Guide d'Exploitation Infrastructure IT

Table des Matières

---

1. Vue d'ensemble

1.1 Architecture

- ***Serveur GLPI/OCS :*** *GESTPARC (192.168.100.222)*
- ***Load Balancer :*** *HAProxy (172.20.30.51)*
- ***Serveurs Backend :*** *Bush (172.20.30.49:8080), Roosevelt (172.20.30.50:8080)*
- ***Réseau :*** *LACP pour redondance*

1.2 Prérequis

- ***OS :*** *Debian 10/11*
- ***RAM :*** *4-8 GB*
- ***Stockage :*** *50 GB OS + 500 GB données*

---

2. Sauvegarde et Restauration

2.1 Stratégie

- ***Complète :*** *Dimanche (rétention 4 semaines)*
- ***Incrémentale :*** *Lundi-Samedi (rétention 1 semaine)*
- ***BDD :*** *Toutes les 6h*

2.2 Script de Sauvegarde

```
#!/bin/bash
```

```
# /usr/local/bin/sauvegarde_complete.sh

DATE=$(date +%Y-%m-%d)
TIMESTAMP=$(date +%Y-%m-%d_%H-%M-%S)
BACKUP_BASE="/dirbackup"
LOG_FILE="/var/log/backup/backup_${DATE}.log"

mkdir -p ${BACKUP_BASE}/{mysql,files,configs}

# Fichiers système
tar -czf ${BACKUP_BASE}/files/system_${TIMESTAMP}.tar.gz \
    /etc /var/www /usr/local/bin /home \
    --exclude=/var/www/html/glpi/files/_sessions

# Bases de données
mysqldump --user=glpibdd --password=m2l4 --single-transaction \
    glpi | gzip > ${BACKUP_BASE}/mysql/glpi_${TIMESTAMP}.sql.gz

mysqldump --user=ocsbdd --password=m2l4 --single-transaction \
    ocsweb | gzip > ${BACKUP_BASE}/mysql/ocsweb_${TIMESTAMP}.sql.gz

# Configurations
tar -czf ${BACKUP_BASE}/configs/configs_${TIMESTAMP}.tar.gz \
    /etc/apache2 /etc/mysql /etc/haproxy /etc/network

# Nettoyage
```

find ${BACKUP_BASE} -name ".tar.gz" -mtime +30 -delete*

find ${BACKUP_BASE} -name ".sql.gz" -mtime +7 -delete*

2.3 Automatisation Cron

 # /etc/crontab

*0 2 * * 0 root /usr/local/bin/sauvegarde_complete.sh*

*0 */6 * * * root /usr/local/bin/backup_db_only.sh*

---

3. GLPI/OCS Inventory

3.1 Installation Base Système

 # Réseau

nano /etc/network/interfaces

auto enp0s3

iface enp0s3 inet static

address 192.168.100.222

netmask 255.255.255.0

gateway 192.168.100.254

echo "GESTPARC" > /etc/hostname

 # LAMP Stack

apt update && apt upgrade -y

apt install apache2 php7.4 mariadb-server -y

```
# Extensions PHP
apt install php7.4-mysql php7.4-mbstring php7.4-curl php7.4-gd \
    php7.4-xml php7.4-intl php7.4-zip php7.4-ldap php7.4-soap -y

systemctl enable apache2 mariadb
```

3.2 Configuration MariaDB

```
mysql -u root -p

CREATE DATABASE glpi CHARACTER SET utf8 COLLATE utf8_unicode_ci;
CREATE USER 'glpibdd'@'localhost' IDENTIFIED BY 'MotDePasseSecurise2024!';
GRANT ALL PRIVILEGES ON glpi.* TO 'glpibdd'@'localhost';

CREATE DATABASE ocsweb CHARACTER SET utf8 COLLATE utf8_unicode_ci;
CREATE USER 'ocsbdd'@'localhost' IDENTIFIED BY 'MotDePasseSecurise2024!';
GRANT ALL PRIVILEGES ON ocsweb.* TO 'ocsbdd'@'localhost';

FLUSH PRIVILEGES;
```

3.3 Installation GLPI

```
cd /tmp
wget https://github.com/glpi-project/glpi/releases/download/10.0.7/glpi-10.0.7.tgz
tar -xzf glpi-10.0.7.tgz
cp -R glpi /usr/share/
chown -R www-data:www-data /usr/share/glpi
chmod -R 755 /usr/share/glpi
```

 # Virtual Host

nano /etc/apache2/sites-available/glpi.conf

Configuration Apache :

```
<VirtualHost *:80>
    ServerName gestparc.local
    DocumentRoot /usr/share/glpi

    <Directory /usr/share/glpi>
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/glpi_error.log
    CustomLog ${APACHE_LOG_DIR}/glpi_access.log combined
</VirtualHost>
```

a2ensite glpi.conf

a2dissite 000-default.conf

systemctl reload apache2

Accès : *http://192.168.100.222/*

- ***BDD :*** *localhost, glpibdd, MotDePasseSecurise2024!, glpi*

3.4 Installation OCS Inventory

 # Dépendances

apt install apache2-dev libmariadbclient-dev build-essential -y

*apt install libxml-simple-perl libdbi-perl libdbd-mysql-perl *

libapache-dbi-perl libnet-ip-perl libsoap-lite-perl -y

 # Installation

cd /tmp

wget https://github.com/OCSInventory-NG/OCSInventory-ocsreports/releases/download/2.11.0/OCSNG_UNIX_SERVER_2.11.0.tar.gz

tar -xzf OCSNG_UNIX_SERVER_2.11.0.tar.gz

cd OCSNG_UNIX_SERVER_2.11.0/

./setup.sh

Configuration : `/etc/apache2/conf-available/z-ocsinventory-server.conf`

PerlSetEnv OCS_DB_HOST localhost

PerlSetEnv OCS_DB_NAME ocsweb

PerlSetEnv OCS_DB_USER ocsbdd

PerlSetVar OCS_DB_PWD MotDePasseSecurise2024!

a2enconf z-ocsinventory-server

a2enconf ocsinventory-reports

systemctl restart apache2

Accès : *http://192.168.100.222/ocsreports (admin/admin)*

3.5 Agents Windows

OCS Agent :

1. *Éditer* `C:\Windows\system32\drivers\etc\hosts`

2. *Ajouter :* `192.168.100.222 GESTPARC`
3. *Installer agent avec URL :* `https://GESTPARC/ocsinventory`
4. *Modifier* `TTO_WAIT=10` *dans* `ocsinventory.ini`

FusionInventory :

1. *Installation complète depuis fusioninventory.org*
2. *Mode Service Windows*
3. *Configuration serveur HTTP intégré*

---

4. HAProxy Load Balancer

4.1 Installation

apt install haproxy -y

systemctl enable haproxy

nano /etc/default/haproxy

ENABLED=1

4.2 Configuration

Fichier : `/etc/haproxy/haproxy.cfg`

```
global
    log 127.0.0.1:514 local0
    chroot /var/lib/haproxy
    stats socket /run/haproxy/admin.sock mode 660 level admin
    user haproxy
    group haproxy
    daemon

defaults
```

```
    mode http
    log global
    option httplog
    option forwardfor
    timeout connect 5000
    timeout client 50000
    timeout server 50000

# Statistiques
stats enable
stats uri /haproxy-stats
stats auth admin:StatsPassword2024

# Frontend
frontend web-frontend
    bind 172.20.30.51:80
    option httplog
    default_backend web-servers

# Backend
backend web-servers
    balance roundrobin
    option httpchk GET /health
    server Bush 172.20.30.49:8080 check inter 3000
    server Roosevelt 172.20.30.50:8080 check inter 3000
```

4.3 Scripts de Gestion

Maintenance serveur :

```
#!/bin/bash

# /usr/local/bin/haproxy_maintenance.sh


ACTION=$1

SERVER=$2

SOCKET="/run/haproxy/admin.sock"


case $ACTION in

   "enable")

      echo "disable server web-servers/$SERVER" | socat stdio $SOCKET

      ;;

   "disable")

      echo "enable server web-servers/$SERVER" | socat stdio $SOCKET

      ;;

esac


echo "show stat" | socat stdio $SOCKET | grep web-servers
```

---

5. LACP (Link Aggregation)

5.1 Configuration Switch Cisco

```
enable

configure terminal
```

interface range gigabitEthernet 1/0/1-2

channel-group 1 mode active

channel-protocol lacp

interface port-channel 1

description "Liaison LACP"

switchport mode trunk

5.2 Configuration Linux

Fichier : `/etc/network/interfaces`

auto bond0

iface bond0 inet static

address 172.20.30.51

netmask 255.255.255.0

gateway 172.20.30.1

bond-mode 802.3ad

bond-miimon 100

bond-lacp-rate fast

bond-slaves eth0 eth1

auto eth0

iface eth0 inet manual

bond-master bond0

auto eth1

iface eth1 inet manual

```
    bond-master bond0
```

5.3 Vérification

```
# État du bond
cat /proc/net/bonding/bond0

# Statistiques
ip -s link show bond0

# Test connectivité
ping -I bond0 172.20.30.1
```

---

6. Monitoring

6.1 Script de Surveillance Global

```
#!/bin/bash
# /usr/local/bin/monitor_infrastructure.sh

LOG_FILE="/var/log/infrastructure_monitor.log"
EMAIL_ADMIN="admin@company.com"

# Apache
systemctl is-active apache2 >/dev/null || {
    echo "[$(date)] Apache2 redémarré" >> $LOG_FILE
    systemctl restart apache2
```

```
}

# MariaDB
systemctl is-active mariadb >/dev/null || {
    echo "[$(date)] MariaDB redémarré" >> $LOG_FILE
    systemctl restart mariadb
}

# GLPI
curl -s http://localhost/glpi >/dev/null || {
    echo "[$(date)] GLPI inaccessible" >> $LOG_FILE
    echo "ALERTE GLPI" | mail -s "Alert" $EMAIL_ADMIN
}

# HAProxy
pgrep haproxy >/dev/null || {
    echo "[$(date)] HAProxy redémarré" >> $LOG_FILE
    systemctl restart haproxy
}

# Espace disque
USAGE=$(df /var | tail -1 | awk '{print $5}' | sed 's/%//')
if [ $USAGE -gt 85 ]; then
    echo "[$(date)] Disque critique: $USAGE%" >> $LOG_FILE
    echo "Disque /var à $USAGE%" | mail -s "Disk Alert" $EMAIL_ADMIN
fi
```

6.2 Automatisation

 # Crontab monitoring

**/5 * * * * /usr/local/bin/monitor_infrastructure.sh*

*0 8 * * * /usr/local/bin/daily_report.sh*

---

7. Maintenance

7.1 Maintenance Hebdomadaire

#!/bin/bash

 # /usr/local/bin/maintenance_weekly.sh

LOG_FILE="/var/log/maintenance.log"

echo "[$(date)] Début maintenance" >> $LOG_FILE

 # Nettoyage logs

find /var/log -name ".log" -mtime +30 -delete*

find /var/log -name ".gz" -mtime +90 -delete*

 # Optimisation MySQL

mysql -u root -p <<EOF

OPTIMIZE TABLE glpi.glpi_logs;

OPTIMIZE TABLE ocsweb.hardware;

EOF

 # Cache GLPI

*rm -rf /usr/share/glpi/files/_cache/**

*rm -rf /usr/share/glpi/files/_sessions/**

echo "[$(date)] Maintenance terminée" >> $LOG_FILE

7.2 Mise à Jour GLPI

#!/bin/bash

GLPI_VERSION="10.0.8"

BACKUP_DIR="/backup/glpi_upgrade"

 # Sauvegarde

mkdir -p $BACKUP_DIR

cp -R /usr/share/glpi $BACKUP_DIR/glpi_$(date +%Y%m%d)

mysqldump -u glpibdd -pm2l4 glpi > $BACKUP_DIR/glpi_$(date +%Y%m%d).sql

 # Arrêt Apache

systemctl stop apache2

 # Mise à jour

cd /tmp

wget https://github.com/glpi-project/glpi/releases/download/${GLPI_VERSION}/glpi-${GLPI_VERSION}.tgz

tar -xzf glpi-${GLPI_VERSION}.tgz

rsync -av --exclude=config --exclude=files glpi/ /usr/share/glpi/

 # Mise à jour BDD

cd /usr/share/glpi

php bin/console glpi:database:update

 # Redémarrage

systemctl start apache2

---

8. Dépannage

8.1 Problèmes Courants

GLPI/OCS inaccessible :

 # Vérifier services

systemctl status apache2 mariadb

 # Logs

tail -f /var/log/apache2/error.log

tail -f /var/log/mysql/error.log

 # Redémarrage

systemctl restart apache2 mariadb

HAProxy en erreur :

 # Test configuration

haproxy -c -f /etc/haproxy/haproxy.cfg

 # État backends

echo "show stat" | socat stdio /run/haproxy/admin.sock

 # Redémarrage

systemctl restart haproxy

LACP défaillant :

 # État bond

cat /proc/net/bonding/bond0

 # Switch Cisco

show etherchannel summary

show lacp neighbor

 # Réinitialisation

ifdown bond0 && ifup bond0

8.2 Logs Importants

 # Apache

/var/log/apache2/error.log

/var/log/apache2/access.log

 # MySQL

/var/log/mysql/error.log

 # HAProxy

/var/log/haproxy.log

 # Système

/var/log/syslog

/var/log/messages

8.3 Commandes Diagnostic

 # État des services

systemctl status apache2 mariadb haproxy

 # Processus

ps aux | grep -E "(apache|mysql|haproxy)"

 # Réseau

netstat -tlnp | grep -E "(80|443|3306)"

ss -tlnp

 # Charge système

top

htop

iotop

 # Espace disque

df -h

*du -sh /var/log/**

8.4 Contacts d'Escalade

***Niveau 1 :** Équipe technique locale **Niveau 2 :** Administrateurs systèmes **Niveau 3 :** Éditeurs (GLPI, OCS)*

Document d'exploitation - Version 1.0
Dernière mise à jour : Juin 2025
Pages : 30