Guacamole

Installation de Guacamole

Déja il faut effectuer une mise à jour des paquets :

apt update

Ensuite il faut installer les paquets nécessaires à l'installation de guacamole :

```
apt install -y build-essential \
∏libcairo2-dev \
□libjpeg62-turbo-dev \
∏libpng-dev \
□libtool-bin \
□libossp-uuid-dev \
∏libavutil-dev \
□libswscale-dev \
     freerdp2-dev \
□libpango1.0-dev \
□libssh2-1-dev \
□libvncserver-dev \
∏libtelnet-dev \
□libwebsockets-dev \
□libssl-dev \
□libvorbis-dev \
□libwebp-dev \
□libpulse-dev \
     sudo \
     vim
```

Pour le bien des commande suivante nous allons définir une variable d'environnement :

Une fois cela fait, nous pouvons télécharger guacamole-server :
wget https://downloads.apache.org/guacamole/\$VER/source/guacamole-server-\$VER.tar.gz
Il faut ensuite extraire l'archive :
tar xzf guacamole-server-\$VER.tar.gz
Il faut ensuite se déplacer dans le dossier :
cd guacamole-server-\$VER
Il faut ensuite configurer guacamole-server :
./configurewith-systemd-dir=/etc/systemd/system/
Il faut ensuite compiler guacamole-server :
make CFLAGS="-Wno-discarded-qualifiers" sudo make install
Nous pouvons ensuite créer les différents liens symboliques :
Idconfig
Il faut ensuite recharger les fichiers de configuration :
systemctl daemon-reload
Puis il faut activer le service :
systemctl enablenow guacd
Nous pouvons ensuite vérifier que le service est bien actif :
systemctl status guacd

VER=1.5.3

Installation de tomcat

Il faut commencer par ajouter le dépôt de tomcat dans le fichier /etc/apt/sources.list :

echo "deb http://deb.debian.org/debian/ bullseye main" > /etc/apt/sources.list.d/bullseye.list

Il faut ensuite mettre à jour les paquets :

```
apt update
```

Nous pouvez alors installer tomcat:

```
apt install tomcat9 tomcat9-admin tomcat9-common tomcat9-user -y
```

On peut ensuite vérifier que tomcat est bien installé :

systemctl status tomcat9.service

Il faut également ouvrir le port 8080 :

```
ufw allow 8080/tcp
```

Installation de guacamole-client

Il faut commencer par créer un dossier de configuration :

mkdir /etc/guacamole

Il faut ensuite télécharger guacamole-client :

wget \

https://downloads.apache.org/guacamole/\$VER/binary/guacamole-\$VER.war \

-O /etc/guacamole/guacamole.war

Nous allons créer un lien symbolique de guacamole-client dans le dossier webapps de tomcat :

In -s /etc/guacamole/guacamole.war /var/lib/tomcat9/webapps/

Il faut ensuite redémarrer tomcat et guacamole :

systemctl restart tomcat9 guacd

Configuration de apache guacamole

Il faut commencer par créer un fichier de configuration :

mkdir /etc/guacamole/{extensions,lib}

Il faut ensuite définir la variable d'environnement du répertoire home de guacamole :

echo "GUACAMOLE_HOME=/etc/guacamole" >> /etc/default/tomcat9

Nous pouvons alors créer le fichier de configuration guacamole.properties dans le dossier /etc/guacamole :

cat > /etc/guacamole/guacamole.properties << EOL

guacd-hostname: 127.0.0.1

guacd-port: 4822

user-mapping: /etc/guacamole/user-mapping.xml

auth-provider: net.sourceforge.guacamole.net.basic.BasicFileAuthenticationProvider

EOL

Ensuite nous pouvons créer un lien symbolique du fichier de configuration dans le dossier /usr/share/tomcat9/.guacamole :

In -s /etc/guacamole /usr/share/tomcat9/.guacamole

Nous pouvons ensuite redémarrer tomcat et guacamole :

systemctl restart tomcat9 guacd

Installation et configuration de mariadb

Nous commençons par installer le paquet MariaDB Server :

sudo apt-get install mariadb-server

Puis il nous faut sécuriser l'installation :

sudo mysql_secure_installation

Une fois ceci fait, vous devrez répondre à une série de questions.

Une fois cette étape effectuée, nous pouvons nous connecter à notre instance MariaDB:

mysql -u root -p

Nous allons ensuite créer la base de données guacamole_db, l'utilisateur guacamole_user ainsi que son mot de passe et nous lui donnons les droits sur la base de données guacamole_db :

CREATE DATABASE guacadb;

CREATE USER 'guaca_nachos'@'localhost' IDENTIFIED BY 'P@ssword!';

GRANT SELECT, INSERT, UPDATE, DELETE ON guacadb.* TO 'guaca_nachos'@'localhost';

FLUSH PRIVILEGES:

EXIT;

Les étapes suivantes consistent à ajouter l'extension mysql à apache guacamole ainsi que le connecteur correspondant :

cd /tmp

wget https://downloads.apache.org/guacamole/1.5.2/binary/guacamole-auth-jdbc-1.5.2.tar.gz

Il faut ensuite extraire l'archive :

tar -xzf guacamole-auth-jdbc-1.5.2.tar.gz

On déplace le fichier .jar de l'extension dans le répertoire [/etc/guacamole/extensions/] créé précédemment :

sudo mv guacamole-auth-jdbc-1.5.2/mysql/guacamole-auth-jdbc-mysql-1.5.2.jar /etc/guacamole/extensions/

Il faut ensuite télécharger le connecteur mysql :

cd /tmp

wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-j-8.0.33.tar.gz

Il faut ensuite extraire l'archive :

tar -xzf mysql-connector-j-8.0.33.tar.gz

On copie (ou déplace) le fichier .jar du connecteur vers le répertoire lib d'Apache Guacamole :

sudo cp mysql-connector-j-8.0.33/mysql-connector-j-8.0.33.jar /etc/guacamole/lib/

Les dépendances sont déployées, mais nous n'avons pas encore fini cette intégration avec MariaDB.

En effet, il faut importer la structure de la base de données Apache Guacamole dans notre base de données guacadb. Pour cela, on va importer tous les fichiers SQL situés dans le répertoire guacamole-auth-jdbc-1.5.2/mysql/schema/. Le mot de passe root de MariaDB doit être saisit pour effectuer l'import.

cd guacamole-auth-jdbc-1.5.2/mysql/schema/ cat *.sql | mysql -u root -p guacadb

Une fois que c'est fait, on va créer et éditer le fichier guacamole.properties pour déclarer la connexion à MariaDB. Ce fichier peut être utilisé pour d'autres paramètres, selon vos besoins.

sudo nano /etc/guacamole/guacamole.properties

On ajoute les lignes suivantes :

MySQL

mysql-hostname: 127.0.0.1

mysql-port: 3306

mysql-database: guacadb

mysql-username: guaca_nachos mysql-password: P@ssword!

Enregistrez et fermez le fichier.

Il faut également éditer le fichier guacd.conf pour déclarer le serveur Guacamole (ici, on déclare une connexion locale sur le port par défaut, à savoir 4822).

sudo nano /etc/guacamole/guacd.conf

Voici le contenu à ajouter :

[server]

 $bind_host = 0.0.0.0$

 $bind_port = 4822$

On enregistre et on termine par redémarrer les trois services liés à Apache Guacamole :

sudo systemctl restart tomcat9 guacd mariadb

En vous rendant sur l'interface web de Guacamole, vous devriez pouvoir vous connecter avec les identifiants par défaut (*guacadmin/guacadmin*).



Vous pouvez alors créer un nouvel utilisateur **nimda/45keuros** et lui donner les droits d'administration. Ensuite vous pouvez vous connecter avec cet utilisateur et supprimer l'utilisateur guacadmin.



Les deux machines seront ajoutées dans la partie configuration.

Configuration administrateur de guacamole

Pour commencer nous allons créer notre nouvel utilisateur, pour cela il nous faut nous rendre dans l'onglet paramètres :

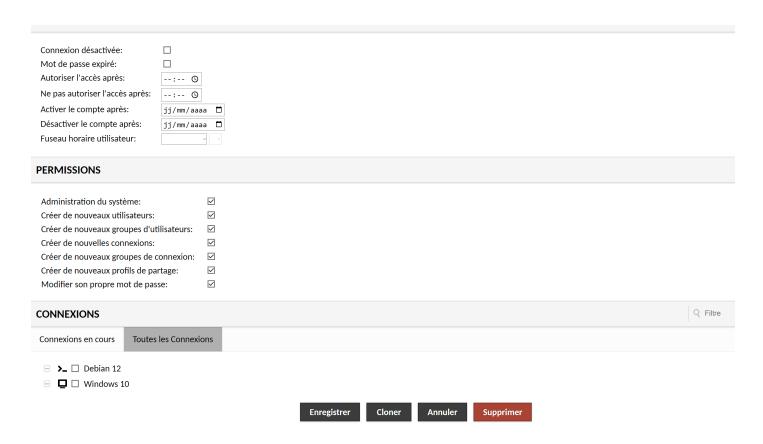


Nous allons ensuite dans l'onglet utilisateurs :



Puis nous cliquons sur ajouter un utilisateur et nous saissons les informations importantes et nous n'oublions pas de cocher toutes les cases afin de donner tous les droits à l'utilisateur :

MODIFIER UTILISATEUR		
Identifiant:	nimda	
Mot de passe:	•••••	
Répéter mot de passe:		
PERMISSIONS		
Administration du système: Créer de nouveaux utilisateurs: Créer de nouveaux groupes d'utilisateurs: Créer de nouvelles connexions: Créer de nouveaux groupes de connexion: Créer de nouveaux profils de partage: Modifier son propre mot de passe:		
CONNEXIONS		
Connexions en cours Toutes les Connexions		
→ □ Debian 12□ □ Windows 10		
	Enregistrer Annuler	
Il nous suffit ensuite de cliquer	sur Enregistrer et notre utilisateur est créé.	
Nous pouvons alors nous déco utilisateur.	nnecter du compte guacadmin et nous connecter avec notre nouvel	
Ainsi nous pouvons supprimer -> guacadmin :	l'utilisateur guacadmin toujours dans l'onglet Paramètres -> Utilisateurs	



Ensuite nous pouvons ajouter nos deux machines dans l'onglet Paramètres -> Connexions -> Nouvelle Connexion :



Debian 12

Déja nous allons commencer par les informations de base :

MODIFIER CONNEXION Nom: Debian 12 Lieu: ROOT Protocole: SSH

Nous nous connectons à notre debian via le protocole SSH, donc dans la liste déroulante Protocole nous sélectionnons SSH.

Ensuite il y a les paramètres de connexion :

F	PARAMÈTRES					
	Réseau					
	Nom d'hôte:	192.168.220.129				
	Port:	2330				
	Clé publique de l'hôte (Base64):					

Authentification

Identifiant:	root	
Mot de passe:	•••••	<u> </u>
Clé privée:		
		li
Phrase secrète:		<u> </u>
		-

--- -

Le nom d'hôte est l'adresse IP de notre debian, le port est le port SSH par défaut est le port 22 mais nous pouvons le changer si nous le souhaitons avec la commande sudo nano /etc/ssh/sshd_config et nous modifions la ligne #Port 22 en Port 2330 par exemple.

Ensuite nous avons les paramètres d'authentification. - *Nom d'utilisateur* : root - *Mot de passe* : Votre mot de passe

Puis tout en bas nous pouvons cliquer sur Enregistrer et notre première machine est ajoutée.

Windows 10

Nous allons commencer par les informations de base :

MODIFIER	CONNEXION
Nom:	Windows 10
Lieu:	ROOT
Protocole:	RDP
déroulante Pr	utilisé pour se connecter à notre machine windows est le RDP, donc dans la liste rotocole nous sélectionnons RDP. les paramètres de connexion :
PARAMÈ	TRES
Réseau	
Nom d'h	nôte: 192.168.220.130
Port:	3389
Authent	ification

Identifiant:	dan.sebag1007@gmail.com	
Mot de passe:	•••••	a
Nom de domaine:		
Mode de Sécurité:		~
Désactiver l'authentification:		
Ignorer le certificat du serveur:	\square	

Nous configurons le clavier et la langue de notre machine windows et nous activons la console d'administration.

Paramètres de base

	Programme de démarrage:	
	Nom du Client:	
	Agencement clavier:	Français (Azerty)
	Fuseau horaire:	Europe - Paris -
	Enable multi-touch:	
	Console Administrateur:	
Puis	s nous pouvons enregistrer notre macl	nine.
Pou	ır que windows RDP fonctionne il faudr	a faire quelques modifications sur notre debian.
	us allons créer un compte utilisateur su acd :	ur notre debian qui sera utilisé pour exécuter le service
m	seradd -M -d /var/lib/guacd/ -r -s /sbin/nologi nkdir /var/lib/guacd hown -R guacd: /var/lib/guacd	n -c "Guacd User" guacd
m cl	nkdir /var/lib/guacd	
m cl	nkdir /var/lib/guacd hown -R guacd: /var/lib/guacd	teur du service guacd :
Ens	nkdir /var/lib/guacd hown -R guacd: /var/lib/guacd suite, nous allons mettre à jour l'utilisa	teur du service guacd : /guacd.service
Ens Ens	nkdir /var/lib/guacd hown -R guacd: /var/lib/guacd uite, nous allons mettre à jour l'utilisa ed -i 's/daemon/guacd/' /etc/systemd/system	teur du service guacd : /guacd.service
Ens Signature of the state of t	nkdir /var/lib/guacd hown -R guacd: /var/lib/guacd uite, nous allons mettre à jour l'utilisa ed -i 's/daemon/guacd/' /etc/systemd/system	teur du service guacd : /guacd.service

Revision #13 Created 4 October 2024 18:15:23 by danette10 Updated 4 October 2024 21:16:13 by danette10