

Aula de informática

Arcadio Ortega Reinoso



Este documento está bajo licencia:



Atribución – No Comercial – Compartir Igual (CC BY-NC-SA 4.0)

Puedes compartir, copiar, redistribuir y adaptar esta obra con fines no comerciales, siempre que cites al autor y mantengas esta misma licencia para cualquier derivado.

Aula Informática

servidor NIS + NFS

```
sudo apt update
sudo apt upgrade
```

```
sudo apt -y install nis
```

```
sudo vi /etc/default/nis
#
# /etc/default/nis Optional configuration settings for the NIS programs.
#
#
# The following two variables are still used in the init script, but
# ignored by systemd. See `nis.debian.howto` in the documentation
# directory for more information.
#
# Are we a NIS server and if so what kind (values: false, slave, master)?
#AOR###
NISERVER=master
#AOR###
# Are we a NIS client?
NISCLIENT=false
...
...
```

```
sudo vi /etc/ypserv.securenets
#
# securenets This file defines the access rights to your NIS server
# for NIS clients. This file contains netmask/network
# pairs. A clients IP address needs to match with at least
# one of those.
#
# One can use the word "host" instead of a netmask of
# 255.255.255.255. Only IP addresses are allowed in this
# file, not hostnames.
#
# Always allow access for localhost, IPv4 and IPv6
255.0.0.0 127.0.0.0
host ::1

# This lines gives access to everybody. PLEASE ADJUST!
#AOR###
#0.0.0.0 0.0.0.0
#::/0
255.255.255.0 192.168.1.0
#AOR###
```

```
sudo vi /etc/hosts
127.0.0.1 localhost
127.0.1.1 aulasrv
#
#AOR###
192.168.1.121 desk1.aula.poveda.com desk1

# The following lines are desirable for IPv6 capable hosts
::1    ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

```
sudo vi /etc/defaultdomain
#AOR##
aula.poveda.com
#AOR##
```

```
sudo systemctl restart rpcbind ypserv yppasswdd ypxfrd
sudo systemctl enable rpcbind ypserv yppasswdd ypxfrd
```

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

```
sudo adduser --disabled-login user1
sudo passwd user1
```

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

```
sudo apt -y install nfs-kernel-server
```

```
sudo vi /etc/ldapd.conf
[General]

Verbosity = 0
# set your own domain here, if it differs from FQDN minus hostname
#AOR###
# Domain = localdomain
Domain = aula.poveda.com
#AOR##

[Mapping]

Nobody-User = nobody
Nobody-Group = nogroup
```

```
sudo vi /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_subtree_check)
#
# 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)
#
#AOR#####
/home 192.168.1.0/24(rw,sync,no_subtree_check,no_root_squash)
#AOR###
```

Aplica los cambios:

```
sudo exportfs -ra
```

Reiniciamos y ponemos disponible el servicio de NFS

```
sudo systemctl enable nfs-server
sudo systemctl restart nfs-server
```

```
cat /etc/passwd
...
user1:x:1001:1001:,,,:/home/user1:/bin/bash
```

Usuarios locales

Configurar `/etc/default/useradd` para usuarios locales

```
sudo vi /etc/default/useradd  
HOME=/localhome
```

Hay que crear primeramente el `/localhome`:

```
sudo mkdir /localhome  
sudo chown root:root /localhome  
sudo chmod 755 /localhome  
sudo useradd -m -d /localhome/usuario usuario
```

NOTA: LO MISMO PARA `/etc/adduser.conf`

cliente NIS + NFS

```
sudo apt update
sudo apt upgrade
```

```
sudo vi /etc/hosts
127.0.0.1 localhost
127.0.1.1 desk1.aula.poveda.com desk1
#AOR#
192.168.1.36 aula.poveda.com aula
#AOR#
...
...
```

```
sudo apt -y install nfs-common
```

```
sudo vi /etc/idmapd.conf
admon@desk1:~$ sudo cat /etc/idmapd.conf
[General]

Verbosity = 0
# set your own domain here, if it differs from FQDN minus hostname
# Domain = localdomain
#AOR###
Domain = aula.poveda.com
#AOR##

[Mapping]

Nobody-User = nobody
Nobody-Group = nogroup
```

```
sudo vi /etc/fstab
# /etc/fstab: static file system information.
...
...
#AOR#
aula:/home /home nfs defaults 0 0
#AOR#
```

```
sudo apt -y install nis
```

```
sudo vi /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.

# ypserver ypserver.network.com
#AOR###
ypserver aula.poveda.com server aula.poveda.com
#AOR###
```

```
sudo vi /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.
#AOR###
passwd:      files systemd nis
group:       files systemd nis
shadow:      files nis
gshadow:     files nis

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

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

netgroup:    nis
```

```
sudo vi /etc/defaultdomain
aula.poveda.com
```

```
sudo vi /etc/pam.d/common-session
# end of pam-auth-update config
#AOR###
session optional      pam_mkhomedir.so skel=/etc/skel umask=077
#AOR##
```

```
sudo systemctl restart rpcbind nsd ypbind
sudo systemctl enable rpcbind nsd ypbind
```

autofs

```
sudo apt -y install autofs
```

```
sudo vi /etc/auto.master  
#  
...  
...  
#  
+auto.master  
#AOR### add to the end  
/home /etc/auto.mount  
#AOR###
```

```
sudo vi /etc/auto.home  
#AOR###  
* -rw,nosuid,soft aula.poveda.com:/home/&  
#AOR###
```

```
root@node01:~# systemctl restart autofs
```


Crear cuentas de usuario con un script

El siguiente script en Bash crea cuentas para los estudiantes, asigna directorios personales y genera contraseñas aleatorias.

Importante: No está verificado, VERLO!!

Script: crear_usuarios.sh

Crea un archivo llamado crear_usuarios.sh con este contenido:

```
#!/bin/bash
#!/bin/bash

# Archivo de entrada con los datos de los usuarios
archivo_usuarios="usuarios.txt"

# Verifica si el archivo existe
if [[ ! -f "$archivo_usuarios" ]]; then
    echo "El archivo $archivo_usuarios no existe. Por favor, crea uno con los datos de los usuarios."
    exit 1
fi

# Crear usuarios
while IFS= read -r linea; do
    # Extraer partes: DNI y nombre completo
    dni=$(echo "$linea" | awk '{print $1}')
    nombre_completo=$(echo "$linea" | cut -d' ' -f2-)

    # Validar el formato del DNI
    if [[ ! "$dni" =~ ^[0-9]{8}[A-Z]$ ]]; then
        echo "El DNI '$dni' no tiene un formato válido. Se omite esta línea."
        continue
    fi

    # Crear el nombre de usuario a partir del DNI (letra + números)
    username="${dni:8:1}${dni:0:8}" # Letra seguida de los 8 dígitos

    # Generar una contraseña aleatoria (fácil de recordar)
    password=$(< /dev/urandom tr -dc 'a-zA-Z0-9' | head -nc8)

    # Crear el usuario con su directorio personal y comentario (nombre completo)
    sudo useradd -m -s /bin/bash -c "$nombre_completo" "$username"

    # Establecer la contraseña
    echo "$username:$password" | sudo chpasswd


    # Mostrar las credenciales
    echo "Usuario: $username | Contraseña: $password | Nombre completo: $nombre_completo"

    # Configurar cuotas (opcional, si las necesitas de inmediato)
    sudo setquota -u "$username" 100000 120000 0 0 /
done < "$archivo_usuarios"

echo "Todos los usuarios han sido creados."
```


Pasos para ejecutar el script

1. Crea un archivo usuarios.txt con los nombres de los estudiantes, un nombre por línea:




```
12345678A Juan Pérez Gómez  
87654321B María López Díaz
```

2. Da permisos de ejecución al script:



```
chmod +x crear_usuarios.sh
```

3. Ejecuta el script:



```
sudo ./crear_usuarios.sh
```

A tener en cuenta

```
admon@povedasrv:~$ rpcinfo -p
```

program	vers	proto	port	service
100000	4	tcp	111	portmapper
100000	3	tcp	111	portmapper
100000	2	tcp	111	portmapper
100000	4	udp	111	portmapper
100000	3	udp	111	portmapper
100000	2	udp	111	portmapper
100024	1	udp	39322	status
100024	1	tcp	32931	status
100005	1	udp	57503	mountd
100005	1	tcp	52153	mountd
100005	2	udp	54466	mountd
100005	2	tcp	37557	mountd
100005	3	udp	35947	mountd
100005	3	tcp	43717	mountd
100003	3	tcp	2049	nfs
100003	4	tcp	2049	nfs
100227	3	tcp	2049	nfs_acl
100021	1	udp	49082	nlockmgr
100021	3	udp	49082	nlockmgr
100021	4	udp	49082	nlockmgr
100021	1	tcp	34017	nlockmgr
100021	3	tcp	34017	nlockmgr
100021	4	tcp	34017	nlockmgr

IMPORTANTE:

```
admon@povedasrv:~$ sudo systemctl start ypserv
admon@povedasrv:~$ sudo systemctl enable ypserv
Synchronizing state of ypserv.service with SysV service script with /usr/lib/systemd/systemd-
sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ypserv
Created symlink /etc/systemd/system/multi-user.target.wants/ypserv.service →
/usr/lib/systemd/system/ypserv.service.
```

Editar las bases de datos NIS: NIS comparte ciertas bases de datos (como los usuarios y grupos), por lo que necesitas asegurarte de que estas bases de datos estén disponibles en el servidor. Los archivos de configuración típicos que debes editar son:

- /etc/passwd: Para los usuarios.
- /etc/group: Para los grupos.
- /etc/shadow: Para las contraseñas.
- /etc/hosts: Para la resolución de nombres.

Crear las bases de datos NIS:

Para generar las bases de datos que se compartirán a través de NIS, utiliza el siguiente comando:

```
admon@povedasrv:~$ sudo /usr/lib/yp/ypinit -m
```

At this point, we have to construct a list of the hosts which will run NIS servers. povedasrv.com 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: povedasrv.com

next host to add:

The current list of NIS servers looks like this:

povedasrv.com

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

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

Building /var/yp/povedasrv/ypservers...

Running /var/yp/Makefile...

gmake[1]: Entering directory '/var/yp/povedasrv'

Updating passwd.byname...

Updating passwd.byuid...

Updating group.byname...

Updating group.bygid...

Updating hosts.byname...

Updating hosts.byaddr...

Updating rpc.byname...

Updating rpc.bynumber...

Updating services.byname...

Updating services.byservicename...

Updating netid.byname...

Updating protocols.bynumber...

Updating protocols.byname...

Updating netgroup...

Updating netgroup.byhost...

Updating netgroup.byuser...

Updating shadow.byname...

gmake[1]: Leaving directory '/var/yp/povedasrv'

povedasrv.com has been set up as a NIS master server.

Now you can run ypinit -s povedasrv.com on all slave server.

```
admon@povedasrv:~$
```

```
admon@povedasrv:~$ sudo cat /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_subtree_check)
```

```
#
```

```
# 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 172.16.0.0/16(rw,sync,no_subtree_check,no_root_squash,sec=krb5)
```

```
sudo systemctl restart nfs-kernel-server.service
```

Este comando te pedirá que confirmes el nombre del servidor NIS (el servidor que tiene el servicio NIS activado) y comenzará a crear las bases de datos necesarias.

Verificaciones

1. Verificar el Usuario en NIS

- Buscar al Usuario en el Servidor NIS.

```
ypcat passwd | grep nombre_usuario
```

- UID: Debe coincidir con el UID del directorio en el servidor NFS.
- Ruta del home: Asegúrate de que /home/nombre_usuario esté correctamente configurado.
- Verificar Resolución de Usuarios

```
getent passwd nombre_usuario
```

2. Verificar el Servidor NFS

- Probar el Montaje Manual del Directorio. En el cliente, intenta montar el directorio del usuario directamente desde el servidor NFS para confirmar que el servidor está compartiendo /home correctamente:

```
sudo mount -t nfs nfs_server:/home /mnt
```

Verifica el contenido: `ls /mnt/nombre_usuario`

- Verificar las Exportaciones en el Servidor. En el servidor NFS, verifica que /home está exportado correctamente:

```
sudo exportfs -v
```

Debería mostrar algo como:

```
/home      <clientes>(rw,sync,no_root_squash)
```

Asegúrate de que la IP o rango de los clientes esté incluido en la configuración.

3. Verificar la Configuración de autofs

- Archivos de Configuración

- Verifica /etc/auto.master: `cat /etc/auto.master`.
- Verifica /etc/auto.home: `cat /etc/auto.home`

- Probar el Montaje Automático. Reinicia el servicio autofs en el cliente para asegurarte de que está funcionando:

```
sudo systemctl restart autofs
```

Intenta acceder al directorio del usuario:

```
ls /home/nombre_usuario
```

4. Verificar el Inicio de Sesión del Usuario

- Probar el Inicio de Sesión. Inicia sesión como el usuario:

```
su - nombre_usuario
```

Si todo está configurado correctamente, el directorio /home/nombre_usuario debería estar montado y accesible.

- Verificar el Montaje

```
mount | grep nfs
```

Debería mostrar algo como:

```
nfs_server:/home/nombre_usuario on /home/nombre_usuario type nfs ...
```

5. Solución de Problemas

- Ver Logs de NIS

```
sudo journalctl -u ypbind
```

- Ver Logs de NFS En el cliente, revisa los logs del cliente NFS:

```
sudo journalctl -u nfs-client.target
```

En el servidor, revisa los logs del servidor NFS:

```
sudo journalctl -u nfs-server
```

- Ver Logs de autofs

```
sudo journalctl -u autofs
```

Enlaces de interés

- [nis-nfs-ssh_key-cluster](#)