

Installation

1.1 LISTE DES PRE-REQUIS

1.1.1 Paquets divers

tofroDOS mailx lsb-release

1.1.2 Compilateurs :

build-essential

1.1.3 Serveur Web et php5

apache2 php5 php5-mysql php-pear php5-ldap php5-snmp php5-gd

1.1.4 Serveur Mysql

mysql-server-5.0 libmysqlclient15-dev

1.1.5 RRDtool

rrdtool librrds-perl

1.1.6 Perl

libconfig-inifiles-perl

1.1.7 Snmp

snmp snmpd libnet-snmp-perl libsnmp-perl

1.1.8 Librairie gd

libgd2-xpm libgd2-xpm-dev libpng-dev libgd-gd2-perl

1.1.9 Librairie Digest

libcrypt-des-perl libdigest-hmac-perl libdigest-sha1-perl

1.1.10 Mail

postfix

1.2 CONFIGURATION DU SYSTEME

1.2.1 Création de l'utilisateur NAGIOS

```
$> useradd -m nagios -d /home/nagios  
$> passwd nagios
```

1.2.2 Création du répertoire de l'utilisateur NAGIOS

```
$>mkdir /home/nagios (optionnel selon les distrib)
```

1.2.3 Attribution des droits sur son repertoire

```
$>chown nagios.nagios /home/nagios
```

1.2.4 Création du groupe des commandes externes

```
$>groupadd nagcmd  
$>usermod -G nagcmd,nagios nagios
```

1.2.5 Création du groupe

```
usermod -G nagios,nagcmd www-data
```

1.3 INSTALLATION DE NAGIOS 3

```
# cd /usr/local/src/  
# wget http://freefr.dl.sourceforge.net/sourceforge/nagios/nagios-3.0.6.tar.gz  
# tar -xvzf nagios-3.0.6.tar.gz  
#cd nagios-3.0  
# ./configure --prefix=/usr/local/nagios --with-nagios-user=nagios --with-nagios-group=nagios --with-command-  
group=nagcmd --enable-nanosleep  
#make all  
#make install  
#make install-init  
#make install-commandmode  
#make install-config  
# update-rc.d nagios defaults
```

1.4 INSTALLATION DES PLUGINS

Comme pour nagios, les plugins sont disponibles sur le site <http://sourceforge.net/projects/nagiosplug/>

```
# cd /usr/local/src/  
# wget http://freefr.dl.sourceforge.net/sourceforge/nagiosplug/nagios-plugins-1.4.13.tar.gz  
# tar -xvzf nagios-plugins-1.4.13.tar.gz  
# cd nagios-plugins-1.4.13  
#./configure --prefix=/usr/local/nagios --with-nagios-user=nagios --with-nagios-group=nagios  
#make  
#make install
```

1.5 INSTALLATION DE NDOUTILS

```
#cd /usr/local/src  
#apt-get install subversion  
#svn checkout http://svn.centreon.com/trunk/Patch-NDOutils/  
# cd Patch-NDOutils  
#sh configure --prefix=/usr/local/nagios/ --enable-mysql --disable-pgsql --with-mysql-lib=/usr/lib/mysql/  
#make  
#cp /usr/local/src/Patch-NDOutils/src/ndomod-3x.o /usr/local/nagios/bin/ndomod.o  
#cp /usr/local/src/Patch-NDOutils/src/ndo2db-3x /usr/local/nagios/bin/ndo2db  
#chown nagios.nagios /usr/local/nagios/bin/ndomod.o
```

1.6 INSTALLATION DE CENTREON2

La dernière version du programme d'installation de Centreon 2 est disponible sur le site officiel du projet à l'adresse suivante : <http://www.centreon.com/Centreon/download.html>

```
#Configuration du path pour faciliter l'installation  
#export PATH="$PATH:/usr/local/nagios/bin"  
#cd /usr/local/src/  
#wget http://download.centreon.com/index.php?id=105  
# tar -xvzf centreon-2.0.2.tar.gz  
#cd centreon-2.0.2  
#./install -i  
You will now read Centreon Licence.  
Press enter to continue.  
Do you accept GPL license ?  
[y/n], default to [n]:  
> y
```

1.7 FINALISATION DE L'INSTALLATION DE NDOUTILS

```
#mysql -u root
mysql> CREATE DATABASE `ndo` DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
mysql>exit
```

Creation de la structure

```
#mysql -u root ndo < /usr/local/src/centreon-2.0.2/www/install/createNDODB.sql
```

Mise a jour de l'utilisateur Centreon

```
#mysql -u root
mysql>GRANT SELECT , INSERT , UPDATE , DELETE ON `ndo` . * TO 'centreon'@'localhost';
```

```
# vi /etc/init.d/ndo2db
```

```
#!/bin/sh
```

```
#
```

```
#
```

```
# chkconfig: 345 99 01
```

```
# description: Nagios to mysql
```

```
#
```

```
# Author : Gaëtan Lucas
```

```
# Realse : 07/02/08
```

```
# Version : 0.1 b
```

```
# File : ndo2db
```

```
# Description: Starts and stops the Ndo2db daemon
```

```
# used to provide network services status in a database.
```

```
#
```

```
status_ndo ()
```

```
{
```

```
    if ps -p $NdoPID > /dev/null 2>&1; then
```

```
        return 0
```

```
    else
```

```
        return 1
```

```
    fi
```

```
    return 1
```

```
}
```

```
printstatus_ndo()
```

```
{
```

```
    if status_ndo $1 $2; then
```

```
        echo "ndo (pid $NdoPID) is running..."
```

```
    else
```

```
        echo "ndo is not running"
```

```
    fi
```

```
}
```

```
killproc_ndo ()
```

```
{
```

```
    echo "kill $2 $NdoPID"
```

```
    kill $2 $NdoPID
```

```
}
```

```
pid_ndo ()
```

```
{
```

```
    if test ! -f $NdoRunFile; then
```

```
    echo "No lock file found in $NdoRunFile"
    echo -n "        checking runing process..."
    NdoPID=`ps h -C ndo2db -o pid`
    if [ -z "$NdoPID" ]; then
        echo "        No ndo2db process found"
        exit 1
    else
        echo "        found process pid: $NdoPID"
        echo -n "        reinit $NdoRunFile ..."
        touch $NdoRunFile
        chown $NdoUser:$NdoGroup $NdoRunFile
        echo "$NdoPID" > $NdoRunFile
        echo "        done"
    fi
fi

NdoPID=`head $NdoRunFile`
}

# Source function library
# Solaris doesn't have an rc.d directory, so do a test first
if [ -f /etc/rc.d/init.d/functions ]; then
    . /etc/rc.d/init.d/functions
elif [ -f /etc/init.d/functions ]; then
    . /etc/init.d/functions
fi

prefix=/usr/local/nagios
exec_prefix=${prefix}
NdoBin=${exec_prefix}/bin/ndo2db
NdoCfgFile=${prefix}/etc/ndo2db.cfg
NdoRunFile=${prefix}/var/ndo2db.run
NdoLockDir=/var/lock/subsys
NdoLockFile=ndo2db.lock
NdoUser=nagios
NdoGroup=nagios

# Check that ndo exists.
if [ ! -f $NdoBin ]; then
    echo "Executable file $NdoBin not found.  Exiting."
    exit 1
fi

# Check that ndo.cfg exists.
if [ ! -f $NdoCfgFile ]; then
    echo "Configuration file $NdoCfgFile not found.  Exiting."
    exit 1
fi

# See how we were called.
case "$1" in

    start)
        echo -n "Starting ndo:"
        touch $NdoRunFile
        chown $NdoUser:$NdoGroup $NdoRunFile
        $NdoBin -c $NdoCfgFile
        if [ -d $NdoLockDir ]; then
            touch $NdoLockDir/$NdoLockFile;
        fi
        ps h -C ndo2db -o pid > $NdoRunFile
```

```
        if [ $? -eq 0 ]; then
            echo " done."
            exit 0
        else
            echo " failed."
            $0 stop
            exit 1
        fi
    ;;

stop)
    echo -n "Stopping ndo: "

    pid_ndo
    killproc_ndo

    # now we have to wait for ndo to exit and remove its
    # own NdoRunFile, otherwise a following "start" could
    # happen, and then the exiting ndo will remove the
    # new NdoRunFile, allowing multiple ndo daemons
    # to (sooner or later) run
    #echo -n 'Waiting for ndo to exit .'
    for i in 1 2 3 4 5 6 7 8 9 10 ; do
        if status_ndo > /dev/null; then
            echo -n '.'
            sleep 1
        else
            break
        fi
    done
    if status_ndo > /dev/null; then
        echo
        echo 'Warning - ndo did not exit in a timely manner'
    else
        echo 'done.'
    fi

    rm -f $NdoRunFile $NdoLockDir/$NdoLockFile
    ;;

status)
    pid_ndo
    printstatus_ndo
    ;;

restart)
    $0 stop
    $0 start
    ;;

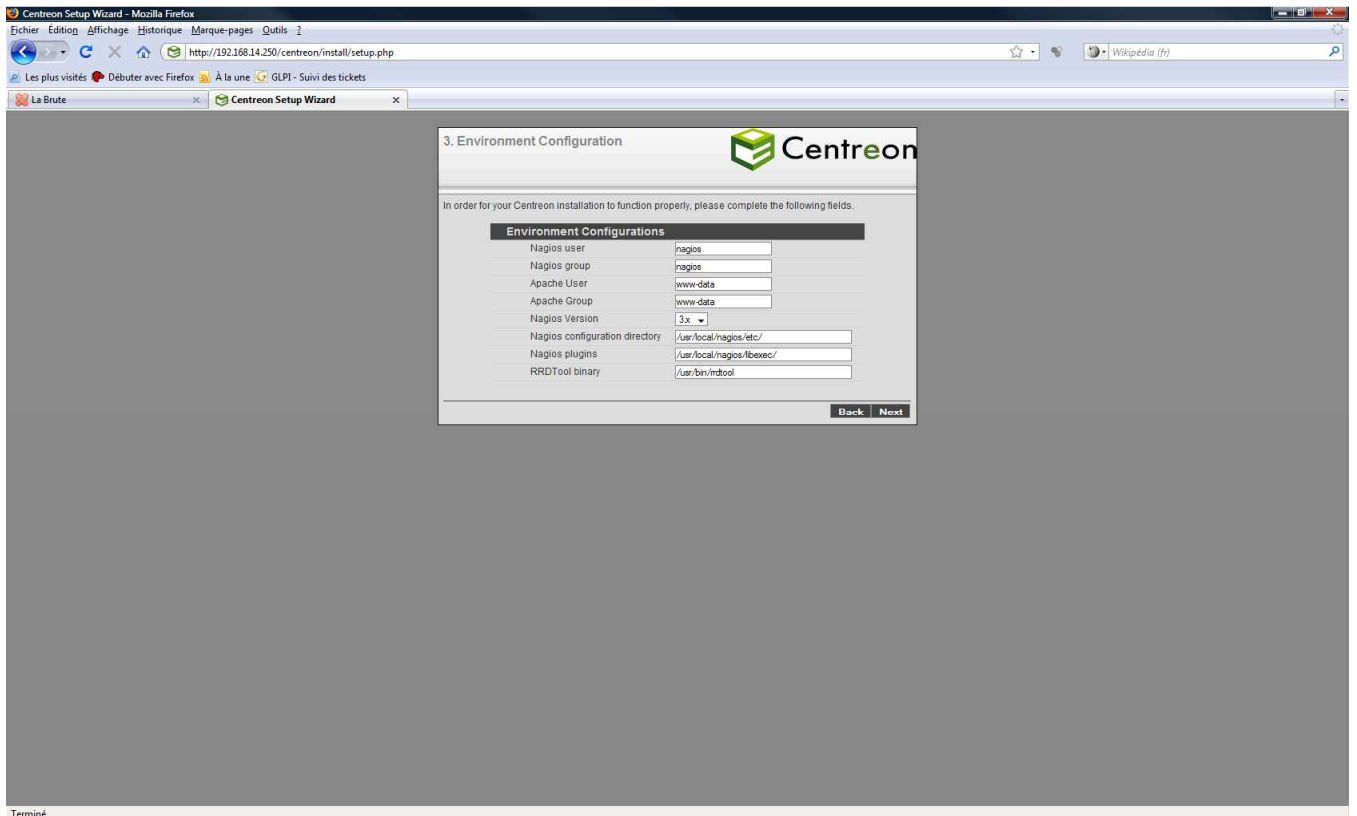
*)
    echo "Usage: ndo {start|stop|restart|status}"
    exit 1
    ;;

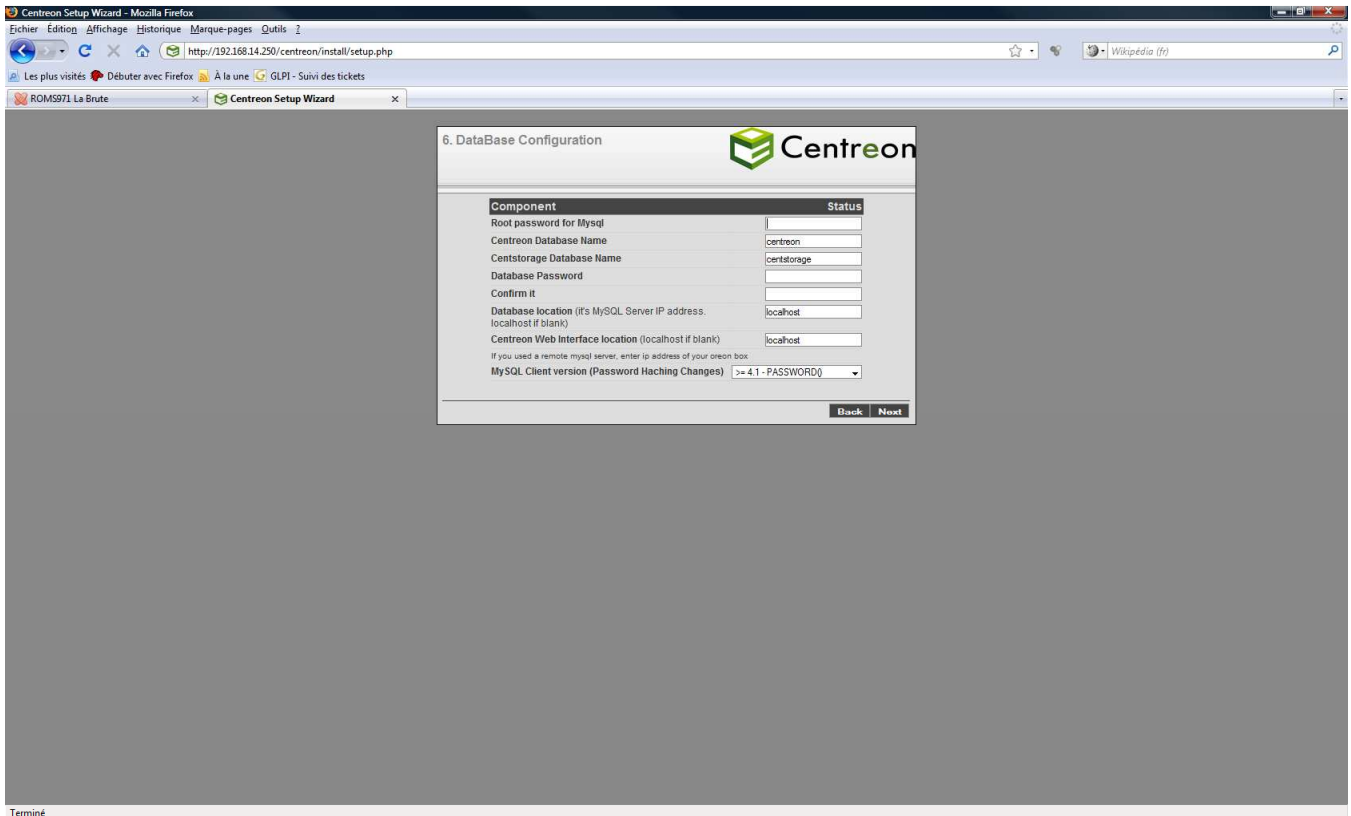
esac

# End of this script
```

```
# chmod +x /etc/init.d/ndo2db
# update-rc.d ndo2db defaults
# /etc/init.d/ndo2db start
```

1.8 CONFIGURATION WEB





Faites un premier test:

Configuration > Nagios > Generate

Sélectionnez les options suivantes:

- Generate Conf guration Files
- Run Nagios debug (-v)

Cliquez sur « Export ». Si aucune erreur n'est signalée, Exportez les configurations:

- Generate Configuration Files
- Run Nagios debug (-v)
- Move Export Files
- Restart Nagios

Cliquez sur « Export ». Nagios va démarrer avec la nouvelle configuration.

```
# /usr/local/nagios/bin/ndo2db -c /usr/local/nagios/etc/ndo2db.cfg  
#/etc/init.d/centstorage start
```