

LAB - VIẾT CHƯƠNG TRÌNH DÙNG NGÔN NGỮ PYTHON SỬ DỤNG RESTCONF ĐỂ THỰC HIỆN LẤY DANH SÁCH THÔNG TIN CÁC CỔNG TRÊN THIẾT BỊ SANDBOX CISCO

I. Sơ đồ



Hình 1: Sơ đồ bài Lab

II. Mô tả

- Sơ đồ bài thực hành gồm 1 PC và 1 thiết bị chạy hệ điều hành IOS XE được đấu nối với nhau như hình 1.
- Trên sơ đồ này, học viên thực hiện kết nối đến thiết bị mạng, viết chương trình để lấy thông tin bằng ngôn ngữ Python.
- Máy PC phải đáp ứng yêu cầu đã cài đặt trạm làm việc cho developer.

III. Yêu cầu kỹ thuật

- Học viên thực hiện kết nối máy tính đến Agent bằng mạng Internet.
- Cài đặt thư viện requests trên máy tính.
- Viết code bằng Python thực hiện yêu cầu:
 - +Viết hàm lấy thông tin các cổng.
 - +Viết hàm main() in ra thông tin các cổng.

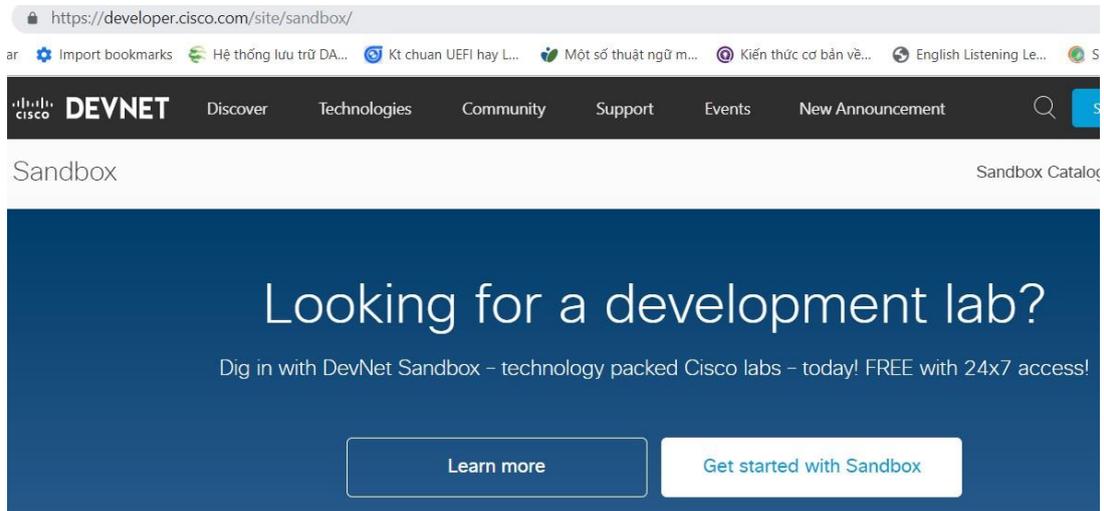
IV. Các bước thực hiện

4.1. Cài đặt thư viện

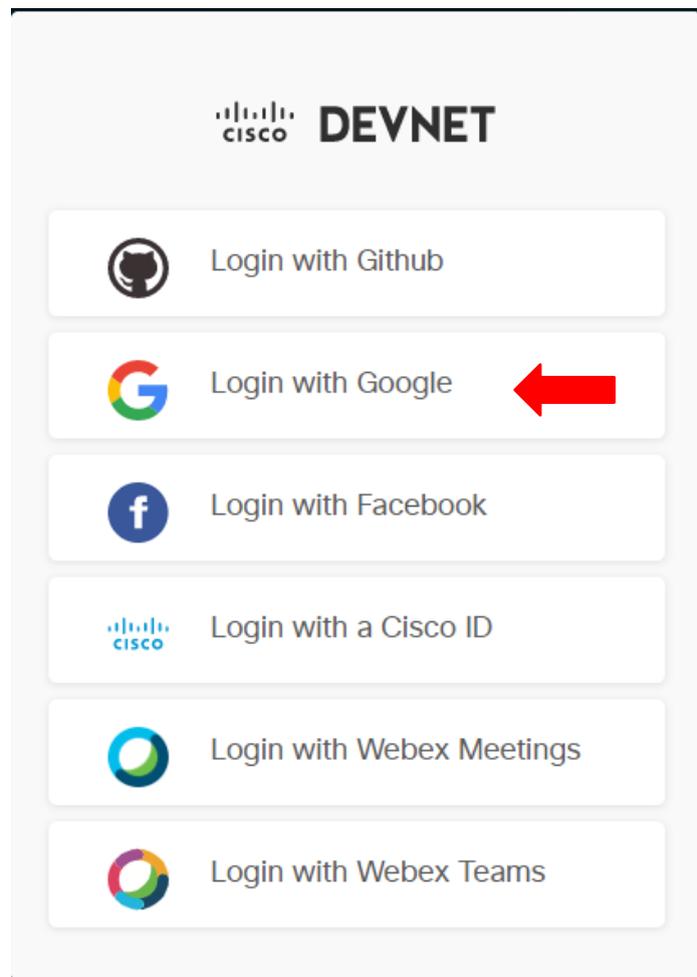
- Mở cmd bằng tổ hợp phím Win+R: gõ cmd.
- Gõ python -m pip install requests - -user

Kiểm tra đi được Internet: ping 8.8.8.8

Vào sandbox của cisco để lấy thông tin CSR1000V. Tìm kiếm với các từ khóa devnetsandbox. Chúng ta sẽ vào được website developer của cisco:



Đăng nhập vào website devnet dùng cisco ID hoặc dùng tài khoản Facebook, Google:



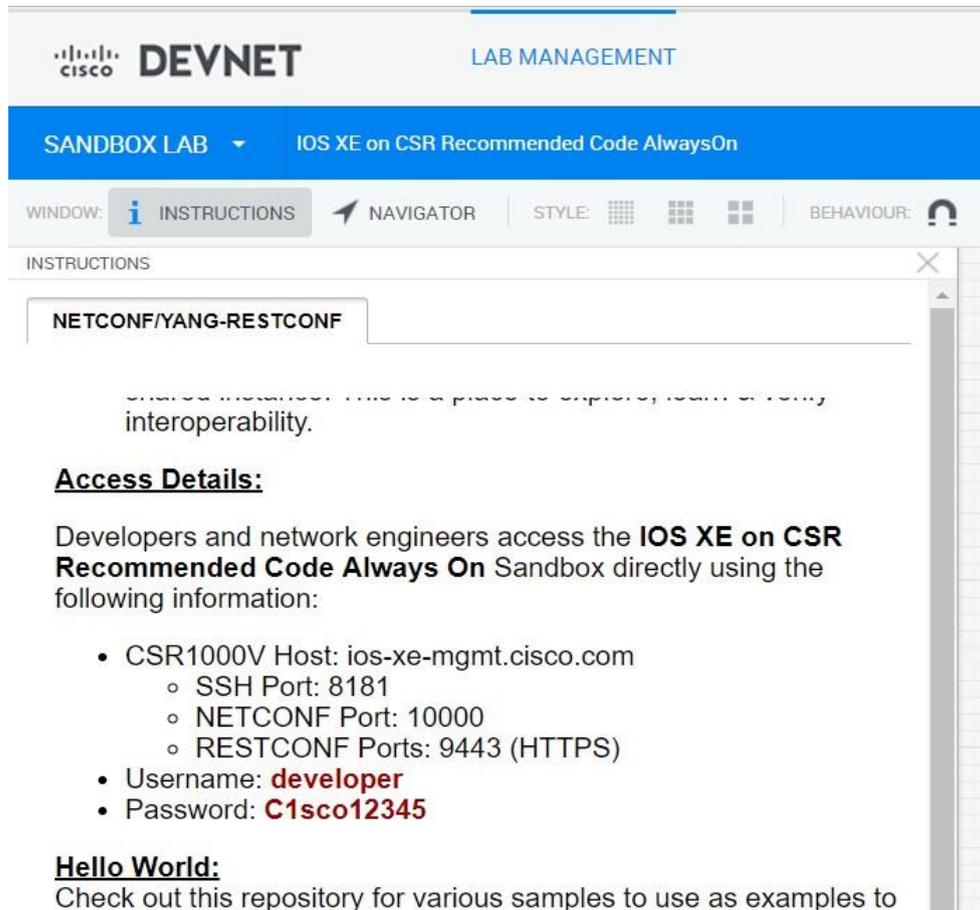
Sau khi đăng nhập xong sẽ hiện lên giao diện web tất cả sandbox lab:

The screenshot shows the DEVNET LAB MANAGEMENT interface. The top navigation bar includes the Cisco logo and 'DEVNET' branding. Below the navigation bar, there is a search bar for 'Sandbox Labs' and a 'LAB CATALOG (77)' header with a 'Reset filters' link. The left sidebar contains a 'Sandbox Labs' menu item (highlighted with a red arrow), 'Reservations' (0 New, 0 Total), and a 'FILTER BY' section for 'Sandbox Lab Status' with options for 'Available', 'Unavailable', and 'View only'. The main content area displays 'All Sandbox Labs (77)' and lists several lab cards. Each card includes a version number, a title, a description, and a button to 'RESERVE' or 'ALWAYS-ON'. A red arrow points to the 'Sandbox Labs' menu item in the sidebar.

Vào Search Sandbox Labs gõ IOS XE để tìm kiếm các lab sử dụng IOS XE, chọn lab có tên Recommended Code AlwaysOn:

The screenshot shows the DEVNET LAB MANAGEMENT interface with search results for 'IOS XE'. The search bar at the top left contains 'IOS XE'. The 'LAB CATALOG (6)' header is visible. The left sidebar is the same as in the previous screenshot. The main content area displays 'All Sandbox Labs (6)' and lists five lab cards. Each card includes a version number, a title, a description, and a button to 'RESERVE' or 'ALWAYS-ON'. A red arrow points to the 'TYPE' dropdown menu in the top right corner of the main content area.

Ở phía bên trái kéo xuống phần Access Details ta có được thông tin để kết nối đến CSR1000V:



DEVNET LAB MANAGEMENT

SANDBOX LAB ▾ IOS XE on CSR Recommended Code AlwaysOn

WINDOW: **i** INSTRUCTIONS NAVIGATOR STYLE: BEHAVIOUR: ↶

INSTRUCTIONS

NETCONF/YANG-RESTCONF

shared instance. This is a place to explore, learn and verify interoperability.

Access Details:

Developers and network engineers access the **IOS XE on CSR Recommended Code Always On** Sandbox directly using the following information:

- CSR1000V Host: ios-xe-mgmt.cisco.com
 - SSH Port: 8181
 - NETCONF Port: 10000
 - RESTCONF Ports: 9443 (HTTPS)
- Username: **developer**
- Password: **C1sco12345**

Hello World:

Check out this repository for various samples to use as examples to

4.2. Viết chương trình

Import các thư viện và khai báo các thông tin cần để giao tiếp với thiết bị:

```
import requests
import sys

requests.packages.urllib3.disable_warnings()

HOST = 'ios-xe-mgmt.cisco.com'
PORT = '9443'
USER = 'developer'
PASS = 'C1sco12345'
```

Viết hàm lấy thông tin các cổng:

```
def get_configured_interfaces():
    url = "https://{h}:{p}/restconf/data/ietf-interfaces:
        interfaces".format(h=HOST, p=PORT)

    headers = {
        'Content-Type': 'application/yang-data+json',
        'Accept': 'application/yang-data+json'
    }

    response = requests.get(url, auth=(USER, PASS),
        headers=headers, verify=False)

    return response.text
```

Gán đường dẫn url:

+ Khai báo headers bao gồm Content-Type, Accept.

+ Khai báo biến response để gán kết quả trả về phương thức get của thư viện requests.

+ “return response.text” nghĩa là sau khi chạy xong hàm get_configured_interfaces() sẽ trả về response dạng text.

- Viết hàm main để chạy hàm get_configured_interfaces() và in ra thông tin các cổng:

```
def main():
    interfaces = get_configured_interfaces()
    print(interfaces)

if __name__ == '__main__':
    sys.exit(main())
```

Kết quả đạt được khi chạy chương trình:

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "GigabitEthernet1",
        "description": "MANAGEMENT INTERFACE - DON'T TOUCH ME",
        "type": "iana-if-type:ethernetCsmacd",
        "enabled": true,
        "ietf-ip:ipv4": {
          "address": [
            {
              "ip": "10.10.20.48",
              "netmask": "255.255.255.0"
            }
          ]
        },
        "ietf-ip:ipv6": {
        }
      },
      {
        "name": "GigabitEthernet2",
        "description": "Configured by NETCONF",
        "type": "iana-if-type:ethernetCsmacd",
        "enabled": true,
        "ietf-ip:ipv4": {
          "address": [
            {
              "ip": "10.255.255.1",
              "netmask": "255.255.255.0"
            }
          ]
        },
        "ietf-ip:ipv6": {
        }
      }
    ]
  }
}
```