

Documentation SP2

Introduction

Nous allons installer un serveur Web qui aura pour IP: 10.31.252.88, ce serveur va nous permettre d'héberger un GLPI qui va nous servir à avoir une vue de la gestion de notre parc informatique grâce à un agent qui sera installé sur les machines clientes. Ce qui nous permettra de voir l'état de notre parc informatique machine par machine. De plus elle va nous permettre d'avoir un service de ticketing qui est très utile en entreprise pour tout ce qui est résolution de problèmes informatiques.

Avantage de GLPI:

- * Service de ticketing.
- * Supervision de machine
- * Peut gérer des informations de tout type tel que financier ou administratif

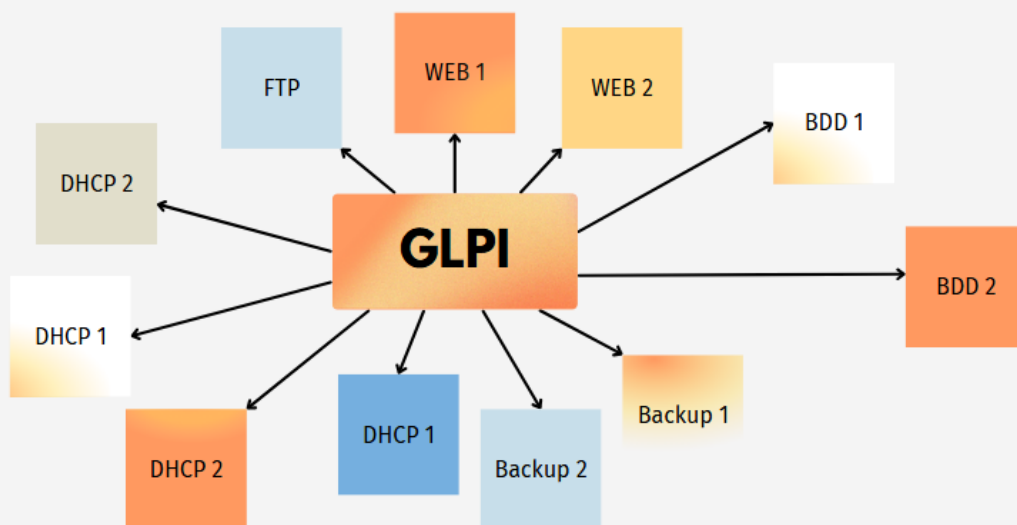
cahier des charges

Création d'un serveur WEB + BDD
Installation d'agent sur toutes les machines de notre réseau
Documentation à fournir

Schema réseau GLPI

Vlan SRV GLPI

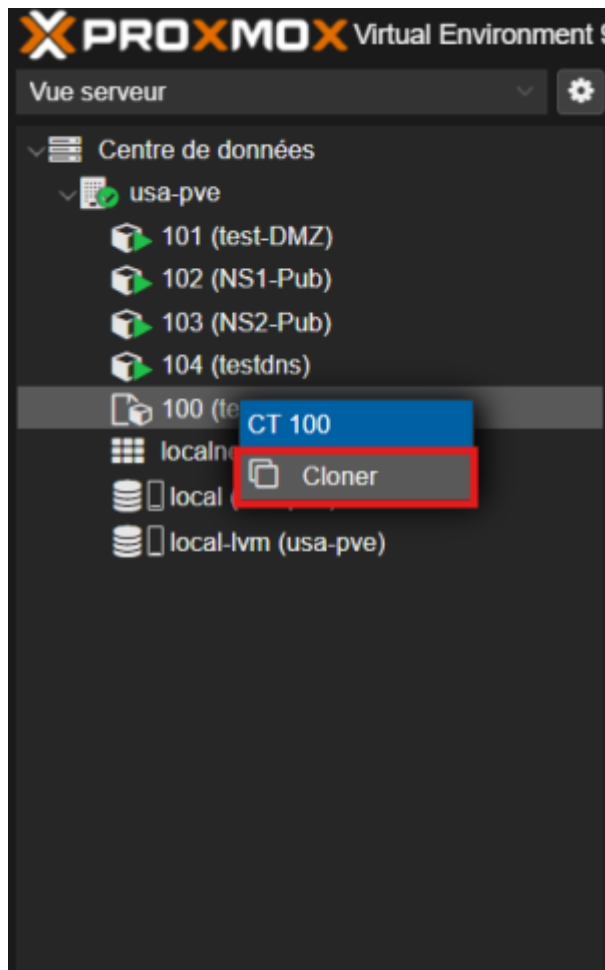
Glpi est un logiciel web qui permet de gérer un parc informatique d'entreprise.



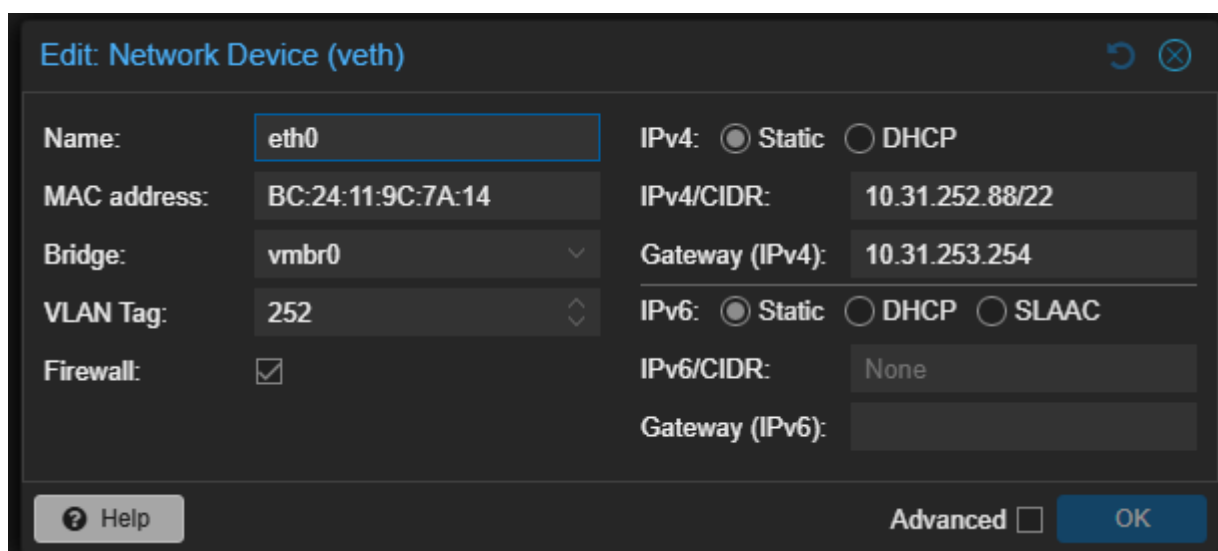
Serveur WEB

Configuration du serveur WEB

Pour installer notre GLPI il va falloir l'héberger sur un serveur Web c'est pour cela que nous allons en faire un a partir de notre template sur proxmox



Une fois le template copier nous allons configurer son adresse IP ainsi que son vlan et tout ses configuration réseau



configuration et installation de GLPI

Installation

Nous allons commencer par installer Apache2 :

```
Microsoft Windows [version 10.0.26200.7922]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\gauti>ssh root@10.31.252.88
Linux GLPI 6.14.11-1-pve #1 SMP PREEMPT_DYNAMIC PMX 6.14.11-1 (2025-
The programs included with the Debian GNU/Linux system are free soft
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Mar  2 07:15:23 2026 from 10.187.20.227
root@GLPI:~# sudo apt-get install apache2 php8.4-fpm mariadb-server
```

```
root@GLPI:~# sudo apt-get install apache2 php8.4-fpm mariadb-server
```

cette ligne va nous permettre d'installer un apache2 ainsi que mariaDB et php8.4.
Une fois cela fait on va installer tout les paquets necessaire au fonctionnement de notre GLPI

```
sudo apt install php8.4-{curl,gd,intl,mysql,zip,bcmath,mbstring,xml,bz2}
sudo apt install php8.4-ldap
```

Installation de MariaDB

Une fois avoir installé apache2 on va installer la base de donnée qui sera une base MariaDB

```
sudo mariadb-secure-installation
```

```
Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n ←
... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y ←
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y ←
SQL executed without errors!
The operation might have been successful, or it might have not done anything.

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y ←
SQL executed without errors!
The operation might have been successful, or it might have not done anything.

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y ←
- Dropping test database ...
SQL executed without errors!
The operation might have been successful, or it might have not done anything.
- Removing privileges on test database ...
SQL executed without errors!
The operation might have been successful, or it might have not done anything.

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y ←
... Success!

Cleaning up ...

All done! If you've completed all of the above steps, your MariaDB
```

Après avoir installé mariaDB nous allons nous y connecter pour créer notre utilisateur

```
sudo mysql -u root -p
CREATE DATABASE User_glpi;
GRANT ALL PRIVILEGES ON User_glpi.* TO glpi_adm@localhost IDENTIFIED BY
"Test";
FLUSH PRIVILEGES;
EXIT
```

Installation de GLPI

Pour la suite nous allons installer GLPI

```
cd /tmp
wget
https://github.com/glpi-project/glpi/releases/download/11.0.4/glpi-11.0.4.tgz
sudo tar -xzvf glpi-11.0.4.tgz -C /var/www/
```

Préparation de l'installation

```
sudo chown www-data /var/www/glpi/ -R
sudo mkdir /etc/glpi
sudo chown www-data /etc/glpi/
sudo mv /var/www/glpi/config /etc/glpi
sudo mkdir /var/lib/glpi
sudo chown www-data /var/lib/glpi/
sudo mv /var/www/glpi/files /var/lib/glpi
sudo mkdir /var/log/glpi
sudo chown www-data /var/log/glpi
```

Nous devons configurer GLPI pour qu'il sache où aller chercher les données. Autrement dit, nous allons déclarer les nouveaux répertoires fraîchement créés. Nous allons créer ce premier fichier :

```
sudo nano /var/www/glpi/inc/downstream.php
<?php
define('GLPI_CONFIG_DIR', '/etc/glpi/');
if (file_exists(GLPI_CONFIG_DIR . '/local_define.php')) {
    require_once GLPI_CONFIG_DIR . '/local_define.php';
}
```

Ensuite, nous allons créer ce second fichier :

```
sudo nano /etc/glpi/local_define.php
<?php
define('GLPI_VAR_DIR', '/var/lib/glpi/files');
define('GLPI_LOG_DIR', '/var/log/glpi');
```

Configuration d'apache pour GLPI

Passons à la configuration du serveur web Apache2. Nous allons créer un nouveau fichier de configuration qui va permettre de configurer le VirtualHost dédié à GLPI. Dans mon cas, le fichier s'appelle support.it-connectlab.fr.conf en référence au nom de domaine choisi pour accéder à GLPI : support.it-connectlab.fr. L'idéal étant d'avoir un nom de domaine (même interne) pour accéder à GLPI afin de pouvoir positionner un certificat TLS/SSL par la suite.

```
sudo nano /etc/apache2/sites-available/support.it-connectlab.fr.conf
```

et voici le Virtualhost :

```
<VirtualHost :80>
    ServerName support.it-connectlab.fr

    DocumentRoot /var/www/glpi/public

    # If you want to place GLPI in a subfolder of your site (e.g. your
    virtual host is serving multiple applications),
    # you can use an Alias directive. If you do this, the DocumentRoot
    directive MUST NOT target the GLPI directory itself.
    # Alias "/glpi" "/var/www/glpi/public"

    <Directory /var/www/glpi/public>
        Require all granted

        RewriteEngine On

        # Ensure authorization headers are passed to PHP.
        # Some Apache configurations may filter them and break usage of API,
        CalDAV, ...
        RewriteCond %{HTTP:Authorization} ^(.+)$
        RewriteRule . - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]

        # Redirect all requests to GLPI router, unless file exists.
        RewriteCond %{REQUEST_FILENAME} !-f
        RewriteRule ^(.*)$ index.php [QSA,L]

    </Directory>

    <FilesMatch .php$>
        SetHandler "proxy:unix:/run/php/php8.4-fpm.sock|fcgi://localhost/"
    </FilesMatch>

</VirtualHost>
```

Une fois le VirtualHost configuré on va activer notre nouveau site dans apache2 grâce à :

```
sudo a2ensite support.it-connectlab.fr.conf
```

Et par la suite désactiver le site par défaut de Apache2 :

```
sudo a2dissite 000-default.conf
```

Nous allons aussi activer le module rewrite (nécessaire pour activer la prise en charge des règles de réécriture) car nous l'avons utilisé dans le fichier de configuration du VirtualHost

```
sudo a2enmod rewrite
```

Il ne reste plus qu'à redémarrer le service Apache2 :

```
sudo systemctl restart apache2
```

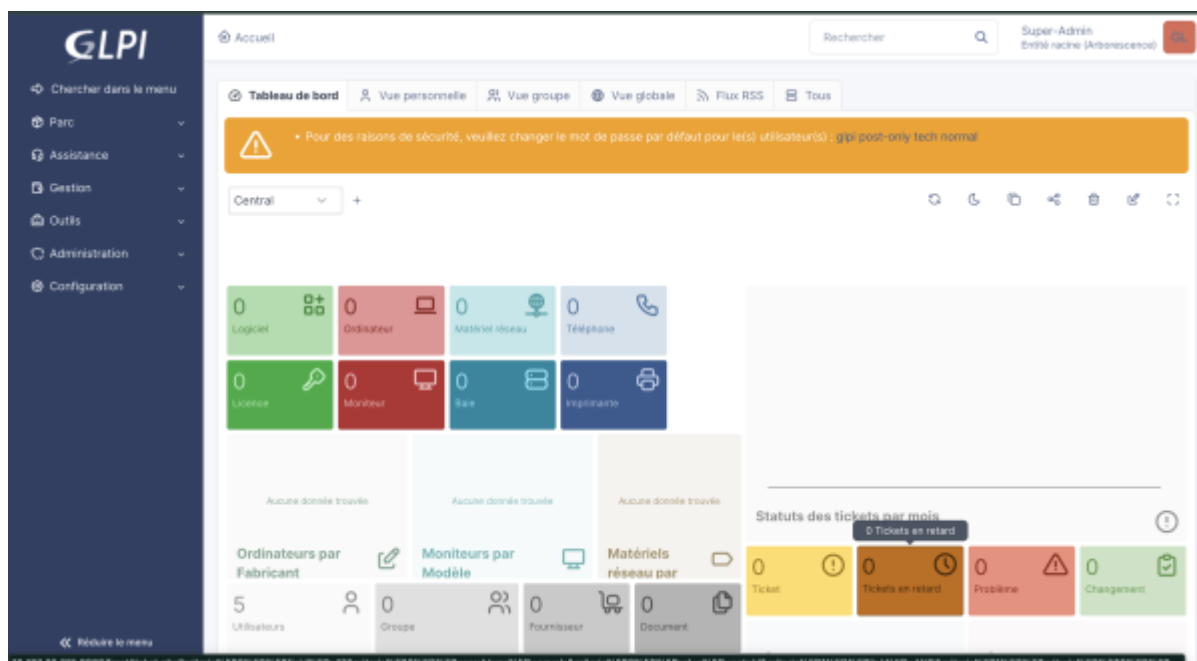
Config agent + GLPI

GLPI

une fois le site activé nous nous rendons sur l'interface graphique de GLPI grace a notre adresse IP 10.31.252.88

on va devvoir rentrer notre base de donn  e ainsi que le compte qui va avec.

Une fois avoir rentr   toute les bonnes informations nous allon arriv   sur cette page :

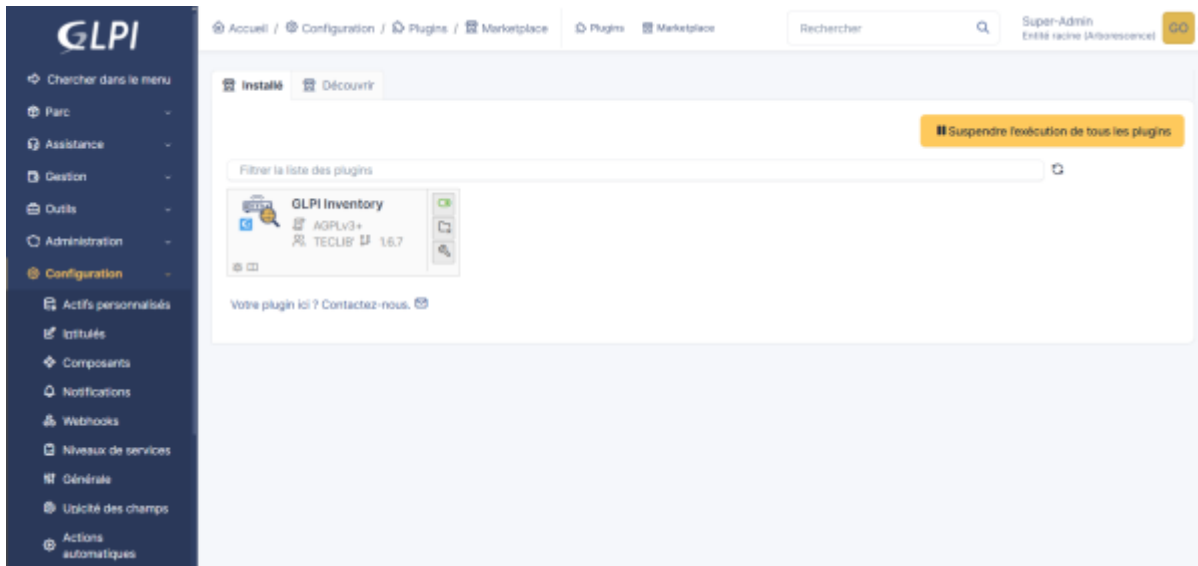


Si cette page est affich   ca veut dire que tout marche bien.

Une fois sur cette page nous allons devoir installer un plugins qui s'appelle GLPI Inventory qui va nous servir    communiquer avec un agent qu'on installera sur les machines de notre parc informatique.

Pour ce faire nous allons aller dans l'onglet : configuration > plugins > marketplace.

Il va falloir se cr  er un compte et par la suite installer GLPI Inventory.



Une fois le plugin installé on va devoir se rendre dans : Administration > Inventaire > activer l'inventaire car sans ça rien ne marche.

Installation + config agent GLPI

Pour installer l'agent de GLPI rien de bien compliqué on va devoir aller chercher le paquet grâce à la commande :

```
wget  
https://github.com/glpi-project/glpi-agent/releases/download/1.16/glpi-agent_1.16-1_all.deb
```

Puis l'extraire :

```
dpkg -i glpi-agent_1.16-1_all.deb
```

puis l'installer :

```
apt install -f
```

Après l'avoir installé il va falloir se rendre dans le fichier de conf :

```
nano /etc/glpi-agent/agent.cfg
```

Une fois dans le fichier de conf il va falloir ajouter la ligne *

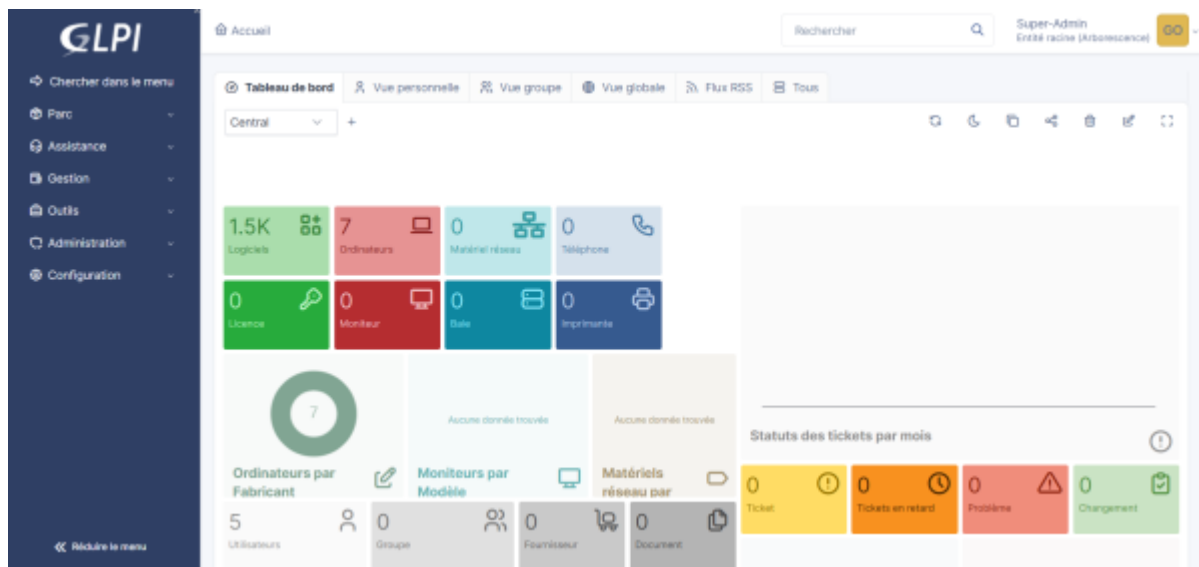
```
server = http://10.31.252.89/marketplace/glpiinventory/,  
http://10.31.252.88/marketplace/glpiinventory/
```

ce sont les 2 adresses de nos GLPI avec moi et Zoé car nous avons le même parc informatique. Ensuite on va devoir redémarrer glpi-agent et le forcer pour qu'il communique avec notre glpi

```
systemctl restart glpi-agent.service
```

glpi-agent --force

Une fois l'agent forcé nous avons normalement nos machines qui remontent sur le GLPI



Zabbix

Afin de réaliser de la supervision sur nos machines nous allons installer un serveur zabbix nous prévenant de toutes alertes sur nos service

Installation

nous allons commencer par installer apache2 qui hébergera notre zabbix

```
apt install apache2 php php-mysql php-mysqldb php-ldap php-bcmath  
phpmbstring php-gd php-pdo php-xml libapache2-mod-php
```

Une fois la procédure d'installation terminée, modifiez le fichier de configuration de zabbix afin de définir les quatre directives suivantes :

```
root@zabbix-srv:~# nano /etc/zabbix/zabbix_server.conf  
DBHost=10.31.252.33  
DBName=zabbix  
DBUser=nolan  
DBPassword=password
```

on va ensuite se connecter sur notre zabbix et le paramétrer

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Welcome to

Zabbix 7.4

Default language

English (en_US)

Back

Next step

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Check of pre-requisites

	Current value	Required	
PHP version	7.4.33	7.4.0	OK
PHP option "memory_limit"	128M	128M	OK
PHP option "post_max_size"	16M	16M	OK
PHP option "upload_max_filesize"	2M	2M	OK
PHP option "max_execution_time"	300	300	OK
PHP option "max_input_time"	300	300	OK
PHP databases support	MySQL		OK
PHP bcmath	on		OK
PHP mbstring	on		OK
PHP option "mbstring.func_overload"	off	off	OK

Back

Next step

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type

MySQL

Database host

10.31.0.33

Database port

3306

0 - use default port

Database name

zabbix

Store credentials in

Plain text

HashiCorp Vault

CyberArk Vault

User

zabbix

Password

Database TLS encryption

☐

Back

Next step

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Settings

Zabbix server name

FLS is GC3

Default time zone

(UTC+01:00) Europe/Paris

Default theme

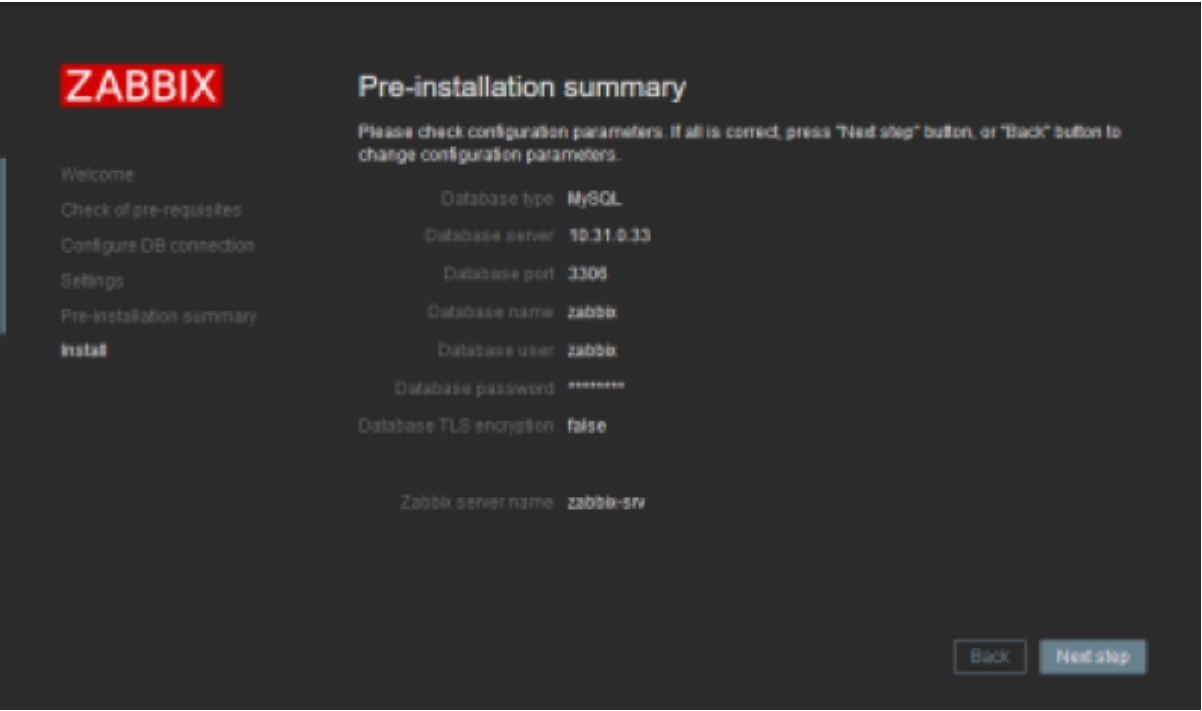
Dark

Encrypt connections from Web interface

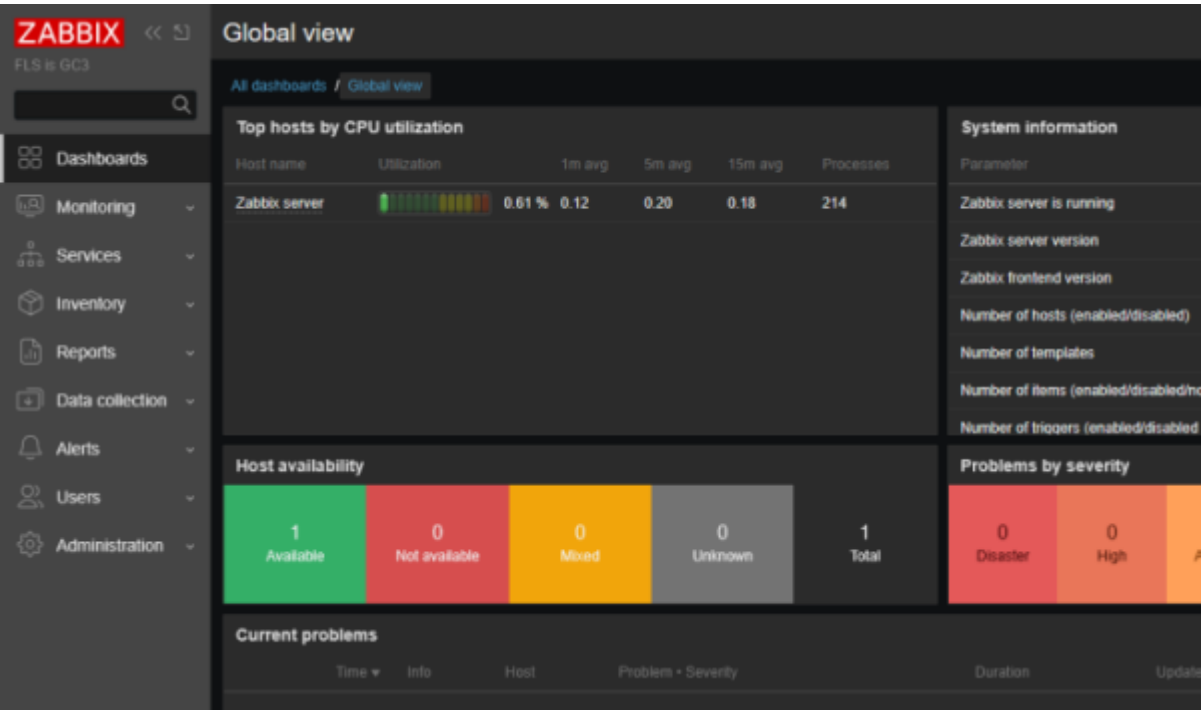
☐

Back

Next step



a la fin de l'installation si tout s'est bien passé le rendu est sencé être celui-ci



From:
<https://sisr2.beaupeyrat.com/> - Documentations SIO2 option SISR

Permanent link:
https://sisr2.beaupeyrat.com/doku.php?id=sisr2-usa:projet_professionnelles

Last update: 2026/05/25 23:08

