

# Documentation technique

## BTS Services informatique aux organisations

*Option Solutions d'Infrastructure, Systèmes et Réseaux (S.I.S.R)*



### SITUATION N°1

**Nom de l'étudiant : LOPEZ-SIGURA Florian | FLS**

Auteur	Date	Description
FLS	23/04/2025	Création du document



## Table des matières :

<b>Table des matières :</b>	<b>2</b>
<b>Time zone :</b>	<b>4</b>
<b>Mode opératoire MariaDB</b>	<b>5</b>
Étape 1 : Connexion à MariaDB	5
Étape 2 : Création de la base de données	5
Étape 3 : Création de l'utilisateur admin avec tous les droits	5
Étape 4 : Création de l'utilisateur applicatif (consultation + mise à jour)	5
Étape 5 : Appliquer les changements de droits	5
Étape 6 : Importation des fichiers SQL	5
<b>Mode opératoire Apache</b>	<b>6</b>
ÉTAPE 1 : Mise à jour du système	6
ÉTAPE 2 : Installation du serveur web Apache	6
ÉTAPE 3 : Vérification	6
ÉTAPE 4 : Vérification du fonctionnement via le navigateur	6
ÉTAPE 5 : Configuration du répertoire web	8
ÉTAPE 6 : Gestion des sites virtuels	8
ÉTAPE 7 : [HTTPS] : Mise à jour des paquets	9
ÉTAPE 8 : Activer les modules nécessaires pour HTTPS	9
ÉTAPE 9 : Créer un certificat SSL auto-signé pour le serveur	9
ÉTAPE 10 : Créer un hôte virtuel HTTPS	9
ÉTAPE 11 : Activer le site HTTPS et désactiver éventuellement le HTTP par défaut	10
ÉTAPE 12 : Redémarrer Apache pour appliquer la configuration	10
ÉTAPE 13 : Apache headers	11
<b>Mode opératoire DNS (Bind9)</b>	<b>12</b>
Étape 1 : Mise à jour des paquets et installation de Bind9 avec ses utilitaires	12
Étape 2 : Déclaration des zones DNS (directe et inverse)	12
Étape 3 : Création du fichier de zone directe	12
Étape 4 : Création du fichier de zone inverse	13
Étape 5 : Vérification de la syntaxe des fichiers de configuration	13
Étape 6 : Redémarrage du service Bind9 pour appliquer la configuration	14
Étape 7 : Activation de Bind9 au démarrage de la machine	14
<b>Mode opératoire HA ( Corosync &amp; pacemaker )</b>	<b>15</b>
Étape 1 : Installation et configuration de Corosync et Pacemaker	15
Étape 2 : Désactivation de stonith	17
Étape 3 : configuration de l'ip flottante (IPFailover)	17
Étape 4 : RéPLICATION de base de donnée	19
Étape 5 : RéPLICATION de base de donnée :	20
Étape 6 : Testing	23
<b>Mode opératoire Nagios :</b>	<b>25</b>
ÉTAPE 1 : Mise à jour du système.	25
ÉTAPE 2 : Installation de Nagios et des composants nécessaires	25
ÉTAPE 3 : Activation du service Nagios	25

ÉTAPE 4 : Démarrage immédiat du service Nagios	25
ÉTAPE 5 : Création d'un utilisateur Nagios pour l'interface web.	25
ÉTAPE 6 : Vérification de la validité de la configuration Nagios	26
ÉTAPE 7 : Redémarrage d'Apache	26
ÉTAPE 8 : Ajout d'un hôte à surveiller	26
ÉTAPE 9 : Redémarrage de Nagios pour appliquer les nouveaux paramètres	28
ÉTAPE 10 : Mise en place de HTTPS pour Nagios	28
<b>Annexe :</b>	<b>30</b>
Schéma infrastructure GSB :	30



## Time zone :

Faire la commande “ TZSELECT ” pour pouvoir avoir le bon fuseau horaire local :

```
root@intralabFLS:/home/sio# tzselect
Please identify a location so that time zone rules can be set correctly.
Please select a continent, ocean, "coord", or "TZ".
1) Africa                                7) Europe
2) Americas                             8) Indian Ocean
3) Antarctica                           9) Pacific Ocean
4) Asia                                  10) coord - I want to use geographical coordinates.
5) Atlantic Ocean                      11) TZ - I want to specify the timezone using the Posix TZ format.
6) Australia
#? 8
Please select a country whose clocks agree with yours.
1) British Indian Ocean Territory
2) French S. Terr.
3) Maldives
4) Mauritius
#? 4

The following information has been given:

        Mauritius

Therefore TZ='Indian/Mauritius' will be used.
Selected time is now:  Sat May 10 23:30:01 +04 2025.
Universal Time is now: Sat May 10 19:30:01 UTC 2025.
Is the above information OK?
1) Yes
2) No
#? y
Please enter a number in range.
#? 1

You can make this change permanent for yourself by appending the line
    TZ='Indian/Mauritius'; export TZ
to the file '.profile' in your home directory; then log out and log in again.

Here is that TZ value again, this time on standard output so that you
can use the /usr/bin/tzselect command in shell scripts:
Indian/Mauritius
root@intralabFLS:/home/sio# date
sam. 10 mai 2025 23:30:49 +04
root@intralabFLS:/home/sio#
```



## Mode opératoire MariaDB

### Étape 1 : Connexion à MariaDB

```
mysql -u root -p
```

### Étape 2 : Création de la base de données

```
CREATE DATABASE gsb_frais CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
```

Étape 3 : Création de l'utilisateur admin avec tous les droits

```
CREATE USER 'gsb_frais'@'localhost' IDENTIFIED BY 'P@ssw0rdGSB';  
GRANT ALL PRIVILEGES ON gsb_frais.* TO 'gsb_frais'@'localhost';
```

### Étape 3 : Création de l'utilisateur admin avec tous les droits

```
CREATE USER 'gsb_frais'@'localhost' IDENTIFIED BY 'P@ssw0rdGSB';  
GRANT ALL PRIVILEGES ON gsb_frais.* TO 'gsb_frais'@'localhost';
```

### Étape 4 : Création de l'utilisateur applicatif (consultation + mise à jour)

```
CREATE USER 'userGsb'@'localhost' IDENTIFIED BY 'secret';  
GRANT SELECT, UPDATE ON gsb_frais.* TO 'userGsb'@'localhost';
```

### Étape 5 : Appliquer les changements de droits

```
FLUSH PRIVILEGES;
```

### Étape 6 : Importation des fichiers SQL

```
exit
```

```
mysql -u gsb_frais -p gsb_frais < gsb_frais_structure.sql  
mysql -u gsb_frais -p gsb_frais < gsb_frais_insert_tables_statiques.sql
```

## Mode opératoire Apache

### ÉTAPE 1 : Mise à jour du système

Avant toute installation, il est important de mettre à jour la base de données des paquets afin d'avoir accès aux dernières versions stables des logiciels disponibles pour Debian.

```
apt update && apt upgrade -y
```

### ÉTAPE 2 : Installation du serveur web Apache

Cette commande installe les services Apache2, php et Mysql les plus utilisés pour les systèmes Linux. Il s'agit uniquement de versions stables et précompilées, livrées avec les modules de base nécessaires.

```
apt install apache2 php libapache2-mod-php mysql-server php-mysql
```

module php :

```
apt install php-curl php-gd php-intl php-json php-mbstring php-xml php-zip
```

### ÉTAPE 3 : Vérification

On vérifie que le service est bien actif et fonctionne correctement après l'installation.

```
systemctl status apache2
```

Si Apache est inactif, on peut le démarrer manuellement :

```
systemctl start apache2  
systemctl enable apache2
```

### ÉTAPE 4 : Vérification du fonctionnement via le navigateur

Depuis un autre poste ou le navigateur du serveur, accéder à l'adresse suivante :

<http://172.18.155.82>

Si Apache est bien installé, vous verrez une page d'accueil "Apache2 Debian Default Page".

## **ÉTAPE 5** : Configuration du répertoire web

Par défaut, Apache utilise le répertoire suivant pour héberger les sites web :

```
/var/www/html
```

Mais nous allons utiliser le répertoire suivant :

```
/var/www/appliGSB
```

```
mkdir /var/www/appliGSB  
usermod -aG www-data sio  
chown -R www-data:www-data /var/www/appliGSB  
chmod -R 755 /var/www/appliGSB
```

## **ÉTAPE 6** : Gestion des sites virtuels

Pour héberger le sites ou séparer la configuration, ont vas créer un VirtualHost :

```
nano /etc/apache2/sites-available/appliGSB.conf
```

```
<VirtualHost *:80>
```

```
Redirect permanent / https://gestionfraisinralabfls.gsb.coop/  
ServerAdmin webmaster@localhost  
ServerName gestionfraisinralabfls.gsb.coop  
ServerAlias www.gestionfraisinralabfls.gsb.coop  
DocumentRoot /var/www/appliGSB/appliFrais_avecMysqli  
ErrorLog ${APACHE_LOG_DIR}/error.log  
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
</VirtualHost>
```

On vas ensuite transférer les fichiers web sur le serveurs via la commande scp

```
scp c:\Users\lsf\Downloads\appliFrais_avecMysqli sio@172.18.155.82:/var/www/appliGSB  
scp c:\Users\lsf\Downloads\bddAppliFrais.zip sio@172.18.155.82:/var/www/appliGSB
```

Activation du site et rechargeement d'Apache

```
a2ensite appliGSB.conf  
a2dissite 000-default.conf  
systemctl reload apache2
```

### ÉTAPE 7 : [HTTPS] : Mise à jour des paquets

```
apt update && apt upgrade -y
```

### ÉTAPE 8 : Activer les modules nécessaires pour HTTPS

ssl : permet de gérer les connexions sécurisées  
headers : utile pour ajouter des en-têtes HTTP strictes (bonnes pratiques de sécurité)

```
a2enmod ssl  
a2enmod headers
```

### ÉTAPE 9 : Créer un certificat SSL auto-signé pour le serveur

Le certificat sera valide 1 an (365 jours)

```
mkdir -p /etc/ssl/appliGSB/  
openssl req -x509 -nodes -days 365 -newkey rsa:2048 \  
-keyout /etc/ssl/appliGSB/intralablsf.key \  
-out /etc/ssl/appliGSB/intralablsf.crt
```

### ÉTAPE 10 : Créer un hôte virtuel HTTPS

On définit : un VirtualHost sur le port 443 avec le certificat SSL

```
nano /etc/apache2/sites-available/appliGSB_ssl.conf
```

```
<VirtualHost *:443>  
    ServerName gestionfraisintralabfls.gsb.coop  
    DocumentRoot /var/www/appliGSB/appliFrais_avecMysqli  
  
    SSLEngine on  
    SSLCertificateFile /etc/ssl/appliGSB/intralablsf.crt
```

```
SSLCertificateKeyFile /etc/ssl/appliGSB/intralablsf.key  
  
<Directory /var/www/appliGSB/appliFrais_avecMysqli>  
    AllowOverride All  
    Require all granted  
</Directory>  
</VirtualHost>
```

**ÉTAPE 11** : Activer le site HTTPS et désactiver éventuellement le HTTP par défaut

a2ensite appliGSB\_ssl.conf  
a2dissite appliGSB.conf

**ÉTAPE 12** : Redémarrer Apache pour appliquer la configuration

systemctl reload apache2

ÉTAPE 13 : Vérifier le bon fonctionnement

Ouvre ton navigateur et va à l'adresse :

<https://172.18.155.82>

Un avertissement de sécurité peut apparaître à cause du certificat auto-signé (normal)

Non sécurisé <https://gestionfraisinralabfls.gsb.coop/cSeConnecter.php>

Nouveaux Bot discord Tahl Jeux Coloriages P... Mandalas Noms de domaine OptiFine\_1.17.1\_H... LE DICTIONNAIRE...

**Suivi du remboursement des frais**

**Identification utilisateur**

\* Login :

\* Mot de passe :

  Cette page est conforme aux standards du Web



## ÉTAPE 13 : Apache headers

à rajouté dans le fichier -> `/etc/apache2/sites-available/appliGSB_ssl.conf`

```
Header always set Strict-Transport-Security "max-age=63072000; includeSubDomains; preload"
```

```
Header always set X-Frame-Options "SAMEORIGIN"
```

```
Header always set X-Content-Type-Options "nosniff"
```

```
Header always set Referrer-Policy "no-referrer-when-downgrade"
```

```
Header always set Permissions-Policy "geolocation=(), microphone=(), camera=()"
```

```
Header always set X-XSS-Protection "1; mode=block"
```

```
Header always set Content-Security-Policy "default-src 'self'; script-src 'self'; style-src 'self'; img-src 'self' data:; font-src 'self'; object-src 'none'; base-uri 'self'; form-action 'self'"
```

Ce que chaque en-tête fait :

- **Strict-Transport-Security** : Forcer HTTPS pour 2 ans + sous-domaines (utilisé pour HSTS preload).
- **X-Frame-Options** : Empêche l'affichage du site dans une iframe (clickjacking).
- **X-Content-Type-Options** : Empêche la détection automatique du type MIME.
- **Referrer-Policy** : Contrôle les données du référent envoyées à d'autres sites.
- **Permissions-Policy** : Bloque l'accès aux API sensibles (géolocalisation, micro, caméra).
- **X-XSS-Protection** : Active le filtre XSS côté navigateur (utile avec anciens navigateurs).
- **Content-Security-Policy (CSP)** : Politique très stricte qui bloque les ressources externes non autorisées (scripts, styles, images, etc.).

## Mode opératoire DNS (Bind9)

### Étape 1 : Mise à jour des paquets et installation de Bind9 avec ses utilitaires

```
apt update && apt install bind9 bind9utils bind9-doc dnsutils -y
```

Cette commande met à jour la liste des paquets disponibles sur le système, puis installe le serveur DNS Bind9, les outils de gestion des zones DNS, la documentation utile pour les administrateurs système, ainsi que dnsutils pour tester la résolution DNS via des commandes comme dig ou nslookup.

### Étape 2 : Déclaration des zones DNS (directe et inverse)

```
echo 'zone "gsbfls" {  
    type master;  
    file "/etc/bind/db.gsbfls";  
};'  
  
zone "155.18.172.in-addr.arpa" {  
    type master;  
    file "/etc/bind/db.172.18.155";  
};' > /etc/bind/named.conf.local
```

Ce bloc définit :

- La zone directe "gsbfls" qui permet de traduire des noms de domaine en adresses IP
- La zone inverse "155.18.172.in-addr.arpa" qui permet de traduire des adresses IP en noms

### Étape 3 : Crédation du fichier de zone directe

```
cp /etc/bind/db.local /etc/bind/db.gsbfls.coop  
echo "$TTL 604800  
@ IN SOA ns1.gsbfls. admin.gsbfls. (  
        2      ; Numéro de série  
        604800   ; Fréquence de rafraîchissement  
        86400    ; Délai de tentative en cas d'échec  
        2419200  ; Délai d'expiration  
        604800 ) ; Durée du cache négatif  
  
@ IN NS ns1.gsbfls.  
@ IN A 172.18.155.84
```

```
ns1 IN A 172.18.155.84
www IN A 172.18.155.84
ftp IN A 172.18.155.84" > /etc/bind/db.gsbfls
```

Ce fichier contient :

- Le SOA (Start of Authority), qui définit l'autorité de la zone
- Les enregistrements A pour le domaine et ses sous-domaines : www, ftp, ns1

#### Étape 4 : Création du fichier de zone inverse

```
cp /etc/bind/db.127 /etc/bind/db.172.18.155
echo "$TTL 604800
@ IN SOA ns1.gsbfls. admin.gsbfls.coop. (
        2      ; Numéro de série
        604800   ; Fréquence de rafraîchissement
        86400    ; Délai de tentative
        2419200  ; Expiration
        604800 ) ; Cache négatif

@ IN NS ns1.gsbfls.coop.
84 IN PTR ns1.gsbfls.coop." > /etc/bind/db.172.18.155
```

Ce fichier permet de faire la correspondance inverse :

Il associe l'adresse IP 172.18.155.84 au nom ns1.gsbfls grâce à l'enregistrement PTR.

#### Étape 5 : Vérification de la syntaxe des fichiers de configuration

```
named-checkconf
named-checkzone gsbfls.coop /etc/bind/db.gsbfls.coop
named-checkzone 155.18.172.in-addr.arpa /etc/bind/db.172.18.155
```

Ces commandes vérifient :

- La configuration générale de Bind9
- La validité syntaxique de la zone directe et de la zone inverse

**Étape 6** : Redémarrage du service Bind9 pour appliquer la configuration

```
systemctl restart bind9
```

**Étape 7** : Activation de Bind9 au démarrage de la machine

```
systemctl enable bind9
```

**Le service Bind9 est maintenant actif, les requêtes DNS sur le domaine gsbfls.coop peuvent être traitées pour les différents services sauf le services web qui reste gsb.coop**

```
root@intralabFLS:/home/sio# named-checkconf
root@intralabFLS:/home/sio# named-checkzone gsbfls /etc/bind/db.gsbfls
zone gsbfls/IN: loaded serial 2
OK
root@intralabFLS:/home/sio# named-checkzone 155.18.172.in-addr.arpa /etc/bind/db
.172.18.155
zone 155.18.172.in-addr.arpa/IN: loaded serial 2
OK
root@intralabFLS:/home/sio#
```

## Mode opératoire HA ( Corosync & pacemaker )

### Étape 1 : Installation et configuration de Corosync et Pacemaker

```
apt update  
apt install corosync pacemaker crmsh
```

Vérification de la ressource : service corosync status

```
root@intralabFLS:/home/sio# service corosync status  
● corosync.service - Corosync Cluster Engine  
   Loaded: loaded (/lib/systemd/system/corosync.service; enabled; vendor preset: enabled)  
   Active: active (running) since Sun 2025-04-06 14:53:55 +04; 1 months 2 days ago  
     Docs: man:corosync  
           man:corosync.conf  
           man:corosync_overview  
   Main PID: 432 (corosync)  
     Tasks: 9 (limit: 1115)  
   Memory: 159.7M  
     CPU: 6h lmin 58.883s  
    CGroup: /system.slice/corosync.service  
           └─432 /usr/sbin/corosync -f
```

Création d'un fichier " authkey " avec la commande " corosync-keygen " ce fichier doit être présent sur tous les nœuds du cluster.

Il faut désormais cloner la VM " Maître " pour pouvoir créer le cluster.

Définir un mot de passe pour l'utilisateur hacluster

```
passwd hacluster
```

Éditer le fichier corosync.conf :

```
nano /etc/corosync/corosync.conf sur le Master et slave
```

On va venir modifier les variables suivantes :

```
bindnetaddr: 172.18.155.0  
mcastaddr: 239.255.1.8
```

```
nodelist {  
    node {  
        name: wan-master  
        nodeid: 1  
        ring0_addr: 172.18.155.82  
    }  
    node {  
        name: wan-slave  
        nodeid: 2  
        ring0_addr: 172.18.155.83  
    }  
}
```

On va démarrer tout les services et faire un crm status pour vérifier la configuration :

```
systemctl restart corosync  
systemctl enable corosync  
crm status
```

```
root@intralabFLS:/home/sio# crm status  
Cluster Summary:  
* Stack: corosync  
* Current DC: wan-MASTER (version 2.0.5-ba59be7122) - partition with quorum  
* Last updated: Fri May 9 14:34:51 2025  
* Last change: Sun Apr 6 14:47:40 2025 by root via crm_attribute on wan-MASTER  
* 2 nodes configured  
* 1 resource instance configured  
  
Node List:  
* Online: [ wan-MASTER wan-slave ]  
  
Full List of Resources:  
* IPFailover (ocf::heartbeat:IPAddr2): Started wan-MASTER  
root@intralabFLS:/home/sio# █
```

## Étape 2 : Désactivation de stonith

C'est en fait un mécanisme pour éteindre complètement le serveur qui vient de flancher en éteignant son onduleur. C'est un procédé surtout utilisé avec des disques partagés car il serait dangereux que l'ordinateur qui est supposé être hors d'état vienne écrire sur le disque partagé et corrompre/altérer les données.

```
crm configure property stonith-enabled=false
```

La vérification (root@intralabXX:~# crm\_verify -L -V ) ne renvoie plus d'erreur.

## Étape 3 : configuration de l'ip flottante (IPFailover)

```
root@intralabXX:~# crm configure primitive IPFailover ocf:heartbeat:IPaddr2 params ip=172.18.155.84 cidr_netmask=21 nic=ens192 iflabel=VIP
```

- **primitive** : argument pour ajouter une primitive regroupant plusieurs valeurs indiquant au Cluster quels scripts utiliser pour la ressource, où le trouver et à quel standard il correspond.
- **ocf** : classe de la ressource (ça pourrait donc aussi être lsb) · heartbeat : fournisseur de la ressource
- **IPaddr2** : ressource gérant les adresses IPv4 virtuelles ==> le script appelé
- **params** : déclaration des paramètres nécessaires à la ressource · IPFailover : le nom de la ressource (il est évidemment libre... mais doit être suffisamment « parlant »),
- **IPaddr2** : le script appelé · params : suivent les différents paramètres à appliquer
- **ip=17218.155.84** : nom et valeurs du paramètre « ip »
- **cidr\_netmask=21** : masque de sous-réseau en notation CIDR
- **nic=ens192** : carte réseau sur laquelle est appliquée l'adresse IP virtuelle
- **iflabel=VIP** : permet de donner un label (étiquette) à la carte réseau virtuelle. Sans ce label, la VIP n'est pas visible avec la commande ifconfig mais seulement avec la commande ip addr show

## Vérification : crmstatus

```
root@intralabFLS:/home/sio# crm status
Cluster Summary:
  * Stack: corosync
  * Current DC: wan-master (version 2.0.5-ba59be7122) - partition with quorum
  * Last updated: Fri May  9 14:45:29 2025
  * Last change: Sun Apr  6 14:47:40 2025 by root via crm_attribute on wan-master
  * 2 nodes configured
  * 1 resource instance configured

Node List:
  * Online: [ wan-master wan-slave ]

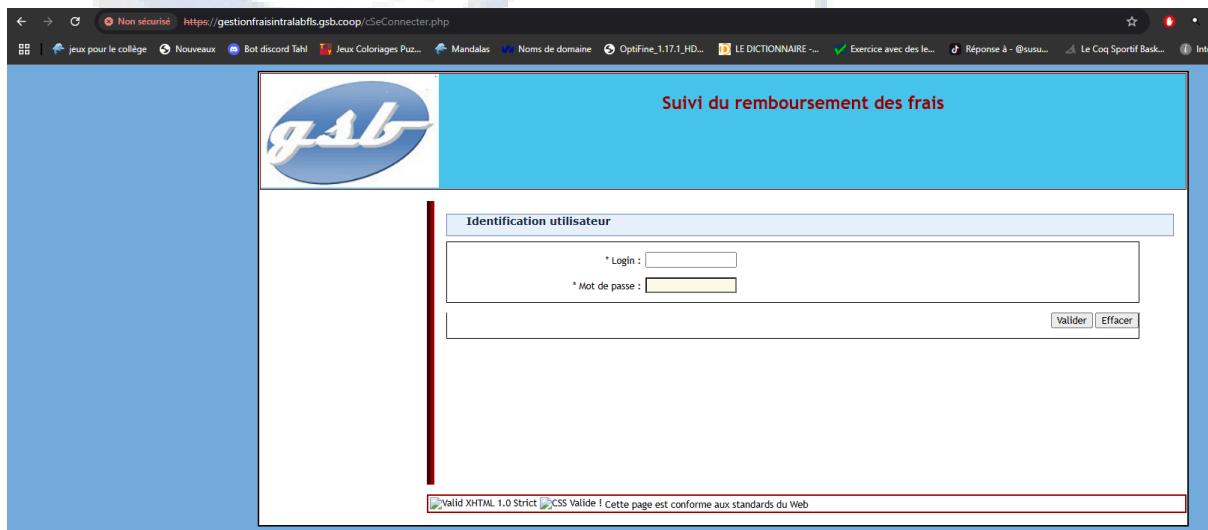
Full List of Resources:
  * IPFailover (ocf::heartbeat:IPAddr2):           Started wan-master
```

cmd windows : nslookup [gestionfraisintralabfls.gsb.coop](https://gestionfraisintralabfls.gsb.coop)

```
C:\Users\Florian>nslookup gestionfraisintralabfls.gsb.coop
Serveur : ADW2019S.sio.lan
Address: 172.18.159.250

Réponse ne faisant pas autorité :
Nom : gestionfraisintralabfls.gsb.coop
Address: 172.18.155.84
```

site web [gestionfraisintralabfls.gsb.coop](https://gestionfraisintralabfls.gsb.coop) :



## **Étape 4 : RéPLICATION de base de donnée**

Sur le serveur MASTER :

Configurer le fichier /etc/mysql/mariadb.conf.d/50-server.cnf

- commenter la ligne bind-address = 127.0.0.1
- Avoir différent server-id sur le master - slave
- Décommenter la ligne log\_bin = /var/log/mysql/mysql-bin.log
- Décommenter la ligne #max\_binlog\_size = 100M
- Rajouter la ligne binlog\_do\_db = gsb\_frais

Bloquer l'écriture via la commande sous mariadb : FLUSH TABLES WITH READ LOCK;

Vérifier que la base de donnée est bien répliqué : show master status;

il faut noter les champs " file " et " position " -> mysql-bin.000001 ; 3921

Redémarrer mariadb

-----

Sur le serveur SLAVE :

Configurer le fichier /etc/mysql/mariadb.conf.d/50-server.cnf

- Avoir différent server-id sur le master - slave
- Décommenter la ligne #max\_binlog\_size = 100M
- Rajouter la ligne master-retry-count = 20
- Rajouter la ligne replicate-do-db = gsb\_frais

Sous mariadb :

```
mysql -u root -p
```

```
stop slave;
```

```
change master to master_host='172.18.155.82', master_user='gsb_frais',  
master_password='P@ssw0rdGSB', master_log_file='mysql-bin.000001',  
master_log_pos=3921;
```

```
start slave ;
```

Redémarrer mariadb

En dernier sur le serveur MASTER : il faut débloquer les tables sur mysql : UNLOCK TABLES;

### Étape 5 : RéPLICATION de base de donnée :

Configuration du fichier /etc/mysql/mariadb.conf.d/50-server.cnf - **MASTER**

- commenter la ligne bind-address
- server-id 1
- Décommenter la ligne #max\_binlog\_size = 100M
- Rajouter la ligne master-retry-count = 20
- Rajouter la ligne replicate-do-db = gsb\_frais

```
sio@intralabFLS: ~  
GNU nano 5.4          /etc/mysql/mariadb.conf.d/50-server.cnf *  
  
# Instead of skip-networking the default is now to listen only on  
# localhost which is more compatible and is not less secure.  
#bind-address        = 127.0.0.1
```

```
GNU nano 5.4          /etc/mysql/mariadb.conf.d/50-server.cnf
#long_query_time      = 10
#log_slow_verbosity   = query_plan,explain
#log-queries-not-using-indexes
#min_examined_row_limit = 1000

# The following can be used as easy to replay backup logs or
# note: if you are setting up a replication slave, see README
#       other settings you may need to change.
server-id              = 1
log_bin                 = /var/log/mysql/mysql-bin.log
expire_logs_days        = 10
max_binlog_size         = 100M
binlog_do_db = gsb_frais
*
```

Après avoir effectué une résolution de problème, il faut impérativement relancer mysql si non les configurations ne seront pas prise en compte et les commandes suivantes ne fonctionnent pas.

On se met sur la base de donnée “ gsb\_frais ”

```
Database changed
MariaDB [gsb_frais]> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0,004 sec)

MariaDB [gsb_frais]>
```

On tape “ FLUSH TABLES WITH READ LOCK; ” pour bloquer l’écriture le temps de la configuration de la réPLICATION. On note les champs “ file ” et “ position ” pour plus tard

```
MariaDB [(none)]> FLUSH TABLES WITH READ LOCK;
Query OK, 0 rows affected (0,002 sec)

MariaDB [(none)]> show master status;
+-----+-----+-----+-----+
| File      | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000001 |      328 | gsb_frais    |                |
+-----+-----+-----+-----+
1 row in set (0,000 sec)
```

## Configurer le fichier /etc/mysql/mariadb.conf.d/50-server.cnf - **SLAVE**

- server-id 2
- Décommenter la ligne #max\_binlog\_size = 100M
- Rajouter la ligne master-retry-count = 20
- Rajouter la ligne replicate-do-db = gsb\_frais

```
sio@hdintralabFLS: ~
GNU nano 5.4          /etc/mysql/mariadb
# Broken reverse DNS slows down connectio
# safe to skip if there are no "host by d
#skip-name-resolve

# Instead of skip-networking the default
# localhost which is more compatible and
#bind-address          = 127.0.0.1
```

## Étape 6 : Testing

Pour effectuer tout les test, j'ai utilisé le compte de Bedos Christian :

### Bedos Christian

#### Visiteur médical

Accueil  
Se déconnecter  
Saisie fiche de frais  
Mes fiches de frais

J'ai stoppé le serveur Master car il était en Current DC

The screenshot shows the vSphere Web Client interface for the 'intralabFLS' virtual machine. The top navigation bar includes icons for power, clone, edit, and settings, followed by 'ACTIONS ▾'. Below the bar, tabs for 'Résumé' (selected), 'Surveiller', 'Configurer', 'Autorisations', and 'Banques de données' are visible. The 'Résumé' tab displays the following information:

Hors tension	SE invité : Debian GNU/Linux 10 (64-bit)
	Compatibilité : ESXi 6.7 et versions ultérieures (VM version 1)
	VMware Tools : Inactif, version :11333 (Invité géré)
	<a href="#">Plus d'infos</a>
	Nom DNS : intralabFLS
	Adresses IP :
	Hôte :

At the bottom left, there are links to 'Lancer la console Web' and 'Lancer Remote Console', along with a Linux icon.

Puis dans la section “ saisie fiche de frais ”, j'ai pu en saisir une :

Descriptif des éléments hors forfait

Date	Libellé	Montant
<b>Nouvel élément hors forfait</b>		
* Date : <input type="text" value="12/02/24"/>		
* Libellé : <input type="text" value="TEST"/>		
* Montant : <input type="text" value="25"/>		

Après avoir rallumer le Master, l'enregistrement de la fiche de frais est bien visible sur le Master. La réPLICATION est bien fonctionnel dans les deux sens :

Non sécurisé https://gestionfraisinalabfls.gsb.coop/cSaisieFicheFrais.php

pour le collège Nouveaux Bot discord Tahl Jeux Coloriages Puzz... Mandalas Noms de domaine OptiFine\_1.17.1\_HD... LE DICTIONNAIRE ... Exercice avec des le... Réponse à - @susu... Le Coq Sportif Bask... Internet Live S...

 Suivi du remboursement des frais

Renseigner ma fiche de frais du mois de Avril 2025

Les modifications de la fiche de frais ont bien été enregistrées

Eléments forfaitisés

* Forfait Etape : <input type="text" value="0"/>
* Frais Kilométrique : <input type="text" value="0"/>
* Nuitée Hôtel : <input type="text" value="0"/>
* Repas Restaurant : <input type="text" value="0"/>

Descriptif des éléments hors forfait

Date	Libellé	Montant	
2025-02-12	TEST	25.00	<input type="button" value="Supprimer"/>
<b>Nouvel élément hors forfait</b>			
* Date : <input type="text" value="12/02/25"/>			
* Libellé : <input type="text" value="TEST"/>			
* Montant : <input type="text" value="25"/>			

  Cette page est conforme aux standards du Web

Active Windows  
Accès aux paramètres

## Mode opératoire Nagios :

### ÉTAPE 1 : Mise à jour du système.

Cette commande met à jour la base de données des paquets disponibles et installe les dernières versions

```
apt update && apt upgrade -y
```

### ÉTAPE 2 : Installation de Nagios et des composants nécessaires

Nagios4 : moteur de supervision principal

nagios-plugins-contrib : ensemble de plugins pour surveiller les services (ping, http, etc.)  
nagios-nrpe-plugin : permet d'interroger des machines distantes via NRPE  
apache2 : serveur web utilisé pour afficher l'interface Nagios

```
apt install nagios4 nagios-plugins-contrib nagios-nrpe-plugin apache2 -y
```

### ÉTAPE 3 : Activation du service Nagios

Pour qu'il démarre automatiquement à chaque démarrage de la machine

```
systemctl enable nagios4
```

### ÉTAPE 4 : Démarrage immédiat du service Nagios

```
systemctl start nagios4
```

### ÉTAPE 5 : Création d'un utilisateur Nagios pour l'interface web.

L'utilisateur "toor" est celui utilisé pour se connecter à l'interface web. Le mot de passe est demandé après cette commande

```
htpasswd -c /etc/nagios4/htpasswd.users toor
```

## ÉTAPE 6 : Vérification de la validité de la configuration Nagios

Cette étape permet de s'assurer qu'il n'y a pas d'erreurs dans les fichiers de configuration

```
nagios4 -v /etc/nagios4/nagios.cfg
```

## ÉTAPE 7 : Redémarrage d'Apache

Pour prendre en compte les fichiers de configuration web de Nagios

```
systemctl restart apache2
```

L'interface web de Nagios est maintenant disponible à l'adresse suivante :

```
http://172.18.155.86/nagios4
```

Utilisation du l'utilisateur "toor" et le mot de passe défini précédemment pour se connecter.

## ÉTAPE 8 : Ajout d'un hôte à surveiller

On crée un fichier de configuration personnalisé pour un hôte nommé "sup" pour les "tâches" de supervision souhaitée.

```
nano /etc/nagios4/conf.d/sup.cfg
```

```
define host {
    use          generic-host
    host_name    MASTER
    alias        Noeud Principale
    address      172.18.155.82
    max_check_attempts 5
    check_period   24x7
    notification_interval 30
    notification_period 24x7
}
```

```
define host {
    use          generic-host
    host_name    SLAVE
    alias        Noeud Secondaire
    address      172.18.155.83
    max_check_attempts 5
    check_period   24x7
    notification_interval 30
```

```

notification_period 24x7
}

define host {
    use          generic-host
    host_name    IP_FAILOVER
    alias        IP flottante HA
    address      172.18.155.84
    max_check_attempts 5
    check_period 24x7
    notification_interval 30
    notification_period 24x7
}

define service {
    use          generic-service
    host_name    MASTER,SLAVE,IP_FAILOVER
    service_description Ping
    check_command  check_ping!100.0,20%!500.0,60%
}

define service {
    use          generic-service
    host_name    MASTER,SLAVE
    service_description Apache HTTP
    check_command  check_http -H $HOSTADDRESS$ 
}

define service {
    use          generic-service
    host_name    MASTER,SLAVE
    service_description MySQL distant
    check_command  check_mysql!-H $HOSTADDRESS$ -u nagios -p tonmotdepasse
}

```

Créer un utilisateur nagios sur chaque serveur MySQL avec accès distant :

```

CREATE USER 'nagios'@'%' IDENTIFIED BY 'tonmotdepasse';
GRANT USAGE ON *.* TO 'nagios'@'%';
FLUSH PRIVILEGES;

```

```
nano /etc/mysql/mysql.conf.d/mysqld.cnf
```

```
bind-address = 0.0.0.0
```

Le système de supervision mis en place via Nagios permet de surveiller en temps réel trois hôtes critiques :

le serveur principal (MASTER)

le serveur secondaire (SLAVE)

l'adresse IP flottante (IP\_FAILOVER) utilisée pour la haute disponibilité.

Pour chaque machine, un contrôle de connectivité réseau (ping) est effectué en continu. Sur les serveurs MASTER et SLAVE, les services essentiels sont également supervisés à distance : le bon fonctionnement du serveur web Apache (via des requêtes HTTP) ainsi que la disponibilité de MySQL (avec vérification d'accès distant authentifié). Ce dispositif garantit la détection rapide d'un dysfonctionnement réseau ou applicatif au sein du cluster HA.

### **ÉTAPE 9** : Redémarrage de Nagios pour appliquer les nouveaux paramètres

`systemctl restart nagios`

`systemctl restart apache2`

Activation des modules Apache nécessaires au bon fonctionnement de l'interface Nagios

`a2enmod cgi`

### **ÉTAPE 10** : Mise en place de HTTPS pour Nagios

Il suffit de rajouter dans le fichier SSL d'Apache fait précédemment :

`nano /etc/apache2/sites-available/appliGSB_ssl.conf`

```
ScriptAlias /nagios/cgi-bin /usr/lib/cgi-bin/nagios4
<Directory "/usr/lib/cgi-bin/nagios4">
    Options ExecCGI
    AllowOverride None
```

```
Require all granted
</Directory>

Alias /nagios /usr/share/nagios4/htdocs
<Directory "/usr/share/nagios4/htdocs">
    Options None
    AllowOverride None
    Require all granted
    AuthType Basic
    AuthName "Nagios Access"
    AuthUserFile /etc/nagios4/htpasswd.users
    Require valid-user
</Directory>
```

#### **Vérification :**

- Accédez à <https://172.18.155.86/nagios4>
- Connectez-vous avec "nagiosadmin"
- Allez dans "Current Status" > "Hosts" pour voir l'état de "monserveur"
- Allez dans "Services" pour voir l'état du ping

Le système Nagios est maintenant opérationnel.

#### **Annexe :**

## Schéma infrastructure GSB :

