

# EzPaarse - Le serveur de production

## L'adresse du serveur :

ezpaarse.univ-lille.fr

<https://wikis.univ-lille.fr/dsi-exploit/dom-doc/vms/vm-udl/serveur-ezpaarse>

## Compte :

le compte de travail est ezpaarse (connexion par cle ssh)

## Chemins :

l'arborescence des répertoires à partir de /home/ezpaarse/

```
.
├── DS
│   ├── EC-CSV
│   └── LOGS
├── SHS
│   ├── EC-CSV
│   └── LOGS
├── ST
│   ├── EC-CSV
│   └── LOGS
├── ULille
│   ├── EC-CSV
│   └── LOGS
├── rapport
├── archives
└── scripts
```

## Workflow de traitement :

Tous les scripts sont dans le répertoire scripts/

### A 01h00 mn du matin :

-Ezpaarse se connecte au 4 serveurs ezproxy pour aller récupérer les logs de la veille

## Ensuite

-ecbulkmaker est utilisé pour analyser les fichiers de logs présents dans les répertoire LOGS et générer les fichiers ezparse csv dans les répertoires EC-CSV

exemple d'appel :

```
/usr/local/ezparse/bin/ecbulkmaker -H "Log-Format-ezproxy:%h %l %u-
%{ezproxy-session}i %t \"%r\" %s %b" /home/ezparse/SHS/LOGS/
/home/ezparse/SHS/EC-CSV/
```

## Ensuite (a faire)

-analyse des résultats présents dans les fichiers rapport html / et recherche de l'éventuel crash avec l'existence d'un fichier .ko

- **Ensuite pour chaque rapport OK,**
- compression du fichier csv avec gzip
- **Ensuite pour chaque rapport NOK,**
- envoie d'un rapport d'échec à dsi-doc

## Ensuite

Téléversement du fichier dans EzMesure avec la fonction ezmesure

## Comment autoriser Ezparse à accéder en SFTP à un serveur EzProxy ?

Lors de la demande de la création de la VM Ezparse à la DSI système, il a été demandé l'autorisation pour le serveur EzParse d'accéder aux 4 serveurs EzProxy par le port 22 (SFTP/SSH) :

```
SRC: ezparse.univ-lille.fr
DST: ressources-electroniques.univ-lille.fr
PORT: 22
```

```
SRC: ezparse.univ-lille.fr
DST: proxy.scd.univ-lille3.fr
PORT: 22
```

```
SRC: ezparse.univ-lille.fr
DST: bu-ezproxy.univ-lille1.fr
PORT: 22
```

```
SRC: ezparse.univ-lille.fr
DST: doc-distant.univ-lille2.fr
PORT: 22
```

Le principe général :

- Le compte ezpaarse du serveur ezpaarse va se connecter en SFTP sur chacun des serveurs Ezproxy pour récupérer les fichiers logs.
- Un compte ezpaarse sur chaque serveur ezproxy va contenir la clé ssh public du compte ezpaarse du serveur ezpaarse.
- Ce compte ezpaarse sur chaque serveur ezproxy doit avoir juste des droits de lecture vers le chemin qui mène au logs de ezproxy.

Il faut sur chaque serveur **EZPROXY** créer un compte ezpaarse, et y mettre la clé public au bon endroit avec les bons droits

```
# en mode sudo
sudo -i

# créer l'utilisateur ezpaarse avec sa home (-m)
useradd -m ezpaarse

# aller dans sa home
cd /home/ezpaarse

# créer le répertoire .ssh
mkdir .ssh

# faire de ezpaarse le propriétaire de .ssh
chown ezpaarse .ssh

# limiter les droits en lecture seule/exécution sur le répertoire
# (obligatoire pour une connexion ssh par clé)
chmod 700 .ssh

# créer le fichier avec la clé public
cd .ssh
echo ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDTaN/nTATQf0r9KaEut4quvQ3lwtk5XJYLpvn9FrS9GVGI
0YRNPYjb6xaNoqCq2FeTZe8wjQv1SskBijg5mgCuDyfp5+/JgaHqEC4PfA9ZzIYBqnsGsZ5hn10X
GUJ5s55aa4Qo8fV0u5Pn9uEIKz0u0hcsxoy3nC6G6SAC4W9zrdzwILAFYDBE2iUD6gtLvtW088ky
EFExnq5afs37wTEAOZjotZhbLYDA/b+gHK0vy3F5J2iRgAwsxKhhS3kft0GqALEAFYFdT5z0pwpZ
jrxj8yQx8FegvpUpSbj2NoEn/zVeAc7yavYIMxRJwjfE4UbEoijUkrD4yI5xVhYKBDJ/
ezpaarse@ezpaarse> authorized_keys

# limiter les droits en lecture sur la clé (obligatoire pour une connexion
ssh par clé)

chown ezpaarse authorized_keys
chmod 600 authorized_keys
```

## Stratégie d'accès aux fichiers de logs

Nous n'allons pas toucher au droits d'accès aux logs de ezproxy là où ils sont.

Nous allons automatiser la recopie compressée des logs de la veille dans le repertoire /home/ezparse

Etape 1 sur serveur **EZPROXY** :

- **Ezproxy doit être configuré pour générer des logs journaliers**
- **Ne pas avoir de logrotate sur le répertoire logs de ezproxy**

Exemple de ligne de configuration (config.txt de ezproxy) qui va permettre à ezproxy de générer un fichier de log par jour (ici dans le repertoire 'logs'):

```
LogFile -strftime logs/ezp%Y%m%d.log
```

Etape 2 sur serveur **EZPROXY** :

en mode root (**sudo -i**) , on va créer un script sur **/root/** : **recopie.sh** avec ce contenu dont il faudra adapter les variables (à priori seulement la variable CHEMINLOG)

```
#!/bin/bash

PREFIX="ezp"
SUFFIX=".log"
CHEMINLOG="/opt/ezproxy/logs/"
CHEMINEZPARSE="/home/ezparse/"

DATE=`date +%Y%m%d --date '1 days ago'`

FICHER=$PREFIX$DATE$SUFFIX

echo "Compression du fichier $FICHER"
echo "-----"
gzip $CHEMINLOG$FICHER

echo "Deplacement du fichier $FICHER.gz vers $CHEMINEZPARSE"
echo "-----"
mv $CHEMINLOG$FICHER.gz $CHEMINEZPARSE

echo "Changement de propriétaire du fichier compressé pour ezparse"
echo "-----"
chown ezparse $CHEMINEZPARSE$FICHER.gz

echo "-----Fin-----"
```

Les variables globales sont accès explicites pour comprendre comment les renseigner (ici figure la version du serveur ezproxy de Lille SHS)

le script fait cela dans l'ordre :

1. Compression du fichier de log de la veille en .gz
2. Déplacement du fichier compressé vers /home/ezpaarse
3. Changement des droits du fichier pour que ezpaarse ait le droit de lire le fichier

Etape 3 sur serveur **EZPROXY** :

programmer le lancement du script tous les jours à 00H15 :

**en root (sudo -i)**

```
crontab -e
```

rendre le fichier recopie.sh executable

```
chmod +x recopie.sh
```

Ajouter ces lignes dans le crontab :

```
# m h dom mon dow  command
15 00 * * * /root/recopie.sh
```

De ce fait, le serveur ezpaarse ira chercher le fichier de la veille dans /home/ezpaarse et n'a pas besoin de droit supplémentaire sur les serveurs proxy.

Par voie de conséquence, on fait un équivalent de logrotate avec compression des fichiers de logs pour qu'ils prennent moins de place.

@prévoir : automatiser l'archivage sur un autre serveur des logs plus vieux de 30 jours : peut-être agrandir l'espace disque du serveur ezpaarse pour que ce soit lui qui héberge les logs sur 1 an avant suppression définitive.

## Automatisation de la chaine de traitement EzProxy -> EzPaarse -> EzMeasure

Sur le serveur **EzPaarse**, il y a un répertoire **/home/ezpaarse/scripts**

**On y trouve plusieurs fichiers :**

Nom du fichier	contenu
ez-proxy-paarse-measure.pl	programme PERL qui organise tous les traitements pour 1 serveur proxy: scp(recupération sftp des logs), ecbulkmaker(traitement des logs par ezpaarse), ezmeasure(injection des fichiers resultats dans EzMsure par webservice)
lilleshs.conf	Fichier de configuration utiliser par le programme Perl pour le traitements des logs de Lille SHS
lilleds.conf	Fichier de configuration utiliser par le programme Perl pour le traitements des logs de Lille DS
lillest.conf	Fichier de configuration utiliser par le programme Perl pour le traitements des logs de Lille ST

Nom du fichier	contenu
ulille.conf	Fichier de configuration utiliser par le programme Perl pour le traitements des logs de ULille
traitement-auto.sh	Programme BASH qui va enchaîner les 4 appels au programme Perl :1 appel par serveur proxy avec sa configuration.  Ce Programme est dans le crontab du compte ezparse : lancement toutes les jours à 1h du matin
—	

Contenu du fichier de conf de SHS : lilleshs.conf

```
serveur=proxy.scd.univ-lille3.fr
chemin=/home/ezparse/
prefixe_fichier_log=ezpDATE.log.gz
prefixe_fichier_ec=ezpDATE.ec.csv
chemin_local_LOGS=/home/ezparse/SHS/LOGS/
chemin_local_EC=/home/ezparse/SHS/EC-CSV/
index_ezmesure=univ-lille-shs
format_log="Log-Format-ezproxy:%h %l %u-%{ezproxy-session}i %t \"%r\" %s %b"
```

Contenu du fichier de conf de DS : lilleds.conf

```
serveur=doc-distant.univ-lille2.fr
chemin=/home/ezparse/
prefixe_fichier_log=ezpDATE.log.gz
prefixe_fichier_ec=ezpDATE.ec.csv
chemin_local_LOGS=/home/ezparse/DS/LOGS/
chemin_local_EC=/home/ezparse/DS/EC-CSV/
index_ezmesure=univ-lille-ds
format_log="Log-Format-ezproxy:%h %{ezproxy-groups}i<.*> %l %u %t \"%r\" %s %b"
```

Contenu du fichier de conf de ST : lillest.conf

```
serveur=bu-ezproxy.univ-lille1.fr
chemin=/home/ezparse/
prefixe_fichier_log=ezpDATE.log.gz
prefixe_fichier_ec=ezpDATE.ec.csv
chemin_local_LOGS=/home/ezparse/ST/LOGS/
chemin_local_EC=/home/ezparse/ST/EC-CSV/
index_ezmesure=univ-lille-st
format_log="Log-Format-ezproxy:%h \"%{ezproxy-groups}i<.*>\" %u %t \"%r\" %s %b \"%{Referer}i<.*>\" \"%{user-agent}i<.*>\" \"%{Cookie}i<.*>\" %{ezproxy-session}i<.*>"
```

Contenu du fichier de conf de Ulille : ulille.conf

```

serveur=ressources-electroniques.univ-lille.fr
chemin=/home/ezpaarse/
prefixe_fichier_log=ezpDATE.log.gz
prefixe_fichier_ec=ezpDATE.ec.csv
chemin_local_LOGS=/home/ezpaarse/ULille/LOGS/
chemin_local_EC=/home/ezpaarse/ULille/EC-CSV/
index_ezmesure=univ-lille
format_log="Log-Format-ezproxy:%h %l %{session-uid}i %{session-idp}i<.*>
%{ezproxy-session}i %t \"%r\" %s %b"

```

Explications :

**serveur** =nom du serveur proxy

**chemin** =sur le serveur ezproxy , chemin où le compte ezpaarse trouvera le fichier de log à récupérer en sftp

**prefixe\_fichier\_log** =comment le fichier de log est nommé exemple : ezpDATE.log.gz Attention **DATE** est un mot clé qui sera remplacé par la date comme cela **yyyymmjj** : ce qui donne ici : ezp20180328.log.gz

**prefixe\_fichier\_ec** =idem pour le fichier EC généré par le traitement EzPaarse : ezpDATE.ec.csv devient ezp20180328.ec.csv lors du traitement

**chemin\_local\_LOGS** =sur le serveur ezpaarse, chemin où mettre le fichier de log généré

**chemin\_local\_EC** =sur le serveur ezpaarse, chemin où mettre le fichier de EC CSV généré par le traitement ezpaarse

**index\_ezmesure** =nom de l'index Elasticsearch du serveur EzMesure national où il faut injecter les événements du fichier EC-CSV (par le webservice)

**format\_log** =chaîne transmis en paramètre à EzPaarse pour décrire le format de log du fichier ezproxy à traiter

**Attention particularité de Ulille :** voilà la vraie ligne de description du LogFormat dans EzProxy :

```
LogFormat %h %l %{session:uid}e %{session:idp}e %{ezproxy-session}i %t "%r"
%s %b
```

Et voilà sa transposition dans EzPaarse :

```
"Log-Format-ezproxy:%h %l %{session-uid}i %{session-idp}i<.*> %{ezproxy-
session}i %t \"%r\" %s %b"
```

- Les attributs générés par les règles **SET** de ezproxy et qui sont déclarés avec un nom qui contient ":" comme **{session:idp}** ont été écrit avec un **tiret** à la place des : → **sinon ezpaarse ne comprend pas l'expression**
- De même le **i** a été remplacé par un **e**

Contenu du fichier bash d'enchaînement des 4 traitements (1 par serveur ezproxy) : **traitement-auto.sh**

La première ligne précise l'environnement de travail pour que le moteur JS utilisé soit la version 6 fournie avec EzPaarse plutôt que la version 4.8 en standard sous Debian( sinon l'injection ezmesure ne marche pas : attention, il y a un espace entre le . et le /usr)

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

```
cd /home/ezpaarse

. /usr/local/ezpaarse/bin/env
# traitement pour Lille DS
perl /home/ezpaarse/scripts/ez-proxy-paarse-mesure.pl
/home/ezpaarse/scripts/lilleds.conf> /home/ezpaarse/rapport/rapport-ds-`date
+%Y-%m-%d-%H%M`.txt

# traitement pour Lille ST
perl /home/ezpaarse/scripts/ez-proxy-paarse-mesure.pl
/home/ezpaarse/scripts/lillest.conf> /home/ezpaarse/rapport/rapport-st-
`date +%Y-%m-%d-%H%M`.txt

#traitement pour Lille SHS
perl /home/ezpaarse/scripts/ez-proxy-paarse-mesure.pl
/home/ezpaarse/scripts/lilleshs.conf> /home/ezpaarse/rapport/rapport-shs-
`date +%Y-%m-%d-%H%M`.txt

#traitement pour ULille
#perl> /home/ezpaarse/scripts/ez-proxy-paarse-mesure.pl
/home/ezpaarse/scripts/ulille.conf> /home/ezpaarse/rapport/rapport-ulille-
`date +%Y-%m-%d-%H%M`.txt
```

**Les rapports de traitement** sont écrits dans le répertoire **rapport/**

Le lien vers le programme Perl est ici :

[ezproxy-chaine-de-traitement.zip](#)

Le programme Perl peu aussi être appelé avec plusieurs paramètres .

Il lui faut au moins 1 paramètre : le fichier de conf  
Et sans plus de paramètre il ira chercher le fichier de la veille.

Sinon, on peut ajouter en paramètre un jour précis qu'il ira chercher à la place du fichier de la veille.

exemple pour qu'il travaille avec le fichier de la veille:

```
perl ez-proxy-paarse-mesure.pl lilleds.conf
```

exemple pour qu'il travaille avec le fichier du 27 mars 2018:

```
perl ez-proxy-paarse-mesure.pl lilleds.conf 2018 03 27
```

Crontab du serveur Ezpaarse exécuté par le compte ezpaarse :

```
00 01 * * * /home/ezpaarse/scripts/traitement-auto.sh
```



```
00 14 15 * * /home/ezpaarse/scripts/archiver.sh
```

Tous les jours à 1h00 du matin : traitement des logs des serveurs ezproxy et injection dans EzMesure  
→ script traitement-auto.sh

Le 15 du mois : les fichiers LOGS et EC-CSV du **mois précédent** sont déplacés dans le répertoire archives (les fichiers EC-CSV sont compressés avant déplacement) →script achiver.sh

From:

<https://wikis.univ-lille.fr/dsi-exploit/> - **DSI - Exploitation / Intégration**

Permanent link:

<https://wikis.univ-lille.fr/dsi-exploit/dom-doc/ezproxy/ezpaarse/ezpaarse-le-serveur>

Last update: **2018/05/21 17:37**