CENTRE HOSPITALIER
DU PAYS D'APT

Documentation Bastion Guacamole

1. Présentation du bastion Guacamole

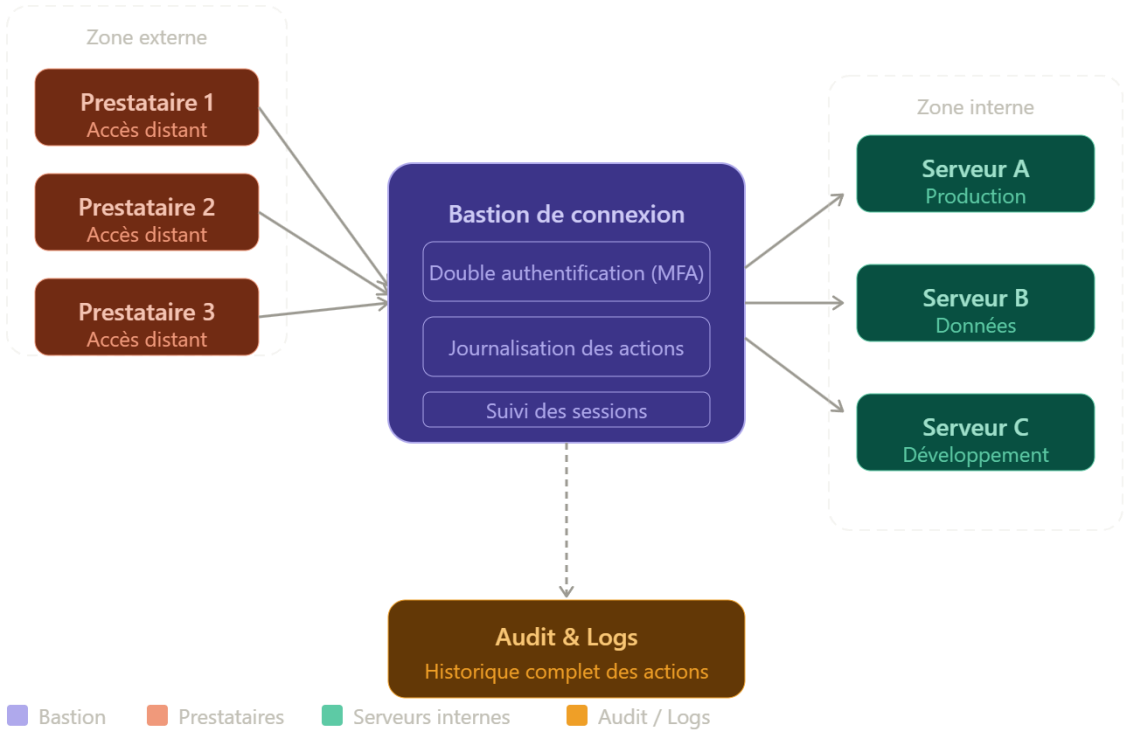
1.1 Qu'est-ce qu'un bastion ?

Un serveur bastion est un point d'entrée unique et sécurisé placé entre les utilisateurs et les machines internes du réseau. Tout accès distant transite par ce point de contrôle centralisé.

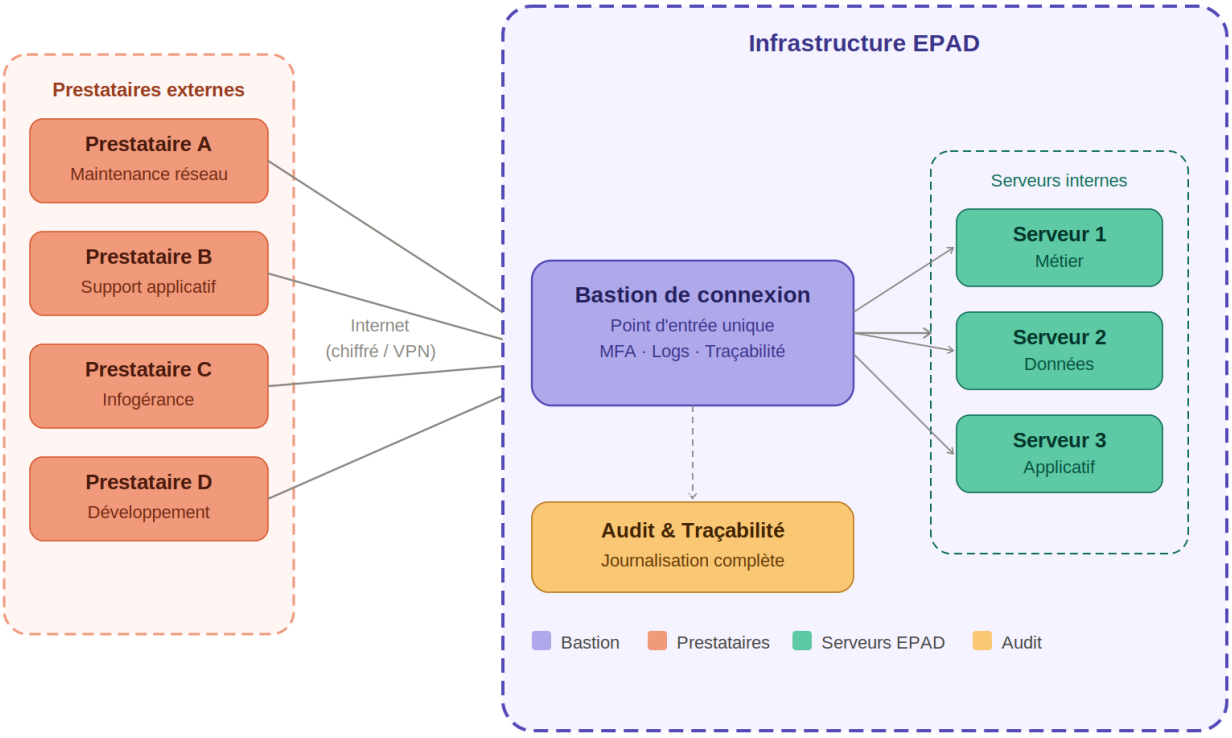
- Centraliser et auditer l'ensemble des sessions distantes (SSH, RDP, VNC)
- Réduire la surface d'attaque en limitant les ports exposés
- Appliquer des politiques d'authentification forte (MFA, LDAP/AD)
- Enregistrer les sessions pour des besoins d'audit et de conformité
- Gérer les droits d'accès par utilisateur ou groupe

1.2 Utilités principales d'un bastion pour L'hôpital ?

Le bastion de connexion joue le rôle de point d'entrée unique et contrôlé entre l'extérieur et votre infrastructure interne. Il contrôle les accès privilégiés, filtre le trafic, enregistre toutes les sessions pour l'audit, protège les identifiants et réduit la surface d'attaque en empêchant les connexions directes aux serveurs internes.
Les prestataires ne se connectent jamais directement aux serveurs.
Ils passent obligatoirement par ce point de passage.



Le bastion de connexion a été déployé au sein de l'EHPAD et permet aux prestataires et entreprises extérieurs de se connecter de manière sécurisée à nos serveurs internes.



1.3 Apache Guacamole

Apache Guacamole est une passerelle de bureau à distance sans client (clientless). L'utilisateur accède aux machines distantes depuis un simple navigateur web, sans installer aucun logiciel supplémentaire. Protocols supportés : RDP, SSH, VNC, Telnet, Kubernetes.

Composant	Technologie	Rôle
guacd	C (natif)	Démon proxy — traduit les protocoles RDP/SSH/VNC en Guacamole Protocol
guacamole-client	Java / Tomcat 10	Application web — interface HTML5 utilisateur
Base de données	MariaDB / MySQL	Stockage : utilisateurs, connexions, groupes, historique
Reverse proxy	Nginx	Exposition HTTPS, terminaison TLS, redirection vers Tomcat
Extensions	JAR Java	Plugins : TOTP, LDAP, Duo, Quick Connect, enregistrement sessions

1.4 Le projet Itiligent Easy-Guacamole-Installer

Déploiement réalisé via le projet open source disponible sur :

https://github.com/itiligent/Easy-Guacamole-Installer

Ce projet propose un ensemble de scripts shell couvrant l'installation complète et la gestion du cycle de vie du bastion.

1.5 Systèmes supportés

- Debian 12 (Bookworm) et Debian 13 (Trixie)
- Ubuntu LTS 22.x et 24.x
- Raspbian

1.6 Architecture déployée

```
Navigateur → Nginx (:443 HTTPS) → Tomcat (:8080) → guacd (:4822) → Machine cible
```

Composant	Port	Description
Nginx (HTTPS)	443	Point d'entrée principal — reverse proxy TLS
Tomcat / Guacamole	8080	Application web Java (via Nginx ou direct)
guacd	4822	Démon proxy natif — connexions aux machines distantes
MariaDB	3306	Base de données locale
SSH	22	Administration du serveur bastion lui-même

> *Les ports 80, 8080 et 443 doivent être Ouvert avant l'installation. Aucun autre service ne doit les utiliser.*

2. Prérequis

2.1 Ressources système

Ressource	Minimum recommandé
CPU	1 cœur par tranche de 25 utilisateurs simultanés
RAM	2 Go par tranche de 25 utilisateurs (+ RAM système de base)
Stockage	Espace système + espace pour les enregistrements de sessions
Connexion Internet	Requise pour télécharger les paquets et scripts
OS	Debian 13 (Trixie) — installation minimale recommandée

2.2 Ports réseau à ouvrir

Port TCP	Protocole	Obligatoire ?	Usage
22	SSH	Oui	Administration du serveur bastion
80	HTTP	Oui	Redirect HTTP vers HTTPS / accès initial
443	HTTPS	Oui	Accès sécurisé à l'interface Guacamole
8080	HTTP	Non (interne)	Accès natif Guacamole sans Nginx

2.3 DNS

- Un enregistrement DNS privé est requis pour le reverse proxy TLS avec Nginx
- Un enregistrement DNS public est requis si vous utilisez Let's Encrypt

> *Sans DNS, utilisez un certificat auto-signé avec l'adresse IP du serveur.*

2.4 Préparation de l'utilisateur

Le script doit être exécuté par un utilisateur NON root disposant des droits sudo. Si besoin, ajouter l'utilisateur au groupe sudo (en root) :

```
usermod -aG sudo nom_utilisateur
```

Vérifiez ensuite :

```
sudo -v
```

> *Ne jamais lancer 1-setup.sh directement en root. Le script demande sudo quand nécessaire.*

3. Maintenance

Script	Description
backup-guacamole.sh	Sauvegarde automatique (cron) de la base MariaDB
upgrade-guacamole.sh	Mise à jour de Guacamole, extensions et connecteur MySQL
branding.jar	Modèle de personnalisation du thème de l'interface

> *Les scripts upgrade et backup sont automatisément personnalisés avec les paramètres spécifiques de votre installation pour garantir la cohérence lors des mises à jour futures.*

4. Installation pas à pas

4.1 Téléchargement du script

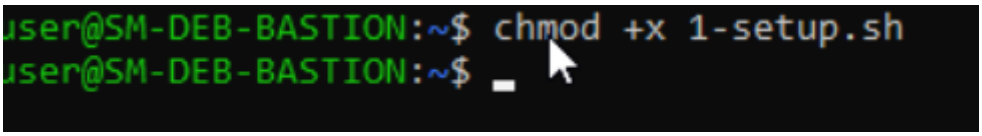
Téléchargez le script principal depuis le dépôt GitHub :

```
wget https://raw.githubusercontent.com/itiligent/Guacamole-Install/main/1-setup.sh
```

Téléchargement du script 1-setup.sh depuis GitHub

Rendez le script exécutable :

```
chmod +x 1-setup.sh
```

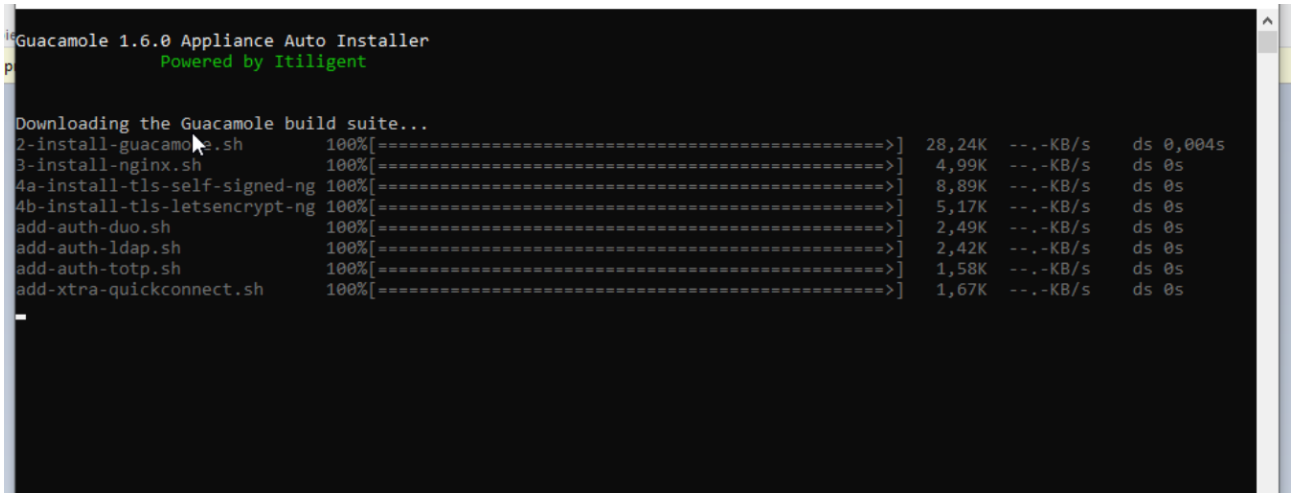


Ajout des droits d'exécution sur 1-setup.sh

4.2 Exécution de l'installateur

```
./1-setup.sh
```

L'installateur télécharge immédiatement la suite complète des scripts (Guacamole, Nginx, TLS, extensions) :

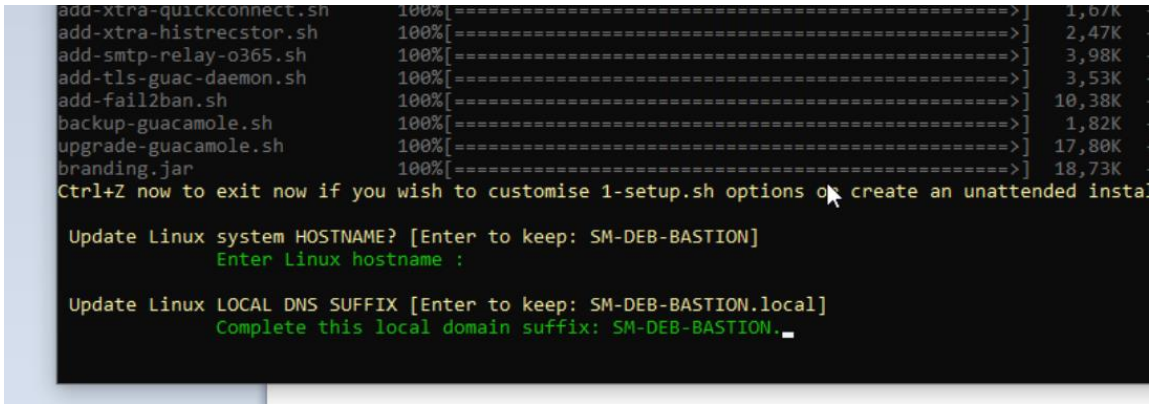


Téléchargement de la suite de scripts Guacamole 1.6.0 par l'installateur Itiligent

4.3 Configuration interactive

Hostname et suffixe DNS

Le script demande le nom d'hôte et le suffixe DNS local. Ces valeurs doivent être cohérentes avec votre infrastructure pour que le reverse proxy TLS fonctionne correctement.

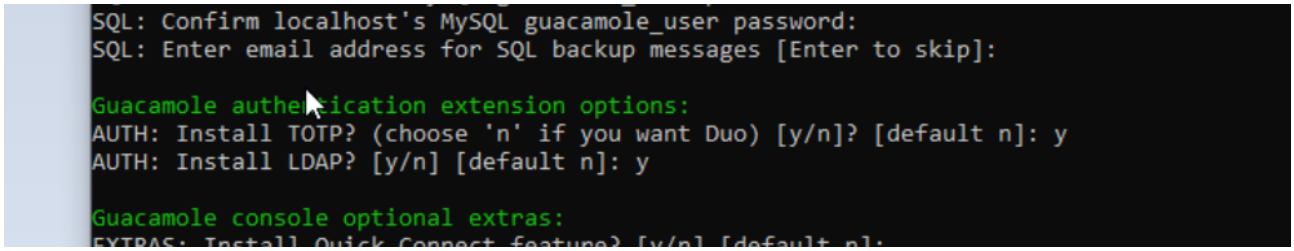


Configuration du hostname (SM-DEB-BASTION) et du suffixe DNS local

Confirmation des paramètres hostname / DNS suffix

Options d'authentification et de configuration

Le script guide l'installation en plusieurs étapes interactives:



Section "Guacamole authentication extension options" — TOTP (y) et LDAP (y) activés

> *Les options TOTP et LDAP peuvent être ajoutées ou modifiées après l'installation avec add-auth-totp.sh et add-auth-ldap.sh.*

> *L'installation complète (compilation de guacd, Tomcat, MariaDB, Nginx, extensions) peut prendre 10 à 30 minutes selon la puissance de la machine.*

5. Accès à l'interface web

5.1 URLs d'accès

Accès	URL
Via Nginx (HTTPS)	https://x.x.x.x.x.x.x.x/guacamole/
Natif Tomcat	http://x.x.x.x:8080/guacamole/

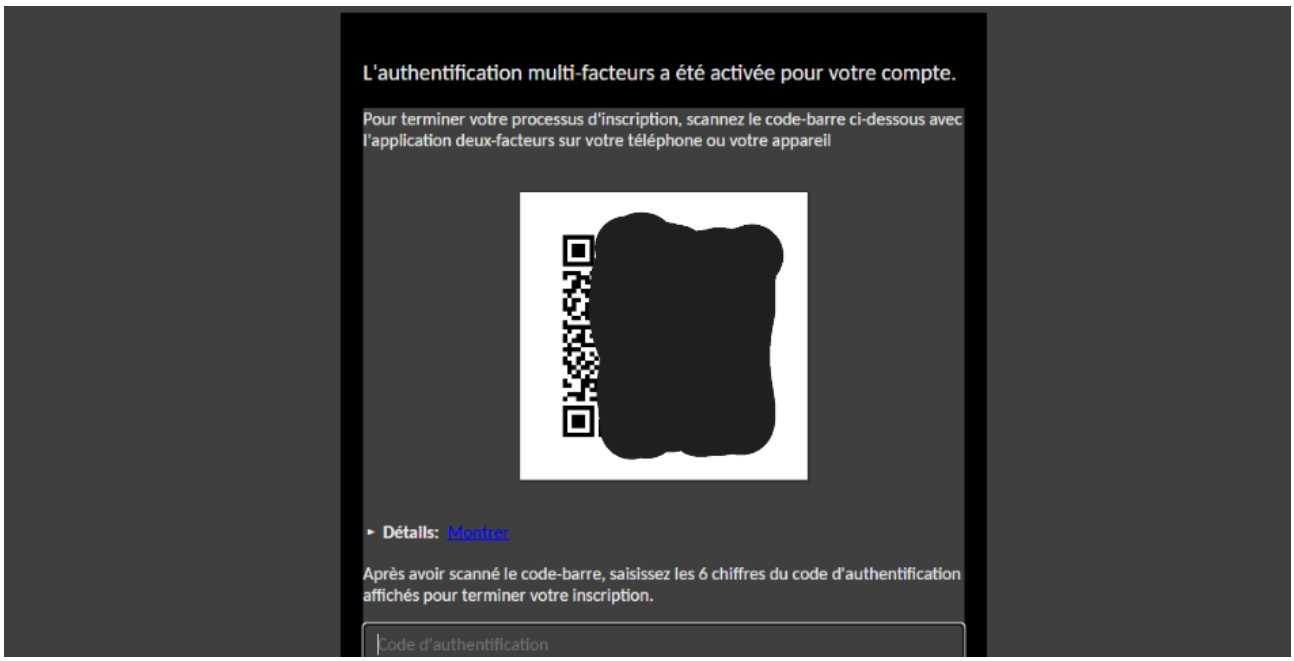
> *Le compte administrateur par défaut est guacadmin. Changez son mot de passe immédiatement via Paramètres > Préférences.*

5.2 Page de connexion

Interface de connexion Guacamole (thème Itiligent) — x.x.x.x:8080/guacamole/

5.3 Enregistrement TOTP

À la première connexion avec TOTP activé, Guacamole affiche un QR code à scanner avec une application d'authentification (Google Authenticator, Authy, Microsoft Authenticator…). Saisissez ensuite le code à 6 chiffres pour valider.



Enregistrement TOTP — scanner le QR code avec l'application d'authentification

> *Le QR code TOTP ne s'affiche qu'une seule fois. En cas de perte, un admin doit réinitialiser la clé TOTP dans la base de données.*

5.4 Dashboard principal

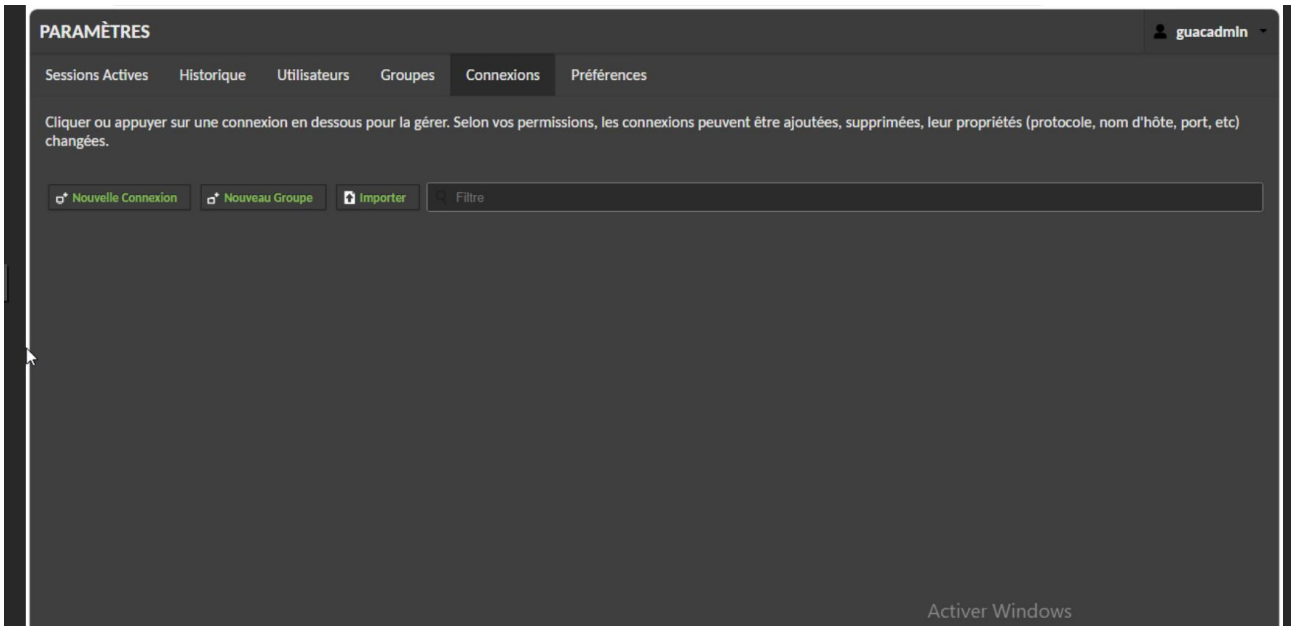
Après authentification, le tableau de bord liste les connexions récentes et toutes les connexions disponibles.

Tableau de bord Guacamole après connexion — aucune connexion configurée initialement

6. Administration du bastion

6.1 Panneau d'administration

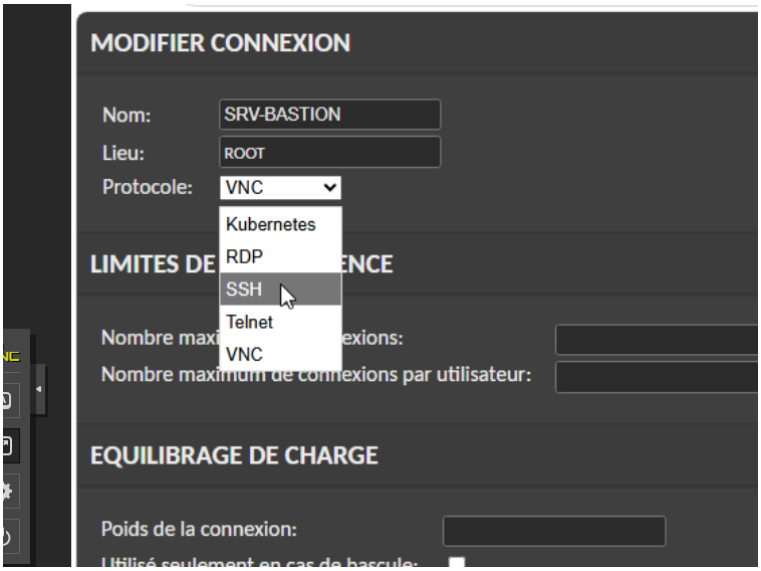
Cliquez sur le nom de l'utilisateur en haut à droite, puis sur Paramètres. Les onglets disponibles sont : Sessions Actives, Historique, Utilisateurs, Groupes, Connexions, Préférences.



Panneau d'administration — onglet Connexions

6.2 Création d'une connexion SSH

Cliquez sur Nouvelle Connexion et renseignez les paramètres. Dans cet exemple, une connexion SSH vers SRV-BASTION est créée :



Création d'une connexion SSH SRV-BASTION — sélection du protocole dans la liste

Paramètre	Description	Exemple
Nom	Nom affiché dans le tableau de bord	SRV-BASTION
Lieu	Groupe de rattachement	ROOT
Protocole	SSH / RDP / VNC / Telnet / Kubernetes	SSH
Hostname	IP ou nom DNS de la machine cible	x.x.x.x
Port	22 (SSH) | 3389 (RDP) | 5900 (VNC)	22
Utilisateur	Compte de connexion sur la machine cible	admin
Authentification	Mot de passe ou clé SSH privée	Clé SSH recommandée

> *Utilisez des clés SSH privées plutôt que des mots de passe pour plus de sécurité (paramètres avancés de la connexion SSH).*

6.3 Utilisation d'une connexion

Un clic sur la vignette du tableau de bord ouvre directement la session dans le navigateur. Les connexions récentes affichent une prévisualisation.

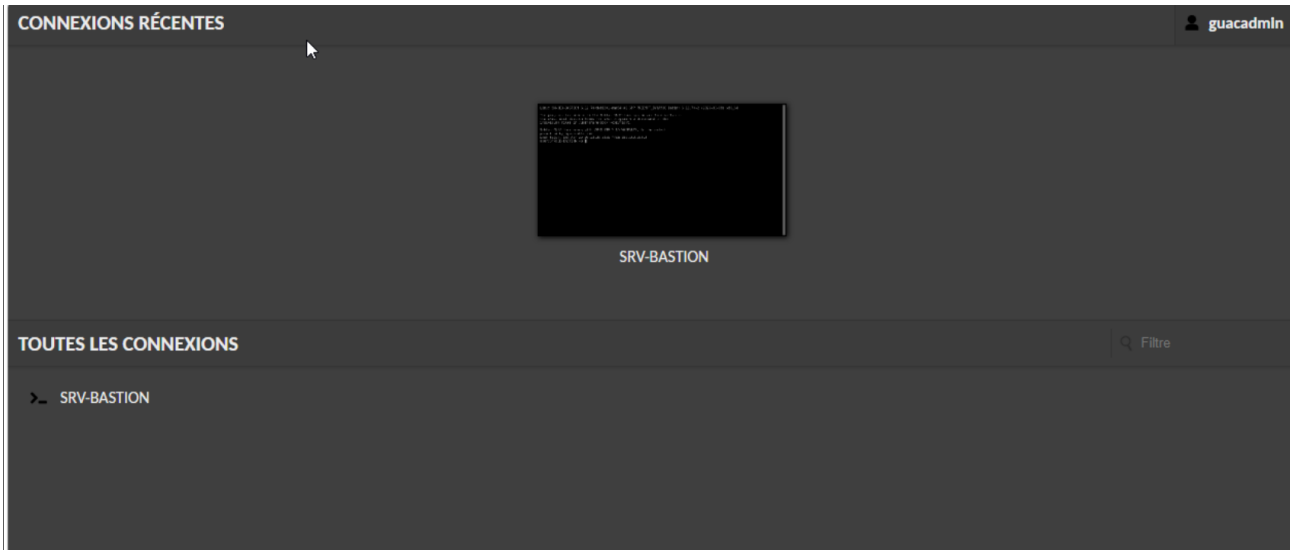


Tableau de bord avec la connexion SSH SRV-BASTION active et récente

6.4 Groupes et utilisateurs

Guacamole permet de créer des groupes de connexions et des groupes d'utilisateurs pour organiser et déléguer les accès. Depuis l'onglet Utilisateurs, créez des comptes et assignez-leur des droits sur des connexions spécifiques.

6.5 Historique des sessions

L'onglet Historique liste toutes les sessions passées avec horodatage, durée, utilisateur et connexion utilisée. Si l'extension History Recorded Storage est installée, les sessions peuvent être relues.

> *L'historique est stocké dans la base MariaDB. Configurez la sauvegarde automatique avec backup-guacamole.sh.*

7. Configuration TLS et sécurité

7.1 Certificat auto-signé

Pour activer HTTPS avec un certificat auto-signé (réseau interne sans DNS public) :

```
bash $HOME/guac-setup/4a-install-tls-self-signed-nginx.sh
```

Les certificats générés sont sauvegardés dans $HOME/guac-setup/tls-certs/[date-heure]/. Le certificat client peut être importé dans le navigateur pour éviter les avertissements. Ce script peut être ré-exécuté pour renouveler.

> *Videz le cache du navigateur après tout changement de certificat TLS.*

7.2 Certificat Let's Encrypt

```
bash $HOME/guac-setup/4b-install-tls-letsencrypt-nginx.sh
```

> *Un enregistrement DNS public pointant vers le serveur est obligatoire pour que Let's Encrypt valide le domaine.*

7.3 Fail2ban — protection anti-bruteforce

```
bash $HOME/guac-setup/add-fail2ban.sh
```

7.4 Chiffrement interne guacd

Pour chiffrer le trafic interne entre Tomcat et guacd en TLS :

```
bash $HOME/guac-setup/add-tls-guac-daemon.sh
```

8. Maintenance

8.1 Mise à jour de Guacamole

Le script upgrade-guacamole.sh est configuré automatiquement avec les paramètres de votre installation (version, base de données, extensions). Il met à jour Guacamole, les extensions et le connecteur MySQL :

```
bash $HOME/guac-setup/upgrade-guacamole.sh
```

8.2 Sauvegarde de la base de données

Configure une tâche cron de sauvegarde quotidienne de la base MariaDB (sous le crontab de l'utilisateur ayant exécuté 1-setup.sh) :

```
bash $HOME/guac-setup/backup-guacamole.sh
```

> *Vérifiez la planification avec : crontab -l*

8.3 Redémarrage des services

```
# guacd + Tomcat
sudo systemctl restart guacd
sudo systemctl restart tomcat10

# Nginx (reverse proxy)
sudo systemctl restart nginx

# MariaDB
sudo systemctl restart mariadb
```

8.5 Personnalisation de l'interface

Un modèle de personnalisation du thème sombre est fourni via branding.jar. Modifiez-le et déposez-le dans le répertoire des extensions :

```
/etc/guacamole/extensions/
```

9. Fonctionnement en mode projet

La mise en place de la solution **Apache Guacamole** (Bastion d'accès distant) a suivi une démarche structurée en quatre phases progressives, permettant de valider chaque étape avant de passer à la suivante.

Phase 1 Recherche et étude préalable

Avant de commencer toute installation, une phase de recherche approfondie a été menée afin de bien comprendre la solution et d'anticiper les besoins techniques.
Cette phase a consisté à :

- Compréhension de la solution
- Analyse des besoins
- Étude technique

Phase 2 Tests en environnement virtuel local

Avant toute intervention sur l'infrastructure de production, une première installation a été réalisée en conditions contrôlées sur un poste de travail personnel, au sein d'une machine virtuelle créée via **VirtualBox.**
Cette approche a permis de :

- Vérifier la faisabilité technique de l'installation de Guacamole
- Tester les configurations réseau, les connexions RDP/SSH et l'interface web
- Identifier et résoudre les éventuels problèmes sans risque pour l'environnement réel
- Valider le bon fonctionnement global de la solution avant tout déploiement

Phase 2 Déploiement sur le serveur hyperviseur

À la suite des tests validés, la solution a été installée sur le **serveur hyperviseur** de l'infrastructure réelle.
Cette phase avait pour objectif de :

- Déployer Guacamole dans un environnement proche de la production
- Observer le comportement de la solution face aux contraintes réelles (ressources, réseau, utilisateurs)
- Mesurer les performances et identifier d'éventuels ajustements nécessaires
- S'assurer de la stabilité et de la compatibilité avec l'infrastructure existante

Phase 3 — Mise en place définitive

Après validation des deux phases précédentes, la solution a été déployée de manière **complète et définitive** sur l'infrastructure. Cette dernière phase comprend la configuration finale, la sécurisation de l'accès et la mise à disposition aux utilisateurs.