Manuel d'installation

Version 4.5.2

Libriciel SCOP

Document

Auteur	Stéphane VAST	***Date de diffusion***	09/03/18
Chef de projet	Stéphane VAST	***N° de version***	4.5.2

Évolution du document

Version	Auteur	Nature des changements	Date
1.0	Stéphane Vast	Adaptations pour nouvelle charte	25/08/2017
1.1	Stéphane Vast	Publication officielle	29/09/2017
1.2	Lukas Hameury	Modifications pour version 4.5.1	14/12/2017
1.3	Lukas Hameury	Modifications pour version 4.5.2	09/03/2018

Licence

Libriciel SCOP

Table des matières

1. PRÉAMBULE À L'INSTALLATION

Ce document est un guide d'installation-type, à destination de techniciens GNU/Linux expérimentés, à suivre et adapter selon l'environnement d'exploitation (système d'exploitation, infrastructure, etc.).

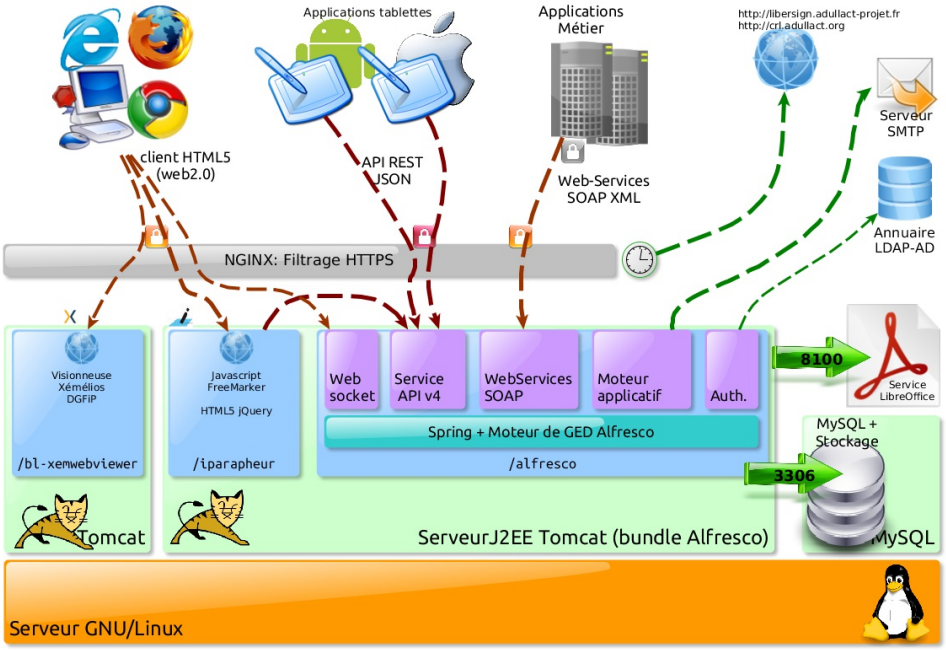
1.1. Champ d'application

Ce guide détaille l'installation de i-Parapheur sur plate-forme GNU/Linux en général, Ubuntu Server LTS en particulier.
Ci-après quelques (vieilles) ressources Web ayant servi de base à cette procédure :

- http://wiki.alfresco.com/wiki/Installing_Alfresco_on_Ubuntu_7.10 (du 29 avr.2008)

Le respect rigoureux des éléments d'intallation décrits ci-après aboutit à une instance fonctionnellement opérationnelle de l'application i-Parapheur.

1.2. Architecture logicielle serveur



Architecture et flux réseaux

Le logiciel i-Parapheur est installé avec les composants suivants :

- système d'exploitation GNU/Linux 64bit, installé en Français, configuré avec encodage UTF-8,
- serveur de base de données MySQL 5.1 ou plus récent,
- les utilitaires de back-office : 'unzip', 'tar', 'at', 'GhostScript', 'LibreOffice'
- serveur Web NginX avec les modules SSL , et donc OpenSSL
- accès à un serveur de messagerie SMTP, optionnellement une boite aux lettres POP3 dédiée
- Alfresco 3.4.c Community (livré en « Bundle » avec TOMCAT préconfiguré, etc.)

Le présent guide détaille l'installation des différents composants serveurs sur une seule machine.
Il est cependant possible de répartir la charge sur différentes plate-formes. L'installation répartie sur différents noeuds n'est pas couverte par ce document.

1.3. Tableau de flux réseaux

Ce chapitre pour énumérer les contraintes de flux sur le réseau entre client et serveur, afin d'ouvrir au besoin les bonnes portes sur les pare-feux, en particulier si le déploiement est réparti sur différents serveurs.

Depuis	Vers	Protocole	port TCP
Navigateur Web (IHM)	Frontal Web (IHM)	HTTPS	~~80,~~ 443
Navigateur Web (webSocket)	Frontal Web (webSocket)	HTTPS	~~80,~~ 443

Navigateur Web (Pes-v2)	Frontal Web (XWV)	HTTPS	~~80,~~ 443
Tablette iOS/Android	Frontal Web (mob.)	HTTPS	443
Application métier (GF...)	Frontal Web (WS)	HTTPS	443
Frontal Web (IHM)	Service IHM Web i-Parapheur	HTTPS	8080
Frontal Web (XWV)	Visionneuse Xémélios	HTTP	9080
Frontal Web (webSocket)	Coeur i-Parapheur	HTTP	8081
Frontal Web (mob.)	Coeur i-Parapheur	HTTP	8080
Frontal Web (WS)	Coeur i-Parapheur	HTTP	8080
Service IHM Web i-Parapheur	Coeur i-Parapheur	HTTP	8080
Coeur i-Parapheur	Service LibreOffice	« socket API »	8100
Coeur i-Parapheur	Service MySQL	« socket API »	3306
Coeur i-Parapheur	TDT externe	HTTPS	443
Coeur i-Parapheur	Frontal Web (XWV)	HTTP	80
Coeur i-Parapheur	Service de messagerie	SMTP	25
Coeur i-Parapheur	Service de messagerie	POP3	110
Coeur i-Parapheur	Annuaire AD / LDAP	LDAP	389
Crontab (récup CA-CRL)	Site crl.adullact.org	HTTP	80
Crontab (entretien LiberSign)	Site libersign.libriciel.fr	HTTPS	443
Postes clients : applet JAVA	Frontal Web (IHM)	HTTPS	443
Postes clients : LiberSign	Sites externes (vérif. CRL)	HTTP, OCSP	80

Ce tableau évoluera en fonction des évolutions techniques de l'application.

1.4. Plates-formes supportées

L'installation a été validée sur les systèmes d'exploitation suivants:

- **Ubuntu 16.04 Server LTS x64** (plate-forme de référence, sur laquelle une offre de « support long terme » est proposée par la société Canonical Ltd.),
- **Debian** 8.

Le serveur parapheur électronique peut également être installé sur d'autres GNU/Linux : Fedora/ CentOS/ RedHat, SUSE... sous réserve que les pré-requis logiciels soient respectés, et sous réserve de validation par les équipes techniques de Libriciel SCOP.

Se référer à la fiche technique des pré-requis à l'installation pour la liste précise des versions de systèmes d'exploitation supportées par Libriciel SCOP.

> *Remarque* : L'installation sur serveur Microsoft™ Windows n'est pas qualifiée. (Problème existant en particulier avec la librairie GhostScript qui n'y serait pas « threadSafe »).
> Idem pour la base de données PostGreSQL, disqualifiée pour le moment par rapport à MySQL.
> Toute contribution sur ce sujet est bienvenue, merci de prendre contact :contact@libriciel.coop .

2. PRÉ-REQUIS LOGICIELS – PLATE-FORME GNU/LINUX

> *RAPPEL*: Pendant l'installation, **le serveur doit être connecté à Internet** afin de récupérer et installer les dernières mises-à-jour des composants logiciels disponibles.

2.1. Système opérateur Ubuntu Server LTS

Localisation: **FR**
Attribuer un nom de machine (pas de `_` dans le nom) :

- *exemple* : `iparapheur.ma-collectivite.fr`

Sélection de logiciels: choisir a minima `OpenSSH server`.

Exemple de partitionnement (avec un espace global de 300Go) :

```
|    /       30G
|    swap     2G
|    /opt   200G   <--- partition LVM, où serait l'entrepôt
|    /var    70G   <--- ici serait la Base de Données
```

NB : Les binaires d'installation seront déposés par convention dans `/opt/_install`.
Si souci avec disposition de clavier : `sudo dpkg-reconfigure keyboard-configuration`

2.2. Configuration du système, connecté à Internet

Vérifier et mettre à jour les dépôts (repository) de logiciels avec les privilèges**administrateur** :

```
sudo -s
vi /etc/apt/sources.list
```

Commenter en préfixant avec le caractère :hash: (dièse) la ligne spécifiant le chemin du CD-ROM, et s'assurer de la présence des dépôts `universe` et `multiverse` (normalement déjà activés) :

```
#deb cdrom:[Ubuntu-Server 16.04 _Xenial Xerus_ - ...]/ trusty main restricted
deb     http://fr.archive.ubuntu.com/ubuntu/ xenial main restricted universe multiverse
deb-src http://fr.archive.ubuntu.com/ubuntu/ xenial main restricted universe multiverse
deb     http://fr.archive.ubuntu.com/ubuntu/ xenial-updates main restricted universe multiverse
deb-src http://fr.archive.ubuntu.com/ubuntu/ xenial-updates main restricted universe multiverse
deb     http://fr.archive.ubuntu.com/ubuntu/ xenial-security main restricted universe multiverse
deb-src http://fr.archive.ubuntu.com/ubuntu/ xenial-security main restricted universe multiverse
```

> Debian : activer les repos 'main' 'contrib', 'non-free'.
> Pour le dépôt Debian 'backports' (pour OOo) nouvelle ligne :
> `deb http://backports.debian.org/debian-backports/  wheezy-backports main contrib non-free`
> puis : `apt-get install debian-backports-keyring`

Mise à jour du système :

```
apt-get update
apt-get -s dist-upgrade        # Simulation
apt-get dist-upgrade           # Mise à jour
```

Naturellement: en cas de mise à jour du kernel (paquet `linux-image...`), redémarrer avec la commande `reboot`.

2.2.1. Réglage du SWAP

Faisons en sorte que le système Linux hôte ne swappe pas trop facilement, en modifiant sa politique de "swappiness".
Créer `/etc/sysctl.d/10-swappiness.conf` , y ajouter la directive suivante :

```
vm.swappiness=10
```

Puis activer (sans reboot nécessaire) avec la commande :

```
sysctl -p
```

Et vérifier que ça marche!! La commande suivante doit donner `10` en résultat:

```
cat /proc/sys/vm/swappiness
```

Si ce n'est toujours pas le cas: ajouter alors la directive `vm.swappiness=10` à la fin du fichier /etc/sysctl.conf, retenter l'activation avec sysctl -p, et revérifier.

2.2.2. Éléments de confort

> NB : éléments de confort pour le technicien (éditeur de texte, complétion automatique CLI) :

```
apt-get install vim-nox   # coloration syntaxique dans vi
vi /etc/bash.bashrc       # pour régler la complétion automatique
```

2.3. Installation des pré-requis logiciels

Base de données MySQL (sauf service externalisé), et quelques outils :

```
apt-get install mysql-server
```

> Si **RedHat/CentOS 6** :
> `yum install mysql-server ; chkconfig mysqld on ; service mysqld start`

> Si **RedHat/CentOS 7**, avec MariaDB (à la place de MySQL) :
> `yum install mariadb-server.x86_64`
> `systemctl enable mariadb.service`
> `systemctl start mariadb.service`
> `mysql_secure_installation`

```
apt-get install ntp xfonts-base psmisc unzip
apt-get install ghostscript gsfonts libxt6 libxi6 libxtst6
```

Polices de caractères[1] TTF standard pour GhostScript, et LibreOffice :

```
apt-get install ttf-mscorefonts-installer
```

> Si RedHat/CentOS: installer composants `xorg-x11-fonts-Type1`
> Voir annexe pour procédure d'installation des polices TTF Microsoft.

Installation du JDK8_u161 téléchargé préalablement depuis le site java.com de Oracle :

```
cd /opt ; chmod +x /opt/_install/jdk-8u161-linux-x64.bin
/opt/_install/jdk-8u161-linux-x64.bin
```

La commande `/opt/jdk1.8.0_161/bin/java -version` doit afficher :

```
java version "1.8.0_161"
Java(TM) SE Runtime Environment (build 1.8.0_161-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.161-b12, mixed mode)
```

Paramétrer la locale, pour être le plus possible en FR-UTF8, en ajoutant dans `/etc/profile` :

```
export LC_ALL=fr_FR.UTF-8
```

Dépaqueter l'archive d'installation préalablement déposée dans le répertoire `/opt/_install` :

```
cd /opt/_install ; tar xzf iParapheur-v4.5.xx.tar.gz
tar xzf iParapheur-v4.5.xx/confs.tar.gz
```

2.3.1. Service SMTP

L'application i-Parapheur envoie des e-mails de notification, il a donc besoin d'un MTA SMTP accessible.

En l'absence de service SMTP directement accessible , on peut en installer un, y vérifier (dans le fichier nommé `/etc/postfix/main.cf`) la variable `relayhost = smtp.macollectivite.org` :

```
apt-get install postfix bsd-mailx
vi /etc/postfix/main.cf
/etc/init.d/postfix restart
```

> Type de configuration : « site internet »

Tester le bon fonctionnement avec :

```
mail -s "test smtp collectivité" mon@mail
```

> Valider la ligne, saisir '.' , puis revalider : un mail part avec uniquement le sujet

On doit voir que le mail est bien parti avec un status `sent 250`, visible dans `/var/log/mail.log`

OPTION : Il est possible de forcer unilatéralement le champ FROM des courriels envoyés.
La configuration décrite ci-dessous change l'adresse émettrice pour les e-mails émis localement ainsi que ceux relayés par postfix. Ajouter dans `/etc/postfix/main.cf` les directives :

```
sender_canonical_classes = envelope_sender, header_sender
sender_canonical_maps = regexp:/etc/postfix/sender_canonical_maps
smtp_header_checks = regexp:/etc/postfix/header_check
```

Ajouter le fichier `/etc/postfix/sender_canonical_maps` : (adapter l'adresse émetteur générique)

```
/.+/    ne-pas-repondre@address.com
```

Ajouter le fichier `/etc/postfix/header_check` : (adapter l'adresse émetteur générique)

```
/From:.*/ REPLACE From: ne-pas-repondre@address.com
```

Enfin redémarrer le service postfix ...

Autres commandes utiles :

```
dpkg-reconfigure postfix     # pour reconfigurer postfix
postconf -n                  # affiche la configuration de postfix
```

> *Remarque* : Sur Debian , c'est le MTA exim qui est installé par défaut.

2.4. Installation du frontal Web HTTPS

Ce composant est nécessaire pour filtrer/orienter/déchiffrer les flux entrants. Le service préconisé est**NginX**. Apache n'est pas supporté pour i-Parapheur 4.5 (contributions bienvenues).

Plus moderne et modulaire, ce composant serveur Web est plus adapté aux déploiements mixtes avec clients Web + WebServices + tablettes. La version livrée dans les distributions Linux est trop bridée (même pour Ubuntu 14.04 qui ne livre NginX qu'en version 1.4.6). L'installation se repose donc sur le dépôt officiel de nginx.org (v1.13 en Juillet 2017), voir ci-après l'exemple pour Ubuntu :

Ajouter les lignes suivantes au fichier `/etc/apt/sources.list` :

```
deb http://nginx.org/packages/ubuntu/ xenial nginx
deb-src http://nginx.org/packages/ubuntu/ xenial nginx
```

> Sur Debian7 : ajouter plutôt la ligne
> `deb http://nginx.org/packages/mainline/debian/ wheezy nginx`

Puis exécuter les commandes :

```
wget http://nginx.org/keys/nginx_signing.key
apt-key add nginx_signing.key
apt-get update
apt-get install nginx
```

> *Remarque* : Si RHEL/CentOS , suivre les instructions sur:http://nginx.org/en/linux_packages.html#stable télécharger le fichier de repository et l'installer avec `yum localinstall`.
> Exemple CentOS7
> http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm Puis : `yum install nginx`

3. CONFIGURATION DU FRONTAL WEB

3.1. Paramétrage NginX

Un exemple de configuration est disponible dans `/opt/_install/confs/nginx` (issu de l'archive `confs.tar.gz`). Cela permettra notamment le déploiement pour l'accueil de connexions tablettes numériques.

Fichiers de configuration à copier dans `/etc/nginx/conf.d/` :

- Personnalisation générale : `/etc/nginx/conf.d/confpara`

```
client_max_body_size 200M; #Pour upload de gros fichier
add_header Cache-Control public;

proxy_intercept_errors on;

root /var/www/parapheur;

error_page 500 /error_pages/500.html;
error_page 404 /error_pages/404.html;
error_page 403 /error_pages/403.html;
error_page 401 /error_pages/401.html;
#error_page 400 /error_pages/400.html;

proxy_set_header        X-Real-IP           $remote_addr;
proxy_set_header        X-Forwarded-For     $proxy_add_x_forwarded_for;
proxy_set_header        Host                $host;
proxy_set_header        X-Forwarded-Server  $host;
proxy_set_header        X-Forwarded-Host    $host;

proxy_redirect off;
```

- Personnalisation générale pour SSL : `/etc/nginx/conf.d/confssl`

```
ssl on;

#VALIDCA
#Fichiers générés avec script nginx
ssl_crl /etc/nginx/ssl/validca/all.crl;
ssl_client_certificate /etc/nginx/ssl/validca/all.pem;

#Configuration ssl conseillée par mozilla (supprimé SSLv3 suite à CVE-2014-3566)
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_prefer_server_ciphers on;
ssl_ciphers "ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-
AES128-GCM-SHA256:DHE-DSS-AES128-GCM-SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-
AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-
AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-SHA:DHE-RSA-AES256-SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:ECDHE-RSA-
RC4-SHA:ECDHE-ECDSA-RC4-SHA:AES128:AES256:RC4-SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!3DES:!MD5:!PSK";

ssl_session_cache shared:SSL:50m;
ssl_session_timeout 10m;
ssl_verify_depth 5;

set $https_enabled on;

proxy_set_header X-Forwarded-Proto  https;
proxy_set_header HTTPS              on;
proxy_set_header X-Forwarded-Ssl    on;
proxy_set_header SSL_VERIFIED       $ssl_client_verify;
proxy_set_header SSL_CLIENT_SERIAL  $ssl_client_serial;
proxy_set_header SSL_CLIENT_CERT    $ssl_client_cert;
proxy_set_header SSL_DN             $ssl_client_s_dn;
proxy_set_header VERIFIED           $ssl_client_verify;
proxy_set_header DN                 $ssl_client_s_dn;
proxy_set_header SSL_PROTOCOL       $ssl_protocol;
proxy_set_header SSL_CIPHER         $ssl_cipher;
proxy_set_header SSL_SESSION_ID     $ssl_session_id;
proxy_set_header SSL_CLIENT_VERIFY  $ssl_client_verify;
```

Fichiers d'hôtes virtuels à adapter (noms FQDN sur variable *server_name*, etc.) :

- « VirtualHost » HTTP – `/etc/nginx/conf.d/parapheur.conf`

```
#PARAPHEUR
server {
    listen 80;
    server_name iparapheur.dom.local secure-iparapheur.dom.local;

    rewrite (?=^[\w\.-]+\/?[\w\.-]*$)^(?![\w\.-]*\/$).*$ $1/ permanent;

    access_log /var/log/nginx/parapheur_access.log;
    error_log /var/log/nginx/parapheur_error.log;
```

```
    include /etc/nginx/conf.d/confpara;

    location /socket.io/ {
        proxy_pass http://127.0.0.1:8081/socket.io/;
    }

    location /bl-xemwebviewer/ {
        proxy_pass http://127.0.0.1:9080/bl-xemwebviewer/;
    }

    #Acces au CANAL HISTORIQUE pour diagnostic/telemaintenance, node-browser...
    location /alfresco/ {
        proxy_pass http://127.0.0.1:8080/alfresco/;
    }

    #On empeche l'accès à l'index des webscripts
    location /index {
        return 404;
    }

    #Location d'accès aux pages statiques
    location ~ ^/(themes|docs|applets|error_pages|favicon.ico)/ {}

    # On redirige en HTTPS pour le reste
    location / {
        proxy_pass https://$server_name/iparapheur/;
    }
}

#REDIRECT MOBILE
server {
    listen 80;
    server_name m.iparapheur.dom.local;

    access_log /var/log/nginx/parapheur_m_access.log;
    error_log /var/log/nginx/parapheur_m_error.log;

    include /etc/nginx/conf.d/confpara;

    location / {
        return https://$server_name/;
    }
}
```

- « VirtualHost » HTTPS – `/etc/nginx/conf.d/parapheur_ssl.conf`

```
## Point d'entree classique web
server {
    listen 443;
    server_name iparapheur.dom.local;

    rewrite (?=^[\w\.-]+\/?[\w\.-]*$)^(?![\w\.-]*\/$).*$ $1/ permanent;

    include /etc/nginx/conf.d/confpara;
    include /etc/nginx/conf.d/confssl;

    #CERTIFICATS SERVER
    ssl_certificate /etc/nginx/ssl/wildcard.dom.local.pem;
    ssl_certificate_key /etc/nginx/ssl/wildcard.dom.local.key;

    access_log /var/log/nginx/parapheur_ssl_access.log;
    error_log /var/log/nginx/parapheur_ssl_error.log;

    ssl_verify_client off;

    error_page 418 = @wsdl;
    recursive_error_pages on;

    location @wsdl {
        try_files /alfresco/iparapheur.wsdl 404;
    }

    #Accès à la visionneuse
    location /bl-xemwebviewer/ {
        #Gestion des PJ absentes
        if ($request_uri ~ .*attachment*.) {
            error_page 500 /error_pages/visionneuse.html;
        }
        proxy_pass http://127.0.0.1:9080/bl-xemwebviewer/;
    }

    #Accès au serveur websocket
    location /socket.io/ {
        proxy_pass http://127.0.0.1:8081/socket.io/;
    }

    #On empeche l'accès à l'index des webscripts
    location /index {
        return 404;
    }
```

```
    #Location d'accès aux pages statiques
    location ~ ^/(themes|docs|error_pages|applets|libersign|favicon.ico)/ {}

    location ~ ^/alfresco/ {
        return 404;
    }

    #Si l'accès ne se fait pas via le root, proxy !
    location /iparapheur/ {
        proxy_pass http://127.0.0.1:8080/iparapheur/;
    }

    location / {
        return https://$server_name/iparapheur/;
    }
}

## Point d'entree WebService et Web avec certificat client
server {
    listen 443 default_server;

    server_name secure-iparapheur.dom.local;

    include /etc/nginx/conf.d/confpara;
    include /etc/nginx/conf.d/confssl;

    #CERTIFICATS SERVER
    ssl_certificate /etc/nginx/ssl/wildcard.dom.local.pem;
    ssl_certificate_key /etc/nginx/ssl/wildcard.dom.local.key;

    access_log /var/log/nginx/parapheur_ssl_secure_access.log;
    error_log /var/log/nginx/parapheur_ssl_secure_error.log;

    ssl_verify_client optional;

    error_page 418 = @wsdl;
    recursive_error_pages on;

    location @wsdl {
        try_files /alfresco/iparapheur.wsdl 404;
    }

    location ~ ^/error_pages/ {}

    location /ws-iparapheur {
        if ($ssl_client_verify != SUCCESS) {
            return 403;
            break;
        }
        if ($args ~ wsdl) {
            return 418;
            break;
        }
        proxy_pass http://127.0.0.1:8080/alfresco/ws-iparapheur;
    }

    location /ws-iparapheur-no-mtom {
        if ($ssl_client_verify != SUCCESS) {
            return 403;
            break;
        }
        if ($args ~ wsdl) {
            return 418;
            break;
        }
        proxy_pass http://127.0.0.1:8080/alfresco/ws-iparapheur-no-mtom;
    }

    #Si l'accès ne se fait pas via le root, proxy !
    location / {
        proxy_pass http://127.0.0.1:8080/;
    }
}

############################################################
## Point d'entree clients TABLETTE
############################################################
#server {
#    listen 443;
#    server_name m-iparapheur.dom.local;
#
#    include /etc/nginx/conf.d/confpara;
#    include /etc/nginx/conf.d/confssl;
#    # CERTIFICATS SERVER (let's encrypt)
#    ssl_certificate /etc/nginx/ssl/m.iparapheur.dom.local.pem;
#    ssl_certificate_key /etc/nginx/ssl/m.iparapheur.dom.local.key;
#
#    access_log /var/log/nginx/parapheur_ssl_m_access.log;
#    error_log /var/log/nginx/parapheur_ssl_m_error.log;
#
#    ssl_verify_client off;
```

```
#
#    # Position des certificats G3 mobiles
#    location /certificates/ {
#        root /var/www/parapheur/;
#    }
#
#    location / {
#        proxy_pass http://127.0.0.1:8080/alfresco/service/;
#    }
#}
```

> *Remarque* : CentOS 7, l'activation du service NginX au démarrage de la machine se fait avec « systemd »

```
systemctl enable nginx.service
```

3.2. Sécurisation avec certificats SSL

Récupération des CRLs : dans `/etc/nginx/ssl/` déposer le script `recup_crl_nginx.sh` qui téléchargera périodiquement la liste des autorités de certification reconnues par la plate-forme (dans un sous-répertoire `validca`):

```
mkdir /etc/nginx/ssl; cd /etc/nginx/ssl/
cp /opt/_install/confs/nginx/ssl/recup_crl_nginx.sh . ; chmod +x recup_crl_nginx.sh
./recup_crl_nginx.sh /etc/nginx/ssl
```

3.3. Configuration HTTPS, certificats Serveurs

> *Remarque* : Le fichier de « virtual host » référence plusieurs fichiers de certificat électronique : ceux-ci auront été acquis au préalable auprès d'une autorité de certification qualifiée, ou à défaut auprès d'une AC de moindre confiance (voire locale et auto-signée mais attention à sa gestion!, cf. Annexe).

Dans le répertoire `/etc/nginx/ssl/` , déposer les certificats TLS/SSL du serveur web (un certificat X509 + une clé privée RSA pour chaque nom FQDN) : dans l'exemple ci-dessus, on remarque :

- **iparapheur.dom.local** : sécurise l'accès principal via navigateur Web ,
- **secure-iparapheur.dom.local** : sécurise l'accès via Certificat client (web-services ou web) ,
- **m.iparapheur.dom.local** : OPTIONNEL à destination des usages tablette Google™ Android ou Apple™ iPad. Ce certificat n'est nécessaire que si il est prévu d'exploiter i-Parapheur avec ces applications. **Important** : Les applications clientes i-Parapheur publiées par Libriciel SCOP sur les kiosques d'application Apple™ « App Store » et Google™ « Play Store » ont besoin d'un tel certificat SSL valide, nécessairement fourni par Libriciel SCOP (gestionnaire de l'AC).

Rappel, chaque certificat est composé de deux fichiers : (cf manuel d'administration sur http://www.nginx.org)

- `fqdn.key` : il contient la clé privée RSA, encodée au format PEM base64
- `fqdn.pem` : il contient le certificat correspondant au format PEM base64, avec la chaîne complète de certification, c'est-à-dire la chaîne des AC jusqu'à l'AC racine. Cela facilite le contrôle de validité pour le navigateur Web.

Une fois les certificats installés et dûment référencés dans le fichier VirtualHost `parapheur_ssl.conf`, vérifier que NginX fonctionne et écoute bien sur le port HTTPS (443):

```
service nginx restart                        # redémarrage de NginX.
netstat -antup | grep 443                    # doit donner un résultat
cd /etc/nginx; rm sites-enabled/000-default  # un peu de ménage
```

Le script de mises-à-jour des CRL (listes des certificats révoqués) doit être appelé régulièrement (via CRON par exemple), voir annexe à ce propos.

> *Remarque* : Attention sur RedHat / CentOS : il peut y avoir des soucis sur le fonctionnement du reverse-proxy, à cause de la présence d'un SE-Linux activé. Voir:
> http://wiki.centos.org/TipsAndTricks/SelinuxBooleans

3.4. Certificats pour applications tierces

La plupart des applications métier (GF, GRH, Courrier, Délibérations, Arrêtés,...) ont besoin d'éléments de sécurité pour se connecter à i-Parapheur. Il faut constituer 2 éléments de sécurité (magasins de certificat) :

- un keyStore (contenant le certificat du client et sa partie privée), pour s'authentifier,

- un trustStore (contenant la chaîne de certification du serveur) pour reconnaître le serveur.

Des outils tels que **Porte-Cle** (logiciel libre écrit en JAVA, multi-plateforme, version courante: 1.9) font cela très bien. Voir le manuel d'administration pour davantage d'informations concernant ce paramétrage.

4. COMPOSANTS I-PARAPHEUR

4.1. Initialisation de la base de données de l'entrepôt

> Rappel: Dans ce manuel, le serveur de base de données MySQL est installé sur le même serveur que l'application i-Parapheur.
> Il convient d'adapter cette procédure dans le cas d'une déportation de la base de données sur un serveur distant ***ET de s'assurer*** que ce service distante apporte au moins le même niveau de performances (ping <2ms) et de ressources qu'en cas d'installation locale.

Tuning de MySQL : dans `/etc/mysql/conf.d` , ajouter le fichier `iParapheur-mysql.cnf` :

```
#
# iParapheur MySQL config file
#
[mysqld]
bind-address                    = 127.0.0.1
#
open_files_limit                = 8192
innodb_file_per_table           = 1
wait_timeout                    = 28800
#
# * Fine Tuning
#
key_buffer                      = 16M
max_allowed_packet              = 16M
thread_stack                    = 192K
thread_cache_size               = 200
max_connections                 = 360
#
# * Query Cache Configuration
#
query_cache_limit               = 32M
query_cache_size                = 64M
tmp_table_size                  = 1024M
max_heap_table_size             = 1024M
table_open_cache                = 500
table_definition_cache          = 400
#
innodb_buffer_pool_size         = 1024M
innodb_read_io_threads          = 32
innodb_file_io_threads          = 32
innodb_locks_unsafe_for_binlog  = 1
```

La base de données va être initialisée avec la commande suivante :

```
service mysql restart
mysql -u root -p*supersecret* < /opt/_install/confs/mysql-init.sql
```

> **ATTENTION** : mot de passe root de MySQL a été déini à l'installation.

4.1.1. Allocation de ressources

ATTENTION : vérifier que le pool de connexions maximum est bien à 360 ! Sinon, l'application aura des problèmes de fonctionnement. Avec les serveurs GNU/Linux sous `systemd` , il y a des ajustements à faire, décrits ci-dessous :

```
cp /lib/systemd/system/mysql.service /etc/systemd/system/
vim /etc/systemd/system/mysql.service
```

Ajouter en fin de fichier les directives :

```
LimitNOFILE=8192
LimitMEMLOCK=infinity
```

Enfin :

```
systemctl daemon-reload
service mysql restart
```

Vérifier au run-time que les autres paramètres sont également pris en compte.

4.2. Installation de Alfresco 3.4.c Community Edition

Le bundle Alfresco Community Edition est librement téléchargeable sur http://sourceforge.net/projects/alfresco/files/.
Dans ce manuel et par convention, l'installation est effectuée dans le répertoire `/opt/iParapheur/` . On procède à l'installation en mode

console (en ligne de commande dans un terminal), ce qui autorise des déploiements en télé-intervention.

Ci-après un exemple avec le package d'installation Alfresco pré-déposé dans `/opt/_install` :

```
mkdir -p /opt/iParapheur; cd /opt/iParapheur
chmod +x /opt/_install/alfresco-community-3.4.c-installer-linux-x64.bin
/opt/_install/alfresco-community-3.4.c-installer-linux-x64.bin --mode text
```

Ci-après un exemple de réponses données lors de l'installation :

```
Language Selection

Please select the installation language
[1] English - English
[2] French - Français
[3] Spanish - Español
[4] Italian - Italiano
[5] German - Deutsch
[6] Japanese - □□□
Please choose an option [1] : 2
-------------------------------------------------------------------------------
Bienvenue dans l'assistant d'installation de Alfresco Community
-------------------------------------------------------------------------------
Sélectionnez des composants que vous désirez installer, décochez ceux que vous
ne voulez pas installer. Cliquez sur Suivant pour continuer.

MySQL : Y (Cannot be edited)
Java : Y (Cannot be edited)
Alfresco : Y (Cannot be edited)

SharePoint [Y/n] :n

Records Management [Y/n] :n

Web Quick Start [Y/n] :n

WCM Alfresco [Y/n] :n

Quickr Connector Support [Y/n] :n

OpenOffice [Y/n] :Y

Est-ce que la sélection est correcte ? [Y/n]: Y

-------------------------------------------------------------------------------
Type d'installation
[1] Facile - Installe les serveurs avec la configuration par défaut
[2] Avancé - Configure les ports du serveur et les propriétés de service
Merci de choisi une option. [1] : 2

-------------------------------------------------------------------------------
Dossier d'installation

Please choose a folder to install Alfresco Community.
Sélectionner un dossier [/opt/alfresco-3.4.c]: /opt/iParapheur

-------------------------------------------------------------------------------
Installation de la base de données

Veuillez sélectionner la configuration de base de données que vous souhaitez utiliser. Si
vous sélectionnez une base de données existante, vous devez configurer l'application Alfresco
avant de procéder à l'installation.

[1] Je souhaite utiliser la base de données MySQL groupée
[2] Je souhaite utiliser une base de données existante
Merci de choisi une option. [1] : 2

-------------------------------------------------------------------------------
Configuration de la base de données

URL JDBC [jdbc:mysql://localhost/alfresco]:
Pilote JDBC [org.gjt.mm.mysql.Driver]:
Nom de la base de données: [alfresco]:
Nom d'utilisateur []: alfresco

Mot de passe :
Saisir à nouveau :

-------------------------------------------------------------------------------
Configuration du port Tomcat

Veuillez saisir les paramètres de configuration Tomcat que vous souhaitez utiliser

Domaine du serveur Web : [127.0.0.1]: iparapheur.dom.local
Port de serveur Tomcat : [8080]:
Port d'arrêt Tomcat : [8005]:
Port SSL Tomcat [8443]:
Port AJP Tomcat : [8009]:
```

```
-------------------------------------------------------------------------------
Port FTP Alfresco

Please choose a port number to use for the integrated Alfresco FTP server.

Port : [21]: 2121

-------------------------------------------------------------------------------
Port RMI Alfresco

Please choose a port number for Alfresco to use to execute remote commands.

Port : [50500]:

-------------------------------------------------------------------------------
Admin Password

Veuillez indiquer un mot de passe afin d'utiliser le compte administrateur Alfresco

Mot de passe admin :
Répéter le mot de passe :
-------------------------------------------------------------------------------
Installer en tant que service

Si vous le souhaitez, vous pouvez enregistrer Alfresco Community en tant que service. Ainsi,
le démarrage s'exécutera automatiquement à chaque démarrage de la machine.

Installer Alfresco Community en tant que service ? [Y/n]: Y

-------------------------------------------------------------------------------
Port de serveur OpenOffice

Veuillez saisir le port que le serveur OpenOffice écoutera par défaut

Port de serveur OpenOffice [8100]:

-------------------------------------------------------------------------------
L'assistant d'installation est maintenant prêt à démarrer l'installation de
Alfresco Community sur votre ordinateur.

Voulez-vous continuer ? [Y/n]: Y

-------------------------------------------------------------------------------
Merci de patienter durant l'installation de Alfresco Community sur votre ordinateur.
Installation en cours
0% _____________ 50% _____________ 100%
########################################
-------------------------------------------------------------------------------
L'assistant vient de finir l'installation de Alfresco Community sur votre ordinateur.

Voir le fichier Lisezmoi ? [Y/n]: n
```

Répertoire pour les traces applicatives : fichier nommé `/var/log/alfresco/alfresco.log`

```
mkdir -p /var/log/alfresco/tomcat/logs ; mkdir -p /var/lib/alfresco/tmp
rm -rf /opt/iParapheur/tomcat/logs
ln -s /var/log/alfresco/tomcat/logs /opt/iParapheur/tomcat/logs
```

4.3. Librairie GhostScript

Il s'agit de permettre l'usage de Ghostscript par Tomcat : en effet, la librairie installée par le système d'exploitation ne semble pas fonctionner correctement. Une version recompilée de GhostScript est fournie pour les systèmes Ubuntu 12.04, 14.04, 16.04, Debian 7 et 8, CentOS 6 et 7.
Recopier la bonne version vers `/opt/iParapheur/common/lib/`, et créer le lien symbolique adéquat, comme suit:

```
cp /opt/_install/confs/libgs-16.04/libgs.so.9.20 /opt/iParapheur/common/lib/
ln -s /opt/iParapheur/common/lib/libgs.so.9.20 /opt/iParapheur/common/lib/libgs.so
```

Pour les autres systèmes d'exploitation non encore couverts par le contenu de l'archive confs.tgz, il faut compiler à partir des sources, prendre la version en date (9.21 fonctionne, en date de rédaction de ce document).

> *Remarque : si RedHat 7 / CentOS 7, voir*http://ghostscript.com/download/gsdnld.html
> http://www.ghostscript.com/doc/9.21/Install.htm#Install_Unix5

Méthode testée avec CentOS 7, les dépôts EPEL nécessaires (installés comme suit au cas où), ainsi que les outils et environnement de compilation et dépendances diverses :

```
yum install epel-release
yum install cabextract
yum install wget rpm-build chkfontpath ttmkfdir
yum grouplist
```

```
yum groupinstall "Development Tools"
yum install zlib-devel libjpeg-devel giflib-devel freetype-devel gcc gcc-c++ make
```

Avec Debian 7, les pré-requis à l'installation s'installent avec la ligne :

```
apt-get install build-essential zlib1g-dev libjpeg62-dev
```

Avec Debian 8, voici les pré-requis à l'installation:

```
apt-get install build-essential zlib1g-dev libjpeg-dev libgif-dev libtiff5-dev
```

Puis la procédure de téléchargement + production de la librairie en elle-même (rien que la librairie ".so") :

```
cd /opt/_install
wget https://github.com/ArtifexSoftware/ghostpdl-downloads/releases/download/gs921/ghostscript-9.21.tar.gz
tar xzf ghostscript-9.21.tar.gz
cd ghostscript-9.21
./configure
make soinstall
cp /usr/local/lib/libgs.so.9.21 /opt/iParapheur/common/lib/
ln -s /opt/iParapheur/common/lib/libgs.so.9.21 /opt/iParapheur/common/lib/libgs.so
```

En cas de besoin de déploiement de Fontes supplémentaires, elles sont à déposer dans le répertoire :
`/usr/local/share/ghostscript/fonts/`

4.4. Fichier 'alfresco-global.properties'

Ajouter les éléments de paramétrage i-Parapheur dans le fichier `alfresco-global.properties`, puis y vérifier les divers paramètres (localisation de l'entrepôt avec configuration de la base de données, stockage, et chemins d'accès) :

```
cd /opt/iParapheur/tomcat/shared/classes
cat /opt/_install/confs/ADD-to_alfresco-global.properties >> alfresco-global.properties
vi /opt/iParapheur/tomcat/shared/classes/alfresco-global.properties
```

```
## ATTENTION : Ce sont des valeurs données à titre d'exemple
##
## Le copier-coller sauvage peut nuire à la santé de l'installation de i-parapheur.

dir.root=/opt/iParapheur/alf_data

### database connection properties
db.driver=org.gjt.mm.mysql.Driver
db.username=alfresco
db.password=alfresco
db.name=alfresco
db.url=jdbc:mysql://127.0.0.1:3306/alfresco

ftp.enabled=false

### External executable locations
ooo.exe=/opt/iParapheur/openoffice/program/soffice.bin
ooo.enabled=true

# ATTENTION : paramètre à ajouter, il n'existe pas (vers ligne numero 35):
#              « open » si openOffice
#              « libre » si libreOffice.
ooo.version=libre

img.root=/opt/iParapheur/common
img.dyn=${img.root}/lib
img.exe=${img.root}/bin/convert
swf.exe=/opt/iParapheur/common/bin/pdf2swf
jodconverter.enabled=true
jodconverter.officeHome=/opt/iParapheur/openoffice
jodconverter.portNumbers=8101

##
# Default properties used in i-parapheur
#-------------
db.pool.initial=100
db.pool.max=350
audit.enabled=true
lucene.query.maxClauses=100000
lucene.indexer.mergerTargetIndexCount=8

# Renseigner l'url de base pour l'applet de signature (v3 only - DEPRECATED)
parapheur.signature.applet.url=http://iparapheur.dom.local/alfresco
## Modeles d'email : emetteur par defaut, et URL de base (sans http://)
parapheur.mail.from=ne-pas-repondre@dom.local
parapheur.mail.baseUrl=iparapheur.dom.local/iparapheur
parapheur.mail.baseUrlSecure=secure-iparapheur.dom.local/iparapheur
# Modeles d'email : prefixe dans l'objet, par exemple [i-Parapheur]
parapheur.mail.objet.prefixe=[i-Parapheur]
parapheur.mail.targetVersion=4

## Proprietes pour generation PDF archive/impression
```

```
parapheur.archive.ttfVerdana.location=/opt/iParapheur/verdanai.ttf
parapheur.archive.iccprofile.location=/opt/iParapheur/srgb.profile
parapheur.archive.tamponActes.prefixe=Acquitté en PREFECTURE le:

## parapheur.habillage (adullact|blex)
parapheur.habillage=adullact
## parapheur.ihm.document.uploadMaxSize :
# - 0 : taille illimitee (valeur par defaut)
# - n : taille limitee a 'n' (en mega-octets)
parapheur.ihm.document.uploadMaxSize=0

## (EXPERIMENTAL) Traitement par lot non bloquant (true|false)
parapheur.jobs.thread.enabled=false
## Affichage d'apercu de dossier : parapheur.preview.enabled (true|false)
parapheur.preview.enabled=true
## parapheur.tdts2low.statutjobinterval: periodicite de connection au TDT
# 30 : valeur par defaut (en minutes)
parapheur.tdts2low.statutjobinterval=30

## parapheur.ws.getdossier : interaction Web-Services
parapheur.ws.getdossier.pdf.enabled=true
parapheur.ws.getdossier.pdf.docName=iParapheur_impression_dossier.pdf

## Generation des apercus bitmap pour client v4 (true|false)
parapheur.mobilepreview.enabled=true

## Resolution GhostScript pour apercus de dossier (150 par defaut)
# n'est plus utilise en 4.2: parapheur.ghostscript.path=/usr/bin/gs
parapheur.ghostscript.dpi=150

## Valeur par defaut de la trigger cron
parapheur.notification.digest.cron=0 0 8 * * ?

## Par defaut, i-Parapheur n'accepte pas les DOCX etc.
parapheur.document.openxml.accept=false

## NOTIFICATIONS & MESSAGE QUEUE ## (nouveaute 4.2) ##
# nombre de threads par TENANT, indiquant la capacite d'actions simultanees par tenant
parapheur.jobs.thread.pool=3
# Taille max (Mo) pour /opt/iParapheur/alf_data/activemq/localhost/tmp_storage
parapheur.jobs.mq.store=512
# Définit si le broker ActiveMQ est embarqué dans l'application I-Parapheur ou non
parapheur.jobs.mq.broker.internal=true
# Connecteur du broker ActiveMQ interne (si parapheur.jobs.mq.broker.internal=true)
parapheur.jobs.mq.broker.connector=vm://localhost
# Connecteur utilisé par les consommateurs ActiveMQ
parapheur.jobs.mq.consumer.connector=vm://localhost

# WebSocket Port d'ecoute du service de notifications
parapheur.notifications.websocketport=8081

## CDC Fast-Service
fastService.repeatintervalMinutes=30
fastService.startDelayMinutes=40

## Duree de validité du ticket de session alfresco
# authentication.ticket.validDuration=P1H

## Pour gerer le claimedRole a partir des proprietes des utilisateurs
parapheur.claimedRole.userMetadataBased=true

## Forcer les liens https dans les mails
parapheur.mail.force.https=true

sso.enabled=false
# Pour autoriser l'authentification par header HTTP (ex. SSO LemonLDAP::NG)
parapheur.auth.external.header.authorize=false

## Nombre max de documents principaux par dossier
parapheur.ihm.creerdossier.maindocuments.max=6

## Surcharge du tag #signature# (localisation image signature numerique)
parapheur.libersign.tag.signature.name=#signature#
parapheur.libersign.tag.signature.name.tenants={}

## Nouveautes v4.4
openOffice.test.cronExpression=*/5 * * * * ?
parapheur.exploit.xemelios.command=/etc/init.d/xemwebview

## AJOUT v4.4.1
parapheur.document.lockedPDF.accept=false
parapheur.hostname=iparapheur.dom.local

## AJOUT v4.5.0
parapheur.cachetserver.security.key=**SECRET**
# Position du cachet serveur
parapheur.libersign.tag.cachet.name=#cachet#
parapheur.libersign.tag.cachet.name.tenants={}
# cron des mails de retard
parapheur.notification.retards.cron=0 0 7 * * ?
# Mail de warning sur l'expiration des certificats cachets serveur
```

```
parapheur.cachetserver.warnexpiration.cronexpression=0 0 8 ? * 1
parapheur.cachetserver.warnexpiration.daysuntilexpiration=30
```

ATTENTION: La valeur de la variable `parapheur.cachetserver.security.key` doit être générée aléatoirement lors de l'installation du i-Parapheur. Il s'agit d'une clé secrète permettant de protéger (par chiffrement crytographique) les mots de passe des certificats enregistrés sur le parapheur. Voici une commande permettant de générer aléatoirement une clé suffisamment forte :

```
< /dev/urandom tr -dc _A-Z-a-z-0-9 | head -c${1:-32};echo;
```

> *Remarque* : ne pas perdre cette clé.

4.5. Fichier TOMCAT 'server.xml'

Par défaut le connecteur AJP13 est activé mais mal configuré.

```
vi /opt/iParapheur/tomcat/conf/server.xml      # (activer AJP13:8009)
```

Ligne **90** : la ligne sur le connecteur AJP13, et vérifier que son paramétrage est de la forme :

```
<Connector port="8009" enableLookups="false" protocol="AJP/1.3"
   URIEncoding="UTF-8" connectionTimeout="600000" redirectPort="8443" />
```

~Ligne **103** : Si usage de Nginx, ajouter une directive `Valve` après la ligne `<Engine name="Catalina" defaultHost="localhost">` :

```
<Valve className="org.apache.catalina.valves.RemoteIpValve"
       protocolHeader="X-Forwarded-Proto" protocolHeaderHttpsValue="https" />
```

4.6. Environnement d'exécution JAVA

Des instabilités ains que des limitations techniques ont été constatées avec le JRE fourni par Alfresco, forçons l'usage de JDK8_161 (fourni par Oracle, à télécharger au préalable), en remplaçant le répertoire `/opt/iParapheur/java/` :

```
rm -rf /opt/iParapheur/java/
ln -s /opt/jdk1.8.0_161 /opt/iParapheur/java
```

> *Remarque importante* : le i-Parapheur utilise le protocole TLS1.2 pour communiquer avec le TDT S2low. Le passage en version 8 de Java est obligatoire pour que cette communication soit fonctionnelle.

4.7. Script de lancement/arrêt TOMCAT `ctl.sh`

Le script pre-installé est nécessaire mais son contenu est insuffisant pour les besoins i-Parapheur.

```
vi /opt/iParapheur/tomcat/scripts/ctl.sh
```

Ajouter en début de fichier :

```
export LANG=fr_FR.UTF-8
```

Modifier la variable `CATALINA_PID` et autres `JAVA_OPTS` comme suit :

```
# Ligne 6
CATALINA_PID=/var/run/parapheur.pid

# Lignes 13 ET 26
export JAVA_OPTS='-XX:MaxPermSize=512m -Xms3072m -Xmx3072m -Xss1024k -XX:PermSize=64m
-XX:NewSize=256m -Dfile.encoding=UTF-8 -Djava.io.tmpdir=/var/lib/alfresco/tmp
-Dalfresco.home=/opt/iParapheur -Dcom.sun.management.jmxremote
-XX:+UseConcMarkSweepGC -XX:+CMSIncrementalMode -XX:CMSInitiatingOccupancyFraction=70'

# Ligne 32
$TOMCAT_BINDIR/shutdown.sh 30 -force
```

> *Remarque importante* : ne pas hésiter à augmenter les valeurs allouées à**Xms** et **Xmx** (taille de la JVM de Tomcat) à 4Go ou davantage si nécessaire : les gros flux PESv2 (à partir de 50Mo) en particulier réclament énormément de ressources... dû à l'empreinte mémoire consommée pour leur traitement.

4.8. Personnalisation du WAR 'alfresco'

1ère injection du fichier AMP (Alfresco Module Package) de i-Parapheur, dans le WAR Alfresco de base :

```
cp /opt/_install/iParapheur-vX.Y.Z/iParapheur-amp*.amp /opt/iParapheur/amps/
cd /opt/iParapheur
bin/apply_amps.sh
```

Résultat de la commande précédente :

```
This script will apply all the AMPs in amps and amps-share to the alfresco.war and share.war files in tomcat/webapps
Press control-c to stop this script...
Press any other key to continue...

Module 'parapheur' installed in 'tomcat/webapps/alfresco.war'
    - Title:        i-Parapheur ADULLACT
    - Version:      3.4
    - Install Date: Mon Dec 24 15:55:34 CET 2012
    - Desription:   Parapheur electronique ADULLACT
No modules are installed in this WAR file
No modules are installed in this WAR file
About to clean out tomcat/webapps/alfresco and share directories and temporary files...
Press control-c to stop this script...
Press any other key to continue...

Cleaning temporary Alfresco files from Tomcat...
```

Mise au propre par une seconde passe, sur le même principe, qui cette fois nettoie certaines librairies superflues et toxiques :

```
mkdir /opt/iParapheur/tomcat/webapps/alfresco
cp /opt/_install/confs/iparaph-updateAMP.sh /opt/iParapheur
cd /opt/iParapheur ; chmod +x iparaph-updateAMP.sh ; ./iparaph-updateAMP.sh
```

Suppression de la web-app inutile 'share', externalisation du fichier de configuration `log4j.properties` :

```
cd /opt/iParapheur/tomcat/webapps ; rm share.war
cd /opt/iParapheur/tomcat/webapps/alfresco/WEB-INF/classes
cp log4j.properties ../../../../shared/classes/alfresco/extension/custom-log4j.properties
```

Renseigner l'emplacement du fichier de log applicative, activer les logs Web-services :

```
vi /opt/iParapheur/tomcat/shared/classes/alfresco/extension/custom-log4j.properties
```

```
# Ligne 16
log4j.appender.File.File=/var/log/alfresco/alfresco.log

# À ajouter en fin de fichier
log4j.logger.org.adullact.spring_ws.iparapheur._1.InterfaceParapheurImpl=info
```

4.9. Connecteur Web-Services

Configuration des URLs du fichier WSDL (remplacement du FQDN pour les connecteurs Web-Services) :

```
cd /opt/iParapheur ; cp /opt/_install/confs/custom-wsdl.sh .
./custom-wsdl.sh iparapheur.dom.local
```

Ce script va remplacer les URLs des services par celle donnée en paramètre préfixée par `secure-`, en fin de fichier :

```
<soap:address location="https://secure-iparapheur.dom.local:443/ws-iparapheur" />
<soap:address location="https://secure-iparapheur.dom.local:443/ws-iparapheur-no-mtom" />
```

Pour information : contenu du script

```
#!/bin/bash
sedpattern="s/iparapheur.demonstrations.adullact.org/$1/g"
sed -i $sedpattern tomcat/webapps/alfresco/WEB-INF/wsdl/iparapheur.wsdl
```

4.10. Client Web i-Parapheur

4.10.1. Déploiement du WAR « iparapheur »

Copie du fichier WAR du client Web i-Parapheur dans le répertoire des webapps de TOMCAT :

```
cd /opt/_install/iParapheur-vX.Y.Z
cp iparapheur-war-*.war /opt/iParapheur/tomcat/webapps/iparapheur.war
cp deployWarIparapheur.sh /opt/iParapheur/
mkdir /opt/iParapheur/tomcat/webapps/iparapheur
cd /opt/iParapheur ; chmod +x deployWarIparapheur.sh
./deployWarIparapheur.sh
```

4.10.2. Fichier `iparapheur-global.properties`

Paramétrage, à partir d'un fichier d'exemple fourni (attention pour l'accès à l'outil de signature électronique LiberSign) :

```
cp /opt/_install/confs/iparapheur-global.properties tomcat/shared/classes/
vi /opt/iParapheur/tomcat/shared/classes/iparapheur-global.properties
```

```
#####################################################################
#   Parametrage i-Parapheur (webapp iparapheur.war)              #
#                            copyleft 2013-2017 Libriciel SCOP  #
#####################################################################

#################################
# URL de localisation de l'applet de signature LiberSign
parapheur.signature.applet.url=/applets

#################################
## PREFERENCES UTILISATEUR     ##
parapheur.ihm.options.password.show=true
parapheur.ihm.options.theme.show=true
parapheur.ihm.options.signature.show=true
parapheur.ihm.options.langue.show=false

#################################
## IHM ADMINISTRATION          ##
parapheur.ihm.admin.users.connected.threshold=10
# Lien vers console Javascript (admin)
parapheur.ihm.admin.mode.advanced=false
# Timeout (600s par defaut) avant alarme (gestion de dossier) si dossier anormalement bloque
parapheur.ihm.admin.dossier.locked.notify=600

#################################
## BAS DE PAGE : Acces support ##
parapheur.ihm.contact.support.text=Propulsé par Libriciel SCOP
parapheur.ihm.contact.support.url=https://www.libriciel.fr/

#################################
## Nombre max de documents principaux dans un dossier (v4.3)
parapheur.ihm.creerdossier.maindocuments.max=6

#################################
## THEMES de l'interface       ##
parapheur.ihm.themes.directory=/var/www/parapheur/themes
parapheur.ihm.themes.disponibles=default,amelia,cerulean,cosmo,darkly,flatly,journal,lumen,readable,simplex,spacelab,superhero,united,yeti

#################################
## VISIBILITE DES DOSSIERS     ##
# Visibilité par défaut de la collectivite principale (racine)
parapheur.ihm.creerdossier.visibilite.defaut=confidentiel
# Visibilité par défaut par tenant (ex. {"tenant1":"confidentiel"})
parapheur.ihm.creerdossier.visibilite.defaut.tenant={}
# Visibilités disponibles à la création (public,group,confidentiel)
parapheur.ihm.creerdossier.visibilite.valeurs=public,group,confidentiel

#################################
## MANUEL UTILISATEUR          ##
parapheur.ihm.aide.utilisateur.text=Télécharger le manuel utilisateur (PDF)
parapheur.ihm.aide.utilisateur.url=/docs/manuel.pdf

# nombre max de lignes possibles par page: 0 signifie pas de limite (dangereux!)
parapheur.ihm.dashboard.lignes.max=50
parapheur.ihm.confirmbox.read=true

#################################
## URL du i-Parapheur          ##
parapheur.url=iparapheur.dom.local/iparapheur

#################################
# Authent. Par certificat client
# obligatoire ?
parapheur.auth.certificate.mandatory=false
# URL de l'authentification par certificat
parapheur.auth.certificate.url=secure-iparapheur.dom.local

#################################
# VISIONNEUSE XEMELIOS         ##
# Mode de presentation d'apercu sur collectivite racine (visuelpdf|xemelios)
parapheur.ihm.apercu.helios=xemelios
# Mode de presentation d'apercu par tenant (ex. {"tenant1":"visuelpdf"})
parapheur.ihm.apercu.helios.tenant={}

#Pour connexion via sso lemonldap
parapheur.auth.external.header.name=Auth-User
#Soit de la forme :
parapheur.auth.external.header.regexp=.*
#Ou bien, pour expression régulière plus complexe, faire un "groupe" entre parenthèse (.*)
# Exemple : parapheur.auth.external.header.regexp=username=(.*)

## Nouveautes v4.4
parapheur.ihm.password.strength=nl6
parapheur.ihm.aide.libersign.url=https://client.libriciel.fr/
parapheur.extension.libersign.firefox.url=https://libersign.libriciel.fr/extension.xpi
```

```
parapheur.extension.libersign.chrome.url=https://chrome.google.com/webstore/detail/jligpldajocilccnnokfnghlamfhnppc
parapheur.extension.libersign.native.url=https://libersign.libriciel.fr/libersign.exe

## Nouveautées v4.5.0
# Afficher ou non l'onglet d'archive
parapheur.ihm.archives.show=false

## Nouveautés v4.5.2
# Permet de cacher la configuration d'attestation de signature
parapheur.ihm.attest.show=false
```

4.10.3. Ressources statiques WEB (thèmes, etc)

Une archive est à dépaqueter dans `/var/www` (répertoire à créer si absent) :

```
mkdir -p /var/www ; cd /var/www
tar xzf /opt/_install/confs/var-www-parapheur.tar.gz
```

Dans `/var/www` , on trouve donc les répertoires suivants :

- `parapheur/applets` : contient l'applet LiberSign
- `parapheur/docs` : documentation PDF téléchargeable (manuel utilisateur)
- `parapheur/error_pages` : modèles de pages d'erreur
- `parapheur/themes` : thèmes graphiques BootStrap. (voir annexe pour la personnalisation)

NB : pour faciliter la maintenance de la politique de signature (ACs de confiance à jour en particulier) et à cause des changements incessants du comportement de JAVA sur le poste de travail (plug-in de navigateur, au fil des mises à jour imposées par Oracle), l'applet de signature LiberSign n'est plus packagée dans les composants applicatifs de i-Parapheur. Elle fait l'objet d'un paquetage distinct, qui pourra être ainsi mis à jour périodiquement par tâche planifiée (CRON), de la même manière que les ACs de confiance pour le frontal Web HTTPS.

```
cd /var/www/parapheur/applets
tar xzf /opt/_install/iParapheur-v4.5.xx/libersignApplet-*.tar.gz
```

Il faut initialiser la mise à jour de LiberSign, et la rendre disponible aussi pour le mode d'applet Java (Microsoft-IE) :

```
cd /var/www/parapheur/applets ; rm SplittedSignatureApplet.jar
cd /var/www/parapheur/libersign ; chmod +x make.sh
./make.sh PROD
cd ../applets/; ln -s ../libersign/update/applet/SplittedSignatureApplet.jar
```

NB : le guide utilisateur est rendu disponible dans le menu « Aide/à-propos ». Par exemple, après avoir récupéré le fichier PDF du manuel utilisateur 4.5, puis l'avoir déposé dans le répertoire `/opt/_install` :
(exemple à adapter selon contexte d'exploitation!)

```
cp /opt/_install/*.pdf /var/www/parapheur/docs/manuel.pdf
```

Enfin, externaliser le fichier WSDL de contrat Web-Services servi statiquement par NginX :

```
mkdir -p /var/www/parapheur/alfresco ; cd /var/www/parapheur
ln -s /opt/iParapheur/tomcat/webapps/alfresco/WEB-INF/wsdl/iparapheur.wsdl alfresco/
chown -R nginx: /var/www/parapheur
```

4.11. Divers réglages finaux

4.11.1. Fichiers de configuration – optimisation

La génération PDF (visuels d'impression, calques,...) nécessite les fichiers suivants :

```
cd /opt/_install/confs ; cp srgb.profile verdanai.ttf /opt/iParapheur/
```

Éditer le script de contrôle `alfresco.sh` (attention aux directives ulimit et à INSTALLDIR), contrôle la conformité du début du script avec ce qui suit :

```
#!/bin/sh

ulimit -Hn 16384
ulimit -Sn 16384

# Disabling SELinux if enabled
if [ -f "/usr/sbin/getenforce" ] && [ `id -u` = 0 ] ; then
    selinux_status=`/usr/sbin/getenforce`
    /usr/sbin/setenforce 0 2> /dev/null
fi

INSTALLDIR=/opt/iParapheur
cd $INSTALLDIR
```

4.11.2. Remplacer OpenOffice.org par LibreOffice

Le bundle « Alfresco 3.4.c Community » livre une version relativement ancienne d'OOo, qui fonctionne pour les opérations courantes (HTML, ODT,...). LibreOffice offre de bien meilleurs filtres pour gérer certains formats de fichier fermés (.DOC .DOCX et autres).
D'une manière générale, considérer le téléchargement de la dernière version stable (dite « Still » ou « stable » dans le jargon LibreOffice, soit v5.2.7 au 13 jui. 2017, par opposition à la version dite « Fresh » ou « Evolution »). i-Parapheur v4.5 supporte les versions 4.2.8 et 5.2.7 de LibreOffice.

http://download.documentfoundation.org/libreoffice/stable/ http://download.documentfoundation.org/libreoffice/old/

```
cd /opt/_install/
 wget <url-de-version-5.2.7>
```

Exemple sur LibreOffice 5.2.7 sur Debian/Ubuntu :

```
wget http://download.documentfoundation.org/libreoffice/stable/5.2.7/deb/x86_64/LibreOffice_5.2.7_Linux_x86-64_deb.tar.gz
```

Exemple sur LibreOffice 5.2.7 sur CentOS / RedHat, adapter avec la version « de production » disponible :

```
wget http://download.documentfoundation.org/libreoffice/stable/5.2.7/rpm/x86_64/LibreOffice_5.2.7_Linux_x86-64_rpm.tar.gz
```

Mise en œuvre :

- Ajouts de dépendances spécifiques :
  - Sur Ubuntu, ajouter si absents les packages `libcups2` `libfontconfig1` `libcairo2` `libgl1-mesa-glx` `libsm6`.

    ```
    apt-get install libcups2 libfontconfig1 libcairo2 libgl1-mesa-glx libsm6
    ```

  - Sur Debian 7, ajouter au préalable le package `libdbus-glib`.

    ```
    apt-get install libdbus-glib-1-2
    ```

  - Sur CentOS 6 / 7, ajouter si absent le package `gnome-vfs2` ainsi que `cups-libs`.

    ```
    yum install gnome-vfs2 cups-libs
    ```

- Installation selon préconisations http://fr.libreoffice.org/home/lisez-moi/#Linux
  - Sur CentOS / RedHat :

    ```
    tar xzf LibreOffice_*rpm.tar.gz
    cd LibreOffice_5.2.7_Linux_x86-64_rpm/RPMS
    yum localinstall *.rpm
    ```

  - Sur Debian / Ubuntu :

    ```
    tar xzf LibreOffice_*.tar.gz
    cd LibreOffice_5.2.7_Linux_x86-64_deb/DEBS
    dpkg -i *.deb
    ```

- Les commandes suivantes permettent de remplacer OpenOffice.org par LibreOffice : `bash cd /opt/iParapheur rm -rf openoffice ln -s /opt/libreoffice5.2 openoffice`

> **ATTENTION** : Dans le cas d'une mise en place sur Debian 9, il faut supprimer la librairie `libz` fournie par alfresco, et utiliser celle du système.
>
> Pour cela, il suffit de supprimer toute référence à cette librairie du dossier `/opt/iParapheur/common/lib` :
>
> ```
> rm /opt/iParapheur/common/lib/libz.so*
> ```

4.11.3. Mise en place de « logrotate »

D'abord, adapter les traces produites par Alfresco, en éditant le fichier *`custom-log4j.properties`* , changer la directive `log4j.appender.File` et commenter `log4j.appender.File.DatePattern` :

```
vi /opt/iParapheur/tomcat/shared/classes/alfresco/extension/custom-log4j.properties
```

```
###### File appender definition #######

# log4j.appender.File=org.apache.log4j.DailyRollingFileAppender
log4j.appender.File=org.apache.log4j.FileAppender
log4j.appender.File.File=/var/log/alfresco/alfresco.log
log4j.appender.File.Append=true
# log4j.appender.File.DatePattern='.'yyyy-MM-dd
```

```
log4j.appender.File.layout=org.apache.log4j.PatternLayout
log4j.appender.File.layout.ConversionPattern=%d{ABSOLUTE} %-5p [%c] %m%n
```

On s'appuie sur `logrotate` ; ajouter le fichier `logrotate-iparapheur.conf` :

```
cp /opt/_install/confs/logrotate-iparapheur.conf /opt/iParapheur/
```

Voici le contenu pour exemple:

```
/var/log/alfresco/alfresco.log {
  weekly
  rotate 10
  copytruncate
  compress
  notifempty
  missingok
}

/var/log/alfresco/tomcat/logs/catalina.out {
  weekly
  rotate 10
  copytruncate
  compress
  notifempty
  missingok
  size 50M
}
```

> *Attention* : Les ressources doivent être libérées par Java/TOMCAT lors de la « bascule de fichiers par logrotate ». Ce script sera donc exécuté **manuellement** lorsque i-Parapheur est stoppé (via CRON, voir plus loin).

> *Remarque* : Si ce script est déposé dans `/etc/logrotate.d/` , on ne maîtrise pas vraiment l'heure d'exécution réelle: par défaut logrotate est déclaré dans `/etc/cron.daily` .
> Or « cron » délègue l'exécution des tâches de cron.daily à « anacron » (voir le contenu de `/etc/crontab`).
> Si anacron est absent alors cron.daily est exécuté à 6h25 par défaut, mais si anacron est présent, alors le lancement est défini par le contenu de `/etc/anacrontab` (soit une fois par jour, délai de 5 minutes).
> A noter enfin que Anacron est lui-même lancé par cron, par sa déclaration dans `/etc/cron.d/anacron`, qui n'exécute anacron qu'à 7h30 ! Ouf...

Pour les autres fichiers de traces qui sont datés par TOMCAT, on ne peut pas faire grand chose via `logrotate` , d'où le script installé ci-après pour nettoyer les logs « trop vieilles » supérieures à quelques jours.

```
cp /opt/_install/confs/nettoieLogs.sh /opt/iParapheur/
chmod +x /opt/iParapheur/nettoieLogs.sh
```

4.11.4. Réglage de la durée de session utilisateur

OPTION Il est possible de modifier la durée de session par défaut (30 minutes) .

Dans `/opt/iParapheur/tomcat/webapps/iparapheur/WEB_INF/web.xml` (vers la ligne n°114) :

```
<session-timeout>30</session-timeout>
```

> *Remarque* : cette 1ère opération sera à reporter lors de chaque redéploiement de iparapheur.war (mises-à-jour...)

Pour la 2nde opération ajouter à la fin du fichier `alfresco-global.properties` :(exemple avec 1 heure)

```
authentication.ticket.validDuration=P1H
```

Pour mémoire, extrait de la JavaDoc de la classe `Duration` utilisée par Alfresco, qui offre ce réglage :

```
 * The lexical representation of duration is PnYnMnDTnHnMnS.
 *
 * P is a literal value that starts the expression
 * nY is an integer number of years followed by the literal Y
 * nM is an integer number of months followed by the literal M
 * nD is an integer number of days followed by the literal D
 * T is the literal that separates the date and time
 * nH is an integer number of hours followed by a literal H
 * nM is an integer number of minutes followed by a literal M
 * nS is a decimal number of seconds followed by a literal S
```

4.11.5. Service Web Visionneuse PESv2

Accessoire précieux pour l'instruction des flux PESv2 (HELIOS de la DGFiP), la visionneuse XEMELIOS (logiciel édité par la DGFiP) est déployée de la façon suivante :

- Déploiement des archives `tar.gz` :

```
cd /opt/
tar xzf _install/iParapheur-v4.5.xx/visionneuse-Xemelios.tar.gz
```

- Création du fichier `setenv.sh` rendu éxecutable :

```
echo 'JAVA_HOME=/opt/jdk1.8.0_161' > /opt/visionneuse-Xemelios/bin/setenv.sh
chmod +x /opt/visionneuse-Xemelios/bin/setenv.sh
```

- Ajout script `xemwebview` dans `/etc/init.d/` :

```
cp /opt/visionneuse-Xemelios/xemwebview /etc/init.d/
chmod +x /etc/init.d/xemwebview
```

- Ajout script de `purge-xemwebview.sh` :

```
cp /opt/_install/confs/purge-xemwebview.sh /opt/iParapheur/
chmod +x /opt/iParapheur/purge-xemwebview.sh
```

- Ajout répertoires temporaires de travail :

```
mkdir -p /var/tmp/bl-xemwebviewer/xwv-cache
mkdir -p /var/tmp/bl-xemwebviewer/xwv-extract
mkdir -p /var/tmp/bl-xemwebviewer/xwv-shared
```

- 1er démarrage de l'application, et surveiller la bonne fin de ce démarrage (peut prendre jusque ~5min, voir chapitre 5.1 pour les détails) :

```
cd /opt/iParapheur ; /etc/init.d/alfresco start
tail -f tomcat/logs/catalina.out
```

- Arrêt de l'application, puis édition du fichier `alfresco.xml` :

```
/etc/init.d/alfresco stop
vi tomcat/conf/Catalina/localhost/alfresco.xml
```

- Y ajouter en fin de fichier la partie surlignée (la nouvelle balise `Environment`) :

```
  <Environment name="iparapheur-blex/bl-xemwebviewer-url"
               value="http://iparapheur.local.dom/bl-xemwebviewer"
               type="java.lang.String" override="false"/>
</Context>
```

A ce stade, c'est achevé. Le lancement du service :

```
/etc/init.d/xemwebview start
```

Arrêt du service :

```
/etc/init.d/xemwebview stop
```

4.11.6. Autres réglages possibles

OPTION Configuration en mode multi-tenants, aujourd'hui DÉPRÉCIÉ : Si nécessaire, l'application i-Parapheur peut fonctionner en « colocation » de collectivité, grâce à l'activation du mode « multi-tenancy » d'Alfresco. L'activation de ce mode s'effectue avec les manipulations suivantes :

```
cd /opt/iParapheur/tomcat/shared/classes/alfresco/extension/mt
mv mt-context.xml.sample mt-context.xml
mv mt-admin-context.xml.sample mt-admin-context.xml
mv mt-contentstore-context.xml.sample mt-contentstore-context.xml
```

Se référer au manuel d'administration (i-Parapheur_v3.2_Admin-multiCollectivite_v1.2.pdf) pour l'exploitation de cette fonctionnalité.

La colocation (multi-tenancy) est limitée arbitrairement à 99 collectivités maximum par l'engineering Alfresco. Il paraît possible d'outrepasser cette limitation en modifiant certains paramètres internes (en particulier la gestion de la taille du cache). Ce dépassement n'est pas supporté par Libriciel. En cas de dépassement de ce maximum, il est recommandé de mettre en place plusieurs serveurs, et répartir les collectivités « locataires » sur ces différentes instances.

> *Attention* : un « tenant » ainsi créé**ne peut pas être supprimé** . C'est une limitation connue d'Alfresco.

> *Remarque* : dans ce mode, aucun couplage annuaire (LDAP, AD), ni SSO n'est possible.

5. VALIDATION DE L'INSTALLATION

5.1. 1er démarrage

Après démarrage de l'application (voir le chapitre suivant pour la commande de démarrage) ou reboot (redémarrage) du serveur, les manipulations suivantes permettent de s'assurer que l'installation s'est bien déroulée.

```
/etc/init.d/alfresco start
```

> *Remarque* : le premier démarrage peut prendre jusqu'à 5 minutes selon la quantité de ressources allouées au serveur. Les démarrages suivants sont plus rapides (de 45 à 100 secondes).

Rappel : La vérification du bon démarrage de TOMCAT peut se faire en examinant les traces `catalina` :

```
tail -f /var/log/alfresco/tomcat/logs/catalina.out
```

Un serveur fonctionnel enregistre dans ces traces le message suivant `INFO: Server startup in xxxx ms`.
Si ce message n'apparaît pas, contrôler l'activité CPU du processus Java de TOMCAT (par exemple avec la commande `top`) : en effet, le serveur peut ne pas avoir fini de démarrer.
Dans le cas contraire (activité nulle), les traces `catalina.out` sont assez verbeuses, et font rapidement état du problème de démarrage.

OPTION Redémarrage du serveur :

Pour vérifier que les services sont bien actifs: `reboot` puis login, `sudo -s`. Chacune des commandes suivantes doit donner un résultat.

```
ps aux | grep -i office
ps aux | grep -i tomcat
ps aux | grep -i mysql
```

5.2. Contrôle des services réseau

La commande suivante liste les ports réseau ouverts en écoute (prêts à servir) :

```
netstat -antup | grep LISTEN
```

Les lignes intéressantes, respectivement pour NginX, MySQL, LibreOffice, i-Parapheur, Xemelios :

```
tcp  0 0 0.0.0.0:80      0.0.0.0:* LISTEN 1599/nginx.conf
tcp  0 0 0.0.0.0:443     0.0.0.0:* LISTEN 1599/nginx.conf
tcp  0 0 127.0.0.1:3306  0.0.0.0:* LISTEN 1353/mysqld
tcp  0 0 127.0.0.1:8100  0.0.0.0:* LISTEN 5715/soffice.bin
tcp6 0 0 127.0.0.1:50500 :::*      LISTEN 5627/java
tcp6 0 0 127.0.0.1:8005  :::*      LISTEN 5627/java
tcp6 0 0 127.0.0.1:9005  :::*      LISTEN 5796/java
tcp6 0 0 :::8009         :::*      LISTEN 5627/java
tcp6 0 0 :::50508        :::*      LISTEN 5627/java
tcp6 0 0 :::35183        :::*      LISTEN 5627/java
tcp6 0 0 :::8080         :::*      LISTEN 5627/java
tcp6 0 0 :::8081         :::*      LISTEN 5627/java
tcp6 0 0 :::35800        :::*      LISTEN 5627/java
tcp6 0 0 :::9009         :::*      LISTEN 5796/java
tcp6 0 0 :::9080         :::*      LISTEN 5796/java
```

> *Remarque* : on constate que MySQL n'écoute qu'en local, parce qu'ici c'est un service dédié à l'application i-Parapheur. Donc pas de connexion distante au serveur de base de données.

5.2.1. Contrôle des accès Web HTTPS

Pré-requis : un navigateur Web sur un poste avec accès au serveur.

Dans un navigateur web, les URLs du tableau ci-après (personnaliser le nom de domaine selon le nom du serveur) doivent donner un écran de connexion (voir les illustrations suivantes) :

N°	URL	Partie testée
1	http://iparapheur.dom.local:8080/alfresco/	Serveur d'application TOMCAT en direct. Si KO, revoir l'installation en chapitre 4. Si OK : en profiter pour se connecter en admin, y définir son certificat de connexion.

2	https://iparapheur.dom.local/iparapheur/	Si N° 1 OK, pour valider l'installation du serveur Web (chapitre 3).
3	https://iparapheur.dom.local	Si N° 2 OK, pour valider la redirection HTTPS

5.2.2. Contrôle d'accès Web-SERVICE HTTPS

Pré-requis : un navigateur Web sur un poste avec accès au serveur, ainsi qu'un certificat électronique client exploitable par le navigateur.

Tester l'accès HTTPS sur:
https://secure-iparapheur.ma-collectivite.fr/ws-iparapheur

Séquence attendue :

- Le navigateur doit réclamer un certificat client (et sans doute râler au préalable s'il ne connait pas l'AC du certificat HTTPS proposé sur ce domaine)
- Sur sélection puis validation d'un certificat client, le message d'erreur suivant apparaît `java.lang.RuntimeException: Utilisateur inconnu`) : c'est parfaitement logique et normal.
- Sur sélection du certificat client associé au compte 'admin' (voir test 1), la connexion sur l'URL ci-dessus affiche un tableau avec points d'entrée WSDL pour les applications métier.

Concernant la partie Web-Services, se référer au manuel administrateur pour la constitution des keystores à utiliser dans le logiciel métier (qui sera techniquement un « client » du i-Parapheur).

6. GUIDE (RAPIDE) D'EXPLOITATION

6.1. Commandes de lancement / arrêt

Lancement de i-Parapheur :

```
/etc/init.d/alfresco start
```

Arrêt de i-Parapheur :

```
/etc/init.d/alfresco stop
```

> *Rappel* : La toute première fois, le lancement va créer les données initiales de i-Parapheur dans la base de données et le système de fichiers (`alf_data`) ; ce processus est relativement long (2 à 5 minutes).

La vérification du bon démarrage de TOMCAT peut se faire en examinant les traces `catalina.out` :

```
tail -f /var/log/alfresco/tomcat/logs/catalina.out
```

Un serveur fonctionnel enregistre dans les traces ce message `INFO: Server startup in xxxx ms` .

6.2. Paramétrage du service de messagerie SMTP

Pour configurer les notifications par mail auprès des acteurs de i-Parapheur.

```
vi /opt/iParapheur/tomcat/shared/classes/alfresco-global.properties
```

Les paramètres de connexion SMTP suivants sont disponibles (à ajouter en fin de fichier par commodité) :

```
#
# Outbound Email Configuration
#-----------------------------
mail.host=monSMTPjoliQuiMarche.dom.local
mail.port=25
mail.username=anonymous
mail.password=
mail.encoding=UTF-8
mail.from.default=ne-pas-repondre-SVP@ma-collectivite.org
mail.smtp.auth=false
```

6.3. Exploitation - sauvegarde des données

6.3.1. Remarque préliminaire

la définition d'une politique de sauvegarde dans le cadre d'une procédure de type PRA/PCA est à la charge de l'exploitant.
Ce chapitre ne fait que **proposer** un process de sauvegarde, et n'a pas vocation à se substituer à la définition d'un PRA ou PCA.
Il conviendra donc que l'exploitant vérifie lui-même le bon fonctionnement de ses procédures de sauvegarde et restauration.

6.3.2. Exemple de mise en oeuvre

Copie du script de sauvegarde dans le répertoire d'installation :

```
cp /opt/_install/confs/backup_parapheur.sh /opt/iParapheur/
cp /opt/_install/confs/send_backup.sh /opt/iParapheur/bin
chmod +x /opt/iParapheur/bin/send_backup.sh /opt/iParapheur/backup_parapheur.sh
```

> *IMPORTANT* : Les backups DOIVENT se faire à froid (application arrêtée), afin que les données d'entrepôt (système de fichier) ne soient pas désynchronisées du contenu de la base de données.

> *Remarque* : Cas de MySQL déporté (pas sur le même serveur que i-Parapheur), attention à l'appel de `mysqldump` ! De plus, ***si le serveur MySQL est en v5.6*** , la commande peut échouer avec un message du genre : `mysqldump: Couldn't execute 'SET OPTION SQL_QUOTE_SHOW_CREATE=1': You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'OPTION SQL_QUOTE_SHOW_CREATE=1' at line 1 (1064).` La solution est d'installer un client MySQL dans une version compatible 5.6 du serveur MySQL.

6.3.3. Tâches planifiées d'exploitation

Régler le CRON afin que la procédure s'exécute tous les jours à partir de 23h00 (dans cet exemple) :

```
crontab -e
```

Contenu:

```
MAILTO=''

## COUPURE DU SERVICE au public a 23h00 (selon version: /sbin/ , /bin/, ou initctl ou 'service ..')
0 23 * * * /sbin/stop nginx >/dev/null 2>&1
1 23 * * * /etc/nginx/ssl/recup_crl_nginx.sh /etc/nginx/ssl

## ARRET DES APPLICATIONS, le plus proprement possible
5 23 * * * /etc/init.d/alfresco stop >/dev/null 2>&1
10 23 * * * /etc/init.d/xemwebview stop
15 23 * * * /usr/bin/killall -q -e -g -s 9 /opt/jdk1.8.0_161/bin/java >/dev/null 2>&1
16 23 * * * /usr/bin/killall -q -e -s 9 /opt/iParapheur/openoffice/program/soffice.bin >/dev/null 2>&1
18 23 * * * /bin/rm -f /var/run/parapheur.pid >/dev/null 2>&1

## inutile si NTPD fonctionne
# 19 23 * * * /usr/sbin/ntpdate ntp.ubuntu.com pool.ntp.org >/dev/null 2>&1

## BACKUP
20 23 * * * /usr/bin/mysqlcheck -o alfresco -u alfresco -palfresco >/dev/null 2>&1
21 23 * * * /opt/iParapheur/backup_parapheur.sh >/dev/null 2>&1
22 23 * * 1 /opt/iParapheur/warn_needPurge.sh  >/dev/null 2>&1

## NETTOYAGE
40 2 * * * /usr/bin/logrotate /opt/iParapheur/logrotate-iparapheur.conf
41 2 * * * /opt/iParapheur/nettoieLogs.sh
42 2 * * * /opt/iParapheur/nettoieEntrepot.sh
43 2 * * * /bin/rm -f /var/tmp/bl-xemwebviewer/xwv-shared/

## Relance application
44 2 * * * /bin/mkdir -p /var/tmp/bl-xemwebviewer/xwv-cache /var/tmp/bl-xemwebviewer/xwv-extract /var/tmp/bl-xemwebviewer/xwv-
shared;/etc/init.d/xemwebview start
45 2 * * * /etc/init.d/alfresco start >/dev/null 2>&1

## OUVERTURE du SERVICE au public pour 3h00
59 2 * * * /sbin/start nginx >/dev/null 2>&1

## VISIONNEUSE XEMELIOS: test purge periodique tous les 1/4 d'heure
0,15,30,45 * * * * /opt/iParapheur/purge-xemwebview.sh

## ENTRETIEN de LiberSign
1 * * * * /var/www/parapheur/libersign/make.sh PROD
```

Explication sommaire : Le 1er groupe stoppe HTTPS et met à jour les ACs et CRLs ; puis arrêt de l'application, rotation des logs. Enfin, une commande de mise à jour de l'heure système, et lancement du script de backup, avant redémarrage de l'application.
En fin, entretien périodique des espaces tampon de visionneuse Xemelios, et entretien de LiberSign.

Enfin, éditer si nécessaire le fichier `send_backup.sh` pour régler les paramètres de serveur distant (FTP ou CIFS), si on souhaite externaliser la sauvegarde ainsi réalisée vers un serveur de fichier distant.

6.3.4. Restauration d'une sauvegarde

Les étapes suivantes sont généralement appliquées (bonnes pratiques) :

- le fichier `alfresco-global.properties` doit être modifié temporairement: positionner le paramètre (l'ajouter si absent, par exemple à proximité de "dir.root" en début de fichier)

  ```
  index.recovery.mode=FULL
  ```

- Arrêter l'application i-Parapheur (voir procédure ci-dessous)
- Deux façons de restaurer les données:
  - Soit manuellement, avec un répertoire de backup, qui contient 3 fichiers:
  - `backup_parapheur.sh` : le script de backup, utile car il contient les éléments techniques de connection BDD MySQL.
  - `alfresco.sql` : dump de la base 'alfresco' (par défaut), à restaurer par suppression préalable de la base existante, puis recréation de la base (vierge) + injection des données de ce "dump".
  - `tgz` : archive compressée de l'entrepôt.
    Supprimer le contenu de `/opt/iParapheur/alf_data`, puis décompresser l'archive, par exemple:

  ```
  cd /opt/iParapheur ; tar xzf /le-bon-chemin-backup/parapheur.tar.gz
  ```

  - Soit on a "confiance" dans le script de restauration `restore_parapheur.sh` préalablement déposé dans le répertoire `/opt/iParapheur/bin` (*RAPPEL*, ce n'est qu'un 'snippet' imparfait, un exemple de code fourni *sans garantie de fonctionnement*):

d'abord vérifier son contenu, l'adapter au contexte d'exploitation, en particulier les éléments de connexion BDD). Puis exécuter le script.

- Les répertoires contenant les indexes LUCENE ne sont plus savegardés par défaut. Donc effacer si présent les répertoires:
  - `/opt/iParapheur/alf_data/lucene-indexes` (si présent)
  - `/opt/iParapheur/alf_data/backup-lucene-indexes` (si présent)
- Lancer l'application (voir procédure ci-dessous), et patienter...
- Surveiller les traces de `catalina.out`, parce que la ré-indexation prend beaucoup de temps (quelques minutes à quelques heures, selon la taille de l'entrepôt).
  Seulement **à la fin de l'indexation**, le service IHM sera opérationnel (identification utilisateur, etc.).
- Après démarrage, dans le fichier `alfresco-global.properties` remettre le paramètre à une valeur normale

```
index.recovery.mode=AUTO
```

> *Remarque* : dans le guide d'administration, ou lors des ateliers de formation, il est dit à plusieurs reprises que i-Parapheur n'est pas une GED, qu'on ne doit pas l'utiliser ainsi. Autrement dit, il convient d'extraire de i-Parapheur les dossiers lorsqu'ils sont arrivés au terme de leur validation, au fur et à mesure.
> Plus il y a de documents dans l'entrepôt de i-Parapheur, plus la sauvegarde prend de la place et du temps. Et mécaniquement, on allonge d'autant la procédure de restauration et surtout de ré-indexation.

> *APPEL à CONTRIBUTION* : les experts en procédure de restauration de données sont plus que bienvenus pour améliorer ce processus, et contribuer leurs bonnes pratiques (licence CeCILLv2, GNU GPL ou équivalent).
> C'est aussi comme cela que le produit se consolide, pour le bénéfice de la communauté des utilisateurs.

6.4. Surveillance – monitoring des services

Il convient de surveiller le service et ses dépendances, pourquoi pas avec des sondes de type Nagios sur :

- l'état des services : LibreOffice, TOMCAT, Apache/Nginx,
- consommation globale : CPU , RAM , partitions disques

Se référer aux guides d'administration Nagios (ou autre: Zabbix, etc.) pour les pratiques de mise en oeuvre de sondes efficientes.

Exemple : pour la consommation disque (si absence de sonde), il est possible de positionner un script BASH très simple nommé `/opt/iParapheur/espace.sh`, déclenché par CRON:

```
#!/bin/bash
# version 1.0.0 par Stephane VAST - Libriciel SCOP

df -h | mail -s "Espace disque sur mon serveur maCollectivite.fr" exploit@maCollectivite.fr
```

Ce script est alors appelé régulièrement par CRON, avec une directive équivalente à :

```
0 8 * * 1 /opt/iParapheur/espace.sh >/dev/null 2>&1
```

6.5. Procédure de mise à jour mineure

Cela ne concerne QUE des mises-à-jour dites « mineures » (4.1.x à 4.1.y par exemple). Elle ne fonctionne pas pour une mise à jour depuis une version 3.0.xx (car changement de socle Alfresco).

Dans le package tar.gz sont livrés un script `iparaph-updateAMP.sh`, et un guide `LISEZ-MOI.txt` :

```
cp /opt/_install/confs/iparaph-updateAMP.sh /opt/iParapheur
```

Lire le fichier `LISEZ-MOI.txt` : il précise la liste des opérations à effectuer, selon l'écart de versions de produit.

NB : pour l'exécution du script `custom-wsdl.sh` (voir § 4.6 sur le connecteur Web-Services) :

```
cd /opt/iParapheur
./custom-wsdl.sh secure-iparapheur.ma-collectivite.fr
```

Cas de mise à jour depuis Ubuntu 10.04 LTS (version supportée jusqu'en avril 2015) : le passage vers la version Ubuntu 12.04 LTS se fait assez simplement (opération identique pour migration 14.04 ou vers 16.04) :

```
apt-get update
apt-get dist-upgrade                        # éventuel reboot si update du kernel
apt-get install update-manager-core
vi /etc/update-manager/release-upgrades     # positionner : Prompt=lts
```

```
do-release-upgrade
```

> *NB* : depuis la version 3.4 et plus, il y a une dépendance sur GhostScript, voir chapitres précédents.

7. ANNEXES

7.1. Polices TTF Microsoft sur RedHat/CentOS

Sur RedHat/CentOS... sans paquet RPM fournissant ces polices de caractère, télécharger et constituer ce paquet « à la main », avec une connexion Internet opérationnelle !

Toujours en mode super-utilisateur, se placer dans le répertoire de travail, installer les pré-requis et télécharger le projet depuis le site SourceForge.net :

```
mkdir -p /opt/_install/msttcorefonts && cd /opt/_install/msttcorefonts
yum install wget cabextract rpm-build chkfontpath ttmkfdir
wget http://corefonts.sourceforge.net/msttcorefonts-2.5-1.spec
```

Construction et installation du paquet RPM (exemple de chemin, vu sur CentOS 5) :

```
rpmbuild -ba msttcorefonts-2.5-1.spec
yum localinstall --nogpgcheck /usr/src/redhat/RPMS/noarch/msttcorefonts-2.5.1.noarch.rpm
/sbin/service xfs reload                              # inutile sur CentOS 6 ou 7
fc-cache -fv
```

NB : CentOS6 :
https://oimon.wordpress.com/2011/09/05/msttcorefonts-on-rhel6-centos-6-sl6/
Pas de XFS à redémarrer.

NB : CentOS7 : activer EPEL pour installer `cabextract` :
http://www.cyberciti.biz/faq/installing-rhel-epel-repo-on-centos-redhat-7-x

En cas de souci, voir le site du projet : http://corefonts.sourceforge.net/

Astuce : D'éventuelles fontes complémentaires 'particulières' peuvent être définies en les copiant dans le répertoire : `/opt/iParapheur/openoffice/basis3.2/share/fonts/truetype/` (adapter selon contexte d'installation)

D'une manière générale, ajouter des polices au système suffit avec (par exemple) :

```
mkdir -p /usr/share/fonts/truetype/mesjoliesfontes
cp /tmp/fonts/*.ttf /usr/share/fonts/truetype/mesjoliesfontes/
fc-cache -f -v
```

Pour les polices Office2007 (Calibri,...) : http://www.oooninja.com/2008/01/calibri-linux-vista-fonts-download.html

Une autre ressource d'informations sur l'installation de polices de caractères supplémentaires :

http://help.accusoft.com/PCC/v8.1/HTML/How%20to%20Install%20Microsoft%20Fonts%20on%20Linux.html

7.2. LibreOffice sur un port particulier

Lorsque l'on choisit un port différent du port par défaut (8100), Alfresco n'honore pas cette configuration (petit bug du moteur).
Application arrêtée (évidemment), il faut modifier la définition du bean suivant :

```
cd /opt/iParapheur/tomcat/webapps/alfresco/WEB-INF/classes/alfresco/subsystems
vi OooDirect/default/openoffice-transform-context.xml
```

Adapter dans ce fichier (vers la ligne 58) :

```
<bean id="openOfficeConnection" class="net.sf.jooreports.openoffice.connection.SocketOpenOfficeConnection"/>
```

par :

```
<bean id="openOfficeConnection" class="net.sf.jooreports.openoffice.connection.SocketOpenOfficeConnection">
    <constructor-arg>
        <value>${ooo.port}</value>
    </constructor-arg>
</bean>
```

7.3. Ressources pour couplage LDAP

Voir la littérature sur Internet, notamment concernant le couplage « alfresco - ldap »:

- http://wiki.alfresco.com/wiki/Alfresco_Authentication_Subsystems

- http://wiki.alfresco.com/wiki/The_Synchronization_Subsystem
- http://www.ochounos.com/#blog/6

D'autres sources d'inspiration pour faciliter l'installation :

- http://howtoforge.org/how-to-install-alfresco-community-3.3-on-ubuntu-server-10.04-lucid-lynx
- http://blog.mycroes.nl/2010/04/installing-alfresco-33-on-ubuntu-lucid.html

7.4. Changer la durée de session

Par défaut la durée de session est pré-réglée sur 30 minutes. Pour changer cette valeur, il faut intervenir sur le paramètre `session-timeout`, dans deux fichiers :

- `/opt/iParapheur/tomcat/alfresco/WEB-INF/web.xml` (vers la ligne n°866)
- `/opt/iParapheur/tomcat/iparapheur/WEB-INF/web.xml` (vers la ligne n°100)

Exemple :

```
<session-config>
    <session-timeout>30</session-timeout>
</session-config>
```

7.5. Service POP3

Optionnel, et si la plate-forme compte exploiter la fonctionnalité « mail-Service » de i-Parapheur, il y a besoin d'un service POP3 accessible (et d'au moins une boite à lettres POP configurée à l'usage de i-Parapheur).

En l'absence de service de boites POP3, on peut en installer un comme suit :

```
apt-get install dovecot-common dovecot-pop3d
```

Contenu de `/etc/dovecot/dovecot.conf` à adapter :

```
protocols = pop3
mail_location = mbox:~/mail:INBOX=/var/mail/%u:INDEX=MEMORY
```

7.6. Lancer i-Parapheur avec utilisateur non 'root'

[A DOCUMENTER... BEAUCOUP MIEUX. POUR LE MOMENT, CE N'EST PAS SUPPORTE...]

L'idée générale est de faire en sorte que le TOMCAT i-Parapheur ne soit pas lancé par `root`, ce qui peut provoquer des réactions cutanées chez certains exploitants paranoïaques.

Attention : les dépendances doivent être cohérentes avec ce passage en mode user-space :

- Créer un utilisateur non-administrateur « iparapheur » par exemple
- Le rendre propriétaire de toute l'arborescence `/opt/iParapheur` et répertoires temporaires
- Idem pour le service Xemelios, et répertoires temporaires
- Attention au `/var/run` ! (ce n'est pas opérationnel tel quel...)

Une des astuces est décrite ci-dessous en éditant le fichier de lancement alfresco.sh :

[CAPTURE ECRAN MANQUANTE]

7.7. I-Parapheur derrière un serveur proxy

L'accès à Internet peut être filtré par un serveur mandataire (proxy), variable d'environnement :

```
http_proxy="http://<username>:<pwd>@<ip>:<port>"  # pour les scripts d'installation
```

i-Parapheur en a besoin pour la connexion au TdT, et éventuellement pour le serveur horodateur.

Éditer le fichier `/opt/iParapheur/alfresco.sh` et ajouter à la clause `JAVA_OPTS` par exemple :

```
-Dhttp.proxyHost=proxy.domain.com -Dhttp.proxyPort=8080 -Dhttp.nonProxyHosts=192.168.0.*|10.1.*
```

Dans cet exemple le proxy est `proxy.domain.com` sur le port `8080`, en précisant que l'accès aux sous-réseaux `192.168.0.0/24` et `10.1.0.0/16` se fait sans passer par le proxy.

En cas de proxy authentifiant (utilisateur « username », et mot de passe « supersecret »), ajouter :

```
-Dhttp.proxyUser=username -Dhttp.proxyPassword=supersecret
```

Cas très particulier de proxy MS ISA Server 2004 « sécurisé » : il utilise une authentification NTLM via le domaine, nom de machine, login, mot de passe ! Installer *NTLM Authorization Proxy Server* :

```
apt-get install ntlmaps
```

Éditer le fichier `/etc/ntlmaps/server.cfg` et positionner les champs suivants :

```
# Dans [GENERAL]
LISTEN_PORT: 5865                   # port du proxy local
PARENT_PROXY: proxy.nom.de.domaine  # le proxy ISA de la collectivité
PARENT_PROXY_PORT: port             # port du proxy ISA de la collectivité
ALLOW_EXTERNAL_CLIENTS: 0           # 1 pour permettre des connections par cet intermédiaire.

# Dans [NTLM_AUTH]
NT_HOSTNAME: Ma_Machine             # nom de la machine connue sur le domaine (Windows)
NT_DOMAIN: Domaine_NT               # le nom de domaine NT de la collectivité
USER: c_est_moi                     # le nom de connexion dans le domaine NT
PASSWORD: mon_mot_de_passe          # le mot de passe correspondant sur le domaine NT
LM_PART:1                           #
NT_PART:1                           #
NTLM_FLAGS: 07820000                #
```

```
/etc/init.d/ntlmaps restart
export http_proxy=http://localhost:5865
```

Éditer le fichier `/opt/iParapheur/alfresco.sh` et régler le proxy sur `localhost:5865`.

> *Autre piste* : Voir le logiciel « cntlm » à utiliser en remplacement de ntlmaps ?

[A FAIRE ? Proposer une configuration basée de***cntlm***]

7.8. Paramétrage avancé du connecteur Web-Services

L'accès à aux Web-services i-Parapheur est *doublement* sécurisé par **MCA + Basic**: *certificat client* d'authentification HTTPS, et identifiant *login/password* vers i-Parapheur. Il est aussi possible de faire automatiquement intervenir le champ 'CN' du certificat dans le login présenté à i-Parapheur : le compte créé dans i-Parapheur devra alors être de la forme `<CN>.<login>`. La syntaxe du séparateur (ici le caractère '.' par défaut) est également paramétrable.

Ce réglage se fait dans le fichier suivant:
`/opt/iParapheur/tomcat/webapps/alfresco/WEB-INF/applicationAcegi.xml`

Aller à la section `bean id="x509AndBasicAuthenticationProcessingFilter"` et adapter la valeur de la propriété `dealWithCertificate` selon le comportement souhaité: `false` ou `true`.

> *NOTE* : L'utilisation des WebServices iParapheur a été expérimentée avec succès avec des clients JAVA (avec JAX-WS), C++ (avec gSOAP), PHP (avec WSO2 wsf-php), C#, NatStar.

Dans le cas de wsf-PHP, il y a un bug de double requête dans la librairie AXIS2/C HTTP embarquée. Le correctif est disponible sous forme de patch à l'URL suivante :
https://issues.apache.org/jira/browse/AXIS2C-1244

7.9. Installation des « swfTools » sur RedHat, Debian, etc.

Inutile depuis l'installeur alfresco 3.4 : le composant logiciel `swftools` (qui fournit l'utilitaire `pdf2swf`) n'est pas disponible dans les dépôts RedHat, ni Ubuntu 10.10 (*swftools is broken by design, that's why it's not in the repositories anymore*), l'installation se fait par compilation des sources...

Pour Suse SLES 10 : https://tpeelen.wordpress.com/2010/04/27/installing-swftools-suse-10/

Pour RedHat :

```
yum install zlib-devel libjpeg-devel giflib-devel freetype-devel gcc gcc-c++ make
```

Pour Ubuntu 10.10 :

```
apt-get install build-essential libgif-dev libjpeg-dev zlib1g-dev libfreetype6-dev
```

Puis dérouler les commandes :

```
wget http://www.swftools.org/swftools-0.9.1.tar.gz
tar xzf swftools-0.9.1.tar.gz
cd swftools-0.9.1
./configure --disable-lame
make && make install
```

L'outil exécutable `pdf2swf` est accessible dans `/usr/local/bin`, le chemin d'accès pour i-parapheur est à renseigner dans `alfresco-global.properties`. Éditer le fichier :

```
vi /opt/iParapheur/tomcat/shared/classes/alfresco-global.properties
```

Dans la zone `External executable location` (vers la ligne 32), localiser le paramètre swf.exe` (généralement ligne 38) et le positionner de la façon suivante :

```
### External executable locations **
ooo.exe=/opt/iParapheur/openoffice/program/soffice.bin
ooo.enabled=true

img.root=/opt/iParapheur/common
img.dyn=${img.root}/lib
img.exe=${img.root}/bin/convert

swf.exe=/usr/local/bin/pdf2swf
```

> NB : une anomalie dans l'installeur Alfresco 3.4.c pour système GNU/Linux 32bits justifie d'installer le composant swftools de cette façon. Cette anomalie n'est pas présente sur l'installeur Alfresco pour GNU/Linux 64bits, exigé (les plate-forme 32bit ne sont pas supportées).

7.10. Certificats et autorités de certification

L'application i-Parapheur s'appuie fortement sur l'usage de certificats électroniques pour sécuriser les communications, produire des signatures ou des cachets électroniques, etc.
Ces certificats sont à acquérir auprès d'une autorité de certification présumée fiable pour l'exploitant.
La force probante des connexions et des signatures est directement liée au niveau de confiance accordé aux certificats utilisés.

OPTION : Si nécessaire et pertinent, voici un bref vademecum pour la création d'une A.C. (autorité de certification, locale et auto-signée).
Tout ce qui suit est sous la responsabilité exclusive de l'exploitant qui devra en assurer la sécurisation et l'entretien. Libriciel SCOP décline toute responsabilité sur les conséquences de l'usage licite ou frauduleux d'une telle AC.

Éditer au préalable `/usr/lib/ssl/openssl.cnf` et décommenter au besoin la ligne relative au `keyUsage` (vers 188) :

```
# This is typical in keyUsage for a client certificate.
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
```

Créer une autorité de certification auto-signée:

```
cd /etc/nginx/ssl
/usr/lib/ssl/misc/CA.pl -newca
```

Penser à modifier le script recup_nginx_crl.sh pour faire ajouter automatiquement cette A.C. dans la liste (validca) des A.C. connues du serveur Web HTTPS

Création d'un certificat serveur pour NginX :

```
cd /etc/nginx/ssl
/usr/lib/ssl/misc/CA.pl -newreq
openssl rsa -in newkey.pem -out iparapheur-serveur_key.pem
/usr/lib/ssl/misc/CA.pl -sign
cp newcert.pem iparapheur-serveur_cert.pem
```

Création de certificat client (pour tests et/ou accès Web-services) :

```
cd /opt/_install/
/usr/lib/ssl/misc/CA.pl -newreq
/usr/lib/ssl/misc/CA.pl -sign
/usr/lib/ssl/misc/CA.pl -pkcs12 'Certificat de Monsieur X'
mv newcert.p12 le-nom-que-je-veux.p12
```

7.10.1. Trucs et astuces

Transformation d'un certificat PKCS12 en fichiers PEM X509 pour NginX

```
openssl pkcs12 -in moncertificat.p12 -nocerts -nodes -out server_key.pem
openssl pkcs12 -in moncertificat.p12 -clcerts -nokeys -out server_cert.pem
```

Opération inverse: certificats de PEM vers P12

```
openssl pkcs12 -export -out certificat.p12 -inkey userkey.pem -in usercert.pem
```

Réencoder un certificat de PEM → DER :

```
openssl x509 -outform der -in moncertificat.pem -out moncertificat.der
```

Vérifier une signature `CMS/PKCS#7` ou CAdES sur PDF:

```
openssl smime -in masignature.p7s -inform PEM -binary -verify \
  -content mondocument.pdf \
  -CApath /chemin/du/validca -purpose any -out /dev/null
```

7.11. En cas de « Proxy AJP » indisponible (Apache)

Se rabattre sur `MOD_JK` : http://wiki.apache.org/tomcat/FAQ/Connectors

```
apt-get install libapache2-mod-jk
```

Activer le module jk: **Attention, Il faut** `mod_JK V.1.2.x`

```
/etc/init.d/apache2 force-reload
cp /tmp/FichiersCONF/apache/mod_jk.conf conf.d/
```

7.12. Souci de connexion Web-Services

Dans certains cas, l'interopérabilité entre application tierce et i-Parapheur peut être compliquée lors de l'établissement de session TLS/SSL : l'erreur retournée étant une exception du style `SSLHandshakeException` assorti d'un message « SSL renegociation failure ».

A l'origine, la correction d'une vulnérabilité connue (CVE-2009-3555), et décrite aux l'URLs :

http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-3555

http://java.sun.com/javase/javaseforbusiness/docs/TLSReadme.html

Une solution consiste à ajouter l'assertion suivante dans les `JAVA_OPTS` au lancement :

`-Dsun.security.ssl.allowUnsafeRenegotiation=true`

A noter également qu'un correctif distribué depuis la version *2.2.14-5ubuntu8.2* de **Apache** et la version *0.9.8k-7ubuntu8.1* de la `libssl` (distribution Ubuntu) permet de contourner également ce problème en cas d'impossibilité de patcher le client.

La `libssl` implémente (en backport de `openssl 0.9.8m`) la RFC5746[2].

Pour Apache, il est possible de forcer l'usage vers l'ancien comportement avec une nouvelle directive à inclure dans la définition du VirtualHost :
`SSLInsecureRenegotiation`

7.13. Message « Too Many Open Files »

Cela peut arriver, des exceptions dans les traces Alfresco, et le serveur d'application qui plante avec ce message : "too many open files".
Cela peut arriver alors même que l'on croit avoir résolu les limites de fichier ouvrables par processus. Le wiki Alfresco propose une réponse :

http://wiki.alfresco.com/wiki/Too_many_open_files

La réponse présentée peut s'avérer inefficace ; la ligne de commande suivante peut révéler la réelle source du problème (une seule ligne[3]):

```
for pid in `pidof java`; do echo "$(< /proc/$pid/cmdline)"; egrep
'files|Limit' /proc/$pid/limits; echo "Currently open files: $(ls -1
/proc/$pid/fd | wc -l)"; echo; done
```

Exemple de résultat :

```
              Limit Soft Limit Hard Limit Units
Max open files       1024       1024       files
```

```
Currently open files: 142
```

Ceci montre que les paramètres système ne sont pas pris en compte.

Une solution de contournement est donnée sur les forums[4] (confirmé par d'autres posts[5]), dans le script `/etc/init.d/alfresco` ajouter en début de fichier les instructions suivantes :

```
ulimit -Hn 16384
ulimit -Sn 16384
```

7.14. Problème de « locale »

(apparu sur ubuntu6.06) Ubuntu14.04 :

- éditer le fichier /etc/default/locale pour avoir les variables LANG et LANGUAGE bien définies
- exécuter la commande `dpkg-reconfigure locales`

7.15. Hôtes virtuels, HTTPS et SNI

Le but : utiliser un serveur Web avec plusieurs hôtes virtuels (virtual hosts) HTTPS sur une seule adresse IP. Le *Server Name Indication*[6] (SNI, extension du protocole TLS) permet le support de plusieurs Virtual Host avec des certificats SSL différents. En effet, il permet d'indiquer au serveur quel FQDN est contacté par le client préalablement à la négociation (handshake) de la session chiffrée.

Le problème des certificats sans SNI se présente dans le cas suivant : lorsque le client demande le certificat au serveur, il ne précise pas de nom de domaine au moment de la négociation SSL. Le serveur est ainsi incapable de savoir quel certificat envoyer en fonction du domaine. Comme un certificat est rattaché à un domaine bien précis, il était nécessaire de mettre en place un nouveau mécanisme d'échange.

Ce mécanisme implique une modification de la phase de négociation des échanges SSL et TLS. La modification est donc à réaliser côté client ET côté serveur. SNI est une extension à TLS.

Parmi les navigateurs, ceux qui supportent le SNI sont (non exhaustif) :

- Internet Explorer 7 ou + (sur Windows Vista et +, mais **pas WindowsXP même avec IE 8**)
- Mozilla Firefox 2.0 ou plus récent
- Google Chrome 6 ou supérieur (Windows, OS X 10.5.7 minimum)
- Safari 3.0 ou supérieur (Mac OS X 10.5.6 minimum)
- MobileSafari sur iOS 4.0 ou supérieur
- Android Honeycomb (3.2) ou supérieur

Au niveau des serveurs, on trouve (non exhaustif) :

- Apache 2.2.12 ou supérieur en utilisant `mod_ssl` ou `mod_gnutls`
- F5 Networks Local Traffic Manager v11.1 ou +
- Lighttpd 1.4.24 ou +, et 1.5.x
- Nginx avec le OpenSSL supportant le SNI
- Apache Tomcat 7 sur Java7 (connexion client)

Les bibliothèques (utilisables sur application client ou serveur) :

- Mozilla NSS client 3.11.1
- OpenSSL
  - 0.9.8f (sorti le 11/10/2007) - pas compilé par défaut, activé avec l'option `--enable-tlsext`
  - 0.9.8j (sorti le 07/01/2009) - compilé par défaut
- GNU TLS
- libcurl 7.18.1 si SNI activé , wget 1.14
- Python 3.2
- Oracle Java7 JSSE

NB : les plate-formes qui ne supportent pas SNI :

- Internet Explorer ou Apple Safari sur Windows XP, IE8 dans certains contextes
- Java 6 !
- Apache TOMCAT (serveur)
- Serveurs : Microsoft IIS toutes versions au moins jusque 7.5

---

1. Se référer à l'annexe pour précisions sur les possibilités de déploiement de polices de caractère ↩

2. Voir http://tools.ietf.org/html/rfc5746↩
3. Source : https://forums.alfresco.com/en/viewtopic.php?f=14&t=40374&start=0↩
4. Voir http://ubuntuforums.org/showthread.php?t=1583041↩
5. Voir http://www.jayway.com/2012/02/11/how-to-really-fix-the-too-many-open-files-problem-for-tomcat-in-ubuntu/↩
6. Voir http://en.wikipedia.org/wiki/Server_Name_Indication↩