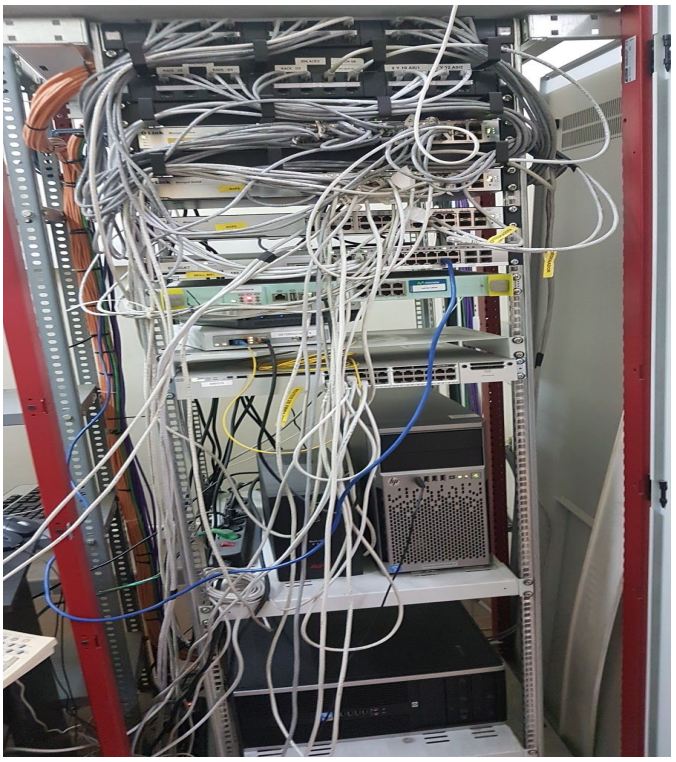
Institut Puig Castellar
Santa Coloma de Gramenet

Mejora sobre la infraestructura de red del centro

CFGM Sistemes microinformàtics i xarxes



➔ **Adrian Garcia Belmonte**
➔ **Pratik Kumar Patel**

Curso 2017/2018

ÍNDICE

1. Introducción

Este proyecto tratará del estudio y mejora de la infraestructura de la red, realizando un estudio sobre sus servicios y aplicaciones para después sugerir mejoras.

Nos hemos propuesto revisar los servicios y aplicaciones del centro, para ello necesitaremos máquinas virtuales para instalar cada uno de los servicios del centro para simular pruebas, información y esquemas sobre cómo está montado en el centro y así poder aplicar las mejoras más eficientemente. Finalmente una vez acabado este trabajo, se presentará en el tribunal.

1.1 Objetivos

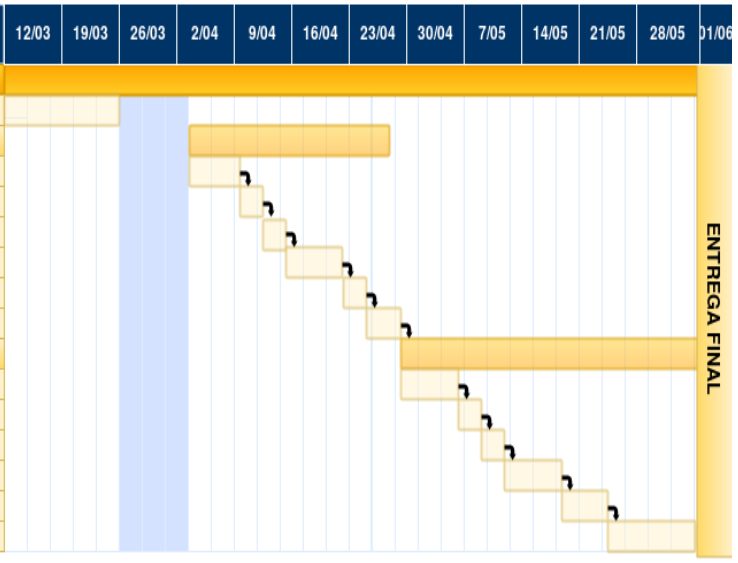
1. Analizar y documentar la infraestructura del centro.
2. Copia de seguridad automatizada en Google Drive
3. Administracion y configuracion de las máquinas de aula mediante Veyon
4. Migración de Owncloud a Nextcloud
5. Creación del servidor torrent integrado en Nextcloud

1.2. Anàlisis de requerimientos

1. Información sobre la arquitectura de la red.
2. Información sobre los requerimientos de los programas
3. Información de la configuración de los servicios.
4. Contenedores / Software para virtualizar. (Virtualbox)
5. Software para documentos y diagramas. (Google Drive)

1.3. Planificación de tareas por semana:

N	Tareas	Duración	Inicio	Finalizado
1	**Mejora sobre la Infraestructura de red del centro**	51H	13.03.18	30.05.18
2	Planificación y memoria del proyecto	10H	13.03.18	21.03.18
3	**Tareas del mes de abril**	17H	3.04.18	27.04.18
4	Estudio sobre la arquitectura de la red	4H	3.04.18	6.04.18
5	Estudio sobre el almacenamiento	2H	10.04.18	10.04.18
6	Implementacion de google-drive-ocamlfuse	2H	11.04.18	13.04.18
7	Mejora sobre la arquitectura de la red	4H	17.04.18	20.04.18
8	Estudio sobre el funcionamiento de Veyon	2H	24.04.18	24.04.18
9	Investigación de la migración de Owncloud/Nextcloud	3H	25.04.18	27.04.18
10	**Tareas del mes de mayo**	24H	1.05.18	30.05.18
11	Implementación de Veyon	5H	1.05.18	4.05.18
12	Estudio del funcionamiento Nextcloud	1H	8.05.18	8.05.18
13	Estudio del programa transmission	2H	8.05.18	9.05.18
14	Implementación de la migración de Owncloud/Nextcloud	5H	10.05.18	16.05.18
15	Creación de la pagina web	5H	17.05.18	23.05.18
16	Preparacion de la presentacion	6H	24.05.18	31.05.18



2. Estudio sobre la infraestructura de red

Antes de empezar a realizar cualquier cambio es conveniente empezar con un estudio sobre lo que hay implementado actualmente, este capítulo consistirá en el análisis y documentación de cómo está actualmente estructurada la red.

2.1 Estructura de la LAN

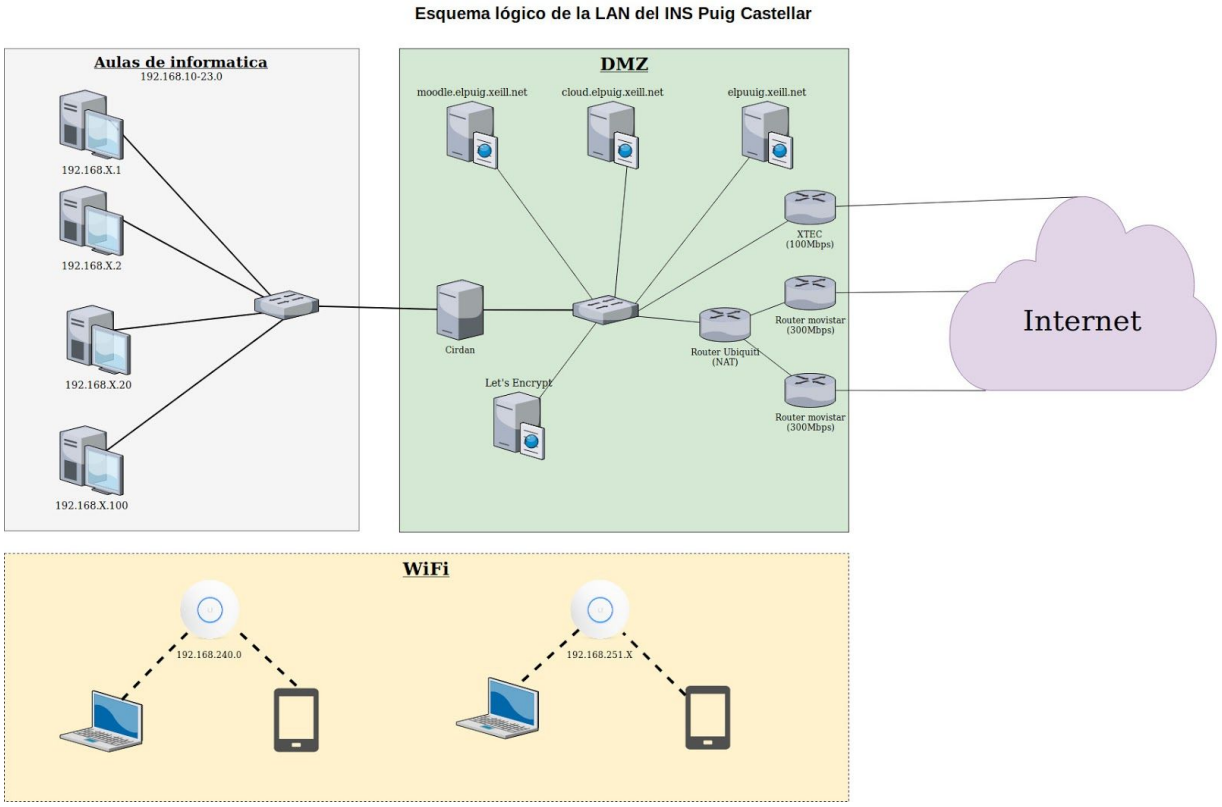


Figura 1: Esquema lógico de la LAN del INS Puig Castellar

La LAN del centro cuenta con el siguiente material siguiente:

- Cableado estructurado de 1 Gbps.
- Puntos de acceso WiFi Ubiquiti.
- Participación en la XEiLL.
- Diversas aulas de informática.
- Diversos ordenadores repartidos en departamentos, seminarios y lugares comunes del centro.
- La máquina física **Rohan,** que cuenta con el siguiente hardware:
    - 2 discos SATA de 4TB Wd.
    - 2 discos SATA SSD de 250GB Samsung 850
    - 32 GiB de RAM ECC Unbuffered 1600MHz (4 módulos Samsung PC3L-12800E).
    - Intel(R) Xeon(R) CPU E3-1240 v3 @ 3.40GHz. Con Hyper-threading activado el SO ve 8 núcleos.
    - Un par de interfaces de red gigabit ethernet.

- La maquina fisica **Gondolin** que realiza los backups semanalmente de todas las máquinas.
- 1 SAI que mantiene las máquinas encendidas en caso de irse la luz

Figura 2: Rohan (Servidor principal)

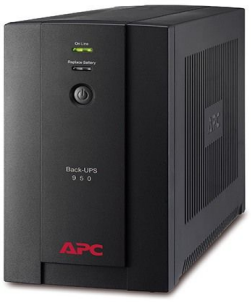


Figura 2: Back-UPS de APC (SAI)

- 1 Router de la XTEC con IP estática (100Mbps simétricos).
- 1 Router de Ubiquiti que hace de NAT.
- 2 Router Movistar de fibra con IP dinámica (Cada uno de 300Mbps simétricos) en modo monopuesto.

Figura 3: Router fibra movistar

Figura 4: Router ubiquiti WiFi

Figura 5: EdgeRouter PoE

2.3 Virtualización

La máquina física **Rohan** ejecuta las siguientes máquinas virtuales (KVM[1]):

- **Cirdan**: Que ofrece la infraestructura de red (DHCP, DNS, NTP, gateway, proxy cache) necesaria para los equipos de cada VLAN.
- **Let's encrypt**: Gestiona los certificados HTTPS y hace de proxy para las máquinas del blog, moodle y otros mediante Apache.
- **Valinor**: Genera gráficas de las máquinas del centro sobre su estado con Munin.
- **Moodle, Blog, Plone, Cloud**: Son las maquinas virtuales que contienen la web del centro, el moodle, el blog, la nube mediante Apache.

2.4 DMZ

La DMZ está ubicada en la subred 192.168.0.0/24 y 192.168.100.0/24 que contienen las máquinas que ofrecen servicios en nuestra red y/o Internet. En esta red se puede encontrar:

- 192.168.0.0/24: Las maquinas **Rohan**, **Gondolin, Valinor, Let's Encrypt.**
- 192.168.100.0/24: Las maquinas **Cirdan, Blog, Moodle, Plone, Cloud.**

2.5 Redes del centro

Las máquinas están distribuidas en distintas redes:

Aula	Direccionamiento utilizado
Torvalds	192.168.10.0/24
Stallman	192.168.11.0/24
Ada	192.168.12.0/24
AIF	192.168.14.0/24
TAC	192.168.15.0/24
ESO/BATX	192.168.16.0/24
Turing	192.168.18.0/24
Darwin	192.168.19.0/24

[1] KVM - Kernel Virtual Machine. Es el hypervisor oficial del núcleo Linux. Más información: https://es.wikipedia.org/wiki/Kernel-based_Virtual_Machine

3. Backups en Google Drive

Actualmente existe un sistema para realizar copias de seguridad de las máquinas virtuales, estos son almacenados semanalmente en la maquina *Gondolin* pero existe un principal inconveniente y es que todas las máquinas se encuentran en el mismo lugar, por eso en el caso de que hubiera un incendio se perdieron todos los datos del centro.

Por eso buscando una solución a este problema, pensamos en la plataforma de Google Drive que ofrece almacenamiento ilimitado a centros escolares. El proyecto google-drive-ocamlfuse ofrece la manera de montar el directorio de una cuenta de google drive en una carpeta de Linux.

3.1 Instalación

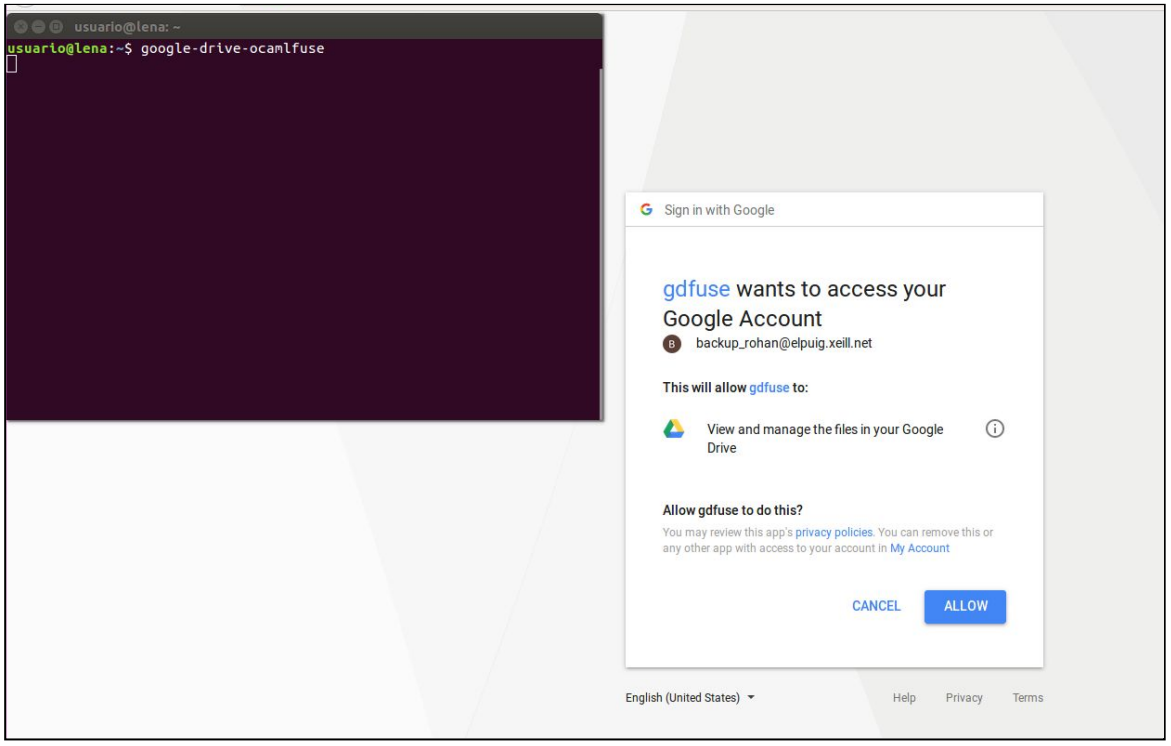
Primero deberemos instalar el programa **google-drive-ocamlfuse** para ello ejecutamos las siguientes órdenes:

Maquina servidor

```
sudo add-apt-repository ppa:alessandro-strada/ppa
sudo apt-get update
sudo apt-get install google-drive-ocamlfuse
```

3.2 Configuración

Una vez que está instalado ejecutaremos el programa con la orden ***google-drive-ocamlfuse***, este nos abrirá una pestaña en el navegador que tendremos que iniciar sesión para aceptar los permisos de la aplicación.



Una vez dados los permisos esperaremos a que el programa acepte el **Token** que nos pide

Cuando el programa finalice copiaremos la carpeta oculta de nuestro directorio personal llamada **.gdfuse/** para después moverla al servidor ya que este no tiene entorno grafico y no podríamos dar los permisos desde allí.

```
Maquina servidor

scp .gdfuse usuario@IP
```

Después de copiar la carpeta **.gdfuse**, crearemos la carpeta donde montaremos nuestro google drive y lo montamos con la siguiente orden:

```
Maquina servidor

google-drive-ocamlfuse ~/google-drive
```

Una vez montada podremos comprobar que el directorio ya se ha montado.

```
usuario@soyuz:~$ df -h
Filesystem              Size  Used Avail Use% Mounted on
udev                    467M     0  467M   0% /dev
tmpfs                    99M  712K   98M   1% /run
/dev/sda2               9.8G  4.3G  5.1G  46% /
tmpfs                   493M     0  493M   0% /dev/shm
tmpfs                   5.0M     0  5.0M   0% /run/lock
tmpfs                   493M     0  493M   0% /sys/fs/cgroup
tmpfs                    99M     0   99M   0% /run/user/1000
google-drive-ocamlfuse  8.0E  400M  8.0E   1% /home/usuario/google-drive
usuario@soyuz:~$ _
```

Como podemos ver, el directorio tiene un tamaño de 8 Exabytes suficiente para subir los backups de Rohan, el fichero no deberá ser más grande de 5 TB (El máximo permitido por google drive).

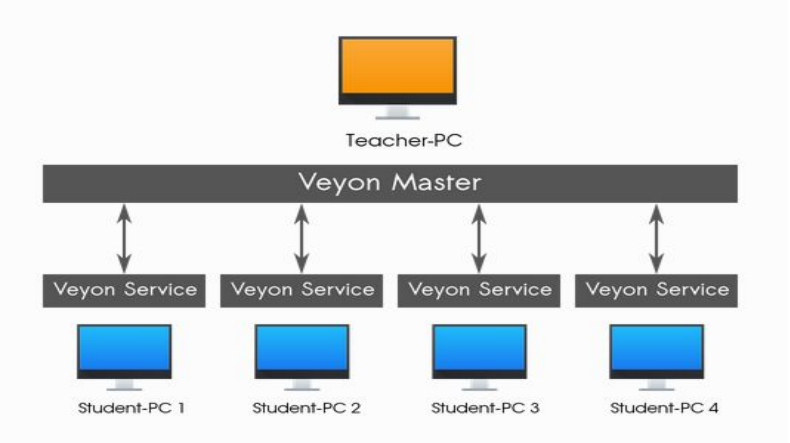
```
google-drive-ocamlfuse  8.0E  400M  8.0E   1% /home/usuario/google-drive
```

4. Administración remota mediante Veyon

Veyon es un software de código abierto para monitorización de computadoras y administración de salas del aula. Permite la observación y el control de las salas de ordenadores, así como la interacción con el usuario. Las funciones principales de Veyon son las siguientes:

- Descripción general de una sala (clase) con todos los contenidos de pantalla en una vista de mosaico.
- Control remoto de computadoras.
- Reflejo de la vista del profesor a todas las demás computadoras en tiempo real (pantalla completa / ventana).
- Bloqueo de las máquinas para obtener la atención de los estudiantes.
- Envío de mensajes de texto a los estudiantes.
- Arranque remoto o apagado de computadoras.
- Desconexión de usuarios.
- Ejecución de programas o apertura de sitios web.

Una diagrama sobre cómo sería la estructura:



4.1 Instalación & Configuración

Veyon se puede obtener desde su página oficial o desde su repositorio en GitHub, tanto la máquina del profesor como la de los alumnos requerirán el paquete principal de Veyon y los siguientes pasos aplicados:

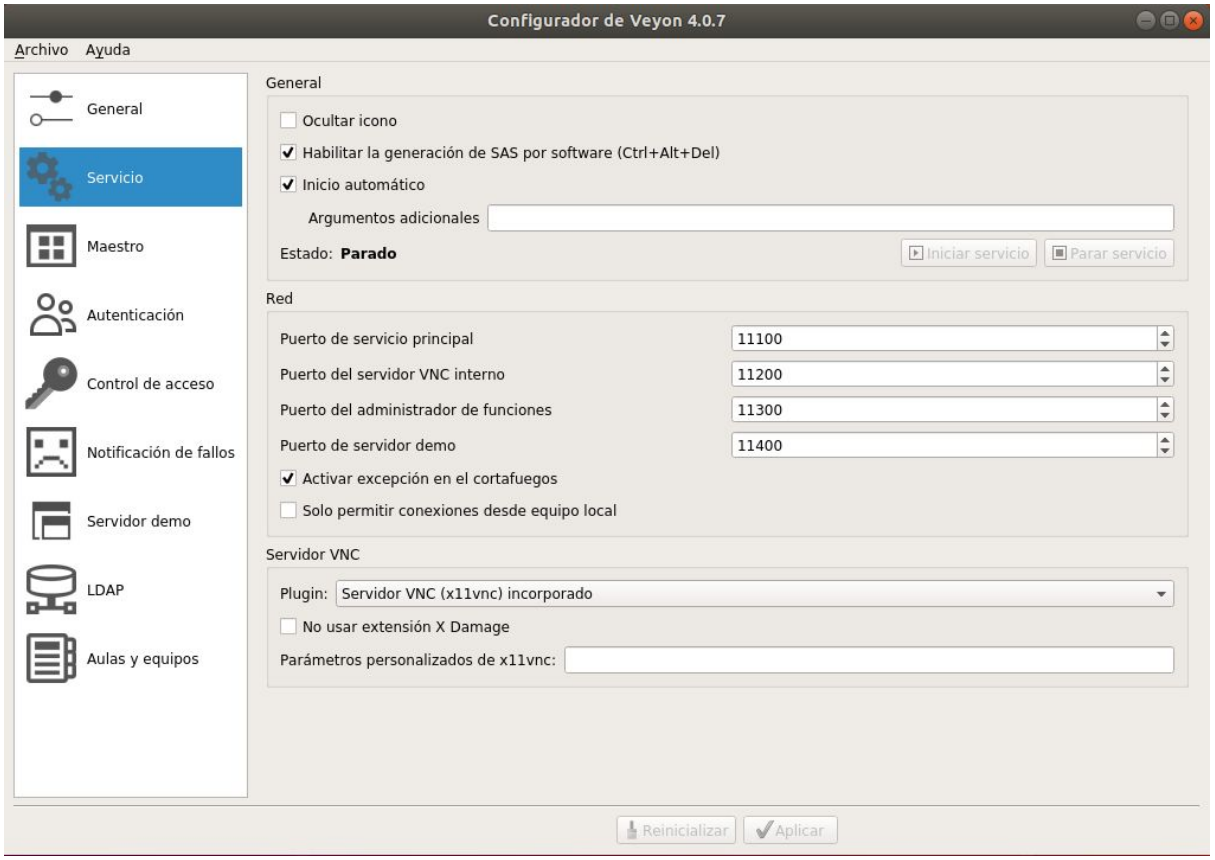
Instalamos el paquete usando la herramienta **apt**.

```
$ sudo apt ./veyon_4.0.7-ubuntu-artful_amd64.deb
```

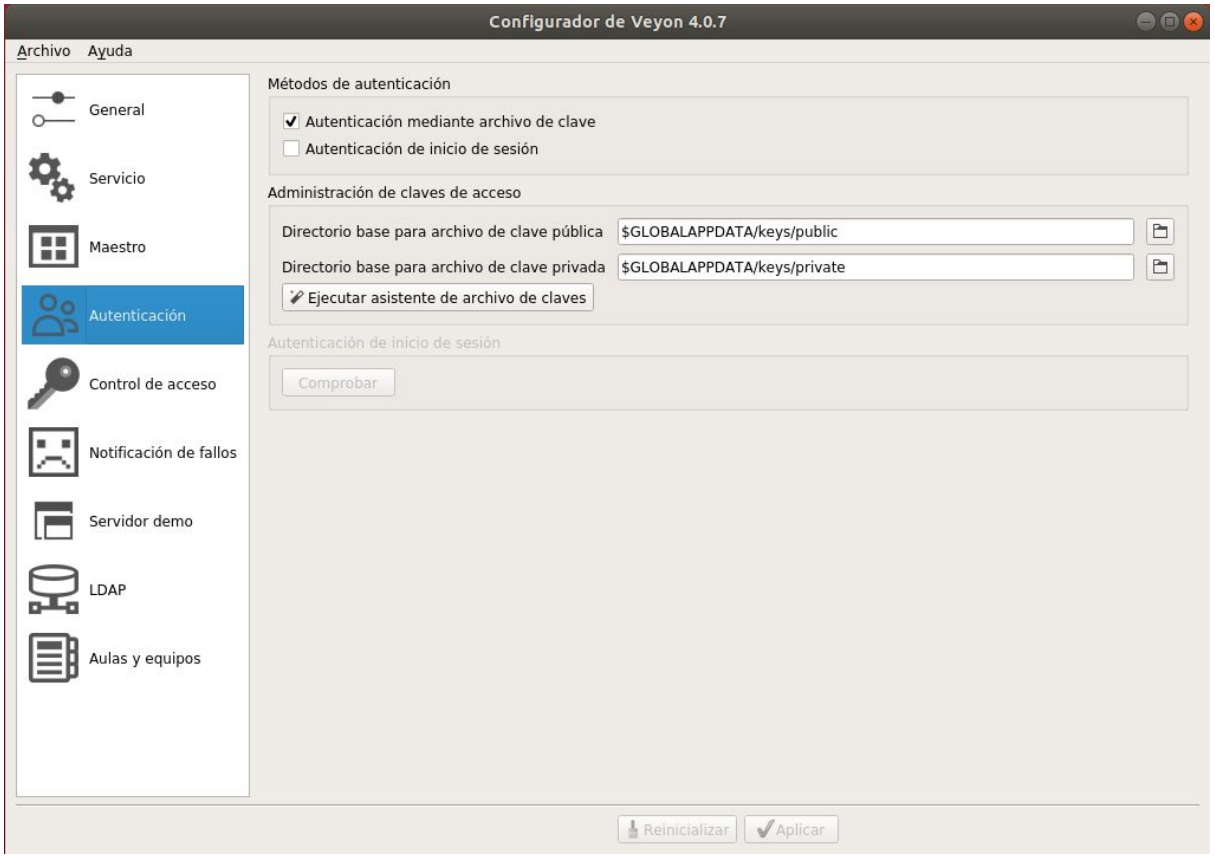
Para configurar las máquinas ejecutaremos la orden del configurador de veyon.

```
$  sudo veyon-configurator &
```

Dentro del configurador primero iremos a *Servicio -> Servidor VNC* y seleccionamos el que dice **Servidor VNC (x11vnc) incorporado**.



Veyon ofrece dos métodos de autenticación, **mediante un archivo de clave** o de **inicio de session,** marcaremos la casilla de **mediante archivo de clave**, ejecutamos el botón de **Aplicar los cambios** y después ejecutaremos el **asistente de archivo de claves** para crear el par de claves.



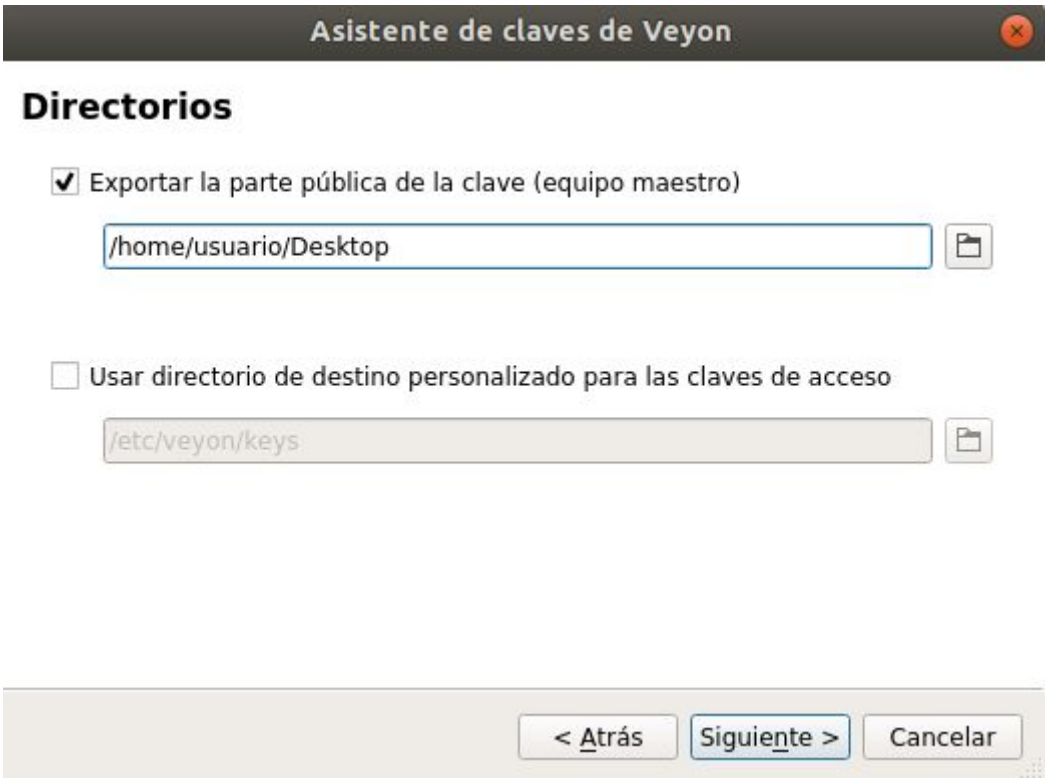
4.1.1 Máquina Maestra (Creación de claves)

A partir de este punto se aplicará a la máquina maestra dentro del asistente nos explicara el funcionamiento de las claves, una vez leído le damos a siguiente.

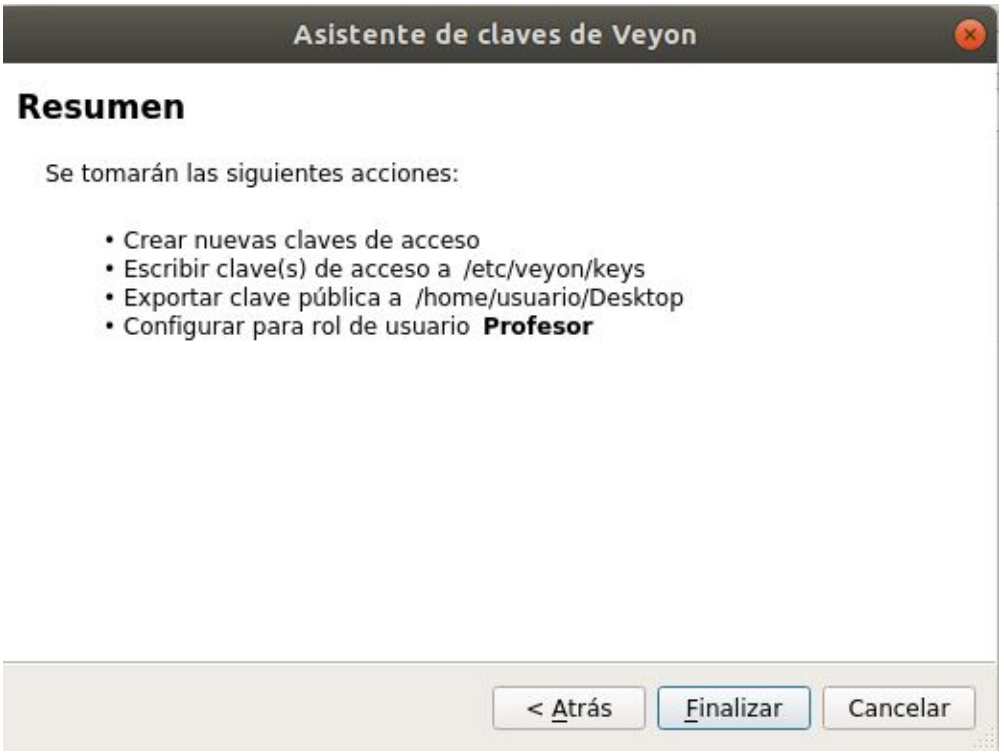
Elegiremos crear un nuevo par de claves de acceso.

Después seleccionamos el rol **profesor**.

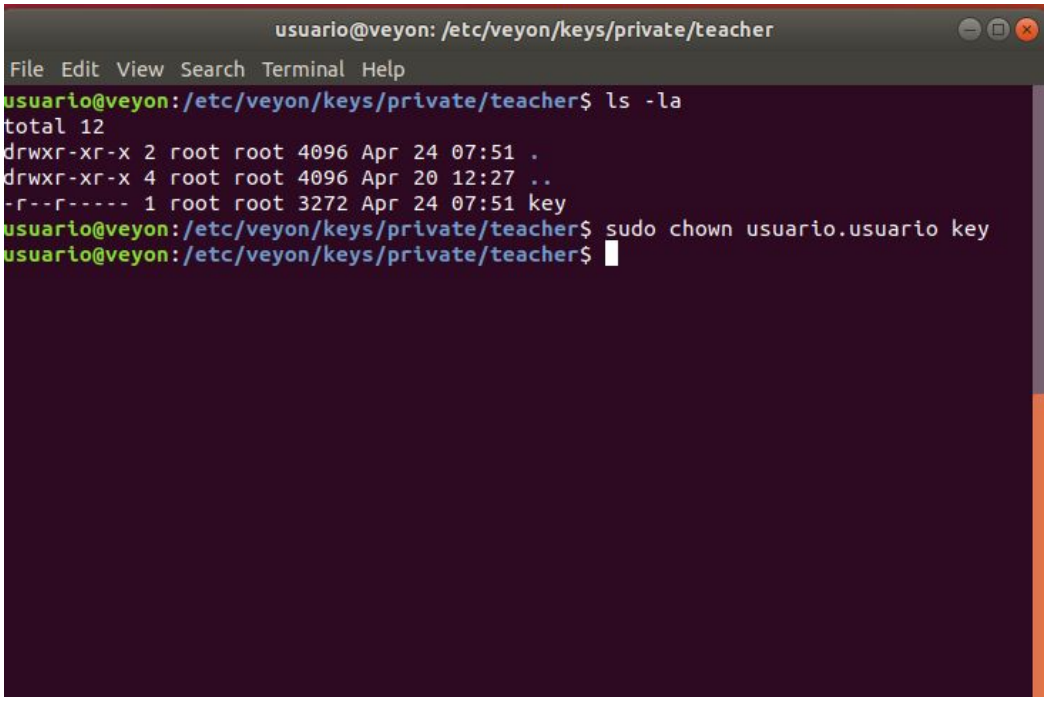
Marcamos la casilla de exportar la clave pública para después importarla en los equipos clientes, le especificaremos el directorio donde será exportada, si no hubiera ninguno se exportan por defecto en el **/etc/veyon/keys**.



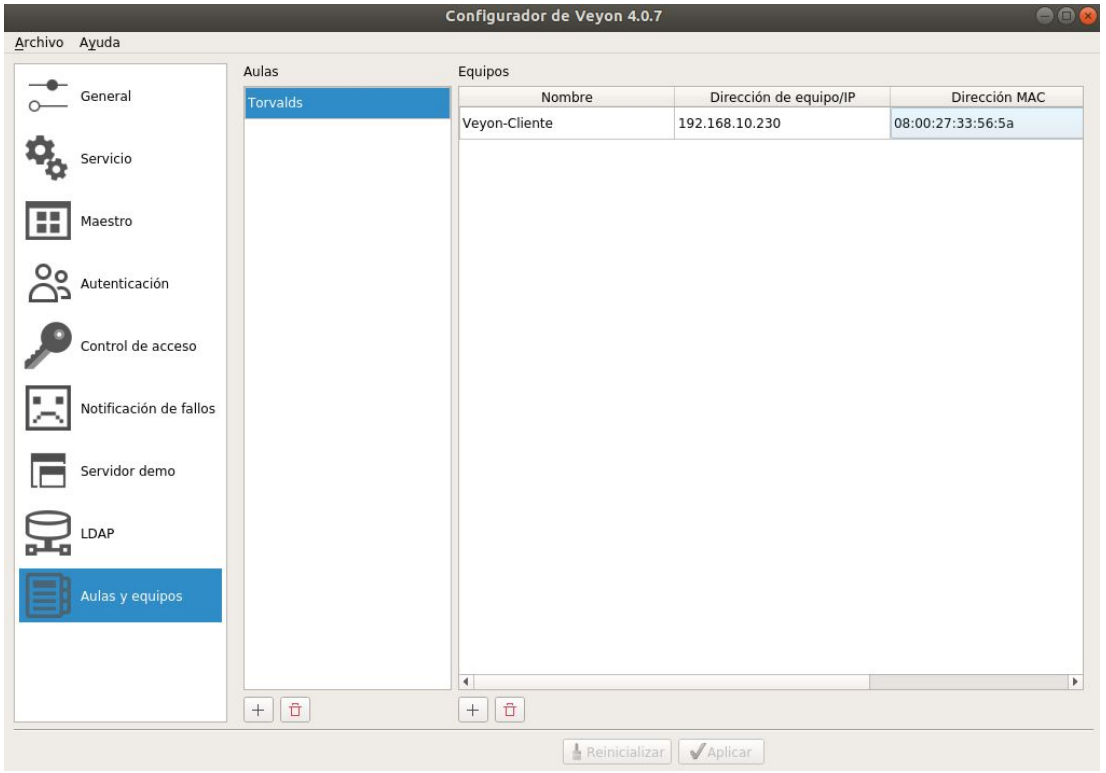
Por último nos dará un resumen de las acciones que hemos tomado y le damos a **finalizar**.



Una vez creadas las claves iremos al directorio **/etc/keys/private/teacher** y aplicamos los **permisos de usuario**, esto sucede debido a que el configurador requiere privilegios sudo causando así que la clave privada se cree con permisos de root y despues no se pueda acceder a máquinas cliente ya que no se tiene acceso a dicha clave.



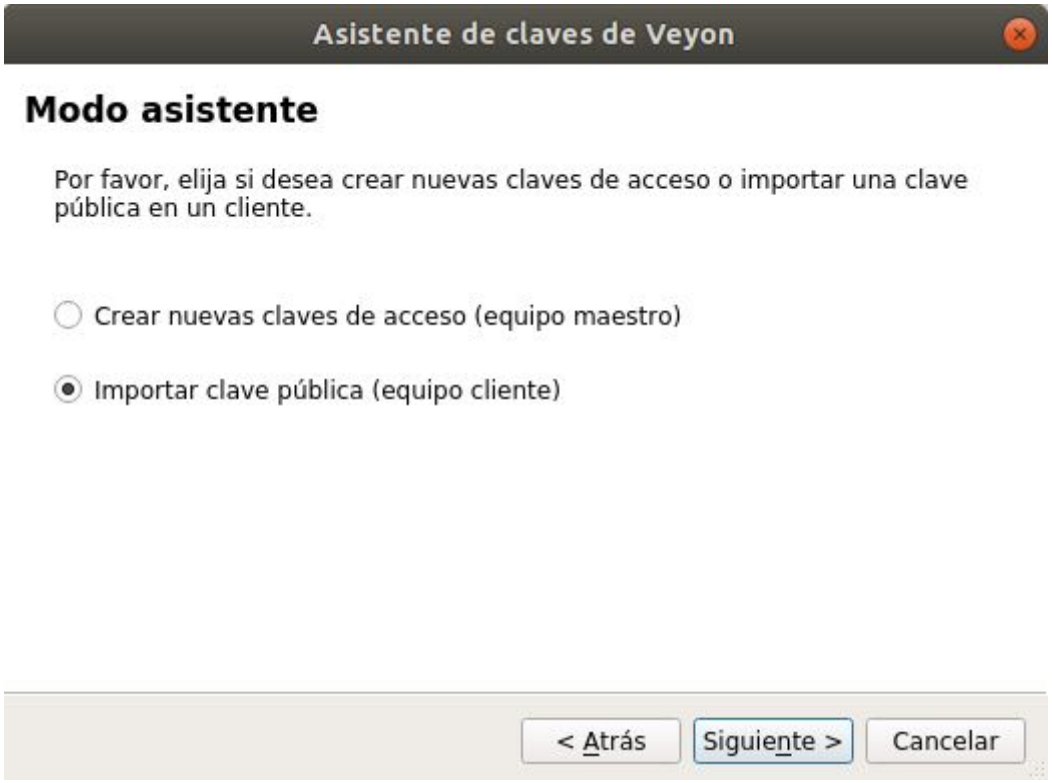
Nuevamente dentro del configurador pulsaremos sobre la sección de **Aulas y equipos**, en esta sección pondremos las aulas y los equipos que controlemos para que luego aparezcan en el programa de Veyon maestro, será necesario por lo menos el nombre de dicho equipo y su dirección IP.



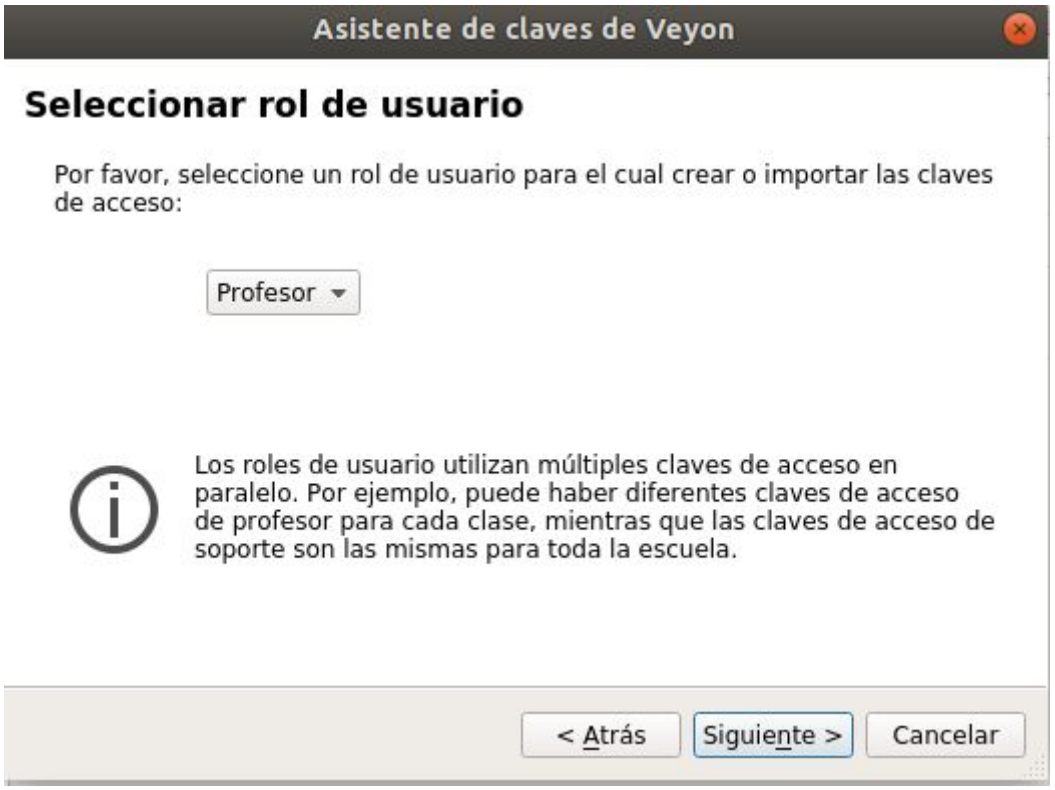
4.1.2 Máquina Cliente (Importación de clave pública)

A partir de este punto se aplicará a la máquina cliente dentro del asistente nos volverá a explicar el funcionamiento de las claves, le damos a siguiente. Deberemos tener también la clave pública previamente movida a los equipos clientes.

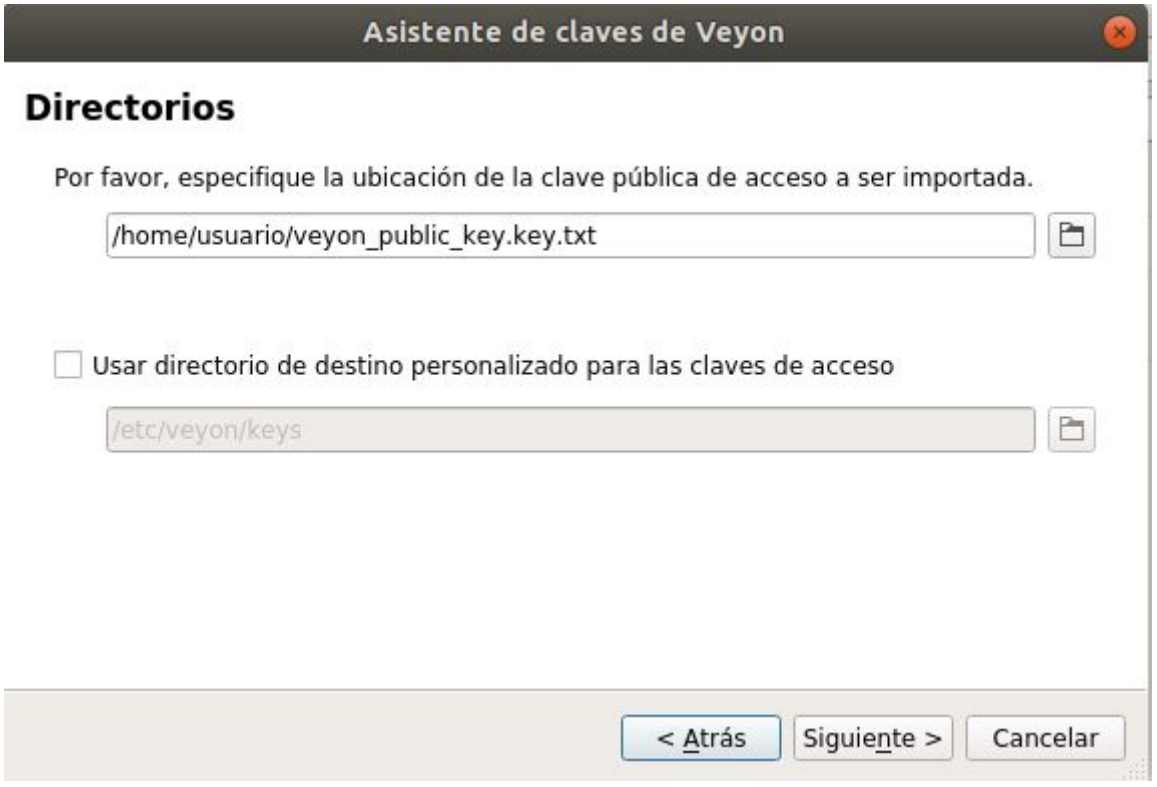
Seleccionamos la importación de la clave.



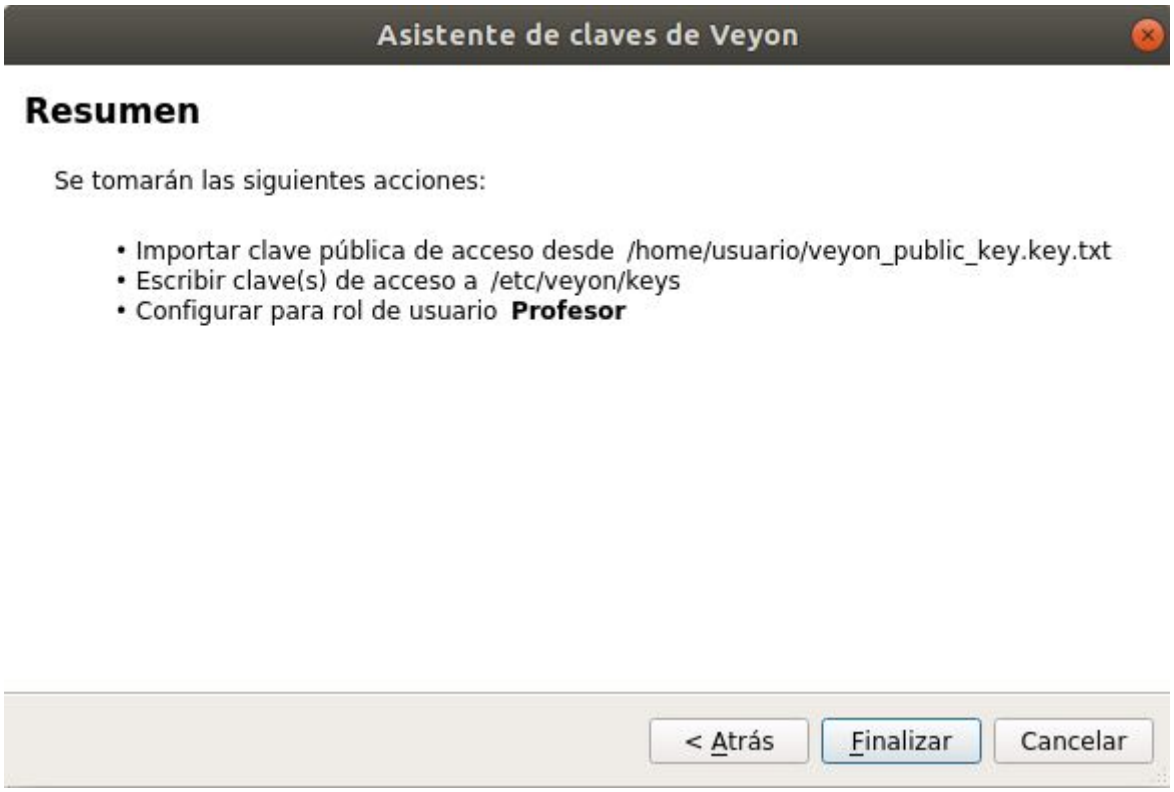
Después le diremos que esta clave pertenecera al rol de profesor.



Seleccionamos la clave pública previamente copiada de la máquina maestra.

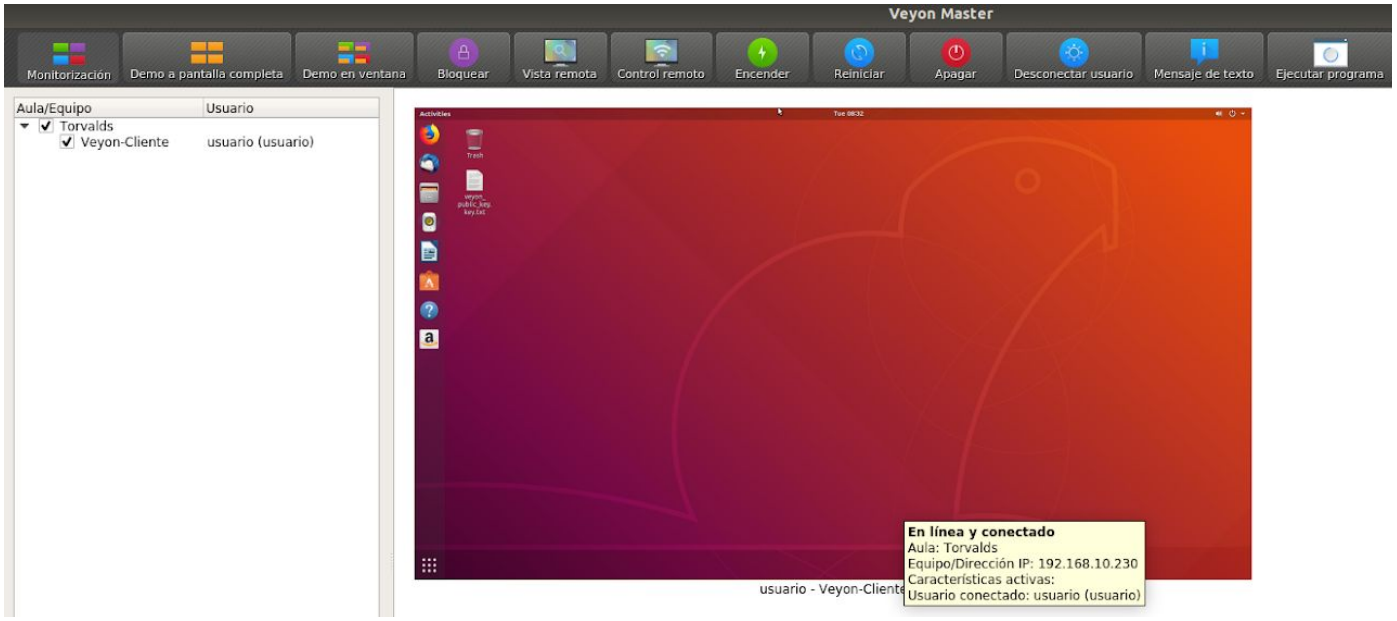


Finalmente nos volverá a dar un resumen de las acciones que hemos tomado y finalizamos.



4.2 Control de equipos

Una vez acabada la configuración ya podremos abrir el programa de **Veyon master**, en él podremos observar en el menú principal las diversas opciones explicadas al principio del capítulo, a la derecha tendremos el menú donde podremos seleccionar los clientes que queremos observar en la vista de pantallas que se encuentra a la izquierda.



5. Actualizando a Ubuntu 18.04

Canonical lanzó la nueva versión de Ubuntu 18.04 con nuevos cambios como el abandono de **lightdm** y dando la bienvenida a GNOME, esta versión cuenta con un soporte extendido hasta el año 2023 además permitir actualizar los paquetes a las últimas versiones.

Pero no todas las cosas son buenas, por ejemplo esta versión no contiene un usuario invitado como lo tenía el anterior ubuntu (Ubuntu 16.04) ya que esto venía implementado en **lightdm**.

En esta actualización hemos pensado la implementación de varias mejoras para esta versión entre las siguientes:

5.1 Usuario invitado

El usuario invitado es de gran utilidad para quien va a utilizar el ordenador temporalmente como para escribir un documento, enviar un correo, etc. Es posible implementarlo en Ubuntu 18.04 utilizando los ejecutables que tiene **GNOME**, estos son los siguientes:

PostLogin: Se ejecutarán las instrucciones dentro de él cuando se **inicie session**
PostSession: Se ejecutarán las instrucciones dentro de él cuando se **cierre la sesión**

Primero vamos a crear un usuario normal y con el comando *mkpasswd* le dejaremos la contraseña en blanco.

```
$ sudo useradd -d /home/convidat convidat
$ sudo passwd -d convidat
```

Después abrimos el archivo PostLogin en el caso de que no exista lo creamos.

```
$ sudo vi  /etc/gdm3/PostLogin/Default
```

Le añadimos las siguientes líneas que consistirán en que cuando el usuario será el de convidat pues este ejecute la creación del directorio

```
#!/bin/sh
guestuser="convidat"

if [ $USER = $guestuser ]; then
        mkdir /home/$guestuser
        cp /etc/skel/.* /home/$guestuser
        chown -R $guestuser:$guestuser /home/$guestuser
fi
exit 0
```

Una vez acabado el **PostLogin** abriremos el archivo **PostSession.**

```
$ sudo vi  /etc/gdm3/PostSession/Default
```

Y le añadimos las siguientes líneas que consistirán en que al cerrar session si es el usuario convidat borre su directorio.

```
#!/bin/sh
guestuser="convidat"

if [ $USER = $guestuser ]; then
   rm -rf /home/$guestuser
fi
exit 0
```

Finalmente le aplicamos los permisos necesarios

```
$ sudo chmod -R 775 /etc/gdm3/PostLogin /etc/gdm3/PostSession
```

5.2 Bloquear el fondo de escritorio, pantalla de inicio y aplicaciones favoritas del dock

Configuración:

Creamos el perfil **user** en el archivo **/etc/dconf/profile/use**r

```
user-db:user
system-db:local
```

La base de datos será local así que creamos su directorio en **/etc/dconf/db/**

```
# mkdir /etc/dconf/db/local.d
```

Dentro de ella tendremos que crear un archivo llamado **00-wallpaper** que contendrá los valores por defecto de las claves de escritorio, esto es necesario ya que el bloqueo no funciona si no está definido.

En fichero **00-wallpaper** añadimos dos segmentos de código que consisten en:

1. Aplicaciones fijadas en el dock por defecto para todos los usuarios.
2. Fondo de escritorio y de inicio por defecto para todos los usuarios.

```
[org/gnome/shell]
favorite-apps = ['nautilus.desktop', 'firefox.desktop', 'chromium-browser.desktop',
'gnome-terminal.desktop']
```

```
[org/gnome/desktop/background]
picture-uri='file:///usr/share/backgrounds/warty-final-ubuntu.png'
picture-options='zoom'
primary-color='#2c001e'
secondary-color='#2c001e'

[org/gnome/desktop/screensaver]
picture-uri='file:///usr/share/backgrounds/warty-final-ubuntu.png'
picture-options='zoom'
primary-color='#2c001e'
secondary-color='#2c001e'
```

Después creamos el directorio **locks** dentro de **/etc/dconf/db/local.d**

```
/etc/dconf/db/local.d# mkdir locks
```

Dentro de **locks** creamos el fichero ***00-wallpaper_lock*** y dentro de él ponemos lo siguientes líneas para Impedir que los usuarios cambien los valores de las siguientes claves:

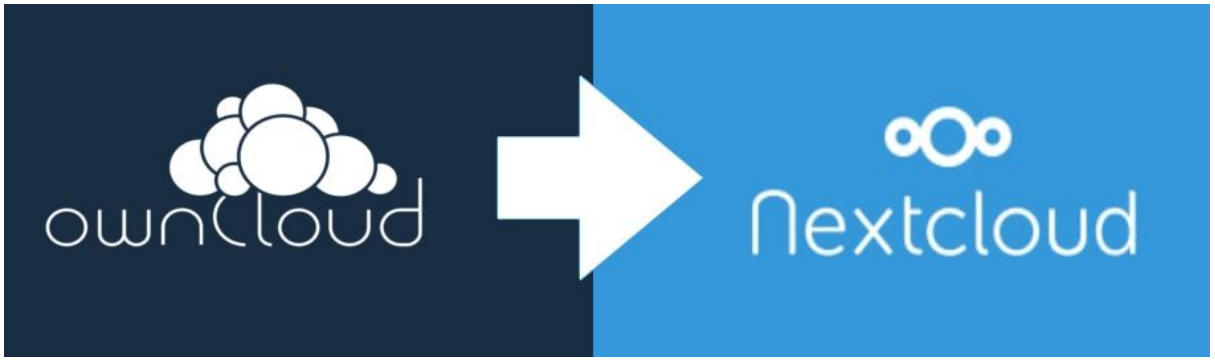
```
# Prevent users from changing values for the following keys:
/org/gnome/desktop/background/picture-uri
/org/gnome/desktop/background/picture-options
/org/gnome/desktop/background/primary-color
/org/gnome/desktop/background/secondary-color

/org/gnome/desktop/screensaver/picture-uri
/org/gnome/desktop/screensaver/picture-options
/org/gnome/desktop/screensaver/primary-color
/org/gnome/desktop/screensaver/secondary-color
```

Por último aplicamos los cambios con **dconf update** y ya tendremos las aplicaciones favoritas en el dock y no se podrá cambiar el fondo de escritorio.

Aunque hayamos definido las aplicaciones fijadas en el dock por defecto, los usuarios podrán seguir poniendo otras aplicaciones ya que no hemos bloqueado la clave.

6. Migración de Owncloud a Nextcloud



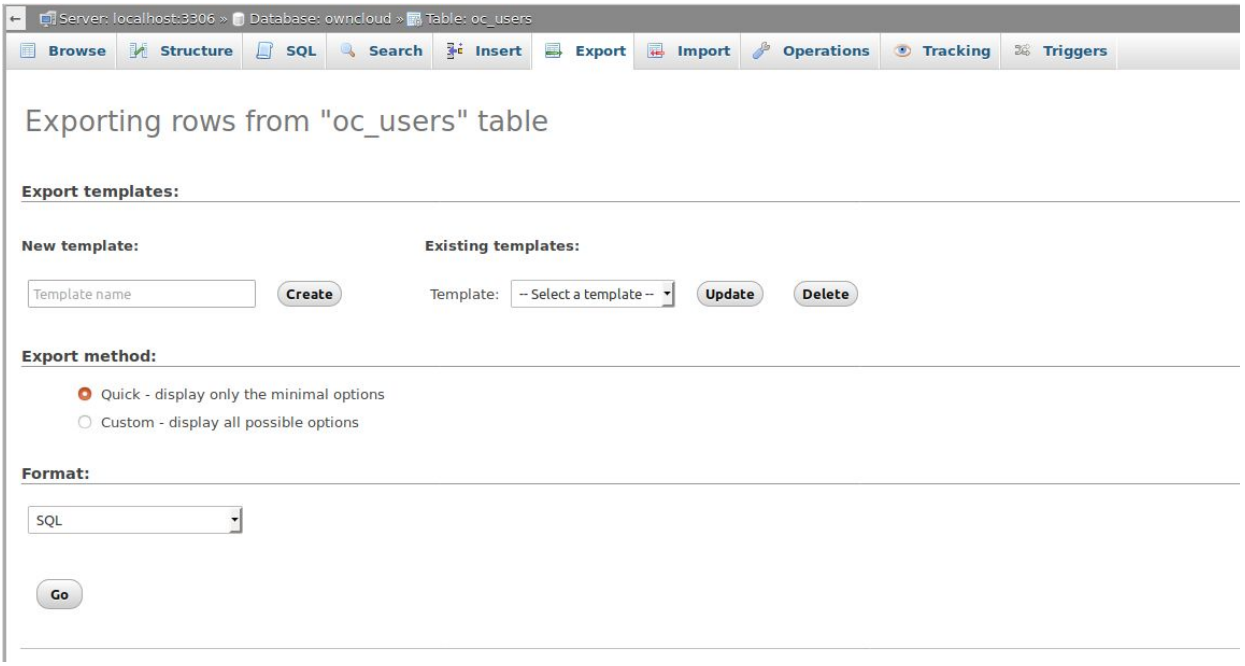
OwnCloud fue uno de los proyectos pioneros en ofrecer a los usuarios la posibilidad de montar su propia nube de forma totalmente gratuita y muy sencilla. Sin embargo, a medida que la plataforma ganó fama, los responsables del proyecto empezaron a pensar en un “modelo de negocio” y lanzaron una versión Enterprise, yendo en contra de los principios de esta plataforma. Por ello, los responsables del proyecto inicial en contra de estas decisiones, hicieron un fork al proyecto lanzando Nextcloud, la nueva nube que busca seguir adelante con los principios de esta plataforma.

Las principales razones para cambiarse a Nextcloud serían las siguientes:

- Los responsables del proyecto inicial se fueron a Nextcloud
- La estabilidad superior de Nextcloud frente a Owncloud
- Tanto desarrolladores como la comunidad están constantemente creando nuevas funciones, características y plugins útiles para los usuarios, funciones que, debido al modelo de negocio de ownCloud, no dejaron implementar antes.

6.1 Migración de los usuarios

Primero migraremos los usuarios para ello accederemos a la base de datos de Owncloud y exportaremos la tabla **oc_users.**



Después iremos a la base de datos de Nextcloud y importamos la tabla que exportamos de la base de datos de Owncloud.

En este caso la tabla de **oc users** tenía la misma estructura tanto en Nextcloud como en Owncloud, si realizamos estos pasos en versiones más antiguas posiblemente tengamos que añadir los usuarios manualmente.

6.2 Migración de los archivos

Una vez migrados los usuarios, podremos empezar a pasar sus archivos para ello, nos dirigiremos al directorio raíz de Owncloud y iremos a la carpeta data.

```
root@soyuz:/var/www/owncloud/data# ll
total 88
drwxrwx---  7 www-data www-data  4096 May 27 15:12 ./
drwxrwxr-x 13 www-data www-data  4096 May 27 14:04 ../
drwxr-xr-x  2 root     root      4096 May 27 15:12 adrian/
drwxr-xr-x  6 www-data www-data  4096 May 27 14:43 avatars/
drwxr-xr-x  2 www-data www-data  4096 May 27 14:35 files_external/
-rw-r--r--  1 www-data www-data   323 May 27 14:05 .htaccess
-rw-r--r--  1 www-data www-data     0 May 27 14:05 index.html
-rw-r--r--  1 www-data www-data     0 May 27 14:05 .ocdata
-rw-r-----  1 www-data www-data 53036 May 27 15:08 owncloud.log
drwxr-xr-x  2 root     root      4096 May 27 15:12 pratik/
drwxr-xr-x  4 www-data www-data  4096 May 27 14:05 root/
root@soyuz:/var/www/owncloud/data#
```

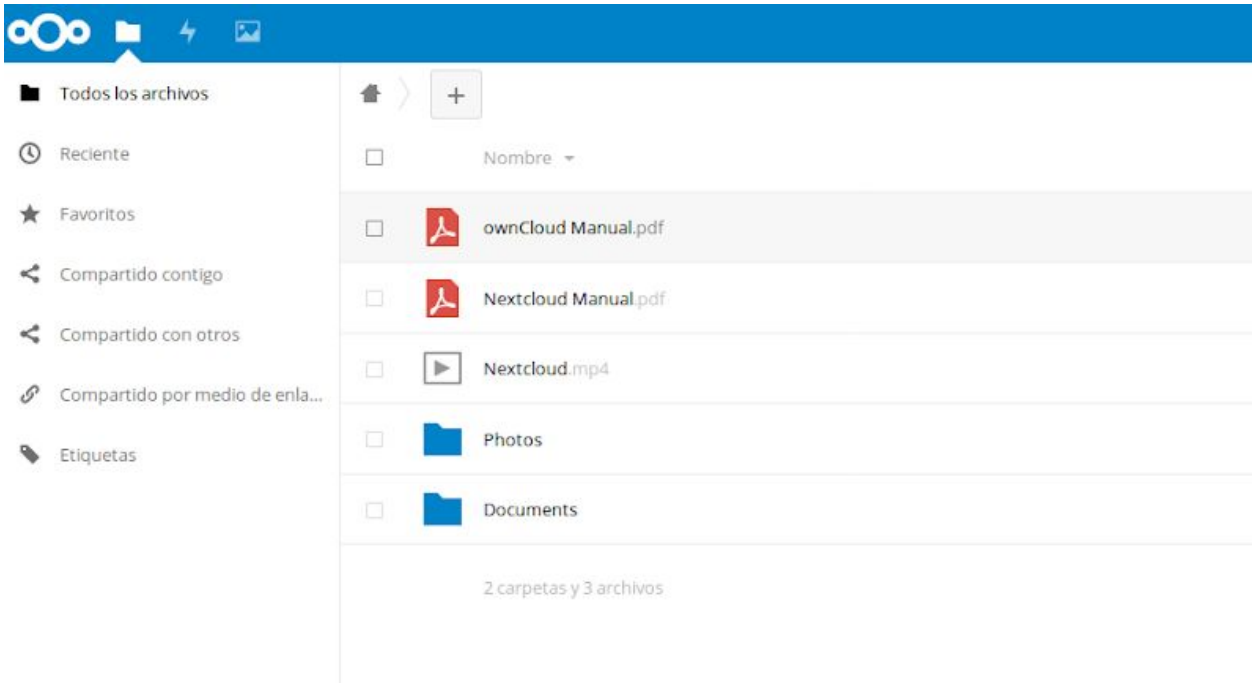
Después copiaremos los respectivos directorios de los usuarios hacia el data de Nextcloud.

```
root@soyuz:/var/www/owncloud/data# cp -r adrian/ pratik/ root/ /var/www/nextcloud/data
root@soyuz:/var/www/owncloud# chown -R www-data:www-data /var/www/nextcloud/data
```

Por último, dentro de la raíz de Nextcloud ejecutamos la siguiente orden para escanear los archivos de los usuarios ya que si no, no aparecían dentro de la nube.

```
usuario@soyuz:/var/www/owncloud# sudo -u www-data php occ files:scan --all
```

Una vez ejecutada la orden ya apareceran nuestros archivos, en este caso solo se ha pasado el manual de Owncloud que venía por defecto en la nube de Owncloud y no había ningún otro.



7. Implementación de BitTorrent en NextCloud

Como ya sabemos Nextcloud es una aplicación de alojamiento de archivos en la web, esta tiene la arquitectura de cliente-servidor. Cuando un usuario baja un fichero grande (10GB) va a una velocidad rápida pero cuando se quiere compartir un fichero grande y lo bajan varios usuarios la velocidad disminuye considerablemente. ¿Cómo se puede solucionar este problema? Precisamente es lo que hace BitTorrent.

BitTorrent es un protocolo que establece las **bases para un intercambio de archivos** basado en la filosofía *peer to peer* (P2P). Para BitTorrent, cada ordenador destino del que hablábamos anteriormente se convierte también en una fuente. De esta forma, los usuarios no sólo descargan el archivo desde Nextcloud, sino que van compartiendo partes ya descargadas entre todos ellos. Ésta es, en esencia, la clave del protocolo BitTorrent en particular y de la tecnología P2P en general.

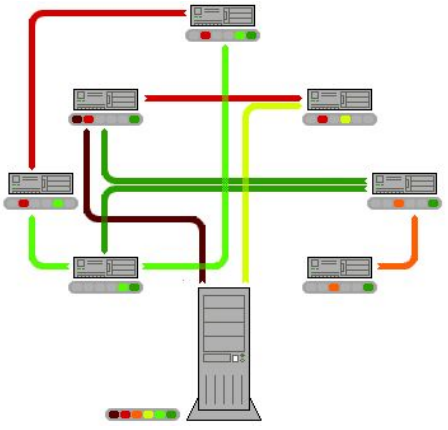


Figura 6: Esquema BitTorrent

7.1 Implementación de BitTorrent

7.1.1 Configuración transmission

Para implementar BitTorrent en Nextcloud hemos usado el programa **Transmission**, que es un cliente P2P, gratuito y de código abierto para la red BitTorrent.

En la configuración de Transmission (*/etc/transmission-daemon/settings.json*) únicamente habilitamos el descubrimiento en red que es el siguiente parámetro:

```
        "lpd-enabled": true
```

7.1.2 Preparación del script

Una vez preparado el transmission hemos programado un script para generar el torrent en bash que es el siguiente:

```
#!/bin/bash
# El Primer parámetro ($1) será el nombre del usuario
USERNAME=$1

# El segundo parametro ($2) sera el nombre del archivo sin la extensión
FILE=$2

# Definimos el directorio donde se comparten los archivos torrent
DOWNLOADDIR="/var/lib/transmission-daemon/downloads"

# Verificamos si el archivo acaba en .torrent (Para no convertir un .torrent a torrent)
if [ ${FILE: -8} == ".torrent" ]
        then
                exit 1;
        fi
# Hacemos un enlace del archivo del usuario hacia el directorio de descargas
ln -s /var/www/nextcloud/data/$USERNAME/files/"$FILE" $DOWNLOADDIR

# Creamos el torrent y lo guardamos en la carpeta del usuario
transmission-create transmission:transmission $DOWNLOADDIR/"$FILE" -o
/var/www/nextcloud/data/$USERNAME/files/"$FILE.torrent"

# Añadimos el torrent al transmission para que los usuarios puedan bajarlo
transmission-remote --auth usuario:usuario --add
/var/www/nextcloud/data/$USERNAME/files/"$FILE.torrent"

# Nos dirigimos al directorio raíz de Nextcloud   y hacemos un scan para que aparezca el
torrent
cd /var/www/nextcloud
php occ files:scan --path $USERNAME
```

7.1.2 Configurando Nextcloud

Una vez tenemos el script y el transmission preparados nos dirigimos al archivo donde se comparten los archivos (*/var/www/nextcloud/apps/files_sharing/templates/public.php*)

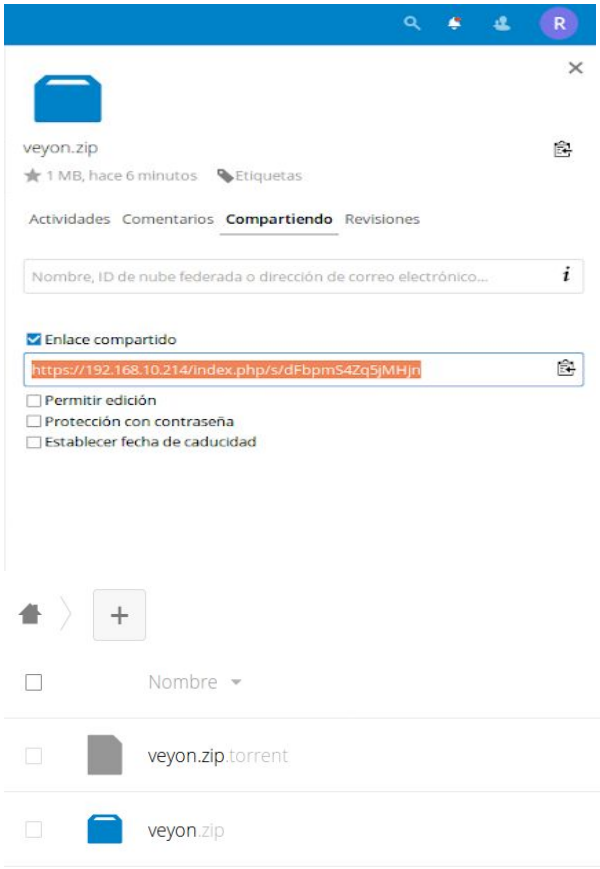
Una vez abierto el archivo buscamos lo siguiente: (*Download %s*) y al final del div añadimos lo siguiente indicado con **+**:

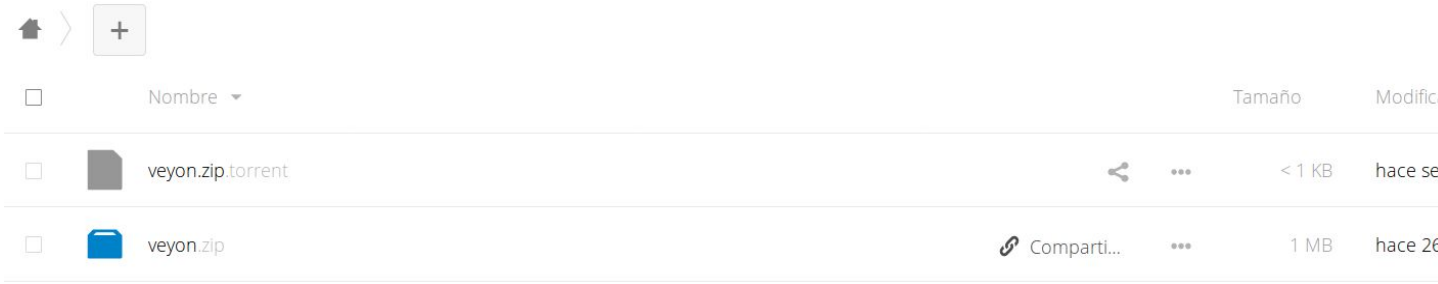
```
 <div class="directDownload">
<a href="<?php p($_['downloadURL']); ?>" id="downloadFile" class="button">
<span class="icon icon-download"></span>
<?php //print_r(array($_['filename'])); ?>
<?php //print_r(p($_['downloadURL'])); ?>
<?php p($l->t('Download %s', array($_['filename'])))?> (<?php p($_['fileSize']) ?>)
</a>
<?php //print_r(p($_['downloadURL'])); ?>
+ <?php shell_exec("bash /var/www/nextcloud/torrent/seed_file.sh '".($_['owner'])."'
'".($_['filename'])."'"); ?>
</div>
```

Al **shell_exec** le diremos que ejecute el script que hemos preparado previamente, enviando los parametros de quien es el usuario y cual es el nombre del archivo.

7.2 Funcionamiento

Una vez implementado el Bittorrent, cuando compartamos un archivo y accedemos a él por primera vez nos generará el archivo .torrent en nuestro directorio de Nextcloud para las próximas veces que lo compartamos.





Al abrir nuestro cliente torrent veremos como nos detecta un cliente mediante descubrimiento en red, este será el transmission que nos compartirá el fichero