

阿里云 阿里云SDK开发指南

Python SDK

文档版本：20181109

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。
斜体	表示参数、变量。	<code>bae log list --instanceid Instance_ID</code>
<code>[]</code> 或者 <code>[a b]</code>	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
<code>{}</code> 或者 <code>{a b}</code>	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 使用手册.....	3
2.1 使用Python SDK.....	3
2.2 Endpoint管理.....	5
2.3 使用CommonRequest进行调用.....	6
2.4 配置STS Token.....	8
2.5 配置RamRole实现ECS实例的无AK访问.....	9
2.6 获取服务地址.....	10
2.7 设置HTTPS请求.....	11
2.8 错误处理.....	12
2.9 部分云产品的SDK使用说明.....	12
3 常见错误.....	13

1 快速开始

欢迎使用阿里云开发者工具套件（SDK）。阿里云Python 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Python 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[提交反馈](#)。

环境准备

- 使用阿里云Python 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 Python SDK

完成以下操作安装Python SDK：

1. 安装SDK核心库。

- 如果您使用Python 2.x，执行以下命令，安装阿里云SDK核心库：

```
pip install aliyun-python-sdk-core
```

- 如果您使用Python 3.x，执行以下命令，安装阿里云SDK核心库：

```
pip install aliyun-python-sdk-core-v3
```

2. 安装云产品的SDK。

以下是安装云服务器ECS的SDK示例：

```
pip install aliyun-python-sdk-ecs
```

使用Python SDK

以下这个代码示例展示了调用阿里云Python SDK的3个主要步骤：

1. 创建Client实例。在创建Client实例时，您需要获取Region ID、AccessKey ID和AccessKey Secret。
2. 创建API请求并设置参数。

3. 发起请求并处理应答或异常。

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.acs_exception.exceptions import ClientException
from aliyunsdkcore.acs_exception.exceptions import ServerException
from aliyunsdkecs.request.v20140526 import DescribeInstancesRequest
from aliyunsdkecs.request.v20140526 import StopInstanceRequest

# 创建AcsClient实例
client = AcsClient(
    "<your-access-key-id>",
    "<your-access-key-secret>",
    "<your-region-id>"
);
# 创建request，并设置参数
request = DescribeInstancesRequest.DescribeInstancesRequest()
request.set_PageSize(10)
# 发起API请求并显示返回值
response = client.do_action_with_exception(request)
print response
```

2 使用手册

2.1 使用Python SDK

本文档介绍如何安装及调用阿里云Python SDK。

安装

阿里云Python SDK支持 Python 2.6.x, 2.7.x 和 3.x及以上环境，并提供pip和GitHub两种安装方式。

- 使用pip安装(推荐)

执行以下命令，通过pip安装SDK。

```
pip install aliyun-python-sdk-core # 安装阿里云SDK核心库
pip install aliyun-python-sdk-ecs # 安装管理ECS的库
```



说明：

如果您使用的是 python3.x，请将 `pip install aliyun-python-sdk-core` 修改为 `pip install aliyun-python-sdk-core-v3`。

- 下载GitHub源码

无执行以下命令，通过GitHub安装Python SDK。

```
git clone https://github.com/aliyun/aliyun-openapi-python-sdk.git
# 安装阿里云 SDK 核心库
cd aliyun-python-sdk-core
python setup.py install
# 安装阿里云 ECS SDK
cd aliyun-python-sdk-ecs
python setup.py install
```

设置身份验证凭据

当使用阿里云SDK访问阿里云服务时，您需要提供阿里云账号进行身份验证。

目前，Python SDK支持以下几种身份验证方式：

验证方式	说明
AccessKey	使用AccessKey ID和AccessKey Secret访问
StsToken	使用STS Token访问
RamRoleArn	使用RAM子账号的AssumeRole方式访问

验证方式	说明
EcsRamRole	在ECS实例上通过EcsRamRole实现免密验证
RsaKeyPair	使用RSA公私钥方式（仅日本站支持）

本文以AccessKey为例说明如何设置身份凭证。为了保证您的账号安全，建议您使用RAM账号来访问阿里云服务。阿里云账号的AccessKey对拥有的资源有完全的权限。RAM账号由阿里云账号授权创建，仅有对特定资源限定的操作权限。参考[创建AccessKey](#)创建RAM账号的AccessKey。

使用AccessKey作为访问凭据，需要在初始化Client时设置。



注意：

确保包含AccessKey的代码不会泄漏（例如提交到外部公开的GitHub项目），否则将会危害您的阿里云账号的信息安全。

```
client = AcsClient(
    "<access-key-id>",
    "<access-key-secret>",
    "<region-id>")
```

发起调用

本文以ECS为例，介绍如何使用阿里云Python SDK发起请求。

1. 导入相关产品的SDK。

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.acs_exception.exceptions import ClientException
from aliyunsdkcore.acs_exception.exceptions import ServerException
from aliyunsdkecs.request.v20140526 import DescribeInstancesRequest
from aliyunsdkecs.request.v20140526 import StopInstanceRequest
```

2. 新建一个AcsClient。

```
client = AcsClient(
    "<your-access-key-id>",
    "<your-access-key-secret>",
    "<your-region-id>"
);
```

3. 创建Request对象。

```
request = DescribeInstancesRequest.DescribeInstancesRequest()
request.set_PageSize(10)
```

4. 发起调用并处理返回。

```
try:
```

```
response = client.do_action_with_exception(request)
print response
except ServerException as e:
    print e
except ClientException as e:
    print e
```

2.2 Endpoint管理

Endpoint是阿里云服务的API服务端地址。针对不同的地域，单个服务可能有不同的Endpoint。例如，云服务器（ECS）在华东1（杭州）地域的Endpoint是 `ecs-cn-hangzhou.aliyuncs.com`，而在日本（东京）地域的Endpoint是 `ecs.ap-northeast-1.aliyuncs.com`。阿里云SDK内置了Endpoint寻址模块，当您调用SDK对一个服务发起请求时，SDK会自动根据您在创建SDK Client时指定的地域ID（Region ID）和产品ID来找到Endpoint。

异常处理

如果遇到了SDK.InvalidRegionID 或者SDK.EndpointResolvingError的错误，建议您参考以下信息解决。

错误码	错误消息	解决方法
SDK.InvalidRegionID	can not find endpoint to access	当前SDK版本过低。请将SDK核心库aliyun-python-sdk-core升级到2.9.0版本或更高版本。
SDK.EndpointResolvingError	No such region <region-id>. Please check your region ID.	请检查您输入的地域ID是否正确。 可以通过DescribeRegionsAPI查找地域ID。
SDK.EndpointResolvingError	No endpoint for product <product-id>.	解决方法： - 当前SDK版本过低。请将SDK核心库aliyun-python-sdk-core升级到2.9.0版本或更高版本，并将使用的产品SDK如aliyun-java-sdk-ecs升级到最新版本。 - 直接设置Endpoint来发送请求，详情参见直接设置Endpoint。
SDK.EndpointResolvingError	No endpoint in the region <region-id> for product <product-id>.	在指定地域下找不到该产品的Endpoint，解决方法如下： 可能该产品在指定地域尚未开放服务，请根据提示信息更换地域ID。

错误码	错误消息	解决方法
		- 将aliyun-python-sdk-core升级到最新版本，以便应用最新的内置Endpoint解析配置。阿里云在服务地址发生新增、变更时会发布新的SDK Core版本。 - 直接设置Endpoint来发送请求，详情参见直接设置Endpoint。

直接设置Endpoint

参考以下代码直接为服务请求设置Endpoint。参考各云产品的API文档，查看可用的Endpoint。



注意：

请根据实际情况替换代码示例中的Endpoint (ecs-cn-hangzhou.aliyuncs.com)。

```
request = DescribeInstancesRequest()  
# 对这个请求直接设置 Endpoint  
request.set_endpoint("ecs-cn-hangzhou.aliyuncs.com")  
# 使用这个 Endpoint发起请求  
response = self.client.do_action_with_exception(request)
```

2.3 使用CommonRequest进行调用

当您要调用的某个产品的API没有提供SDK时，可以采用泛用型的API调用方式 (CommonRequest)。使用CommonRequest调用方式可实现任意Open API接口的调用。

CommonRequest调用特点

CommonRequest的特点如下：

1. 轻量：只需Core包即可发起调用，无需下载安装各产品的SDK。
2. 简便：无需更新SDK即可调用最新发布的API。
3. 快速迭代。

使用CommonRequest

阿里云产品的API有RPC和RESTful两种风格，不同风格的API的CommonRequest的调用方法也不同。

通常API参数中包含Action参数的是RPC风格，包含PathPattern参数的是RESTful风格。一般情况下，每个产品内，所有API的调用风格是统一的。每个API仅支持特定的一种风格调用，传入错误的标识，可能会调用到其他API，或收到ApiNotFound的错误信息。

发起一次CommonRequest请求，您需要获取以下几个参数的值。您可以在[文档中心](#)各产品的API文档中获取以下参数的值。此外，部分产品也可以通过[OpenAPI Explorer](#)来获取API的参数信息。

- 域名(domain)：该产品的服务地址。
- API版本(version)：该API的版本号，格式为YYYY-MM-DD。

您可以在各产品的API文档的公共参数部分获取API版本。

- 接口信息：要调用的接口名称。

— 当调用的API为RPC风格时（大部分阿里云产品API为RPC风格）如ECS和RDS，需要获取**Action**参数，使用`request.ApiName = "<Action>"`的方式来指定API名称。

例如RunInstances接口，在发起CommonRequest请求时，需要使用`request.ApiName = "RunInstances"`来指定API名称。

— 当调用的API为RESTful风格时如容器服务，需要获取**PathPattern**参数，使用`request.PathPattern = "<PathPattern>"`的方式来指定RESTful路径。

例如容器服务的查看所有集群实例的API的**PathPattern**为/clusters，在发起CommonRequest请求时，要使用`request.PathPattern = "/clusters"`指定RESTful路径。

示例：调用RPC风格的API

以下代码展示了如何使用CommonRequest的方式调用ECS的DescribeInstanceStatus接口：

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.request import CommonRequest
client = AcsClient('{your_access_key_id}', '{your_access_key_secret}',
                  '{your_region_id}')
request = CommonRequest()
request.set_domain('ecs.aliyuncs.com')
request.set_version('2014-05-26')
request.set_action_name('DescribeInstanceStatus')
# or:
# request = CommonRequest(domain='ecs.aliyuncs.com', version='2014-05-26',
#                           action_name='DescribeInstanceStatus')
request.add_query_param('PageNumber', '1')
request.add_query_param('PageSize', '30')
response = client.do_action_with_exception(request)
```

示例：调用RESTful风格的API

以下代码展示了如何使用CommonRequest的方式调用容器服务的查看所有集群实例接口：

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.request import CommonRequest
client = AcsClient('{your_access_key_id}', '{your_access_key_secret}',
                  '{your_region_id}')
```

```
request = CommonRequest()
request.set_domain('cs.aliyuncs.com')
request.set_version('2015-12-15')
request.set_uri_pattern('/clusters')
# or:
# request = CommonRequest(domain='ecs.aliyuncs.com', version='2014-05-26', uri_pattern='/clusters')
response = client.do_action_with_exception(request)
```

2.4 配置STS Token

直接使用阿里云账号的主账号的AccessKey ID和AccessKey Secret进行应用开发会有一定的安全风险，为了提升安全性，除了通过RAM角色控制权限范围外，您还可以使用为RAM角色签发的STS Token来访问阿里云服务。

STS Token介绍

阿里云STS (Security Token Service) 是为阿里云账号（或RAM用户）提供短期访问权限管理的云服务。通过STS，您可以为联盟用户（您的本地账号系统所管理的用户）颁发一个自定义时效和访问权限的访问凭证。联盟用户可以使用STS短期访问凭证直接调用阿里云服务API，或登录阿里云管理控制台操作被授权访问的资源。

使用STS Token调用SDK有以下优势：

- 减少了主账号AccessKey ID和AccessKey Secret泄露的风险，特别是移动设备等场景。
- 能使用灵活的权限控制，STS Token有一定的时间限制，并且根据RAM角色的灵活设置对ECS、SLB等资源的精细授权。



说明：

在使用STS Token前，确保该产品支持STS Token验证。

设置STS Token

您可以通过以下两种方式设置STS Token：

- 方式一：直接使用STS Token

直接使用STS Token时，您需要自行维护STS Token的周期性更新。

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.auth.credentials import StsTokenCredential
sts_token_credential = StsTokenCredential('sts_access_key_id', '
sts_access_key_secret', 'sts_session_token')
```

```
acs_client = AcsClient(region_id='cn-hangzhou', credential=
sts_token_credential)
```

其中：

- **region-id**是您正在使用的地域ID。您可以通过调用**DescribeRegions**接口查看地域ID。
- **sts-access-key-id**、**sts-access-key-secret**和**sts-session-token**是调用**AssumeRole**接口返回的授权信息。
- 方式二：使用SDK自动管理STS Token

您可以通过指定RAM的角色信息，让SDK帮您自动申请并维护STS Token。

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.auth.credentials import RamRoleArnCredential

ram_role_arn_credential = RamRoleArnCredential('access_key_id', '
access_key_secret', 'role_arn', 'role_session_name')
acs_client = AcsClient(region_id='cn-hangzhou', credential=
ram_role_arn_credential)
```

其中：

- **role-arn**是角色全局资源描述符，您可以通过访问[RAM控制台](#)，单击角色名，进入详情页后查询角色名对应的role-arn。
- **role-session-name**是临时角色名称，您可以通过调用**AssumeRole**接口来获取一个扮演该角色的临时身份，创建成功后，便可以使用创建时的**RoleSessionName**参数作为本方式的**role-session-name**参数。

2.5 配置RamRole实现ECS实例的无AK访问

为了提高应用部署的安全性的同时提升便利性，阿里云SDK支持通过[实例元数据](#)服务来获取ECS RAM角色的授权信息来访问阿里云资源和服务。使用这种方式，您部署在ECS上的应用程序，无需在SDK上配置授权信息即可访问阿里云API（即不需要配置AccessKey）。通过这种方式授权的SDK，可以拥有这个ECS RAM角色的权限。



说明：

确保ECS实例已经配置了RAM角色。

示例

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.auth.credentials import EcsRamRoleCredential

ecs_ram_role_credential = EcsRamRoleCredential("role_name")
```

```
acs_client = AcsClient(region_id='region-id', credential=ecs_ram_role_credential)
```

其中：

- **role-name**是与ECS实例关联的RAM角色名称。
- **region-id**是使用的地域的ID，你可以通过调用**DescribeRegions**接口查看地域ID。



说明：

示例中的**region-id**是目标服务(RAM角色有访问权限)的API所在地域，不一定等于这个ECS实例的地域。

2.6 获取服务地址

使用阿里云SDK调用云产品API时，需要获取每个云产品API的服务地址（Endpoint）。

配置方法

您只需在初始化 **AcsClient**实例时指定一个地域 ID即可，系统会根据指定的地域ID和产品名称，返回对应的服务地址。



说明：

不允许随意改变初始化**AcsClient**实例的参数顺序，必须先输入**AccessKey ID**，然后是**Accesskey Secret**，最后是**Region ID**。

```
client = AcsClient(  
    "<your-access-key-id>",  
    "<your-access-key-secret>",  
    "<your-region-id>"    # 注意参数顺序，Region ID 是第三个参数  
);
```

故障排查

在使用阿里云Python SDK时，如果出现**SDK.InvalidRegionId** 或 **Can not find endpoint to access**的错误，可能有如下原因：

- 原因 1：该产品未在指定区域开通服务，或该产品在指定区域开通了服务，但未注册Endpoint。
解决方案：提交工单来注册 Endpoint。
- 原因 2：参数填写错误。

解决方案：检查各个参数的值是否正确，并且保证初始化**AcsClient**实例时参数顺序是正确的。

2.7 设置HTTPS请求

阿里云Python SDK支持使用HTTP和HTTPS协议发起 API 请求。大部分产品使用HTTP协议，但访问控制(RAM)，安全令牌 (STS) 和密钥管理 (KMS) 等产品默认使用HTTPS协议发起API请求。

使用Python SDK时，您可以为某个请求指定使用HTTP或HTTPS协议，您也可以设置全局默认协议。



说明：

产品的默认协议 (HTTP/HTTPS) 优先于设置的全局默认协议。此外，访问控制 (RAM)，安全令牌 (STS) 和密钥管理 (KMS) 的默认协议为 HTTPS，不能使用 HTTP 协议。

添加OpenSSL依赖

阿里云Python SDK的HTTPS协议依赖Python的OpenSSL支持。要使用阿里云 SDK通过HTTPS协议发送请求，您需要在 Python 中添加 OpenSSL 支持。

运行`python -c "import ssl"`查看Python环境是否支持OpenSSL。运行后，如果没有出现 `ImportError: No module named ssl`的错误信息，说明已经支持OpenSSL。

若没有OpenSSL支持，运行以下命令安装：

```
pip install pyopenssl
```

设置单个请求的HTTP/HTTPS协议

参考以下代码示例为一个接口设置HTTPS调用：

```
request = CreateInstanceRequest.CreateInstanceRequest()  
request.set_protocol_type("https")  
# 取值："https" 或 "http"
```

设置全局默认协议

参考以下代码示例设置全局默认协议：

```
import aliyunsdkcore.request  
aliyunsdkcore.request.set_default_protocol_type("https")
```

```
# 创建请求并调用 client.do_action_with_exception() 来发送请求
```

2.8 错误处理

在使用阿里云Python SDK时，当服务端或者SDK端出错时，SDK会返回相应的异常信息。这些异常信息中会包含具体的错误信息包括错误码 (Error Code) 和错误消息 (Error Message) 。

一般情况下，您不需要对阿里云Python SDK返回的错误进行处理。您只需要根据服务端的错误提示处理服务端错误即可。

- `ServerException`是服务端错误信息。
- `ClientException`是SDK端错误信息。

例如，当出现如下错误时，您可以根据具体的错误信息，修改AccessKey的ID即可。

```
aliyun sdkcore.acs\_exception.exceptions.ServerException: HTTP Status  
: 404 Error:InvalidAccessKeyId.NotFound Specified access key is not  
found.
```

如果您需要对SDK的错误进行处理，可以参考如下示例编写代码：

```
try:  
    response = client.do_action_with_exception(request)  
except ServerException as e:  
    # 这里可以添加您自己的错误处理逻辑  
    # 例如，打印具体的错误信息  
    print e.get_http_status()  
    print e.get_error_code()  
    print e.get_error_msg()
```

2.9 部分云产品的SDK使用说明

目前部分阿里云产品的Python SDK有完全不同的依赖、使用步骤和功能，关于这些云产品的Python SDK使用方法，请参考具体的产品文档。

产品	使用说明
对象存储 (OSS)	使用说明 使用说明
表格存储 (Table Store)	使用说明 使用说明
日志服务 (Log Service)	使用说明 使用说明
大数据计算 (MaxCompute)	使用说明 使用说明
消息服务 (MNS)	使用说明 使用说明

3 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.InvalidRegionId	Can not find endpoint to access.	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>from aliyunsdkcore.profile import region_provider region_provider.modify_point('Ecs', 'cn-zhangjiakou', 'ecs.aliyuncs.com')</pre>
SDK.TimeoutError	—	请求超时,并且重试全部失败。	- 如果存在跨region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。 - 如果频繁出现该问题,且排除低质量网络场景的情况下,建议您提交工单进行处理。
SDK.ServerError: InvalidProtocol.NeedSsl	Your request is denied as lack of ssl protect .Recommend :https://error-center.aliyun.com/status/search?Keyword=InvalidProtocol.NeedSsl&source=PopGw	API只接受HTTPS请求,不接受HTTP请求。	在发送request前,添加以下代码: <pre>request.set_protocol_type('https')</pre>