

文章

[Hao Ma](#) · 一月 15, 2021 阅读大约需 3 分钟

IAM实践指南——OAuth 2.0下的API保卫战（第二部分）

在这个由三部分组成的系列文章中，我们将展示如何在OAuth 2.0标准下使用IAM简单地为IRIS中的未经验证的服务添加安全性。

在[第一部分](#)中，我们介绍了一些OAuth 2.0背景知识，以及IRIS和IAM的初始定义和配置，以帮助读者理解确保服务安全的整个过程。

现在，本文将详细讨论和演示配置IAM所需的步骤——验证传入请求中的访问令牌，并在验证成功时将请求转发到后端。

本系列的[最后一部分](#)将讨论和演示IAM生成访问令牌（充当授权服务器）并对其进行验证时所需的配置，以及一些重要的最终考虑事项。

如果您想试用IAM，请联系InterSystems销售代表。

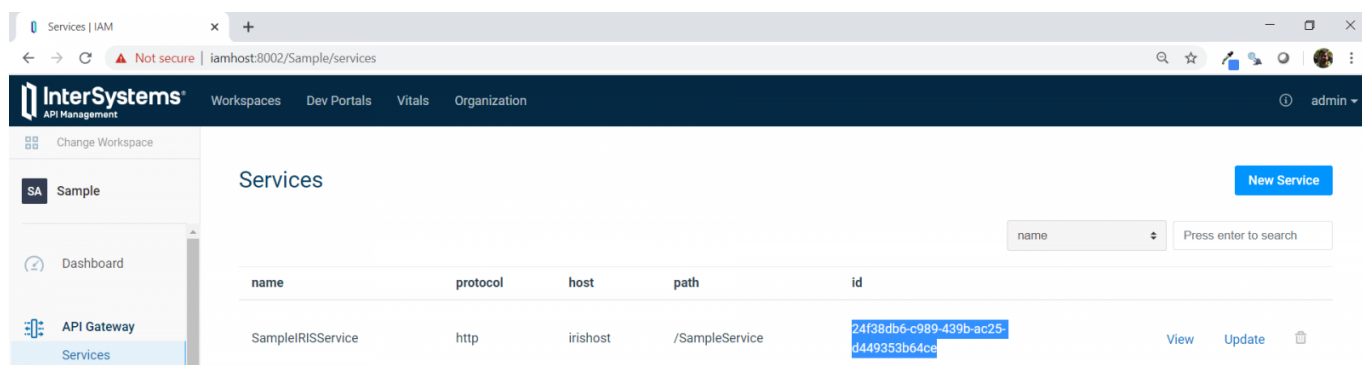
场景1：IAM作为访问令牌验证器

在该场景中，需要使用一个外部授权服务器生成JWT（JSON Web Token）格式的访问令牌。该JWT使用了RS256算法和私钥签名。为了验证JWT签名，另一方（本例中是IAM）需要拥有授权服务器提供的公钥。

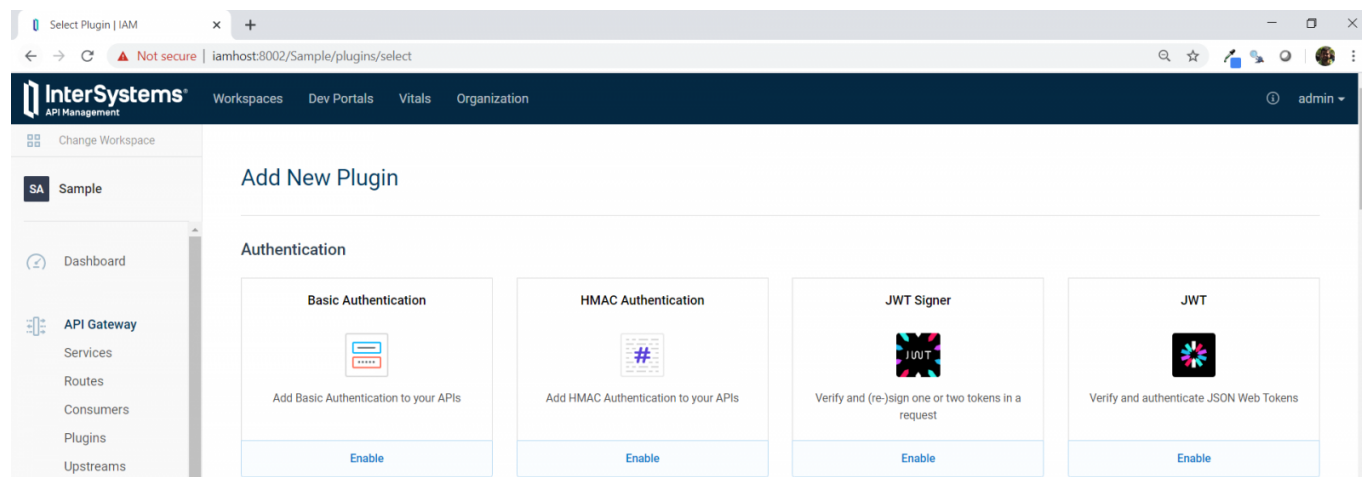
由外部授权服务器生成的JWT主体中还包括一个名为“exp”的声明（包含该令牌过期的时间戳），以及另一个名为“iss”的声明（包含授权服务器的地址）。

因此，IAM需要先使用授权服务器的公钥和JWT内部“exp”声明中包含的过期时间戳对JWT签名进行验证，然后再将请求转发给IRIS。

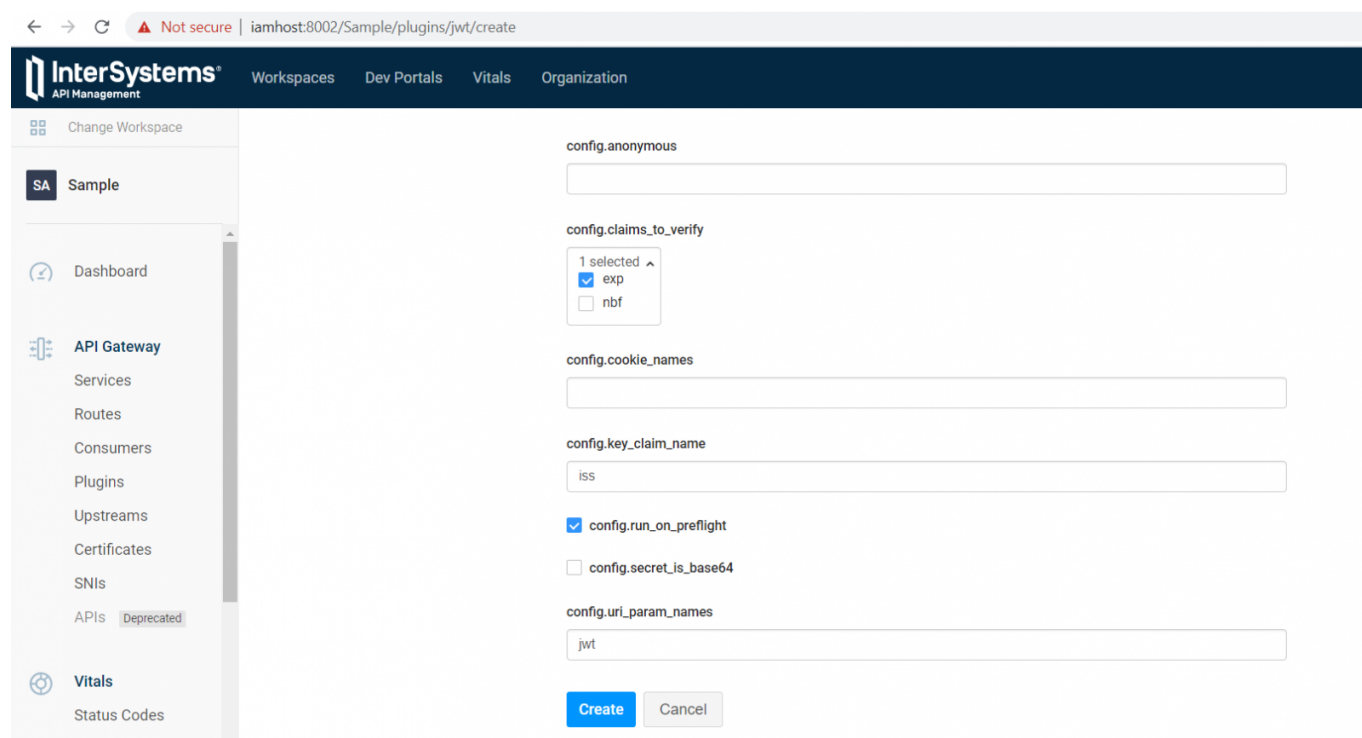
对IAM进行相应配置时，首先要向IAM中的“SampleIRISService”添加一个名为“JWT”的插件。为此，请转到IAM中的Services页面并复制“SampleIRISService”的ID，稍后会用到。



之后，打开插件，点击“New Plugin”按钮，找到“JWT”插件，点击启用。



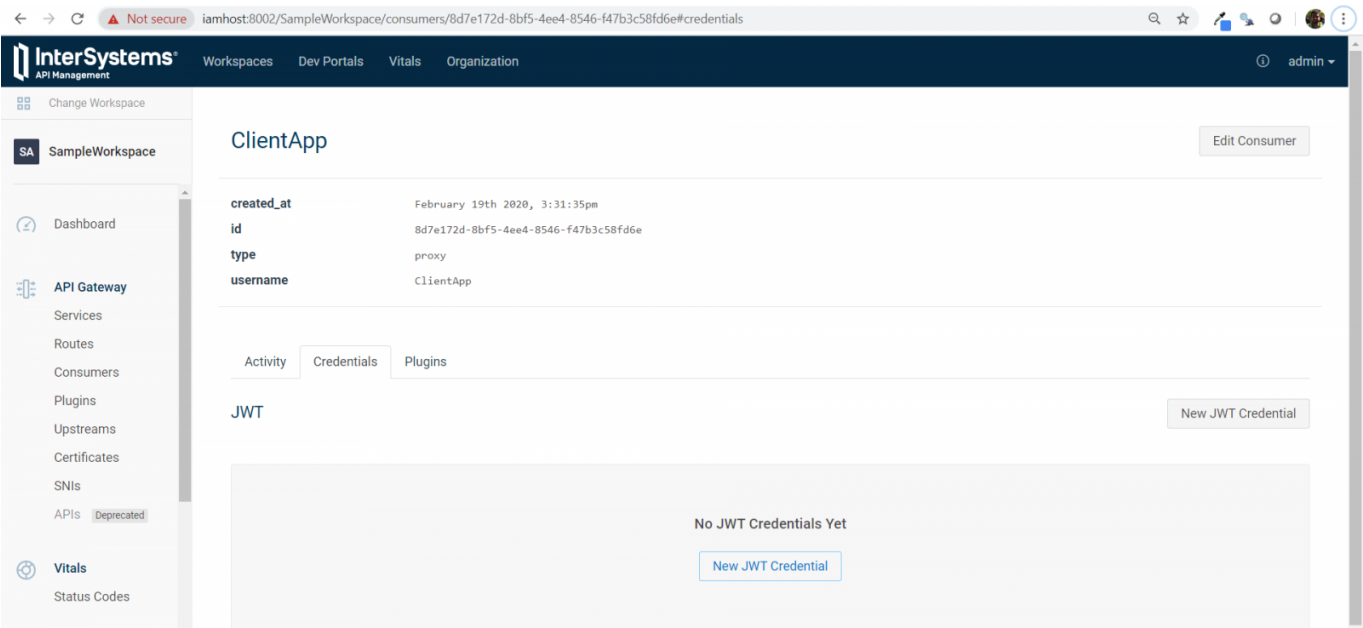
在下个页面中，将“SampleIRISService”ID粘贴在“serviceid”字段中，然后在“config.claimstoverify”参数中选中“exp”框。



注意，“config.keyclaimname”参数的值是“iss”。后面会用到。

然后，点击“Create”按钮。

完成操作后，找到左侧菜单中的“Consumers”部分，然后单击先前创建的“ClientApp”。点击“Credentials”标签，然后单击按钮“New JWT Credential”。



在下一页中，选择JWT签名算法（本例中为RS256），并将公钥（这是授权服务器提供的PEM格式的公钥）粘贴到“rsapublickey”字段中。

在“key”字段中，在添加JWT插件时需要用到之前在“config.keyclaimname”字段中输入的JWT声明内容。所以在本例中，需要插入的是JWT的iss声明内容（本例中是授权服务器的地址）。

← → ↺

Not secure | iamhost:8002/Sample/consumers/ebf70f90-6342-47dc-877e-a2f4b45657cd/jwt/create

InterSystems®
API Management

Workspaces

Dev Portals

Vitals

Organization

Change Workspace

SA Sample

Dashboard

API Gateway

Services

Routes

Consumers

Plugins

Upstreams

Certificates

SNIs

APIs Deprecated

Vitals

Status Codes

Dev Portal

Create JWT Credential

key

https://authorizationserver:5001

algorithm

RS256

rsa_public_key

-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqt6NlnZQ8Ty9wfpXhPx
G16zV7Svd2DhJ3j6ZS/zfkThSeuo11fTcoxt8Ma5orORQDfRKLtwYd9KnDR93Y
I6JFyPDEzAoXkV9RRig2NiOHoOR368mu7NbA2R3F8jqlER42R24/AKUTkU9LPjus
KZBUcADImLg9698GM5125x2s4i2U+T68jERRO4iEIPG9dV/K98Y+vZ8XN29kb4AOK
OtK04ZFCrN2R6GbpvACShqczMqxb4q3BZnB/RVkrFv4sW0AdO5jie7WKpP8XHeT
3eifdt77uholjUSvpqs6VLYAc9xEpxCmzu63XNqVV7S8Qnzs05wyJk5LytdtEnnH
XwIDAQAB
-----END PUBLIC KEY-----

secret

Create

Cancel

之后，单击“Create”按钮。

提示：出于调试目的，可以使用一个在线工具对JWT进行解码，将公钥粘贴进去就可以检查声明内容及其值，并且验证签名。该在线工具的链接如下：<https://jwt.io/#debugger>

现在，添加了JWT插件后，就不能发送未经身份验证的请求了。如下所示，对URL的一个简单GET请求（未经身份验证）：

<http://iamhost:8000/event/1>

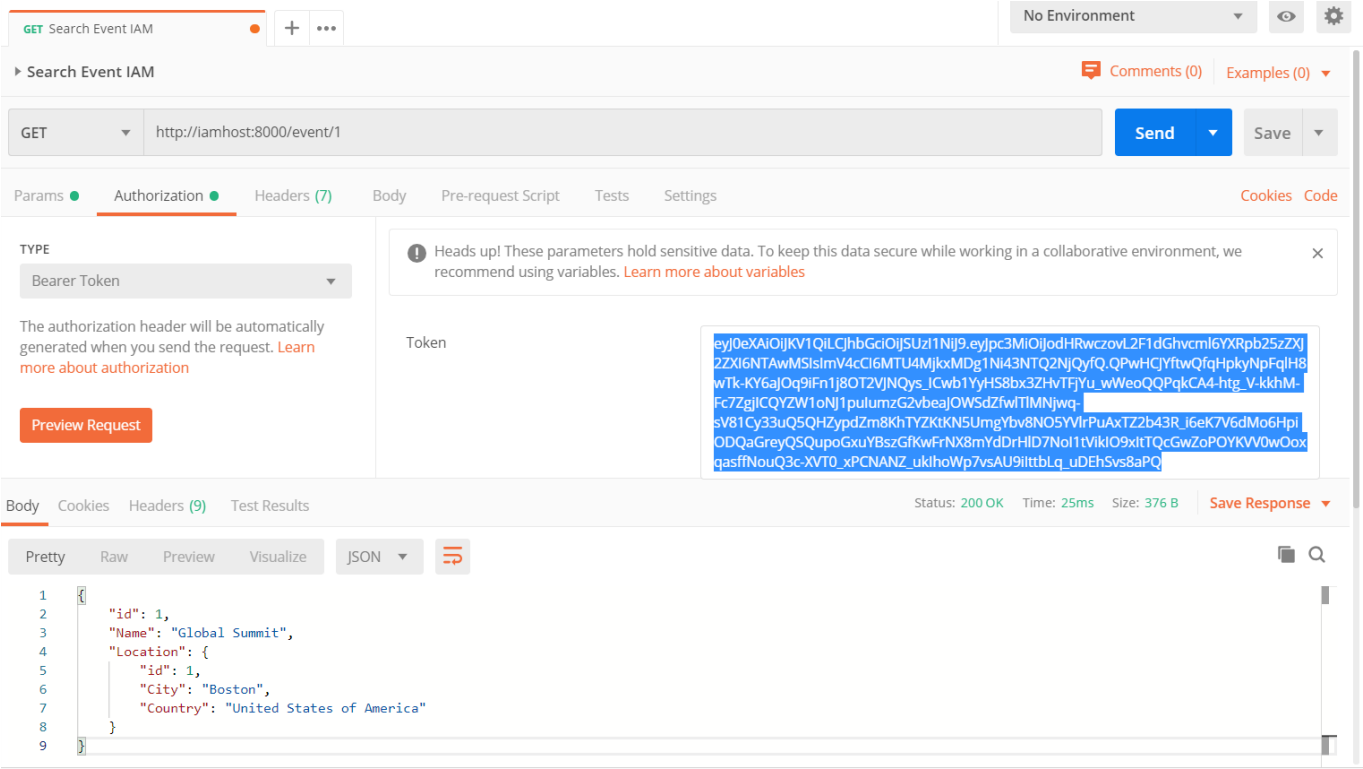
返回一个未经授权的信息，以及状态码“401未经授权”。



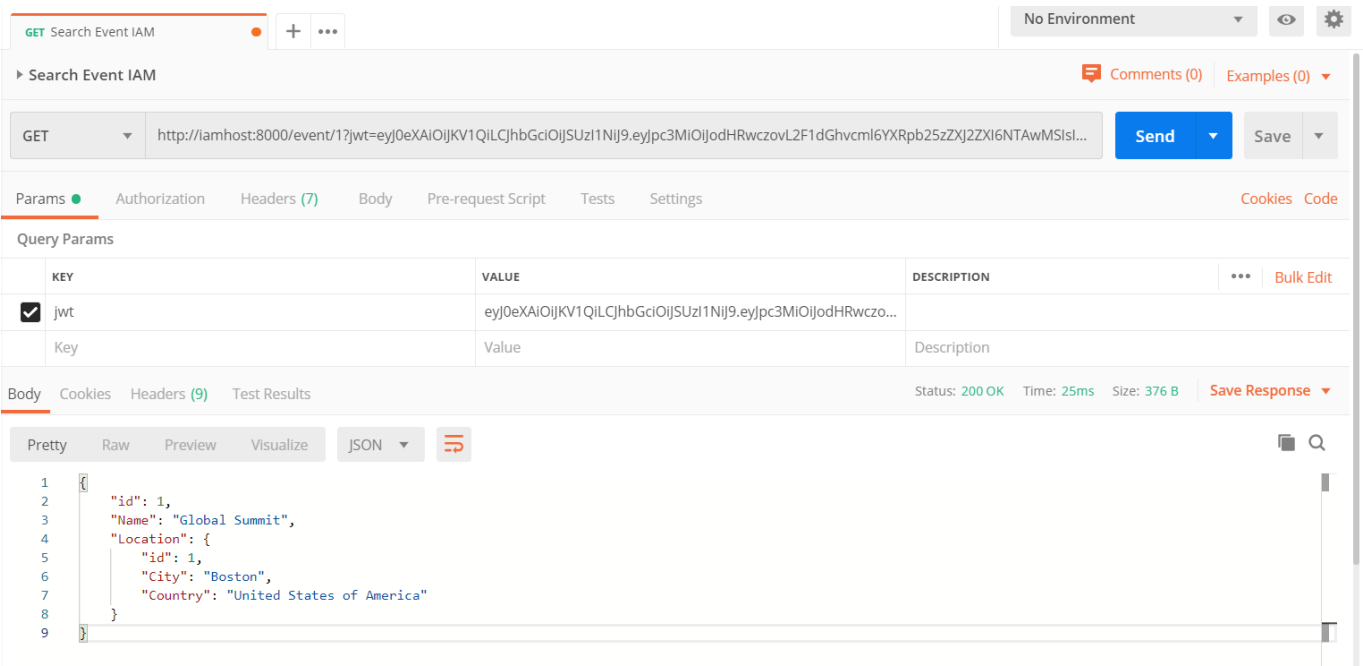
<https://authorizationserver:5001/auth>

[illegible]

<http://iamhost:8000/event/1>



或者将它作为querystring参数添加进去。当添加JWT插件（本例中是“jwt”）时，querystring关键字是在“config.uriparams”字段中指定的值



最后，如果在“config.cookieparams”字段中输入任意名称，选择将JWT作为cookie包含在请求中。

请继续阅读本系列的第三部分也即最后一部分，了解IAM生成和验证访问令牌所需的配置，以及一些重要的最终考虑因素。

#API #OAuth2 #REST API #安全 #InterSystems IRIS
