

Rapport Hebergement7.lan

ATTENTION :

Pour tous les serveurs, modifiez le fichier /etc/hostname et mettre les bons noms de machines

Serveur BDD (Sous Debian 10) :

On lui assigne une carte réseau en static dans le LAN (nano /etc/network/interfaces)

```
link/ether 00:0c:29:3a:49:c1 brd ff:ff:ff:ff:ff:ff
inet 192.168.207.2/24 brd 192.168.207.255 scope global ens33
    valid_lft forever preferred_lft forever
inet6 fe80::20c:29ff:fe3a:49c1/64 scope link
    valid_lft forever preferred_lft forever
```

Ainsi que les paramètres DNS dans /etc/resolv.conf

```
domain heberge7.lan
search heberge7.lan
nameserver 192.168.207.1
```

Paquets à installer (en bridge) : Apt install php-mysqli make build-essential mariadb-server

Mysql -u root -p

Create table bddhebergement

```
MariaDB [(none)]> use bddhebergement
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [bddhebergement]>
```

MariaDB [(none)]> use bddhebergement ; // N'oubliez pas les points virgules à la fin des lignes

MariaDB [bddhebergement]> CREATE USER ON bddhebergement.* 'admin'@'%' IDENTIFIED BY 'p@ssw0rd'

MariaDB [bddhebergement]> GRANT ALL PRIVILEGES ON *.* TO 'admin'@'%' IDENTIFIED BY 'p@ssw0rd' WITH GRANT OPTION;

MariaDB [bddhebergement]> flush privileges;

MariaDB [bddhebergement]> exit;

Ajoutez à la fin du fichier /etc/mysql/my.cnf : la ligne suivante : max_allowed_packet = 32M

Nano /etc/mysql/mariadb.conf.d/50-server.cnf :

Bind-address=0.0.0.0

Extension=mysqli enlever « ; »

Serveur Web (Sous Debian 10) :

On lui assigne une carte réseau en static dans le LAN (nano /etc/network/interfaces)

```
link/ether 00:0c:29:d3:4f:86 brd ff:ff:ff:ff:ff:ff
inet 192.168.207.1/24 brd 192.168.207.255 scope global ens33
    valid_lft forever preferred_lft forever
inet6 fe80::20c:29ff:fed3:4f86/64 scope link
    valid_lft forever preferred_lft forever
```

Log (avoir 20lignes) : tail -20 /var/log/apache2/error.log

Paquets à installer (en bridge) : apache2, php, php7.3, php-mysqli, php-mysql, ssh, curl, wget, bind9

php7.3-ldap php7.3-curl php-soap php-mbstring php-zip php7.3-mysql

php7.3libapache2-mod-php7.3php7.3-common php-pear php7.3-cli php7.3-gd

Activez le module permettant la gestion des sites personnels avec la commande : a2enmod userdir

Relancez apache avec la commande : service apache2 restart

Installation du service DNS :

/etc/resolv.conf :

```
domain heberge7.lan
search heberge7.lan
nameserver 127.0.0.1
```

Dans le fichier /etc/bind/named.conf.local :

```
GNU nano 3.2 /etc/bind/named.conf.local
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
zone "heberge7.lan" {
    type master;
    allow-update { none; };
    file "/etc/bind/db.heberge7.lan";
};
zone "207.168.192.in-addr.arpa" {
    type master;
    allow-update { none; };
    file "/etc/bind/db.207.168.192.in-addr.arpa";
};
```

Dans le fichier /etc/bind/db.heberge7.lan :

```
GNU nano 3.2 /etc/bind/db.heberge7.lan
$ttl 86400
@      IN      SOA      SrvWeb.heberge7.lan.  root.heberge7.lan. (
1
10800
3600
604800
38400)
@      IN      NS       SrvWeb.heberge7.lan.
SrvWeb IN      A       192.168.207.1
SrvBDD IN      A       192.168.207.2
SrvGipi IN     A       192.168.207.4
www    IN      CNAME   SrvWeb
```

Configuration de la zone inverse :

Dans le fichier /etc/bind/db.207.168.192.in-addr.arpa :

```
GNU nano 3.2 /etc/bind/db.207.168.192.in-addr.arpa
$ttl 86400
@      IN      SOA     SrvWeb.heberge7.lan  root.heberge7.lan. (
1
10800
3600
604800
38400 )
@      IN      NS      SrvWeb.heberge7.lan.
1      IN      PTR     SrvWeb
2      IN      PTR     SrvBDD
4      IN      PTR     SrvGipi
```

Testez la configuration avec la commande host :

host SrvWeb.heberge7.lan

```
root@SrvWeb:~# host SrvWeb.heberge7.lan
SrvWeb.heberge7.lan has address 192.168.207.1
```

host 192.168.207.1

```
root@SrvWeb:~# host 192.168.207.1
1.207.168.192.in-addr.arpa domain name pointer SrvWeb.207.168.192.in-addr.arpa.
```

Serveur ftp sur le SrvWeb :

apt install vsftpd

Configurez le service FTP en modifiant le fichier /etc/vsftpd.conf comme suit :

Décommentez la ligne #write_enable=YES (supprimez le # en début de ligne)

Décommentez la ligne #local_umask=022 (supprimez le # en début de ligne)

Remplacez vsftpd par ftp sur la ligne pam_service_name = vsftpd

Redémarrez le service : service vsftpd restart

On créer un user web admin pour nos programmeur

Useradd webadmin -d /var/www/html/

Passwd webadmin : on lui assigne un mot de passe

Chown webadmin * , Chgrp webadmin , chown webadmin /var/www -R

Installation du proftpd sur le SrvWeb (connexion en FTP):

Apt install proftpd

On modifie la ligne ServerName «debian» dans

Nano /etc/proftpd/proftpd.conf

```
ServerName                "SrvWeb"  
# Set to inetd only if you would run proftpd
```

Script Bash pour le serveur web (création, suppression, modification):

Dans /var/www/html créer le fichier avec le script

```
GNU nano 3.2                creationUtil.sh  
#!/bin/bash  
username=$1  
password=$2  
useradd $username -md /home/$username  
echo $username:$password | chpasswd  
mkdir /home/$username/public_html  
touch /home/$username/public_html/index.html_
```

D'abord on affecte 2 variables username et password

Ensuite on crée l'utilisateur linux ensuite on lui affecte son mot de passe, puis on lui crée son dossier de partage et pour finir on lui ajoute le fichier index.html.

Dans le fichier /etc/apache2/sites-available on copie le fichier 000-default.conf et on le renomme en XXX.conf (c'est notre modèle)

On modifie ce fichier dans ce style en y ajoutant le nom de son serveur ainsi que l'endroit où se stocke son site web:

```
ServerName XXXX.heberge7.lan  
DocumentRoot /home/XXXX/public_html
```

```
cp /etc/apache2/sites-available/XXXX.conf /etc/apache2/sites-available/$username.conf  
sed -i -e "s/XXXX/$username/g" /etc/apache2/sites-available/$username.conf  
a2ensite $username  
service apache2 restart  
echo "$username IN CNAME SrvWeb">>/etc/bind/db.heberge7.lan  
service bind9 restart_
```

Pour la suppression d'un utilisateur :

```
GNU nano 3.2 suppressionUtil.sh
#!/bin/bash
username=$1
password=$2
userdel $username
rm -fr /home/$username/public_html

rm /etc/apache2/sites-available/$username.conf
a2dissite $username
service apache2 reload
perl -ni.orig -e "print unless(/$username IN/)" /etc/bind/db.heberge7.1an
service bind9 reload
```

D'abord on affecte 2 variables username et password

On supprime l'utilisateur linux ensuite on lui efface son dossier de stockage qui contient le fichier index.html

On supprime le fichier de configuration apache et on désactive l'utilisateur puis on redémarre le service

Pour finir l'enregistrement DNS ce supprime et le service bind redémarre

Pour la modification d'un utilisateur :

Pour la modification des informations de l'utilisateur :

```
GNU nano 3.2 /var/www/html/modificationUtil.sh
#!/bin/bash
username=$1
password=$2

echo $username:$password | chpasswd_
```

Le mot de passe doit se changer automatiquement

SrvGipi (Sous Debian 10) :

On lui assigne une carte réseau en static dans le LAN (nano /etc/network/interfaces)

```
link/ether 00:0c:29:3e:19:3e brd ff:ff:ff:ff:ff:ff
inet 192.168.207.4/24 brd 192.168.207.255 scope global ens37
    valid_lft forever preferred_lft forever
inet6 fe80::20c:29ff:fe3e:193e/64 scope link
    valid_lft forever preferred_lft forever
```

Ainsi que les paramètres DNS dans /etc/resolv.conf

```
domain heberge7.lan
search heberge7.lan
nameserver 192.168.207.1
```

Configuration pour accéder à Internet via le proxy du BTS SIO

Modifier fichier /etc/wgetrc

```
http_proxy = http://NomUtil:MotDePasse@192.168.1.21:3128/
https_proxy = http://NomUtil:MotDePasse@192.168.1.21:3128/
ftp_proxy = http://NomUtil:MotDePasse@192.168.1.21:3128/
```

Prendre la dernière version de gipi pour notre cas 9.5.6 et fusioninventory ici 9.5+3.0

```
gipi-9.5.6.tgz fusioninventory-9.5+3.0.tar.bz2
```

Placez-vous dans le dossier **/home**.

Téléchargez la version 9.4.5 de GLPI avec la commande :

```
wget https://github.com/gipi-project/gipi/releases/download/9.4.5/gipi-9.4.5.tgz
```

Décompressez ce fichier avec la commande : **tar -xzf gipi-9.4.5.tgz**

Vérifiez que le dossier **gipi** a bien été créé avec la commande **ls**.

Copiez ce dossier dans l'arborescence du serveur web (**/var/www/html**) :

```
cp gipi /var/www/html/ -R
```

Changez le propriétaire de ce nouveau répertoire : `chown -R www-data /var/www/html/glpi`

Donner les droits en écriture au service web sur le dossier `/var/www/html/glpi/files` et sur le dossier `/var/www/html/glpi/config` :

```
cd /var/www/html/glpi
chmod 777 files config
cd files
chmod 777 _*
```

Dans le fichier `/etc/php/7.3/apache2/php.ini`, modifiez la ligne: `max_execution_time = 600`

Relancez le service **Apache** par la commande : `service apache2 restart`

Connectez-vous à **MariaDB** avec la commande `mysql -u root -p`

```
MariaDB [none]> CREATE DATABASE glpidb; // N'oubliez pas les points virgules à la fin des lignes
MariaDB [none]> use mysql;
MariaDB [mysql]> GRANT ALL PRIVILEGES ON glpidb.* TO glpiuser@localhost IDENTIFIED by 'glpi';
MariaDB [mysql]> exit;
```

Placez-vous sur votre client **Cit10** et ouvrez le navigateur à l'adresse : `http://10.10.0.1/glpi`

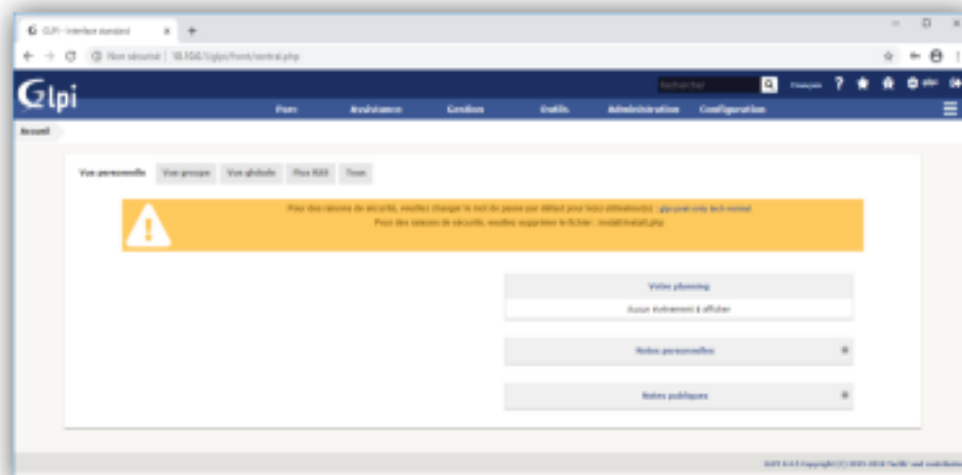
1. Select your language ➡ **Français**, puis **OK**.
2. Acceptez les termes de la licence ➡ Acceptez ➡ Cliquez sur **Continuer**.
3. Début de l'installation ➡ Cliquez sur **Installer**.
4. Etape 0 (tous les résultats doivent être au vert voire orange, sauf l'accès web au répertoire des fichiers qui doit être en rouge) ➡ Cliquez sur **Continuer**.
5. Etape 1 : ➡ Serveur MySQL : **localhost**
Utilisateur MySQL : **root**
Mode de passe MySQL : **root**
6. Etape 2 (Test de connexion à la bdd) : ➡ Saisissez **glpidb**, puis **Continuer**.
7. Etape 3 (Initialisation de la base) ➡ Cliquez sur **Continuer**.
8. Etape 4 (Récouter des données) ➡ Décocher "Envoyer "statistiques d'usage"
➡ Cliquez sur **Continuer**.
9. Etape 5 (Une dernière chose avant de démarrer) ➡ Cliquez sur **Continuer**.
10. Etape 6 (L'installation est terminée) ➡ Cliquez sur **Utiliser GLPI**.

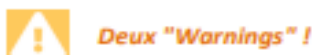
Les identifiants et mots de passe par défaut sont :

- glpi/glpi pour le compte administrateur
- tech/tech pour le compte technicien
- normal/normal pour le compte normal
- post-only/postonly pour le compte postonly


Connectez-vous à **GLPI** avec le compte "glpi" et son mot de passe "glpi".

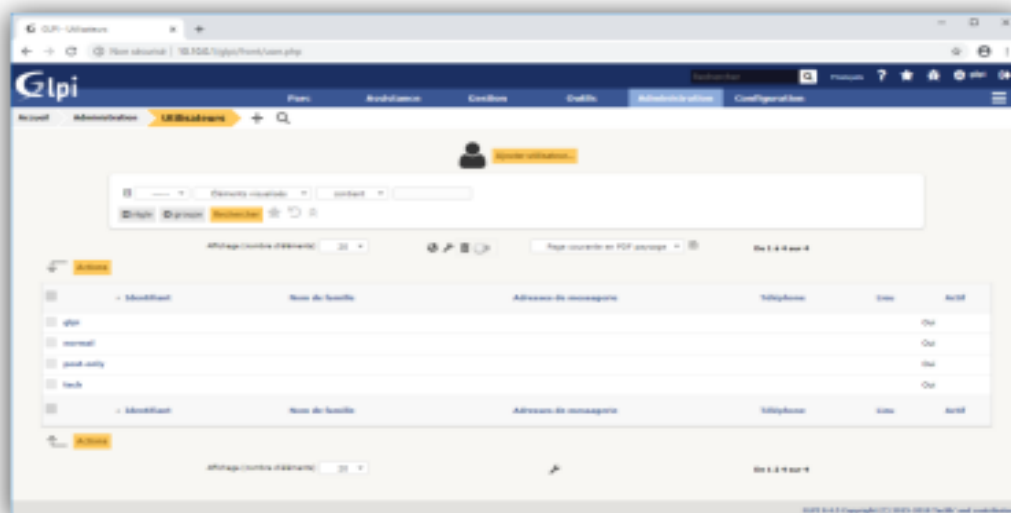
On obtient :



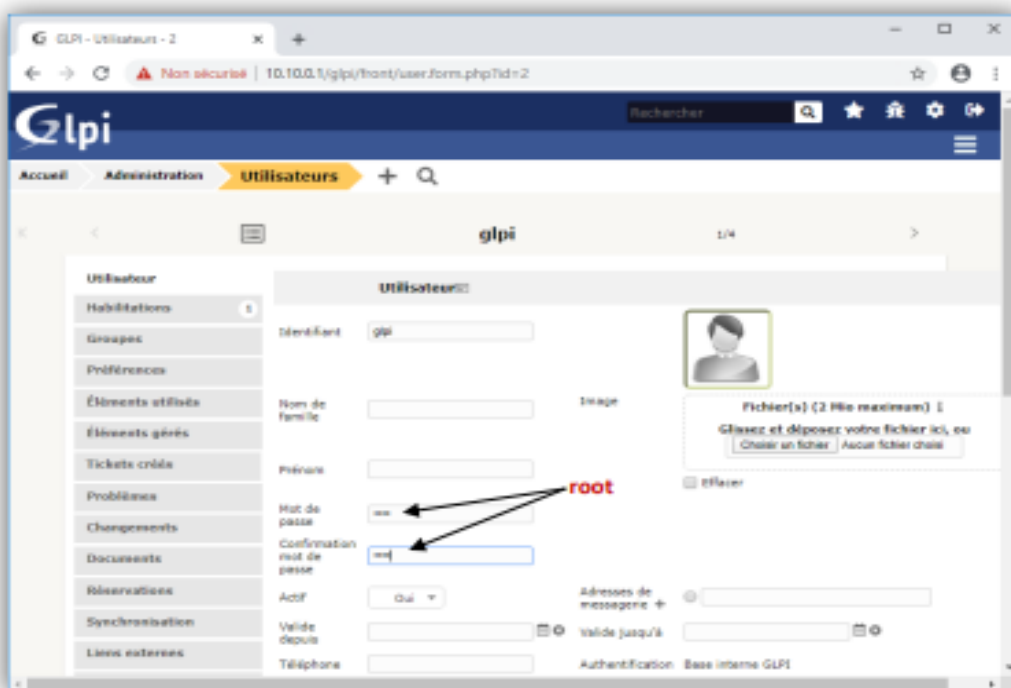


Suppression du fichier d'installation : `rm /var/www/html/gli/install/install.php`

Changement des mots de passe par défaut. Cliquez sur  puis sur l'option "Utilisateurs" du menu "Administration".



Cliquez sur gli.



Cliquez sur **Sauvegarder**.

Faites la même opération pour les autres comptes.

- le compte **normal** ➔ Mot de passe : **normalroot**
- le compte **post-only** ➔ Mot de passe : **post-onlyroot**
- le compte **tech** ➔ Mot de passe : **techroot**

Cliquez sur **Accueil**.

GLPI est prêt !

Compte admin :

Login : Administrateur


mdp : P@sswOrd

Ouvrez **GLPI** avec le compte **glpi**.

Sélectionnez "Plugins" du menu "Configuration".

Cliquez sur "Voir le catalogue des plugins".

Cliquez sur "FusionInventory For GLPI".

Cliquez sur .

Puis faites un clic droit sur "fusioninventory-9.4+2.4.tar.bz2" le lien devrait être du genre :

<https://github.com/fusioninventory/fusioninventory-for-glpi/releases/download/glpi9.4%2B2.4/fusioninventory-9.4+2.4.tar.bz2>

Revenez sur votre serveur **SrvLinux**.

Placez vous dans **/home**

Téléchargez la version 9.4 de FusionInventory avec la commande (une esule ligne) :

```
wget https://github.com/fusioninventory/fusioninventory-for-glpi/releases/download/glpi9.4%2B2.4/fusioninventory-9.4+2.4.tar.bz2
```

Décompressez l'archive avec la commande : **tar -xvf fusioninventory-9.4+2.4.tar.bz2**

Vérifiez que le dossier **fusioninventory** a bien été créé avec la commande **ls**.

Copiez le dossier **fusioninventory** dossier **/var/www/html/glpi/plugins** comme ceci :

```
cp fusionInventory /var/www/html/glpi/plugins -R
```

Revenez sur votre machine cliente **Clt10**.

Sélectionnez **Plugins** du menu **Configuration**. Vous devez obtenir ceci :



- New	Dossier	Version	Licence	Statut	Auteurs	Site Web	Actions
FusionInventory	fusioninventory	9.4+2.4	AGPLv3+	Non installé	David DURIEUX & FusionInventory team	🔗	📁 🗑️

- New	Dossier	Version	Licence	Statut	Auteurs	Site Web	Actions
-------	---------	---------	---------	--------	---------	----------	---------

Cliquez sur , puis sur . Vous devez obtenir : 

Cliquez sur l'option "FusionInventory" du menu "Administration".

On obtient un Warning :  Le cron de GLPI ne fonctionne pas, voir documentation

En cliquant sur "Voir documentation", on obtient :

```
For Linux, add in crontab:  
***** /usr/bin/php5 /var/www/glpi/front/cron.php &>/dev/null
```

La version de php que nous avons installée est **php7.3**, il nous faudra modifier un peu la **cron**.

Revenez sur le serveur **SrvLinux**.

Pour créer une **cron**, il faut utiliser la commande : **crontab -e**

Le système nous demande ensuite quel éditeur nous désirons, choisissons **nano** (option 1).

Ajoutez à la fin du fichier :

```
***** /usr/bin/php7.3 /var/www/html/glpi/front/cron.php &>/dev/null
```

On relance **cron** par la commande **/etc/init.d/cron restart**

Revenez sur **Clt10**. Et cliquez sur l'option "FusionInventory" du menu "Administration". Cette fois-ci, il n'y a plus de Warning.



On lui installe phpmyadmin afin que nos programmeur puisse accéder à la base glpidb le principe est le même qu'auparavant.

Une fois cela fait on peut créer des tickets personnalisé dans l'onglet configuration :



Ensuite choisir :



Une fois ici on peut créer nos tickets en cliquant sur le + :

