|  |
| --- |
| Projet  **Configuration des serveurs DNS, HTTPS, FTP, IMAP, SMTP et LDAP** |

**Réalisé par : Encadré par:**

SAMI Ayoub M. RGHIOUI Anass

EL FAHIM Brahim

SEFFRAOUI Anas

**Année scolaire :** 2023/2024

**Table de matières :**

[**Introduction 3**](#_8w1hcfmpswn9)

[**I. Configuration du DNS 3**](#_rfqw5p6un21g)

[1. Définition 3](#_d364fs4cvac6)

[2. les étapes de la configuration 3](#_vjasycxs0l52)

[**II. Configuration du HTTPS 8**](#_6srwxeg2kz5e)

[1. Définition 8](#_7y8ibivj077q)

[2. Les étapes de la configuration 8](#_yxo5j83gwy2k)

[**III. Configuration du FTP 21**](#_mal166824mz1)

[1. Définition 21](#_lyznq4h7a9l9)

[2. Les étapes de la configuration 21](#_anp1wkuvfgj0)

[**IV. Configuration du SMTP et IMAP 27**](#_so6bvor41021)

[1. Définition 27](#_13o9z7n2synb)

[2. Les étapes de la configuration 27](#_g51d9tamxgex)

[**V. Configuration du LDAP 38**](#_x8ui595vbav6)

[1. Définition 38](#_b9dx2g8cl4cc)

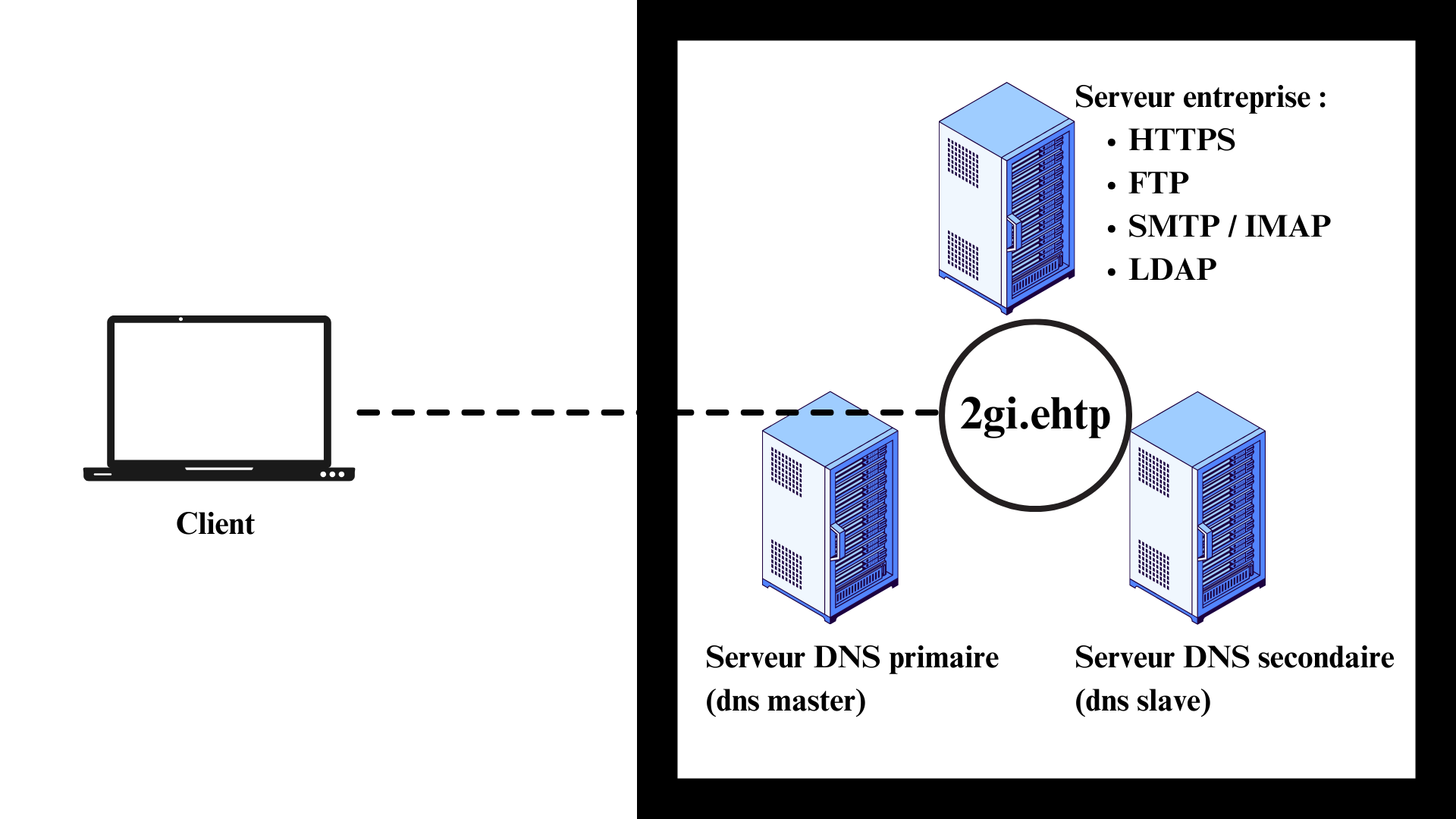
[2. Les étapes de la configuration 39](#_75ivtssv5jc8)

[**Conclusion 44**](#_ausyv4xo943z)

# 

# Introduction

Dans le cadre de notre sous module intitulée « services et protocoles de communication », on est censé configurer les serveurs DNS, HTTP, SMTP, FTP, IMAP et LDAP. Ce rapport englobe les différentes étapes de réalisation et la configuration de ces serveurs.



# Configuration du DNS

## Définition

Le DNS ou le Domain Name System permet de convertir le nom de l' hôte (comme www.exemple.com) en une adresse IP. Les internautes accèdent aux informations en ligne via des noms de domaine tandis que les navigateurs interagissent par le biais d'adresses IP. En résumé, le DNS traduit le nom des domaines en une adresse IP pour que les navigateurs puissent charger les ressources web.

## les étapes de la configuration

On crée deux serveurs DNS : dns-master et dns-slave.

On commence à configurer le dns-master. La commande suivante permet de configurer le nom d’hôte de la machine et par conséquent le nom est devenue « **dnsmaster.2gi.ehtp** » :



Après on déconnecte par la commande : « logout », pour prise en compte la modification du nom d’hôte de la machine. Pour vérifier si la commande a bien été exécutée on va utiliser la commande : « hostname » qui va nous afficher le nom d’hôte de la machine.



Pour le dns-slave, on fait la même chose : On change le nom de la machine par la commande:



Le nom est devenue « dnsslave.2gi.ehtp » :

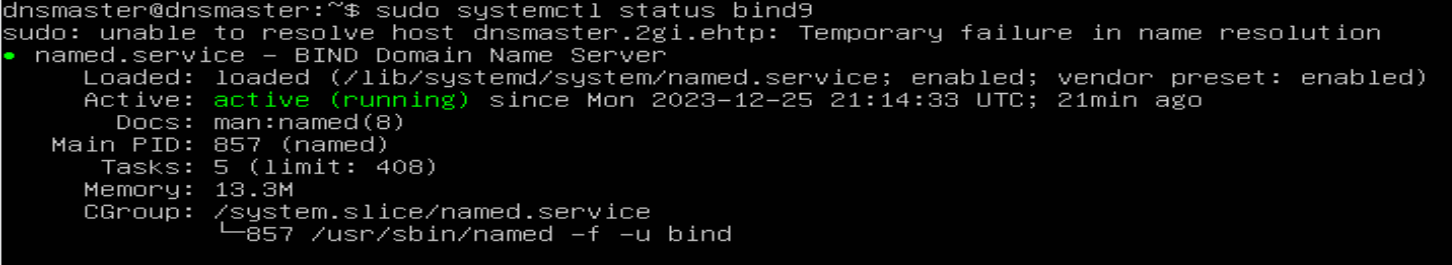


***Pour le dns-master :***

On installe le paquet bind sur le serveur. Le paquet BIND (Berkeley Internet Name Domain), qui est le serveur DNS le plus couramment utilisé sous Linux. On install bind en utilisant la commande :

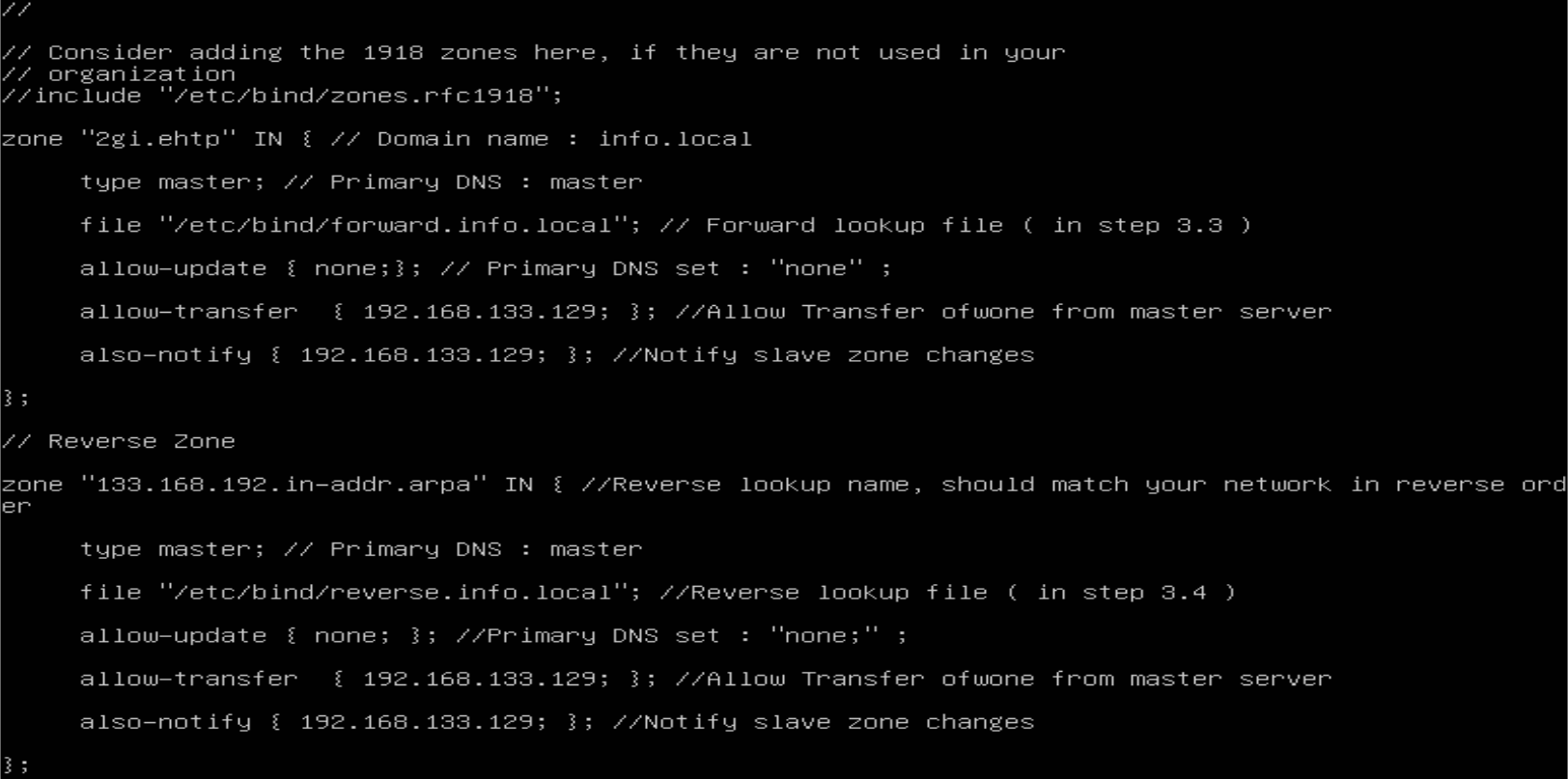


puis on vérifier son statut par la commande :



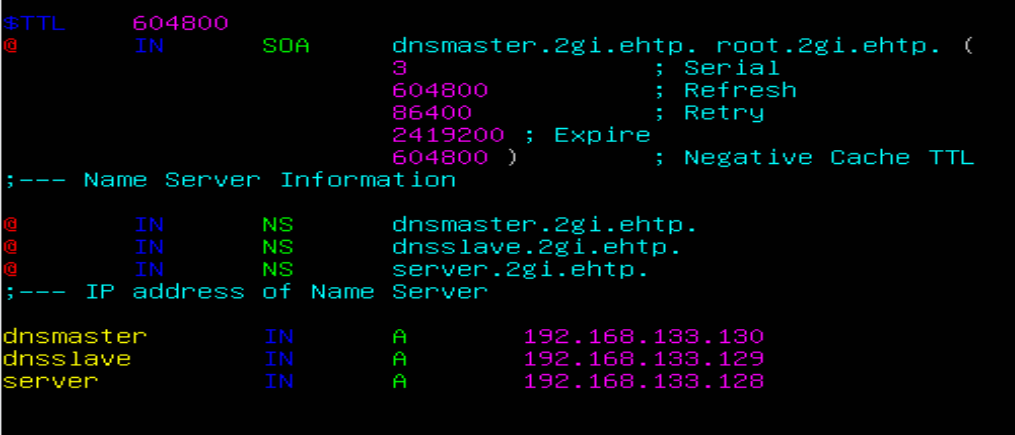
Après, on ajoute dans le fichier /etc/bind/named.conf.local les lignes suivantes:





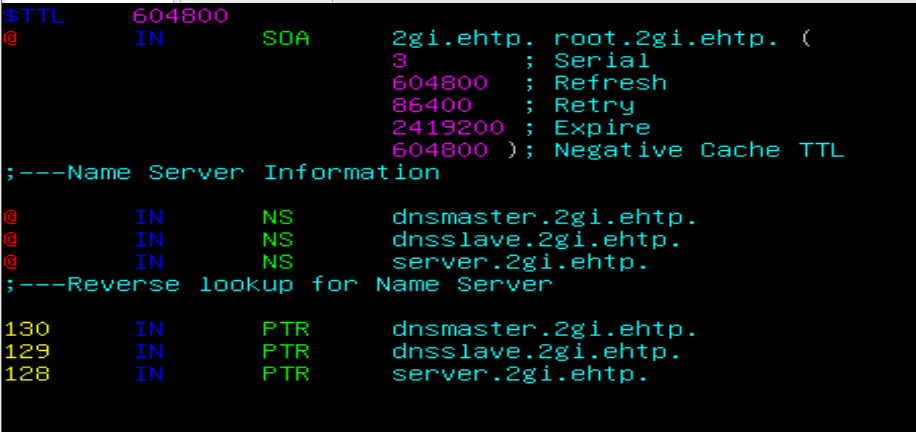
Puis on modifie le fichier /etc/bind/forward.info.local en ajoutant les informations suivantes :



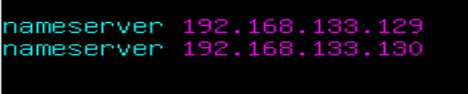


Et on modifie aussi le fichier /etc/bind/reverse.info.local, en ajoutant quelques lignes :





Pour terminer la configuration du serveur dns-master, il faut rajouter dans le fichier /etc/resolv.conf les informations suivantes :

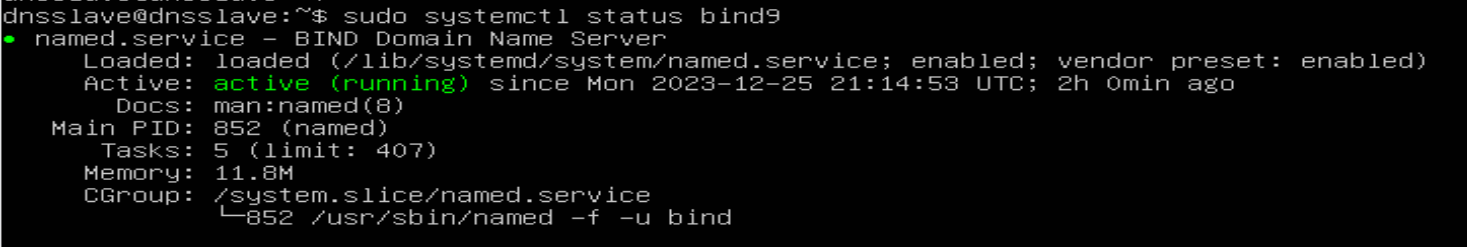


***Pour le dns-slave :***

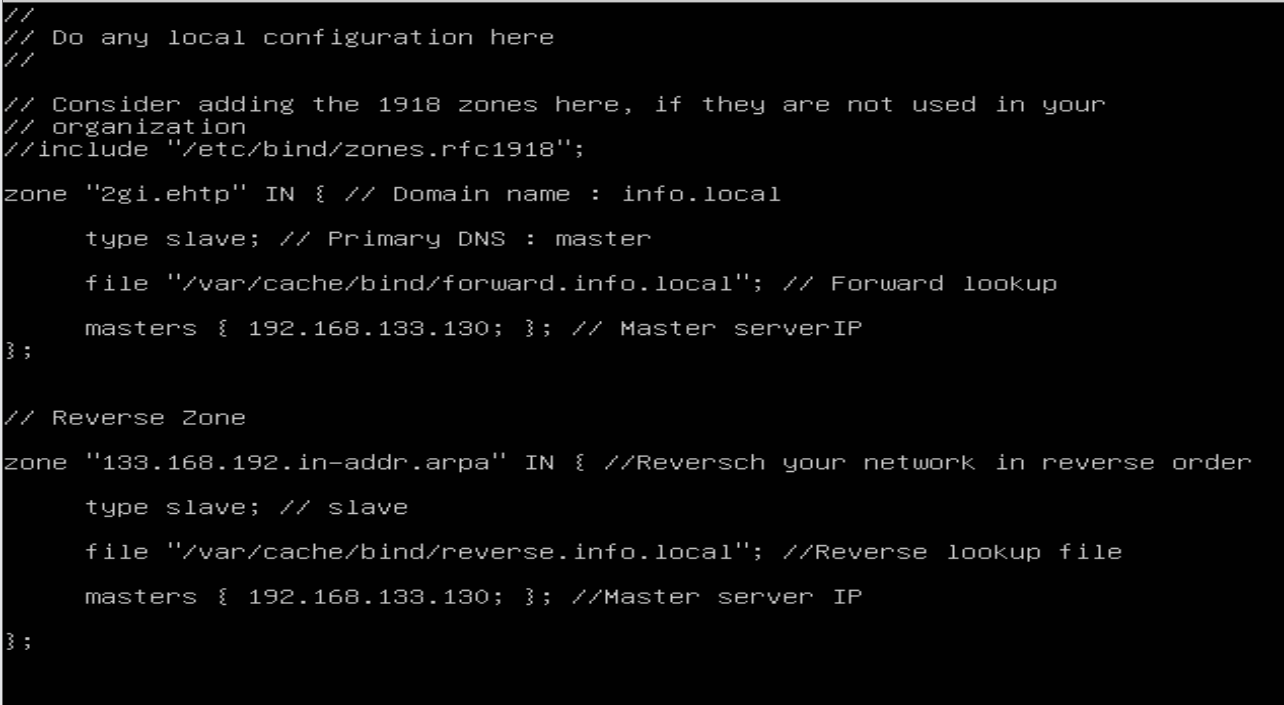
On installe bind9 par la commande :



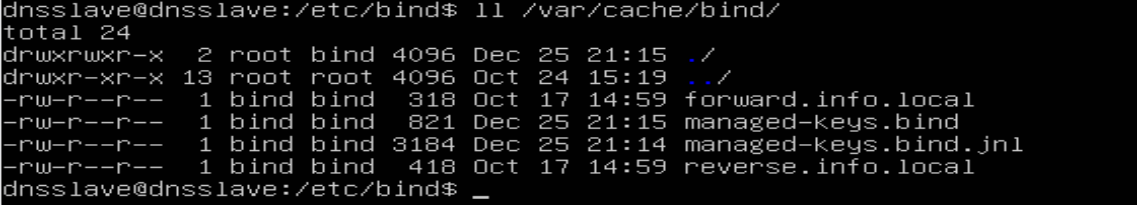
Et on vérifie son statut :



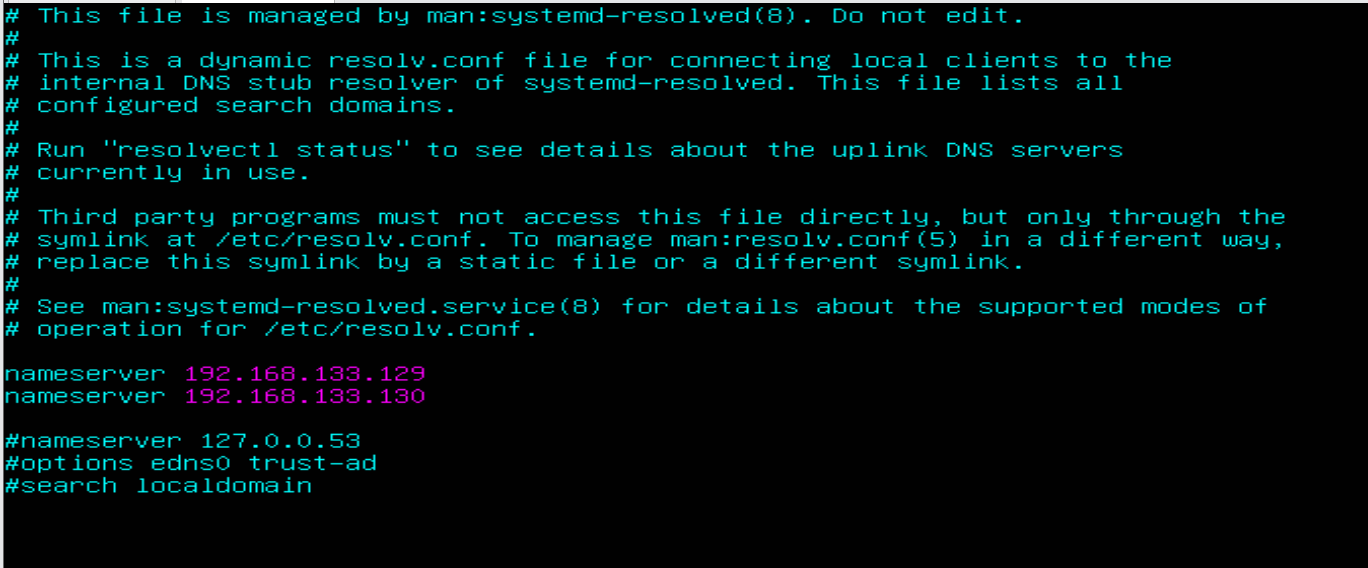
Ensuite, on modifie le fichier /etc/bind/named.conf.local en ajoutant les lignes suivantes :



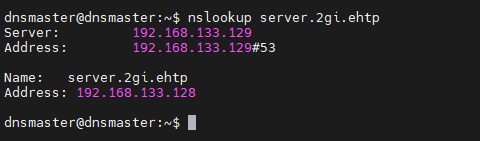
On tape la commande : « ll /var/cache/bind » pour vérifier l’apparition des fichiers forward.info.local et reverse.info.local. S’ils n’apparaissent pas, il faut redémarrer le bind9 par la commande : « sudo systemctl restart bind9 »



On modifiera le fichier /etc/resolv.conf comme pour le serveur dns-master



Après la terminaison de la configuration des deux serveurs, il faut tester la connectivité entre eux par la commande : « nslookup [adresse IP] ».

****

# 

# Configuration du HTTPS

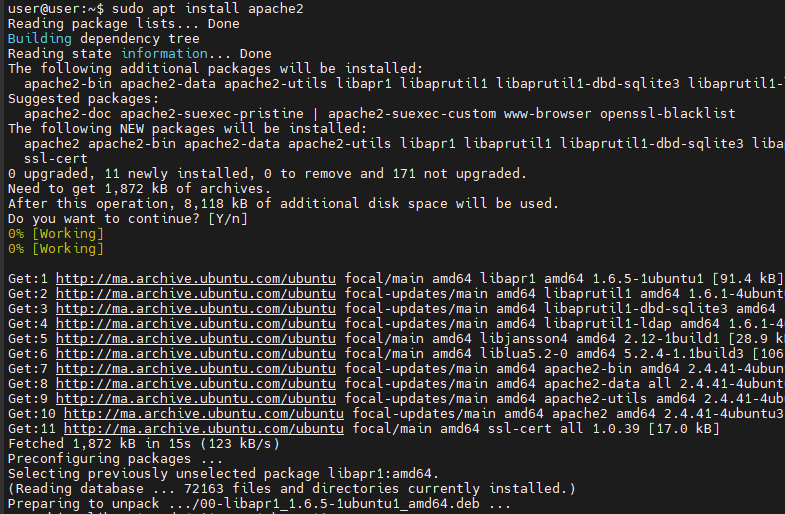
## Définition

Un serveur HTTP (Hypertext Transfer Protocol) est un logiciel qui exécute un serveur Web et qui permet aux utilisateurs de se connecter à un site Web en utilisant un navigateur Web. Lorsqu'un utilisateur tape une URL dans son navigateur, le navigateur envoie une demande HTTP au serveur Web associé à l'adresse IP de l'URL. Le serveur HTTP reçoit la demande et envoie une réponse au navigateur sous la forme d'un fichier HTML (Hypertext Markup Language), qui est le langage utilisé pour afficher les pages Web.

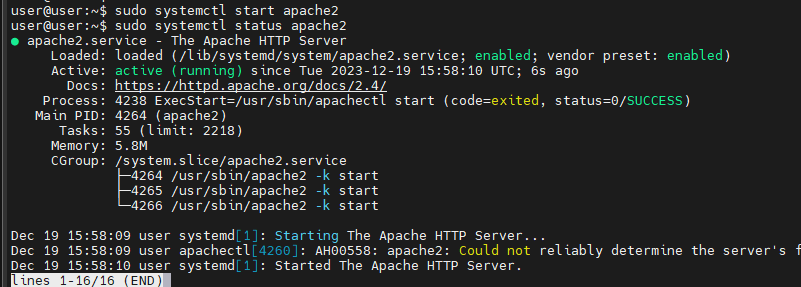
## Les étapes de la configuration

Pour configurer le serveur HTTP on doit commencer par installer les outils apache2. Dans notre configuration nous nous baserons sur le serveur Apache2 pour configurer notre serveur HTTP. Il existe plusieurs autres types de serveurs HTTP tels que NGINX et Microsoft IIS.

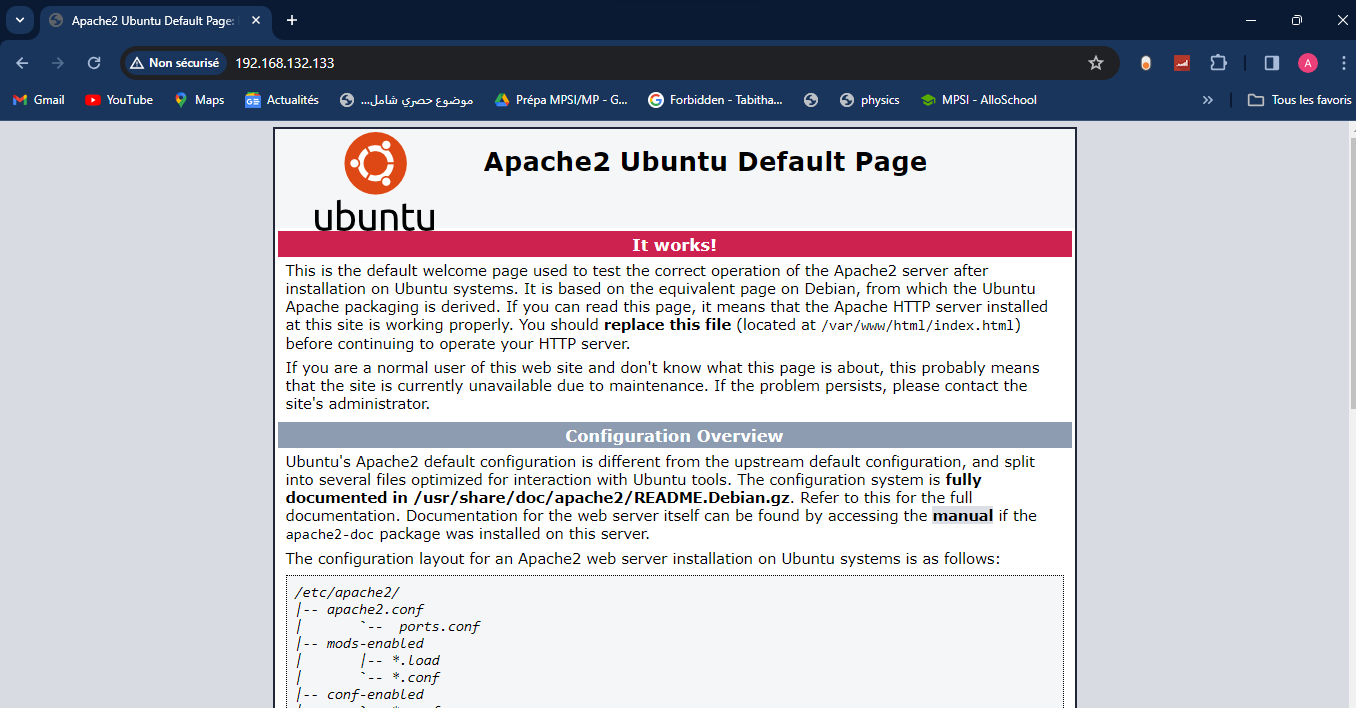
La première commande sera donc :



Par la suite on active le serveur Apache2 et on visualise son statut actuel pour s’assurer que tout marche bien :



On teste par suite cette configuration en tapant dans un navigateur web l’adresse-ip de notre serveur (dans ce cas c’est 192.168.132.133) :



En fait, c’est la page html par défaut du serveur Apache2. Dans le cadre de ce projet, on a pas à la modifier mais si c’est le cas il suffira de modifier dans le code de la page index.html se trouvant dans le dossier /var/www/html.

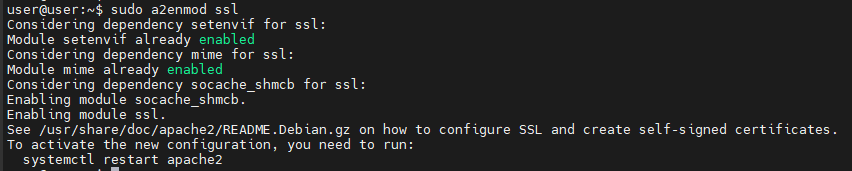
1. **Configuration de l’HTTPS (SSL)**

Selon le cahier de charges de ce projet, on est restreint à ajouter au serveur http un certificat SSL pour réaliser la configuration de l’HTTPS.   
 L’HTTPS en fait, n’est qu’une version sécurisée par le protocole SSL du protocole HTTP. Ceci se réalise en effet en générant pour son nom de domaine une certification de SSL.

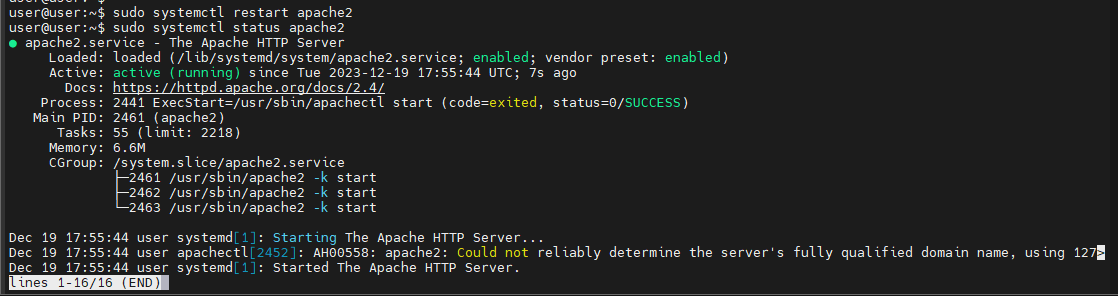
La configuration du protocole HTTPS en suivant l’ancienne configuration du serveur http qu’on vient de réaliser se fait comme suit :

D’abord, on désactive tout firewall afin d’éliminer tout obstacle lié à cette configuration :

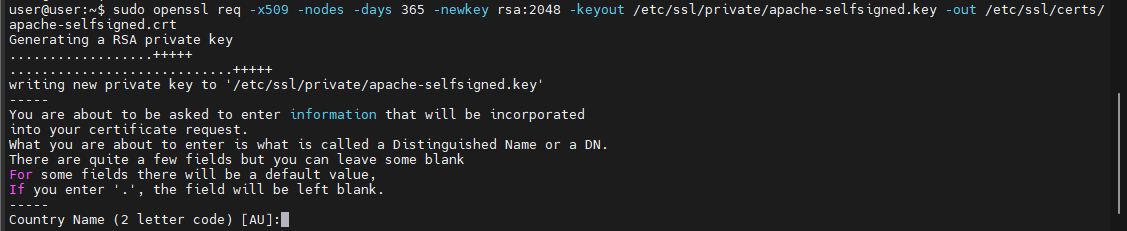


On active par la suite le module SSL pour commencer la configuration de notre certificat : 

Comme est cité dans la réponse donné par le système à cette instruction, il faudra redémarrer le serveur Apache2 :



On pourra après taper la commande suivante de OpenSSL pour générer un certificat auto-signé (self-signed certificate) valable pour 365 jours :

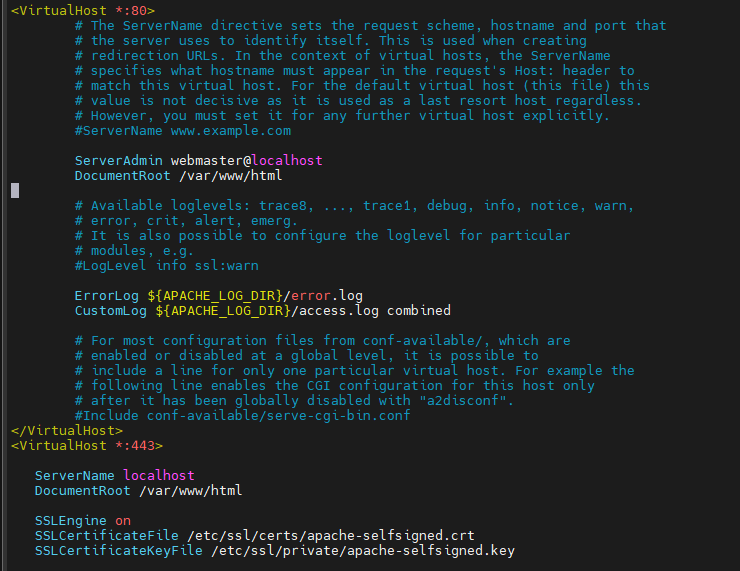


On visualise les fichiers de la directive /etc/apache2 :

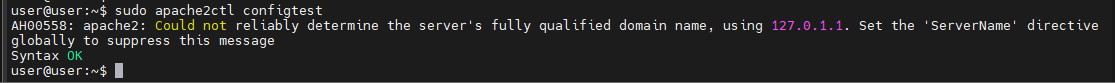


On devra maintenant modifier dans la configuration du site par défaut d’Apache :

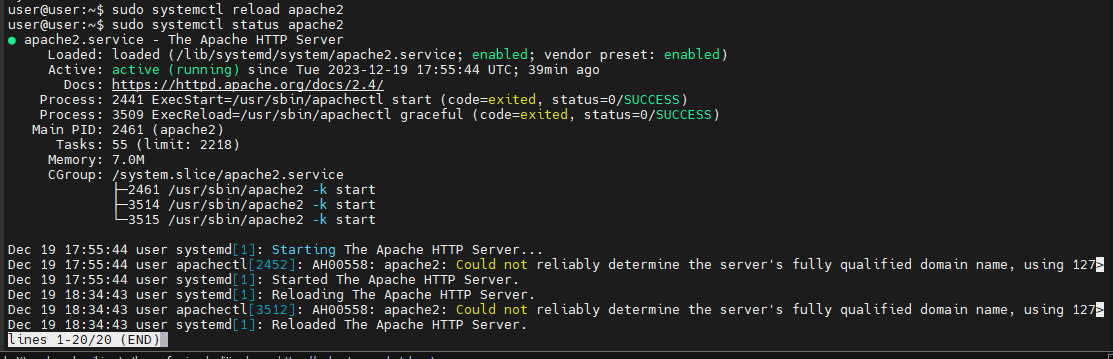


Dans ce fichier, on ajoute la partie associée au HTTPS ayant pour port : 443   


On teste la syntaxe de ce qu’on a modifié avec l’instruction : *sudo apache2ctl configtest*:

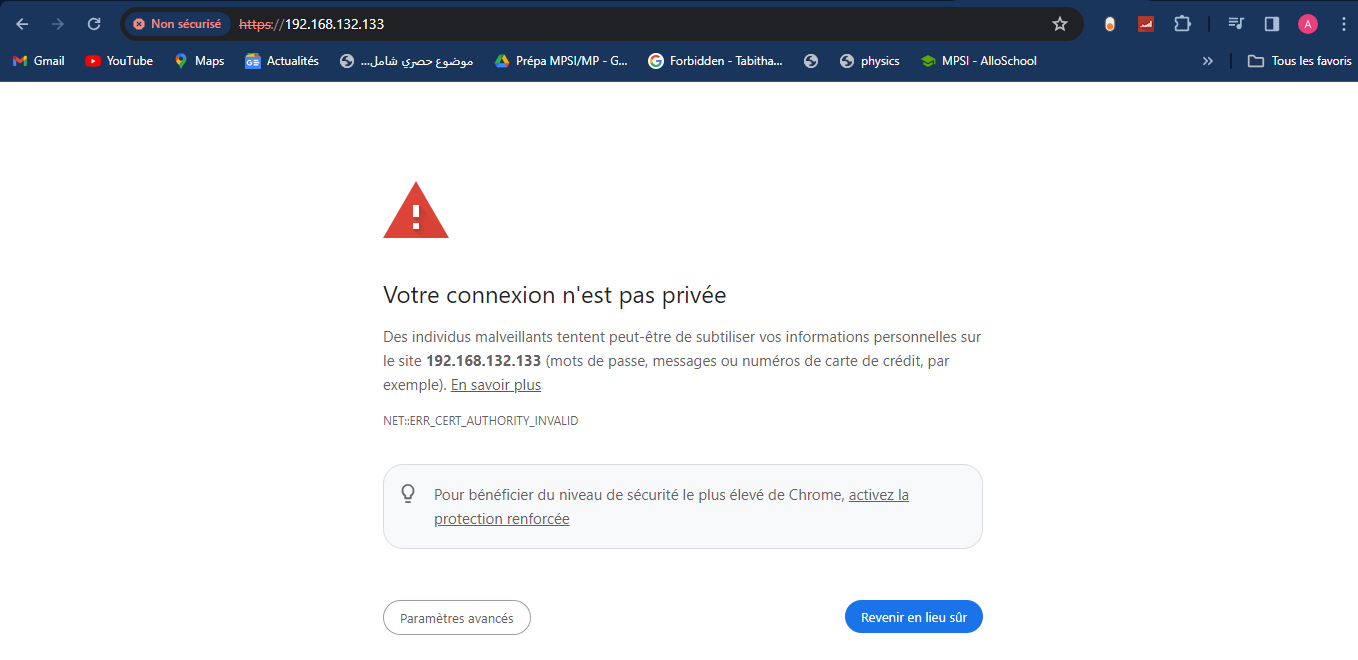


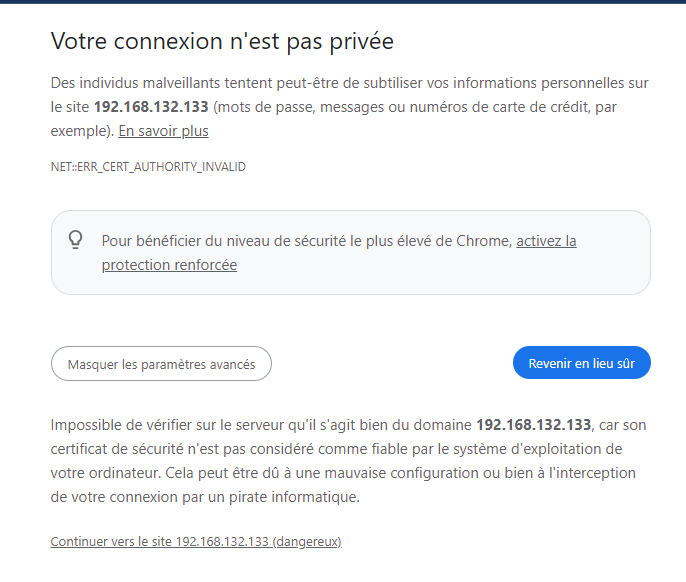
On redémarre et on visualise le statut de Apache2 maintenant :

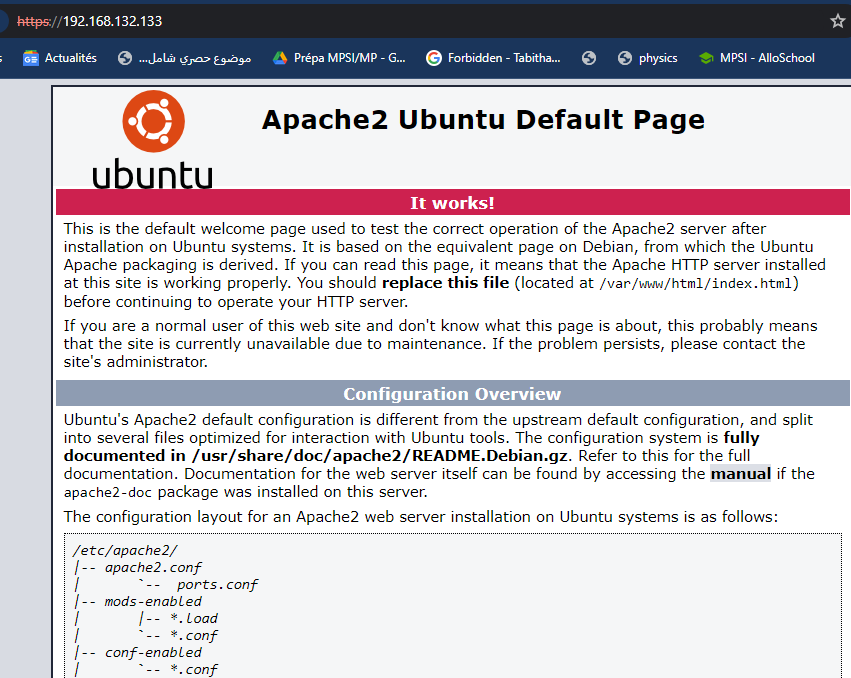


Maintenant pour le vrai test : sur notre navigateur web, on tape [*https://[adresse-ip-serveurhttps*](about:blank)*]*:









En effet, comme notre nom de domaine n’est pas usuel, notre certificat n’est pas reconnu par le navigateur web, qui nous annonce que ce site pourra avoir des soucis de sécurité, mais on peut toujours y accéder en choisissant de continuer vers le site.

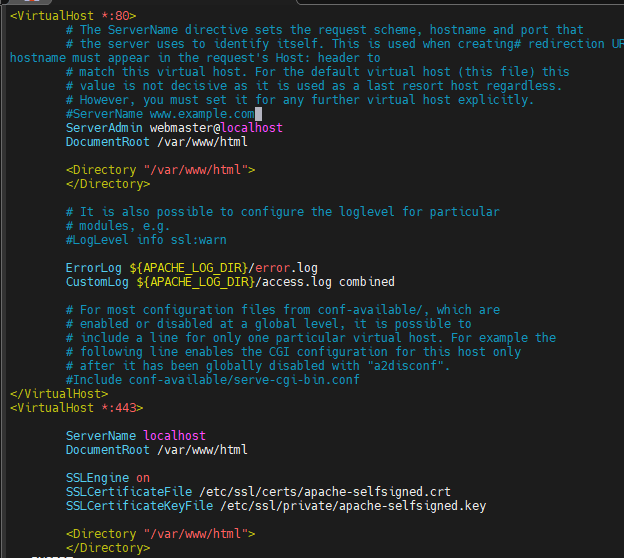
1. **L’authentification**

Pour configurer une authentification à certaines pages de notre site web, on peut suivre la méthodologie suivante :   
 On visualise le username et le password chiffrés à l’aide de la commande :

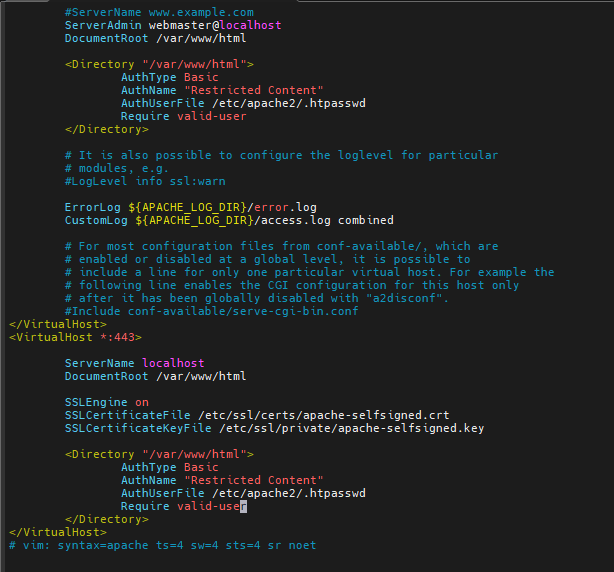


On change par la suite dans la configuration des sites-enabled à travers la commande :

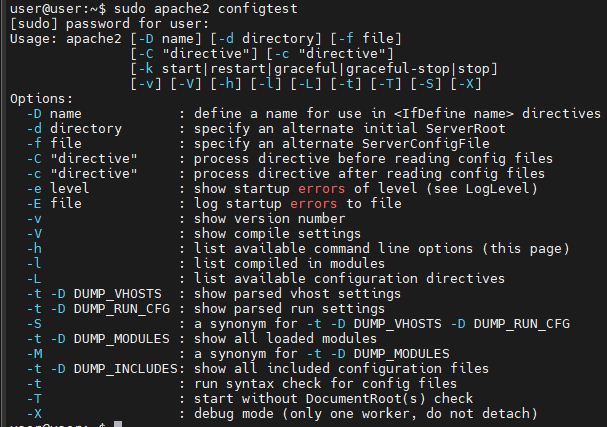


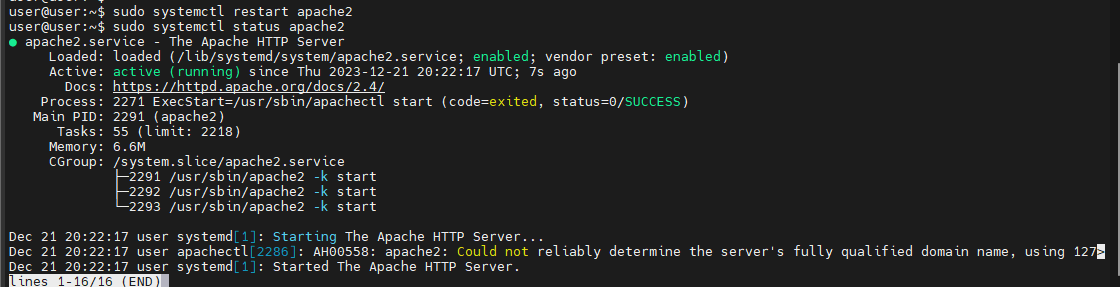


Pour les deux ports de HTTP et HTTPS, on ajoutera une balise Directory qui définit l’ensemble des pages que l’authentification va atteindre (avec le chemin /var/www/html )  et à l’intérieur l’ensemble des paramètres liés à l’authentification :



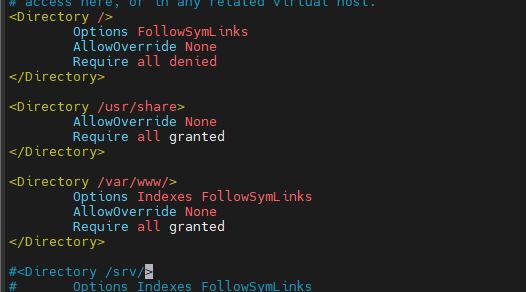
De même on teste pour les erreurs de syntaxe, puis on redémarre le serveur Apache2 :

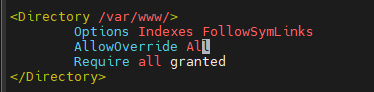




Maintenant pour le fichier de configuration par défaut de Apache :

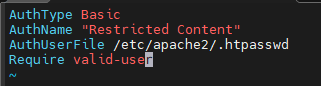




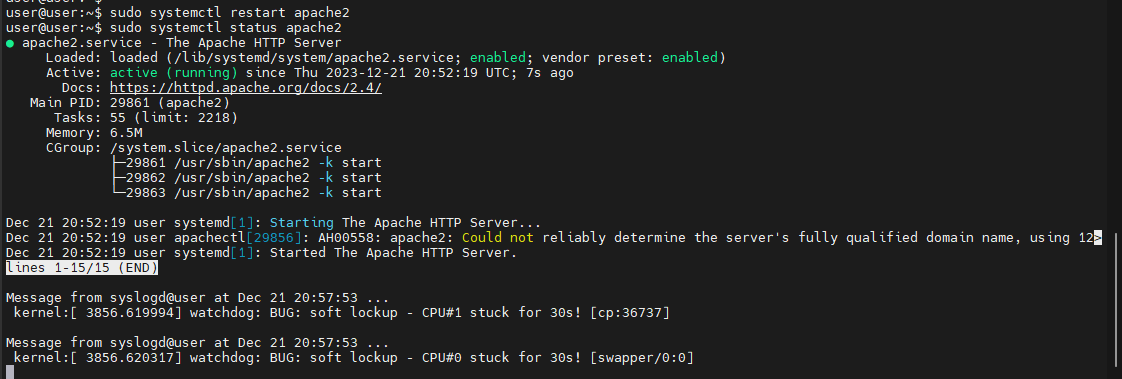


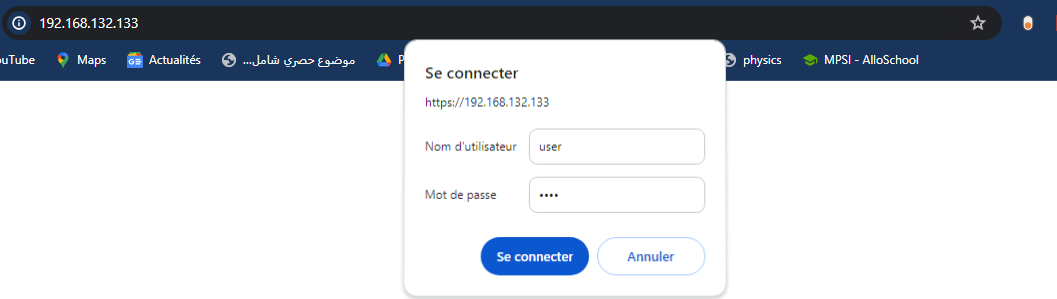
Maintenant, pour les fichiers qu’on veut sécuriser avec une authentification :

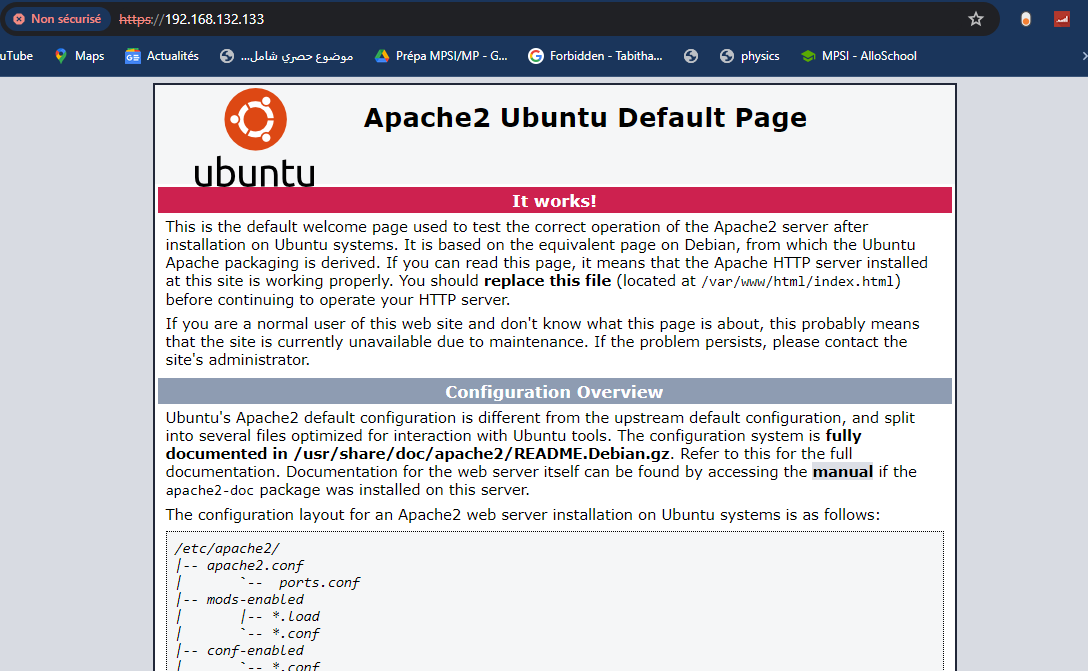




Et on teste :

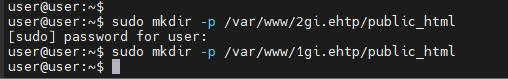




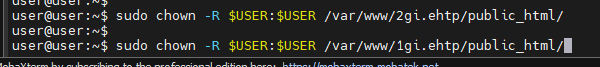


1. **Les Virtual Hosts**

Dans le contexte d’Apache, les Virtual Hosts permettent de configurer plusieurs sites web sur un seul serveur. Les Virtual Hosts sont particulièrement utiles quand on a plusieurs domaines ou sous-domaines qu’on veut héberger sur le même serveur.   
La configuration des Virtual Hosts commence par créer deux sous-directives dans les fichiers html d’Apache :



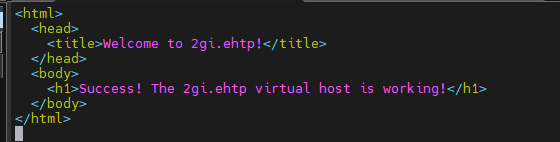
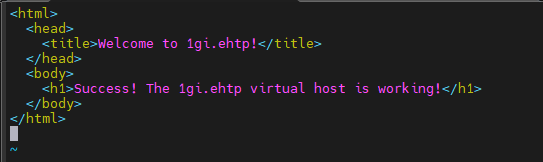
Après on devra donner au user présent les droits d’accès à ces directives :





La commande **sudo chmod -R 775 /var/www** modifie les permissions de manière récursive (**-R**) pour le répertoire **/var/www** et son contenu.ù

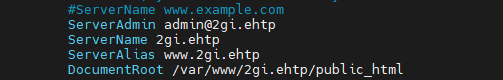
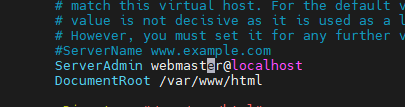
On donnera des pages index.html pour chaque virtual Host pour tester avec un simple code de test :

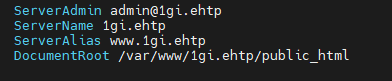


Par suite, on copiera la configuration par défaut des pages existantes déjà sur la configuration des nouvelles pages de virtuals hosts créés :

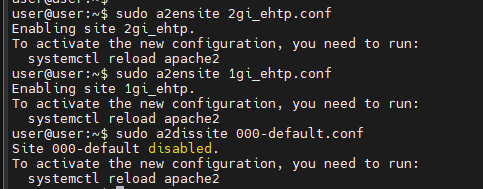


On modifiera dans la configuration de chacun de ces deux fichiers comme suit :

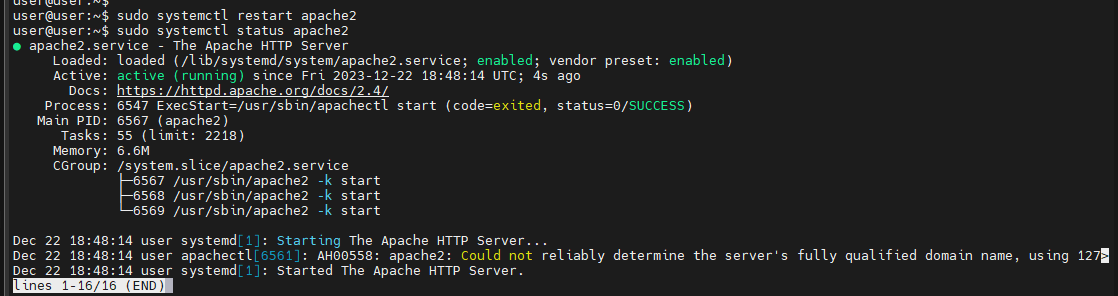
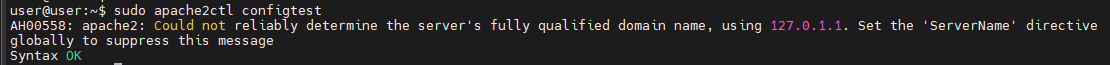




Maintenant, pour l’ancienne configuration du serveur Apache avec la page /var/www/html/index.html, on désactive sa configuration par défaut, et on l’active pour les nouveaux fichiers des virtuals hosts :



Maintenant, on teste la syntaxe et le fonctionnement du serveur Apache :



C’est bon, tout concernant le serveur HTTP et HTTPS est configuré.

# Configuration du FTP

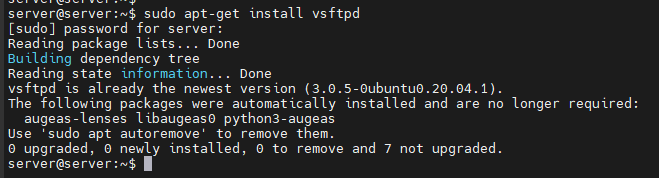
## Définition

FTP est un protocole de communication destiné au partage de fichiers sur un réseau TCP/IP. Il permet, depuis un ordinateur, de copier des fichiers vers un autre ordinateur du réseau, ou encore de supprimer ou de modifier des fichiers sur cet ordinateur.

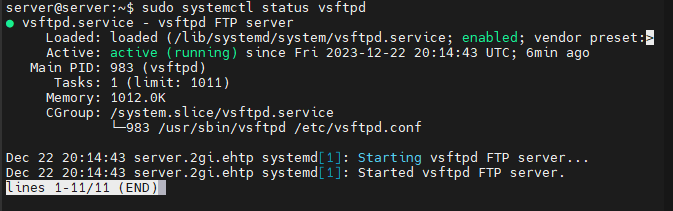
## Les étapes de la configuration

Pour configurer le serveur FTP on doit installer le package VsFTPd. VsFTPd est un serveur FTP qui mise beaucoup sur la sécurité, Contrairement aux autres serveurs FTP (ProFTPd, PureFTPd, etc.), aucune faille majeure de sécurité n'a jamais été décelée dans VsFTPd.

On installe le package comme suit :



Puis on s’assure que le serveur vsftpd est active :



* **La configuration du serveur**

Maintenant en procède la configuration :

L’installation de notre serveur est bien effectuée, maintenant on va commencer de configurer ce dernier, pour cela on va éditer le fichier ‘**vsftpd.conf**’



* On ajoute un message de bienvenue



* On autorise l’accès à notre serveur FTP à certains utilisateurs locaux



* On autorise les commandes d’écriture FTP



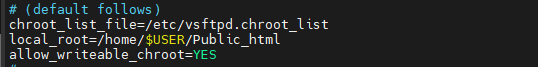
* On désactive le jeu de caractères ASCII



* On modifie les paramètres du chroot.



* On modifie les paramètres chroot par défaut.



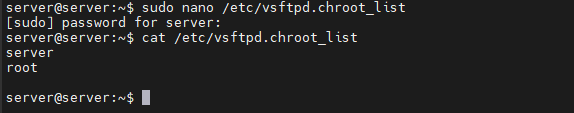
* On active l’option **ls recurse**



Après la configuration de notre serveur FTP on enregistre les modifications.

* **L’ajout des noms d’utilisateurs à la liste CHROOT**

Maintenant on doit ajouter des noms d’utilisateurs des personnes autorisées à partager le contenu du serveur FTP. Pour faire cela on va éditer le fichier “**vsftpd.chroot\_list**“ , et on tape les noms d’utilisateurs acceptés :



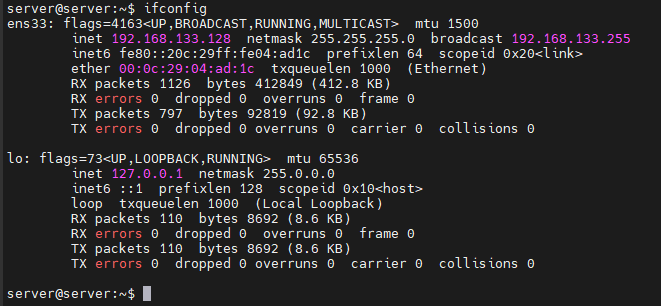
Maintenant on redémarre notre serveur FTP, VSFTPd est alors arrêté, puis redémarré, cette simple procédure permet d’enregistrer les changements faits. À ce stade, nous pouvons accéder à notre serveur FTP.



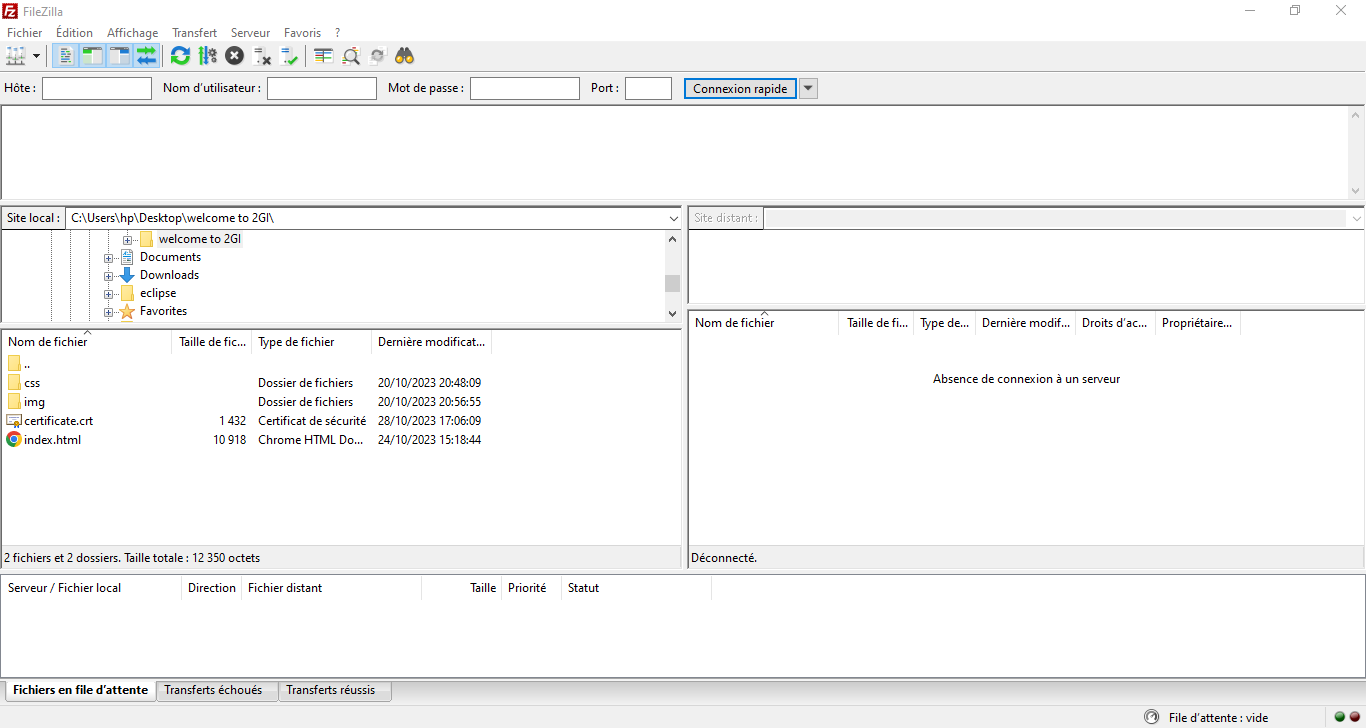
* **L’accès au serveur ftp**

On peut se connecter sur le serveur en entrant les identifiants dans les champs prévus à cet effet sur un client FTP (par exemple avec **FilleZilla**).

On a l’adresse IP du notre serveur ftp est :

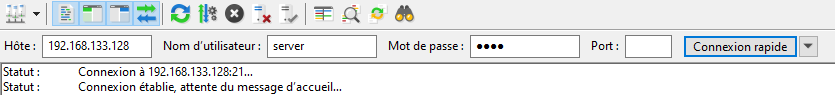


On installe FTP puis on le démarre on trouve sa console comme suit :

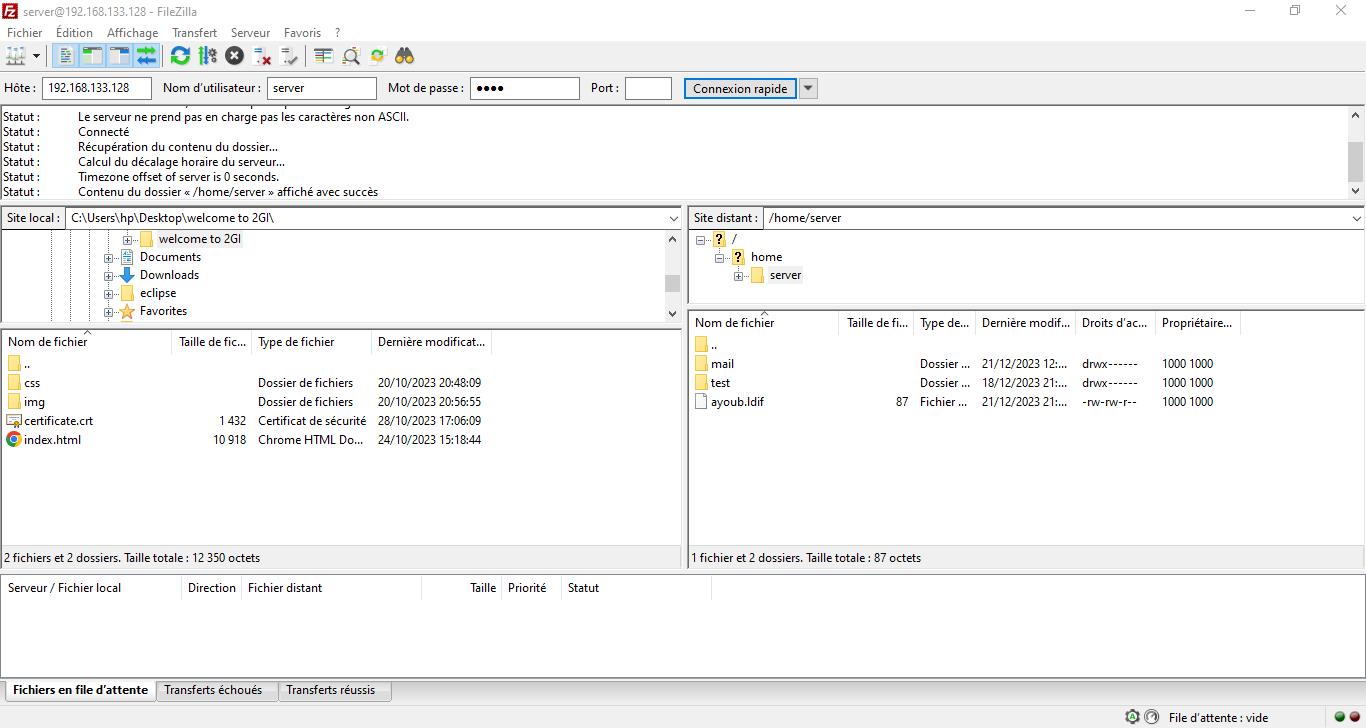


On introduit les identifiants nécessaires pour s’authentifier au serveur, à savoir, l’adresse IP, le mot de passe, le port et le nom d’utilisateur.

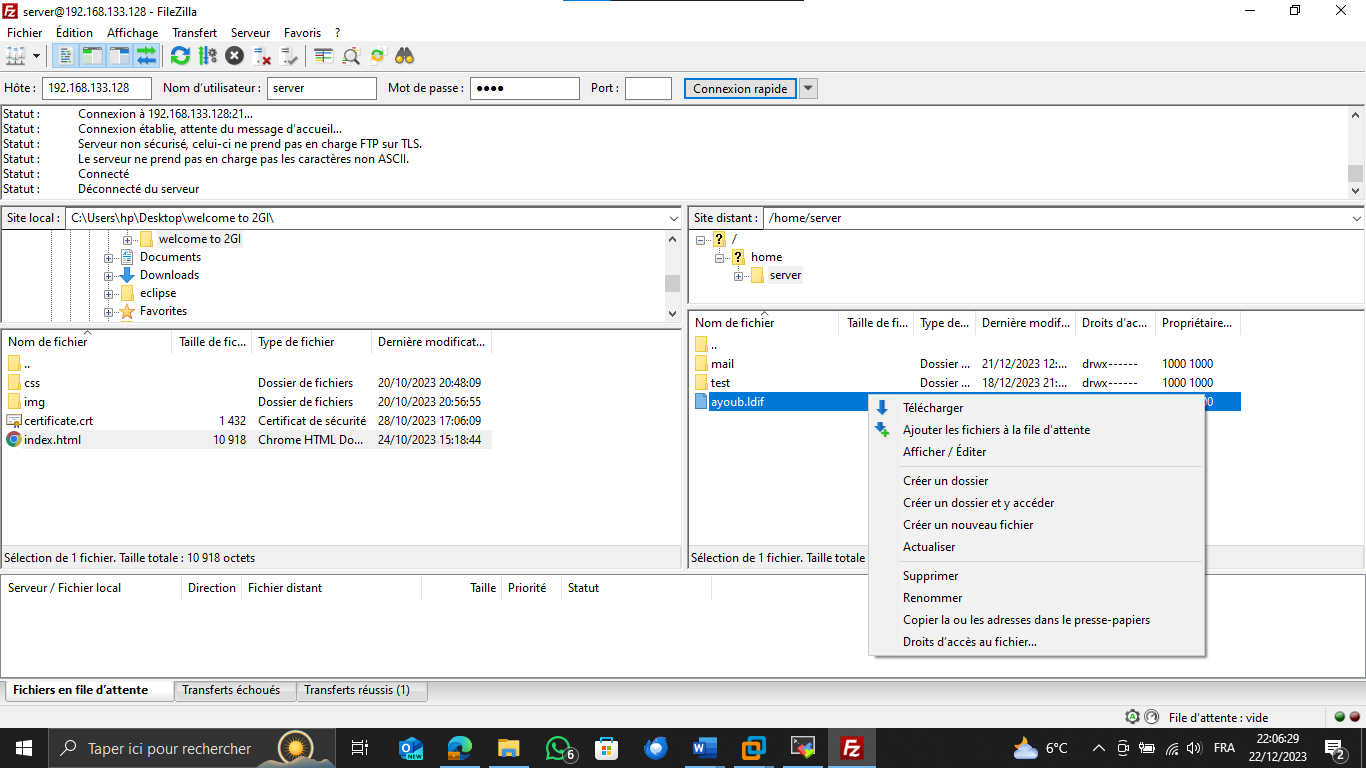
Dans notre cas IP : **192.168.133.128**, nom d’utilisateur : **server et port : 21**

****

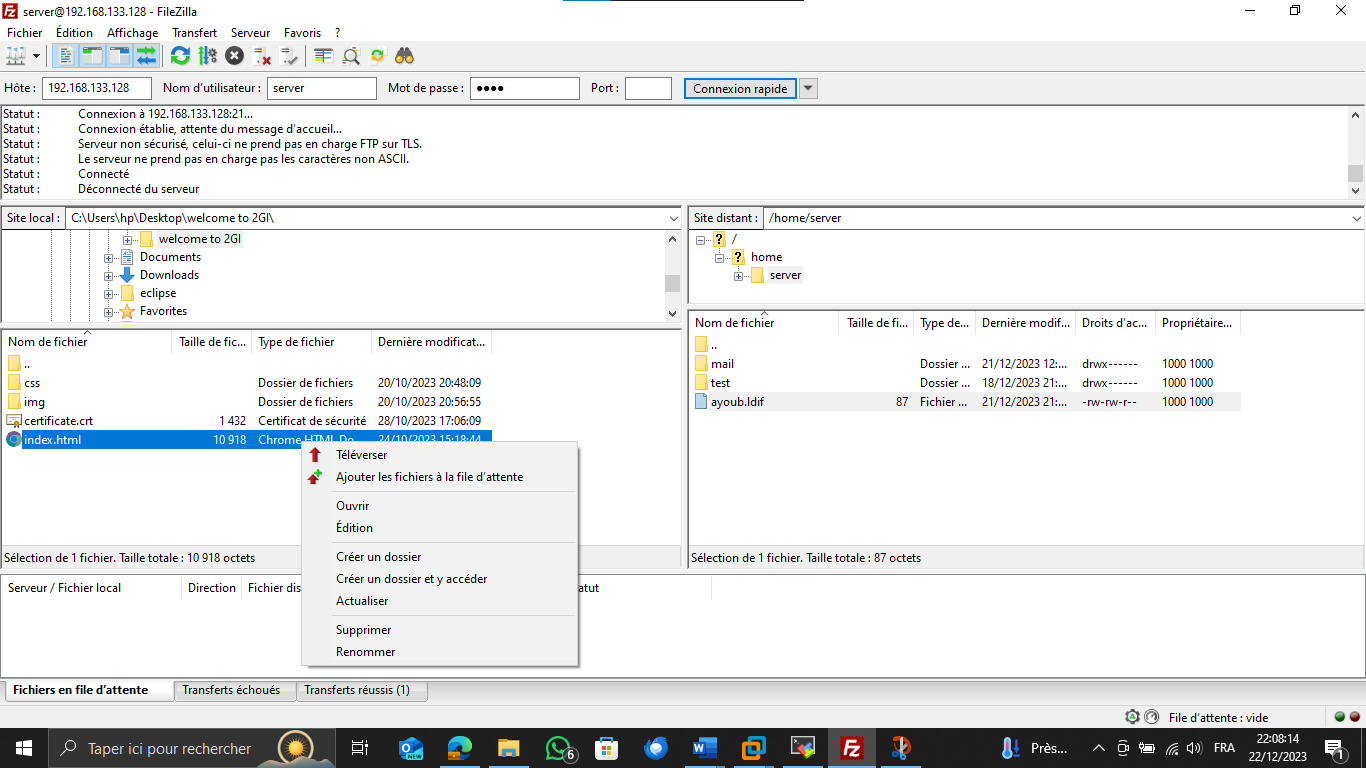
Dès qu'on soit connecté on voit tous les dossiers de notre serveur et on peut télécharger et/ou téléverser des fichiers du et vers notre serveur ftp.



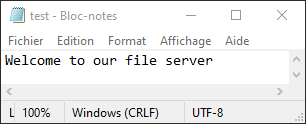
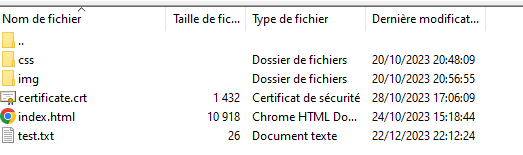
Dans le côté droit de la console de FileZilla nous voyons les fichiers disponibles sur le serveur des fichiers et on peut parcourir les répertoires.

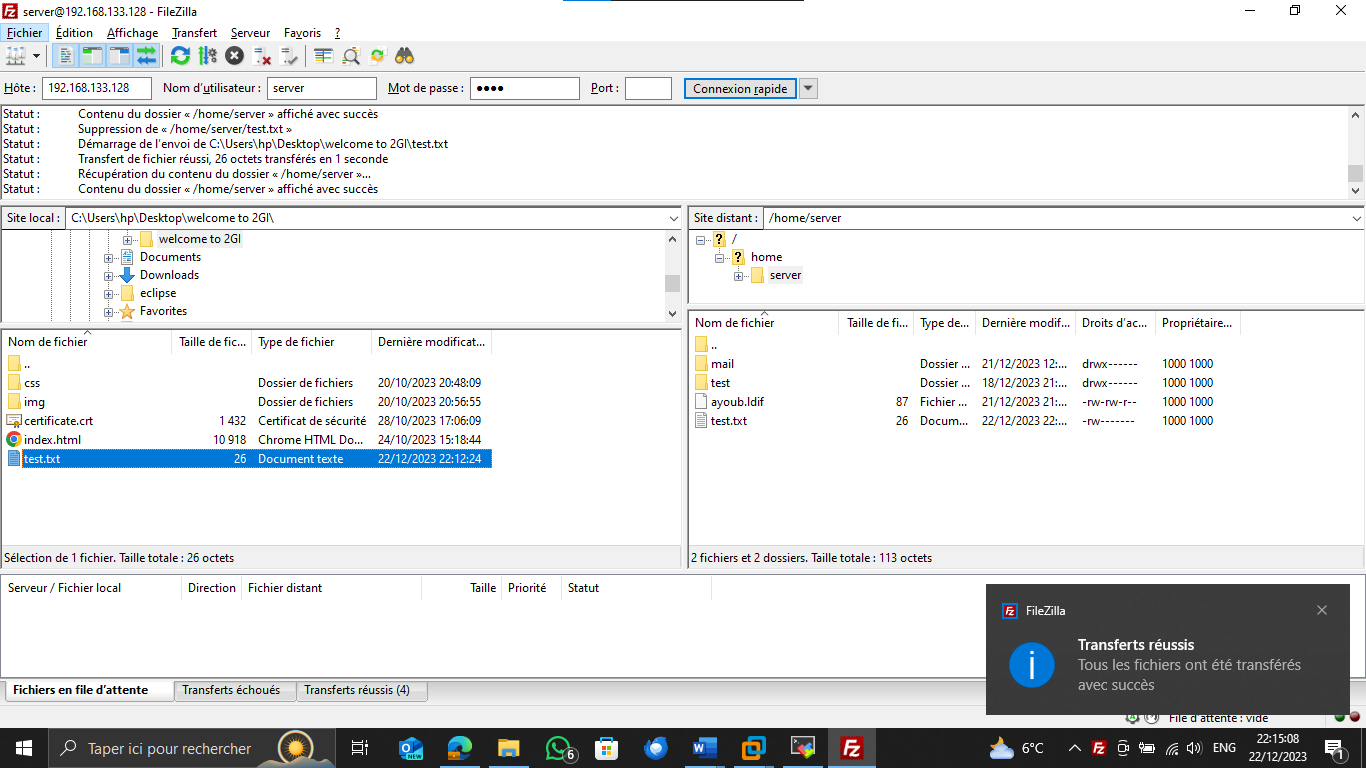


Dans le côté gauche de la console de FileZilla nous voyons les fichiers qu’on peut téléverser sur le serveur des fichiers.



Maintenant on va faire un test pour téléverser un fichier test.txt vers le serveur.



On voit qu'il est bien téléversé au niveau du serveur ftp.



# Configuration du SMTP et IMAP

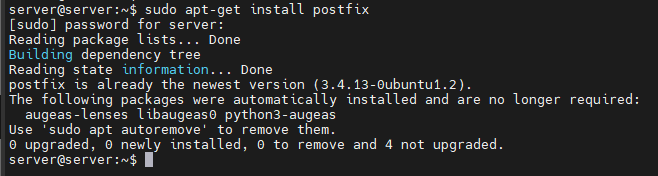
## Définition

Un serveur SMTP (Simple Mail Transfer Protocol) est un serveur qui permet d'envoyer des courriels sur internet. Lorsqu’on utilise un client de courriel (comme Outlook ou Gmail) pour envoyer un courriel, le client de courriel envoie le courriel à un serveur SMTP qui se charge de l'acheminer vers sa destination finale. Les serveurs SMTP utilisent le protocole SMTP pour communiquer entre eux et acheminer les courriels à leur destination finale. Les serveurs SMTP sont également utilisés pour recevoir des courriels et les stocker dans un compte de courriel.

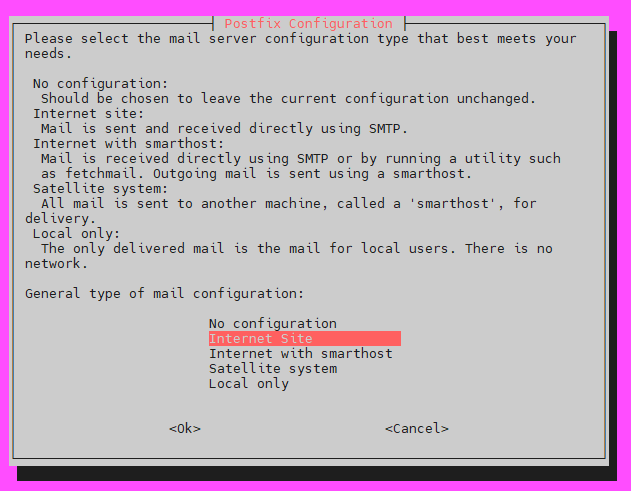
## Les étapes de la configuration

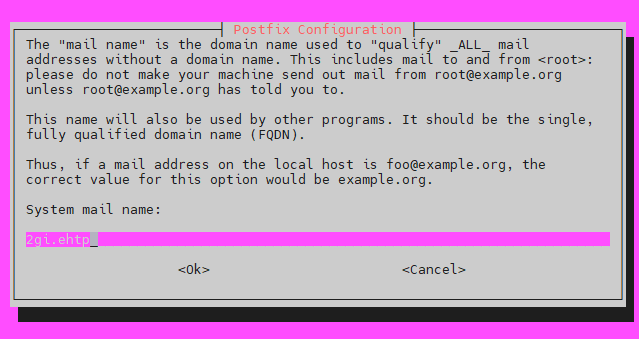
Pour configurer le serveur SMTP dans notre serveur d’entreprise, on a suivi les étapes suivantes :

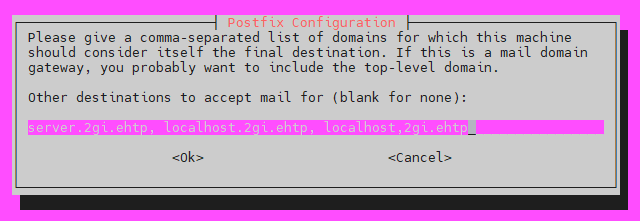
* On a installé le serveur de messagerie Postfix sur notre serveur d’entreprise. Postfix est l'un des serveurs de messagerie les plus populaires.

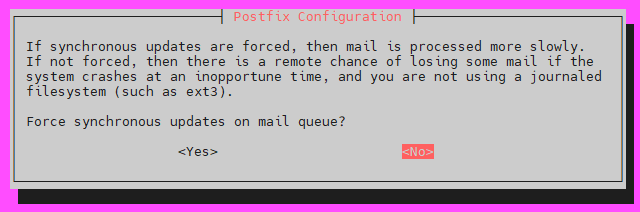


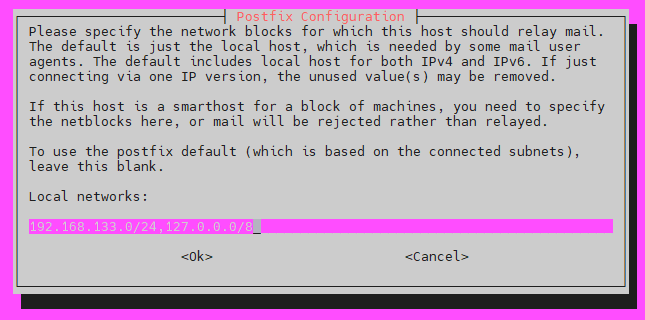
Lors de l'installation de Postfix, on sélectionne le type de configuration de notre serveur de messagerie. On sélectionne "Internet Site" pour configurer Postfix en tant que serveur SMTP.

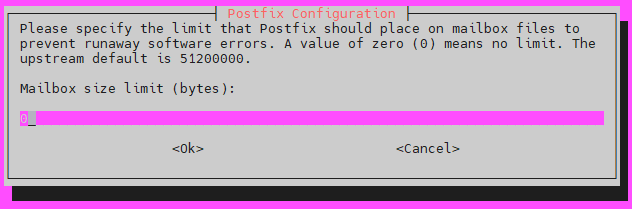


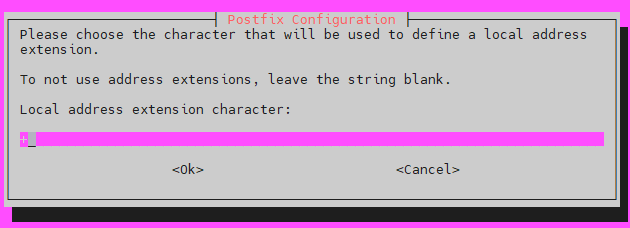


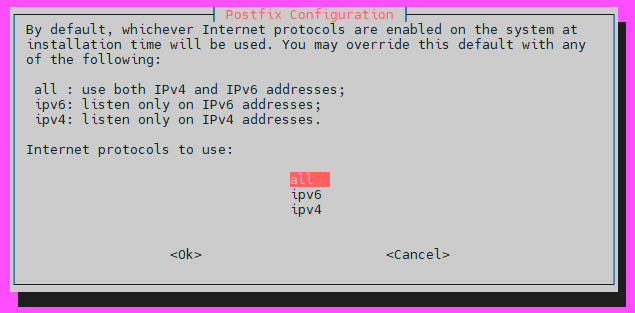




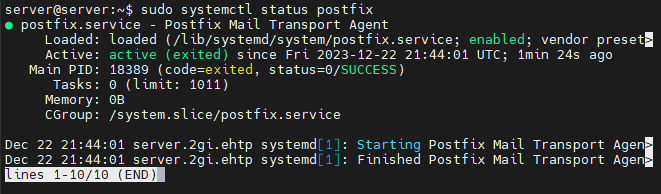




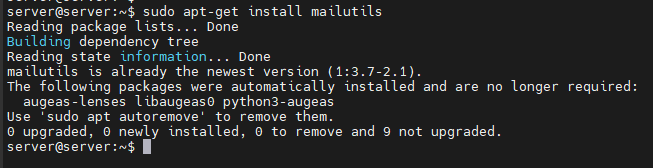




* On vérifier le statut de postfix et de voir si elle est active ou pas



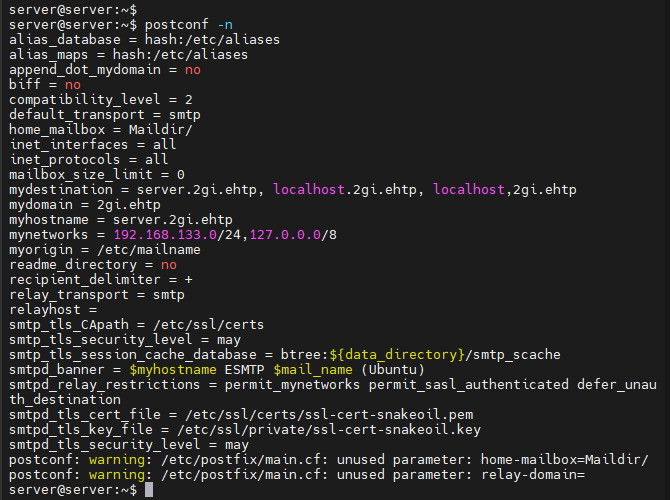
* On installe les outils de mail



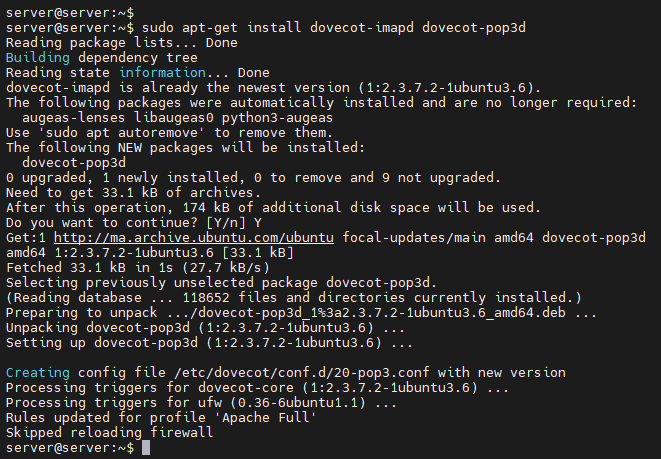
* On configure le « home\_mailbox » en lui affectant le répertoire « Maildir »



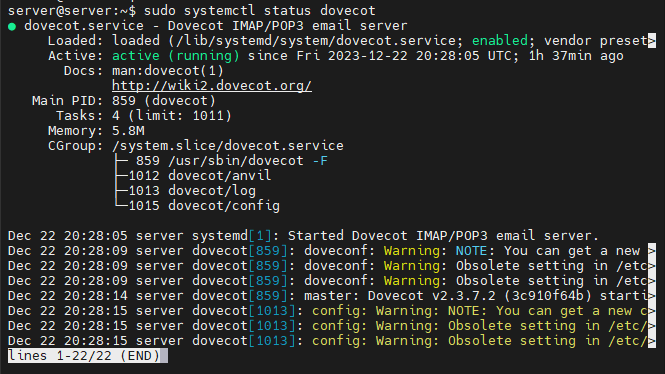
* On teste si la dernière commande a été exécuté avec succès



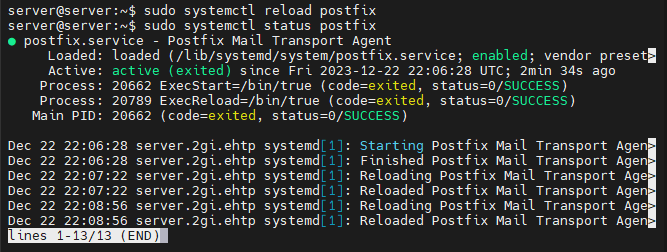
* On installe les packages **dovecot-imapd** et **dovecot-pop3d :** on a installé imap et pop3 dans notre serveur.



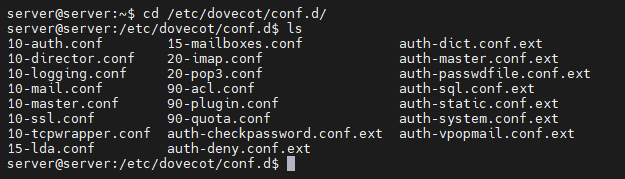
* On vérifier que **dovecot(imap)** est bien active



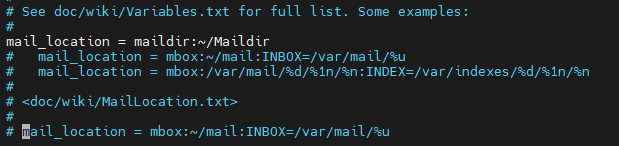
* On a redémarré le « postfix » et on afficher son statut par la suite



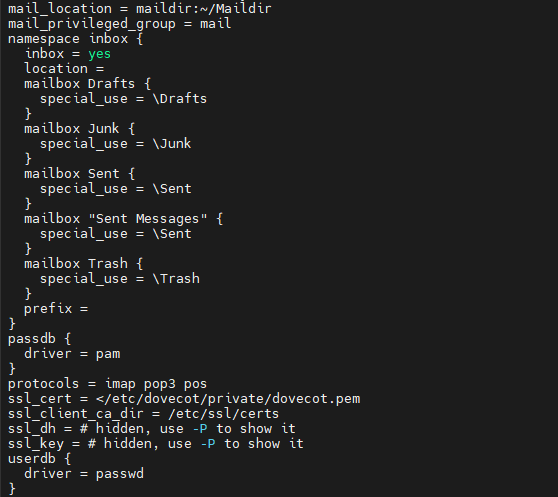
On s’est déplacé vers ce répertoire **/etc/dovecot/conf.d/**



* On modifie le fichier **10-mail.conf**



* On teste si tous les modifications sont ajoutées avec succès



On se déplace vers le répertoire /**etc/ skel/ :**

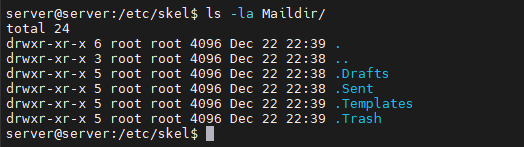
* On a créé des répertoires et des sous-répertoires

sudo mkdir -p Maildir/.Drafts Maildir/.Drafts/cur Maildir/.Drafts/new Maildir/.Drafts/tmp

sudo mkdir -p Maildir/.Sent Maildir/.Sent/cur Maildir/.Sent/new Maildir/.Sent/tmp

sudo mkdir -p Maildir/.Trash Maildir/.Trash/cur Maildir/.Trash/new Maildir/.Trash/tmp

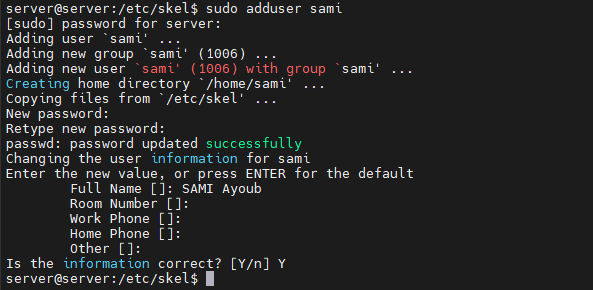
sudo mkdir -p Maildir/.Drafts Maildir/.Templates/cur Maildir/.Templates/new Maildir/.Templates/tmp



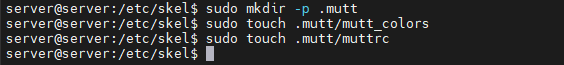
* On a changé les droits d’accès



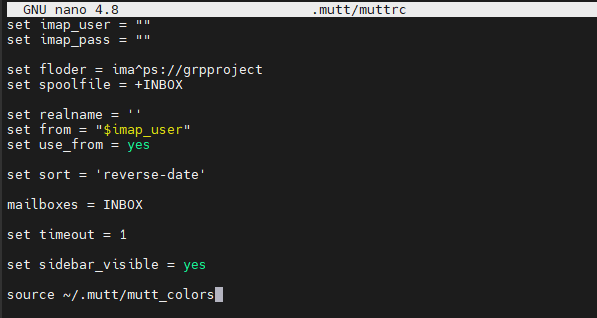
* On créé 2 utilisateurs : ayoub et sami

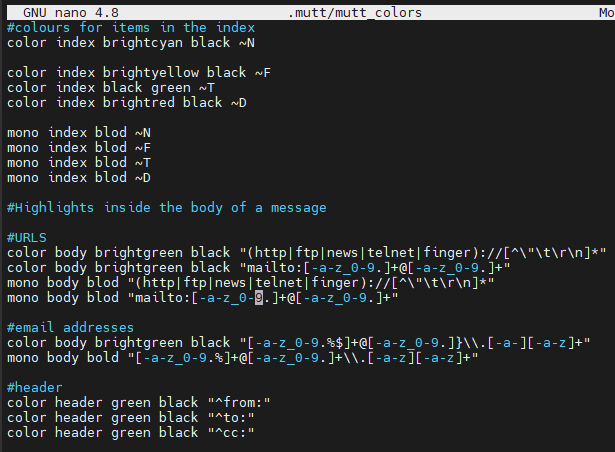


* On a créé le répertoire ‘.**mutt’** et avec ses fichiers

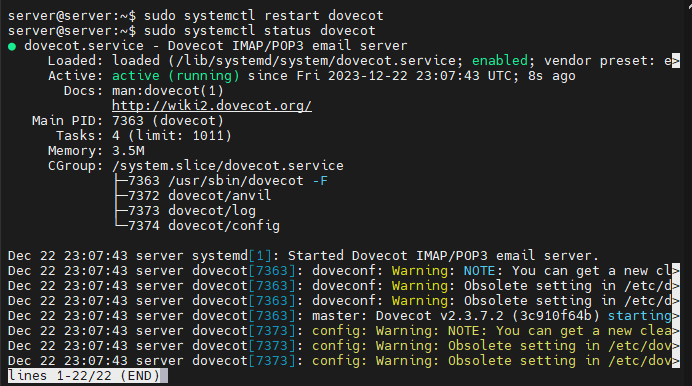


* On a saisi les données de chaque fichier du répertoire « .mutt »





* On redémarre **dovecot** puis affiche son statut

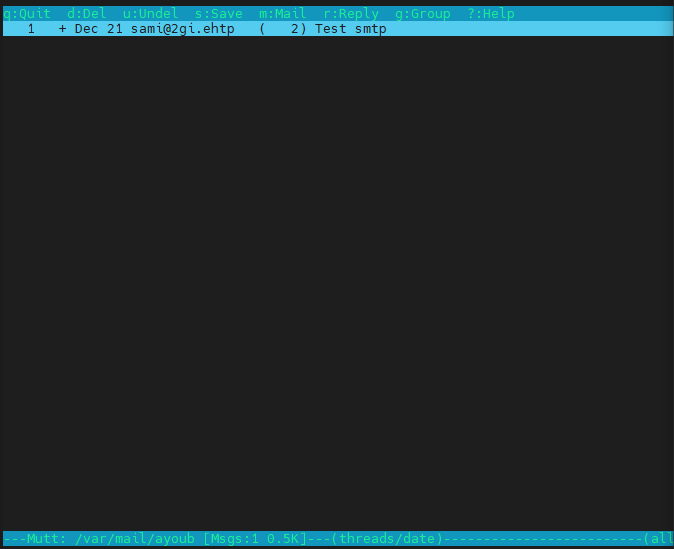


1. **La mise en oeuvre de la réalisation**

Maintenant qu’on a fini la configuration de serveur on va utiliser s’authentifier à l’utilisateur **ayoub** :



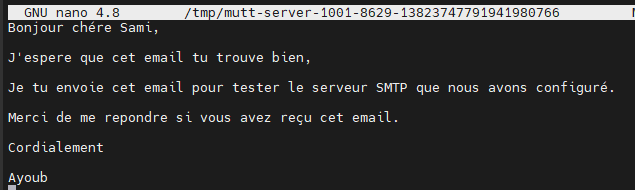
Maintenant on va envoyer un email à l’utilisateur **sami** grâce à la commande **mutt :**



Pour rédiger un email en tape m :





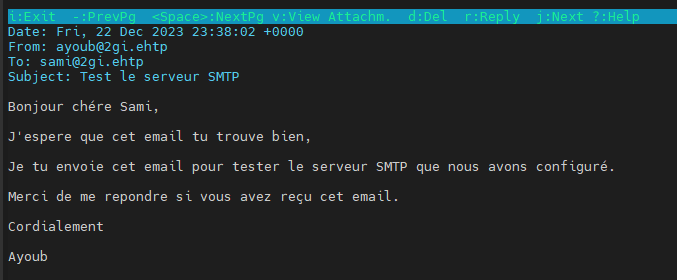


On ferme le corps de l’email par ctrl+C, puis on click y pour envoyer l’email.



Maintenant, on se connecte au compte **sami**. Et on tape grace a la command mutt on voit les emails reçus :





# Configuration du LDAP

## Définition

LDAP (Lightweight Directory Access Protocol) est un protocole de réseau utilisé pour accéder et gérer des informations stockées dans un annuaire. Un annuaire est une base de données qui stocke des informations structurées sur des utilisateurs, des groupes et des ressources.

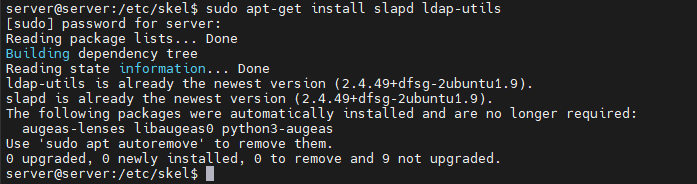
LDAP permet aux applications de se connecter à un annuaire et de lire, d'ajouter, de supprimer et de modifier des informations stockées dans l'annuaire. Il est souvent utilisé pour authentifier les utilisateurs et autoriser l'accès aux ressources, ainsi que pour stocker des informations de profil d'utilisateur, comme le nom, l'adresse e-mail et le numéro de téléphone.

LDAP utilise un modèle de données hiérarchique qui est similaire à l'arborescence des dossiers de fichiers sur un ordinateur. Chaque entrée de l'annuaire est un objet qui peut avoir des attributs (comme le nom, l'adresse email, etc.) et des relations avec d'autres objets.

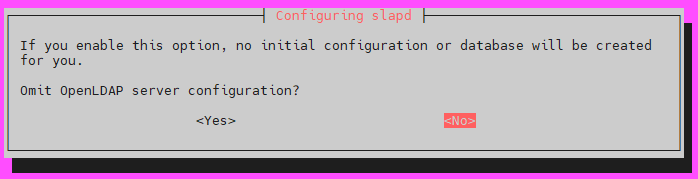
LDAP utilise un système de nommage basé sur des DN (Distinguished Names) pour identifier les objets dans l'annuaire. LDAP est souvent utilisé pour l'authentification unique (Single Sign-On, SSO), ce qui permet à un utilisateur de s'authentifier une seule fois pour accéder à plusieurs ressources et systèmes. LDAP est également souvent utilisé en tant que source d'informations pour les annuaires intégrés dans les systèmes d'exploitation et les applications.

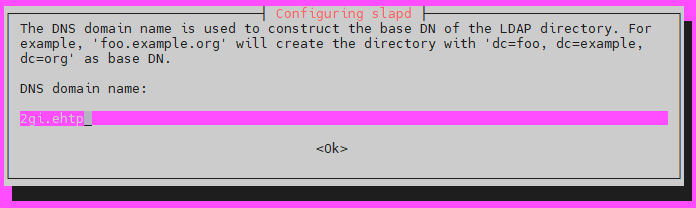
## Les étapes de la configuration

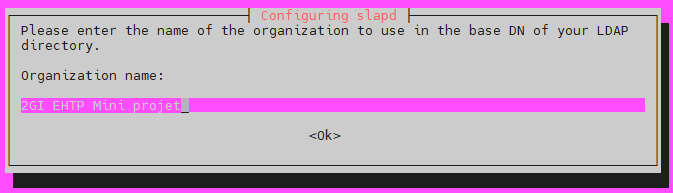
Pour configurer LDAP sur notre serveur d’entreprise, on installe OpenLDAP et un client LDAP. En utilisant la commande suivante :

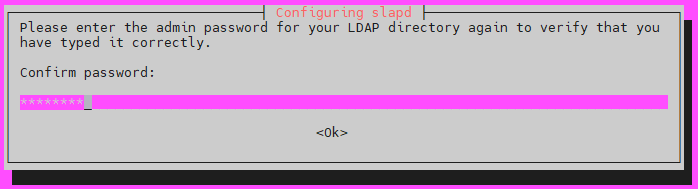


Une fois que OpenLDAP est installé, on configure notre base de données LDAP. En exécutant la commande : **sudo dpkg-reconfigure slapd.** On suit les étapes suivantes :





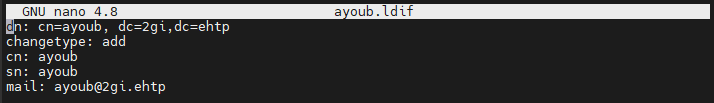




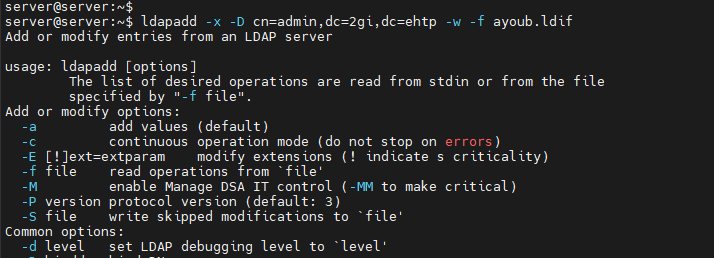
Une fois qu’on a configuré LDAP et ajouté des entrées, on peut utiliser les utilitaires LDAP tels que **ldapsearch** et **ldapmodify** pour gérer la base de données LDAP. On utilisera également des outils tels que **phpldapadmin** pour gérer votre base de données LDAP via une interface Web.

* On crée un fichier **ayoub.ldif**

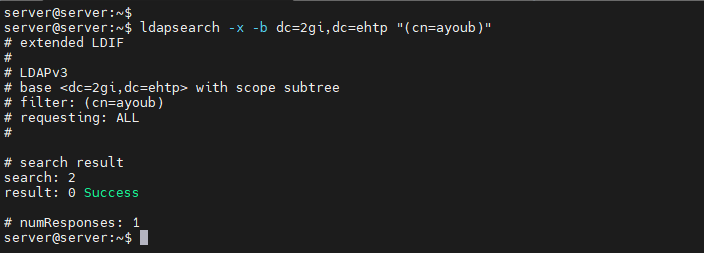




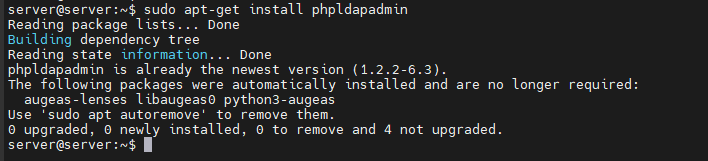
* On ajout l’entrée **ayoub.ldif**



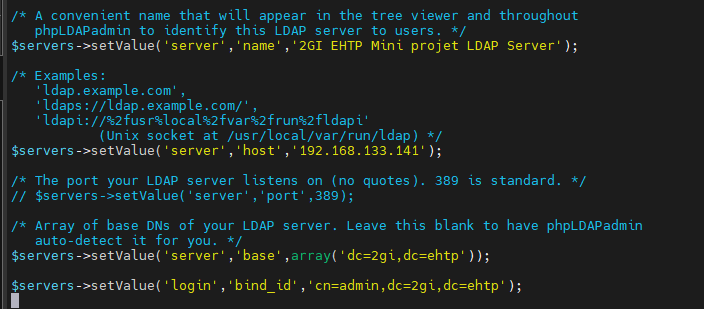
* On cherche l’entrée qu’on vient d’ajouter



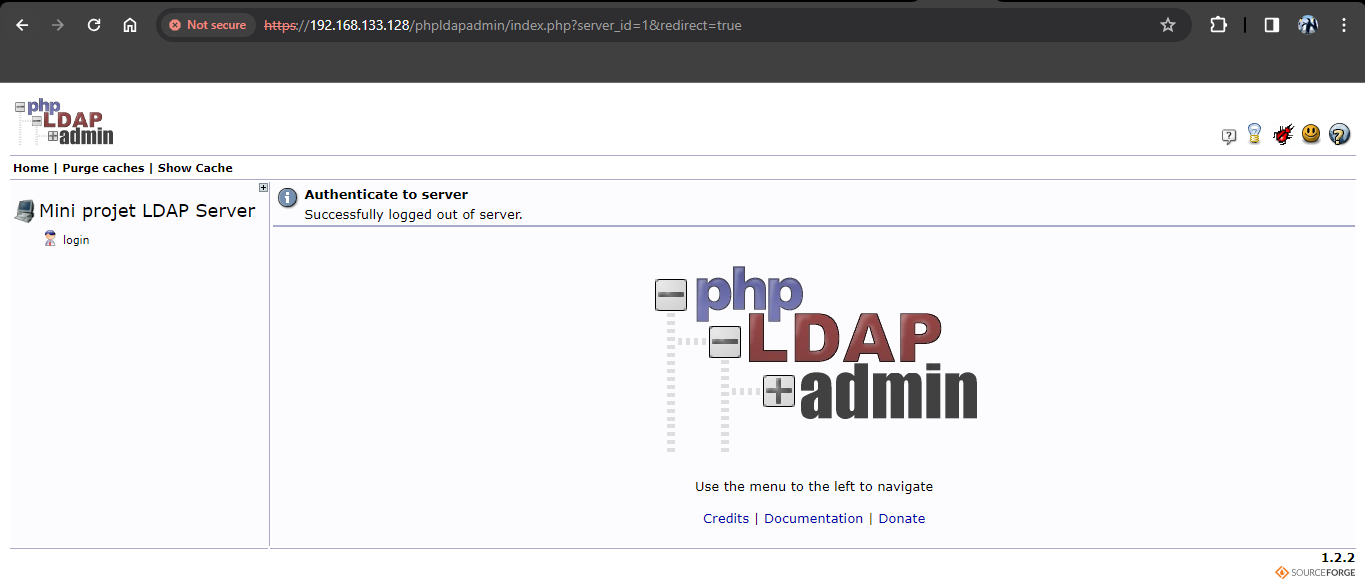
* **Installation et configuration d’une interface web LDAP :  PhpLDAPadmin**
* On installe d’abord phpLDAPadmin

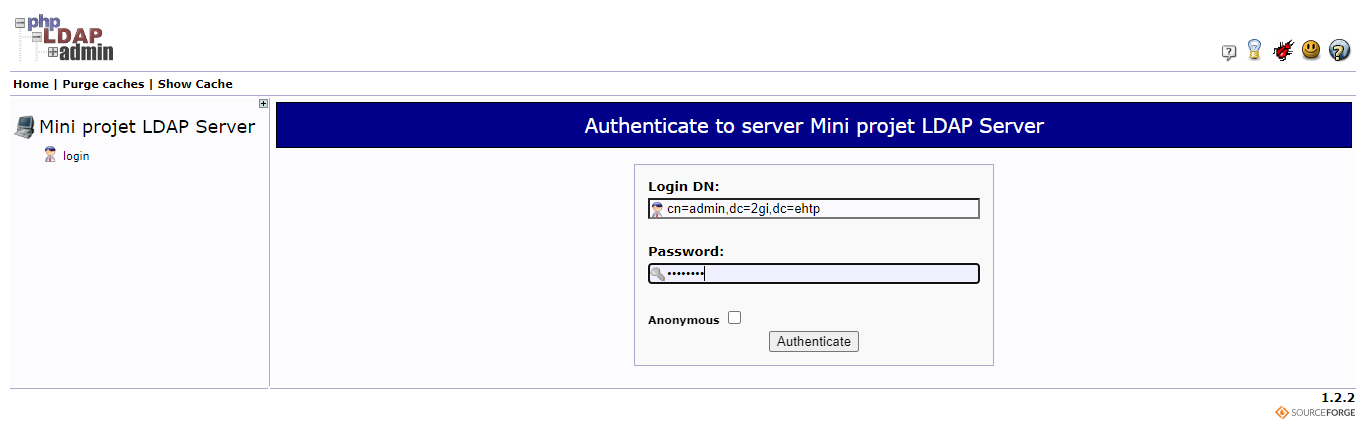


* On ouvre le fichier config.php de PhpLDAPadmin et on modifie les paramètres de configuration suivants :



* Maintenant on se connecte à l’interface web du phpLDAPadmin pour la gestion de LDAP :



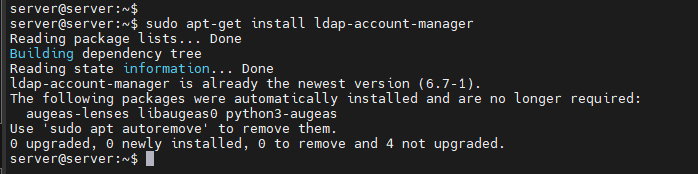




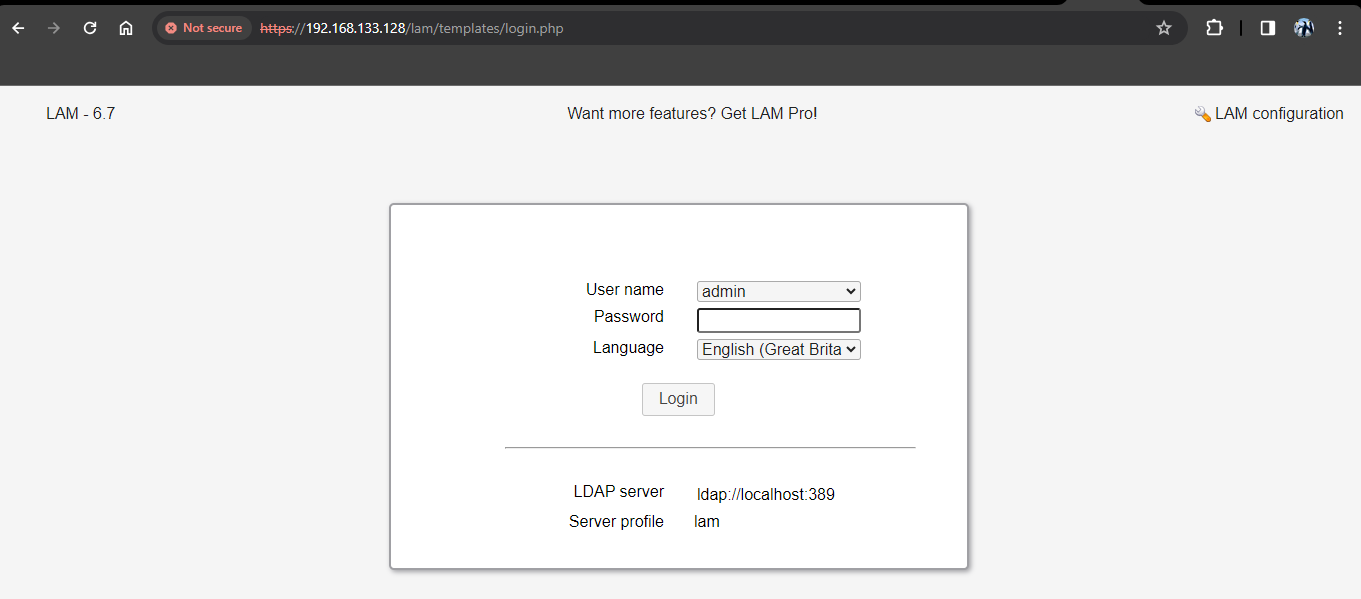
* **Installation et configuration d’une interface web LDAP : LAM (LDAP Account Manager)**

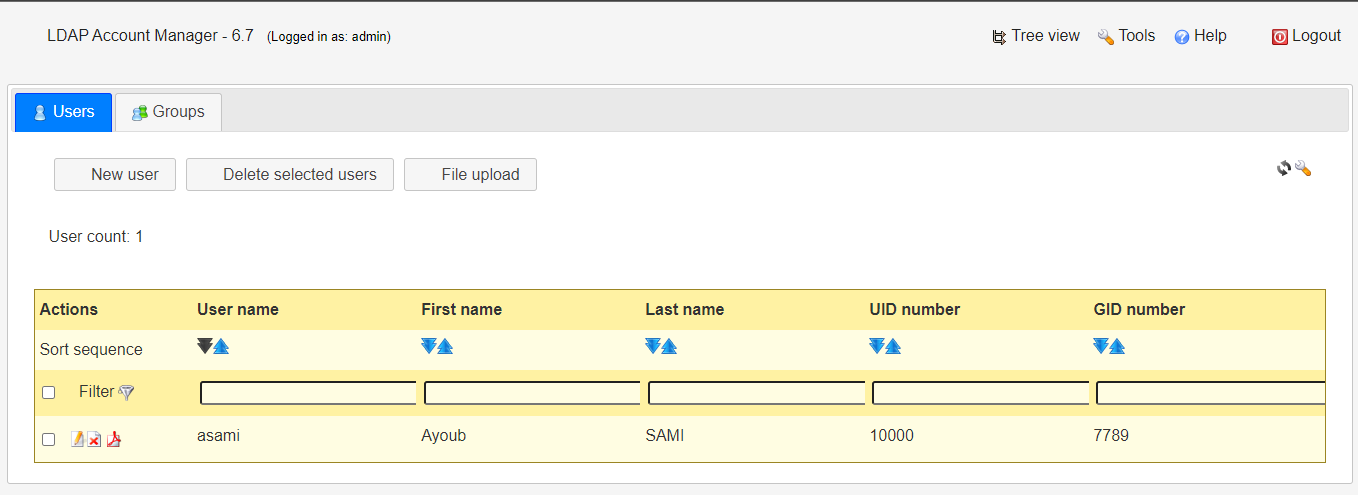
**LDAP Account Manager** est une [application Web](https://en.wikipedia.org/wiki/Web_application) permettant de gérer différents types de comptes dans un Annuaire [LDAP](https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol). Il est écrit en [PHP](https://en.wikipedia.org/wiki/PHP). Contrairement aux outils tels que [PhpLDAPadmin](https://en.wikipedia.org/wiki/PhpLDAPadmin), l'accent est mis sur le compte et donne à l'utilisateur une vue plus abstraite d'un répertoire. Cela vise à permettre à des personnes ayant peu de connaissances techniques de gérer les données LDAP. L'application de base est sous licence [Licence publique générale GNU](https://en.wikipedia.org/wiki/GNU_General_Public_License), et une version étendue est disponible sous licence commerciale.

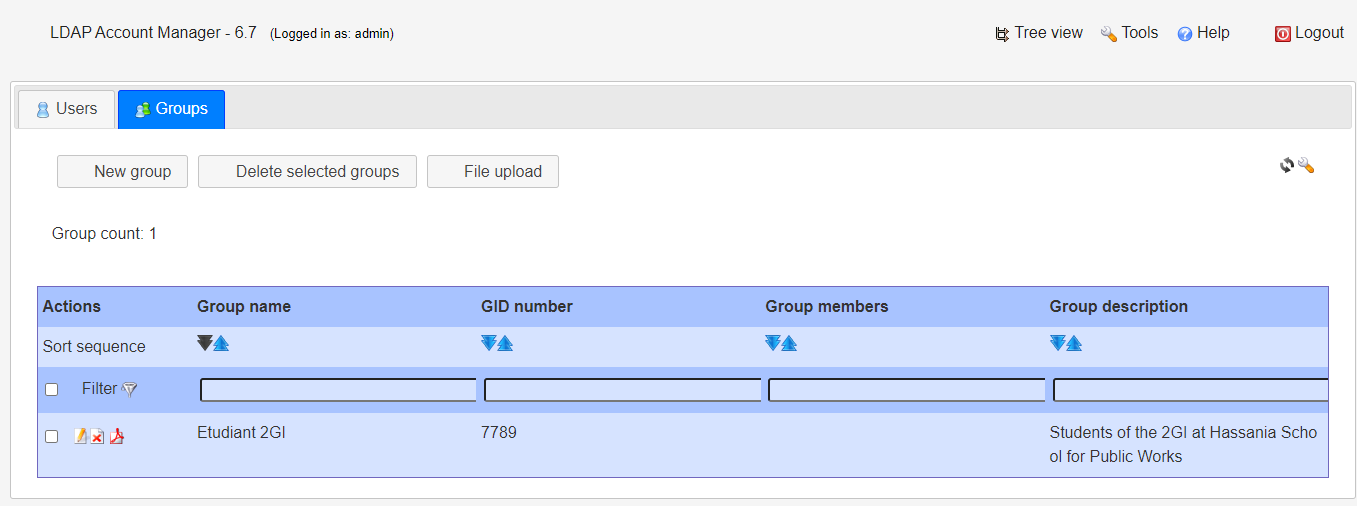
* On installe d’abord lam

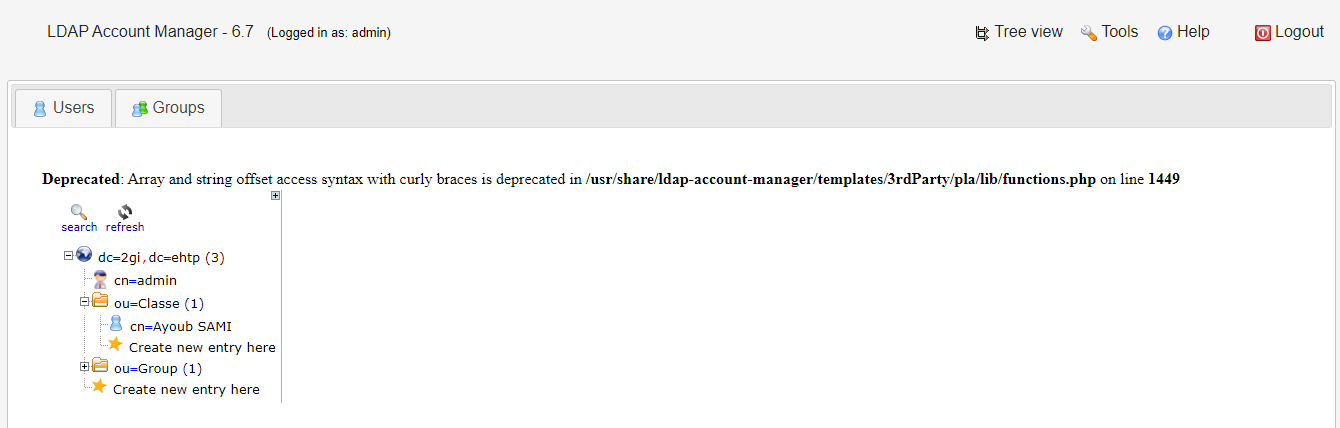


* Maintenant on se connecte à l’interface web lam de gestion de LDAP :









# Conclusion

Au cours de ce projet, on a pu configurer les six serveurs et bien assimiler les concepts de ces protocoles et leurs configurations ainsi que leurs fonctions dans le réseau.