TP 2

Accès à des systèmes distants

Durée prévue : 1 séance (1h30)

Pour ces exercices, vous allez vous connecter à l'un des serveurs utilisables au département Info dans le cadre de l'UE : commencez par aller regarder dans TOMUSS le nom du serveur que l'on vous demande d'utiliser, dans la case `lifasr5-k`. Ce nom est de la forme `lifasr5-k`, où `k` est égal à 1, 2, ..., ou 6. **Dans la suite, on parlera toujours de `lifasr5-k`, mais il faudra utiliser le nom que vous avez trouvé dans TOMUSS**. De plus, **pNUMETU désigne votre login étudiant**.

2.1 Connexion `ssh` sur les machines `lifasr5-k`

De but est ici de réviser l'**utilisation élémentaire** de la commande `ssh` (déjà vue en LIFASR2), ainsi que quelques autres commandes (`ls`, `cd`, `pwd`...), et de voir comment se connecter sur l'une des machines dédiées à LIFASR5.

Lorsque vous lancez un terminal sur un système Linux, celui-ci vous permet d'accéder, (*via* un shell) d'intéragir avec la machine sur laquelle vous êtes connecté·e : on appellera cette machine la **machine locale**. Il est aussi possible d'ouvrir un terminal sur une **machine distante** au travers du réseau : dans ce cas, les commandes que vous tapez sur votre clavier, et dont vous voyez le résultat sur votre écran sont en fait exécutées sur la machine distante.

Il existe différents logiciels et différents protocoles pour faire cela. Nous utiliserons le logiciel `ssh`, car il s'agit d'une solution sécurisée très souvent utilisée en pratique. Pour pouvoir ouvrir une connexion `ssh`, il faut :

— que le le serveur `sshd` s'exécute sur la machine distante,

— pouvoir exécuter le client `ssh` sur la machine locale.

Ça tombe bien, car `sshd` tourne sur les machines `lifasr5-k`.

Les ordinateurs des salles de TP de Lyon 1 (par exemple, dans les salles du bâtiment Nautibus) sont directement connectés au réseau de l'Université. Depuis ces ordinateurs, vous pouvez directement vous connecter à une machine `lifasr5-k` grâce à la commande `ssh pNUMETU@lifasr5-k`, qui se trouve sur le même réseau, comme le montre la Figure 2.1.

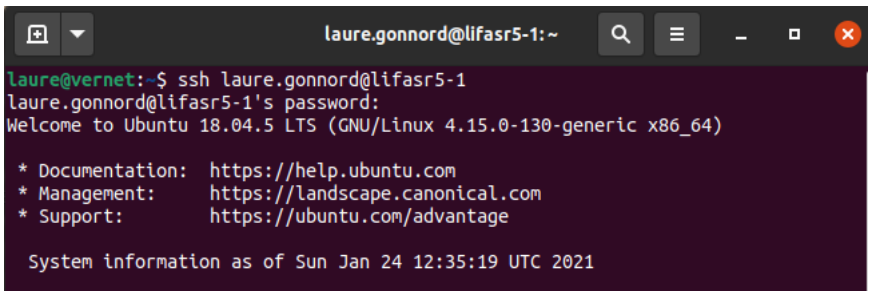

FIGURE 2.1 – Login "direct" (le login des enseignant·e·s est de la forme `prenom.nom` et non `pNUMETU`)

Il n'en est plus de même si vous travaillez depuis chez vous, ou tout simplement sur un ordinateur connecté en wifi au réseau Eduroam : dans ce cas, il faut utiliser une machine passerelle (dont le nom complet est `linuxetu.univ-lyon1.fr`) pour vous permettre d'entrer sur le réseau de l'université.

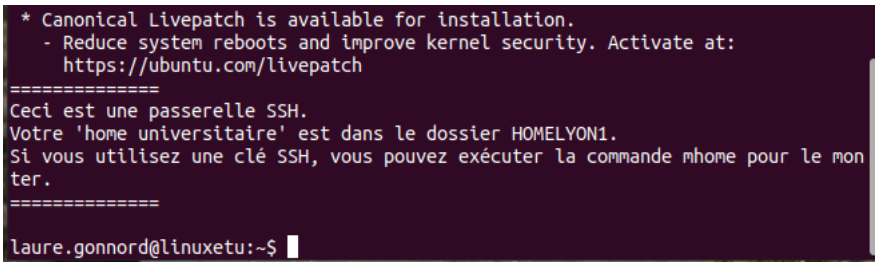

FIGURE 2.2 – Connexion sur `linuxetu.univ-lyon1.fr`

Dans la suite, même si vous êtes connecté.e sur un ordinateur de l'université, vous devez passer par la passerelle, et suivre le sujet ! Comme cela, vous apprendrez comment faire pour venir vous connecter une machine de l'université depuis chez vous.

EXERCICE 1 ▶ Connexion `ssh` sur les machines `lifasr5-k`

Connexion en deux étapes

Vous allez déjà commencer par aller vous connecter sur `lifasr5-k` en procédant en deux étapes :

— d'abord en vous connectant sur `linuxetu.univ-lyon1.fr`

— puis en vous connectant sur `lifasr5-k`

1) Connectez-vous sur `linuxetu`) en utilisant la commande `ssh pNUMETU@linuxetu.univ-lyon1.fr` ; il est probable que le client `ssh` vous demande de lui confirmer, à la première connexion, que vous connaissez bien l'identité de la machine `linuxetu`. En l'occurrence, vous pouvez répondre `yes`. Ensuite, entrez le mot de passe associé à votre login pour vous connecter. Cela va ressembler à ce que vous voyez sur la Figure 2.2.

2) Une fois connecté·e à `linuxetu`, quel est le chemin absolu de votre répertoire personnel sur cette machine ? Avez-vous accès à vos fichiers sur votre compte de l'université ?

.....

3) Connectez-vous [1] sur `lifasr5-k` avec la commande `ssh pNUMETU@lifasr5-k` ; à nouveau, il est probable que le client `ssh` vous demande de lui confirmer que vous connaissez bien l'identité de la machine `lifasr5-k` : répondez `yes`, comme dans l'exemple ci-dessous.

```
laure.gonnord@linuxetu:~$ ssh laure.gonnord@lifasr5-2
The authenticity of host 'lifasr5-2 (192.168.78.37)' can't be established.
ECDSA key fingerprint is SHA256:D0BrzN2rr8BPlSU4VlR2/nk+DOFrF9pbeONhICqIwtk.
Are you sure you want to continue connecting (yes/no)? yes
```

4) Une fois connecté.e à `lifasr5-k`, quel le chemin absolu de votre répertoire personnel ?

.....

5) Où se trouvent les fichiers de votre compte habituel de l'université ?

.....

6) Lancez un éditeur de texte en mode graphique, par exemple `geany &` ou bien `gedit &` : que se passe-t-il et pourquoi ?

.....

7) Déconnectez-vous de `lifasr5-k` en entrant `exit` ou en tapant le caractère de fin de fichier EOF (Ctrl + D). Sur quelle machine êtes vous désormais connecté·e ?

.....

8) Déconnectez-vous à nouveau avec `exit` ou Ctrl + D : êtes vous bien connecté·e sur votre ordinateur local ?

.....

EXERCICE 2 ▶ Connexion avec un rebond

Dans l'exercice précédent, nous avons d'abord ouvert un terminal sur `linuxetu`, puis de `linuxetu` nous avons ouvert un terminal sur `lifasr5-k` : `linuxetu` nous a servi de machine intermédiaire pour rebondir vers le réseau de l'Université.

Il est possible de faire la même chose en une seule commande avec :

```
ssh -J pNUMETU@linuxetu.univ-lyon1.fr lifasr5-k
```

1) Testez la commande précédente en l'adaptant avec votre login `pNUMETU`. Vous devez effectivement vous connecter deux fois : qu'est-ce qui permet de l'affirmer ?

.....

2) Déconnectez vous avec `exit` ou Ctrl + D : dans quel terminal êtes-vous maintenant connecté·e ?

.....

3) Consultez la page de manuel de `ssh` (`man ssh`), et trouvez la documentation pour l'option `-J` : pourquoi l'option s'appelle t'elle `"-J"` ? Notez aussi le nom de la directive de configuration qui correspond à cette option.

.....

4) Reconnectez-vous, en ajoutant les options `-X` et `-C` à la commande, ce qui peut s'écrire : `ssh -CX -J pNUMETU@linuxetu.univ-lyon1.fr pNUMETU@lifasr5-k`. Tentez de lancer à nouveau un éditeur graphique, avec `geany &` ou `gedit &` : cette fois-ci, vous devez voir apparaître la fenêtre sur votre machine locale. Quelle est le rôle de chacune des deux options `-X` et `-C` (`man ssh`) ?

1. à partir du terminal de la question précédente, dans lequel vous êtes encore connecté·e sur linuxetu.

.....

.....

EXERCICE 3 ► **Fichier de configuration pour vos connexions**

Il faut bien reconnaître qu'il n'est pas facile de retenir toutes les commandes précédentes, surtout leurs options : on va tâcher de créer, sur votre machine locale, un fichier de configuration pour votre client `ssh`, de façon à ne pas avoir de commandes trop compliquées à retenir !

1) À quoi sert l'option `-d` de `ls` ?

.....

2) Sur votre machine locale, placez vous à la racine de votre répertoire personnel (avec `cd`) : est-ce qu'un répertoire `.ssh` est présent ? Si oui, qui en est le propriétaire, et quel droits sont positionnés pour ce fichier ? Que signifie ces droits ?

.....

3) Imaginons que le répertoire `.ssh` ne soit pas présent à la racine de votre répertoire personnel : quelles commandes utiliser pour le créer, pour vérifier que vous en êtes bien le/la propriétaire, et pour que seul·e le/la propriétaire puisse lister son contenu, ajouter des fichiers dedans ou aller dans un de ses sous-répertoires ?

.....

4) Editez un fichier `~/.ssh/config` avec un éditeur de texte tout simple. Ce fichier, s'il existe, est consulté par la commande `ssh` lors de son lancement, et peut contenir des directives de configuration. Ajoutez les lignes suivantes :

```
Host linuxetu
        HostName linuxetu.univ-lyon1.fr
        User pNUMETU
        Port 22

Host lifasr5-k
        HostName lifasr5-k
        User pNUMETU
        ProxyJump linuxetu
        ForwardX11 yes
        Port 22
```

Veillez à bien remplacer `lifasr5-k` par le nom de la machine qui vous est attribuée dans TOMUSS, et `pNUMETU` par votre login ; enregistrez bien votre fichier !

5) En consultant la page de manuel `ssh_config`, déterminez pour chaque directive du fichier de configuration ci-dessus quel est son rôle. Est-ce que `ProxyJump` vous rappelle quelque chose ?

.....

.....

.....

.....

6) Maintenant, vous devez pouvoir ouvrir un terminal sur `lifasr5-k` simplement avec la commande : `ssh lifasr5-k` ; si ça ne fonctionne pas, corrigez avec votre chargé·e de TP.

2.2 Utilisation d'une paire de clés

Le but de l'utilisation de clés est de vous éviter d'avoir à entrer votre *password* (mot de passe) à chaque fois que vous vous connectez à une machine avec `ssh`. Une fois que tout est mis en place, vous n'avez plus qu'à entrer une fois une *passphrase* (phrase de passe ?) pour permettre à `ssh` d'utiliser votre clé pour toutes les connexions suivantes.

Pour `ssh`, une clé est en fait constituée d'une paire de clés :

— une clé privée, que vous conservez précieusement dans le répertoire `~/.ssh/` de votre compte sur votre machine locale, et que vous ne confiez à rien ni personne !

— une clé publique, que vous pouvez déposer dans le fichier `~/.ssh/authorized_keys` de votre compte sur différentes machines distantes.

Après avoir déposé votre clé publique sur une machine distante, vous pourrez vous connecter dessus sans avoir à retaper votre *password*, pourvu que vous ayez déjà entré votre *passphrase* pour déverrouiller votre clé privée. En pratique, une clé (privée ou publique), est simplement un grand entier codé en hexadécimal dans un fichier, avec un peu de blabla administratif en plus. Selon le type de clé utilisé, les fichiers d'une paire de clé peuvent s'appeler :

— `id_rsa` et `id_rsa.pub`
— ou `id_dsa` et `id_dsa.pub`
— ou `id_ed25519` et `id_ed25519.pub`

Le fichier se terminant par `.pub` contient la clé publique, l'autre fichier contient la clé privée.

EXERCICE 4 ► **Réutilisation ou création d'une paire de clés**

— Si vous avez déjà créé une paire de clés dans une autre UE, et que vous vous souvenez de votre *passphrase*, il semble logique que vous vous en resserviez. Vous devriez retrouver cette paire de clés dans votre répertoire `~/.ssh/`. Dans la suite, nous partirons du principe que vous avez une paire de clé pour laquelle les fichiers s'appellent `id_rsa` et `id_rsa.pub`, mais charge à vous d'adapter les noms de fichiers à votre situation !
— Si vous avez déjà créé une paire de clés dans une autre UE, mais que vous ne vous souvenez pas de votre *passphrase* : supprimer les fichiers correspondant à cette paire de clé inutile dans votre répertoire `~/.ssh/`. Attention de ne pas supprimer le fichier `~/.ssh/config` que nous avons crée tout à l'heure !
— Si vous n'avez pas de paire de clés dans votre répertoire `~/.ssh/`, vous allez en créer une avec la commande :

```
ssh-keygen -t rsa -b 4096
```

Vous devez utiliser une *passphrase* : tâchez de ne pas l'oublier ! Une fois la création terminée, vous devez trouver deux fichiers, `id_rsa` et `id_rsa.pub`, dans votre répertoire `~/.ssh/` : vérifiez que c'est bien le cas avant de continuer. Consultez le contenu des fichiers avec la commande `cat` (sans montrer à personne votre clé privée) : vous pouvez constater que les fichiers ne contiennent que des caractères ASCII.

EXERCICE 5 ► **Copie de la clé publique sur les machines distantes**

Normalement, il existe la commande `ssh-copy-id` pour aller placer votre clé publique sur les machines distantes où vous souhaitez aller vous connecter. Malheureusement, on ne peut pas s'en servir sur `linuxetu`, en raison des limitations de `lshell` sur cette machine. Nous allons contourner le problème, en plaçant nous-même votre clé publique dans `~/.ssh/authorized_keys` de votre répertoire personnel sur cette machine.

1) Dans un terminal, depuis votre ordinateur personnel, connectez-vous à `linuxetu` avec `ssh linuxetu`. Sur `linuxetu`, vérifiez que vous avez bien un répertoire `~/.ssh/` avec `ls -ld ~/.ssh`; assurez vous que seul le propriétaire du répertoire peut le lire, l'écrire, et le traverser. Corrigez si nécessaire (déjà fait avant dans le TP).

2) Dans un autre terminal de votre ordinateur personnel, créez le fichier `~/.ssh/authorized_keys` sur `linuxetu` avec la commande :

```
scp ~/.ssh/id_rsa.pub linuxetu:~/.ssh/authorized_keys
```

3) Dans le terminal connecté à `linuxetu`, vérifiez avec `ls -l ~/.ssh/authorized_keys` que le fichier est bien désormais présent dans `~/.ssh/`; si tel est bien le cas, assurez vous que seule le propriétaire du fichier (vous) peut écrire et lire dedans avec `chmod u=rw,g=,o= ~/.ssh/authorized_keys`; vérifiez que les droits sont corrects.

4) Ouvrez un troisième terminal sur votre machine locale, et essayez de vous connecter sur `linuxetu` avec `ssh linuxetu` : logiquement `ssh` va vous demander d'entrer votre *passphrase*, puis vous connecter (si tel n'est pas le cas, il faut revoir les étapes précédentes).

5) Supposons que vous vous êtes bien connecté avec votre *passphrase* : déconnectez-vous, puis reconnectez-vous avec `ssh linuxetu`; logiquement, vous n'avez pas eu à nouveau à entrer votre *passphrase* (si tel est le cas, lisez mais ne faites rien au point suivant).

6) Si `ssh` vous redemande votre *passphrase* à chaque tentative de connexion, cela signifie que vous n'avez pas de programme installé sur votre distribution Linux pour mémoriser votre clé privée le temps de votre session. Dans ce cas :
— entrez la commande *ssh-add* dans un terminal de votre machine locale, puis entrez votre *passphrase*; vous devriez au moins pouvoir vous connectez sans avoir à retaper votre *passphrase* dans ce terminal.
— plus tard, vous chercherez une solution pour installer un « agent ssh » sur votre distribution !

A ce stade, vous pouvez vous devez pouvoir vous connecter sans problème et sans *passphrase* sur *linuxetu* : il est temps de placer votre clé publique dans `~/.ssh/authorized_keys` de `lifasr5-k`. Cette fois-ci, on peut utiliser `ssh-copyid`; en plus, le rebond sur `linuxetu` est déjà configuré : tout cela va bien simplifier les choses !

1) Dans un terminal, depuis votre ordinateur personnel, entrez la commande suivante en remplaçant bien comme toujours `lifasr5-k` par le nom de la machine qui vous est affectée :

```
ssh-copyid lifasr5-k
```

Logiquement, la commande ne doit pas vous demander votre `passphrase` si vous l'avez déjà entrée pour la série de questions précédentes. Par contre, vous allez devoir entrez votre *password* l'accès à `lifasr5-k`.

2) Connectez-vous à `lifasr5-k` : logiquement, ni `passphrase` ni `password` ne sont plus nécessaires !

3) Consulter le contenu du fichier `~/.ssh/authorized_keys` sur `lifasr5-k` avec `cat ~/.ssh/authorized_keys` : retrouvez-vous bien votre clé publique ?