

Utilisation de tmux

Survivre à la déconnexion

Imaginez que vous êtes connecté·e à une machine distante via SSH, que vous êtes en train de faire une mise à jour de cette machine (ou toute opération qu'il ne faut surtout pas interrompre au risque de tout casser) et que soudain la connexion se perd (ce qui est fort possible puisque le protocole IP est sans garantie, et le revendique)...

`tmux` permet d'avoir une session qui reste persistante lorsque la connexion SSH se rompt, et de rattacher cette session une fois qu'on est reconnecté·e à la machine distante :

1. Connectez-vous sur votre conteneur
2. Lancez `tmux` (si `tmux` n'est pas installé, vous avez raté une étape lors des TP précédents, refaites le « TP installer quelques paquets de base sur son conteneur »).
3. Lancez la commande `htop` qui montre l'utilisation des ressources de la machine.
4. Déconnectez-vous salement (par exemple en fermant votre terminal).
5. Reconnectez-vous sur votre conteneur.
6. Ré-attachez votre session précédente avec la commande `tmux attach`.
7. Observez que `htop` tourne toujours.
8. Quittez `htop` (plusieurs possibilités).
9. Désormais, faites un `tmux` sur vos machines distantes avant de les mettre à jour.
10. Cependant, ne laissez pas un `tmux` tourner inutilement si vous n'avez pas d'opérations en cours lorsque vous vous déconnectez.

Plusieurs terminaux dans un seul

`tmux` est aussi bien utile pour avoir plusieurs shells en même temps à travers une seule connexion SSH, comme des onglets :

11. Lancez `tmux` et créez des nouveaux onglets avec `CTRL+b c` et passez d'un onglet aux onglets voisins avec `CTRL+b n` (pour le suivant) et `CTRL+b p` pour le précédent. Pensez `c` comme "create", `n` comme "next", `p` comme "previous". Vous pouvez aussi aller directement à la Nième fenêtre avec `CTRL+b N`.

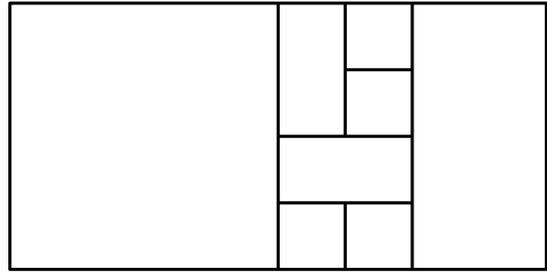
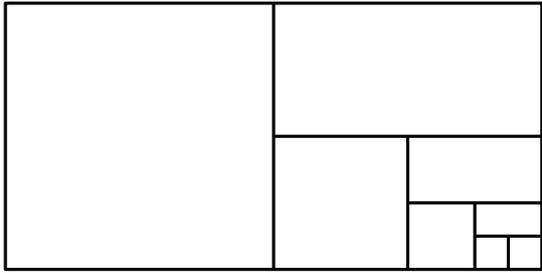
Aller plus loin pour s'amuser (facultatif)

`tmux` peut être utilisé comme un gestionnaire de fenêtres en mode texte.

Pour connaître l'ensemble des raccourcis claviers de `tmux`, vous pouvez regarder le manuel de `tmux` ou faire un `CTRL+b ?` une fois que `tmux` est lancé.

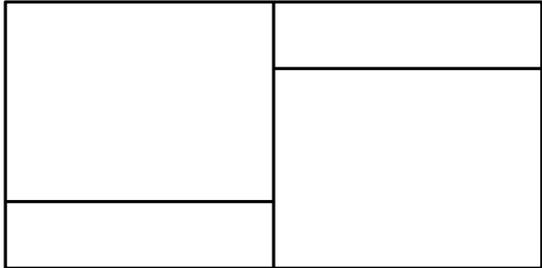
Vous verrez par exemple comment couper (en: split) un terminal en deux verticalement ou horizontalement avec `CTRL+b %` et `CTRL+b "`, et comment passer d'un terminal à l'autre avec `CTRL+b <flèche>`.

12. En découpant les fenêtres en 2, reproduisez les mosaïques suivantes :



Vous verrez aussi comment déplacer les bords à l'aide de combinaisons comme `CRTL+b` `CTRL+<flèche>`.

13. En découpant les fenêtres et puis en déplaçant les bords, reproduisez les mosaïques suivantes :



Plusieurs personnes sur le même terminal

Par ailleurs, `tmux attach` permet à plusieurs personnes différentes d'utiliser le même shell simultanément, ce qui est bien pratique pour s'entraider.

Et en salle TP ?

Pour éviter que des étudiant·es laissent tourner des processus indéfiniment sur les machines de salles TP, un mécanisme fait que lorsque vous vous déconnectez, tous vos processus sont détruits, y compris `tmux`.

Si vous voulez avoir des processus persistants avec `tmux` sur une machine du sercal, suivez la page https://si-galilee.univ-paris13.fr/index.php?title=Processus_persistants (`screen` peut être considéré comme un ancêtre de `tmux`).

Objectifs du TP :

- objectifs opérationnels du TP :
 - ne pas planter ses prochaines mises à jour
- objectifs pédagogiques du TP :
 - se mettre bien