

阿里云 Go SDK

Go SDK

文档版本：20190221

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 。
<code>courier</code> 字体	命令。	执行 <code>cd /d C:/windows</code> 命令，进入Windows系统文件夹。
##	表示参数、变量。	<code>bae log list --instanceid Instance_ID</code>
[]或者[a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ }或者{a b}	表示必选项，至多选择一个。	<code>swich {stand slave}</code>

目录

法律声明.....	I
通用约定.....	I
1 快速开始.....	1
2 常见错误.....	3

1 快速开始

欢迎使用阿里云开发者工具套件（SDK）。阿里云Go SDK让您不用复杂编程即可访问云服务器、云数据库RDS、云监控等多个阿里云服务。本操作介绍如何获取阿里云Go SDK并开始调用。

如果您在使用SDK的过程中遇到任何问题，欢迎前往阿里云SDK[问答社区](#)提问，提问前请阅读提问引导[#unique_4](#)。

在线调试和生成SDK示例

[OpenAPI Explorer](#)提供在线调用云产品API、动态生成 SDK示例代码和快速检索接口等功能，能显著降低使用API 的难度，推荐您使用。

环境准备

- 使用阿里云Go SDK，您需要一个RAM账号以及一对AccessKey ID和AccessKey Secret。请在阿里云控制台中的[AccessKey管理页面](#)上创建和查看您的AccessKey，或联系您的系统管理员。
- 使用阿里云SDK调用某个产品的API前，确保您已经在阿里云控制台开通了该产品。

安装阿里云 Go SDK

执行以下命令，安装阿里云Go SDK：

```
glide get github.com/aliyun/alibaba-cloud-sdk-go
```

使用Go SDK

以下这个代码示例展示了调用阿里云Go SDK的3个主要步骤：

1. 创建Client实例。
2. 创建API请求并设置参数。
3. 发起请求并处理应答或异常。

```
package main

import (
    "github.com/aliyun/alibaba-cloud-sdk-go/services/ecs"
    "github.com/aliyun/alibaba-cloud-sdk-go/sdk/requests"
    "fmt"
)

func main() {
    // 创建ecsClient实例
    ecsClient, err := ecs.NewClientWithAccessKey(
        "<your-region-id>",           // 地域ID
        "<your-access-key-id>",       // 您的Access Key ID
        "<your-access-key-secret>"    // 您的Access Key Secret
    )
    if err != nil {
        fmt.Println("Error creating ECS client:", err)
        return
    }

    // 创建API请求并设置参数
    req := requests.NewRequest()
    req.Method = "GET"
    req.Path = "/api/v1/instances"
    req.QueryParams = map[string]string{
        "RegionId": "<your-region-id>",
        "AccessKeyId": "<your-access-key-id>",
        "AccessKeySecret": "<your-access-key-secret>",
    }

    // 发起请求并处理应答或异常
    resp, err := ecsClient.GetInstances(req)
    if err != nil {
        fmt.Println("Error calling GetInstances:", err)
        return
    }

    // 处理应答
    fmt.Println("Instances:", resp)
}
```

```
    if err != nil {
        // 异常处理
        panic(err)
    }
    // 创建API请求并设置参数
    request := ecs.CreateDescribeInstancesRequest()
    // 等价于 request.PageSize = "10"
    request.PageSize = requests.NewInteger(10)
    // 发起请求并处理异常
    response, err := ecsClient.DescribeInstances(request)
    if err != nil {
        // 异常处理
        panic(err)
    }
    fmt.Println(response)
}
```

2 常见错误

您可以参考本文档中提供的常见问题排查SDK使用中的问题。

错误码	错误信息	原因	解决方法
SDK. CanNotResolveEndpoint	Can not resolve endpoint, please check the user guide	SDK无法自动获取调用的产品在特定region的endpoint（域名）。	检查提供的地域ID和endpoint是否正确。获取endpoint后，可以通过以下代码设定endpoint： <pre>request.Domain = "<endpoint>"</pre>
SDK. JsonUnmarshalError	Failed to unmarshal response, but you can get the data via response.GetHttpStatusCode() and response.GetHttpContentString()	SDK反序列化应答失败，大多情况下是由于SDK实际收到的应答结构与ApiMeta不符，例如字段不匹配，格式不正确等原因导致的。	应答解析
SDK. TimeoutError	The request timed out 4 times(3 for retry), perhaps we should have the threshold raised a little?	请求超时，并且所有重试均失败。	· 如果存在跨地域调用或其他低质量网络场景，建议考虑适当延长超时时间或最大重试次数。 · 如果频繁出现这个问题，且排除低质量网络场景的情况下，建议您提交工单。

错误码	错误信息	原因	解决方法
SDK. AsyncFunctionNotEnabled	Async function is not enabled in client, please invoke 'client.EnableAsync()' function	在未开启异步请求开关的情况下，调用了异步接口。	开启异步功能即可，详情参见 异步调用 。

应答解析

在SDK反序列化失败时，您可以忽略，通过提取response类中的原始HTTP应答，依然可以获取应答中的信息。



说明:

阿里云Go SDK依赖了jmespath，因此当应答内容为JSON时，您可以直接使用jmespath来解析应答。

```
response, err := client.CreateInstance(request)
if err != nil {
    // 异常处理
    if clientErr, isClientErr := err.(*errors.ClientError);
isClientErr {
    if clientErr.ErrorCode() == errors.JsonUnmarshalErrorCode
    {
        // 您可以通过我们的封装方法来分别获取原始应答中的数据
        // 应答中的http状态码，类型: int
        statusCode := response.GetHttpStatus()
        // 应答中的http头，类型: map[string][]string
        responseHeader := response.GetHttpHeaders()
        // 应答中的body内容，类型: []byte
        responseBytes := response.GetHttpContentBytes()
        // 应答中的body内容，类型: string，等价于`string(response.
        GetHttpContentBytes())`
        responseContent := response.GetHttpContentString()
        // 您也可以直接获取原始应答，类型: net/http.Response
        httpResponse := response.GetOriginHttpResponse()
    }
}
}
```