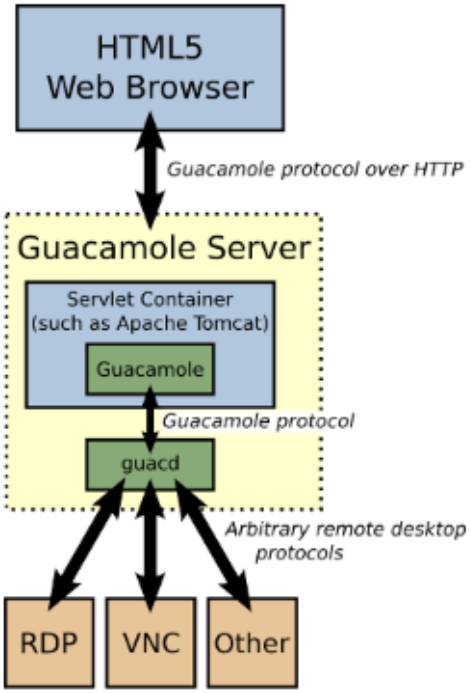
Fuentes - Documentation technique : Bastion

Système : Ubuntu 22.04 LTS (Jammy) | **Version Guacamole :** 1.6.0 | **Moteur RDP :** FreeRDP 3.2.0 | **Entreprise :** SIS 67 – Service d'Incendie et de Secours du Bas-Rhin



> **Contexte — Origine du projet**

> Le système d'information du SIS 67 comporte des serveurs critiques (contrôleurs de domaine, hyperviseurs vSphere, équipements réseau) qui étaient administrés via des connexions RDP directes depuis les postes utilisateurs. Cette approche présentait trois risques majeurs : absence de traçabilité des actions administratives, non-respect du principe de moindre privilège, et incompatibilité avec la politique de sécurité Active Directory imposant le groupe **Protected Users** (qui interdit l'authentification NTLM).

> Ce projet répond à ce besoin en déployant un **bastion d'administration centralisé** basé sur Apache Guacamole — point d'entrée unique, auditable et sécurisé pour l'ensemble des accès aux serveurs sensibles.

Synthèse des compétences E5 couvertes

Compétence E5	Ce que ce projet démontre concrètement
C1 – Gérer le patrimoine informatique	Recensement des serveurs administrés, gestion des habilitations nominatives, politique de sauvegarde des logs (90 jours)
C2 – Répondre aux incidents	Rapport d'incident lié au Kerberos
C4 – Travailler en mode projet	Analyse du besoin, phases préproduction → production, documentation complète, passage de relais à l'administrateur système
C5 – Mettre à disposition un service	Déploiement, tests d'intégration (RDP/SSH/Kerberos), rédaction d'une documentation d'exploitation, rapport d'incident

Table des matières

19. Bilan et Résultats du Projet
20. Axes d'amélioration et Perspectives

1. Architecture et Principes de Sécurité

Le bastion repose sur une **isolation stricte des flux et des privilèges**. Contrairement à une installation standard, ce serveur est conçu pour supporter les contraintes des comptes Active Directory **"Protected Users"** (NTLM interdit, Kerberos obligatoire).

> **Principe de la porte d'entrée unique**
>
> Les administrateurs ne se connectent jamais directement aux serveurs cibles. Ils passent obligatoirement par Guacamole, le seul hôte autorisé à joindre ces serveurs — les flux directs sont bloqués au pare-feu.

Matrice des composants

Composant	Rôle	Sécurité
Tomcat 9	Serveur d'application – interface web	Isolé via utilisateur système dédié, GUACAMOLE_HOME externalisé
guacd	Proxy de connexion – moteur RDP/SSH	Utilisateur dédié sans shell, support GSSAPI/Kerberos
FreeRDP 3.2.0	Bibliothèque protocole RDP	Compilée sur mesure pour supporter Kerberos/NLA – indispensable pour Protected Users
MariaDB	Base de données de gestion	Stockage des droits, connexions et historique d'audit
TOTP	Second facteur d'authentification	Obligatoire pour toute session administrative

Matrice de flux réseau simplifiée

Source	Destination	Port	Protocole	Justification
Poste admin	Bastion Guacamole	443	HTTPS	Accès interface web uniquement
Bastion (guacd)	Serveurs cibles Tier 0/1	3389	RDP/NLA+Kerberos	Flux interne contrôlé
Bastion (guacd)	Serveurs Linux	22	SSH	Administration systèmes Linux

Source	Destination	Port	Protocole	Justification
Bastion	Contrôleurs de domaine	88	Kerberos	Authentification AD

2. Préparation du Système et Hardening

Avant d'installer Guacamole, l'environnement est préparé pour garantir l'isolation des données, l'accès aux paquets et la synchronisation temporelle.

2.1. Activation des dépôts et Réseau

Pour installer les composants de sécurité (Tomcat 9, MariaDB), les dépôts `universe` doivent être actifs.

Configuration Netplan (`/etc/netplan/00-installer-config.yaml`) : La syntaxe doit utiliser des séquences (tirets) et les permissions doivent être strictes.

```
network:
  version: 2
  ethernets:
    ens160:
      addresses:
        - ...
      routes:
        - to: default
          via: ...
      nameservers:
        addresses:
          - ...
        search:
          - ...
```

```
sudo chmod ...
```

2.2. Installation des Socles et Synchronisation

Kerberos est extrêmement sensible au décalage horaire (max 5 min). Les services sont installés maintenant pour créer les utilisateurs système nécessaires.

```
# Installation de Chrony, Tomcat 9 et MariaDB
```

```
sudo apt install -y chrony tomcat9 mariadb-server
```

Configuration de Chrony :

```
# sudo nano /etc/chrony/chrony.conf
server dc... iburst
server dc... iburst

sudo systemctl restart chrony
```

2.3. Arborescence de Sécurité et Permissions

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

# Enregistrements de sessions (Audit)
sudo mkdir -p /var/...

# Attribution des droits
sudo chown ...
```

2.4. Hardening de la résolution locale (`/etc/hosts`)

Indispensable pour le **Tier 0** afin de réduire la dépendance au DNS externe.

```
# sudo nano /etc/hosts
ip...    dc...
ip...    dc...
```

2.5. Validation de la Résolution DNS (FQDN)

Le bastion doit résoudre parfaitement les noms complets pour Kerberos.

```
sudo apt install -y dnsutils iputils-ping
```

1. **Test Forward :** `host dc...`
2. **Test Inverse (Critique) :** `host <IP_DU_DC>`

> Si l'IP ne retourne pas le FQDN (PTR record), Kerberos échouera avec l'erreur `KRB_AP_ERR_MODIFIED`.

3. Installation du Moteur RDP (FreeRDP 3)

> **Étape critique pour le Tier 0**

> La version native d'Ubuntu 22.04 (FreeRDP 2.x) est **insuffisante** pour les comptes "Protected Users" : elle ne supporte pas Kerberos/GSSAPI. La compilation depuis les sources de FreeRDP 3.2.0 est obligatoire.

3.1. Dépendances de compilation

```
sudo apt install -y cmake git build-essential libssl-dev libx
11-dev libxext-dev \
libxinerama-dev libxcursor-dev libxdamage-dev libxv-dev libxk
bfile-dev \
libasound2-dev libcups2-dev libxml2-dev libxrandr-dev libgstr
eamer1.0-dev \
libgstreamer-plugins-base1.0-dev libkrb5-dev libgssapi-krb5-2
libwayland-dev \
libusb-1.0-0-dev libcjson-dev libpkcs11-helper1-dev libxkbcom
mon-dev \
libfuse3-dev liburiparser-dev xsltproc docbook-xsl \
libavcodec-dev libavutil-dev libswresample-dev libcairo2-dev
libpango1.0-dev \
libssh2-1-dev libssl-dev libjpeg-turbo8-dev libpng-dev libavf
ormat-dev \
libswscale-dev
```

3.2. Compilation de FreeRDP 3.2.0

```
cd /tmp
git clone https://github.com/FreeRDP/FreeRDP.git
cd FreeRDP && git checkout 3.2.0
mkdir build && cd build

# Vider le dossier build si besoin
rm -rf /tmp/FreeRDP/build/*

cd /tmp/FreeRDP/build

cmake -DCMAKE_BUILD_TYPE=Release \
```

```
  -DWITH_KDCRAW=ON \
  -DWITH_GSSAPI=ON \
  -DWITH_SSE2=ON \
  -DWITH_MANPAGES=OFF \
  -DCHANNEL_URBDRC=OFF \
  -DWITH_SERVER=OFF ..

make -j$(nproc)
sudo make install
sudo ldconfig
```

4. Installation de Guacamole Server (guacd)

Le service Guacamole est lié à la version spécifique de FreeRDP compilée à l'étape précédente.

4.1. Compilation du serveur

```
wget https://downloads.apache.org/guacamole/1.6.0/source/guac
amole-server-1.6.0.tar.gz
tar -xzf guacamole-server-1.6.0.tar.gz
cd guacamole-server-1.6.0/

make clean 2>/dev/null

# Forcer l'utilisation du nouveau FreeRDP via PKG_CONFIG_PATH
export PKG_CONFIG_PATH=/usr/local/lib/pkgconfig
./configure --with-systemd-dir=/lib/systemd/system \
            --with-rdp \
            --with-ssh \
            --without-vnc \
            --without-telnet \
            --without-kubernetes \
            --disable-oss

# ⚠ VÉRIFIER LE TABLEAU DE SYNTHÈSE AVANT DE CONTINUER

make -j$(nproc)
```

```
sudo make install
sudo ldconfig
```

4.2. Sécurisation de l'utilisateur guacd

Le service ne tourne pas sous `root` ou `daemon`. Un utilisateur dédié sans shell est créé pour limiter l'impact d'une compromission potentielle.

```
sudo useradd -M -d /var/lib/guacd/ -r -s /sbin/nologin -c
"..." guacd
sudo mkdir -p /var/lib/...
sudo chown ...

# Modification du service systemd
sudo sed -i 's/User=.../User=.../' /lib/systemd/system/guacd.
service
sudo systemctl daemon-reload
sudo systemctl enable --now guacd
```

> **Note :** À ce stade, le moteur est capable de négocier Kerberos avec les serveurs Windows, même pour les comptes ultra-restreints du groupe Protected Users.

5. Configuration de l'Authentification Kerberos

Pour que Guacamole puisse négocier avec l'Active Directory, le système Linux doit d'abord être un client Kerberos irréprochable.

5.1. Configuration du Client (`/etc/krb5.conf`)

> Les noms de domaines doivent impérativement être en **MAJUSCULES** dans ce fichier.

```
sudo nano /etc/krb5.conf

[libdefaults]
    default_realm = EXEMPLE.LOCAL
    dns_lookup_realm = false
    dns_lookup_kdc = true
    rdns = false
```

```
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true

[realms]
    EXEMPLE.LOCAL = {
        kdc = dc...
        kdc = dc...
        admin_server = dc...
    }

[domain_realm]
    .exemple.local = EXEMPLE.LOCAL
    exemple.local  = EXEMPLE.LOCAL
```

5.2. Création du Keytab

Plutôt que de stocker un mot de passe en clair, on génère un fichier **Keytab**. Il permet à `guacd` d'obtenir ses tickets Kerberos (TGT) de manière autonome et sécurisée.

```
sudo apt update && sudo apt install -y krb5-user
```

```
sudo ktutil
addent -password -p compte.adm@TEST.LOCAL -k 1 -e aes256-cts-
hmac-sha1-96
# [Saisir le mot de passe de l'administrateur quand demandé]
wkt /etc/guacamole/compte.adm.keytab
q
```

Sécurisation stricte (lecture seule, propriétaire uniquement) :

```
sudo chown ... /etc/guacamole/compte.adm.keytab
sudo chmod ...

# Vérification du contenu
sudo klist -kt /etc/guacamole/compte.adm.keytab
```

6. Automatisation et Persistance des Tickets

Les comptes "Protected Users" ont des tickets Kerberos de durée limitée. Un mécanisme de renouvellement automatique garantit la continuité du service.

6.1. Script de renouvellement

```
sudo nano /usr/local/bin/renew-guac-kerberos.sh

#!/bin/bash
# Renouvellement automatique du ticket Kerberos pour guacd
/usr/bin/kinit -kt /etc/guacamole/compte.adm.keytab compte.adm@EXEMPLE.LOCAL
logger "Guacamole-Kerberos: Ticket renouvelé avec succès pour compte.adm."
```

```
sudo chmod ...
```

6.2. Planification (Cron)

Renouvellement toutes les 8 heures — bien avant l'expiration standard de 10h.

```
(sudo crontab -l 2>/dev/null; echo "0 */8 * * * /usr/local/bin/renew-guac-kerberos.sh") | sudo crontab -
```

7. Liaison Kerberos avec guacd

```
sudo systemctl edit guacd
```

```
[Service]
User=...
Group=...

# Variables d'environnement Kerberos
Environment="KRB5_CONFIG=..."
Environment="KRB5_CLIENT_KTNAME=..."
Environment="KRB5CCNAME=FILE:..."
```

```
RuntimeDirectory=...
RuntimeDirectoryMode=...
```

```
# Initialisation manuelle pour le premier lancement
sudo -u ...

# Supprimer l'ancien cache pour éviter les conflits de permis
sions
sudo rm -f ...

sudo systemctl daemon-reload
sudo systemctl restart guacd
```

8. Déploiement du Client Web (Tomcat 9)

Tomcat 9 héberge l'interface web Guacamole. C'est ici que le fichier `.war` est installé et que la configuration centrale est déclarée.

8.1. Installation du binaire `.war`

```
wget https://downloads.apache.org/guacamole/1.6.0/binary/guac
amole-1.6.0.war
sudo cp guacamole-1.6.0.war /etc/guacamole/guacamole.war
sudo ln -sf /etc/guacamole/guacamole.war /var/lib/tomcat9/web
apps/guacamole.war
```

8.2. Déclaration du GUACAMOLE_HOME

Tomcat doit savoir où chercher ses fichiers de configuration.

```
sudo systemctl edit tomcat9
```

```
[Service]
Environment="GUACAMOLE_HOME=/etc/guacamole"
```

```
sudo systemctl daemon-reload
sudo systemctl restart tomcat9
```

9. Configuration de la Persistance (MariaDB)

MariaDB stocke les connexions, les utilisateurs et l'historique d'audit. Le mot de passe de la base n'est jamais exposé inutilement.

9.1. Initialisation de la Base de Données

```
sudo mariadb
```

```
CREATE DATABASE guacamole_db;

CREATE USER 'guacamole_user'@'localhost' IDENTIFIED BY 'UnMD
P';

GRANT SELECT,INSERT,UPDATE,DELETE, CREATE TEMPORARY TABLES, L
OCK TABLES
  ON guacamole_db.* TO 'guacamole_user'@'localhost';

FLUSH PRIVILEGES;
EXIT;
```

9.2. Installation de l'extension JDBC et du Driver

1. Extension Guacamole JDBC :

```
wget https://downloads.apache.org/guacamole/1.6.0/binary/guac
amole-auth-jdbc-1.6.0.tar.gz
tar -xzf guacamole-auth-jdbc-1.6.0.tar.gz
sudo cp guacamole-auth-jdbc-1.6.0/mysql/guacamole-auth-jdbc-m
ysql-1.6.0.jar /etc/guacamole/extensions/
```

2. Driver MySQL (Connector/J) :

```
wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-co
nnector-j-8.2.0.tar.gz
tar -xzf mysql-connector-j-8.2.0.tar.gz
sudo cp mysql-connector-j-8.2.0/mysql-connector-j-8.2.0.jar /
etc/guacamole/lib/
```

3. Injection du schéma SQL :

```
cat guacamole-auth-jdbc-1.6.0/mysql/schema/*.sql | sudo maria
db guacamole_db
```

10. Configuration de Guacamole (guacamole.properties)

Ce fichier est le cerveau du client web. On y définit les accès à la base de données, les chemins d'audit et les paramètres Kerberos.

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

# Connexion au proxy guacd
guacd-hostname: localhost
guacd-port:     4822

# Authentification MariaDB
mysql-hostname: localhost
mysql-port:     3306
mysql-database: guacamole_db
mysql-username: guacamole_user
mysql-password: UnMDP

# Chemin d'audit vidéo
recording-search-path: /var/...

# Royaume Kerberos par défaut
krb5-realm: TEST.LOCAL
```

```
sudo chmod ...
sudo chown ...
```

11. Sécurisation MFA — Extension TOTP

> **Avertissement de sécurité**

> Dans un environnement de rebond, le couple identifiant/mot de passe est insuffisant. L'activation du TOTP (Google Authenticator, Microsoft Authenticator) est **obligatoire**.

11.1. Installation

```
wget https://downloads.apache.org/guacamole/1.6.0/binary/guac
amole-auth-totp-1.6.0.tar.gz
tar -xzf guacamole-auth-totp-1.6.0.tar.gz
sudo cp guacamole-auth-totp-1.6.0/guacamole-auth-totp-1.6.0.j
ar /etc/guacamole/extensions/
```

11.2. Paramétrage

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

# --- Configuration TOTP ---
totp-issuer: Mon-Bastion-Securise
totp-digits: 6
totp-period: 30
```

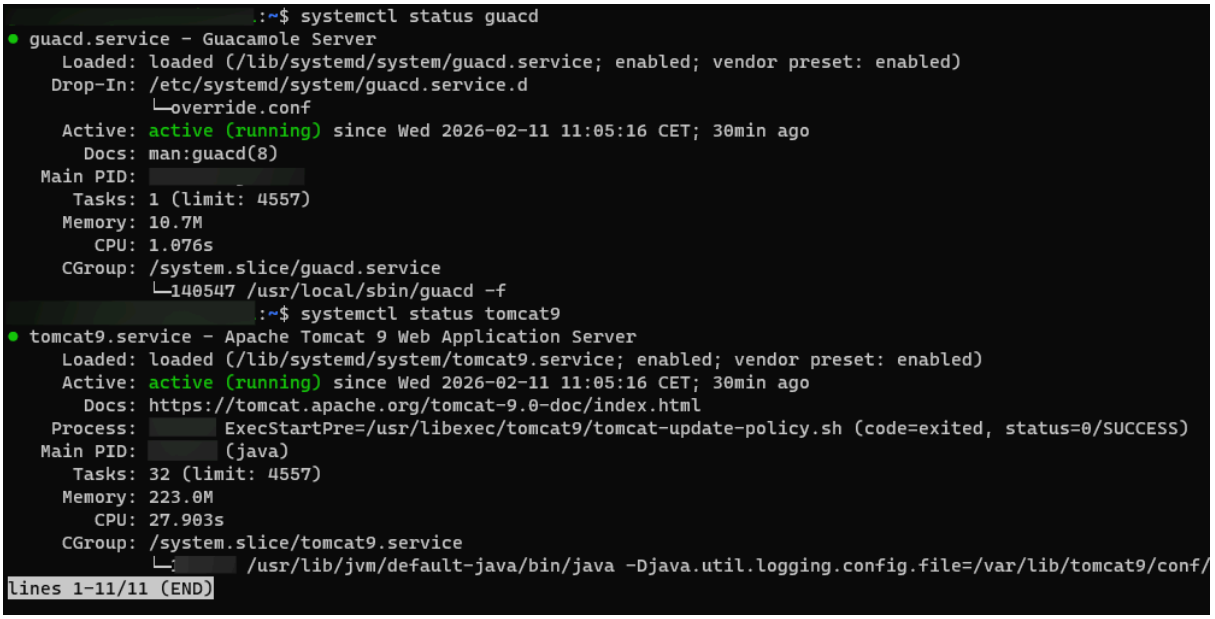
12. Finalisation et Lancement

```
sudo systemctl restart tomcat9 guacd
```

Checklist de bon fonctionnement

Élément	Commande de vérification	Résultat attendu	
Interface Web	`http://<ip>:8080/guacamole`	Page de login active avec champ	

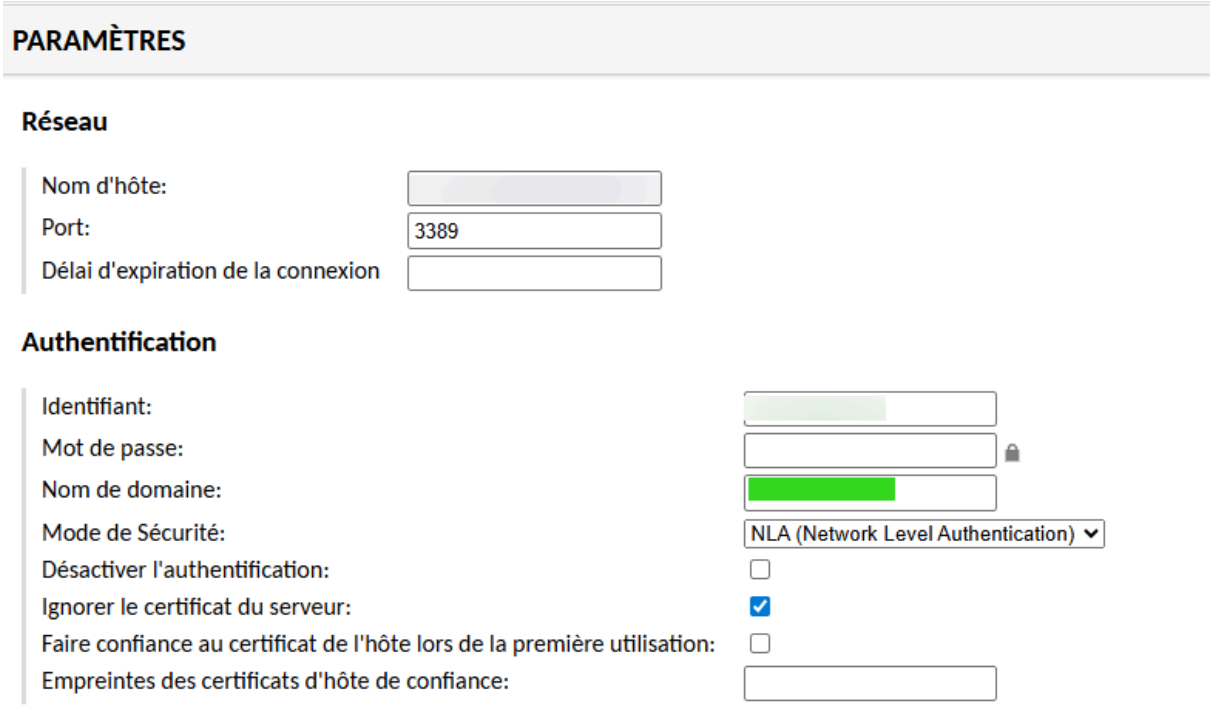
Élément	Commande de vérification	Résultat attendu	
		TOTP	
Extensions chargées	`sudo journalctl -u tomcat9 \	grep "Extension"`	Chaque extension listée au démarrage
Base de données	`sudo journalctl -u tomcat9 \	grep "MySQL"`	Connexion MariaDB établie
Kerberos guacd	`sudo -u guacd klist`	Ticket TGT valide	



13. Configuration des Connexions

Paramètres **obligatoires** dans l'interface Guacamole pour chaque serveur cible afin que le compte Protected Users soit accepté :

Paramètre	Valeur / Réglage
Hostname	**FQDN Obligatoire** (ex: `srv-test.test.local`)
Port	3389
Security Mode	**NLA** (Network Level Authentication)
Domain	`TEST.LOCAL`
Username	compte.adm
Password	*Laisser vide — le ticket Kerberos est utilisé*
Ignore Server Certificate	Coché (ou déployer le certificat CA)
Nom d'hôte (proxy Guacamole)	localhost
Port (proxy Guacamole)	4822



14. Système d'Audit et Visualisation des Sessions

Par défaut, Guacamole peut enregistrer les sessions. Pour les lire directement depuis l'interface Web, une extension dédiée est nécessaire.

14.1. Installation de l'extension de lecture

```
wget https://downloads.apache.org/guacamole/1.6.0/binary/guac
amole-history-recording-storage-1.6.0.tar.gz
tar -xzf guacamole-history-recording-storage-1.6.0.tar.gz
sudo cp guacamole-history-recording-storage-1.6.0/guacamole-h
istory-recording-storage-1.6.0.jar /etc/guacamole/extensions/
```

14.2. Droits d'accès du dossier d'Audit

Le moteur `guacd` écrit les fichiers, Tomcat doit pouvoir les lire.

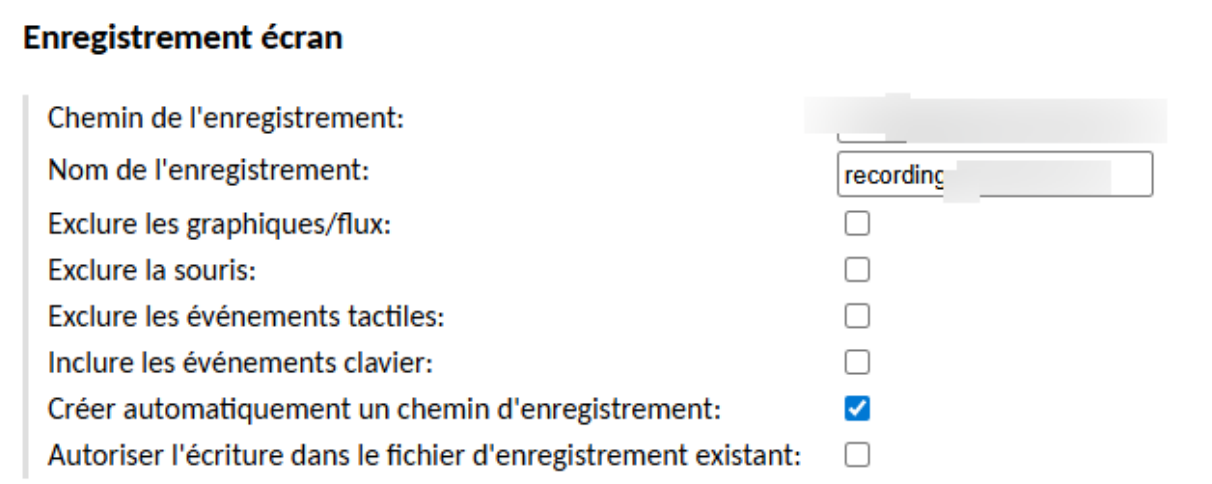
```
# Propriétaire : guacd (écriture), Groupe : tomcat (lecture)
sudo chown ...

# Droits 2750 : le bit SGID force les nouveaux fichiers à app
artenir au groupe tomcat
sudo chmod ...
```

```
sudo systemctl restart guacd tomcat9
```

14.3. Configuration des Connexions pour l'indexation

Dans l'interface Web, section **"Session Recording"** de chaque connexion :

Paramètre	Valeur
Recording path	`/var/.../${HISTORY_UUID}`
Recording name	`recording`
Automatically create path	**Coché**



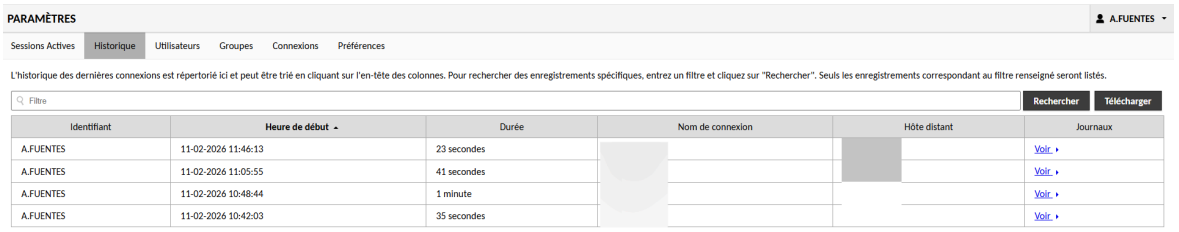
Pourquoi des UUID pour les enregistrements ?

Chaque session génère un dossier nommé par son `HISTORY_UUID` — un identifiant unique non prédictible. Cette approche garantit :

- **1 session = 1 dossier = 1 enregistrement** → aucun risque d'écrasement
- **Impossible à deviner** → un attaquant ne peut pas cibler un enregistrement précis par son chemin
- **Aucune information sensible** visible dans le nom du fichier
- **Conformité aux bonnes pratiques d'audit** et de traçabilité

Si Guacamole utilisait le nom du serveur, le nom de l'utilisateur ou une date lisible, un attaquant pourrait corréler les accès sensibles et cibler les enregistrements. L'UUID élimine ce vecteur.

> L'extension de lecture scanne les dossiers nommés par l'UUID de session, ce qui fait apparaître le bouton **"View"** directement dans l'onglet "Historique" — sans téléchargement externe.



15. Politique de Rétention et Maintenance

Pour éviter la saturation disque, le nettoyage des données d'audit est entièrement automatisé.

15.1. Purge automatique des Vidéos (90 Jours)

```
sudo nano /usr/local/bin/guac-purge-recordings.sh

#!/bin/bash
TARGET="/var/..."
if [ -d "$TARGET" ]; then
    find "$TARGET" -mindepth 1 -maxdepth 1 -type d -mtime +90
-exec rm -rf {} \;
    logger "Guacamole-Audit: Purge des enregistrements termin
```

```
ée."
fi
```

```
sudo chmod +x /usr/local/bin/guac-purge-recordings.sh
(sudo crontab -l 2>/dev/null; echo "0 0 * * * /usr/local/bin/
guac-purge-recordings.sh") | sudo crontab -
```

16. Gestion des Logs Système (Journald)

16.1. Configuration de la rétention

```
sudo nano /etc/systemd/journald.conf

[Journal]
# Rétention maximale de 90 jours (cohérent avec la rétention
vidéo)
MaxRetentionSec=90day

# Rotation des fichiers à 100 Mo
SystemMaxFileSize=100M
```

```
sudo systemctl restart systemd-journald
```

17. Rotation des Logs Applicatifs (Logrotate)

```
sudo nano /etc/logrotate.d/tomcat9

/var/log/tomcat9/*.log {
    copytruncate
    daily
    rotate 90
    compress
    missingok
    notifempty
}
```

Synthèse de la Politique de Rétention

Type de donnée	Emplacement	Durée	Méthode
Vidéos d'Audit	`/var/...`	**90 jours**	Script Cron (`find -rm`)
Logs Système	`/var/log/journal/`	**90 jours**	`journald.conf`
Logs Tomcat	`/var/log/tomcat9/`	**90 jours**	`logrotate`

18. Guide de Troubleshooting

Ordre d'analyse en cas de dysfonctionnement : **Réseau → Kerberos → Logs Tomcat → Logs guacd**

Cible	Commande	Utilité
Kerberos	`sudo -u guacd klist`	Vérifier si le ticket TGT est valide pour le moteur
Authentification	`sudo journalctl -u tomcat9 -f`	Voir les erreurs de login ou d'extensions
Flux RDP/SSH	`sudo journalctl -u guacd -f`	Voir les rejets de certificats ou erreurs Kerberos
Espace Disque	`du -sh /var/...`	Surveiller le poids des audits vidéo
DNS inverse	`host <IP_DU_DC>`	Vérifier le PTR record (critique pour Kerberos)
Synchronisation NTP	`chronyc tracking`	Vérifier l'offset temporel (< 5 min obligatoire)

19. Bilan et Résultats du Projet

> **Résultats obtenus**

> À l'issue du déploiement en préproduction, le bastion est **pleinement opérationnel**. L'administrateur système dispose d'une documentation complète permettant le passage en production et l'intégration de l'ensemble des comptes administrateurs et du parc de serveurs.

Ce que ce bastion apporte concrètement au SIS 67

Avant le projet	Après le projet
Connexions RDP directes depuis les postes	Passage obligatoire par le bastion
Aucune traçabilité des actions admin	Enregistrement vidéo de chaque session (90 jours)

Avant le projet	Après le projet
Mot de passe seul suffit à se connecter	MFA TOTP obligatoire (2 facteurs)
NTLM utilisé → incompatible Protected Users	Kerberos/NLA via FreeRDP 3 → compatible Tier 0
Droits admin partagés, non nominatifs	Comptes nominatifs, moindre privilège
Aucun audit possible post-incident	Replay de session directement depuis l'interface

Difficultés rencontrées et solutions

Problème	Cause identifiée	Solution apportée
Connexion RDP échoue pour compte Protected Users	NTLM bloqué par le groupe AD + FreeRDP 2.x sans support Kerberos	Compilation FreeRDP 3.2.0 avec flags `-DWITH_GSSAPI=ON`
Ticket Kerberos expiré après 10h	Durée de vie limitée par la politique AD	Script cron de renouvellement toutes les 8h via keytab
Saturation disque potentielle	Enregistrements vidéo sans limite	Purge automatique cron à 90 jours

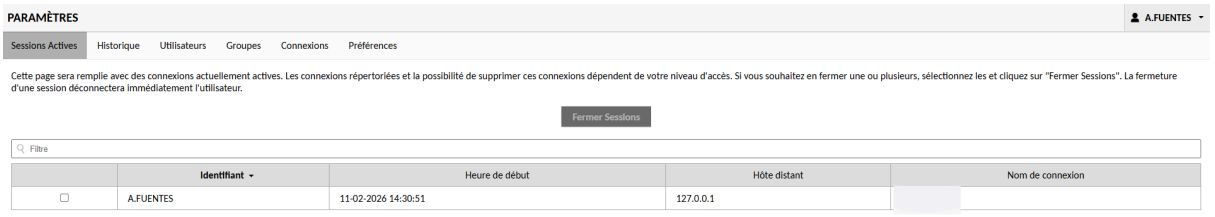
20. Axes d'amélioration et Perspectives

Ce projet constitue la première brique d'une architecture d'administration sécurisée complète. Les évolutions envisagées sont :

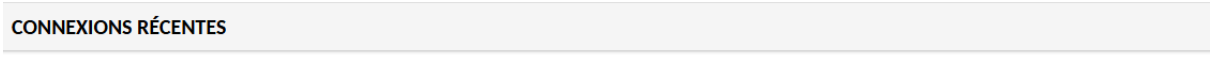
1. **Clonage et Tiering** — Dupliquer la VM Guacamole sur plusieurs segments réseau pour couvrir l'ensemble des Tiers AD (Tier 0 : DC / Tier 1 : Serveurs / Tier 2 : Postes). Chaque administrateur n'accède qu'au Tier correspondant à son niveau d'habilitation.
2. **Déploiement en production** — Finalisation avec l'intégration de l'ensemble du parc de serveurs et des comptes administrateurs.
3. **Supervision Zabbix** — Intégration du bastion comme hôte supervisé dans Zabbix pour monitorer la disponibilité du service et déclencher des alertes en cas de panne.

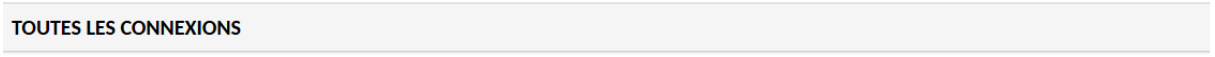
Annexe — Captures d'écran

Interface de connexion Guacamole, écran TOTP, historique des sessions avec bouton View, enregistrements vidéo par UUID.

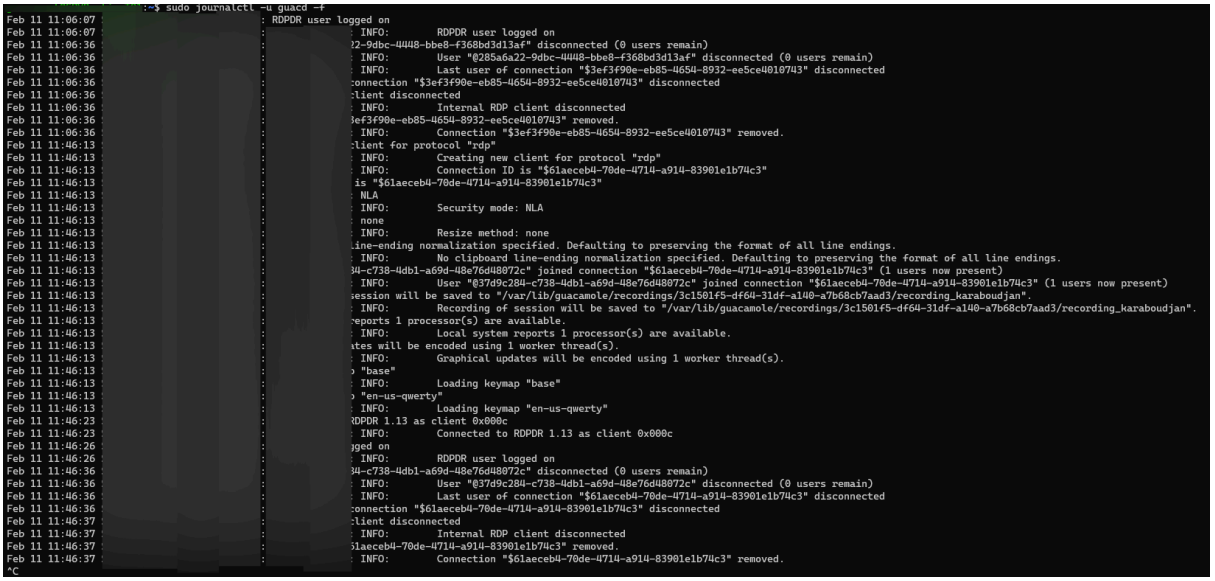

PARAMÈTRES
A.FUENTES
Sessions Actives
Historique
Utilisateurs
Groupes
Connexions
Préférences
Cette page sera remplie avec des connexions actuellement actives. Les connexions répertoriées et la possibilité de supprimer ces connexions dépendent de votre niveau d'accès. Si vous souhaitez en fermer une ou plusieurs, sélectionnez les et cliquez sur "Fermer Sessions". La fermeture d'une session déconnectera immédiatement l'utilisateur.
Fermer Sessions
Filtre
Identifiant
Heure de début
Hôte distant
Nom de connexion
A.FUENTES
11-02-2026 14:30:51
127.0.0.1


A.FUENTES
Presse-papiers
Texte copié/coupé dans Guacamole apparaîtra ici. Changer le texte ci-dessous affectera le presse-papiers distant.
Cliquez pour afficher le contenu du presse-papiers.
Méthode de saisie
Aucune
Aucune méthode de saisie utilisée. Clavier accepté depuis un clavier physique connecté.
Clavier
Affiche et utilise la saisie du clavier virtuel intégré dans Guacamole. Ceci est nécessaire pour les périphériques mobiles ne disposant pas de clavier physique.
Clavier virtuel
Affiche et utilise la saisie du clavier virtuel intégré dans Guacamole. Le clavier virtuel permet d'utiliser des


CONNEXIONS RÉCENTES


TOUTES LES CONNEXIONS


Serveurs



Rédigé par Alessandro FUENTES — Stage SIS 67, 2026