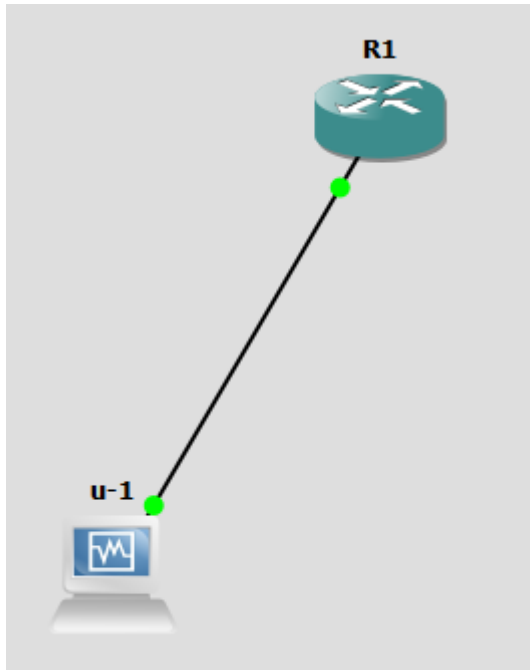


TP : Automation part 1

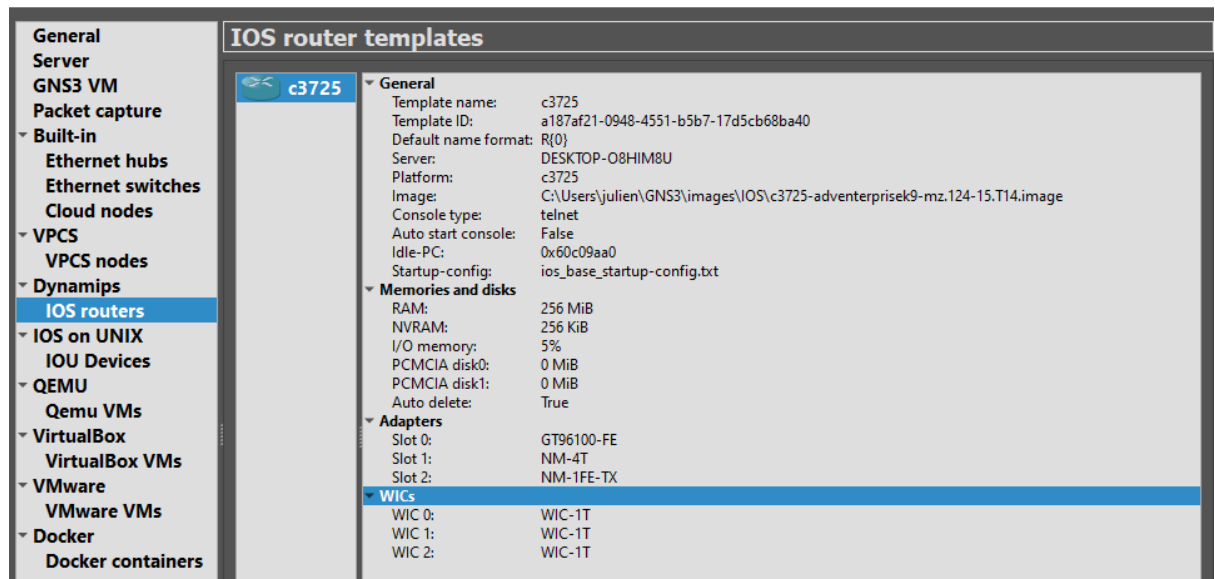
Le but de ce TP est de connecter dans GNS3 un routeur cisco type c3725 avec une VM virtualbox (ubuntu ou autre) et de piloter ce routeur via ansible installé sur cette VM.



Installation

1. Installer GNS3 (pas la version VM)
2. Télécharger cet OS
https://drive.google.com/file/d/1SdPDs8umg9rj0Qg2HxOglPLu5imT08Cs/view?usp=s_haring
 - a. RAM 256 Mb
 - b. IDLE-PC 0x60c09aa0
 - c. Adapters
 - i. GT96100-FE
 - ii. NM-4T
 - iii. NM-1FE-TX

3. Ajouter l'OS dans Edit > Préférences > Dynamips > IOS routers



4. Installer Virtualbox
 - a. Installer la VM
 - b. Installer net-tools, ansible, iperf
 - c. Installer les collections ansible ansible.netcommon et cisco.ios via ansible-galaxy
 - d. configurer un second adaptateur réseau en Generic Driver
5. Importer la VM dans GNS3 (Edit > Préférences)
 - a. dans Edit > Network bien augmenter le nombre d'adapters a >1
 - b. Activer Allow GNS3 to use any configured VirtualBox adapters

Configuration

1. Ajouter un Router R1 c3725 et une VM au projet.
2. Connectez les entres eux
3. Appliquez cette conf réseau : 10.0.0.0/24. R1 aura l'ip .1 et la VM la .10.
4. Configurer le routeur R1 pour accepter SSH depuis la VM.
5. Validez le ping depuis la VM puis SSH (attention il vous faudra ajouter des ciphers / kexAuth dans le .ssh/config)

Mumuse

Via la VM lancez cette commande :

```
> ansible all -i 10.0.0.1, -c ansible.netcommon.network_cli -u <username> -k -m cisco.ios.ios_facts -e ansible_network_os=cisco.ios.ios
```

renvoyez moi une screen de l'output de la commande.

TP : Automation part 2 (lundi 11/10)

Mumuse suite !

Creez le fichier d'inventaire **hosts** contenant le routeur, les informations pour s'y connecter
Puis lancez cette commande jusqu'à ce qu'elle vous renvoie le même résultat que la précédente :

```
> ansible all -i hosts -m cisco.ios.ios_facts
```

Playbook

1. On récupère la version
 - a. Créez le fichier **play.yml** qui sera le playbook réseau que nous utiliserons pour ce TP.
 - b. Ajouter à ce playbook la première tâche "récupération de la version" qui va utiliser le module **cisco.ios.ios_command** pour récupérer la version d'OS du routeur.
 - c. Enregistrer cette valeur dans une variable dans le playbook (voir **register**)
 - d. Ajouter une tâche de debug pour afficher cette valeur
2. On fait une backup !
 - a. Ajouter une tâche backup via le module **cisco.ios.ios_config**
 - b. Toutes les options nécessaires pour se connecter au routeur doivent être dans le fichier d'inventaire ou dans le dossier vars.
3. Interfaces FastEthernet 0/1
 - a. Via les modules **cisco.ios.ios_interfaces** & **cisco.ios.ios_l3_interfaces** changez le statut, la description et l'adresse IP de l'interface FastEthernet 0/1 du routeur.
4. Access-list
 - a. configurez l'acl extended **cloud_computing** permettant de se connecter depuis le réseau 10.0.0.0/24 au réseau 10.0.0.0/24 en ssh
5. Tags
 - a. Modifiez cette tâche avec le mot clé **acl** pour que ce bout de playbook puisse être lancé séparément
6. Router 2
 - a. Ajouter un routeur **r2** dans la simulation
 - b. il portera l'ip 10.0.0.2
 - c. placez la vm, et les 2 routeurs sur un switch dans le même vlan pour qu'ils puissent communiquer
 - d. Adapter le playbook pour que le playbook soit fonctionnel avec ce routeur

7. Restructurer votre projet comme suit :
- a. Mettez les routeurs r1 et r2 dans le groupe routers

```
|— ansible.cfg
|— group_vars
|   |— routers.yml
|— host_vars
|   |— r1.yml
|   |— r2.yml
|— inventory.yml
|— play.yml
```

Envoyez l'output de l'exécution du playbook et votre projet zipé.