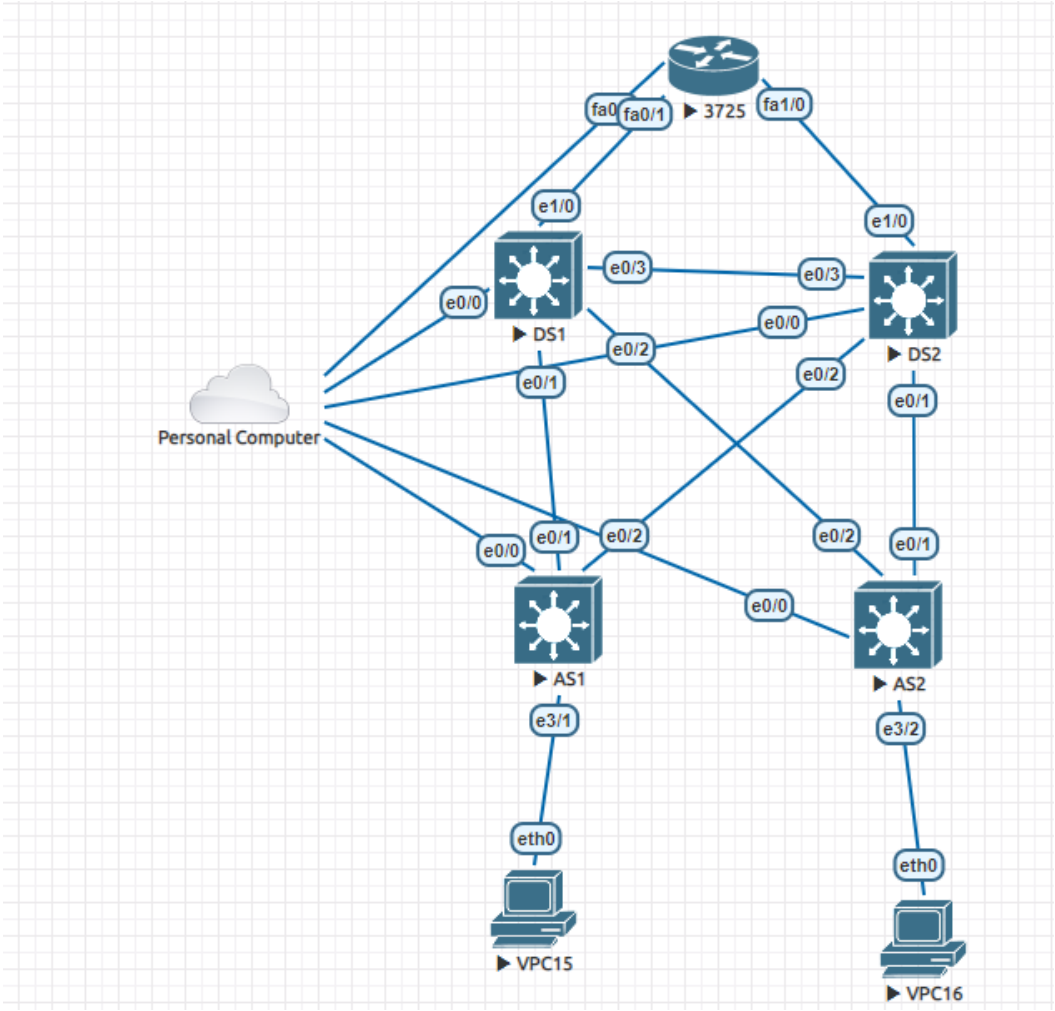
Lab - Bảo mật dữ liệu trong Ansible playbook với Ansible Vault

I. Ý tưởng:

Dựa trên bài lab HSRP trước đây. Ta sẽ để ssh username, ssh password và enable password vào 1 file riêng mà không nằm chung file chứa IP các thiết bị. Sau đó sẽ mã hóa file này với Ansible Vault. Mục đích chính là **bảo mật thông tin** tránh bị tấn công.

Sơ đồ mạng:



Mô tả:

- Sơ đồ gồm 1 Router, 2 Distributed Switch, 2 Access Switch, 2 PC cho 2 Vlan, được kết nối với Ansible Server. Mô hình thực hiện trên Linux OS và các thiết bị ảo.

II. Yêu cầu:

Bài lab được thực hiện trên môi trường Linux. Máy thực hiện đã được cài Ansible. Máy kết nối được đến các thiết bị bằng các công cụ như Putty hay CLI.

- Thực hiện các công việc sau:
  - Các cú pháp cơ bản với Ansible Vault.
  - Tạo file playbook pass.yml.
  - Chạy playbook đã được mã hóa.

Các bước thực hiện:

Cấu hình Router và 4 Switch, các bạn tham khảo bài trước:

https://www.forum.vnpro.org/forum/ccna%C2%AE/devnet-associate/421460-lab-qu%E1%BA%A3n-l%C3%BD-v%C3%A0-tri%E1%BB%83n-khai-t%E1%BB%B1-%C4%91%E1%BB%99ng-h%E1%BA%A1-t%E1%BA%A7ng-m%E1%BA%A1ng-d%C3%B9ng-ansible

Code các file thực hiện, các bạn có thể tham khảo ở đây:
https://github.com/vnpro149/Ansible/tree/master/HSRP

Lưu ý trong khi viết file playbook thực thi tác vụ:

- Thêm khai báo file vào code đoạn sau trước mỗi phần tasks trong file playbook để bảo mật cho các tác vụ khi thực thi các cấu hình lên thiết bị:

```
name: AS1
hosts: AS1
gather_facts: no
vars_files:
  - pass.yml
tasks:
```

- Đây là khai báo file pass.yml để lấy ssh user, ssh password,… nếu đặt file không nằm chung với file playbook, ta cần khai báo đường dẫn chính xác đến vị trí file vd: **roles/group_vars/pass.yml**
- Các giải thích chi tiết về cú pháp file YAML đã có ở bài trước, nếu quên, các bạn có thể xem lại
- Tạo file playbook là pass.yml chứa các thông tin cần bảo mật: ssh username, ssh password, enable password, sudo password, ….

Một vài cú pháp cơ bản khác khi dùng Ansible Vault:

Bước 1: Tạo file Ansible Vault

- Để tạo file Ansible Vault gồm 2 cách:
  - **Cách 1:** Tạo playbook như bình thường bằng bất cứ công cụ editor nào bạn muốn: vi, vim, nano,…. sau đó mã hóa file đó với Ansible Vault bằng câu lệnh: sudo ansible-vault encrypt <filename>.yml
  - **Cách 2:** Tạo 1 file Ansible Vault với câu lệnh: sudo ansible-vault create <filename>.yml. Tuy nhiên, cách này mặc định ansible sẽ chọn **vi** làm editor. Các bạn có thể chỉnh thành editor mình muốn trong file ~/.bashrc: sudo nano ~/.bashrc và thêm dòng export EDITOR=nano hoặc bất cứ công cụ editer nào mà bạn muốn.
- Để sửa file:
  - sudo ansible-vault edit <filename>.yml hoặc đơn giản là giải mã file và sửa nó, sau đó mã hóa lại.
  - **Mã hóa file:** sudo ansible-vault encrypt <filename>.yml
  - **Giải mã file:** sudo ansible-vault decrypt <filename>.yml
  - **Thay đổi mật khẩu mã hóa:** sudo ansible-vault rekey <filename>.yml
  - Xem nội dung file đã mã hóa: sudo ansible-vault view <filename>.yml
- **Lưu ý:** các câu lệnh rekey, encrypt, decrypt có thể sử dụng cho nhiều file cùng 1 lúc, vd: **ansible-vault rekey file1.yml file2.yml**

Nội dung file **pass.yml** của mình:

```
---
ansible_ssh_user: admin
ansible_ssh_pass: 123
ansible_become_password: 321
```

Nội dung file pass.yml sau khi mã hóa:

```
khoa@ubuntu /etc/ansible sudo cat pass.yml
$ANSIBLE_VAULT;1.1;AES256
64353832653530336535373232646636656161626161633038346337656164393133663539326235
35306161366646433733623735346438343937386163616 50a6265623339303135653438383734 62
64666433343166306264393638653161653263373634353263643962643037313937613663663164
61356433386435353 30a3232366631323337616630386337663165383030386537396665303237 64
30623232623563335656131343263373830663664633430373938323830393538336238326538 3839
62366339346534613435373266643130346462326635333065313038333834353938383363316361
65636432616361363135383063363534613037356166643461346162333264306362663730326363
31616630663662313036333630623761383938313064393962663063633766306613338323931 65
3563
```

Lưu ý: cần ghi nhớ mật khẩu mã hóa. Có thể đặt nhiều password cho 1 file.

- Chạy các playbook với pass.yml đã được mã hóa.

  sudo ansible-playbook <filename>.yml --ask-vault-pass sau đó nhập mật khẩu mã hóa

- Có thể lưu mật khẩu mã hóa riêng ở 1 file hoặc 1 script và chạy với lệnh sau:

  sudo ansible-playbook <filename>.yml --vault-password-file password.txt

  hoặc

  sudo ansible-playbook <filename>.yml --vault-password-file password.py

  Giải thích:

- **--vault-password-file** là lấy password file chứa password.

- **--ask-vault-pass** là cho nhập password khi chạy

Kết quả:

- Chạy file AS.yml:

```
ok: [AS2]

TASK [Chuyen cong e0/1 - e0/2 sang mode trunk] ******************************
changed: [AS2]

TASK [Show Trunking] ********************************************************
ok: [AS2]

TASK [debug] ****************************************************************
ok: [AS2] => {
    "show_vlan.stdout_lines": [
        [
            "VLAN Name                             Status    Ports",
            "---- -------------------------------- --------- -------------------------------",
            "1    default                          active    Et0/1, Et0/2, Et0/3, Et1/0, Et1/1, Et1/2, Et1/3, Et2/0, Et2/1, Et2/2, Et2/
3, Et3/0, Et3/3",
            "10   KeToan                           active    Et3/1",
            "20   Kythuat                          active    Et3/2",
            "1002 fddi-default                     act/unsup ",
            "1003 token-ring-default               act/unsup ",
            "1004 fddinet-default                  act/unsup ",
            "1005 trnet-default                    act/unsup"
        ]
    ]
}

PLAY RECAP ******************************************************************
AS1                        : ok=6    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
AS2                        : ok=6    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Chạy file DS_HSRP.yml:

```
TASK [Creat Vlan on DS1] ****************************************************************************************************************
ok: [DS2]

TASK [Set mode trunk on DS2] ************************************************************************************************************
ok: [DS2]

TASK [Chuyen cong e0/1 - e0/3 sang mode trunk] ******************************************************************************************
changed: [DS2]

TASK [Show Vlan On DS1] *****************************************************************************************************************
ok: [DS2]

TASK [HSRP Vlan 10] *********************************************************************************************************************
changed: [DS2]

TASK [HSRP Vlan 20] *********************************************************************************************************************
changed: [DS2]

TASK [OSPF] *****************************************************************************************************************************
ok: [DS2]

TASK [Set ip] ***************************************************************************************************************************
changed: [DS2]

TASK [Set Ethernet1/0 IPv4 address] *****************************************************************************************************
ok: [DS2]

PLAY RECAP ******************************************************************************************************************************
DS1                        : ok=9    changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
DS2                        : ok=9    changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Chạy router.yml:

```
TASK [Set fastEthernet1/0 IPv4 address] ****************************************************************************************************
changed: [Router]

TASK [Set OSPF] ****************************************************************************************************************************
changed: [Router]

TASK [Set ip OSPF f0/1] ********************************************************************************************************************
changed: [Router]

TASK [Set ip OSPF f1/0] ********************************************************************************************************************
changed: [Router]

TASK [Show ip] *****************************************************************************************************************************
ok: [Router]

TASK [debug] *******************************************************************************************************************************
ok: [Router] => {
    "show_ip.stdout_lines": [
        [
            "Interface                  IP-Address      OK? Method Status                Protocol",
            "FastEthernet0/0            10.215.26.164   YES DHCP   up                    up      ",
            "FastEthernet0/1            172.16.12.1     YES manual up                    up      ",
            "FastEthernet1/0            172.16.13.1     YES manual up                    up      ",
            "Loopback1                  10.0.0.1        YES manual up                    up"
        ]
    ]
}

PLAY RECAP *********************************************************************************************************************************
Router                     : ok=10   changed=7    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```