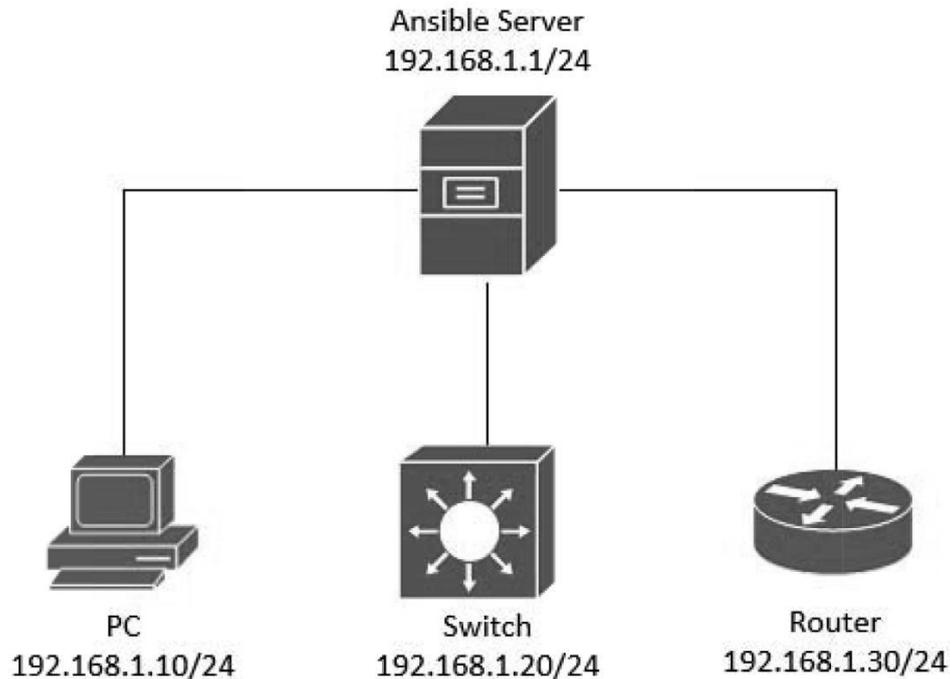


LAB - QUẢN LÝ VÀ TRIỂN KHAI TỰ ĐỘNG HẠ TẦNG MẠNG DÙNG ANSIBLE

I. Sơ đồ



Hình 1: Sơ đồ bài Lab

II. Mô tả

- Sơ đồ gồm 1 Router, 1 Switch và 1 PC được kết nối vào Ansible Server. Yêu cầu các thiết bị ping thông tới Ansible Server.

- Máy tính của học viên có thể truy cập vào Ansible Server thông qua trình duyệt web.

III. Yêu cầu

Học viên truy cập vào Ansible Server để:

- + Thêm thiết bị vào Inventory.
- + Thêm credential để xác thực thiết bị.
- + Thêm Playbook vào project/jobs để đẩy cấu hình xuống thiết bị.
- + Chạy thử playbook thêm VLAN cho Switch và thiết lập cổng cho Router.

IV. Các bước thực hiện

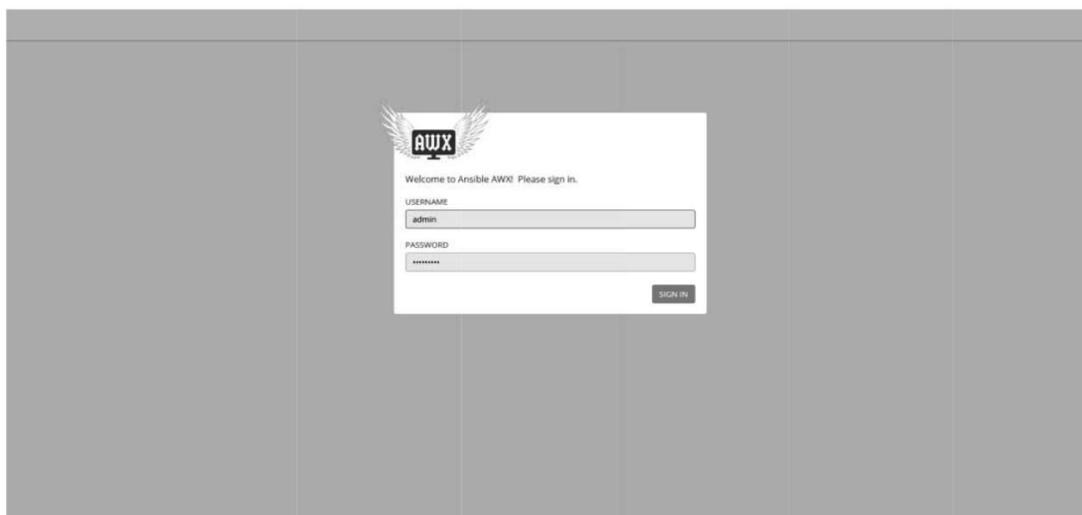
Cấu hình SSH Router:

```
R# configure terminal
R(config)# username admin password 123
R(config)# enable password 321
R(config)# crypto key generate rsa
1024
R(config)#line vty 0 4
R(config-line)# password 123
R(config-line)# login local
```

Cấu hình SSH Switch:

```
SW# configure terminal
SW(config)# username admin password 123
SW(config)# enable password 321
SW(config)# crypto key generate rsa
1024
SW(config)#line vty 0 4
SW(config-line)# password 123
SW(config-line)# login local
```

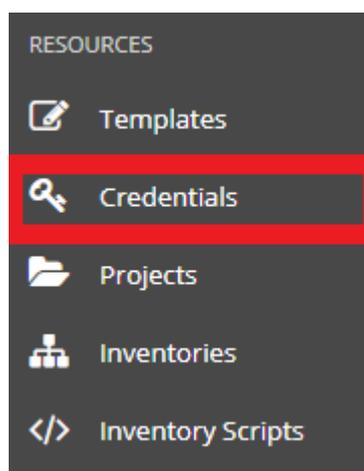
Truy cập vào Ansible Server bằng trình duyệt web:



Giao diện chính của Ansible Server:



Từ giao diện chính, vào Credentials để thiết lập mã SSH:

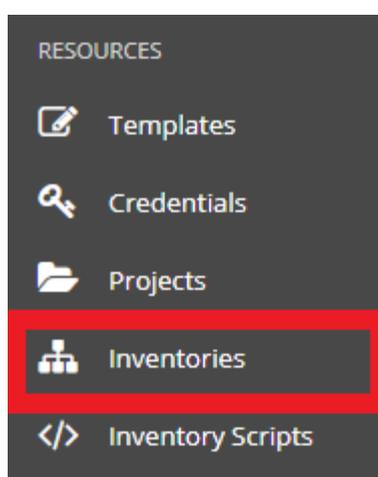


Sau khi vào Credentials, bấm vào icon  để thêm Username/Password SSH:
Trong đó:

- Name: Đặt tên để quản lý mật khẩu SSH.
- Credential Type: Vì phương thức xác thực là SSH nên chúng ta chọn “Machine”.
- Username/Password: Tên đăng nhập và mật khẩu SSH của thiết bị.

Sau khi cài đặt xong nhấn Save để lưu lại.

Để thêm thiết bị vào Inventories, từ giao diện chính vào Inventories để tiến hành thêm thiết bị:



Giao diện Inventories:

Bấm vào icon  để thêm thiết bị vào:

NAME	TYPE	ORGANIZATION	ACTIONS
Demo Inventory	Inventory	Default	  
Devices	Inventory	Default	  

Trong đó:

[1]: Nhập tên Inventory để quản lý.

[2]: Chọn nhóm cho Inventory, ở đây mặc định để default.

[3]: Khai báo các thông tin cơ bản của các thiết bị.

Khai báo [3]:

Giải thích:

- Ansible_connection: network_cli là phương thức kết nối đến thiết bị thông qua SSH.
- Ansible_become: 'yes' và ansible_become_method: enable là cho phép ansible có thể vào privilege mode trước khi thực thi các task.
- Ansible_network_os: ios ở đây do dùng các thiết bị của Cisco nên khai báo như vậy.

Lưu ý: Thứ tự khai báo có thể thay đổi mà không ảnh hưởng đến kết quả.

```
---
ansible_connection: network_cli
ansible_become: 'yes'
ansible_become_method: enable
ansible_network_os: ios
```

Sau khi khai báo xong vào mục [4] để khai báo IP và Enable Password của thiết bị:

Khai báo [4]

Trong đó:

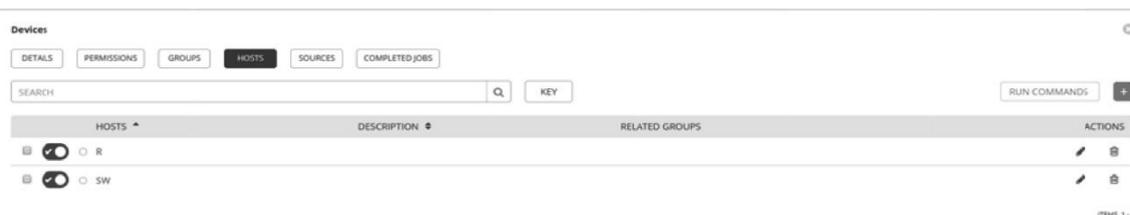
- Host name: Tên host để quản lý.

- Variable: Lưu ý: bắt đầu với dấu '---'
- + Ansible_host: Địa chỉ IP của thiết bị dùng để SSH.
- + Ansible_become_password: Là enable password của thiết bị.

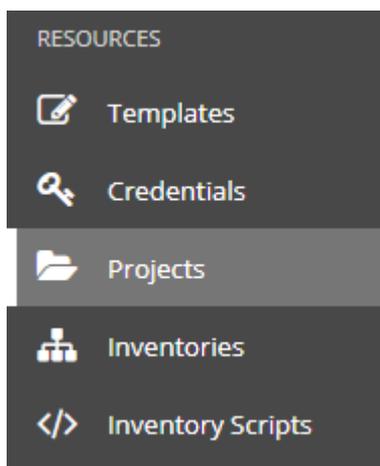
```
---  
ansible_host: '192.168.1.30'  
ansible_become_password: '123'
```

Nhấn Save để lưu lại cấu hình.

Với thiết bị Switch cũng cấu hình tương tự. Sau khi cấu hình xong trong Inventory/Devices sẽ có 2 mục thiết bị như sau:



Tiếp theo, để tạo Playbook trong project ta vào mục project



Bấm vào icon  để thêm Project mới vào:

Ở đây cần điền vào các thông số sau:

- Name: Tên để quản lý các Project
- SCM type: Vì Playbook để trên github nên chúng ta sẽ chọn Git.
- SCM URL: Địa chỉ URL của github chứa các Playbook.
- Tích vào Update Revision on launch để update các project trước khi thực thi các job.

Sau khi hoàn tất nhấn Save để lưu cấu hình.

Vào Job templates để tạo các tác vụ thực hiện Playbook:

Ở đây cần điền các mục sau:

- Name: Tên Job để quản lý.
- Job type: Để thực thi file Playbook chúng ta chọn Run.
- Inventory: Chọn thiết bị để thực thi Playbook.
- Project: Chọn Project chứa các Playbook để thực thi.
- Playbook: Các Playbook được chứa trong Github có đuôi yml.

Project 1



Bấm vào icon  → Job templates để thêm Job mới 



Nội dung file vlan.yml:

```
---
- name: tao vlan
  hosts: all # tên của thiết bị trong Ansible server( vì ở đây chỉ có router nên có thể để all)
  gather_facts: no # lấy thông tin của host (khi thực hiện playbook sẽ tự động lấy thông tin của host,
  # ở đây chọn no hoặc false để khi chạy playbook đỡ tốn thời gian)

  tasks:
    - name: Create vlan #Đặt tên task để quản lý
      ios_vlan: # Cú pháp để thiết lập vlan, vlan_id, name,state,... là các tham số để cấu hình vlan
        vlan_id: "{{ item.number }}" #item.number thì nó sẽ tìm giá trị với key là number trong phần with_items
        name: "{{ item.name }}" #item.name thì nó sẽ tìm giá trị với key là name trong phần with_items
        state: present # trạng thái của vlan: present: tạo, absent:xóa
      with_items:
        - { number: "33", name: "phong_nhan_su" }
        - { number: "34", name: "phong_ke_toan" }
      #Hiển thị các vlan lên màn hình
    - name: Show vlan
      ios_command: # Cú pháp để viết lệnh để gửi ở mode command line của switch
        commands:
          - show vlan brief
      register: show_vlan #gắn kết quả thực thi từ switch vào biến show_vlan

    - debug: var=show_vlan.stdout_lines #debug: in ra màn hình, đặt biến var
      # (lấy thông tin của các dòng từ show_vlan gắn vào),in biến var
```

Tương tự với Router.yml:

```
---
- name: dat ip cho loopback
  hosts: all # tên của thiết bị trong Ansible server( vì ở đây chỉ có router nên có thể để all)
  # các tác vụ
  tasks:
    - name: Set loopback IPv4 address # Đặt tên task để quản lý
      ios_l3_interface: # do cấu hình router nên bắt buộc khai báo như vậy
        name: loopback 2 # tên cổng
        ipv4: 10.0.0.1/24 # địa chỉ IP của loopback

    - name: Set GigabitEthernet0/3 IPv4 address
      ios_l3_interface:
        name: GigabitEthernet0/3
        ipv4: 192.168.0.1/24
      #Xem lại tất cả các cổng
    - name: Show ip
      ios_command: # Viết lệnh để gửi ở mode command line của router
        commands:
          - show ip int brief
      register: show_ip # gắn kết quả vào biến show_ip

    - debug: var=show_ip.stdout_lines # debug: in ra màn hình, đặt biến var
      # (lấy thông tin của các dòng từ show_ip gắn vào),in biến var
```

Ví dụ một số lệnh sử dụng trên thiết bị Cisco IOS để viết các nhiệm vụ trong playbook bằng ngôn ngữ YAML, chạy trên Ansible server:

Ios_command: Thực hiện khi thiết bị ở mode Privileged.

```
tasks:
- name: Chạy lên show version
  ios_command:
  commands: Show version

- name: Chạy lệnh show version bao gồm IOS
  ios_command:
  commands: Show version
  wait_for: Result[0] contains IOS

- name: Chạy nhiều lệnh
  ios_command:
  commands:
  - show version
  - show interfaces

- name: Chạy lệnh yêu cầu trả lời prompt
  ios_command:
  commands:
  - command: 'clear counters GigabitEthernet0/1'
    prompt: 'Clear "show interface" counters on this interface \[confirm\]'
    answer: 'y'
  - command: 'clear counters GigabitEthernet0/2'
    prompt: '[confirm]'
    answer: '\r'
```

Ios_config: Thực hiện khi thiết bị ở mode Configuration:

```
- name: Cấu hình hostname trong inventory
  ios_config:
  lines: hostname {{ inventory_hostname }}
- name: Cấu hình cổng
```

```
ios_config:
lines:
-description test interface
-ip address 172.31.1.1 255.255.255.0
parents:interface Ethernet1

-name: Cấu hình ip helper-address trên nhiều cổng
ios_config:
lines:
-ip helper-address 172.26.1.10
-ip helper-address 172.26.3.8
parents: "{{item}}"
with_items:
-interface Ethernet1
-interface Ethernet2
-interface GigabitEthernet1
```

ios_interface: Thực hiện trên cổng:

```
-name: Cấu hình cổng
ios_interface:
name: GigabitEthernet0/2
description:test-interface
speed: 100
duplex: half
mtu: 512
-name: Xóa cổng loopback
ios_interface:
name: Loopback9
state: absent
-name: Mở cổng (up)
ios_interface:
name: GigabitEthernet0/2
```

```
enabled: True
-name: Tắt cổng (down)
ios_interface:
name: GigabitEthernet0/2
enabled: False
```

Ios_vlan: thực hiện trên VLAN

```
-name: Tạo vlan
ios_vlan:
vlan_id: 100
name: test-vlan
state: present

-name: Thêm cổng vào vlan
ios_vlan:
vlan_id: 100
interfaces:
-GigabitEthernet0/0
-GigabitEthernet0/1

-name: Xóa vlan
ios_vlan:
vlan_id: 100
state: absent

-name: Thêm vlan dùng aggregate
ios_vlan:
aggregate:
- { vlan_id: 100, name: test-vlan, interfaces: [GigabitEthernet0/1,
GigabitEthernet0/2], delay: 15, state: suspend}
- { vlan_id: 101, name: test-vlan, interfaces: GigabitEthernet0/3}
```

Credentials: Chọn tài khoản và mật khẩu xác thực SSH cho thiết bị.

Sau khi hoàn thành bấm Save rồi Launch để khởi chạy Job.

Khi chạy file vlan.yml thành công:

```
SSH password:
PLAY [tao vlan] ***** 02:06:33

TASK [Create vlan] ***** 02:06:33
changed: [SW] => (item={u'number': u'33', u'name': u'phong_nhan_su'})
changed: [SW] => (item={u'number': u'34', u'name': u'phong_ke_toan'})
fatal: [R]: FAILED! => {"msg": "timed out"}

TASK [Show vlan] ***** 02:07:04
ok: [SW]

TASK [debug] ***** 02:07:06
ok: [SW] => {
  "show_vlan.stdout_lines": [
    [
      "VLAN Name                Status    Ports",
      "-----",
      "1    default                active   Fa0/1, Fa0/2, Fa0/3, Fa0/4, Fa0/5, Fa0/6, Fa0/7, F
a0/8, Fa0/9, Fa0/10, Fa0/11, Fa0/12, Fa0/13, Fa0/14, Fa0/15, Fa0/16, Fa0/17, Fa0/18, Fa0/19, Fa0/20, Fa0/21, Fa
0/22, Fa0/23, Fa0/24, Gi0/1, Gi0/2",
      "33   phong_nhan_su           active   ",
      "34   phong_ke_toan            active   ",
      "1002 fddi-default            act/unsup ",
      "1003 token-ring-default    act/unsup ",
      "1004 fddinet-default       act/unsup ",
      "1005 trnet-default         act/unsup"
    ]
  ]
}
```

Vì Router không khởi tạo VLAN được nên báo lỗi.

Kết quả khi chạy file Router.yml:

```
ok: [R] => {
  "show_ip.stdout_lines": [
    [
      "Interface                IP-Address    OK? Method Status          Protocol",
      "FastEthernet0/0          10.215.11.68  YES DHCP    up              up",
      "FastEthernet0/1          unassigned    YES unset    administratively down down",
      "Serial0/0/0              unassigned    YES unset    administratively down down",
      "Loopback2                 10.0.0.1      YES manual  up              up"
    ]
  ]
}

PLAY RECAP ***** 02:25:13
R                : ok=4    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ig
nored=0
SW                : ok=4    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ig
nored=0
```

Khi có chỉnh sửa trên Github và muốn đồng bộ ngay với Ansible Server ta vào Project nhấn vào icon như sau:

