVeTn2dxV7ovXXfLtAqLLERAH****
client_id=98989****
```

```
grant_type=refresh_token
```

表 7-11: 返回参数

参数名称	描述
access_token	新的访问令牌。应用使用新的访问令牌来访问阿里云 API。
expires_in	访问令牌的剩余有效时间，单位为秒。
token_type	访问令牌的类型。取值为：Bearer。

返回示例

```
{
  "access_token": "eyJraWQiOiJrMTIzNCIsImVuYyI6****",
  "token_type": "Bearer",
  "expires_in": 3600,
}
```

说明:

本次请求的返回值与用授权码换取访问令牌的返回值一致，但不包含 refresh_token 和 id_token。

撤销刷新令牌

Native 应用获取了刷新令牌后，在用户退出登录应用时或用户将自己的账号从应用中移除时，必须撤销刷新令牌。

撤销刷新令牌的请求地址：`https://oauth.aliyun.com/v1/revoke`。

表 7-12: 请求参数

参数名称	是否必选	描述
token	是	需要撤销的刷新令牌。
client_id	是	应用的身份 ID。

7.2.3 通过 OIDC 获取用户信息

OIDC（OpenID Connect）是建立在 OAuth 2.0 基础上的一个认证协议，本文为您介绍应用如何使用 OIDC 获取阿里云登录用户的信息。

前提条件

应用获取用户登录信息首先需要创建应用，为应用提供恰当的名称、OAuth 范围、回调地址等关键信息，并为应用生成应用密钥。详情请参考：#unique_146。

OIDC 基本概念

身份令牌	OIDC 可以给应用下发代表登录用户的身份令牌。身份令牌用于获取姓名，登录名等用户信息，不能用于访问阿里云服务。
OIDC discovery endpoint	OIDC 协议包含了不同的 endpoint 用于不同的目的，discovery endpoint 中包含 OIDC 协议所需要的所有配置信息，方便开发者使用。

说明:

Discovery endpoint 是通过 JSON 文档来提供一系列键值，其中包含主要的提供者信息，例如：协议支持的响应类型，令牌颁发者的取值，身份令牌签名密钥的地址，签名算法等。

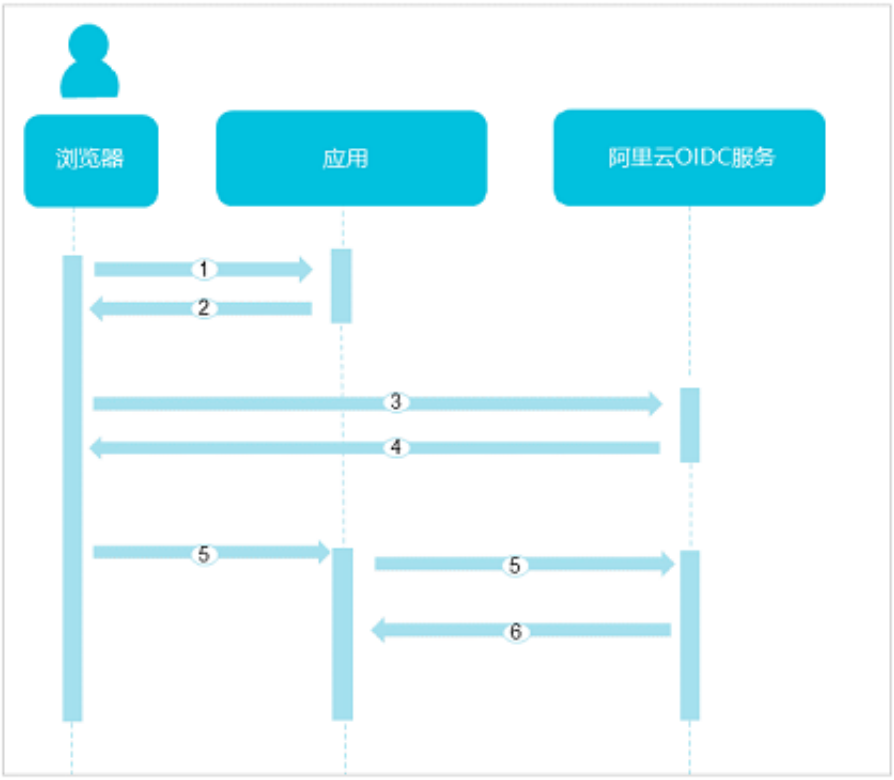
阿里云作为 OIDC 服务提供者，提供了一个 discovery endpoint：`https://oauth.aliyun.com/.well-known/openid-configuration` 来简化配置流程。

Discovery endpoint 包含的内容示例如下：

```
{
  "code_challenge_methods_supported": [
    "plain",
    "S256"
  ],
  "subject_types_supported": [
    "public"
  ],
  "response_types_supported": [
    "code"
  ],
  "issuer": "https://oauth.aliyun.com",
  "jwks_uri": "https://oauth.aliyun.com/v1/keys",
  "revocation_endpoint": "https://oauth.aliyun.com/v1/revoke",
  "token_endpoint": "https://oauth.aliyun.com/v1/token",
  "id_token_signing_alg_values_supported": [
    "RS256"
  ],
  "scopes_supported": [
    "openid",
    "aliuid",
    "profile"
  ],
  "authorization_endpoint": "https://signin.aliyun.com/oauth2/v1/auth"
```

```
}
```

基本流程



1. 用户通过浏览器登录应用。
2. 应用重定向到阿里云 OIDC 服务并将 URL 返回给浏览器。

   说明:
   如果用户还未登录，则会进一步重定向到阿里云登录服务。
3. 用户通过浏览器登录阿里云 OIDC 服务并申请授权码。
4. 阿里云 OIDC 服务重定向到应用并返回授权码。
5. 应用使用授权码向阿里云 OIDC 服务申请身份令牌。

   如何获取身份令牌的签名密钥，请参考：应用获取身份令牌签名密钥。
6. 阿里云 OIDC 服务向应用返回身份令牌，应用通过身份令牌便可以获取用户信息。

   - 在身份令牌用于获取用户信息的场景下，由于身份令牌是由应用直接从 OIDC 服务获取的，因此应用不需要验证令牌的签名。

     用户信息返回示例请参考：用户信息返回示例。
   - 在身份令牌用于应用的各个不同模块之间通信的场景下，为保证信息安全，建议应用接受对身份令牌进行验证。

     如何验证身份令牌，请参考：验证身份令牌的 JWT (JSON Web Token) 签名。

应用获取身份令牌签名密钥

请求示例

```
private List getSignPublicKey() {
  HttpResponse response = HttpClientUtils.doGet("https://oauth.aliyun.
com/v1/keys");
  List rsaKeyList = new ArrayList();
  if (response.getCode() == 200 && response.isSuccess()) {
    String keys = JSON.parseObject(response.getData()).getString("keys
");
    try {
      JSONArray publicKeyList = JSON.parseArray(keys);
      for (Object object : publicKeyList) {
        RSAKey rsaKey = RSAKey.parse(JSONObject.toJSONString(object));
        rsaKeyList.add(rsaKey);
      }
      return rsaKeyList;
    } catch (Exception e) {
      LOG.info(e.getMessage());
    }
  }
  LOG.info("GetSignPublicKey failed:{}", response.getData());
  throw new AuthenticationException(response.getData());
}
```

验证身份令牌的 JWT (JSON Web Token) 签名

阿里云颁发的身份令牌是带有签名的 JWT，签名算法为 JWS 标准 RS256。当应用请求获取用户信息时，阿里云需要对身份令牌进行验证，包含以下几个方面：

- **签名验证：通过 OAuth 服务公布的签名公钥，验证身份令牌的真实性和完整性。**
- **有效期验证：检查令牌颁发时间和令牌过期时间的有效性。**
- **检查令牌接收者：防止颁发给其他应用的身份令牌被传递给本应用。**

请求示例

```
public boolean verifySign(SignedJWT signedJWT) {
  List publicKeyList = getSignPublicKey();
  RSAKey rsaKey = null;
  for (RSAKey key : publicKeyList) {
    if (signedJWT.getHeader().getKeyID().equals(key.getKeyID())) {
      rsaKey = key;
    }
  }
  if (rsaKey != null) {
    try {
      RSASSAVerifier verifier = new RSASSAVerifier(rsaKey.toRSAPubli
cKey());
      if (signedJWT.verify(verifier)) {
        return true;
      }
    } catch (Exception e) {
      LOG.info("Verify exception:{}", e.getMessage());
    }
  }
  throw new AuthenticationException("Can't verify signature for id
token");
```

```
}
```

用户信息返回示例

表 7-13: Header 返回参数

参数名称	描述	需要的 OAuth 范围
alg	签名算法。	openid
kid	验证身份令牌签名使用的公钥，用户需要使用此公钥验证签名，防止身份令牌被篡改。	openid

表 7-14: Body 返回参数

参数名称	描述	需要的 OAuth 范围
exp	令牌过期时间。	openid
sub	令牌主体，登录用户的 UID。	openid
aud	令牌接收者，OAuth 应用 ID。	openid
iss	令牌颁发者。取值为：https://oauth.aliyun.com。	openid
iat	令牌颁发时间。	openid
name	登录用户的显示名称。	profile
upn	登录用户的 UPN（User Principal Name）。	profile
login_name	主账号的登录名。	profile
aid	登录用户所属的阿里云账号 ID。	aliuid
uid	登录用户的阿里云账号 ID。	aliuid

返回示例

返回 Header

```
{
  "alg": "RS256",
  "kid": "JC9wxzrhqJ0gtaCEt2QLUfevEUIwltFhui4O1bh****"
}
```

返回 Body

为了阅读方便，这里展示的是未加密的身份令牌的返回。实际返回为加密的身份令牌，详情请参考：验证身份令牌的 JWT (JSON Web Token) 签名。

- 主账号请求时的返回 Body

```
{
  "exp": 1517539523,
  "sub": "123456789012****",
  "aud": "4567890123456****",
  "iss": "https:\/\/oauth.aliyun.com",//中国站
  "iat": 1517535923,
  "name": "alice", //登录用户的显示名称
  "login_name":"alice@example.com", // 主账号的登录名
  "aid": "123456789012****", //对应的阿里云主账号 ID
  "uid": "123456789012****", //登录用户的阿里云账号 ID
}
```

- RAM 用户请求时的返回 Body

```
{
  "exp": 1517539523,
  "sub": "123456789012****",
  "aud": "4567890123456****",
  "iss": "https:\/\/oauth.aliyun.com",
  "iat": 1517535923,
  "name": "alice", //登录用户的显示名称
  "upn": "alice@example.com",// RAM 用户的登录名
  "aid": "123456789012****", //对应的阿里云主账号 ID
  "uid": "234567890123****", //登录用户的阿里云账号 ID
}
```

更多信息

除了直接获取身份令牌，您也可以在获取访问令牌后通过调用 UserInfo 接口获取用户信息，该接口必须使用访问令牌才能访问，返回信息不加密。

说明:

OIDC 场景下，即只有 openid，aliuid 和 profile 这几个范围的时候，也会返回访问令牌，此时的访问令牌只能用于调用UserInfo接口。

获取用户信息的请求地址：`https://oauth.aliyun.com/v1/userinfo`。

请求示例

```
GET v1/userinfo HTTP/1.1
Host: oauth.aliyun.com
Authorization: Bearer SlAV32hkKG
```

返回 Body

- 主账号请求时的返回 Body

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
{
  "sub": "123456789012****",
  "name": "alice", //登录用户的显示名称
  "login_name":"alice@example.com", // 主账号的登录名
  "aid": "123456789012****", //对应的阿里云主账号 ID
  "uid": "123456789012****", //登录用户的阿里云账号 ID
}
```

- RAM 用户请求时的返回 Body

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "sub": "123456789012****",
  "name": "alice", //登录用户的显示名称
  "upn": "alice@example.com",// RAM 用户的登录名
  "aid": "123456789012****", //对应的阿里云主账号 ID
  "uid": "234567890123****", //登录用户的阿里云账号 ID
}
```

7.3 管理 OAuth 应用

7.3.1 创建应用

您可以通过 OAuth 来创建 Web 应用或 Native 应用，从而获取用户信息或访问阿里云 API。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 OAuth 应用管理。
3. 单击新建应用，填写应用名称和应用显示名称。
4. 选择应用类型。
   - WebApp：指基于浏览器交互的网络应用。
   - NativeApp：指操作系统中运行的本地应用，主要为运行在桌面操作系统或移动操作系统中的应用。
5. 根据需要修改访问令牌有效期时长。

   说明:

   有效期可设置范围为：15 分钟至 3 小时，默认为 3600 秒。
6. 根据需要修改刷新令牌有效期时长。

   说明:

   有效期可设置范围为：2 小时至 1 年，默认为 2592000 秒。
7. 填写回调地址。

8. 单击确认。

7.3.2 查看应用基本信息

本文为您介绍如何查看应用基本信息，包括应用名称、显示名称和应用类型等信息。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 OAuth 应用管理。
3. 在应用名称列表下，单击目标应用名称。
4. 在基本信息区域下，可以查看应用基本信息。

7.3.3 修改应用基本信息

本文为您介绍如何修改应用基本信息，包括应用名称、应用显示名称和令牌有效期等信息。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 OAuth 应用管理。
3. 在应用名称列表下，单击目标应用名称。
4. 在基本信息区域下，单击编辑基本信息。
5. 修改完成后，单击确认。

7.3.4 添加应用范围

OAuth 服务通过为应用添加范围来限定应用扮演用户登录阿里云后的权限。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 OAuth 应用管理。
3. 在应用名称列表下，单击目标应用名称。
4. 在应用 OAuth 范围页签下，单击添加 OAuth 范围。
5. 从左侧范围列表下，勾选需要的范围。

> **说明:**
> `openid`、`aliuid`、`profile` 这几个范围与身份令牌相关，其他范围都与访问令牌相关。

6. 单击确定。

7.3.5 创建应用密钥

如果应用需要访问阿里云 API，需要创建应用密钥用作换取访问令牌时鉴定应用身份的密码。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 OAuth 应用管理。
3. 在应用名称列表下，单击目标应用名称。
4. 在应用密钥页签下，单击创建密钥。
5. 在弹出的对话框中查看应用密钥，单击确认。

> 说明:
> 最多可以创建 2 个应用密钥。应用密钥只在创建时显示，不支持查询，请妥善保管。

7.3.6 删除应用

如果不再需要使用应用获取用户信息或访问阿里云 API，可以删除应用。删除应用后，企业用户将不能正常登录。

操作步骤

1. 登录 RAM 控制台。
2. 在左侧导航栏，单击 OAuth 应用管理。
3. 在应用名称列表下，找到目标应用，单击删除。
4. 单击确认。

7.4 OAuth 常用的 SDK 示例

本文基于 Spring Boot OAuth2 和 Pac4J，为您介绍 OAuth 常用的 SDK 示例的相关配置。

Spring Boot OAuth2 示例

参考 Spring Boot and OAuth2文档及示例，主要做以下两点修改：

- 配置文件改为阿里云对应的配置。

```
aliyun:
    client:
      clientId: 4151950823846923577
      clientSecret: 6EwN4qutnZuchG6n677Ie33SsjAhwyTpcOMSoIo6
v0gqJtw4QcHhERVXfqzcWgMB
      accessTokenUri: https://oauth.aliyun.com/v1/token
      userAuthorizationUri: https://signin.aliyun.com/oauth2/v1/auth
      tokenName: access_token
      authenticationScheme: query
      clientAuthenticationScheme: form
    resource:
```

```
      userInfoUri: https://oauth.aliyun.com/v1/userinfo
```

- **修改重定向 URI。**

  修改`OAuth2ClientAuthenticationProcessingFilter`中的回调地址，改成与应用中的配置相同。 例如：应用配置的是`http://localhost:8080/login/aliyun`，则把代码中的回调地址替换为`/login/aliyun`。

  整体示例代码如下：

```
public class WebApplication extends WebSecurityConfigurerAdapter {

  @Autowired
  OAuth2ClientContext oauth2ClientContext;

  @RequestMapping("/user")
  public Principal user(Principal principal) {
    return principal;
  }

  @Override
  protected void configure(HttpSecurity http) throws Exception {
    // @formatter:off
    http.antMatcher("/**").authorizeRequests().antMatchers("/", "/
login**", "/webjars/**").permitAll().anyRequest()
        .authenticated().and().exceptionHandling()
        .authenticationEntryPoint(new LoginUrlAuthenticationEntryPoi
nt("/")).and().logout()
        .logoutSuccessUrl("/").permitAll().and().csrf()
        .csrfTokenRepository(CookieCsrfTokenRepository.withHttpOn
lyFalse()).and()
        .addFilterBefore(ssoFilter(), BasicAuthenticationFilter.
class);
    // @formatter:on
  }

  public static void main(String[] args) {
    SpringApplication.run(WebApplication.class, args);
  }

  @Bean
  public FilterRegistrationBean oauth2ClientFilterRegistration(
OAuth2ClientContextFilter filter) {
    FilterRegistrationBean registration = new FilterRegistrationBean
();
    registration.setFilter(filter);
    registration.setOrder(-100);
    return registration;
  }

  private Filter ssoFilter() {
    OAuth2ClientAuthenticationProcessingFilter aliyunFilter = new
OAuth2ClientAuthenticationProcessingFilter(
        "/login/aliyun");
    OAuth2RestTemplate aliyunTemplate = new OAuth2RestTemplate(
aliyun(), oauth2ClientContext);
    aliyunFilter.setRestTemplate(aliyunTemplate);
    UserInfoTokenServices tokenServices = new UserInfoTokenServices(
aliyunResource().getUserInfoUri(),
        aliyun().getClientId());
    tokenServices.setRestTemplate(aliyunTemplate);
```

```
    aliyunFilter.setTokenServices(tokenServices);
    return aliyunFilter;
  }

  @Bean
  @ConfigurationProperties("aliyun.client")
  public AuthorizationCodeResourceDetails aliyun() {
    return new AuthorizationCodeResourceDetails();
  }

  @Bean
  @ConfigurationProperties("aliyun.resource")
  public ResourceServerProperties aliyunResource() {
    return new ResourceServerProperties();
  }

}
```

Pac4J 示例

参考 Pac4J 的 spring-webmvc-pac4j 示例，进行以下操作：

- 创建`AliyunOidcClient`。

```
public class AliyunOidcClient extends OidcClient<OidcProfile,
OidcConfiguration> {
    public AliyunOidcClient() {

    }

    public AliyunOidcClient(OidcConfiguration configuration) {
        super(configuration);
    }

    @Override
    protected void clientInit() {
        CommonHelper.assertNotNull("configuration", this.getConfigu
ration());
        this.getConfiguration().defaultDiscoveryURI("https://oauth.
aliyun.com/.well-known/openid-configuration");
        OidcProfileCreator<OidcProfile> profileCreator = new
OidcProfileCreator(this.getConfiguration());
        profileCreator.setProfileDefinition(new OidcProfileDefinitio
n((x) -> {
            return new OidcProfile();
        }));
        this.defaultProfileCreator(profileCreator);
        super.clientInit();
    }
}
```

- 添加`oidcConfig`的bean。

```
<bean id="oidcConfiguration" class="org.pac4j.oidc.config.OidcConfig
uration">
        <property name="clientId" value=your application id />
        <property name="secret" value=your application secret />
        <property name="useNonce" value="false" />
        <property name="scope" value="openid profile aliuid" />
        <property name="clientAuthenticationMethod" value="
client_secret_post" />
```

```
</bean>
```

- 添加aliyunOidcClient 的 bean。

```
<bean id="aliyunOidClient" class="org.pac4j.demo.spring.AliyunOidc
Client">
        <constructor-arg name="configuration" ref="oidcConfiguration
" />
        <property name="authorizationGenerator">
            <bean class="org.pac4j.demo.spring.RoleAdminAuthGenerat
or" />
        </property>
</bean>
```

- 配置bean clients的 callbackUrl 以及 client 属性。

说明:

其中 callbackUri 需要配置在阿里云控制台中应用的回调地址里。

```
<bean id="clients" class="org.pac4j.core.client.Clients">
        <constructor-arg name="callbackUrl" value="http://127.0.0.1:
8080/callback" />
        <constructor-arg name="clients">
            <list>
                <ref bean="aliyunOidClient" />
            </list>
        </constructor-arg>
</bean>
```

- 添加oidc拦截器。

```
<mvc:interceptor>
            <mvc:mapping path="/oidc/*" />
            <bean class="org.pac4j.springframework.web.SecurityIn
terceptor">
                <constructor-arg name="config" ref="config" />
                <constructor-arg name="clients" value="AliyunOidc
Client" />
            </bean>
</mvc:interceptor>
```

- 启动示例，访问http://localhost:8080/oidc/index.html。