

Administrer

Bastien Helec
Semestre 4

Sommaire

les liens en dessus sont clickable et renvoi vers les démonstrations correspondantes

- [Introduction](#) : Rappel de la compétence Administrer
- [Le niveau attendu en fin d'année](#)
- [Mon niveau actuel](#)
- [Ce que je prévois de faire pour améliorer mon niveau](#)
- [Ce que j'ai réaliser](#)
- [Ce que j'en conclu](#)

Introduction : Rappel de la compétence Administrer

[Retour au sommaire](#)

La compétence Administrer a pour objectif d'administrer et de concevoir un réseau. Les apprentissages attendus sont :

- AC 21.01 : Configurer et dépanner le routage dynamique dans un réseau
- AC 21.02 : Configurer et expliquer une politique simple de QoS et les fonctions de base de la sécurité d'un réseau
- AC 21.03 : Déployer des postes clients et des solutions virtualisées adaptées à une situation donnée
- AC 21.04 : Déployer des services réseaux avancés
- AC 21.05 : Identifier les réseaux opérateurs et l'architecture d'Internet
- AC 21.06 : Travailler en équipe pour développer ses compétences professionnelles

Celle-ci demande de savoir en situation professionnelle :

- Concevoir et administrer l'infrastructure du réseau informatique d'une entreprise
- Installer et administrer les services réseaux informatiques d'une entreprise
- Déployer et administrer des solutions fixes pour les clients d'un opérateur de télécommunication
- Gestion des postes clients
- Utilisation des technologies de virtualisations

Demandant ainsi au futur professionnel de répondre aux composantes essentielles suivantes :

- Choisir les solutions et technologies adaptées à la situation
- Respecter les principes fondamentaux de la sécurité informatiques
- Utiliser une approche rigoureuse pour la résolution des dysfonctionnements
- Respecter les règles métiers
- Assurer une veille technologique
- Communiquer avec les clients et les différents acteurs impliqués
- Utiliser une approche rigoureuse et méthodique (démarche scientifique)

Le niveau attendu en fin d'année:

[Retour au sommaire](#)

Selon moi le niveau attendu en fin d'année pour la compétence Administrer est de savoir concevoir un réseau en le créant de A à Z de le dimensionner en fonction du cahier des charges , de le configurer et de le dépanner. Ainsi que pouvoir assurer le bon fonctionnement des équipements lié a l'utilisation d'internet et a l'entreprise :

- Les équipements réseaux (routeurs, switchs, firewall, etc...)
- Les équipements de virtualisations (serveurs, hyperviseurs, etc...)
- Les équipements clients (ordinateurs, tablettes, smartphones, etc...)

Tout ce travail doit être effectuer en réalisant de la documentation et pouvoir obtenir des traces de ce qui a été fait pour pouvoir le reproduire ou le modifier si besoin s'en avoir a tout redimensionner et ainsi pouvoir communiquer et travailler en équipe.

Mon niveau actuel :

[Retour au sommaire](#)

Je pense que mon niveau dans la compétence Administrer correspond au niveau attendu en fin d'année. Malgré certains défauts liés à une documentation trop différentes d'un projet ou d'un rendu à l'autre et ainsi devoir tout reprendre depuis le début.

Je pense que mon niveau est bon car j'ai pu mettre en place des réseaux avec des protocoles de routage dynamique (BGP, OSPF, EIGRP, IBGP, RIP) pouvoir mettre en place des loopback et des VLAN, j'ai tendance à observer que le réseau suit bien la demande en appliquant du QoS de manière manuelle (vérification humaine), ainsi que le déploiement sans problème d'un serveur à l'autre en vérifiant les adresses IP occupées ou les réseaux occupés avec nmap, puis vérifier le bon fonctionnement en utilisant des outils comme ping, traceroute, mtr, etc...

Puis surtout j'ai pris l'habitude de réaliser l'administration système sur des conteneurs ou sur des machines virtuelles dans un premier temps de part ma distribution de choix : Fedora. Mais aussi pour me permettre de garder un ordinateur sain et ainsi ne pas avoir des problèmes différents d'un projet à l'autre, ce qui m'arrivait trop souvent en première année.

De plus aujourd'hui je communique auprès des premières années des conseils pour pouvoir permettre de ne pas se retrouver dans la difficulté que j'ai pu avoir en première année.

Avec l'expérience obtenue en deuxième année avec les SAE clouds, j'ai pu mettre en place des infrastructures plus complexes tout comme en interne avoir mis en place une infrastructure chez moi.

Ce que je prevois de faire pour améliorer mon niveau :

[Retour au sommaire](#)

Je prevois de me renseigner plus sur les documentations des différents protocoles de routage dynamique comme par exemple les RFC du BGP , du RIP etc afin des les comprendre et pouvoir ainsi utiliser tout le potentiel associés au ceux-ci.

Je pense aussi réaliser des infrastructures toujours plus complexe en me permettant de mettre en pratique toutes les informations que j'aurais apprise auprès des documentations, Et donc ainsi réaliser moi meme une documentation sous forme de tutoriel.

Ce que j'ai réaliser :

Introduction :

Dans cette AC je doit ainsi prouver que je suis capable de configurer et donc de depanner le routage dynamique dans un réseau.

Grâce au cours exploiter durant le S3, mon stage et S4 et ma premiere année de formation R&T, j'ai pu apprendre a configurer et depanner des routeurs et des switchs Cisco et mikrotik.

Mais également de configurer les ordinateurs clients et serveurs pour qu'ils puissent correspondre au besoin demander.

Les protocoles de routage dynamique :

Dans ma formation nous avons pu voir les protocoles de routage dynamique suivant elle utilise ainsi la couches transport du modèle OSI :

- RIP (Routing Information Protocol)
- OSPF (Open Shortest Path First)
- EIGRP (Enhanced Interior Gateway Routing Protocol)
- BGP (Border Gateway Protocol)
- STP (Spanning Tree Protocol)

Definitions :

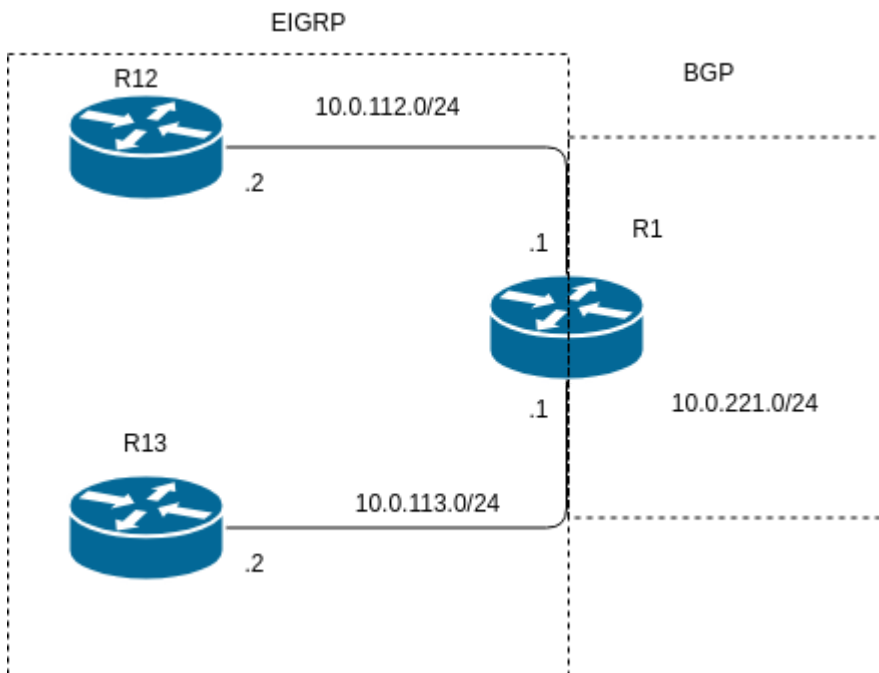
- **RIP** : Le protocole RIP est un protocole de routage. Il utilise le nombre de sauts comme métrique. Il est limité à 15 sauts. Il est donc utilisé pour des réseaux de petite taille. Il est très simple à configurer et à déployer. Il est très peu utilisé de nos jours, sa sécurité est très faible et il est très lent à converger.
- **OSPF** : Le protocole OSPF est un protocole de routage créer par Cisco à état de lien. Il utilise la bande passante comme métrique. Il est utilisé pour des réseaux de taille moyenne à grande. Il est plus complexe à configurer et à déployer que RIP. Il est plus sécurisé et plus rapide à converger de part l'utilisation de l'algorithme de Dijkstra.
- **EIGRP** : Le protocole EIGRP est un protocole de routage. Il utilise la bande passante et la charge comme métrique. Il est utilisé pour des réseaux de taille moyenne à grande. Il est plus complexe à configurer et à déployer que RIP. Il est plus sécurisé et plus rapide à converger de part l'utilisation de l'algorithme de Dijkstra
- **BGP** : Le protocole BGP est un protocole de routage. Il utilise la bande passante et la charge comme métrique. Il est utilisé pour des réseaux de taille moyenne à grande. Il est plus complexe à configurer et à déployer que RIP. Il est plus sécurisé et plus rapide à converger de part l'utilisation de l'algorithme de Dijkstra. De plus il utilise le système d'autonomie (AS) pour définir les routes, ainsi celle-ci permettent de definir plusieurs réseaux dans une meme zone et ainsi de pouvoir faire de l'agrégation de routes. J'ajouterais aussi que la configuration BGP permet de faire des redistribution de routes entre les protocoles de routage dynamique pour permettre de voir apparaitres que certains réseau soit sur d'autre AS ou bien utilise d'autre protocole de routage.

- **STP** : Le protocole STP est un protocole qui a pour but d'éviter les bouclages réseaux et ainsi limité le calcul du switch, ce n'est pas a proprement parait un protocole de routage mais est très utile pour depanner et configurer un routage dynamique.

Démonstration de mes compétences :

- Configuration est mise en place d'un réseau avec EIGRP : Dans le contexte de la formation dans la ressource R302 j'ai pu mettre en place via l'application GNS3 un réseau interne avec le protocole EIGRP et utiliser l'attributs **weight** pour le BGP qui nous permet de choisir la route que l'on souhaite utiliser :

Tout d'abord j'ai mis en place le réseau suivant :



Puis on a configurer les routeurs de cette manière :

```
R12(config)# interface FastEthernet0/0
R12(config-if)# ip address 10.0.112.2 255.255.255.0
R12(config-if)# no shutdown
R12(config-if)# exit
R12(config)# do write memory
R12(config)# router eigrp 100
R12(config-router)# network 0.0.0.0 255.255.255.255
R12(config-router)# exit
R12(config)# do write memory
```

Ici on a configurer le routeur R12 avec une adresse IP et le protocole eigrp avec le numéro de l'AS 100. J'ai terminer par sauvegarder la configuration.

Les mêmes commandes ont été effectuées sur les autres routeurs en changeant l'adresse IP de l'interface FastEthernet0/0. Sauf R1 qui a été configurer en BGP

Pour etres sur que cela fonctionne nous avons effectuer un **show ip route** sur le routeur R12 :

```
show ip route
```

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-
2       ia - IS-IS inter area, * - candidate default, U - per-user static
route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

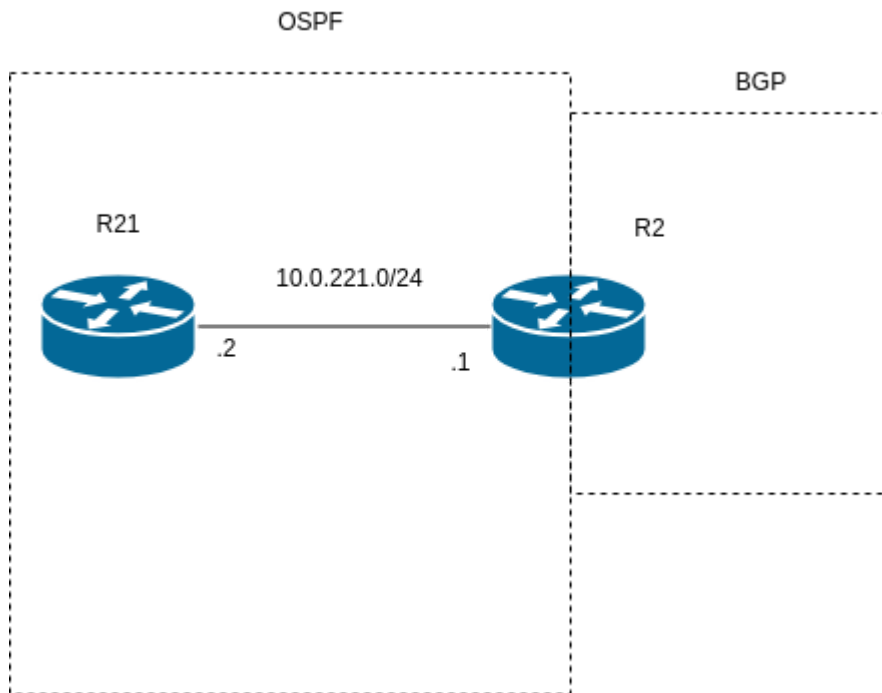
```
      100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
D      100.0.0.0/8 is a summary, 00:30:36, Null0
C      100.1.0.0/24 is directly connected, Loopback0
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
D      10.0.14.0/24 [90/30720] via 10.0.112.1, 00:30:36, FastEthernet0/0
D      10.0.12.0/24 [90/30720] via 10.0.112.1, 00:30:36, FastEthernet0/0
D      10.0.0.0/8 is a summary, 00:30:37, Null0
D EX   10.0.23.0/24 [170/25602816] via 10.0.112.1, 00:30:36,
FastEthernet0/0
C      10.0.112.0/24 is directly connected, FastEthernet0/0
D      10.0.113.0/24 [90/30720] via 10.0.112.1, 00:30:39, FastEthernet0/0
D EX   10.0.221.0/24 [170/25602816] via 10.0.112.1, 00:30:39,
FastEthernet0/0
```

On peut voir que le réseau est bien détecté en EIGRP et que les routeurs sont bien connectés entre eux.

- Configuration est mise en place d'un réseau avec OSPF :

Dans le contexte de la formation dans la ressource R302 j'ai pu mettre en place via l'application GNS3 un réseau interne avec le protocole OSPF :

Tout d'abord j'ai mis en place le réseau suivant :



Puis on a configurer le routeurs de cette manière :

```
R21(config)# interface FastEthernet0/0
R21(config-if)# ip address 10.0.221.2 255.255.255.0
R21(config-if)# no shutdown
R21(config-if)# exit
R21(config)# do write memory

# Configuration de l'OSPF

R21(config)# router ospf 1
R21(config-router)# network 10.0.221.0 255.255.255.0 area 0
R21(config-router)# exit
R21(config)# do write memory
```

Ici on a une configuration OSPF classique en définissant le numéro de l'AS 1 et le réseau

J'ai ainsi configurer également sur le router R2 uniquement la partie ospf presente ci-dessus.

Puis on a réaliser un `show ip route` sur le routeur R21 :

```
show ip route
```

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-
2      ia - IS-IS inter area, * - candidate default, U - per-user static
route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

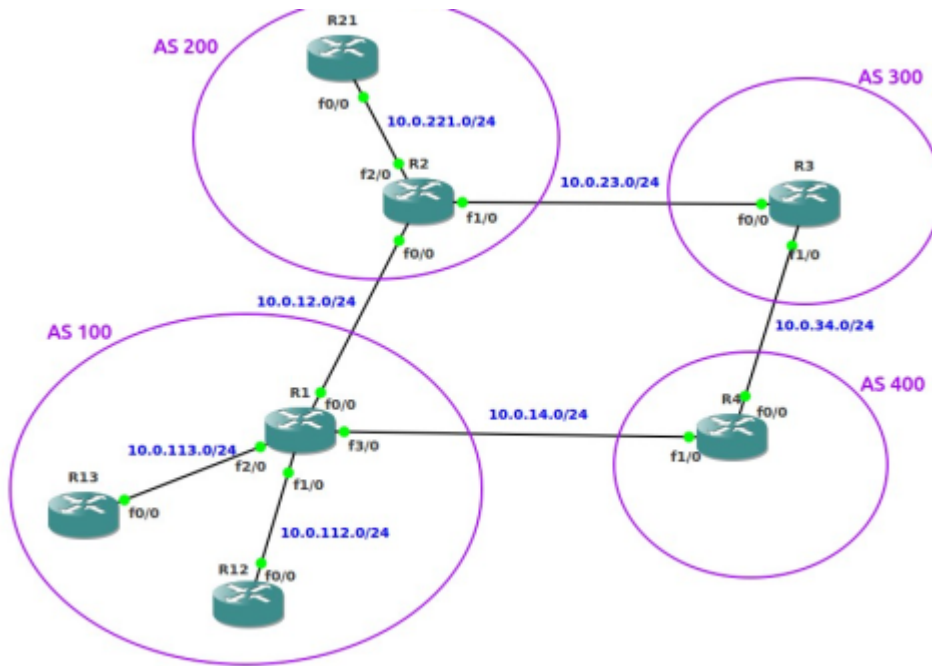
```
      100.0.0.0/8 is variably subnetted, 4 subnets, 3 masks
O      100.2.0.2/32 [110/2] via 10.0.221.2, 00:02:43, FastEthernet0/0
O E2   100.0.0.0/8 [110/1] via 10.0.221.2, 00:02:43, FastEthernet0/0
O E2   100.1.0.0/24 [110/1] via 10.0.221.2, 00:02:43, FastEthernet0/0
C      100.2.0.0/24 is directly connected, Loopback0
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
O E2   10.0.14.0/24 [110/1] via 10.0.221.2, 00:02:43, FastEthernet0/0
O      10.0.12.0/24 [110/2] via 10.0.221.2, 00:02:44, FastEthernet0/0
O E2   10.0.0.0/8 [110/1] via 10.0.221.2, 00:02:44, FastEthernet0/0
O      10.0.23.0/24 [110/2] via 10.0.221.2, 00:02:44, FastEthernet0/0
O E2   10.0.112.0/24 [110/1] via 10.0.221.2, 00:02:44, FastEthernet0/0
O E2   10.0.113.0/24 [110/1] via 10.0.221.2, 00:02:44, FastEthernet0/0
C      10.0.221.0/24 is directly connected, FastEthernet0/0
```

On peut voir que le réseau est bien détecté en OSPF et que les routeurs sont bien connectés entre eux.

- Configuration est mise en place d'un réseau avec BGP :

Dans le contexte de la formation dans la ressource R302 j'ai pu mettre en place via l'application GNS3 un réseau interne avec le protocole BGP :

Tout d'abord j'ai mis en place le réseau suivant :



On a configuré les routeurs R1 R2 R3 et R4 de sorte à avoir les configurations principales pour les réseaux BGP.

Puis en fonction des AS on a configuré le BGP de cette manière :

```
R1(config)# router bgp 100
R1(config-router)# neighbor 10.0.12.2 remote-as 200
R1(config-router)# neighbor 10.0.14.4 remote-as 400
R1(config-router)# exit
R1(config)# do write memory
```

Sur le routeur R1 on a configuré le BGP avec le numéro de l'AS 100 et les voisins avec leur AS respectif.

Les mêmes commandes ont été effectuées sur les autres routeurs en correspondance avec les AS et les réseaux.

Quand tous les routeurs BGP sont créés on reçoit des messages entre les routeurs :

```
*Mar 1 00:00:17.923: %BGP-5-ADJCHANGE: neighbor 192.168.23.3 Up
```

puis on voit les routes BGP apparaître sur les routeurs comme ici sur R3:

```
show ip bgp
```

```
BGP table version is 27, local router ID is 100.3.0.3
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

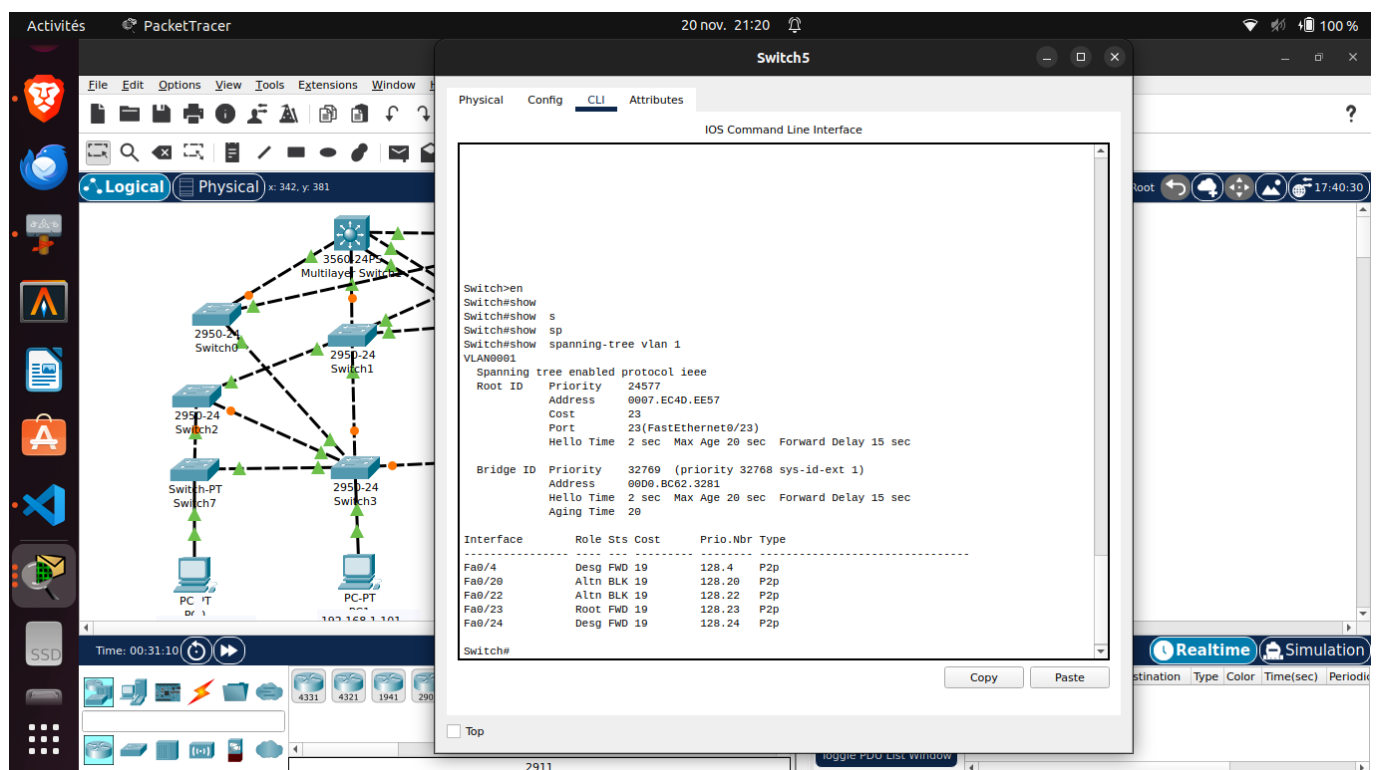
```
r> 10.0.23.0/24      100.2.0.2          0          0 200 ?
```

```
> 10.0.221.0/24     100.2.0.2          0          0 200 ?
```

On peut voir que le réseau est bien détecté en BGP et que les routeurs sont bien connectés entre eux.

- Configuration est mise en place d'un réseau avec STP :

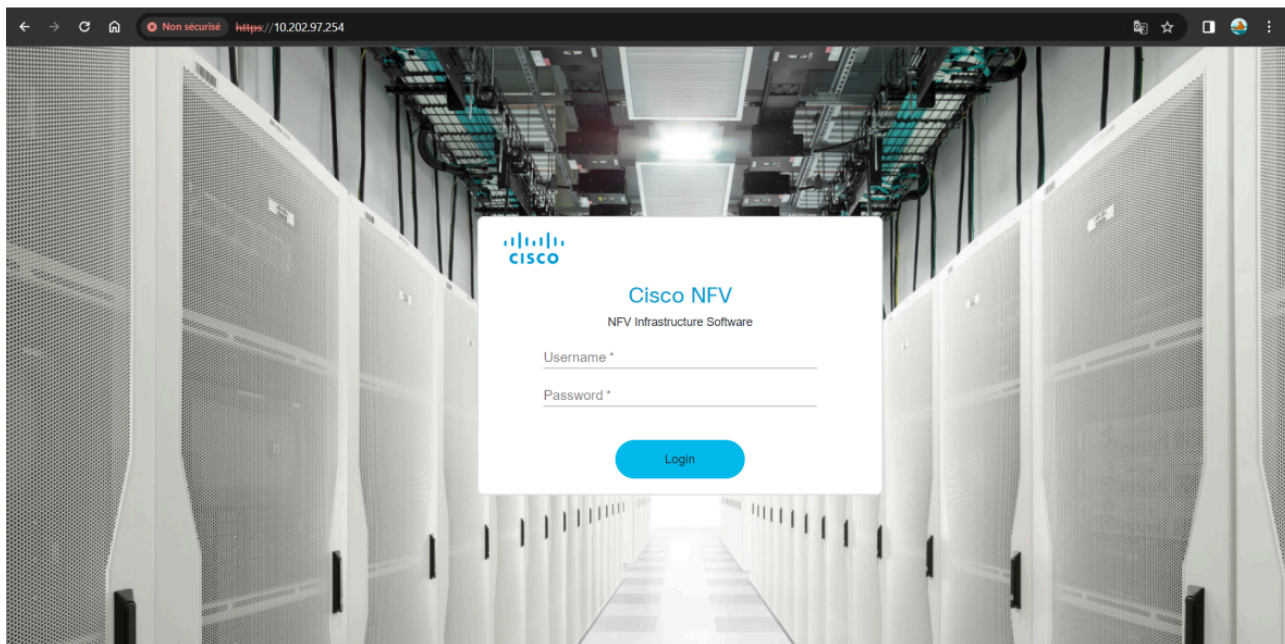
Dans la ressource R301 j'ai pu mettre en place via l'application Cisco Packet Tracer un réseau interne avec le protocole STP :



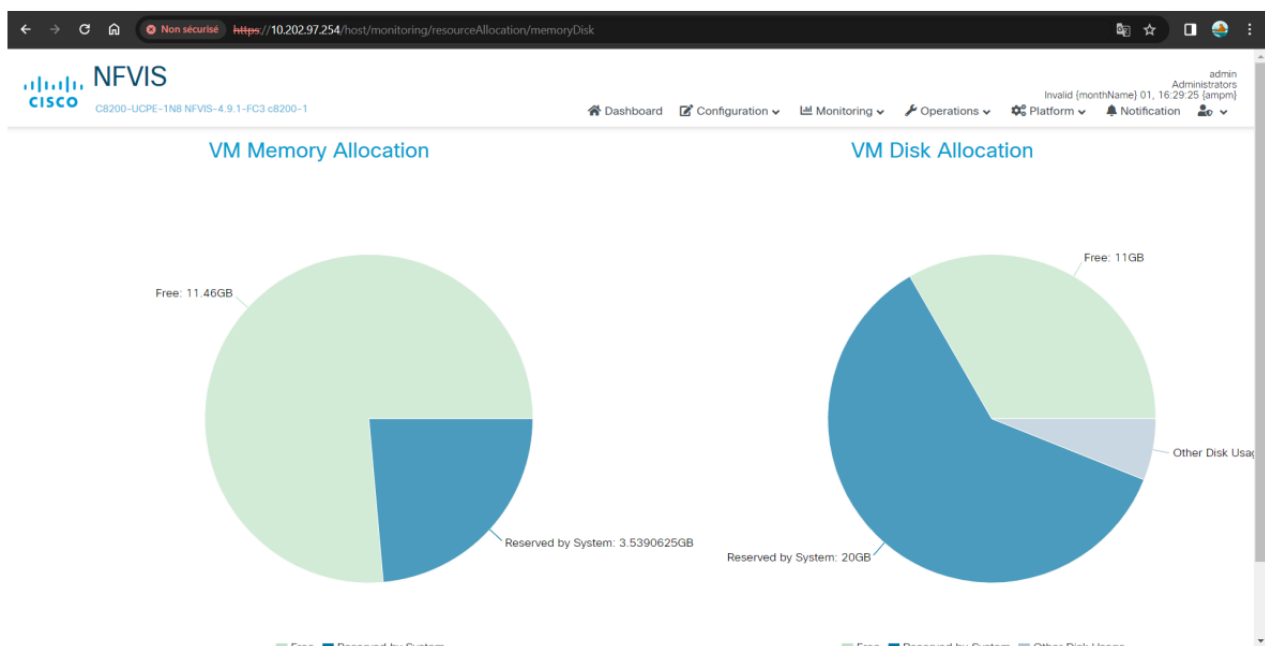
On peut voir pour un réseau complexe les routes prises par le protocole STP pour éviter les bouclages réseaux.

- Virtualisation du réseau :

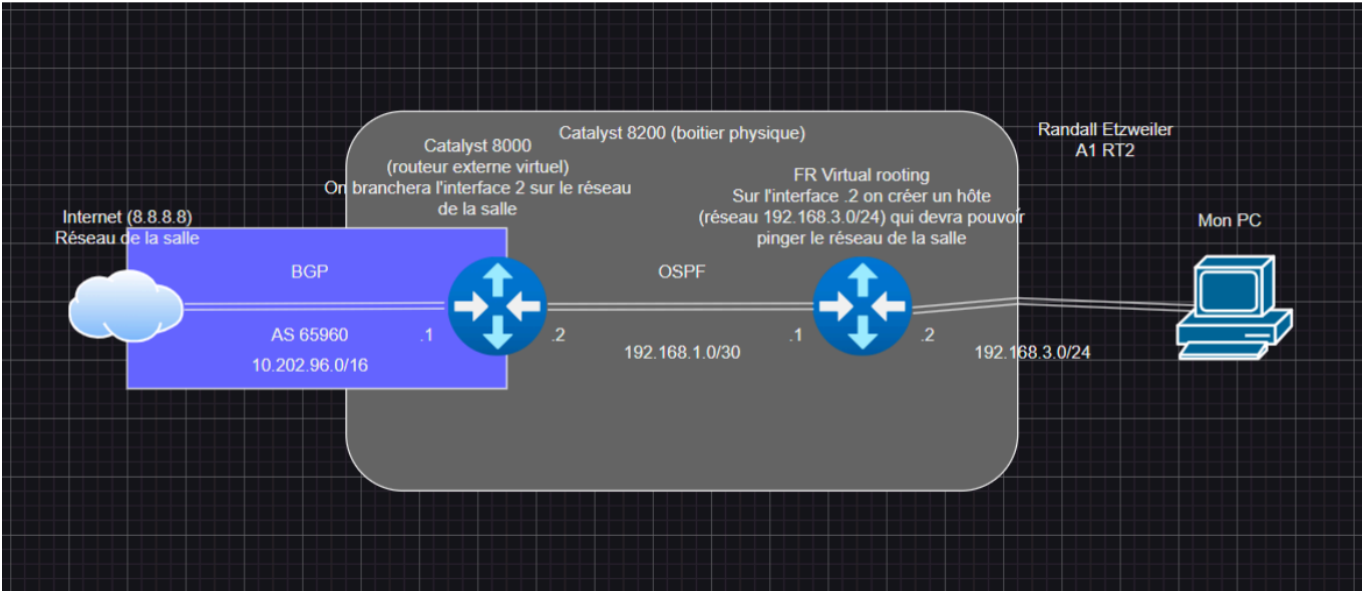
Durant la SAE.3.Devcloud.03 On a pu virtualiser des appareils de routages comme le routeur c8000 avec le router edge csr1000v.



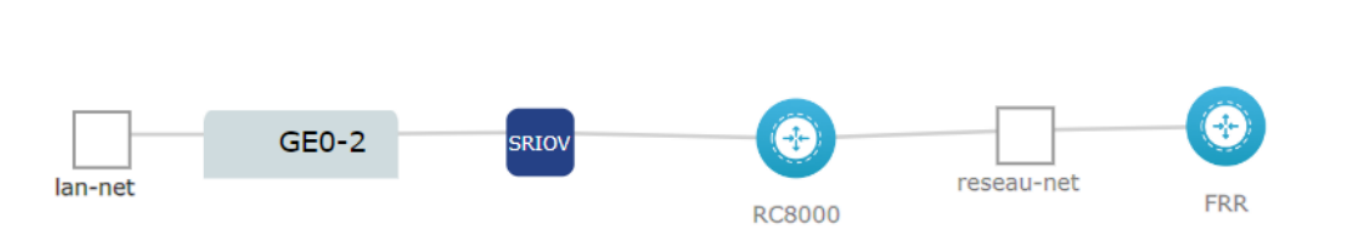
Le router edge csr1000v étant notre appareil de virtualisation global nous avons mis en place avec mon binôme le logiciel NFVIS qui nous permet de voir la consommation actuelles du hardware de l'appareil



Une fois que notre appareil était configuré nous avons réfléchi à notre réseau à l'intérieur de notre routeur edge arrivant ainsi à cette solution :

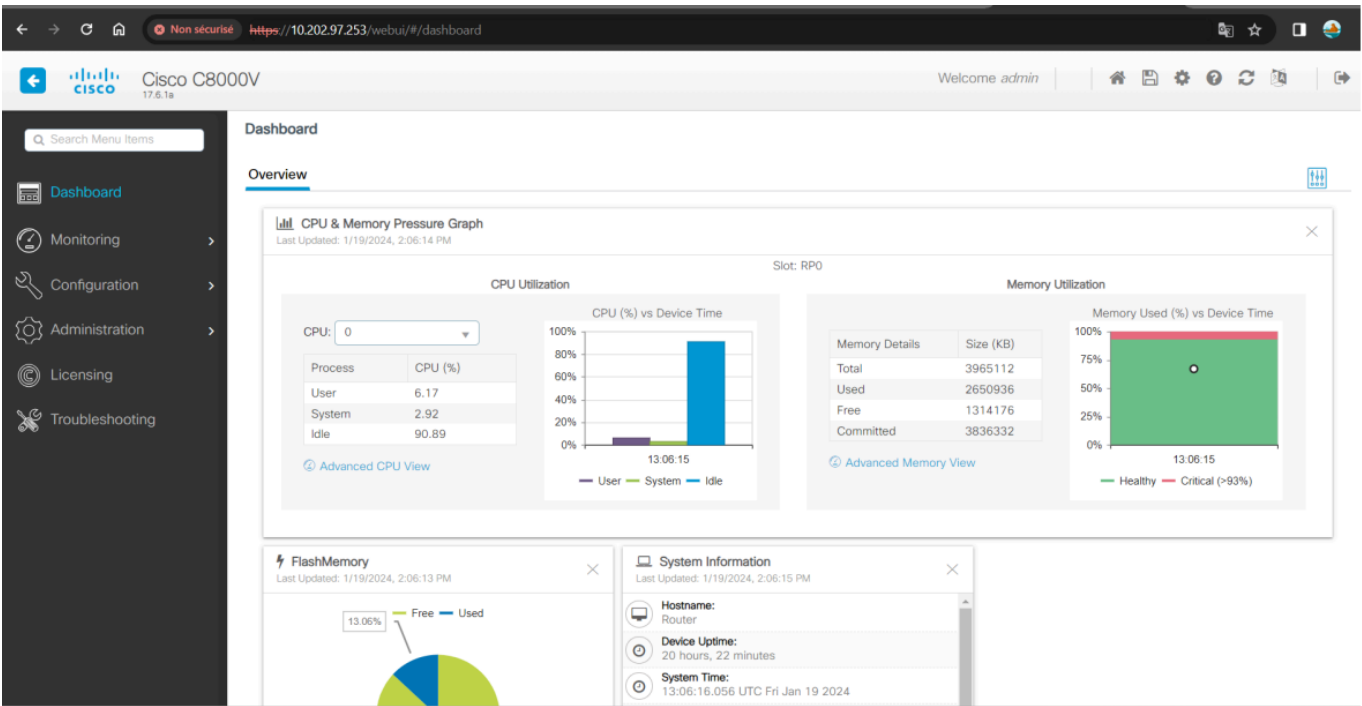


En outre passant ici les configurations des routeurs FRR et C8000 effectués afin d'arriver à une communication entre le réseau de la salle et le PC de mon binôme. Nous avons pu créer notre réseau virtuellement une fois les 2 routeurs créés grâce à NFVIS :

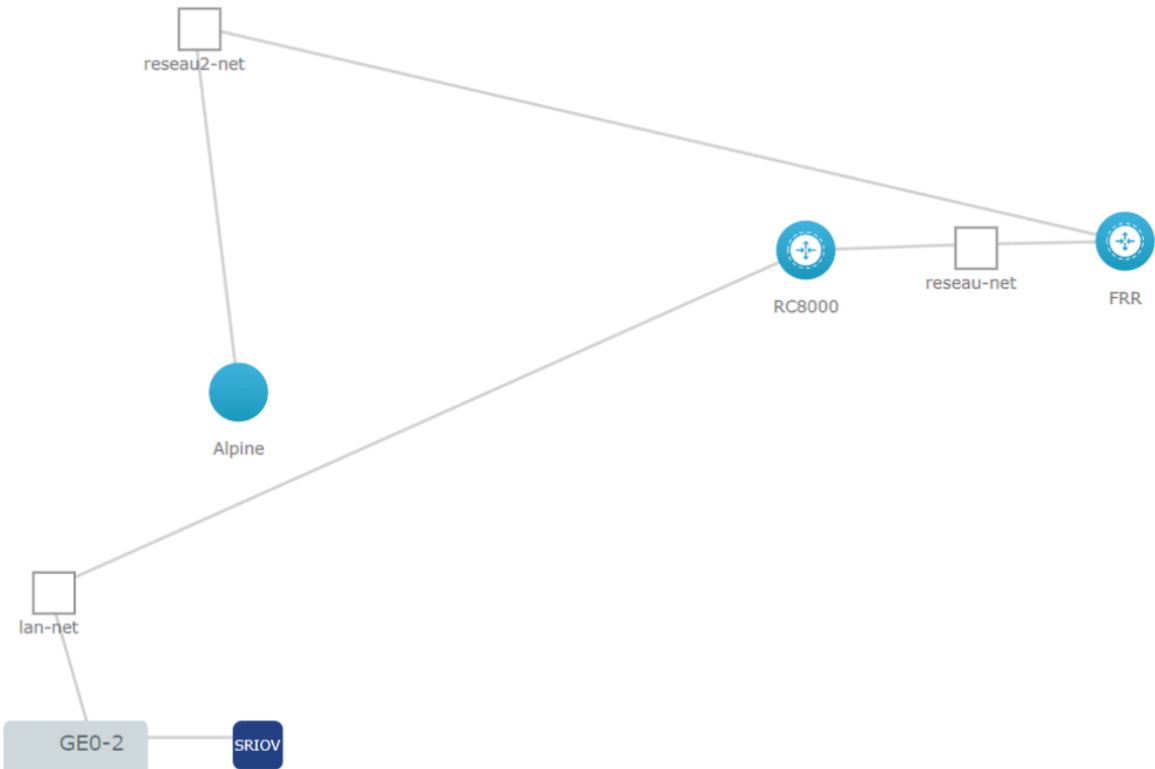


GE0-2 correspondant à une interface physique de notre routeur edge

Une fois notre réseau initial terminé nous avons pu nous connecter à l'interface web du routeur C8000 voyant ainsi son utilisation :

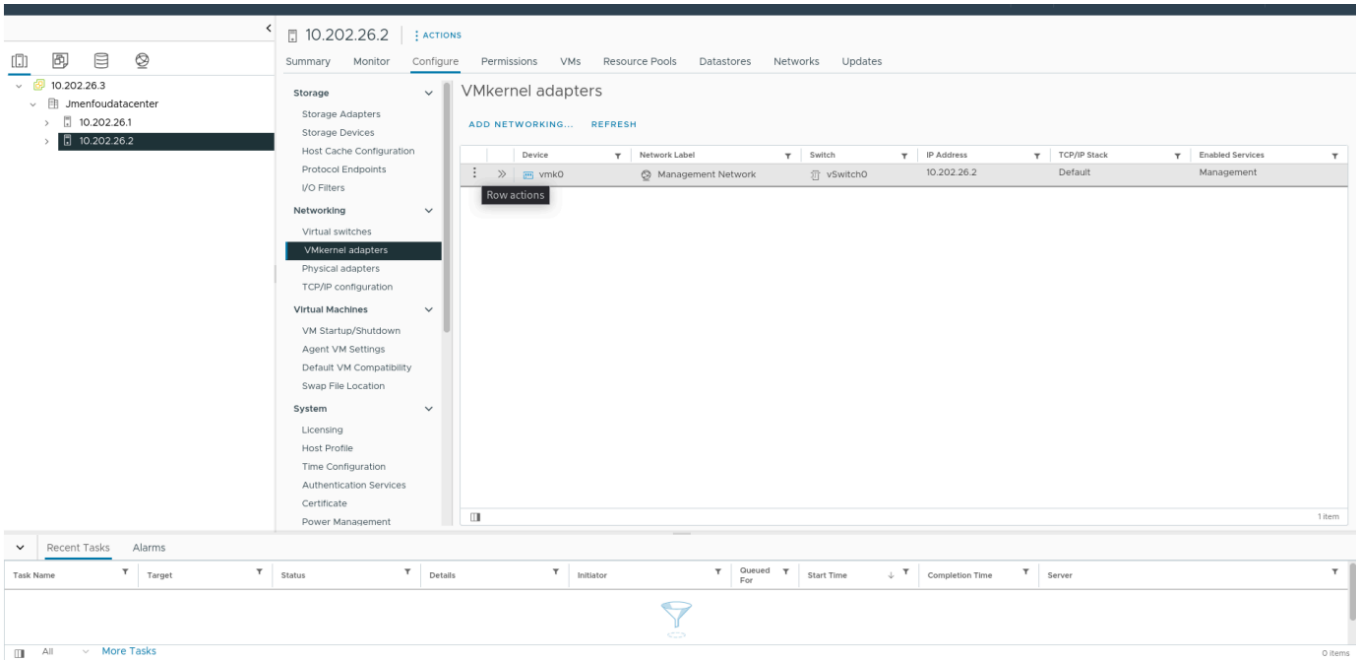


Pour arriver a ce réseau final :



- Virutalisaton d'une infrastructure :

En parallele lors de la SAE.3.Devcloud.04 j'ai pu mettre en place une infrastrucutre virtualisé via 2 serveurs a dispositions, l'objectif était de mettre en application et de ce familiariser avec les solutions Cloud comme ESXI de VMware ou encore Proxmox

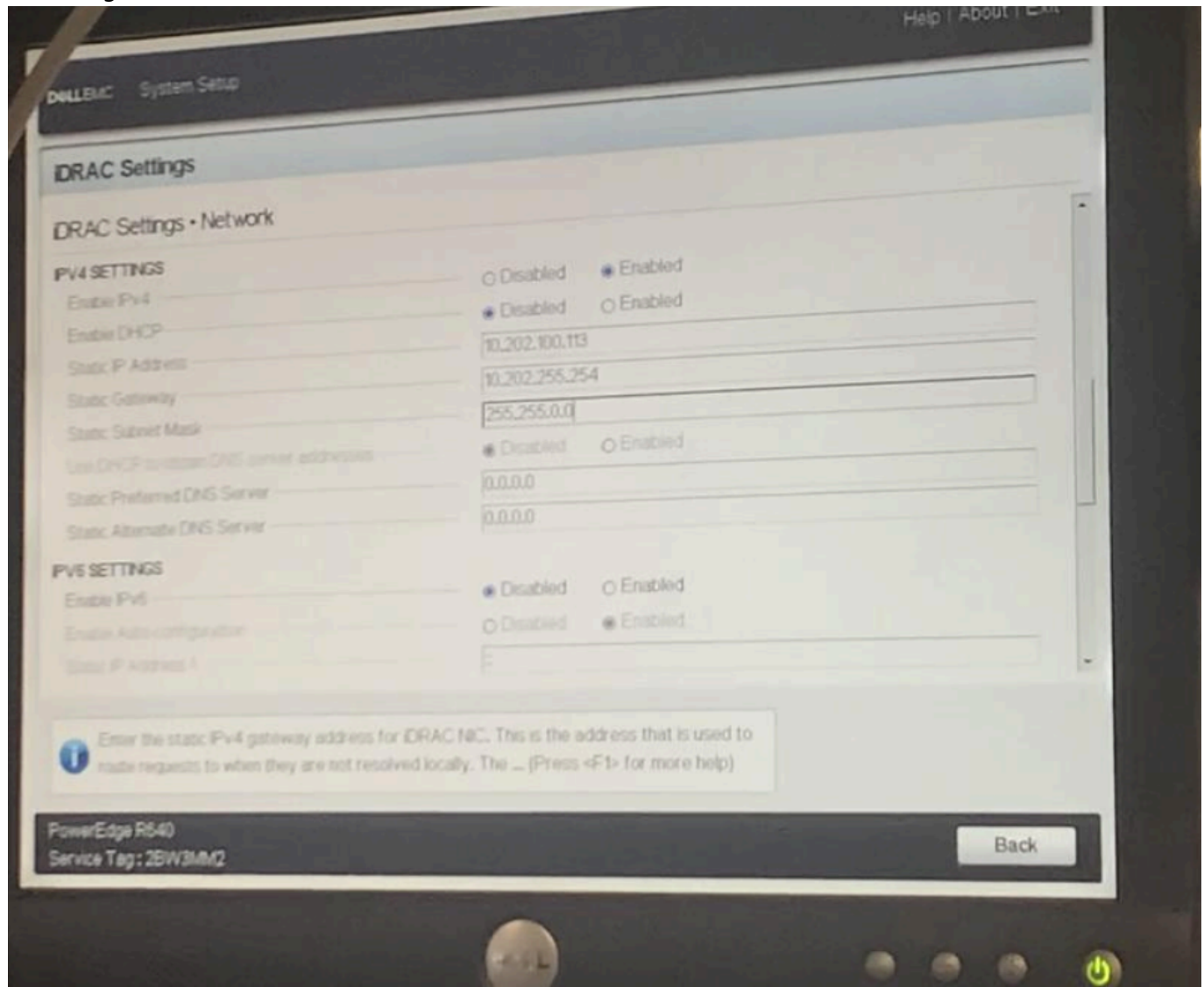


Pour cela nous avons, mon binôme et moi, utiliser VM ware en réalisant une installation classique sur un serveur HP Proliant DL360 et un serveur Dell PowerEdge R640. L'avantage avec les serveur est qu'ils possèdent 2 carte réseau dont l'une accessible lorsque le serveur est atteint sur Dell on l'appel l'iDRAC et sur HP l'iLO.

Ainsi nos deux serveurs ce sont retrouver dans l'adressage IP suivant :

iLO : 10.202.100.112 partie ESXi : 10.202.26.1 iDRAC : 10.202.100.113 partie ESXi : 10.202.26.2

Les configurations des interfaces iLO et iDRAC on était effectué directement sur les serveurs associées.

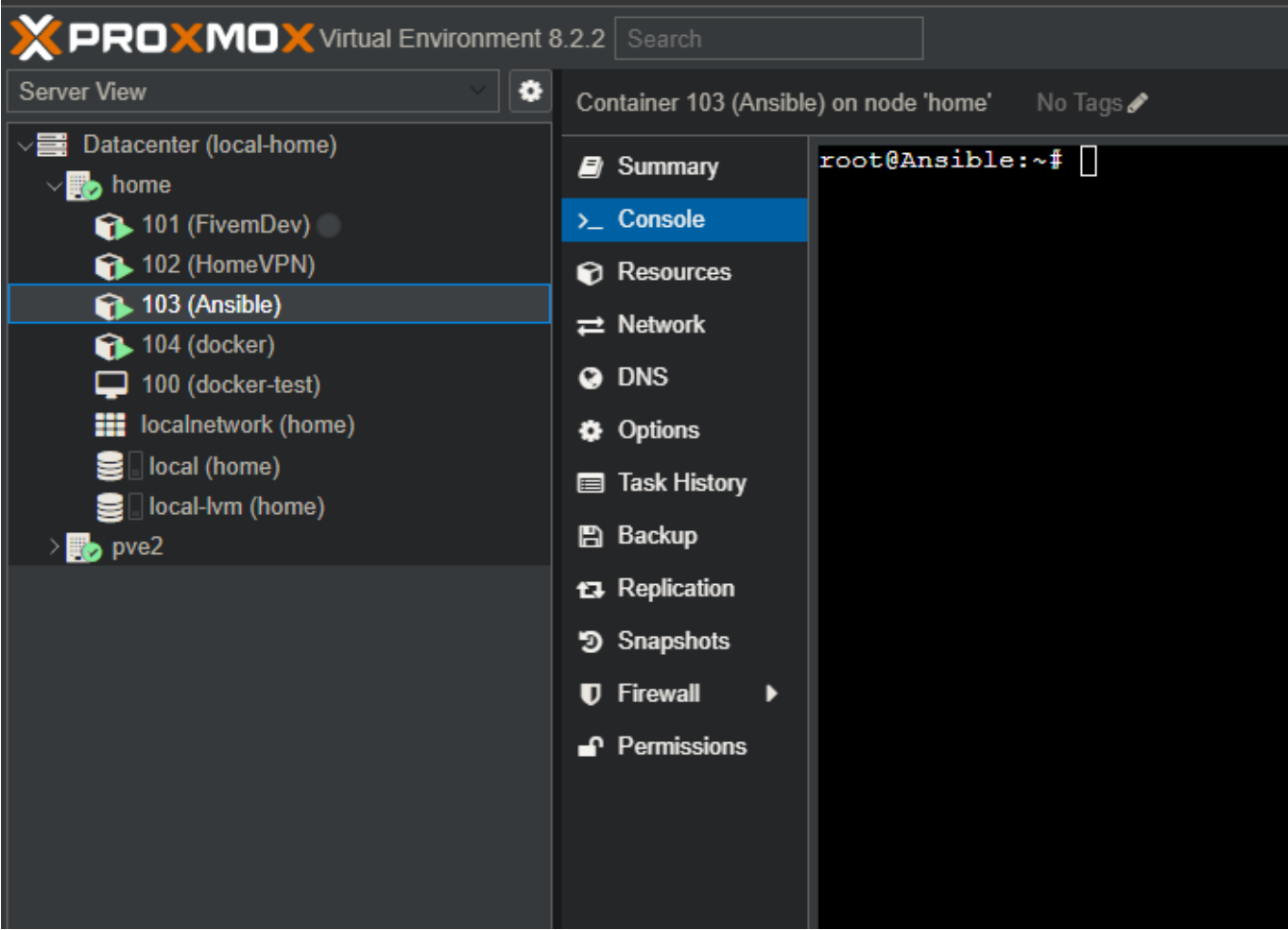


La différence est notables lorsqu'il s'agit de la configuration du serveur sous iDRAC nous avons eu tendance a chercher plus comparer a l'iLO qui celui-ci indiqué d'office ou était les points demander.

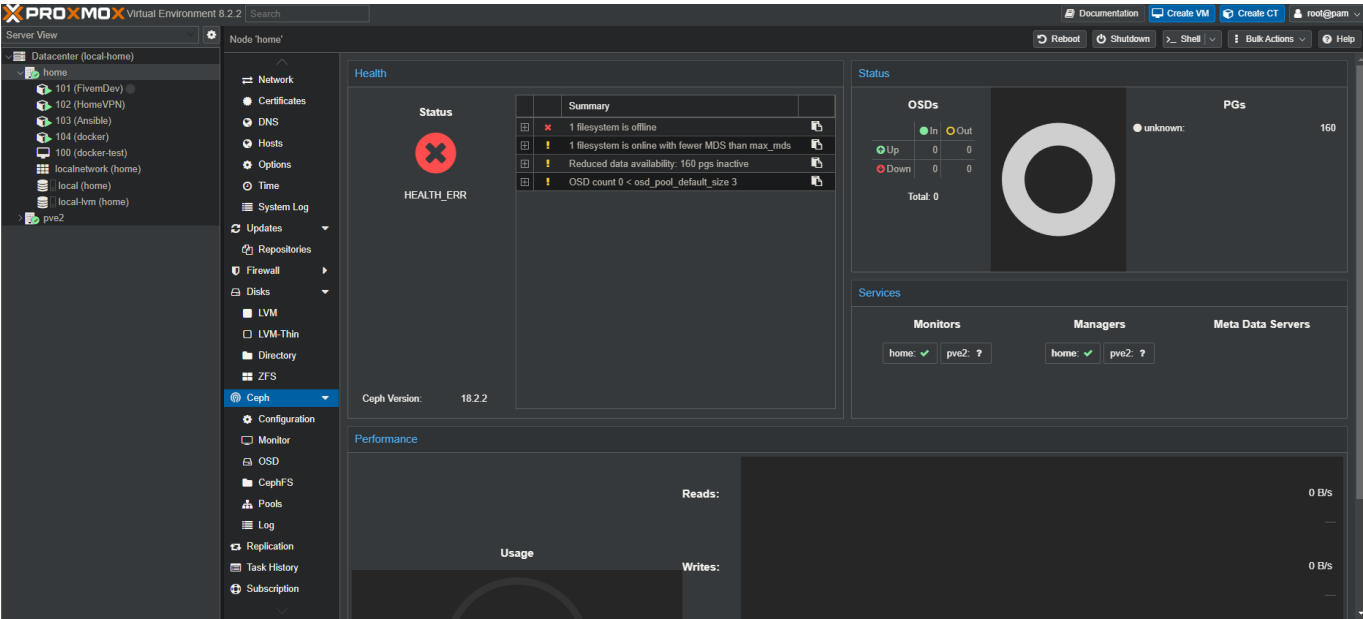
Une fois la configuration et l'installation sur ESXi terminer nous avons eu besoin d'une VM pour nous permettre d'obtenir le Vsphere client de chaque serveur. Un point qui paru assez étrange de mon cas étant donné que nous aurions pu directement installer le Vsphere client sur les serveurs.

Ainsi en finalité nous avons pu obtenir un réseau de 2 serveurs correspondant au serveur configuré pour nous permettre de créer des machine virtuel et conteneur.

De mon coté a titre personnel j'ai pu mettre en place en cluster Proxmox me permettant ainsi de me faire du développement informatique, une phase de test et une connectivité VPN via un conteneur LXC.



Ce ne fu pas de tout repos car l'un de mes serveurs ne pouvait pas etre démarrer par affichage de par un default interne , la RAM était mal intégrer dans le boitier donnant l'impression que celui-ci pouvait avoir un affichage sans réellement l'avoir ce qui complexifier la taches afin d'entamer le cluster. Une fois ce probleme résolu j'ai ainsi donc pu mettre en place un ceph nous permettant d'afficher un résumé succints de la consommation de chaque appareil.



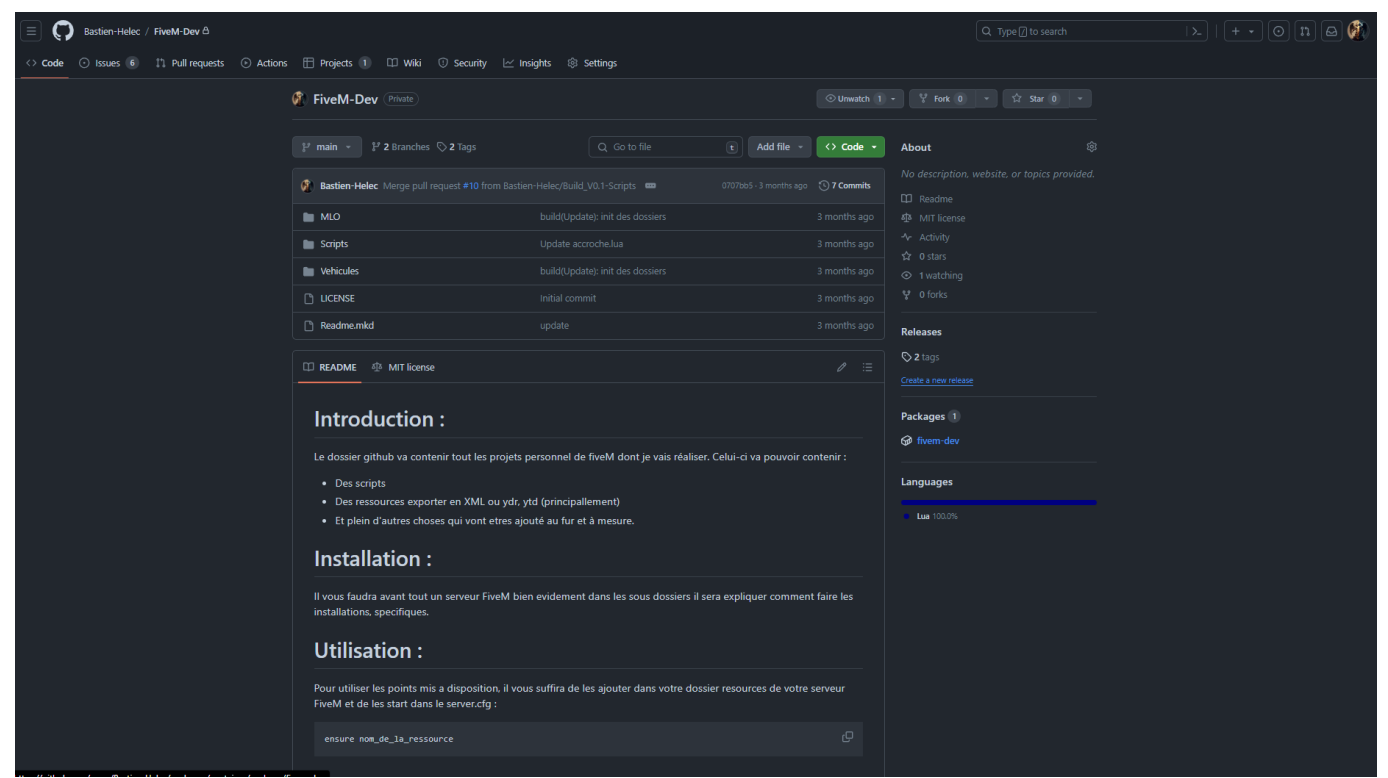
J'ai décidé d'utiliser des conteneurs de par la présence d'un hardware peu performant avec plusieurs VM a l'interieur.

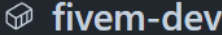
Ainsi cela ma permit de mettre en place un VPN wireguard, ainsi qu'un serveur fivem qui correspond a du developpement de jeu video me permettant ainsi de réaliser des scripts et tout ce qui s'en suit pour une communauté de jeu de rôles.

Fivem est un produit de l'entreprise cfx.re permettant d'ajouter une surcouche a un jeu video internationnellement connu (GTA 5) offrant la possibilité au joueurs de développer une toute nouvelle alternative au jeu d'origine ici du roleplay.

également a l'interieur ma phase de test avant le deploiment de plusieurs solutions.

Comme la création d'un container docker permettant un deploiement rapide d'un serveur fivem ainsi qu'une documentation permettant le deploiement efficaces. Initialiser sur mon github :





Install from the command line

Learn more about packages

```
$ docker pull ghcr.io/bastien-helec/fivem-dev:v1.0
```

Recent tagged image versions

V1.0

Published 2 months ago · Digest

26

View and manage all versions

Readme.mkd

Introduction :

Le dossier github va contenir tout les projets personnel de fiveM dont je vais réaliser. Celui-ci va pouvoir contenir :

- Des scripts
- Des ressources exporter en XML ou ydr, ytd (principalement)

Details

Bastien-Helec

FiveM-Dev

MIT License

Last published

2 months ago

Issues

6

Total downloads

26

Collaborators

1

Bastien-Helec

bastien_helec

Open an issue

Package settings

Installation :

Il vous faudra avant tout un serveur FiveM bien evidement dans les sous dossiers il sera expliquer comment faire les installations, specifiques.

Il vous faudra avant tout un serveur FiveM bien evidement dans les sous dossiers il sera expliquer comment faire les installations, specifiques.

Je met a disposition un serveur fivem a configurer pour Attention c'est un docker il vous faudra docker pour pouvoir le recuperer.

Vous devrez créer un réseau avec l'adresse réseau de la ou vous etes :

```
docker network create --subnet=<nouvelle_adresse_IP> <votre réseau>
```

Une fois que vous avez votre conteneur vous devez le run avec cette commande :

```
docker run --net="<votre réseau>" -d -p 30120:30120 -p 30120:30120/udp -p 8080:8080 -p 30110:30110 -p 4
```

Je vous inviterais a directement vous connectez sur le package : User : fivem Password : fivem

Note : Lisez le readme.me présent dans son /home/ il vous indiquera ce qu'il faut faire

Merci de changer le mot de passe une fois que vous avez récupérer le conteneur avec

```
sudo fivem passwd
```

Et la préparation des playbooks ansible comme ceux utilisé pour la SAE4.devcloud.01 ayant pour objectif le déploiement de microservice avec une solution de supervision des microservices.

Egalement sur mon serveur j'ai mis en place un systeme de replication qui récupèrent les dernieres sauvegarde réaliser sur les machines présentes, ainsi qu'un démarrage automatique pour les conteneurs essentielles comprenant le démarrage via init.d d'un nouveau services créer.

- Configuration des VPN sur serveur et client :

De part les blocages de port de l'IUT j'ai du mettre en place un VPN entre mon clusters proxmox chez moi, pour me permettre d'accéder a tout les ports extérieur.

Pour ce faire j'ai créer un VPN avec wireguard sur un serveur proxmox dans un contener LXC, puis une configuration entre mon serveur proxmox chez moi et les différents clients.

De part la sécurité du VPN je ne vais pas montrer par des captures d'écrans les configurations mais je vais expliquer comment j'ai fait :

1. Installation de wireguard sur le serveur proxmox :

```
apt install wireguard
```

2. Génération des clés sur le serveur proxmox :

```
wg genkey | tee privatekey | wg pubkey > publickey
```

3. Configuration du serveur wireguard et du client wireguard:

```
[Interface]
PrivateKey= [ Clefs priver du serveur Clefs priver du client]
Address = [ Adresse IP du serveur wireguard Adresse IP du client wireguard]
ListenPort = 51820

[Peer]
PublicKey = [ Clefs public du client wireguard Clefs public du serveur wireguard]
AllowedIPs = [ Adresse IP du client wireguard Adresse IP du serveur wireguard (ou
0.0.0.0/0 pour le full tunnel)]
Endpoint = [ Adresse IP du client wireguard ou ip publique ]
PersistentKeepalive = 25
```

Bien evidement on a du configurer a l'interieur du serveur VPN les regles de forwarding et de NAT pour permettre au client de pouvoir accéder a internet.

```
iptables -t nat -A POSTROUTING -s [ Réseau wireguard ] -o eth0 -j MASQUERADE
```

Et sur le routeur on autorise l'accès au serveur wireguard en ouvrant le port 51820.

Sous linux on utilise seulement les CLI pour pouvoir configurer le VPN.

Sous windows et android l'entreprise Wireguard a créer une application pour pouvoir configurer le VPN de manière graphique. Sans pour autant changer les configurations de bases.

Pour ma part j'ai configurer mon VPN en full tunnel pour pouvoir acceder a tout les ports et mon cluster proxmox.

- Support client :

Dans mon stage j'ai pu réaliser un dépannage auprès de plusieurs clients sur l'installation d'un nouvel ordinateur avec les logiciels primordiaux, également un depannage sur une imprimantes qui suite au passage d'un technicien de l'entreprise de l'imprimantes a perturber la cliente sur le nom de l'imprimante permettant l'impression.

- Configuration de machine :

Durant mon stage j'ai du préparer une nouvelle machine pour le responsable technique du pole informatique de l'entreprise, pour cela j'ai alors commencer par configurer les nouveaux composants sur la machine : (RAM , carte graphique).

Puis ensuite faire l'installation classique ainsi que les mises a jour de la nouvelle machine. J'ai également associer le NAS de l'entreprise via une procédure qui explique de A a Z comment faire la configuration d'un NAS synology, ce qui ma aider pour les autres configurations de PC demander dans l'entreprise.

Je suis également venu en aide a des petits dépannages dans l'entreprise pour un HDMI non fonctionel, un probleme de surchauffe jusqu'a la mise a jour de la machine etc ...

- Supervisions des machines :

En entreprise on a pu me demander de faire une installation via GPO d'un agent de stockage de document (zeendoc) pour cela j'ai trouver une command le permettant de le faire en mode silencieux sous windows :

```
.\\Setup_Agent_ZeenDoc_v2.0.26.exe /SILENT /LOG=log2.txt
```

J'ai pu également venir en aide a un autre stagiaire de l'entreprise avec la solution de supervision DATTO pour l'installation silencieuse d'un antivirus pour cela il avait entamer une programmation qui c'est averer inutile de part la solution deja présente sur le logiciel

En suivant les cours et la demande en SAE j'ai était amener a deployer une solution ansible pour la supervisions des microservices pour cela j'ai utilisé des playbooks en yaml me permettant ainsi de gerer les microservices ainsi que de les stocker dans un registry harbor relatif a la SAE je prendrais l'exemple du serveur nextcloud :

Nextcloud nous demande ainsi d'avoir un service de base de données , un service web et le service nextcloud. Pour cela dans le playbook j'ai indiquer un service maraidb un service PHPmyadmin permettant d'observer la base de données relative et le service nextcloud alloué avec son service web en NGINX. j'avais entamer la chose avec docker compose qui c'est averer utile lors du deploiement sous ansible.

le fichier relatif au playbooks ansible de nextcloud :

```
- include_tasks: install.yml
  tags: install_docker

- include_tasks: pull_images.yml
  tags: pull_nextcloud

- include_tasks: manage_nextcloud.yml
  tags:
    - create_nextcloud
    - start_nextcloud
    - stop_nextcloud

- include_tasks: info_containers.yml
  tags: nextcloud_info

- include_tasks: network_containers.yml
  tags: network_info
```

et sont fichier docker compose :

```
services:
  db:
    image: mariadb
    volumes:
      - /var/lib/docker/disk01:/var/lib/mysql
    environment:
      MYSQL_ROOT_PASSWORD: root
      MYSQL_USER: debian
      MYSQL_PASSWORD: root
      MYSQL_DATABASE: nextcloud_db
    ports:
      - 3306:3306
    networks:
      macvlan-docker-nextcloud:
        ipv4_address: 192.168.1.90

  phpmyadmin:
    image: phpmyadmin
    environment:
      PMA_HOST: 192.168.1.90
      MYSQL_ROOT_PASSWORD: root
    ports:
      - 3356:3356/tcp
    networks:
      macvlan-docker-nextcloud:
        ipv4_address: 192.168.1.83

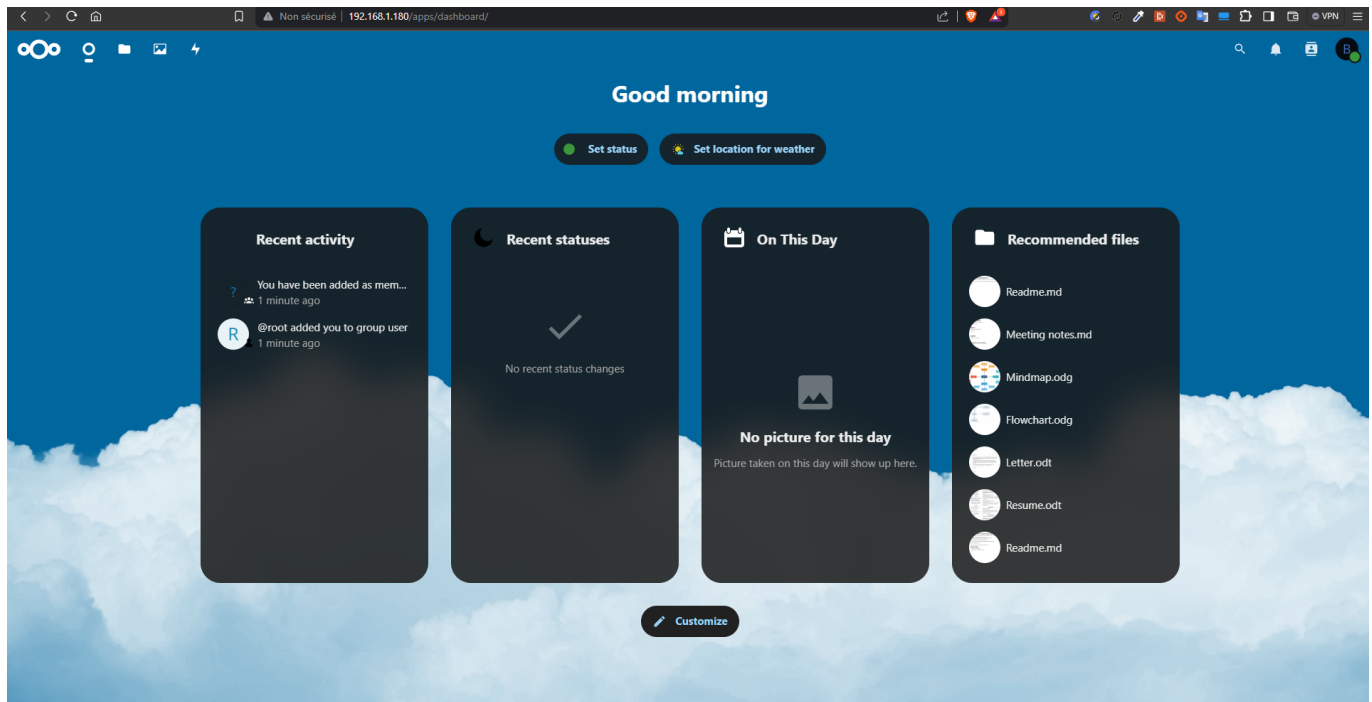
  nextcloud:
    image: nextcloud
    volumes:
```

```
- /var/lib/docker/nextcloud:/var/www/html
environment:
  MYSQL_HOST: 192.168.1.90
  MYSQL_DATABASE: nextcloud_db
  MYSQL_USER: debian
  MYSQL_PASSWORD: root
depends_on:
  - db
networks:
  macvlan-docker-nextcloud:
    ipv4_address: 192.168.1.18

nginx:
  image: nginx
  volumes:
    - /var/lib/docker/nginx:/etc/nginx/conf.d
    - /var/lib/docker/nginx/logs:/var/log/nginx
  ports:
    - 80:80
    - 443:443
  depends_on:
    - web
  networks:
    macvlan-docker-nextcloud:
      ipv4_address: 192.168.1.150

web:
  image: nginx # Remplacez ceci par l'image de votre service web
  volumes:
    - /var/lib/docker/web:/usr/share/nginx/html # Chemin exemple, adaptez selon
vos besoins
  networks:
    macvlan-docker-nextcloud:
      ipv4_address: 192.168.1.151

networks:
  macvlan-docker-nextcloud:
    external: true
```



Ce que j'en conclu :

[Retour au sommaire](#)

De la part les demonstrations réaliser dans le contenu ultérieur J'estime que je suis capable de configurer et de depanner le routage dynamique dans un réseau.

Egalement de configurer les ordinateurs clients et serveurs pour qu'ils puissent correspondre au besoin demander et faire la planification du réseau complet d'une infrastructure.

Avec la création d'une infrastructure a domicile cela ma également permis de développer concrètement les compétences relatif au réseau et télécommunications

J'en conclu aujourd'hui que je serais en capacité de d'administrer de concevoir et de gerer un réseau ainsi que les demandes clients sur des points précis.

6/20/2024, 5:39:32 PM © Helec Bastien. All Rights Reserved.
