



# Tecnicatura Universitaria en Software Libre



---

## **Administración de GNU/Linux I**

Unidad 6

Compartir recursos

Autor

Leonardo Martinez

## Contenido

<b>1 Recursos de archivos</b>	<b>3</b>
1.1 Sistemas de archivos distribuidos (DFS - Distributed File Systems)	3
1.1.1 Network File Server: NFS	4
1.1.1.1 El servidor	4
1.1.1.2 El cliente	5
1.1.1.3 Ejemplo de implementación	6
1.1.2 Configuración de espacios compartidos con Samba	8
1.1.2.1 Instalación y configuración del servidor	8
1.1.2.2 Conexión de los clientes	10
<b>2 Bibliografía</b>	<b>15</b>

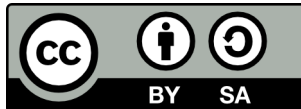
Edición 2022.

**Autor:** Leonardo Martinez

### ¡Copia este texto!

Los textos que componen este trabajo se publican bajo formas de licenciamiento que permiten la copia, la redistribución y la realización de obras derivadas, siempre y cuando éstas se distribuyan bajo las mismas licencias libres y se cite la fuente. El copyright de los textos individuales corresponde a los respectivos autores.

Este trabajo está licenciado bajo un esquema Creative Commons Atribución Compartir Igual (CC-BY-SA) 4.0 Internacional. <http://creativecommons.org/licenses/by-sa/4.0/deed.es>



## 1 Recursos de archivos

En toda organización es necesario el intercambio de información entre las personas que realizan su trabajo diario. Con el avance tecnológico y la aparición de las redes de comunicaciones aparece un nuevo paradigma de conexión entre usuarios que permite compartir sus archivos utilizando estas redes.

### 1.1 Sistemas de archivos distribuidos (DFS - Distributed File Systems)

Un sistema de archivos distribuido almacena archivos en uno o más computadores denominados servidores y los hace accesibles a otros computadores denominados clientes, donde se manipulan como si fueran locales.

Existen muchas ventajas en el uso de servidores de archivos, los archivos están más accesibles si desde varios computadores se puede acceder a los servidores y compartir los archivos de una única ubicación. Es más sencillo que distribuir copias de los archivos a todos los clientes.

Las copias de respaldo y la seguridad son más fáciles de manejar cuando sólo hay que tener en cuenta a los servidores. Los servidores pueden ofrecer un gran espacio de almacenamiento que sería costoso y poco práctico suministrar a cada cliente.

La utilidad de un *sistema de archivos distribuido* se ve claramente cuando se considera a un grupo de empleados que tienen que compartir documentos.

Los *sistemas de archivos distribuidos* basados en *GNU/Linux* son utilizados para centralizar la administración de discos y proveer la facilidad de compartir archivos transparentemente en la red.

Los paquetes de *GNU/Linux* que proveen **DFS** casi siempre incluyen el programa cliente y el programa server. Un servidor **DFS** comparte archivos locales en la red, un cliente **DFS** monta archivos compartidos localmente. Un sistema *GNU/Linux* puede ser tanto un cliente, un server, o ambos dependiendo de la configuración definida.

**NFS** fué desarrollado por *Sun Microsystems* en el 1984 y licenciado a muchos vendedores. Esencialmente permite a servidores hacer disponible su sistema de archivos y directorios, típicamente aplicaciones y los directorios **HOME**, a clientes que los montan como directorios locales.

El cliente entonces ve la información como si fuese local; el montaje debe ser transparente para el cliente. **NFS** elimina la necesidad de instalar aplicaciones y múltiple copias de información idéntica en más de un computador.

**Samba** (*Server Message Block*, or **SMB**) fue desarrollado por *Andrew Tridgell* en 1991. Desarrollado para lograr que su PC, corriendo un *Xserver* en estado beta mediante un protocolo de red basado en *NetBIOS* propiedad de la *DEC*, se pueda comunicar con su estación de trabajo.

Oficialmente el inicio *NetBIOS* para *Unix* es a finales del 1993. **Samba** es al día de hoy incluida en casi todas las distribuciones de *GNU/Linux*.

### 1.1.1 Network File Server: NFS

**NFS** es un protocolo que permite acceso remoto a un sistema de archivos a través de la red. Todos los sistemas *Unix* pueden trabajar con este protocolo.

Es una herramienta muy útil, pero se debe tener en cuenta sus limitaciones, especialmente en cuestiones de seguridad ya que todos los datos pasan a través de la red sin cifrar permitiendo que un sniffer pueda interceptarlos.

El servidor fuerza restricciones de acceso basado en la dirección *IP* del cliente que puede ser falsificada; y, finalmente, cuando se provee acceso a una máquina cliente a un espacio **NFS** compartido mal configurado, el usuario `root` del cliente puede acceder a todos los archivos en el espacio compartido, aún aquellos que pertenezcan a otros usuarios, ya que el servidor confía en el nombre de usuario que recibe del cliente, siendo esta es una limitación histórica del protocolo.

La mayoría de estos problemas fueron subsanados en la versión 4 del protocolo. Pero esto trajo una nueva complejidad, el servidor **NFS** es más difícil de configurar cuando se quiere utilizar las características básicas como autenticación o cifrado ya que están basadas en *Kerberos*.

Y sin esas características, el uso del protocolo **NFS** se tiene que acotar a una red local de confianza ya que la información se transmite por la red sin cifrar y los permisos de acceso se otorgan en base a la dirección *IP* del cliente.

#### 1.1.1.1 El servidor

El servidor **NFS** es parte del núcleo *Linux*; en los núcleos que *Debian* provee está compilado como un módulo de núcleo. Para ejecutar el servidor **NFS** automáticamente al iniciar, se debe instalar el paquete `nfs-kernel-server` que contiene los *scripts* de inicio relevantes.

El archivo de configuración del servidor **NFS**, `/etc/exports`, enumera los directorios disponibles en la red, *exportados*. Para cada espacio compartido **NFS**, sólo tendrán acceso las máquinas especificadas aunque se puede obtener un control más detallado con unas pocas opciones.

La sintaxis para este archivo es bastante simple:

```
/directorio/a/compartir maquina1(opcion1,opcion2,...) maquina2(...) ...
```

Si se utiliza **NFSv4**, todos los directorios exportados deben ser parte de una misma jerarquía y que el directorio raíz de esa jerarquía se debe exportar e identificar con la opción `fsid=0` o `fsid=root`.

Se puede identificar cada máquina mediante su nombre *DNS* o su dirección *IP*. También puede especificar conjuntos completos de máquinas utilizando una sintaxis como `*.unl.edu.ar` o un rango de direcciones *IP* `192.168.0.0/255.255.255.0` o `192.168.0.0/24`.

```
/srv/compartido *.unl.edu.ar
```

De forma predeterminada, o si utiliza la opción `ro`, los directorios están disponibles sólo para lectura. La opción `rw` permite acceso de lectura/escritura. Los clientes **NFS** típicamente se conectan desde un puerto restringido sólo a `root` menor a `1024`; puede eliminar esta restricción con la opción `insecure`, la opción `secure` es implícita, pero se puede hacer explícita para más claridad.

También de forma predeterminada, el servidor sólo responde consultas **NFS** cuando se complete la operación actual de disco, la opción `sync`; puede desactivarse esto con la opción `async`.

Las escrituras asíncronicas aumentarán un poco el rendimiento pero disminuirán la fiabilidad debido al riesgo de pérdida de datos en caso de un cierre inesperado del servidor entre que recibió el pedido de escritura y los datos sean escritos realmente en el disco.

Para no proveerle acceso de `root` al sistema de archivos a ningún cliente **NFS**, el servidor considerará todas las consultas que parezcan provenir de un usuario que se identifique `root` como si provinieran del usuario `nobody`. Este comportamiento corresponde a la opción `root_squash` y está activado de forma predeterminada. La opción `no_root_squash`, que desactiva este comportamiento, es riesgosa y sólo debe ser utilizada en entornos controlados. Las opciones `anonuid=uid` y `anongid=gid` permiten especificar otro usuario para ser utilizado en lugar de `UID/GID 65534` que corresponde al usuario `nobody` y el grupo `nogroup`.

Con **NFSv4** se puede agregar la opción `sec` para indicar el nivel de seguridad que se desea configurar:

- `sec=sys` es el valor por defecto y no posee características de seguridad especiales.
- `sec=krb5` habilita sólo la autenticación.
- `sec=krb5i` agrega protección de integridad.
- `sec=krb5p` es el nivel más completo que incluye protección de privacidad con el uso de cifrado.

Para que los tres niveles `krb5` funciones se debe disponer de un servicio *Kerberos* funcionando en la red.

Existen otras opciones disponibles que están documentadas en la página de manual `exports`.

#### 1.1.1.2 El cliente

Como con cualquier otro sistema de archivos, incorporar un espacio compartido **NFS** en el jerarquía del sistema es necesario montarlo. Debido a que este sistema de archivos tiene sus peculiaridades son necesarios unos pocos ajustes en la sintaxis del comando `mount` y en el archivo `/etc/fstab`.

Montaje manual con el comando `mount`

```
# mount -t nfs -o rw,nosuid nfserver.unl.edu.ar:/srv/compartido /nfsunl
```

Elemento **NFS** en el archivo `/etc/fstab`

```
nfserver.unl.edu.ar:/srv/compartido /nfsunl nfs rw,nosuid 0 0
```

El elemento descrito monta, al iniciar el sistema, el directorio **NFS** `/srv/compartido/` en el servidor **nfserver** en el directorio local `/nfsunl/`. Se configura con acceso de lectura-escritura usando el parámetro `rw`. La opción `nosuid` es una medida de protección que elimina cualquier *bit setuid* o *setgid* de los programas almacenados en el espacio compartido. Si el espacio compartido **NFS** está destinado sólo a almacenar documentos, también se recomienda utilizar la opción `noexec` que evita la ejecución de programas almacenados en el espacio compartido.

La página de manual `nfs` describe todas las opciones con más detalle.

#### 1.1.1.3 Ejemplo de implementación

En la máquina virtual de *Xubuntu* que se usó como ejemplo en el desarrollo de la materia se configura el servidor de **NFS**. El primer paso es preparar el directorio que se *exportará*.

```
$ sudo mkdir /srv/compartido
$ sudo chmod 777 /srv/compartido/
```

Luego se debe editar el archivo de configuración `/etc/exports` para exportar el directorio `/srv/compartido` a la red `192.168.0.0/24`.

```
$ 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)
#
/srv/compartido  192.168.0.0/24(rw,sync,no_subtree_check)
```

Se debe forzar al servicio **NFS** que lea la nueva configuración en el archivo `/etc/exports` y aplique los cambios.

```
$ sudo exportfs -ra
$
```

Luego se crea un archivo dentro del directorio para que se pueda modificar en el equipo cliente.

```
$ echo "Creación de un archivo de texto en NFS" > /srv/compartido/textonfs.txt
```

En el equipo cliente se crea el directorio donde se va a montar el recurso **NFS** exportado por la máquina virtual.

```
leonardo@tarod [~]
-> % sudo mkdir -p /mnt/nfsdir
```

Luego se realiza el montaje del recurso remoto en el directorio creado previa verificación de los recursos compartidos por el servidor.

```
leonardo@tarod [~]
-> % showmount -e 192.168.0.102
Export list for 192.168.0.102:
/srv/compartido 192.168.0.0/24

leonardo@tarod [~]
-> % cd /mnt

leonardo@tarod [/mnt]
-> % sudo mount -t nfs -o rw,nosuid 192.168.0.102:/srv/compartido /mnt/nfsdir

leonardo@tarod [/mnt]
-> % mount | grep 192.168.0.102
192.168.0.102:/srv/compartido on /mnt/nfsdir type nfs4 (rw,nosuid,relatime,vers=
4.0,rsize=131072,wsiz=131072,namlen=255,hard,proto=tcp,port=0,timeo=600,retrans
=2,sec=sys,clientaddr=192.168.0.106,local_lock=none,addr=192.168.0.102)
```

Una vez montado el recurso **NFS** se verifica la existencia el archivo creado localmente en el servidor.

```
leonardo@tarod [/mnt]
-> % cd nfsdir

leonardo@tarod [/mnt/nfsdir]
-> % l
total 12K
drwxrwxrwx 2 root      root      4,0K sep 22 20:17 .
drwxr-xr-x 3 root      root      4,0K sep 22 19:52 ..
-rw-rw-r-- 1 leonardo leonardo   40 sep 22 20:17 textonfs.txt
```

Se realiza una modificación al archivo.

```
leonardo@tarod [/mnt/nfsdir]
-> % echo "Actualización de un archivo de texto en NFS" >> textonfs.txt

leonardo@tarod [/mnt/nfsdir]
-> % ll
total 4,0K
-rw-rw-r-- 1 leonardo leonardo 85 sep 22 20:19 textonfs.txt

leonardo@tarod [/mnt/nfsdir]
```



```
-> % cat textonfs.txt
Creación de un archivo de texto en NFS
Actualización de un archivo de texto en NFS

leonardo@tarod [/mnt/nfsdir]
-> %
```

### 1.1.2 Configuración de espacios compartidos con Samba

**Samba** <sup>1</sup> es un conjunto de programas originalmente creados por *Andrew Tridgell* y actualmente mantenidos por *The SAMBA Team*, que administran el protocolo *SMB*, también llamado *CIFS*) en *GNU/Linux*. *Windows* utiliza este protocolo para compartir recursos de archivos e impresoras por la red.

También puede configurarse para trabajar como un *controlador de dominio Windows*. Esta es una herramienta sobresaliente para asegurar una integración perfecta entre servidores *GNU/Linux* y las máquinas de escritorios en las oficinas que todavía utilizan *Windows*.

*SMB*, acrónimo de *Server Message Block* es un protocolo, del *Nivel de Presentación* del modelo *OSI* de *TCP/IP*, creado en 1985 por *IBM*. Algunas veces es referido también como *CIFS*, acrónimo de *Common Internet File System* <sup>2</sup>, tras ser renombrado por *Microsoft* en 1998.

Entre otras cosas, *Microsoft* añadió al protocolo soporte para enlaces simbólicos y duros así como también soporte para archivos de gran tamaño.

#### 1.1.2.1 Instalación y configuración del servidor

Para poder compartir directorios e impresoras de un sistema *GNU/Linux* con el protocolo *SMB* en un entorno de red heterogéneo se debe instalar el paquete de **Samba** en el equipo que expondrá los recursos.

```
$ sudo apt update
$ sudo apt -y install samba
```

Hay varias opciones para realizar la configuración de **Samba**, varias son aplicaciones gráficas, sin embargo, para conocer mejor el software se puede trabajar desde la línea de comando.

El archivo de configuración es el archivo de texto `/etc/samba/smb.conf` que se puede editar con cualquier programa de edición, tanto en la consola como de interfaz gráfica. Como todo archivo de configuración, requiere permisos de administrador para poder editarlo.

**Samba** es un conjunto de paquetes de software maduro y muy complejo, por lo que su configuración puede llevar tiempo y ser engorrosa. Se puede confiar en la configuración por defecto que provee el paquete de instalación y concentrarse en cambiar las líneas necesarias para el servicio que se pretende ofrecer. Para más información sobre los parámetros de configuración se puede consultar la página del manual de `smb.conf`.

Las líneas que comienzan con los caracteres `;` y `#` son comentarios y no se reconocen como valores de configuración activos. Para que la línea sea efectiva hay que quitar ese carácter.

La configuración por defecto del software en la mayoría de las distribuciones *GNU/Linux* viene preparada para que con sólo agregar la definición de los recursos de archivos que se quieren compartir es suficiente para tenerlos disponibles.

En el caso de impresoras, la configuración por defecto comparte automáticamente todas las impresoras definidas en el servicio de impresión **CUPS**.

Para definir un recurso público en el cual los usuarios no necesiten identificarse, se puede utilizar la siguiente plantilla:

```
[Recurso Público]
path = /ubicacion/del/directorio
guest ok = yes
read only = no
browseable = yes
```

La expresión **Recurso Público** es el nombre que se expondrá a los clientes que accedan al recurso. La línea `guest ok = yes` es la que especifica que el recurso es de acceso público. El parámetro `browseable = yes` muestra el recurso como visible a todos los usuarios, el valor `no` obliga al usuario a escribir el nombre manualmente al definir el acceso en el cliente. Y por supuesto, se debe configurar el parámetro `read only = yes` para restringir que los usuarios realicen cambios en el recurso.

La plantilla para los recursos privados, donde el usuario debe identificarse incluye la definición del parámetro `valid users = [lista de usuarios separada por espacios]` que explicita los usuarios con acceso permitido al recurso.

```
[Recurso Privado]
path = /ubicacion/del/directorio
valid users = user1 user2 userN
read only = no
browseable = yes
```

Los usuarios tienen que estar definidos en la computadora que comparte el recurso.

Con estas dos plantillas es posible configurar recursos compartido en la máquina virtual que se viene utilizando en el curso. Para ello editamos el archivo de configuración de **Samba** `/etc/samba/smb.conf` y se agrega al final lo siguiente:

```
# Configuraciones personales

[Recurso Público]
path = /srv/samba/publico
guest ok = yes
read only = no
browseable = yes

[Recurso Privado]
```

```
path = /srv/samba/privado
valid users = administrador
read only = no
browseable = yes
```

Los directorios definidos en esta sección deben estar creados:

```
$ sudo mkdir -p /srv/samba/publico
$ sudo mkdir -p /srv/samba/privado
$ sudo chown administrador:administrador privado/
$ sudo chmod 775 publico/
```

Los usuarios que tienen que identificarse para acceder a los recursos privados se deben que agregar a la base de contraseñas de **Samba** con el comando:

```
$ sudo smbpasswd -a user
```

El comando pedirá una contraseña para ese usuario. Esta contraseña es exclusiva para la gestión de los accesos a los recursos compartidos por **Samba**.

Con todo definido, solo resta reiniciar el servicio **Samba** para que tome los nuevos parámetros.

```
$ sudo invoke-rc.d samba restart
```

Para comprobar que los recursos definidos estén disponibles, en el mismo equipo se puede utilizar el cliente `smbclient` para visualizar lo exportado.

```
$ smbclient -L localhost
WARNING: The "syslog" option is deprecated
Enter administrador's password:
Domain=[WORKGROUP] OS=[Windows 6.1] Server=[Samba 4.3.9-Ubuntu]

      Sharename      Type      Comment
      -----
      print$         Disk      Printer Drivers
      IPC$           IPC       IPC Service (xubuntu1604 server (Samba, Ubuntu))
      Recurso Público Disk
      Recurso Privado Disk
Domain=[WORKGROUP] OS=[Windows 6.1] Server=[Samba 4.3.9-Ubuntu]

      Server          Comment
      -----
      ALBION2
      XUBUNTU1604      xubuntu1604 server (Samba, Ubuntu)

      Workgroup       Master
      -----
      WORKGROUP       ALBION2

$
```

### 1.1.2.2 Conexión de los clientes

Desde otro equipo con *GNU/Linux* se puede verificar lo mismo desde la línea de comando con el comando `smbclient`.

```
leonardo@tarod [~]
-> % smbclient -L 192.168.0.102 -U administrador
Enter administrador's password:
Domain=[WORKGROUP] OS=[Windows 6.1] Server=[Samba 4.3.9-Ubuntu]

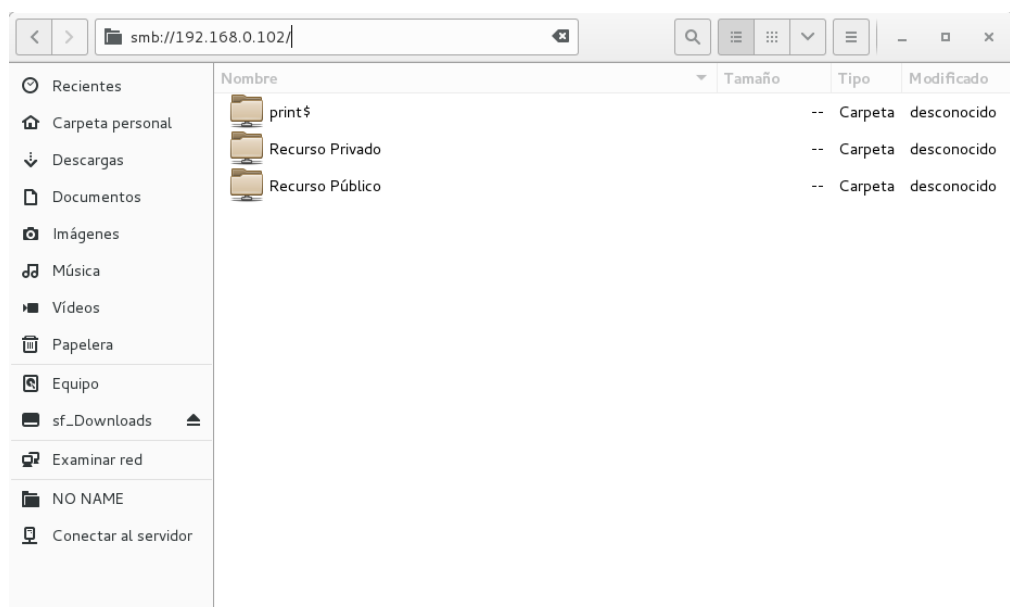
      Sharename      Type      Comment
      -----
      print$         Disk      Printer Drivers
      IPC$           IPC       IPC Service (xubuntu1604 server (Samba, Ubuntu))
      Recurso Público Disk
      Recurso Privado Disk
Domain=[WORKGROUP] OS=[Windows 6.1] Server=[Samba 4.3.9-Ubuntu]

      Server          Comment
      -----
      ALBION2
      XUBUNTU1604      xubuntu1604 server (Samba, Ubuntu)

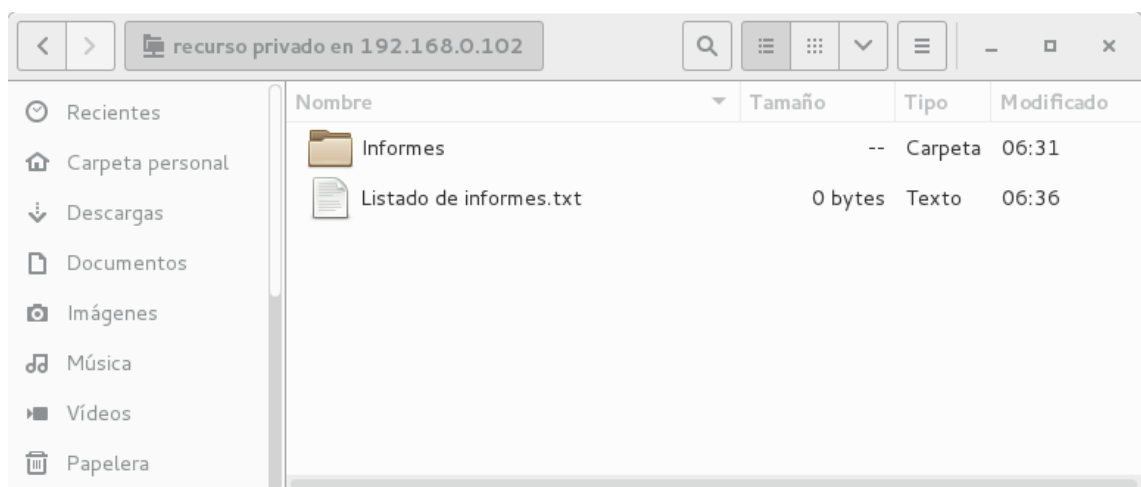
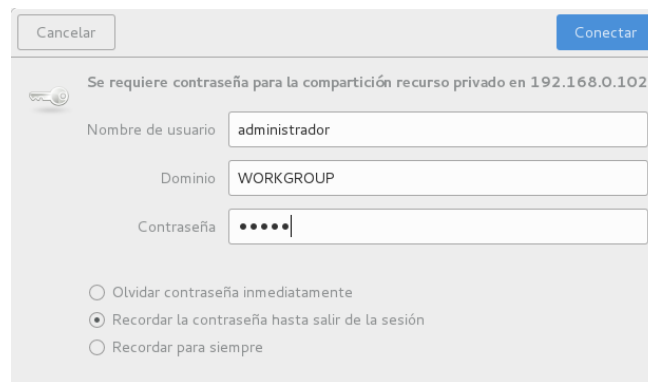
      Workgroup       Master
      -----
      WORKGROUP       ALBION2

leonardo@tarod [~]
-> %
```

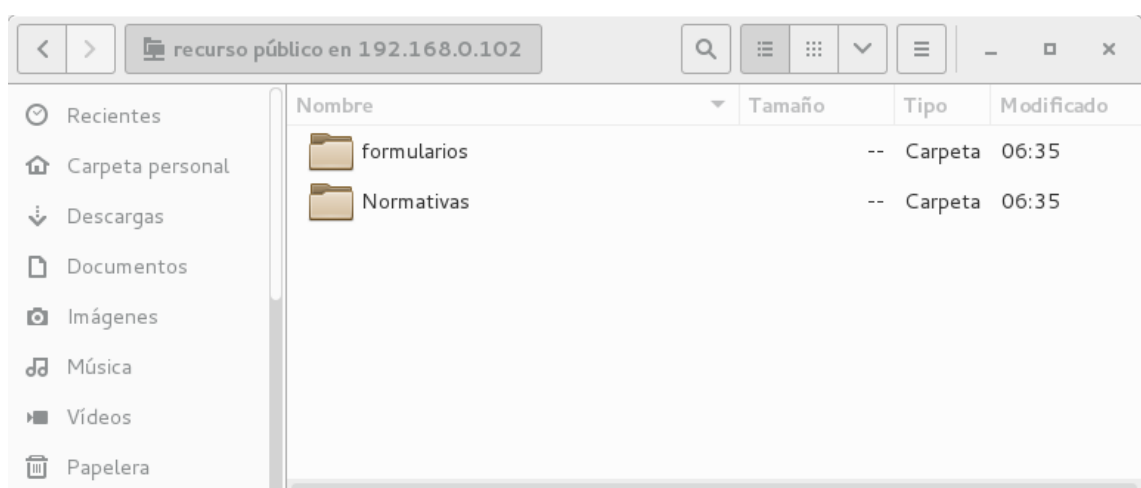
Y también desde el entorno gráfico.



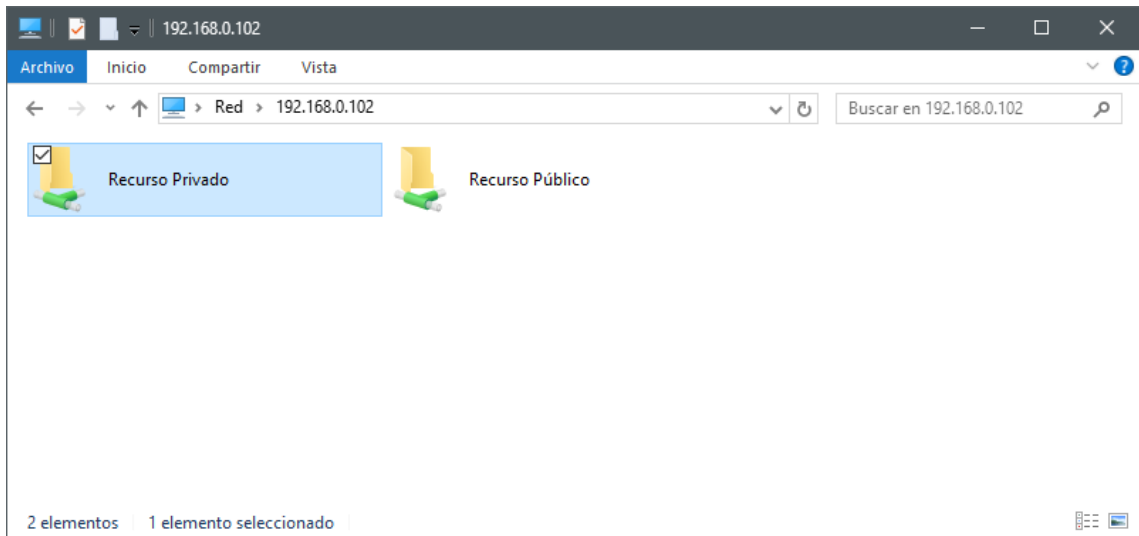
Al acceder al recurso privado se debe ingresar usuario y contraseña.



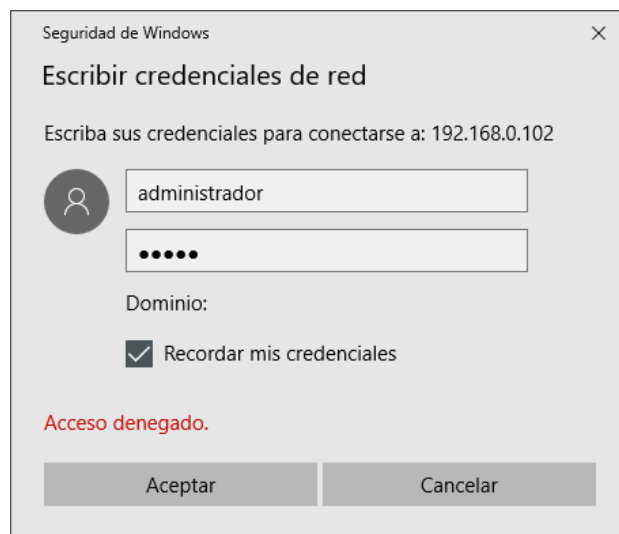
En el caso del recurso público, no es necesario la autenticación.

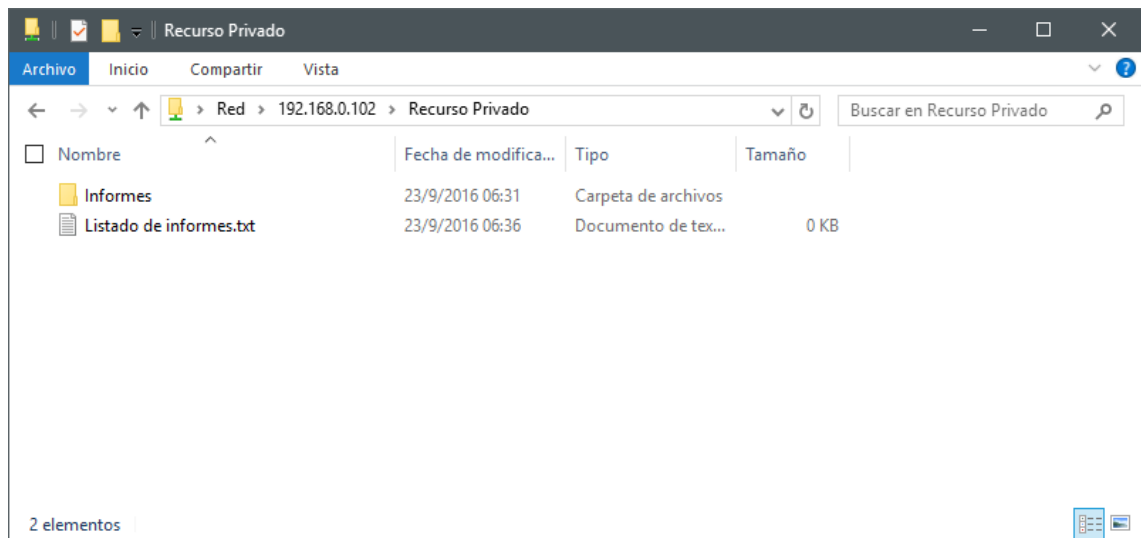


También desde un equipo con *Windows*.

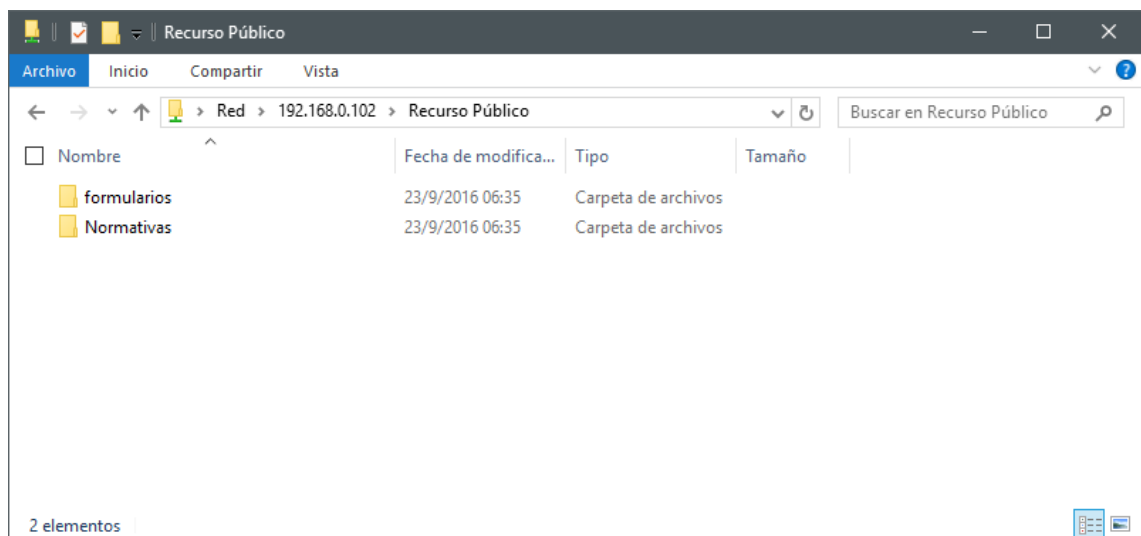


Ingresando los datos de autenticación al acceder al recurso privado.





Y el acceso al recurso público sin necesidad de autenticar con el usuario.



## 2 Bibliografía

- Páginas de Manual (Man Pages) del sistema operativo.

La bibliografía que se indica a continuación corresponde a material que provee una visión interesante sobre los temas propuestos en esta unidad. Es documentación más completa e incluso más extensiva en el desarrollo de algunos temas.

- The Debian Administrator's Handbook, Raphaël Hertzog and Roland Mas, (<https://debian-handbook.info/>)
- [Administración de sistemas GNU/Linux](#), Máster universitario en Software Libre, Universitat Oberta de Catalunya
- Administración de sistemas GNU/Linux, Antonio Perpiñan, Fundación Código Libre Dominicano (<http://www.codigolibre.org>)

---

1	<a href="http://samba.org/">http://samba.org/</a>
2	<a href="http://samba.org/cifs/">http://samba.org/cifs/</a>