ANNÉE 2022/2023

TP n° 2 : Outils pour les réseaux

NET4101

Ingénieur généraliste, 2ème année

Ce document est soumis à une licence Creative Commons Attribution
Partage dans les Mêmes Conditions 4.0 International —

Rédacteurs

Rémy Grünblatt
Maître de conférences
remy.grunblatt@telecom-sudparis.eu

Jehan Procaccia
Ingénieur systèmes et réseaux
jehan.procaccia@imtbs-tsp.eu

Équipe enseignante

Andrea Araldo
Maître de Conférences
andrea.araldo@telecom-sudparis.eu

Laurent Bernard
Directeur d'études
laurent.bernard@telecom-sudparis.eu

Franck Gillet
Ingénieur R&D — Plateforme
franck.gillet@telecom-sudparis.eu

Antoine Lavignotte
Directeur d'études
antoine.lavignotte@telecom-sudparis.eu

Objectifs de ce TP et compétences à acquérir

— Connaître les interfaces des matériels utilisés dans NET 4101
— Savoir se connecter aux équipements réseaux
— Configurer et utiliser un client SSH
— Connaître les outils et commandes de base de configuration réseau sous différents systèmes
— Mettre en œuvre une configuration réseau simple en ligne de commande
— Diagnostiquer et réparer des problèmes réseaux classiques
— Manipuler et utiliser wireshark

Important : Pour la réalisation de ce TP, vous devez récupérer une fiche avec des informations spécifiques à votre groupe de binôme auprès de votre chargé de TP.

Des outils, oui, mais pour quoi faire ?

L'informatique et les réseaux sont des disciplines *artisanales*. Quand on programme, quand on conçoit et déploie des réseaux, on n'est en général pas dans un contexte industriel de masse mais dans un contexte particulier, qui nécessite un certain *savoir-faire*. De la même manière qu'un bon cuisinier possède des couteaux bien affutés, des casseroles, des faitouts… pour faire du réseau, il est nécessaire de connaître (bien) et de savoir manipuler (mieux) un certain nombre d'outils. En bref, d'avoir différentes options pour diagnostiquer, débuguer, reconfigurer, *comprendre* ce que l'on manipule.

Avant de commencer à s'intéresser aux *logiciels*, nous allons cependant commencer par un rappel autour de l'environnement *matériel* de l'ingénieur systèmes et réseaux.

1 Environnement Matériel

Le but de cette partie est de vous familiariser avec la salle de TP que vous avez déjà vu à la première séance, ainsi qu'avec le matériel à disposition.

1.1 Organisation générale physique des salles

Deux salles de TP sont utilisées dans ce module, la b101 et la b109. Bien que leur organisation soit globalement commune, elles diffèrent les unes des autres par de petites variations.

1.1.1 Postes

Les postes de chaque salle utilisent la distribution Linux CentOS. Au démarrage de la machine, une session s'ouvre automatiquement sans avoir besoin de rentrer de mot de passe, respectivement sous l'utilisateur b101 et l'utilisateur b109, selon la salle dans laquelle vous vous trouvez. Vous n'êtes pas *root* (super-utilisateur) sur ces machines, mais vous n'avez pas besoin de l'être car la majorité de vos TPs se passeront dans des machines virtuelles « hébergées » par ces machines, machines dans lesquelles vous pourrez être *root*. Majoritairement, les machines virtuelles utiliseront la distribution Linux Ubuntu, mais parfois aussi Windows.
L'utilisation de ces machines virtuelles permet de personnaliser l'environnement pour chaque TP, par exemple en proposant certains outils spécifiques, sans avoir besoin de reconfigurer les machines hôtes.

1.1.2 Câblage et réseau

Chaque poste possède deux cartes réseaux physiques, l'une intégrée à sa carte mère et l'autre ajoutée en sus, toutes deux connectées par bus PCI.

Information : Sous les systèmes d'exploitation basés sur Linux, il est possible d'utiliser respectivement les programmes « `lspci` », « `lsusb` » et « `lshw` » pour lister les composants matériels sur le bus PCI, le bus USB, ou plus généralement obtenir les caractéristiques de l'ensemble des composants connectés à l'ordinateur. On peut par exemple utiliser « `lshw -C network` » pour se limiter aux cartes réseaux.

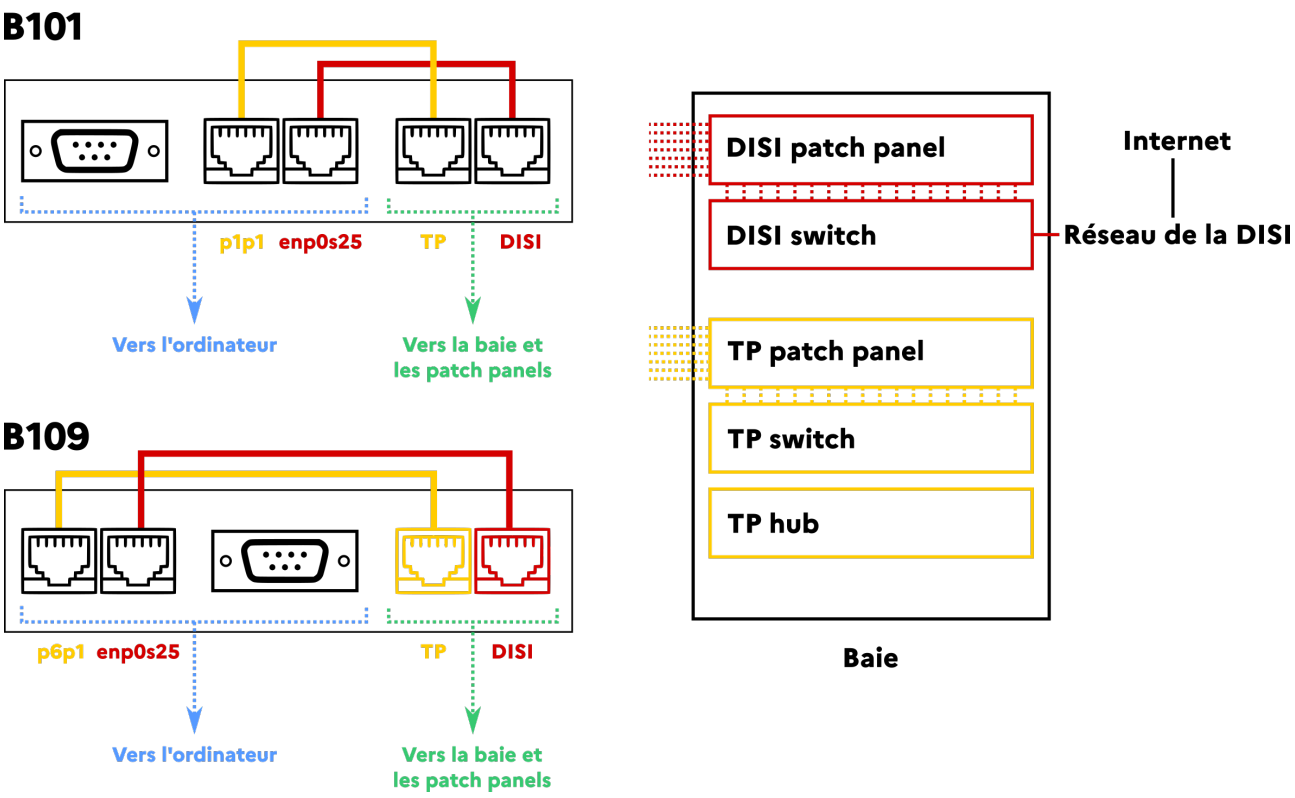

FIGURE 1 – Zoom sur le câblage des *patch panels* de bureau de la B101 et de la B109. Dans cette image, le réseau les câbles sont connectés au switch TP, mais on peut aussi décider de les relier au hub TP.

Pour chaque machine, chacune de ces cartes réseaux est connectée, par le biais de « *patch panel* » disponibles sur chaque bureau, à l'un des deux segment réseaux utilisé dans les salles : le segment réseau « DISI », sur lequel les machines sont connectées à un switch géré par la DISI, et sur lequel un routeur permet un accès à Internet (configuration automatique par DHCP), et un segment réseau « TP », commun aux deux salles, sur lequel les machines sont connectées à un hub. Un accès au port série de l'ordinateur peut également se faire via le *patch panel*, évitant de manipuler les unités centrales. Les configurations et branchements dans les salles B101 et B109 sont décrites par la figure 1.

Important : Il est important de rebrancher les câbles dans cette configuration à la fin des TPs, pour éviter aux suivants (e.g. potentiellement vous) de se retrouver face à une configuration inconnue. En cas de doutes, demandez nous !

Le nom de l'interface intégrée à la carte mère est nommée **« `enp0s25` »**, et le nom de l'interface additionnelle est nommée **« `p1p1` »** en B101 et **« `p6p1` »** en B109.

1.2 Système d'exploitation hôte et Virtualbox

Dans le cadre des TPs, nous utiliserons principalement des machines virtuelles et des logiciels dans ces machines virtuelles. Ces machines virtuelles peuvent accéder aux périphériques de la machine hôte de différentes manières. On détaille ici les configurations la plus utilisées :

- **Cartes réseaux** : les cartes réseaux *virtuelles* de la machine *virtuelle* sont en général « bridgées » (ou « pontées ») avec les interfaces physiques de la machine hôte. Il s'agit de faire comme si un *switch* était présent en amont de l'interface physique, *switch* connecté à l'interface physique et à l'interface virtuelle. Du point de vue de l'extérieur (par exemple, du point de vue de la DISI), deux cartes réseaux sont visibles, notamment avec deux adresses MAC et deux adresses IP différentes.
- **Périphériques USB** : les périphériques USB (souris, clavier, caméra, …) peuvent être connectés soit à l'hôte, soit à une **unique** machine virtuelle. Dans ce second cas, l'hôte effectue du « passthrough » pour envoyer les données échangées avec le périphérique directement à la machine virtuelle, sans interagir avec le périphérique : il ne sert que de passeur de plats. On peut choisir de rattacher un périphérique en utilisant le menu « `> Devices > USB >` » puis en sélectionnant le périphérique à connecter, menu disponible depuis la barre d'outils de la machine virtuelle.

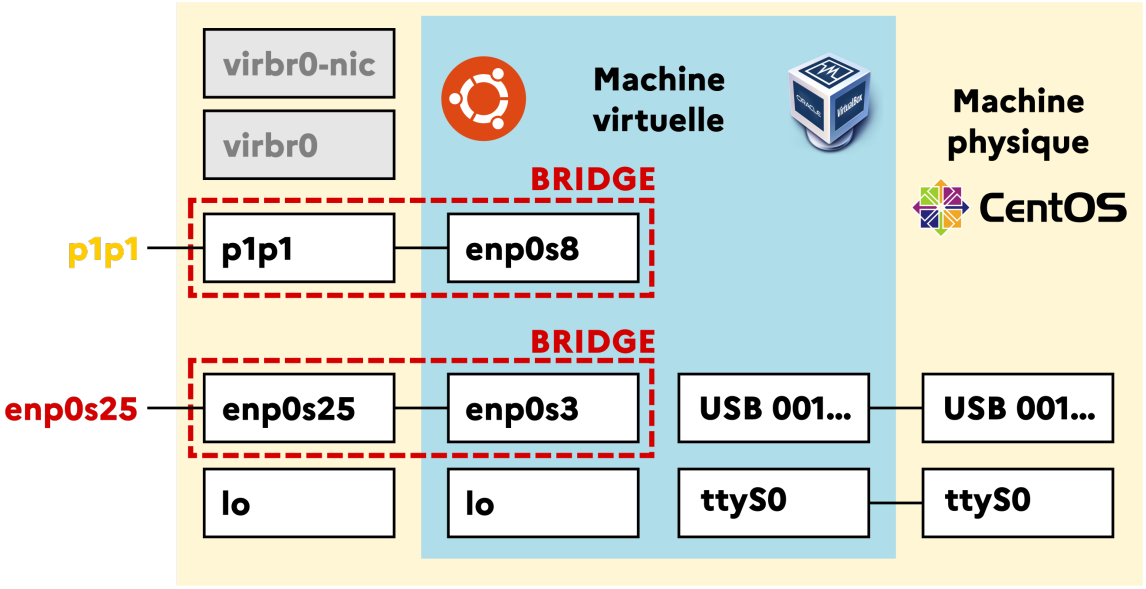

FIGURE 2 – Schéma logique d'une machine virtuelle dans la salle b101 — Exemple de configuration où la machine virtuelle est bridgée avec l'interface Ethernet « `enp0s25` », c'est-à-dire l'interface *normalement* connectée au segment réseau DISI. Dans cette configuration, la machine virtuelle peut récupérer une adresse IP qui lui est propre en utilisant le protocole DHCP comme si elle était directement connectée au segment réseau DISI.

— **Port série** : De la même manière que les périphériques USB, le port série peut être utilisé au choix par l'hôte, au choix par une **unique** machine virtuelle. Attention : plusieurs machines virtuelles peuvent *techniquement* écrire dans le même port série, résultant alors en l'écriture de données incorrectes sur le port série. Il faut bien faire attention à n'utiliser le port série que depuis une unique machine virtuelle ou uniquement depuis l'hôte.

Pour accéder à la configuration d'une machine virtuelle, il suffit de sélectionner la machine virtuelle dans la fenêtre de VirtualBox et d'appuyer sur le bouton « `Settings` ».

Information : La configuration des bridges de VirtualBox n'est pas visible dans les outils classiquement utilisés pour configurer les bridges sous Linux. En effet, VirtualBox utilise son propre module noyau pour implémenter son propre système de bridge, et il n'est donc pas possible de configurer le bridging entre machine virtuelle et hôte en utilisant par exemple le programme « `ip` » ou le programme « `bridge` ».

Question 1 : Lancez la machine virtuelle nommée "Ubuntu-22-04-Generic" et vérifiez qu'elle est fonctionnelle. Vérifiez qu'elle est notamment connectée à Internet en lançant un navigateur Web et en allant observer votre adresse IP telle que perçue par le site Web `https://monip.org/`. Comparez avec l'adresse IP obtenue en répétant la procédure sur la machine hôte. Que peut-on en déduire quant à la configuration réseau de cette machine ?

Question 2 : Branchez une Webcam USB à votre machine et vérifiez qu'il est possible de déléguer ce périphérique à votre machine virtuelle. On pourra par exemple utiliser le programme « `cheese` » pour tester que la Webcam est bien reconnue et fonctionne.

Question 3 : Remplissez les noms des membres de votre binôme sur votre fiche de réponse et prenez une capture d'écran (depuis la VM) où est visible cette fiche dans « `cheese` » (toujours dans la VM). Partagez cette capture d'écran entre votre VM et l'hôte en utilisant les dossiers partagés (dont des raccourcis sont disponibles sur le bureau dans la machine virtuelle). Visualisez ce fichier sur la machine hôte, et prenez une seconde capture d'écran, cette fois-ci depuis la machine hôte. Gardez cette dernière capture qui sera utilisée par la suite du TP.

1.3 Équipements Cisco Catalyst 2950 et 2960 (switchs) et Cisco ISR 4331 (routeurs)

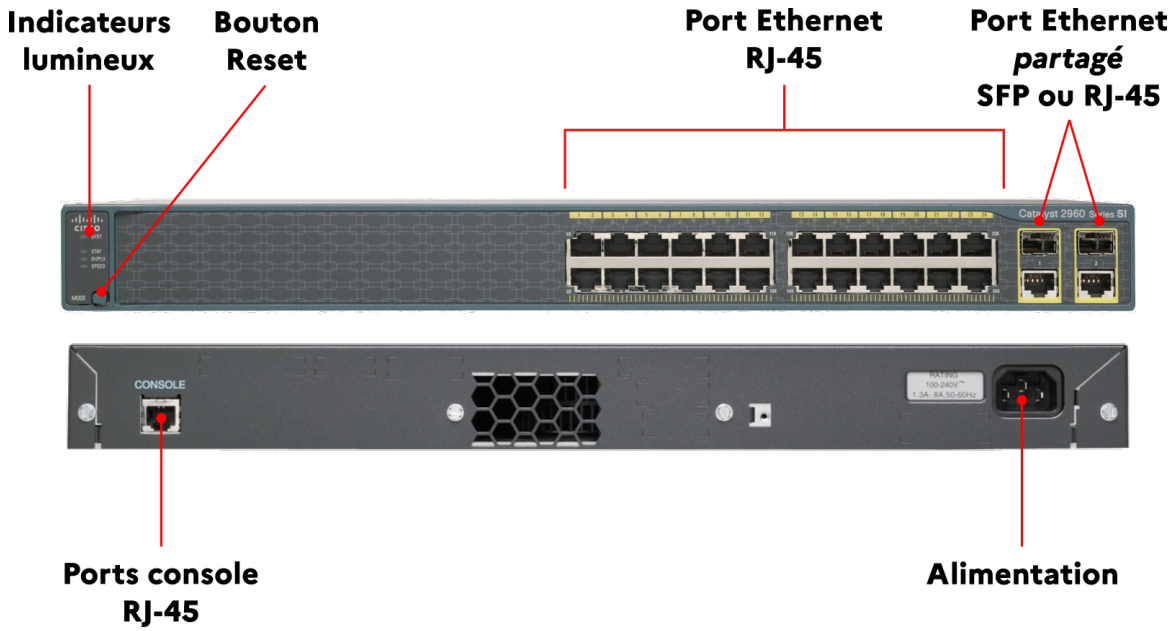

FIGURE 3 – Cisco Catalyst 2960, l'un des switchs utilisés dans ce module

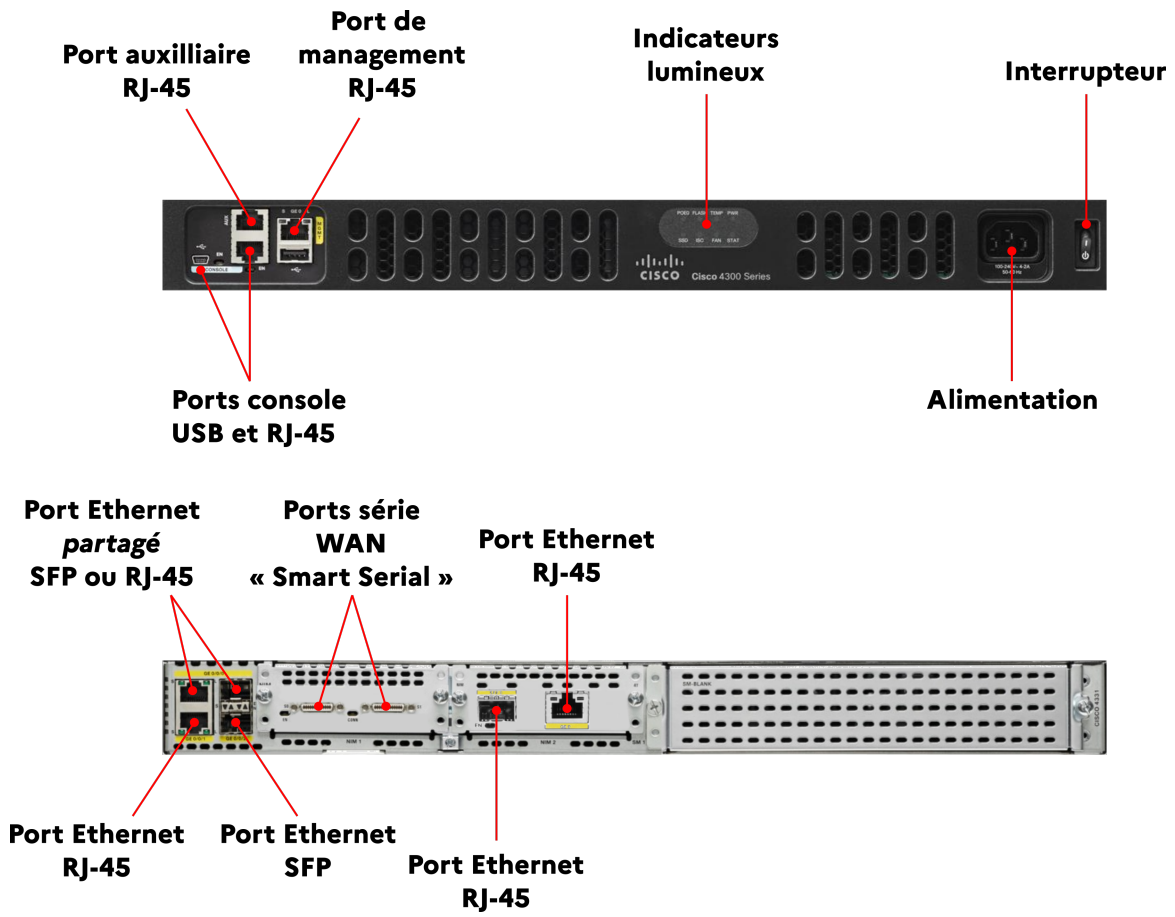

FIGURE 4 – Cisco ISR 4331, l'un des routeurs utilisés dans ce module

Attention : Lors de leur démarrage, les routeurs Cisco ISR 4331 font tourner leur ventilateur à la vitesse maximale pendant environ 7 minutes, ce qui a tendance à faire *beaucoup* de bruit. Il convient d'éviter de les redémarrer trop souvent pour le confort auditif de l'ensemble de la salle.

Question 4 : Vérifiez que votre routeur ISR 4331 est bien allumé et connecté sur son port console à votre machine. Utilisez le programme « `minicom` » (en terminal) pour vous connecter au routeur ISR 4331 par le biais de ce port console, au choix depuis l'hôte ou depuis une machine virtuelle. Sous Windows, on pourra utiliser l'utilitaire « `Putty` » ou utiliser WSL (Windows Subsystem for Linux) pour utiliser « `minicom` » en terminal. Pour vérifier que tout fonctionne bien, appuyez simplement sur la touche Entrée plusieurs fois, pour vérifier si une nouvelle ligne vous est proposée dans le shell Cisco.

Note : Pour les équipements Cisco, la configuration du port console est en général 9600-8-N-1 : 9600 bits par seconde, 8 bits de données (8) , pas de bit de parité (N), et 1 bit stop (1).

2 Outils logiciels

2.1 Linux

Note : Dans toutes cette section, les opérations se feront depuis la VM Ubuntu-22-04-Generic. Votre utilisateur « `utilisateur` » est sudoers et son mot de passe est « `motdepasse` ».

Sous Linux, quelques commandes permettent d'interagir avec le système pour gérer et configurer le réseau. Certains programmes prennent cependant la main sur la configuration du réseau, auquel cas utiliser ces commandes pour modifier la configuration n'aura probablement pas l'effet escompté. En effet, le programme, fonctionnant très souvent sous la forme d'un démon logiciel, écrasera votre configuration manuelle ou interférera avec vos actions. Parmi ces programmes, on peut noter deux programmes bien connus :

- **NetworkManager** : utilisé notamment sous Ubuntu, il permet de gérer et de configurer les connexions filaires et sans fils en utilisant une interface graphique (mais il fonctionne aussi en ligne de commande)
- **Systemd-networkd** : intégré au système d'init systemd, il est principalement utilisé en ligne de commande et configuré en éditant des fichiers de configurations. Il est principalement à destination des serveurs.

D'autres programmes comme « `dhcpcd` » (pour **dhcp c**lient **d**aemon) ou « `wpa_supplicant` » sont aussi utilisés pour configurer le réseau automatiquement (parfois comme dépendances).

Important : Il convient donc de s'assurer que les interfaces que l'on configure « manuellement » ne sont pas déjà configurées automatiquement par des programmes comme NetworkManager. Pour NetworkManager, on pourra utiliser la commande « `nmcli` » pour accéder à sa configuration en ligne de commande.

Connaître la configuration réseau

Question 5 : Dans un terminal, lancez la commande « `ip` » et déterminez comment utiliser les sous-commandes « `address` », « `link` », « `neighbour` » et « `route` » pour **lire** la configuration réseau. À quoi servent chacune de ces sous-commandes ? Avec quelle(s) commande(s), et sur quelle(s) machine(s) est-il possible de retrouver les adresses visibles dans la question 1 ?

Question 6 : À quoi correspondent les options -4 et -6 du programme ip ?

Information : N'hésitez pas à placer « `alias ip="ip -c=auto"` » dans le fichier de configuration de votre shell, e.g. « `~/.bashrc` », afin d'utiliser automatiquement la coloration de la sortie de la commande ip

Question 7 : À quoi servent les fichiers « `/etc/resolv.conf` », « `/etc/hostname` » et « `/etc/hosts` » ?

Information : Pour afficher des fichiers en ligne de commande sous linux, on utilise le programme « `cat` », par exemple « `cat /etc/foo/bar` ».

Modifier la configuration réseau

Pour modifier la configuration d'une machine sous Linux, plusieurs s'options s'offrent à nous. On pourra modifier des paramètres directement dans des fichiers de configurations (à l'aide d'éditeurs de textes) comme « `/etc/hostname` », « `/etc/resolv.conf` », ou encore « `/etc/NetworkManager/NetworkManager.conf` ». On pourra aussi utiliser des programmes en ligne de commande (e.g. « `nmcli` », « `hostnamectl` », « `networkctl` » pour systemd-networkd, resolvectl si le fichier « `resolv.conf` » est géré par « `systemd-resolved` ») pour aller modifier de manière *programmatique* ces fichiers. On pourra aussi s'appuyer sur l'utilitaire « `ip` », qui permet de gérer les interfaces, les adresses, les routes, et d'en modifier la configuration (en plus de la consulter, comme vu précédemment).

Important : La syntaxe pour modifier des paramètres réseaux avec « `ip` » peut sembler compliquée au premier abord. N'hésitez pas à utiliser le manuel (commande « `man` ») pour consulter des exemples sur l'utilisation d' « `ip` »! Par exemple, on pourra utiliser « `man ip` », « `man ip link` », « `man ip route` » ou « `man ip address` », puis chercher la section « `EXAMPLES` » (située à la fin de la page de manuel) en tapant « `/EXAMPLES` » pour se déplacer directement vers la section.

Question 8 : Dans la VM, utilisez « `ip link` » pour placer l'interface (virtuelle) bridgée avec l'interface (physique) `p1p1` ou `p1p6` dans son état "UP". L'interface possède-t-elle des adresses IPs ? Si oui, de quelle famille ? Pourquoi ?

Question 9 : Ajoutez une adresse IPv4 (en utilisant « `ip address` ») à l'interface virtuelle bridgée avec l'interface physique du segment « TP » (`p1p1` ou `p1p6`) dans le range « `10.13.37.0/24` » (l'adresse en « `.254` » est déjà utilisée). Pour éviter les collisions, merci d'utiliser l'adresse mentionnée sur la fiche que vous avez reçu au démarrage du TP. Spécifiez ensuite que le réseau « `10.13.37.0/24` » est disponible directement sur votre segment réseau (*on-link*) en utilisant « `ip route` ». Chargez la page web `http://10.13.37.254/`, et notez le *flag* affiché sur votre fiche de réponse.

Question 10 : Sur le réseau « `10.13.37.0/24` », un routeur est disponible à l'adresse « `10.13.37.254` ». Ajoutez une route par défaut en utilisant ce routeur avec « `ip route` », en spécifiant une métrique plus basse que la route par défaut déjà présente sur la machine virtuelle. Utilisez ensuite un navigateur web (ou « `curl` », ou « `wget` ») pour vérifiez avec quelle IP vous sortez sur Internet, toujours en utilisant le service web *https ://-monip.org*. Notez cette IP sur votre fiche de réponse.

2.2 Cisco

Note : Dans toutes cette section, on suppose que l'on est connecté par le port série aux équipements CISCO. Cette connexion peut se faire depuis l'hôte, depuis une VM Ubuntu, ou depuis une VM Windows.

La CLI Cisco n'est pas la CLI Linux ou la CLI Windows. Ainsi, son fonctionnement est différent, et on n'aura a priori pas accès aux utilitaires que l'on a l'habitude d'utiliser sur ces systèmes, même si *sous le capot*, Cisco IOS XE (utilisé par les ISR, mais pas par les switchs) utilise Linux… La CLI Cisco se caractérise par plusieurs niveaux de privilèges dont un résumé est disponible sur la figure 1. À tout instant, dans la CLI Cisco, il est possible d'utiliser le point d'interrogation « « ? » » pour lister les commandes disponibles dans le terminal. Pour naviguer dans la sortie d'une commande Cisco, on peut utiliser les flèches directionnelles ou la barre espace pour faire défiler plus rapidement les commandes.

Nom du mode	État de la CLI	Utilisation
User exec	Router> Switch>	Mode dans lequel on se connecte généralement aux équipments. Permet d'exécuter des commandes de base de type « `ping` ». Utiliser « `logout` » pour se déconnecter.
Privileged exec	Router# Switch#	Pour passer dans ce mode, utiliser « `enable` » depuis le mode *user exec*. Pour le quitter, utiliser « `disable` ». Permet d'afficher la configuration de l'équipement avec e.g. « `show running-config` », de redémarrer la machine, de débuguer…
Global configuration	Router#(config) Switch#(config)	Pour passer dans ce mode, utiliser « `configure terminal` » depuis le mode *privileged exec*. Pour le quitter, utiliser « `exit` » ou « `<Ctrl>+z` ». Permet d'éditer la configuration de l'équipement, de ses interfaces…

TABLE 1 – Vue d'ensemble des différents modes de la ligne de commande Cisco

Exemple de passage d'un niveau de privilège à l'autre

```
Router>
Router>?
Exec commands:
  access-profile   Apply user-profile to interface
  app-hosting      Application hosting
  [...]
Router>enable
Router#
Router#?
Exec commands:
  access-profile   Apply user-profile to interface
  access-session   Access-session options for eEdge
  ↪  Cette commande n'était pas disponible en mode user exec !
  app-hosting      Application hosting
  [...]
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
Router(config)#exit
```

Note : Sauf mention contraire, les mots de passes utilisés pour « protéger » l'accès au mode privileged exec ou au terminal virtuel sont toujours « `cisco` » ou « `cisco123` ».

Connaître la configuration réseau

Pour connaître la configuration d'un équipement cisco (notamment un routeur), on utilise la commande « `show` » qui permet de consulter l'état courant de cet équipement.

Question 11 : À partir de quel niveau de privilège la commande « `show running-config` » est elle disponible ? À quoi sert cette commande ?

Question 12 : Quel est l'uptime et le numéro de version de votre routeur ? Indice : « `show v?` ».

Question 13 : À quoi sert la commande « `show ip interface brief` »? Quel est son principal intérêt par rapport à la commande « `show interfaces` » ?

Modifier la configuration réseau

Pour modifier la configuration d'un équipement Cisco, il est nécessaire d'avoir recours au terminal de configuration (mode « Global configuration »). La syntaxe à utiliser dans ce terminal est la même que celle visible dans la sortie de « `show running-config` ». Par exemple, on peut voir dans la sortie de « `show running-config` » une ligne du type « `hostname routeur-XX` ». Dans le terminal de configuration, il suffira donc de taper la commande « `hostname routeur-YY` » pour changer ce nom de machine vers « `routeur-YY` ».

Information : Les commandes Cisco peuvent être abrégées lorsqu'il n'y a pas d'ambiguïté sur ce qu'elles désignent. Par exemple, au lieu de taper « `configure terminal` », on pourra taper « `conf t` » car il n'existe pas d'autre commande que « `configure` » commençant par « `conf` », et il n'existe pas d'autre sous-commande que « `terminal` » commençant par « `t` ».

Certaines options de configurations, par exemple celles liées aux interfaces, possèdent des sous-options de configuration, qui sont visible dans la sortie de « `show run` » car elles présentent une indentation :

Exemple d'option de configuration avec sous-configuration

```
!
interface GigabitEthernet0/2/0
 no ip address
 shutdown
 negotiation auto
!
```

Pour modifier les sous-options de configuration, il faut utiliser le nom de l'option principale, par exemple « `interface GigabitEthernet0/2/0` », qui va permettre d'entrer un terminal de configuration spécifique à cette option, à la manière de la commande « `conf term` » (mode « `interface configuration` ») :

```
routeur-ABCD(config)#interface GigabitEthernet0/2/0
routeur-ABCD(config-if)#!On configure de la manière souhaitée...
routeur-ABCD(config-if)#exit
routeur-ABCD(config)#
```

Dans la sortie de « `show running-config` », on peut voir certaines lignes préfixées de « no », par exemple « `no ip http server` ». Ce préfixe sert à désactiver la fonctionnalité, par exemple ici la présence d'un serveur web de configuration hébergé sur le routeur. Pour réactiver l'option, il suffit de taper « `ip http server` » (sans le « no »).

Question 14 : Activez l'ensemble des interfaces Ethernet de votre routeur, puis configurez l'adresse IP déjà utilisée dans la partie Linux et disponible sur votre fiche à l'interface de votre choix. Vous brancherez ensuite cette interface au segment réseau « TP », ajouterez une route statique sur le routeur avec la commande « `ip route 10.13.37.0 255.255.255.0 GigabitEthernetX/Y/Z` ». Lorsque c'est fait, pinguez l'adresse « `10.13.37.254` » pour vérifier que tout fonctionne bien (commande « `ping` », depuis votre routeur).

Question 15 : Connectez une interface de votre routeur au segment réseau DISI, configurez cette interface en dhcp client (« `ip address dhcp` » en mode de configuration d'interface), et vérifiez que vous récupérez bien une IP sur le réseau de la DISI. Tentez ensuite de pinguer une adresse bien connue.

SSH : Se connecter à une machine distante

SSH est un logiciel (et un protocole) permettant de se connecter à des machines distantes de manière sécurisée. Dans le cadre de ce TP, chacun d'entre vous a reçu une feuille avec une adresse d'un serveur, un numéro de port, un nom d'utilisateur et un mot de passe. Ces informations vont pouvoir vous servir à vous connecter à une machine virtuelle distante sur laquelle vous pourrez exécuter des commandes de test.

Information : Votre utilisateur est sudoers sur la machine distante : vous pouvez utiliser sudo pour passer root.

Question 16 : Depuis la VM Ubuntu-22-04-Generic, connectez vous à la VM distante en utilisant le nom d'utilisateur, le mot de passe, le port et l'adresse mentionnés sur votre fiche. Déconnectez vous de la VM distante. Dans la VM Ubuntu-22-04-Generic, générez une clef SSH avec l'utilitaire « `ssh-keygen` » (options par défaut) et copiez cette clef vers la VM distante en utilisant l'utilitaire « `ssh-copy-id` ». Essayez de vous connecter à nouveau à la VM distante. Qu'observez vous ?

Question 17 : Dans le home de la VM distante, créez un fichier nommé "groupe.txt" dans votre home (par exemple avec la commande « `touch` »), puis éditez le pour y faire figurer le nom des deux membres de votre groupe (par exemple, avec « `nano` » ou avec « `echo` » et une redirection de flux).

Question 18 : Quel est le système d'exploitation de la VM distante ? Depuis combien de temps la machine est elle allumée ? On pourra regarder du côté des commandes « `uname` » (avec une option spécifique pour tout afficher) et « `uptime` ».

Question 19 : Transférez la seconde capture d'écran (effectuée au début de la séance) vers la VM distante (éventuellement en ré-utilisant les dossiers partagés), avec « `scp` ». La syntaxe de base de « `scp` » est la suivante : « `scp <source> <destination>` », où « `<source>` » et « `<destination>` » sont du type « `utilisateur@hostname:/chemin/vers/le/fichier` » pour un fichier distant, et « `/chemin/vers/le/fichier` » pour un fichier local.

Information : Savoir accéder à une machine « distante » est important, que ce soit pour l'utilisation des ressources de cette machine, pour le diagnostic de problèmes réseaux, ou tout simplement pour expérimenter l'envoi et la réception de données à travers Internet. SSH est la méthode préférentielle pour l'administration des équipements réseaux : la connexion console / série ne devrait être qu'une solution de secours / de configuration initiale !

tcpdump et tshark / wireshark : observer et analyser le trafic réseau

Les outils tcpdump (CLI) et wireshark (graphique, tshark en CLI) permettent d'observer et d'analyser le traffic réseau passant sur les interfaces réesaux de la machine sur laquelle ils sont exécutés. Pour pouvoir capturer ce trafic, il est nécessaire d'avoir des droits super-utilisateurs : on imagine très bien quels problèmes de confidentialité pourraient se présenter si l'on laissait n'importe quel utilisateur capturer le trafic des autres utilisateurs de la machine. En général, sous Linux, ces droits super-utilisateurs peuvent être obtenus en utilisant les commandes « `su` » (**s**witch **u**ser), qui demande le mot de passe de l'utilisateur que l'on souhaite utiliser, ou « `sudo` » (**s**witch **u**ser and **do**), qui demande le mot de passe de l'utilisateur actuel. Une autre manière de permettre la capture est d'attribuer une fois pour toute le bit SUID au binaire utilisé (tcpdump, tshark, …) ce qui permettra à n'importe quel utilisateur, même non super-utilisateur.

Information : Le programme tshark fait partie de la suite logicielle Wireshark. À ce titre, la syntaxe des filtres de tshark correspond au format des filtres de wireshark. Un autre utilitaire premettant d'observer et d'enregistrer le traffic en ligne de console est tcpdump, e.g. « `tcpdump -i any 'port 53 and udp' -w capture.pcapng` ». Malgré son nom, tcpdump n'est pas limité à la capture du trafic en TCP.

Question 20 : Sur la VM distante, le programme « `tshark` » est installé. Dans la commande « `tshark -i any -f 'tcp port 80' -l -w capture.pcapng` », à quoi servent les différentes options ?

Question 21 : Lancez une capture sur la VM distante d'environ une minute **sans aucun filtre**. En parallèle, lancez une seconde connexion SSH vers la machine (ou utilisez un multiplexeur de terminal comme tmux) et lancez les commandes

- « `$ wget -4 http://telecom-sudparis.eu/` »
- « `$ wget -6 http://telecom-sudparis.eu/` »
- « `$ dig telecom-sudparis.eu` »
- « `$ dig telecom-sudparis.eu @8.8.8.8` »

Stoppez ensuite la capture avec « `<Ctrl>+C` » puis utilisez « `scp` » pour récupérer le fichier « `capture.pcapng` » sur votre VM locale.

Information : « `tmux` » est un multiplexeur de terminal : en lançant tmux, on récupère un terminal virtuel que l'on peut « détacher » en tapant sur <CTRL>+b puis d (pour *detach*), et ré-attacher en utilisant la commande « `tmux attach` ». Ainsi, on peut lancer un processus long dans un tmux lancé sur une machine distante, détacher ce terminal et couper la connexion SSH : le processus continuera de s'exécuter même en l'absence de connexion SSH ! Tmux permet aussi de créer des terminaux virtuels, avec <CTRL>+b puis c pour *create*), et de se déplacer entre ces terminaux virtuels avec les commandes <CTRL>+b puis n pour *next* et <CTRL>+b puis p pour *previous*.

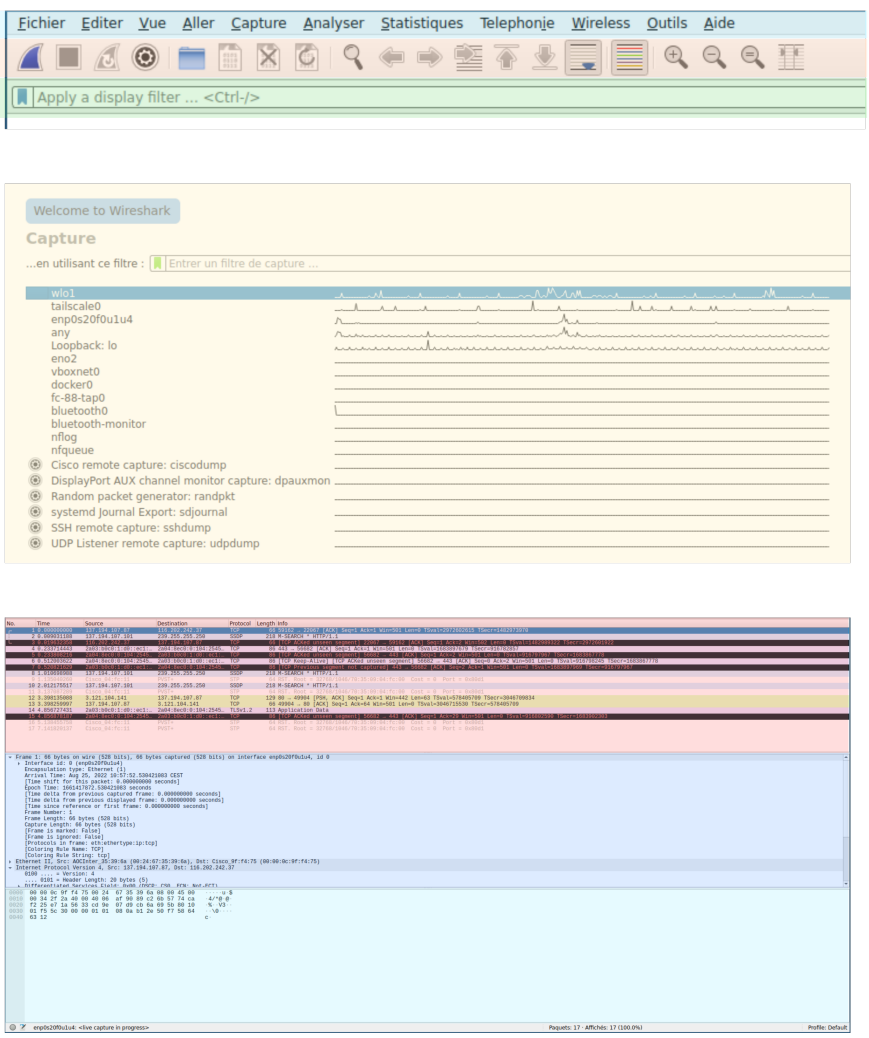

FIGURE 5 – Vue d'ensemble de l'interface graphique de Wireshark. Les parties soulignées sont les plus importantes. Au lancement de wireshark (appelé sans arguments), il est possible de sélectionner une interface pour effectuer une capture locale « en direct ». Il est aussi possible d'ouvrir un fichier de capture (de type « `.pcap` » ou « `.pcapng` ») pour une analyse « hors-ligne »

Question 22 : Ouvrez ce fichier dans Wireshark pour procéder à une analyse avec une interface graphique. Dans la capture, quel protocoles observez vous ? (Indice : utiliser « `Statistiques > Hierarchie des protocoles` »). Quelles adresses IPv4 sont visibles dans la capture ? (Indice : utiliser « `Statistiques > IPv4 Statistics` ». Prenez une adresse au hasard, et utilisez la commande « `whois <ADRESSE-IP>` » pour obtenir plus d'informations à son propos. Quel organisme possède cette adresse IP ?

Question 23 : À quoi ont servi les commandes *wget* et *dig* ? Retrouvez, en utilisant des filtres, ces requêtes que vous avez effectué manuellement, dans l'interface de Wireshark. On pourra noter que les filtres wireshark s'écrivent dans une syntaxe proche de celle du python, e.g. il est possible d'écrire des filtres comme « `tcp.port in {80, 443, 1337} or tcp.sport == 1234` ».

Attention : Tshark et tcpdump capturent le trafic au niveau de l'interface réseau, avant même que le pare-feu de la machine n'entre en jeu. Ainsi, si un pare-feu (comme nftables) bloque le trafic sur une machine, il est tout de même possible d'observer le trafic arriver sur l'interface avec ces outils.

Question 24 : Lancez Wireshark en capture sur toutes les interfaces de votre VM locale (Ubuntu-22-04-Generic). Depuis la VM distante (connexion via SSH), pinguez les adresses « publiques » IPv4 et IPV6 de cette VM locale. Les pings fonctionnent-ils ? Arrivez vous à observer les paquets dans Wireshark ?

2.3 Nmap : le standard pour scanner les réseaux

Pour débuguer certains problèmes réseaux, il est possible d'utiliser l'outil Nmap (en console) ou Zenmap (interface graphique à nmap) qui est un scanner pour les réseaux. Ce scanner va servir à découvrir les machines sur un segment réseau (par exemple, savoir quelles sont les machines allumées possédant une IP dans « `10.13.37.0/24` ») et les services fonctionnant sur ces réseaux (par exemple, détecter les ports ouverts en UDP ou TCP, et obtenir les versions des programmes écoutant sur ces ports).

Le syntaxe de nmap est la suivante : « `nmap <OPTIONS> <TARGET>` ». La partie « `<OPTIONS>` » permet de manipuler le comportement de Nmap, par exemple « `-sn` » pour n'effectuer qu'un « ping scan » pour vérifier si les machines sont allumées ou non, « -F » pour ne scanner que les ports les plus courants, « -p 1-100 » pour scanner les ports de 1 à 100… La partie « `<TARGET>` » représente l'IP, le nom d'hôte, ou le réseau à scanner, par exemple « `remy.grunblatt.org` », « `10.13.37.0/24` » ou « `192.168.1.3` ».

Attention : Par sécurité, assurez vous bien que vous n'êtes plus connecté au segment réseau « DISI ». Un scan réseau peut vite déclencher des alertes chez nos amis de la DISI, ce que nous ne souhaitons pas, et même si théoriquement vous ne devriez pas sortir sur ce réseau, il peut être une bonne idée de se sécuriser de ce côté là.

Question 25 : Certains ports sont ouverts sur une machine « cachée » dont l'IP est comprise dans le réseau « `10.13.37.0/24` » et le dernier octet est supérieur à *64*. Effectuez un scan réseau avec nmap (ou zenmap) pour découvrir sure quelle IP et quels sont ces ports, et notez les sur votre fiche de réponse. Que pouvez vous dire de cette machine cachée ? (par exemple, arrivez vous à déterminer si elle a un autre rôle dans le réseau ?).

2.4 Windows

Le système d'exploitation Windows est bien moins intéressant pour l'ingénieur et l'ingénieur réseau que les systèmes d'exploitation Linux, BSD, ou même OSX. En effet, la majorité des serveurs liés à Internet (Web, DNS, …) fonctionnent sous Linux, la totalité des super-calculateurs fonctionnent sous Linux, les équipements réseaux fonctionnent presque tous sur des systèmes d'exploitation basés sur Linux, Android est basé sur Linux… Cependant, Windows est encore largement utilisé sur les postes utilisateurs dans certains champs d'application (santé, bureautique, …), et l'administration de ces postes va souvent de pair avec l'utilisation de serveurs sous Windows. Windows n'étant pas fondamentalement nécessaire aux séances suivantes de ce module, cette section sera donc principalement informative.

Note : Dans toutes cette section, les opérations se feront depuis la VM Windows10Generic.

Connaître la configuration réseau

Outre les applications graphiques de Windows comme le « Centre Réseau et Partage » permettant d'afficher la configuration réseau sous Windows, l'invite de commande Windows est intéressante car elle permet de centraliser l'ensemble des informations liées à la configuration réseau de Windows.

Question 26 : Ouvrez un invite de commande PowerShell (raccourci « `Windows+r` » puis cmd.exe, ou en cliquant directement sur le raccourci). Testez les commandes « `ipconfig /all` », « `winver` », « `arp -a` », « `uptime` », « `route print` » et placez les dans la section « Pierre de Rosette ».

Modifier la configuration réseau

Pour modifier la configuration réseau, il est possible de passer par le centre réseau et partage et par l'interface graphique. Il est cependant possible d'utiliser la ligne de commande et les commandes mentionnées à la question précédente pour modifier ces paramètres.

3 Pierre de Rosette

Le but de cette section est d'accueillir vos notes autour de différentes commandes que vous utilisez dans ce TP. Toutes les lignes n'ont pas vocation à être remplies, car certaines manipulations n'ont pas vraiment de sens dans certains contextes (par exemple, remettre à zéro la configuration réseau de Windows ou de Linux ne peut pas se faire en une unique commande).

Raccourci pour ouvrir un terminal / une invite de commande :

LINUX : .....

WINDOWS : .....

CISCO : .....

Remettre à zéro la configuration réseau :

LINUX : .....

WINDOWS : .....

CISCO : .....

Connaitre l'uptime d'une machine :

LINUX : .....

WINDOWS : .....

CISCO : .....

Connaitre la version du système d'exploitation (noyau, ...) :

LINUX : .....

WINDOWS : .....

CISCO : .....

Lister les interfaces réseaux :

LINUX : .....

WINDOWS : .....

CISCO : .....

Effectuer un ping vers une adresse IP :

LINUX : .....

WINDOWS : .....

CISCO : .....

Lister les adresses IP des interfaces :

LINUX : .....

WINDOWS : .....

CISCO : .....

Lister les routes utilisées :

LINUX : .....

WINDOWS : .....

CISCO : .....

Lister les entrées du cache ARP :

LINUX : .....

WINDOWS : .....

CISCO : .....

Ajouter une adresse IP à une interface :

LINUX : .....

WINDOWS : .....

CISCO : .....

Obtenir la route utilisée pour contacter une adresse IP donnée :

LINUX : .....

WINDOWS : .....

CISCO : .....

Ajouter une route à une interface :

LINUX : .....

WINDOWS : .....

CISCO : .....

Désactiver / Activer une interface réseau :

LINUX : .....

WINDOWS : .....

CISCO : .....

Filtre pour n'afficher que le trafic ARP ou DHCP dans Wireshark

.....

Ping scan du réseau 10.0.0.0/24 en utilisant nmap

.....

Scan des ports les plus courants de la machine 10.13.37.254 avec nmap

.....