

Bastion d'administration

Mise en place d'Apache Guacamole 1.6.0 sur Debian 12 — Tomcat 9, HTTPS et double authentification

🎯 Objectifs

- Comprendre le rôle d'un bastion d'administration dans un système d'information.
- Savoir où le positionner dans l'architecture réseau (DMZ, pare-feu, VPN).
- Installer Apache Guacamole 1.6.0 (serveur guacd + application web) sur Debian 12.
- Adosser Guacamole à MariaDB pour gérer utilisateurs, droits et traçabilité.
- Durcir le bastion : HTTPS, double authentification, anti-force brute, enregistrement des sessions, pare-feu.
- Utiliser le bastion au quotidien et **gérer les droits des utilisateurs** selon le principe du moindre privilège (partie 2).

1. Qu'est-ce qu'un bastion ?

Un bastion, comme **Apache Guacamole**, est une machine (virtualisée ou non) qui sert de **point d'accès sécurisé** pour se connecter aux autres systèmes d'une infrastructure informatique.

Il joue un rôle crucial en matière de sécurité, car il permet de **concentrer, contrôler et sécuriser** les accès aux systèmes critiques : serveurs, bases de données ou toute autre ressource sensible.

Apport	Ce que cela signifie
Contrôle d'accès centralisé	Le bastion est la porte d'entrée unique vers les systèmes internes.
Amélioration de la sécurité	Les accès peuvent être surveillés, enregistrés et filtrés.
Traçabilité et audit	Les connexions sont journalisées, et les sessions peuvent être enregistrées puis relues.
Simplification des accès distants	Guacamole fonctionne depuis un navigateur, sans client à installer.
Réduction des risques de configuration	L'interface web limite les erreurs de configuration côté poste client.

Le serveur Apache Guacamole sera le **point d'entrée unique** pour accéder aux serveurs et équipements de l'infrastructure, via les protocoles **RDP**, **SSH**, **VNC** et même **Kubernetes**. Que l'on soit en externe ou en interne, les connexions aux serveurs passeront obligatoirement par l'hôte Guacamole.

⚠ Protocoles non sécurisés : à bannir

Guacamole sait aussi gérer **Telnet**, mais ce protocole transmet tout **en clair**, identifiants compris. On fait l'impasse dessus et on le remplace par **SSH**. Dans cette procédure, Telnet n'est même **pas compilé** : moins de code, moins de surface d'attaque.

2. Où positionner le bastion ?

Positionner un bastion dans un système d'information nécessite une réflexion stratégique pour assurer la **sécurité**, la **disponibilité** et la **performance** des accès à distance. Voici les recommandations principales et leurs justifications.

2.1 Segment réseau isolé ou DMZ

Le bastion doit idéalement être placé dans une **zone démilitarisée (DMZ)**, segment tampon entre le réseau interne et l'extérieur, ce qui limite l'exposition directe du réseau interne.

Pourquoi ? La DMZ permet de contrôler les flux entrants et sortants et fournit une couche de protection supplémentaire en cas de compromission du bastion : seuls les services strictement nécessaires sont accessibles.

2.2 Contrôle d'accès via un pare-feu

En plaçant le bastion derrière un **pare-feu** entre la DMZ et le réseau interne, on restreint l'accès aux ressources internes aux seuls ports et protocoles nécessaires.

Pourquoi ? Tous les accès passent par une couche de filtrage et de surveillance, ce qui limite les risques d'intrusion.

2.3 Connexion sécurisée par VPN ou tunnel SSH

Depuis l'extérieur, l'accès au bastion ne doit être possible qu'au travers d'un **VPN** ou d'un **tunnel SSH**.

Pourquoi ? Ces connexions ajoutent une couche de chiffrement et d'authentification, et évitent d'exposer directement le portail web sur Internet.

2.4 Accès restreint aux réseaux internes

Le bastion ne doit atteindre que les **sous-réseaux ou machines nécessaires**, grâce à des règles de pare-feu internes.

Pourquoi ? En limitant ses permissions, on réduit la surface d'attaque et on contrôle la propagation en cas de compromission.

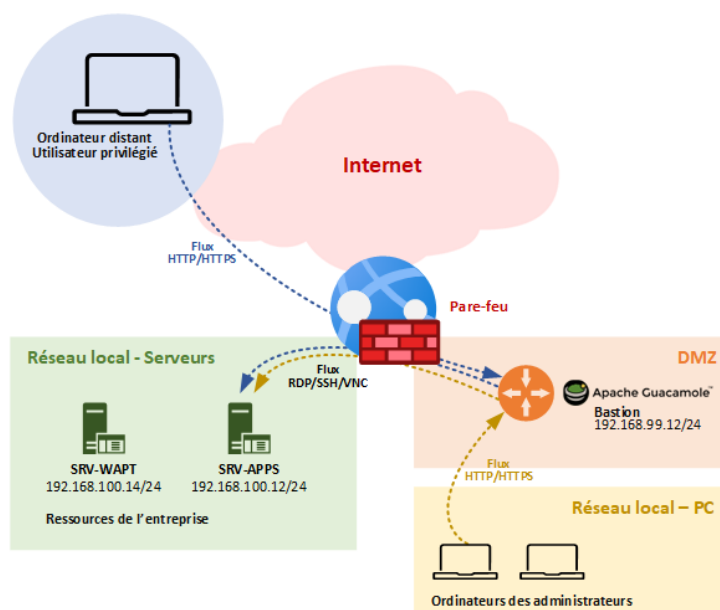
2.5 Authentification multifacteur (MFA) et journalisation

La **MFA** et des **journaux d'accès détaillés** sont essentiels sur le bastion (mises en œuvre en section 9).

Pourquoi ? La MFA garantit que seules les personnes autorisées accèdent au bastion ; les journaux permettent de tracer toute activité suspecte ou anormale.

✓ À retenir

Le **meilleur emplacement** pour Apache Guacamole est la **DMZ, derrière un pare-feu**, avec des restrictions strictes d'accès au réseau interne. Les utilisateurs accèdent aux ressources nécessaires tout en minimisant l'exposition des ressources internes.



Exemple de positionnement du bastion dans une DMZ

§ Lecture du schéma

Les flux « HTTP/HTTPS » vers le bastion s'entendent **en HTTPS uniquement** : le port 80 ne sert qu'à rediriger vers le 443. L'ordinateur distant passe par le **VPN** de l'entreprise avant d'atteindre le bastion.

3. Architecture cible et prérequis

Tous les composants sont installés sur la même machine Debian 12. Seul **Nginx** est joignable depuis le réseau ; les autres services n'écoutent que **localement**.

Composant	Rôle	Écoute
Nginx	Reverse proxy HTTPS : seule porte d'entrée web du bastion.	0.0.0.0:443 (et 80 → redirection)
Tomcat 9	Exécute l'application web Guacamole (fichier .war).	127.0.0.1:8080
guacd	Démon qui ouvre les connexions RDP / SSH / VNC vers les serveurs.	127.0.0.1:4822
MariaDB	Stocke utilisateurs, droits, connexions et historique.	127.0.0.1:3306

Élément	Version utilisée	Où vérifier
Debian	12 « bookworm » (LTS jusqu'en juin 2028)	debian.org/releases
Apache Guacamole	1.6.0 (22 juin 2025)	guacamole.apache.org/releases
Tomcat	9.0.122 (archive officielle Apache)	tomcat.apache.org/download-90.cgi
Java	OpenJDK 17 (dépôts Debian 12)	—
MySQL Connector/J	9.7.0 (avril 2026)	dev.mysql.com/downloads/connector/j

§ Pourquoi Debian 12 et pas Debian 13 ?

Debian 13 fournit **FreeRDP 3**, que Guacamole 1.6.0 ne prend en charge qu'à titre **expérimental** : la compilation du support RDP y pose problème. Debian 12 fournit FreeRDP 2 et reste maintenue (LTS) jusqu'en juin 2028. À réévaluer à la prochaine version de Guacamole.

⚠ Vous avez suivi l'ancienne version de ce document ?

Elle installait Tomcat 9 depuis le dépôt **Debian 11 « bullseye »**, en **fin de vie depuis le 31 août 2026** (plus aucun correctif de sécurité). Tomcat 9 reste le bon choix, mais pas ce paquet. Avant de reprendre la procédure, supprimez ce dépôt, ces paquets et les anciens fichiers :

```
rm /etc/apt/sources.list.d/bullseye.list
apt-get purge tomcat9 tomcat9-admin tomcat9-common tomcat9-user
apt-get autoremove && apt-get update
rm /etc/guacamole/extensions/*1.5.5*.jar
rm /etc/guacamole/lib/mysql-connector-j-9.1.0.jar
```

Si la base de données existe déjà, ne réimportez pas le schéma : appliquez les scripts du dossier [mysql/schema/upgrade/](#) correspondant à votre version de départ (voir les notes de version 1.6.0).

Toutes les commandes s'exécutent avec des **droits d'administration** (root ou **sudo**). C'est une nécessité pour installer des paquets et des services, pas une bonne pratique en soi : une fois le bastion en place, on n'utilise plus root au quotidien.

4. Préparer le système

On met le système à jour, puis on installe les dépendances de compilation. La liste complète est dans la documentation officielle :

- [Documentation — Installation d'Apache Guacamole \(dépendances\)](#)

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

§ Nouveautés par rapport à l'ancienne procédure

- **apt-get update** ne fait que **rafraîchir la liste** des paquets ; c'est **apt-get upgrade** qui **met à jour** le système. On le fait **avant** d'installer quoi que ce soit.
- **libtelnet-dev** a été **retiré** : sans lui, le protocole Telnet n'est pas compilé.

5. Compiler et installer guacd (Guacamole Server)

A. Télécharger et vérifier les sources

On fixe la version dans une variable, réutilisée dans toute la suite. **Si vous ouvrez un nouveau terminal, redéfinissez-la.**

```
VER=1.6.0
cd /tmp
wget https://downloads.apache.org/guacamole/${VER}/source/guacamole-server-${VER}.tar.gz
wget https://downloads.apache.org/guacamole/${VER}/source/guacamole-server-${VER}.tar.gz.sha256
sha256sum guacamole-server-${VER}.tar.gz
cat guacamole-server-${VER}.tar.gz.sha256
```

⚠ Vérifier l'intégrité

Les deux empreintes affichées doivent être **identiques**. Sinon, l'archive a été corrompue ou modifiée : on ne l'installe pas.

Le site downloads.apache.org ne conserve que les versions courantes ; les anciennes sont sur archive.apache.org/dist/guacamole/.

B. Configurer et compiler

```
tar -xzf guacamole-server-${VER}.tar.gz
cd guacamole-server-${VER}/
./configure --with-systemd-dir=/etc/systemd/system/
```

En fin de sortie, on contrôle le support des protocoles. Résultat attendu :

```
Protocol support:
  Kubernetes .... yes
  RDP ..... yes
  SSH ..... yes
  Telnet ..... no      <- voulu
  VNC ..... yes
Services / tools:
  guacd ..... yes
```

⚠ Erreur « guacenc_video_alloc »

Elle concerne **guacenc**, l'outil qui convertit les enregistrements de session en vidéo (via FFmpeg). On peut relancer la configuration en le désactivant :

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

Les sessions restent enregistrables et consultables dans l'interface web (section 9B) ; seule l'export vidéo est perdu.

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

C. Configurer guacd en écoute locale

```
mkdir -p /etc/guacamole/{extensions,lib}
nano /etc/guacamole/guacd.conf
```

```
[server]
bind_host = 127.0.0.1
bind_port = 4822
```

⚠ Jamais 0.0.0.0 ici

Le protocole de guacd **n'est pas authentifié** : quiconque joint le port 4822 peut lui demander d'ouvrir des connexions. Tomcat étant sur la même machine, guacd n'a besoin d'écouter que sur **127.0.0.1**.

D. Donner un répertoire de travail à guacd

guacd tourne sous le compte système **daemon**, dont le répertoire personnel n'est pas modifiable. Or, pour les connexions **RDP**, la bibliothèque FreeRDP a besoin d'y écrire sa configuration : sans répertoire accessible, les connexions RDP échouent (en particulier avec NLA). On lui attribue un répertoire dédié par une **surcharge systemd**, sans modifier le fichier de service installé.

```
mkdir -p /etc/systemd/system/guacd.service.d
nano /etc/systemd/system/guacd.service.d/override.conf
```

```
[Service]
StateDirectory=guacd
Environment=HOME=/var/lib/guacd
```

§ À savoir

- **StateDirectory=guacd** demande à systemd de créer **/var/lib/guacd**, propriété du compte **daemon**, avec des droits limités à ce compte.
- **systemctl edit guacd** produit normalement le même fichier, mais il peut échouer sur une unité installée par **make install** : la création manuelle ci-dessus fonctionne dans tous les cas.
- Symptôme si l'étape est oubliée : **journalctl -u guacd** signale un **Permission denied** sur un chemin en **.config/freerdp**, et la connexion RDP se ferme aussitôt.

E. Démarrer guacd

```
systemctl daemon-reload
systemctl enable --now guacd
systemctl status guacd
systemctl show guacd -p Environment
```

Le statut doit être **active (running)**, avec la ligne **Listening on host 127.0.0.1, port 4822**. La dernière commande doit afficher **Environment=HOME=/var/lib/guacd**. Le service tourne sous le compte système **daemon**, pas en root.

6. Installer Tomcat 9 et l'application web

Guacamole 1.6.0 s'exécute sur **Tomcat 9**. Tomcat 10, fourni par Debian 12, utilise une autre API Java (**jakarta.servlet** au lieu de **javax.servlet**) : le projet Guacamole ne le prendra en charge qu'à partir de sa version 2.0.0. Debian 12 ne proposant plus Tomcat 9, on l'installe depuis l'**archive officielle d'Apache**, sans dépôt tiers ni conversion.

§ Tomcat 9 est-il encore maintenu ?

Oui. Tomcat 9 est la dernière version compatible Java EE (javax) : la fondation Apache en assure le support **au moins jusqu'au 31 mars 2027**, puis prévoit une branche **9.1.x** pour prolonger ce support, avec un préavis d'au moins 12 mois avant son arrêt.

Contrepartie : Tomcat n'étant plus installé par apt, **ses mises à jour de sécurité sont manuelles** (voir l'encadré « Maintenance » ci-dessous).

A. Installer Java et créer le compte de service

Tomcat tourne sous un compte système dédié, sans shell ni droit de connexion : **jamais en root**.

```
apt-get install -y openjdk-17-jre-headless
useradd --system --user-group --home-dir /opt/tomcat --shell /usr/sbin/nologin tomcat
```

B. Télécharger et vérifier Tomcat

Vérifiez la dernière version 9.0.x sur la page de téléchargement officielle :

- [Apache Tomcat 9 — Téléchargement](#)

```
TOMCAT_VER=9.0.122
cd /tmp
wget https://downloads.apache.org/tomcat/tomcat-9/v${TOMCAT_VER}/bin/apache-tomcat-${TOMCAT_VER}.tar.gz
wget https://downloads.apache.org/tomcat/tomcat-9/v${TOMCAT_VER}/bin/apache-tomcat-${TOMCAT_VER}.tar.gz.sha512
sha512sum -c apache-tomcat-${TOMCAT_VER}.tar.gz.sha512
```

La dernière commande doit afficher « **Réussi** » (ou « OK »). Sinon, on n'installe pas l'archive.

C. Installer Tomcat

On installe Tomcat dans `/opt/tomcat`, avec un lien `latest` vers la version en service : une mise à jour consistera à faire pointer ce lien vers la nouvelle version.

```
mkdir -p /opt/tomcat
tar -xzf apache-tomcat-${TOMCAT_VER}.tar.gz -C /opt/tomcat
ln -sf /opt/tomcat/apache-tomcat-${TOMCAT_VER} /opt/tomcat/latest
# Applications livrées par défaut : inutiles sur un bastion
rm -rf /opt/tomcat/latest/webapps/*
# Binaires et configuration : propriété de root, lisibles par le groupe tomcat
chown -R root:tomcat /opt/tomcat/apache-tomcat-${TOMCAT_VER}
chmod -R g+rX,o-rwx /opt/tomcat/apache-tomcat-${TOMCAT_VER}
# Seuls ces répertoires sont modifiables par Tomcat
mkdir -p /opt/tomcat/latest/conf/Catalina/localhost
chown -R tomcat:tomcat /opt/tomcat/latest/{logs,temp,work,webapps,conf/Catalina}
```



Moindre privilège

Les applications d'exemple et surtout l'interface d'administration **manager** sont supprimées : elles n'ont rien à faire sur un bastion. Le compte tomcat ne peut **pas modifier** ses propres binaires ni sa configuration : un attaquant qui prendrait le contrôle de Tomcat ne pourrait pas les altérer.

D. Créer le service systemd

```
nano /etc/systemd/system/tomcat.service
```

```
[Unit]
Description=Apache Tomcat 9 (Guacamole)
After=network.target

[Service]
Type=simple
User=tomcat
Group=tomcat
Environment="JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64"
Environment="CATALINA_HOME=/opt/tomcat/latest"
Environment="CATALINA_BASE=/opt/tomcat/latest"
Environment="GUACAMOLE_HOME=/etc/guacamole"
ExecStart=/opt/tomcat/latest/bin/catalina.sh run
UMask=0027
NoNewPrivileges=true
PrivateTmp=true
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

```
systemctl daemon-reload
systemctl enable --now tomcat
systemctl status tomcat
```

`GUACAMOLE_HOME` indique explicitement à Guacamole où trouver sa configuration (`/etc/guacamole`). `NoNewPrivileges` et `PrivateTmp` durcissent le service.

E. Déployer l'application web Guacamole

```
cd /tmp
wget https://downloads.apache.org/guacamole/${VER}/binary/guacamole-${VER}.war
install -o tomcat -g tomcat -m 640 guacamole-${VER}.war
/opt/tomcat/latest/webapps/guacamole.war
systemctl restart tomcat
```

§ Maintenance de Tomcat

Tomcat ne se met plus à jour avec apt : il faut **surveiller les annonces de sécurité** de Tomcat 9 (liste de diffusion announce@tomcat.apache.org ou page « Security » du site). Pour passer à une nouvelle version 9.0.x :

- télécharger et vérifier la nouvelle archive, l'extraire dans `/opt/tomcat` ;
- recopier `conf/server.xml` et le contenu de `webapps/` depuis l'ancienne version, puis appliquer les mêmes droits ;
- faire pointer le lien `latest` vers la nouvelle version et redémarrer le service ;
- garder l'ancienne version quelques jours pour pouvoir revenir en arrière.

7. Base de données MariaDB

Une base de données permet de gérer **utilisateurs et permissions**, de conserver l'**historique des connexions** (audit, conformité) et de s'intégrer à un **annuaire LDAP / Active Directory**. Guacamole prend en charge MySQL / MariaDB et PostgreSQL ; on utilise ici **MariaDB**.

A. Installer et sécuriser MariaDB

```
apt-get install -y mariadb-server  
mariadb-secure-installation
```

Le script supprime les comptes anonymes et la base de test, et interdit la connexion root à distance. Sur Debian, root se connecte ensuite simplement avec la commande **mariadb** (authentification par socket local).

B. Créer la base et le compte applicatif

On génère d'abord un mot de passe fort :

```
openssl rand -base64 24
```

```
mariadb  
CREATE DATABASE bastion_db CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;  
CREATE USER 'bastion_app'@'localhost' IDENTIFIED BY '<MotDePasseFort>';  
GRANT SELECT,INSERT,UPDATE,DELETE ON bastion_db.* TO 'bastion_app'@'localhost';  
FLUSH PRIVILEGES;  
EXIT;
```

Ne recopiez pas les noms d'exemple

Les noms de base et de compte d'un tutoriel sont **publics**. Choisissez vos propres noms, et le mot de passe généré ci-dessus. Le compte applicatif ne reçoit que les droits **SELECT, INSERT, UPDATE, DELETE** sur sa seule base : principe du **moindre privilège**.

C. Extension JDBC et import du schéma

```
cd /tmp  
wget https://downloads.apache.org/guacamole/${VER}/binary/guacamole-auth-jdbc-${VER}.tar.gz  
tar -xzf guacamole-auth-jdbc-${VER}.tar.gz  
cat guacamole-auth-jdbc-${VER}/mysql/schema/*.sql | mariadb bastion_db  
cp guacamole-auth-jdbc-${VER}/mysql/guacamole-auth-jdbc-mysql-${VER}.jar  
/etc/guacamole/extensions/
```

D. Connecteur MySQL (Connector/J)

Le connecteur se télécharge sur le site de MySQL, que l'on utilise MariaDB ou MySQL. Vérifiez la dernière version (**Platform Independent**, archive **.tar.gz**) :

- [Télécharger MySQL Connector/J](#)

```
JDBC_VER=9.7.0
```

```
cd /tmp
wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-j-${JDBC_VER}.tar.gz
tar -xzf mysql-connector-j-${JDBC_VER}.tar.gz
cp mysql-connector-j-${JDBC_VER}/mysql-connector-j-${JDBC_VER}.jar
/etc/guacamole/lib/
```

E. Fichier guacamole.properties

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

```
# guacd
guacd-hostname: 127.0.0.1
guacd-port: 4822
# MariaDB
mysql-hostname: 127.0.0.1
mysql-port: 3306
mysql-database: bastion_db
mysql-username: bastion_app
mysql-password: <MotDePasseFort>
```

Ce fichier contient un mot de passe **en clair** : seuls root et Tomcat doivent pouvoir le lire.

```
chown root:tomcat /etc/guacamole/guacamole.properties
chmod 640 /etc/guacamole/guacamole.properties
systemctl restart tomcat
```

8. Publier le bastion en HTTPS

Par défaut, Tomcat sert Guacamole **en clair** sur le port 8080 : identifiants et sessions circuleraient sans chiffrement. On place **Nginx** devant, en HTTPS, et on ferme Tomcat au réseau.

A. Tomcat en écoute locale et vraie IP des clients

```
nano /opt/tomcat/latest/conf/server.xml
```

Dans le connecteur HTTP (port 8080), on ajoute l'attribut **address** ; les autres attributs présents restent tels quels :

```
<Connector port="8080" protocol="HTTP/1.1"
  address="127.0.0.1"
  connectionTimeout="20000"
  redirectPort="8443" />
```

Puis, à l'intérieur de la balise **<Host>**, on ajoute la valve suivante. Sans elle, Tomcat verrait toutes les connexions venir de 127.0.0.1 (Nginx) : l'**historique** serait inexploitable et l'**anti-force brute** bannirait le proxy lui-même.

```
<Valve className="org.apache.catalina.valves.RemoteIpValve"
  internalProxies="127\.0\.0\.1|0:0:0:0:0:0:0:1"
  remoteIpHeader="x-forwarded-for"
  remoteIpProxiesHeader="x-forwarded-by"
  protocolHeader="x-forwarded-proto" />
```

B. Certificat TLS

Pour la maquette, on génère un **certificat autosigné** (le navigateur affichera un avertissement). En production, on utilise un certificat émis par l'**autorité de certification interne** de l'entreprise.

```
apt-get install -y nginx
mkdir -p /etc/nginx/ssl
openssl req -x509 -nodes -newkey rsa:3072 -days 825 \
  -keyout /etc/nginx/ssl/guacamole.key -out /etc/nginx/ssl/guacamole.crt \
  -subj "/CN=bastion.mondomaine.local" \
  -addext "subjectAltName=DNS:bastion.mondomaine.local,IP:192.168.99.12"
chmod 600 /etc/nginx/ssl/guacamole.key
```

C. Site Nginx

```
nano /etc/nginx/sites-available/guacamole
```

```
server {
    listen 80;
    server_name bastion.mondomaine.local;
    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl http2;
    server_name bastion.mondomaine.local;
    ssl_certificate /etc/nginx/ssl/guacamole.crt;
    ssl_certificate_key /etc/nginx/ssl/guacamole.key;
    ssl_protocols TLSv1.2 TLSv1.3;

    location = / { return 301 /guacamole/; }

    location /guacamole/ {
        proxy_pass http://127.0.0.1:8080;
        proxy_buffering off;
        proxy_http_version 1.1;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $http_connection;
        access_log off;
    }
}
```

```
ln -s /etc/nginx/sites-available/guacamole /etc/nginx/sites-enabled/  
rm /etc/nginx/sites-enabled/default  
nginx -t && systemctl restart nginx tomcat
```

§ À savoir

- `proxy_buffering off` est **indispensable** : avec la mise en tampon, les sessions Guacamole se figent.
- `access_log off` évite de remplir les journaux Nginx avec le flux des sessions ; l'historique des connexions reste dans Guacamole.

Le bastion est accessible sur <https://bastion.mondomaine.local/guacamole/>.

9. Sécuriser le bastion

A. Remplacer le compte par défaut

Le schéma de base crée un administrateur **guacadmin** / **guacadmin**, connu de tous. À faire **immédiatement**, avant toute autre configuration :

- Se connecter avec **guacadmin** / **guacadmin**.
- **Paramètres → Utilisateurs → Nouvel utilisateur** : créer un compte administrateur **nominatif**, lui cocher toutes les permissions d'administration.
- Se déconnecter, se reconnecter avec ce nouveau compte.
- Supprimer le compte **guacadmin**.

B. Double authentification, anti-force brute et enregistrement des sessions

Trois extensions officielles complètent le dispositif. Elles se téléchargent depuis le dépôt officiel et se copient dans `/etc/guacamole/extensions/` :

Extension	Rôle
guacamole-auth-totp	Double authentification par code à usage unique (application d'authentification sur smartphone).
guacamole-auth-ban	Bloque temporairement une adresse IP après plusieurs échecs de connexion. Activée d'office avec Docker, mais à installer à la main ici.
guacamole-history-recording-storage	Permet de relire les sessions enregistrées directement dans l'historique de l'interface web.

```
cd /tmp
for EXT in auth-totp auth-ban history-recording-storage; do
  wget https://downloads.apache.org/guacamole/${VER}/binary/guacamole-${EXT}-${VER}.tar.gz
  tar -xzf guacamole-${EXT}-${VER}.tar.gz
  cp guacamole-${EXT}-${VER}/guacamole-${EXT}-${VER}.jar /etc/guacamole/extensions/
done
```

On crée le répertoire des enregistrements : **guacd** (compte daemon) y écrit, **Tomcat** y lit.

```
mkdir -p /var/lib/guacamole/recordings
chown daemon:tomcat /var/lib/guacamole/recordings
chmod 2750 /var/lib/guacamole/recordings
```

On ajoute à la fin de `/etc/guacamole/guacamole.properties` (valeurs par défaut, affichées pour être explicites) :

```
# Double authentification
totp-issuer: Bastion BTS SIO
# Anti-force brute : 5 échecs = 5 minutes de blocage
ban-max-invalid-attempts: 5
ban-address-duration: 300
```

```
systemctl restart tomcat
```

À la connexion suivante, chaque utilisateur doit **scanner un QR code** avec une application d'authentification, puis saisir le code à 6 chiffres à chaque connexion.

Pour enregistrer les sessions d'une connexion : dans ses paramètres, section **Enregistrement d'écran**, renseigner :

Paramètre	Valeur
Chemin de l'enregistrement	<code>\${HISTORY_PATH}/\${HISTORY_UUID}</code>
Créer automatiquement le chemin	coché
Nom de l'enregistrement	par exemple <code>session-\${GUAC_DATE}-\${GUAC_TIME}</code>

Enregistrer, c'est collecter des données

Les enregistrements montrent tout ce qui s'affiche à l'écran, y compris des données personnelles. En entreprise : **informer les administrateurs** concernés, fixer une **durée de conservation** et restreindre l'accès aux enregistrements (RGPD).

C. Pare-feu local (nftables)

Même derrière le pare-feu de la DMZ, le bastion filtre lui-même ses flux entrants : **défense en profondeur**. Adaptez le réseau des postes d'administration autorisés en SSH.

```
apt-get install -y nftables
nano /etc/nftables.conf
```

```
#!/usr/sbin/nft -f
flush ruleset
table inet filter {
    chain input {
        type filter hook input priority 0; policy drop;
        iif "lo" accept
        ct state established,related accept
        ct state invalid drop
        ip protocol icmp accept
        ip6 nexthdr icmpv6 accept
        tcp dport { 80, 443 } accept
        ip saddr 192.168.50.0/24 tcp dport 22 accept    # postes admin (à adapter)
    }
    chain forward {
        type filter hook forward priority 0; policy drop;
    }
}
```

```
nft -c -f /etc/nftables.conf
systemctl enable --now nftables
```

⚠ Attention au verrouillage

Si vous êtes connecté **en SSH**, vérifiez que votre poste fait bien partie du réseau autorisé **avant** d'activer les règles, sinon la session est coupée. La commande `nft -c` vérifie la syntaxe sans rien appliquer.

10. Vérifications finales

```
ss -tlnp
```

Résultat attendu (extrait) : seuls Nginx et SSH écoutent sur toutes les interfaces.

```
LISTEN 0.0.0.0:443      users:(("nginx",...))
LISTEN 0.0.0.0:80       users:(("nginx",...))
LISTEN 0.0.0.0:22       users:(("sshd",...))
LISTEN 127.0.0.1:8080    users:(("java",...))
LISTEN 127.0.0.1:8005    users:(("java",...))  <- arrêt de Tomcat
LISTEN 127.0.0.1:4822    users:(("guacd",...))
LISTEN 127.0.0.1:3306    users:(("mariadb",...))
```

Contrôle	Attendu
<code>http://</code> + adresse du bastion	Redirection automatique vers HTTPS
<code>http://<IP>:8080</code> depuis un autre poste	Connexion refusée
Connexion avec guacadmin	Refusée : le compte n'existe plus
Connexion avec le compte nominatif	Mot de passe puis code TOTP demandés
6 mots de passe faux depuis un poste	Adresse IP bloquée pendant 5 minutes
Session RDP / SSH vers un serveur	Fonctionne ; relecture possible dans Historique

§ Une connexion RDP échoue ?

Toujours commencer par le journal de guacd, qui donne la cause exacte : `journalctl -u guacd -n 50 --no-pager`.

- **Permission denied** sur `.config/freerdp` : la surcharge `systemd` de la section 5D manque.
- **Erreur de certificat** : le serveur Windows présente un certificat autosigné ; cocher « Ignorer le certificat du serveur » dans la connexion, ou mieux, déployer un certificat émis par l'autorité interne.
- **Échec d'authentification avec NLA** : vérifier identifiant et domaine (nom NetBIOS du domaine dans le champ Domaine), l'appartenance au groupe « Utilisateurs du Bureau à distance », et qu'aucune GPO n'interdit NTLM (le NLA de FreeRDP 2 utilise NTLM).
- **Délai dépassé** : tester le port depuis le bastion avec `timeout 3 bash -c '</dev/tcp/<IP-serveur>/3389'` ; en cas d'échec, voir le pare-feu Windows et le filtrage entre DMZ et serveurs.

 Rappel cybersécurité

L'objectif d'un bastion est d'**améliorer et de sécuriser les accès distants** par la mise en place de droits, de groupes et d'utilisateurs. Sa mise en place implique une bonne maîtrise des concepts de cybersécurité.

N'hésitez pas à **changer les noms d'utilisateurs, les mots de passe, les noms de bases de données et les ports** ! Les attaquants lisent les mêmes tutoriels que vous : quelques recherches et un peu d'**ingénierie sociale** leur suffisent.

Un bastion concentre tous les accès : s'il tombe, tout tombe. Il doit être **mis à jour en priorité** (paquets Debian et versions de Guacamole).

 Pour aller plus loin

- Relier Guacamole à l'**Active Directory** avec l'extension LDAP, pour ne plus gérer de comptes locaux.
- Donner l'accès externe via un **VPN WireGuard** sur le pare-feu, sans exposer le port 443 sur Internet.
- Remplacer le certificat autosigné par un certificat de l'**autorité de certification interne**.
- Redéployer le bastion avec les **images Docker officielles** (guacamole/guacd et guacamole/guacamole) et comparer les deux approches.

PARTIE 2

Utiliser et administrer le bastion

Se connecter, créer des connexions, travailler dans une session, gérer les droits, superviser

La première partie a construit un bastion sûr. Mais un bastion bien installé et mal utilisé reste un bastion vulnérable : la sécurité se joue aussi **au quotidien**, dans la façon de créer les connexions, d'attribuer les droits et de surveiller les accès. Chaque section de cette partie signale donc explicitement ce qui est **important côté cyber**.

Dans le cadre de vos épreuves

Pour vos activités d'étudiant, et notamment les **épreuves pratiques**, un bastion facilite l'accès à vos machines : le **jury** peut consulter votre réalisation depuis un simple navigateur, sans configuration sur son poste. C'est aussi l'occasion de démontrer une **gestion des accès maîtrisée** : compte dédié au jury, droits limités à vos machines, accès borné dans le temps (section 14.6).

11. Se connecter au bastion

11.1 Accéder à la page de connexion

Le bastion s'ouvre depuis n'importe quel navigateur récent, à l'adresse :

```
https://bastion.mondomaine.local/guacamole/
```

Il n'y a **pas de numéro de port** à indiquer : Nginx répond sur le port HTTPS standard (443). Le port 8080 de Tomcat n'est pas joignable depuis le réseau (partie 1, section 8).

[Capture à insérer : page de connexion Guacamole, avec le cadenas HTTPS visible dans la barre d'adresse]

Important côté cyber — vérifier où l'on tape son mot de passe

- On ne saisit ses identifiants que si l'adresse commence par **https://** et correspond au bastion. Une page qui imite le bastion à une adresse voisine est un classique de l'**hameçonnage** (phishing).
- Avec le certificat autosigné de la maquette, le navigateur affiche un avertissement : on vérifie l'**empreinte du certificat** une première fois, puis on l'accepte. En production, un certificat de l'autorité interne supprime l'avertissement : un avertissement qui apparaît soudain doit alors **alerter**, pas être contourné.

11.2 Première connexion et double authentification

- Saisir son **nom d'utilisateur** et son **mot de passe**, puis cliquer sur **Se connecter**.
- Si le compte a été créé avec l'option « Mot de passe expiré » (section 14.5), Guacamole impose immédiatement le choix d'un **nouveau mot de passe**, conforme à la politique définie en section 14.7.
- Lors de la toute première connexion, Guacamole affiche un **QR code** : le scanner avec une application d'authentification (sur smartphone), puis saisir le code à 6 chiffres affiché.
- Aux connexions suivantes, le code à 6 chiffres est demandé après le mot de passe.

[Capture à insérer : écran d'enrôlement TOTP avec le QR code (flouter le QR code avant de publier la capture)]

⚠ Important côté cyber — le second facteur

Le QR code contient la **clé secrète** du compte : ne jamais le photographier, le partager ni le laisser à l'écran. Une capture d'écran de ce QR code publiée dans un compte rendu suffit à cloner le second facteur. En cas de perte du téléphone, c'est l'**administrateur** qui réinitialise l'enrôlement depuis la fiche de l'utilisateur ; l'utilisateur ne peut pas le faire seul.

11.3 Se repérer dans l'interface

Zone	Contenu
Page d'accueil	« Connexions récentes » (vignettes des dernières sessions) et « Toutes les connexions », rangées par groupes.
Menu utilisateur (en haut à droite)	Accueil, Paramètres, Se déconnecter.
Paramètres	Onglets Sessions actives, Historique, Utilisateurs, Groupes, Connexions, Préférences . Un utilisateur ne voit que les onglets correspondant à ses droits.
Préférences	Langue de l'interface, méthode de saisie par défaut, changement de mot de passe .

[Capture à insérer : page d'accueil d'un compte nominatif, avec les groupes de connexions dépliés]

§ À savoir

Les libellés de l'interface peuvent varier légèrement selon la version de Guacamole et la langue choisie dans les préférences. Ce livret utilise ceux de la version 1.6.0 en français.

12. Créer une connexion vers une machine distante

Une **connexion** est une fiche enregistrée dans le bastion : elle décrit **comment** joindre une machine (protocole, adresse, port) et **avec quelles options de sécurité**. Les utilisateurs n'ont jamais à connaître l'adresse des serveurs : ils cliquent sur le nom de la connexion.

Pour en créer une : **Paramètres → Connexions → Nouvelle connexion**. Le formulaire est découpé en sections ; seules les rubriques utiles sont détaillées ci-dessous.

[Capture à insérer : formulaire « Nouvelle connexion », sections repliées]

12.1 Rubriques communes à toutes les connexions

Rubrique	Valeur conseillée	Pourquoi
Nom	Explicite et normalisé, ex. SRV-DC01 – Contrôleur de domaine (RDP)	L'historique et les journaux doivent être lisibles par quelqu'un d'autre que vous.
Emplacement	Le groupe de connexions adapté (section 14.3)	Les groupes structurent l'attribution des droits.
Protocole	RDP (Windows) ou SSH (Linux, équipements réseau)	Jamais Telnet : non compilé sur ce bastion.
Limites de simultanéité	Serveur sensible : 1 connexion par utilisateur	Évite les sessions fantômes ouvertes sur plusieurs postes.
Enregistrement d'écran	Chemin <code>\${HISTORY_PATH}/\${HISTORY_UUID}</code> , création automatique cochée	Traçabilité (partie 1, section 9).

12.2 Connexion RDP (serveurs et postes Windows)

Section	Paramètre	Valeur
Réseau	Nom d'hôte	Adresse IP ou nom DNS du serveur
Réseau	Port	3389
Authentification	Nom d'utilisateur / Mot de passe	Voir l'encadré « identifiants » ci-dessous
Authentification	Domaine	Nom NetBIOS du domaine (ex. DOMBTSIG) pour un compte Active Directory
Authentification	Mode de sécurité	NLA (authentification avant ouverture de session)
Authentification	Ignorer le certificat du serveur	Coché uniquement si le serveur utilise un certificat autosigné
Presse-papiers	Désactiver la copie / le collage	Selon la sensibilité du serveur (encadré ci-dessous)
Partage de périphériques	Activer le lecteur	Non coché : pas de transfert de fichiers par défaut

[Capture à insérer : section Authentification d'une connexion RDP : NLA, domaine, certificat]

§ Pourquoi NLA ?

Avec **NLA** (Network Level Authentication), l'utilisateur est authentifié **avant** que Windows n'ouvre une session graphique. Sans NLA, n'importe qui joignant le port 3389 obtient l'écran de connexion Windows, ce qui consomme des ressources et expose davantage le serveur aux attaques. NLA est le mode à conserver en production.

12.3 Connexion SSH (serveurs Linux, équipements réseau)

Section	Paramètre	Valeur
Réseau	Nom d'hôte	Adresse IP ou nom DNS
Réseau	Port	22 (ou le port SSH du serveur)
Réseau	Clé publique de l'hôte	La clé publique SSH du serveur (voir ci-dessous)
Authentification	Nom d'utilisateur	Compte nominatif sur le serveur, jamais root
Authentification	Clé privée (et phrase de passe)	À privilégier par rapport au mot de passe

La clé publique du serveur se récupère **sur le serveur lui-même** (par une voie sûre, par exemple la console de la VM) :

```
cat /etc/ssh/ssh_host_ed25519_key.pub
```

⚠ Important côté cyber — vérifier l'identité du serveur SSH

Renseigner la **clé publique de l'hôte** permet à guacd de vérifier qu'il parle bien au vrai serveur. Si un attaquant s'intercale (attaque de **l'homme du milieu**) ou si l'adresse IP est réattribuée à une autre machine, la clé ne correspond plus et la connexion est **refusée** au lieu de transmettre vos identifiants à l'imposteur.

12.4 Où mettre les identifiants des machines ?

C'est la décision la plus importante lors de la création d'une connexion. Trois possibilités :

Option	Fonctionnement	Avantages	Risques
A. Identifiants enregistrés dans la connexion	Nom d'utilisateur et mot de passe saisis dans la fiche	Connexion en un clic	Stockés en clair dans la base ; tous les utilisateurs de la connexion agissent sous le même compte : on ne sait plus qui a fait quoi sur le serveur
B. Champs laissés vides	Guacamole demande les identifiants au moment de la connexion	Chacun utilise son propre compte ; rien n'est stocké	Une saisie de plus à chaque connexion
C. Jetons \$ {GUAC_USERNAME} et \${GUAC_PASSWORD}	Guacamole réutilise les identifiants saisis pour se connecter au bastion	Compte personnel, sans ressaisie, rien n'est stocké	Suppose les mêmes identifiants sur le bastion et sur les serveurs, donc un bastion relié à l'annuaire (extension LDAP)

⚠ Important côté cyber — ne pas stocker de mots de passe dans les connexions

La documentation officielle le précise : les paramètres des connexions, **mots de passe compris**, sont enregistrés **en clair** dans la base de données. Quiconque obtient une copie de la base (sauvegarde égarée, compte applicatif compromis) récupère donc tous les mots de passe des serveurs.

Choix recommandé : **option B** avec des comptes nominatifs, ou **option C** quand le bastion est relié à l'Active Directory. L'option A est à réserver à une maquette sans données réelles.

12.5 Presse-papiers et transfert de fichiers

Par défaut, le presse-papiers est partagé entre le poste de l'utilisateur et la machine distante. C'est pratique pour coller une commande, mais c'est aussi un **canal de fuite** : on peut copier des données sensibles hors du serveur, ou coller un contenu malveillant sur le serveur.

Type de machine	Copie depuis la session	Collage vers la session	Lecteur / transfert
Poste ou serveur de TP	Autorisée	Autorisé	Selon le besoin
Serveur de production	Désactivée	Autorisé	Désactivé
Serveur très sensible (contrôleur de domaine, hyperviseur)	Désactivée	Désactivé	Désactivé

13. Travailler dans une session

13.1 Le menu Guacamole

Pendant une session, l'écran de la machine distante occupe tout le navigateur : il n'y a **pas de barre d'outils permanente**. Les commandes sont regroupées dans le **menu Guacamole**, qui s'ouvre et se ferme avec la combinaison :

Ctrl + Alt + Maj

[Capture à insérer : menu Guacamole ouvert sur une session RDP]

Élément du menu	Usage
Presse-papiers	Zone de texte synchronisée avec la machine distante (si le presse-papiers n'est pas désactivé pour la connexion).
Méthode de saisie : aucune	Fonctionnement normal au clavier physique.
Méthode de saisie : saisie de texte	Utile sur tablette ou pour certaines dispositions de clavier.
Méthode de saisie : clavier à l'écran	Clavier virtuel, qui permet notamment d'envoyer Ctrl+Alt+Suppr à la machine distante.
Mode souris	Pointeur absolu (souris) ou relatif (écran tactile).
Zoom	Ajuster l'affichage à la taille de la fenêtre.

Élément du menu	Usage
Partager	Générer un lien de partage de la session, uniquement si un profil de partage existe (section 14.8).
Nom de la connexion / Déconnecter	Revenir à l'accueil, basculer vers une autre session ouverte, ou fermer la connexion.

§ Astuces

- **Ctrl+Alt+Suppr** tapé au clavier est intercepté par votre propre poste : passez par le **clavier à l'écran** du menu.
- Le **plein écran** se fait avec la touche **F11** du navigateur.
- La résolution et la qualité d'affichage se règlent dans la **connexion** (section Écran), par l'administrateur : elles ne se changent pas en cours de session.

13.2 Terminer proprement une session

Fermer l'onglet du navigateur **ne ferme pas** la session sur le serveur : elle reste ouverte, déconnectée. On termine donc toujours par la **déconnexion de la session distante** (menu Démarrer de Windows, commande **exit** sous Linux), puis par la déconnexion du bastion (menu utilisateur → **Se déconnecter**).

⚠ Important côté cyber — les sessions laissées ouvertes

- Une session administrateur restée ouverte sur un serveur peut être **reprise** par une autre personne, ou exploitée par un programme malveillant qui profite des droits déjà obtenus.
- Poste de travail quitté, même deux minutes : **verrouillage** (Windows + L).
- Côté administrateur, l'onglet **Sessions actives** permet de repérer et de fermer les sessions oubliées (section 15).

14. Gérer les droits des utilisateurs

C'est le cœur du bastion. Toutes les protections de la partie 1 (HTTPS, TOTP, pare-feu) servent à une seule chose : garantir que **chaque personne n'accède qu'aux machines dont elle a besoin, et que l'on sait qui a fait quoi**. Une gestion des droits approximative annule tous ces efforts.

⚠ Important côté cyber — pourquoi les droits du bastion sont critiques

Le bastion concentre **tous les accès d'administration**. Un compte aux droits trop larges, s'il est compromis (mot de passe deviné, poste infecté, départ d'un stagiaire non désactivé), donne à l'attaquant **toutes les machines** de ce compte d'un seul coup : c'est le scénario classique de **déplacement latéral** vers les serveurs critiques.

Une bonne gestion des droits limite les dégâts d'une compromission : l'attaquant n'obtient que ce que le compte volé avait le droit de faire.

14.1 Les principes à appliquer

Principe	Ce qu'il signifie	Traduction dans Guacamole
Moindre privilège	Chacun reçoit uniquement les droits nécessaires à sa mission, rien de plus.	Aucune permission système par défaut ; accès aux seules connexions utiles.
Besoin d'en connaître	On ne voit même pas l'existence des ressources qui ne nous concernent pas.	Un utilisateur ne voit que les connexions sur lesquelles il a un droit.
Comptes nominatifs	Un compte = une personne identifiée. Jamais de compte partagé.	Pas de compte « étudiant » ou « technicien » commun : l'historique doit désigner une personne.
Séparation des rôles	Celui qui administre les comptes n'est pas forcément celui qui utilise les serveurs, ni celui qui contrôle.	Rôles distincts : administrateur du bastion, administrateurs système, auditeur.
Gestion par groupes (RBAC)	On attribue les droits à des rôles , puis on place les personnes dans les rôles.	Groupes d'utilisateurs, groupes de connexions ; jamais de droit attribué directement à une personne sauf exception documentée.
Revue périodique	Les droits se vérifient régulièrement : ils s'accumulent avec le temps.	Revue à chaque semestre et à chaque départ (section 14.9).

14.2 Le modèle de droits de Guacamole

Guacamole manipule quatre types d'objets : les **utilisateurs**, les **groupes d'utilisateurs**, les **connexions** et les **groupes de connexions** (plus les profils de partage). Il existe deux familles de permissions.

1. Les permissions système : ce qu'un compte peut faire sur le bastion lui-même.

Permission (nom interne)	Ce qu'elle autorise	À qui la donner
ADMINISTER — Administrer le système	Tout, sans restriction : c'est un « joker » qui englobe toutes les autres permissions.	2 personnes maximum : les administrateurs du bastion.
AUDIT	Consulter l'historique des connexions et des authentifications de tout le système.	Un auditeur ou un responsable sécurité qui contrôle sans administrer.
CREATE_USER — Créer des utilisateurs	Créer de nouveaux comptes.	Administrateurs du bastion uniquement.
CREATE_USER_GROUP	Créer de nouveaux groupes d'utilisateurs.	Administrateurs du bastion uniquement.
CREATE_CONNECTION	Créer de nouvelles connexions.	Éventuellement un administrateur système chargé de déclarer ses serveurs.
CREATE_CONNECTION_GROUP	Créer des groupes de connexions.	Administrateurs du bastion.
CREATE_SHARING_PROFILE	Créer des profils de partage de session.	Administrateurs du bastion, et seulement si le partage est souhaité.

Dans le formulaire d'un utilisateur, une case supplémentaire « **Modifier son propre mot de passe** » permet à la personne de changer elle-même son mot de passe : elle doit être **cochée pour tous**.

2. Les permissions sur objet : ce qu'un compte peut faire sur **un** utilisateur, **une** connexion ou **un** groupe précis.

Permission	Sur une connexion	Sur un utilisateur ou un groupe
READ (lecture)	Voir la connexion et l'utiliser : c'est le droit d'accès normal.	Voir le compte ou le groupe.
UPDATE (modification)	Modifier ses paramètres (adresse, identifiants, options).	Modifier le compte, y compris son mot de passe .
DELETE (suppression)	Supprimer la connexion.	Supprimer le compte ou le groupe.
ADMINISTER	Donner ou retirer des droits sur cette connexion à d'autres.	Donner ou retirer des droits à ce compte .

⚠ Important côté cyber — les droits qui permettent de devenir administrateur

- **UPDATE sur un compte administrateur** permet de changer son mot de passe, donc de **prendre son identité**.
- **ADMINISTER sur son propre compte ou son propre groupe** permet de s'attribuer n'importe quel droit.
- **UPDATE sur une connexion** permet de la rediriger vers une autre machine ou d'y lire les identifiants stockés.

Ces droits ne se donnent qu'aux administrateurs du bastion. Pour un utilisateur ordinaire, la seule permission sur objet est **READ** sur les connexions dont il a besoin.

§ Les droits s'additionnent, ils ne se retirent pas

Guacamole n'a **pas de permission d'interdiction**. Un utilisateur membre de plusieurs groupes reçoit la **somme** de tous leurs droits, y compris ceux hérités des groupes parents (l'héritage est récursif, quelle que soit la profondeur). Conséquence pratique : pour retirer un accès, il faut retirer la personne du groupe qui le donne, ou retirer le droit à ce groupe. Ajouter un autre groupe ne sert à rien.

14.3 Organiser les connexions en groupes

Un **groupe de connexions** de type **organisationnel** fonctionne comme un dossier. On regroupe les machines selon un critère qui correspond à la façon dont on attribue les accès, en général la **fonction** et la **sensibilité** des serveurs :

Groupe de connexions	Contenu (exemple du plateau technique)	Sensibilité
Contrôleurs de domaine	SRV-DC01, SRV-DC02	Critique
ESX	Hyperviseurs de la ferme de virtualisation	Critique
Serveurs applicatifs Linux	Serveurs web, bases de données	Élevée
Serveurs gestion de parc Linux	GLPI, supervision, WAPT...	Élevée

Groupe de connexions	Contenu (exemple du plateau technique)	Sensibilité
Machines de TP — [groupe d'étudiants]	VM des étudiants pour leurs réalisations	Modérée

§ À savoir

Un groupe de type « **répartition de charge** » (balancing) est un cas particulier : il envoie l'utilisateur vers la connexion la moins chargée du groupe. On ne l'utilise que pour des serveurs identiques (par exemple une ferme de serveurs Bureau à distance), pas pour ranger des machines.

14.4 Méthode : construire une matrice des droits

Avant de cliquer dans l'interface, on construit **sur papier** une matrice qui croise les **rôles** (groupes d'utilisateurs) et les **groupes de connexions**. Elle sert de référence, se relit facilement, et permet de vérifier les droits lors des revues.

Étape 1 — Lister les rôles, c'est-à-dire les profils de personnes ayant des besoins différents :

Groupe d'utilisateurs	Qui ?	Permissions système
GRP-Admins-Bastion	Les 2 administrateurs du bastion	ADMINISTER
GRP-Admins-Infra	Enseignants responsables de l'infrastructure	aucune
GRP-SISR2-[groupe]	Étudiants de 2e année SISR d'un groupe de projet	aucune
GRP-Jury-E5	Membres du jury, le temps de l'épreuve	aucune
GRP-Auditeurs	Responsable sécurité ou enseignant chargé du contrôle	AUDIT

Étape 2 — Remplir la matrice : quelles connexions pour quel rôle ?

	Contrôleurs de domaine	ESX	Serveurs applicatifs Linux	Gestion de parc Linux	Machines de TP du groupe
GRP-Admins-Bastion	Accès	Accès	Accès	Accès	Accès
GRP-Admins-Infra	Accès	Accès	Accès	Accès	—
GRP-SISR2-[groupe]	—	—	—	—	Accès
GRP-Jury-E5	—	—	—	—	Accès, limité dans le temps
GRP-Auditeurs	—	—	—	—	—

« Accès » signifie la permission **READ** : voir et utiliser les connexions. L'auditeur n'a aucun accès aux machines : il consulte l'historique et les enregistrements, ce qui garantit qu'il contrôle sans pouvoir agir.

Étape 3 — Appliquer dans Guacamole :

- **Paramètres → Groupes → Nouveau groupe** : créer chaque groupe d'utilisateurs.

- Dans chaque groupe, onglet **Connexions** : cocher les groupes de connexions et les connexions prévus par la matrice.
- Dans la fiche de chaque utilisateur, onglet **Groupe(s)** : l'ajouter au(x) groupe(s) correspondant à son rôle.
- Ne cocher **aucune permission directement** sur les utilisateurs, sauf « Modifier son propre mot de passe ».

[Capture à insérer : fiche d'un groupe d'utilisateurs, onglet Connexions, avec les cases cochées]

Étape 4 — Tester avec un compte de test : se connecter avec un compte de chaque rôle et vérifier qu'il voit **exactement** ce que prévoit la matrice, ni plus ni moins. C'est la **recette des droits** : sans ce test, on ne sait pas si les droits sont corrects.

14.5 Créer un compte utilisateur

Paramètres → Utilisateurs → **Nouvel utilisateur**. Le formulaire comporte plusieurs rubriques.

Rubrique	Réglage	Explication
Identifiant	<code>prenom.nom</code>	Nominatif, selon une convention unique pour tout le bastion.
Mot de passe	Mot de passe provisoire aléatoire	Transmis à la personne par un canal séparé (de vive voix, pas dans le même courriel que l'identifiant).
Mot de passe expiré	Coché	Oblige la personne à choisir son propre mot de passe à la première connexion : l'administrateur ne le connaît plus.
Connexion désactivée	Non coché (coché pour suspendre un compte)	Un compte désactivé est refusé comme s'il n'existait pas, mais son historique est conservé.
Profil	Nom complet, adresse électronique	Permet de savoir qui est derrière l'identifiant lors d'un audit.
Permissions	Uniquement « Modifier son propre mot de passe »	Les autres droits viennent des groupes.
Groupe(s)	Le ou les groupes du rôle	Voir la matrice (section 14.4).

14.6 Restreindre un compte dans le temps

Chaque compte peut être limité **dans le temps**, ce qui est particulièrement utile pour les comptes temporaires (jury, stagiaire, prestataire) :

Restriction	Effet	Exemple
Activer le compte après (date)	Aucune connexion possible avant cette date.	Jour de l'épreuve
Désactiver le compte après (date)	Aucune connexion possible après cette date.	Lendemain de l'épreuve
Autoriser l'accès après (heure)	Connexion refusée avant cette heure, chaque jour.	07:30
Ne pas autoriser l'accès après (heure)	Connexion refusée après cette heure, chaque jour.	18:30

Restriction	Effet	Exemple
Fuseau horaire	Fuseau utilisé pour interpréter ces dates et heures.	Europe/Paris

✓ Cas pratique : le compte du jury

Pour une épreuve pratique, on crée un compte **nominatif** par membre du jury (`prenom.nom.jury`), placé dans **GRP-Jury-E5**, qui ne donne accès **qu'aux machines du candidat**. On règle :

- **Activer le compte après** : le jour de l'épreuve ; **Désactiver le compte après** : le lendemain.
- **Autoriser l'accès** de 08:00 à 18:00, fuseau **Europe/Paris**.
- Enregistrement des sessions **activé** sur les connexions concernées.

Le compte se ferme **tout seul** : aucun risque d'oubli. C'est un excellent point à présenter au jury lui-même, comme démonstration d'une gestion des accès maîtrisée.

14.7 Imposer une politique de mots de passe

La politique s'applique à tous les comptes de la base. On ajoute à `/etc/guacamole/guacamole.properties` :

```
# Politique de mots de passe des comptes Guacamole
mysql-user-password-min-length: 12
mysql-user-password-require-multiple-case: true
mysql-user-password-require-digit: true
mysql-user-password-require-symbol: true
mysql-user-password-prohibit-username: true
mysql-user-password-history-size: 5
```

```
systemctl restart tomcat
```

Paramètre	Effet
<code>min-length: 12</code>	12 caractères minimum.
<code>require-multiple-case</code>	Au moins une majuscule et une minuscule.
<code>require-digit / require-symbol</code>	Au moins un chiffre et un caractère spécial.
<code>prohibit-username</code>	Le mot de passe ne peut pas contenir l'identifiant.
<code>history-size: 5</code>	Impossible de réutiliser l'un des 5 derniers mots de passe.

§ Et l'expiration périodique ?

Guacamole peut imposer un changement tous les N jours (`mysql-user-password-max-age`). L'ANSSI déconseille les changements fréquents, qui poussent aux mots de passe faibles : elle privilégie un mot de passe **long et robuste**, associé à un **second facteur**, et un changement immédiat **en cas de doute de compromission**. Pour les **comptes à privilèges**, comme ceux d'un bastion, elle recommande toutefois un renouvellement **peu fréquent**, de l'ordre de l'année : `mysql-user-password-max-age: 365` est un réglage cohérent.

14.8 Les profils de partage

Un **profil de partage** permet à un utilisateur de générer un **lien** donnant accès à sa session en cours, par exemple pour qu'un enseignant observe la manipulation d'un étudiant. Le profil peut être en **lecture seule** (option « Lecture seule » du profil).

Important côté cyber — les liens de partage

Une personne qui possède le lien accède à la session **sans compte Guacamole**, donc sans mot de passe ni TOTP. Un lien transmis par messagerie peut être réutilisé par n'importe qui tant que la session est ouverte.

Recommandation : ne créer des profils de partage que si le besoin est réel, toujours en **lecture seule**, et ne les rendre disponibles que sur les connexions de TP, **jamais** sur les serveurs critiques.

14.9 Faire vivre les droits : arrivées, changements, départs

Événement	Action	Délai
Arrivée	Création du compte nominatif (14.5), ajout au groupe du rôle, remise du mot de passe provisoire.	Le jour de l'arrivée, pas avant
Changement de rôle	Ajout au nouveau groupe et retrait de l'ancien .	Immédiat
Départ	Cocher « Connexion désactivée » ; suppression après la durée de conservation de l'historique.	Le jour même
Perte du téléphone (TOTP)	Réinitialisation de l'enrôlement TOTP dans la fiche de l'utilisateur, après vérification de son identité.	À la demande
Revue des droits	Comparer les groupes et leurs membres avec la matrice ; supprimer les comptes inutilisés.	Chaque semestre

Important côté cyber — l'accumulation des droits

Le phénomène s'appelle l'**accumulation de privilèges** : on ajoute des droits à chaque nouveau besoin, on oublie de retirer les anciens. Au bout de deux ans, un étudiant devenu apprenti puis salarié a gardé tous les accès de ses différentes missions. Les comptes d'anciens utilisateurs jamais désactivés sont l'une des portes d'entrée les plus fréquentes lors des attaques : la revue périodique et la désactivation **le jour du départ** ne sont pas des formalités.

Exercice : matrice des droits de votre projet

- Listez les rôles qui interviennent sur votre projet (vous, votre binôme, l'enseignant, le jury, un auditeur).
- Rangez vos machines en groupes de connexions selon leur sensibilité.
- Construisez la matrice rôles × groupes de connexions, puis appliquez-la dans Guacamole.
- Créez un compte de test par rôle et réalisez la recette des droits : chaque compte voit-il exactement ce qui est prévu ?
- Configurez le compte du jury avec des dates et des horaires de validité, puis vérifiez qu'une connexion en dehors de ces créneaux est refusée.

15. Superviser l'activité

Onglet (Paramètres)	Ce qu'il montre	Usage
Sessions actives	Toutes les sessions en cours : utilisateur, connexion, adresse IP d'origine, durée.	Repérer une session oubliée ou suspecte, et la fermer (cocher, puis « Terminer les sessions »).
Historique	Les connexions passées : qui, quelle machine, quand, combien de temps, et si un enregistrement existe.	Répondre à « qui s'est connecté au contrôleur de domaine mardi soir ? » ; relire la session enregistrée.

[Capture à insérer : onglet Historique avec la colonne de relecture des enregistrements]

⚠ Important côté cyber — surveiller, c'est aussi détecter

Les journaux ne servent que s'ils sont **lus**. Quelques signaux qui doivent alerter : une connexion en pleine nuit, une adresse IP d'origine inhabituelle, un compte qui ouvre des sessions sur des machines qu'il n'utilise jamais, des échecs d'authentification répétés.

Consulter l'historique **chaque semaine** est un bon réflexe. En entreprise, ces journaux sont envoyés vers un outil centralisé (SIEM, par exemple Wazuh ou Graylog) qui génère des alertes automatiquement.

16. Panorama des solutions de bastion

Les solutions de bastion d'administration permettent de gérer de manière sécurisée les accès aux systèmes critiques.

Solutions libres (open source)

Solution	Caractéristiques
Apache Guacamole	Bastion RDP, VNC et SSH accessible depuis un navigateur web. Très populaire pour les accès distants. Licence Apache 2.0.
The Bastion (OVHcloud)	Bastion SSH open source développé et utilisé par OVHcloud : traçabilité fine, enregistrement des sessions, gestion des clés.
Teleport (Gravitational)	Accès unifié SSH, Kubernetes, bases de données... Code source sous AGPLv3, mais depuis la v16 les binaires « Community » sont sous licence commerciale : gratuits seulement pour les structures de moins de 100 salariés et 10 M\$ de chiffre d'affaires.
Pritunl Zero	Bastion d'administration à approche Zero Trust, multi-protocoles, interface simplifiée.

Solutions propriétaires

Solution	Caractéristiques
Wallix Bastion	Solution française de gestion des accès et sessions à privilèges ; certifiée par l'ANSSI (CSPN).
CyberArk Privileged Access Security	Une des références de la gestion des accès à privilèges.
BeyondTrust Privileged Remote Access	Sécurisation des accès administratifs, avec enregistrement et audit.
Delinea (ex-Thycotic et Centrify)	Plateforme complète de gestion des accès à privilèges, dont Secret Server , coffre-fort de mots de passe.
Senhasegura	Concurrent de CyberArk et Wallix, présent en Europe et en Amérique latine.
JumpCloud	Plateforme SaaS de gestion des identités et des accès, avec des fonctions d'accès distant.

§ Bastion ou gestion d'identité ?

Des outils comme **Keycloak** (open source) sont des gestionnaires d'**identité** (IAM / SSO), pas des bastions : ils ne relaient pas les sessions RDP ou SSH. Ils se combinent en revanche très bien avec un bastion : Guacamole peut déléguer l'authentification à Keycloak via OpenID Connect ou SAML.

Référence : guide de l'ANSSI « **Recommandations relatives à l'administration sécurisée des systèmes d'information** ».

Le choix dépend des besoins : l'open source pour la **flexibilité**, le propriétaire pour la **conformité**, les enregistrements avancés et le **support technique**.

17. Évolutions par rapport à la version précédente

Élément	Avant	Maintenant
Guacamole	1.5.5	1.6.0, archive vérifiée par empreinte SHA-256
Tomcat	Tomcat 9 via le dépôt Debian 11 (fin de vie)	Tomcat 9 officiel Apache, archive vérifiée, compte de service dédié, applications par défaut supprimées
guacd	Écoute sur 0.0.0.0	Écoute sur 127.0.0.1
Accès web	HTTP en clair sur 8080	HTTPS via Nginx ; Tomcat en local
Compte guacadmin	Non mentionné	Remplacé par un compte nominatif
MFA	Recommandée, non mise en œuvre	Extension TOTP
Force brute	—	Extension auth-ban
Enregistrement des sessions	—	Activé et relisible dans l'interface

Élément	Avant	Maintenant
Pare-feu local	—	nftables, SSH limité aux postes d'administration
Telnet	Compilé	Non compilé
guacd et RDP	Connexions RDP en échec (FreeRDP sans répertoire accessible)	Surcharge systemd : répertoire /var/lib/guacd pour le compte daemon
Secrets	guacamole.properties lisible par tous	chmod 640, propriétaire root:tomcat
Panorama	Doublons et erreurs de licence	Corrigé ; ajout de The Bastion et du guide ANSSI
Utilisation	Document séparé, en partie inexact (URL en :8080, barre d'outils, captures avec guacadmin)	Partie 2 réécrite : connexion, création des connexions RDP/SSH, menu de session
Droits des utilisateurs	Absent	Section 14 : principes, modèle de droits, matrice rôles × connexions, restrictions temporelles, politique de mots de passe, cycle de vie
Dépannage RDP	Absent	Annexe A : démarche, tableau de diagnostic, pièges de la partie 2, réflexion sur NLA, fiche de suivi d'incident
Identifiants des serveurs	Enregistrés dans les connexions	Déconseillé (stockage en clair) : saisie à la connexion ou jetons \${GUAC_USERNAME}/\${GUAC_PASSWORD}

Relecture du 29 septembre 2026. Les captures d'écran de l'ancienne version (1.5.5, Debian 11) ont été retirées : les résultats attendus sont donnés sous forme de texte. Source d'origine : IT-Connect, procédure adaptée et mise à jour d'après la documentation officielle Apache Guacamole 1.6.0.

Annexe A. Dépannage des connexions RDP

Une connexion RDP qui échoue à travers le bastion peut avoir une dizaine de causes différentes, situées à des endroits différents : le bastion, le réseau, le serveur Windows ou l'Active Directory. Plutôt que de modifier des réglages au hasard, on suit une **démarche** : lire le message, en déduire où se situe le problème, tester une hypothèse à la fois.

La règle d'or du dépannage

- **Lire avant d'agir** : le journal de guacd donne presque toujours la cause exacte.
- **Un seul changement à la fois**, puis un nouveau test : sinon on ne sait pas ce qui a réglé le problème.
- **Noter** ce qui a été modifié, pour pouvoir revenir en arrière et documenter la solution.
- **Ne jamais « débloquer » en affaiblissant la sécurité** sans comprendre la cause (voir A.5 sur NLA).

A.1 Recueillir les informations

Avant tout, relancer une tentative de connexion, puis noter **trois choses** :

- le **message affiché** dans le navigateur par Guacamole ;
- si le problème touche **toutes** les connexions RDP ou **une seule** ;
- **ce qui a changé** depuis la dernière fois où cela fonctionnait (nouvelle connexion, nouveau réglage, mise à jour Windows, nouvelle GPO).

Ensuite, lire le journal de guacd au moment de l'échec :

Installation	Commande
Native (partie 1)	<code>journalctl -u guacd --since "5 min ago" --no-pager</code>
Conteneurs (variante Docker)	<code>cd /opt/bastion && docker compose logs guacd --since 5m</code>

§ Toutes ou une seule ?

- **Toutes les connexions RDP échouent** : la cause est probablement sur le **bastion** (guacd, son répertoire de travail, une mise à jour).
- **Une seule connexion échoue** : la cause est probablement dans **cette connexion** (paramètres) ou sur **ce serveur** (pare-feu, NLA, certificat, compte).

A.2 Tableau de diagnostic

Repérer dans le journal ou à l'écran le message le plus proche, puis appliquer la vérification correspondante.

Message ou symptôme	Cause probable	Vérification / correction
<code>Permission denied</code> sur un chemin en <code>.config/freerdp</code> ; la session se ferme aussitôt	guacd (compte daemon) ne peut pas écrire sa configuration FreeRDP	Surcharge systemd de la section 5D : <code>systemctl show guacd -p Environment</code> doit afficher <code>HOME=/var/lib/guacd</code>

Message ou symptôme	Cause probable	Vérification / correction
Erreur de certificat ou de TLS	Le serveur présente un certificat autosigné	Cocher « Ignorer le certificat du serveur » dans la connexion, ou déployer un certificat de l'autorité interne
Server refused connection (wrong security type?)	Le mode de sécurité de la connexion ne correspond pas à celui du serveur	Comparer le mode choisi dans Guacamole (NLA) et le réglage du serveur Windows (A.5)
Échec d'authentification alors que le mot de passe semble bon	Identifiant, domaine, droits RDP, compte verrouillé ou NTLM restreint	Voir A.3
Connexion « déjà utilisée » ou refusée dès la deuxième tentative	Limite de 1 connexion par utilisateur et session fantôme restée ouverte	Paramètres → Sessions actives : terminer l'ancienne session (A.4)
Délai dépassé , connexion réseau refusée	Le port 3389 n'est pas joignable depuis le bastion	Voir A.6

A.3 Échec d'authentification avec NLA

Avec NLA, Windows vérifie les identifiants **avant** d'ouvrir la session. Si l'un des points suivants est faux, la connexion est refusée :

Point à vérifier	Valeur attendue	Où vérifier
Nom d'utilisateur	Identifiant seul, ex. jdupont (sans domaine)	Paramètres de la connexion → Authentification
Domaine	Nom NetBIOS : DOMBTSIG (ou bien identifiant jdupont@dombtsig.local et champ Domaine vide)	Paramètres de la connexion → Authentification
Droit de connexion RDP	Compte membre du groupe Utilisateurs du Bureau à distance du serveur (ou administrateur)	Gestion de l'ordinateur → Groupes locaux, ou GPO
Compte non verrouillé	Pas de verrouillage après des tentatives répétées	« Utilisateurs et ordinateurs Active Directory » → propriétés du compte → onglet Compte
NTLM autorisé	Aucune GPO « Sécurité réseau : restreindre NTLM » qui refuse NTLM vers ce serveur	gpresult /h rapport.html sur le serveur

§ Pourquoi NTLM ?

Le client RDP de Guacamole (FreeRDP 2) s'authentifie en NLA via **NTLM** ; il ne sait pas utiliser Kerberos. Un poste Windows avec **mstsc** peut donc se connecter (il utilise Kerberos) alors que le bastion échoue sur le même serveur : c'est le signe d'une restriction NTLM.

⚠ Important côté cyber — les échecs répétés

Une connexion mal configurée (ancien mot de passe enregistré, mauvais domaine) qui réessaie peut **verrouiller le compte dans l'Active Directory**. À l'inverse, des échecs répétés que l'on n'a pas provoqués soi-même sont un **signal d'alerte** : quelqu'un tente peut-être de deviner un mot de passe. Il faut toujours identifier l'origine des échecs avant de déverrouiller un compte.

A.4 Pièges liés aux réglages de la partie 2

Réglage	Effet inattendu	Solution
Jetons <code>\${GUAC_USERNAME}</code> / <code>\${GUAC_PASSWORD}</code> (section 12.4, option C)	Guacamole transmet à Windows les identifiants du compte Guacamole . Si le bastion n'est pas relié à l'AD, ce compte n'existe pas dans le domaine : l'authentification NLA échoue.	Laisser les champs vides (option B) tant que l'extension LDAP n'est pas en place.
Limite de 1 connexion par utilisateur (section 12.1)	Fermer l'onglet ne ferme pas la session : la tentative suivante est refusée.	Terminer la session dans Paramètres → Sessions actives , puis prendre l'habitude de se déconnecter proprement (section 13.2).
Restrictions horaires du compte (section 14.6)	La connexion au bastion lui-même est refusée en dehors des créneaux.	Vérifier dates, heures et fuseau horaire dans la fiche de l'utilisateur.

A.5 Réflexion : faut-il désactiver NLA pour que « ça marche » ?

Lorsque l'authentification NLA échoue, la tentation est grande de décocher l'option NLA sur le serveur Windows (**Propriétés système → Utilisation à distance**). Souvent, la connexion passe alors... mais le problème n'est pas résolu, il est **masqué**, et le serveur est affaibli.

Ce que NLA protège

Avec NLA, Windows vérifie l'identité **avant** de créer quoi que ce soit. Sans NLA, toute machine qui joint le port 3389 obtient l'écran de connexion Windows, c'est-à-dire une **vraie session graphique créée sans aucune authentification**.

Risque sans NLA	Explication
Surface d'attaque avant authentification	Les failles RDP les plus graves (par exemple BlueKeep , CVE-2019-0708, exécution de code à distance) s'exploitent à ce stade, avant toute authentification. NLA bloque leur exploitation par un attaquant non authentifié.
Déni de service	Chaque tentative de connexion consomme une session et de la mémoire sur le serveur.
Force brute	Les mots de passe sont testés directement sur l'écran de connexion du serveur.
Fuite d'informations	L'écran de connexion peut révéler le nom du domaine, le nom du serveur ou le dernier utilisateur connecté.

Utiliser la désactivation comme test de diagnostic

Désactiver NLA quelques minutes, sur une machine de test, est en revanche un bon outil de diagnostic :

Résultat du test	Conclusion
La connexion pass e sans NLA	Le réseau, le port et le certificat sont hors de cause : le blocage vient de l' authentification NLA (A.3 : identifiants, domaine, compte verrouillé, NTLM).
La connexion échoue toujours	NLA n'est pas en cause : chercher ailleurs (journal guacd, réseau, certificat).

Dans les deux cas, **on réactive NLA** immédiatement après le test, puis on corrige la vraie cause.

Conduire le test proprement

- Uniquement sur une **VM de test**, jamais sur un contrôleur de domaine ou un serveur de production.
- Dans Guacamole, choisir le mode de sécurité **TLS**, jamais « RDP » (ancien chiffrement RDP, faible).
- Noter l'heure du test et le résultat ; **réactiver NLA** dès la fin.

Important côté cyber — si un serveur doit rester sans NLA

Cela ne peut être qu'une **exception documentée et temporaire**, compensée par :

- un pare-feu Windows qui n'ouvre le port 3389 **qu'à l'adresse du bastion** ;
- le mode de sécurité **TLS** côté Guacamole ;
- une **stratégie de verrouillage** de compte dans l'Active Directory ;
- la GPO « **Ne pas afficher le dernier nom d'utilisateur** » ;
- une date de fin prévue pour la remise en place de NLA.

Le bastion réduit l'exposition (en principe, lui seul joint le port 3389), mais dans un réseau pédagogique d'autres postes peuvent souvent atteindre ce port : **il faut le vérifier**, pas le supposer.

Question de réflexion

Un technicien désactive NLA sur un serveur parce que « ça ne marchait pas avec le bastion », puis oublie de le réactiver. Six mois plus tard, une nouvelle faille RDP exploitable **avant authentification** est publiée. Quels serveurs de l'entreprise sont exposés ? Comment aurait-on pu détecter cette configuration ? Quelle procédure aurait évité la situation ?

A.6 Le serveur n'est pas joignable

Depuis le bastion, tester le port RDP du serveur (remplacer l'adresse) :

```
timeout 3 bash -c '</dev/tcp/192.168.100.12/3389' && echo "port 3389 ouvert"
```

Résultat	Où chercher
« port 3389 ouvert »	Le réseau fonctionne : revenir au journal de guacd (A.2).

Résultat	Où chercher
Pas de réponse (délai)	Un pare-feu bloque : pare-feu Windows du serveur, règles du pare-feu entre la DMZ et le réseau des serveurs (pfSense), routage.
« Connexion refusée » immédiate	Le serveur répond mais le Bureau à distance n'écoute pas : vérifier qu'il est activé, et que le service « Services Bureau à distance » est démarré.

§ Le pare-feu du bastion n'est pas en cause

Les règles nftables du bastion (section 9C) ne filtrent que les connexions **entrantes**. Les connexions que guacd ouvre vers les serveurs, et leurs réponses, sont autorisées.

A.7 Fiche de suivi d'incident

À compléter pour chaque problème : elle sert à ne pas tourner en rond, et à **documenter** la solution pour les suivants.

Rubrique	À compléter
Date, heure, connexion concernée	
Message affiché par Guacamole	
Extrait du journal guacd	
Toutes les connexions ou une seule ?	
Ce qui a changé récemment	
Hypothèses testées (une à la fois) et résultats	
Cause identifiée	
Correction appliquée	

Rubrique	À compléter
Réglage de sécurité modifié temporairement ? Remis en place le :	