

# SRV-SUPERVISION

Libre NMS

Noah MAILLET

Projet-SANDBOX





# Table of Contents

1	Preface .....	3
1.1	Presentation Technologie – Libre NMS.....	3
1.2	Configuration Minimale. ....	4
2	Création de la machine virtuelle. ....	5
3	Installation de rocky 8.9 .....	8
3.1	Localisation. ....	8
3.2	Paramètres utilisateurs.....	9
3.3	Logiciel. ....	9
3.4	Système.....	10
4	Configuration Rocky Linux 8.9.....	11
4.1	Configuration connexion guacamole.....	12
4.1.1	Nouvelle connexions. ....	12
4.1.2	Paramètres. ....	13
4.1.3	Test de la connexion. ....	14
5	Installation LibreNMS. ....	15
5.1	Installation paquet requis.....	15
5.2	Add librenms user.....	15
5.3	Téléchargement LibreNMS.....	16
5.4	Configuration permissions. ....	16
5.5	Installation dépendances PHP.....	17
5.6	Configuration Time-Zone.....	18
5.7	Configuration MariaDB. ....	19
5.8	Configuration PHP-FPM. ....	20



# 1 Preface

## 1.1 Présentation Technologie – Libre NMS.

LibreNMS est une plateforme de supervision réseau open source et communautaire conçue pour surveiller et gérer les réseaux informatiques de toutes tailles. En tant que solution flexible et évolutive, LibreNMS offre une gamme étendue de fonctionnalités de surveillance, de gestion des périphériques et de génération de rapports, le tout dans une interface utilisateur intuitive.

Caractéristiques principales :

1. **Surveillance automatisée** : LibreNMS permet la surveillance continue des dispositifs réseau, y compris les commutateurs, les routeurs, les serveurs, les équipements sans fil, etc. Il offre une surveillance en temps réel des performances, de la disponibilité et de la santé des périphériques.
2. **Découverte automatique** : La fonction de découverte automatique identifie automatiquement les périphériques réseau et leurs interfaces, réduisant ainsi le temps nécessaire à la configuration initiale.
3. **Notifications et Alertes** : LibreNMS génère des notifications en temps réel sur les problèmes de performances ou de disponibilité des périphériques via divers canaux tels que l'email, SMS, Slack, etc., permettant une réactivité immédiate aux incidents.
4. **Gestion des Périphériques** : La plateforme offre des fonctionnalités de gestion des périphériques telles que la visualisation de l'état, la configuration des seuils de performance, la journalisation des événements, et la gestion des mises à jour du firmware.
5. **Collecte de Données** : LibreNMS collecte une variété de données de performance, y compris les statistiques sur le trafic réseau, l'utilisation du CPU, la mémoire, et d'autres métriques pertinentes pour une analyse approfondie.
6. **Personnalisation** : Les utilisateurs peuvent personnaliser l'interface, les tableaux de bord et les rapports en fonction de leurs besoins spécifiques, permettant une expérience utilisateur optimisée.
7. **Intégrations** : LibreNMS s'intègre avec d'autres outils de surveillance, de notification et de gestion de systèmes, offrant une connectivité transparente avec l'écosystème existant.

Architecture technique :

- LibreNMS est basé sur une architecture moderne et extensible, comprenant les éléments suivants :
- **Base de Données** : Utilise une base de données MySQL ou MariaDB pour stocker les données de surveillance, assurant une haute disponibilité et une performance optimale.
- **Collecteurs** : Des collecteurs pollent régulièrement les périphériques pour collecter des données de performance, qui sont ensuite stockées dans la base de données.
- **Interface Web** : Fournit une interface web conviviale pour la configuration, la visualisation et l'analyse des données de surveillance.
- **Modules d'Extension** : Permettent l'ajout de fonctionnalités supplémentaires, de plug-ins et d'intégrations avec des systèmes tiers.
- LibreNMS est une solution de supervision réseau puissante, flexible et open source, adaptée aux besoins des petites et grandes entreprises. Grâce à ses fonctionnalités avancées, son architecture extensible et sa communauté active, LibreNMS offre une visibilité complète sur les réseaux informatiques, permettant une gestion proactive et efficace des infrastructures réseau.



## 1.2 Configuration minimale.

La configuration minimale requise pour exécuter LibreNMS est relativement modeste. Voici les spécifications minimales recommandées :

### Matériel :

- Processeur : 1 GHz CPU (multi-core recommandé pour une meilleure performance)
- Mémoire : 2 Go de RAM (4 Go ou plus recommandés pour des réseaux plus importants)
- Espace disque : 20 Go d'espace disque disponible pour les données et les logs (l'espace requis dépend de la taille du réseau surveillé et de la rétention des données)

### Logiciel :

- Système d'exploitation : Linux (Ubuntu, Debian, CentOS, etc.) ou FreeBSD sont les plus couramment utilisés
- Serveur Web : Apache ou Nginx
- Base de données : MySQL (ou MariaDB) version 5.7 ou supérieure
- PHP : PHP 8.1 ou supérieur avec les extensions requises (gd, mbstring, mysql, xml, json, etc.)

### Réseau :

- Accès Internet pour les mises à jour automatiques et les téléchargements de paquets
- Ports réseau ouverts pour la communication avec les périphériques surveillés (par exemple, SNMP)

Il est important de noter que ces spécifications sont une recommandation minimale. Pour des réseaux plus importants ou pour une utilisation plus intensive, il est conseillé d'augmenter les ressources matérielles (CPU, RAM, espace disque) en conséquence.

De plus, la configuration et l'optimisation du serveur, du système d'exploitation et des logiciels dépendront également des besoins spécifiques de votre déploiement LibreNMS, notamment en ce qui concerne la taille du réseau, la charge de surveillance et les fonctionnalités utilisées.

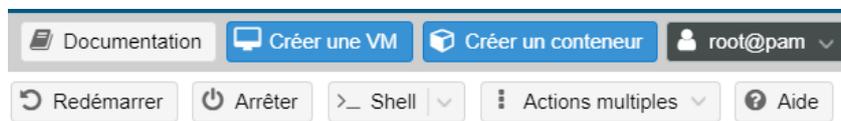


## 2 Création de la machine virtuelle.

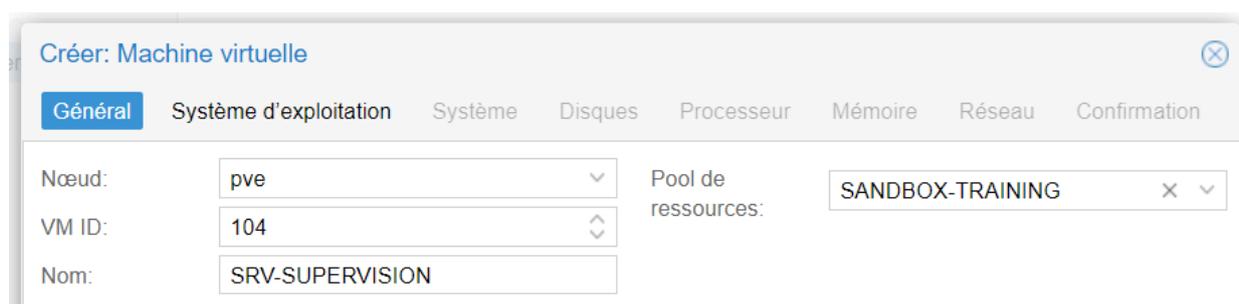
Pour ce projet j'utilise une ferme de serveur proxmox.

Vous pouvez utiliser tout type d'hyperviseur tant que vous respectez les préconisations matérielles.

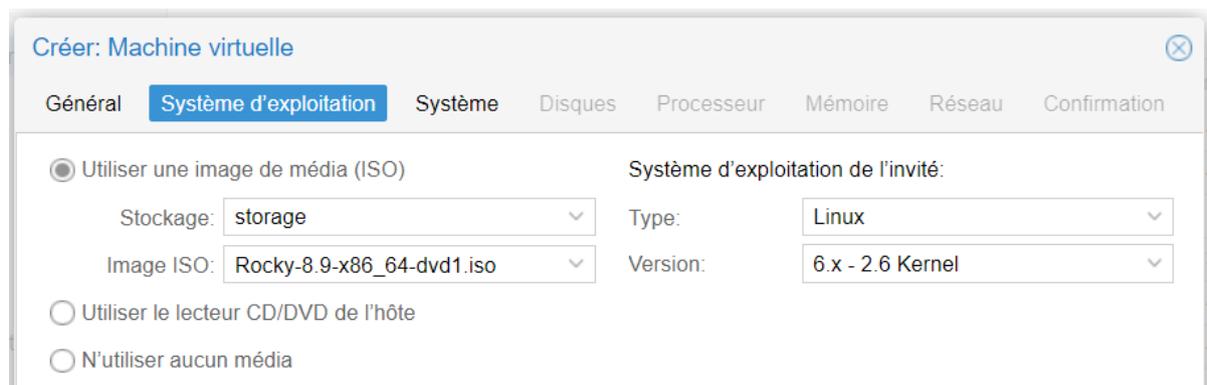
1. Je me connecte à la ferme de serveur.
2. Créer une VM.



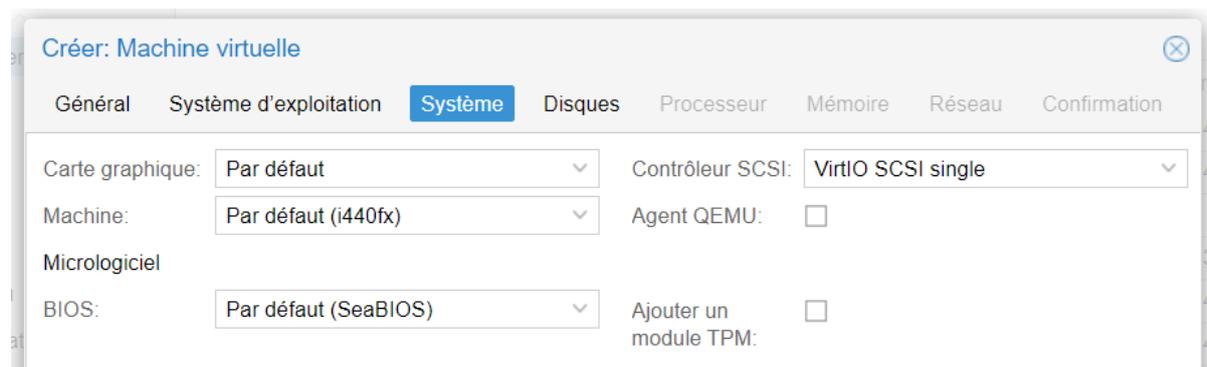
3. Je nomme la machine virtuelle « SRV-SUPERVISION » et je le sélectionne le pool de ressources « SANDBOX-TRAINING » → suivant.



4. Je sélectionne l'iso ici « Rocky-8.9-Dvd1.iso » → suivant.



5. Je laisse les informations par défaut → suivant.





6. J'alloue « 52 Go » de stockage à la machine virtuel → Suivant.

Créer: Machine virtuelle

Général Système d'exploitation Système **Disques** Processeur Mémoire Réseau Confirmation

scsi0  Disque Bande passante

Bus/périphérique: SCSI 0 Cache: Par défaut (Aucun ca)

Contrôleur SCSI: VirtIO SCSI single Abandonner:

Stockage: local-lvm IO thread:

Taille du disque (Gio): 52

Format: Image disque brute (r)

7. J'alloue « 2 cœurs » de processeur et je sélectionne le type « host » → Suivant.

Créer: Machine virtuelle

Général Système d'exploitation Système Disques **Processeur** Mémoire Réseau Confirmation

Supports de processeur: 1 Type: host

Cœurs: 2 Total de cœurs: 2

8. J'alloue « 2 Go » de ram → Suivant.

Créer: Machine virtuelle

Général Système d'exploitation Système Disques Processeur **Mémoire** Réseau Confirmation

Mémoire (MiB): 2048

9. Je choisis « Intel E1000 » en modèle, je désactive le pare-feu et désactive le VLAN → Suivant.

Créer: Machine virtuelle

Général Système d'exploitation Système Disques Processeur Mémoire **Réseau** Confirmation

Aucun périphérique réseau

Pont (bridge): vubr0 Modèle: Intel E1000

Étiquette de VLAN: 1 Adresse MAC: auto

Pare-feu:



10. Je vérifie que les informations sont correctes → terminer.

Créer: Machine virtuelle

Général Système d'exploitation Système Disques Processeur Mémoire Réseau Confirmation

Key ↑	Value
cores	2
cpu	host
ide2	storage.iso/Rocky-8.9-x86_64-dvd1.iso,media=cdrom
memory	2048
name	SRV-SUPERVISION
net0	e1000,bridge=vibr0,tag=1
nodename	pve
numa	0
ostype	l26
pool	SANDBOX-TRAINING
scsi0	local-lvm:52,iotread=on
scsihw	virtio-scsi-single
sockets	1
vmid	104

Démarrer après création

Avancé  Retour Terminer

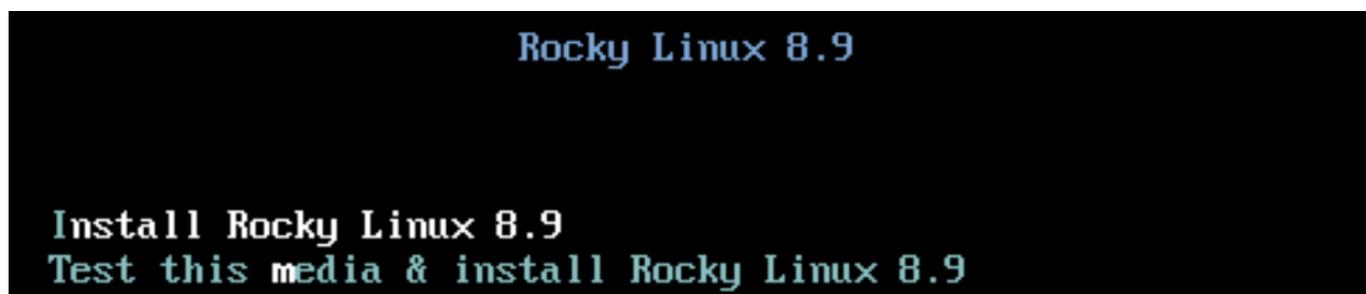
11. La machine virtuelle à bien été créer :

- ✓ pve
  - 101 (SRV-LLDAP)
  - 102 (SRV-DNS)
  - [Redacted]
  - 100 (BigFirewall)
  - 103 (SRV-GUACAMOLE)
  - 104 (SRV-SUPERVISION)
  - 110 (HOST-ADMIN)

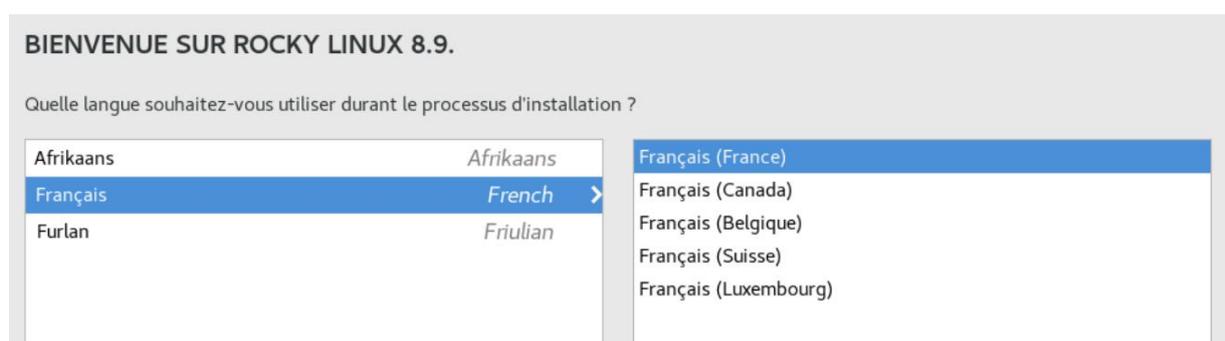


## 3 Installation de rocky 8.9

1. Je me connecte à la ferme de serveur.
2. Je démarre la machine virtuelle.
3. Je choisis « Install Rocky Linux 8.9 ».

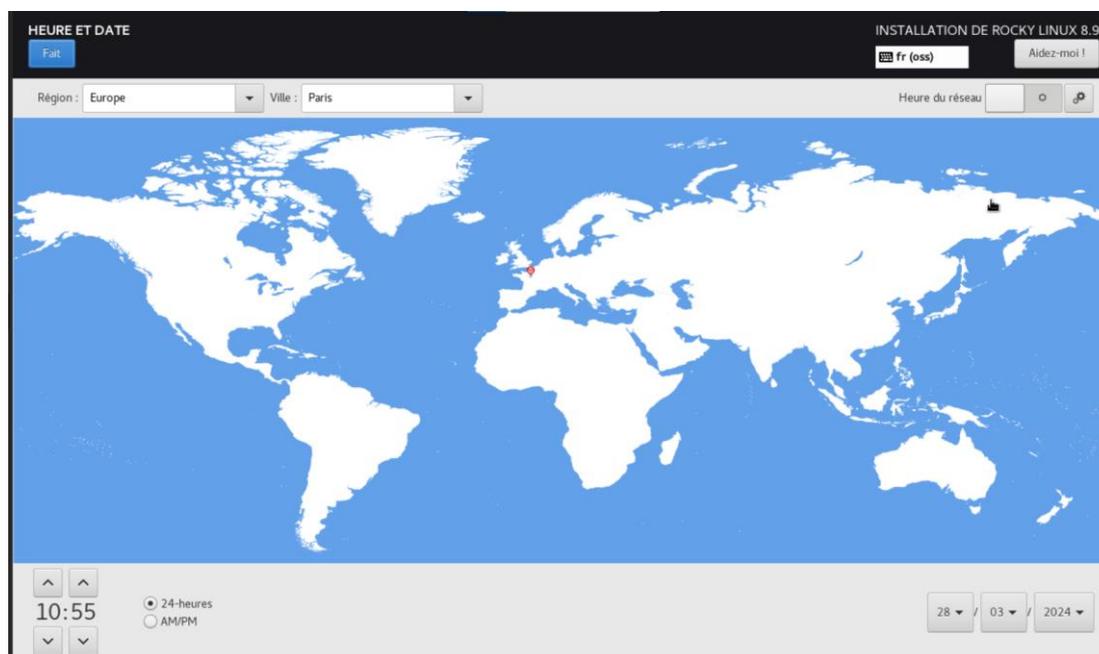


4. Je choisis la langue française → continuez.



### 3.1 Localisation.

5. Heure et date → Je choisis Europe ; Paris. → Fait.





## 3.2 Paramètres utilisateurs

- Je définis le mot de passe administrateur → Fait.

Le compte root est utilisé pour administrer le système. Entrez un mot de passe pour l'utilisateur root.

Mot de passe administrateur :

Confirmé :

Faible

- Création utilisateur → Je rentre les informations de l'utilisateur → Fait.

Nom et prénom

Nom d'utilisateur

Faire de cet utilisateur un administrateur

Exiger un mot de passe pour utiliser ce compte

Mot de passe

Confirmé le mot de passe

Faible

Avancé...

## 3.3 Logiciel.

- Sélection de logiciels.
  - Je sélectionne serveur.
  - Je choisis les paquets suivants :
    - Utilitaire de surveillance du matériel,
    - Outils système.

SÉLECTION DE LOGICIELS

INSTALLATION DE ROCKY LINUX 8.9

Fait

fr (oss)

Aidez-moi !

Environnement de base

- Serveur avec GUI  
Un serveur intégré, facile à gérer, avec une interface graphique.
- Serveurs**  
Un serveur intégré, facile à gérer.
- Installation minimale  
Fonctionnalité de base.
- Station de travail  
Une station de travail est un système de bureau convivial pour les ordinateurs portables et les PC.
- Custom Operating System  
Basic building block for a custom Rocky Linux system.
- Hôte de virtualisation  
Hôte de virtualisation minimal.

Logiciel supplémentaire pour l'environnement sélectionné

- Utilitaires de surveillance du matériel**  
Ensemble d'outils pour surveiller le matériel du serveur.
- Serveur de fichiers Windows  
Ce groupe de packages vous permet de partager des fichiers entre les systèmes Linux et Windows (tm).
- Outils de débogage  
Outils pour déboguer les applications ayant un mauvais comportement et diagnostiquer les problèmes de performance.
- Serveur de nom DNS  
Ce groupe de packages vous permet d'exécuter un serveur de noms DNS (BIND) sur le système.
- Serveur de fichiers et de stockage  
Serveur de stockage réseau CIFS, SMB, NFS, iSCSI, iSER et iSNS.
- Serveur FTP  
Ces outils vous permettent d'exécuter un serveur FTP sur le système.



## 3.4 Système.

9. Installation Destination → je sélectionne le disque → fais.

**Sélection des périphériques**  
Sélectionnez le périphérique sur lequel vous souhaitez faire l'installation. Il restera intact jusqu'à ce que vous cliquiez sur le bouton « Commencer l'installation » du menu principal.

Disques locaux standards

52 Gio



**QEMU QEMU HARDDISK**  
sda / 52 Gio d'espace libre

*Les disques décochés ne seront pas modifiés.*

Disques spéciaux et réseau

 Ajouter un disque...

*Les disques décochés ne seront pas modifiés.*

**Configuration du stockage**

Automatique  Personnalisé

Je voudrais libérer plus d'espace.

**Chiffrement**

Chiffrer mes données. Vous définirez une phrase de passe plus tard.

10. Réseau et Nom d'hôte → je rentre le nom de la machine virtuelle, je rentre les informations du réseau → fais.

**RÉSEAU ET NOM D'HÔTE** INSTALLATION DE ROCKY LINUX 8.9

 fr (oss)

**Ethernet (ens18)**  
Intel Corporation 82540EM Gigabit Ethernet Controller (QEMU Virtual Machine)

 **Ethernet (ens18)**  Connecté

Adresse matérielle: BC:24:11:51:E3:C9

Vitesse: 1000 Mb/s

Adresse IP: 10.16.1.4/28

Route par défaut: 10.16.1.14

DNS: 10.16.1.2  
192.168.1.254  
1.1.1.1

Nom d'hôte :   Nom d'hôte actuel : srv-supervision.sandbox.local

11. Commencer l'installation.



## 4 Configuration Rocky Linux 8.9

1. Je me connecte à la machine virtuelle.
2. Mise à jour des dépôts.

Commandes :

```
Dnf update
```

```
Résumé de la transaction
-----
Installer      4 Paquets
Mettre à niveau 42 Paquets

Taille totale des téléchargements : 152 M
Voulez-vous continuer ? [o/N] : o
```

3. Redémarrage de Rocky.

Commandes :

```
Reboot
```

4. J'installe le serveur SSH.

Commandes :

```
Dnf install openssh-server
```

```
[root@srv-supervision ~]# dnf install openssh-server
Dernière vérification de l'expiration des métadonnées effectuée il y a 0:07:19 le lun. 15 avril 2024 05:27:32 EDT.
Le paquet openssh-server-8.0p1-19.el8_9.2.x86_64 est déjà installé.
Dépendances résolues.
Rien à faire.
Terminé !
```



## 4.1 Configuration connexion guacamole.

1. Je me connecte sur le serveur guacamole avec un compte administrateur.
2. Je me rends dans paramètres → Connexions → je déplie le dossier « administration » → nouvelle connexion.

**PARAMÈTRES**

Sessions Actives   Historique   Utilisateurs   Groupes   **Connexions**   Préférences

Cliquer ou appuyer sur une connexion en dessous pour la gérer. Selon vos permissions, les connexions peuvent être supprimées.

- Administration
  - HOST-ADMIN
  - SRV-DNS
  - SRV-GUACAMOLE
  - SRV-LLDAP
  - SRV-SUPERVISION
  - Nouvelle Connexion
  - Nouveau Groupe

### 4.1.1 Nouvelles connexions.

3. Je rentre le nom du serveur et je choisis le protocole.

Nom:

Lieu:

Protocole:  ▼



## 4.1.2 Paramètres.

1. Réseau – Je rentre le nom d'hôte, le port.

### Réseau

Nom d'hôte:	<input type="text" value="srv-supervision.sandbox.loc"/>
Port:	<input type="text" value="22"/>
Clé publique de l'hôte (Base64):	<input type="text"/>

2. Authentification – Je rentre l'Identifiant et le mot de passe.

### Authentification

Identifiant:	<input type="text" value="sandbox"/>
Mot de passe:	<input type="password" value="....."/>
Clé privée:	<input type="text"/>
Phrase secrète:	<input type="password"/>

3. Affichage – Je sélectionne la palette de couleur.

### Affichage

Palette de couleurs:	<input type="text" value="Vert sur noir"/>
Nom police:	<input type="text"/>
Taille police:	<input type="text" value="v"/>
Taille maximum du défilement arrière:	<input type="text"/>
Lecture seule:	<input type="checkbox"/>

4. Session / Environnement – Je rentre la version du clavier et le fuseau horaire.

### Session / Environnement

Exécuter une commande:	<input type="text"/>
Langue/Locale (\$LANG):	<input type="text" value="fr"/>
Fuseau horaire (\$TZ):	<input type="text" value="Europe"/> <input type="text" value="Paris"/>
Intervalle keepalive Serveur:	<input type="text"/>



5. Enregistrement Écran – Je renseigne le chemin de l'enregistrement et le nom de l'enregistrement, je coche la case Créer automatiquement le chemin d'enregistrement.

#### Enregistrement Ecran

Chemin de l'enregistrement:	<input type="text" value="{HISTORY_PATH}/{HIST"/>
Nom de l'enregistrement:	<input type="text" value="{GUAC_DATE}-{GUAC_"/>
Exclure les graphiques/flux:	<input type="checkbox"/>
Exclure la souris:	<input type="checkbox"/>
Inclure les événements clavier:	<input type="checkbox"/>
Créer automatiquement le chemin d'enregistrement:	<input checked="" type="checkbox"/>

6. Enregistrer.

### 4.1.3 Test de la connexion.

1. Je me connecte au serveur guacamole.
2. Je déplie le dossier administration → SRV-SUPERVISION.
3. La connexion est fonctionnelle.

```
[sandbox@srv-supervision ~]$ dig www.google.fr
; <<>> DiG 9.11.36-RedHat-9.11.36-11.el8_9.1 <<>> www.google.fr
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 7116
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1280
;; QUESTION SECTION:
;www.google.fr.                IN      A
;; ANSWER SECTION:
www.google.fr.                72     IN      A      142.250.201.163
;; Query time: 10 msec
;; SERVER: 192.168.1.254#53(192.168.1.254)
;; WHEN: Mon Apr 15 05:52:19 EDT 2024
;; MSG SIZE rcvd: 58
[sandbox@srv-supervision ~]$
```



## 5 Installation LibreNMS.

1. Je me connecte sur le Serveur de supervision.
2. Je me connecte en super utilisateur.

Commande :

```
su -
```

```
[sandbox@srv-supervision ~]$ su -  
Password:  
[root@srv-supervision ~]# █
```

### 5.1 Installation paquet requis.

Installation des dépendances requise pour le bon fonctionnement de librenms.

Commande :

```
dnf install epel-release  
dnf install dnf-utils http://rpms.remirepo.net/enterprise/remi-release-8.rpm  
dnf module reset php  
dnf module install php:remi-8.1  
dnf module enable phpremi-:8.1  
dnf install bash-completion cronie fping git ImageMagick mariadb-server mtr net-snmp net-  
snmp-utils nginx nmap php-fpm php-cli php-common php-curl php-gd php-gmp php-json php-  
mbstring php-process php-snmp php-xml php-zip php-mysqlnd python3 python3-PyMySQL python3-  
redis python3-memcached python3-pip python3-systemd gcc rrdtool unzip
```

### 5.2 Add librenms user.

Ajout d'un utilisateur de service pour Librenms.

Commandes :

```
useradd librenms -d /opt/librenms -M -r -s "$(which bash)"
```

```
[root@srv-supervision ~]# useradd librenms -d /opt/librenms -M -r -s "$(which bash)"  
[root@srv-supervision ~]# █
```



## 5.3 Téléchargement LibreNMS.

Téléchargement de LibreNMS.

Commandes :

```
cd /opt  
git clone https://github.com/librenms/librenms.git
```

```
[root@srv-supervision ~]# cd /opt  
[root@srv-supervision opt]# git clone https://github.com/librenms/librenms.git  
Clonage dans 'librenms'...  
remote: Enumerating objects: 214382, done.  
remote: Counting objects: 100% (7771/7771), done.  
remote: Compressing objects: 100% (769/769), done.  
remote: Total 214382 (delta 7378), reused 7135 (delta 7002), pack-reused 206611  
Réception d'objets: 100% (214382/214382), 288.46 Mio | 9.34 Mio/s, fait.  
Résolution des deltas: 100% (161348/161348), fait.  
Mise à jour des fichiers: 100% (14134/14134), fait.  
[root@srv-supervision opt]# █
```

## 5.4 Configuration permissions.

Configuration des permissions.

Commandes :

```
# changement de propriétaire du fichier « /opt/librenms ».  
chown -R librenms:librenms /opt/librenms  
  
# changement des droit du fichier « /opt/librenms ».  
chmod 771 /opt/librenms  
  
# Création d'ACL pour les répertoire spcifié.  
setfacl -d -m g::rwx /opt/librenms/rrd /opt/librenms/logs /opt/librenms/bootstrap/cache/  
/opt/librenms/storage/  
setfacl -R -m g::rwx /opt/librenms/rrd /opt/librenms/logs /opt/librenms/bootstrap/cache/  
/opt/librenms/storage/
```

```
[root@srv-supervision opt]# chown -R librenms:librenms /opt/librenms/  
[root@srv-supervision opt]# chmod 771 /opt/librenms/  
[root@srv-supervision opt]# setfacl -d -m g::rwx /opt/librenms/rrd /opt/librenms/logs /opt/librenms/bootstrap/cache/ /opt/librenms/storage/  
[root@srv-supervision opt]# setfacl -R -m g::rwx /opt/librenms/rrd /opt/librenms/logs /opt/librenms/bootstrap/cache/ /opt/librenms/storage/  
[root@srv-supervision opt]# █
```



## 5.5 Installation dépendances PHP.

Commande :

```
su - librenms
```

```
./scripts/composer_wrapper.php install --no-dev
```

Exit

```
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

  INFO  Discovering packages.

laravel-notification-channels/webpush ..... DONE
laravel/socialite ..... DONE
laravel/tinker ..... DONE
laravel/ui ..... DONE
librenms/laravel-vue-i18n-generator ..... DONE
mews/purifier ..... DONE
nesbot/carbon ..... DONE
nunomaduro/termwind ..... DONE
php-flasher/flasher-laravel ..... DONE
silber/bouncer ..... DONE
spatie/laravel-ignition ..... DONE
tightenco/ziggy ..... DONE

79 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> LibreNMS\ComposerHelper::postInstall
> Illuminate\Foundation\ComposerScripts::postInstall
> @php artisan vue-i18n:generate --multi-locales --format=umd
> @php artisan view:cache

  INFO  Blade templates cached successfully.

> @php artisan optimize

  INFO  Caching the framework bootstrap files.

config ..... 25ms DONE
routes ..... 54ms DONE

> @php artisan config:clear
```



## 5.6 Configuration Time-Zone.

Configuration du fuseau horaire.

Commande :

```
nanny /etc/php.ini
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;  
; Module Settings ;  
;;;;;;;;;;;;;;;;;;;;;;;;;;  
  
[CLI Server]  
; Whether the CLI web server uses ANSI color coding in its terminal output.  
cli_server.color = 0n  
  
[Date]  
; Defines the default timezone used by the date functions  
; https://php.net/date.timezone  
date.timezone = "Europe/Paris"
```

Commande :

```
timedatectl set-timezone Europe/Paris
```

```
[root@srv-supervision ~]# timedatectl set-timezone Europe/Paris  
[root@srv-supervision ~]# date  
lun. avril 15 13:55:41 CEST 2024  
[root@srv-supervision ~]#
```



## 5.7 Configuration MariaDB.

Configuration MariaDB.

Commande :

```
nanny /etc/my.cnf.d/mariadb-server.cnf
```

Modification du fichier :

```
[mysqld]
```

```
innodb_file_per_table=1
```

```
lower_case_table_names=0
```

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
log-error=/var/log/mariadb/mariadb.log
pid-file=/run/mariadb/mariadb.pid
innodb_file_per_table=1
lower_case_table_names=0
```

Redémarrage et activation de mariadb :

Commandes :

```
Systemctl enable mariadb
```

```
Systemctl restart mariadb
```

```
[root@srv-supervision ~]# systemctl enable mariadb
Created symlink /etc/systemd/system/mysql.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/mysqld.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /usr/lib/systemd/system/mariadb.service.
[root@srv-supervision ~]# systemctl restart mariadb
[root@srv-supervision ~]#
```

Création de la base de données :

Commande :

```
mysql -u root
```

```
CREATE DATABASE librenms CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```

```
CREATE USER 'librenms'@'localhost' IDENTIFIED BY '';
```

```
GRANT ALL PRIVILEGES ON librenms.* TO 'librenms'@'localhost';
```

```
Exit
```

```
[root@srv-supervision ~]# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.3.39-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE librenms CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
Query OK, 1 row affected (0,000 sec)

MariaDB [(none)]> CREATE USER 'librenms'@'localhost' IDENTIFIED BY '' ;
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON librenms.* TO 'librenms'@'localhost';
Query OK, 0 rows affected (0,000 sec)

MariaDB [(none)]> exit;
Bye
[root@srv-supervision ~]#
```



## 5.8 Configuration PHP-FPM.

Configuration de PHP-FPM.

Commandes :

```
cp /etc/php-fpm.d/www.conf /etc/php-fpm.d/librenms.conf
vi /etc/php-fpm.d/librenms.conf
```

Modification à effectuer :

# Modifier [www] → [librenms]

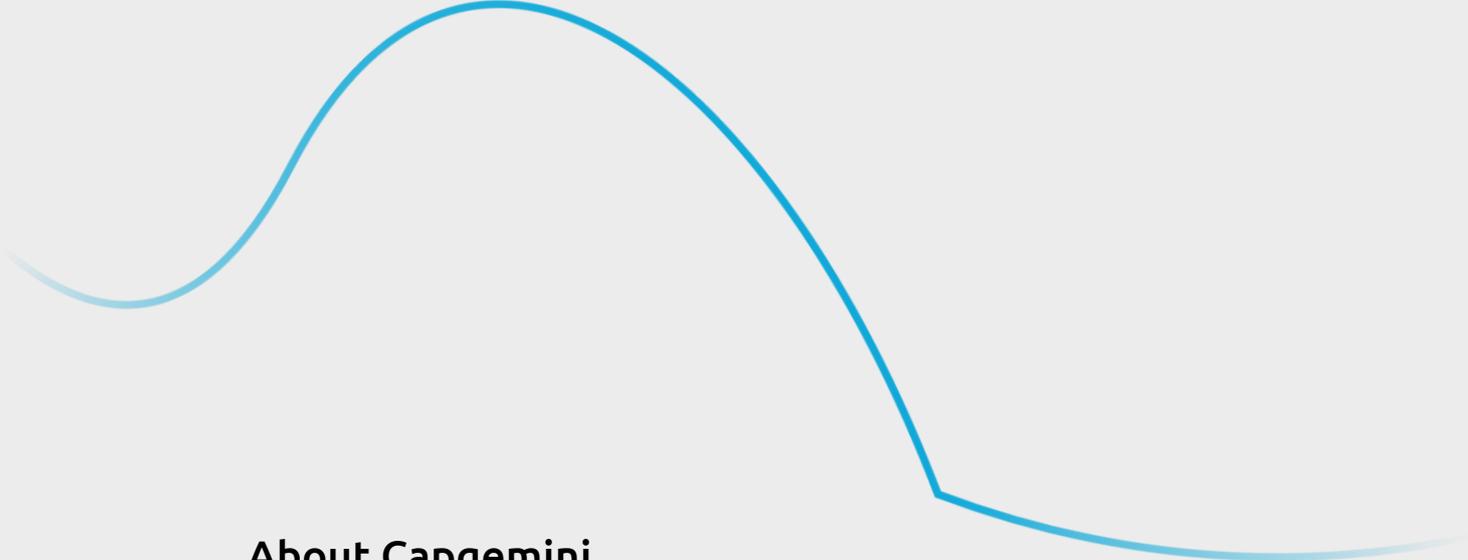
```
; Start a new pool named 'www'.
; the variable $pool can be used in any directive and will be replaced by the
; pool name ('www' here)
[librenms]
```

# Changer user et group → libre nms.

```
; Unix user/group of processes
; Note: The user is mandatory. If the group is not set, the default user's group
; will be used.
; RPM: apache user chosen to provide access to the same directories as httpd
user = librenms
; RPM: Keep a group allowed to write in log dir.
group = librenms
```

# Modification → listen = /run/php-fpm-librenms.sock

```
; The address on which to accept FastCGI requests.
; Valid syntaxes are:
; 'ip.add.re.ss:port'    - to listen on a TCP socket to a specific IPv4 address on
;                        a specific port;
; '[ip:6:addr:ess]:port' - to listen on a TCP socket to a specific IPv6 address on
;                        a specific port;
; 'port'                - to listen on a TCP socket to all addresses
;                        (IPv6 and IPv4-mapped) on a specific port;
; '/path/to/unix/socket' - to listen on a unix socket.
; Note: This value is mandatory.
listen = /run/php-fpm-librenms.sock
```

A large, light blue wave graphic that starts on the left, rises to a peak, and then descends towards the right, ending in a small curve.

## About Capgemini

Capgemini is a global leader in partnering with companies to transform and manage their business by harnessing the power of technology. The Group is guided everyday by its purpose of unleashing human energy through technology for an inclusive and sustainable future. It is a responsible and diverse organization of over 360,000 team members in more than 50 countries. With its strong 55-year heritage and deep industry expertise, Capgemini is trusted by its clients to address the entire breadth of their business needs, from strategy and design to operations, fueled by the fast evolving and innovative world of cloud, data, AI, connectivity, software, digital engineering and platforms. The Group reported in 2022 global revenues of €22 billion.

Get the Future You Want | [www.capgemini.com](http://www.capgemini.com)



This document contains information that may be privileged or confidential and is the property of the Capgemini Group.

**Company Confidential.** Copyright © 2023 Capgemini. All rights reserved.