**AWS综合实验S3触发Lambda**

**实验目的**

学会使用S3的事件触发Lambda函数,本次实验将以S3上传文件触发Lambda对对象进行备份

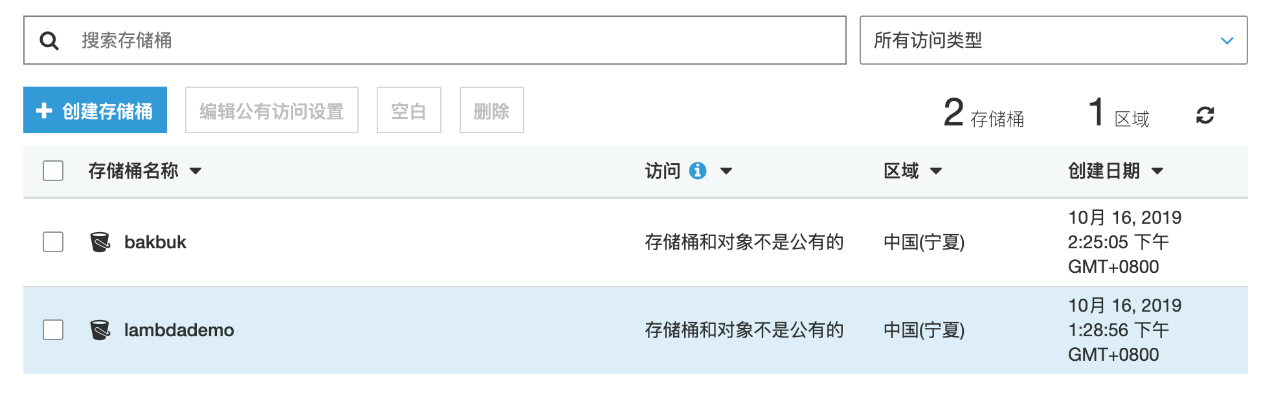
**实验要求**

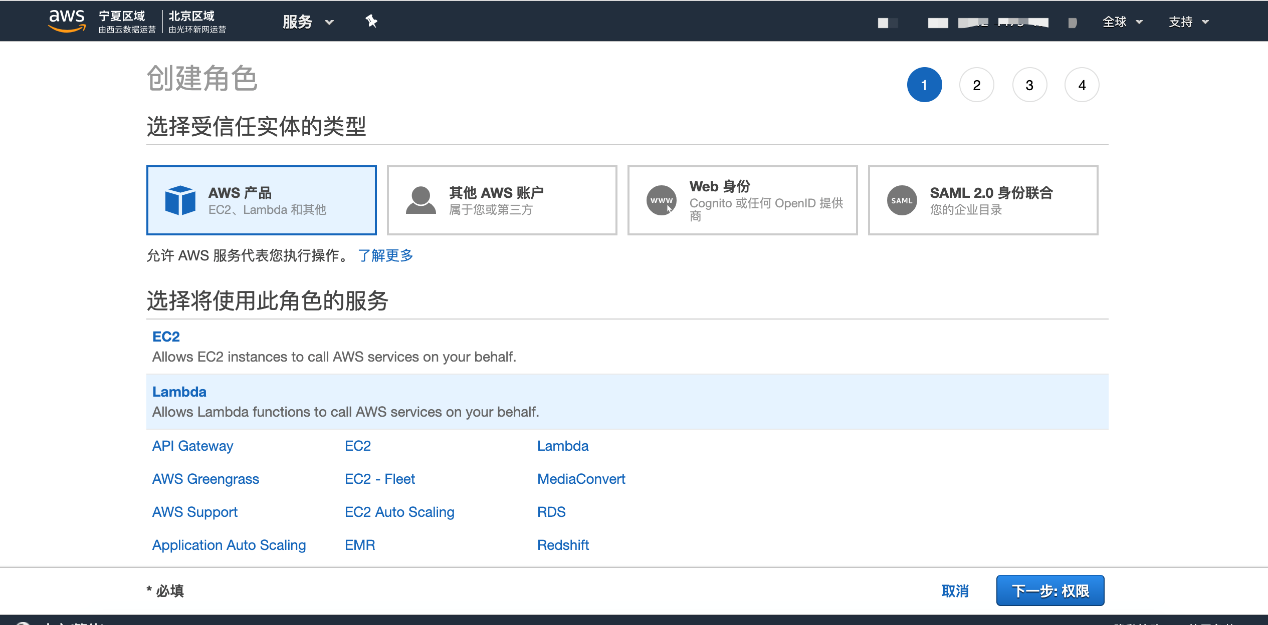
了解Amazon Web Services  
了解S3  
了解Lambda  
了解IAM  
了解至少一种编程语言

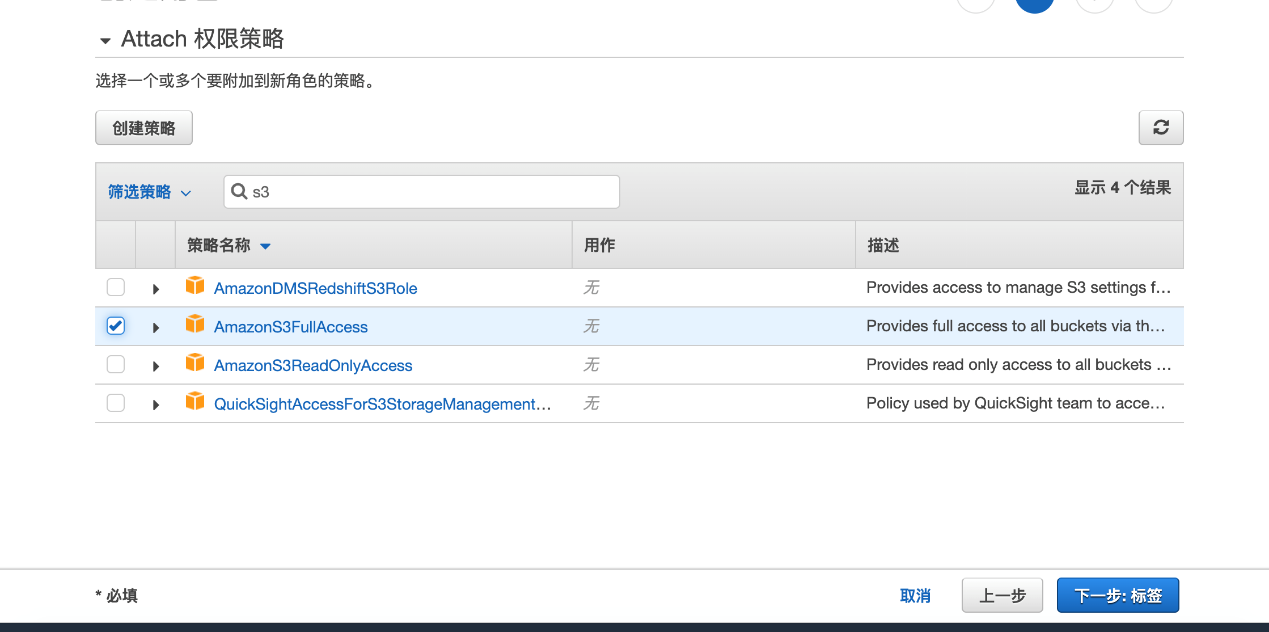
**实验原理**

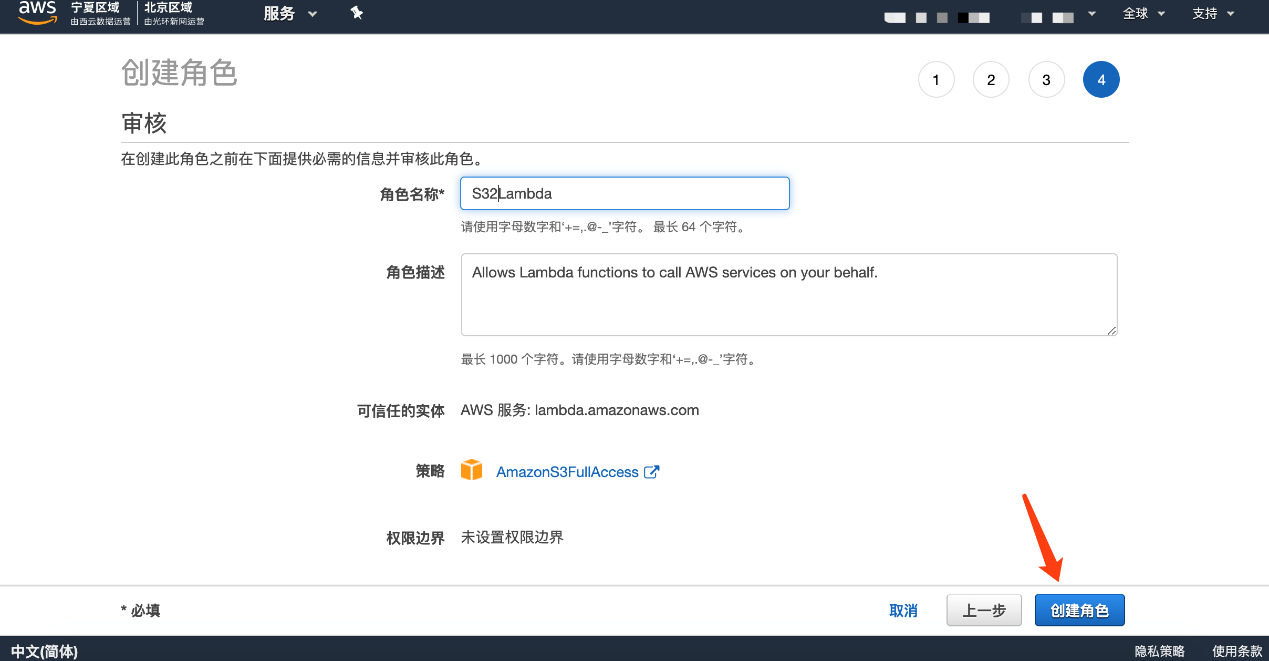
您可以启用特定的 Amazon S3 存储桶事件，以便每次发生这些事件时都向目标发送通知消息。

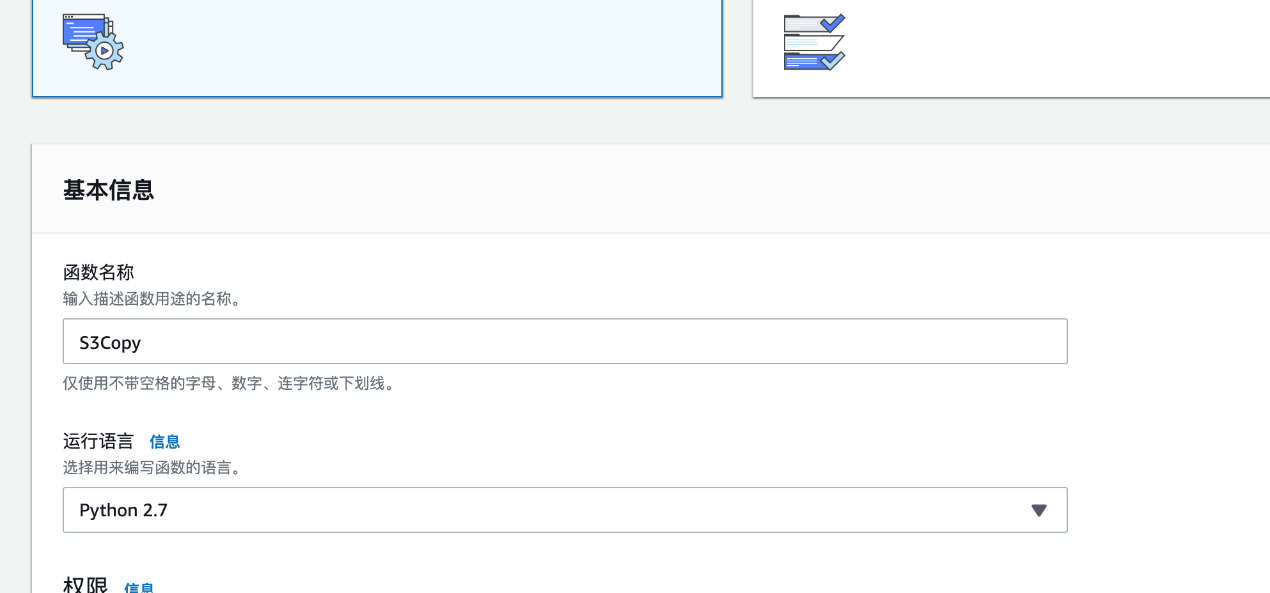
**实验步骤**

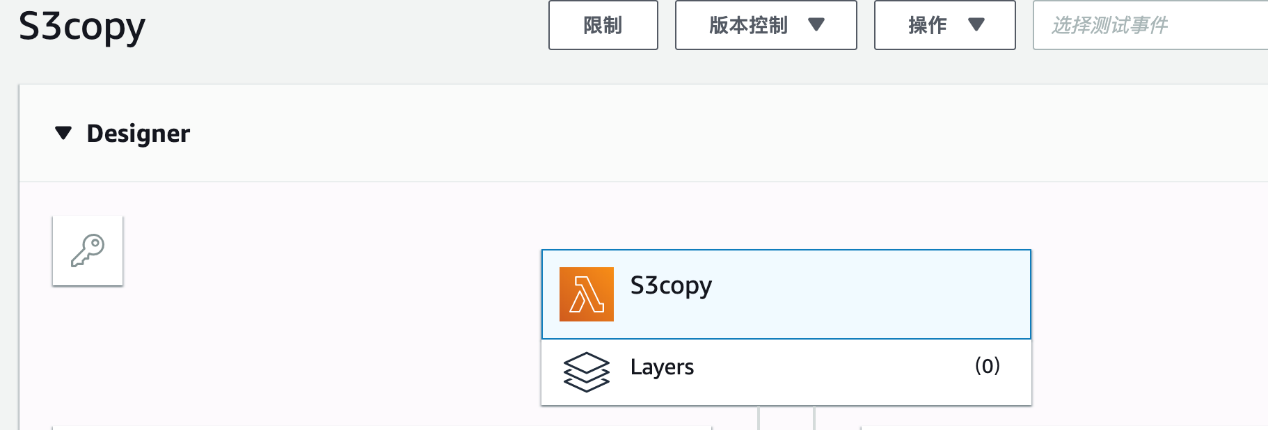
1.创建一个实验用的存储桶。另外一个作为备份桶。  


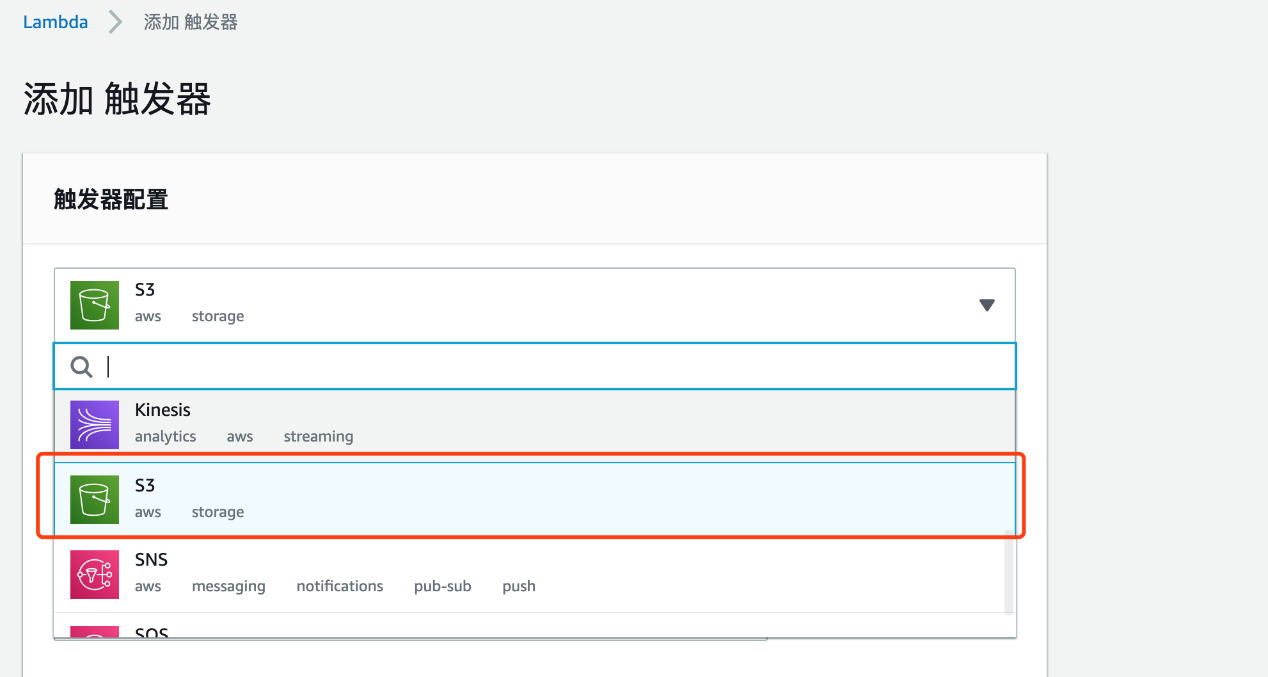
2.首先创建一个IAM角色用于执行Lambda，将使用此角色的服务选择Lambda，点击【下一步】  


3.在点击下一步之后再附加的策略上搜索S3FullAccess，并且选择该策略。点击【下一步】  


4.下一步，输入函数名称，最后点击【创建角色】  


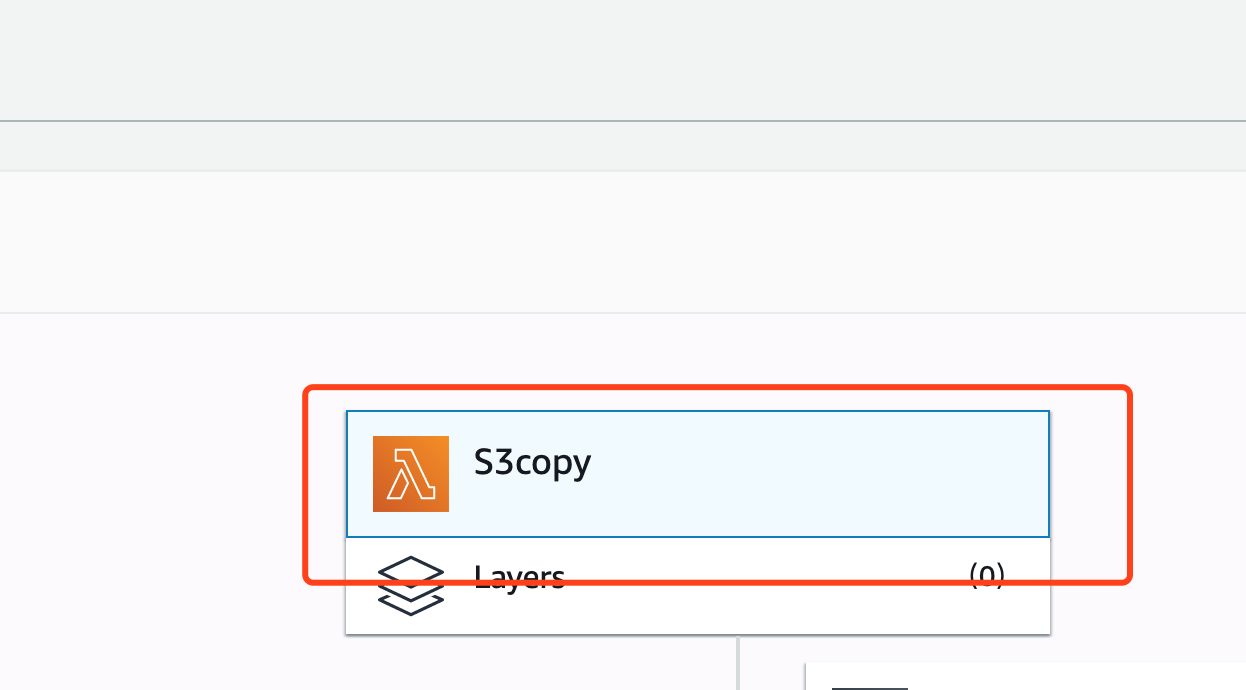
5.切换到Lambda服务中，创建一个Lambda函数用于被触发。定义一个函数名称。  
语言环境上选择python2.7  
  
并且角色选择刚刚创建的角色，最后点击【创建函数】。  


6.创建完成后，可以看到如下界面，这时点击添加触发器。  


7.触发器配置上选择S3  


8.选择刚刚创建的存储桶，事件类型上选择“所有对象创建时间”  


9.其余保持默认，勾选立即启用触发器点击【添加】  


10.在添加完成后点击创建的函数。  


11.添加一下代码到函数中: #bakbuk为你创建的备份桶，需要修改为你创建的桶

import json

import boto3

import urllib

buk = boto3.resource('s3')

def lambda\_handler(event, context):

bucket = event['Records'][0]['s3']['bucket']['name']

key = urllib.unquote\_plus(event['Records'][0]['s3']['object']['key'].encode('utf8'))

copy\_source = {

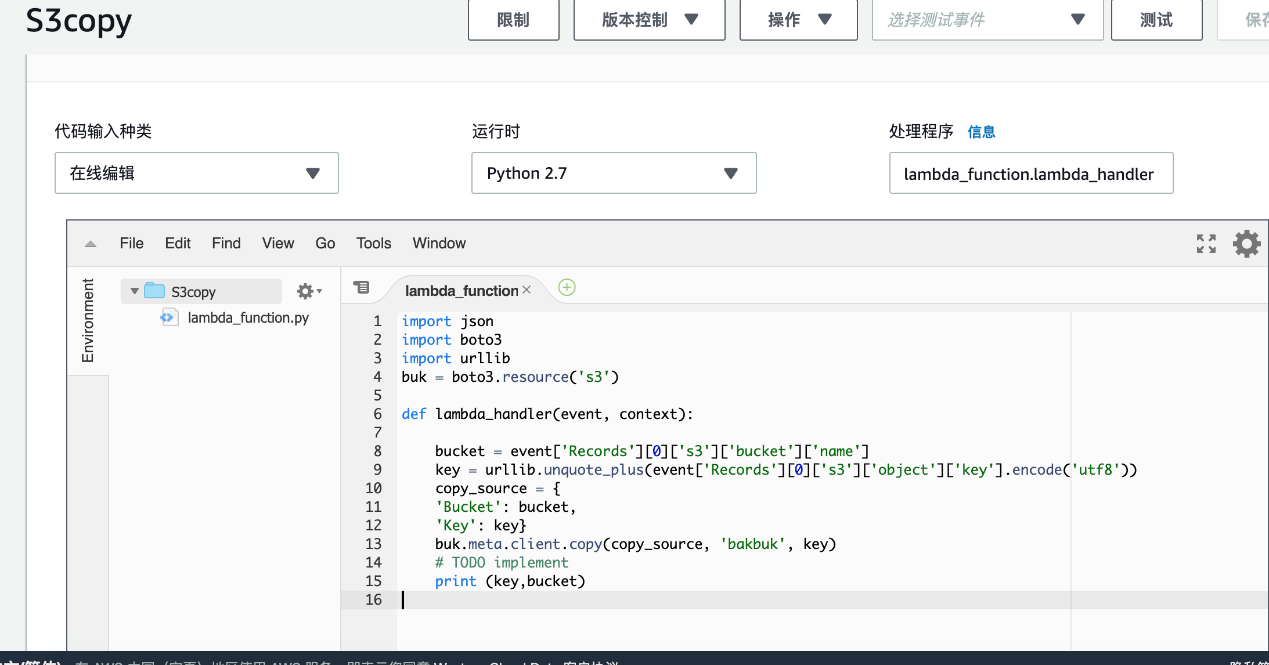
'Bucket': bucket,

'Key': key}

buk.meta.client.copy(copy\_source, 'bakbuk', key)

# TODO implement

print (key,bucket)

添加完成后点击【保存】  


**实验结果**

随机上传一个文件后可以看到文件被自动备份到了另外一个桶中  
  
