

Documentation Raspberry

Rédacteur : Angelo LARIVIERE

Groupe SAE : Quentin Rocher, Khaoula Hajbi, Erwan Barbier, Angelo LARIVIERE

connexion au Raspeberry

Pour ce connecter au raspbeberry nous avons a notre disposiion deux moyens :

Nous pouvons nous connecter garce au serveur titan directement depuis un terminal linux en entrant la ligne de comamnde suivant

```
ssh saepi@192.168.1.171
```

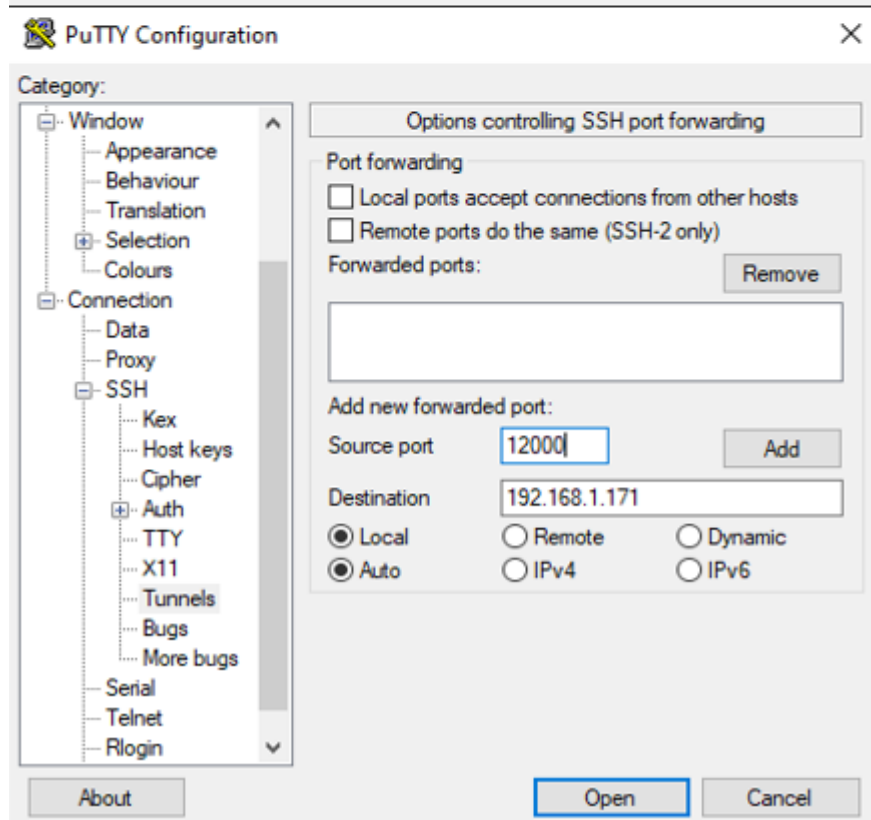
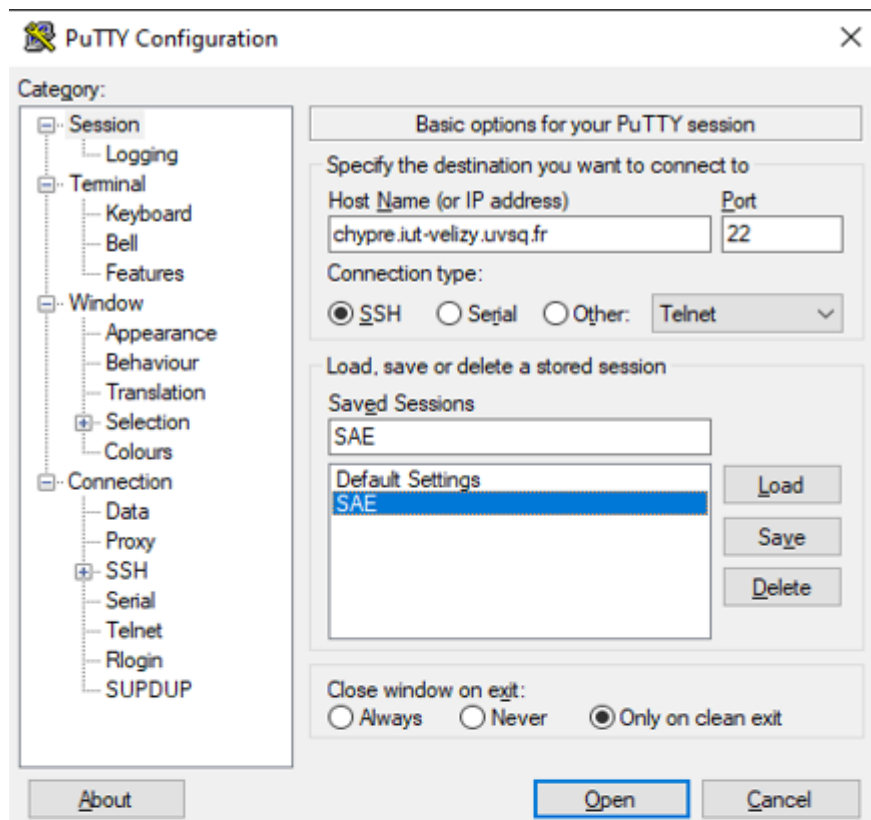
Mais nous avons aussi mit en place un tunnel ssh pour pouvoir nous connecter en dehors du resaux de l'iut pour se fair nous avons installer **MobaXterm** et **Putty**.

Tout d'abord **MobaXterm** qui vas nous servir a nous conecter direcetement a Serveur du Raspebery est d'y accéder par navigateur web.

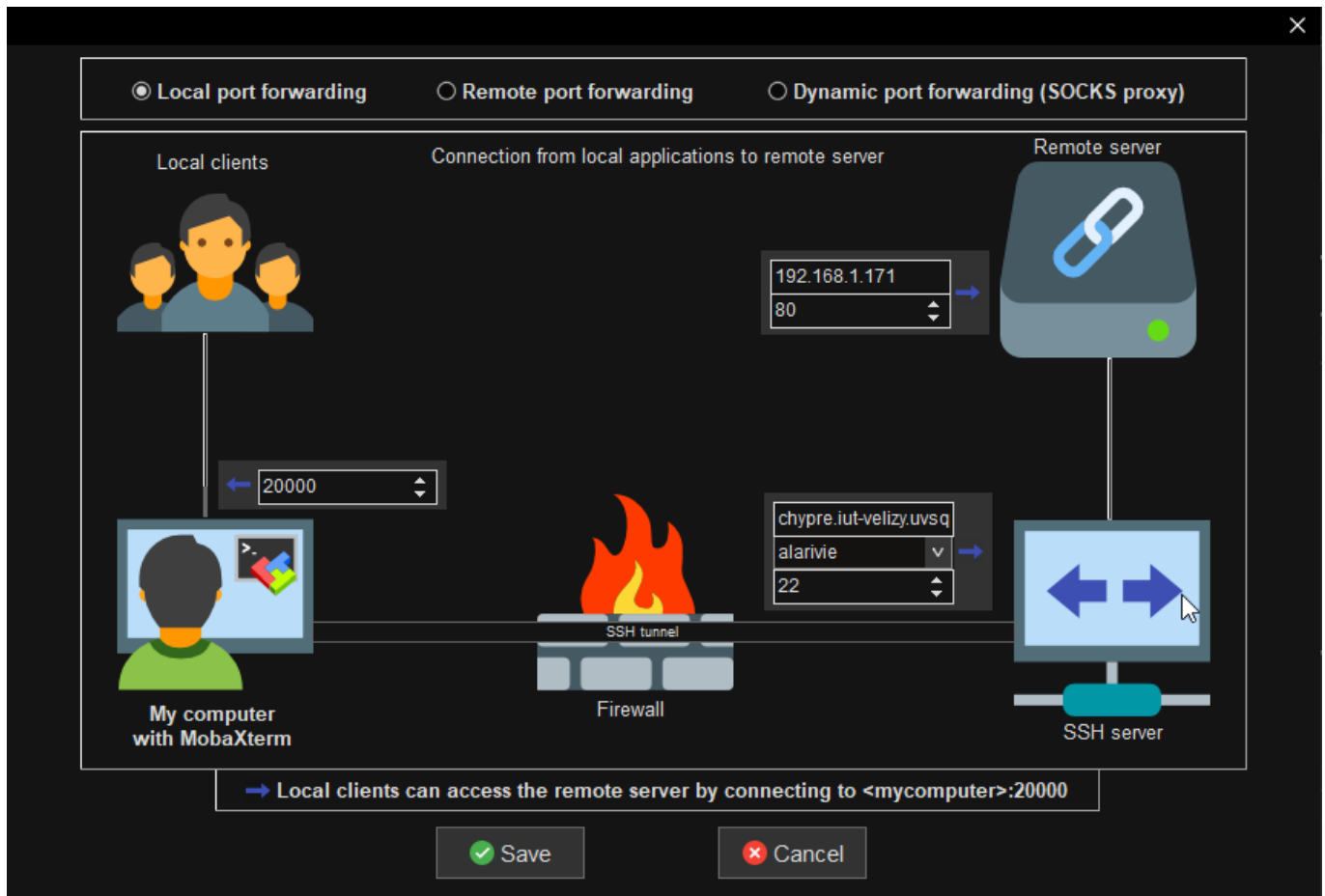
Putty, luis nous permet de nous connecter d'abors a titant par tunnelle ssh puis ensuit on peut reaccéder au raspberri grace a la commande

```
ssh seapi@192.168.1.171
```

Configuration Putty



Configuration MobaXterm



Ici l'utilisation de Putty et de MobaXterm a été choisi car MobaXterm permet d'utiliser putty avec une interface graphique permettant une meilleur faciliter d'utilisation, mais il permet aussi de réaliser un tunnel ssh pour se connecter au raspberry depuis son ordinateur personnel.

Mettre en place les différents serveurs

Pour cette SEA nous avons besoin d'installer deux serveur tout d'abord pour mettre notre serveur qui contientras le site web avec les simulation nous avons choisit d'installer **apache** ; pour stocker la base de donner nous avons choisit d'installer **Mysql**. Mais avant de faire tout les instalations nous devons verifiez que le rasberry est à jour :

Mise a jour du systeme du raspberri

pour mettre a jour le systeme du rasberri nous avons besoin de la commande

```
sudo apt update
```

mais nous avons aussi besoin de la commande

```
sudo apt upgrade
```

apres avoir taper ces commandes on pourras passer aux instalations des serveur sur le rasberri

Mise en place de apache

Pour installer apache sur les rasberrie il faut taper la commande

```
sudo apt install apache2
```

cette commande installera apache sur le rasberri; a la fin de l'execution de la commande nous pourrons configurer le serveur afin d'afficher notre site web

Mise en place de Mysql

pour installer Mysql nous avons besoin d'autre commande permettant l'instalation

```
wget https://dev.mysql.com/get/mysql-apt-config_0.8.22-1_all.deb
```

permetant de récupérer les paquets pourt installer Mysql

```
sudo apt-get install ./mysql-apt-config_0.8.22-1_all.deb
```

puis permettant d'installer les package utile a l'installation de Mysql

on fini par installer Mysql-server nous permettant d'avoir un server utilisant Mysql que nous pourront utiliser avec notre code

```
sudo apt-get install mysql-server
```

suite a cette commande nous aurons acces a des parametre de configuration nous permettant entre autre de créé une base de donnée que nous appelleron **notrebase**, mais aussi un utilisateur **root** nous permettant d'accéder plutard a la fin de la configuration de Mysql

Configuration de apache

Dans notre projet nous avons organise notre site web en plusieurs sous fichier et tout c'est fichier on été déposer sur git et sont donc dans un dépôt; Apache fonctionne grace a un fichier source stocker dans les Dossier `var/www/html` et s'appel `index.html` notre but est donc de remplacer ce fichier par le notre pour que apache démarre sur notre site:

Plusieur problème se sont présenter a nous tout d'abords des sécuriter en peche de déplacer l'emplacement de `index.html` dans notre depot git se trouvant dans `/home/saepi/sae_semestre3_projet/` car `/home` n'apparaît n'est pas sous le même utilisateur, ma seconde idée était donc de mettre le dépôt git dans le dossier `var/www/html` mais des sécuriter empeche de clonner un dépôt dans ce dossier

la solution que nous avons appliquer est de déplacer le dossier de l'application web dans le dossier `var/www/html` avec la commande :

```
cp -r /home/saepi/sae_semestre_projet/ApplicationWeb /var/www/html
```

on noublie pas de supprimer le `index.html` en se placant dans le dossier `var/www/html`

```
sudo rm index.html
```

en ajoutant dans le fichier `000-default.conf` ce trouvant dans le dossier `etc/apache2/site-enable` et en ajoutant de le fichier `apache2.conf` ce trouvant dans le dossier `etc/apache2/` dans ces deux fichiers on ajoute la ligne `/var/www/html/ApplicationWeb`

```
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port th
at
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html/ApplicationWeb

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
~
~
~
~
~
~
~
~
~
"000-default.conf" 31L, 1347B 4,39-46 All
```

```
Include ports.conf
```

```
# Sets the default security model of the Apache2 HTTPD server. It does
# not allow access to the root filesystem outside of /usr/share and /var/www.
# The former is used by web applications packaged in Debian,
# the latter may be used for local directories served by the web server. If
# your system is serving content from a sub-directory in /srv you must allow
# access here, or in any related virtual host.
```

```
<Directory />
    Options FollowSymLinks
    AllowOverride None
    Require all denied
</Directory>
```

```
<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>
```

```
<Directory /var/www/html/ApplicationWeb>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

```
#<Directory /srv/>
#     Options Indexes FollowSymLinks
#     AllowOverride None
#     Require all granted
#</Directory>
```

```
# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives. See also the AllowOverride
# directive.
```

```
#
AccessFileName .htaccess
```

```
-- VISUAL --
```

```
2
```

```
168,13
```

```
80%
```

Configuration de Mysql

la configuration de mysql est plus simple car il suffi de ce connecter au service de mysql :

```
mysql -u root -r
```

on vas par la suite crée un utilisateur *user* avec un mot de passe **test** cela vas nous permettre d'accéder au tables garce a nos script php

```
create user user identified by 'test';
```

on ajoute les droit a l'utilisateur

```
grant all privileges on notrebase.* to 'user';
```

```
flush privileges;
```

suite a cela en executant notre scripte pour crée la base de donne on optient nos tables

```
MariaDB [notrebase]> show tables;
+-----+
| Tables_in_notrebase |
+-----+
| Historique           |
| Log                  |
| Modules              |
| Statistique          |
| Utilisateur          |
+-----+
5 rows in set (0.001 sec)

MariaDB [notrebase]> █
```

et la configuration du raspberry est fini

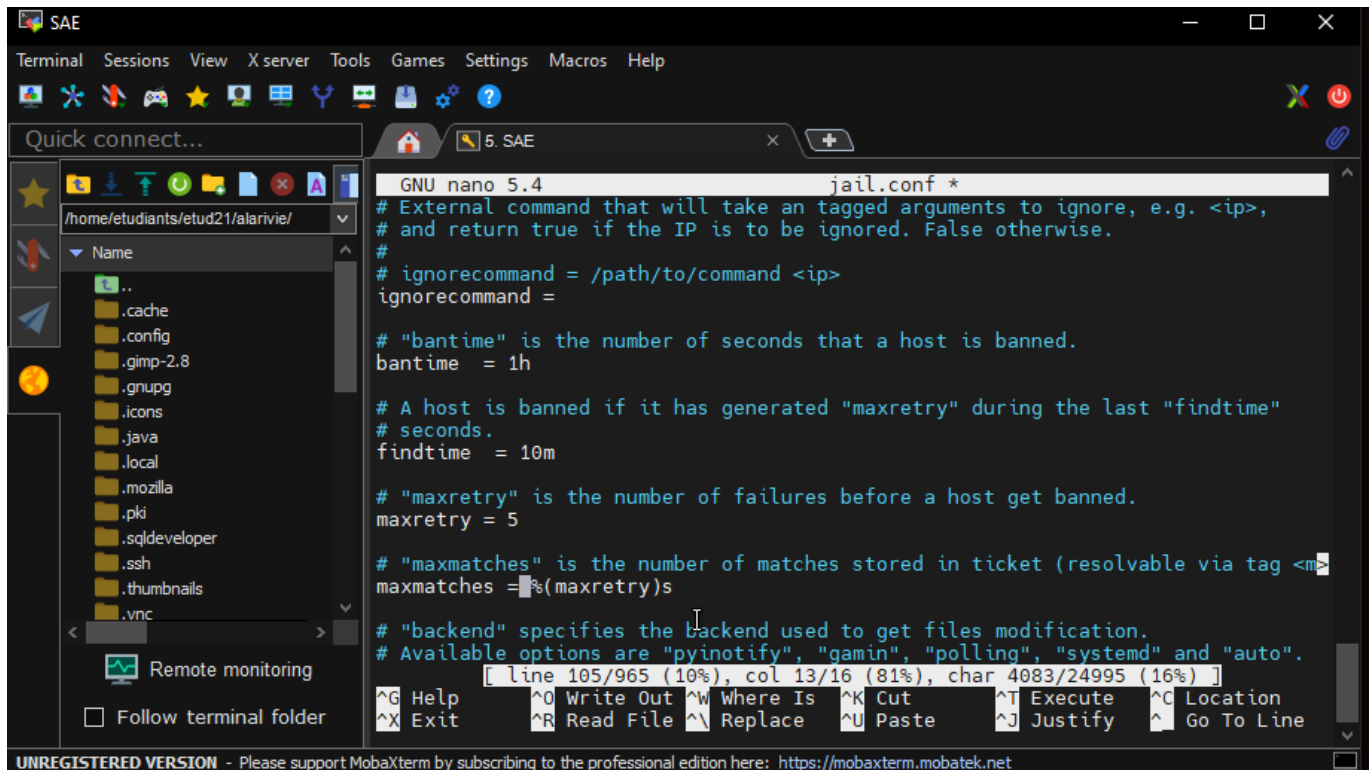
Securisation du Raspberry

Pour la scéruter du raspberry nous avons tous d'abord verifier que le raspberry etait a jours

```
sudo apt update
```

```
sudo apt upgrade
```

suite a sa nous avavons en suite installer un processus permettant de mettre en quarentaine les ip des personne voulant se connecter est c'étant trompé 5 fois.



On peut donc voir dans cette image la plusieurs ligne :

```
bantime = 1h
```

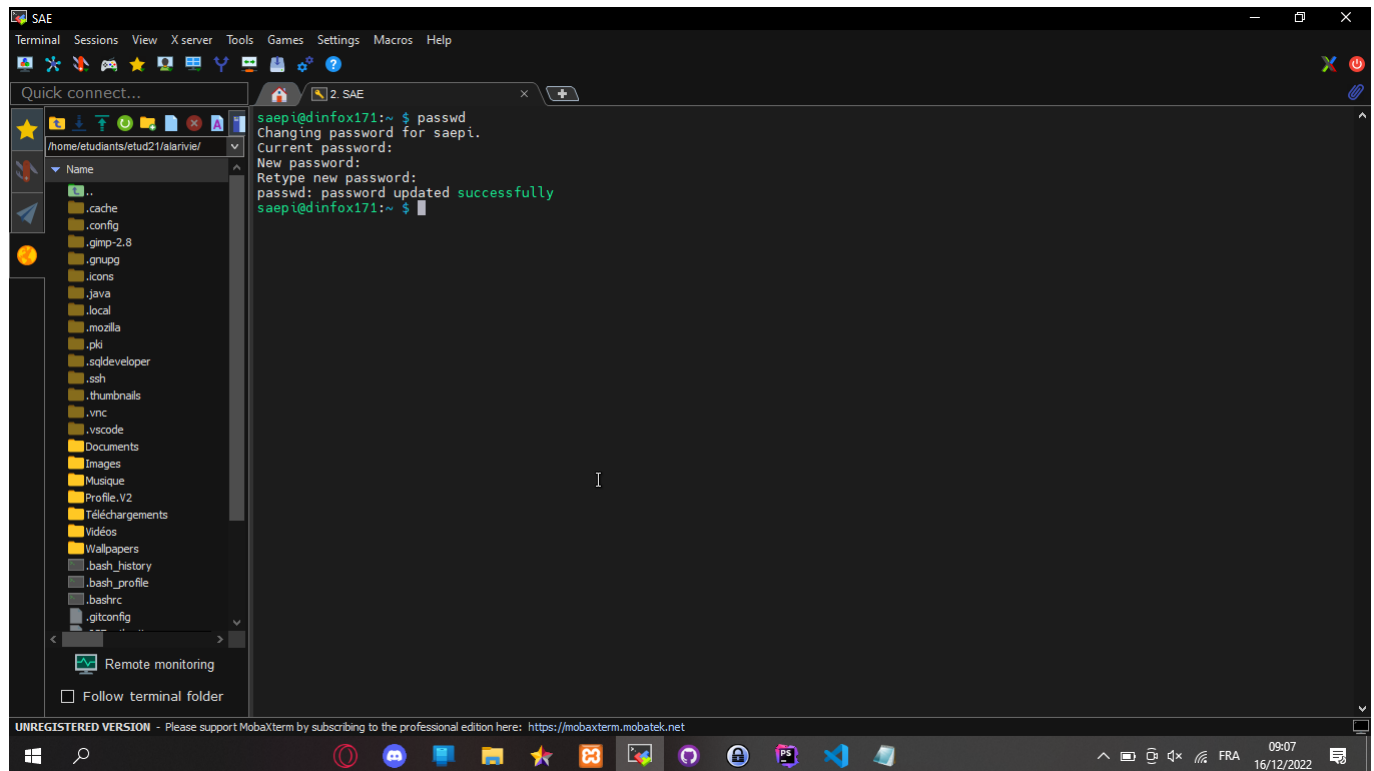
cette ligne sert a definir le temps de ban des ips

Changer le mots de passes

mot de passe de connexion au raspberry

pour modifier le raspberry nous utilisons la commande :

```
passwd
```

The screenshot shows the MobaXterm application window. On the left is a file explorer showing the directory structure of the remote host. The main terminal pane shows a session with the user 'saepi' on the host 'dinfox171'. The user has executed the 'passwd' command to change their password. The output shows the current password being prompted, followed by the new password and its confirmation. The message 'passwd: password updated successfully' is displayed twice. The terminal prompt returns to the user's shell.

```
saepi@dinfox171:~$ passwd
Changing password for saepi.
Current password:
New password:
Retype new password:
passwd: password updated successfully
saepi@dinfox171:~$
```

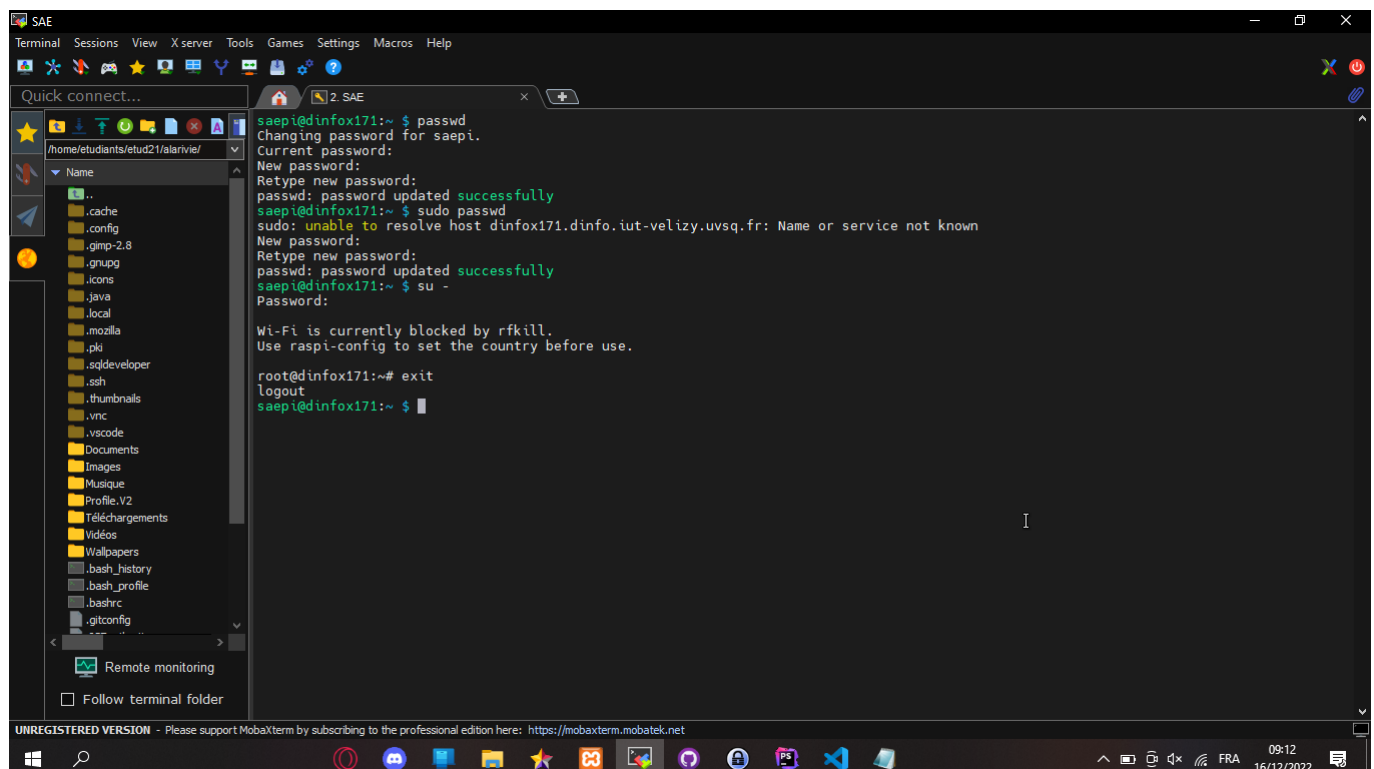
suite a cette commande le nouveaux mot de passe du raspberry :

48h56h

mot de passe root du raspberry

pour le mot de passe de root on tape la commande :

```
sudo passwd
```



The screenshot shows the MobaXterm application window. The terminal pane shows the user 'saepi' executing 'sudo passwd' to change the root password. The output shows the current password being prompted, followed by the new password and its confirmation. The message 'passwd: password updated successfully' is displayed twice. The terminal prompt returns to the user's shell. Below the password change, there is a message about Wi-Fi being blocked by rfkill and a prompt to use 'raspi-config' to set the country before use. The user then enters 'exit' to return to the root shell, followed by 'logout' and 'su -' to return to the user's shell.

```
saepi@dinfox171:~$ sudo passwd
Changing password for saepi.
Current password:
New password:
Retype new password:
passwd: password updated successfully
saepi@dinfox171:~$ sudo passwd
Changing password for saepi.
Current password:
New password:
Retype new password:
passwd: password updated successfully
saepi@dinfox171:~$ su -
Password:

Wi-Fi is currently blocked by rfkill.
Use raspi-config to set the country before use.

root@dinfox171:~# exit
logout
saepi@dinfox171:~$
```

suite a cette commande le nouveaux mot de passe root du raspberry :

~~48-56-~~