

Mise en oeuvre de GNS3

Sommaire

1. Installation	2
2. GNS3 GUI	2
2.1. Les noeuds	2
2.1.1. VPCS	2
2.1.2. Conteneurs Docker	2
2.1.3. Machines virtuelles	6
2.1.4. Appareils (<i>appliance</i>)	8
webterm	8
Open vSwitch	8
Routeur Open source	10
2.2. Configuration d'un noeud	11
2.2.1. Réseau	11
2.2.2. Conteneur Docker	13
2.2.3. Type de console	15
2.3. Topologie d'une maquette	15
2.4. Capture de trames	17
2.5. Divers	18
3. Web-Ui	19
4. Liste des TPs	26

Thierry Vaira - <tvaira@free.fr> - version v0.1 - 15/01/2022 - tvaira.free.fr

Objectif : Mettre en oeuvre GNS3

GNS3 (*Graphical Network Simulator*) est un logiciel libre permettant l'émulation ou la simulation de réseaux informatiques.



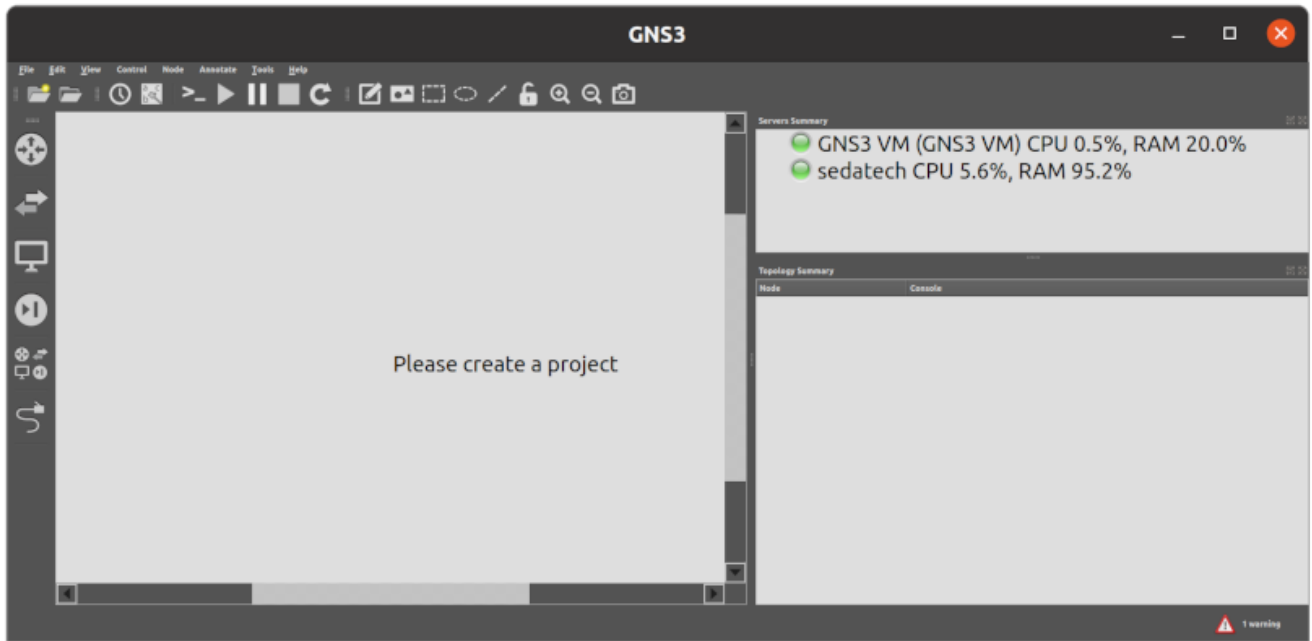
- Site officiel : <https://www.gns3.com/>
- Documentations : <https://docs.gns3.com/docs/>

1. Installation

Suivre la procédure donnée dans ce document : [installation-gns3.pdf](#)

Voir aussi : <https://docs.gns3.com/docs/getting-started/installation/linux/>

2. GNS3 GUI



2.1. Les noeuds

2.1.1. VPCS

VPCS (*Virtual PC Simulator*) est le simulateur fourni par défaut pour les noeuds. Ce programme, écrit par Paul Meng, permet de simuler un PC léger prenant en charge DHCP et ping. Il ne consomme que 2 Mo de RAM par instance et ne nécessite pas d'image supplémentaire.

Documentation : <https://docs.gns3.com/docs/emulators/vpcs/>

2.1.2. Conteneurs Docker

Il est possible d'utiliser des conteneurs Docker pour les noeuds dans GNS3 (<https://docs.gns3.com/docs/emulators/docker-support-in-gns3>).

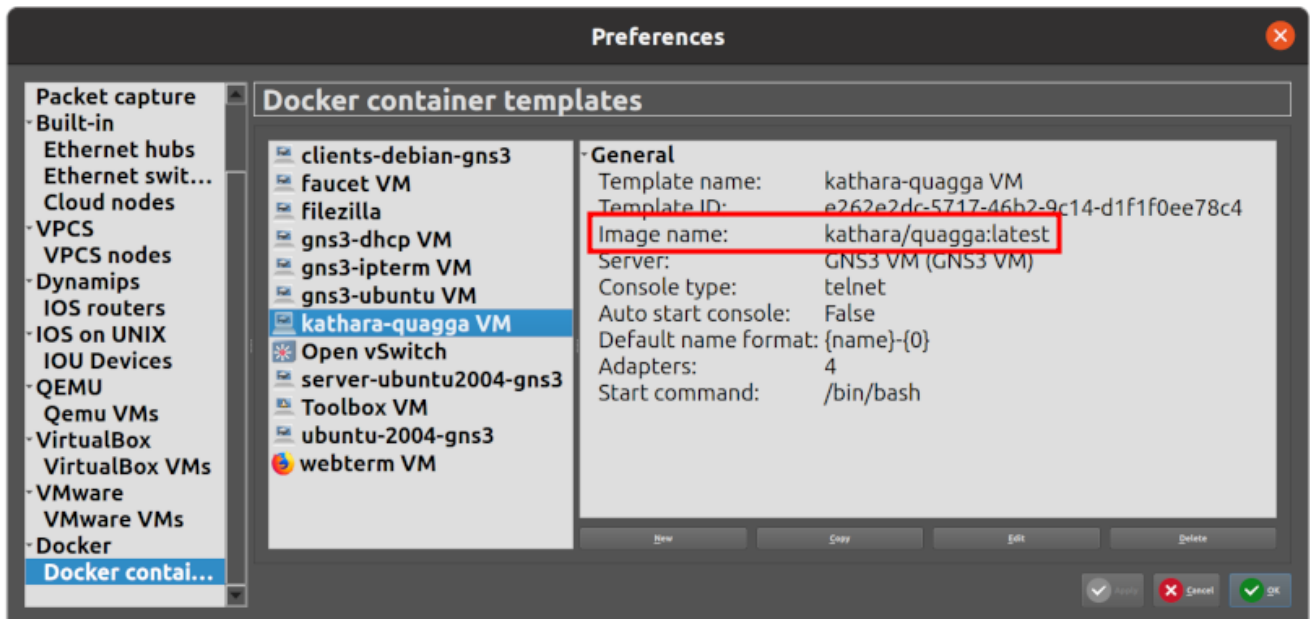


Des conteneurs Docker pour GNS3 sont disponibles ici :

- <https://hub.docker.com/u/gns3/>
- <https://hub.docker.com/r/tvaira/>

Dans les maquettes des TPS, on utilisera :

- pour les machines hôtes, on utilisera `tvaira/ubuntu-2004-gns3:latest`
- pour les machines clientes GUI, on utilisera `tvaira/clients-debian-gns3:latest`
- pour les machines serveurs, on utilisera `tvaira/server-ubuntu2004-gns3:latest`
- pour les routeurs, on utilisera `kathara/quagga:latest`

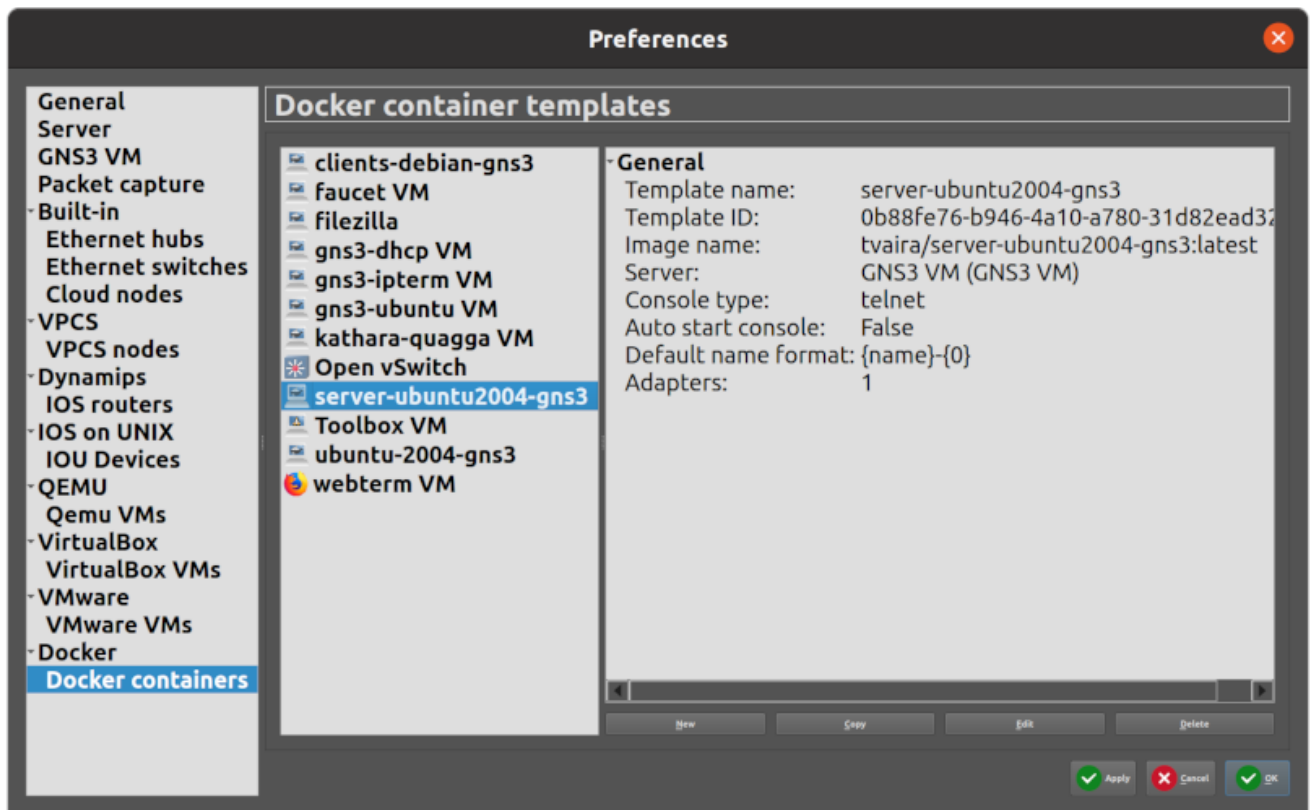


L'image `tvaira/clients-debian-gns3:latest` fournit Firefox et Filezilla et nécessite `vncviewer` pour s'y connecter à partir d'une console :

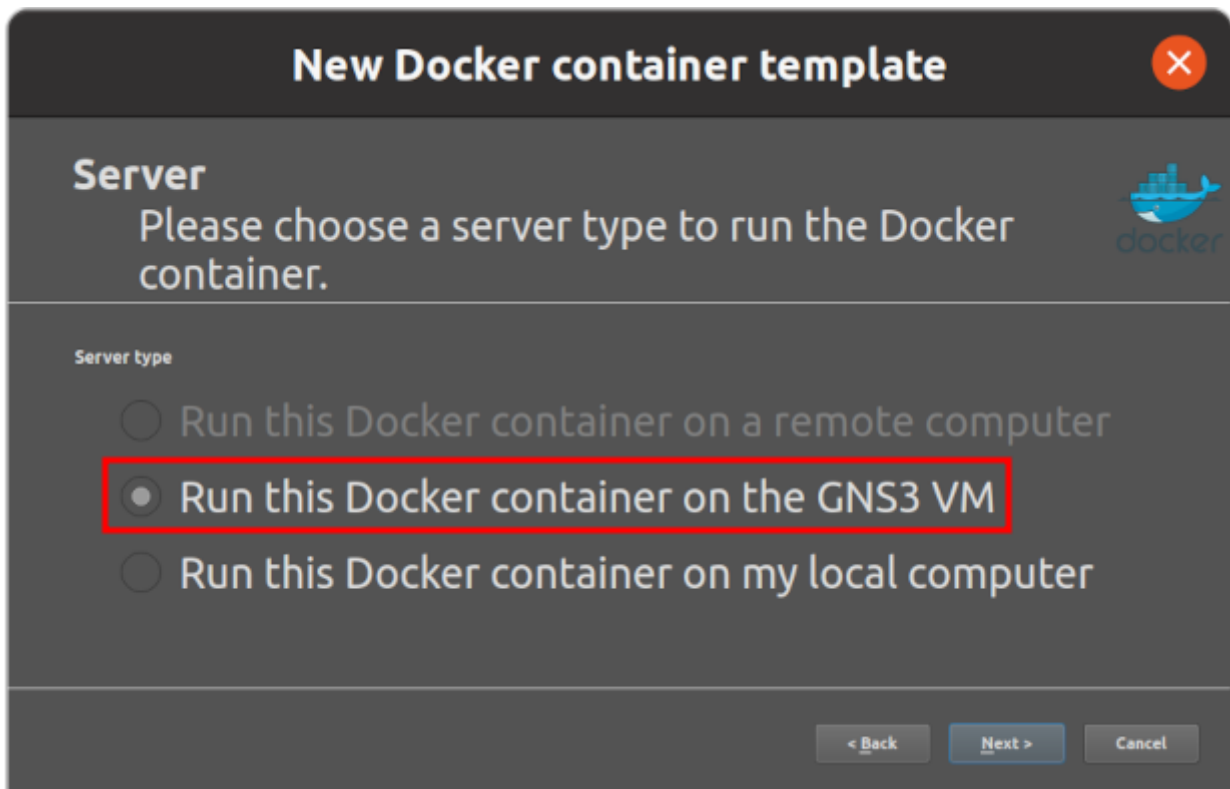


```
$ wget https://www.realvnc.com/download/file/viewer.files/VNC-Viewer-6.21.1109-Linux-x64.deb
$ sudo dpkg -i VNC-Viewer-6.21.1109-Linux-x64.deb
```

Pour ajouter une nouveau conteneur Docker, aller dans **Edit** → **Preferences** → **Docker containers** et cliquer sur **New** :



Puis :



Choisir entre une image existante :

New Docker container template

Docker Virtual Machine

Please choose a Docker virtual machine from the list or provide an image name on Docker hub.

☒ Existing image

☐ New image

Image list: tvaira/server-ubuntu2004-gns3:latest

< Back

Next >

Cancel

ou nouvelle image (<https://hub.docker.com/>) :

New Docker container template

Docker Virtual Machine

Please choose a Docker virtual machine from the list or provide an image name on Docker hub.

☐ Existing image

☒ New image

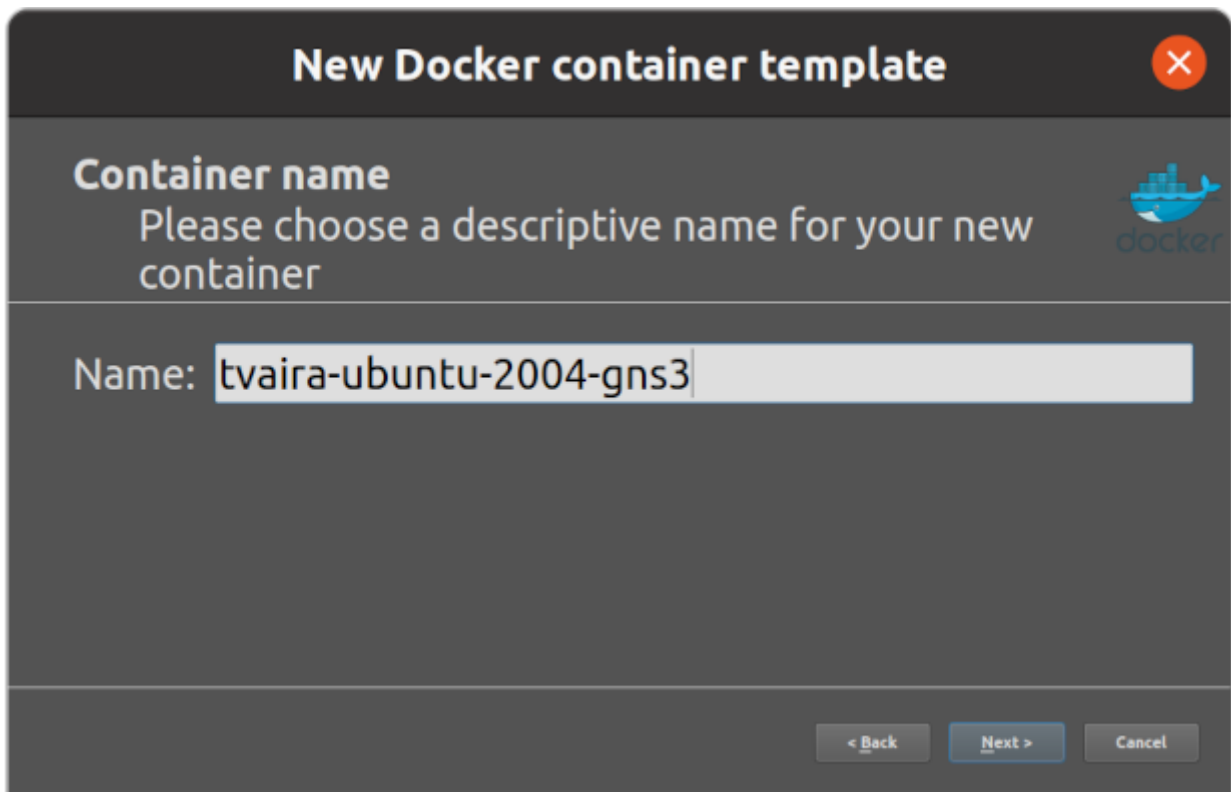
Image name: tvaira/ubuntu-2004-gns3:latest

< Back

Next >

Cancel

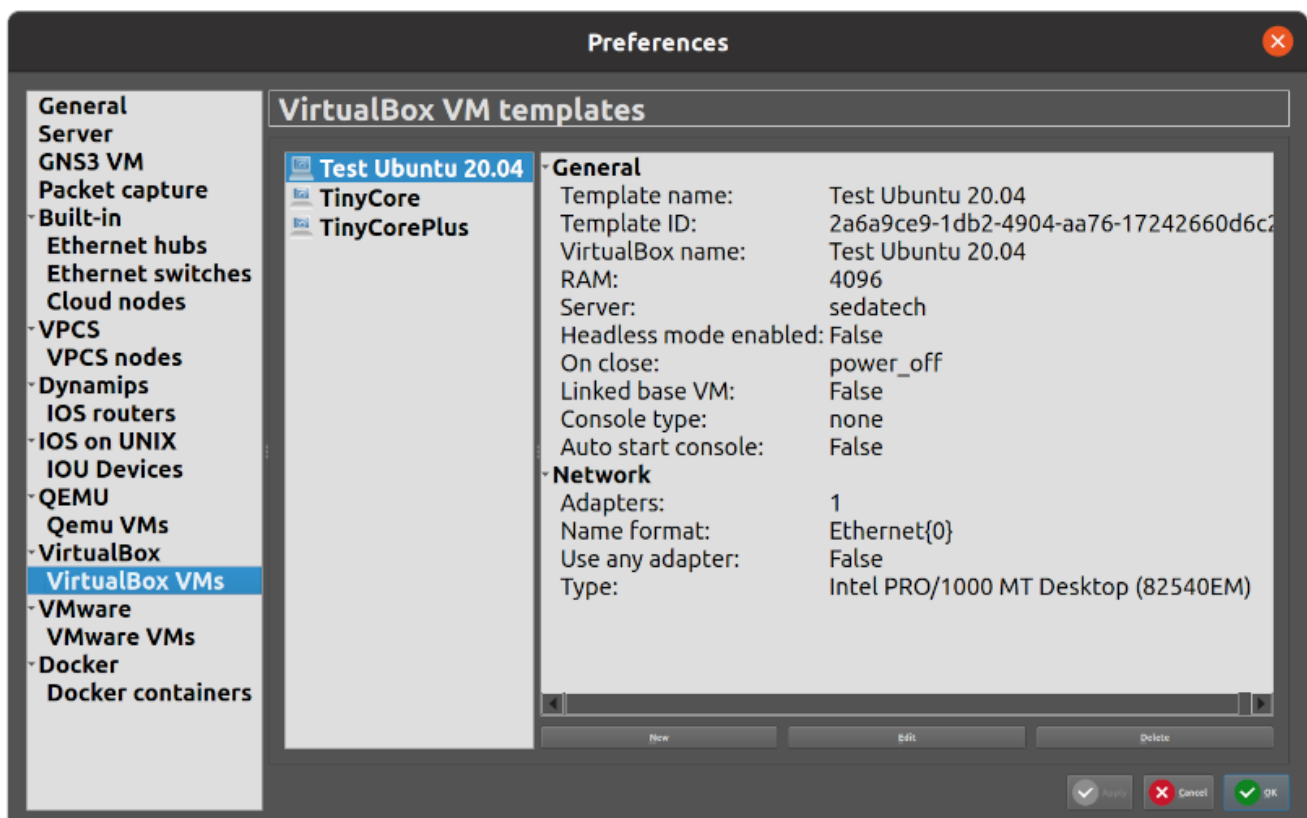
Terminer en donnant un nom :

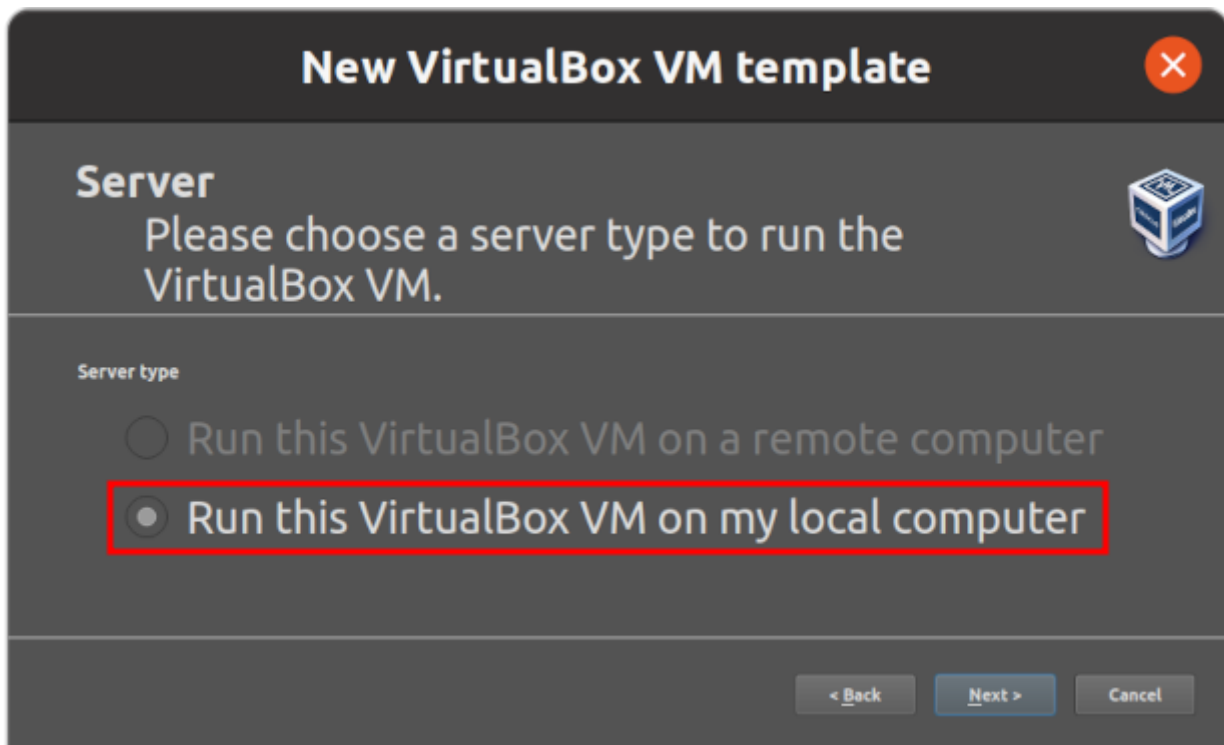


2.1.3. Machines virtuelles

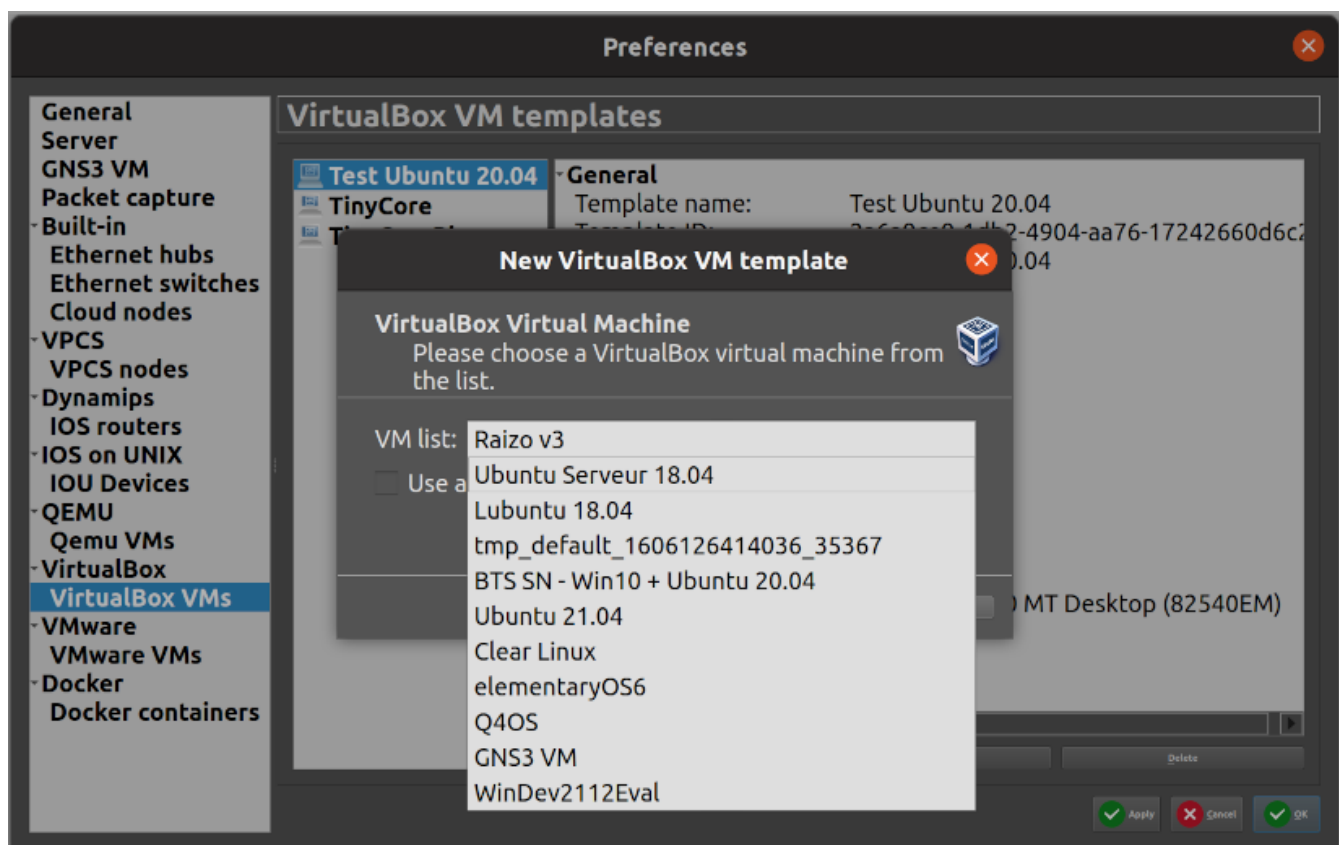
Il est possible d'utiliser des VM de VirtualBox ou VMWare.

Pour ajouter une VM de VirtualBox, aller dans **Edit** → **Preferences** → **VirtualBox VMs** et cliquer sur **New** :

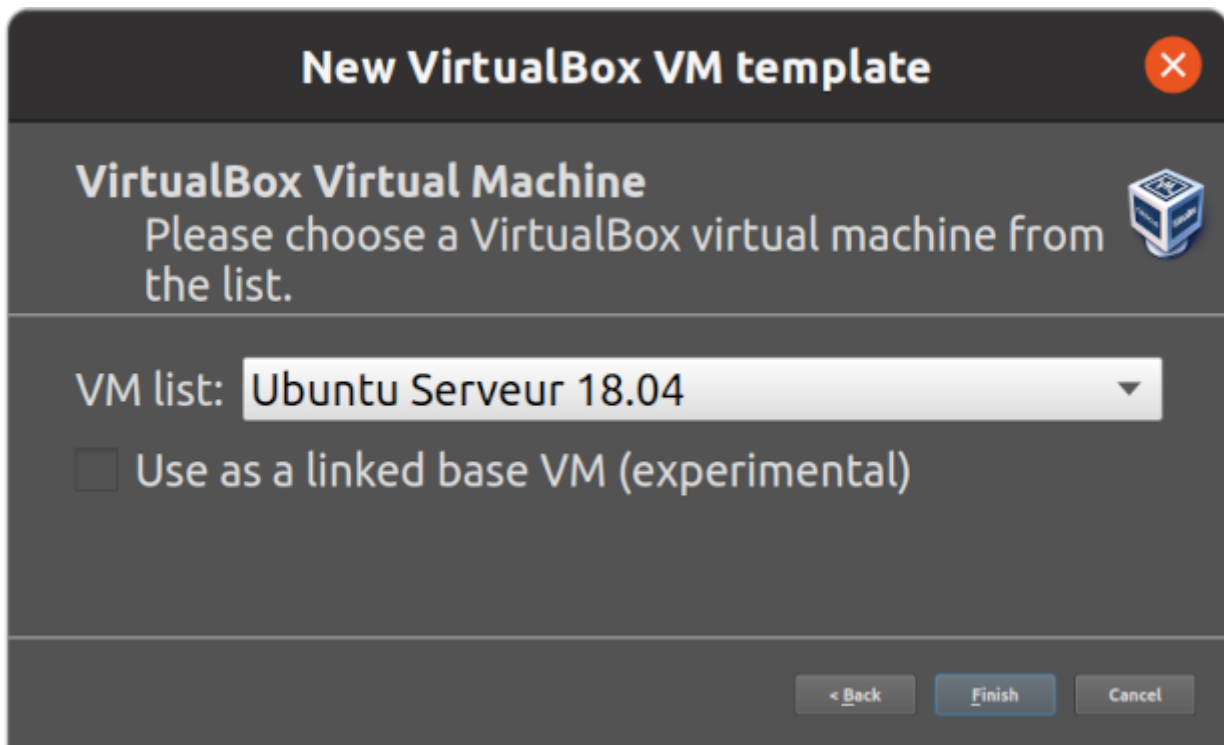




Choisir une VM dans la liste :



Terminer en cliquant sur **Finish** :



2.1.4. Appareils (appliance)

Les *appliances* sont disponibles au téléchargement sur le *marketplace* de GNS3 : <https://gns3.com/marketplace/appliances/>

Voir aussi : <https://github.com/GNS3/gns3-registry/tree/master/appliances>



Certains appareils nécessitent **vncviewer** pour s'y connecter à partir d'une console :

```
$ wget https://www.realvnc.com/download/file/viewer.files/VNC-Viewer-6.21.1109-Linux-x64.deb
$ sudo dpkg -i VNC-Viewer-6.21.1109-Linux-x64.deb
```

webterm

webterm est une boîte à outils réseau basée sur Debian. Il contient le navigateur web Firefox plus les utilitaires suivants : **net-tools**, **iproute2**, **ping**, **traceroute**, **curl**, **host**, **iperf3**, **mtr**, **socat**, **ssh**, **tcpdump**, ...

Télécharger l'*appliance* sur le *marketplace* de GNS3 : <https://gns3.com/marketplace/featured/webterm>



L'image **tvaira/clients-debian-gns3:latest**, basée sur *webterm*, ajoute **Filezilla**.

Open vSwitch

Open vSwitch est un commutateur virtuel multicouche.

Documentation : <http://openvswitch.org/support/>

Télécharger l'*appliance* sur le *marketplace* de GNS3 : <https://gns3.com/marketplace/appliances/open-vswitch>

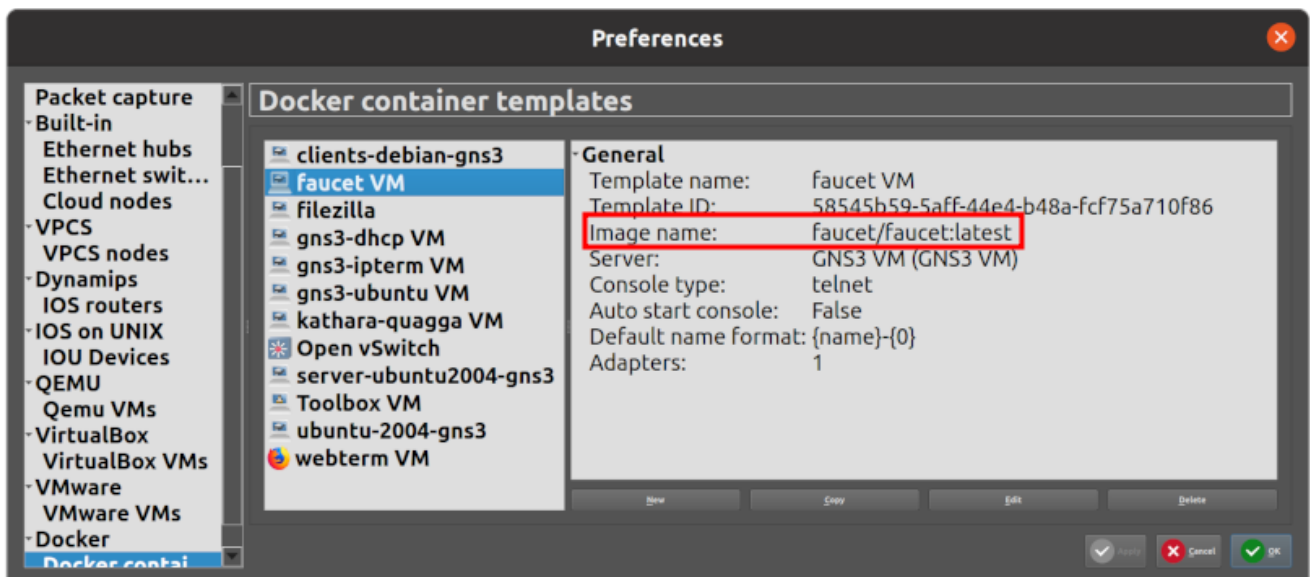
Open vSwitch supportant le protocole OpenFlow, il est possible d'ajouter une administration à distance des commutateurs via un contrôleur **Faucet**.



OpenFlow est un protocole réseau standard qui permet de réaliser une architecture Software-defined networking (SDN). Ce protocole est constitué d'instructions qui permet de programmer le plan de contrôle d'un équipement réseau.

Les contrôleurs comme Faucet utilisent OpenFlow pour contrôler la façon dont Open vSwitch (et d'autres commutateurs) traitent les trames et les paquets dans le réseau.

On ajoute un conteneur Docker avec l'image `faucet/faucet:latest` :



Puis on ajoute la commande à exécuter au démarrage :

Node properties

faucetVM-1 configuration

General settings | Advanced | Usage

Name: faucetVM-1

Start command: sh -c 'cd; faucet & sleep 10; ash -i -l'

Adapters: 1

Custom adapters: [Configure custom adapters](#)

Console type: telnet ☐ Auto start console

VNC console resolution: 1024x768

HTTP port in the container: 80

HTTP path: /

Environment variables:
(KEY=VALUE, one per line)

Network configuration [Edit](#)

[Reset](#) [Apply](#) [Cancel](#) [OK](#)

Routeur Open source

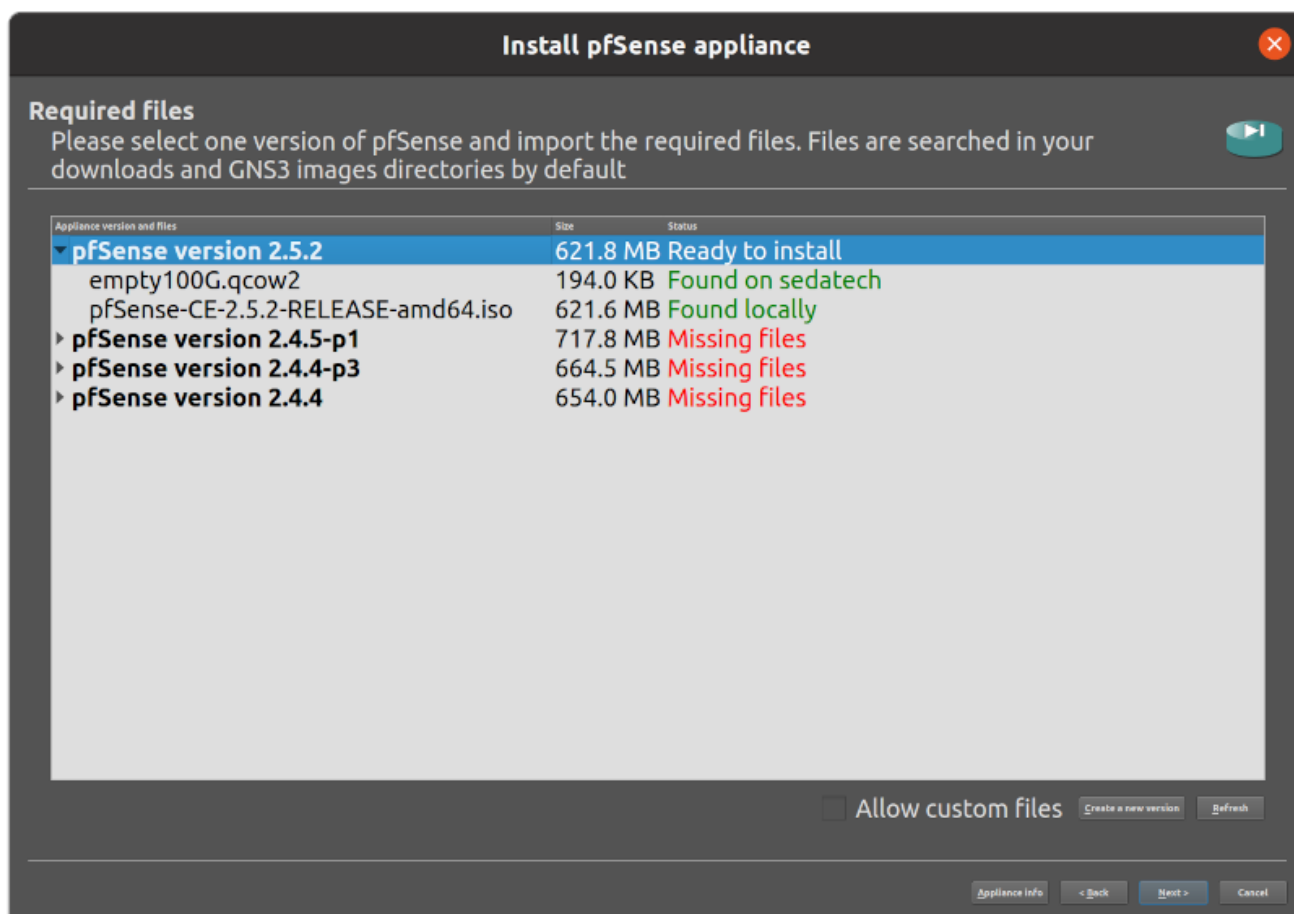
pfSense est un projet de routeur open source basé sur FreeBSD (une version de linux) prenant en charge le routage, le DHCP et le NAT. De plus, il possède un *firewall*.

Liens :

- <https://www.pfsense.org/about-pfsense/features.html>
- https://doc.pfsense.org/index.php/Main_Page

Télécharger l'*appliance* : <https://gns3.com/marketplace/featured/pfsense>

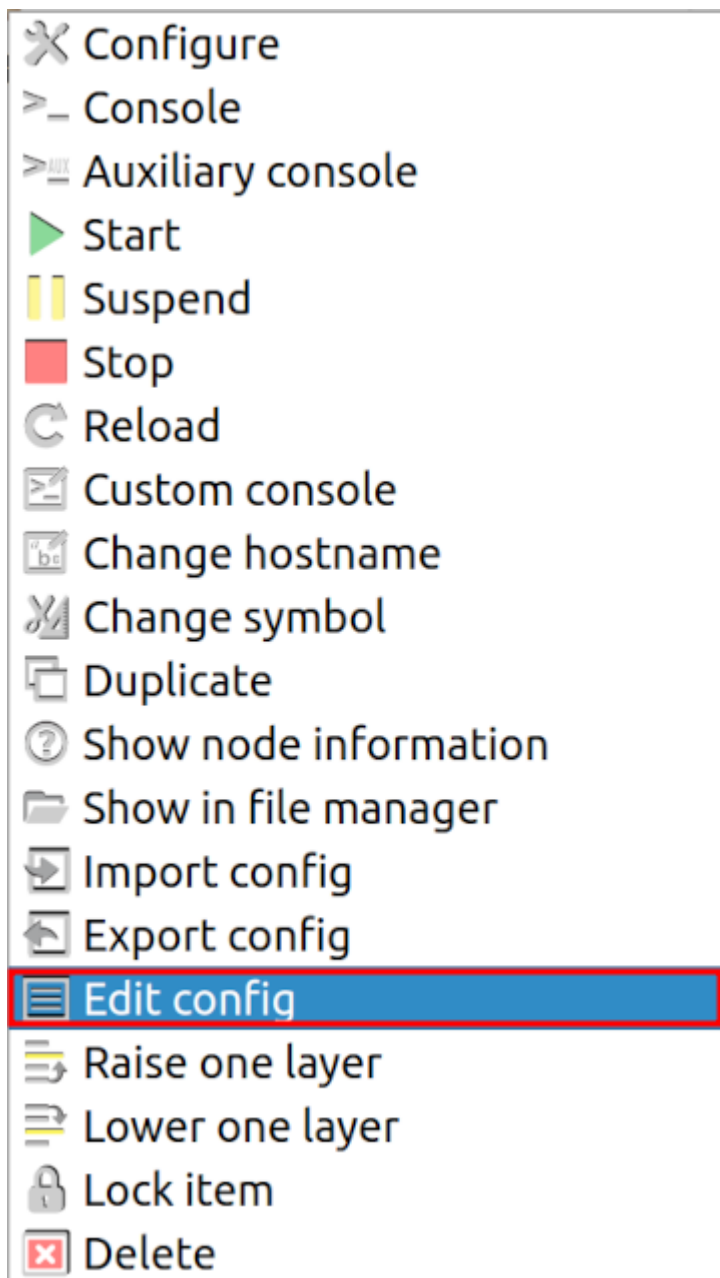
Puis l'installer dans GNS3 :



2.2. Configuration d'un noeud

2.2.1. Réseau

Il est habituel d'éditer la configuration réseau d'un noeud :



Puis par exemple pour l'interface `eth0` :

webterm-1 interfaces

```
#
# This is a sample network config uncomment lines to
# configure the network
#

# Static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.0.1
    netmask 255.255.255.0
#    gateway 192.168.0.1
    up echo nameserver 192.168.0.254 > /etc/resolv.conf

# DHCP config for eth0
# auto eth0
# iface eth0 inet dhcp
```

Refresh Cancel Save



Le mot clé `up` permet d'ajouter des commandes au démarrage de l'interface, par exemple : `up route add -net 172.16.32.0 netmask 255.255.255.0 gw 192.168.0.254`

2.2.2. Conteneur Docker

Pour les conteneurs Docker, on configure généralement :

- la commande à exécuter au démarrage (*start command*), par exemple : `sh -c "/bin/bash -i"` ou tout simplement `/bin/bash`
- le nombre d'interface réseau (*adapters*), 1 par défaut
- le type de console (`telnet`, `vnc` ou `http`)
- les variables d'environnement, par exemple : `PROMPT_COMMAND=history -a` (pour conserver l'historique des commandes saisies)

Node properties

serveur-http configuration

General settings | **Advanced** | Usage

Name:

Start command:

Adapters:

Custom adapters:

Console type: ☐ Auto start console

VNC console resolution:

HTTP port in the container:

HTTP path:

Environment variables:
(KEY=VALUE, one per line)

Network configuration

Dans l'onglet **Advanced** :

- ajouter des noms d'hôte dans le fichier `/etc/hosts`
- ajouter des répertoires persistants au conteneur Docker, par exemple :

Node properties

serveur-http configuration

General settings | **Advanced** | Usage

Extra hosts added to the `/etc/hosts` file.
(hostname:IP one per line)

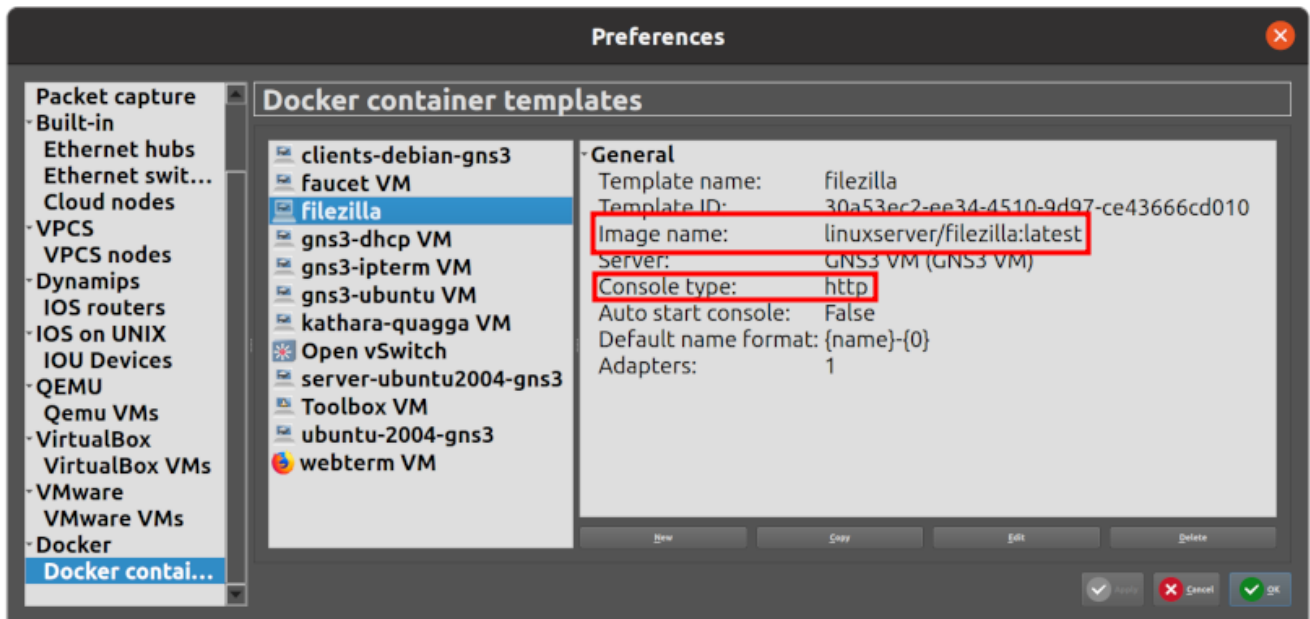
Additional directories to make persistent that are not included in the image VOLUMES config. One directory per line.

2.2.3. Type de console

On utilise habituellement une console de type **telnet** pour opérer avec un noeud. Si le noeud fournit une GUI, il faudra utiliser **vncviewer** pour s'y connecter à partir d'une console :

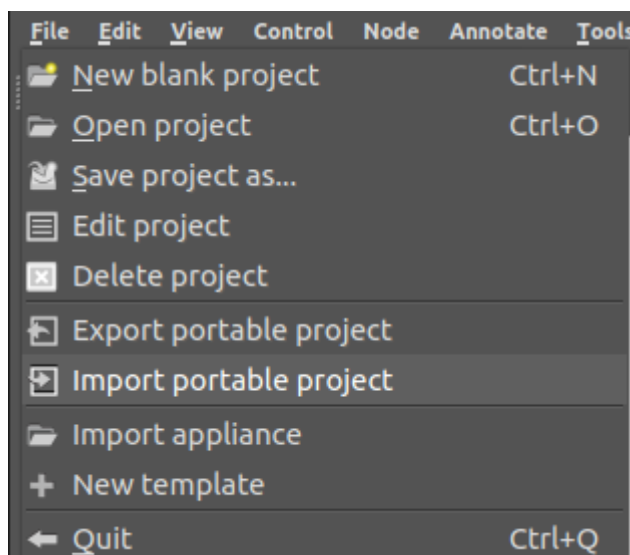
```
$ wget https://www.realvnc.com/download/file/viewer.files/VNC-Viewer-6.21.1109-Linux-x64.deb
$ sudo dpkg -i VNC-Viewer-6.21.1109-Linux-x64.deb
```

Certains conteneurs utilise **http** :

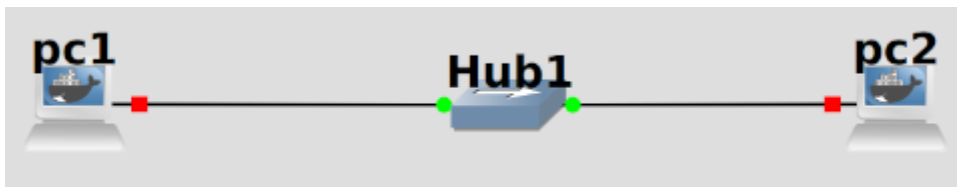


2.3. Topologie d'une maquette

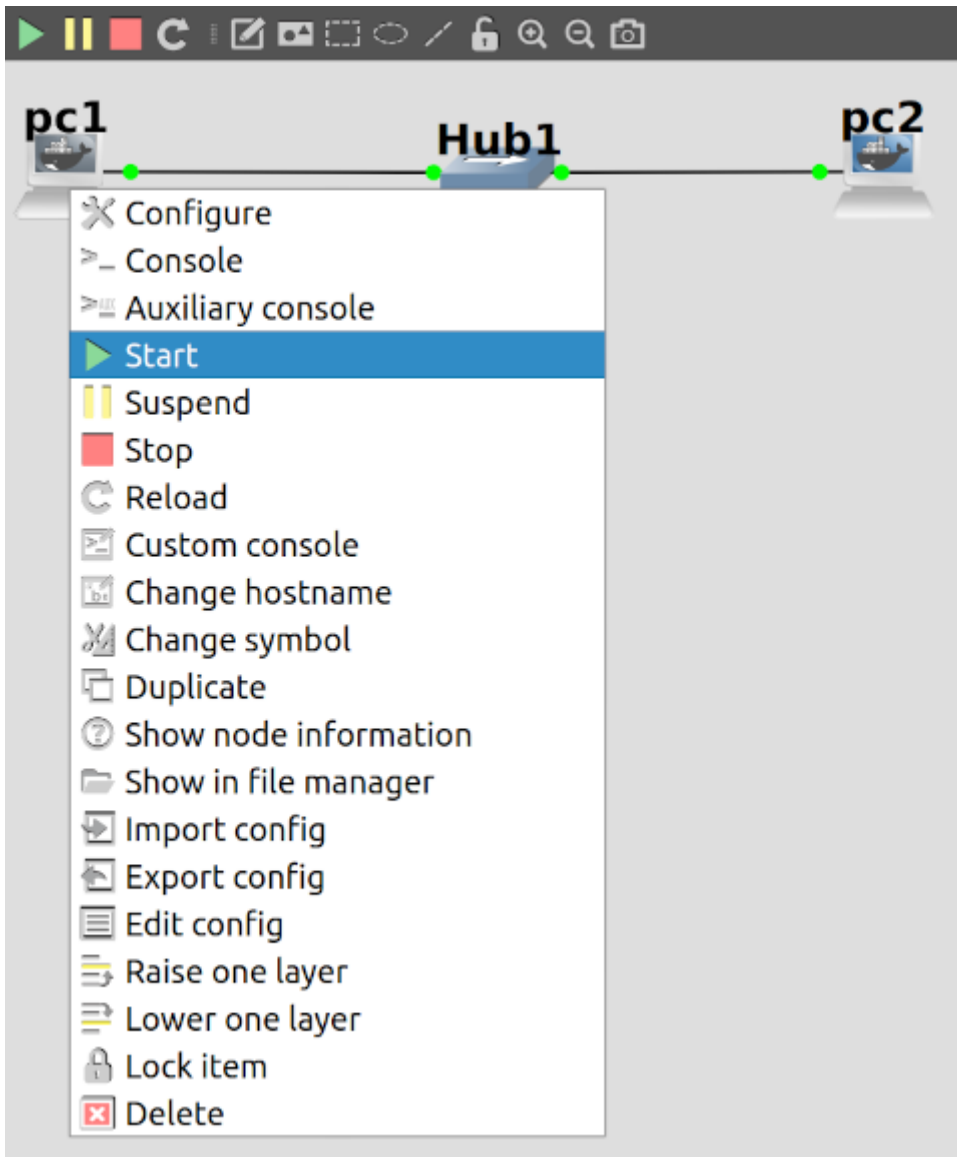
Créer un nouveau projet ou télécharger et importer dans GNS3 une maquette fournie pour un TP :



Exemple de maquette basique :



Il faut démarrer les machines de la maquette (toutes avec la flèche verte de la barre d'outils ou individuellement avec le menu contextuel clic droit d'une machine):

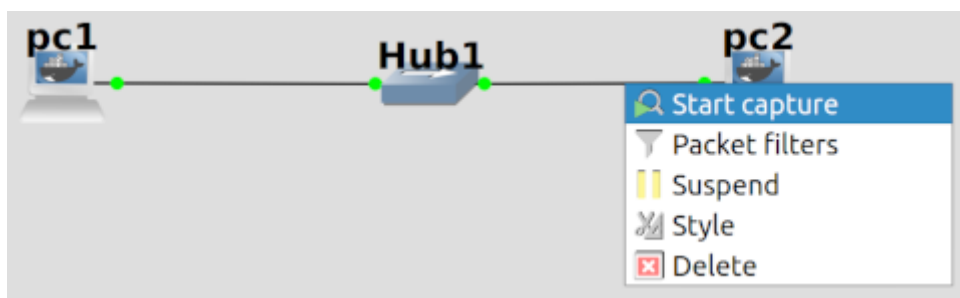


L'ensemble des noeuds :

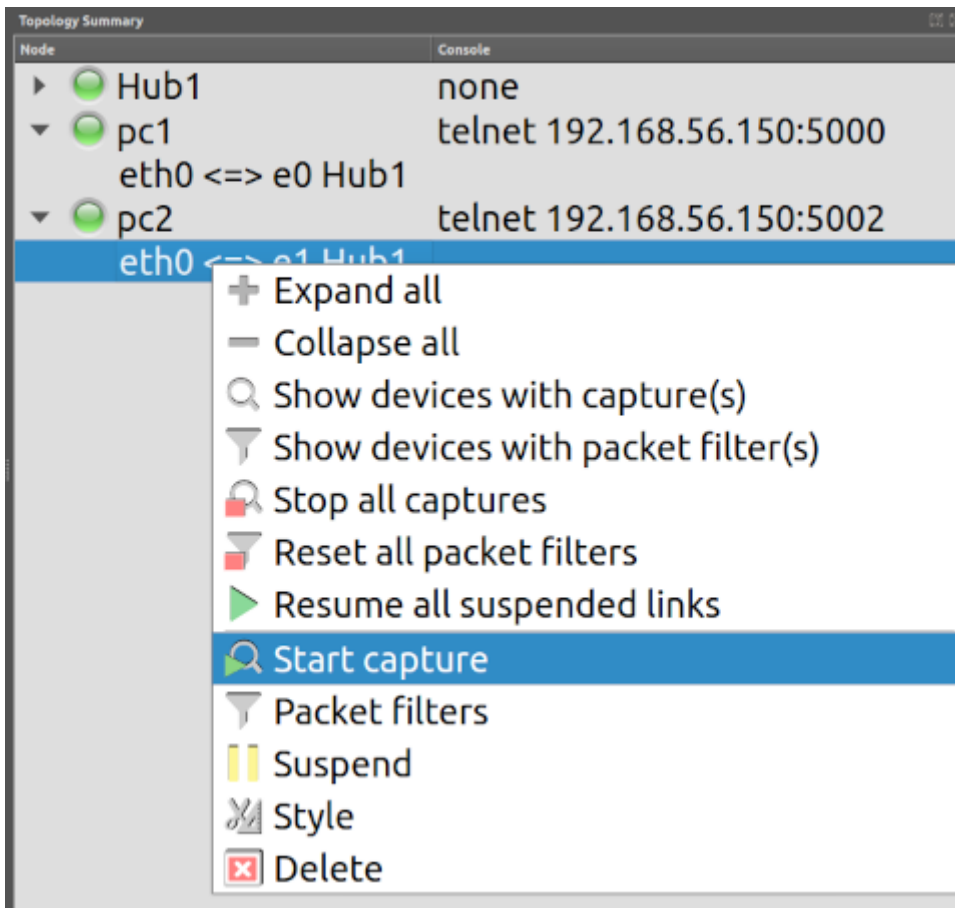
Servers Summary	
<ul style="list-style-type: none"> GNS3 VM (GNS3 VM) CPU 0.5%, RAM 22.7% <ul style="list-style-type: none"> Hub1 pc1 pc2 sedatech CPU 5.8%, RAM 92.8% 	
Topology Summary	
Node	Console
<ul style="list-style-type: none"> Hub1 <ul style="list-style-type: none"> e0 <=> eth0 pc1 e1 <=> eth0 pc2 pc1 <ul style="list-style-type: none"> eth0 <=> e0 Hub1 pc2 <ul style="list-style-type: none"> eth0 <=> e1 Hub1 	<ul style="list-style-type: none"> none telnet 192.168.56.150:5000 telnet 192.168.56.150:5002

2.4. Capture de trames

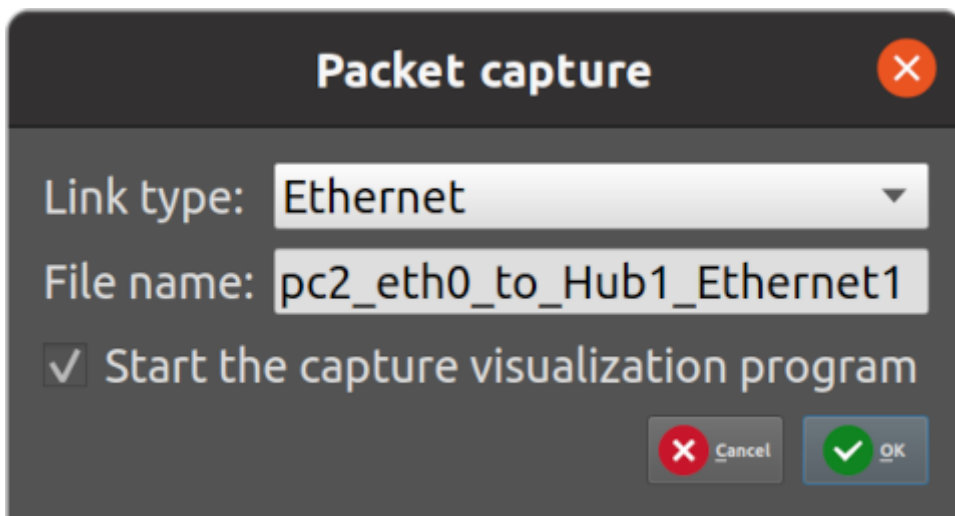
Il est possible de démarrer une (ou plusieurs) capture **Wireshark** sur un lien de la maquette :



ou :



Puis :



2.5. Divers

Au démarrage d'une machine, il est possible que les liens symboliques (déterminant les noms par défaut de certaines commandes par exemple) soient manquants. Une méthode simple pour corriger les alternatives cassées est d'utiliser :

```
# yes '0' | update-alternatives --force --all
```

3. Web-Ui

L'accès à GNS3 peut aussi se faire par une interface Web : <http://<adresse-ip>/>



Utiliser l'adresse IP de l'interface réseau avec un Accès par pont.

Servers

ID	Name	Location	Host	Port	Actions
1	local	bundled	192.168.52.29	80	→ 🗑️

[Add server](#)

GNS3 Web UI © 2020 - v2.2.29

Créer et ouvrir un projet pour accéder à l'interface de GNS3

Projects

[Go to system status](#) [Go to preferences](#) [Add blank project](#) [Import project](#)

Exemple de topologie réseau :

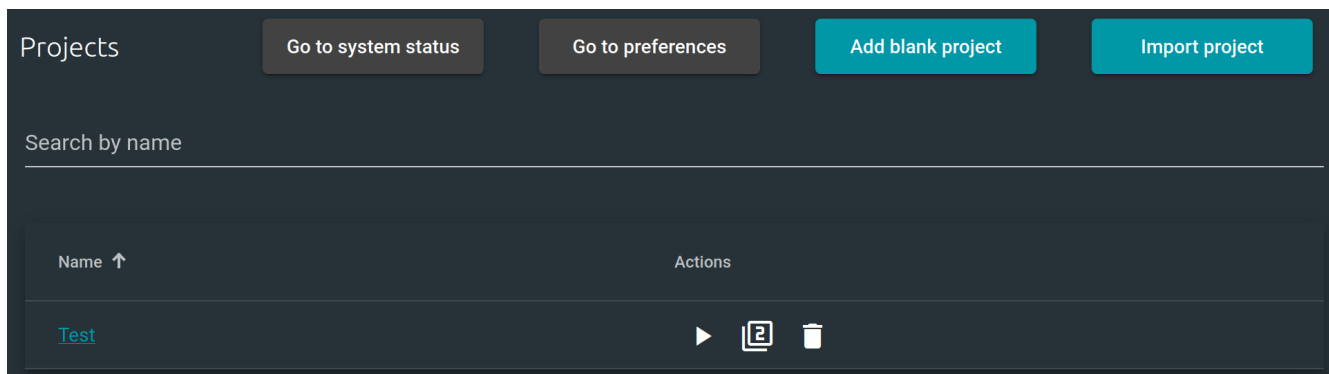
Map topology

Filter nodes

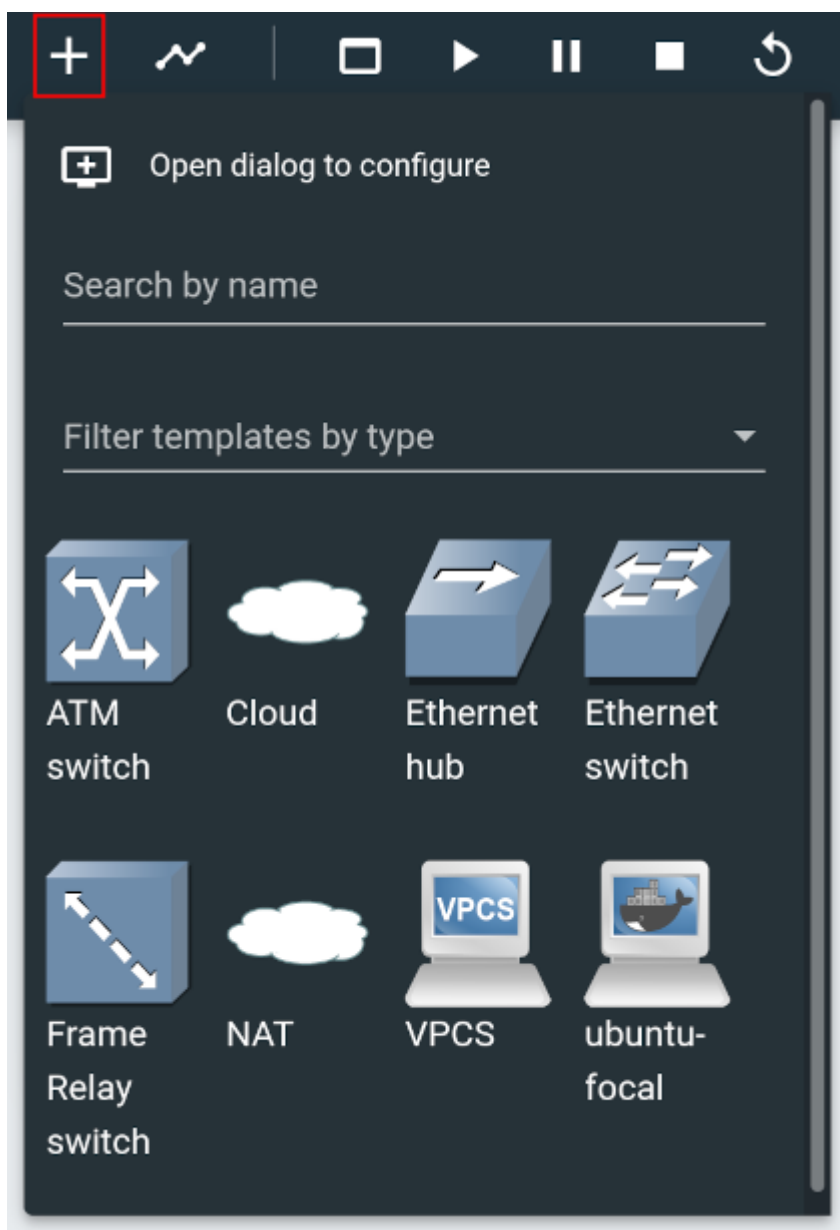
sort by name ascending

Node	IP Address
Hub1	none
PC1	telnet 192.168.52.29:5000
PC2	telnet 192.168.52.29:5002
ubuntu-focal-1	telnet 192.168.52.29:5006

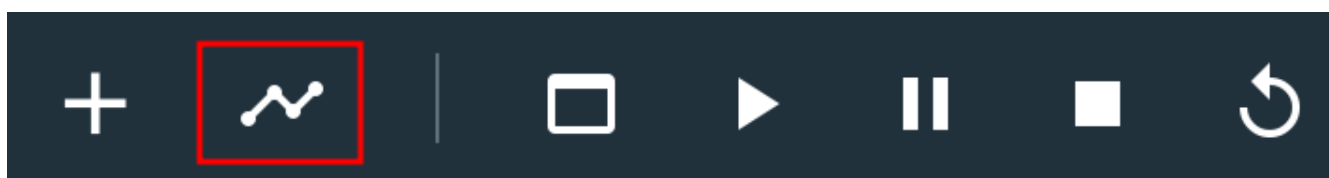
Dans l'interface Web-Ui, ouvrir un projet :



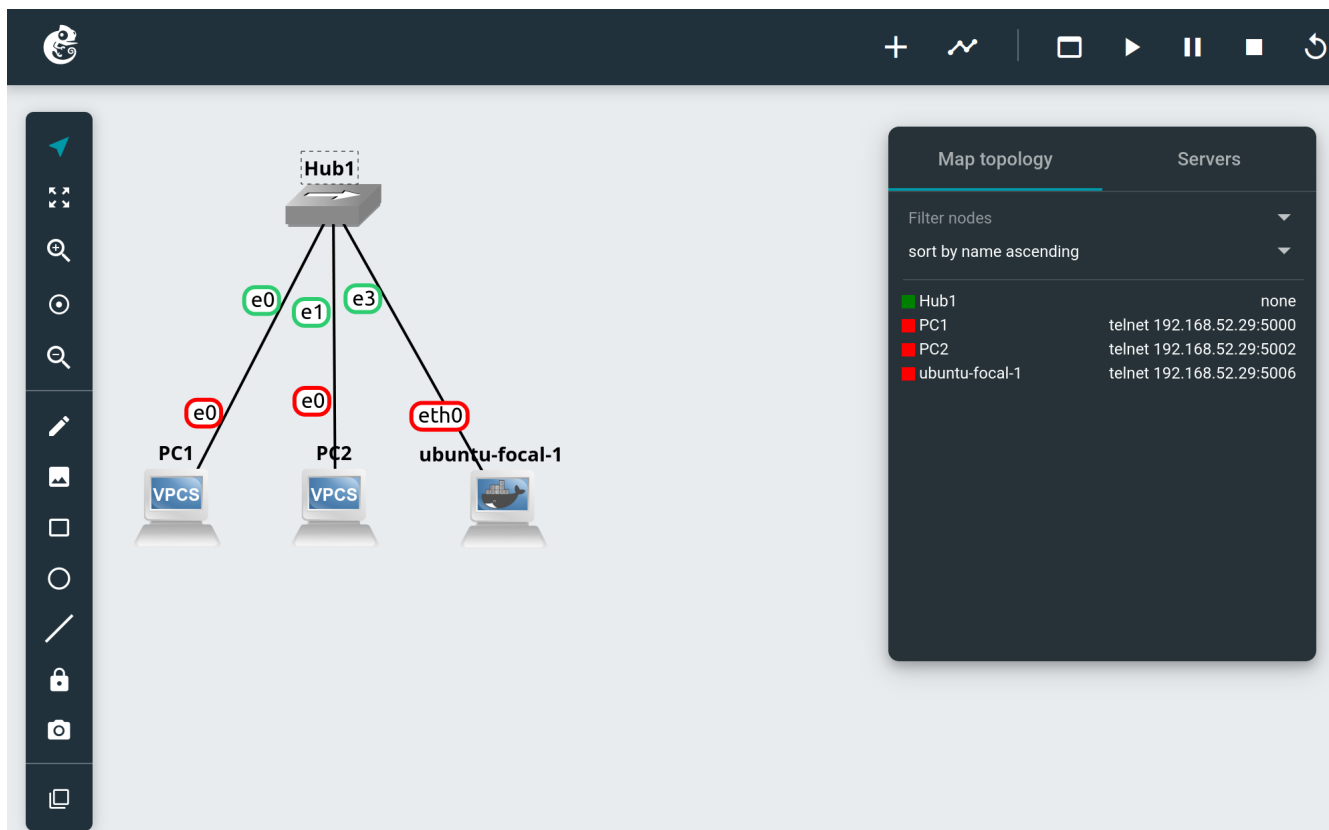
Puis ajouter des noeuds :



Et des liens :















Pour obtenir le réseau suivant par exemple :



Sélectionner un noeud et faire un clic droit :

ubuntu-focal-1



-  Show node information
-  Configure
-  Start
- HTTP Web console
- HTTP Web console in new tab
-  Console
-  Change hostname
-  Change symbol
-  Duplicate
-  Move layer up
-  Move layer down
-  Bring to front
-  Lock item
-  Delete

- **Start** : démarrer le noeud puis,
- **Web console in new tab** : ouvrir une console dans un onglet

```

ubuntu-focal-1 console is now available... Press RETURN to get started.
root@ubuntu-focal-1:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::24a1:c6ff:fe13:205b prefixlen 64 scopeid 0x20<link>
    ether 26:a1:c6:13:20:5b txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 7 bytes 586 (586.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

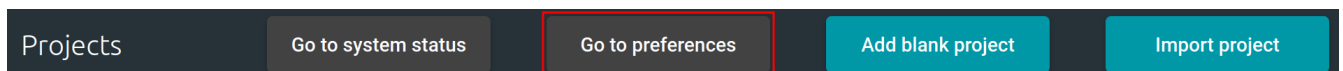
root@ubuntu-focal-1:/# ifconfig eth0 192.168.1.3/24
root@ubuntu-focal-1:/# ping 192.168.1.1 -c 1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.939 ms

--- 192.168.1.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.939/0.939/0.939/0.000 ms
root@ubuntu-focal-1:/# █

```

- Pour les images Docker :

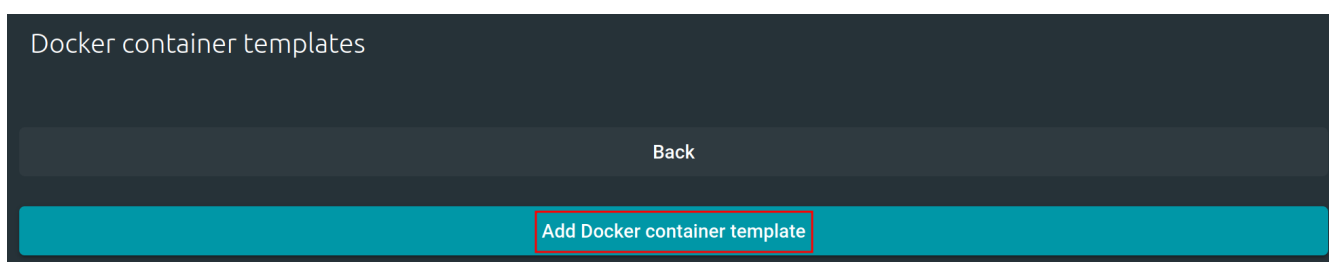
Dans l'interface Web-Ui, aller dans Préférences :



Sélectionner Docker :



Puis ajouter un nouveau *template* :



Paramétrer le nouveau conteneur :

New Docker container template

1 Server type

☒ Run this Docker container locally

☐ Run this Docker container on the GNS3 VM

Docker Virtual Machine

3 Container name

Network adapters

Start command

Console type

7 Environment

Cancel

Add template

2 Docker Virtual Machine

☐ Existing image

Image name

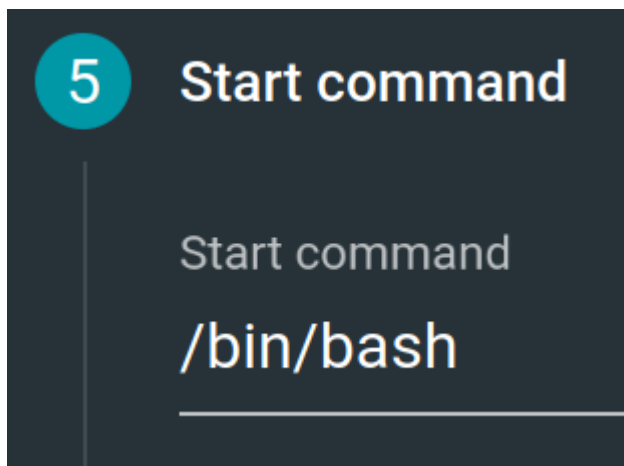
gns3/ubuntu:focal

☒ New image

3 Container name

Container name

ubuntu-focal



Finaliser en cliquant sur **Add template**.

4. Liste des TPs

Les TPs sont disponibles depuis **Github Classroom** :

- <https://github.com/btssn-lasalle84/tp-reseau-interface>
- <https://github.com/btssn-lasalle84/tp-reseau-routage-statique>
- <https://github.com/btssn-lasalle84/tp-reseau-ipv4>
- <https://github.com/btssn-lasalle84/tp-reseau-services>
- <https://github.com/btssn-lasalle84/tp-reseau-parefeu>
- <https://github.com/btssn-lasalle84/tp-reseau-vlan>
- <https://github.com/btssn-lasalle84/tp-reseau-routage-dynamique>
- <https://github.com/btssn-lasalle84/tp-reseau-tcp-udp>
- <https://github.com/btssn-lasalle84/tp-reseau-ipv6>
- <https://github.com/btssn-lasalle84/tp-reseau-windows>
- <https://github.com/btssn-lasalle84/tp-admin-http>

Thierry Vaira - <tvaira@free.fr> - version v0.1 - 15/01/2022 - tvaira.free.fr