

Guide d'installation d'Apache Guacamole

Apache Guacamole est une passerelle d'accès à distance « sans client » (via navigateur) pour RDP, SSH et VNC. Ce guide propose une installation rapide en production légère via Docker Compose (recommandée), ainsi qu'une alternative d'installation manuelle.

1. Prérequis

- **Serveur Linux** (Ubuntu/Debian recommandés) avec accès sudo.
- **Ressources** : 1 vCPU / 2 Go RAM minimum (plus selon le nombre de sessions).
- **Réseau** : accès sortant aux dépôts, et accès depuis les utilisateurs (HTTP/HTTPS). Ouvrir **TCP 443** (ou 80) côté pare-feu.
- **Nom de domaine** (optionnel mais conseillé) : ex. *guacamole.exemple.tld*.
- **Certificat TLS** (conseillé en production) : via reverse proxy (Nginx/Traefik/Caddy) ou terminaison TLS en amont.

2. Installation (recommandée) avec Docker Compose

Cette méthode déploie trois services : **guacd** (proxy Guacamole), **guacamole** (application web) et une base **PostgreSQL** pour l'authentification et la configuration.

2.1 Installer Docker et Docker Compose

Installez Docker Engine et le plugin Compose selon votre distribution. Sur Ubuntu, la voie la plus fiable est d'utiliser le dépôt officiel Docker. Vérifiez ensuite :

```
docker --version
docker compose version
```

2.2 Créer le répertoire du projet

```
sudo mkdir -p /opt/guacamole
cd /opt/guacamole
```

2.3 Créer le fichier docker-compose.yml

Créez un fichier **docker-compose.yml** (adapter les mots de passe) :

```
version: "3.8"

services:
  guacd:
    image: guacamole/guacd:latest
    restart: unless-stopped
```

```
postgres:
image: postgres:16
restart: unless-stopped
environment:
POSTGRES_DB: guacamole_db
POSTGRES_USER: guacamole_user
POSTGRES_PASSWORD: ChangeMe_StrongPassword
volumes:
- postgres_data:/var/lib/postgresql/data
```

```
guacamole:
image: guacamole/guacamole:latest
restart: unless-stopped
depends_on:
- guacd
- postgres
environment:
GUACD_HOSTNAME: guacd
GUACD_PORT: 4822
POSTGRES_HOSTNAME: postgres
POSTGRES_DATABASE: guacamole_db
POSTGRES_USER: guacamole_user
POSTGRES_PASSWORD: ChangeMe_StrongPassword
ports:
- "8080:8080"

volumes:
postgres_data:
```

2.4 Initialiser la base de données

1. Démarrez uniquement PostgreSQL :

```
sudo docker compose up -d postgres
```

1. Récupérez le script d'initialisation fourni par Guacamole :

```
sudo docker run --rm guacamole/guacamole:latest /opt/guacamole/bin/initdb.sh -
-postgres > initdb.sql
```

1. Importez le schéma dans PostgreSQL :

```
sudo docker compose exec -T postgres psql -U guacamole_user -d guacamole_db <
initdb.sql
```

Astuce : une fois importé, vous pouvez conserver *initdb.sql* pour audit ou le supprimer.

2.5 Démarrer Guacamole et se connecter

```
sudo docker compose up -d
```

Accédez ensuite à : **http://<IP_ou_DNS>:8080/guacamole**.

Identifiants initiaux (par défaut) : **guacadmin / guacadmin**. Changez immédiatement le mot de passe et/ou désactivez ce compte si vous mettez en place un SSO/LDAP.

3. Mettre Guacamole derrière HTTPS (conseillé)

- Placez un **reverse proxy** (Nginx, Traefik, Caddy) devant le port 8080 et terminez TLS (Let's Encrypt, PKI interne, etc.).
- Publiez idéalement sur **443** et redirigez **80** → **443**.
- Ajoutez des en-têtes de sécurité usuels (HSTS, X-Content-Type-Options, etc.) selon votre politique.

4. Configuration de base

1. **Créer un utilisateur** (ou groupes) : *Paramètres* → *Utilisateurs*.
2. **Créer une connexion** : *Paramètres* → *Connexions* → *Nouvelle connexion*, puis choisissez le protocole (RDP/SSH/VNC).
3. **Renseigner la cible** : hôte, port, identifiants (ou clés SSH), et options (clipboard, résolution, audio selon protocole).
4. **Affecter les droits** : donnez l'accès à la connexion aux utilisateurs/groupe nécessaires.

5. Alternative : installation “manuelle” (sans Docker)

En mode “manuel”, vous installez : **guacd** (daemon), **Tomcat** (ou autre conteneur servlet Java) pour l'application Guacamole, et une **base de données** (PostgreSQL/MySQL) + extensions d'authentification (JDBC, LDAP, etc.). Cette approche demande plus de maintenance (versions Java/Tomcat, librairies, compilation de guacd selon l'OS).

- Installer les dépendances système (build-essential, bibliothèques RDP/VNC/SSH selon vos besoins).
- Installer **guacd** (paquets de la distribution ou compilation) et l'activer en service.
- Installer **Tomcat** + Java, puis déployer le fichier *guacamole.war*.
- Configurer **GUACAMOLE_HOME** (propriétés, extensions, pilote JDBC) et initialiser la base (script fourni).

6. Sécurité et bonnes pratiques

- **HTTPS obligatoire** en production (cookies, identifiants, sessions).
- **Comptes** : changez le mot de passe admin par défaut, appliquez une politique de mots de passe, limitez les droits.
- **Authentification centralisée** : privilégiez LDAP/SAML/OIDC (si disponible dans votre contexte) plutôt que des comptes locaux.
- **Mises à jour** : mettez à jour régulièrement les images/paquets (guacamole, guacd, base, reverse proxy).
- **Réseau** : restreignez les flux sortants de Guacamole vers les sous-réseaux cibles (RDP/SSH/VNC) ; évitez l'accès "vers tout".
- **Logs** : centralisez et surveillez les journaux (authentications, erreurs guacd, reverse proxy).

7. Dépannage (check-list)

- **Service up ?** : `docker compose ps`
- **Logs** : `docker compose logs -f guacamole` et `docker compose logs -f guacd`
- **Base** : vérifiez les variables JDBC et que le schéma est importé (erreurs d'auth DB = identifiants/host/port).
- **Connexion RDP/SSH/VNC** : testez la connectivité réseau depuis le serveur Guacamole (pare-feu, routes, DNS).
- **Reverse proxy** : en cas de boucle de redirection/erreurs 502, vérifiez la cible (8080), les en-têtes et le TLS.