

Le but de ce TP est de découvrir le logiciel de simulation de réseau *GNS3* afin de construire et d'administrer des réseaux locaux

Exercice 1. Prise en main de GNS3

GNS3 est un logiciel qui sert à concevoir et à construire des réseaux. Il vous permet de manipuler différents équipements (hub, switch, routeur, pc ...), de les connecter, de les paramétrer et de visualiser les différents échanges.

1.1 Lancer GNS3 sur votre machine, créer un nouveau projet. Vous obtenez la fenêtre suivante :

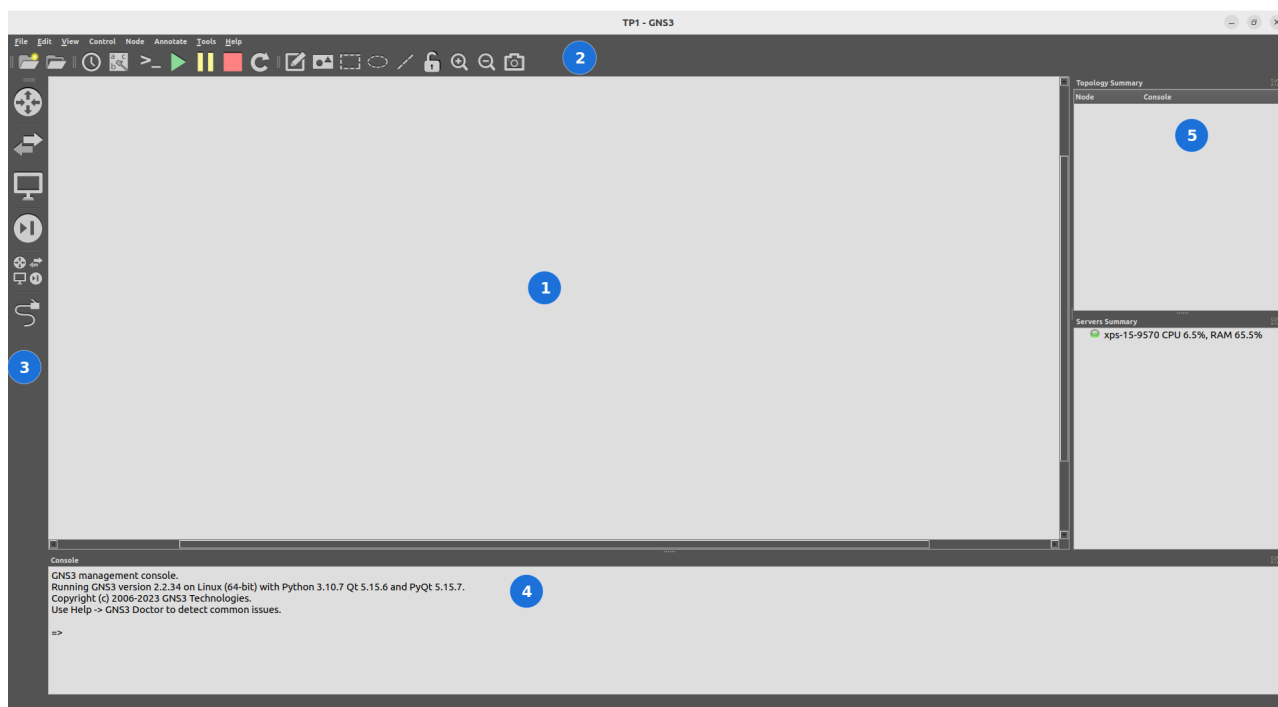


FIGURE 1 – GNS3

1. Fenêtre principale : Sur cet espace de travail, vous allez pouvoir construire votre réseau.
2. La barre d'icônes permet d'effectuer les tâches les plus courantes, dont le lancement et l'arrêt de la simulation
3. Le menu équipement : Pour ajouter un équipement, il suffit de le faire glisser de ce menu vers la fenêtre principale. La dernière icône permet de relier vos différents éléments

Réseau 1A (TP n°1)

4. Console : pour administrer en ligne de commande
5. Liste des différents équipements de votre projet et leur état : rouge si le matériel est inactif, vert lorsqu'il est actif

Votre projet peut être sauvegardé pour être repris plus tard.

1.2 Créer un nouveau projet, et sauvegarder-le.

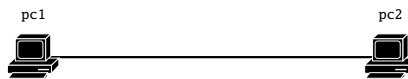


FIGURE 2 – Réseau de deux pc reliés directement

1.3 Créer un réseau avec deux machines (VPCS), et relier-les. Vous pouvez noter que lorsque vous avez cliqué sur une machine, le menu Ethernet0 est apparu. C'est l'**interface** de votre machine, c'est par cet élément du pc que les communications réseau entreront et sortiront. Une machine peut avoir plusieurs interfaces réseau.

1.4 Démarrer le réseau (via le bouton triangle vert dans la barre des tâches)

Vous avez construit et démarré votre premier réseau.

Exercice 2. Communication entre deux machines

On veut tester si les deux machines arrivent à communiquer entre elles

Il est nécessaire d'identifier les machines sur le réseau pour pouvoir leur envoyer un message : c'est le rôle d'une *adresse IP*. C'est l'interface d'une machine qui est associée à cet identifiant. On dira par exemple que l'adresse 192.168.0.1 est associée à l'interface e0 du pc1. Par abus de langage lorsqu'un pc a une seule interface, on pourra parler d'une adresse IP d'un pc.

Pour plus d'explications sur ces adresses IP consultez le TD1.

2.1 Ouvrir une console sur une machine (clic droit sur un pc, puis console). Avec cette console vous allez pouvoir administrer un pc. Vous pouvez entrer ? pour avoir la liste des commandes disponibles

2.2 Pour consulter l'adresse IP actuelle, entrer la commande

```
| show ip
```

Vous obtenez une liste de champs et de leur valeur associée

Réseau 1A (TP n°1)

```
NAME      : PC4[1]
IP/MASK    : 0.0.0.0/0
GATEWAY    : 0.0.0.0
DNS        :
MAC        : 00:50:79:66:68:03
LPORT     : 10014
RHOST:PORT : 127.0.0.1:10015
MTU        : 1500
```

On peut voir que l'adresse IP actuelle est 0.0.0.0/0, c'est-à-dire qu'elle n'est pas encore attribuée. Nous nous intéresserons aux autres champs plus tard.

2.3 Nous allons affecter une adresse IP à chaque interface de machines avec la commande suivante :

```
ip 192.168.0.1/24
```

Entrer cette commande sur pc1

Affecter 192.168.0.2/24 à pc2. Vérifier que l'adresse IP a bien été modifiée.

2.4 Vos deux machines ont donc désormais une adresse. Il faut tester si les deux machines peuvent communiquer. C'est le rôle de la commande `ping`. Cette commande prend en argument l'adresse de la machine que l'on souhaite joindre. Par exemple pour tester que la connexion entre pc1 pc2 fonctionne, on entre (sur pc1) :

```
ping 192.168.0.2
```

Vous obtenez le résultat suivant

```
84 bytes from 192.168.0.2 icmp_seq=1 ttl=64 time=0.811 ms
84 bytes from 192.168.0.2 icmp_seq=2 ttl=64 time=0.720 ms
...
```

Lorsque vous lancez la commande, plusieurs ping successifs sont lancés. Chaque ligne correspond à la réponse à un ping. On peut voir sur cet exemple que pc1 arrive à communiquer avec pc2. Chaque réponse précise

- la quantité de données reçues,
- l'adresse IP à l'origine de cette réponse,
- le numéro identifiant le ping,
- et le temps entre l'émission et la réception de la réponse

Réseau 1A (TP n°1)

(nous verrons la signification du ttl plus tard)

2.5 Tester le ping de pc2 vers pc1

Une fois votre réseau stoppé, les configurations des machines sont perdues. Il est donc nécessaire de savoir les sauvegarder et les restaurer. Il suffit dans un terminal d'une machine de saisir

```
| save
```

pour sauvegarder la configuration d'une machine (dans un fichier `startup.vpc` par défaut)

et

```
| load
```

pour recharger une configuration

2.6 Sauvegarder votre configuration pour pc1 et pc2

Exercice 3. Trois machines

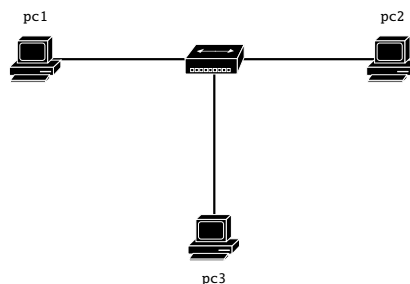


FIGURE 3 – Réseau de trois pc reliés par un hub

Vous avez à présent un réseau de deux machines, avec chacune une adresse IP, ce qui leur permet de communiquer.

Comment faire pour avoir un réseau de trois machines (ou plus) ?

Nous allons créer un réseau en étoile, c'est-à-dire qu'un équipement central sera relié à tous les nœuds du réseau. Cet équipement supplémentaire peut être un *hub* ou un *switch*. Cette topologie est la plus courante.

3.1 Arrêter votre réseau, et supprimer la liaison entre les deux machines. Ajoutez un troisième pc. Ajouter un **hub** au centre et relier chaque pc à celui-ci.

3.2 Démarrer le réseau. Recharger la configuration pour pc1 et pc2 (Si nécessaire). Attribuer une adresse IP à pc1, par exemple 192.168.0.3/24 (pensez à sauvegarder la configuration de pc3)

3.3 Vérifier avec un ping que toutes les machines arrivent à communiquer

Réseau 1A (TP n°1)

Exercice 4. Écoute des communications

Vous savez maintenant configurer un réseau local avec plusieurs machines. Que se passe-t-il sur une connexion lorsque deux machines communiquent, et comment un hub arrive à distribuer les échanges.

Il est possible de répondre à ces questions en écoutant les échanges sur une connexion grâce au logiciel *wireshark*.

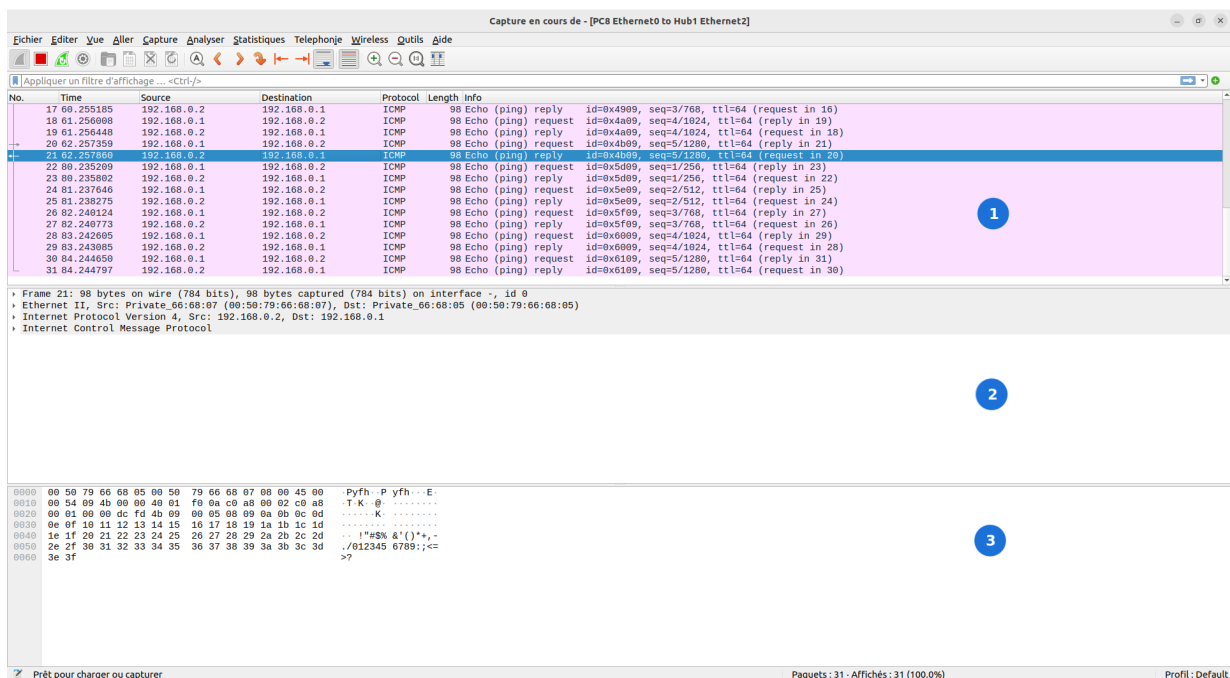


FIGURE 4 – Wireshark, outils de capture de paquets

Wireshark se décompose en trois fenêtres :

1. La liste des différents échanges. Chaque ligne représente une communication entre machines, qu'on appellera paquet dans la suite
2. Cette fenêtre affiche de manière détaillée le paquet sélectionné dans la fenêtre 1. Le paquet est présenté en différenciant les différentes couches, et avec un système de champs/valeur. Grâce à cette vue *wireshark* permet d'interpréter plus facilement un paquet
3. Cette vue présente le paquet sous forme brute, où les octets sont affichés en représentation hexadécimale

4.1 Lancer une capture (clic droit sur une liaison) sur la liaison entre le hub et pc2. Lancer le ping depuis pc1 vers pc2. Comparer le nombre de ping, avec le nombre de paquets capturés. Pouvez-vous expliquer cette différence ?

Réseau 1A (TP n°1)

4.2 Refaire la question précédente (ping de pc1 vers pc2), mais en lançant la capture sur la liaison entre pc3 et le hub. Que constatez-vous ?

4.3 Expliquer comment un hub fonctionne pour distribuer les paquets.

Exercice 5. Switch

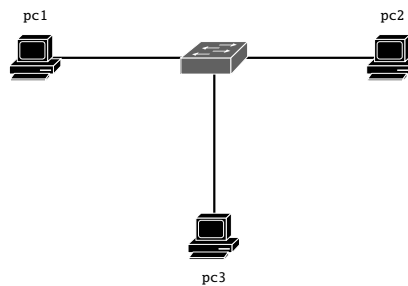


FIGURE 5 – Réseau de trois pc reliés par un switch

5.1 Arrêter le réseau, et remplacer le hub par un switch. Redémarrez le réseau. Faites un ping entre pc1 et pc2, et écoutez les échanges sur la liaison entre pc3 et switch. Que constatez-vous ? Quelle est la différence entre le hub et le switch ?

5.2 Expliquer comment le switch fonctionne pour distribuer les paquets

Points à retenir :

- Construire un réseau avec GNS3 et savoir sauvegarder et charger une configuration
- Affecter une adresse IP
- Faire un ping
- capturer des paquets avec *wireshark*
- le rôle d'un hub et d'un switch