Configurer SSH sur Linux, et accéder à distance via terminal powershell windows

1

[Updated] Configuration d'une clef SSH pour les serveurs Ubuntu/Debian (GNU/Linux)

Sur la VM Linux, se placer dans le répertoire SSH de l'utilisateur

cd /home/titi/.ssh

Si le répertoire n'existe pas

mkdir /home/titi/.ssh

cd /home/titi/.ssh

Générer une clef SSH, garder le path et le nom par défaut id\_rsa

ssh-keygen

Ensuite, ajoutez la clé publique au fichier authorized\_keys en saisissant ce qui suit :

cat /home/titi/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys

Appliquer une permissions sur le dossier 700

chmod -R 700 /home/titi/.ssh/

Sur votre poste Windows, importer la clef privée avec SCP dans votre dossier utilisateur Windows:

scp titi@10.0.X.X:/home/titi/.ssh/id\_rsa "C:\Users\votre\_user\_windows\.ssh\titikey"

(Adapter IP et Path aux besoins)

Configuration SSH-Server

Une fois la clef privé importé sur votre machine Windows, il va falloir aller modifier le fonctionnement du service SSH sur la VM Linux:

nano /etc/ssh/sshd\_config -c

Désactiver l'authentification par mot de passe et n'accepter que l'authentification par certificat

Commenter ou Dé-Commenter les lignes suivantes et changer les valeurs:

# Include /etc/ssh/sshd\_config.d/\*.conf

PermitRootLogin without-password

PasswordAuthentication no

PubkeyAuthentication yes

ChallengeResponseAuthentication no

Redémarrer le service SSH

systemctl restart ssh

Pour se connecter depuis Windows Terminal:

ssh -i "C:\Users\votre\_user\_windows\.ssh\titikey" titi@10.0.0.XXX

2

Modifier le fichier de configuration "YAML" dans /etc/netplan/ pour assigner une @IP static à la VM :

netplan generate

nano /etc/netplan/00-installer-config.yaml

# This is the network config written by 'subiquity' network: ethernets: ensXXX: addresses: [192.168.1.150/24] routes: - to: 0.0.0.0/0 via: 192.168.1.1 metric: 3 nameservers: addresses: [8.8.8.8, 8.8.4.4] dhcp4: false dhcp6: false version: 2

netplan apply Pour débug: netplan --debug apply

Etape 1.1 > Vérifier les modifications réseau et réaliser des tests de connectivités

ip a pour vérifier toujours, et ip route

ping votre\_paserelle

ping google.com

ETAPE 2.1 > Changer le nom d'hôte d'une VM Ubuntu

hostnamectl hostname lenomdevotrechoix

Lancer la commande hostnamectl pour observer la modification

Vérifier si le nom de l'hôte correspond dans le fichier de configuration suivant:

hostnamectl

nano /etc/hostname

nano /etc/hosts

ETAPE 3 > Editer le fichier de source des paquets > sources.list suivant votre version:

Vérifier la présence des sources des paquets >

nano /etc/apt/sources.list

Mettre à jour les sources des paquers APT

apt-get update

Tester si vous pouvez rechercher netstat et l'installer

apt search netstat

apt-get install net-tools

netstat -laputen

ETAPE 5 > Configurer l'heure de la VM en fonction de la TimeZone Europe/Paris

sudo timedatectl set-timezone Europe/Paris

Afficher la nouvelle configuration

Timedatectl

LAB - Allow SSH Users

Trouver une solution pour n'autoriser que certains utilisateurs à se connecter en SSH.

Disons titi et tata cette fois, mais pas toto.

Pour tester, depuis votre PC avec PowerShell

ssh [titi@192.168.X.X](mailto:titi@192.168.X.X)

Correction - interdire des utilisateurs sur SSH

Trouver une solution pour n'autoriser que certains utilisateurs à se connecter en SSH. Disons titi et tata cette fois, mais pas toto.

Créer un groupe qui permettra aux utilisateurs membre d'accéder au serveur en SSH.

addgroup authorized \_ssh\_group

Ajouter les utilisateurs que l'on souhaite qui disposeront de permissions à SSH

usermod -a -G authorized\_ssh\_group titi

usermod -a -G authorized\_ssh\_group tata

Editer le fichier de configuration du service SSH:

nano /etc/ssh/sshd\_config

Ajouter la ligne suivante:

AllowGroups authorized\_ssh\_group

Redémarrer le service SSH:

systemctl restart ssh

Tester depuis le PC avec PowerShell

ssh titi@192.168.X.X

Note: On pourrait à l'inverse de l'option Allow utiliser l'option Deny:

DenyUsers les\_users

Ou encore

DenyGroups forbiden\_ssh\_group

LAB - Sudoers

Faites en sorte que l'utilisateur titi puisse avoir des droits sudoers, par exemple qu'il puisse redémarrer des services...

usermod -a -G sudo titi