ALIBABA CLOUD

Alibaba Cloud

容器镜像服务

用户指南

文档版本：20201015

阿里云

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
危险	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。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会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，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	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。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击**确定**。
`Courier字体`	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	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。	`switch {active|stand}`

目录

1.容器镜像服务默认实例版

1.1. 构建仓库与镜像

当您对命名空间数、私有仓库数、构建规则数等规格要求不高时，建议使用支持基础镜像功能的默认实例版。本文介绍如何为默认实例创建镜像仓库、设置构建规则以及构建镜像。

镜像仓库 构建镜像 镜像构建规则

功能特点

- 代码变更时自动触发构建

  开启**代码变更自动构建镜像**后，每次提交代码将自动触发镜像构建，减少手动触发构建的繁琐工作。

  i. 登录容器镜像服务控制台。
  ii. 在顶部菜单栏，选择所需地域。
  iii. 在左侧导航栏，选择**默认实例 > 镜像仓库**。
  iv. 在**镜像仓库**页面，单击目标仓库右侧操作列的**管理**。
  v. 在左侧导航栏，选择**构建**，然后开启**代码变更自动构建镜像**。

- 海外构建

  代码构建过程中可能会依赖国外源，但由于网络环境，我们提供**海外机器构建**功能。在海外构建完成后，将镜像推送到指定地域的仓库中。

  > **说明** 有时海外回大陆的网络不稳定，可能会导致镜像推送超时失败。

- 不使用缓存

  开启**不使用缓存**后，每次构建都会重新拉取基础依赖镜像，这可能会增加镜像拉取时间，因此建议关闭该选项。

- 多阶段构建

创建镜像仓库

创建镜像仓库前，需要在所需地域下已创建命名空间，详情参见命名空间的基本操作。

1. 登录容器镜像服务控制台。
2. 在顶部下拉菜单中选择所需地域，在左侧导航栏中选择**默认实例 > 镜像仓库**，然后单击**创建镜像仓库**。
3. 在**创建镜像仓库**对话框中，设置命名空间、仓库名称、摘要和仓库类型，本例选择**私有**镜像仓库类型。然后单击**下一步**。
4. 在设置代码源对话框中，将代码源设为**云Code**，然后单击**创建镜像仓库**。
   - **代码变更时自动构建镜像**：勾选后，当分支有代码提交后会自动触发构建规则。
   - **海外机器构建**：勾选后，构建时会在海外机房构建，构建成功后推送到指定地域。

- **不使用缓存**：勾选后，每次构建时会强制重新拉取基础依赖镜像，可能会增加构建时间。

设置构建规则

1. 登录容器镜像服务控制台。
2. 在顶部菜单栏，选择所需地域。
3. 在左侧导航栏，选择**默认实例 > 镜像仓库**。
4. 单击目标仓库右侧操作列中的**管理**，进入仓库详情页面。

5. 单击左侧导航栏中的**构建**，在**构建规则设置**区域的右侧单击**添加规则**。

> **说明** 如需修改构建规则，单击目标规则操作列中的**修改**。

6. 设置构建规则，然后单击**确认**。

- **类型**：设置源代码仓库的类型，可以是 Branch 或 Tag。
- **选择Branch或Tag**：设置构建的代码分支。
- **Dockerfile目录**：设置 Dockerfile 文件所在的目录。这里的目录指的是相对目录，以代码分支的根目录为父目录。
- **Dockerfile文件名**：设置 Dockerfile 文件名，默认为 Dockerfile。
- **镜像版本**：设置镜像 Tag，例如 latest。

构建镜像

1. 在仓库详情页面，单击左侧导航栏中的**构建**。
2. 在**构建规则设置**区域，单击目标规则操作列中的**立即构建**。

构建完成后，生成新的构建记录。

3. 单击右侧操作列表的**日志**按钮，查看构建日志记录。
4. 等待镜像构建完成后，单击左侧菜单栏中的**镜像版本**，查看已构建完成的镜像列表。

5. 如果您需要查看所有镜像，请单击**构建**页签，在**构建设置**中开启**海外机器构建**和**不使用缓存**，默认开启**代码变更自动构建镜像**，可以看到所有镜像版本列表。

后续步骤

- 使用镜像创建无状态Deployment应用
- 使用镜像创建有状态StatefulSet应用
- 使用镜像创建Job类型应用

1.2. 命名空间的基本操作

通过配置命名空间，可以有效管理该命名空间下的仓库集合，包括仓库权限和仓库属性。本文介绍命名空间的基本操作。

命名空间的最佳实践

命名空间作为一些仓库的集合，推荐将一个公司或组织的仓库集中在一个命名空间下面。

- 以公司名称作为命名空间：aliyun、alibaba
- 以团队、组织作为命名空间：misaka-team

命名空间的基本操作

- 创建命名空间
  - 登录容器镜像服务控制台，在左侧导航栏选择**默认实例 > 命名空间**，单击页面右上角**创建命名空间**，输入命名空间名称。
  - 目前一个账号可以创建3个命名空间。
- 命名空间设置
  - 服务目前默认允许用户直接推送镜像，系统自动根据仓库名称创建对应仓库。

    您可以通过将**自动创建仓库**设置为关闭，关闭这一自动创建的功能。
  - 服务目前对于推送镜像自动创建的仓库，默认是私有的。

    您可以将**默认仓库属性**设置为公有，使得自动创建的仓库默认为公有。

1.3. 仓库的主要功能

仓库是镜像的集合，建议将一个应用不同版本的镜像放置在一个仓库中。本文介绍仓库的主要功能。

镜像仓库 仓库功能

仓库的命名

建议以软件包名或应用名作为仓库名称。

- 以软件包命名：例如 centos、jetty
- 以应用命名：例如 console-web、console-service

仓库的主要功能

- 仓库可见性设置
  - 设置为公有仓库，仓库是开放的，允许所有用户匿名下载镜像。
  - 设置为私有仓库，仓库是其他用户不可见的，只有有权限的账户登录才能下载镜像。
- 镜像部署

  通过仓库页面的**部署应用**按钮，可以直接前往容器服务进行部署。
- 仓库镜像查询
  - 列举仓库内的镜像，并获得镜像的 `Digest` 和 `ImageId` 。

- 检查镜像的层信息，查看镜像的每一层大小和每一层的构建元信息。
- 镜像安全扫描功能，对镜像中存在的漏洞进行扫描，并对部分漏洞提供解决方案。

- Webhook
  - 提供仓库镜像的消息触发功能，当镜像上传之后主动触发用户设置的访问地址。
  - 串联镜像服务的下游流程。
- 仓库授权

  支持 RAM 用户粒度控制仓库的访问权限。
- 镜像构建服务
  - 管理用户的源代码仓库，当代码提交后按照用户设置的构建规则构建镜像，并推送到用户仓库。
  - 串联镜像服务的上游流程。

1.4. 仓库访问控制

阿里云权限管理机制包括访问控制（简称RAM）和安全凭证管理（简称STS），灵活使用RAM和STS，可以极大地提高管理的灵活性和安全性。本文介绍如何在不同的场景下配置仓库的访问控制。

镜像仓库 访问控制 RAM 安全凭证

背景信息

默认情况下，主帐号对自己的资源拥有完整的操作权限。借助RAM和STS，可以使不同的子账号拥有访问镜像资源的不同权限，同时也支持为用户提供临时的访问授权。在了解如何配置授权策略前，请先详细阅读RAM产品文档。

系统策略配置

- AliyunContainerRegistryFullAccess

  子用户拥有该授权后，对于镜像资源的权限等同于主账号，可以做任意操作。

```
{
  "Statement": [
    {
      "Action": "cr:*",
      "Effect": "Allow",
      "Resource": "*"
    }
  ],
  "Version": "1"
}
```

- AliyunContainerRegistryReadOnlyAccess

  子用户拥有该授权后，对于所有镜像资源有只读权限，例如：可以查看仓库列表，Pull镜像等。

```
{
 "Statement": [
  {
   "Action": [
    "cr:Get*",
    "cr:List*",
    "cr:PullRepository"
   ],
   "Effect": "Allow",
   "Resource": "*"
  }
 ],
 "Version": "1"
}
```

典型场景策略配置

- 场景1

  场景描述：授权子账号某个命名空间（例如：juzhong）的读权限。子账号登录Registry后pull所有该命名空间下的镜像，可以通过OpenAPI查看到该命名空间的信息及该命名空间下所有镜像仓库的相关信息。

```
{
 "Statement": [
  {
   "Action": [
    "cr:Get*",
    "cr:List*",
    "cr:PullRepository"
   ],
   "Effect": "Allow",
   "Resource": [
    "acs:cr:*:*:repository/juzhong/*"
   ]
  }
 ],
 "Version": "1"
}
```

> 注意 如果同时需要子账号在控制台查看命名空间，需要进行如下授权。子账号可以看到全量的命名空间及仓库列表，但仅能Pull其中juzhong这个命名空间下的仓库。

```
{
 "Statement": [
  {
   "Action": [
    "cr:Get*",
    "cr:List*",
    "cr:PullRepository"
   ],
   "Effect": "Allow",
   "Resource": [
    "acs:cr:*:*:repository/juzhong/*"
   ]
  },
  {
   "Action": [
    "cr:ListNamespace",
    "cr:ListRepository"
   ],
   "Effect": "Allow",
   "Resource": [
    "*"
   ]
  }
 ],
 "Version": "1"
}
```

- 场景2

  场景描述：授权子账号某个镜像仓库（例如：镜像仓库名为nginx，所属命名空间名为juzhong，所属地域为华东1）的所有权限。

  > 注意 如需通过子账号在控制台上管理镜像仓库，依然需要参考场景1配置。

```
{
 "Statement": [
  {
   "Action": [
    "cr:*"
   ],
   "Effect": "Allow",
   "Resource": [
    "acs:cr:cn-hangzhou:*:repository/juzhong/nginx"
   ]
  },
  {
   "Action": [
    "cr:Get*",
    "cr:List*"
   ],
   "Effect": "Allow",
   "Resource": [
    "acs:cr:*:*:repository/juzhong"
   ]
  }
 ],
 "Version": "1"
}
```

- 场景3

  场景描述：授权子账号某命名空间的所有操作权限 。

  > 注意 此场景仅可以通过OpenAPI调用。如果需要同时在控制台看到所有仓库，请参照场景1。

```
{
  "Statement": [
    {
      "Action": [
        "cr:*"
      ],
      "Effect": "Allow",
      "Resource": [
        "acs:cr:cn-hangzhou:*:repository/juzhong",
        "acs:cr:cn-hangzhou:*:repository/juzhong/*"
      ]
    }
  ],
  "Version": "1"
}
```

RAM说明

在使用RAM对子账号授权时，请特别关注下面的说明，以免您为子账号授予过大的权限。

如果您通过RAM为某一个子账号授予阿里云所有资源的管理权限（即AdministratorAccess），无论您之前是否为该子账号授予过镜像服务的权限，该子账号都将拥有对镜像服务的全部权限。

镜像服务鉴权规则

- 资源描述

  在通过RAM进行授权时，资源的描述方式如下表所示：

资源类型	授权策略中的资源描述
repository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositorynameacs:cr:$regionid:$accountid:repository/$namespacename/*acs:cr:$regionid:$accountid:repository/$namespacename

  参数说明如下表所示：

参数名称	说明
$regionid	地域ID，可用*代替。
$accountid	云账号数字ID，可用 * 代替。
$namespacename	命名空间名称。
$repositoryname	镜像仓库名称。

- 鉴权规则

  子账号或者STS方式访问镜像服务API时，镜像服务会向RAM进行权限检查，以确保调用者拥有相应权限。每个API会根据涉及到的资源以及API的语义来确定需要检查哪些资源的权限。每个API的鉴权规则如下表所示：

API	鉴权Action	鉴权Resource
创建命名空间	cr:CreateNamespace	*
删除命名空间	cr:DeleteNamespace	acs:cr:$regionid:$accountid:repository/$namespacename
更新命名空间信息	cr:UpdateNamespace	acs:cr:$regionid:$accountid:repository/$namespacename
获取指定命名空间	cr:GetNamespace	acs:cr:$regionid:$accountid:repository/$namespacename
获取命名空间列表	cr:ListNamespace	*
创建仓库	cr:CreateRepository	acs:cr:$regionid:$accountid:repository/$namespacename
删除仓库	cr:DeleteRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
更新仓库信息	cr:UpdateRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
查询仓库信息	cr:GetRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
查询仓库列表信息	cr:ListRepository	*
根据命名空间查询仓库列表信息	cr:ListRepository	*
查询仓库标签信息	cr:ListRepositoryTag	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
删除镜像版本	cr:DeleteRepositoryTag	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
查询镜像Manifest信息	cr:GetRepositoryManifest	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
查询镜像层信息	cr:GetRepositoryLayers	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname

API	鉴权Action	鉴权Resource
获取临时Token	cr:GetAuthorizationToken	*
Pull镜像	cr:PullRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Push镜像	cr:PushRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname

1.5. 触发器管理

阿里云镜像服务为每个仓库提供了触发器的功能，帮助用户在镜像构建成功后进行消息的推送，实现持续集成的后续流程。如果用户设置了一个容器服务的触发器，那么当镜像构建成功后，将会自动触发容器服务上应用拉取新的镜像，并进行重新部署。本文介绍如何使用触发器。

触发器安全规则

- HTTP服务：默认使用80端口。

  如果要使用其他端口，请在触发器URL末尾手动添加端口号，而且只能选择以下端口号：80、21、443、70、210、280、488、591、777、1025-65535。

- HTTPS服务：默认使用443端口。

  只支持默认的443端口，如果要使用其他端口号，请使用HTTP服务。

触发条件

容器镜像服务ACR提供了两种不同的方式来设置触发器的触发条件：表达式触发和Tag触发。之前未设置触发条件的触发器会默认为全部触发类型。

- 表达式触发：基于正则表达式来进行Tag的过滤，只有当符合正则表达式的Tag才会继续触发触发器。

  可以填写简单的正则表示，如 `release-v.*` ，则只有Tag为 `release-v` 开头的镜像版本在构建后，才会触发后续持续集成的流程；否则为未触发状态，访问记录中访问状态码将显示“未触发”。

  单击**访问记录**，可以查看触发器的历史访问详情。

- Tag触发：基于用户筛选的Tag列表来进行触发。

  可以在列表中筛选需要触发的Tag，最多可选10个，则只有当Tag在列表中时，才会在镜像构建后触发触发器；否则为未触发状态，访问记录中访问状态码将显示“未触发”。

  单击**访问记录**，可以查看触发器的历史访问详情。

通知内容

触发器的通知内容包含了镜像仓库的信息，以及镜像的版本信息，如下所示。其中，镜像仓库信息包括了仓库的Namespace、Name以及仓库对应的Region等。

```
POST /payload HTTP/1.1

Content-Type: application/json
Request URL: https://cs.console.aliyun.com/hook/trigger?triggerUrl=YzRmMWE5YzM2ZjMzYzQ0NmFiM
GYzNWJlMmM2MjM2NzIyfGV4cHJlc3N8cmVkZXBsb3l8MThlMmllY2drdXYyZXw=&secret=365a4a664b456154
38716a487a75695a7ac48329224b35b073c2197374e7d62a
Request method: POST

{
  "push_data": {
    "digest": "sha256:457f4aa83fc9a6663ab9d1b0a6e2dce25a12a943ed5bf2c1747c58d48bbb4917",
    "pushed_at": "2016-11-29 12:25:46",
    "tag": "latest"
  },
  "repository": {
    "date_created": "2016-10-28 21:31:42",
    "name": "repoTest",
    "namespace": "namespace",
    "region": "cn-hangzhou",
    "repo_authentication_type": "NO_CERTIFIED",
    "repo_full_name": "namespace/repoTest",
    "repo_origin_type": "NO_CERTIFIED",
    "repo_type": "PUBLIC"
  }
}
```

1.6. 安全扫描容器镜像

容器镜像服务ACR（Container Registry）支持所有基于Linux的容器镜像安全扫描，可以识别镜像中所有已知的漏洞信息。您可以收到相应的漏洞信息评估和相关的漏洞修复建议。

背景信息

在云原生交付链，ACR能在推送完成后自动进行安全扫描。若您设置过安全阻断策略，其会识别镜像的安全风险并阻断高风险的容器镜像。通过安全策略的容器镜像才会进行交付链后续的分发和部署环节。交付链能保证容器应用的安全交付和高效部署。您也可以集成安全扫描的相关API，实现自定义周期的镜像安全扫描功能。

安全扫描的时长主要取决于镜像的大小。一般情况下扫描一个镜像可以在三分钟之内完成。

操作步骤

1. 登录容器镜像服务控制台。

2. 在控制台页面左上角选择容器镜像所在地域。

   选择地域

3. 单击**默认实例 > 镜像仓库**，选择所需的镜像仓库或单击右侧的**管理**，进入**仓库信息**管理页面。
4. 在左侧导航栏中选择**镜像版本**，单击右侧操作列中的**安全扫描**。

   镜像安全扫描

执行结果

镜像安全扫描完成后，您可以看到安全漏洞详细信息，如下图所示。

镜像安全扫描结果

扫描结果按照**高危**、**中危**、**低危**、**未评级**四个漏洞等级汇总漏洞信息，并且会展示所有漏洞的具体信息以及相应的漏洞修复版本提示。