

Apache Guacamole un bastion d'administration

- INSTALLER LES PRÉREQUIS
- TÉLÉCHARGER ET COMPILER GUACAMOLE SERVER
- INSTALLER GUACAMOLE CLIENT
- CONFIGURER GUACAMOLE
- ACCÉDER À L'INTERFACE
- AJOUTER DES CONNEXIONS
- SÉCURISER L'ACCÈS



DOCUMENT CREATED
BY ERICSSON

Étape 1 : Installer les prérequis d'Apache Guacamole

Installer les prérequis

Guacamole nécessite plusieurs composants :

```
apt-get update
apt-get install build-essential libcairo2-dev libjpeg62-turbo-dev libpng-dev libtool-bin uuid-dev libossp-uuid-dev libavcodec-
dev libavformat-dev libavutil-dev libswscale-dev freerdp2-dev libpango1.0-dev libssh2-1-dev libtelnet-dev libvncserver-dev
libwebsockets-dev libpulse-dev libssl-dev libvorbis-dev libwebp-dev
```



```
apt-get update
apt-get install build-essential libcairo2-dev libjpeg62-turbo-dev libpng-dev libtool-bin uuid-dev libossp-uuid-dev libavcodec-dev
libavformat-dev libavutil-dev libswscale-dev freerdp2-dev libpango1.0-dev libssh2-1-dev libtelnet-dev libvncserver-dev
libwebsockets-dev libpulse-dev libssl-dev libvorbis-dev libwebp-dev
```

On attend gentiment la fin de l'installation.

La partie "cliente" d'Apache Guacamole nécessite d'installer un serveur Tomcat, mais nous allons effectuer cette opération plus tard.

Pour effectuer l'installation depuis un compte utilisateur, sans utiliser le compte "**root**" directement, pensez à installer "**sudo**" et à ajouter un utilisateur au groupe correspondant. L'exemple ci-dessous donne les permissions à l'utilisateur "**flo**"

```
apt-get install sudo
usermod -aG sudo flo
```

Ensuite, préfixez par "**sudo**" les commandes qui nécessitent une élévation de privilèges.

```
sudo apt-get update
```



```
apt-get install sudo
usermod -aG sudo flo
sudo apt-get update
```

Étape 3 : Compiler et installer Apache Guacamole "Server"

On va se positionner dans le répertoire "/tmp" et télécharger l'archive tar.gz :

```
cd /tmp
wget https://downloads.apache.org/guacamole/1.5.5/source/guacamole-server-1.5.5.tar.gz
```

Une fois le téléchargement terminé, on décompresse l'archive tar.gz et on se positionne dans le répertoire obtenu :

```
tar -xzf guacamole-server-1.5.5.tar.gz
cd guacamole-server-1.5.5/
```



```
cd /tmp
wget https://downloads.apache.org/guacamole/1.5.5/source/guacamole-server-1.5.5.tar.gz
tar -xzf guacamole-server-1.5.5.tar.gz
cd guacamole-server-1.5.5/
```

On exécute la commande ci-dessous pour se préparer à la compilation, ce qui va permettre de vérifier la présence des dépendances :

```
sudo ./configure --with-systemd-dir=/etc/systemd/system/
```



```
sudo ./configure --with-systemd-dir=/etc/systemd/system/
```

Regardez bien la sortie de la commande précédente, afin de vérifier la présence éventuelle d'une erreur. Si vous obtenez une erreur qui spécifie "**guacenc_video_alloc**", c'est lié au composant "**guacenc**" qui est utilisé pour créer les enregistrements au format vidéo (lié à FFmpeg). Dans ce cas, vous pouvez relancer la commande précédente en désactivant ce composant :

```
sudo ./configure --with-systemd-dir=/etc/systemd/system/ --disable-guacenc
```

```
sudo make
```

```
sudo make install
```



```
sudo ./configure --with-systemd-dir=/etc/systemd/system/ --disable-guacenc
```

```
sudo make
```

```
sudo make install
```

La commande ci-dessous sert à mettre à jour les liens entre guacamole-server et les librairies (cette commande ne retourne aucun résultat) :

```
sudo ldconfig
```

Ensuite, on va **démarrer le service "guacd"** correspondant à Guacamole et **activer son démarrage automatique**. La première commande sert à prendre en compte le nouveau service.

```
sudo systemctl daemon-reload  
sudo systemctl enable --now guacd
```

```
sudo systemctl status guacd
```



```
sudo ldconfig
```

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable --now guacd
```

```
sudo systemctl status guacd
```

Étape 4 : Créer le répertoire de configuration

Dernière étape avant de passer à la partie client d'Apache Guacamole, **on crée l'arborescence pour la configuration d'Apache Guacamole**. Cela va donner le répertoire **"/etc/guacamole"** avec les sous-répertoires **"extensions"** et **"lib"**. Nous en aurons besoin par la suite pour mettre en place le stockage des données dans une base de données MariaDB / MySQL.

```
sudo mkdir -p /etc/guacamole/{extensions,lib}
```



```
sudo mkdir -p /etc/guacamole/{extensions,lib}
```

Étape 5 : Installer Guacamole Client (Web App)

Pour la **Web App** correspondante à Apache Guacamole, et donc à la partie cliente, nous avons besoin d'un serveur **Tomcat 9**. J'insiste sur le fait que **Tomcat 10, distribué par défaut via les dépôts de Debian 12, n'est pas pris en charge par Apache Guacamole**. Nous devons **ajouter le dépôt de Debian 11** sur notre machine Debian 12 afin de pouvoir **télécharger les paquets correspondants à Tomcat 9**.

Nous allons ajouter un nouveau fichier source pour Apt. Créez le fichier suivant :

```
sudo nano /etc/apt/sources.list.d/bullseye.list
```

Ajoutez cette ligne, enregistrez et fermez le fichier.

```
deb http://deb.debian.org/debian/ bullseye main
```

Mettez à jour le cache des paquets :

```
sudo apt-get update
```

Effectuez l'**installation des paquets Tomcat 9 sur Debian 12** avec cette commande :

```
sudo apt-get install tomcat9 tomcat9-admin tomcat9-common tomcat9-user
```

```
sudo nano /etc/apt/sources.list.d/bullseye.list
```

```
deb http://deb.debian.org/debian/ bullseye main
```

```
sudo apt-get update
```

```
sudo apt-get install tomcat9 tomcat9-admin tomcat9-common tomcat9-user
```



Puis, nous allons **télécharger la dernière version de la Web App d'Apache Guacamole** depuis le dépôt officiel (même endroit que pour la partie serveur). On se positionne dans **"/tmp"** et on télécharge la Web App, ce qui revient à télécharger un fichier avec l'extension **".war"**. Ici, la **version 1.5.5** est téléchargée.

```
cd /tmp
wget https://downloads.apache.org/guacamole/1.5.5/binary/guacamole-1.5.5.war
```

Une fois que le fichier est téléchargé, on le déplace dans la librairie de Web App de Tomcat9 avec cette commande :

```
sudo mv guacamole-1.5.5.war /var/lib/tomcat9/webapps/guacamole.war
```

Puis, on relance les services Tomcat9 et Guacamole :

```
sudo systemctl restart tomcat9 guacd
```

```
cd /tmp
```

```
wget https://downloads.apache.org/guacamole/1.5.5/binary/guacamole-1.5.5.war
```



```
sudo mv guacamole-1.5.5.war /var/lib/tomcat9/webapps/guacamole.war
```

```
sudo systemctl restart tomcat9 guacd
```

Étape 6 : Base de données MariaDB pour l'authentification

On commence par installer le paquet MariaDB Server :

```
sudo apt-get install mariadb-server
```

Puis, on exécute le script ci-dessous pour **sécuriser un minimum notre instance** (changer le mot de passe root, désactiver les accès anonymes, etc...).

```
sudo mysql_secure_installation
```

Réponses aux questions de l'assistant : pour **Mysql secure install**

- Enter current password for root (enter for none) ➡ ne rien renseigner et faire entrer
- Set root password ? ➡ Entrer le mot de passe du futur admin de mariaDB (exemple : "Adm.2020") (c'est le mot de passe root du serveur mysql !)
- Remove anonymous user ➡ yes
- Disallow root login remotely ? ➡ Yes (on ne pourra se connecter en root qu'en local)
- Remove test database and access to it ? ➡ yes
- Reload privilege tables now ? ➡ yes

Une fois cette étape effectuée, on va se connecter en tant que root à notre instance MariaDB :

```
mysql -u root -p
```

```
sudo apt-get install mariadb-server
```



```
sudo mysql_secure_installation
```

```
mysql -u root -p
```

Ceci est utile pour **créer une base de données et un utilisateur dédié pour Apache Guacamole**. Les commandes ci-dessous permettent de créer la base de données "**guacadb**", avec l'utilisateur "**guaca_nachos**" associé au mot de passe "**P@ssword!**" (adaptez ces valeurs). Cet utilisateur dispose de quelques droits sur la base de données.

```
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;
```

```
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;
```

La suite va consister à **ajouter l'extension MySQL à Apache Guacamole** ainsi que le connecteur correspondant. Encore quelques fichiers à télécharger depuis Internet.

Toujours depuis le dépôt officiel, on télécharge cette extension :

```
cd /tmp
```

```
wget https://downloads.apache.org/guacamole/1.5.5/binary/guacamole-auth-jdbc-1.5.5.tar.gz
```

Puis, on décompresse l'archive tar.gz obtenue :

```
tar -xzf guacamole-auth-jdbc-1.5.5.tar.gz
```

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

```
sudo mv guacamole-auth-jdbc-1.5.5/mysql/guacamole-auth-jdbc-mysql-1.5.5.jar /etc/guacamole/extensions/
```

```
cd /tmp
```

```
wget https://downloads.apache.org/guacamole/1.5.5/binary/guacamole-auth-jdbc-1.5.5.tar.gz
```



```
tar -xzf guacamole-auth-jdbc-1.5.5.tar.gz
```

```
sudo mv guacamole-auth-jdbc-1.5.5/mysql/guacamole-auth-jdbc-mysql-1.5.5.jar /etc/guacamole/extensions/
```

Télécharger le connecteur MySQL

On lance le téléchargement :

```
cd /tmp
```

```
wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-j-9.1.0.tar.gz
```

Puis, on décompresse l'archive tar.gz :

```
tar -xzf mysql-connector-j-9.1.0.tar.gz
```

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

```
sudo cp mysql-connector-j-9.1.0/mysql-connector-j-9.1.0.jar /etc/guacamole/lib/
```

```
cd /tmp
```

```
wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-j-9.1.0.tar.gz
```



```
tar -xzf mysql-connector-j-9.1.0.tar.gz
```

```
sudo cp mysql-connector-j-9.1.0/mysql-connector-j-9.1.0.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-1.5.5/mysql/schema/"**. Le mot de passe root de MariaDB doit être saisi pour effectuer l'import.

```
cd guacamole-auth-jdbc-1.5.5/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
```

Dans ce fichier, insérez les lignes ci-dessous en adaptant les trois dernières lignes avec vos valeurs :

```
# MySQL  
mysql-hostname: 127.0.0.1  
mysql-port: 3306  
mysql-database: guacadb  
mysql-username: guaca_nachos  
mysql-password: P@ssword!
```

```
cd guacamole-auth-jdbc-1.5.5/mysql/schema/  
cat *.sql | mysql -u root -p guacadb
```

```
sudo nano /etc/guacamole/guacamole.properties
```

```
# MySQL  
mysql-hostname: 127.0.0.1  
mysql-port: 3306  
mysql-database: guacadb  
mysql-username: guaca_nachos  
mysql-password: P@ssword!
```



Tant que l'on est dans la configuration, **éditez 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 code à intégrer :

```
[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
```

Voilà, l'installation de base est terminée !

```
sudo nano /etc/guacamole/guacd.conf
```

```
[server]
```

```
bind_host = 0.0.0.0
```

```
bind_port = 4822
```

```
sudo systemctl restart tomcat9 guacd mariadb
```

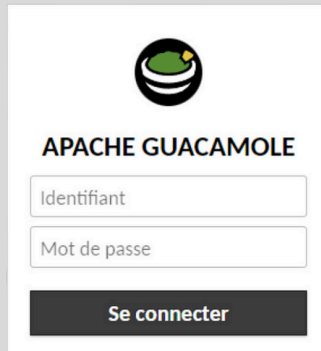


Étape 7 : Premiers pas avec Apache Guacamole

On va pouvoir se connecter à Apache Guacamole pour effectuer nos premiers pas sur l'interface de la Web App.

```
http://<Adresse IP>:8080/guacamole/
```

Une page de connexion va s'afficher :



Pour se connecter, on va utiliser les identifiants par défaut :

- Utilisateur : **guacadmin**
- Mot de passe : **guacadmin**

Étape 8 : Créer un nouveau compte admin

Tout d'abord, nous allons **créer un nouveau compte d'administration** et supprimer le compte par défaut, pour des raisons de sécurité.

Notre objectif est le suivant :

- Créer un nouveau compte administrateur (avec un nom personnalisé)
- Se déconnecter du compte "**guacadmin**"
- Se reconnecter avec le nouveau compte administrateur
- Supprimer le compte "**guacadmin**" par défaut (ou à minima changer son mot de passe et le désactiver)

Pour accéder aux paramètres, il faut cliquer sur le nom d'utilisateur en haut à droite puis sur "Paramètres".

[←](#) [→](#) [↻](#) ⚠ Non sécurisé | ██████████:8080/guacamole/#/settings/users

PARAMÈTRES

Sessions ActivesHistoriqueUtilisateursGroupesConnexionsPréférences

Cliquez ou appuyez sur un utilisateur en dessous pour le gérer. Selon vos permissions, les utilisateurs peuvent être ajoutés:

Nouvel Utilisateur

Filtre

Identifiant ▾	Organisation
guacadmin	

Un formulaire s'affiche. **Indiquez un nom d'utilisateur**, en évitant les traditionnels "Administrateur", "Admin", etc.... Et choisissez **un mot de passe robuste**. Cochez l'ensemble des permissions pour que cet utilisateur soit administrateur de la plateforme Guacamole.

MODIFIER UTILISATEUR

Identifiant:

Mot de passe:

Répéter mot de passe:

PROFIL

Nom:

Adresse Mail:

Organisation:

Rôle:

RESTRICTIONS DE COMPTE

Connexion désactivée: ☐

Mot de passe expiré: ☐

Autoriser l'accès après:

Ne pas autoriser l'accès après:

Activer le compte après:

Désactiver le compte après:

Fuseau horaire utilisateur:

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: ☒

PARAMÈTRES

Réseau

Nom d'hôte:

Port:

Authentification

Identifiant:

Mot de passe:

Nom de domaine:

Mode de Sécurité:

Désactiver l'authentification: ☐

Ignorer le certificat du serveur: ☒

Passerelle du bureau à distance

Nom d'hôte:

Port:

Identifiant:

Mot de passe:

Nom de domaine:

Paramètres de base

Programme de démarrage:

Nom du Client:

Agencement clavier:

Fuseau horaire:

Enable multi-touch: ☐

Console Administrateur: ☐

Il y a de **nombreuses options disponibles**, notamment pour faire remonter les périphériques locaux, ou passer par une passerelle de bureau à distance. Au début, il faut passer du temps à trouver la bonne formule pour que la connexion RDP intègre toutes les fonctions dont on a besoin. Toutefois, si l'on veut simplement se connecter et avoir le contrôle à distance, ce n'est pas utile de modifier profondément la configuration.

Pour que l'expérience soit un peu plus agréable, on peut cocher les options ci-dessous (mais cela reste facultatif - *testez avec et sans*) :

Performance

Activer fond d'écran: ☒

Activer thématisation: ☒

Activer le lissage des polices (ClearType): ☒

Activer pleine fenêtre de glisser: ☒

Activer la composition du bureau (Aero): ☒

Activer les animations de menu: ☒

Désactiver le cache bitmap: ☐

Désactiver le cache hors écran : ☐

Désactiver le cache glyph: ☐

Étape 9 : Apache Guacamole : erreur de connexion en RDP

Que faire si la connexion RDP ne se lance pas ou qu'elle affiche une erreur ?

Retournez sur la ligne de commande de votre serveur et **vérifiez les dernières lignes de logs** qui s'affichent lorsque l'on regarde le statut du service guacd :

```
sudo systemctl status guacd
```

Si le certificat RDP ne peut pas être vérifié (auto-signé par exemple) et que l'option "**Ignorer le certificat du serveur**" n'est pas cochée dans les paramètres de la connexion Guacamole, alors cette erreur se produira.

Une autre erreur que vous pourriez rencontrer **si vous avez besoin d'établir des connexions en RDP**, c'est celle-ci :

```
RDP server closed/refused connection: Security negotiation failed (wrong security type?)
```

```
fl@srv-guacamole:/tmp/guacamole-auth-jdbc-1.5.5/mysql/schema$ sudo journalctl -f -u guacd
nov. 08 16:30:49 srv-guacamole guacd[35429]: Loading keymap "fr-fr-azerty"
nov. 08 16:30:49 srv-guacamole guacd[35429]: guacd[35429]: INFO: Loading keymap "fr-fr-azerty"
nov. 08 16:30:49 srv-guacamole guacd[35429]: RDP server closed/refused connection: Security negotiation failed (wrong security type?)
nov. 08 16:30:49 srv-guacamole guacd[35429]: guacd[35429]: INFO: RDP server closed/refused connection: Security negotiation failed (wrong security type?)
nov. 08 16:30:49 srv-guacamole guacd[35429]: User "@eed24332-87a7-474f-b0ea-5d5e02f178c2" disconnected (0 users remain)
nov. 08 16:30:49 srv-guacamole guacd[35429]: Last user of connection "$56155ddc-8d64-41a5-ac7d-9ae42b19d121" disconnected
nov. 08 16:30:49 srv-guacamole guacd[35429]: guacd[35429]: INFO: User "@eed24332-87a7-474f-b0ea-5d5e02f178c2" disconnected (0 users remain)
nov. 08 16:30:49 srv-guacamole guacd[35429]: guacd[35429]: INFO: Last user of connection "$56155ddc-8d64-41a5-ac7d-9ae42b19d121" disconnected
nov. 08 16:30:49 srv-guacamole guacd[31513]: Connection "$56155ddc-8d64-41a5-ac7d-9ae42b19d121" removed.
nov. 08 16:30:49 srv-guacamole guacd[31513]: guacd[31513]: INFO: Connection "$56155ddc-8d64-41a5-ac7d-9ae42b19d121" removed.
```

Ce problème est lié au compte utilisateur "**daemon**" utilisé par défaut pour exécuter le service "**guacd**". Vous pouvez le vérifier avec cette commande :

```
sudo useradd -M -d /var/lib/guacd/ -r -s /sbin/nologin -c "Guacd User" guacd
sudo mkdir /var/lib/guacd
sudo chown -R guacd: /var/lib/guacd
sudo sed -i 's/daemon/guacd/' /etc/systemd/system/guacd.service
sudo systemctl daemon-reload
sudo systemctl restart guacd
```

Puis, vérifiez l'état du service :

```
sudo systemctl status guacd
```

```
sudo systemctl status guacd
```

```
sudo useradd -M -d /var/lib/guacd/ -r -s /sbin/nologin -c "Guacd User" guacd
```

```
sudo mkdir /var/lib/guacd
```

```
sudo chown -R guacd: /var/lib/guacd
```

```
sudo sed -i 's/daemon/guacd/' /etc/systemd/system/guacd.service
```

```
sudo systemctl daemon-reload
```

```
sudo systemctl restart guacd
```

```
sudo systemctl status guacd
```

