Déploiement d'OCS 1.02 RC2 sous Debian Etch 64

Par Big_orneau

Note : Utilisation ici d'OCS sur un domU Xen. Les commandes sont en italiques.

- Avant toute chose vérifier absolument :

  La date sur le serveur (*tzconfig*, *ntpdate* à faire sur le Dom0)

  L'hostname *vim /etc/hosts*
  127.0.0.1 localhost
  19x.xxx..xxx.xxx dom0.univ-xxx.fr dom0
  19x.xxx..xxx.xxx domU.univ-xxx.fr domU

  Puis, *echo domU.univ-xxx.fr > /etc/hostname*

  Les DNS vim /etc/resolv.conf

  search xxx.univ-xxx.fr
  nameserver 19x.xxx.xxx.xxx

- Rebooter

  Hostname et *hostname –f* doivent indiquer domU.univ-xxx.fr

- Télécharger le packet OCS avec *wget*

- Installer le nécessaire :

apt-get install mysql-server apache2 php5 php5-ldap php5-imap php5-gd php5-mysql

- Installer également les lib nécessaires

apt-get install libapache2-mod-php5 libapache2-mod-perl2
apt-get install libxml-simple-perl libcompress-zlib-perl
apt-get install libdbi-perl libdbd-mysql-perl
apt-get install libnet-ip-perl libphp-pclzip make apt-get install libapache-dbi-perl
(apt-get install lipsoap-lite-perl si nécessaire)

- Dezipper le packet OCS

tar -xvzf OCSNG_LINUX_SERVER_XX.tar.gz

et exécuter *setup.sh*. Vérifiez l'installation des librairies au passage.

Attention, la structure a changée depuis la dernière version. Le répertoire /var/www est vide (alias utilisés).

Décommenter AddDefaultCharset ISO-8859-1 dans apache2.conf

Créer un utilisateur Mysql avec mot de passe :
Mysql –u root –p
Set password for root@localhost=password('xxxx');

Une fois terminé, pointer vers **http:// adresse/ocsreports/install.php** et suivre le setup (attention à la configuration du site défault).

Modification du document root

- Il est plus simple d'accéder au serveur par http://nom-serveur que http://nom-server/ocsreports Pour cela :

vim /etc/apache2/conf.d/ocsinventory-reports.conf

Modifier :

Alias /ocsreports /usr/share/ocsinventory-server/ocsreports en
DocumentRoot /usr/share/ocsinventory-server/ocsreports

Decommenter également *LimitRequestBody 4194304*

- On va ensuite nettoyer notre site default qui doit ressembler à ca :

cat /etc/apache2/sites-available/default

```
NameVirtualHost *
<VirtualHost *>
        ServerAdmin webmaster@localhost
        ServerName ocs-xen.xxx.fr
        ErrorLog /var/log/apache2/error.log
        # Possible values include: debug, info, notice, warn, error, crit,
        # alert, emerg.
        LogLevel warn
        CustomLog /var/log/apache2/ocs80.log combined
        ServerSignature Off
</VirtualHost>
```

Passage en SSL et déploiement d'un packet

Il faut un site OCS accessible en SSL pour déployer les packets.

Activer SSL : a2enmod ssl

Modifier l'écoute des ports. Ajouter à /etc/apache2/ports.conf :
Listen 443

Puis :

Apt-get install ssl-cert (ou utiliser le script fourni par OCS en annexe)

Modifier le fichier /usr/sbin/make-ssl-cert

openssl req -config $TMPFILE -new -x509 -nodes -out $output -keyo
ut $output > /dev/null 2>&1

en

openssl req -config $TMPFILE -new -x509 **-days 1825** -nodes -out $output -keyo
ut $output > /dev/null 2>&1

Puis :

make-ssl-cert /usr/share/ssl-cert/ssleay.cnf /etc/apache2/ssl/cacert.pem

Editer le fichier suivant pour inclure les options SSL :

vim /etc/apache2/conf.d/ssl

```
SSLProtocol all
SSLOptions +StdEnvVars
SSLCipherSuite
ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
SSLCertificateFile /etc/apache2/ssl/cacert.pem
SSLCertificateKeyFile /etc/apache2/ssl/cacert.pem
```

Créer une copie du site default et le modifier comme suit :

cp /etc/apache2/site-available/default /etc/apache2/site-available/ocsssl
vim /etc/apache2/site-available/ocsssl

```
NameVirtualHost *:443
<VirtualHost *:443>
        ServerAdmin webmaster@localhost
        ServerName ocs-xen.xxx.fr
        SSLEngine On
        ErrorLog /var/log/apache2/errorssl.log
        # Possible values include: debug, info, notice, warn, error, crit,
        # alert, emerg.
        LogLevel warn
        CustomLog /var/log/apache2/ocs443.log combined
        ServerSignature Off
        <Directory "/usr/lib/cgi-bin">
                SSLOptions +StdEnvVars
```

```
        </Directory>
        SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown
</VirtualHost>
```

Activer : a2ensite ocsssl

Création d'un packet (méthode Lancer)

Avant toute chose, il faut copier le cacert.pem du serveur sur le repertoire ocs du client ! Pour éviter des prises de tête possibles, **je déconseille de télécharger le certificat copié à partir d'un navigateur Web.** Utilisez plutôt un copier/coller à partir d'un partage samba ou un transfert en FTP ou une clef Usb, etc.

Ensuite créer un répertoire de la forme suivante sur votre poste ou sur le serveur :

OS-Application-Version-Action-Suffix

Ou OS = wxp, w2k….
Action = config, install, remove ,script
Suffix = a, b ,c ….

Exemple : mkdir wxp-photofiltre-621-install-a

Copier l'application à déployer dans le répértoire (ici **pf-setup631.exe)**. L'application doit obligatoirement posséder des switchs pour l'installation silencieuse !

Zipper le dossier entier (le dossier, pas l'application seule)

Allez sur Ocs puis « Télédéploiement de packet ». On va uploader le packet crée. Rentrer les infos suivantes :

Nom : wxp-photofiltre-621-install-a
Fichier : wxp-photofiltre-621-install-a .zip
Action : Lancer
Chemin : **wxp-photofiltre-631-install-a\\pf-setup631.exe /S**

Le chemin doit être de la forme nom_zip\\commande

Envoyez le fichier. Fragmentez le en plusieurs packets si nécessaire.

Il faut ensuite activer le packet dans ocs. Dans l'interface d'activation, sélectionnez le package, cliquez sur l'icône Activer puis rentrer :

Serveur https : adresse-serveur/download
Serveur de fichiers : adresse-serveur/download

Vérifier au passage que le répertoire download est accessible en https dans un navigateur !

Ensuite sélectionnez une machine dans l'interface de recherche ou le listing, et lui associer le packet voulu. (il faut cliquer sur la machine, puis sur la double flèche, puis ajouter packet). Modifier si nécessaire les options du serveur pour le déploiement. Attention en cas de déploiement important à ne pas mettre un timing trop serré. (Au passage, ne pas oublier d'activer l'option de déploiement !)

Pour vérifier si tout marche bien sans attendre, sur le client, lancer un cmd et faire ocsinventory.exe /np /debug /server :adresse

Vérifier ensuite les logs du client (download.log et machine.log)

Voilà, l'application est déployée ! Vous pouvez vérifiez le statut sur le serveur, en cliquant sur la machine dans OCS, puis sur la double flèche !

ANNEXE

```
#!/bin/sh
#
# First, generate apache server certificate request
# Generate 1024 bits RSA key, store private key in a
# no password protected PEM file server.key, using
# system default openssl configuration file.
#
echo
echo Generating Apache server private key...
echo
openssl genrsa -out server.key 1024
#
# Next, sign the apache server certificate with the apache
# server key
#
# Sign with PEM certificate server.crt, using PEM file
# server.key for server private key, using system default
# openssl configuration file.
#
# The produced certificate will be valid for 1825 days (about 5 years)
#
echo
echo Generating Apache server self signed certificate...
echo
openssl req -outform PEM -new -key server.key -x509 -days 1825 -out server.crt
```

Attention à bien renommer le server.crt en cacert.pem sur la machine cliente !