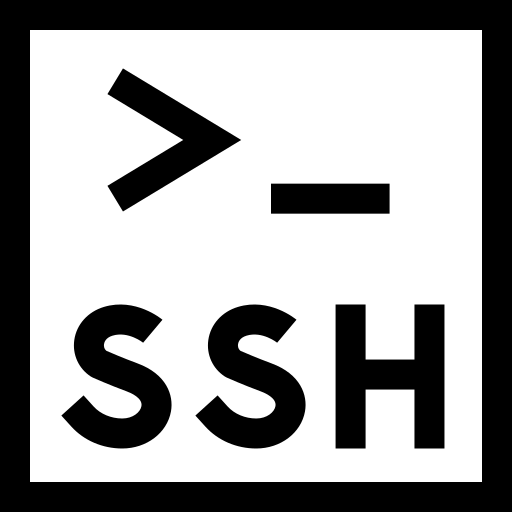


**Rapport**

Configuration de Connexion SSH entre Serveur et client

****

Réalisé par :

HOUDA KAISSI

ESSALHI SARA

I. Objectifs

L’objectif de ce projet est l’utilisation du protocole SSH pour établir une connexion Client-Serveur avec authentification par mot de passe

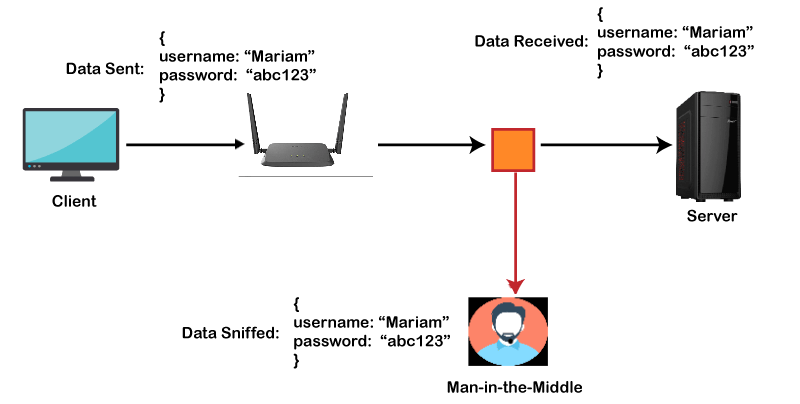
**II.SSH**

SSH, ou Secure Shell, est un protocole de communication sécurisé conçu pour assurer des connexions sécurisées sur un réseau non sécurisé. Il est largement utilisé pour administrer des serveurs à distance, transférer des fichiers de manière sécurisée et exécuter des commandes à distance

Port : Par défaut, le port TCP utilisé par SSH est le port 22

Authentification : SSH prend en charge plusieurs méthodes d'authentification, y compris l'authentification par mot de passe

Sécurité : SSH assure la confidentialité et l'intégrité des données échangées



III. Dispositif du projet





PC avec Ubuntu (client) Server routeur

IV. Réalisation

Nous avons rencontré un problème d'authentification avec VMware, ce qui nous a poussés à rechercher une solution dans le menu du BIOS. En exploitant les capacités de surveillance des ressources du serveur fournies par Ilo 5, nous avons identifié une solution de contournement. Cependant, en raison de contraintes de licence, nous avons choisi de remplacer un disque dur et de déployer Ubuntu comme système d'exploitation.

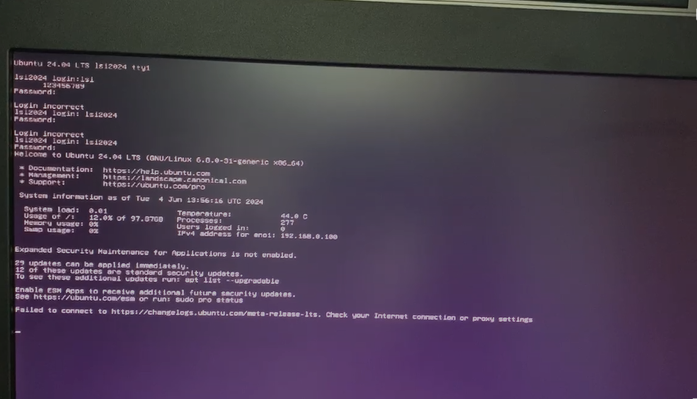


Configuration du Serveur SSH

Après l'installation d'Ubuntu, nous avons procédé à la configuration de SSH pour un accès distant sécurisé aux machines serveur et client.

Détails d'Accès au Serveur :

* Nom d'Utilisateur : LSI2024
* Mot de Passe : 123456789



C**réation d'Utilisateur SSH**

Pour faciliter les connexions SSH vers le serveur, nous avons mis en place un utilisateur dédié. SSH a été installé avec succès sur le serveur en utilisant la configuration par défaut via la commande **sudo apt install ssh.**

Maintenant, pour renforcer la sécurité du serveur, nous allons configurer le pare-feu pour permettre uniquement le trafic provenant du port SSH désigné.

**Configuration router**



Pour accéder au réseau et vous connecter à Internet via un câble Ethernet, branchez-le sur le réseau de la faculté. Une fois connecté, accédez aux paramètres du routeur depuis votre PC à l'URL : 192.168.1.1. Entrez le mot de passe TP-Link lorsque vous y êtes invité. Ensuite, sélectionnez l'option "Répéteur" et configurez les paramètres souhaités du réseau WiFi, tels que les filtres WiFi. Utilisez votre appareil mobile pour vous connecter au réseau WiFi configuré afin de poursuivre l'installation des packages requis.

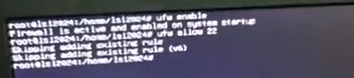
Configuration du Pare-feu :

* Autoriser le trafic entrant uniquement sur le port SSH (généralement le port 22) à l'aide de la commande suivante :

**Copy code**

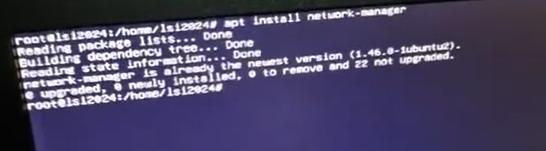
**sudo ufw allow ssh**

Cette commande permet les connexions SSH entrantes à travers le pare-feu.

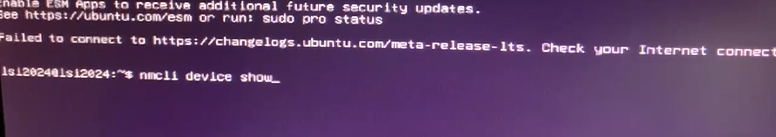


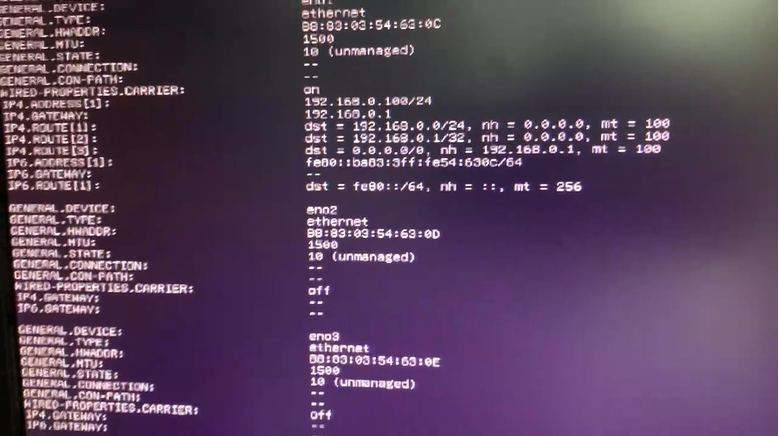
Installer package pour voir l’addresse ip de ssh server

apt install network-manager



Une fois connecté au serveur, ouvrir un terminal et saisir la commande suivante pour obtenir l'adresse IP : **nmcli device show.**





**Installation de SSH client et configuration**

Nous avons commencé par l’installation du service SSH sur le PC via la commande suivante :

**sudo apt install openssh-server**

Une fois installé, le service SSH est généralement automatiquement démarré. Pour vérifier son statut et s'assurer qu'il est en cours d'exécution, nous avons utilisé la commande suivante :

**sudo systemctl status ssh**

Pour permettre l'accès SSH au PC, nous avons également vérifié si le pare-feu autorisait le trafic sur le port 22,

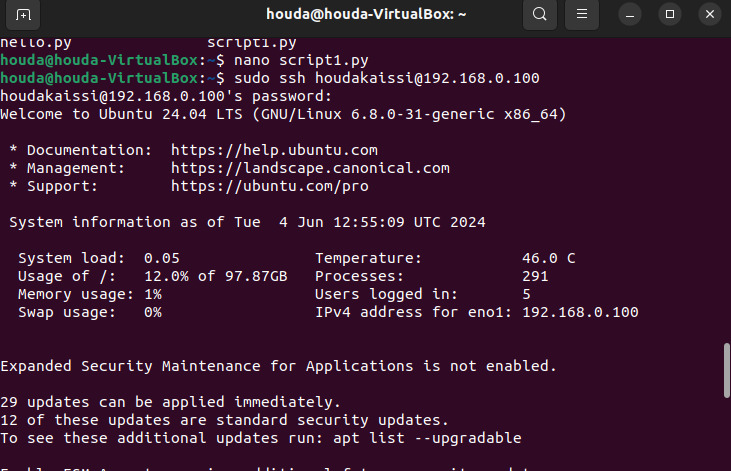
Après l'installation, nous avons ouvert le fichier **"sshd\_config"** situé dans le répertoire "/etc/ssh". À l'intérieur de ce fichier, nous avons rencontré plusieurs instructions, telles que :

* Le numéro de port utilisé : Port 22
* Pour autoriser la connexion en tant que root : PermitRootLogin yes
* Le chemin vers le fichier contenant les clés publiques autorisées : AuthorizedKeysFile .ssh/authorized\_keys
* La permission d'authentification par mot de passe : PasswordAuthentication yes

**connexion avec serveur**

Démarrer le service SSH avec la commande : **sudo systemctl start ssh**.

Sur votre PC, ouvrir un terminal et saisir la commande SSH pour vous connecter au serveur : **ssh** [**LSI2024@add\_server**](mailto:LSI2024@add_server)**.**



Une fois que la connexion est établie, saisir le mot de passe requis.

Maintenant, vous êtes connecté au serveur Ubuntu LSI024 depuis votre PC

Après avoir établi une connexion SSH en utilisant la commande ssh nom\_utilisateur@adresse\_IP\_serveur et fourni votre mot de passe, vous pouvez créer un fichier Python nommé fact.py et l'exécuter en suivant ces étapes :

Connexion SSH au serveur : Utilisez la commande suivante pour vous connecter au serveur distant en remplaçant nom\_utilisateur par le nom d'utilisateur approprié et adresse\_IP\_serveur par l'adresse IP du serveur :

**ssh nom\_utilisateur@adresse\_IP\_serveur**

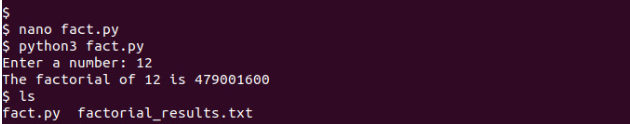
Authentification : Entrez votre mot de passe lorsque vous y êtes invité pour vous authentifier sur le serveur distant.

**création de script python**

Création du fichier Python : Une fois connecté au serveur, utilisez un éditeur de texte ou une commande de création de fichier pour créer un fichier Python nommé fact.py. Par exemple, vous pouvez utiliser l'éditeur de texte nano :

Cela ouvrira l'éditeur de texte nano, où vous pourrez saisir votre code Python. Écrivez votre script Python dans le fichier fact.py et enregistrez-le.

Exécution du fichier Python : Après avoir créé le fichier hello.py, vous pouvez l'exécuter en utilisant l'interpréteur Python 3. Utilisez la commande suivante pour exécuter le script Python :



->Assurer que vous êtes dans le même répertoire que le fichier fact.py lors de son exécution, ou spécifiez le chemin complet du fichier si nécessaire.