

Mode opératoire d'installation de Dokeos 1.8.2 sur Ubuntu Server

Dokeos 1.8.2 LMS
Oogie Rapid Authoring
Dokeos Live Conferencing



Sommaire

A qui est destiné ce mode opératoire ?	5
Introduction.....	5
Présentations.....	7
Parti 1 – Présentation des commandes linux utilisées	7
Parti 2 – Post-installation de notre serveur	9
Pré- installation pour « Ubuntu 6.06 LTS (Dapper) ».....	9
Pré- installation pour « Ubuntu 6.06 LTS (Dapper) » et « Ubuntu 7.04 »	9
Parti 3 – Installation d'un serveur LAMP.....	10
Installation d'Apache2.....	10
Installation de MySQL	10
Installation de PHP 5 pour Apache et MySQL.....	10
Pré-tests	10
Tester le fonctionnement de PHP	11
Installation des modules complémentaires pour PHP 5	11
Configuration de PHP.....	11
Sécuriser MySQL.....	12
Installation de phpmyadmin	12
Parti 4 – Installation d'un Dokeos.....	13
Parti 5 – Installation de JAVA et TOMCAT	13
Installation de java.....	13
Vérification de la version de java	13
Installation de Tomcat.....	14
<i>Configuration de Tomcat</i>	14
Parti 6 – Installation du serveur openoffice pour Oogie.....	14
Pré-installation.....	14
Installation de open office	15
Installation des polices Microsoft.....	15
Configuration du serveur openoffice	15
1 ^{er} lancement d' openoffice.....	15

Installation du script de démarrage	15
1 ^{er} lancement de oooserver.....	19
Configuration de Oogie.....	19
Parti 7 – Installation de la vidéoconférence	21
Installation de RED5.....	21
Installation du module RED5 Dokeos	21
Configuration de RED5.....	21
Installation de OpenLaszlo.....	22
Configuration de OpenLaszlo	22
Installation de « mod_jk » pour Apache2.....	22
Configuration du module « mod_jk ».....	23
Configuration de la vidéoconférence.....	23

A qui est destiné ce mode opératoire ?

Ce mode opératoire est destiné à toute personne souhaitant, installer une application web d'e-learning.

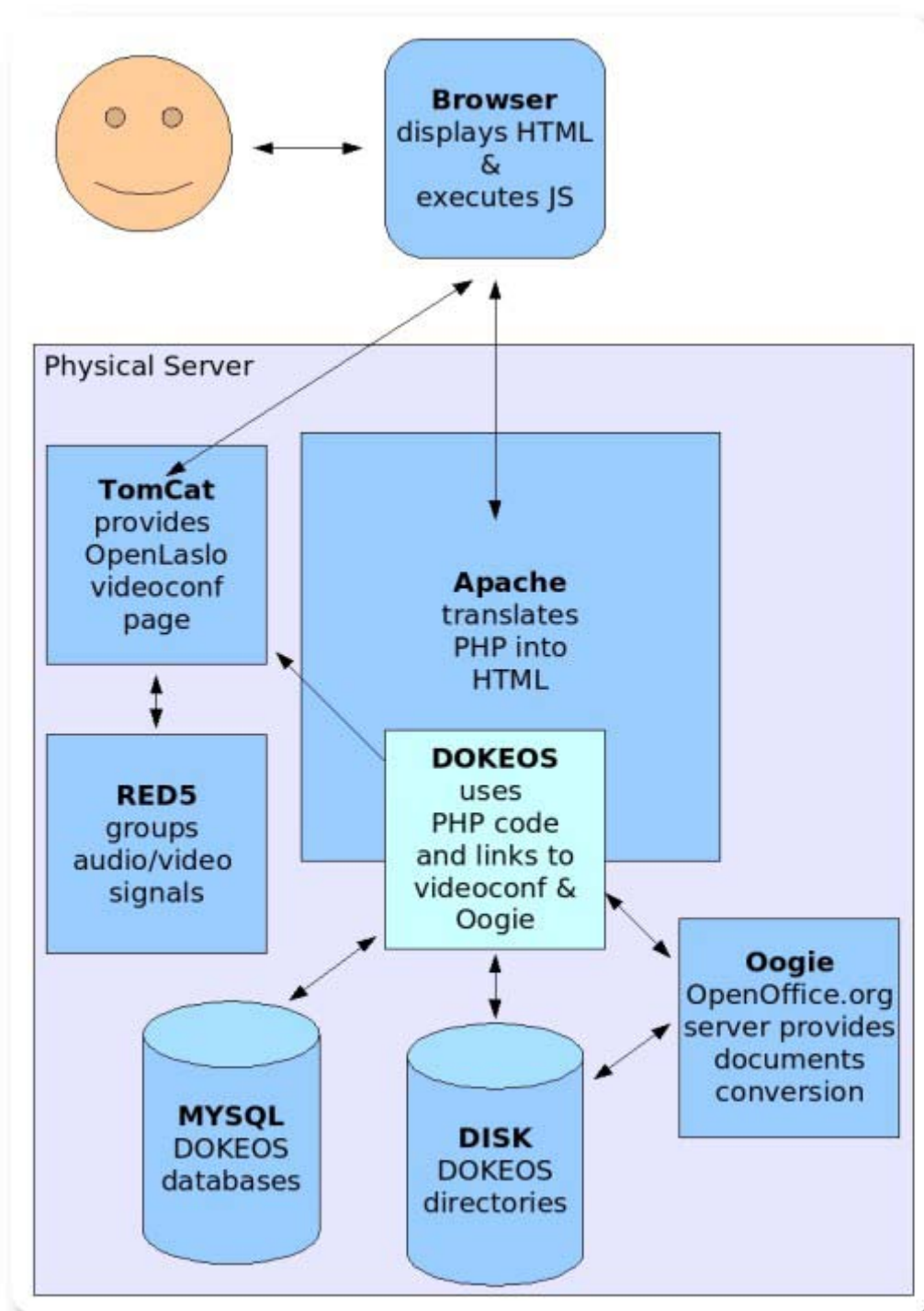
Vous devez toutefois posséder une connaissance minimale du Web et des commandes linux.

Contenu de ce mode opératoire.

Introduction

Ce mode opératoire, décrit l'installation de A à Z d'une solution d'e-learning.





Présentations

Dokeos est une plate-forme d'apprentissage en ligne (ou LMS) sous licence GPL, ce logiciel se veut international, en effet son architecture est multilingue et il supporte 34 langues. Le logiciel est écrit en PHP utilise le SGBDR MySQL.

C'est une plate-forme d'apprentissage qui permet de scénariser et de diffuser des cours en ligne pouvant intégrer :

- des documents à télécharger,
- des liens hypertextes,
- des outils de communication (chat, forum, vidéoconférence...)

Parti 1 – Présentation des commandes linux utilisées

Se déplacer dans les répertoires

```
root@ubuntu :~# cd /etc
```

Se déplace dans le répertoire etc.

Copier un fichier (ou un répertoire) : cp.

cp [option] fichier-origine fichier-destination

Exemple pour faire une copie de notre fichier «fichier-linux» en un fichier «fichier-linux2», il suffit de faire :

```
root@ubuntu :~# cp fichier-linux fichier-linux2
```

Lister les fichiers d'un répertoire (ls)

La commande ls et ses très nombreuses options vous permettront d'obtenir beaucoup d'informations sur les fichiers présents dans un répertoire : déplaçons nous par exemple dans le répertoire "/bin" et listons le contenu de ce répertoire :

```
root@ubuntu :~# cd /etc  
root@ubuntu :~# ls
```

Déplacer ou renommer un fichier (mv)

Pour comprendre la commande mv, voyons une suite de commandes qui effectuent des opérations différentes :

```
root@ubuntu :~# mv fichier-linux linux
```

Renomme le fichier « fichier-linux » en « linux ».

```
root@ubuntu :~# mv linux linux
```

Ecrase le fichier existant avec la source.

```
root@ubuntu :~# mv document mes-documents
```

Renomme le répertoire « document » en « mes-documents ».

Supprimer un fichier "rm".

La syntaxe de la commande rm est la suivante :

rm fichier-supprimer

Exemple pour supprimer le fichier " fichier-linux2" :

```
root@ubuntu :~# rm fichier-linux2
```

Archivage de données : la commande "tar"

La commande tar permet d'archiver ou de désarchiver des répertoires et des fichiers de façon optimale.

Une des commandes dont vous aurez certainement le plus besoin est :

```
root@ubuntu :~# tar xzf nom_du_fichier.tar.gz
```

Arrêter le système : la commande shutdown

```
root@ubuntu :~# shutdown -r now
```

Cette commande vous permet de rebooter l'ordinateur.

```
root@ubuntu :~# shutdown -h now
```

Cette commande vous permet d'arrêter complètement le système. Vous pouvez éteindre l'ordinateur lorsque vous verrez affiché :

"System halted

The system is halted".

Attribuer les droits à un fichier (répertoire) : chmod

La commande chmod attribut les droits aux fichiers ou répertoires.

```
root@ubuntu :~# chmod 750 /votre/repertoire
```

```
root@ubuntu :~# chmod -R 750 /votre/repertoire
```

Attribution récursive.

Placer les propriétés (chmod)

Introduction : linux permet de spécifier les droits qu'ont les utilisateurs sur un fichier.

Pour voir ces droits, il suffit d'utiliser la commande ls -l :

```
root@ubuntu :~# ls -l perso
```


Parti 2 – Post-installation de notre serveur



Pré- installation pour « Ubuntu 6.06 LTS (Dapper) ».

Récupérer la liste des dépôts pour votre version d'ubuntu à l'adresse suivante :

<http://doc.ubuntu-fr.org/depots>

Dans notre exemple nous allons utiliser les dépôts universe et multiverse « Ubuntu 6.06 LTS (Dapper) ».

```
jean@ubuntu : sudo nano /etc/apt/sources.list
```

A la fin du fichier taper ou coller

Code:

```
## SERVEURS FRANÇAIS
deb http://fr.archive.ubuntu.com/ubuntu/ dapper main restricted
universe multiverse
deb http://fr.archive.ubuntu.com/ubuntu/ dapper-security main
restricted universe multiverse
deb http://fr.archive.ubuntu.com/ubuntu/ dapper-updates main
restricted universe multiverse
```

Note: CTRL+o pour enregistrer et CTRL+x pour quitter.

Pré- installation pour « Ubuntu 6.06 LTS (Dapper) » et « Ubuntu 7.04 ».

Mise à jour de votre serveur :

```
jean@ubuntu : sudo apt-get update
jean@ubuntu : sudo apt-get upgrade
jean@ubuntu : sudo shutdown -r now
```

Note: pour utiliser SSH faire

```
jean@ubuntu : sudo apt-get ssh
```

Parti 3 – Installation d'un serveur LAMP

Installation d'Apache2



```
jean@ubuntu : sudo apt-get install apache2-mpm-prefork
```

Installation de MySQL



```
jean@ubuntu : sudo apt-get install mysql-server-5.0
```

Installation de PHP 5 pour Apache et MySQL



```
jean@ubuntu : sudo apt-get install libapache2-mod-php5 php5-mysql
```

Pré-tests

Recharger la configuration d'Apache


```
jean@ubuntu : sudo /etc/init.d/apache2 reload
```

Vous pouvez tester votre serveur dans votre navigateur à l'adresse suivante :

<http://192.168.0.60/>

Où 192.168.0.60 est l'adresse IP de votre serveur

Index of /

Name	Last modified	Size	Description
 apache2-default/	26-Jul-2006 20:50	-	

Apache/2.0.55 (Ubuntu) PHP/5.1.2 Server at localhost Port 80

Tester le fonctionnement de PHP

```
jean@ubuntu : sudo nano /var/www/phpinfo.php
```

Editer le texte suivant et enregistrer :

```
<?php phpinfo() ; ?>
```

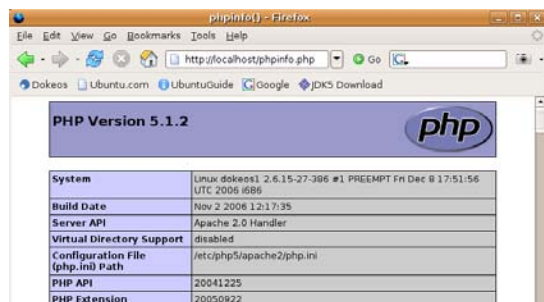
Attribution des droits sur le fichier :

```
jean@ubuntu : sudo chmod 755 /var/www/phpinfo.php
```

Vous pouvez tester votre serveur dans votre navigateur à l'adresse suivante :

<http://192.168.0.60/phpinfo.php>

Où 192.168.0.60 est l'adresse IP de votre serveur



Installation des modules complémentaires pour PHP 5

```
jean@ubuntu : sudo apt-get install php-pear php5-gd php5-xsl php5-mcrypt php5-ldap
```

Configuration de PHP

```
jean@ubuntu : sudo nano /etc/php5/apache2/php.ini
```

Rechercher et remplacer les variables par les valeurs suivantes :

```
max_execution_time = 300 ; Maximum execution time of each script, in seconds
```

```
max_input_time = 600 ; Maximum amount of time each script may spend
```

```
parsing request data
```

```
memory_limit = 32M ; Maximum amount of memory a script may  
consume (8MB)
```

```
post_max_size = 16M ; Maximum size of a POST request
```

```
upload_max_filesize = 100M ; Maximum upload file size
```

Sécuriser MySQL

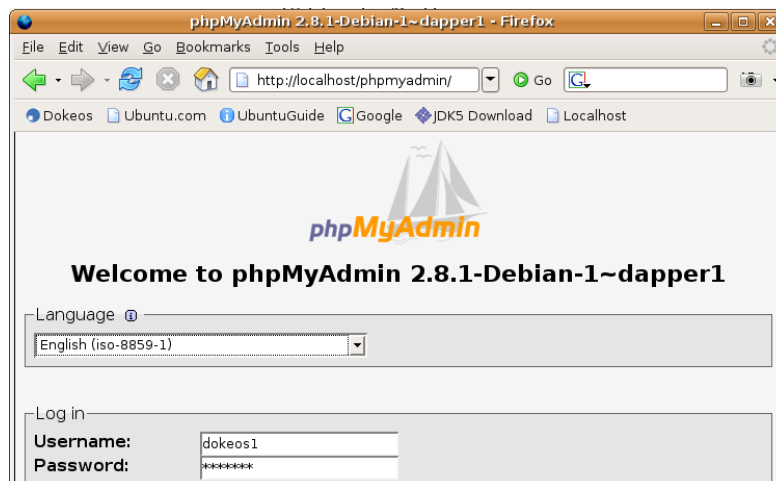
```
jean@ubuntu : sudo mysql_secure_installation
```

Installation de phpmyadmin

```
jean@ubuntu : sudo apt-get install phpmyadmin
```

Vous pouvez tester phpmyadmin à l'adresse suivante :

<http://192.168.0.60/phpmyadmin/>



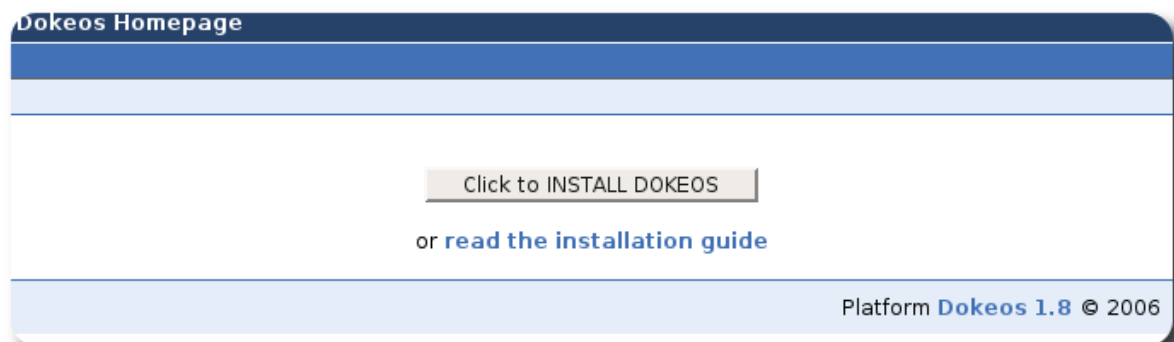
Parti 4 – Installation d'un Dokeos

Téléchargement et décompression de l'archive Dokeos.

```
jean@ubuntu : cd /var/www
jean@ubuntu : sudo wget http://www.dokeos.com/download/dokeos-1.8.2.zip
jean@ubuntu : sudo apt-get install unzip
jean@ubuntu : sudo unzip dokeos-1.8.2.zip
jean@ubuntu : sudo chmod -R 0777 dokeos/
```

Installation de Dokeos à partir de votre navigateur à l'adresse suivante :

<http://192.168.0.60/dokeos/>



Attribution des droits.

```
jean@ubuntu : sudo chmod 444
/var/www/dokeos/main/inc/conf/configuration.php
jean@ubuntu : sudo chmod 444 /var/www/dokeos/main/install/index.php
```

Parti 5 – Installation de JAVA et TOMCAT

Installation de java

```
jean@ubuntu : sudo apt-get install sun-java5-jdk
```

Vérification de la version de java

```
jean@ubuntu : sudo java-version
```

Note : Si la version est différente de java-1.5

```
jean@ubuntu : sudo update-alternatives --config java
```

Choisir la version java-1.5.0 en tapant « 1 ».

```
Sélection Alternative
-----
1 /usr/lib/jvm/java-1.5.0-sun/jre/bin/java
2 /usr/bin/gij-wrapper-4.1
*+ 3 /usr/lib/jvm/java-gcj/jre/bin/java
```

Installation de Tomcat

```
jean@ubuntu : sudo apt-get install tomcat5 tomcat5-admin tomcat5-
webapps
```

Configuration de Tomcat

```
jean@ubuntu : sudo nano /etc/default/tomcat5
```

Editer à la fin du document le texte suivant :

```
JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun
```

```
jean@ubuntu : sudo export JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun
jean@ubuntu : sudo /etc/init.d/tomcat5 restart
```

Parti 6 – Installation du serveur openoffice pour Oogie

Pré-installation

```
jean@ubuntu : sudo apt-get install xvfb
jean@ubuntu : sudo apt-get install xbase-clients
```

Installation de open office

```
jean@ubuntu : sudo apt-get install openoffice.org
```

Installation des polices Microsoft

```
jean@ubuntu : sudo apt-get install msttcorefonts
```

Configuration du serveur openoffice

```
jean@ubuntu : sudo apt-get nano  
/usr/lib/openoffice/share/registry/data/org/openoffice/Setup.xcu
```

Ajouter le texte suivant :

```
<prop oor:name="ooSetupConnectionURL">  
<value>  
socket,host=localhost,port=2002;urp;StarOffice.ServiceManag  
er  
</value>  
</prop>
```

A coté du texte suivant :

```
<prop oor:name="ooSetupInstCompleted">  
<value>>false</value>  
</prop>
```

1^{er} lancement d' openoffice

```
jean@ubuntu : sudo xvfb-run --server-args='-screen 0 800x600x16' -a  
/usr/lib/openoffice/program/soffice -headless -nologo -norestore  
jean@ubuntu : sudo ooffice -headless -display :99
```

Installation du script de démarrage

```
jean@ubuntu : sudo nano /etc/init.d/oooserver
```

Editer le texte suivant :

Penser à modifier le nom de l'utilisateur ex : USER=jean

```
#!/bin/sh
### BEGIN INIT INFO
# Provides:          oooserver
# Required-Start:    $local_fs $remote_fs
# Required-Stop:     $local_fs $remote_fs
# Default-Start:     2 3 4 5
# Default-Stop:      S 0 1 6
# Short-Description: Initscript to start OOo as server
# Description:       This file should be used to construct scripts to be
#                   placed in /etc/init.d.
### END INIT INFO

# Author: Jerome Warnier <jwarnier@beeznest.net>
#
# Do NOT "set -e"

# PATH should only include /usr/* if it runs after the mountnfs.sh script
PATH=/usr/sbin:/usr/bin:/sbin:/bin
DESC="OpenOffice.org server"
NAME=oooserver
DAEMON=/usr/bin/xvfb-run
#DAEMON_ARGS="--error-file=/tmp/xvfb-run.log -a
/usr/lib/openoffice/program/soffice -headless -nologo -norestore"
DAEMON_ARGS="-a /usr/lib/openoffice/program/soffice -headless -nologo -
norestore -
accept=socket,host=localhost,port=2002;urp;StartOffice.ServiceManager"
PIDFILE=/var/run/$NAME.pid
SCRIPTNAME=/etc/init.d/$NAME
USER=dlc

# Exit if the package is not installed
[ -x "$DAEMON" ] || exit 0

# Read configuration variable file if it is present
[ -r /etc/default/$NAME ] && . /etc/default/$NAME

# Load the VERBOSE setting and other rcS variables
[ -f /etc/default/rcS ] && . /etc/default/rcS

# Define LSB log_* functions.
# Depend on lsb-base (>= 3.0-6) to ensure that this file is present.
. /lib/lsb/init-functions

#
# Function that starts the daemon/service
#
do_start()
{
    # Return
    # 0 if daemon has been started
    # 1 if daemon was already running
    # 2 if daemon could not be started
    start-stop-daemon --start --quiet --pidfile $PIDFILE --chuid $USER -
-background --exec $DAEMON -- \
        --server-args='-screen 0 800x600x16' $DAEMON_ARGS
    # Add code here, if necessary, that waits for the process to be
    ready
    # to handle requests from services started subsequently which depend
```



```

        # on this one.  As a last resort, sleep for some time.
    }

#
# Function that stops the daemon/service
#
do_stop()
{
    # Return
    # 0 if daemon has been stopped
    # 1 if daemon was already stopped
    # 2 if daemon could not be stopped
    # other if a failure occurred
    start-stop-daemon --stop --quiet --retry=TERM/30/KILL/5 --pidfile
$PIDFILE --name $NAME
    RETVAL="$?"
    [ "$RETVAL" = 2 ] && return 2
    # Wait for children to finish too if this is a daemon that forks
    # and if the daemon is only ever run from this initscript.
    # If the above conditions are not satisfied then add some other code
    # that waits for the process to drop all resources that could be
    # needed by services started subsequently.  A last resort is to
    # sleep for some time.
    start-stop-daemon --stop --quiet --oknodo --retry=0/30/KILL/5 --exec
$DAEMON
    [ "$?" = 2 ] && return 2
    # Many daemons don't delete their pidfiles when they exit.
    rm -f $PIDFILE
    return "$RETVAL"
}

#
# Function that sends a SIGHUP to the daemon/service
#
do_reload() {
    #
    # If the daemon can reload its configuration without
    # restarting (for example, when it is sent a SIGHUP),
    # then implement that here.
    #
    start-stop-daemon --stop --signal 1 --quiet --pidfile $PIDFILE --
name $NAME
    return 0
}

#
# Function that checks status of the daemon/service
#
do_status()
{
    # Return
    # 0 One or more processes matched the criteria.
    # 1 No processes matched.
    # 2 Syntax error in the command line.
    # 3 Fatal error: out of memory etc.
    pgrep -u $USER soffice.bin > /dev/null
    RETVAL="$?"
    return "$RETVAL"
}

```

```

case "$1" in
start)
    [ "$VERBOSE" != no ] && log_daemon_msg "Starting $DESC" "$NAME"
    do_start
    case "$?" in
        0|1) [ "$VERBOSE" != no ] && log_end_msg 0 ;;
        2) [ "$VERBOSE" != no ] && log_end_msg 1 ;;
    esac
    ;;
stop)
    [ "$VERBOSE" != no ] && log_daemon_msg "Stopping $DESC" "$NAME"
    do_stop
    case "$?" in
        0|1) [ "$VERBOSE" != no ] && log_end_msg 0 ;;
        2) [ "$VERBOSE" != no ] && log_end_msg 1 ;;
    esac
    ;;
#reload|force-reload)
    #
    # If do_reload() is not implemented then leave this commented out
    # and leave 'force-reload' as an alias for 'restart'.
    #
    #log_daemon_msg "Reloading $DESC" "$NAME"
    #do_reload
    #log_end_msg $?
    #;;
restart|force-reload)
    #
    # If the "reload" option is implemented then remove the
    # 'force-reload' alias
    #
    log_daemon_msg "Restarting $DESC" "$NAME"
    do_stop
    case "$?" in
        0|1)
            do_start
            case "$?" in
                0) log_end_msg 0 ;;
                1) log_end_msg 1 ;; # Old process is still running
                *) log_end_msg 1 ;; # Failed to start
            esac
            ;;
        *)
            # Failed to stop
            log_end_msg 1
            ;;
    esac
    ;;
status)
    #log_daemon_msg "Checking status of $DESC" "$NAME"
    [ "$VERBOSE" != no ] && log_daemon_msg "Checking status of $DESC"
"$NAME"
    do_status
    log_end_msg "$?"
    echo $?
    exit $?
    ;;

```

```
* )
    #echo "Usage: $SCRIPTNAME {start|stop|restart|reload|force-reload}"
>&2
    echo "Usage: $SCRIPTNAME {start|stop|restart|force-reload|status}"
>&2
    exit 3
    ;;
esac
:
```

```
jean@ubuntu : sudo chmod +x /etc/init.d/oooserver
jean@ubuntu : sudo ln -s /etc/init.d/oooserver
/etc/rc2.d/S51oooserver
```

1^{er} lancement de oooserver

```
jean@ubuntu : sudo /etc/init.d/oooserver start
```

Vérifier que oooserver est opérationnel

```
jean@ubuntu : sudo /etc/init.d/oooserver status
```

Vous devez obtenir : **...done.**

Configuration de Oogie

Hôte : **localhost**

Nom d'utilisateur :

Mot de passe FTP :

Chemin vers les fichiers LZX : **/lpslatest/dokeos/videoconference/ppt2lpreorder.lzx**



Basic overview Display

New chapter

New step

scenarioen

- Sample test
- slide01
- slide02
- slide03
- slide04
- slide05

Manipulate the current document


Edit Move Delete

Audiorecorder

Loudness

A Comment does exist. Listen or record new one.

Writing an e-course scenario



Parti 7 – Installation de la vidéoconférence

Installation de RED5

```
jean@ubuntu : cd /urs/src
jean@ubuntu : sudo wget
http://dl.fancycode.com/red5/debian/0.6.2/red5_0.6.2-1_all.deb
jean@ubuntu : sudo dpkg -i red5_0.6.2-1_all.deb
```

Installation du module RED5 Dokeos

```
jean@ubuntu : sudo wget http://www.dokeos.com/download/dokeos-red5APP-1\_8.tar.gz
jean@ubuntu : sudo tar zfvx dokeos-red5APP-1_8.tar.gz
jean@ubuntu : sudo cp -R dokeos-red5APP-1.8/* /usr/lib/red5/webapps
```

Configuration de RED5

```
jean@ubuntu : sudo nano /usr/lib/red5/conf/red5.properties
```

Modifier ou non les ports et les adresses IP en fonction de votre configuration

```
jean@ubuntu : sudo nano /usr/lib/red5/webapps/dokeosrecorder/WEB-INF/red5-web.properties
```

Modifier « **webapp.virtualHosts=*,localhost, 127.0.0.1** » par
« **webapp.virtualHosts=192.168.0.60,localhost, 127.0.0.1** » ainsi seul votre serveur
pourra se connecter à RED5 par mesure de sécurité.

```
jean@ubuntu : sudo nano /usr/lib/red5/conf/realm.properties
```

Modifier le mot de passe du compte administrateur « **admin: admin,admin** » par
exemple « **admin: F56xWry,admin** ».

Installation de OpenLaszlo

```
jean@ubuntu : sudo cd /home/jean
jean@ubuntu : sudo wget http://www.dokeos.com/download/dokeos-openlaszlo-1\_8.tar.gz
jean@ubuntu : sudo tar -xzf dokeos-openlaszlo-1_8.tar.gz
```

Configuration de OpenLaszlo

```
jean@ubuntu : sudo nano /home/jean/dokeos-openlaszlo-1.8/Server/tomcat-5.0.24/bin/startup.sh
```

Modifier le « `export JAVA_HOME=...` » par « `export JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun` »

```
jean@ubuntu : sudo nano /home/jean/dokeos-openlaszlo-1.8/Server/tomcat-5.0.24/bin/shutdown.sh
```

Modifier le « `export JAVA_HOME=...` » par « `export JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun` »

```
jean@ubuntu : sudo nano /home/jean/dokeos-openlaszlo-1.8/Server/lps-latest/dokeos/videoconference/host.lzx
```

Modifier « `value="your-hostname"` » par « `value="192.168.0.60"` » où 192.168.0.60 est l'IP de votre serveur RED5.

Modifier « `name="rmptTunnelport" value="80"` » par « `name="rmptTunnelport" value="8088"` » où 8088 est le port de votre serveur RED5.

Relancer votre serveur Tomcat :

```
jean@ubuntu : sudo sh /home/jean/dokeos-openlaszlo-1.8/Server/tomcat-5.0.24/bin/shutdown.sh
jean@ubuntu : sudo sh /home/jean/dokeos-openlaszlo-1.8/Server/tomcat-5.0.24/bin/startup.sh
```

Installation de « `mod_jk` » pour Apache2

```
jean@ubuntu : sudo apt-get install libapache2-mod-jk
jean@ubuntu : sudo nano /etc/libapache2-mod-jk/workers.properties
```

Modifier « **workers.java_home=...** » par « **workers.java_home=/usr/lib/j2sdk1.5-sun** ».
Modifier « **workers.tomcat_home=...** » par «
workers.tomcat_home=/home/jean/dokeos-openlaszlo-1.8/Server/tomcat-5.0.24 ».

```
jean@ubuntu : sudo nano /etc/apache2/mods-available/jk.load
```

Editer à la fin du document le texte suivant :

```
JkLogFile /var/log/apache2/mod_jk.log
JkLogLevel info
JkLogStampFormat "[%a %b %d %H:%M:%S %Y]"
JkMount /lps-latest ajp13
JkMount /lps-latest/dokeos/* ajp13
```

Configuration du module « mod_jk »

```
jean@ubuntu : sudo cp /etc/apache2/mods-enabled/jk.load
/etc/apache2/mods-available/jk.load.original
jean@ubuntu : sudo rm /etc/apache2/mods-enabled/jk.load
jean@ubuntu : sudo ln -s /etc/apache2/mods-available/jk.load
/etc/apache2/mods-enabled/jk.load
jean@ubuntu : sudo /etc/init.d/apache2 reload
```

Configuration de la vidéoconférence

Dans votre navigateur à l'adresse suivante :

<http://192.168.0.60/dokeos/>

Dans le panel d'administration

administration > Configurer les services > Vidéo-conférence

Chemin vers la visioconférence : **/lps-**

latest/dokeos/videoconference/videoconference.html

Chemin vers la visioconférence de classe : **/lps-**

latest/dokeos/videoconference/videoconference2.html

Hôte pour la vidéoconférence : **192.168.0.60**

Le protocole de la vidéoconférence fonctionne en mode web (faux recommandé) :
5080

Port du protocole RTMPT pour la vidéoconférence : **1935**

Port tunnel du protocole RTMPT pour la vidéoconférence : **8088**

Remarque :

Dans le cas où votre installation n'est pas à la racine de votre site, vous devez éditer le document suivant :

```
jean@ubuntu : sudo nano /home/jean//dokeos-openlaszlo-1.8/Server/lps-  
latest/dokeos/videoconference/conference/global/functions.lzx
```

à la ligne 73 on a

```
dataset name="getSessionVarsData" request="false" proxied="false"  
  querytype="POST" src="{  
'http://' + canvas.rmpthost + '/main/webrooms/checksession.php' }"  
  type="http"
```

Vous devez corriger « **/main/webrooms/checksession.php** » avec le chemin d'installation complet de Dokeos par rapport à la racine du serveur web, dans notre cas il a été remplacé par « **/dokeos/main/webrooms/checksession.php** »