

了解并使用Catalyst Center API

目录

[简介](#)

[背景信息](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[HTTP简介](#)

[HTTP请求](#)

[HTTP响应](#)

[HTTP方法 \(动词 \)](#)

[HTTP状态代码](#)

[REST概述](#)

[Catalyst中心和REST API](#)

[使用Bruno检索身份验证令牌](#)

[1.确定API终端](#)

[2.配置Bruno进行身份验证](#)

[3.在Bruno中设置全局标题](#)

[示例 1：创建站点楼层](#)

[步骤 1：获取父ID](#)

[步骤 2：创建楼层](#)

[示例 2：检索接口信息](#)

[步骤 1：识别设备ID](#)

[步骤 2：在Bruno中创建参数化请求](#)

[结论](#)

简介

本文档介绍如何使用API客户端和平台内置开发人员工具访问和使用CaTC北向 (意图) API。

背景信息

它涵盖基本HTTP/REST概念，并提供自动化和测试的实际示例。

先决条件

要求

建议掌握下列主题的相关知识：

- HTTP/1.1协议和状态代码。
- REST架构风格。
- 基本了解Cisco Catalyst Center(CaTC)
- 基本了解Bruno (或类似API客户端，如Postman)。

使用的组件

本文档中的信息基于下列硬件和软件版本：

- Cisco Catalyst Center版本2.3.7.x和3.x
- Bruno版本4.1.0

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始 (默认) 配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

Catalyst中心北向API(Intent API)允许用户与图形用户界面之外的控制器交互。这样可以实现定制工作流程的自动化、与第三方工具的集成，以及规模化的高效网络管理。

HTTP简介

超文本传输协议(HTTP)是Web数据通信的基础。它遵循客户端 — 服务器模型，在该模型中，客户端发送请求，而服务器返回响应。

HTTP请求

HTTP请求包括四个部分：

1. 请求行：指定方法（包括GET和POST）、URI和协议版本（例如GET /api/v1/nodes HTTP/1.1）。
2. 标头：提供元数据，例如身份验证凭据和内容类型。
3. 空行：将信头与正文分开。
4. 邮件正文（可选）：包含发送到服务器的数据（常见于POST/PUT/PATCH）。

HTTP响应

HTTP响应反映请求结构：

1. 状态行:包括状态代码（例如200 OK）。
2. 标头：有关服务器和响应数据的详细信息。
3. 邮件正文：请求的实际数据（对于Catalyst Center通常采用JSON格式）。

HTTP方法（动词）

- GET :从资源中检索数据。
- POST:创建新资源。
- PUT:替换现有资源。
- 补丁程序:将部分修改应用于资源。
- DELETE :删除资源。

HTTP状态代码

HTTP状态代码是服务器作为HTTP响应的一部分返回的标准化数字代码。它们表示客户端请求的一般结果，并有助于了解请求是否成功、是否需要执行其他操作或是否遇到错误。

这些代码按其第一个数字分为五个主要类别。

- 1xx代码仅供参考，表示已收到请求且正在继续处理。
- 2xx代码表示请求成功，表示服务器成功接收、理解并处理了请求。
- 3xx代码用于重定向，表示可能需要执行其他操作才能完成请求。
- 4xx代码表示客户端错误，这意味着客户端发送的请求存在问题。
- 5xx代码代表服务器错误，这意味着服务器在尝试处理请求时遇到问题。

REST概述

REST（具象状态传输）是设计网络应用的体系结构风格。RESTful API使用HTTP作为传输机制

，并且通常使用JSON进行数据交换，因为它具有轻量性质。

当客户端与REST API交互时，服务器返回资源的当前状态的表示形式。此表示作为响应的一部分传输到客户端，并且可以用不同的格式表示。

REST不是协议。相反，它是一种分层式风格，定义了一系列构建分布式系统和API的原则和限制。在现代API中，JSON是最常用的格式之一，因为它轻量且易于处理。

典型的REST交互遵循基于客户端的模型（这已在文章中提及）。客户端使用GET、POST、PUT、PATCH或DELETE等方法发送HTTP请求。服务器处理并返回包含请求的信息或操作结果的HTTP响应。

RESTful API通常使用HTTP作为通信机制，以对资源执行操作。在此模型中，客户端标识资源/终结点，然后使用HTTP方法指示要对该资源执行的操作。REST API可以使用几个主要元素进行描述。首先，URI用于标识客户端要访问的资源或端点。URI提供该资源在API中的位置或标识符。其次，客户端使用标准HTTP方法（如GET、POST、PUT、PATCH和DELETE）来定义请求操作。例如，GET可用于检索资源，POST用于创建资源，而DELETE则完全删除资源。最后，资源通过表示形式进行交换，表示形式由资源介质类型定义。这种表示方式是在客户端和服务器之间传输资源数据的形式，如JSON或XML。

Catalyst中心和REST API

CaTC通过集中自动化、配置、策略管理、监控和分析，帮助简化园区网络运营。网络管理员可以更高效地部署和管理设备、维护一致的策略、监控网络行为和提高整体运营可视性。这些功能可以通过图形用户界面(GUI)以及通过API和外部集成来访问。

CaTC平台还提供可用于构建自定义集成和网络操作自动化的API。通过将CaTC API与现有思科解决方案或第三方应用相结合，组织可以扩展平台功能并创建针对特定运营或业务需求的定制工作流程。

在介绍了基本概念之后，本文档重点介绍Cisco Catalyst Center中的北向API（也称为目的API）。这些API通过基于策略的业务意图抽象化来展示特定的Catalyst Center功能，使管理员和开发人员可以专注于期望的结果，而不是实现目标所需的单个配置步骤。

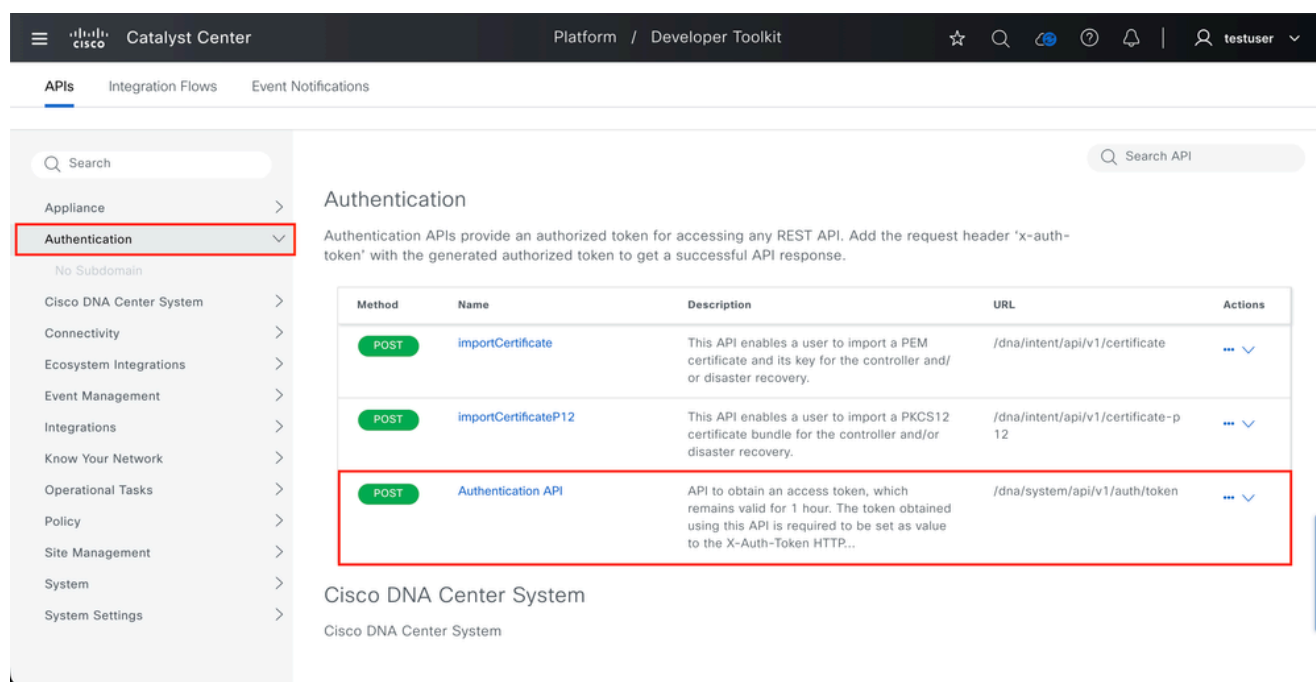
Catalyst Center Intent API是使用JSON数据格式与平台交互的RESTful API。它们可用于在整个网络中发现网络信息、检索操作数据，以及执行支持的控制和自动化任务。

使用Bruno检索身份验证令牌

在进行API调用之前，您必须进行身份验证才能接收令牌。

1. 确定API终端

1. 登录到Catalyst Center。
2. 导航到菜单>平台>开发人员工具包。
3. 在左侧菜单中，选择Authentication > Authentication API。

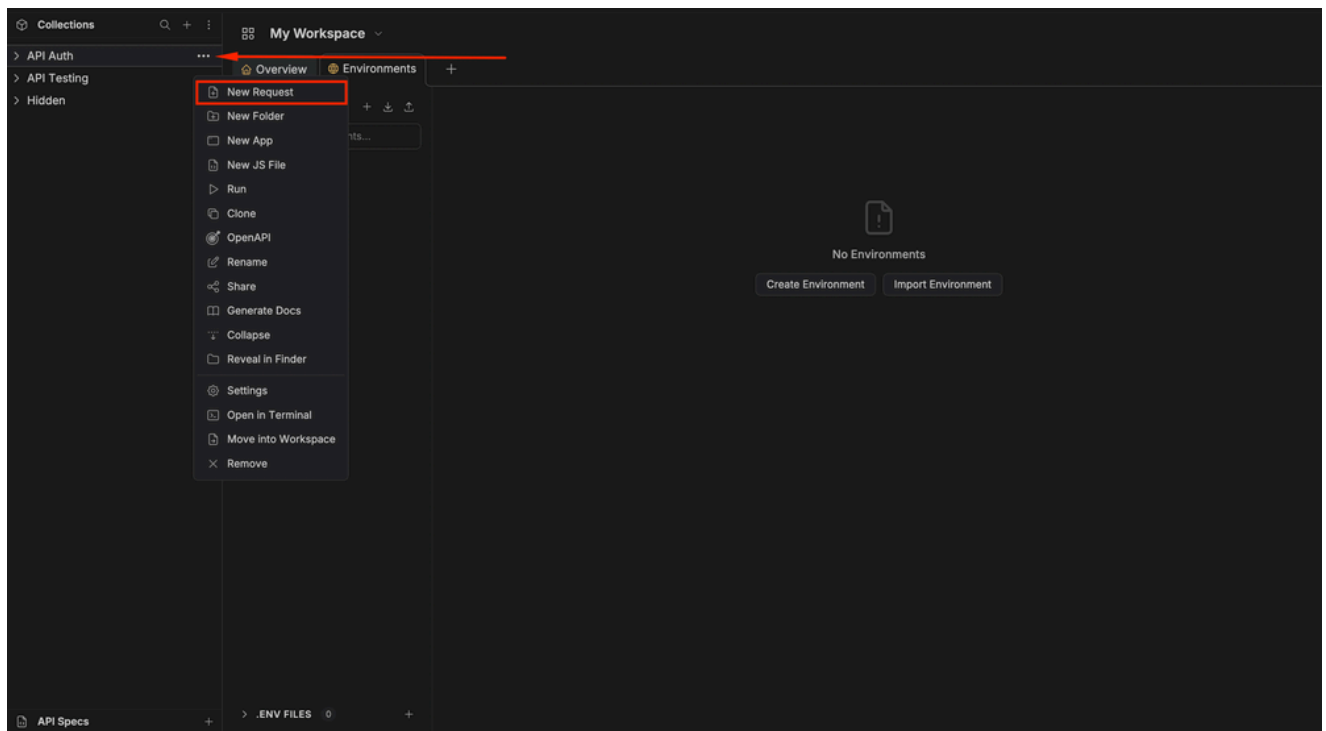


Catalyst Center开发人员工具包

4. 复制URL (通常为POST /dna/system/api/v1/auth/token)。

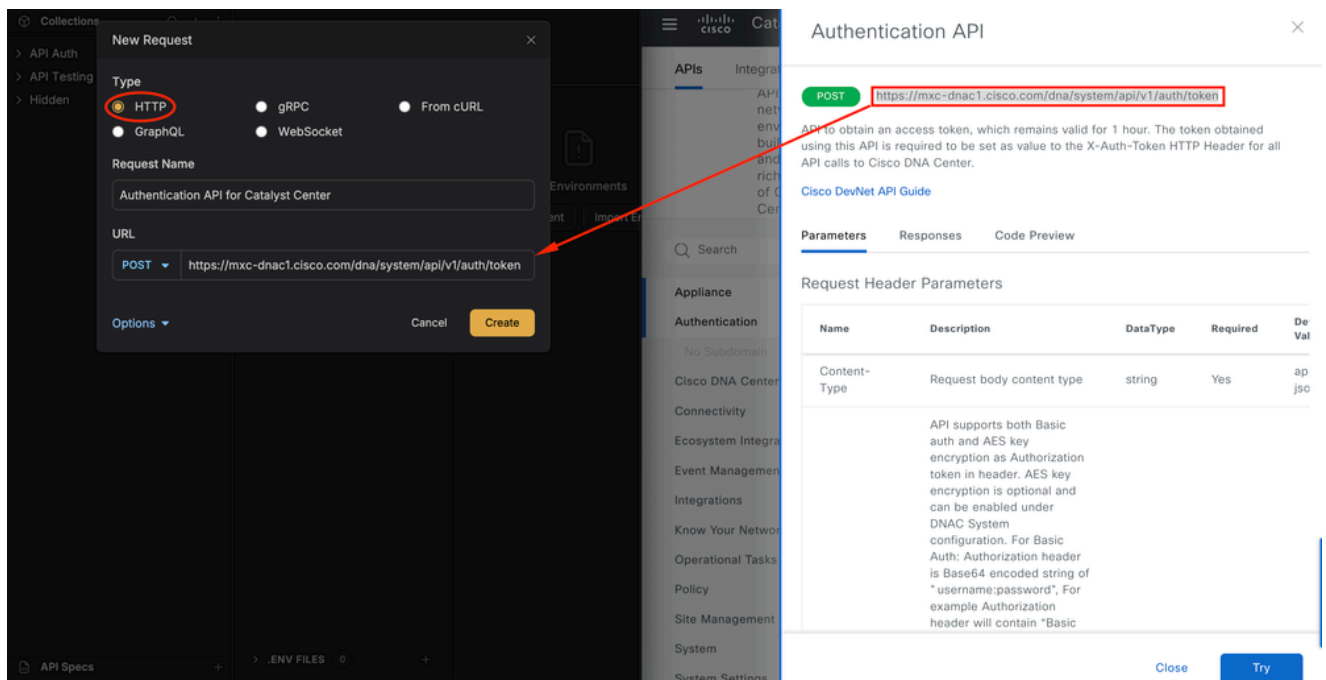
2. 配置Bruno进行身份验证

1. 在Bruno中，创建新请求。
2. 将Type设置为HTTP，将Method设置为POST。



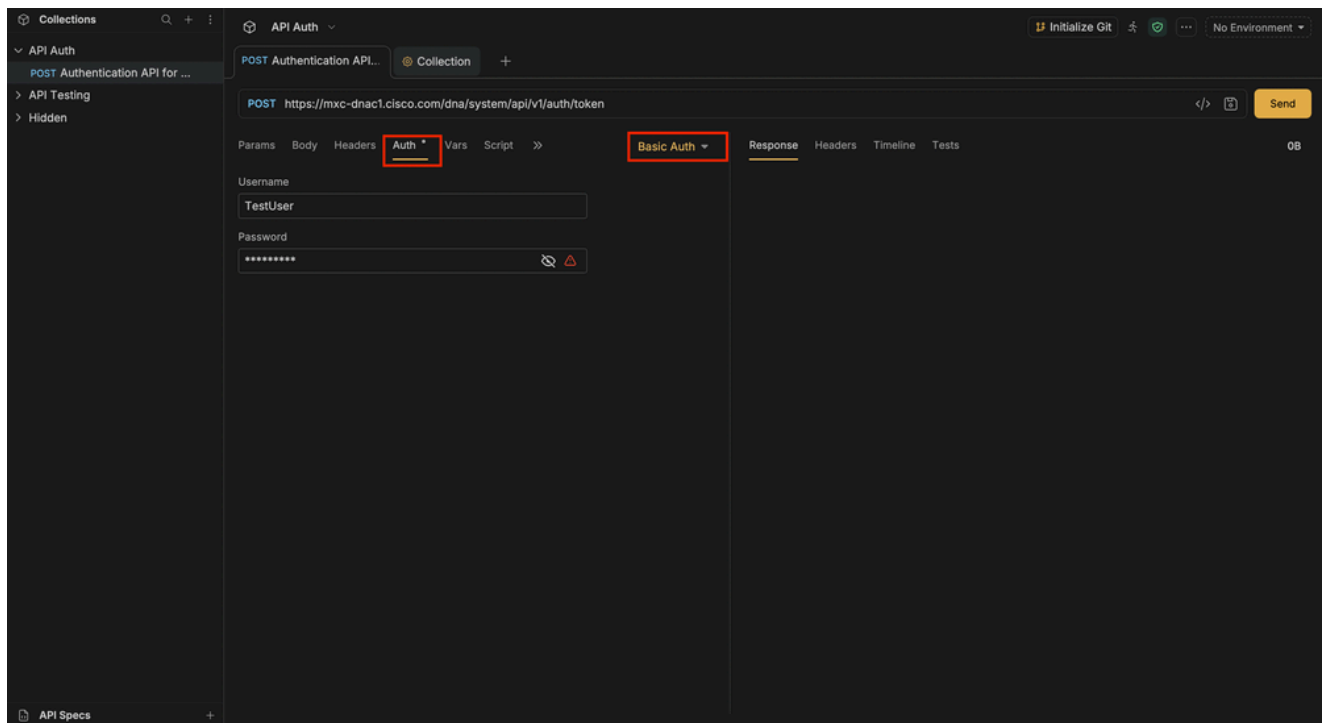
在Bruno创建新请求

3. 输入身份验证API的URL。



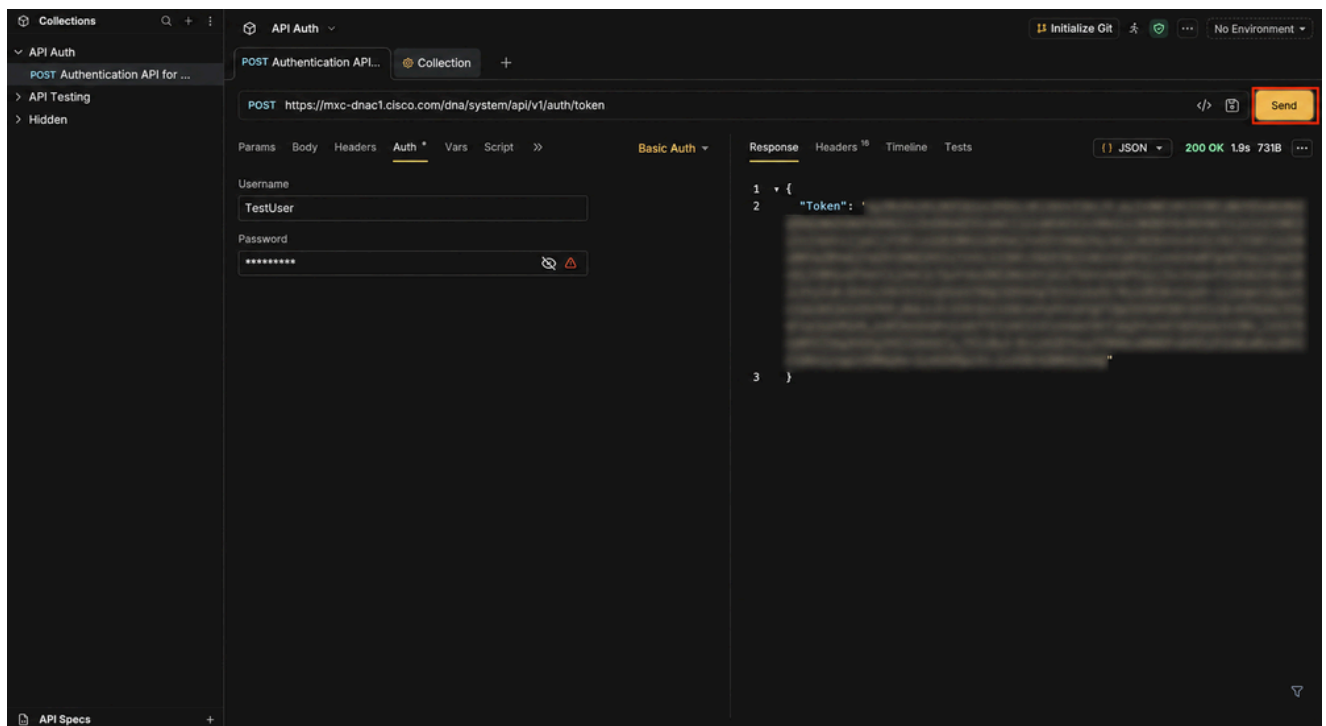
在Bruno创建新请求

4. 导航到Auth选项卡并选择Basic Auth。



身份验证API的基本身份验证

5. 输入您的Catalyst Center SUPER ADMIN凭证。
6. 单击发送。



运行身份验证API

服务器返回200 OK状态和包含令牌的JSON负载。

Alternative (Terminal): You can also generate the token by base64-encoding your `username:password` and

```
[ ~ % echo -n 'TestUser:Test12345678' | base64
VGZzdFVzZXI6VGZzdDEyMyEhIQ==
```

获取用户名和密码的base64编码

Try 'Authentication API'

Method: **POST** Public URL :https://mxc-dnac1.cisco.com/dna/system/api/v1/auth/token

PARAMETERS

HEADERS

Content-Type* application/json

Authorization* Basic VGZzdFVzZXI6VGZzdDEyMyEhIQ==

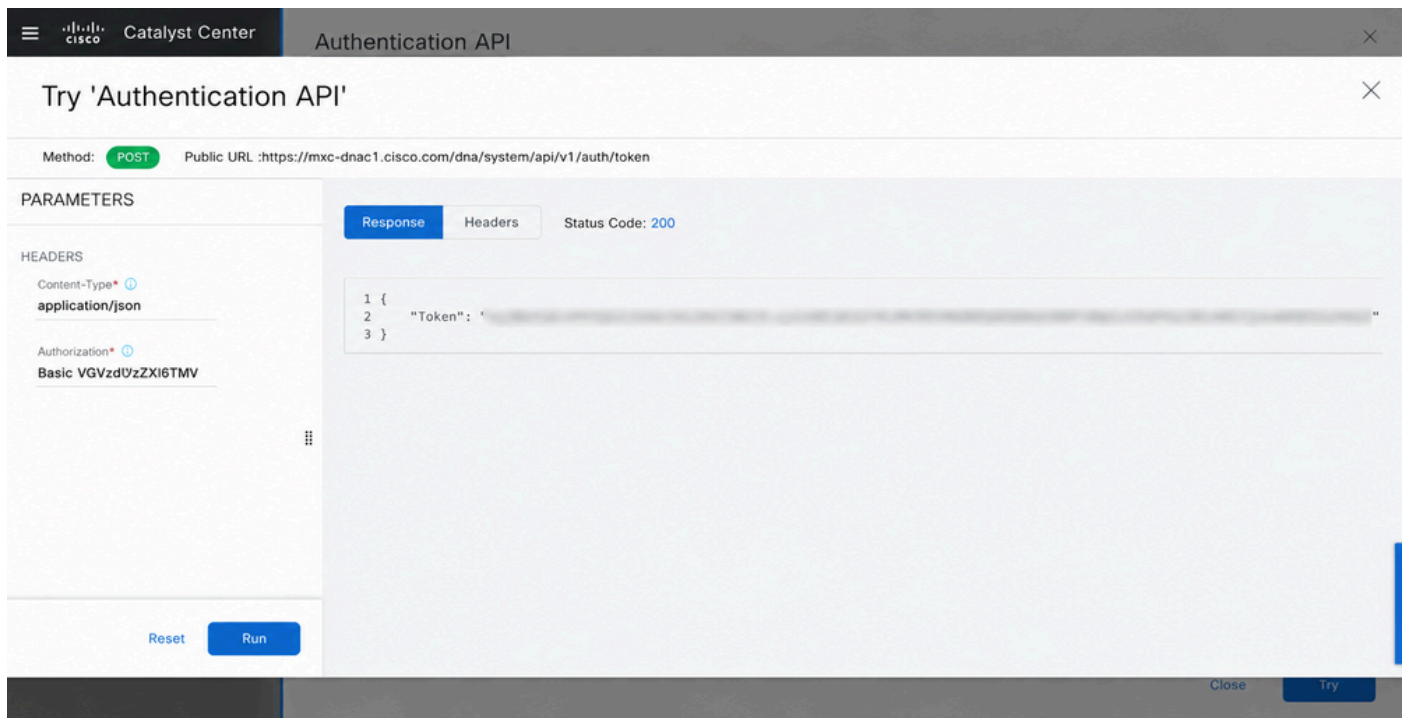
Reset Run

Enter Parameters

First enter in some necessary parameters on the left and then click Run.

Close Try

使用编码的base64用户名和密码输入基本身份验证



从Catalyst Center GUI获取身份验证令牌。

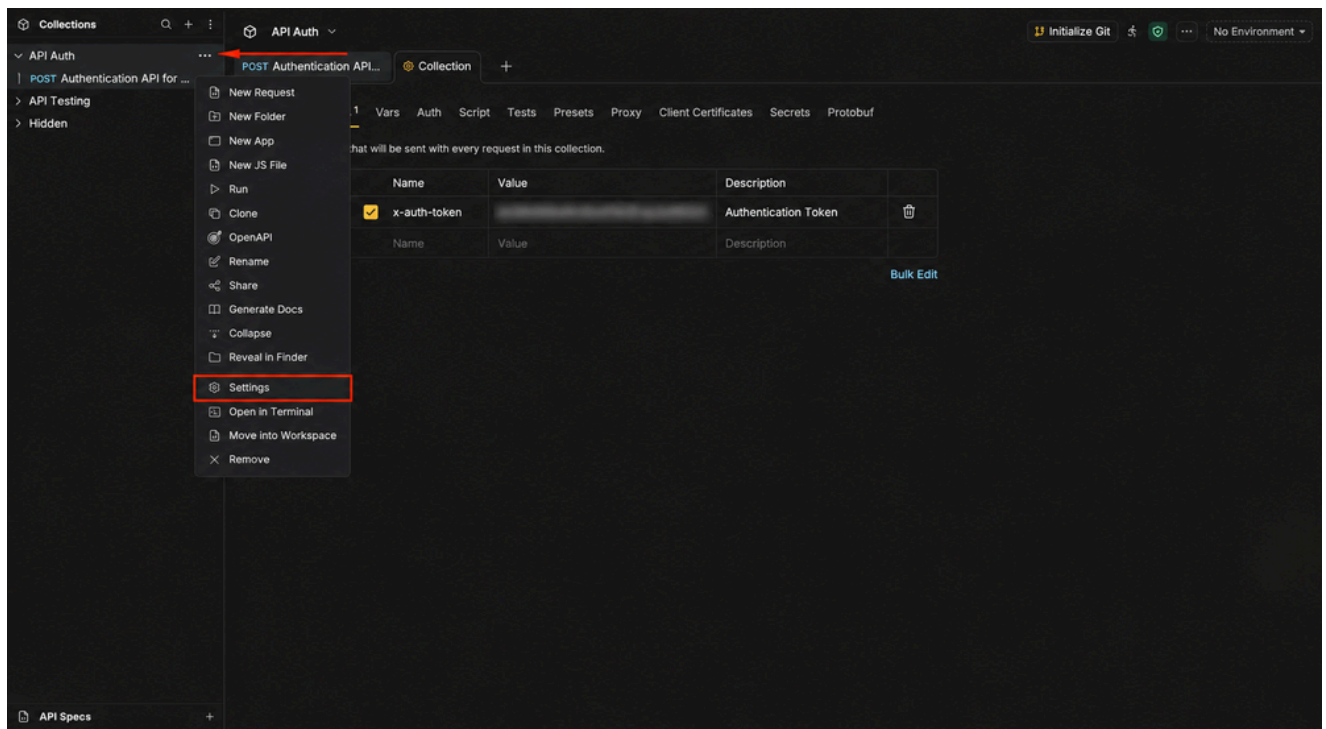
此时，您在Bruno响应或通过Catalyst Center Developer Toolkit完成的响应上拥有身份验证令牌。

现在在Bruno中将令牌添加为集合设置上的Header，这样便于集合内创建的任何其他请求使用该令牌，因此您无需手动将令牌输入为每个请求的标头。

3.在Bruno中设置全局标题

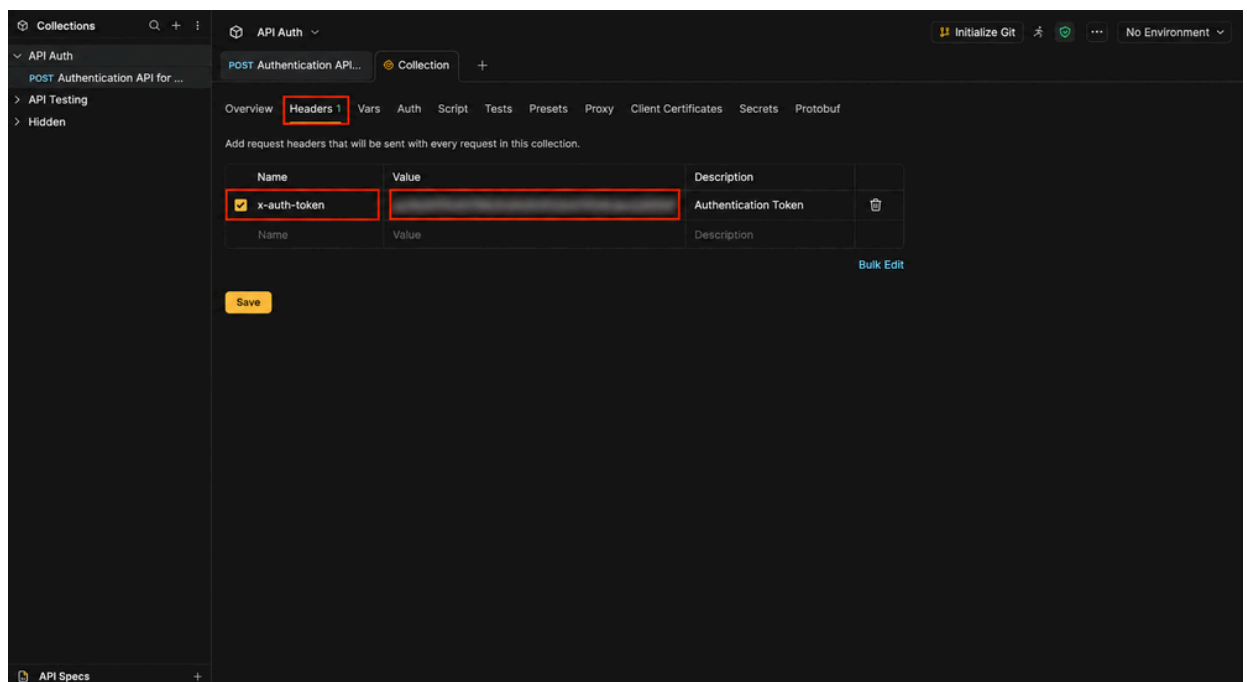
要避免对每个请求进行重新身份验证，请将令牌设置为集合级别报头：

1. 右键单击您的收藏并选择设置。



集合设置

2. 转至Headers选项卡。
3. 添加新信头：
 - 名称：x-auth-token
 - 值： [在此处粘贴令牌]



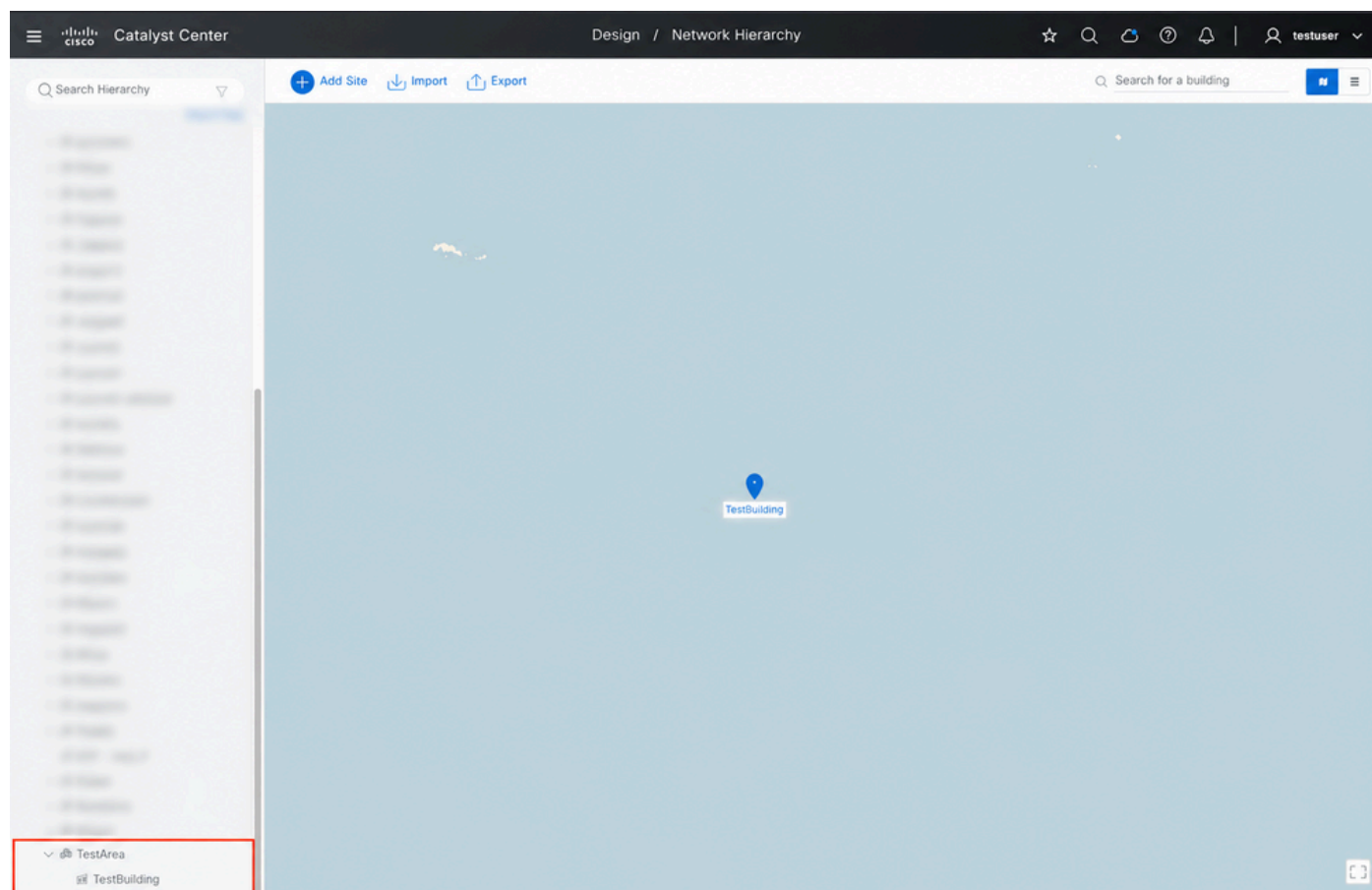
在Bruno中输入API集合的全局标头。

4. Click Save.

此时，您已生成身份验证令牌以执行任何其他Catalyst Center API请求。

示例 1：创建站点楼层

此示例演示如何在网络层次结构中创建楼层(Global > TestArea > TestBuilding)。

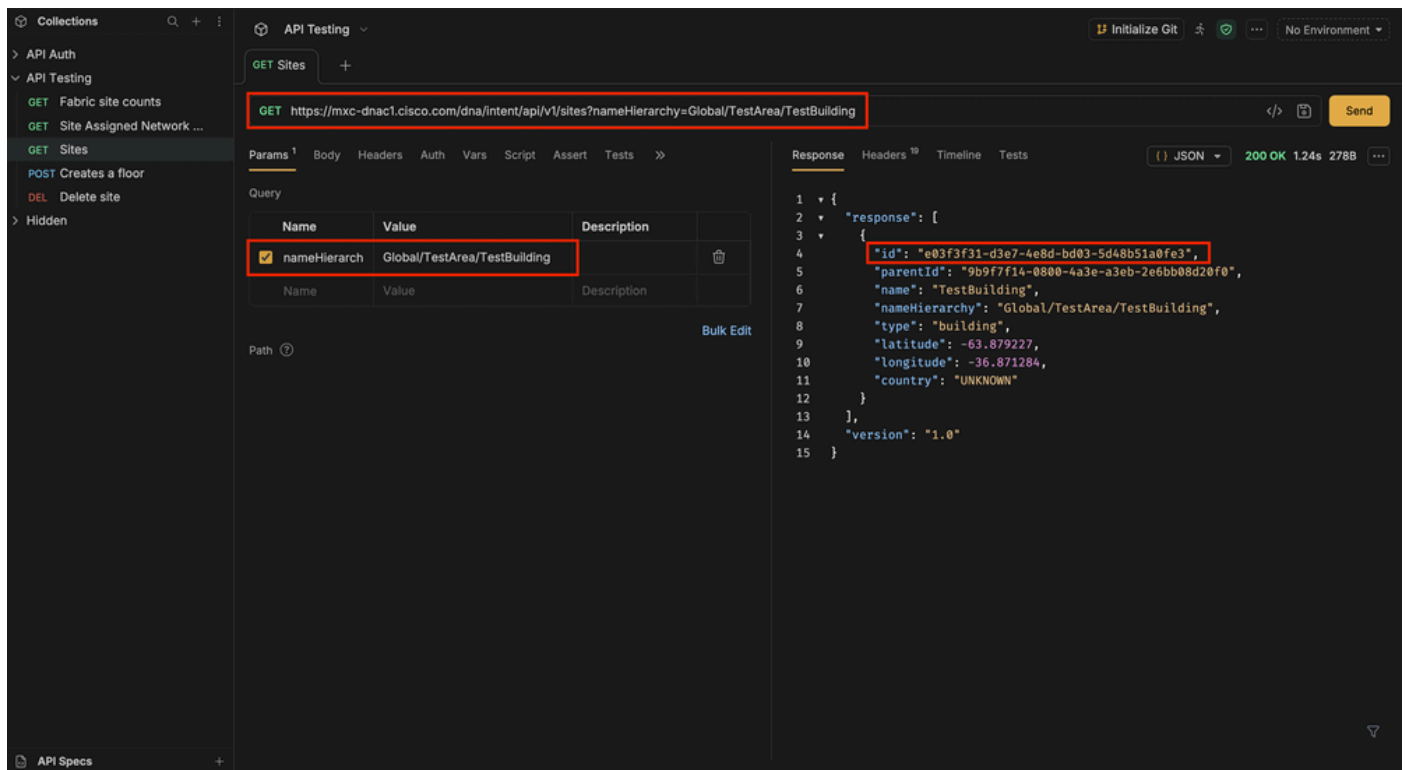


网络层次结构

步骤 1：获取父ID

使用GET /dna/intent/api/v1/sites API查找TestBuilding的ID。

- 查询参数：nameHierarchy=全局/测试区域/测试构建
- 从响应中复制ID;这是你的parentId。



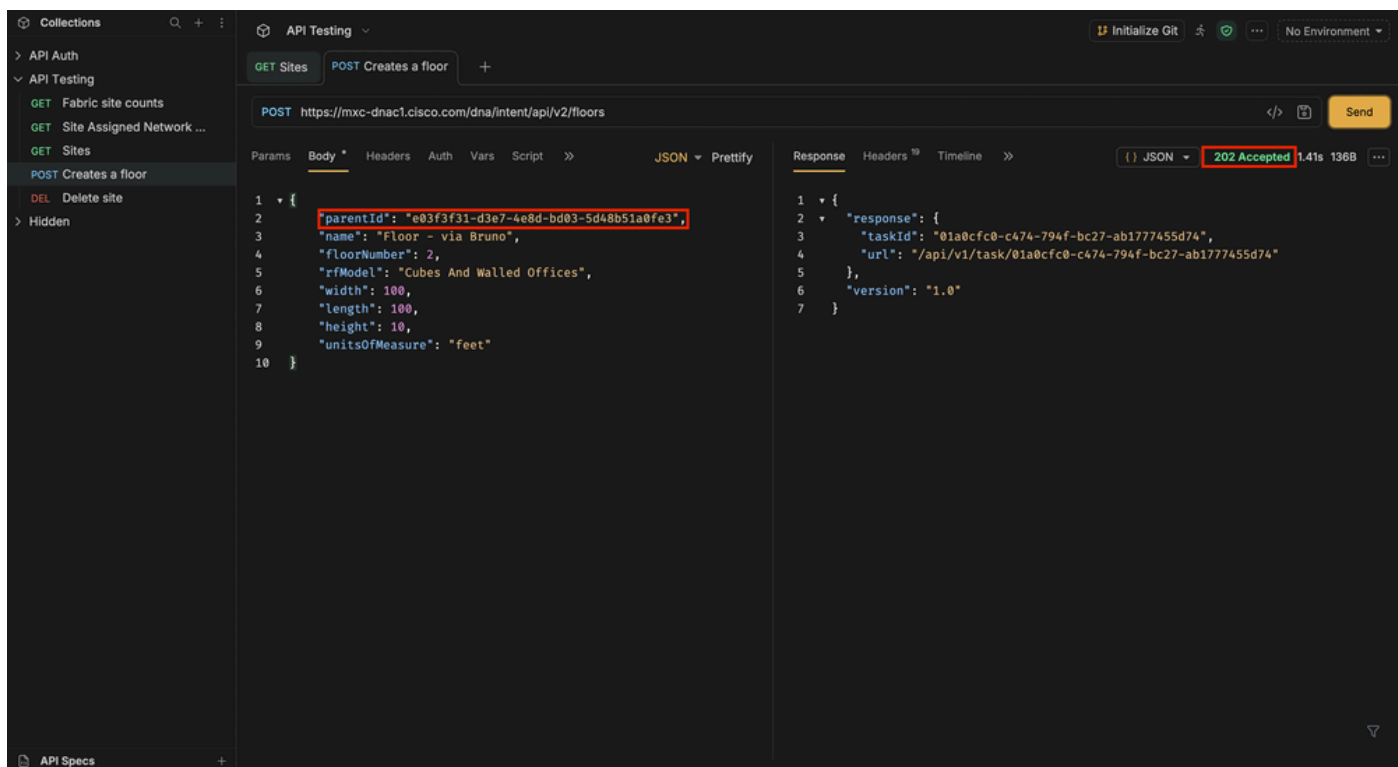
GET请求/dna/intent/api/v1/sites?nameHierarchy=Global/TestArea/TestBuilding

步骤 2：创建楼层

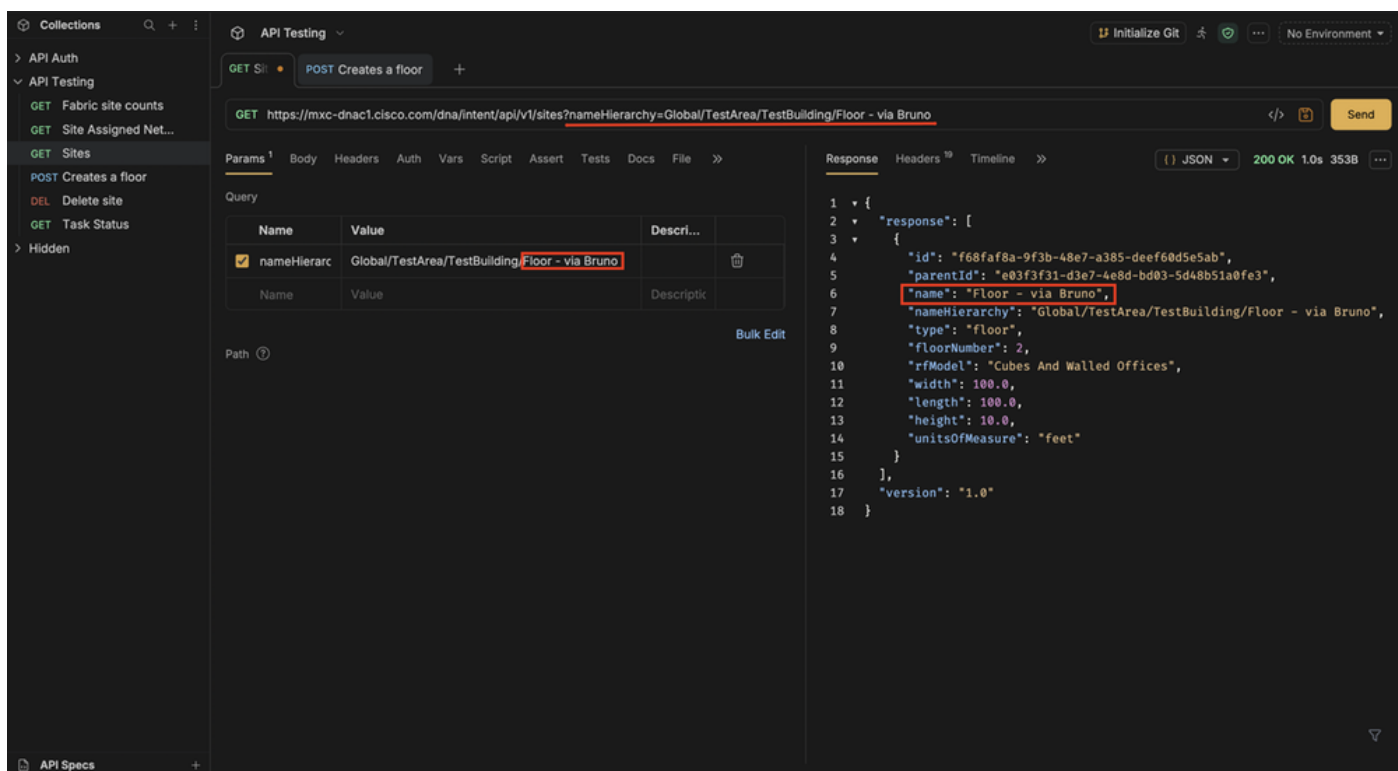
1. 创建/dna/intent/api/v2/floors的POST请求。
2. 在Body选项卡(JSON)中，输入所需的参数：

```
{
  "name": "Floor 1",
  "parentId": "your-building-id-here",
  "floorNumber": 1,
  "rfModel": "Cubic-Office",
  "width": 100,
  "length": 100,
  "height": 10,
  "unitsOfMeasure": "feet"
}
```

3. 单击发送。成功响应确认已创建楼层。



创建楼层的POST请求



通过POST请求验证Floor创建的GET API。

最后，要验证已成功创建楼层，可以运行之前的API并修改查询以匹配完整的层次路径，如上所示。

示例 2：检索接口信息

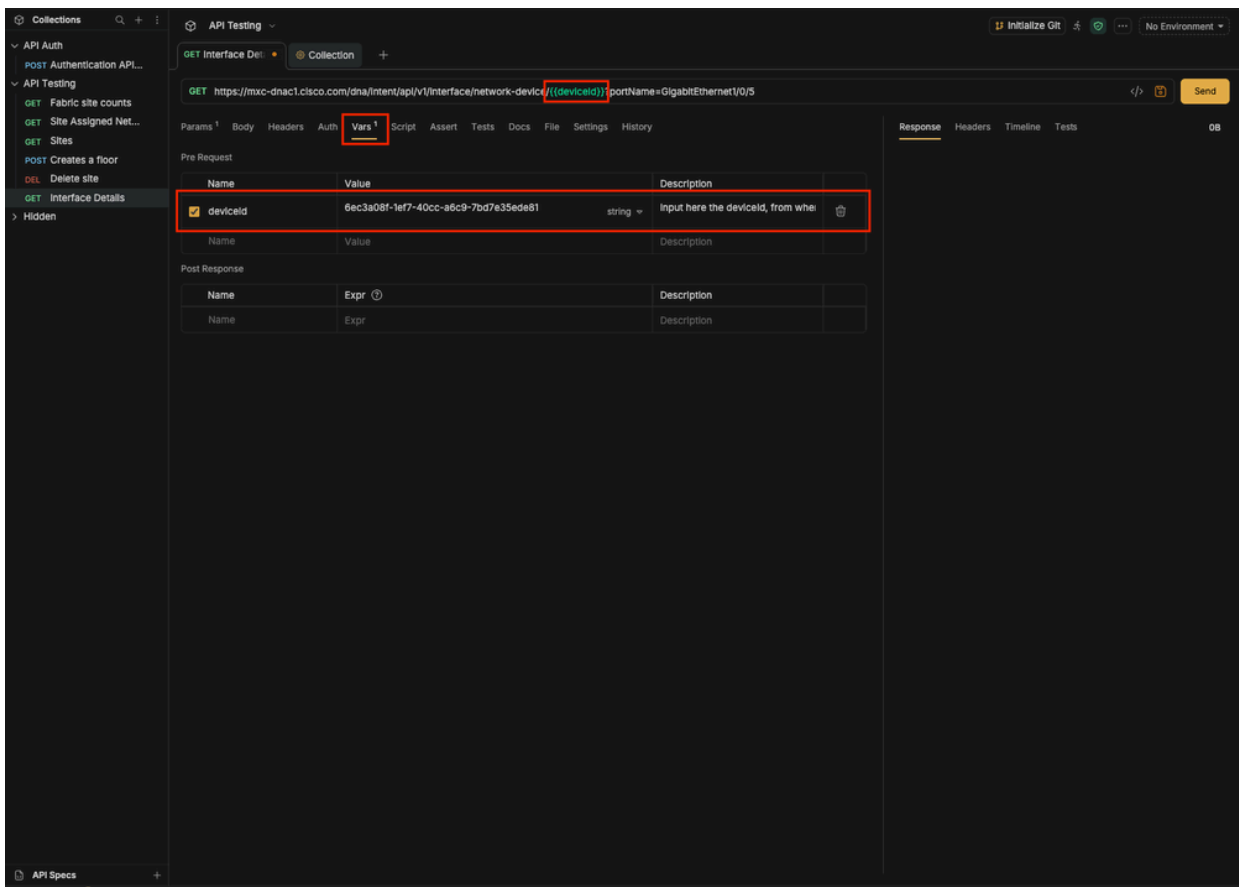
此示例说明如何获取网络设备上的特定接口的状态。

步骤 1：识别设备ID

在Catalyst Center GUI中，转到Inventory，单击您的设备，然后在浏览器中查看URL。最后一个数据段是设备ID。

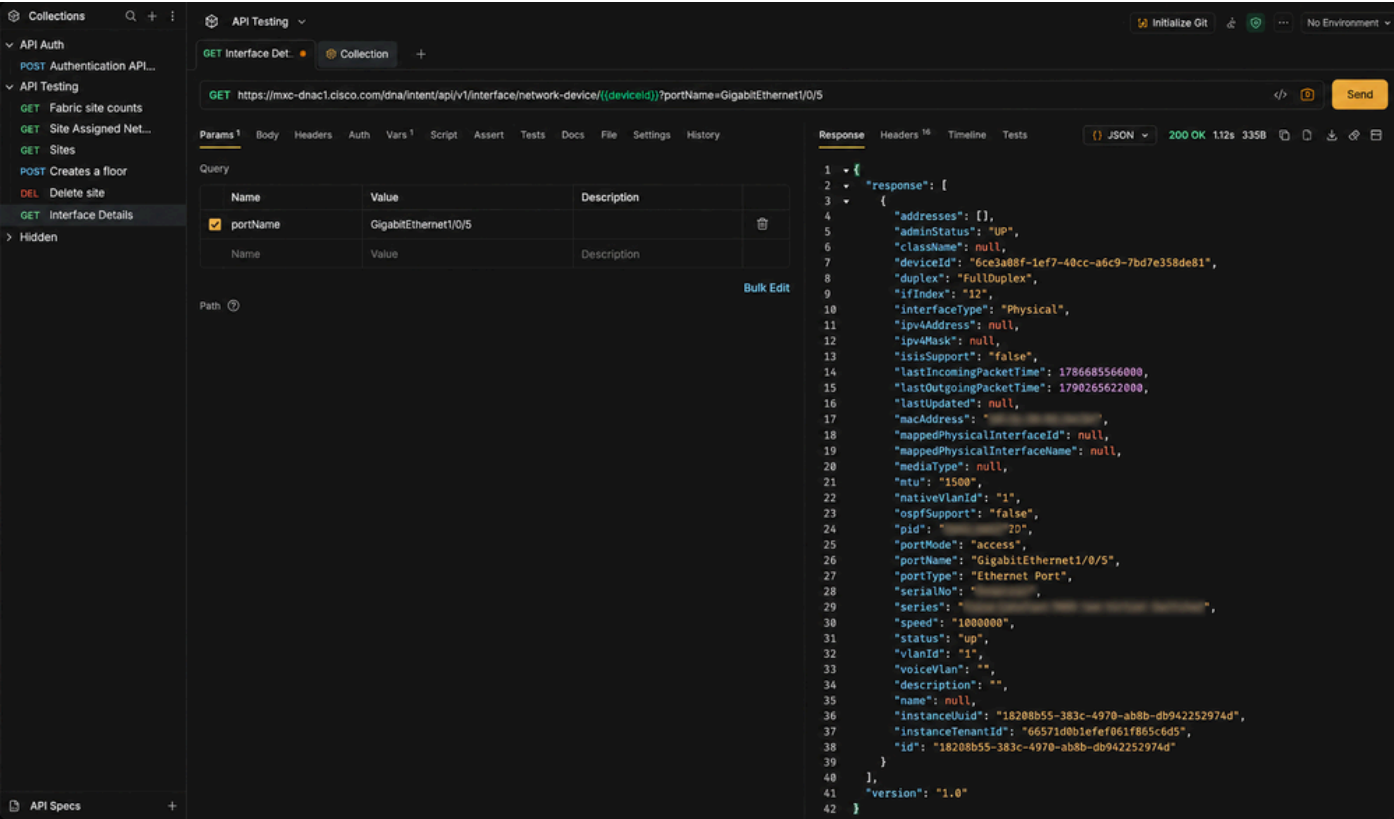
步骤 2：在Bruno中创建参数化请求

- 1. 为/dna/intent/api/v1/interface/network-device/{{deviceId}}创建GET请求。
- 2. 在请求的Vars选项卡中，添加名为deviceId变量并粘贴您的ID。
- 3. 要过滤特定端口，请添加查询参数:
 - 密钥:名称
 - 值：GigabitEthernet1/0/5



在Bruno中创建变量

4. 单击发送。



GET请求应用查询

结论

Catalyst Center北向API为网络管理员提供了强大的灵活性。通过从基于GUI的管理过渡到API驱动的工作流程，您可以自动执行重复性任务、确保配置一致性，并将网络智能集成到更广泛的IT生态系统中。

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。