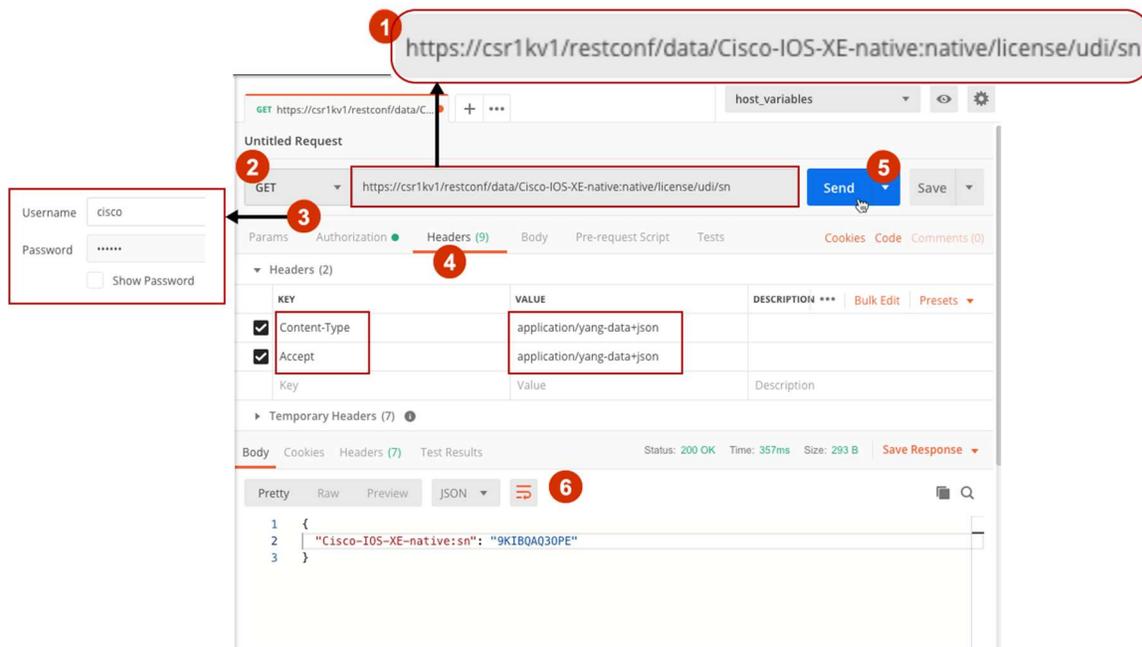


# Hướng dẫn sử dụng Postman để gửi yêu cầu API REST

## Vì sao nên học dùng Postman trong DevOps/Automation?

Khi làm việc với các hệ thống tự động hóa hạ tầng (ví dụ như Cisco Catalyst Center, RESTCONF API, hoặc quản lý cloud qua Terraform/Ansible), bạn sẽ phải tương tác với rất nhiều API. Nếu không có công cụ kiểm thử API trực quan như Postman, việc debug và kiểm tra API sẽ cực kỳ tốn thời gian.

## Cách gửi yêu cầu API với Postman – Từ lý thuyết đến thực hành



Postman là một ứng dụng độc lập, bạn có thể tải về và cài đặt trên Windows, macOS hoặc Linux. GUI của Postman cho phép bạn:

- Nhập URL endpoint
- Chọn HTTP method như GET, POST, PUT, DELETE
- Thêm headers như Content-Type: application/json
- Chọn loại xác thực (Basic Auth, Bearer Token, OAuth2, v.v.)
- Nhấn nút Send và xem ngay kết quả trả về

Ví dụ đơn giản:

- URL: `http://192.168.100.10/api/v1/devices`
- Phương thức: GET
- Header: Accept: `application/json`
- Xác thực: Basic Auth (username/password)
- Nhấn Send

→ Bạn sẽ nhận được mã trạng thái HTTP 200 (OK) cùng với nội dung phản hồi (response body) dạng JSON.

### Các công cụ khác ngoài Postman

- cURL: Công cụ dòng lệnh phổ biến, tích hợp dễ dàng vào shell script hoặc CI/CD pipeline.
- Python (requests): Dành cho người viết mã automation, giúp xây dựng quy trình tự động hóa API.

### Ví dụ với Python:

```
import requests

response = requests.get(
    'http://192.168.100.10/api/v1/devices',
    auth=('admin', 'vnpro123'),
    headers={'Accept': 'application/json'})

print(response.status_code)

print(response.json())
```

### Những điểm cần lưu ý khi dùng Postman

- Postman hỗ trợ nhiều loại xác thực, không chỉ Basic Auth.
- Không bị giới hạn bởi định dạng XML – hỗ trợ cả JSON, YAML...
- Giao diện giúp thử nghiệm dễ dàng các chức năng như phân trang, lọc dữ liệu, sắp xếp.

## Câu hỏi ôn tập

Hai phát biểu đúng về Postman là:

1. Postman là một ứng dụng GUI rất dễ dùng để gọi API REST.
2. Postman có thể thực hiện các yêu cầu GET, PUT, POST và DELETE.

Sai:

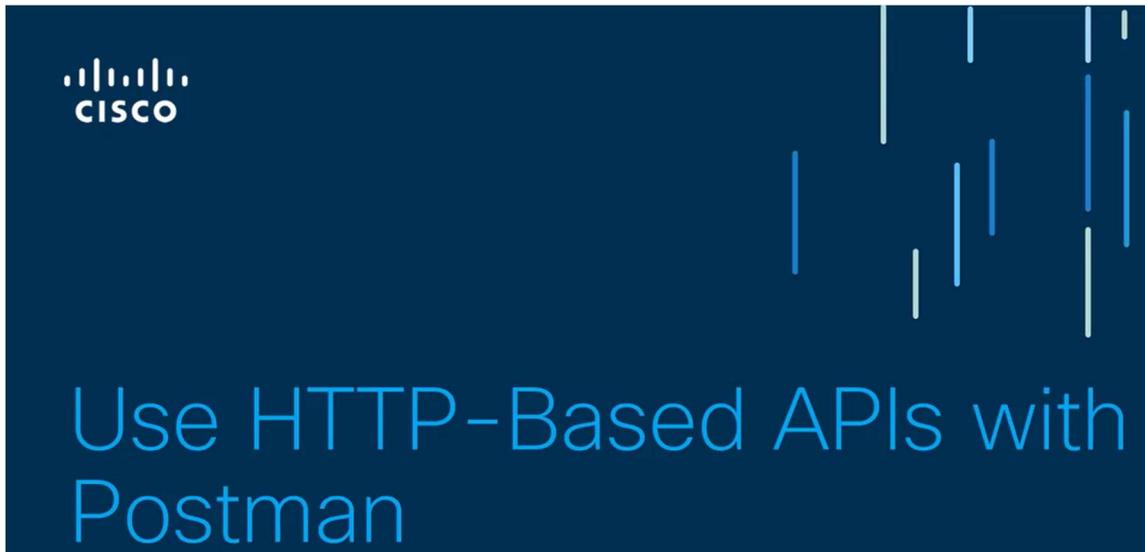
- Chỉ hỗ trợ XML (sai)
- Postman là công cụ duy nhất (sai)
- Chỉ có xác thực cơ bản (sai)

## Lời kết dành cho cộng đồng VnPro

Học Postman không chỉ là học một công cụ, mà là nền tảng để bước vào thế giới API Automation chuyên nghiệp. Bạn có thể triển khai mọi thứ từ kiểm thử bảo mật API đến xây dựng dashboard giám sát tự động. Bạn đang học CCNA DevNet hoặc DEVCOR, hãy xem Postman như “Cisco Packet Tracer” cho thế giới lập trình API.

# Tự động hóa cấu hình mạng bằng RESTCONF API và Postman

Một trong những bài lab nền tảng và rất quan trọng trong hành trình trở thành DevNet/NetDevOps chuyên nghiệp: Tự động hóa cấu hình mạng bằng RESTCONF API và Postman.



💡 Tóm tắt nhanh cho cộng đồng DevOps / Automation Engineer Việt Nam  
Nếu bạn đang làm với Cisco IOS XE, thì RESTCONF API chính là cây cầu nối giữa cấu hình truyền thống (CLI) và tự động hóa hiện đại (programmability). Với Postman, bạn có thể:

## ◇ Gửi yêu cầu GET để:

- Truy xuất toàn bộ cấu hình:

```
GET https://csr1kv1/restconf/data/Cisco-IOS-XE-native:native?content=config
```

- Xem phiên bản thiết bị:

```
GET https://csr1kv1/restconf/data/Cisco-IOS-XE-native:native/version
```

- Xem serial number:

```
GET https://csr1kv1/restconf/data/Cisco-IOS-XE-native:native/license/udi/sn
```

- Lấy thông tin interface cụ thể:

```
GET https://csr1kv1/restconf/data/Cisco-IOS-XE-native:native/interface/GigabitEthernet=2
```

## ◇ Gửi yêu cầu PATCH để:

- Cập nhật cấu hình SNMP contact:

```
{  
  "Cisco-IOS-XE-snmp:contact": "cisco_devnet"  
}
```

- Thêm IP phụ, mô tả cho interface Loopback:

```
{  
  "Cisco-IOS-XE-native:interface": {  
    "Loopback": [  
      {  
        "name": 200,  
        "description": "Configured_with_Postman",  
        "ip": {  
          "address": {  
            "secondary": [  
              {  
                "address": "10.202.202.1",  
                "mask": "255.255.255.0"  
              }  
            ],  
            "primary": {  
              "address": "10.200.2.2",  
              "mask": "255.255.255.0"  
            }  
          }  
        }  
      }  
    ]  
  }  
}
```

#### ◇ Gửi yêu cầu POST để tạo tài nguyên mới:

- Tạo interface Loopback100:

```
{  
  "Cisco-IOS-XE-native:Loopback": {  
    "name": 100,  
    "ip": {  
      "address": {  
        "primary": {  
          "address": "10.100.2.2",  
          "mask": "255.255.255.0"  
        }  
      }  
    }  
  }  
}
```

}

- ◆ Gửi yêu cầu PUT để thay thế toàn bộ cấu hình interface:
  - Loại bỏ các IP phụ khỏi Loopback200 bằng cách gửi lại toàn bộ nội dung cấu hình bạn muốn giữ.

### Một số lưu ý cực kỳ quan trọng cho anh em CCNA Automation:

- PATCH = merge nhẹ, không thay đổi cấu hình không khai báo
- PUT = thay thế toàn bộ cấu hình cho object
- POST = tạo mới; gửi lại lần 2 sẽ bị lỗi 409 Conflict
- GET = quan sát trước khi thao tác; luôn dùng để xác thực sau cấu hình
- Header: luôn cần Accept và Content-Type dạng application/yang-data+json

### Ứng dụng thực chiến:

- Tự động hóa cấu hình SNMP, Interface, ACL, Route...
- Tích hợp vào Jenkins, Ansible, CI/CD pipeline cho DevSecOps
- Kiểm thử nhanh API Cisco với GUI thân thiện của Postman trước khi viết Python/Ansible