

# TP NIS

## 1 PREREQUIS

Nous utilisons deux machines virtuelles Ubuntu sur Oracle VirtualBox.

Voici leur adresse IP :

- NIS-Server : 192.168.46.2
- tp-client (Ubuntu) : 192.168.46.3

Les deux machines peuvent se ping et leur correspondance est effectuée dans /etc/hosts :

```
thibaut@Ubuntu:~$ ping nis-server
PING nis-server (192.168.46.2): 56(84) bytes of data:
64 bytes from nis-server (192.168.46.2): icmp_seq=1 ttl=64 time=0.653 ms
64 bytes from nis-server (192.168.46.2): icmp_seq=2 ttl=64 time=0.968 ms
64 bytes from nis-server (192.168.46.2): icmp_seq=3 ttl=64 time=3.93 ms
^C
--- nis-server ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2019ms
rtt min/avg/max/mdev = 0.653/1.849/3.928/1.475 ms

thibaut@nis-server:~$ ping tp-client
PING tp-client (192.168.46.3) 56(84) bytes of data:
64 bytes from tp-client (192.168.46.3): icmp_seq=1 ttl=64 time=0.693 ms
64 bytes from tp-client (192.168.46.3): icmp_seq=2 ttl=64 time=0.887 ms
64 bytes from tp-client (192.168.46.3): icmp_seq=3 ttl=64 time=0.941 ms
^C
--- tp-client ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2803ms
rtt min/avg/max/mdev = 0.693/0.840/0.941/0.106 ms
```

*Vérification des hosts sur machine serveur (gauche) et machine client (droite)*

Nous avons nettoyé les comptes utilisateurs inutiles du dernier TP.

Nous vérifions les paramètres exports et possibilité de monter le /home à distance sur le client.

```
thibaut@nis-server: ~
GNU nano 6.2 /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_sub>
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/home tp-client(rw,no_root_squash)
```

*Paramètres exports*

Ajout d'un utilisateur « user » (en lieu et place des utilisateurs « tann » et « colnot ») sur le serveur et le client, puis d'un utilisateur testNIS sur le serveur.

```
sudo useradd user -p user -m
sudo useradd testNIS -p testNIS -m
```

```
thibaut@nis-server:~$ sudo nano /etc/exports
thibaut@nis-server:~$ sudo systemctl restart rpcbind
thibaut@nis-server:~$ sudo systemctl restart nfs-kernel-server
thibaut@nis-server:~$ sudo exportfs -a
exportfs: /etc/exports [2]: Neither 'subtree_check' or 'no_subtree_check' specified for export "tp-client:/home".
    Assuming default behaviour ('no_subtree_check').
    NOTE: this default has changed since nfs-utils version 1.0.x

thibaut@nis-server:~$ sudo exportfs
/home                tp-client
```

*Restart des services, mise à jour exportfs*

Sur le client, le montage fonctionne automatiquement grâce à la configuration du dernier tp dans /etc/fstab :

```
thibaut@Ubuntu:~$ sudo ls -l /mnt/FromNFS
total 12
drwxr-x---  2    1201    1201 4096 mai   17 14:06 testNIS
drwxr-x--- 21 thibaut thibaut 4096 mai   17 13:49 thibaut
drwxr-x---  2 user    user    4096 mai   17 14:05 user
```

*Le dossier est bien monté sur le client*

## 2 CONFIGURATION SERVEUR

Ensuite, nous vérifions la présence des paquets nécessaires et/ou les installons:

```
# Installation de ypserv
sudo apt install nis

# Vérification presence ypserv
which ypserv
which rpcbind

# verification portmapper
rpcinfo | grep portmapper
```

```
thibaut@nis-server:~$ which ypserv
/usr/sbin/ypserv
thibaut@nis-server:~$ which rpcbind
/usr/sbin/rpcbind
thibaut@nis-server:~$ rpcinfo | grep portmapper
100000  4  tcp6      ::0.111      portmapper superuser
100000  3  tcp6      ::0.111      portmapper superuser
100000  4  udp6      ::0.111      portmapper superuser
100000  3  udp6      ::0.111      portmapper superuser
100000  4  tcp       0.0.0.0.0.111 portmapper superuser
100000  3  tcp       0.0.0.0.0.111 portmapper superuser
100000  2  tcp       0.0.0.0.0.111 portmapper superuser
100000  4  udp       0.0.0.0.0.111 portmapper superuser
100000  3  udp       0.0.0.0.0.111 portmapper superuser
100000  2  udp       0.0.0.0.0.111 portmapper superuser
100000  4  local     /run/rpcbind.sock portmapper superuser
100000  3  local     /run/rpcbind.sock portmapper superuser
```

*Les services sont installés et portmapper est en service.*

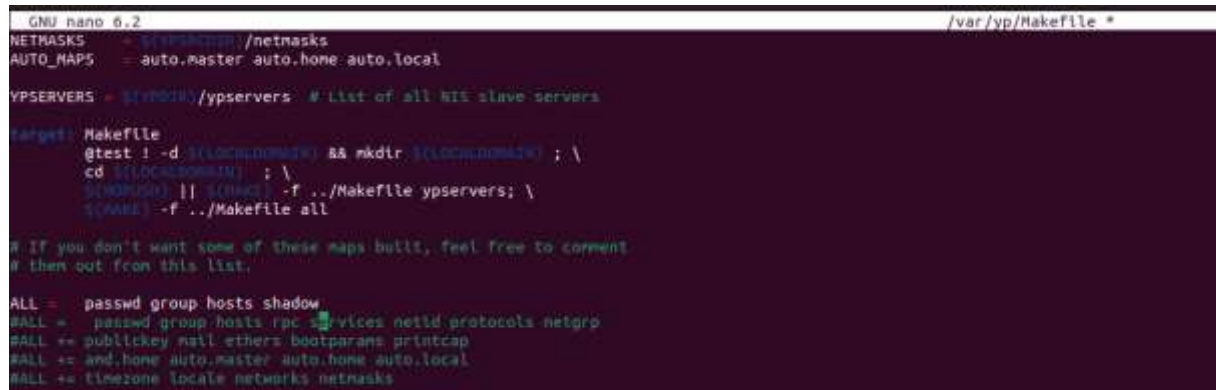
Modification du nom de domaine nis à Efrei, dans le fichier `/etc/defaultdomain` :

```
echo 'efrei' | sudo tee /etc/defaultdomain
```

Modification du makefile :

```
sudo nano /var/yp/Makefile
```

Les maps sont créées pour les bases de données renseignées dans `ALL` :



```
GNU nano 6.2 /var/yp/Makefile *
NETMASKS = $(YPSERVER)/netmasks
AUTO_MAPS = auto.master auto.home auto.local

YPSERVERS = $(YPSERVER)/ypservers # List of all NIS slave servers

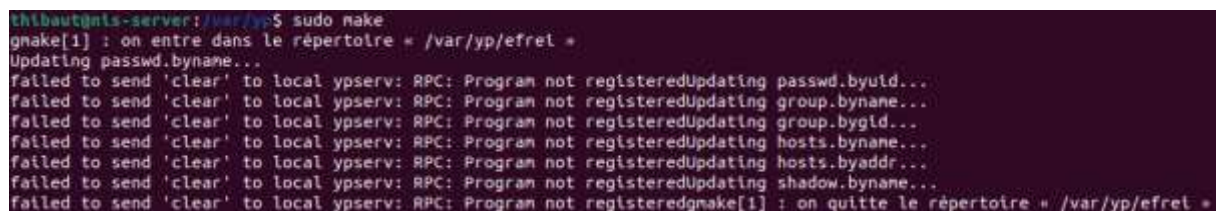
target: Makefile
@test ! -d $(LOCALDOMAIN) && mkdir $(LOCALDOMAIN) ; \
cd $(LOCALDOMAIN) ; \
$(MAKE) || $(MAKE) -f ../Makefile ypservers; \
$(MAKE) -f ../Makefile all

# If you don't want some of these maps built, feel free to comment
# them out from this list.

ALL = passwd group hosts shadow
#ALL += passwd group hosts rpc services netid protocols netgrp
#ALL += publickey nall ethers bootparams printcap
#ALL += and.home auto.master auto.home auto.local
#ALL += timezone locale networks netmasks
```

*Fichier makefile dans `/var/yp`*

Construction des maps :



```
thibaut@nis-server:/var/yp$ sudo make
make[1]: on entre dans le répertoire « /var/yp/efrei »
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 hosts.byname...
failed to send 'clear' to local ypserv: RPC: Program not registeredUpdating hosts.byaddr...
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]: on quitte le répertoire « /var/yp/efrei »
```

*Résultat de l'exécution de make*

```
sudo systemctl start ypserv
sudo systemctl start yppasswdd
```

Afin de voir les différentes possibilités de configuration, nous parcourons `/etc/ypserv.conf` :

```
sudo nano /etc/ypserv.conf
```

```

GNU nano 6.2
#
# ypserv.conf  In this file you can set certain options for the NIS ser
#               and you can deny or restrict access to certain maps base
#               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.
#
# How many map file handles should be cached ?
files: 30
#
# Should we register ypserv with SLP? Only available if SLP support
# is compiled in. Deprecated functionality.
slp: no
# After how many seconds we should re-register ypserv with SLP?
slp_timeout: 3600
#
# xfr requests are only allowed from ports < 1024
xfr_check_port: yes
#
# 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.
# IMPORTANT: this rules will be ignored for IPV6 connections!
#
# Host          : Domain  : Map          : Security
# *              : *      : passwd.byname : port
# *              : *      : passwd.byuid  : port
#
# This is the default - restrict access to the shadow password file,
# allow access to all others.
*              : *      : 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

```

*Fichier yp.conf*

Lorsque nous réexécutons le Makefile après le démarrage de ypserv, nous n'avons plus d'erreur :

```

thibaut@nls-server:/var/yp$ sudo make
gnake[1] : on entre dans le répertoire « /var/yp/efrei »
gnake[1]: rien à faire pour « all ».
gnake[1] : on quitte le répertoire « /var/yp/efrei »
thibaut@nls-server:/var/yp$

```

Vérification des services :



```
thibaut@nis-server:/var/yp$ sudo systemctl status yppasswdd
● yppasswdd.service - NIS Users Passwords Change Server
   Loaded: loaded (/lib/systemd/system/yppasswdd.service; disabled; vendor preset: enabled)
   Active: active (running) since Fri 2024-05-17 15:25:28 CEST; 7min ago
     Process: 4929 ExecStartPre=/bin/domainname -F /etc/defaultdomain (code=exited, status=0/SUCCESS)
     Process: 4930 ExecStart=/usr/sbin/rpc.yppasswdd -D $YPPWDDIR -e $YPCHANGEOK $YPPASSWDDARGS (code=exited, status=0/SUCCESS)
    Main PID: 4932 (rpc.yppasswdd)
      Tasks: 1 (limit: 4598)
     Memory: 624.0K
        CPU: 10ms
    CGroup: /system.slice/yppasswdd.service
            └─4932 /usr/sbin/rpc.yppasswdd -D /etc -e chsh

mai 17 15:25:28 nis-server systemd[1]: Starting NIS Users Passwords Change Server...
mai 17 15:25:28 nis-server systemd[1]: Started NIS Users Passwords Change Server.
thibaut@nis-server:/var/yp$ sudo systemctl status ypserv
● ypserv.service - NIS/YP (Network Information Service) Server
   Loaded: loaded (/lib/systemd/system/ypserv.service; disabled; vendor preset: enabled)
   Active: active (running) since Fri 2024-05-17 15:25:21 CEST; 7min ago
     Process: 4905 ExecStartPre=/bin/domainname -F /etc/defaultdomain (code=exited, status=0/SUCCESS)
     Process: 4906 ExecStart=/usr/sbin/ypserv $YPSERVARGS (code=exited, status=0/SUCCESS)
    Main PID: 4908 (ypserv)
      Tasks: 1 (limit: 4598)
     Memory: 608.0K
        CPU: 11ms
    CGroup: /system.slice/ypserv.service
            └─4908 /usr/sbin/ypserv

mai 17 15:25:21 nis-server systemd[1]: Starting NIS/YP (Network Information Service) Server...
```

*Les services sont bien actifs sur le serveur*

### 3 CONFIGURATION CLIENT

Nous vérifions si les paquets rpcbind, yptools, ypbind sont installés sur notre machine et, dans le cas contraire, les installons :

```
sudo apt install yp-tools
which rpcbind
which yp-tools
which ypbind
nano /etc/yp.conf
```

```
GNU nano 6.2 /etc/yp.conf
#
# yp.conf Configuration file for the ypbind process. You can define
# NIS servers manually here if they can't be found by
# broadcasting on the local net (which is the default).
#
# See the manual page of ypbind for the syntax of this file.
#
# IMPORTANT: For the "ypserver", use IP addresses, or make sure that
# the host is in /etc/hosts. This file is only interpreted
# once, and if DNS isn't reachable yet the ypserver cannot
# be resolved and ypbind won't ever bind to the server.
domain efrei server nis-server
ypserver nis-server
```

*La configuration du serveur pour le client*

- ⇒ Nous définissons le domaine efrei du serveur « nis-server » (déclaré dans /etc/hosts). Ensuite, nous définissons le

```
nano /etc/nsswitch.conf
```

```

GNU nano 6.2 /etc/nsswitch.conf
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the 'glibc-doc-reference' and 'info' packages installed, try:
# 'info libc "Name Service Switch"' for information about this file.

passwd:      nis files
group:       nis files
shadow:      nis files
gshadow:     nis files

hosts:       nis files dns mdns4_minimal [NOTFOUND=return]
networks:    files

protocols:   db files
services:    db files
ethers:      db files
rpc:         db files

netgroup:    nis

```

*Configuration nsswitch, ajouté nis sur passwd, group, shadow et hosts*

Création du defaultdomain et lancement du service nis (ypbind) sur le client:

```

thibaut@Ubuntu:~$ sudo nano /etc/defaultdomain
thibaut@Ubuntu:~$ sudo service ypbind restart
thibaut@Ubuntu:~$ sudo cat /etc/defaultdomain
efrei
thibaut@Ubuntu:~$ sudo systemctl status ypbind.service
● ypbind.service - NIS Binding Service
   Loaded: loaded (/lib/systemd/system/ypbind.service; disabled; vendor preset: enabled)
   Active: active (running) since Fri 2024-05-17 15:48:18 CEST; 1min 1s ago
     Process: 7416 ExecStartPre=/bin/domainname -F /etc/defaultdomain (code=0)
     Process: 7417 ExecStart=/usr/sbin/ypbind $YPBINDARGS (code=0)
    Main PID: 7419 (ypbind)
       Tasks: 3 (limit: 4598)
      Memory: 760.0K
         CPU: 22ms
       CGroup: /system.slice/ypbind.service
               └─7419 /usr/sbin/ypbind

```

*Modif du defaultdomain et vérif statut ypbind*

## 4 TEST DE FONCTIONNEMENT NIS

```

thibaut@Ubuntu:~$ ypcat -d efrei -h nis-server passwd
testNIS:x:1201:1201::/home/testNIS:/bin/sh
user:x:1200:1200::/home/user:/bin/sh
thibaut:x:1000:1000:Thibaut,,,:/home/thibaut:/bin/bash
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin

```

*Essai de la command ypcat : on voir le fichier passwd*

Nous arrivons bien à nous login en tant que « testNIS » sur le client :

```
thibaut@Ubuntu:~$ su - testNIS
Mot de passe :
su: warning: cannot change directory to /home/testNIS: No such file or directory
$ whoami
testNIS
$
```

*La connexion est réussie*

```
thibaut@Ubuntu:~$ ypwhich
nis-server
thibaut@Ubuntu:~$ ypmatch -d efrei user passwd
user:x:1200:1200::/home/user:/bin/sh
thibaut@Ubuntu:~$
```

*Essai des commandes ypwhich et ypmatch*

## 5 ADJONCTION NFS

Le montage du répertoire `/home` du serveur sur notre client, dans `/mnt/FromNFS`, se fait déjà au démarrage ou manuellement (avec `mount -a`) grâce à la configuration dans `/etc/fstab` (voir prérequis).

Le répertoire utilisateur de testNIS est défini sur le serveur dans le fichier `/etc/passwd`. Nous pouvons donc modifier ce fichier sur le serveur :

```
thibaut@nis-server:/var/yp$ sudo nano /etc/passwd
[sudo] Mot de passe de thibaut :
thibaut@nis-server:/var/yp$ sudo getent passwd | grep testNIS
testNIS:x:1201:1201::/mnt/FromNFS/home/testNIS:/bin/sh
```

*Configuration du répertoire utilisateur sur le serveur*

Maintenant, connectons-nous sur le client avec testNIS et observons le résultat :

```
root@Ubuntu:~# su - testNIS
$ pwd
/mnt/FromNFS/testNIS
$ touch test-file
$ nano test-file
$ cat test-file
-sh: 4: st-file: not found
$ cat test-file
Test
$
```

Regardons maintenant sur le serveur :



```

root@nis-server:~# ls -l /home/testNIS
total 4
-rw-rw-r-- 1 testNIS testNIS 5 mai 17 19:42 test-file
root@nis-server:~# cat /home/testNIS/test-file
Test
root@nis-server:~#

```

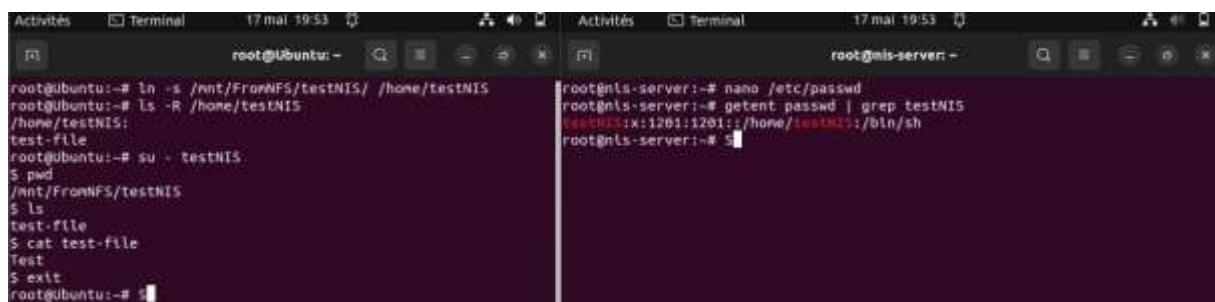
→ Le fichier est bien présent !

Nous pourrions alternativement faire un lien symbolique de `/home/testNIS` à `/mnt/FromNFS/home/testNIS` (avant cela, sur le serveur, nous reconfigurons le home de testNIS à `/home/testNIS`).

Le désavantage de cette dernière approche est qu'elle est décentralisée et nécessite une action supplémentaire, à effectuer sur tous les clients. De plus, si le répertoire cible est renommé, supprimé ou déplacé, on aura une rupture de lien. Néanmoins, l'utilisateur testNIS pourra accéder de manière transparente à `/home/testNIS` sur le client.

Commande pour créer le lien symbolique :

```
ln -s /mnt/FromNFS/testNIS /home/testNIS
```



The image shows two terminal windows side-by-side. The left window is on a client machine (root@ubuntu) and shows the command `ln -s /mnt/FromNFS/testNIS /home/testNIS` being executed. It then shows `ls -R /home/testNIS` outputting `/home/testNIS: test-file`. The user then switches to the `testNIS` user (`su - testNIS`), runs `pwd` showing `/mnt/FromNFS/testNIS`, `ls` showing `test-file`, and `cat test-file` outputting `Test`. The right window is on the nis-server machine (root@nis-server) and shows the user editing `/etc/passwd` with `nano` to add the `testNIS` user entry: `testNIS:x:1201:1201::/home/testNIS:/bin/sh`.

## 6 CONCLUSION

Grâce à NFS et NIS, nous avons atteint le but du TP :

- L'authentification et les droits utilisateurs sont centralisés sur la machine nis-server.
- Sur le client, un utilisateur testNIS peut s'authentifier sans que son identité figure dans les fichiers d'administration locale.
- L'utilisateur a accès à un répertoire personnel, centralisé sur nis-server. Ces fichiers sont absents du système local du client. Il aura accès aux mêmes fichiers s'il se connecte sur un autre client configuré de la même manière.