

Installer un domaine NIS

Olivier Hoarau (olivier.hoarau@funix.org)

V1.8 26 décembre 2009

1	Historique.....	2
2	Préambule.....	2
3	Introduction.....	3
4	Présentation.....	3
5	Installation et configuration du serveur.....	5
5.1	Installation.....	5
5.2	Configuration sous Mandriva.....	5
5.3	Configuration sous ubuntu.....	7
5.4	Exporter les home directories.....	8
6	Installation et configuration d'un client.....	9
6.1	Installation.....	9
6.2	Configuration NIS sous Mandriva.....	9
6.3	Lancement du client NIS sous Mandriva.....	9
6.4	Test de fonctionnement.....	10
6.5	Configuration du client si vous utilisez NFS.....	10
6.6	Configuration du client si vous utilisez l'automontage.....	10
6.7	Modifier /etc/passwd.....	11
6.8	Modifier /etc/shadow.....	12
7	Test de fonctionnement.....	12
8	Commandes diverses.....	12
8.1	Visualiser les maps.....	12
8.2	Changer son mot de passe	12
8.3	En cas de modif d'une map.....	12

1 Historique

V1.0	16 janvier 2000	Création du document
V1.1	29 octobre 2000	Mise à jour du document (changement de version des différents packages)
V1.2	3 décembre 2000	changement de version des différents packages
V1.3	16 décembre 2001	Passage à Mandrake 8.1, changement de version
V1.4	24 décembre 2002	Passage à Mandrake 9.0, changement de version.
V1.5	4 mai 2003	Passage à Mandrake 9.1, changement de version
V1.6	6 août 2005	Rajout des fichiers de sécurisation, toilettage
V1.7	26 juillet 2006	Installation du serveur sous ubuntu
V1.8	26 décembre 2009	rajout dans la configuration, modification de /etc/shadow

2 Préambule

Ce document a pour objet de présenter la manière d'installer un domaine NIS.
La dernière version de ce document est téléchargeable à l'URL <http://www.funix.org>

Ce document est sous licence Creative Commons Attribution-ShareAlike 3.0 Unported, le détail de la licence se trouve sur le site <http://creativecommons.org/licenses/by-sa/3.0/legalcode>. Pour résumer, vous êtes libres

- de reproduire, distribuer et communiquer cette création au public
- de modifier cette création

suivant les conditions suivantes:

- **Paternité** — Vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'oeuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'oeuvre).
- **Partage des Conditions Initiales à l'Identique** — Si vous transformez ou modifiez cette oeuvre pour en créer une nouvelle, vous devez la distribuer selon les termes du même contrat ou avec une licence similaire ou compatible.

Par ailleurs ce document ne peut pas être utilisé dans un but commercial sans le consentement de son auteur. Ce document vous est fourni "dans l'état" sans aucune garantie de toute sorte, l'auteur ne saurait être tenu responsable des quelconques misères qui pourraient vous arriver lors des manipulations décrites dans ce document.

3 Introduction

Pourquoi cette page alors qu'il existe un NIS-HOWTO fort bien détaillé, c'est justement parce qu'il était trop bien détaillé sans peut être aller à l'essentiel, j'avoue que je m'y suis un peu perdu, qui plus est il est peu "ancien" et aurait besoin d'un petit toilettage pour s'appliquer aux nouvelles versions de linux. Vous découvrirez dans ce HOW-TO, qu'il existe plusieurs types de NIS, le NIS traditionnel, le NIS+, mais aussi le NYS ! Pour ma part je me limite au NIS traditionnel, le tout testé sur un serveur et un client NIS Mandriva et/ou ubuntu.

4 Présentation

Admettons que vous disposiez de plusieurs postes linux connectés en réseau, vous êtes obligés de créer autant de comptes sur chaque machine que vous avez d'utilisateurs, le problème est que chaque utilisateur se retrouve avec une home directory différente sur chaque poste, mais aussi un mot de passe distinct, voire éventuellement un uid et gid différent. De même pour les fichiers **/etc/hosts**, quand vous rajoutez une machine, vous êtes obligés de modifier un à un tous les fichiers **/etc/hosts** de chaque machine. Avec un ensemble pareil, il est difficile de garder une certaine cohérence entre les machines, et c'est particulièrement déroutant pour les utilisateurs (mots de passe, homedirectory). D'où l'intérêt du domaine NIS, anciennement appelé YP (pour Yellow Page devenu marque déposée), dans un domaine NIS, on dispose d'un serveur NIS qui contient des fichiers de référence (appelés map), comme **/etc/passwd** ou **/etc/hosts**, les clients NIS vont consulter ces fichiers de référence. Ainsi si l'on rajoute un utilisateur sur le réseau, c'est uniquement le fichier de référence **/etc/passwd** du serveur qui sera modifié et distribué aux clients, on dispose donc d'un **/etc/passwd** identique pour tout le réseau qui se trouve physiquement sur le réseau, seul lui est modifié, le principe est le même pour les autres map disponibles (**hosts**, **group**, ...).

On verra par la suite que NIS s'utilise conjointement avec NFS (ou éventuellement l'automontage **autofs**). On présentera les deux méthodes, sachant que je préfère personnellement la dernière, car elle permet de pouvoir travailler en local sans le serveur NIS. On suppose dans la suite des opérations que NFS ou l'automontage est configuré sur votre machine (en serveur et client), si ce n'est pas le cas, référez vous au document "mountage" en tout genre.

En plus du serveur NIS vous pouvez disposer d'un serveur de secours, en cas de défaillance du premier, les clients iront consulter les maps sur le serveur de secours, appelé serveur slave, donc en cas de modification d'une map du serveur principal, si vous disposez d'un serveur slave vous devez immédiatement répercuter la modification des maps sur le serveur slave. Je n'ai personnellement pas testé cette fonctionnalité de serveur slave mais j'en parlerai quand même dans cette page, je ne garantis pas par contre que ça fonctionnera du premier coup.

Avant de commencer vous devez choisir un nom pour votre domaine NIS, on prendra pour l'exemple **alphaville**, vous pouvez évidemment choisir le nom de domaine qui vous plaît, vous êtes simplement limité par votre imagination. On suppose aussi que votre réseau utilise les adresses IP 192.168.13.X et votre serveur NIS a l'adresse 192.168.13.1 et a pour nom **obelix**.

Le but de la manœuvre est le suivant, qu'un utilisateur du réseau quelque soit le poste utilisé puisse retrouver automatiquement sa homedirectory et que la modification de son mot de passe sur n'importe quel poste entraîne la modification sur l'ensemble des postes.

Deux cas de figures possibles seront abordés:

cas 1: répertoire /home exporté

Tous les homes directory se trouvent sous **/home** sur le serveur, tous les utilisateurs du réseau y ont été déclarés. Sur les postes clients vous n'avez déclaré aucun utilisateur à part les utilisateurs système (dont root) qu'on trouve par défaut, en tout état de cause, vous devez avoir **/home** vide, de même que **/home** ne doit pas être le point de montage d'une quelconque partition. Par NFS ou automontage, le répertoire **/home** du serveur sera monté sur chacun des clients et donc l'utilisateur retrouvera sa home directory quelque soit le poste qu'il utilisera.

cas 2: répertoire /export/home exporté

Vous ne pouvez pas vider **/home** sur les postes clients pour diverses raisons. Dans ce cas, sur le serveur créer l'homedirectory de chaque utilisateur sous **/home**, puis on va faire un lien symbolique de **/home** vers **/export/home** et dans le fichier **/etc/passwd** au niveau des homedirectory on remplacera **/home** par **/export/home**. Sur les clients on montera **/home** du serveur sous un répertoire **/export**, chaque utilisateur retrouvera donc sa home directory sous **/export/home**, tout en ne touchant pas au répertoire **/home** des postes clients.

5 Installation et configuration du serveur

5.1 Installation

La mise en place d'un serveur nécessite l'installation du package **ypserv** sous Mandriva (avec **urpmi** pour gérer les dépendance). Cette installation va créer un ensemble de fichier se trouvant sous **/usr/sbin** commençant par **yp**, et des fichiers sous **/etc/rc.d** pour le lancement automatique et encore d'autres sous **/var/yp** et **/usr/lib/yp**.

Les daemons à lancer sont **ypserv** et **yppasswdd**, le premier et le daemon NIS lui-même, le second le daemon permettant de changer les mots de passe contenus dans la map **passwd** quelque soit le poste où l'on se trouve.

Sous ubuntu on installe tout simplement le package **nis**.

5.2 Configuration sous Mandriva

A présent éditer le fichier **/etc/sysconfig/network** et rajouter:

```
NISDOMAIN=alphaville
```

Maintenant on va dans **/var/yp** et on édite le fichier **Makefile** qui s'y trouve, voici les modifications que j'ai apporté au **Makefile** d'origine et les lignes que je juge importantes :

```
# If we have only one server, we don't have to push the maps to the  
# slave servers (NOPUSH=true). If you have slave servers, change this  
# to "NOPUSH=false" and put all hostnames of your slave servers in the file  
# /var/yp/ypservers.  
# si vous avez pas de serveur slave mettre true, sinon mettre false  
NOPUSH=true  
  
# We do not put password entries with lower UIDs (the root and system  
# entries) in the NIS password database, for security. MINUID is the  
# lowest uid that will be included in the password maps.  
# MINGID is the lowest gid that will be included in the group maps.  
# dans la table NIS des password il n'y aura pas les comptes dont l'UID est inférieur  
# à 500, cela correspond aux utilisateurs systèmes  
# idem pour les groupes  
# vos utilisateurs et vos groupes doivent donc avoir un numéro d'identification supérieur  
à 500  
MINUID=500  
MINGID=500  
  
(...)  
  
# Should we merge the passwd file with the shadow file ?  
# MERGE_PASSWD=true|false  
# Si vous utilisez les shadow password et que souhaitez fusionner  
# les fichiers passwd et shadow mettre true  
MERGE_PASSWD=false  
  
# Should we merge the group file with the gshadow file ?  
# MERGE_GROUP=true|false  
# si vous n'utilisez pas de shadow password pour les groupes laissez à false  
MERGE_GROUP=false  
  
(...)
```

```

# If you don't want some of these maps built, feel free to comment
# them out from this list.
# la ligne la plus important c'est la liste des maps susceptibles d'être gérées
# pour ma part je me suis limité à /etc/passwd /etc/group et /etc/hosts
#all: passwd group hosts rpc services netid protocols netgrp mail \
    #shadow publickey # networks ethers bootparams amd.home \
# auto.master auto.home passwd.adjunct
all: passwd group shadow

```

Maintenant on va pouvoir construire les maps proprement dites, en fait il va partir des fichiers ascii (**/etc/passwd**, **/etc/group**, et **/etc/hosts**) et générer des fichiers binaires (maps). Pour cela taper en tant que root:

domainname alphaville

Vous ne devez pas oublier aussi de lancer **ypserv** et **yppasswdd** en tapant :

```

/etc/rc.d/init.d/ypserv start
/etc/rc.d/init.d/yppasswdd start

```

A présent sous **/var/yp** tapez:

```
make
```

Un répertoire **alphaville** va se créer sous **/var/yp** avec les maps à l'intérieur:

```

gmake[1]: Entering directory `/var/yp/kervao'
Updating passwd.byname...
Updating passwd.byuid...
Updating group.byname...
Updating group.bygid...
Updating shadow.byname...
gmake[1]: Leaving directory `/var/yp/kervao'

```

A présent on va éditer le fichier **/etc/ypserv.conf** qui est le fichier de config du serveur nis (**ypserv**): voici son contenu:

```

#
# ypserv.conf  In this file you can set certain options for the NIS server,
#              and you can deny or restrict access to certain maps based
#              on the originating host.
#
#              See ypserv.conf(5) for a description of the syntax.
#
# Some options for ypserv.  This things are all not needed, if
# you have a Linux net.
dns: no
# The following, when uncommented, will give you shadow like passwords.
# Note that it will not work if you have slave NIS servers in your
# network that do not run the same server as you.
# Host          : Map          : Security : Passwd_mangle
#

```

```

# Host          : Domain : Map          : Security
#
#*              : *      : passwd.byname : port
#*              : *      : passwd.byuid  : port

# Not everybody should see the shadow passwords, not secure, since
# under MSDOG everybody is root and can access ports < 1024 !!!

*              : *      : shadow.byname : port
*              : *      : passwd.adjunct.byname : port

# If you comment out the next rule, ypserv and rpc.ypxfrd will
# look for YP_SECURE and YP_AUTHDES in the maps. This will make
# the security check a little bit slower, but you only have to
# change the keys on the master server, not the configuration files
# on each NIS server.
# If you have maps with YP_SECURE or YP_AUTHDES, you should create
# a rule for them above, that's much faster.
# *              : *      : none

```

Vous pouvez restreindre l'accès des tables NIS à votre sous réseau 192.168.13 en tapant par exemple :

```

192.168.13.      : shadow.byname : port
192.168.13.      : passwd.adjunct.byname : port

```

Dans un but de sécurisation on créera le fichier `/var/yp/securenets` qui aura cette tête là

```

# pour permettre l'accès sur le serveur même
255.0.0.0 127.0.0.0
# pour permettre l'accès de toutes les machines du sous-réseau (masque et adresse
réseau)
255.255.255.0 192.168.13.0

```

5.3 Configuration sous ubuntu

Il faut éditer le fichier `/etc/default/nis` et rajouter

```
NISSERVER=master
```

on édite maintenant le fichier `/etc/defaultdomain` où on indique le nom de domaine. Il n'est pas nécessaire de modifier les fichiers `/etc/yp.conf` et `/etc/ypserv.conf`. Maintenant on ouvre le fichier `/var/yp/Makefile`, pour exporter les utilisateurs et les groupes il suffit d'avoir la ligne suivante

```
all: passwd group shadow
```

Maintenant dans le fichier `/etc/ypserv.securenets` vous pouvez restreindre l'accès aux clients, pour rajouter le client 192.168.26.50 par exemple il suffit de taper

```

# This line gives access to everybody. PLEASE ADJUST!
#0.0.0.0      0.0.0.0
host 192.168.26.50

```

Maintenant pour diffuser les tables il suffit de taper

```
sudo /usr/lib/yp/ypinit -m
```

Voilà le résultat

At this point, we have to construct a list of the hosts which will run NIS servers. **obelix.kervao.fr** is in the list of NIS server hosts. Please continue to add the names for the other hosts, one per line. When you are done with the list, type a <control D>.

```
next host to add: obelix.kervao.fr
next host to add: CTRL-D
```

The current list of NIS servers looks like this:

```
obelix.kervao.fr
```

Is this correct? [y/n: y] y

We need a few minutes to build the databases...

Building /var/yp/alphaville/ypservers...

Running /var/yp/Makefile...

```
make[1]: entrant dans le répertoire « /var/yp/alphaville »
```

```
Updating passwd.byname...
```

```
failed to send 'clear' to local ypserv: RPC: Program not registeredUpdating
passwd.byuid...
```

```
failed to send 'clear' to local ypserv: RPC: Program not registeredUpdating
group.byname...
```

```
failed to send 'clear' to local ypserv: RPC: Program not registeredUpdating
group.bygid...
```

```
failed to send 'clear' to local ypserv: RPC: Program not registeredUpdating
shadow.byname...
```

```
failed to send 'clear' to local ypserv: RPC: Program not registeredmake[1]: quittant le
répertoire « /var/yp/alphaville »
```

obelix.kervao.fr has been set up as a NIS master server.

Now you can run **ypinit -s obelix.kervao.fr** on all slave server.

5.4 Exporter les home directories

Qu'on veuille utiliser NFS ou l'automontage, qu'on soit dans le cas 1 ou 2 énoncé plus haut dans le chapitre de présentation, la manip est la même, on doit exporter le répertoire **/home**, pour cela on rajoutera la ligne suivante au fichier **/etc/exports**:

```
/home
```

Eventuellement si vous voulez limiter l'accès aux machines **toto,titi,tata** en lecture/écriture sur le répertoire en question:

```
/home titi(rw) tata(rw) toto(rw)
```

Pour que le fichier soit pris en compte, n'oubliez pas de taper:

```
exportfs -a
```

Maintenant si vous êtes dans le cas 1, c'est terminé (ou presque), dans le cas 2, vous devez faire en plus sur le serveur, un lien de **/home** vers **/export**


```
mkdir /export  
ln -s /home /export
```

Et modifiez **/etc/passwd** pour remplacer **/home** par **/export/home** comme la ligne donnée en exemple:

```
veronique:3GGWf/uWiPoWM:4002:4000::/export/home/veronique:/bin/bash
```

NOTE: Dans le cas de shadow password, au lieu du mot de passe crypté on voit un **x**.

Maintenant vous devez rebatir la map **passwd**, puisque vous venez de modifier **/etc/passwd**, sous **/var/yp**, tapez:

```
make
```

6 Installation et configuration d'un client

6.1 Installation

Vous devez installer **yp-tools**, **portmap** et **ypbind** avec **urpmi**. Ca va vous installer des exécutables commençant par **yp** sous **/usr/bin** et **/usr/sbin** et un répertoire **/var/yp**. **yp-tools** est un package contenant une ensemble d'outils pour le NIS, **ypbind** est le daemon qui va tourner sur les postes clients.

6.2 Configuration NIS sous Mandriva

Editez le fichier **/etc/sysconfig/network** et rajoutez:

```
NISDOMAIN=alphaville
```

Ensuite éditez le fichier **/etc/yp.conf** et rajouter

```
domain alphaville server obelix
```

Vous pouvez éventuellement mettre l'adresse IP de votre serveur NIS. Eventuellement vous pouvez rajouter les serveurs slaves en rajoutant à la suite:

```
ypserver nom-du-serveur-slave
```

Maintenant vous devez modifier votre fichier **/etc/nsswitch.conf** pour les lignes **hosts**, **group** et **passwd** (uniquement les maps exportés) pour prendre en compte d'abord les fichiers **/etc/hosts**, **/etc/group** et **/etc/passwd** locaux puis ceux du serveur NIS, on aura donc pour les lignes en question:

```
passwd: compat
```

```
shadow: compat
```

```
group: compat
```

```
hosts: files nis dns
```

L'argument **compat** permet de pouvoir mixer des informations du serveur NIS et des informations du poste client (voir illustration plus bas).

6.3 Lancement du client NIS sous Mandriva

Tapez maintenant les commandes suivantes:

```
domainname alphaville  
/etc/rc.d/init.d/ypbind start  
/etc/rc.d/init.d/portmap start
```

6.4 Test de fonctionnement

Pour voir si tout marche, il suffit de taper sur le client:

```
ypwhich
```

Vous devriez voir en réponse:

```
obelix
```

C'est à dire le nom de votre serveur NIS.

6.5 Configuration du client si vous utilisez NFS

Cas 1

Rajoutez dans **/etc/fstab**

```
tavel:/home /home nfs defaults 0 0
```

Cas 2

Vous devez créer un répertoire **/export/home**

```
mkdir /export  
mkdir /export/home
```

Puis rajoutez dans **/etc/fstab** la ligne suivante:

```
tavel:/home /export/home nfs defaults 0 0
```

Dans les deux cas, tapez:

```
mount -a
```

Pour que le répertoire soit effectivement monté.

6.6 Configuration du client si vous utilisez l'automontage

Cas 1

Rajoutez à votre fichier **/etc/auto.master** la ligne suivante:

```
/ /etc/auto.home --timeout=60
```

Cas 2

Créer un répertoire **/export** :

```
mkdir /export
```

Rajoutez au fichier **/etc/auto.master** la ligne suivante:

```
/export /etc/auto.home --timeout=60
```

Dans les deux cas le fichier **/etc/auto.home** contiendra:

```
home -fstype=nfs obelix:/home
```

Pour relancer l'automontage et prendre en compte les nouvelles modifications, il suffit de taper:

```
/etc/rc.d/init.d/autofs restart
```

6.7 Modifier /etc/passwd

En toute rigueur au fichier **/etc/passwd** du client vous devez rajouter la ligne suivante tout à la fin:

```
+::::::
```

Mais ce n'est pas absolument nécessaire, ça marchera très bien sans. Mais on peut aller plus loin, les ":" correspondent en fait aux champs de **/etc/passwd**, qui je le rappelle sont:

```
login:password:uid:gid:commentaire:homedirectory:shell
```

Si vous voulez que l'utilisateur toto déclaré sur le serveur maitre utilise un shell **/bin/ksh**, vous rajouterez donc:

```
+toto::::::/bin/ksh
```

Si vous voulez que celui ci ait comme homedirectory **/tmp**

```
+toto::::::/tmp:
```

C'est l'intérêt de l'argument **compat** du fichier **/etc/nsswitch.conf**, sans quoi il ne sera pas possible de panacher des informations du serveur NIS et des informations des fichiers locaux.

Enfin si vous voulez limiter le nombre de comptes accessibles par nis (utilisateur toto, titi, tata), il suffira de rajouter:

```
+toto::::::
```

```
+titi::::::
```

```
+tata::::::
```

Concrètement, j'ai un serveur sous kubuntu et un poste client sous Mandriva avec une version de KDE différente, du coup sous kubuntu ma home directory est **/export/home/user** sur le poste Mandriva **/home/user** qui est un répertoire local qui me permet d'avoir des environnements KDE distincts mais par contre en retrouvant toutes mes données (je fais des liens pour le répertoire **Documents**, **.thunderbird**, etc.).

6.8 Modifier /etc/shadow

Vous devez mettre un + tout à la fin sinon ça ne marchera pas !

7 Test de fonctionnement

Le meilleur test consiste en tant que simple utilisateur est de vous loguer sur tous vos postes (client et serveur) et de retrouver votre unique home directory.

8 Commandes diverses

8.1 Visualiser les maps

Vous disposez de la commande **ypcat** qui vous permet d'éditer les maps, ainsi pour éditer la map **passwd** il suffit de taper:

```
ypcat passwd
```

Vous verrez donc tous les comptes accessibles sur le réseau.

8.2 Changer son mot de passe

Si vous voulez modifier votre mot de passe, vous ne devez plus taper tout simplement **passwd** mais **yppasswd** pour que ça soit pris en compte dans la map **passwd** présente sur le serveur NIS.

Pour que ce soit transparent pour l'utilisateur vous pouvez très bien renommer **passwd** sur vos postes clients.

```
mv /usr/bin/passwd /usr/bin/passwd.sansnis
```

Puis faire un lien de **passwd** vers **yppasswd**

```
ln -s /usr/bin/yppasswd /usr/bin/passwd
```

ATTENTION: il se peut qu'il y ait un léger temps de latence entre le temps où vous avez tapé votre mot de passe et le temps où le serveur NIS va reconstruire ses maps, donc ne vous étonnez pas si après modif du mot de passe, vous essayez de vous loguer et que vous vous faites jeter, il suffit d'attendre un peu ou de se loguer encore avec l'ancien mot de passe.

8.3 En cas de modif d'une map

Si sur le serveur NIS vous avez à changer une map (pour rajouter un poste dans **/etc/hosts** par exemple), il suffit d'aller dans **/var/yp** et de taper **make**, normalement les maps des serveurs slaves devraient être mises à jour automatiquement.

Sous l'unix d'HP, HP-UX, on trouve la commande **yppmake** qui réalise les fonctions suivantes :

```
cd /var/yp  
make
```

Sous linux rien ne vous empêche de créer le script **ypmake** sous **/usr/sbin** contenant

```
#!/bin/bash  
cd /var/yp  
make
```

Avec les droits 744

```
chmod 744 /usr/sbin/ypmake
```

Plus simplement vous disposez également de la commande **ypinit** à taper avec l'option **-m**