

# 阿里云 应用实时监控服务 ARMS

API 参考

文档版本：20200630

# 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. 阿里云文档中所有内容，包括但不限于图片、架构设计、页面布局、文字描述，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。	 <b>禁止：</b> 重置操作将丢失用户配置数据。
	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。	 <b>警告：</b> 重启操作将导致业务中断，恢复业务时间约十分钟。
	用于警示信息、补充说明等，是用户必须了解的内容。	 <b>注意：</b> 权重设置为0，该服务器不会再接受新请求。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 <b>说明：</b> 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	单击 <b>设置 &gt; 网络 &gt; 设置网络类型</b> 。
<b>粗体</b>	表示按键、菜单、页面名称等UI元素。	在 <b>结果确认</b> 页面，单击 <b>确定</b> 。
Courier字体	命令。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	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>switch {active stand}</code>

# 目录

---

法律声明.....	I
通用约定.....	I
<b>1 API概览.....</b>	<b>1</b>
<b>2 公共参数.....</b>	<b>2</b>
<b>3 安装 ARMS SDK.....</b>	<b>4</b>
<b>4 应用监控.....</b>	<b>6</b>
4.1 GetTrace.....	6
4.2 SearchTraceAppByName.....	9
4.3 SearchTraceAppByPage.....	11
4.4 SearchTraces.....	14
4.5 ListTraceApps.....	16
4.6 QueryMetric（应用监控）.....	19
<b>5 前端监控.....</b>	<b>27</b>
5.1 CreateRetcodeApp.....	27
5.2 ListRetcodeApps.....	28
5.3 SearchRetcodeAppByPage.....	31
5.4 DeleteRetcodeApp.....	33
5.5 SetRetcodeShareStatus.....	35
5.6 GetRetcodeShareUrl.....	36
5.7 QueryMetric（前端监控）.....	38
<b>6 Prometheus监控.....</b>	<b>48</b>
6.1 AddGrafana.....	48
6.2 AddIntegration.....	49
6.3 GetPrometheusApiToken.....	50
6.4 ListDashboards.....	52
<b>7 自定义监控.....</b>	<b>55</b>
7.1 QueryDataset（下钻数据集）.....	55
7.2 QueryDataset（通用数据集）.....	58
<b>8 报警.....</b>	<b>61</b>
8.1 CreateAlertContact.....	61
8.2 CreateAlertContactGroup.....	62
8.3 SearchAlertContactGroup.....	64
8.4 ImportAppAlertRules.....	66
8.5 SearchAlertContact.....	74
<b>9 附录.....</b>	<b>78</b>
9.1 如何通过内网调用 API.....	78

# 1 API概览

应用实时监控服务提供以下相关API接口。

## 应用监控

API	描述
<a href="#">SearchTraceAppByName</a>	调用 SearchTraceAppByName 查询应用监控任务列表。
<a href="#">SearchTraceAppByPage</a>	调用 SearchTraceAppByPage 分页查询应用监控任务。
<a href="#">ListTraceApps</a>	调用 ListTraceApps 列出指定地域下的全部应用监控任务。

## 前端监控

API	描述
<a href="#">CreateRetcodeApp</a>	调用 CreateRetcodeApp 创建前端监控任务。
<a href="#">ListRetcodeApps</a>	调用 ListRetcodeApps 列出指定地域下全部前端监控任务。
<a href="#">SearchRetcodeAppByPage</a>	调用 SearchRetcodeAppByPage 分页查询前端监控任务。
<a href="#">DeleteRetcodeApp</a>	调用 DeleteRetcodeApp 删除前端监控任务。

## 报警

API	描述
<a href="#">CreateAlertContact</a>	使用 CreateAlertContact 创建报警联系人。
<a href="#">CreateAlertContactGroup</a>	调用 CreateAlertContactGroup 创建联系人分组。
<a href="#">SearchAlertContactGroup</a>	调用 SearchAlertContactGroup 查询报警联系人分组。
<a href="#">ImportAppAlertRules</a>	调用 ImportAppAlertRules 导入应用报警规则。
<a href="#">SearchAlertContact</a>	调用 SearchAlertContact 查询报警联系人。

## 2 公共参数

本文列出了 ARMS API 的公共参数和接入地址。



### 说明：

POP 网关是面对公网环境提供服务的，如果不能访问公网，则无法连接服务。

### 公共请求参数

参数	说明
<b>region</b>	API 网关所在地域。支持的地域请参见 <a href="#">接入地址</a> 。
<b>accessKeyId/accessKeySecret</b>	<ul style="list-style-type: none"><li>如果您以阿里云账号（主账号）或 RAM 用户（子账号）调用 API，则此参数为阿里云账号或 RAM 用户的 <b>accessKeyId/accessKeySecret</b>。</li><li>如果您以 RAM 角色调用 API，则此参数为您获取的 STS 安全令牌中的 <b>accessKeyId/accessKeySecret</b>。详情参见<a href="#">#unique_17</a>。</li></ul>
<b>endpoint</b>	接入点名称与 <b>region</b> 保持一致，例如 cn-hangzhou。
<b>productName</b>	API 的产品名称，填写 ARMS 即可。
<b>domain</b>	值为 arms.[region].aliyuncs.com，例如华东 1（杭州）地域为 arms.cn-hangzhou.aliyuncs.com，请参见 <a href="#">接入地址</a> 。

### 接入地址

为减少网络延时，建议您根据业务调用来源配置接入地址。ARMS API 的服务接入地址，参见以下表格。

地域名称（部署地位置）	接入地址
华东 1（杭州）	arms.cn-hangzhou.aliyuncs.com
华东 2（上海）	arms.cn-shanghai.aliyuncs.com
华北 1（青岛）	arms.cn-qingdao.aliyuncs.com
华北 2（北京）	arms.cn-beijing.aliyuncs.com
华南 1（深圳）	arms.cn-shenzhen.aliyuncs.com
华北 3（张家口）	arms.cn-zhangjiakou.aliyuncs.com
中国（香港）	arms.cn-hongkong.aliyuncs.com
新加坡（新加坡）	arms.ap-southeast-1.aliyuncs.com

**说明:**

- API 对单 IP 调用次数限制为 100 次/秒。
- 由于 ARMS 提供的接入地址（Endpoint）为公网服务地址，当您的 ECS 实例没有分配公网带宽或者不存在公网 IP 地址时，无法使用阿里云 CLI 或 SDK 等工具发起 API 请求。如果您的请求端为专有网络 VPC 类型的无公网访问的 ECS 实例，并有内网调用需求，请参见[如何通过内网调用 API](#)。

**相关文档**[#unique\\_17](#)

## 3 安装 ARMS SDK

ARMS 提供 Java、Python 和 PHP 的 SDK，本文介绍安装方法。

### 安装 Java SDK

要安装 ARMS Java SDK，只需在 pom.xml 文件中添加 Maven 依赖即可。

```
<dependencies>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>aliyun-java-sdk-arms</artifactId>
    <version>2.5.9</version>
  </dependency>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>aliyun-java-sdk-core</artifactId>
    <version>3.5.0</version>
  </dependency>
</dependencies>
```

### 安装 Python SDK

要安装 ARMS Python SDK，请运行以下命令：

```
pip install aliyun-python-sdk-arms
```

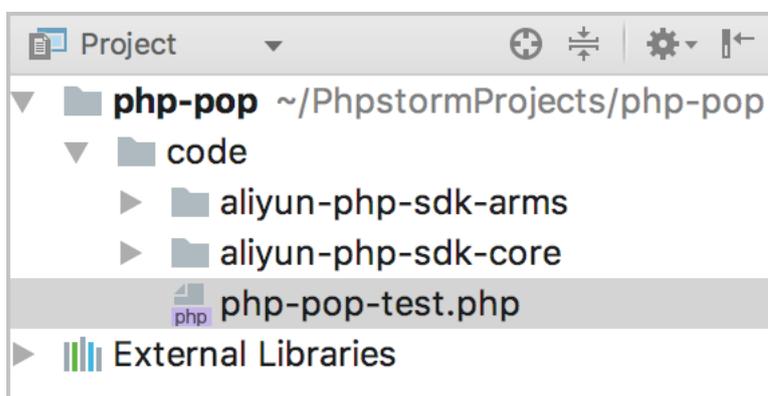
### 安装 PHP SDK

请按照以下步骤安装 PHP SDK。

1. 运行以下命令，将 PHP SDK 克隆到本地的 aliyun-openapi-php-sdk 文件夹。

```
git clone https://github.com/aliyun/aliyun-openapi-php-sdk
```

2. 将 aliyun-openapi-php-sdk 下的 aliyun-php-sdk-arms 和 aliyun-php-sdk-core 文件夹复制到您的 PHP 项目目录，目录结构如下图所示。



[MetricQuery 接口（应用监控）](#)

[MetricQuery 接口（前端监控）](#)

[下钻数据集查询接口](#)

[通用数据集查询接口](#)

## 4 应用监控

### 4.1 GetTrace

调用GetTrace接口获取调用链详情。

#### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

#### 请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetTrace	系统规定参数。取值：GetTrace。
RegionId	String	是	cn-hangzhou	地域ID。
TraceID	String	是	swg-1231-xsdad-dddasds	调用链ID。

#### 返回数据

名称	类型	示例值	描述
RequestId	String	626037F5-FDEB-45B0-804C-B3C92797A64E	请求 ID
Spans	Array		调用链路详情信息
Duration	Long	10000	调用链路耗时，单位是微秒。
HaveStack	Boolean	false	是否有方法栈
LogEventList	Array		调用链路中的日志事件
TagEntryList	Array		返回的Tag列表
Key	String	http.status.code	Tag Key
Value	String	200	Tag Value

名称	类型	示例值	描述
Timestamp	Long	1590388651	13位时间戳
OperationName	String	/api/demo	埋点的接口名称
ResultCode	String	222	返回码
RpcId	String	swg-xsdf-12fscf-qddf	RPC ID
RpcType	Integer	1	RPC类型
ServiceIp	String	172.20.XX.XX	应用所在的机器IP地址
ServiceName	String	arms-demo	应用名称
TagEntryList	Array		调用链路的Tag列表
Key	String	xxx	Tag Key
Value	String	aaaaaaaa	Tag Value
Timestamp	Long	1590388651	时间戳
TraceID	String	swg-1231-xsdad-dddasds	调用链路ID

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/trace/Trace.json?RegionId=cn-hangzhou
&TraceID=swg-1231-xsdad-dddasds
&<公共请求参数>
```

### 正常返回示例

#### XML 格式

```
<RequestId>626037F5-FDEB-45B0-804C-B3C92797A64E</RequestId>
<Spans>
  <HaveStack>>false</HaveStack>
  <ServiceIp>123.12.12.12</ServiceIp>
  <OperationName>getXxx</OperationName>
  <ServiceName>service-01</ServiceName>
  <RpcId>swg-xsdf-12fscf-qddf</RpcId>
  <RpcType>1</RpcType>
```

```
<TraceID>swg-1231-xsdad-dddasds</TraceID>
<Duration>10000</Duration>
<Timestamp>1590388651</Timestamp>
<ResultCode>222</ResultCode>
<TagEntryList>
  <Value>aaaaaaaa</Value>
  <Key>xxx</Key>
</TagEntryList>
<LogEventList>
  <Timestamp>1590388651</Timestamp>
  <TagEntryList>
    <Value>yyyyyyyyyy</Value>
    <Key>xxx</Key>
  </TagEntryList>
</LogEventList>
</Spans>
```

### JSON 格式

```
{
  "RequestId": "626037F5-FDEB-45B0-804C-B3C92797A64E",
  "Spans": [
    {
      "HaveStack": "false",
      "ServiceIp": "123.12.12.12",
      "OperationName": "getXxx",
      "ServiceName": "service-01",
      "RpcId": "swg-xsdf-12fscf-qddf",
      "RpcType": "1",
      "TraceID": "swg-1231-xsdad-dddasds",
      "Duration": "10000",
      "Timestamp": "1590388651",
      "ResultCode": "222",
      "TagEntryList": [
        {
          "Value": "aaaaaaaa",
          "Key": "xxx"
        }
      ],
      "LogEventList": [
        {
          "Timestamp": "1590388651",
          "TagEntryList": [
            {
              "Value": "yyyyyyyyyy",
              "Key": "xxx"
            }
          ]
        }
      ]
    }
  ]
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 4.2 SearchTraceAppByName

调用 SearchTraceAppByName 查询应用监控任务列表。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
RegionId	String	是	cn-hangzhou	地域 ID
TraceAppName	String	是	demo	应用名称

### 返回数据

名称	类型	示例值	描述
RequestId	String	514078F3-3DA0-4180-95A0-D8F300CB71D2	地域 ID
TraceApps	Array		应用监控信息
AppId	Long	6549	应用 ID
Pid	String	atc889zkcf@d8deedfa9bfxxxx	应用的 ID 标识串
AppName	String	3123-docker-demo	应用名称
Type	String	TRACE	监控类型。 <b>TRACE</b> 为应用监控， <b>RETCODE</b> 为前端监控。
UserId	String	12341234	用户 ID
RegionId	String	cn-hangzhou	地域 ID
CreateTime	Long	1531291867000	创建时间
UpdateTime	Long	1531291867000	更新时间

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/trace/SearchTraceAppByName.json?
RegionId=cn-hangzhou
&TraceAppName=demo
```

### 正常返回示例

#### XML 格式

```
<SearchTraceAppByName>
  <requestId>514078F3-3DA0-4180-95A0-D8F300CB71D2</requestId>
  <traceApps>
    <traceApp>
      <appId>6549</appId>
      <appName>3123-docker-demo</appName>
      <createTime>1531291867000</createTime>
      <pid>xxxxxxxxxxxxxxxxxxxx</pid>
      <regionId>cn-hangzhou</regionId>
      <type>TRACE</type>
      <updateTime>1531291867000</updateTime>
      <userId>xxxxxxxxxxx</userId>
    </traceApp>
    <traceApp>
      <appId>10198</appId>
      <appName>13312-demo</appName>
      <createTime>1540189384000</createTime>
      <pid>xxxxxxxxxxxxxxxxxxxx</pid>
      <regionId>cn-hangzhou</regionId>
      <type>tRACE</type>
      <updateTime>1540189384000</updateTime>
      <userId>xxxxxxxxxxx</userId>
    </traceApp>
    <traceApp>
      <appId>128985</appId>
      <appName>dubboDemoConsumer</appName>
      <createTime>1565147950000</createTime>
      <pid>xxxxxxxxxxxxxxxxxxxx</pid>
      <regionId>cn-hangzhou</regionId>
      <type>tRACE</type>
      <updateTime>1565147950000</updateTime>
      <userId>xxxxxxxxxxx</userId>
    </traceApp>
  </traceApps>
</SearchTraceAppByName>
```

#### JSON 格式

```
{
  "requestId": "514078F3-3DA0-4180-95A0-D8F300CB71D2",
  "traceApps": [
    {
      "appId": 6549,
      "pid": "xxxxxxxxxxxxxxxxxxxx",
      "appName": "3123-docker-demo",
      "type": "TRACE",
      "userId": "xxxxxxxxxxx",
      "regionId": "cn-hangzhou",
```

```

    "createTime": 1531291867000,
    "updateTime": 1531291867000
  },
  {
    "appId": 10198,
    "pid": "xxxxxxxxxxxxxxxx",
    "appName": "13312-demo",
    "type": "TRACE",
    "userId": "xxxxxxxxxxxx",
    "regionId": "cn-hangzhou",
    "createTime": 1540189384000,
    "updateTime": 1540189384000
  },
  {
    "appId": 128985,
    "pid": "xxxxxxxxxxxxxxxx",
    "appName": "dubboDemoConsumer",
    "type": "TRACE",
    "userId": "xxxxxxxxxxxx",
    "regionId": "cn-hangzhou",
    "createTime": 1565147950000,
    "updateTime": 1565147950000
  }
]
}

```

### 错误码

访问[错误中心](#)查看更多错误码。

## 4.3 SearchTraceAppByPage

调用 SearchTraceAppByPage 分页查询应用监控任务。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
<b>Action</b>	String	是	SearchTraceAppByPage	系统规定参数。取值：SearchTraceAppByPage。
<b>PageNumber</b>	Integer	是	1	当前查询页码
<b>PageSize</b>	Integer	是	2	每页数据大小
<b>RegionId</b>	String	是	cn-hangzhou	地域 ID

名称	类型	是否必选	示例值	描述
TraceAppName	String	是	Demo	应用名称

### 返回数据

名称	类型	示例值	描述
RequestId	String	18B01385-A3B9-4BCF-988A-2064D77CF2C7	请求 ID
PageBean	Struct		返回页面信息
TotalCount	Integer	22	查询结果总计
PageNumber	Integer	1	当前查询页码
PageSize	Integer	2	每页数据行数
TraceApps	Array		应用监控信息
AppId	Long	6549	应用 ID，数据库自增字段。
Pid	String	atc889zkcfd8deedfa9bfxxxx	应用的 ID 标识串
AppName	String	4123-docker-demo	应用监控名称
Type	String	TRACE	监控类型。 <b>TRACE</b> 为应用监控， <b>RETCODE</b> 为前端监控。
UserId	String	12341234	用户 ID
RegionId	String	cn-hangzhou	地区 ID
CreateTime	Long	1531291867000	创建时间
UpdateTime	Long	1531291867000	更新时间

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/trace/SearchTraceAppByPage.json?
pageNumber=1
&pageSize=2
&regionId=cn-hangzhou
&traceAppName=Demo
```

### 正常返回示例

### XML 格式

```
<SearchTraceAppByPage>
  <RequestId>18B01385-A3B9-4BCF-988A-2064D77CF2C7</RequestId>
  <PageBean>
    <PageNumber>1</PageNumber>
    <PageSize>2</PageSize>
    <TotalCount>22</TotalCount>
    <TraceApps>
      <TraceApp>
        <AppId>6549</AppId>
        <AppName>4123-docker-demo</AppName>
        <CreateTime>1531291867000</CreateTime>
        <Pid>xxxxxxxxxxxxxxxxxxxx</Pid>
        <RegionId>cn-hangzhou</RegionId>
        <Type>TRACE</Type>
        <UpdateTime>1531291867000</UpdateTime>
        <UserId>xxxxxxxxxxxx</UserId>
      </TraceApp>
      <TraceApp>
        <AppId>10198</AppId>
        <AppName>132-k8s-demo</AppName>
        <CreateTime>1540189384000</CreateTime>
        <Pid>xxxxxxxxxxxxxxxxxxxx</Pid>
        <RegionId>cn-hangzhou</RegionId>
        <Type>TRACE</Type>
        <UpdateTime>1540189384000</UpdateTime>
        <UserId>xxxxxxxxxxxx</UserId>
      </TraceApp>
    </TraceApps>
  </PageBean>
</SearchTraceAppByPage>
```

### JSON 格式

```
{
  "requestId": "18B01385-A3B9-4BCF-988A-2064D77CF2C7",
  "pageBean": {
    "totalCount": 22,
    "pageNumber": 1,
    "pageSize": 2,
    "traceApps": [
      {
        "appId": 6549,
        "pid": "xxxxxxxxxxxxxxxxxxxx",
        "appName": "4123-docker-demo",
        "type": "TRACE",
        "userId": "xxxxxxxxxxxx",

```

```

    "regionId": "cn-hangzhou",
    "createTime": 1531291867000,
    "updateTime": 1531291867000
  },
  {
    "appId": 10198,
    "pid": "xxxxxxxxxxxxxxxxxxxx",
    "appName": "132-k8s-demo",
    "type": "TRACE",
    "userId": "xxxxxxxxxxxx",
    "regionId": "cn-hangzhou",
    "createTime": 1540189384000,
    "updateTime": 1540189384000
  }
]
}

```

### 错误码

访问[错误中心](#)查看更多错误码。

## 4.4 SearchTraces

调用SearchTraces接口查询调用链列表信息，可根据时间、应用名称、IP地址、Span名称和Tag等信息对调用链进行过滤查询。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
<b>Action</b>	String	是	SearchTraces	系统规定参数。取值：SearchTraces。
<b>EndTime</b>	Long	是	1591255112346	结束时间，精确到ms。
<b>RegionId</b>	String	是	cn-hangzhou	地域ID
<b>StartTime</b>	Long	是	1591255112345	开始时间，精确到ms。
<b>ServiceName</b>	String	否	arms-demo	应用名称
<b>OperationName</b>	String	否	/api	埋点的接口名称

名称	类型	是否必选	示例值	描述
<b>MinDuration</b>	Long	否	1200	最小耗时，单位为ms。
<b>Tag.N.Key</b>	String	否	http.status_code	Tag Key
<b>Tag.N.Value</b>	String	否	200	Tag Value
<b>Reverse</b>	Boolean	否	false	按照时间正序或者倒序排列。取值为： <ul style="list-style-type: none"> <li>• true：表示倒序。</li> <li>• false：表示正序。</li> </ul> 默认值为false。
<b>ServiceIp</b>	String	否	172.20.XX.XX	应用所在的机器IP地址

### 返回数据

名称	类型	示例值	描述
RequestId	String	626037F5-FDEB-45B0-804C-B3C92797A64E	请求ID
TraceInfos	Array		返回的调用链路详细信息
Duration	Long	1200	耗时，单位为ms。
OperationName	String	getXxx	埋点的接口名称
ServiceIp	String	172.20.XX.XX	应用所在的机器IP地址
ServiceName	String	arms-demo	应用名称
Timestamp	Long	1591255112346	13位时间戳
TraceID	String	swg-1231-xsdad-dddasds	调用链路ID

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/trace/Traces.json?RegionId=cn-hangzhou
&EndTime=342341121
&StartTime=234213211
&<公共请求参数>
```

### 正常返回示例

#### XML 格式

```
<RequestId>626037F5-FDEB-45B0-804C-B3C92797A64E</RequestId>
<TraceInfos>
  <ServiceIp>122.12.12.12</ServiceIp>
  <OperationName>getXxx</OperationName>
  <ServiceName>service-01</ServiceName>
  <TraceID>swg-1231-xsdad-dddasds</TraceID>
  <Duration>1200</Duration>
  <Timestamp>1590388651</Timestamp>
</TraceInfos>
```

#### JSON 格式

```
{
  "RequestId": "626037F5-FDEB-45B0-804C-B3C92797A64E",
  "TraceInfos": [
    {
      "ServiceIp": "122.12.12.12",
      "OperationName": "getXxx",
      "ServiceName": "service-01",
      "TraceID": "swg-1231-xsdad-dddasds",
      "Duration": "1200",
      "Timestamp": "1590388651"
    }
  ]
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 4.5 ListTraceApps

调用 ListTraceApps 列出指定地域下的全部应用监控任务。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

## 请求参数

名称	类型	是否必选	示例值	描述
RegionId	String	是	cn-hangzhou	地域 ID

## 返回数据

名称	类型	示例值	描述
RequestId	String	959C77E2-04E9-4FD4-848A-B8C1A48BEF51	请求 ID
Success	Boolean	true	请求是否成功
Code	Integer	200	返回码
Message	String	true	返回信息
TraceApps	Array		返回应用监控列表信息
AppId	Long	5918	应用 ID，数据库自增字段。
Pid	String	atc889zkcfd8deedfa9bfxxx	应用的 ID 标识串
AppName	String	arms-pop-hz	应用名称
Type	String	TRACE	监控类型。 <b>TRACE</b> 为应用监控， <b>RETCODE</b> 为前端监控。
UserId	String	12341234	用户 ID
CreateTime	Long	1529667762000	创建时间
UpdateTime	Long	1529667762000	更新时间
RegionId	String	cn-hangzhou	地域 ID

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/trace/ListTraceApps.json?Region=cn-hangzhou
```

### 正常返回示例

### XML 格式

```
<?xml version="1.0" encoding="UTF-8" ?>
<o>
  <requestId type="string">959C77E2-04E9-4FD4-848A-B8C1A48BEF51</requestId>
  <traceApps class="array">
    <e class="object">
      <appId type="number">5918</appId>
      <appName type="string">arms-pop-hz</appName>
      <createTime type="number">1529667762000</createTime>
      <pid type="string">xxxxxxxxxxxxxxxx</pid>
      <regionId type="string">cn-hangzhou</regionId>
      <type type="string">TRACE</type>
      <updateTime type="number">1529667762000</updateTime>
      <userId type="string">xxxxxxxxxxxxxxxx</userId>
    </e>
    <e class="object">
      <appId type="number">6549</appId>
      <appName type="string">docker-demo</appName>
      <createTime type="number">1531291867000</createTime>
      <pid type="string">xxxxxxxxxxxxxxxx</pid>
      <regionId type="string">cn-hangzhou</regionId>
      <type type="string">TRACE</type>
      <updateTime type="number">1531291867000</updateTime>
      <userId type="string">xxxxxxxxxxxxxxxx</userId>
    </e>
    <e class="object">
      <appId type="number">7451</appId>
      <appName type="string">arms-console-hz</appName>
      <createTime type="number">1533525518000</createTime>
      <pid type="string">xxxxxxxxxxxxxxxx</pid>
      <regionId type="string">cn-hangzhou</regionId>
      <type type="string">TRACE</type>
      <updateTime type="number">1533525518000</updateTime>
      <userId type="string">xxxxxxxxxxxxxxxx</userId>
    </e>
  </traceApps>
</o>
```

### JSON 格式

```
{
  "requestId": "959C77E2-04E9-4FD4-848A-B8C1A48BEF51",
  "traceApps": [
    {
      "appId": 5918,
      "pid": "xxxxxxxxxxxxxxxx",
      "appName": "arms-pop-hz",
      "type": "TRACE",
      "userId": "xxxxxxxxxxxxxxxx",
      "createTime": 1529667762000,
      "updateTime": 1529667762000,
    },
    {
      "appId": 6549,
      "pid": "xxxxxxxxxxxxxxxx",
      "appName": "docker-demo",
      "type": "TRACE",
      "userId": "xxxxxxxxxxxxxxxx",
      "createTime": 1531291867000,
      "updateTime": 1531291867000,
    },
    {
      "appId": 7451,
      "pid": "xxxxxxxxxxxxxxxx",
      "appName": "arms-console-hz",
      "type": "TRACE",
      "userId": "xxxxxxxxxxxxxxxx",
      "createTime": 1533525518000,
      "updateTime": 1533525518000,
    }
  ]
}
```

```
    "regionId": "cn-hangzhou"
  },
  {
    "appId": 6549,
    "pid": "xxxxxxxxxxxxxxxxxxxx",
    "appName": "docker-demo",
    "type": "TRACE",
    "userId": "xxxxxxxxxxxxxxxxxxxx",
    "createTime": 1531291867000,
    "updateTime": 1531291867000,
    "regionId": "cn-hangzhou"
  },
  {
    "appId": 7451,
    "pid": "xxxxxxxxxxxxxxxxxxxx",
    "appName": "arms-console-hz",
    "type": "TRACE",
    "userId": "xxxxxxxxxxxxxxxxxxxx",
    "createTime": 1533525518000,
    "updateTime": 1533525518000,
    "regionId": "cn-hangzhou"
  }
]
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 4.6 QueryMetric（应用监控）

调用QueryMetric接口查询应用监控的相关监控指标。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 描述

QueryMetric接口可用于查询应用监控和前端监控的相关监控指标。本文以查询应用监控的相关指标为例。

API名称	Request	Response
QueryMetric	QueryMetricRequest	QueryMetricResponse

### 请求参数

请求参数包含公共参数和业务参数。

#### 公共参数

公共请求参数参见[公共参数](#)。

## 业务参数

阿里云将用户的所有请求参数封装在一个Request中，返回一个Response。

字段名称	字段类型	设置方法	字段含义	是否必选	备注
StartTime	Long	setStartTime	查询数据的起始时间	是	取值为13位时间戳
EndTime	Long	setEndTime	查询数据的截止时间	是	取值为13位时间戳
Metric	String	setMetric	指标	是	需查询的指标不可自定义输入，请参见 <a href="#">可查询的应用监控指标</a> 。
Measures	List[String]	setMeasures	指标的测量数据	是	对应metric的测量数据，请参见 <a href="#">可查询的应用监控指标</a> 。
Dimensions	List[String]	setDimensions	维度	否	对应metric的维度，请参见 <a href="#">可查询的应用监控指标</a> 。
Filters	List[Filter]	setFilters	过滤条件	是	必选字段，必须添加 <b>pid</b> 和 <b>regionId</b> 条件， <b>pid</b> 获取方式请参见 <a href="#">如何获取应用pid</a> 。
IntervalInSec	Integer	setIntervalInSec	数据片的时间间隔（ms）	否	<ul style="list-style-type: none"> <li>单位：ms</li> <li>取值范围：≥ 60000</li> </ul>
OrderBy	String	setOrderBy	排序依据	否	无
Limit	Integer	setLimit	返回结果的个数限制	否	无

字段名称	字段类型	设置方法	字段含义	是否必选	备注
Order	String	setOrder	排序	否	取值范围： <ul style="list-style-type: none"> <li>ASC：升序</li> <li>DESC：降序</li> </ul>
SecurityToken	String	setSecurityToken	STS securityToken	否	采用RAM用户角色模式时需要设置该字段。详情参见 <a href="#">#unique_24</a> 。

### Filter复合字段说明

字段名称	字段类型	设置方法	字段含义	示例值
key	String	setKey	维度名称	pid
value	String	setValue	维度值	xxx@74xxx



#### 注意：

应用监控所有的Metric查询中必须在**filters**中添加**pid**条件和**regionId**条件。

### 如何获取应用pid

在**ARMS控制台**左侧导航栏中单击**应用监控 > 应用列表**，然后在**应用列表**页面单击目标应用名称，进入该应用的总览页面。

此时浏览器地址栏中的URL即包含应用的**pid**，格式为pid=xxx。由于浏览器进行了编码，除EDAS应用之外的其他应用需要对**pid**稍作修改。例如，如果URL中包含的pid为xxx%4074xxx，则需要将%40替换为@，即：xxx@74xxx。

### 返回参数

返回值为JSON串形式，可通过QueryMetricResponse.getdata()获取。

字段名称	字段含义	备注
data	数据点	无

### 可查询的应用监控指标

您可以使用**QueryMetric**接口查询应用监控的以下指标。



#### 说明：

已知具体的查询条件时，应将值传入 **filters** 参数中，用于限定查询结果的范围。如果不知道具体的查询条件，可以将下表中的维度传入 **dimensions** 参数，从而获得该维度所有可能值的列表。

指标集 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
appstat.vm	通用指标，对应应用详情下的JVM监控图表，包括GC指标、堆内存和非堆内存详情以及线程数。	<ul style="list-style-type: none"> <li>pid</li> <li>rootIp</li> </ul>	GC指标： <ul style="list-style-type: none"> <li>youngGcCount // JVM监控Young GC次数</li> <li>oldGcCount // JVM监控Full GC次数</li> <li>youngGcTime // JVM监控Young GC耗时</li> <li>oldGcTime // JVM监控Full GC耗时</li> <li>youngGcCountInstant //JVM监控Young GC次数瞬时值</li> <li>oldGcCountInstant //JVM监控Full GC次数瞬时值</li> <li>youngGcTimeInstant //JVM监控Young GC耗时瞬时值</li> <li>oldGcTimeInstant //JVM监控Full GC耗时瞬时值</li> </ul>

指标集 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
			<p>堆内存和非堆内存详情：</p> <ul style="list-style-type: none"> <li>• edenSpace //年轻代-eden区</li> <li>• oldGen //老年代</li> <li>• survivorSpace //年轻代-survivor区</li> <li>• metaSpace</li> <li>• nonHeapCommitted //非堆内存</li> <li>• nonHeapInit //非堆内存初始值</li> <li>• nonHeapMax //非堆内存最大值</li> <li>• nonHeapUsed //非堆内存使用量</li> <li>• directUsed //直接缓冲区</li> <li>• directCapacity //直接缓冲区</li> </ul> <p>线程数：</p> <ul style="list-style-type: none"> <li>• threadCount</li> <li>• threadNewCount</li> <li>• threadDeadlockCount</li> <li>• threadRunnableCount</li> <li>• threadTerminatedCount</li> <li>• threadTimedWaitCount</li> <li>• threadWaitCount</li> <li>• threadBlockedCount</li> </ul>

指标集 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
appstat.host	主机监控, 包括实例数、CPU、物理内存、磁盘、负载、网络流量 (Bytes) 和网络数据包数量。	<ul style="list-style-type: none"> <li>pid</li> <li>rootlp</li> </ul>	<p>实例数:</p> <ul style="list-style-type: none"> <li>instanceCount</li> </ul> <p>CPU:</p> <ul style="list-style-type: none"> <li>systemCpuIdle //页面未展示 - CPU 空等待</li> <li>systemCpuSystem</li> <li>systemCpuUser</li> <li>systemCpuIoWait</li> </ul> <p>物理内存:</p> <ul style="list-style-type: none"> <li>systemMemFree</li> <li>systemMemUsed</li> <li>systemMemTotal //页面未展示 - 系统内存总计</li> <li>systemMemBuffers</li> <li>systemMemCached //页面缓存</li> </ul> <p>磁盘:</p> <ul style="list-style-type: none"> <li>systemDiskFree</li> <li>systemDiskUsed</li> <li>systemDiskTotal //公共云控制台页面未展示</li> </ul> <p>负载:</p> <ul style="list-style-type: none"> <li>systemLoad</li> </ul>

指标集 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
			网络： <ul style="list-style-type: none"> <li>systemNetInPackets</li> <li>systemNetOutputPackets</li> <li>systemNetInErrs</li> <li>systemNetOutErrs</li> <li>systemNetInBytes</li> <li>systemNetOutputBytes</li> </ul>
appstat.database	数据库调用。	<ul style="list-style-type: none"> <li>pid</li> <li>rpcType //调用类型</li> <li>endpoint // 数据库地址：localhost:3306</li> <li>destId // 库名：arms</li> </ul>	<ul style="list-style-type: none"> <li>rt // 响应时间</li> <li>count // 请求数</li> <li>error // 错误数</li> </ul>
appstat.txn	接口调用。	<ul style="list-style-type: none"> <li>pid</li> <li>rpcType</li> <li>rpc// 接口：/demo/oracleTwo</li> </ul>	<ul style="list-style-type: none"> <li>rt</li> <li>count</li> <li>error</li> <li>errRate // 错误率</li> </ul>
appstat.incall	应用详情。	<ul style="list-style-type: none"> <li>pid</li> <li>rpcType</li> <li>rootIp</li> <li>rpc</li> <li>ppid</li> </ul>	<ul style="list-style-type: none"> <li>rt</li> <li>count</li> <li>error</li> </ul>
appstat.exception	异常。	<ul style="list-style-type: none"> <li>pid</li> <li>rpc</li> <li>endpoint</li> <li>exceptType</li> <li>exceptInfo</li> </ul>	<ul style="list-style-type: none"> <li>rt</li> <li>count</li> </ul>

指标集 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
appstat.sql	慢SQL。	<ul style="list-style-type: none"> <li>pid</li> <li>rpc</li> <li>endpoint</li> <li>sqlId</li> </ul>	<ul style="list-style-type: none"> <li>rt</li> <li>count</li> <li>error</li> <li>slow</li> </ul> <div style="border: 1px solid #ccc; background-color: #f0f0f0; padding: 5px; margin-top: 10px;">  <b>说明:</b> slow=true时, limit条件不生效。         </div>
appstat.mq.send	MQ发送。	无	<ul style="list-style-type: none"> <li>rt</li> <li>count</li> <li>error</li> </ul>
appstat.mq.receive	MQ接收。	无	<ul style="list-style-type: none"> <li>rt</li> <li>count</li> <li>error</li> </ul>

### 示例代码

使用**QueryMetric**接口查询应用监控JVM老年代与Eden区的示例代码如下:

## 5 前端监控

### 5.1 CreateRetcodeApp

调用 CreateRetcodeApp 创建前端监控任务。

#### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

#### 请求参数

名称	类型	是否必选	示例值	描述
RegionId	String	是	cn-hangzhou	地域 ID。
RetcodeApp Name	String	是	SdkTest	前端监控应用名称。
RetcodeApp Type	String	是	web	前端监控应用类型，包括 web、weex、mini_dd、mini_alipay、mini_wx 和 mini_common。

#### 返回数据

名称	类型	示例值	描述
RequestId	String	A5EC8221-08F2-4C95-9AF1-49FD998C647A	请求 ID。
RetcodeApp DataBean	Struct		返回前端监控创建信息。
AppId	Long	135143	应用 ID。
Pid	String	aokcdqn3ly@a195c6d6421****	PID。

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/retcode/CreateRetcodeApp.json?
RegionId=cn-hangzhou
&RetcodeAppName=SdkTest
&RetcodeAppType=web
```

### 正常返回示例

#### XML 格式

```
<CreateRetcodeApp>
  <requestId>A5EC8221-08F2-4C95-9AF1-49FD998C647A</requestId>
  <retcodeAppDataBean>
    <appId>135143</appId>
    <pid>aokcdqn3ly@a195c6d6421****</pid>
  </retcodeAppDataBean>
</CreateRetcodeApp>
```

#### JSON 格式

```
{
  "requestId": "A5EC8221-08F2-4C95-9AF1-49FD998C647A",
  "retcodeAppDataBean": {
    "appId": 135143,
    "pid": "aokcdqn3ly@a195c6d6421****"
  }
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 5.2 ListRetcodeApps

调用 ListRetcodeApps 列出指定地域下全部前端监控任务。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListRetcodeApps	系统规定参数，取值为 ListRetcodeApps。

名称	类型	是否必选	示例值	描述
RegionId	String	是	cn-hangzhou	地域 ID
AccessKeyId	String	否	****	忽略

### 返回数据

名称	类型	示例值	描述
RequestId	String	99A663CB-8D7B-4B0D-A006-03C8EE38E7BB	请求 ID
RetcodeApps			前端监控应用列表信息
AppId	Long	16064	应用 ID
AppName	String	a3	应用名称
Pid	String	xxxxxxxxxxxxxxxxxxxx	PID

### 示例

#### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/retcode/ListRetcodeApps.json?RegionId=cn-hangzhou
```

#### 正常返回示例

#### XML 格式

```
<0>
  <requestId type="string">99A663CB-8D7B-4B0D-A006-03C8EE38E7BB</requestId>
  <retcodeApps class="array">
    <e class="object">
      <appId type="number">16064</appId>
      <appName type="string">a3</appName>
      <pid type="string">xxxxxxxxxxxxxxxxxxxx</pid>
    </e>
    <e class="object">
      <appId type="number">38093</appId>
      <appName type="string">ARMS页面</appName>
      <pid type="string">xxxxxxxxxxxxxxxxxxxx</pid>
    </e>
    <e class="object">
      <appId type="number">77494</appId>
      <appName type="string">ARMS-Retcode页面</appName>
      <pid type="string">xxxxxxxxxxxxxxxxxxxx</pid>
    </e>
  </retcodeApps>
</0>
```

```
</e>
<e class="object">
  <appld type="number">77499</appld>
  <appName type="string">测试1</appName>
  <pid type="string">xxxxxxxxxxxxxxxxxxxx</pid>
</e>
<e class="object">
  <appld type="number">77501</appld>
  <appName type="string">前端测试</appName>
  <pid type="string">xxxxxxxxxxxxxxxxxxxx</pid>
</e>
</retcodeApps>
</o>
```

## JSON 格式

```
{
  "requestId": "99A663CB-8D7B-4B0D-A006-03C8EE38E7BB",
  "retcodeApps": [
    {
      "appName": "a3",
      "appld": 16064,
      "pid": "xxxxxxxxxxxxxxxxxxxx"
    },
    {
      "appName": "ARMS页面",
      "appld": 38093,
      "pid": "xxxxxxxxxxxxxxxxxxxx"
    },
    {
      "appName": "ARMS-Retcode页面",
      "appld": 77494,
      "pid": "xxxxxxxxxxxxxxxxxxxx"
    },
    {
      "appName": "测试1",
      "appld": 77499,
      "pid": "xxxxxxxxxxxxxxxxxxxx"
    },
    {
      "appName": "前端测试",
      "appld": 77501,
      "pid": "xxxxxxxxxxxxxxxxxxxx"
    }
  ]
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 5.3 SearchRetcodeAppByPage

调用 SearchRetcodeAppByPage 分页查询前端监控任务。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
PageNumber	Integer	是	1	当前查询页码
PageSize	Integer	是	5	每页数据行数
RegionId	String	是	cn-hangzhou	地域 ID
RetcodeAppName	String	是	App1	前端监控应用名称

### 返回数据

名称	类型	示例值	描述
RequestId	String	626037F5-FDEB-45B0-804C-B3C92797A64E	请求 ID
PageBean	Struct		每页返回信息
TotalCount	Integer	8	查询结果总数
PageNumber	Integer	1	当前查询页码
PageSize	Integer	2	每页数据行数
RetcodeApps	Array		每页返回前端监控任务信息
AppId	Long	16064	应用 ID，数据库自增字段。
Pid	String	aokcdqn3ly@741623b4e91****	应用的 ID 标识串

名称	类型	示例值	描述
AppName	String	a3	应用名称
Type	String	RETCODE	监控类型。 <b>TRACE</b> 为应用监控， <b>RETCODE</b> 为前端监控。
UserId	String	12341234	用户 ID
RegionId	String	cn-hangzhou	地域 ID
CreateTime	Long	1545363321000	创建时间
UpdateTime	Long	1545363321000	更新时间

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/retcode/SearchRetcodeAppByPage.json?
PageNumber=1
&PageSize=5
&RegionId=cn-hangzhou
&RetcodeAppName=App1
```

### 正常返回示例

#### XML 格式

```
<SearchRetcodeAppByPage>
  <requestId>626037F5-FDEB-45B0-804C-B3C92797A64E</requestId>
  <pageBean>
    <pageNumber>1</pageNumber>
    <pageSize>2</pageSize>
    <totalCount>8</totalCount>
    <retcodeApps>
      <retcodeApp>
        <appId>16064</appId>
        <appName>a3</appName>
        <createTime>1545363321000</createTime>
        <pid>xxxxxxxxxxxxxxxx</pid>
        <regionId>cn-hangzhou</regionId>
        <type>RETCODE</type>
        <updateTime>1545363321000</updateTime>
        <userId>xxxxxxxxxxxxxxxx</userId>
      </retcodeApp>
      <retcodeApp>
        <appId>38093</appId>
        <appName>TestApp</appName>
        <createTime>1553239261000</createTime>
        <pid>xxxxxxxxxxxxxxxx</pid>
        <regionId>cn-hangzhou</regionId>
        <type>RETCODE</type>
      </retcodeApp>
    </retcodeApps>
  </pageBean>
</SearchRetcodeAppByPage>
```

```
        <updateTime>1553239261000</updateTime>
        <userId>xxxxxxxxxxxx</userId>
    </retcodeApp>
</retcodeApps>
</pageBean>
</SearchRetcodeAppByPage>
```

### JSON 格式

```
{
  "requestId": "626037F5-FDEB-45B0-804C-B3C92797A64E",
  "pageBean": {
    "totalCount": 8,
    "pageNumber": 1,
    "pageSize": 2,
    "retcodeApps": [
      {
        "appId": 16064,
        "pid": "xxxxxxxxxxxx",
        "appName": "a3",
        "type": "RETCODE",
        "userId": "xxxxxxxxxxxx",
        "regionId": "cn-hangzhou",
        "createTime": 1545363321000,
        "updateTime": 1545363321000
      },
      {
        "appId": 38093,
        "pid": "xxxxxxxxxxxxxxxxxxxxxxxx",
        "appName": "TestApp",
        "type": "RETCODE",
        "userId": "xxxxxxxxxxxx",
        "regionId": "cn-hangzhou",
        "createTime": 1553239261000,
        "updateTime": 1553239261000
      }
    ]
  }
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 5.4 DeleteRetcodeApp

调用 DeleteRetcodeApp 删除前端监控任务。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

## 请求参数

名称	类型	是否必选	示例值	描述
<b>Action</b>	String	是	DeleteRetcodeApp	系统规定参数，取值为 DeleteRetcodeApp。
<b>AppId</b>	String	是	1231	应用 ID
<b>RegionId</b>	String	是	cn-hangzhou	地域 ID

## 返回数据

名称	类型	示例值	描述
Data	String	true	删除成功或失败
RequestId	String	01FF8DD9-A09C-47A1-895A-B6E321BE77B6	请求 ID

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/retcode/DeleteRetcodeApp.json?AppId=1231&RegionId=cn-hangzhou
```

### 正常返回示例

#### XML 格式

```
<0>
  <data type="string">true</data>
  <requestId type="string">01FF8DD9-A09C-47A1-895A-B6E321BE77B6</requestId>
</0>
```

#### JSON 格式

```
{
  "requestId": "01FF8DD9-A09C-47A1-895A-B6E321BE77B6",
  "data": "true"
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 5.5 SetRetcodeShareStatus

使用 SetRetcodeShareStatus 设置前端监控站点的分享开关。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	SetRetcodeShareStatus	要执行的操作，取值： <b>SetRetcodeShareStatus</b> 。
Pid	String	是	123123	应用 ID。
Status	Boolean	是	true	前端监控站点的分享开关的状态。 <ul style="list-style-type: none"><li>true：开启</li><li>false：关闭</li></ul>

### 返回数据

名称	类型	示例值	描述
IsSuccess	Boolean	true	设置成功或失败
RequestId	String	01FF8DD9-A09C-47A1-895A-B6E321BE77B6	请求 ID

### 示例

#### 请求示例

```
http(s)://arms.cn-hangzhou.aliyun-inc.com:8099/retcode/SetRetcodeShareStatus.json?Pid=123123&Status=true&<公共请求参数>
```

#### 正常返回示例

#### XML 格式

```
<SetRetcodeShareStatus>
  <IsSuccess>true</IsSuccess>
  <RequestId>01FF8DD9-A09C-47A1-895A-B6E321BE77B6</RequestId>
```

```
</SetRetcodeShareStatus>
```

### JSON 格式

```
{
  "IsSuccess": true,
  "RequestId": "01FF8DD9-A09C-47A1-895A-B6E321BE77B6"
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 5.6 GetRetcodeShareUrl

使用 GetRetcodeShareUrl 获取前端监控站点的分享地址。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetRetcodeShareUrl	要执行的操作，取值： <b>GetRetcodeShareUrl</b> 。
Pid	String	是	123123	应用 ID。

### 返回数据

名称	类型	示例值	描述
RequestId	String	01FF8DD9-A09C-47A1-895A-B6E321BE77B6	请求 ID

名称	类型	示例值	描述
Url	String	http://arms-daily.console.aliyun.com:8080/shareapi/retcode.json?login_arms_t3h_token=XXXxxx&action=RetcodeAction&eventSubmitDoGetData=1	前端监控站点的分享地址

## 示例

### 请求示例

```
http(s)://arms.cn-hangzhou.aliyun-inc.com:8099/retcode/GetRetcodeShareUrl.json?Pid=123123
&<公共请求参数>
```

### 正常返回示例

#### XML 格式

```
<GetRetcodeShareUrl>
  <requestId>01FF8DD9-A09C-47A1-895A-B6E321BE77B6</requestId>
  <url>http://arms-daily.console.aliyun.com:8080/shareapi/retcode.json?login_arms_t3h_token=XXXxxx&action=RetcodeAction&eventSubmitDoGetData=1</url>
</GetRetcodeShareUrl>
```

#### JSON 格式

```
{
  "requestId": "01FF8DD9-A09C-47A1-895A-B6E321BE77B6",
  "url": "http://arms-daily.console.aliyun.com:8080/shareapi/retcode.json?login_arms_t3h_token=XXXxxx&action=RetcodeAction&eventSubmitDoGetData=1"
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 5.7 QueryMetric（前端监控）

调用QueryMetric接口查询前端监控的相关监控指标。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 描述

QueryMetric接口可用于查询应用监控和前端监控的相关监控指标。本文以查询前端监控的相关指标为例。

API名称	Request	Response
QueryMetric	QueryMetricRequest	QueryMetricResponse

### 请求参数

请求参数包含公共参数和业务参数。

#### 公共参数

公共请求参数请参见[公共参数](#)。

#### 业务参数

阿里云将用户的所有请求参数封装在一个Request中，返回一个Response。

字段名称	字段类型	设置方法	字段含义	是否必选	备注
startTime	Long	setStartTime	查询数据的起始时间	是	取值为13位时间戳
endTime	Long	setEndTime	查询数据的截止时间	是	取值为13位时间戳
metric	String	setMetric	指标	是	需查询的指标不可自定义输入，请参见 <a href="#">可查询的应用监控指标</a> 。
measures	List[String]	setMeasures	指标的测量数据	是	对应metric的测量数据，请参见 <a href="#">可查询的应用监控指标</a> 。最多可添加5个。

字段名称	字段类型	设置方法	字段含义	是否必选	备注
dimensions	List[String]	setDimensions	维度	否	对应metric的维度，请参见 <a href="#">可查询的应用监控指标</a> 。最多可添加5个。
filters	List[Filter]	setFilters	过滤条件	是	必选字段，必须添加pid和regionId条件，pid获取方式请参见 <a href="#">如何获取应用pid</a> 。
intervalInSec	Integer	setIntervalInSec	数据片的时间间隔	否	<ul style="list-style-type: none"> <li>单位：ms</li> <li>取值范围：≥ 60000</li> </ul>
orderBy	String	setOrderBy	排序依据	否	N/A
limit	Integer	setLimit	返回结果的个数限制	否	N/A
order	String	setOrder	排序	否	取值范围： <ul style="list-style-type: none"> <li>ASC：升序</li> <li>DESC：降序</li> </ul>
securityToken	String	setSecurityToken	STS securityToken	否	采用RAM用户角色模式时需要设置该字段。详情参见 <a href="#">#unique_24</a> 。

### Filter复合字段说明

字段名称	字段类型	设置方法	字段含义	取值
key	String	setKey	维度名称	<ul style="list-style-type: none"> <li>pid</li> <li>regionId</li> <li>environment</li> </ul> 其中，pid和regionId为必填项，environment为可选项。

字段名称	字段类型	设置方法	字段含义	取值
value	String	setValue	维度值	<ul style="list-style-type: none"> <li>• pid的取值: xxx@74xxx</li> <li>• regionId的取值: cn-hangzhou</li> <li>• environment的取值:               <ul style="list-style-type: none"> <li>- prod: 表示线上环境</li> <li>- gray: 表示灰度环境</li> <li>- pre: 表示预发环境</li> <li>- daily: 表示日常环境</li> <li>- local: 表示本地环境</li> </ul> </li> </ul>

**注意:**

前端监控所有的Metric查询中必须在**filters**中添加**pid**和**regionId**条件。

**如何获取前端应用pid**

在控制台左侧导航栏中选择**前端监控**，然后在**前端监控**页面单击目标前端应用名称，进入该前端应用的总览页面。

此时浏览器地址栏中的URL即包含前端应用的**pid**，格式为pid=xxx。由于浏览器进行了编码，前端应用需要对**pid**稍作修改。例如，如果URL中包含的pid为xxx%4074xxx，则需要将%40替换为@，即：xxx@74xxx。

**返回参数**

返回值为JSON串形式，可通过QueryMetricResponse.getData()获取。

字段名称	字段含义	备注
data	数据点	无

**可查询的应用监控指标**

您可以使用**QueryMetric**接口查询应用监控的以下指标。

**说明:**

已知具体的查询条件时，应将值传入**filters**参数中，用于限定查询结果的范围。如果不知道具体的查询条件，可以将下表中的维度传入**dimensions**参数，从而获得该维度所有可能值的列表。

指标 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
webstat.index	总览 (访问量)	<ul style="list-style-type: none"> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr (屏幕分辨率)</li> <li>ip_isp (运营商)</li> <li>page</li> <li>ct (网络制式)</li> </ul>	<ul style="list-style-type: none"> <li>pv</li> <li>uv</li> </ul>
webstat.satisfy	满意度	<ul style="list-style-type: none"> <li>page</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> </ul>	<ul style="list-style-type: none"> <li>satisfy</li> <li>good (fpt &lt; 2000)</li> <li>bad (fpt &gt; 8000)</li> <li>neutral (fpt &gt; 2000 AND fpt &lt; 8000)</li> </ul>
webstat.url	访问的URL	<ul style="list-style-type: none"> <li>page</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>username</li> <li>uid</li> </ul>	<ul style="list-style-type: none"> <li>pv</li> <li>uv</li> </ul>

指标 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
webstat.session	慢加载追踪	N/A	<ul style="list-style-type: none"> <li>page</li> <li>sid</li> <li>browser</li> <li>load</li> <li>date</li> <li>browser_version</li> <li>dom</li> <li>ip_country</li> <li>ip_country_id</li> <li>ip_region</li> <li>ip_region_id</li> </ul>
webstat.speed	访问速度	<ul style="list-style-type: none"> <li>page</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>release</li> </ul>	<ul style="list-style-type: none"> <li>avg_fpt</li> <li>avg_ready</li> <li>avg_load</li> <li>avg_dns</li> <li>avg_tcp</li> <li>avg_ssl</li> <li>avg_ttfb</li> <li>avg_trans</li> <li>avg_dom</li> <li>avg_res</li> <li>avg_tti</li> <li>avg_fmp</li> <li>avg_cfpt</li> <li>avg_ctti</li> <li>avg_t1 ~ avg_t10</li> <li>count</li> </ul>

指标 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
webstat.stable	错误率排行	<ul style="list-style-type: none"> <li>page</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>release</li> <li>detector_app_version</li> </ul>	<ul style="list-style-type: none"> <li>rate</li> <li>count (样本量)</li> <li>error_pv</li> </ul>
webstat.errcate	错误聚类排行	<ul style="list-style-type: none"> <li>msg</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>page</li> <li>release</li> <li>detector_app_version</li> </ul>	<ul style="list-style-type: none"> <li>count (错误次数)</li> </ul>
webstat.msg.top	页面高频错误	<ul style="list-style-type: none"> <li>page</li> <li>msg</li> <li>file</li> <li>date</li> </ul>	<ul style="list-style-type: none"> <li>count (错误次数)</li> <li>error_uv (影响用户数)</li> </ul>
webstat.api	API成功率	<ul style="list-style-type: none"> <li>api</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>detector_app_version</li> <li>release</li> </ul>	<ul style="list-style-type: none"> <li>rate</li> <li>count (请求次数)</li> <li>avg_time (平均耗时)</li> </ul>

指标 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
webstat.apimsg	API Message聚类	<ul style="list-style-type: none"> <li>msg</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>code</li> <li>success</li> <li>release</li> <li>detector_a</li> <li>pp_version</li> </ul>	<ul style="list-style-type: none"> <li>count (请求次数)</li> </ul>
webstat.apicost	API成功耗时	<ul style="list-style-type: none"> <li>api</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>code</li> <li>success</li> <li>release</li> <li>detector_a</li> <li>pp_version</li> </ul>	<ul style="list-style-type: none"> <li>count (成功次数)</li> <li>avg_time (成功耗时均值)</li> </ul>
webstat.apifailtime	API失败耗时	<ul style="list-style-type: none"> <li>api</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> <li>code</li> <li>success</li> <li>release</li> <li>detector_a</li> <li>pp_version</li> </ul>	<ul style="list-style-type: none"> <li>count (错误次数)</li> <li>avg_time (失败耗时均值)</li> </ul>

指标 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
webstat.api.detail	API详情	<ul style="list-style-type: none"> <li>api</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>ct</li> <li>ip_isp</li> <li>sr</li> <li>page</li> <li>msg</li> <li>domain</li> </ul>	<ul style="list-style-type: none"> <li>success_rate</li> <li>success_count</li> <li>fail_count</li> <li>count (请求次数)</li> <li>success_time (成功耗时)</li> <li>fail_time (失败耗时)</li> <li>fail_uv (失败影响用户数)</li> </ul>
webstat.sum	自定义统计: 求和统计	<ul style="list-style-type: none"> <li>key</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> </ul>	<ul style="list-style-type: none"> <li>sum_val</li> <li>count (错误次数)</li> <li>uv</li> </ul>
webstat.avg	自定义统计: 均值统计	<ul style="list-style-type: none"> <li>key</li> <li>ip_country_id</li> <li>ip_region_id</li> <li>detector_browser</li> <li>detector_device</li> <li>detector_os</li> <li>sr</li> </ul>	<ul style="list-style-type: none"> <li>avg_val</li> <li>count (错误次数)</li> </ul>

指标 (Metric)	描述 (Description)	维度 (Dimensions)	测量数据 (Measures)
webstat.resource	资源错误排行	<ul style="list-style-type: none"> <li>• page</li> <li>• node_name</li> <li>• src</li> <li>• ip_country_id</li> <li>• ip_region_id</li> <li>• detector_browser</li> <li>• detector_device</li> <li>• detector_os</li> <li>• ct</li> <li>• ip_isp</li> <li>• sr</li> </ul>	count (资源错误数)

### 示例代码

使用**QueryMetric**接口查询前端监控总览页面的PV与UV的示例代码如下：

### 常见问题

- 调用时为什么会出现RAM的权限问题？  
这可能是由于该RAM角色没有权限，您可以为用户添加权限，详情请参见[#unique\\_24](#)。
- 怎么拉取列表数据（不考虑时间粒度）？  
将intervalInSec设置为2147483647。
- 为什么返回的数据值都为0？
  - 请检查时间间隔是否设置过小，intervalInSec 需要大于或等于6000。
  - 请检查regionId是否设置正确，这个regionId是根据日志接收的服务端划分的地域，而不是用户所在的地域，您可以根据项目的上报日志地址来区分regionId：
    - 杭州地域前端监控上报日志地址：<https://arms-retcode.aliyuncs.com/r.png?>
    - 新加坡地域前端监控上报日志地址：<https://arms-ap-southeast-1.console.aliyun.com/r.png?>
    - 美西地域前端监控上报日志地址：<http://arms-us-west-1.console.aliyun.com/r.png?>
- 在调用模拟器或代码接口时为什么会报错？
  - 请检查regionId是否已填写，filters中是否已添加pid。
  - 查看对应Metric的measures或dimensions是否正确，详情请参见[可查询的应用监控指标](#)。

- 报错信息以及对应解决方案

- **Metric查询错误，请联系管理员。**

- 请检查Metric字段是否正确。
- 请检查measures或dimensions是否正确。
- 请检查filters中是否已添加pid。

- **InvalidIntervalInSec**

请检查intervalInSec 是否超过最大值（2147483647）。

- **MissingMeasures**

请检查measures 是否已填写。

- **NonsequenceParameter.Dimensions**

- 请检查dimensions 是否已填写正确。
- 请检查模拟器dimensions 是否有多余的空格。

- **ServiceUnavailable**

请检查regionId 是否正确。

- **前端监控地域不合法**

请检查regionId是否设置正确，这个regionId是根据日志接收的服务端划分的地域，而不是用户所在的地域，您可以根据项目的上报日志地址来区分regionId：

- 杭州地域前端监控上报日志地址：<https://arms-retcode.aliyuncs.com/r.png?>
- 新加坡地域前端监控上报日志地址：<https://arms-ap-southeast-1.console.aliyun.com/r.png?>
- 美西地域前端监控上报日志地址：<http://arms-us-west-1.console.aliyun.com/r.png?>

- 为什么数据集中会缺失一些 **measures**或**dimensions**参数？

因为**measures**和**dimensions**最多设置5个，如果超过5个，将导致参数无法返回。

- 为什么一些指标的总数在聚合后的比未聚合时的明显变少？例如：**webstat.index**聚合后求和的pv总数比未聚合时得到的pv总数少。

因为POP网关的限制，超出10000条的数据将被去除，所以当聚合造成数据量超过限制时，返回的数据量比实际的量小，所以指标求和的总数会明显减少。您需要将每次请求后返回的数据量控制在10000条以内，以获得准确的数据。

## 6 Prometheus监控

### 6.1 AddGrafana

调用AddGrafana接口集成ARMS Prometheus监控的大盘。

#### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

#### 请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	AddGrafana	系统规定参数。取值：AddGrafana。
ClusterId	String	是	DFADGSDSF****	阿里云容器服务Kubernetes版的Kubernetes集群的ID。
Integration	String	是	asm	ARMS已支持的软件缩写。取值范围：ASM、IoT和Flink。
RegionId	String	是	cn-hangzhou	地域ID。

#### 返回数据

名称	类型	示例值	描述
Data	String	success	操作是否成功。
RequestId	String	959C77E2-04E9-4FD4-848A-B8C1A48BEF51	请求ID。

#### 示例

##### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/prometheus/AddGrafana.json?Region=cn-hangzhou&ClusterId=DFADGSDSF****
```

```
&Integration=asm
```

正常返回示例

XML 格式

```
<AddGrafana>
  <data>success</data>
  <requestId>959C77E2-04E9-4FD4-848A-B8C1A48BEF51</requestId>
</AddGrafana>
```

JSON 格式

```
{
  "data": "success",
  "requestId": "959C77E2-04E9-4FD4-848A-B8C1A48BEF51"
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 6.2 AddIntegration

调用AddIntegration接口集成ARMS Prometheus监控的大盘以及采集规则。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
<b>Action</b>	String	是	AddIntegration	系统规定参数。取值：AddIntegration。
<b>ClusterId</b>	String	是	ADFAGDSD****	阿里云容器服务Kubernetes版的Kubernetes集群的ID。
<b>Integration</b>	String	是	asm	ARMS已支持的软件缩写。取值范围：ASM、IoT和Flink。
<b>RegionId</b>	String	是	cn-hangzhou	地域ID。

## 返回数据

名称	类型	示例值	描述
Data	String	success	操作是否成功。
RequestId	String	959C77E2-04E9-4FD4-848A-B8C1A48BEF51	请求ID。

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/prometheus/AddIntegration.json?Region=cn-hangzhou&ClusterId=DFADGSDSF****&Integration=asm
```

### 正常返回示例

#### XML 格式

```
<AddIntegration>
  <data>success</data>
  <requestId>959C77E2-04E9-4FD4-848A-B8C1A48BEF51</requestId>
</AddIntegration>
```

#### JSON 格式

```
{
  "data": "success",
  "requestId": "959C77E2-04E9-4FD4-848A-B8C1A48BEF51"
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 6.3 GetPrometheusApiToken

调用GetPrometheusApiToken接口获取集成ARMS Prometheus监控所需的Token。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

## 请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetPrometheusApiToken	系统规定参数。取值：GetPrometheusApiToken。
RegionId	String	是	cn-hangzhou	地域ID。

## 返回数据

名称	类型	示例值	描述
RequestId	String	626037F5-FDEB-45B0-804C-B3C92797A64E	请求ID。
Token	String	20200521626037F5-FDEB-45B0-B3C92797****	集成ARMS Prometheus监控所需的Token。

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/prometheus/GetPrometheusApiToken.json?Region=cn-hangzhou
```

### 正常返回示例

#### XML 格式

```
<GetPrometheusApiToken>
  <requestId>01FF8DD9-A09C-47A1-895A-B6E321BE77B6</requestId>
  <token>20200521626037F5-FDEB-45B0-B3C92797****</token>
</GetPrometheusApiToken>
```

#### JSON 格式

```
{
  "requestId": "01FF8DD9-A09C-47A1-895A-B6E321BE77B6",
  "token": "20200521626037F5-FDEB-45B0-B3C92797****"
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 6.4 ListDashboards

调用ListDashboards接口列出指定地域下的某个集群的Grafana大盘。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
<b>Action</b>	String	是	ListDashboards	系统规定参数。取值：ListDashboards。
<b>ClusterId</b>	String	是	SDGDGSFSDG SDGF	阿里云容器服务Kubernetes版的Kubernetes集群的ID。
<b>RegionId</b>	String	是	cn-hangzhou	地域ID。

### 返回数据

名称	类型	示例值	描述
DashboardVos	Array		Grafana大盘信息
Exporter	String	Kubernetes	Exporter接入源的类型
Id	String	171	Grafana大盘ID，仅在安装Grafana大盘时是唯一的。
IsArmsExporter	Boolean	false	是否属于ARMS提供的Exporter
Tags	List	["arms-k8s", ccc8ce1fe0 c9543629e3 9ee657e34****"]	Grafana大盘标签
Time	String	1590136924	Grafana大盘创建时间
Title	String	ApiServer	Grafana大盘名称

名称	类型	示例值	描述
Type	String	dash-db	Grafana大盘类型
Uid	String	1411998742 223274-****-02	用于唯一标识安装多个Grafana大盘时的大盘ID，是展示在页面上的唯一业务ID。
Url	String	http://grafana .console. aliyun.com/d /1411998742 223274-****-02/ apiserver?orgId= 142&refresh=60s	Grafana大盘所在的完整URL地址
RequestId	String	959C77E2-04E9 -4FD4-848A- B8C1A48BEF51	请求ID

## 示例

### 请求示例

```
http://arms.cn-hangzhou.aliyun-inc.com:8099/prometheus/ListDashboards.json?
Region=cn-hangzhou
&ClusterId=DFADGSDSFASDF
```

### 正常返回示例

#### XML 格式

```
<DashboardVos>
  <Uid>1411998742223274-****-02</Uid>
  <Type>dash-db</Type>
  <Title>ApiServer</Title>
  <Time>1590136924</Time>
  <Id>171</Id>
  <Exporter>Kubernetes</Exporter>
  <Url>http://grafana.console.aliyun.com/d/1411998742223274-****-02/apiserver?
orgId=142&refresh=60s</Url>
  <IsArmsExporter>>false</IsArmsExporter>
</DashboardVos>
<DashboardVos>
  <Tags>["arms-k8s","ccc8ce1fe0c9543629e39ee657e34****"]</Tags>
</DashboardVos>
<RequestId>959C77E2-04E9-4FD4-848A-B8C1A48BEF51</RequestId>
```

#### JSON 格式

```
{
  "DashboardVos": [
    {
      "Uid": "1411998742223274-****-02",
```

```
"Type": "dash-db",
>Title": "ApiServer",
>Time": 1590136924,
>Id": 171,
>Exporter": "Kubernetes",
>Url": "http://grafana.console.aliyun.com/d/1411998742223274-****-02/apiserver
?orgId=142&refresh=60s",
>IsArmsExporter": false
},
{
  "Tags": ["arms-k8s","ccc8ce1fe0c9543629e39ee657e34****"]
}
],
"RequestId": "959C77E2-04E9-4FD4-848A-B8C1A48BEF51"
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 7 自定义监控

### 7.1 QueryDataset（下钻数据集）

调用QueryDataset查询下钻数据集中的数据。

#### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

#### 描述

QueryDataSet可用于查询ARMS下钻数据集中的数据。



#### 说明：

关于下钻数据集和通用数据集的区别，请参见[#unique\\_36](#)。

API名称	Request	Response
QueryDataSet	QueryDataSetRequest	QueryDataSetResponse

#### 请求参数

请求参数包含公共参数和业务参数。

#### 公共参数

公共请求参数请参见[公共参数](#)。

#### 业务参数

阿里云将用户所有的请求参数封装在一个Request中，返回一个Response。**QueryDataSetRequest**包含以下字段：

字段名称	字段类型	设置方法	字段含义	是否必选	备注
datasetId	Long	setDatasetId	数据集ID	是	请参见 <a href="#">如何获取datasetId</a> 。
minTime	Long	setMinTime	查询数据的起始时间	是	<ul style="list-style-type: none"><li>单位：ms</li><li>取值为13位时间戳</li></ul>

字段名称	字段类型	设置方法	字段含义	是否必选	备注
maxTime	Long	setMaxTime	查询数据的截止时间	是	<ul style="list-style-type: none"> <li>单位：ms</li> <li>取值为13位时间戳</li> </ul>
intervalInSec	Integer	setIntervalInSec	数据片的时间间隔	是	<ul style="list-style-type: none"> <li>单位：s</li> <li>取值范围：≥60</li> </ul>
measures	List[String]	setMeasures	查询指标列表	否	列表最长支持3个元素。如果为空，则返回所有指标数据。
dimensions	List[Dimension]	setDimensions	查询维度列表	否	下钻数据集字段，dimensions为复合参数，列表最长支持3个元素。Dimension的定义见下方表格。
orderByKey	String	setOrderByKey	orderBy指标	否	N/A
limit	Integer	setLimit	限制的返回个数	否	N/A
reduceTail	Boolean	setReduceTail	是否把limit之外的数据合并到一起	否	N/A
securityToken	String	setSecurityToken	STS securityToken	否	采用RAM用户角色模式时需要设置该字段。详情参见 <a href="#">#unique_24</a> 。

## Dimensions复合字段说明

字段名称	字段类型	设置方法	字段含义	备注
key	String	setKey	维度名称	如：区域
value	String	setValue	维度值	如：北京

字段名称	字段类型	设置方法	字段含义	备注
type	String	setType	取值方式	分别为：STATIC、All和DISABLED

- 当您想选择该维度下面的所有的维度值时，**type**设置为ALL，**value**为null。
- 当您想选择该维度下面的其中某个维度值时，**type**设置为STATIC（静态值），**value**为输入维度值。
- 当您不想选择该维度时候，可以直接忽略，或者将**type**设置为DISABLED。

### 如何获取datasetId

1. 在左侧导航栏选择**自定义监控 > 数据集管理**，在**实例列表**页面顶部选择目标地域。
2. 在**数据集管理**页签的数据集列表中，查看目标**数据集名称**对应的**数据集ID**，即可获取datasetId。

### 返回参数

返回值为JSON串形式返回，可通过ARMSQueryDataSetResponse.getdata()获取。

**ARMSQueryDataSetResponse**主要字段有：

字段名称	字段含义	备注
dimensions	时序数据的维度值	当该维度选为ALL时，会有多个。
measures	数据点中的指标	N/A
resultSize	返回的所有数据点个数	N/A
dimData	多条时序数据	N/A

### 完整使用示例

1. 在不选择维度情况下，查询所有数据的汇总信息。
2. 选择第一个维度，将维度值设为空字符串（""），类型为ALL（全部）。结果返回按第一维度分组的数据。
3. 选择第一个维度，将维度值设为固定值，如hangzhou，类型为STATIC（固定值）。结果返回按第一维度的hangzhou数据。
4. 选择要查询的指标列表，返回结果只返回您指定的指标。

### 相关文档

[公共参数](#)

## 7.2 QueryDataset（通用数据集）

调用QueryDataset查询通用数据集中的数据。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 描述

QueryDataSet可用于查询ARMS通用数据集中的数据。



#### 说明：

关于下钻数据集和通用数据集的区别，请参见[#unique\\_36](#)。

API名称	Request	Response
QueryDataSet	QueryDataSetRequest	QueryDataSetResponse

### 请求参数

请求参数包含公共参数和业务参数。

#### 公共参数

公共请求参数参见[公共参数](#)。

#### 业务参数

阿里云将用户所有的请求参数封装在一个Request中，返回一个Response。QueryDataSetRequest包含以下字段：

字段名称	字段类型	设置方法	字段含义	是否必选	备注
datasetId	Long	setDatasetId	数据集ID	是	请参见 <a href="#">如何获取datasetId</a> 。
minTime	Long	setMinTime	查询数据的起始时间	是	<ul style="list-style-type: none"> <li>单位：ms</li> <li>取值为13位时间戳</li> </ul>
maxTime	Long	setMaxTime	查询数据的截止时间	是	<ul style="list-style-type: none"> <li>单位：ms</li> <li>取值为13位时间戳</li> </ul>

字段名称	字段类型	设置方法	字段含义	是否必选	备注
intervalInSec	Integer	setIntervalInSec	数据片的时间间隔	是	<ul style="list-style-type: none"> <li>单位：s</li> <li>取值范围：≥60</li> </ul>
measures	List[String]	setMeasures	查询指标列表	否	列表最长支持3个元素。如果为空，则返回所有指标数据。
requiredDims	List[Dimension]	setRequiredDims	通用数据集的必选维度	否	如果您配置了必选维度，则此项为必填项。
optionalDims	List[Dimension]	setOptionalDims	通用数据集的可选维度	否	N/A
orderByKey	String	setOrderByKey	orderBy指标	否	N/A
limit	Integer	setLimit	限制的返回个数	否	N/A
reduceTail	Boolean	setReduceTail	是否把limit之外的数据合并到一起	否	N/A
securityToken	String	setSecurityToken	STS securityToken	否	采用RAM用户角色模式时需要设置该字段，请参见 <a href="#">#unique_24</a> 。

#### RequiredDims(OptionalDims)复合字段说明

字段名称	字段类型	设置方法	字段含义	备注
key	String	setKey	维度名称	如：区域
value	String	setValue	维度值	如：北京
type	String	setType	取值方式	分别为：STATIC、All和DISABLED

- 当您想选择该维度下面的所有的维度值时，**type**设置为ALL，**value**为null。

- 当您想选择该维度下面的其中某个维度值时，**type**设置为STATIC（静态值），**value**为输入维度值。
- 当您不想选择该维度时候，可以直接忽略，或者将**type**设置为DISABLED。

### 如何获取datasetId

1. 登录 [ARMS 控制台](#)。
2. 在左侧导航栏选择**自定义监控 > 数据集管理**，在**实例列表**页面顶部选择目标地域。
3. 在**数据集管理**页签的数据集列表中，查看目标**数据集名称**对应的**数据集ID**，即可获取datasetId。

### 返回参数

返回值为JSON串形式返回，可通过QueryDataSetResponse.getdata()获

取。**QueryDataSetResponse**主要字段有：

字段名称	字段含义
measures	数据点中的指标
resultSize	返回的所有数据点个数
dimData	多条时序数据

### 完整使用示例

1. 在不选择维度情况下，查询所有数据的汇总信息。
2. 请检查您是否配置了必选维度。如果配置了，那么在查询时一定要设置必选维度相关配置，否则无法进行查询。
3. 选择要查询的指标列表，返回结果只返回指定的指标。

### 相关文档

[公共参数](#)

## 8 报警

### 8.1 CreateAlertContact

调用 CreateAlertContact 创建报警联系人。

#### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

#### 请求参数

名称	类型	是否必选	示例值	描述
<b>Action</b>	String	是	CreateAlertContact	系统规定参数。取值：CreateAlertContact。
<b>ContactName</b>	String	是	JohnDoe	报警联系人名称
<b>DingRobotWebhookUrl</b>	String	是	https://oapi.dingtalk.com/robot/send?access_token=91f2f6****	钉钉机器人 Webhook URL。获取方式参见 <a href="https://dingdoc.dingtalk.com/doc#/serverapi2/qf2nxq">https://dingdoc.dingtalk.com/doc#/serverapi2/qf2nxq</a> 。注意：钉钉机器人安全设置中的自定义关键词请填写“报警”。
<b>Email</b>	String	是	someone@example.com	联系人邮箱地址
<b>PhoneNum</b>	String	是	1381111****	联系人手机号码
<b>RegionId</b>	String	是	cn-hangzhou	地域 ID。默认请填写 <b>cn-hangzhou</b> 。
<b>SystemNoc</b>	Boolean	是	true	是否接收系统通知
<b>ProxyUserId</b>	String	否	12341234	内部参数

邮箱地址、手机号码和钉钉机器人 Webhook URL 这三种联系方式中必须至少填写一种。

## 返回数据

名称	类型	示例值	描述
RequestId	String	C21AB7CF-B7AF-410F-BD61-82D1567F65F8	请求 ID
ContactId	String	6258	报警联系人 ID

## 示例

### 请求示例

```
http(s)://[Endpoint]/?ContactName=JohnDoe
&DingRobotWebhookUrl=https://oapi.dingtalk.com/robot/send?access_token=xxxxxxx
&Email=johndoe@example.com
&PhoneNum=1381111****
&RegionId=cn-hangzhou
&SystemNoc=true
&<公共请求参数>
```

### 正常返回示例

#### XML 格式

```
<CreateAlertContact>
  <RequestId>C21AB7CF-B7AF-410F-BD61-82D1567F65F8</RequestId>
  <ContactId>6258</ContactId>
</CreateAlertContact>
```

#### JSON 格式

```
{
  "requestId": "C21AB7CF-B7AF-410F-BD61-82D1567F65F8",
  "contactId": "6258"
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 8.2 CreateAlertContactGroup

调用 CreateAlertContactGroup 创建报警联系人分组。

## 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

## 请求参数

名称	类型	是否必选	示例值	描述
<b>ContactGroupName</b>	String	是	TestGroup	报警联系人分组名称
<b>ContactIds</b>	String	是	123 234 245	报警联系人分组内的联系人 ID。多个联系人 ID 以空格分隔。
<b>RegionId</b>	String	是	cn-hangzhou	地域 ID。默认请填写 <b>cn-hangzhou</b> 。
<b>ProxyUserId</b>	String	否	12341234	内部参数

## 返回数据

名称	类型	示例值	描述
RequestId	String	E9354A61-9535-4CB3-8E1B-C12E177D2450	请求 ID
ContactGroupId	String	4466	报警联系人分组 ID

## 示例

## 请求示例

```
http(s)://[Endpoint]/?ContactGroupName=TestGroup
&ContactIds=123 234 245
&RegionId=cn-hangzhou
&<公共请求参数>
```

## 正常返回示例

## XML 格式

```
<CreateAlertContactGroup>
  <RequestId>E9354A61-9535-4CB3-8E1B-C12E177D2450</RequestId>
  <ContactGroupId>4466</ContactGroupId>
</CreateAlertContactGroup>
```

## JSON 格式

```
{
  "requestId": "E9354A61-9535-4CB3-8E1B-C12E177D2450",
  "contactGroupId": "4466"
}
```

}

## 错误码

访问[错误中心](#)查看更多错误码。

## 8.3 SearchAlertContactGroup

调用 SearchAlertContactGroup 查询报警联系人分组。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	SearchAlertContactGroup	系统规定参数。取值：SearchAlertContactGroup。
ContactGroupName	String	是	TestGroup	报警联系人分组名称
RegionId	String	是	cn-hangzhou	地域 ID。默认请填写 <b>cn-hangzhou</b> 。
ProxyUserId	String	否	12341234	内部参数
ContactName	String	否	John Doe	报警联系人名称
ContactId	Long	否	123	报警联系人 ID

### 返回数据

名称	类型	示例值	描述
RequestId	String	620BC657-3D59-4DC9-A5CE-D09CE3834C1D	请求 ID
ContactGroups	Array		报警联系人分组信息
ContactGroupId	Long	746	报警联系人分组 ID

名称	类型	示例值	描述
ContactGroupName	String	TestGroup	报警联系人分组名称
UserId	String	12341234	用户 ID
CreateTime	Long	1529668855000	创建时间
UpdateTime	Long	1529668855000	更新时间

## 示例

### 请求示例

```
http(s)://[Endpoint]/?Action=SearchAlertContactGroup
&ContactGroupName=TestGroup
&RegionId=cn-hangzhou
&<公共请求参数>
```

### 正常返回示例

#### XML 格式

```
<SearchAlertContactGroup>
  <requestId>620BC657-3D59-4DC9-A5CE-D09CE3834C1D</requestId>
  <contactGroups>
    <contactGroup>
      <contactGroupId>746</contactGroupId>
      <contactGroupName>TestGroup1</contactGroupName>
      <createTime>1529668855000</createTime>
      <updateTime>1529668855000</updateTime>
      <userId>xxxxxxxxxxx</userId>
    </contactGroup>
    <contactGroup>
      <contactGroupId>3849</contactGroupId>
      <contactGroupName>TestGroup2</contactGroupName>
      <createTime>1561428915000</createTime>
      <updateTime>1561428915000</updateTime>
      <userId>xxxxxxxxxxx</userId>
    </contactGroup>
  </contactGroups>
</SearchAlertContactGroup>
```

#### JSON 格式

```
{
  "requestId": "620BC657-3D59-4DC9-A5CE-D09CE3834C1D",
  "contactGroups": [
    {
      "contactGroupId": 746,
      "contactGroupName": "TestGroup1",
      "userId": "xxxxxxxxxxx",
      "createTime": 1529668855000,
      "updateTime": 1529668855000
    }
  ]
}
```

```

    },
    {
      "contactGroupId": 3849,
      "contactGroupName": "TestGroup2",
      "userId": "xxxxxxxxxxx",
      "createTime": 1561428915000,
      "updateTime": 1561428915000
    }
  ]
}

```

## 错误码

访问[错误中心](#)查看更多错误码。

## 8.4 ImportAppAlertRules

调用 ImportAppAlertRules 创建应用报警规则。



### 说明：

此报警规则导入接口仅适合于应用监控和前端监控报警规则，包括默认生成的应用监控、前端监控报警规则。不适用于自定义监控报警规则、Prometheus 监控报警规则、默认紧急报警规则等。

## 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

## 请求参数

名称	类型	是否必选	示例值	描述
<b>ContactGroupIds</b>	String	是	[123, 234]	报警联系人分组 ID，格式为 JSONArrayStr。
<b>Pids</b>	String	是	["atc889zkcfd8deedfa9bfxxxx", "acd129bfcfd5daebfa6cdxxxx"]	常见报警规则关联的 ARMS 应用 ID（即 PID），格式为 JSONArrayListStr。
<b>RegionId</b>	String	是	cn-hangzhou	应用报警规则关联的应用所属地域的 ID。

名称	类型	是否必选	示例值	描述
<b>TemplateAlertId</b>	String	是	324324234	报警模板 ID。TemplateAlertId 和 TemplateAlertConfig 必须至少填写一个。如果两个参数都填写，则 TemplateAlertId 优先。
<b>IsAutoStart</b>	Boolean	否	true	生成报警规则后是否自动启动报警规则。 <b>true</b> 表示生成报警规则后启动报警规则， <b>false</b> 表示生成报警规则后不启动报警规则，默认为 <b>false</b> 。

名称	类型	是否必选	示例值	描述
TemplageAlertConfig	String	否	<pre>[{"contactGroups": "258", "alertType": 5, "alarmContext": { "subTitle": "", "content": "报警名称: \$报警名称\n筛选条件: \$筛选\n报警时间: \$报警时间\n报警内容: \$报警内容\n注意: 该报警未收到恢复邮件之前, 正在持续报警中, 24小时后会再次提醒您! ", "alertLevel": "WARN", "metricParam": { "appId": "7579", "pid": "arms-console:arms-consolehost", "type": "TXN", "dimensions": { "type": "DISABLED", "key": "rpc" } }, "alertWay": ["DING_ROBOT"], "alertRule": { "rules": { "measure": "appstat.txn.rt", "alias": "调用响应时间_ms", "aggregates": "AVG", "nValue": 1, "value": 1, "operator": "CURRENT_GT_E" } }, "operator": "" }, {"title": "这是个导入配置报警模板", "config": {"continuous"</pre>	<p>ARMS 报警规则的配置 JSON 串。TemplateAlertId 和 TemplageAlertConfig 必须至少填写一个。如果两个参数都填写, 则 TemplateAlertId 优先。关于此字段的详细说明参见<a href="#">关于参数 TemplageAlertConfig 的补充说明</a>。</p>
68				文档版本: 20200630

名称	类型	是否必选	示例值	描述
ProxyUserId	String	否	12341234	内部参数

## 关于参数 TemplageAlertConfig 的补充说明

### 如何在 ARMS 控制台生成报警模板 JSON 串

首先，在 ARMS 控制台的**报警管理 > 报警策略管理**页面，创建一个应用监控报警作为模板。然后在**报警规则**页签上选择此报警模板，并单击左下角的**批量导出报警**，即可生成报警模板 JSON 串。

### JSON 串示例

```
[
  {
    "contactGroupIds": "381", // 报警通知发送的联系人分组 ID，以“,”分隔。报警模板中忽略，导入时会映射对应值。
    "alertType": 5, // 目前支持4为前端监控报警，5为应用监控报警。
    "alarmContext": { // 报警通知内容模板。
      "subTitle": "",
      "content": "报警名称:$报警名称\n筛选条件: $筛选\n报警时间: $报警时间\n报警内容: $报警内容\n注意: 该报警未收到恢复邮件之前，正在持续报警中，24小时后会再次提醒您！"
    },
    "alertLevel": "WARN", // 报警等级: FATAL、ERROR、WARN。
    "metricParam": {
      "appId": "70901", // 报警关联的应用 appId。模板报警配置中可随意填写，导入时会映射对应值。
      "pid": "atc889zkcf@d8deedfa9bf1520", // 报警关联的应用 pid。模板报警配置中可随意填写，导入时会映射对应值。
      "type": "TXN", // 报警指标类型，详见下一节。
      "dimensions": [ // 报警关联的维度，详见下一节。
        {
          "type": "STATIC",
          "value": "\/hello_test_api_address\/test1",
          "key": "rpc"
        }
      ]
    },
    "alertWay": [ // 报警发送通知方式: SMS (开启短信)、MAIL (开启邮件通知)、DING_ROBOT (开启钉钉机器人通知)。
      "SMS",
      "MAIL",
      "DING_ROBOT"
    ],
    "alertRule": {
      "rules": [ // 报警判断规则列表
        {
          "measure": "appstat.txn.rt", // 报警规则请求指标，详见下一节。
          "alias": "入口调用响应时间_ms", // 报警规则请求指标展示字段，报警模板无需填此字段。
          "aggregates": "AVG", // 数据请求后聚合算子。AVG为取平均，SUM为取和，MIN为取最小值，MAX为取最大值。
          "nValue": 1, // 报警规则每分钟轮询请求几分钟的数据。
          "value": 1, // 报警规则判断阈值。
          "operator": "CURRENT_GTE" // CURRENT_GTE: 大于等于; CURRENT_LTE: 小于等于; PREVIOUS_UP: 环比上升; PREVIOUS_DOWN: 环比下降; HOH_UP: 与上小时同
```

```

比上升; HOH_DOWN: 与上小时同比下降; DOD_UP: 与昨日同比上升 %; DOD_DOWN: 与昨日同比下降 %。
    }
  ],
  "operator": "&" // 表示多个判断条件的组合方式。& 为与逻辑, | 为或逻辑。
},
"title": "报警模板报警名", // 模板报警名称。导入后, 会生成 "{title}-对应用名" 作为新报警名称。
"config": "{\"continuous\":false,\"dataRevision\":2,\"ownerId\": \" 123412341234\"}",
// continuous 为 true 表示连续报警, continuous 为 false 表示以 24 小时为静默期的静默报警策略。dataRevision 字段为数据修订策略, 0 为补零、1 为补1、2 为空值不处理。
"notice": { // unix timestamp ms 时间戳,
noticeStartTime 与 noticeEndTime 时间戳代表通知时间范围, startTime 与 endTime 代表报警生效时间范围, timestamp 转换为时间戳代表当天时间点, 如 1565964097071 对应 2019-08-16 22:01:37, 代表 22:01:37 时间点。
"noticeStartTime": 1480521600000,
"startTime": 1480521600000,
"endTime": 1480607940000,
"noticeEndTime": 1480607940000
},
"status": "NON" // 报警当前启动装填。报警模板中忽略此字段, 导入时会映射对应值。
}
]

```

### 报警模板中 Measure 与 Dimension、metricParam.type 说明

每个报警属于一个类型, 由 metricParam.type 字段控制。每类报警都能设置一种维度 (Dimension) 作为筛选条件。每类报警都可以配置多个报警规则 (alertRule), 每个 alertRule 可以配置多个属于此 type 类型的数据请求指标进行计算。

Dimensions.type 有三种选项:

- **STATIC**: 固定匹配此维度值需要填 dimensions.value。
- **ALL**: 遍历所有维度值, 如此接口所有接口名的指标依次判断, 哪个接口触发阈值引起报警, 会在报警内容中体现接口名, 此时不需要填 dimensions.value。
- **DISABLE**: 聚合所有维度值为一个值 (求和), 此时不需要填 dimensions.value。

### 报警指标取值枚举

报警类型 metricParam.type	此类报警维度 dimensions.key	此类报警数据请求指标 alertRule.rules.measure
TXN (应用监控入口调用量)	rpc (接口名称)	appstat.txn.rt (入口调用响应时间_ms)、appstat.txn.count (入口调用次数)、appstat.txn.errcount (入口调用错误次数)

报警类型 <code>metricParam.type</code>	此类报警维度 <code>dimensions.key</code>	此类报警数据请求指标 <code>alertRule.rules.measure</code>
TXN_TYPE（应用监控调用类型统计）	rpcType（接口调用类型）（如 HTTP、DUBBO）	appstat.inbound.rt（应用提供服务调用响应时间_ms）、appstat.inbound.count（应用提供服务调用次数）、appstat.inbound.error（应用提供服务调用错误数）、appstat.outbound.rt（应用依赖服务调用响应时间_ms）、appstat.outbound.count（应用依赖服务调用次数）、appstat.outbound.error（应用依赖服务调用错误数）
DB（数据库指标）	endpoint（数据库名称）	appstat.database.rt（数据库调用响应时间_ms）、appstat.database.count（数据库调用次数）、appstat.database.errcount（数据库调用错误次数）

报警类型 metricParam.type	此类报警维度 dimensions.key	此类报警数据请求指标 alertRule.rules.measure
JVM (JVM 监控)	rootIp (机器 IP)	<p>appstat.jvm.heap_used (JVM 堆内总内存量_byte)、appstat.jvm.GcPsScavengeCount (JVM 垃圾回收次数)、appstat.jvm.GcPsMarkSweepCount (JVM 标记清除次数)、appstat.jvm.GcG1OldGenCount (JVM_Old 区 G1GC 次数)、appstat.jvm.GcG1YoungGenCount (JVM_Young 区 G1GC 次数)、appstat.jvm.gc.YoungGcCountInstant (JVM_YoungGC 次数)、appstat.jvm.gc.OldGcCountInstant (JVM_FullGC 次数)、appstat.jvm.gc.YoungGcTimeInstant (JVM_YoungGC 耗时_ms)、appstat.jvm.gc.OldGcTimeInstant (JVM_FullGC 耗时_ms)、appstat.jvm.ThreadCount (JVM_线程总数)、appstat.jvm.non_heap_used (JVM 非堆总使用内存量_byte)、appstat.jvm.non_heap_max (JVM 非堆内存最大值_byte)、appstat.jvm.non_heap_init (JVM 非堆内存初始值_byte)、appstat.jvm.non_heap_committed (JVM 非堆内存提交值_byte)</p>

报警类型 metricParam.type	此类报警维度 dimensions.key	此类报警数据请求指标 alertRule.rules.measure
HOST (主机监控)	rootIp (机器 IP)	appstat.jvm.SystemCpuUser (节点机用户使用 CPU_百分比)、appstat.jvm.SystemMemFree (节点机空闲内存_byte)、appstat.jvm.SystemDiskFree (节点机空闲磁盘_byte)、appstat.jvm.SystemNetInErrs (节点机接收错误报文数)、appstat.jvm.SystemNetOutErrs (节点机发送错误报文数)、appstat.jvm.SystemLoad (节点机系统负载)
EXCEPTION (异常接口调用)	rpc (接口名称)	appstat.exception.rt (应用异常接口调用响应时间_ms)、appstat.exception.count (应用异常接口调用次数)

### 返回数据

名称	类型	示例值	描述
RequestId	String	A5EC8221-08F2-4C95-9AF1-49FD998Cxxxx	请求 ID
Data	String	399363	返回值

### 示例

#### 请求示例

```
http(s)://[Endpoint]/?ContactGroupIds=1234
&Pids=["xxx@xxxxxx", "xxx@xxxxxx2"]
&RegionId=cn-hangzhou
&TemplateAlertId=324324234
```

&<公共请求参数>

正常返回示例

XML 格式

```
<Data>[1217430]</Data>
<RequestId>9CF78EE0-90E2-4BC3-AA95-227378F913AC</RequestId>
```

JSON 格式

```
{
  "Data": "[1217430]",
  "RequestId": "9CF78EE0-90E2-4BC3-AA95-227378F913AC"
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 8.5 SearchAlertContact

调用 SearchAlertContact 查询报警联系人。

### 调试

您可以在 OpenAPI Explorer 中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer 可以自动生成 SDK 代码示例。

### 请求参数

名称	类型	是否必选	示例值	描述
<b>Action</b>	String	是	SearchAlertContact	系统规定参数。取值：SearchAlertContact。
<b>ContactName</b>	String	是	John Doe	报警联系人名称。非必填。
<b>CurrentPage</b>	String	是	1	查询分页的当前页码。非必填，默认为 1。
<b>Email</b>	String	是	someone@example.com	联系人邮箱。非必填。
<b>PageSize</b>	String	是	20	查询分页的每页项数。非必填，默认为 10。

名称	类型	是否必选	示例值	描述
Phone	String	是	1381111*****	报警联系人电话号码。非必填。
RegionId	String	是	cn-hangzhou	地域 ID。默认请填写 <b>cn-hangzhou</b> 。
ProxyUserId	String	否	12341234	内部参数

### 返回数据

名称	类型	示例值	描述
RequestId	String	7FE5346F-F26E-47B2-812A-1BA0CDB021B9	请求 ID
PageBean	Struct		返回结构体
TotalCount	Integer	23	总结果项数
PageNumber	Integer	1	返回结果页码
PageSize	Integer	10	返回结果每页项数
Contacts	Array		联系人对象列表
ContactId	Long	123	联系人 ID
ContactName	String	John Doe	联系人名称
Phone	String	1381111*****	联系人电话号码
Email	String	someone@example.com	联系人邮箱地址
UserId	String	123412341234	用户 ID
DingRobot	String	https://oapi.dingtalk.com/robot/send?access_token=91f2f6****	钉钉机器人 Webhook URL
CreateTime	Long	1579508519683	创建时间

名称	类型	示例值	描述
UpdateTime	Long	1579508519683	更新时间
SystemNoc	Boolean	false	是否接收系统通知

## 示例

### 请求示例

```
http(s)://[Endpoint]/?Action=SearchAlertContact
&ContactName=Darth Vader
&CurrentPage=1
&Email=12341234@gmail.com
&PageSize=20
&Phone=15966666666
&RegionId=cn-hangzhou
&<公共请求参数>
```

### 正常返回示例

### XML 格式

```
<RequestId>7FE5346F-F26E-47B2-812A-1BA0CDB021B9</RequestId>
<PageBean>
  <PageNumber>1</PageNumber>
  <TotalCount>2</TotalCount>
  <PageSize>10</PageSize>
  <Contacts>
    <Phone>123412341234</Phone>
    <Email>yll_2@qq.com</Email>
    <ContactName>测试_1</ContactName>
    <SystemNoc>>false</SystemNoc>
    <CreateTime>1489566438000</CreateTime>
    <UpdateTime>1489566438000</UpdateTime>
    <DingRobot/>
    <ContactId>313</ContactId>
    <UserId>123412341234</UserId>
  </Contacts>
  <Contacts>
    <Phone>152342234342</Phone>
    <Email>darth_vader@alibaba-inc.com</Email>
    <ContactName>测试_2</ContactName>
    <SystemNoc>>false</SystemNoc>
    <CreateTime>1515859703000</CreateTime>
    <UpdateTime>1572602781000</UpdateTime>
    <DingRobot/>
    <ContactId>1023</ContactId>
    <UserId>123412341234</UserId>
  </Contacts>
</PageBean>
```

### JSON 格式

```
{
  "RequestId": "7FE5346F-F26E-47B2-812A-1BA0CDB021B9",
  "PageBean": {
```

```
"pageNumber": 1,
"totalCount": 2,
"pageSize": 10,
"contacts": [
  {
    "phone": "123412341234",
    "email": "yll_2@qq.com",
    "contactName": "测试_1",
    "systemNoc": false,
    "createTime": 1489566438000,
    "updateTime": 1489566438000,
    "dingRobot": "",
    "contactId": 313,
    "userId": "123412341234"
  },
  {
    "phone": "152342234342",
    "email": "darth_vader@alibaba-inc.com",
    "contactName": "测试_2",
    "systemNoc": false,
    "createTime": 1515859703000,
    "updateTime": 1572602781000,
    "dingRobot": "",
    "contactId": 1023,
    "userId": "123412341234"
  }
]
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 9 附录

### 9.1 如何通过内网调用 API

本文通过配置云解析 PrivateZone，为无公网访问能力的专有网络 VPC 类型的 ECS 实例提供了通过阿里云内网调用 API 的方案。

#### 背景信息

由于 ARMS 提供的接入地址（Endpoint）为公网服务地址，当您的 ECS 实例没有分配公网带宽或者不存在公网 IP 地址时，则无法使用阿里云 CLI 或者 SDK 等工具发起 API 请求。此时，您可以为 ECS 实例所在地域下的专有网络 VPC 关联云解析 PrivateZone，即可完成在阿里云内网调用 API 的目的。

#### 使用说明

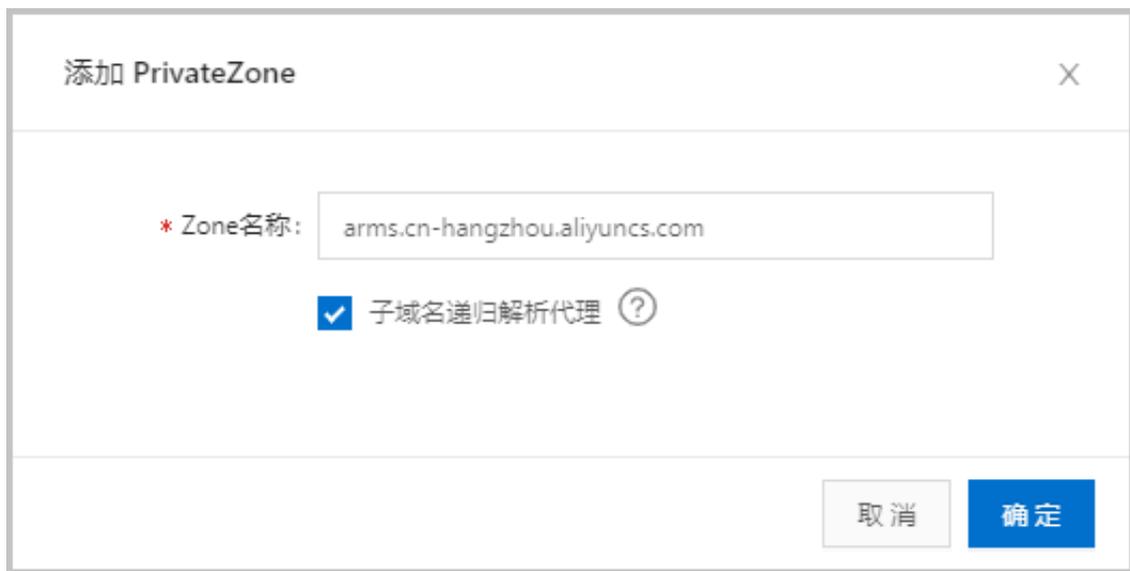
- 此方案仅适用于专有网络 VPC 类型的 ECS 实例所在的地域，不支持跨地域配置云解析 PrivateZone。
- 建议您使用已部署了阿里云 CLI 或者 SDK 的自定义镜像创建 ECS 实例，避免实例在无公网访问的条件下无法加载相关依赖。
- 目前，支持云解析 PrivateZone 的 ARMS 接入地址（Endpoint）如下表所示，请确保您使用的 Endpoint 在列举范围内。

阿里云地域	地域 ID	CNAME 记录值	接入地址（Endpoint）
华东 1（杭州）	cn-hangzhou	popunify-vpc.cn-hangzhou.aliyuncs.com	arms.cn-hangzhou.aliyuncs.com
华东 2（上海）	cn-shanghai	popunify-vpc.cn-shanghai.aliyuncs.com	arms.cn-shanghai.aliyuncs.com
华北 1（青岛）	cn-qingdao	popunify-vpc.cn-qingdao.aliyuncs.com	arms.cn-qingdao.aliyuncs.com
华北 2（北京）	cn-beijing	popunify-vpc.cn-beijing.aliyuncs.com	arms.cn-beijing.aliyuncs.com
华南 1（深圳）	cn-shenzhen	popunify-vpc.cn-shenzhen.aliyuncs.com	arms.cn-shenzhen.aliyuncs.com
华北 3（张家口）	cn-zhangjiakou	popunify-vpc.cn-zhangjiakou.aliyuncs.com	arms.cn-zhangjiakou.aliyuncs.com

阿里云地域	地域 ID	CNAME 记录值	接入地址 (Endpoint)
中国 (香港)	cn-hongkong	popunify-vpc.cn-hongkong.aliyuncs.com	arms.cn-hongkong.aliyuncs.com
新加坡	ap-southeast-1	popunify-vpc.ap-southeast-1.aliyuncs.com	arms.ap-southeast-1.aliyuncs.com

### 操作步骤

1. 登录[云解析 DNS 控制台](#)。
2. 在左侧导航栏中，单击 **PrivateZone** 后，在 **PrivateZone** 页面单击**添加 Zone**。
3. 在**添加 PrivateZone** 对话框中，完成以下设置后，单击**确定**。
  - **Zone 名称**：设置一个已支持云解析 PrivateZone 的 ARMS 接入地址，例如 arms.cn-hangzhou.aliyuncs.com。
  - **子域名递归解析代理**：选择该功能后，在 VPC 内查询 Zone 命名空间内未配置的子域名时，PrivateZone 会代理公网递归解析，并将递归解析结果做为 DNS 查询响应，返回 VPC。



4. 在 Zone 列表找到已创建的 PrivateZone，单击**操作列**的**解析设置**。
5. 在**解析设置**页面的**解析设置**页签，单击**添加记录**。

6. 在**添加记录**对话框中，完成以下设置后，单击**确定**。

- **记录类型**：选择 **CNAME**。
- **主机记录**：填写为 @，可以解析 @.exmaple.com 域名。
- **记录值**：设置为对应地域下的 CNAME 记录值，详情请参见[使用说明](#)。
- **TTL 值**：递归 DNS 缓存域名解析结果的时间，本示例选择了 **1 分钟**。

添加记录

记录类型: CNAME 将域名指向另外一个域名

主机记录: @ .arms.cn-hangzhou.aliyuncs.com

\* 记录值: popunify-vpc.cn-hangzhou.aliyuncs.com

TTL值: 1 分钟

取消 确定

7. 返回 **PrivateZone** 页面，找到已创建的 PrivateZone，单击**操作**列的**关联VPC**。

- 在**关联 VPC** 对话框中，选择与 PrivateZone 相同的地域，并勾选 ECS 实例所在的专有网络 VPC（可多选），单击**确定**。



## 执行结果

为专有网络 VPC 关联了云解析 PrivateZone 后，您可以通过远程连接登录实例，在 ECS 实例内部测试是否能访问对应地域的接入地址，详情请参见[#unique\\_40](#)。

以下以 `arms.cn-hangzhou.aliyuncs.com` 为例，使用 **ping** 功能测试数据包收发状况。

```
ping arms.cn-hangzhou.aliyuncs.com
```

当出现类似以下显示结果时，表示可以访问对应地域的接入地址。

```
[root@arms-dev-kevin ~]# ping arms.cn-hangzhou.aliyuncs.com
PING popunify-vpc.cn-hangzhou.aliyuncs.com.gds.alibabadns.com (100.103.7.26) 56(
84) bytes of data.
64 bytes from 100.103.7.26: icmp_seq=1 ttl=102 time=1.11 ms
64 bytes from 100.103.7.26: icmp_seq=2 ttl=102 time=1.13 ms
64 bytes from 100.103.7.26: icmp_seq=3 ttl=102 time=1.14 ms
64 bytes from 100.103.7.26: icmp_seq=4 ttl=102 time=1.13 ms
64 bytes from 100.103.7.26: icmp_seq=5 ttl=102 time=1.13 ms
64 bytes from 100.103.7.26: icmp_seq=6 ttl=102 time=1.13 ms
^C
--- popunify-vpc.cn-hangzhou.aliyuncs.com.gds.alibabadns.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5859ms
rtt min/avg/max/mdev = 1.114/1.132/1.149/0.044 ms
[root@arms-dev-kevin ~]#
```