

# Outils libres côté client

BOFFI Aurélien  
SCHWARTZ Nicolas

13 février 2022

## Table des matières

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>TP1 - Efficacité de l'environnement</b> | <b>2</b> |
| 1.1      | .....                                      | 2        |
| 1.2      | .....                                      | 2        |
| 1.3      | .....                                      | 2        |
| 1.4      | .....                                      | 3        |
| 1.5      | .....                                      | 3        |
| 1.6      | .....                                      | 3        |
| 1.7      | .....                                      | 4        |
| 1.8      | .....                                      | 4        |
| 1.9      | .....                                      | 5        |
| <b>2</b> | <b>TP2 - SSH</b>                           | <b>6</b> |
| 2.1      | .....                                      | 6        |
| 2.2      | .....                                      | 6        |
| 2.3      | .....                                      | 6        |
| 2.4      | .....                                      | 6        |
| 2.5      | .....                                      | 6        |
| 2.6      | .....                                      | 6        |
| 2.7      | .....                                      | 7        |
| <b>3</b> | <b>TP3 - GIT</b>                           | <b>8</b> |
| 3.1      | .....                                      | 8        |
| 3.2      | .....                                      | 8        |
| 3.3      | .....                                      | 8        |

# 1 TP1 - Efficacité de l'environnement

## 1.1

| Description                              | Raccourci        |
|--|------------------|
| Discord changer de serveur               | CTRL + tab       |
| Brave changer d'onglet                   | CTRL + tab       |
| Brave nouvel onglet                      | CTRL + t         |
| Brave nouvelle fenetre                   | CTRL + n         |
| Terminator diviser écran verticalement   | CTRL + SHIFT + e |
| Terminator diviser écran horizontalement | CTRL + SHIFT + o |

TABLE 1 – Raccourcis.

## 1.2

Le site que nous avons trouvé est le site [Fastfingers](#) qui est assez connu.



FIGURE 1 – Capture d'écran.

## 1.3

— Paramétrer Vim comme éditeur par défaut :

```
% sudo update-alternatives --config editor
Il existe 4 choix pour l'alternative editor (qui fournit /usr/bin/editor).

Sélection  Chemin          Priorité  État
-----
0          /bin/nano             40       mode automatique
1          /bin/nano             40       mode manuel
2          /usr/bin/code         0        mode manuel
* 3        /usr/bin/emacs        0        mode manuel
4          /usr/bin/vim.tiny     15       mode manuel

Appuyez sur <Entrée> pour conserver la valeur par défaut[*] ou choisissez le numéro sélectionné :4
update-alternatives: utilisation de « /usr/bin/vim.tiny » pour fournir « /usr/bin/editor » (editor) en mode manuel
```

FIGURE 2 – Capture d’écran

— `sudo update-alternatives --config editor`  
On choisit ensuite le numéro de Vim et on confirme :

## 1.4

L’history peut dévoiler l’emplacement de fichier sensible, il est donc important de bien le configurer. Afin d’éviter les très nombreux `ls` et `cd` il faut ajouter : Dans le `/.bashrc` : `HISTIGNORE="ls :cd :history"`

## 1.5

Pour la commande `mkcd` : Dans le fichier `/.bashrc` ajouter :

```
function mkcd() { mkdir -p "$@" && cd "$_"; }
alias mkcd="mkcd"
```

Pour le script `git emergency`, voici le code :

```
#!/bin/bash
gitemergency () {
    git add .
    git commit -m "commit d'urgence" $1
    git push
}
```

## 1.6

— Créer un script vide `backup.sh`.  
— Ajouter le script dans `/etc/bash_completion` :

```
#!/bin/bash
_backup() {
    local cur prev opts

    cur="${COMP_WORDS[COMP_CWORD]}"
    prev="${COMP_WORDS[COMP_CWORD-1]}"
```

```

local files=("${cur})*")

case $COMP_CWORD in
  1) opts='getent passwd | cut -d: -f1 ';;
  2) opts="now tonight tomorrow";;
  3) opts="${files[@]}";;
  *);;
esac

COMPREPLY=()
COMPREPLY=( $(compgen -W "$opts" -- "${cur}") )
return 0
}
complete -o nospace -F _backup backup

```

## 1.7

- Pour ajouter le plugin vagrant prompt nous devons nous rendre dans le fichier de notre theme.
- Les themes sont dans le dossier `/oh-my-zsh/themes`.
- Une fois dedans nous devons rajouter les lignes suivantes :

```

# Oh-My-Zsh Themes

if [[ $TERM != "dumb" ]] || [[ $DISABLE_LS_COLORS != "true" ]]; then
  PROMPT="%{fg:red}%n%{$reset_color%}%n%{$fg:magenta}%n%{$reset_color%}%n%{$fg:blue}%n%{$reset_color%}%n%{$prompt_info%} %{$vagrant_prompt_info%}"
fi

ZSH_THEME_GIT_PROMPT_PREFIX="%{fg:green}%n"
ZSH_THEME_GIT_PROMPT_SUFFIX="%{$reset_color%}"
ZSH_THEME_GIT_PROMPT_DIRTY=""
ZSH_THEME_GIT_PROMPT_CLEAN=""

# display exitcode on the right when =0
return_code="%{fg:red}%n%{$reset_color%}"

RPROMPT="%{$return_code%}${git_prompt_status}%{$reset_color%}"

ZSH_THEME_GIT_PROMPT_ADDED="%{fg:green}%n"
ZSH_THEME_GIT_PROMPT_MODIFIED="%{fg:green}%n"
ZSH_THEME_GIT_PROMPT_DELETED="%{fg:red}%n"
ZSH_THEME_GIT_PROMPT_RENAMED="%{fg:magenta}%n"
ZSH_THEME_GIT_PROMPT_UNMERGED="%{fg:yellow}%n"
ZSH_THEME_GIT_PROMPT_UNTRACKED="%{fg:cyan}%n"

git
PROMPT="%{$prompt%}%n%{$git_prompt_info%}"

ZSH_THEME_GIT_PROMPT_PREFIX=""
ZSH_THEME_GIT_PROMPT_SUFFIX=""
ZSH_THEME_GIT_PROMPT_DIRTY=""
ZSH_THEME_GIT_PROMPT_CLEAN=""

# display exitcode on the right when =0
return_code="%{fg:red}%n%{$reset_color%}"

RPROMPT="%{$return_code%}${git_prompt_status}"

ZSH_THEME_GIT_PROMPT_ADDED=""
ZSH_THEME_GIT_PROMPT_MODIFIED=""
ZSH_THEME_GIT_PROMPT_DELETED=""
ZSH_THEME_GIT_PROMPT_RENAMED=""
ZSH_THEME_GIT_PROMPT_UNMERGED=""
ZSH_THEME_GIT_PROMPT_UNTRACKED=""

fi

ZSH_THEME_VAGRANT_PROMPT_PREFIX="%{fg:bold:blue}%n"
ZSH_THEME_VAGRANT_PROMPT_SUFFIX="%{fg:bold:blue}%n%{$reset_color%}"
ZSH_THEME_VAGRANT_PROMPT_RUNNING="%{fg:bold:green}%n"
ZSH_THEME_VAGRANT_PROMPT_POWEROFF="%{fg:bold:red}%n"
ZSH_THEME_VAGRANT_PROMPT_SUSPENDED="%{fg:bold:yellow}%n"
ZSH_THEME_VAGRANT_PROMPT_NOT_CREATED="%{fg:bold:white}%n"

```

FIGURE 3 – Capture d'écran.

## 1.8

- Code à ajouter dans le fichier `/.zshrc`
- ```

double-up-history() {

```

```

    apache=$(service --status-all | grep apache2 | cut -c4)
    echo $apache
    if [[ $apache == "+" ]]
    then
        echo "apache va se stop"
        sudo service apache2 stop
    else
        echo " apache va se lancer"
        sudo service apache2 start
    fi
}

zle -N double-up-history
bindkey "\C-A" double-up-history

```

## 1.9

- Terminaux testé : Terminator, kitty, coolretroterminal
- Terminator : plusieurs onglets ouverts assez facilement + performant
- Kitty : rapide
- CoolRetroTerminal : difficile a utiliser en raison de son interface

## 2 TP2 - SSH

### 2.1

- `ssh alice@192.168.56.3`
- `ssh carole@192.168.56.3`
- `ssh bob@192.168.56.3`

Les history sont vides car c'est la première fois que les machines sont utilisés.

### 2.2

- `ssh-keygen`
- `ssh-copy-id alice@192.168.56.3`

Pour déposer manuellement :

- `cat /vagrant/id_rsa.pub > ~/.ssh/authorized_keys`

### 2.3

Ajout de l'host bc :

- Host  
    Hostname 192.168.56.3  
    User bob

### 2.4

- Commande :

```
ssh -L 8000:srv.local:80 alice@cli.local
```

### 2.5

- Sur notre machine : `ssh -D 9000 alice@cli.local`
- Editer le fichier `/etc/tsocks.conf`
- Commenter les lignes suivantes et modifier la valeur de `server_port` :

```
#local = 192.168.0.0/255.255.255.0  
#local = 10.0.0.0/255.0.0.0  
server_port = 9000
```

### 2.6

- `ssh bc`
- `apt install x11-apps`
- `apt install xauth`
- `#ForwardX11 yes`
- `ssh -X 192.168.56.3`
- `xeyes`

## 2.7

— Dans `/.ssh/config` :

```
Host srv
  Hostname 192.168.56.3
  User alice
  ProxyJump cli
#Pour ProxyCommand
ProxyCommand ssh cli -W%h:%p
```

## 3 TP3 - GIT

### 3.1

```
[aurelien@aurelienperso:~/Documents/OutilsLibre/Giit on master ]
% git status
Sur la branche master

Aucun commit

Fichiers non suivis:
  (utilisez "git add <fichier>..." pour inclure dans ce qui sera validé)
    Vagrantfile
    srv/

Aucune modification ajoutée à la validation mais des fichiers non suivis sont présents (utilisez "git add" pour les suivre)
```

FIGURE 4 – Capture d'écran.

```
zsh: command not found: vagrant prompt info
[aurelien@aurelienperso:~/Documents/OutilsLibre/Giit on master ]
% git commit -m "MAJ Vagrant"
[master 75059c3] MAJ Vagrant
1 file changed, 1 insertion(+), 1 deletion(-)
```

FIGURE 5 – Capture d'écran.

### 3.2

```
[aurelien@aurelienperso:~/Documents/OutilsLibre/Giit on branche ]
% git add -p Vagrantfile
diff --git a/Vagrantfile b/Vagrantfile
index b8f4404..831ab69 100644
--- a/Vagrantfile
+++ b/Vagrantfile
@@ -12,6 +12,7 @@ Vagrant.configure("2") do |config|
   useradd --shell /bin/bash --create-home alice || true
   useradd --shell /bin/bash --create-home bob || true
   useradd --shell /bin/bash --create-home carol || true
+  useradd --shell /bin/bash --create-home patrick || true
   echo alice:1234 | chpasswd
   echo bob:azerty | chpasswd
   echo carol:secret | chpasswd
(1/2) Stage this hunk [y,n,q,a,d,j,J,g,/,e,]? y
@@ -32,8 +33,8 @@ Vagrant.configure("2") do |config|
   b.vm.hostname = "srv"

   b.vm.provision "shell", inline: <<-SHELL
-   apt-get -y install apache2 ruby
-   a2enmod cgi
+   apt-get -y install apache2 ruby php
+   a2enmod cgi php
   systemctl restart apache2.service
   cp /vagrant/srv/test1.cgi /usr/lib/cgi-bin/test1.cgi
   cp /vagrant/srv/test2.cgi /usr/lib/cgi-bin/test2.cgi
(2/2) Stage this hunk [y,n,q,a,d,K,g,/,e,]? y
```

FIGURE 6 – Capture d'écran.

### 3.3



```

[aurelien@aurelienperso:~/Documents/OutilsLibre/Giit on branche ]
% git commit -m "Ajout de l'user Patrick et activation de PHP"
[branche 45b7c65] Ajout de l'user Patrick et activation de PHP
1 file changed, 3 insertions(+), 2 deletions(-)

```

FIGURE 7 – Capture d'écran.

```

[aurelien@aurelienperso:~/Documents/OutilsLibre/Giit on master ]
% git status
Sur la branche master
rien à valider, la copie de travail est propre

```

FIGURE 8 – Capture d'écran.

```

commit 45b7c6517e47a01ffc97392181746312c222e2bd (HEAD -> master, branche)
Author: Aurélien <you@yourdomain.example.com>
Date: Thu Feb 10 18:19:32 2022 +0100

```

FIGURE 9 – Capture d'écran.

```

[aurelien@aurelienperso:~/Documents/OutilsLibre/Giit on master ]
% git branch -d branche
Branche branche supprimée (précédemment 45b7c65).

```

FIGURE 10 – Capture d'écran.

```

commit 75059c39e4c839f2a430aaf4143299bb0c8bf776 (HEAD -> master)
Author: Aurélien <you@yourdomain.example.com>
Date: Thu Feb 10 18:12:51 2022 +0100

    MAJ Vagrant

commit 58fe50619c841cc2c56da8d783a458ede0321b0c
Author: Aurélien <you@yourdomain.example.com>
Date: Thu Feb 10 18:02:38 2022 +0100

    On ajoute Vagrant
(END)

```

FIGURE 11 – Capture d'écran.

```

[aurelien@aurelienperso:~/Documents/OutilsLibre/Giit on master ]
% git commit -m "Forward port to 8080"
[master 6343f0c] Forward port to 8080
1 file changed, 1 insertion(+), 1 deletion(-)

```

FIGURE 12 – Capture d'écran.

```

[aurelien@aurelienperso:~/Documents/OutilsLibre/Git on forward-new-port ]
% git commit -m "forward to 8081"
[forward-new-port 757e687] forward to 8081
1 file changed, 1 insertion(+), 1 deletion(-)
zsh: command not found: vagrant_prompt_info
[aurelien@aurelienperso:~/Documents/OutilsLibre/Git on forward-new-port ]
% git checkout master
Basculement sur la branche 'master'
zsh: command not found: vagrant_prompt_info
[aurelien@aurelienperso:~/Documents/OutilsLibre/Git on master ]
% git merge forward-new-port
Fusion automatique de Vagrantfile
CONFLIT (contenu) : Conflit de fusion dans Vagrantfile
La fusion automatique a échoué ; réglez les conflits et validez le résultat.
zsh: command not found: vagrant_prompt_info

```

FIGURE 13 – Capture d'écran.

```

[aurelien@aurelienperso:~/Documents/OutilsLibre/Git on master ]
% git status
Sur la branche master
Vous avez des chemins non fusionnés.
(réglez les conflits puis lancez "git commit")
(utilisez "git merge --abort" pour annuler la fusion)

Chemin non fusionné :
(utilisez "git add <fichier>..." pour marquer comme résolu)
modifié des deux côtés : Vagrantfile

aucune modification n'a été ajoutée à la validation (utilisez "git add" ou "git commit -a")
zsh: command not found: vagrant_prompt_info
[aurelien@aurelienperso:~/Documents/OutilsLibre/Git on master ]
% git merge --abort
zsh: command not found: vagrant_prompt_info
[aurelien@aurelienperso:~/Documents/OutilsLibre/Git on master ]
% git status
Sur la branche master
rien à valider, la copie de travail est propre
zsh: command not found: vagrant_prompt_info

```

FIGURE 14 – Capture d'écran.

```

# Et essayez à nouveau.
<<<<<< HEAD
config.vm.network "forwarded_port", guest: 80, host: 8081
=====
config.vm.network "forwarded_port", guest:80, host: 80802
>>>>>> forward

```

FIGURE 15 – Capture d'écran.