
# TRABAJO FINAL DE GRADO

**TÍTULO DEL TFG:** Supervivencia de datos y resistencia a fallos en redes multiservicio con software Open Source

**TITULACIÓN:** Grado en ingeniería Telemática

**AUTOR:** Carles Artigas Morales

**DIRECTOR:** Roc Meseguer

**FECHA:** 13/01/2016

**Título:** Supervivencia de datos y resistencia a fallos en redes multiservicio con software Open Source

**Autor:** Carles Artigas Morales

**Director:** Roc Meseguer Pallarès

**Fecha** 13/01/2016

**Resumen**

El objetivo de este proyecto es diseñar un servidor multifunciones seguro y fiable en dos ámbitos muy diferentes , uno de ellos el empresarial y otro el particular , utilizando software Open Source.

El proyecto describe dos escenarios , uno de ellos enfocado al ámbito corporativo, donde se implementará un sistema de seguridad de datos , en el cual se protegerán tanto físicamente como por software el acceso a la información así como su almacenamiento y otro enfocado al ámbito particular , el ámbito de casa , donde se facilitará al usuario el acceso a contenido multimedia y a su vez se garantizará la protección de datos

En la introducción se introduce al lector en diferentes problemas y como y porque FreeNAS es la herramienta escogida para solventar tales cuestiones.

En Tecnologías utilizadas se explica las diferentes tecnologías que se usan en los dos escenarios, de esta manera el lector podrá seguir sin ningún problema los capítulos posteriores.

En el capítulo 4 y 5 se explica los dos escenarios, empresarial y home server respectivamente y se expondrá como FreeNAS ofrece diferentes soluciones las cuales se usarán para crear , diseñar e implementar el proyecto

Se concluye el proyecto explicando que posibles soluciones podrían haber sido implementadas usando la herramienta FreeNAS y las previsiones de futuro

En los Anexos se introduce todo aquello que provoca que la lectura de esta memoria se entorpezca pero que a su vez tiene una explicación mas en detalle sobre los escenarios.

Todo este proyecto ha sido diseñado, implementado y pensado por Carles

Artigas Morales.

FreeNAS además ofrece cientos de posibilidades que no han sido implementadas en este proyecto y que de igual manera presentan una potencia parecida a la que se ha implementado en este proyecto.

**Title:** Data survivability and failure resilience in a multiservice network with Open Source software

**Author:** Carles Artigas Morales

**Director:** Roc Meseguer Pallarès

**Date:** 13th January 2016

**Overview**

The goal in this project is to design multifunction server, reliable and safe.

Two different topics will be described, one of them related on corporation field and the other one related with home.

This project describes both scenarios, the corporative one's implements a data security system where information would be secure using physical protection or software protection. The one focused on home, ease the user access to get multimedia content and protect data using similar mechanism of corporation one.

In the introduction , reader will be introduce in a bunch of problems and how and why FreeNAS it's the tool chosen to get issue solved.

In used technologies, it is explained the many technologies that would be applied in both scenarios .This it is presented to the reader to avoid missing some important concept.

In chapter 4 and 5 the scenarios would be described and would be exposed how FreeNAS would offer many appliance.

The project would be concluded explaining which possible appliances would be able to be implemented using FreeNAS Tool and forecasting implementations

In the annexes, explanations would get in to deeper understanding to avoid hinder the and try to provide more details.

This project has been designed, implemented and thought by Carles Artigas Morales

FreeNAS offers as well hundreds of new possibilities that couldn't been implemented in this project due the hugeness of possibilities that FreeNAS offers . All possibilities have the same capabilities as the ones implemented

# ÍNDICE

**FIGURAS**

**TABLAS**

# CAPÍTULO 1. INTRODUCCIÓN

En este proyecto se diseñaran dos escenarios. El primero de ámbito empresarial y el segundo de ámbito particular.

En el primer escenario empresarial se trata la seguridad de datos, los cuales durante años son el activo más importante de muchas empresas así como multinacionales como datos de cliente, cuentas bancarias, desarrollo de productos, etc... Todos estos datos se guardan en formato digital y, por tanto, están sujetan a todas los posibles fallos y/o imprevistos que sucedan en la infraestructura así como también están sujetos a errores de empleados, virus informáticos y atacantes maliciosos. Por tanto, es necesario encontrar una manera de proteger y asegurar que los datos no se pierdan ni sean modificados.

La herramienta escogida para solventar tal reto es FreeNAS (Fig. 1. 1) , el cual ofrece un amplio abanico de soluciones. FreeNAS provee a los datos empresariales y/o particulares con soluciones para sobrevivir a los posibles fallos que puedan suceder tanto físicos como de software e incluso errores de usuario ,con la infraestructura empresarial. Además una gran ventaja es que los datos pueden almacenarse remotamente y gestionarse con la facilidad que se gestionaría los datos en local, pero con las ventajas que ofrece tanto en seguridad, redundancia, variedad y servicios que facilitan la gestión y el cifrado de la información


**Fig. 1. 1.** Logotipo de FreeNAS

La herramienta FreeNAS a su vez, es también la herramienta escogida como solución para el ámbito particular, ya que además de su capacidad para proteger los datos ante fallos también tiene muchos servicios que se pueden implementar.Por ejemplo en forma de discos compartidos a través de la red o en forma de *jails* (instancias virtuales de máquinas basadas en FreeBSD ), estas *jails* se utilizan para implementar servicios usando el espacio de almacenamiento y los recursos de la maquina que virtualiza)

FreeNAS se define como un software Open Source gratuito de *Network-Attached Storage*, basado en FreeBSD y que utiliza el sistema de archivos OpenZFS. FreeNAS soporta Windows, MAC, ESX ,clientes Unix, Android y IOS. Asimismo posee una arquitectura pensada para poder implementar *software* de terceros ampliando exponencialmente sus funciones y soportando

gran cantidad protocolos de almacenamiento, de cifrado, de *sharing* , de Microsoft y otros varios.

## 1.1 Objetivo del proyecto

El objetivo principal de este proyecto es la creación de un servidor multiservicio y multiplaforma siendo Open Source enfocado en dos diferentes ámbitos como son el empresarial y el particular.En el caso de el empresarial supliendo las necesidades básicas de cualquier *Startup* como son la protección y el cuidado de la información , el almacenamiento y los servicios básicos TIC.En el caso particular facilitar , automatizar el acceso a contenido multimedia así como su reproducción a demanda y el acceso a los archivos locales con los dispositivos móviles independientemente de donde este situado el dispositivo móvil.

Asegurar un *data resilience*, que sería la capacidad de adaptarse y recuperarse a cualquier incidencia / desastre producido en los datos , es importante asegurar lo antes dicho para evitar la pérdida de información . Dependiendo de lo sensible que sea esta información perdida , puede causar un gran desastre económico y estructural en un organización según sea la magnitud de la pérdida o en cuanto en el uso personal la perdida de fotografías, videos y documentos. Por tanto hoy en día cualquier organización necesita poder asegurar esta condición y para ello existen muchas herramientas y mecanismos para evitar que suceda, uno de estos mecanismos es FreeNAS y su capacidad de asegurar a partir de configuraciones RAID, *backups*, *snapshots* y replicaciones esta característica tan demandada y vital.

La seguridad en datos y su acceso es otro objetivo tan importante como *data resilience*. Una brecha de seguridad en un proyecto puntero, en información sensible que pueda beneficiar a la competencia o simplemente dejar en evidencia la organización. Es necesaria una seguridad en los datos los cuales queremos compartir , así como seguridad en aquellos que no queremos que nadie pueda acceder, pero a su vez queremos tener un fácil y rápido acceso desde cualquier lado, por eso , es necesario adoptar unas políticas de seguridad en datos bastante altas, las cuales FreeNAS es capaz de asegurar

Otro objetivo será el asegurar un acceso universal (cualquier *OS* puede interactuar con el servidor), es necesario que se pueda acceder a la información desde cualquier dispositivo con estándares de red y protocolos, por tanto cualquier servicio debe poder efectuarse y usarse desde cualquier dispositivo.

También se quiere asegurar un acceso a los documentos desde cualquier ubicación , sin dejar de lado la seguridad y permitiendo que los usuarios puedan trabajar cómodamente como si de una red local se tratase.

Otro objetivo es tener la capacidad de virtualizar la infraestructura y los servicios y que toda ella sea controlable de manera centralizada desde la red y fácil , permitiendo el archivado y el clonado según necesidad

En lo que se refiere al objetivo relacionado con el escenario particular corresponde a un *home server* multiservicio .

Este servidor brinda un servicio automatizado de búsquedas según la calidad deseada y utilizando el famoso servicio Torrent (p2p) o Usenet .Además ofrece auto organización de Torrents , NZB y todo archivo multimedia que exista en el servidor, renombración automática de archivos , movimientos automáticos de datos entre volúmenes , búsqueda automática de series,libros,música,películas que en el momento que exista el Torrent o el archivo NZB serán automáticamente encontrados, descargados y auto organizados y todo ello con el mínimo esfuerzo para el usuario y sin la pérdida de tiempo que supone el hecho de buscar , encontrar y descargar el archivo deseado en la calidad deseada y con el menor tiempo posible de descarga , ya que analiza los *seeders*(en cuanto a Torrent) , la calidad del archivo descargado y utiliza los metadatos de las paginas oficiales de cine, música y libros para complementar el archivo (caratulas, director, subtítulos etc.. . Además también ofrecerá la posibilidad de reproducción por *streaming* a demanda a cualquier dispositivo que haya conectado con el servidor , independientemente de la localización y el *SO* utilizado para acceder. Para facilitar la gestión también ofrecerá un servicio muy parecido al de Dropbox o drive pero con la ventaja de que es gratuito y sin cuotas de espacios, ya que el único límite lo pone la infraestructura del usuario, este servicio permitirá gestionar las fotos de familia, las películas , la música , los libros y cualquier tipo de dato remotamente y de una manera muy simple.

## 1.2 Bussines Resilience

Hoy en día todas las empresas generan cantidades enormes de información, a mayor empresa, mayor cantidad de datos generados por segundo, para muchas de ellas, una pérdida de información puede ser un factor crítico el cual condicione el fracaso o el éxito de la organización, además de generar millones de euros en pérdidas económicas.


**Fig. 1. 2.** Diagrama de capacidad de recuperación ante fallos

La idea de este proyecto es proveer un *framework* (Fig. 1. 3) seguro que se pueda ajustar a los estándares del mercado y que que contenga todas las herramientas para hacerlo posible, para ello el *software* escogido es FreeNAS por tanto en cuanto a tratamiento de datos se refiere, en el cual se mitigan los riesgos, se reduce el impacto del daño y aumenta la capacidad de recuperación a fallos de la empresa provocando que esta sea más competitiva en el mercado.

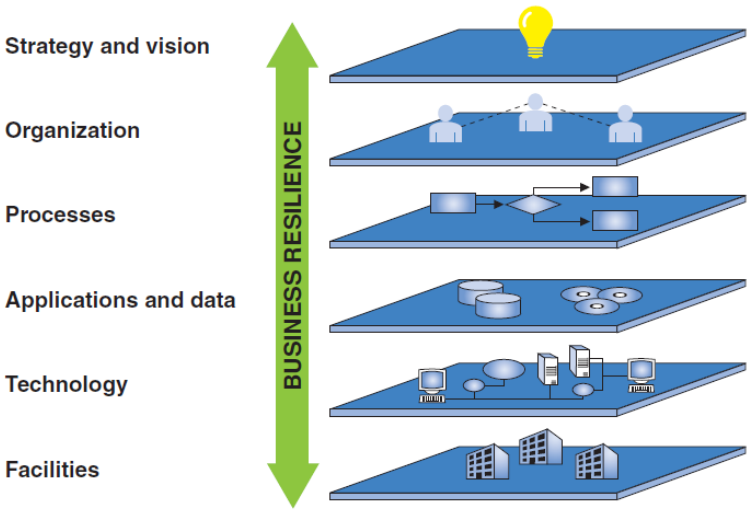



**Fig. 1. 3** Crecimiento favorecedor en una empresa

FreeNAS se puede enfocar  en el *framework* (Fig. 1. 3) de *busines resilience* tocando las capas Facilities , Technology, Applications and data, Processes y Organization Issues.

¿Qué es *Bussines Resilence* y porqué es importante?

Es la habilidad de rápidamente adaptarse y responder a los problemas así como la continuidad de las operaciones empresariales. *Bussines resilence* debe interpretarse como el ticket a la continuidad empresarial.

Centrarse en el desastre lleva a la organización a defenderse del daño pero es mejor un acercamiento proactivo que ayude a la empresa a responder a una fallida inesperada de manera más rápida y con un coste menor. Asimismo, esto ayuda a una empresa a demostrar su cumplimiento con los requerimientos regulatorios del país.

# CAPITULO 2: TECNOLOGIAS UTILIZADAS

## 2.1 Zettabyte File System

ZFS es un sistema de archivos y volúmenes desarollado por SUN Microsystems .

A diferencia de otros sistemas de archivos, no es necesario definir tamaños de partición a la hora de crear el sistema de archivos, en su lugar , agrupas un numero de discos en un ZFS pool(la manera de organizar volúmenes en FreeNAS) , una vez ha sido creado, dinámicamente puedes crear Datasets. Los Datasets son un espacio de almacenamiento que se puede utilizar para optimizar el espacio usado según qué tipo de datos se quieran guardar , además permite la personalización de permisos según que usuarios y grupos quieran acceder.

Este sistema de archivos fue creado con la idea que nunca en la práctica se pudiese llegar a sus límites teóricos.

ZFS es un sistema de 128 bits es decir su capacidad es $2^7$ veces el limite teórico de un sistema de 64 bits. Jeff Bonwick ,el arquitecto jefe de Sun dijo que:
"Llenar un sistema de archivos de 128 bits excedería los limites cuánticos de almacenamiento en la Tierra, no puedes rellenarlo sin hervir los océanos"

Actualmente a día de hoy, el prefijo *Tera*, es el utilizado en las empresas cuando se habla de capacidad, es casi equivalente proporcionalmente en órdenes de magnitud a la diferencia entre *Zetta*($10^{21}$) y *Tera* ($10^{12}$) a la diferencia en capacidad de un byte respecto a un terabyte.

Este sistema de archivos permite una capacidad de almacenamiento de 293 Zettabytes ,es decir aproximadamente 3 x ($10^{23}$) bytes. Para hacernos una idea de la magnitud que estamos hablando, podemos observar el tráfico previsto total para el año 2016 en internet según cisco, que será de 1.1 Zettabytes al año según cisco

$$\frac{293*10^{21}}{1{,}1*10^{21}} = 266{,}36 \text{ años}$$

Si quisiéramos llenar un sistema de archivos ZFS usando el bandwitch total del tráfico de internet para el año 2016 , es decir 1,1 zettabytes por año , necesitaríamos un total 266.36 años para llenar toda la capacidad.

Algunos límites teóricos de ZFS son:

$2^{48}$ — Número de snapshots en cualquier sistema de ficheros ($2 \times 10^{14}$)

$2^{48}$ — Número de ficheros en un sistema de ficheros ($2 \times 10^{14}$)

16 exabytes — Tamaño máximo de un sistema de ficheros.

16 exabytes — Tamaño máximo de un fichero.

16 exabytes — Tamaño máximo de cualquier atributo.

$3 \times 10^{23}$ bytes (293 zettabytes aprox.) — Tamaño máximo de un zpool.

$2^{56}$ — Número de atributos de un fichero (realmente limitado a $2^{48}$ que es el número de ficheros que puede contener un sistema de ficheros ZFS).

$2^{56}$ — Número de ficheros en un directorio (realmente limitado a $2^{48}$ que es el número de ficheros que puede contener un sistema de ficheros ZFS).

$2^{64}$— Número de dispositivos en cualquier zpool.

$2^{64}$— — Número de zpools en un sistema.

$2^{64}$— — Número de sistemas de ficheros en un zpool.

### 2.1.2 Características

**Integridad de datos** una de las mayores características de ZFS respecto a otros sistemas de archivos es que ZFS se centra en la integridad de los datos, está diseñado para proteger la información del usuario contra corrupción silenciosa de datos debido a la degradación de datos, bugs en el firmware del disco , escrituras fantasma , escritura/lectura fallida (acceso al bloque equivocado) , errores de driver, sobre escrituras accidentales.

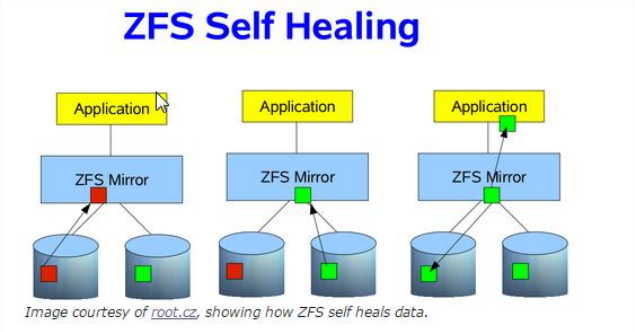



**Fig 2. 1.** ZFS self healing

Cuando se accede a un bloque de datos, independientemente si hay datos o metadatos , se calcula un *checksum* y se compara con el *checksum*

previamente guardado para comparar como es y cómo debería ser , si el *checksum* encaja , los datos se pasan al *stack* de aplicación que procesa los datos. Si no encaja , ZFS comenzara la reparación (Fig 2. 1 ) siempre y cuando se esté usando paridad y redundancia(mirror o RAID)

ZFS no tiene una herramienta de reparación equivalente a Fsck de Unix o *scandisk* de Windows , en lugar de ello utiliza una herramienta denominada scrub . Esta herramienta lee cada bloque , tanto si son datos o metadatos y detecta los errores y bloques corruptos para su reparación.

**ZFS es transaccional, Copy-on-write(COW)** Por cada petición de escritura se crea una copia del bloque de datos asociado y todos los cambios se hacen en la copia en lugar de modificar el bloque original. Después se actualizan los punteros para que señalen a la localización del bloque nuevo.

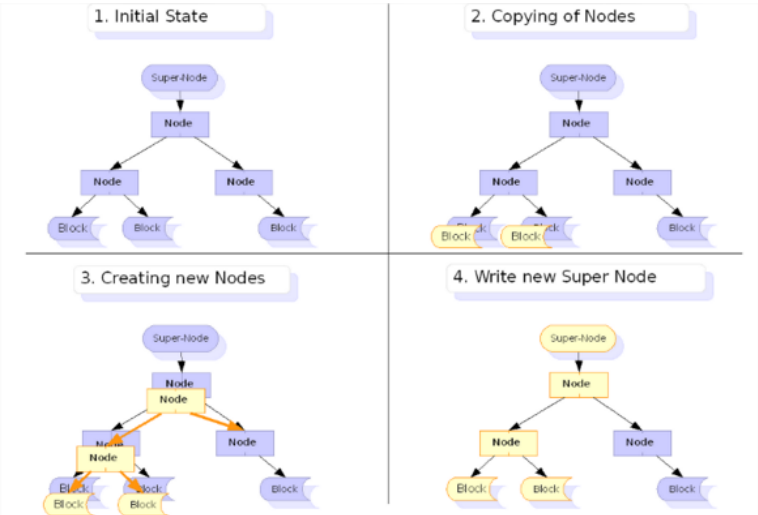


*Image showing the COW model when a data block is created, and the hash tree is updated. Courtesy of dest-unreachable.net.*

**Fig 2. 2** Diagrama Copy-on-write

Esto significa que ZFS siempre escribe en espacio libre (Si hay disponible, Fig 2. 2) y la mayoría de escrituras serán secuenciales.
La mayor característica del COW es que se pueden hacer *snapshots* de los datos , ya que como siempre queda el antiguo bloque en algún lugar del disco y aunque este haya sido marcado como espacio libre se puede recuperar. Si el bloque ha sido sobrescrito después de haber sido *snapshoted* , se copia en el sistema de archivos del *snapshot*. Esto es posible porque el *snapshot* es una copia del hash del árbol en un momento exacto. Gracias a esta característica los *snapshots* no ocupan apenas espacio siempre y cuando los bloques no hayan sido sobrescritos.

Debido a esta característica COW suele dejar los discos con una fragmentación muy grande , provocando caídas de rendimiento, por tanto es necesario

realojar los bloques. Para combatir este defecto, a su vez utiliza bloques de 128 KB que minimizan la fragmentación 32 veces

**Compresión a tiempo real** La compresión se hace cuando el bloque se escribe en el disco , pero solo si los datos se benefician de la compresión, cuando un bloque comprimido es accedido , se auto descomprime automáticamente , como a diferencia de otros sistemas ZFS comprime a nivel de bloque, no a nivel de archivo, por tanto es transparente para las aplicaciones que aceden a los datos comprimidos

## 2.2 Almacenamiento en FreeNAS y FreeBSD

FreeNAS ofrece distintos tipos de almacenamiento, además de aceptar la totalidad de tipos de disco (SCSI ,SAS, Sata, SSD …).

- **Stripe:** requiere al menos un disco, consiste en dividir la información en bloques y guardarla en diferentes discos de manera parcial, de esta manera se aumenta la velocidad de escritura/lectura. Si solo hay un disco lo guarda en bloques pero en el mismo disco, además utiliza el máximo espacio disponible.Fig 2. 3

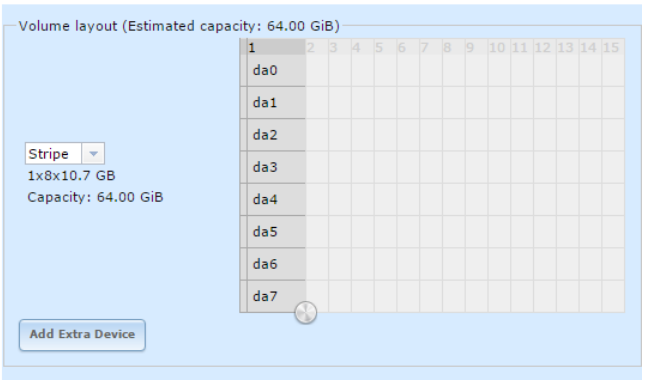



**Fig 2. 3** Organización en stripe

En esta configuración en *stripe*, se dividirá la información en 8 discos diferentes , pero en el caso de fallida de uno de ellos se pierde toda la información, En el ejemplo se ha usado discos que poseen 10 GB de capacidad(ya que es de pruebas) , pero la información almacenable total son 8 gb por disco. Por tanto la total son 64GB

- **Mirror:** Significa que cada disco es un clon y tiene la misma copia de la información. En el caso propuesto debajo, son cuatro discos, los cuales, cada uno de ellos posee una copia exacta de la información , por ello el espacio total es de 8 GB.Fig 2. 4

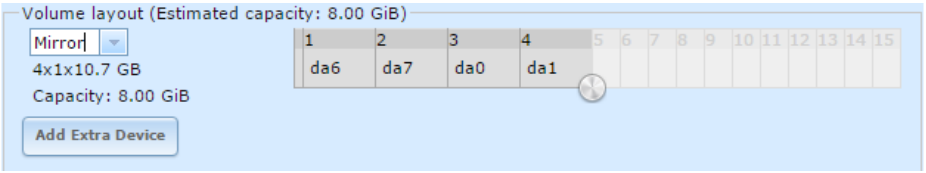



**Fig 2. 4** Organización Mirror

En este caso se agrupan los discos en pares , así duplicamos el tamaño y dejamos el número de copias a la mitad. Por tanto en el caso de arriba si se pierden tres discos no pasaría nada, la información no se perdería, además que la velocidad de lectura/escritura aumentaría por 4. En el caso de abajo si perdiéramos da2 y da3 perderíamos toda la información ya que guardan lo mismo ,pero la perdida de da2 y da5 o de da4 y da3 simultaneas no nos haría perder la información, en este caso la velocidad de escritura/lectura es un poco mayor que el doble.Fig 2. 5

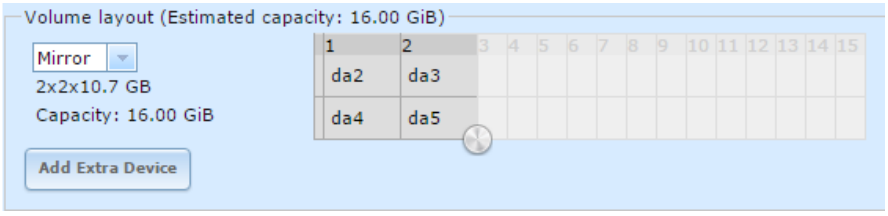



**Fig 2. 5** Organización Mirror 2

- **RAIDZ1:** Requiere de al menos 3 discos, contiene una gran similitud con RAID 5 y RAID 3 ( Fig 2. 6) , se diferencia en que distribuye los bloques de paridad entre discos, y además corta los bloques de paridad y los distribuye entre discos a su vez, además RAID z1 debe acceder a la información leyendo todos los discos a la vez, también la información se posiciona entre discos dinámicamente y la paridad se añade cuando es necesaria.

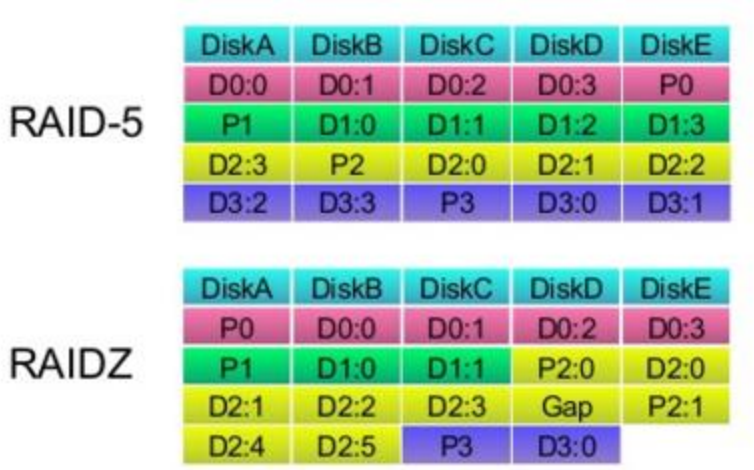



**Fig 2. 6** Comparativa RAID5 - RAIDZ

- **RAIDZ2:** Posee las mismas características que RAID Z1 pero usando doble paridad.
- **RAIDZ3:** las mismas características que RAID Z1 pero usando triple paridad.
- **log device:** Sera un disco dedicado exclusivamente a guardar los logs de error
- **cache device:** Será un disco dedicado exclusivamente a hacer de cache, su función es mantener en memoria, todo aquello que la RAM no haya almacenado (por saturación o para acelerar la lectura/escritura de los datos)
- **spare:** Es un tipo de configuración que se suele añadir a raíz z , z2, z3 por si en algún momento fallara un disco , este comenzaría la reconstrucción de la información automáticamente, además se puede configurar para que sea global, esto implica que podrá reconstruir la información de diferentes data pools del mismo volumen.

El espacio faltante en cualquier configuración se lo reserva el sistema para utilizarlo como swap o para metadatos , así como cuando el volumen se llena , tener la capacidad de poder borrar archivos. FreeNAS calcula el espacio disponible usando bloques de 128 KiB , por eso visualmente parece que hay menor espacio que el tamaño total de los discos, además cualquier archivo copiado en el sistema ocupara menos espacio que en un sistema que no sea ZFS

## 2.3 Usenet

Usenet se creó en el año 1979 una década antes de que se creara el *world wide web*, por esto, Usenet es la red de computadoras más antigua que existe que aún sigue siendo ampliamente usada.

Usenet es un sistema de discusión distribuido mundialmente, basado en la arquitectura UUCP(*Unix-to-Unix Copy*), término referido al conjunto de programas y protocolos que permiten la ejecución remota de comandos , archivos ,emails , noticias entre ordenadores)

Usenet se parece a un sistema BBS (*bulletin board system*, un software que permite a los usuarios conectarse a un sistema usando una terminal y a partir de ahí, los usuarios leen y escriben post de diferentes categorías, así como suben o descargan datos y intercambian mensajes entre ellos.

Usenet se puede ver de manera superficial como un hibrido entre email y foros web. Los post y la información se guardan de manera secuencial. Por tanto cuando un usuario postea un artículo, este se organiza en categorías llamadas *newsgroups*, que estos grupos están auto organizados en jerarquías y sujetos. Por ejemplo *sci.math*, *sci* se refiere a *science* y *math* a *mathematics*. Una vez posteado y organizado cada servidor se comunica con otros(Fig 2. 7), haciendo que los artículos se copien de servidor en servidor, por tanto un artículo eventualmente puede llegar a todos los servidores de la red.

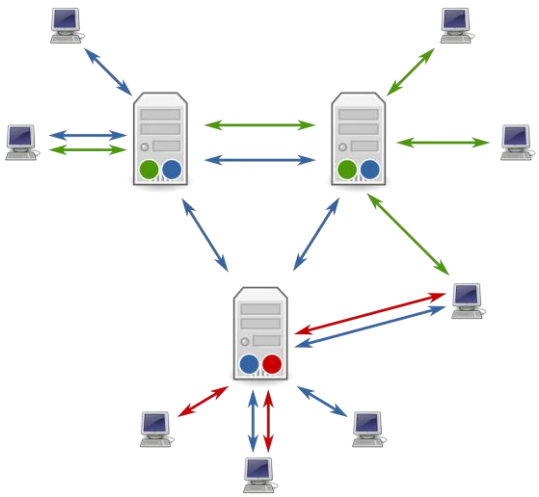


**Fig 2. 7** Usenet diagrama servidor-cliente

Usenet es normalmente el que envía el que inicia la transferencia en lugar del que recibe, esto puede parecer ineficiente, pero Usenet fue creado cuando las redes de telecomunicaciones eran lentas y el ancho de banda era un recurso preciado.

Hoy en día Usenet ha perdido importancia respecto a foros y blogs, Usenet necesita de un cliente de noticias para poder leer los post. Además existen grupos de binarios que son utilizados muy ampliamente para transferencia de datos.

Usenet se puede pensar como una red de *flodding* , ya que a la que un servidor recibe datos nuevos, los transfiere al siguiente , hasta que todos los servidores de la red tienen una copia de la información

Ya que Usenet se creó originalmente para distribuir contenido codificado en 7-bit ASCII, para distribuir cualquier tipo de contenido multimedia es necesario primero convertirlo en texto utilizando algún programa de codificación en binario. Usenet a su vez tiene un tiempo de retención, este tiempo será el cual un binario o un artículo estará en el servidor, cuando este tiempo se haya agotado, se liberara el espacio, para poder ser ocupado por otro binario

### 2.3.1 Organización

Los Principales grupos de noticias de Usenet son:

- Comp. ➔ Noticias sobre ordenadores
- Humanities. ➔ Noticias sobre humanidades
- Misc ➔ Noticias sobre educación, niños etc..
- news ➔ discusiones sobre noticias
- rec ➔ noticias sobre recreación y ocio
- sci ➔ noticias sobre ciencia
- soc ➔ noticias sobre social
- talk ➔ noticias sobre tópicos varios
- alt.binaries ➔ Es la jerarquía más grande de todas, es donde se cuelgan los ficheros de media para su posterior transferencia

### 2.3.2 NZB

NZB es un formato de archivo basado en XML para coger post de los servidores Usenet, NZB específica que necesita ser descargado . Cada mensaje de Usenet tiene un identificador único, que generalmente se subdivide en múltiples mensajes, cada uno de ellos teniendo su id, un cliente de NBZ será capaz de leer el id de todos los mensajes necesarios con tal de decodificar el archivo y devolverlo a su estado original.(Fig 2. 8)

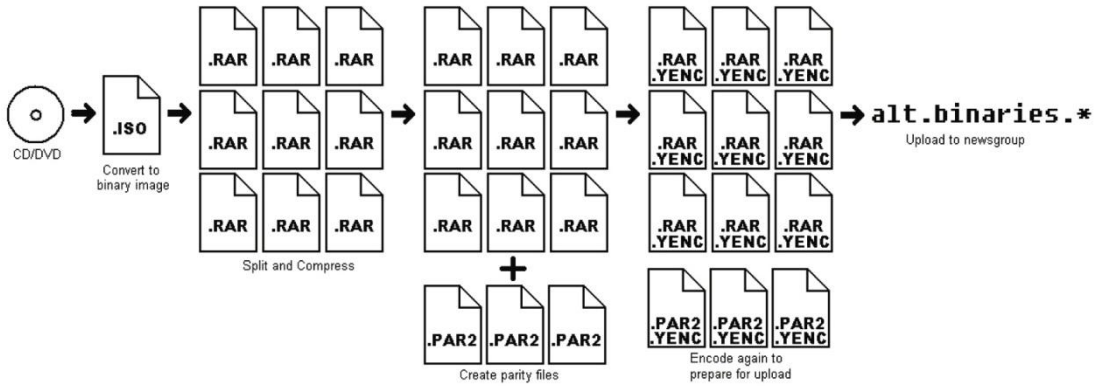



**Fig 2. 8** Conversion NZB

Archivo NZB codificado en XML

```
<?XML version="1.0" encoding="iso-8859-1" ?>
<!DOCTYPE NZB PUBLIC "-//newzBin//DTD NZB 1.1//EN"
"http://www.newzbin.com/DTD/NZB/NZB-1.1.dtd">
<NZB xmlns="http://www.newzbin.com/DTD/2003/NZB">
 <head>
   <meta type="title">Your File!</meta>
   <meta type="password">secret</meta>
   <meta type="tag">HD</meta>
   <meta type="category">TV</meta>
</head>
 <file poster="Joe Bloggs &amp;lt;bloggs@nowhere.example&amp;gt;"
date="1071674882" subject="Here's your file!  abc-mr2a.r01 (1/2)">
   <groups>
     <group>alt.binaries.newzbin</group>
     <group>alt.binaries.mojo</group>
   </groups>
   <segments>
     <segment bytes="102394"
number="1">123456789abcdef@news.newzbin.com</segment>
     <segment bytes="4501"
number="2">987654321fedbca@news.newzbin.com</segment>
   </segments>
 </file>
</NZB>
```

## 2.4 JAILS implementadas y servicios

Una *jail* es una maquina independiente, virtualizada y basada en FreeBSD que utiliza los recursos del servidor FreeNAS

- **MediaPLEX** es una jail de FreeNAS que es la encargada de ofrecer el servicio de *streaming* a demanda , este servicio tiene dos maneras de ofrecer el *streaming* , usando PLEX.tv como en la imagen de estados de la ¡Error! No se encuentra el origen de la referencia., que es una solución muy buena si el usuario no dispone de una IP estática. O utilizando una IP publica que sigue la siguiente estructura http://[IPPUBLICA]:32400/web/index.htlm evitando el intermediario.
  Además MediaPLEX ofrece la posibilidad de sincronizar con canales de todo el mundo y permite reproducirlos en *streaming*, estos canales suelen tener una retención de 7 días.

  La comunicación usando el servidor PLEX.tv seria de la siguiente manera Fig 2. 9.

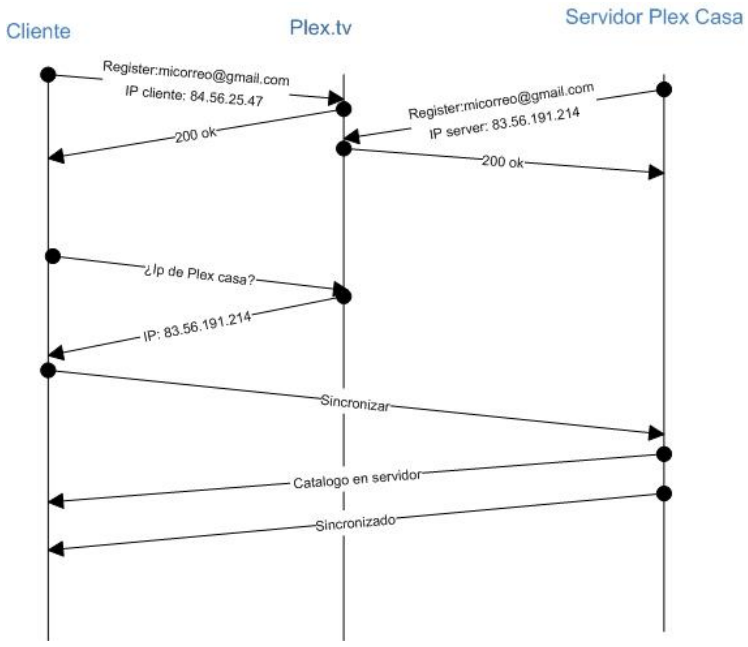



**Fig 2. 9** Diagrama PLEX Tv

- **Owncloud** será la *jail* encargada de crear una nube auto hospedada, facilitando el acceso remoto al usuario, permitiendo la gestión de usuarios, permisos, cuotas y datos de una manera muy simple e intuitiva. Así como transferencia de ficheros entre cliente servidor.

- **Headphones , Lazylibrarian , Couchpotato y Sickbeard** serán las *jails* encargadas de descargar metadatos de las paginas oficiales de música, libros, series y películas utilizándolos para auto organizar Autores y sus canciones ,películas y series respectivamente según el criterio del usuario. Además será la encargada de descargar los Torrents o los archivos NZB de la mayor calidad posible y transmitirlos a *Nas* para que otras *jails* puedan utilizarlas y gestionarlas.

- **Sabnzbd** es un lector de noticias en binario *Open Source* y escrito en Python , es totalmente gratis y funciona prácticamente en cualquier OS, Sabnzbd convierte el uso de Usenet fácil y ágil , evitando cualquier tipo de interacción humana ya que las otras *jails* se encargaran de transmitirle el fichero NZB.

- **Transmission** es un cliente *Open Source* de ficheros Torrent centralizado por red, es decir, cualquier dispositivo podrá acceder a transmisión utilizando un navegador y una url , a partir de su interfaz web se podrán gestionar los Torrents, que a su vez serán transmitidos a transmisión por otras *jails*.

  Transmission es proyecto basado en voluntarios. Transmission tiene la ventaja de que se integra con el sistema operativo convirtiéndolo en nativo, además es uno de los clientes p2p que menor cantidad de CPU utilizan. Tiene la ventaja de que es un cliente p2p centralizado en la red, accesible desde dentro de la red local así como desde fuera con cualquier dispositivo.

- **Virtual Box hypervisor**, muchas empresas utilizan servicios virtualizados debido a la versatilidad que da y a la facilidad de gestión, administración y archivado de máquinas virtuales, ya que la infraestructura requerida es mucho menor y se pueden aprovechar los recursos de una manera muchísimo más óptima, virtual box *hypervisor* también correrá en FreeBSD dentro de un *jail* de FreeNAS pero ofrecerá la posibilidad de crear otras máquinas virtuales basadas en otros SO, y todo ello de manera centralizada, es decir, se utilizara una interfaz web para administrar , archivar , encender y apagar las máquinas virtuales y un visor VNC para poder visualizar cualquier maquina e interactuar con ella como si de una maquina local se tratara con la ventaja que se podrá acceder desde cualquier lugar

- **OpenVPN** server es el encargado de brindar al escenario empresarial una manera de permitir a sus empleados el acceso al servidor como si de local se tratase permitiendo el acceso a documentos y la comunicación con toda la red empresarial desde cualquier ubicación .

- **autoridad certificadora** es una entidad de confianza que es la encargada de emitir y revocar certificados digitales, utilizados para

firmar, cifrar y/o verificar la identidad de un dispositivo/cliente, esta autoridad(Fig 2. 10) será necesaria para auto firmarse el certificado del servidor FreeNAS. Así como para verificar la autentidad de los clientes la identidad de los usuarios.

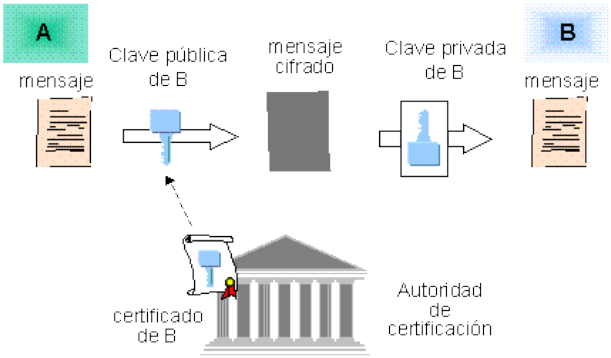



**Fig 2. 10** Esquema autoridad certificadora

# CAPÍTULO 3. ESCENARIO EMPRESARIAL

## 3.1 Idea y Background del escenario

El escenario empresarial simula una pequeña empresa de unos 70 empleados, es decir una S*tartup*, esta empresa ha crecido en los últimos años y su necesidad de infraestructura ha crecido con ella. La empresa tiene unos objetivos claros: Dar de baja maquinaria antigua y virtualizar sus servicios para poder asignar recursos a voluntad según el crecimiento de la empresa, quiere además asegurar que los datos que se usan diariamente tengan una alta redundancia ,incluidas maquinas virtualizadas. Además han abierto nuevas oficinas repartidas por el mundo y necesitan que estas oficinas puedan conectar con la sede central y trabajar con los documentos y maquinas virtuales como si de local se tratase, También y debido a su falta de capital , no pueden invertir en toda la seguridad que les gustaría, por tanto han decidido que la mejor opción y para evitar el envío por internet, es trasladar físicamente el disco o archivarlo en un almacén con la información cifrada para evitar cualquier tipo de violación de seguridad

La empresa quiere virtualizar el servidor de correos que corre en una maquina muy antigua que ya no puede servir tantas peticiones, ni puede ser actualizada a una nueva versión del *software* de correos debido a la falta de recursos, por tanto, han decidido que utilizando la virtualización existente en FreeNAS, virtualizar su antiguo servidor de correos para correrlo utilizando el *hypervisor* de *virtual box*, de esta manera pueden adaptar de manera cómoda la cantidad de recursos que la maquina utiliza
Para conectar entre oficinas la solución que han decidido es la creación de un servidor VPN , de esta manera se aseguran el acceso a todos los documentos por cualquier oficina.

También y debido a que muchos usuarios utilizan Linux,mac's ,tablets móviles necesitan asegurar un acceso universal, es decir que cualquier SO pueda acceder a la información

Debido a la falta de infraestructura de seguridad , el administrador de la empresa ha decidido que la mejor manera de evitar accesos no autorizados al servidor , es eliminando posibles *passwords* y permitir únicamente el acceso por SSH al servidor utilizando un certificado digital y una clave privada generados por la autoridad certificadora creada en FreeNAS.

Para asegurar una alta redundancia han decidido que la mejor manera es la creación de un pool *raídZ1* , *snapshots* diarios y cada 10 días el análisis usando la herramienta *scrub* que posee FreeNAS para corregir errores de escritura, ya que en un pool RAIDZ1 la pérdida de un disco + errores de escritura en datos de paridad puede ser fatal.

### 3.1.1Ejemplos de posibles fallos

A continuación se describen tres ejemplos de posibles fallo cotidianas con las que se puede encontrar una empresa.

Por tanto, cuales son las preguntas que cualquier empresa debe hacerse:

Como protejo/ recupero la información en caso de desastre?

Puede acceder alguien sin autorización a mis datos?

Qué mecanismos aseguran una mitigación de daño en caso de pérdida y un plan de recuperación?

Que puedo perder sin que sea critico para mi empresa?

Cuáles son los mayores riesgos que pueden poner en duda mi continuidad como empresa?

Como se adecua tu  sistema de recuperación en momentos en el que el trabajo está en un pico?

#### 3.1.1.1 Ejemplo 1

En este ejemplo se encuentra un escenario en el que una empresa tiene un proyecto con un cliente muy importante en el que almacena sus datos en un servidor y no cuenta con una estructura de protección de datos suficiente. Es decir, no cubre replicación ni *backups* ni cifrado. Dicho proyecto sufre una pérdida de un disco duro de terabytes de capacidad perdiendo el trabajo de meses de este como de muchos otros datos importantes.

Eso provoca que no se cumplan los plazos acordado con el cliente ya que hay que volver a empezar debido a la perdida de información. Con el uso de FreeNAS  se evita esta situación ya que ofrece la posibilidad de almacenar los datos en RAID Z1, Z2 , Z3 o mirror, guardando la información con paridad de bit, permitiendo la reconstrucción de la información perdida.

#### 3.1.1.2 Ejemplo 2

Un servidor, en este caso el *Domain controller* deja de funcionar en una mediana empresa, provocando que ningún usuario del dominio pueda *logear* en ningún PC de la organización , eso provoca que ningún usuario del dominio (empleados y jefes) puedan usar sus ordenadores y trabajar generando al minuto miles de euros en perdidas. Con FreeNAS se evitaría eso, ya que permite generar *backups* usando *cronjobs*.

#### 3.1.1.3Ejemplo 3:

La empresa sufre un ataque malintencionado y se pierde toda la información del servidor Exchange, al perderse el servidor , toda la información de cuentas de correo, emails, contactos se pierde. Eso puede provocar el caos dentro de la organización

## 3.2 Requisitos

Los requisitos planteados a la hora de diseñar el proyecto son los siguientes:

- Un sistema seguro y fiable
- Seguridad física y redundancia en los datos empresariales
- Cifrado de disco
- Eliminación de *passwords* para administración del servidor
- Virtualización de la infraestructura , incluye maquinaria antigua.
- Acceso Universal
- Asignación de recursos a a las maquinas virtualizadas
- *Backups y Snapshots*
- Inversión de capital mínima
- Sistema de permisos potente
- Transparente para el usuario corriente
- Alta velocidad de escritura y lectura
- Capacidad de servir los datos sin saturaciones.

En este escenario se cumplen los siguientes objetivos: Acceso Universal, Seguridad de datos y acceso *, data resilience* , acceso a la red local desde cualquier ubicación y virtualización.

Este escenario se ha simulado utilizando 3 discos físicos diferentes, la razón de esta elección es intentar acercarse lo máximo a un escenario real.

Los discos son de 5400 rpm , 7200 rpm y un SSD ,el ancho de banda del disco de 5400 llega a un máximo de 28 MB/s debido a su antigüedad, por tanto debido a las características de ZFS y de que el sistema RAIDz1 obliga a que todos los discos tengan la misma velocidad y el mismo espacio de almacenamiento , todas las pruebas de aquí en adelante estarán limitadas por esta característica física de la maquina que está haciendo la simulación. El escenario cuenta con 3 CPU’s dedicadas a la simulación así como 9,8 GB de RAM

## 3.3 Diseño del escenario

El diseño escogido es el mostrado en **Fig 3. 1** Escenario empresarial y la herramienta que más se adapta para los requisitos de la empresa es FreeNAS , que es la encargada de cumplir cada uno de los requisitos expuestos anteriormente.

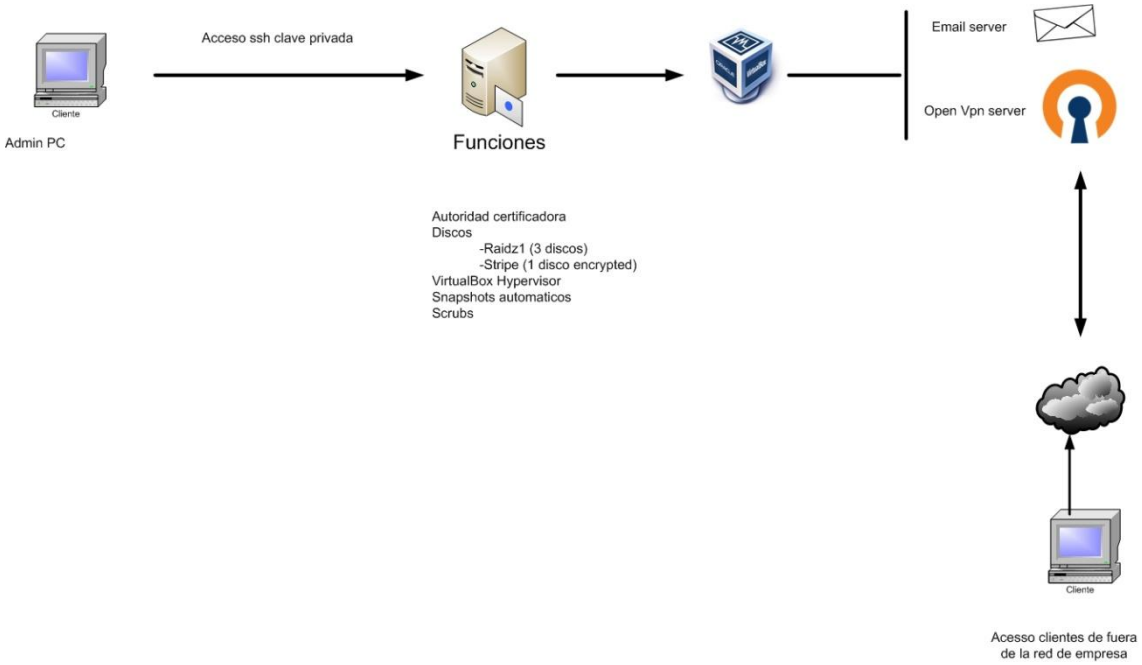



**Fig 3. 1** Escenario empresarial

## 3.4 SSH con “public key authentification”

Es un protocolo de red criptográfico que permite autentificarse remotamente y operar de manera segura en una red insegura.
SSH provee de un canal seguro utilizando una arquitectura cliente-servidor. Generalmente para el acceso remoto hay que suministrar un usuario y un *password* pero la variante implementada en este proyecto funcionara ligeramente diferente. Este método utilizara una clave privada que es la que sirve como autentificación , el método funciona enviando una firma creada con la clave privada del usuario , el servidor ara un *check* para ver que la signatura es válida , utilizando la clave pública del usuario que estará previamente en el servidor, la clave privada se guardara en el host, de esta manera se aumenta la seguridad y se reduce uno de los mayores riesgos que es el factor humano, ya que no sirve el *password* del usuario para *logear* remotamente.

Ya que la mayoría de servidores suelen estar en unas salas climatizadas y aisladas , el acceso por SSH es una forma de evitar desplazamientos para llegar la ubicación física del servidor, por tanto es necesaria una alta seguridad.

En las opciones de configuración de cuenta de administrador en FreeNAS , se almacena la clave publica de acceso por SSH Fig 3. 2 , por tanto , cuando un usuario acceda deberá suministrar su *login* y una firma con su clave para que el servidor verifique

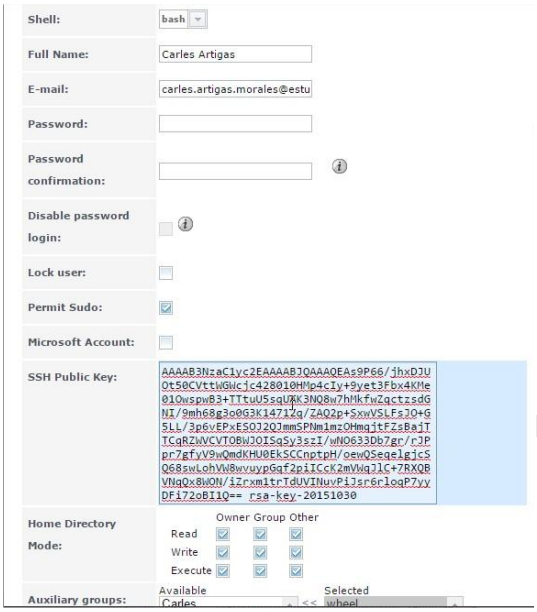


**Fig 3. 2** Introducción Clave publica

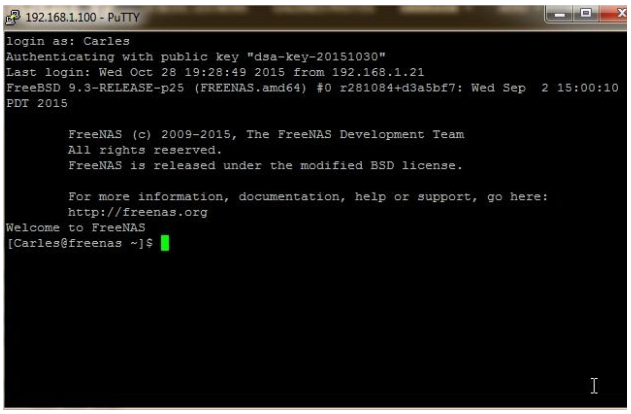


**Fig 3. 3** Acceso con clave

Una vez hecho esto podremos conectar con el servidor usando simplemente nuestra clave privada, que es única por cada cliente .Fig 3. 3

## 3.5 OpenVPN

Actualmente muchas empresas tienen empleados viajando por el mundo o visitando clientes, estos empleados muchas veces necesitan acceder a documentos y/o carpetas en la empresa, además muchas empresas tienen oficinas repartidas por el mundo y tienen documentos que deben ser guardados en los servidores de la sede, OpenVPN ofrece una forma de poder conectarse fácilmente a la red de la empresa y trabajar como si en local se tratase.

La maquina escogida para hacer de servidor , es una *jail* de FreeNAS estándar basada en FreeBSD, la razón de la elección es que es la forma que consume menos recursos que ofrece FreeNAS , ya que si estuviera basada en otro SO habría que emularlo con el virtual box de FreeNAS basado en FreeBSD, por tanto de esta manera evitamos un nivel de virtualización más.

Esta máquina con IP 192.168.1.240 es la encargada de hacer de servidor, es importante recalcar que para que el cliente se conecte con el servidor deben de tener tanto servidor como cliente una clave y un certificado firmado por la autoridad certificadora que está dada de alta en FreeNAS.

| | |
|---|---|
| Ca.crt | certificado de la autoridad |
| Cliente.key | clave de cliente |
| Cliente.crt | certificado de cliente |
| Cliente.conf | archivo de configuración del cliente |
| Server.crt | certificado servidor |
| Server.key | clave de servidor |
| Server.conf | archivo de configuración del servidor |

También es necesario que el *router* y el servidor FreeNAS tengan un ruta estática que permita acceder a la red interna , ya que el servidor está configurado para dar IP's con la subred del rango 10.8.0.0/30 Fig 3. 4, por tanto tanto FreeNAS como el *router* deben tener constancia de la existencia de esa red para poder *enrutar*.

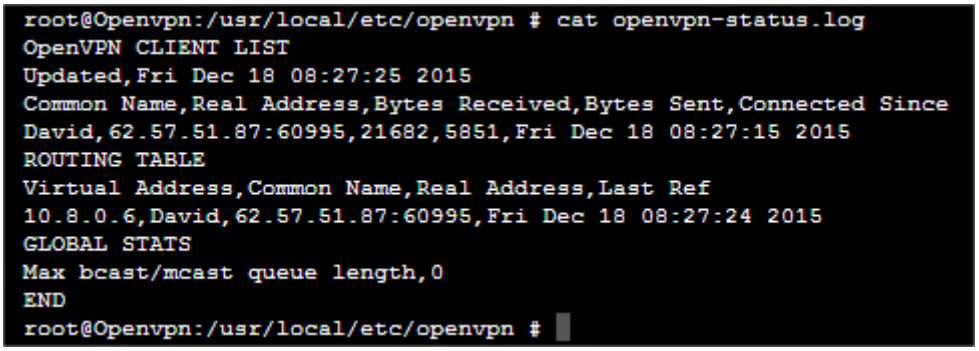



**Fig 3. 4** Log de conexión al servidor

El cliente debe tener a su vez , un cliente que le permita la conexión, los propios creadores de openVPN ofrecen distintos clientes para diferentes SO.

Log simplificado de conexión usando un cliente Windows.

*Data Channel Encrypt: Cipher 'AES-128-CBC' initialized with 128 bit key*

*Data Channel Encrypt: Using 160 bit message hash 'SHA1' for HMAC authentication*

*Data Channel Decrypt: Cipher 'AES-128-CBC' initialized with 128 bit key*

*Data Channel Decrypt: Using 160 bit message hash 'SHA1' for HMAC authentication*

*Notified TAP-Windows driver to set a DHCP IP/netmask of 10.8.0.6/255.255.255.252*

*Initialization Sequence Completed*

## 3.6 Virtual Box hypervisor

FreeNAS nos ofrece *jails* basadas en FreeBSD , pero ¿qué pasa si necesitamos maquinas basadas en otro sistema operativo?, la respuesta es que FreeNAS nos ofrece una *jail* de virtual box para virtualizar otros sistemas operativos y todos ellos centralizados desde la red, en otras palabras , se puede acceder a ellas remotamente como si de una *jail* mas se tratara o de manera muy parecida a la que ofrece *Vmware EsX* . Con el termino centralizado me refiero , que desde cualquier lugar de la red local se puede acceder a la maquina virtual, pero la diferencia principal con el sistema EsX de vmware, es que hay una maquina dedicada a organizar las otras , además de moverlas entre maquinas clientes, característica con la que no cuenta virtual box *hypervisor*.

Esta *jail* es la encargada de hospedar el servidor de correo, la razón es, muchas empresas suelen tener maquinas físicas dedicadas a ciertos servicios, como es el correo, esta al ser física, puede sufrir cualquier tipo de catástrofe, sea que se rompa un componente, que el servidor deje de funcionar por tocar la configuración o la perdida de algún buzón de voz, además que el uso de recursos no es eficiente, de esta manera se asegura una eficiencia mas óptima. Por tanto si la máquina se virtualiza y se introduce dentro de FreeNAS , obtendrá todas las ventajas que FreeNAS ofrece, gracias a esto , se acabaron posibles pérdidas, ya que la función de *snapshots*, la redundancia en discos eliminan sustancialmente los riesgos, además el hecho de virtualizar el sistema permite clonar las maquinas tantas veces como queramos por si en algún momento una de ellas dejará de estar online , un clon puede sustituir la función.

También convierte el servidor de correo, en una máquina independiente de la infraestructura , por tanto, si la máquina se quiere migrar a una infraestructura más actual, se puede virtualizar una maquina física y instalarla dentro de la *jail* de virtual box, además si queremos añadir más recursos dedicados a la maquina siempre tendremos la posibilidad de hacerlo.

Esta máquina será utilizada a su vez por FreeNAS para enviar notificaciones al administrador de los diferentes procesos que el servidor lleve acabo, de esta manera si en algún momento hubiese una incidencia , como que un disco dejase de funcionar, o un servicio se parase , FreeNAS automáticamente enviaría un correo a la cuenta del administrador (Fig 3. 5)


**Fig 3. 5** Envió a la cuenta de mail Administrador

Ejemplo de importación y acceso

Una vez el archivo .ova ha sido creado, importar la maquina es tan sencillo como guardarla en un Dataset que este montado en la *jail* de Virtual box y importarla Fig 3. 6

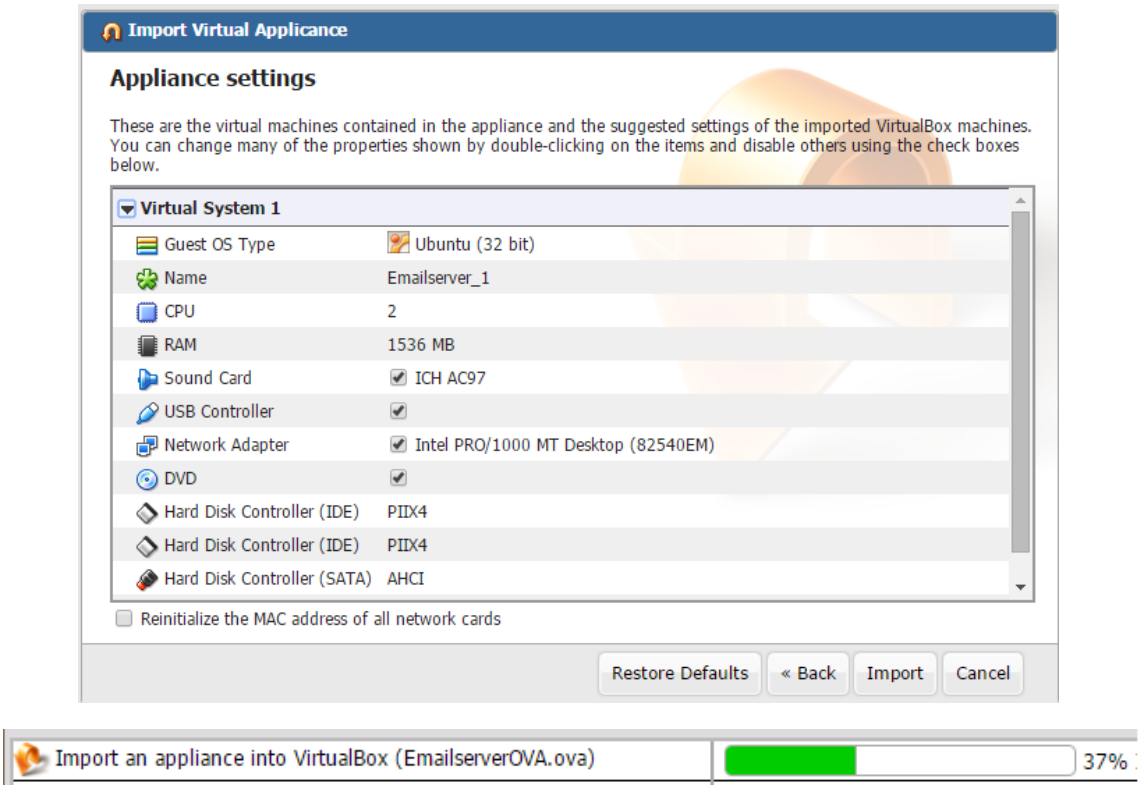



**Fig 3. 6** Importación de una maquina virtual

Cuando la maquina haya sido importada (Fig 3. 6), habrán dos maneras de acceder a ella, la primera es utilizando SSH que solo podrá hacerse si la maquina tiene configurada una IP o utilizando un visor de VNC como VNC-Viewer(Fig 3. 7), en el cual para acceder hay que poner la IP de la *jail* de Virtual box así como el puerto que identificara a la maquina a la que queremos acceder, en otras palabras , la IP siempre será la misma tengamos 10

máquinas que 1 , lo que cambiara será el puerto que es lo que identifica que máquina queremos acceder

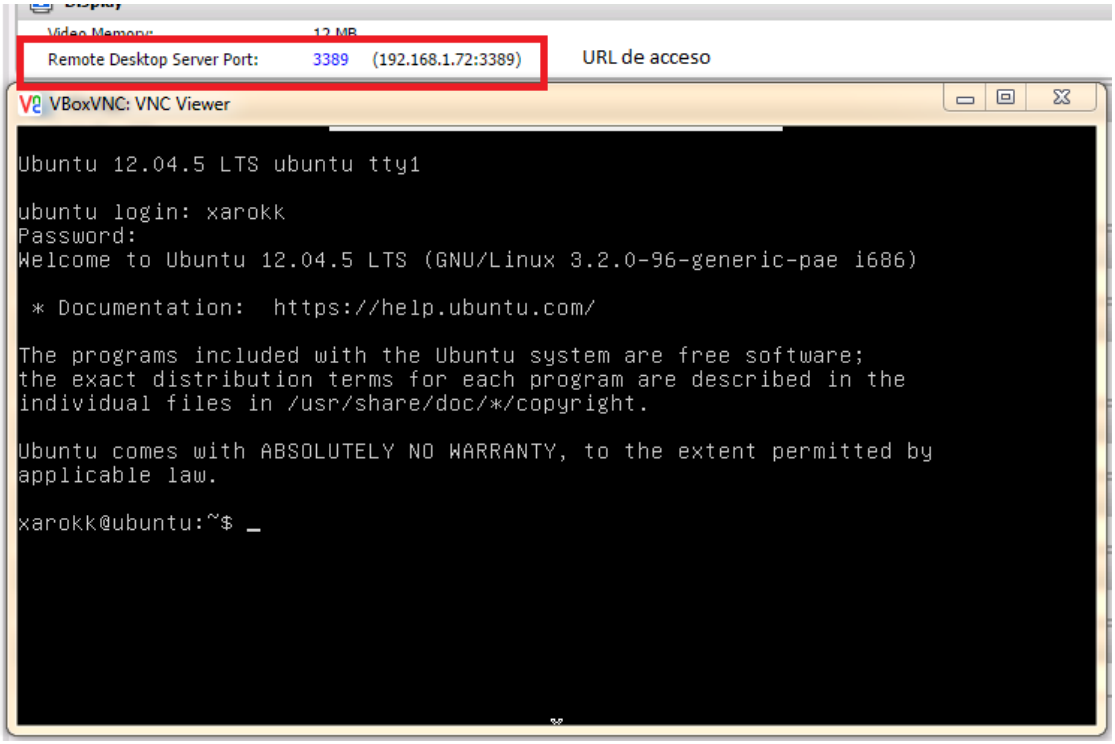



**Fig 3. 7** Acceso usando Vnc Viewer con url 192.168.1.72:3389

## 3.7 Almacenamiento , IO en discos y Rendimiento

En este escenario que es de ámbito empresarial, es necesaria una alta redundancia en la información, FreeNAS como hemos visto anteriormente ofrece distintas maneras de organizar los discos para proporcionar la redundancia antes dicha. En este escenario la empresa consta de 4 discos SAS, estos discos son muy caros si los comparamos con los discos Sata, pero a cambio tienen un rendimiento mucho mayor y pueden estar encendidos 24x7 , que es lo que a cualquier organización que utilice servidores le interesa.

3 de estos discos los quieren dedicar a guardar información poco sensible pero importante , ya que es necesaria para el trabajo del día a día, el disco faltante en cambio se quiere usar para guardar la información más sensible de la empresa, información vital de la cual la organización depende, por tanto se quiere cifrar para evitar que nadie pueda leer la información y solo el que este autorizado pueda descifrarla, ya que la información que posee el disco es tan importante que no se mandara jamás por internet, por tanto cuando sea necesaria mover la información de lugar , se desenganchará el disco del servidor y se moverá a otras instalaciones de la empresa.Este disco se desenganchara del servidor y se trasladará físicamente a otras instalaciones

que posee la empresa, donde se volverá a insertar dentro de una maquina FreeNAS .De igual manera los discos se encriptaran para almacenarlos.

Por tanto en el ámbito empresarial lo que nos interesa es maximizar el rendimiento, minimizar los costes y maximizar el espacio disponible, entonces ¿Cual es la mejor manera de organizarlos? Para ello comparo debajo las prestaciones para 3 discos, ya que el tipo de organización dependerá de la cantidad de discos disponibles.

Para acercarse a una simulación real y debido a la falta de infraestructura donde implementar la simulación los discos serán de 30 gb pero en un ejercicio de imaginación diremos que son de 1 TB

Los cálculos de espacio se han hecho utilizando la siguiente URL:

Drive size: 1 TB
Number of drives: 3

**Mirror**

| | TiB | TB | % |
|---|---|---|---|
| Drive Size | 0.91 | 1 | N/A |
| Total Parity Space | 1.364 | 2 | 66.67 |
| Total Data Space | 0.91 | 1 | 33.33 |
| Total RAID Space | 2.729 | 3 | 100 |
| Minimun Free Space | 0.179 | 0.197 | 19.68 |
| Usable data space | 0.716 | 0.787 | 78.72 |

**Tabla 3. 1** Calculos mirror

**RAID Z1**

| | TiB | TB | % |
|---|---|---|---|
| Drive Size | 0.91 | 1 | N/A |
| Total Parity Space | 0.91 | 1 | 33.33 |
| Total Data Space | 1.819 | 2 | 66.67 |
| Total RAID Space | 2.729 | 3 | 100 |
| Minimun Free Space | 0.358 | 0.394 | 19.68 |
| Usable data space | 1.432 | 1.574 | 78.72 |

**Tabla 3. 2** Calculos RAIDz1

**Stripe**

| | TiB | TB | % |
|---|---|---|---|
| Drive Size | 0.91 | 1 | N/A |
| Total Parity Space | 0 | 0 | 0 |
| Total Data Space | 2.729 | 3 | 100 |
| Total RAID Space | 2.729 | 3 | 100 |
| Minimun Free Space | 0.537 | 0.59 | 19.68 |
| Usable data space | 2.148 | 2.362 | 78.72 |

**Tabla 3. 3** Calculos stripe

Como se puede ver en las tablas: Tabla 3. 1,Tabla 3. 2 yTabla 3. 3 , con 3 discos disponibles la mejor organización es RAIDZ1 ya que nos da paridad a diferencia de stripe, con lo que podemos perder un disco entero y nos da más espacio usable que mirror , ya que mirror con 3 discos es aproximadamente el inverso de RAID z1 cuando comparamos espacio de paridad/data.

Además se asignan cuotas a los volúmenes para que nunca superen el 80% del espacio usable, recordemos que los sistemas basados en ZFS necesitan un espacio mínimo disponible ya que en el momento en el que el espacio del disco cae por debajo del 20% es decir, tenemos ocupado el 80% el rendimiento decae (Fig 3. 8). Por tanto el rendimiento del volumen cae de manera drástica, en otras palabras , la velocidad de escritura/lectura disminuye. La grafica inferior muestra como cae el rendimiento conforme vamos llenando el disco.

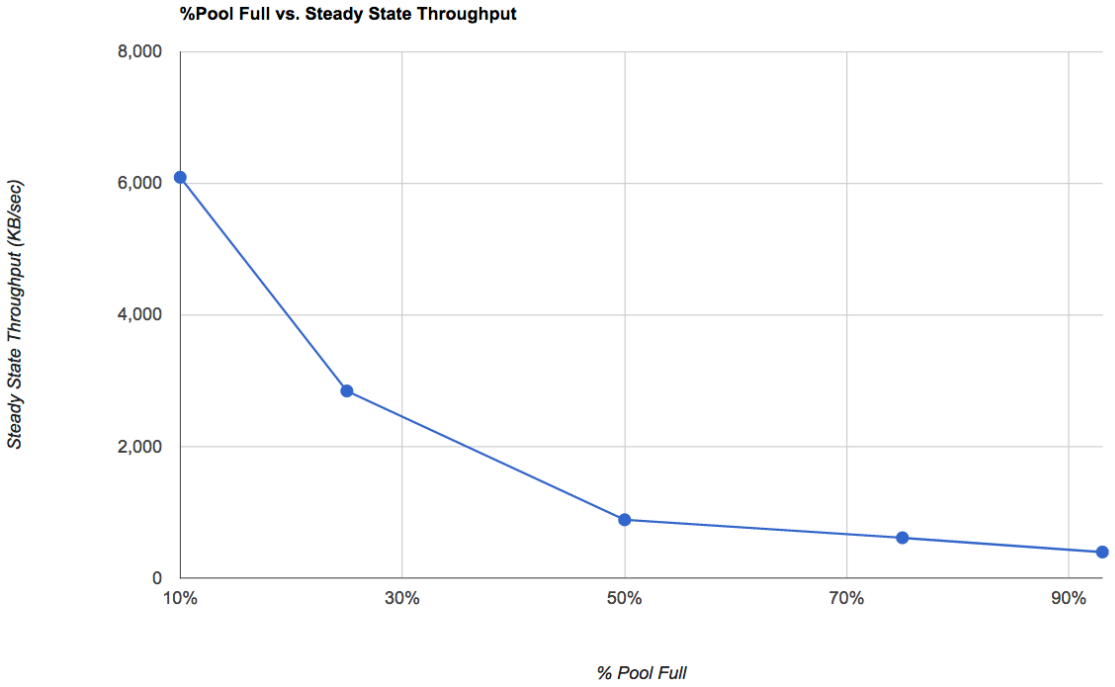



**Fig 3. 8** Espacio vs ancho de banda

La razón de esta caída de rendimiento  es debida a la fragmentación que sufren los discos debido al copy-on-write además del mecanismo utilizado por ZFS para descubrir bloques libres en el disco y liberar antiguos.

Al principio la escritura en el disco sigue un proceso secuencial de la siguiente manera

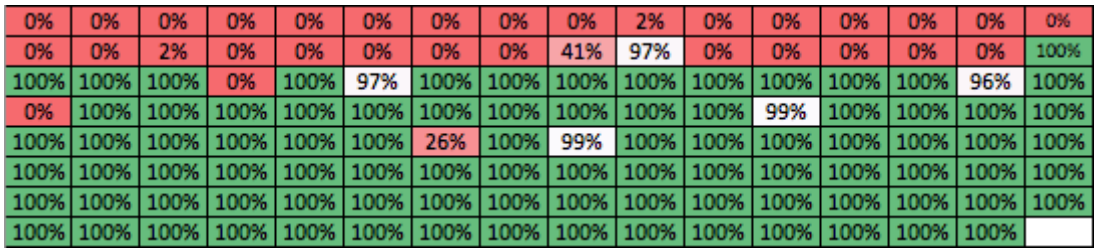



**Fig 3. 9** Copia secuencial

Pero conforme vamos modificando archivos, FreeNAS va liberando trozos de bloques(Se modifican los punteros, es decir, apuntan a la nueva ubicación de la información ) , debido a la característica de copy-on-write, por tanto conforme el tiempo pasa, el disco está cada vez mas fragmentado.

Tanto en la imagen Fig 3. 9 como en la imagen Fig 3. 10 el espacio usado es el 25 % , la diferencia es que en la Fig 3. 9 está toda la información junta en los bloques originales y en la Fig 3. 10 los punteros señalan a la nueva posición de los datos por eso el espacio usado por bloque se ha ido redistribuyendo a nuevas posiciones. Hay que tener en cuenta que la información antigua , sin modificar queda presente en el disco.

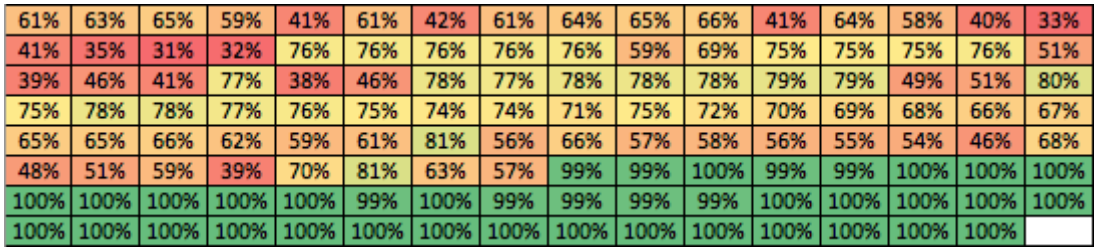



**Fig 3. 10** Copy-on-write

Si tuviésemos un escenario con más discos disponibles , la organización de los discos cambiaria , ya que con 4 y 5 discos los creadores de FreeNAS nos recomiendan RAIDz2 y RAIDz3 respectivamente, aunque todo depende de lo que estemos dispuestos a perder y cuanto espacio queramos aprovechar .

El cuarto disco se destina a cifrado, pero además se quiere disponer del máximo espacio del disco para  por tanto se utiliza stripe , al ser un  disco solo no tenemos posibilidad de organizarlo de otra manera.

La encriptación en FreeNAS funciona por disco es decir, es diferente a los cifrados de archivos, lo que se está cifrando es el disco entero a nivel de bloque, es transparente para la capa aplicación  y es útil para usuarios que quieran guardar información sensible y quieran poder mover físicamente los discos de emplazamiento.

Mientras el disco y la clave estén intactos, el sistema se puede desencriptar, por tanto la clave debe ser guardada en un lugar seguro y protegida por una *passphrase.*

La encriptación va por volúmenes es decir, si tienes x discos dentro de un volumen , todos los discos de ese volumen tendrán la misma clave , incluidos los Datasets dentro del volumen. Sin clave, la información se da por perdida.

Hay que recalcar que la información en la RAM esta sin encriptar por tanto existe una vulnerabilidad ya que los sistemas basados en FreeBSD suelen mantener siempre la RAM hasta que es necesaria para otro uso, por tanto si un archivo ha sido accedido y se ha guardado en la RAM, alguien con los conocimientos técnicos suficientes , podría recuperar parte de la información.

Una diferencia importante es que la encriptación en FreeNAS está diseñada para evitar robos (sustracción de discos) pero en ningún caso protege de accesos no autorizados por software debido a esto los permisos deben estar bien definidos y que solo pueda acceder quien está autorizado.

Ejemplo encriptación

Una vez creado el volumen encriptado, creada la clave de acceso y asignada la *passphrase*, el volumen tendrá la opción de cerrarse, una vez esta en estado *locked* ( Fig 3. 11) , nadie puede usar el volumen sin insertar primero la *passphrase* y la clave de encriptación.

| Name | Used | Available |
|---|---|---|
| ◢ Empresa | 1.1 GiB (1%) | 82.4 GiB |
| ▸ Empresa | 735.7 MiB (1%) | 53.2 GiB |
| Encriptado | Locked | Locked |

**Fig 3. 11** Volumenes escenario

Tanto si el volumen es extraído como si el volumen es cerrado para desencriptar el proceso es parecido, con la diferencia en que si se extrae primero es necesario importar el volumen y si se cierra simplemente es poner las dos claves en la GUI de FreeNAS . Hay que decir que a mayor cantidad de información en el disco más tiempo tardará en cambiar de estado *locked* a *unlocked* y viceversa ( Fig 3. 12)

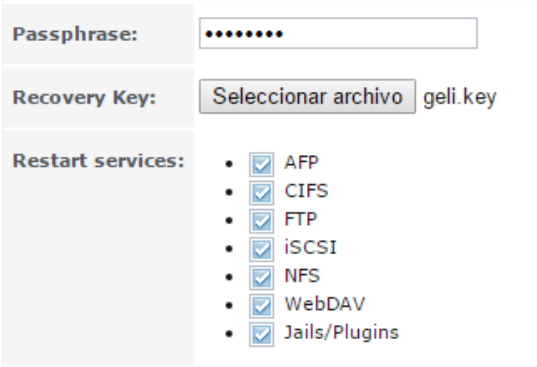



**Fig 3. 12** Desencriptación disco Encriptado

Ejemplo de recuperación de un disco roto y recuperación usando *Snapshots*

Para simular la pérdida de un disco, se suprime el archivo virtual relacionado con el disco duro del volumen Empresa, este volumen es el que se corresponde con la agrupación de los 3 discos en RAIDZ1 antes dicho.

Cuando el disco deja de ser accesible por FreeNAS , este envía un aviso al administrador por email utilizando el servidor de correos previamente configurado, Además salta un alerta en la GUI (Fig 3. 13) de FreeNAS


**Fig 3. 13** Alarma de aviso de pérdida de disco

Por tanto aunque el disco haya sido destruido, el sistema será capaz de seguir funcionando con 2 discos y recuperar toda la información aunque uno de ellos falte y información continuará siendo accesible, pero a partir de ahora sin redundancia ya que falta el tercer disco. Si hubiésemos tenido el caso de que hubiera habido un cuarto disco en el volumen en modo *spare*, la reconstrucción de los datos hubiese sido transparente para el administrador , es decir , FreeNAS al detectar el fallo automáticamente hubiera utilizado el cuarto disco para sustituir al faltante, aunque hubiese avisado del proceso por correo al administrador.

Cuando sustituimos el disco por un tercer disco manualmente , automáticamente se copiaran todos los archivos y los datos de paridad para volver a reconstruir el disco faltante en el nuevo disco.Fig 3. 15

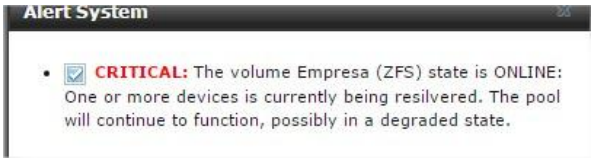



**Fig 3. 14** Alarma que avisa de que hay un disco nuevo pero sigue degradado

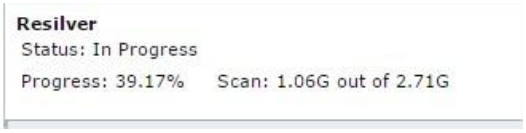



**Fig 3. 15** Proceso de recuperación

Como se puede ver en la imagen de la Fig 3. 16, los dos discos que seguían en el sistema están transmitiendo la información al disco recién añadido

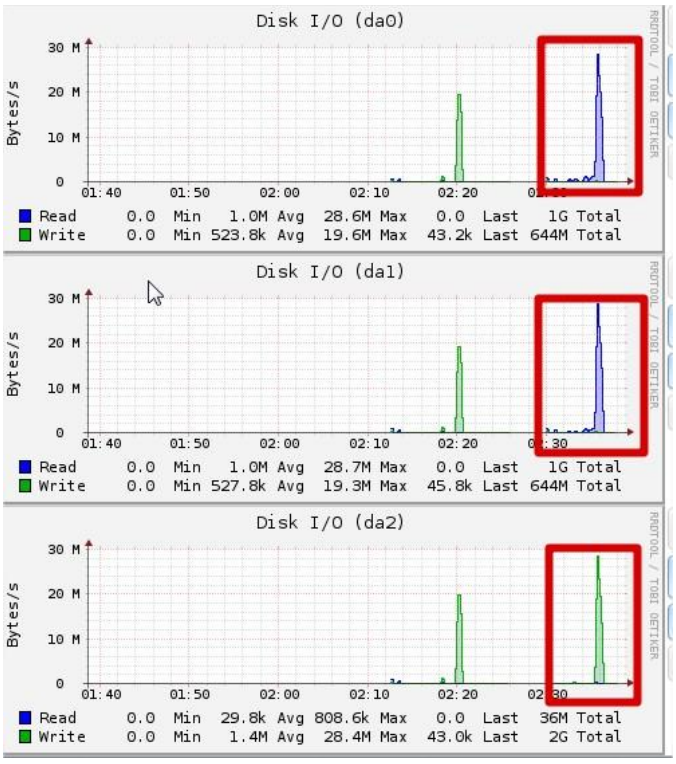



**Fig 3. 16** Gráfica en la que se observa la lectura de dos discos y la escritura en el tercero

Para añadir un nivel mas de seguridad en los datos, se configuran *snapshots* periódicos , que serán instantáneas en un momento dado del estado de los discos, estas instantáneas y debido al funcionamiento de ZFS , serán de rápida recuperación , es decir, si un usuario elimina algún dato empresarial importante de manera maliciosa o sin querer, el propio tipo de funcionamiento ZFS de copy-on-write y de bloques explicado anteriormente hace que el sistema pueda recuperar una instantánea en cuestión de segundos, ya que no es reescribir la

información a partir de un archivo de diferencias , si no , cambiar la posición donde señalan los punteros , ya que ZFS , nunca elimina los datos , ni sobrescribe, siempre utiliza bloques libres a menos que estos se hayan acabado.

Una ventaja de RAIDZ1 es que utiliza los discos simultáneamente, en otras palabras, escribe y lee la información de los tres discos a la vez, por ejemplo, en una trasferencia de archivos de un PC al servidor


**Fig 3. 17** Transferencia

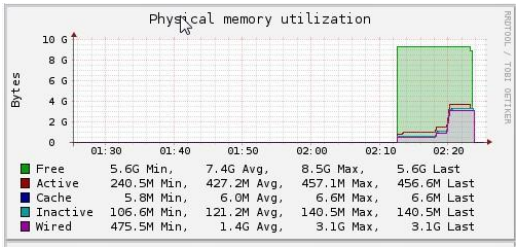



**Fig 3. 18** Uso RAM

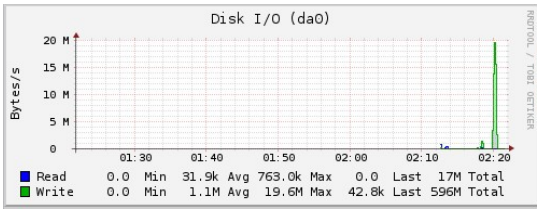



**Fig 3. 19** Escritura disco da0

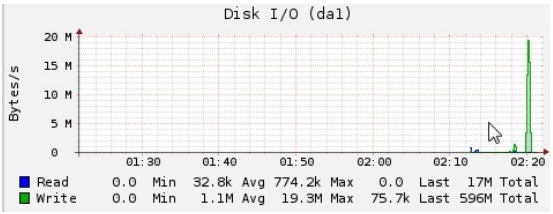



**Fig 3. 20** Escritura disco da1

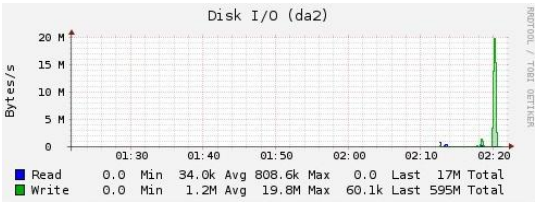



**Fig 3. 21** Escritura disco da2

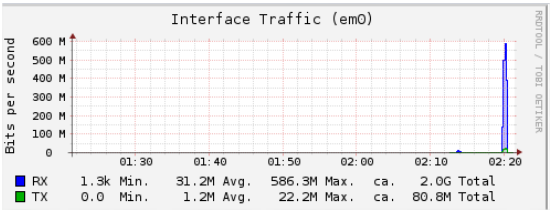



**Fig 3. 22** Recepción NIC

**Fig 3. 23** Diferentes graficas de la transferencia

Como se puede observar en Fig 3. 17 la transferencia tiene un ancho de banda de 76.8 MB/s que se corresponde con la suma de escritura en los 3 discos ( Fig 3. 19, Fig 3. 20 y Fig 3. 21)aproximadamente, Además la recepción de datos también encaja con el *troughput* de la interfaz de red em0 (Fig 3. 22). da0 + da1 + da2 = em0 / 8 bits

## 3.8 Costes escenario

Para que este escenario vaya fluido es necesaria una gran cantidad de RAM, los propios desarrolladores de FreeNAS recomiendan que por cada Terabyte

de espacio haya dedicada 2 GB de RAM, pero esto sería solo en el caso de que el sistema solo se usará como servidor de archivos, como además se implementa en la infraestructura una *jail* de virtual box y una de FreeBSD con el servidor openVPN la demanda de RAM es aún mayor .

Este sería el coste según las dimensiones de la empresa

Para una empresa de entre 50 a 150 trabajadores con 8 TB de almacenamiento.Tabla 3. 4

| **Componente** | **Precio** |
|---|---|
| HP ProLiant ML150 Gen9 E5-2620v3 16GB H240 | 2,339.99$ |
| Ram =16gb RDDIM | |
| Procesadores = 2 procesadores 6 cores | |
| Hitachi Ultrastar 7K4000 4TB SAS Hard | 264x2$ |
| Total | 2867.99 |

**Tabla 3. 4** Precio infraestructura

Para una empresa de 1000 trabajadores con 60 TB de almacenamiento ver Tabla 3. 5

| Componente | Tipo |
|---|---|
| HP ProLiant BL660c Gen9 E5-4620v3 128GB-R 4P Server Blade: | 19.800 |
| Ram = 128 GB | |
| Procesadores = 10 procesadores con 8 cores a 2.0Ghz | |
| 10 NICS = 10GB/s | |
| Dell Storage MD1400 and MD1420 Direct-Attached Storage | 4663 |
| Hitachi Ultrastar 7K4000 4TB SAS Hard Drive x15 | 264 x15 |
| Total | 28.423 $ |

**Tabla 3. 5** Precio infraestructura 1000 trabajadores

# CAPITULO 4 HOME SERVER

## 4.1 Idea y Background del escenario

Hoy en día y debido a diferentes leyes como la ley de economía sostenible (ley SINDE) , no se puede acceder con la misma facilidad que hace 3 o 4 años a contenidos protegidos con derechos de autor, sea películas , sean series o música. Debido a estas leyes muchas páginas han sido cerradas y sus creadores perseguidos por la ley como si de delincuentes se tratasen . Este escenario ha sido desarrollado para facilitar el acceso a todas aquellas familias/personas que siguen consumiendo contenido audiovisual protegido con derechos de autor y no desean pagar los altos precios por contenidos que solo verán 1 vez y que consideran que el canon pagado por dispositivos de reproducción/grabación ya es suficiente para el acceso libre a estos contenidos.

Por tanto este escenario se centra en la creación de un sistema totalmente automatizado en que la interacción con el usuario sea mínima y que consiga obtener todos estos contenidos deseados por el usuario de la forma más rápida y fácil posible. Además brindando un acceso fácil, universal y rápido a todos los archivos contenidos en el servidor y poder brindar independientemente de la ubicación del usuario video a demanda de aquellos contenidos previamente descargados con el sistema de automatización.

También en muchos hogares hay ya smart TV , que permiten conectarse a internet y acceder al servidor de casa, este escenario facilitaría la reproducción de contenido multimedia en el caso de ser poseedor de dicha televisión, además ya que el acceso al servidor multimedia es universal y desde cualquier lugar , si tenemos más de una residencia y en la segunda residencia se posee una smart TV o cualquier dispositivo con acceso a internet que sea capaz de abrir un navegador o usar la app de PLEX tv podrá conectarse remotamente y reproducir los videos a demanda.

Este servidor a su vez tendrá su propia nube de hospedaje para facilitar hacer *backups* de los archivos multimedia de los dispositivos móviles del usuario como son las fotos y videos del móvil.

En este escenario se cumplen los objetivos de acceso universal , *data resilience*, acceso desde cualquier lugar y el objetivo relacionado con el home *multiservice* server completamente automatizado

Este escenario ha sido pensado y desarrollado para su uso en casa. Para poder desarrollar este escenario ha habido que dedicar 8GB de RAM, 2 discos virtuales de virtual box (emulando dos discos duros, cada uno en un disco físico diferente (5400rpm y 7200 rpm respectivamente) y un tercer disco duro de 8GB emulando una memoria USB, que es donde el SO correrá ya que en FreeBSD

el SO es independiente de los discos duros, también ha sido necesario ceder 3 cores de la CPU(i7-3770k(8cores)) .

## 4.2 Requisitos

- Facilidad de uso
- Comodidad
- Utilidad
- Automatización de búsquedas y descargas
- Video a demanda desde cualquier ubicación
- Variedad de servicios
- Acceso a los datos desde cualquier ubicación
- Acceso Universal
- Capacidad de descargar archivos multimedia automáticamente cuando se publiquen
- Rapidez

## 4.3 Diseño

Las siguientes *jails* estarán virtualizadas dentro del servidor FreeNAS

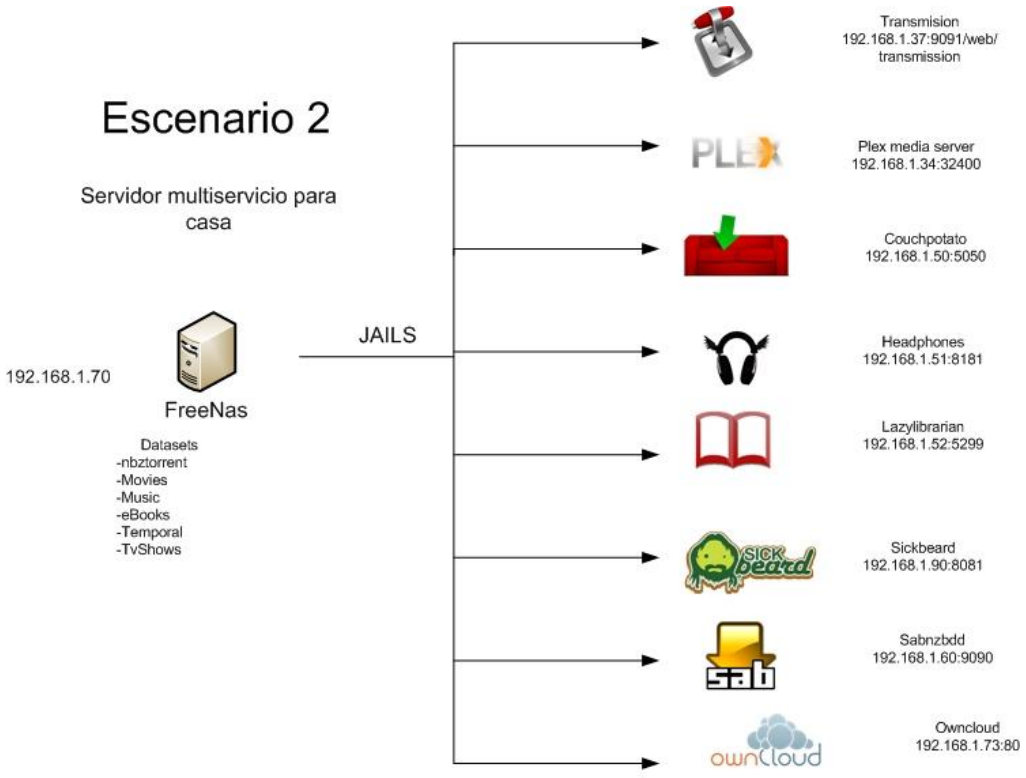



**Fig 4. 1** Diagrama escenario de jails

En este escenario constara de un servidor FreeNAS que contara con los siguientes Datasets Tabla 4. 1. Estos Datasets se podrán acceder en una red local a partir de la IP 192.168.1.70 ya que estarán compartidos usando Cifs y WebDav para que cualquier dispositivo pueda acceder de manera cómoda y simple. Además existirá la opción de usar la *jail* de Owncloud para acceder remotamente desde cualquier lugar como se explicará más adelante.

| Datasets | Usuario | *Jail* que accede |
|---|---|---|
| NZB | Transmission, media, xarokk, Sabnbz,headphones, lazylibrarian, sickbeard, Couchpotato | Sabnbz,Headphones, Lazylibrarian, Owncloud, Sickbeard, Transmission, Couchpotato |
| Music | Headphones, xarokk,media | Headphones ,Owncloud |
| eBooks | LazyLibrarian,xarokk | LazyLibrarian, Owncloud |
| Temporal | Transmission, Headphones, LazyLibrarian Couchpotato xarokk | Couchpotato , Headphones Owncloud Transmission LazyLibrarian |
| TvShows | Sickbeard,xarokk | Sickbeard, Owncloud , PLEX |
| Movies | Couchpotato, xarokk, cuentas de PLEX, Transmission | Couchpotato, xarokk, PLEX,Transmission, Owncloud |
| Fotos y videos | Xarokk y cuentas de Owncloud | Owncloud |

**Tabla 4. 1** Datasets con usuarios y jails que acceden.

Estos Datasets tendrán permisos según que *jails* y usuarios vayan a acceder, de esta manera se evitan accesos no autorizados. Cada Dataset tendrá una función específica, por ejemplo el Dataset NZB será donde se almacenen todos los Torrents y archivos NZB descargados automáticamente por las *jails* Couchpotato , Headphones y Sickbeard, que a su vez podrán leer las *jails* Sabnbzd, transmisión y Owncloud. La idea es separar según que contenido y que *jail* vaya a acceder , así como el tipo de datos que se quieran almacenar por Dataset. De esta manera se consigue gracias al sistema ZFS una optimización del espacio de almacenamiento usado.

Todos estos Datasets colgaran de la siguiente estructura: (Fig 4. 2)

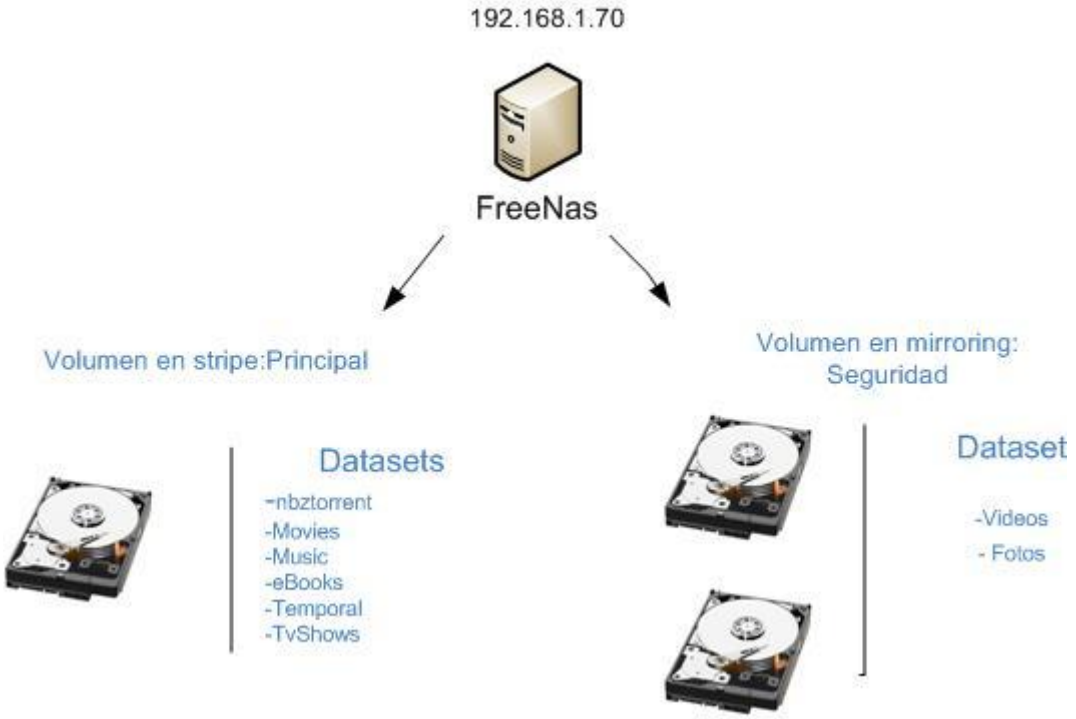



**Fig 4. 2** Distribución de Datasets entre discos

El Volumen en *stripe* se crea para maximizar el espacio disponible de almacenamiento, además como todo el contenido es obtenido a partir de los Torrents o Usenet, es desechable, es decir, en el caso de rotura del disco, todo el contenido puede volver a ser recuperado al volverse a descargar, por eso será el encargado de contener los siguientes Datasets NZB, Movies, Music, ebooks, temporal, tvshows que en caso de pérdida la importancia es mínima, en cambio el volumen Seguridad, está en *mirror*, con lo que si uno de los discos se rompiese, podríamos recuperar la información ya que cada disco guarda una copia de la información. Por esta razón será el encargado de almacenar los Videos las fotos y cualquier cosa que queramos tener guardado redundantemente.

El esquema general de funcionamiento se muestra en la figura Fig 4. 3.Para facilitar la comprensión del diagrama y la explicación se separara en bloques para poder profundizar en mayor detalle.

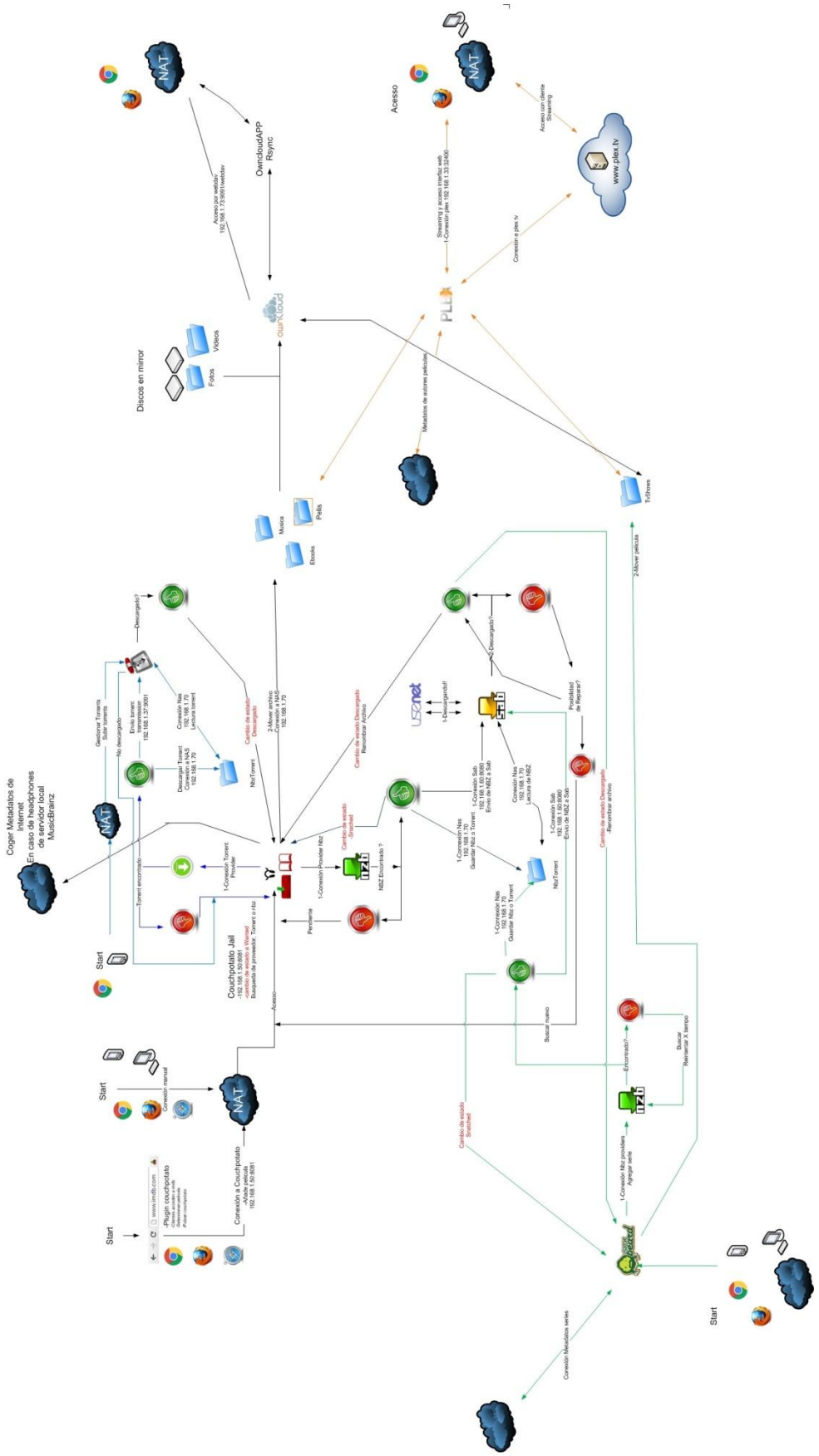



**Fig 4. 3** Diagrama de estados y de Dataset del escenario 2

### 4.3.1. Proveedores NZB y Torrent configurados

Para utilizar los proveedores de NZB Fig 4. 4 y Fig 4. 5, será necesario registrarse en sus respectivas páginas webs, estos proveedores ofrecen un número de llamadas a sus APIS, que permitirán encontrar el archivo de NZB que contiene en qué punto de Usenet se encuentran los archivos deseados. En cambio para Torrent, simplemente hay que poner la url de la web y la *jail* ya se encargará de lo demás.

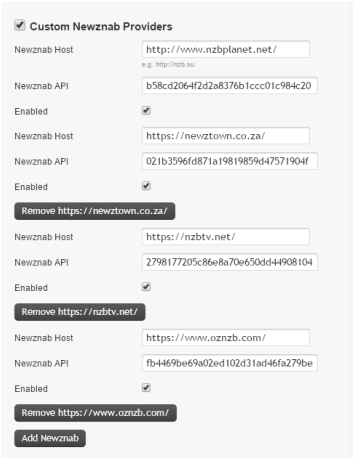



**Fig 4. 4** Proovedor de NZB.

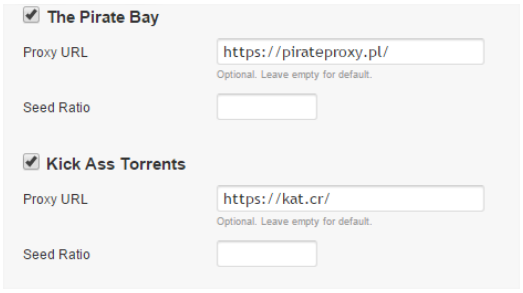



**Fig 4. 5** Proveedor de Torrent.

### 4.3.2. Bloque Couchpotato, Headphones, LazyLibrarian, Sickbeard, Transmision y Sabnzbd

En este bloque se hablará sobre la automatización que el conjunto de jails Couchpotato ,Headphones, LazyLibrarian, Transmision y Sabnzbd de FreeNAS ofrecerán al usuario. El esquema de funcionamiento de este bloque será el de la figura Fig 4. 6

Si el usuario desea ver una película, escuchar una canción o leer un libro en especifico y no quiere perder tiempo en buscar por internet, deberá usar las *jails* Couchpotato, Headphones y Lazylibrarian o en el caso de Couchpotato su *plugin* para navegador, estas *jails* serán las encargadas de buscar, encontrar , renombrar y mover los archivos de audio, texto o video según unas preferencias que el usuario haya configurado en las respectivas interfaces webs de cada jail. Sabnzbd y Transmision serán los encargados de descargar a partir de la información proporcionada por las jails anteriores.

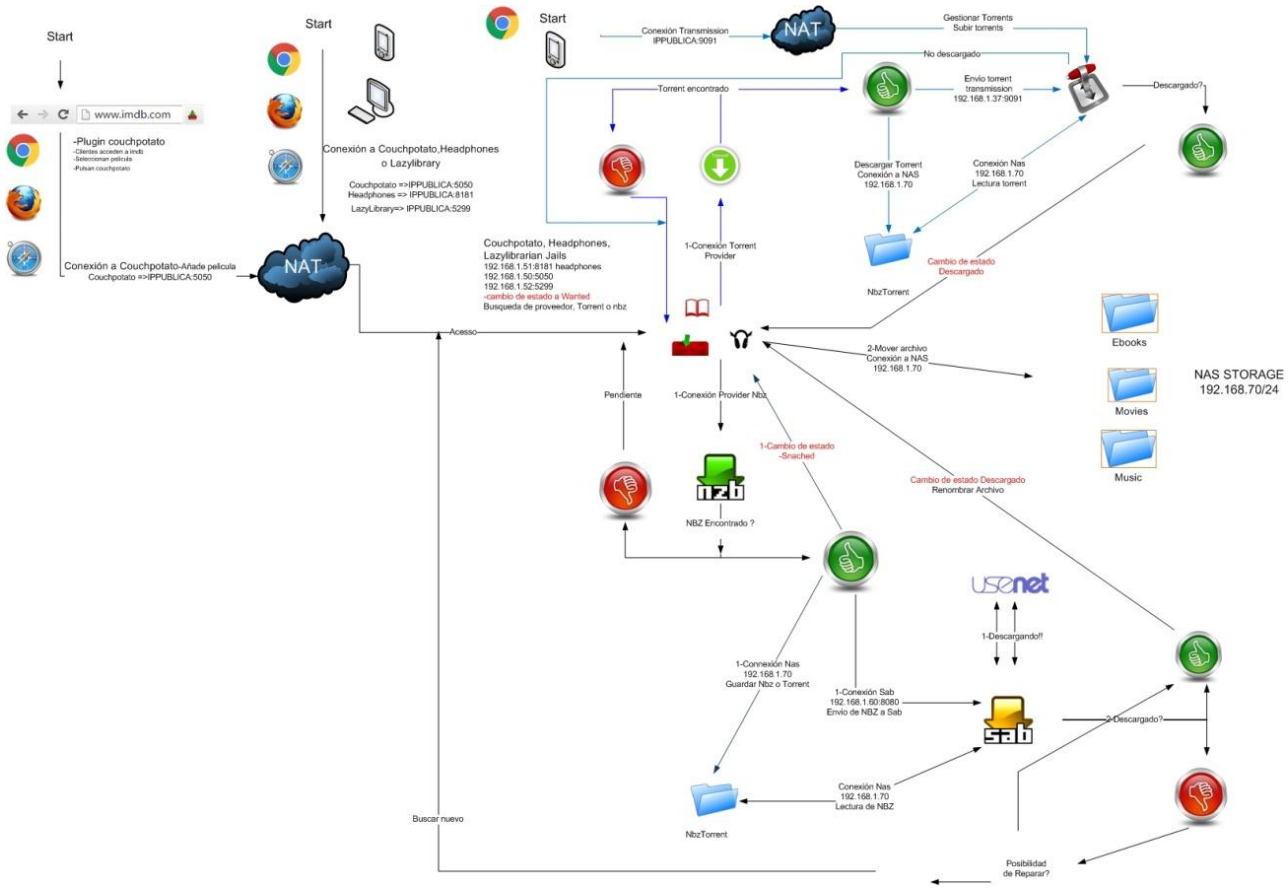



**Fig 4. 6** Diagrama de estados de cómo funciona la descarga

### 4.3.2.1. Funcionamiento del bloque: Ejemplo con Couchpotato

Si el usuario desea descargarse una película puede hacerlo de dos maneras, la primera es a partir del *plugin* Couchpotato para chrome,mozilla o safari, este *plugin* debe enlazarse con la IP privada o la IPPublica:5050 a la que responda la *jail*, dependiendo del uso que quiera darse, si se enlaza usando una IP privada la conexión al servidor para la búsqueda de películas solo puede hacerse de manera local, en cambio si se utiliza la IP pública el acceso es universal, es decir, mientras se tenga acceso a internet el *plugin* para navegador será capaz de encontrar el servidor.

Para seleccionar la película la manera más cómoda y simple para el usuario es acceder a la página de cine www.imdb.com y pulsar el botón del plugin.

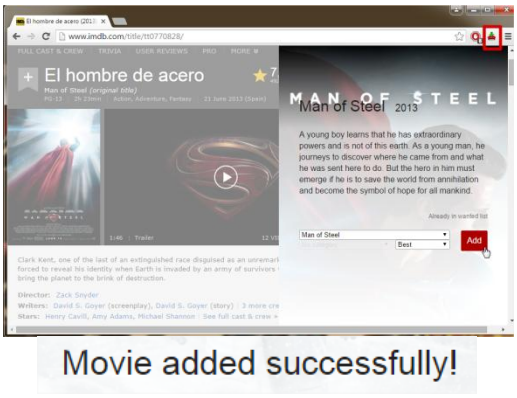



**Fig 4. 7** *Plugin* de navegador de Couchpotato para añadir una película al servidor

Una vez hecho esto el *plugin* se conectará con la *Jail* Couchpotato y Couchpotato se descargará de imdb los metadatos de la película, es decir, el director, autores, trailers etc.. , Entonces se asignará una etiqueta de *Wanted* a la película dentro de la *jail* Couchpotato y este comenzará a buscar automáticamente a cualquier proveedor de Torrents o NZB que se haya configurado utilizando unos parámetros previamente configurados en la *jail* como es la calidad de la película y el idioma .

En la Fig 4. 8 Couchpotato *parsea* imdb y la api de couchpotato para descargar los metadatos correspondientes a la película.


**Fig 4. 8** Log que muestra el *parseo* de las páginas de la web de imdb.

Después de parsear los metadatos, empieza a buscar el archivo por internet que contenga la mayor puntuación.

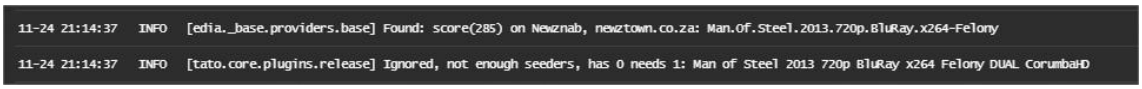



**Fig 4. 9** Log cuando ha encontrado una película pero no tiene suficientes *seeders*.

En el caso de que el archivo tenga mucha puntuación pero no tenga suficientes *seeders*, parámetro que configura el usuario, se ignorará. Fig 4. 9


**Fig 4. 10** Log cuando ha encontrado un proveedor de NZB**.**

Cuando se haya localizado el mejor archivo Fig 4. 10, se enviara automáticamente a Sabnbzd o transmisión dependiendo de que tipo de fichero se haya encontrado, se descargara el NZB o el Torrent y se guardará en el *network storage* NZB,a continuación comenzará la descarga y Couchpotato cambiará el estado de la película a *snatched*, para que en el momento en que este descargada, automáticamente renombrarla y moverla al *network storage Movies*.Fig 4. 11

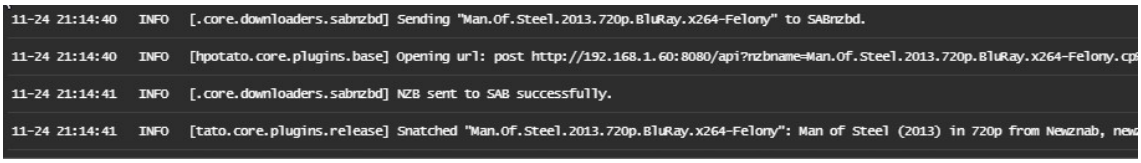



**Fig 4. 11** Log cuando contacta con el SABNZBD**.**


**Fig 4. 12** Barra de descargas de la película a descargar.

En el caso de los Torrents y aunque Couchpotato descargara automáticamente la película usando transmisiónFig 4. 14, permite ver un menú donde se encuentran todas las versiones de las películas ordenadas por puntación para el caso en el cual la descarga fuera mal Fig 4. 13 o se quedara a medias, además permite visualizar el tráiler de la película. La puntuación es una mezcla entre *seeders peers* y diferentes palabras como 1080p,720p etc.

| Release name | Status | Quality | Size | Age | Score | Provider |
|---|---|---|---|---|---|---|
| Man Of Steel 2013 1080p BluRay x264 AC3-ETRG | available | 1080p | 4GB | 14 | 830 | KickAssTorrents |
| Man.Of.Steel.2013.1080p.BluRay.x264.AC3-ETRG | available | 1080p | 4GB | 0 | 637 | ThePirateBay |
| Man.Of.Steel.2013.1080p.BluRay.x264-SECTOR7 [P... | available | 1080p | 9.8GB | 0 | 322 | ThePirateBay |

**Fig 4. 13** Opciones alternativas en caso de que la descarga no haya ido bien en Couchpotato.

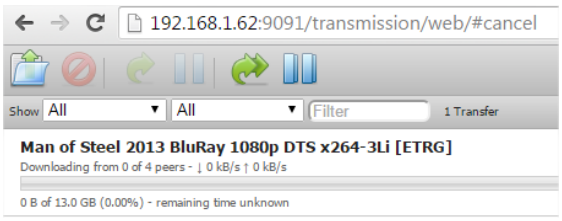



**Fig 4. 14** Jail transmisión que indica la descarga.

### 4.3.2.2. Funcionamiento del bloque: Ejemplo Headphones, Lazylibrarian y Sickbeard

Estas dos *jails* actuaran utilizando la misma metodología que Couchpotato pero la diferencia será en el tipo de archivo que buscaran por internet , además tanto Headphones,Lazylibrarian y Couchpotato permiten una auto organización de todo el contenido, llegando al punto de convertirse en una verdadera biblioteca de música, audio y video. Sickbeard Fig 4. 16 a diferencia de los anteriores solo utilizará Sabnbzd .

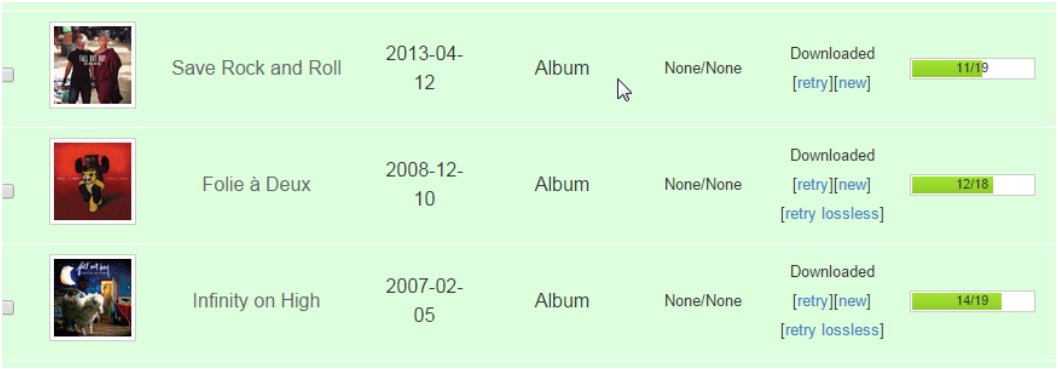



**Fig 4. 15** Álbumes de distintos grupos de la biblioteca.

Una vez se haya acabado de descargar el archivo la jail que lo haya descargado lo renombrará, por tanto Headphones Fig 4. 15 guardará en Music, Lazylibrarian en Ebooks, Sickbeard en TvShows, y Couchpotato en Movies.

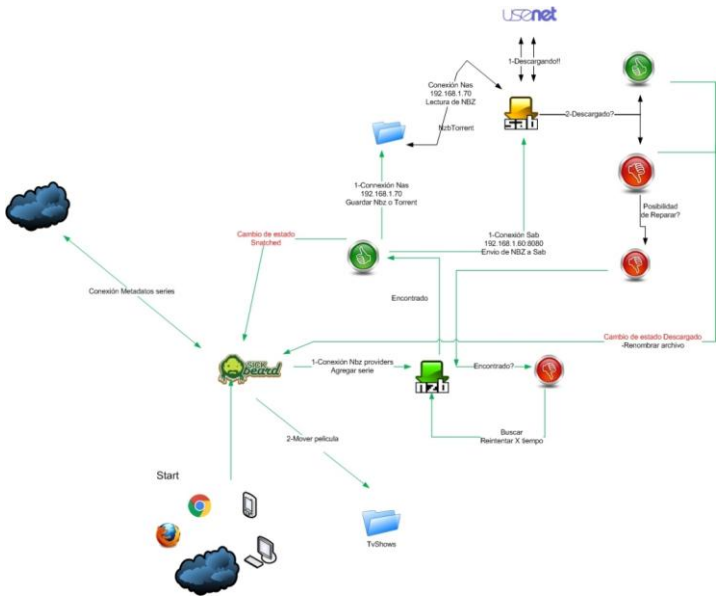



**Fig 4. 16** Diagrama de funcionamiento de Sickbeard.

#### 4.3.2.3. MediaPLEX y Owncloud

Ahora que ya esta automatizado todo el proceso de búsqueda y descarga, se configura la *jail* MediaPLEX que permitirá la reproducción en *streaming*, desde cualquier lugar y dispositivo con conexión a internet, bien sea usando la aplicación de android/IOS o conectando vía IP pública.

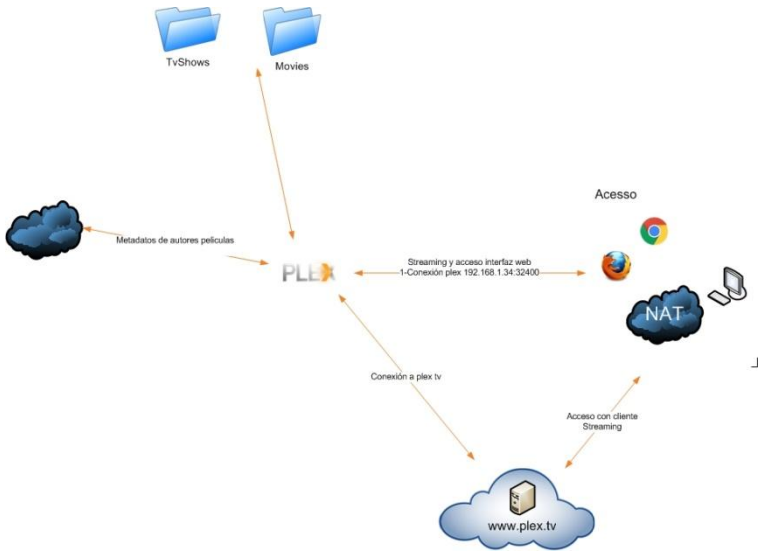



**Fig 4. 17** Diagrama de MediaPLEX.

PLEX leerá los archivos de media contenidos en el *network storage Tvshows* y *Movies (*Fig 4. 17*)*, descargará los metadatos correspondientes que serán independientes de aquellos descargados por Sickbeard o Couchpotato, organizará series según temporadas y los servirá a demanda a cualquier dispositivo que los solicite.

En el caso de usarse una aplicación propietaria de PLEX, no será necesario recordar la IP pública del servidor ya que se conectara con plex.tv de la manera que esta explicada en el anexo.
Para finalizar la jail Owncloud Fig 4. 18, permitirá gestionar todos los Datasets remotamente y como si de un cloud storage se tratase, permitiendo la gestión de usuarios y permisos según archivos y carpetas.

Ya que existen Datasets más importantes que otros, como el de fotos y videos, que es donde se guarda la información personal de los usuarios , estos dos Datasets estarán en modo *mirror* ya que es la opción que mayor redundancia da para dos discos, ya que si uno de los discos duros se rompiese , siempre se puede recuperar la información debido a que, este tipo de configuración guarda copias iguales de la información en cada uno de los discos

.

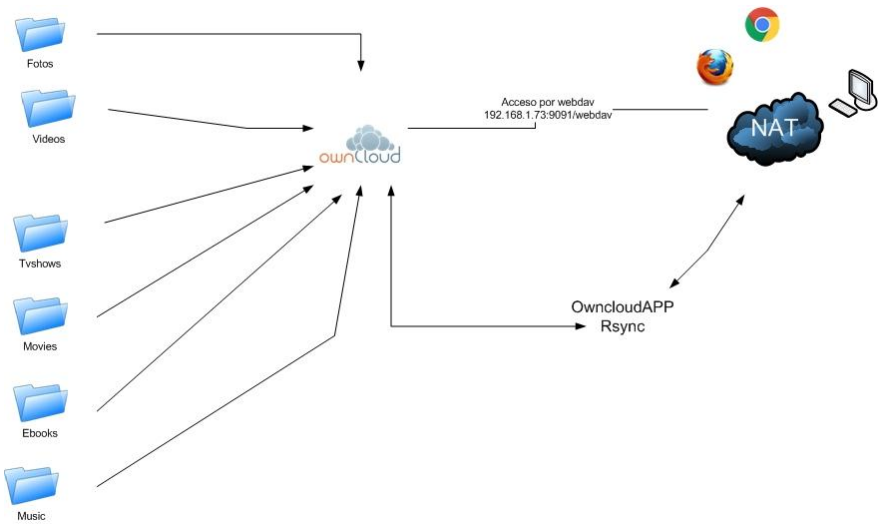



**Fig 4. 18** Diagrama de Owncloud.

Owncloud permite la conexión usando la aplicación para android/IOS y también se puede utilizar un acceso por Webdav utilizando la IPpublica:puerto del servidor.

## 4.4. Network attached storage del escenario

La imagen de la Fig 4. 19 muestra como el *network attached storage* esta enlazado entre *jails*, es decir, que cada *jail* tiene montado aquel Dataset que necesita para operar

| | | |
|---|---|---|
| sabnzbd_1 | /mnt/Principal/Nbz | /mnt/Nbz |
| sabnzbd_1 | /mnt/Principal/Tvshows | /mnt/tvshows |
| sickbeard_1 | /mnt/Principal/Nbz | /mnt/Nbz |
| sickbeard_1 | /mnt/Principal/Tvshows | /mnt/Tvshows |
| transmission_1 | /mnt/Principal/Nbz | /mnt/Torrentprincipal |
| transmission_1 | /mnt/Principal/Peliculas | /mnt/peliculasprincipal |
| couchpotato_1 | /mnt/Principal/Nbz | /mnt/nbz |
| couchpotato_1 | /mnt/Principal/Peliculas | /mnt/peliculas |
| owncloud_1 | /mnt/Principal | /mnt/nube |
| lazylibrarian_1 | /mnt/Principal/Ebooks | /mnt/Ebooksprincipal |
| headphones_1 | /mnt/Principal/Musica | /mnt/music |
| plexmediaserver_1 | /mnt/Principal/Peliculas | /mnt/pelisprincipal |

**Fig 4. 19** Relaciones entre *jails* y *Datasets*.

Por ejemplo ya que Owncloud tiene que poder acceder a todos los archivos, su punto de montaje en *NAS* colgara del Dataset Principal, dentro de este Dataset colgaran los demás puntos de montajes que estarán distribuidos por cada jail que necesite acceso.

Otro ejemplo seria Transmisión , Couchpotato y PLEX necesitan tener acceso al Dataset de películas, transmisión para poder compartir, Couchpotato para saber que ya esta descargada y PLEX para poder reproducirla en *streaming*.

## 4.5. Rendimiento

Una vez acabado el escenario, nos damos cuenta que los recursos , sobretodo la RAM son un bien preciado para FreeNAS, ya que son 8 *jails* , lo que implica que realmente se están virtualizando con los mismos recursos 9 maquinas independientes, si bien de CPU tienen un consumo muy reducido, ya que para el escenario que he montado el procesado que hay es mínimo, pero en cambio para la RAM es lo contrario, debido a la mecánica propia de funcionar de FreeBSD.

Para comprobarlo se activará en los gestores de descargas Sabnbzd y transmisión diferentes Torrents para comparar distintos componentes del sistema.

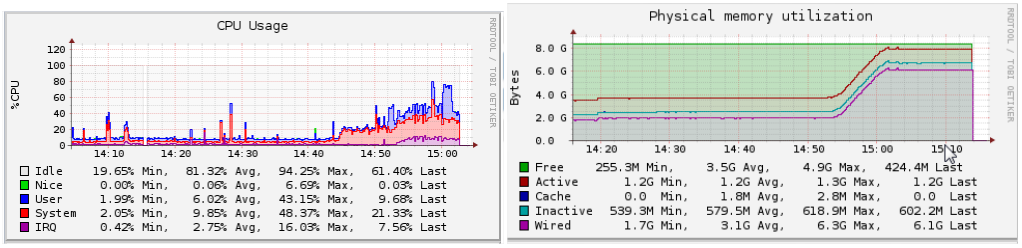



**Fig 4. 20** Consumo de la CPU a la izquierda y de la RAM a la derecha.

Como se puede observar en estas gráficas, una vez se empieza a descargar el sistema empieza a ocupar toda la RAM disponible y no la libera a menos que sea necesaria para otra cosa, es decir, hasta que el sistema no se reinicie o la RAM sea necesaria para otro uso, toda esta información se guardara en él la memoria, se use o no. En cambio en CPU, no tendrá tanto consumo , ya que apenas llega al 50 % del uso de los 3 *cores* cedidos.

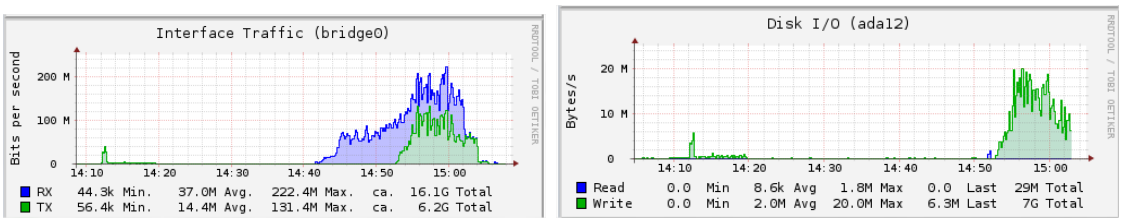



**Fig 4. 21** La transferencia por la interfaz de red a la izquierda y a la derecha la escrito en el disco.

La interfaz bridge, es la suma de todas las interfaces de las *jails* y la de FreeNAS, por tanto en la gráfica se representará en azul lo que se está descargando hacia FreeNAS y en verde la transferencia de FreeNAS a las *jail* transmisión, si lo desglosamos:

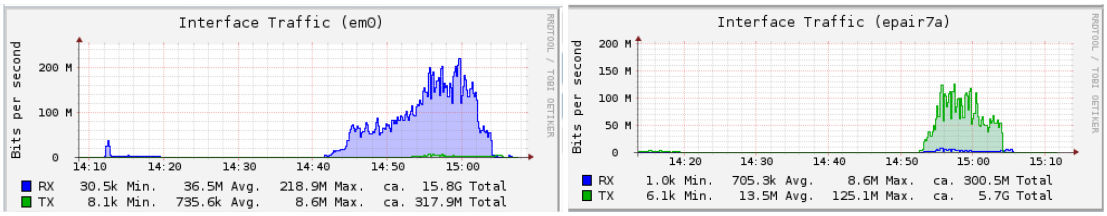



**Fig 4. 22** Transferencia por la interfaz de FreeNAS a la izquierda y a la derecha interfaz de *Jail* SABNZBD.

Se puede observar la interfaz em0 que es la correspondiente a FreeNAS y epair7a que es la que se corresponde a la *jail* de transmisión, que si las solapamos las dos graficas nos dan la interfaz bridge0.

## 4.6. Costes del Escenario

Con los componentes escogidos abajo nos acercaríamos al máximo a la simulación pero lo ideal para este escenario serian 16gb de RAM, el servidor irá fluido, pero con 16gb, se aumenta el rendimiento y solo por un aumento de 20 euros.

| **Componente** | **Tipo** | **Precio** |
|---|---|---|
| Gigabyte GA-H81M-S2H | Placa Base | 50.95 |
| Intel Core i3-4160 3.6Ghz Box | Procesador | 106.95 |
| G.Skill Ripjaws X DDR3 1600 PC3-12800 8GB CL9 | Memoria | 41.95 |
| L-Link Fuente de Alimentación ATX 500W | fuente | 15.95 |
| Tacens Mars Gaming MC0 | Torre | 19.95 |
| Seagate Desktop 7200.14 2TB SATA3 64MB | Disco | 67.95 |
| Seagate Barracuda 7200.14 1TB SATA3 Reacondicionado x2 | Disco | 39.95x2 |
| Total | | **383.6** |

**Tabla 4. 2** Coste de adquisición de la infraestructura para el home server.

Por tanto, por 383,6 -406,6 (dependiendo de cuanta RAM queramos) euros podemos tener el servidor con 2 terabytes para descargar y 1 terabyte en mirror para los datos más importantes como son las fotos y videos personales.

# CAPITULO 5: CONCLUSIONES Y PREVISIÓN DE FUTURO

## 5.1 Conclusiones

ZFS es un sistema de archivos de 128 bits a diferencia de ext4 y NTFS que actualmente están basados en 64 bits que son los correspondientes a Unix y Microsoft . La ventaja de NTFS es que tiene compatibilidad con cualquier OS , pero su tamaño máximo de archivo es de 16 TB y de volumen de 256 TB , además es más lento a la hora de leer y/o escribir en el disco que ZFS o ext4 . Ext4 es el comúnmente utilizado en sistemas Unix , tiene compatibilidad con la mayoría de sistemas que no sean de Microsoft, su tamaño máximo de archivo es de 16TB pero a diferencia de NTFS puede tener un volumen máximo de 1 exabyte. ZFS es el que asegura mayor protección de datos pero también el que mayor cantidad de memoria consume, su tamaño máximo de archivo y volumen es de 16 exabytes , además es el que mayor rendimiento da a la hora de servir peticiones o escritura/lectura de disco.

En el escenario empresarial se han suplido las necesidades básicas de una pequeña empresa como son el almacenamiento , la virtualización de servicios e infraestructura, la seguridad de datos y de acceso . Para ello se ha diseñado un escenario que intenta simular una pequeña parte de la infraestructura de una empresa diseñada para poder dar servicio 24x7.Se ha intentado dar una idea de cómo el sistema de archivos ZFS puede ser útil para maximizar la seguridad en datos y aumentar la velocidad a la hora de servir peticiones , así como de asegurar una rápida recuperación respecto a fallos gracias al funcionamiento interno del sistema ZFS a la hora de tratar la información a nivel de bloque en lugar de a nivel de aplicación.

En el escenario particular se ha creado un servidor multiservicio que supla las necesidades de recreo en cuanto a contenido multimedia se refiere. Facilitando el acceso multimedia, incrementando la velocidad de adquisición de contenido, completando el contenido descargado con metadatos de las paginas oficiales de video, música y libros, permitiendo un servicio de video en *streaming* auto organizado de calidad y un acceso universal, desde cualquier lugar a los datos contenidos en el servidor gracias a la creación de una nube auto hospedada.

Por tanto este proyecto me ha permitido plantear una problemática en dos escenarios muy diferentes con necesidades muy dispares las cuales han sido solucionadas por la herramienta base FreeNAS que permite la integración con cientos de servicios y la aplicación de soluciones conjuntas con otras aplicaciones de terceros como son Openvpn , Virtual box , Headphones etc.. y todo esto se ha cumplido utilizando exclusivamente Software Open Source

Sobre la efectividad y la utilidad de las *jails* del escenario de home server hay que recalcar que Couchpotato, Sickbeard , Headphones han dado resultados

excelentes la hora de encontrar el contenido pero LazyLibrarian no, ya que los libros en formato epub o pdf suelen encontrarse con más facilidad por descarga directa que por Torrent o Usenet , solamente recomendable para trilogías o libros *Trending topic* .Los gestores de descargas han demostrado un resultado excelente a la hora de descargar pero debido al funcionamiento de ZFS suelen colapsar el uso de RAM si se mantienen encendidos mucho rato con lo cual es recomendable una correcta configuración en numero de conexiones máximas, *upload* máximos y cantidad de archivos que se estén descargando.

## 5.2 Previsión de futuro

Ampliaciones posibles del escenario empresarial

Ya que la mayoría de empresas utilizan un dominio de Microsoft , el proyecto podría ser ampliable utilizando la integración nativa que ofrece FreeNAS con *active directory* o en su defecto con Kerberos.

La automatización de *backups* hacia Amazon también es una herramienta nativa existente en FreeNAS que permite a partir del uso de una *jail* crashplans la recuperación de esos datos previamente hechos *backups*.En caso de querer backups locales la herramienta necesaria seria la *jail* bacula-sd.
Otra posible mejora sería la posibilidad de clonar e instalar maquinas utilizando el *multicast* gracias a una *jail* nativa denominada Cruciblewds
idencia

Ampliaciones escenario home server

Una mejora posible seria añadir una *jail* encargada de reproducir la música en *streaming* rivalizando con spotify pero sin anuncios. Esa *jail* se denomina SubSonic

# GLOSARIO

**Active Directory:** Directorio activo de un dominio de Microsoft

**API :** Application Program Interface

**Backups:** Copias de seguridad

**ESX :** Sistema operativo para el gestion de máquinas virtuales

***Jails* :** instancias virtuales de máquinas basadas en FreeBSD

**NZB :** Archivo codificado en texto de contenido multimedia

**Peers :** instancias de Torrents corriendo en ordenadores remotos

**RAID :** redundant array of inexpensive disk

**Seeders :** ordenadores que ya tienen el archivo y siguen compartiendolo

**SO:** Operative system

**SSH :** Secure Shell

**Streaming:** Reproducción online sin descargar contenido

**ZFS :** Zetta Byte File System

***Snapshots* :** Instantanias de un momento dado del estado de la información

# BIBLIOGRAFIA


[1] Gary *Sims Leaning FreeNAS configure and manage a network attached storage solution* Packt Birmingham publishing august 2008

[2] Ixsystems *FreeNAS User Guide 9.3.1* [en linea] disponible a (http://doc.FreeNAS.org/9.3/FreeNAS.html).

[3] Lousise k Comfort, Yesim Sungu, David Johnson and Mark Dunn , *ComPLEX Systems in Crisis: Anticipation and Resilience in Dynamic Environments*, Blackwell Publishes Volume 9 Number 3 September 2001 pag 1-4.

[4] Sungard, *Risk-resilience-recovery The R3 Approach to Data* Governance,2012

[5] Uday Vallamsetty, *ZFS write Performance (Impact of Fragmentation)*,19th February 2013 [en linea] disponible en(http://blog.delphix.com/uday/2013/02/19/78/) .

[6] Cisco, The *Zettabyte Era:Trends and Analysis* [en linea] Disponible en (http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/VNI_Hyperconnectivity_WP.pdf) page 1-6

[7] Margared Rouse, *Disk Stripping*[en línea] Disponible en (*http://searchstorage.techtarget.com/definition/disk-striping*)

[8] Babin Lonston,*Configuring FreeNAS to Setup ZFS Storage Disks and Creating NFS Shares On FreeNAS*,November 27, 2014 [en linea] (*http://www.tecmint.com/add-storage-disks-to-FreeNAS-server/*)

[9] Adam Leventhal's Weblog*, What is RAID-Z* [en linea] Disponible en (*https://blogs.oracle.com/ahl/entry/what_is_RAID_z*)

[10] Durindiel.fr , *Certificats et Webdav* [en linea ] Disponible en (http://www.durindel.fr/informatique/tuto-FreeNAS-9-3-certificats-et-webdav)

[11] Owncloud, *Owncloud User manual 8.2* [en línea] Disponible en (https://doc.Owncloud.org/server/8.2/Owncloud_User_Manual.pdf)

# ANEXOS

## Anexo A : JAILS Escenarios

FreeNAS ofrece la posibilidad de instalar *jails* que aumentan la funcionalidad así como los servicios brindados, los plugins se instalan en *jails*, que son sistemas operativos completamente aislados del servidor, es decir, lo único que comparten es el hardware. Estas *jails* están basadas en FreeBSD y por lo tanto los diferentes servicios como *Couchpotato* , *Headphones* y todas las demás *jails* contendrán un sistema de archivos con la estructura de FreeBSD
Ya que la *jail* que contiene el servidor, está aislada, es necesario configurarle una IP con la que poder acceder a la *jail*, esto se hará desde la interfaz web de FreeNAS que posee una Shell por cada *jail* que exista, de esta manera podremos acceder a ella y asignarle la IP , una vez asignada la IP, se podrá acceder  la maquina remotamente, como si de una maquina mas se tratará
Además también se puede acceder a la consola de cualquier máquina utilizando la consola principal de FreeNAS.

Las jails del escenario corporativo son las siguientes:**A.  1** Escenario Corporativo Jails

| Jail | IPv4 Address | Autostart |
|---|---|---|
| VirtualBox | 192.168.1.200 | true |
| Openvpn | 192.168.1.240 | true |

**A.  1** Escenario Corporativo Jails

Las jails del escenario home server son las siguientes: A.  2

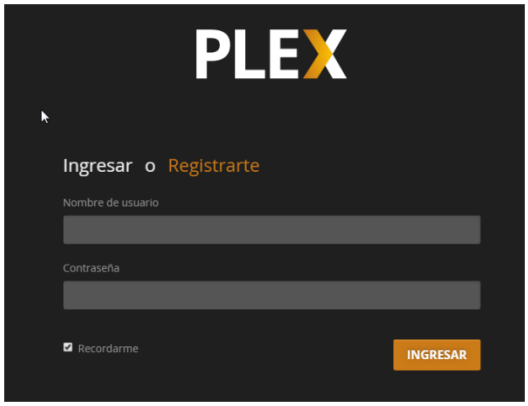
| Jail | IPv4 Address | Autostart | Status | Ty |
|---|---|---|---|---|
| sickbeard_1 | 192.168.1.90 | true | Running | plu |
| sabnzbd_1 | 192.168.1.60 | true | Running | plu |
| owncloud_1 | 192.168.1.80 | true | Running | plu |
| plexmediaserver_1 | 192.168.1.34 | true | Running | plu |
| transmission_1 | 192.168.1.37 | true | Running | plu |
| couchpotato_1 | 192.168.1.50 | true | Running | plu |
| headphones_1 | 192.168.1.51 | true | Running | plu |
| lazylibrarian_1 | 192.168.1.52 | true | Running | plu |

**A.  2** Escenario Home server Jails

## Anexo B: MediaPLEX server

Con  el servicio activado se podrá acceder usando la siguiente url. B. 1

http://[IP]:[puerto]/web/index.html




**B. 1** Ingreso en PLEX

Una vez que podemos acceder al servicio *PLEX* remotamente, deberemos introducir contenido en el, para ello habrá que asignar Datasets, estos Dataset deben estar asignados al usuario *nobody* y al grupo media, al ser el Dataset propiedad del usuario *nobody* implica que cualquier *user* podrá acceder, modificar y/o eliminar archivos siempre y cuando este en el grupo media.

Ya que el servidor *PLEX* y FreeNAS están aislados , cualquier usuario existente en PLEX , no existe en FreeNAS y viceversa, debido a esto habrá que crear un usuario que exista en ambas maquinas y que usen el mismo id, creamos en FreeNAS un usuario llamado media propietario del grupo media y en la maquina MediaPLEX asignamos al usuario PLEX(administrador en el servidor) el grupo media con el id de FreeNAS usando los siguientes comandos:

$pw groupadd media -g 1003  ## Creamos el grupo media y le asignamos el id 1003 que es el id del grupo

$pw usermod PLEX -G media  ## asignamos al usuario PLEX al grupo media

Una vez asignados los permisos podremos asignar almacenamiento que este contenido en el servidor de FreeNAS, este almacenamiento debe ser propiedad de media y estar en el grupo media, ya que si no el usuario PLEX del servidor MediaPLEX no tendrá permisos y no será capaz de leer la información contenida en  el espacio de almacenamiento adicional.

En la siguiente imagen, vemos que el servidor MediaPLEX contiene 2 espacios de almacenamiento adicionales. La pestañita source indica en que parte del

servidor FreeNAS está el almacenamiento y la pestañita Destination indica donde se montara el almacenamiento dentro de la maquina MediaPLEX.

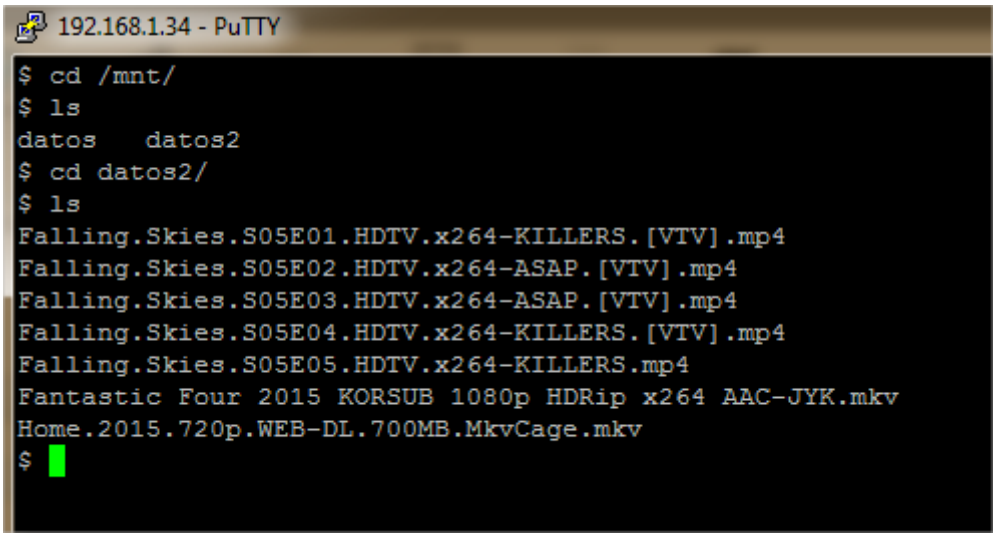
| Jail | Source | Destination |
|---|---|---|
| plexmediaserver_1 | /mnt/JAILS/media | /mnt/datos |
| plexmediaserver_1 | /mnt/JAILS/Controller/Pelis | /mnt/datos2 |

**B. 2** PLEX Nas storage

Para poder acceder remotamente y no tener que acceder a la interfaz web de FreeNAS cada vez que se quiera cambiar algo de configuración de la maquina MediaPLEX habilitaremos el acceso por SSH ,para ello habrá que habilitar el acceso sshd enable =yes (B. 3),además será necesario generar las claves con el siguiente comando /usr/bin/ssh-keygen –A y reiniciar el servicio

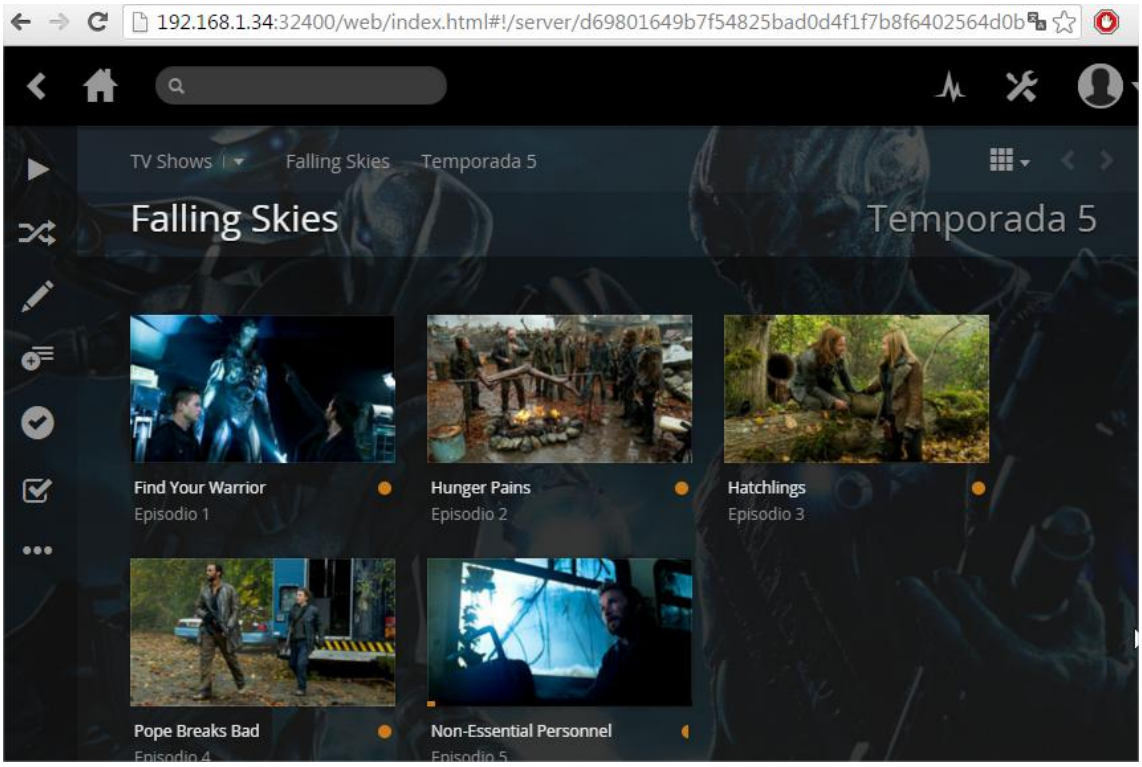

**B. 3** archivo /etc/rc.conf

Ahora que ya podemos acceder remotamente se puede comprobar que efectivamente la información que había */mnt/JAILS/media* y */mnt/JAILS/Pelis* del servidor FreeNAS está contenida dentro de la carpeta datos y datos2 del servidor MediaPLEX B. 4, con su contenido multimedia



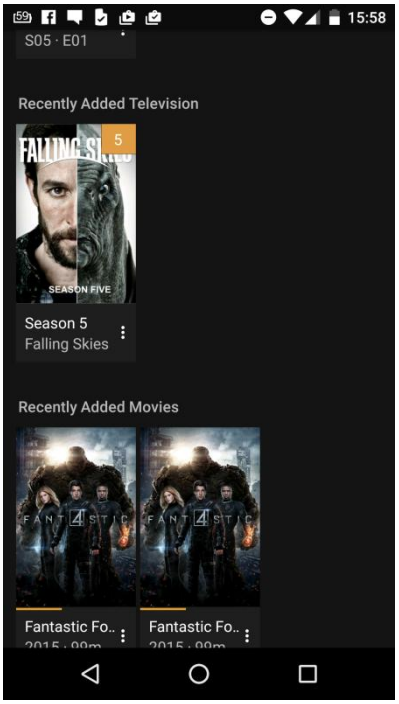

**B. 4** Contenido de la carpeta montada

Como se puede observar en la imagen B. 5, el contenido pirata en este caso ha sido analizado y a partir de metadatos de internet se han auto organizado todos los capítulos de Falling skies .

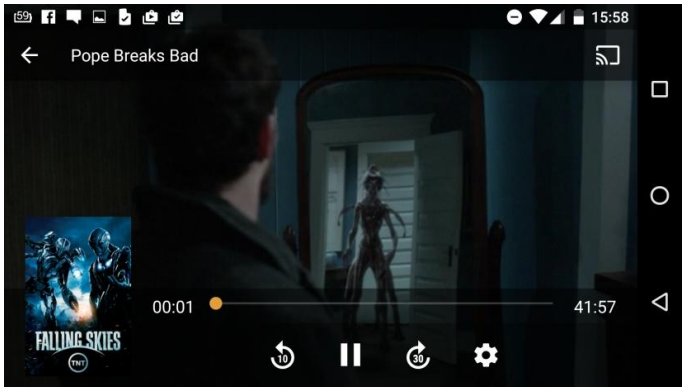



**B. 5** PLEX reproducción

La reproducción del contenido es a demanda y multiplataforma B. 6 y B. 7es decir , cualquier tipo de dispositivo con acceso a internet/red local será capaz de reproducirlo , bien sea a partir de una aplicación y/o a partir del navegador

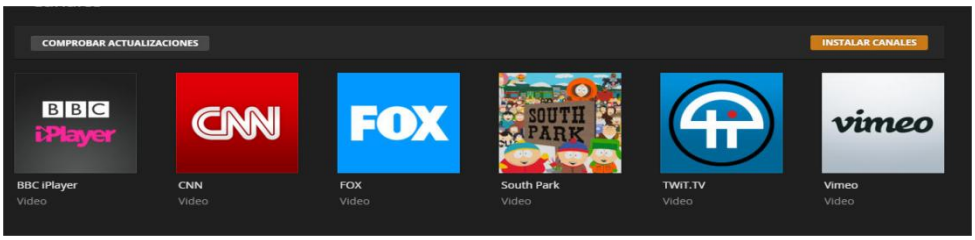



**B. 6** Movil

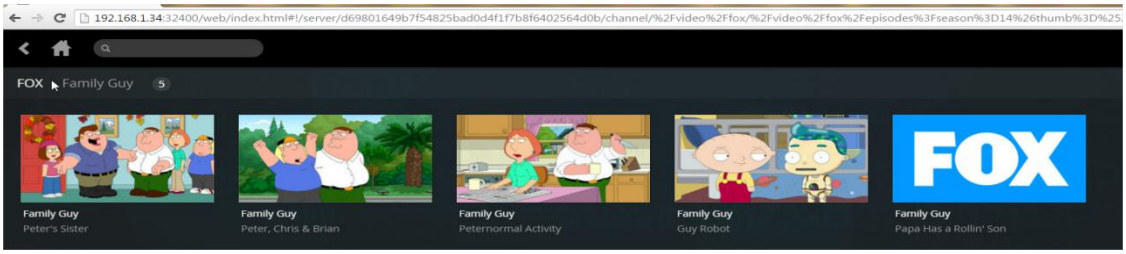



**B. 7** Reproducción desde el móvil

El servidor MediaPLEX a su vez permite la sincronización de canales de tv o radio de todo el mundo como la BBC, o vimeo B. 8 permitiendo al usuario ver desde los programas emitidos hace 7 días hasta los de hoy, también brindando esta posibilidad a cualquier dispositivo que se conecte al servidor de casa



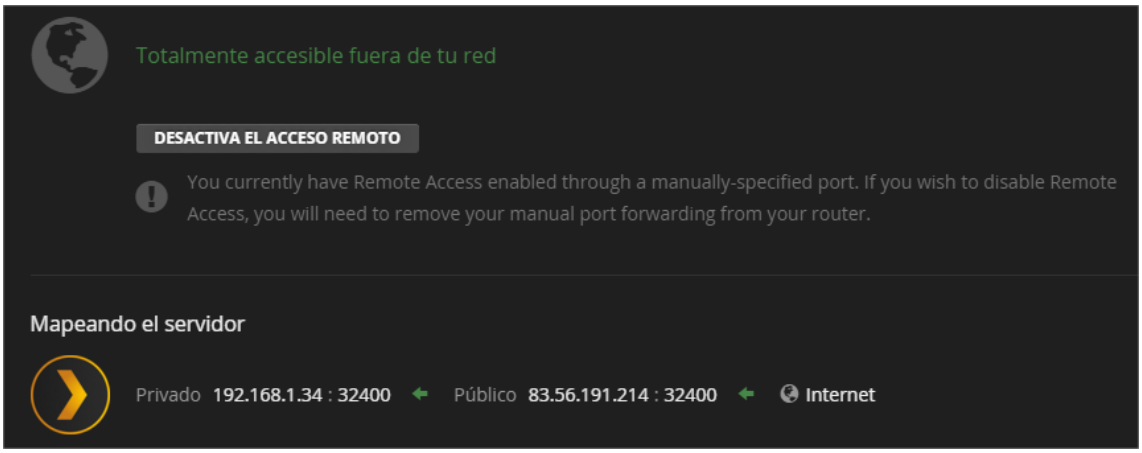

**B. 8** Canales PLEX




**B. 9** Canal Fox

## B.1 Acceso remoto

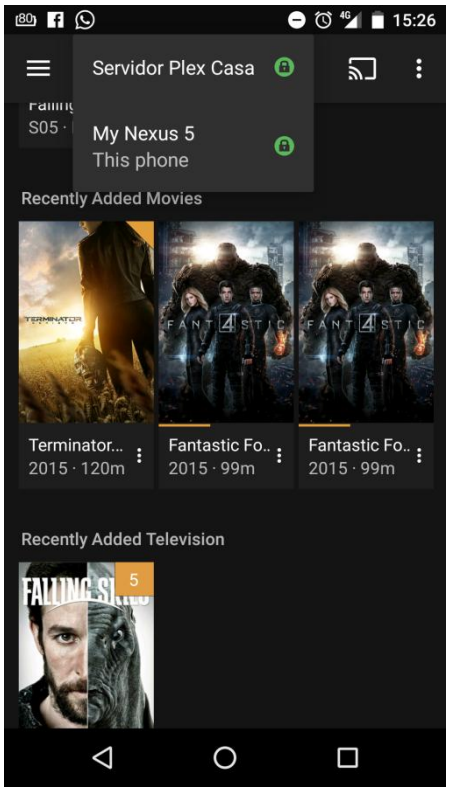
MediaPLEX, además permite que cualquier dispositivo que se haya conectado a tu home media server , volverse a conectar usando PLEX.tv, siempre y cuando se haya configurado el acceso remoto B. 10 .Si el acceso remoto esta activado desde cualquier punto del mundo con conexión a internet ,cualquier dispositivo que ya se haya conectado usando la red local, recordara el id de tu PLEX server y será capaz de encontrarlo usando PLEX.tv




**B. 10** Acceso remoto habilitado

Primero es necesario al creación de una cuenta en PLEX.tv, una vez creada será necesario *logearse* en cada dispositivo, así como en el servidor. En el momento en que servidor y dispositivo estén *logeados* en PLEX.tv , el dispositivo será capaz de encontrar el servidor , ya que PLEX.tv (B. 11) notificara en que IP publica se encuentra nuestro servidor , después se enviara un sincronizar para recibir el catalogo disponible en el servidor

Una vez ha sido sincronizado el catalogo el dispositivo podrá hacer una petición *on demand*. PLEX tiene la característica de que puede brindar usando *multicast* los paquetes de video y servir diferentes flujos de video/audio sin problemas, siempre y cuando no se sature la RAM del servidor.




**B. 11** Conexion usando la app de PLEX tv

Hay que recalcar que también se puede acceder usando la siguiente dirección

http://[IPPUBLICA]:32400/web/index.htlm

al acceder usando esta dirección , se evita el uso de PLEX.tv por tanto esta no participara en el intercambio de mensajes

## A.2.2 MediaPLEX + Transmission

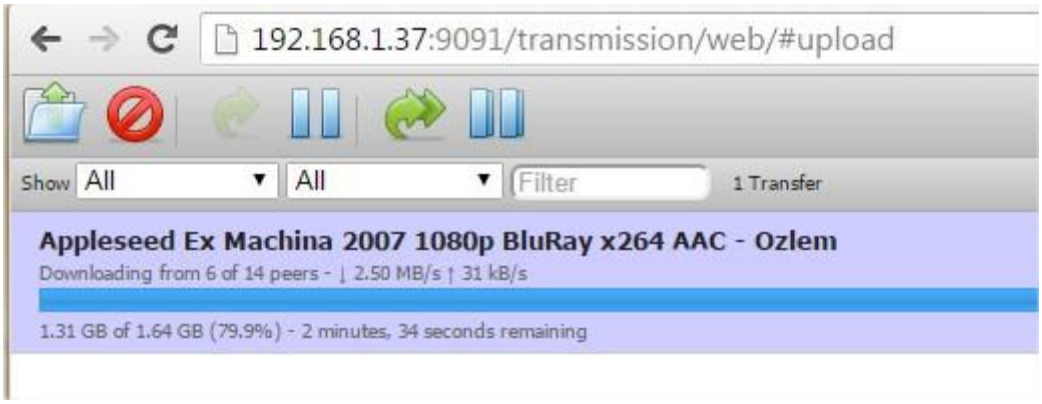
Transmission es una *jail* de FreeNAS , que hace la función de cliente P2P. Este *jail* como PLEX y todos las *jails* de FreeNAS corre sobre FreeBSD , que lo emula en una *jail* aislada del sistema *Nas* .

Una vez instalado el plugin y para unirlo con MediaPLEX, será necesario configurar como almacenamiento el mismo Dataset que MediaPLEX

| Jail | Source | Destination |
|---|---|---|
| plexmediaserver_1 | /mnt/JAILS/media | /mnt/datos |
| plexmediaserver_1 | /mnt/JAILS/Controller/Pelis | /mnt/datos2 |
| Virtual | /mnt/JAILS/Controller/ISOS | /mnt/datos |
| transmission_1 | /mnt/JAILS/Controller/Pelis | /mnt/datos |

**B. 12** Carpetas entre transmision y PLEX

Este Dataset tendrá permisos para el grupo media, ya que así se había configurado para el servidor PLEX y Couchpotato, por tanto para que el usuario de transmission pueda crear los Torrents y empezar a descargar B. 13, primero habrá que crear tres carpetas watch, complete y temp , las cuales se asignara como propietario el usuario transmisión y se le asignaran permisos 775.Cuando esto ya este hecho, los Torrents empezaran a descargarse sin ningún tipo de error.




**B. 13** Transmision descargando

Transmision al igual que MediaPLEX server permite el acceso remoto desde fuera de la red local, para ello solo hay que acceder en el router para habilitar el *forwarding* la IP de transmission

IPPUBLICA:9091/transmission/web/#files

Una vez la descarga se haya efectuado, automáticamente se añadirá a la librería de MediaPLEX y couchpotato cambiará el estado a *downloaded*, en la que este organizara de manera automática usando los metadatos de internet

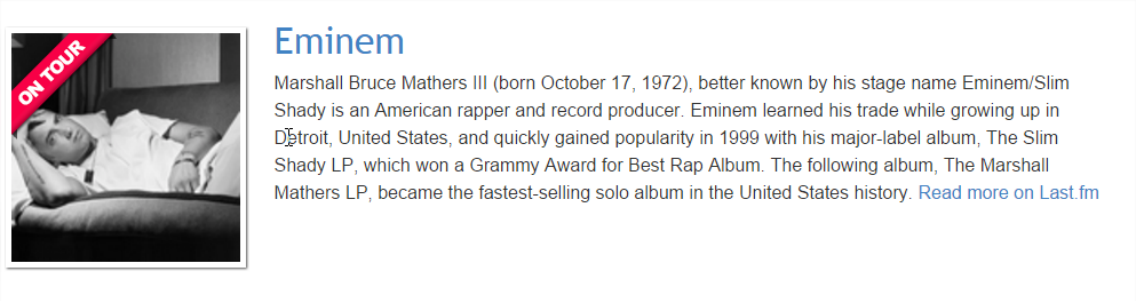
## Anexo C: Headphones y LazyLibrarian

Headphones utiliza como indexador musicbrainz , que es de dónde saca toda la información de los artistas y sus álbumes, antiguamente había infinitas llamadas a la api de musicbrainz por IP pero desde hace un tiempo, los administradores de la pagina web han decidido limitar el servicio a 1 *request* por segundo por IP y a 50 por segundo por todas las IP, es decir, actualmente el servicio gratuito ha quedado completamente inservible, por tanto no se puede indexar a la *jail* de headphones la información necesaria para la búsqueda de Torrents , para solventar este problema , en este proyecto se creará un servidor de indexación propio, la otra opción es pagar el servicio VIP , que equivale a 10 euros al mes.

Para que nuestro propio servidor de indexación este actualizado será necesario registrarse en la página web de MusicBrainz, seguir los comandos que ponen en esta página https://musicbrainz.org/doc/MusicBrainz_Server/Setup para habilitar la replicación y tener la base de datos actualizada, luego será necesario solicitar un token que nos permita sincronizar con su base de datos

token 9iJXP3xUKuSZOoQ1rSJAzbiM8adMqOzMbOd6k9mG

Cuando el servidor de Musicbrainz este activo y después de configurar. Headphones se conectará y descargará los metadatos de nuestro servidor Musicbrainz, este empezara a recopilar información de los artistas que hayamos seleccionado.
En esta información vienen incluidos los álbumes, conciertos futuros, canciones y bibliografía del artista.(C. 1)




**C. 1** Descripción Eminem

### Ejemplo del proceso con Eminem

En los logs extraídos de Headphones se puede observar que esta sincronizando con la base de datos de musicbrainz los álbumes de Eminen

```
Fetching Metacritic reviews for: Eminem
Seeing if we need album art for: Eminem
```

[Eminem] No new releases, so no changes made to How to Be an MC, Volume 13 (Eminem Instrumentals)

[Eminem] Now adding/updating: How to Be an MC, Volume 13 (Eminem Instrumentals) (Comprehensive Force)

[Eminem] Seeing if we need album art for Relapse

[Eminem] Packaging Relapse releases into hybrid title

[Eminem] New release Relapse (c0c2f1bf-57d9-4d4f-88a0-4da54ac7c65e) added

[Eminem] New release Relapse: Refill (a5afe885-aac7-4dd0-98c6-e59507663d8c) added

[Eminem] New release Relapse (88a856de-d5d1-4225-9612-8858b2f5353f) added

[Eminem] New release Relapse (5fe6d9b3-3602-4315-ad7c-9aa0c8feade7) added

[Eminem] New release Relapse (25130d2d-8a82-4956-99e7-30efd0f9ff89) added

[Eminem] New release Relapse (249a5c51-7002-309e-9871-95cdcea77153) added

[Eminem] Now adding/updating: Relapse (Comprehensive Force)

[Eminem] Seeing if we need album art for The Eminem Show

[Eminem] Packaging The Eminem Show releases into hybrid title

[Eminem] New release The Eminem Show (af71f60c-a8e8-4774-a2b3-30dbfaa13bd6) added

[Eminem] New release The Eminem Show (923de299-21a3-4bf9-89c1-6d45f051a2e0) added

[Eminem] New release The Eminem Show (5c3da192-e366-4b55-8174-85e20f900dbb) added

[Eminem] New release The Eminem Show (40a7080b-40e4-3b24-847a-d21aaee5f414) added

[Eminem] Now adding/updating: The Eminem Show (Comprehensive Force)

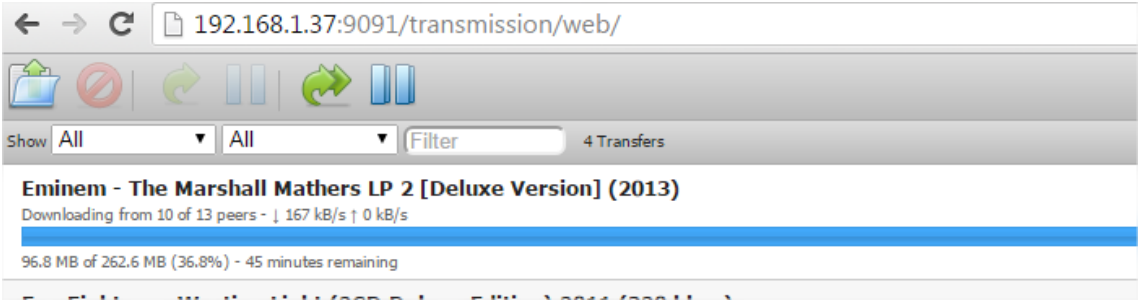
Una vez la sincronización se haya completado aparecerán todos sus álbumes de la siguiente manera


**C. 2** Albumes descargados

Cuando seleccionemos el álbum/canción que queramos Headphones empezará su búsqueda automatizada

```
Search for wanted albums complete
Could not get seed ratio for The Pirate Bay
Torrent folder name: Eminem+-+The+Marshall+Mathers+LP+2+[Deluxe+Version]
+2013+320kbps
Torrent sent to Transmission successfully
Sending torrent to Transmission
Found best result from The Pirate Bay:
 Eminem - The Marshall Mathers LP 2 [Deluxe Version] 2013 320kbps - 250.0 MB
Making sure we can download the best result
Parsing results from Old Pirate Bay
```

Cuando encuentre el mejor resultado se enviará automáticamente a transmisión (C. 3) o a Sabnzbd+ dependiendo de cuál sea el mejor resultado




**C. 3** Descarga automática

Cuando la descarga se haya completado transmisión se comunicara con Headphones para notificarle que la descarga se ha completado, entonces Headphones escaneara el archivo, lo moverá y renombrara, además marcará como descargadas en la biblioteca las canciones

Updating scanned artist track counts

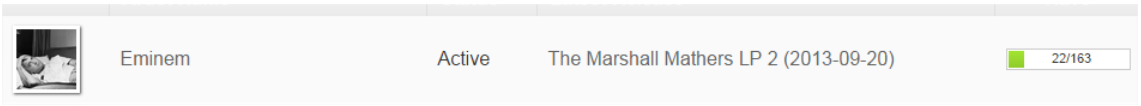
Completed matching tracks from directory: /mnt/music

Track matching is 100.0% complete

Now matching songs by Eminem

Found 34 new/modified tracks in: '/mnt/music'. Matching tracks to the appropriate releases....

34 new/modified songs found and added to the database




**C. 4** Progresion de completada

Lazylibarian utiliza como indexador goodreads.com , a diferencia de musicbrainz de Headphones, goodreads no tiene ninguna limitación de llamadas a su api, debido a que los ebooks no suelen tener una mayor dificultad de encontrarse en Torrents ya que la mayoría de ellos suelen estar en descarga directa, por tanto , Lazylibrarian no vale la pena como descargador automatizado de libros pero en cambio , si que funciona muy bien como biblioteca de libros.

## Anexo D Sickbeard

Sickbeard utiliza como indexador la siguiente pagina http://thetvdb.com/.

Utiliza los metadatos de las series que hayamos seleccionado para luego poder encontrar el archivo NZB correspondiente al capítulo que queramos, a su vez también utiliza las fechas de publicación de los capítulos por si alguno que hayamos seleccionado no existiese , empezar la búsqueda en el momento de

su salida, es decir, en el momento en que el capitulo se publica, Sickbeard empezará a hacer búsquedas periódicas y en el momento en que alguien haya publicado el Torrent o el NZB , el propio Sickbeard iniciará la descarga del archivo NZB para posteriormente enviársela a Sabnzbd.

Sending NZB to Sabnzbd: The.Walking.Dead.S06E08.HDTV.x264-KILLERS

Downloading episode from http://lolo.sickbeard.com/getNZB/b9ae698a64c0c1f0bcf2a97f287d55d2.NZB&i=0&r=f9c9e596a1b33ae68834e40fcb223271

:: Picked The.Walking.Dead.S06E08.HDTV.x264-KILLERS as the best

:: Quality of The.Walking.Dead.S06E08.HDTV.x264-KILLERS is SD TV

Quality of The.Walking.Dead.S06E08.HDTV.x264-KILLERS is SD TV

:: Quality of The.Walking.Dead.S06E08.HDTV.x264-KILLERS is SD TV

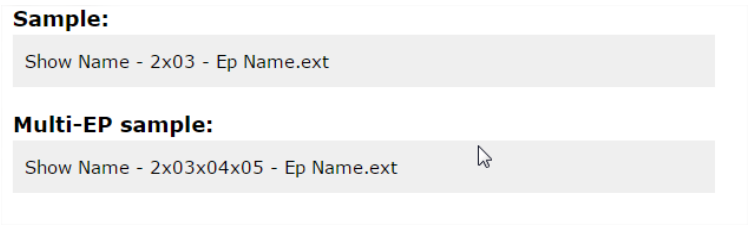
:: Quality of The.Walking.Dead.S06E08.HDTV.x264-KILLERS is SD TV

Provider gave result Little.House.On.The.Prairie.S06E08.iNTERNAL.BDRip.x264-LiBRARiANS but that doesn't seem like a valid result for The Walking Dead so I'm ignoring it

Clearing newtown cache and updating with new information

Searching newtown for The Walking Dead - 6x08 - Start to Finish

Cuando el archivo haya sido descargado por Sabnzbd, este se lo comunicará a Sickbeard , este procesara el archivo y le cambiara el nombre según una mecánica que haya decidido el usuario (D. 1)




**D. 1** Opciones renombración

Después moverá el archivo al *network storage* Tvshows

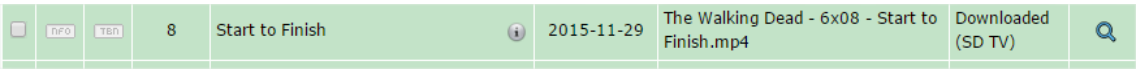
:: Performing refresh on The Walking Dead

Analyzing name u'The.Walking.Dead.S06E08.HDTV.x264-KILLERS.mp4'

Analyzing name None

```
Processing /mnt/Tvshows/Completo/The.Walking.Dead.S06E08.HDTV.x264-KILLERS/The.Walking.Dead.S06E08.HDTV.x264-KILLERS.mp4 (None)
```

Cuando haya acabado el procesado , automáticamente cambiara el estado del capítulo.




**D. 2** Descarga completada

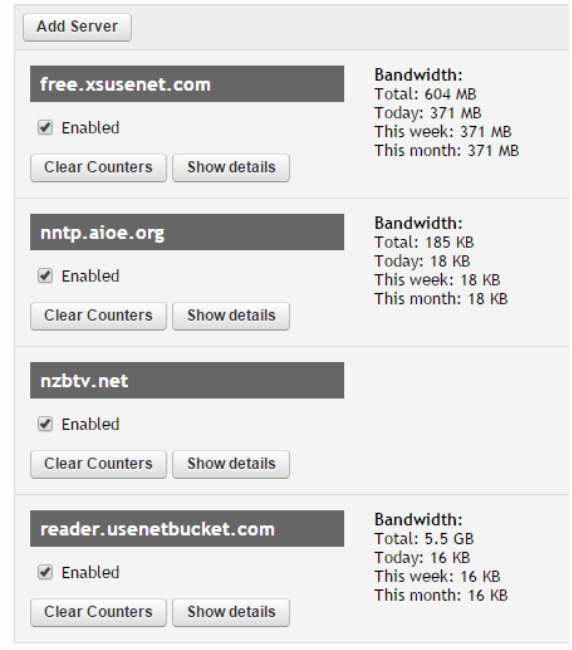
Para que Sickbeard sea capaz de encontrar archivos NBZ será necesario registrarse en las páginas web de proveedores de NZB. Los proveedores utilizados por mi simulación son los de la imagen E. 1 Proveedores Usenet.

| Web | Api-key |
|---|---|
| https://newztown.co.za/ | 021b3596fd871a19819859d47571904f |
| https://NZBtv.net/ | 2798177205c86e8a70e650dd44908104 |
| https://www.ozNZB.com/ | fb4469be69a02ed102d31ad46fa279be |
| http://www.NZBplanet.net/ | b58cd2064f2d2a8376b1ccc01c984c20 |

## Anexo E Sabnzbd

Para que Sabnzbd funcione es necesario tener configurados proveedores de Usenet, sin ellos los archivos NZB son inservibles. La gran mayoría de proveedores de Usenet son de pago, excepto free.xsusenet.com, que es el único de todos que permite acceso gratuito a sus binarios aunque tienen una sola restricción, el ancho de banda

**Proveedores usados**



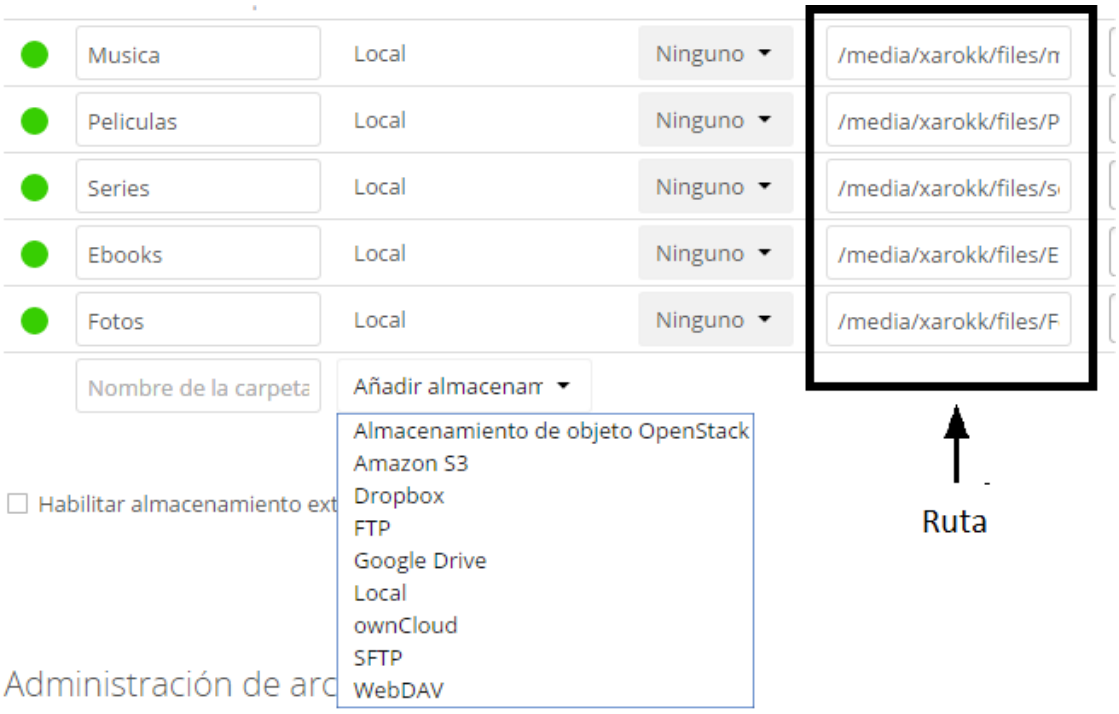

**E. 1** Proveedores Usenet

## Anexo F Owncloud

Owncloud es nuestra *jail* encargada de la gestión de archivos y acceso remoto de archivos.

Es muy parecido a dropbox o drive pero auto hospedado, una diferencia importante es que puedes sincronizar Owncloud con dropbox, drive, servidores ftp , carpetas compartidas en otros ordenadores, Amazon S3, y permite la gestión de usuarios. Para compartir una carpeta que este fuera de la *jail* hay que agregar un modulo de Owncloud que permite la gestión de contenido de almacenamiento externo, sin ese modulo Owncloud es incapaz de sincronizar con ubicaciones de red externas .

Cuando las carpetas hayan sido localizadas en sus respectivas ubicaciones F. 1 de red aparecerá un punto verde indicando que la sincronización ha sido correcta y que Owncloud puede disponer a voluntad de los datos



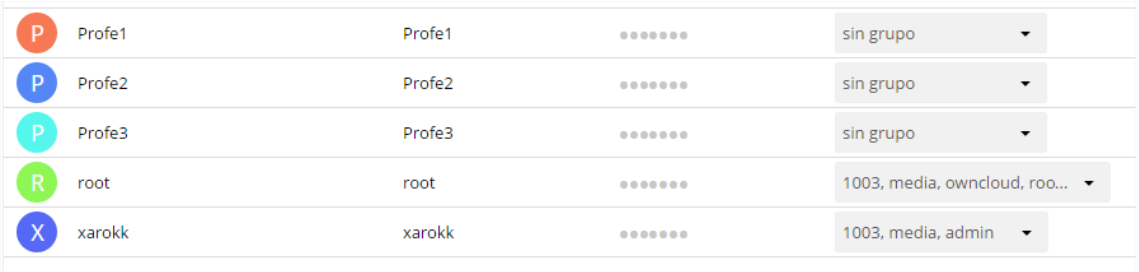

**F. 1** Puntos de montaje Owncloud

Las carpetas compartidas pueden ser por usuario, globales o por grupo y además se les puede asignar cuotas por carpetas o por usuarios , limitando así el espacio utilizado por cada usuario.


**F. 2** Diferencia entre remota y local

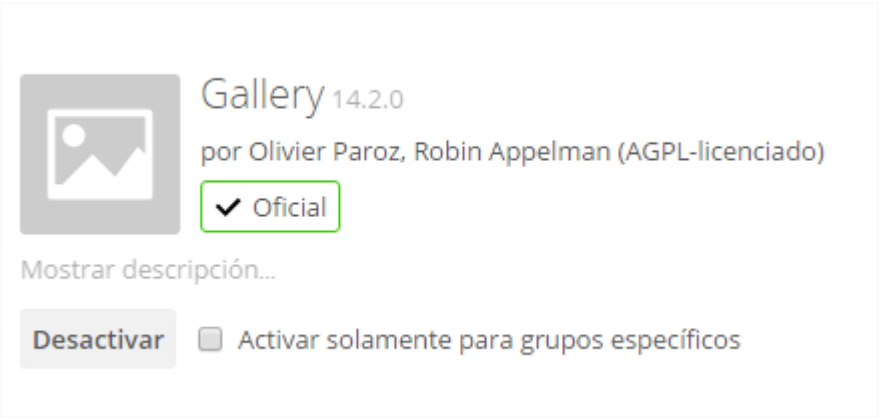
Las carpetas de ubicaciones de red aparecen con un cuadrado y una flecha en ellas, las locales en cambio aparecen con el símbolo de una carpeta azul.

Owncloud permite también la creación de usuarios (F. 3),los cuales aumentan la seguridad, ya que puedes permitir el acceso según que usuario grupo pertenezca




**F. 3** Usuarios y grupos

Para que Owncloud sea capaz de abrir pdf's, abrir words , visionar imágenes , reproducir videos o encriptar es necesario habilitar los modulos correspondientes ya que sin ellos Owncloud simplemente es un gestor de archivos. Por ejemplo en la imagen F.  4 se puede ver el modulo de galería que permite abrir fotografias




**F.  4** Modulo galeria

En el archivo siguiente hay que asignar donde se encuentra la carpeta raíz de toda la información que queramos guardar en local , además también hay que asignar que dominios Ip/nombres de máquina pueden acceder , este archivo es el que se carga a la hora de iniciar el servicio Owncloud.

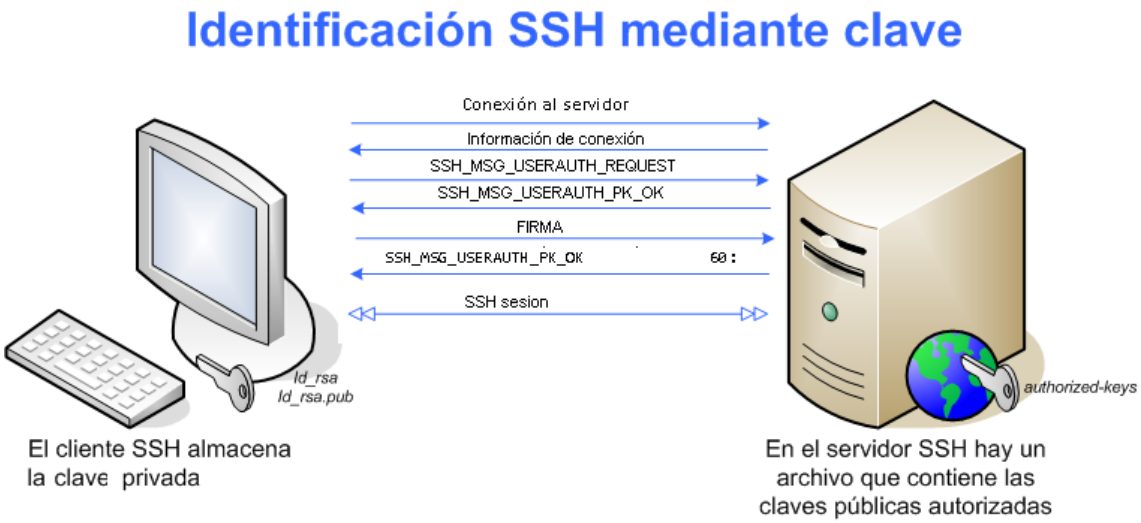
Config.php

```
<?php
$CONFIG = array (
  'memcache.local' => '\\OC\\Memcache\\APCu',
  'instanceid' => 'ocw79m81t6uv',
  'passwordsalt' => 'hISdK4o4I8mg6Fz88cg3IwHSl6A/+a',
  'secret' => 'AuXiOfjEKLyFe1GQv3tkFMUkvLWchN81wA8tsm7sUMg3+Isb',
  'trusted_domains' =>
  array (
    0 => '192.168.1.73',
    1 => '147.83.0.0/16',
  ),
  'datadirectory' => '/media',
  'overwrite.cli.url' => 'https://192.168.1.73',
  'dbtype' => 'sqlite3',
  'version' => '8.2.1.4',
  'logtimezone' => 'UTC',
  'installed' => true,
```

Si se quiere añadir un modulo que no exista en la pestaña de aplicaciones , hay que copiar el modulo descomprimido entero dentro de la carpeta /path/ruta/Owncloud/apps/root
Despues reniciar el servicio y ya se podrá habilitar desde la GUI de Owncloud

## Anexo G: SSh con clave y certificado

Según el rfc4252 el dialogo detallado entre cliente y servidor será el siguiente:G. 1




**G. 1** Diagrama SSH con clave

----------Server----------
Esta información se añade para evitar negociaciones innecesarias

byte SSH_MSG_USERAUTH_REQUEST
string user name in ISO-10646 UTF-8 encoding [RFC3629]
string service name in US-ASCII
string "publickey"
boolean FALSE
string public key algorithm name
string public key blob

----------Cliente----------

string session identifier
byte SSH_MSG_USERAUTH_REQUEST
string user name
string service name
string "publickey"
boolean TRUE

string public key algorithm name
string public key to be used for authentication

Si se acepta el siguiente mensaje

----------Server----------
byte SSH_MSG_USERAUTH_PK_OK
string public key algorithm name from the request
string public key blob from the request

*Si no se acepta*

byte SSH_MSG_USERAUTH_FAILURE

----------Cliente----------

Después el cliente debe enviar una firma generada con la clave privada y hace eso sin verificar si la clave es aceptable.

byte SSH_MSG_USERAUTH_REQUEST
string user name
string service name
string "publickey"
boolean TRUE
string public key algorithm name
string public key to be used for authentication
string signature

El valor signature es lo siguiente

string session identifier
byte SSH_MSG_USERAUTH_REQUEST
string user name
string service name
string "publickey"
boolean TRUE
string public key algorithm name
string public key to be used for authentication

Una vez el servidor haya recibido el mensaje deberá comprobar si la clave suplida es válida para este usuario.

----------Server----------
SSH_MSG_USERAUTH_PK_OK 60:

## Anexo H: Openvpn archivos de configuración

### Ca.crt

-----BEGIN CERTIFICATE-----
MIIE/TCCA+WgAwIBAgIJAJyZAnq/QZ6FMA0GCSqGSIb3DQEBCwUAMIGvM
QswCQYD
VQQGEwJFUzESMBAGA1UECBMJQmFyY2Vsb25hMRIwEAYDVQQHEwlCY
XJjZWxvbmEx
DDAKBgNVBAoTA3VwYzEdMBsGA1UECxMUTXlPcmdhbml6YXRpb25hbFVu
aXQxFjAU
BgNVBAMTDU9wZW52cG5zZXJ2ZXIxDzANBgNVBCkTBnVwY2tleTEiMCAGC
SqGSIb3
DQEJARYTa2N4YXJva2tjQGdtYWlsLmNvbTAeFw0xNTEyMTgxNDA5MzRaFw
0yNTEy
MTUxNDA5MzRaMIGvMQswCQYDVQQGEwJFUzESMBAGA1UECBMJQmFy
Y2Vsb25hMRIw
EAYDVQQHEwlCYXJjZWxvbmExDDAKBgNVBAoTA3VwYzEdMBsGA1UECx
MUTXlPcmdh
bml6YXRpb25hbFVuaXQxFjAUBgNVBAMTDU9wZW52cG5zZXJ2ZXIxDzANB
gNVBCkT
BnVwY2tleTEiMCAGCSqGSIb3DQEJARYTa2N4YXJva2tjQGdtYWlsLmNvbTC
CASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAL/vhrD1A78IjzhDBBhyyP
Azw2uc
pqeQZYntuL05GpyQyl8+PVUD2XHAM5TWUHGOpN/tuIy0sJyqIHrPZv3TsfR/tT
FD
ZBwpf0OPyEqmgcLEMxsvIbA9k1y+CepP47pWBBC/VHx5p0H995anc7xLqNiQ
S7za
P60KDQoS9fJpp1t5qWIDmNgKfYfezgqWd5BbnftBG7h0snroTNGU1155sw7Bv
H+a
g3fuxg1DOo9sDy5d8Elar+Bd2nw8hW6wADdqzOKj3tvMlCYK4iO5V8GsnqpTfB
0m
TgoF69mz7xzDqCVDUgoBor6fIENeIESAggd1qtux/xhGI1fnz73FRKMQpz8CAw
EA
AaOCARgwggEUMB0GA1UdDgQWBBT0vh6uxvpAchld0siYA2iA2flHnjCB5AY
DVR0j
BIHcMIHZgBT0vh6uxvpAchld0siYA2iA2flHnqGBtaSBsjCBrzELMAkGA1UEBhM
C
RVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb25h
MQwwCgYD
VQQKEwN1cGMxHTAbBgNVBAsTFE15T3JnYW5pemF0aW9uYWxVbml0MR
YwFAYDVQQD
Ew1PcGVudnBuc2VydmVyMQ8wDQYDVQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w
0BCQEW
E2tjeGFyb2trY0BnbWFpbC5jb22CCQCcmQJ6v0GehTAMBgNVHRMEBTADA
QH/MA0G
CSqGSIb3DQEBCwUAA4IBAQBzOYRI5J1dOtRQuUFU95po4Sb3YzEUba+2q
HdA+NMo

7gFtWkUhb+R45/h66L/iLB6DB10elRFjJYrSXPGCwusM4Nqr3Y0x+CVAecTUk
HBm
wdObMsU8/1gQUm71c1ndI+6oAxspmIMXSzW4mjMmsUxPp5+uxIIsRvyWRHe
0rdNP
X82mrey2EQByOmwno7m1W5CZtrkUjV179KIrn35XPMW2Y28518jw31iwMAet
awrI
BO1+gGo1dPfk2isYyRoPuB1E4w/q377lA1iHbG2dhkB2mmL5QHatVFC1+hRu4
Qv4
ub6zyr2e1hzdwk1kEfzN9Y9pDaj5LprzWnEQnYGpasR9
-----END CERTIFICATE-----

## Cliente.crt

Certificate:
Data:
Version: 3 (0x2)
Serial Number: 2 (0x2)
Signature Algorithm: sha256WithRSAEncryption
Issuer: C=ES, ST=Barcelona, L=Barcelona, O=upc,
OU=MyOrganizationalUnit,
CN=Openvpnserver/name=upckey/emailAddress=kcxarokkc@gmail.com
Validity
Not Before: Dec 18 14:30:33 2015 GMT
Not After : Dec 15 14:30:33 2025 GMT
Subject: C=ES, ST=Barcelona, L=Barcelona, O=upc,
OU=MyOrganizationalUnit,
CN=David/name=DavidClient/emailAddress=morerasole@gmail.com
Subject Public Key Info:
Public Key Algorithm: rsaEncryption
RSA Public Key: (2048 bit)
Modulus (2048 bit):
00:b3:8e:47:30:9d:d7:73:c1:06:c1:6f:65:c2:5d:
a1:fa:3f:7c:00:c0:33:79:6e:ce:f0:20:00:a5:ff:
c7:97:4b:44:40:23:89:26:a8:52:32:b3:72:da:d2:
d7:e8:78:4b:00:bf:3a:ea:ae:07:d9:ff:d5:4f:fc:
31:83:e9:c5:58:5d:b9:c0:2f:c2:f3:3a:81:54:27:
c3:63:73:a4:56:f5:9a:8d:cc:b8:9e:67:e4:b6:93:
dc:c3:1b:61:ee:9c:a2:89:46:11:99:23:53:fd:42:
1b:c2:c3:09:fd:fc:5c:7f:a3:69:f3:57:d8:ac:40:
79:e0:7f:b9:20:1b:60:9b:97:c5:0f:3c:0b:a7:1d:
65:16:73:08:a0:41:93:2e:7a:ec:29:9a:77:57:1e:
34:0f:77:0e:9b:1e:50:e5:13:9b:8c:b5:b5:e0:79:
6c:4a:47:b0:86:14:83:02:27:ec:ca:1b:46:58:ab:
98:1d:77:4f:ca:40:73:1e:a9:3f:a7:ed:ed:45:db:
ca:19:e0:55:7c:cc:e3:b4:98:20:3a:3d:dc:85:ab:
07:6d:5d:94:c1:35:47:fd:ea:a6:6c:02:6e:7d:59:
26:f5:a2:96:b5:ac:6c:23:a4:fa:84:d7:8e:c7:c2:
be:d1:93:8f:0b:cc:13:c4:75:09:dc:44:3a:55:85:
b1:fb
Exponent: 65537 (0x10001)

```
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                Easy-RSA Generated Certificate
            X509v3 Subject Key Identifier:
                C6:AF:97:D8:EF:62:1E:66:E4:8F:97:68:B6:8B:DC:F1:DC:C2:32:82
            X509v3 Authority Key Identifier:

keyid:F4:BE:1E:AE:C6:FA:40:72:19:5D:D2:C8:98:03:68:80:D9:F9:47:9E

DirName:/C=ES/ST=Barcelona/L=Barcelona/O=upc/OU=MyOrganizationalUnit/
CN=Openvpnserver/name=upckey/emailAddress=kcxarokkc@gmail.com
                serial:9C:99:02:7A:BF:41:9E:85

            X509v3 Extended Key Usage:
                TLS Web Client Authentication
            X509v3 Key Usage:
                Digital Signature
    Signature Algorithm: sha256WithRSAEncryption
        ba:fa:65:45:3e:9c:e9:9b:e0:a8:27:66:f3:59:60:b6:1a:7a:
        79:a8:25:b6:5f:6b:1e:4c:bd:2f:a5:e5:0d:d7:1b:53:d6:64:
        cd:8d:e3:0b:01:e3:4b:aa:b5:96:ee:2e:57:2d:58:82:15:60:
        11:32:3d:a5:b6:76:f7:4f:82:a9:e3:39:5e:33:08:fd:d0:65:
        27:fe:5b:1d:de:ea:e2:35:a1:72:7f:34:85:f4:d3:cd:a5:61:
        54:10:a1:56:af:cf:0d:b3:3c:84:a9:a6:6b:0c:a5:bc:8e:be:
        5a:3b:96:03:1e:89:4f:87:92:40:7c:a6:59:73:bf:2c:ab:f6:
        37:86:e0:6d:4c:b2:52:86:b6:2c:01:cb:1b:1c:9e:8c:9b:6c:
        27:00:9a:d1:ce:6a:47:a5:a0:1b:0c:65:25:2a:6e:8d:6b:ec:
        f1:55:ee:7d:92:29:a1:1b:7c:3c:c0:52:91:e8:9b:36:6a:47:
        1e:9d:e0:4d:82:a8:af:77:f7:e2:3b:26:90:a7:2c:94:d3:d3:
        65:9e:79:71:10:96:90:eb:9e:9d:98:79:08:1e:87:7a:df:c2:
        e7:eb:99:c0:2a:8a:a3:d4:1d:54:28:8c:84:a6:16:b2:04:03:
        db:d4:9d:50:81:f5:1d:bc:64:e4:f7:27:91:28:ac:47:6e:c3:
        a4:f7:dd:51
-----BEGIN CERTIFICATE-----
MIIFQTCCBCmgAwIBAgIBAjANBgkqhkiG9w0BAQsFADCBrzELMAkGA1UEBh
MCRVMx
EjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb25hMQw
wCgYDVQQK
EwN1cGMxHTAbBgNVBAsTFE15T3JnYW5pemF0aW9uYWxVbml0MRYwFAY
DVQQDEw1P
cGVudnBuc2VydmVyMQ8wDQYDVQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w0BCQ
EWE2tj
eGFyb2trY0BnbWFpbC5jb20wHhcNMTUxMjE4MTQzMDMzWhcNMjUxMjE1M
TQzMDMz
WjCBrTELMAkGA1UEBhMCRVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBA
GA1UEBxMJ
QmFyY2Vsb25hMQwwCgYDVQQKEwN1cGMxHTAbBgNVBAsTFE15T3JnYW
5pemF0aW9u
```

YWxVbml0MQ4wDAYDVQQDEwVEYXZpZDEUMBIGA1UEKRMLRGF2aWRD
bGllbnQxIzAh
BgkqhkiG9w0BCQEWFG1vcmVyYXNvbGVAZ21haWwuY29tMIIBIjANBgkqhki
G9w0B
AQEFAAOCAQ8AMIIBCgKCAQEAs45HMJ3Xc8EGwW9lwl2h+j98AMAzeW7O
8CAApf/H
l0tEQCOJJqhSMrNy2tLX6HhLAL866q4H2f/VT/wxg+nFWF25wC/C8zqBVCfDY
3Ok
VvWajcy4nmfktpPcwxth7pyiiUYRmSNT/UIbwsMJ/fxcf6Np81fYrEB54H+5IBtg
m5fFDzwLpx1lFnMIoEGTLnrsKZp3Vx40D3cOmx5Q5RObjLW14HlsSkewhhSD
Aifs
yhtGWKuYHXdPykBzHqk/p+3tRdvKGeBVfMzjtJggOj3chasHbV2UwTVH/eqmb
AJu
fVkm9aKWtaxsI6T6hNeOx8K+0ZOPC8wTxHUJ3EQ6VYWx+wIDAQABo4IBZj
CCAWIw
CQYDVR0TBAIwADAtBglghkgBhvhCAQ0EIBYeRWFzeS1SU0EgR2VuZXJhd
GVkIENl
cnRpZmljYXRlMB0GA1UdDgQWBBTGr5fY72IeZuSPl2i2i9zx3MIygjCB5AYDV
R0j
BIHcMIHZgBT0vh6uxvpAchld0siYA2iA2flHnqGBtaSBsjCBrzELMAkGA1UEBhM
C
RVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb25h
MQwwCgYD
VQQKEwN1cGMxHTAbBgNVBAsTFE15T3JnYW5pemF0aW9uYWxVbml0MR
YwFAYDVQQD
Ew1PcGVudnBuc2VydmVyMQ8wDQYDVQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w
0BCQEW
E2tjeGFyb2trY0BnbWFpbC5jb22CCQCcmQJ6v0GehTATBgNVHSUEDDAKBg
grBgEF
BQcDAjALBgNVHQ8EBAMCB4AwDQYJKoZIhvcNAQELBQADggEBALr6ZUU+
nOmb4Kgn
ZvNZYLYaenmoJbZfax5MvS+l5Q3XG1PWZM2N4wsB40uqtZbuLlctWIIVYBEy
PaW2
dvdPgqnjOV4zCP3QZSf+Wx3e6uI1oXJ/NIX0082lYVQQoVavzw2zPISppmsMp
byO
vlo7lgMeiU+HkkB8pllzvyyr9jeG4G1MslKGtiwByxscnoybbCcAmtHOakeloBsM
ZSUqbo1r7PFV7n2SKaEbfDzAUpHomzZqRx6d4E2CqK939+I7JpCnLJTT02W
eeXEQ
lpDrnp2YeQgeh3rfwufrmcAqiqPUHVQojISmFrIEA9vUnVCB9R28ZOT3J5EorEd
u
w6T33VE=
-----END CERTIFICATE-----

## Cliente.key

-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAs45HMJ3Xc8EGwW9lwl2h+j98AMAzeW7O8CAApf/Hl0tE
QCOJ
JqhSMrNy2tLX6HhLAL866q4H2f/VT/wxg+nFWF25wC/C8zqBVCfDY3OkVvWaj
cy4
nmfktpPcwxth7pyiiUYRmSNT/UIbwsMJ/fxcf6Np81fYrEB54H+5IBtgm5fFDzwL

px1lFnMIoEGTLnrsKZp3Vx40D3cOmx5Q5RObjLW14HlsSkewhhSDAifsyhtGW
KuY
HXdPykBzHqk/p+3tRdvKGeBVfMzjtJggOj3chasHbV2UwTVH/eqmbAJufVkm9a
KW
taxsI6T6hNeOx8K+0ZOPC8wTxHUJ3EQ6VYWx+wIDAQABAoIBABUc064ZwS
2xMrQn
/xUdoWAletUzxFmw99AwUyMxsixOA1ZvVz8eZ5vikcy9IfoJuvf5hFn34KB3foOL
Kd9S64t6ac75Trj+1hopGsZk4JwBAQ1PVx35JWF6fWjLnXZLIKl3fbY9+zA6R71
c
5Z3hwJaZhmnAmI+YfTwyiHgIABILUgrUr2b8cce+PS8kvu8fue8LFkvuZV4HTJE
C
E+rbQxLKquHigZ9ON52xJCozLzUM92S4K+jeg2a4tsq0itTo5EQ6SEwG7+F5X
3HU
FufxBWFSu1bh7A+Whgc0AHDatA9VRBt0aYtvyWsZ/5gPdDSJ92nkhQm5QAE
+Uv7+
nCUfGtECgYEA17xFeYixn3n2Veveuol6YAxHHDIAwEPeY9SMfhvnQEuuncC8/
Xtt
XBI9W5cxKyMPGx13FIJcaHZHJsI1s/DXWJo9066RlJssrDNcXxXhf5bV0XA752
78
PcdKg0kwPHIa5A2BGgDDInUDL369yPaP9d0k7s8njg6Mdfz4vMdLx0cCgYEA1
RFc
aw/gFf/iH7CI6jCCAdLggA2BVmn4p9eOq3zb5dmFJSDRxup/eTKizXpEYU3ggH
yl
JLFIVz3HSKjhp7SSlqBewlGNvkbGY+oYDQR4n5AES8tkJ2QTg1iChBzNpaa9Q
Aac
PP58B3oHjwZl5/ftnubhVwKQIjsc/gCYZGIRQa0CgYAqtuxKk/YXTV6oxHrlFDN
W
sRejTYM4jjeKLJtknMG8yT9ZnL/OsJGGmVXsvJLPp7Fl/P+G4AjL5h9QXUOkzR
rW
QDPdpcLNWawOrFyttvde+856GW7C9AaddMgcFnGmhjEs1j+pll0MhX6L+Nw6
3hbC
53ZYkkuSpO7KrbvSJOOsIQKBgBIHAwPH9xTFbrNxs1PlQMq4tef/vIa8Np603U
xj
JxPegKZwjW9AHgL6Js4t3yVAepNEdhxsp4tCHd9m8pjG7XQzI3FbhxunT1fa0a
gg
TbhqbgKftdyjWgdN8NUOvtOaNP760DNU8NeNGqCeUPb8gi1kqYI8JEfRKpXP
JRWx
OnaNAoGBAM1n4uPrUHJ+xKDqrxaQuPTWrGzMXxFMiNGGfUbGy1Nx0QY9G
OXMPIjE
XB2shRq9j1W6+/ERDDCb8KmYN6QNrHGYolrxzyBg+eFxK3cVJQwiSb07F/4xr
Dv4
LHxQL5Y77Ix9Ks6CcM0mSQpgkhI0VugegCEbKWd+tMyLu3WcjnZU
-----END RSA PRIVATE KEY-----

## Cliente.conf

```
#################################################
# Sample client-side OpenVPN 2.0 config file #
# for connecting to multi-client server.     #
#                                            #
```

```
# This configuration can be used by multiple #
# clients, however each client should have   #
# its own cert and key files.                #
#                                            #
# On Windows, you might want to rename this  #
# file so it has a .ovpn extension           #
##############################################

# Specify that we are a client and that we
# will be pulling certain config file directives
# from the server.
client

# Use the same setting as you are using on
# the server.
# On most systems, the VPN will not function
# unless you partially or fully disable
# the firewall for the TUN/TAP interface.
;dev tap
dev tun

# Windows needs the TAP-Win32 adapter name
# from the Network Connections panel
# if you have more than one.  On XP SP2,
# you may need to disable the firewall
# for the TAP adapter.
;dev-node MyTap

# Are we connecting to a TCP or
# UDP server?  Use the same setting as
# on the server.
;proto tcp
proto udp

# The hostname/IP and port of the server.
# You can have multiple remote entries
# to load balance between the servers.
;remote my-server-1 1194
;remote my-server-2 1194
remote 88.12.141.22 1194
# Choose a random host from the remote
# list for load-balancing.  Otherwise
# try hosts in the order specified.
;remote-random

# Keep trying indefinitely to resolve the
# host name of the OpenVPN server.  Very useful
# on machines which are not permanently connected
# to the internet such as laptops.
resolv-retry infinite
```

```
# Most clients don't need to bind to
# a specific local port number.
nobind

# Downgrade privileges after initialization (non-Windows only)
;user nobody
;group nobody

# Try to preserve some state across restarts.
persist-key
persist-tun

# If you are connecting through an
# HTTP proxy to reach the actual OpenVPN
# server, put the proxy server/IP and
# port number here.  See the man page
# if your proxy server requires
# authentication.
;http-proxy-retry # retry on connection failures
;http-proxy [proxy server] [proxy port #]

# Wireless networks often produce a lot
# of duplicate packets.  Set this flag
# to silence duplicate packet warnings.
;mute-replay-warnings

# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
ca ca.crt
cert David.crt
key David.key

# Verify server certificate by checking that the
# certicate has the correct key usage set.
# This is an important precaution to protect against
# a potential attack discussed here:
#  http://openvpn.net/howto.html#mitm
#
# To use this feature, you will need to generate
# your server certificates with the keyUsage set to
#   digitalSignature, keyEncipherment
# and the extendedKeyUsage to
#   serverAuth
# EasyRSA can do this for you.
remote-cert-tls server
```

```
# If a tls-auth key is used on the server
# then every client must also have the key.
;tls-auth ta.key 1

# Select a cryptographic cipher.
# If the cipher option is used on the server
# then you must also specify it here.
cipher aes-128-cbc

# Enable compression on the VPN link.
# Don't enable this unless it is also
# enabled in the server config file.
;comp-lzo

# Set log file verbosity.
verb 3

# Silence repeating messages
;mute 20

<ca>
-----BEGIN CERTIFICATE-----
MIIE/TCCA+WgAwIBAgIJAJyZAnq/QZ6FMA0GCSqGSIb3DQEBCwUAMIGvM
QswCQYD
VQQGEwJFUzESMBAGA1UECBMJQmFyY2Vsb25hMRIwEAYDVQQHEwlCY
XJjZWxvbmEx
DDAKBgNVBAoTA3VwYzEdMBsGA1UECxMUTXlPcmdhbml6YXRpb25hbFVu
aXQxFjAU
BgNVBAMTDU9wZW52cG5zZXJ2ZXIxDzANBgNVBCkTBnVwY2tleTEiMCAGC
SqGSIb3
DQEJARYTa2N4YXJva2tjQGdtYWlsLmNvbTAeFw0xNTEyMTgxNDA5MzRaFw
0yNTEy
MTUxNDA5MzRaMIGvMQswCQYDVQQGEwJFUzESMBAGA1UECBMJQmFy
Y2Vsb25hMRIw
EAYDVQQHEwlCYXJjZWxvbmExDDAKBgNVBAoTA3VwYzEdMBsGA1UECx
MUTXlPcmdh
bml6YXRpb25hbFVuaXQxFjAUBgNVBAMTDU9wZW52cG5zZXJ2ZXIxDzANB
gNVBCkT
BnVwY2tleTEiMCAGCSqGSIb3DQEJARYTa2N4YXJva2tjQGdtYWlsLmNvbTC
CASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAL/vhrD1A78IjzhDBBhyyP
Azw2uc
pqeQZYntuL05GpyQyl8+PVUD2XHAM5TWUHGOpN/tuIy0sJyqIHrPZv3TsfR/tT
FD
ZBwpf0OPyEqmgcLEMxsvIbA9k1y+CepP47pWBBC/VHx5p0H995anc7xLqNiQ
S7za
P60KDQoS9fJpp1t5qWIDmNgKfYfezgqWd5BbnftBG7h0snroTNGU1155sw7Bv
H+a
```

g3fuxg1DOo9sDy5d8Elar+Bd2nw8hW6wADdqzOKj3tvMlCYK4iO5V8GsnqpTfB
0m
TgoF69mz7xzDqCVDUgoBor6fIENeIESAggd1qtux/xhGI1fnz73FRKMQpz8CAw
EA
AaOCARgwggEUMB0GA1UdDgQWBBT0vh6uxvpAchld0siYA2iA2flHnjCB5AY
DVR0j
BIHcMIHZgBT0vh6uxvpAchld0siYA2iA2flHnqGBtaSBsjCBrzELMAkGA1UEBhM
C
RVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb25h
MQwwCgYD
VQQKEwN1cGMxHTAbBgNVBAsTFE15T3JnYW5pemF0aW9uYWxVbml0MR
YwFAYDVQQD
Ew1PcGVudnBuc2VydmVyMQ8wDQYDVQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w
0BCQEW
E2tjeGFyb2trY0BnbWFpbC5jb22CCQCcmQJ6v0GehTAMBgNVHRMEBTADA
QH/MA0G
CSqGSIb3DQEBCwUAA4IBAQBzOYRI5J1dOtRQuUFU95po4Sb3YzEUba+2q
HdA+NMo
7gFtWkUhb+R45/h66L/iLB6DB10elRFjJYrSXPGCwusM4Nqr3Y0x+CVAecTUk
HBm
wdObMsU8/1gQUm71c1ndI+6oAxspmIMXSzW4mjMmsUxPp5+uxIIsRvyWRHe
0rdNP
X82mrey2EQByOmwno7m1W5CZtrkUjV179KIrn35XPMW2Y28518jw31iwMAet
awrI
BO1+gGo1dPfk2isYyRoPuB1E4w/q377lA1iHbG2dhkB2mmL5QHatVFC1+hRu4
Qv4
ub6zyr2e1hzdwk1kEfzN9Y9pDaj5LprzWnEQnYGpasR9
-----END CERTIFICATE-----
</ca>

<cert>
-----BEGIN CERTIFICATE-----
MIIFQTCCBCmgAwIBAgIBAjANBgkqhkiG9w0BAQsFADCBrzELMAkGA1UEBh
MCRVMx
EjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb25hMQw
wCgYDVQQK
EwN1cGMxHTAbBgNVBAsTFE15T3JnYW5pemF0aW9uYWxVbml0MRYwFAY
DVQQDEw1P
cGVudnBuc2VydmVyMQ8wDQYDVQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w0BCQ
EWE2tj
eGFyb2trY0BnbWFpbC5jb20wHhcNMTUxMjE4MTQzMDMzWhcNMjUxMjE1M
TQzMDMz
WjCBrTELMAkGA1UEBhMCRVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBA
GA1UEBxMJ
QmFyY2Vsb25hMQwwCgYDVQQKEwN1cGMxHTAbBgNVBAsTFE15T3JnYW
5pemF0aW9u
YWxVbml0MQ4wDAYDVQQDEwVEYXZpZDEUMBIGA1UEKRMLRGF2aWRD
bGllbnQxIzAh
BgkqhkiG9w0BCQEWFG1vcmVyYXNvbGVAZ21haWwuY29tMIIBIjANBgkqhki
G9w0B

AQEFAAOCAQ8AMIIBCgKCAQEAs45HMJ3Xc8EGwW9lwl2h+j98AMAzeW7O
8CAApf/H
l0tEQCOJJqhSMrNy2tLX6HhLAL866q4H2f/VT/wxg+nFWF25wC/C8zqBVCfDY
3Ok
VvWajcy4nmfktpPcwxth7pyiiUYRmSNT/UIbwsMJ/fxcf6Np81fYrEB54H+5IBtg
m5fFDzwLpx1lFnMIoEGTLnrsKZp3Vx40D3cOmx5Q5RObjLW14HlsSkewhhSD
Aifs
yhtGWKuYHXdPykBzHqk/p+3tRdvKGeBVfMzjtJggOj3chasHbV2UwTVH/eqmb
AJu
fVkm9aKWtaxsI6T6hNeOx8K+0ZOPC8wTxHUJ3EQ6VYWx+wIDAQABo4IBZj
CCAWIw
CQYDVR0TBAIwADAtBglghkgBhvhCAQ0EIBYeRWFzeS1SU0EgR2VuZXJhd
GVkIENl
cnRpZmljYXRlMB0GA1UdDgQWBBTGr5fY72IeZuSPl2i2i9zx3MIygjCB5AYDV
R0j
BIHcMIHZgBT0vh6uxvpAchld0siYA2iA2flHnqGBtaSBsjCBrzELMAkGA1UEBhM
C
RVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb25h
MQwwCgYD
VQQKEwN1cGMxHTAbBgNVBAsTFE15T3JnYW5pemF0aW9uYWxVbml0MR
YwFAYDVQQD
Ew1PcGVudnBuc2VydmVyMQ8wDQYDVQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w
0BCQEW
E2tjeGFyb2trY0BnbWFpbC5jb22CCQCcmQJ6v0GehTATBgNVHSUEDDAKBg
grBgEF
BQcDAjALBgNVHQ8EBAMCB4AwDQYJKoZIhvcNAQELBQADggEBALr6ZUU+
nOmb4Kgn
ZvNZYLYaenmoJbZfax5MvS+l5Q3XG1PWZM2N4wsB40uqtZbuLlctWIIVYBEy
PaW2
dvdPgqnjOV4zCP3QZSf+Wx3e6uI1oXJ/NIX0082lYVQQoVavzw2zPISppmsMp
byO
vlo7lgMeiU+HkkB8pllzvyyr9jeG4G1MslKGtiwByxscnoybbCcAmtHOakeloBsM
ZSUqbo1r7PFV7n2SKaEbfDzAUpHomzZqRx6d4E2CqK939+I7JpCnLJTT02W
eeXEQ
lpDrnp2YeQgeh3rfwufrmcAqiqPUHVQojISmFrIEA9vUnVCB9R28ZOT3J5EorEd
u
w6T33VE=
-----END CERTIFICATE-----
</cert>

<key>
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAs45HMJ3Xc8EGwW9lwl2h+j98AMAzeW7O8CAApf/Hl0tE
QCOJ
JqhSMrNy2tLX6HhLAL866q4H2f/VT/wxg+nFWF25wC/C8zqBVCfDY3OkVvWaj
cy4
nmfktpPcwxth7pyiiUYRmSNT/UIbwsMJ/fxcf6Np81fYrEB54H+5IBtgm5fFDzwL
px1lFnMIoEGTLnrsKZp3Vx40D3cOmx5Q5RObjLW14HlsSkewhhSDAifsyhtGW
KuY

HXdPykBzHqk/p+3tRdvKGeBVfMzjtJggOj3chasHbV2UwTVH/eqmbAJufVkm9a
KW
taxsI6T6hNeOx8K+0ZOPC8wTxHUJ3EQ6VYWx+wIDAQABAoIBABUc064ZwS
2xMrQn
/xUdoWAletUzxFmw99AwUyMxsixOA1ZvVz8eZ5vikcy9IfoJuvf5hFn34KB3foOL
Kd9S64t6ac75Trj+1hopGsZk4JwBAQ1PVx35JWF6fWjLnXZLIKl3fbY9+zA6R71
c
5Z3hwJaZhmnAmI+YfTwyiHgIABILUgrUr2b8cce+PS8kvu8fue8LFkvuZV4HTJE
C
E+rbQxLKquHigZ9ON52xJCozLzUM92S4K+jeg2a4tsq0itTo5EQ6SEwG7+F5X
3HU
FufxBWFSu1bh7A+Whgc0AHDatA9VRBt0aYtvyWsZ/5gPdDSJ92nkhQm5QAE
+Uv7+
nCUfGtECgYEA17xFeYixn3n2Veveuol6YAxHHDIAwEPeY9SMfhvnQEuuncC8/
Xtt
XBI9W5cxKyMPGx13FIJcaHZHJsI1s/DXWJo9066RlJssrDNcXxXhf5bV0XA752
78
PcdKg0kwPHIa5A2BGgDDInUDL369yPaP9d0k7s8njg6Mdfz4vMdLx0cCgYEA1
RFc
aw/gFf/iH7CI6jCCAdLggA2BVmn4p9eOq3zb5dmFJSDRxup/eTKizXpEYU3ggH
yl
JLFIVz3HSKjhp7SSlqBewlGNvkbGY+oYDQR4n5AES8tkJ2QTg1iChBzNpaa9Q
Aac
PP58B3oHjwZl5/ftnubhVwKQIjsc/gCYZGIRQa0CgYAqtuxKk/YXTV6oxHrlFDN
W
sRejTYM4jjeKLJtknMG8yT9ZnL/OsJGGmVXsvJLPp7Fl/P+G4AjL5h9QXUOkzR
rW
QDPdpcLNWawOrFyttvde+856GW7C9AaddMgcFnGmhjEs1j+pll0MhX6L+Nw6
3hbC
53ZYkkuSpO7KrbvSJOOsIQKBgBIHAwPH9xTFbrNxs1PlQMq4tef/vIa8Np603U
xj
JxPegKZwjW9AHgL6Js4t3yVAepNEdhxsp4tCHd9m8pjG7XQzI3FbhxunT1fa0a
gg
TbhqbgKftdyjWgdN8NUOvtOaNP760DNU8NeNGqCeUPb8gi1kqYI8JEfRKpXP
JRWx
OnaNAoGBAM1n4uPrUHJ+xKDqrxaQuPTWrGzMXxFMiNGGfUbGy1Nx0QY9G
OXMPIjE
XB2shRq9j1W6+/ERDDCb8KmYN6QNrHGYolrxzyBg+eFxK3cVJQwiSb07F/4xr
Dv4
LHxQL5Y77Ix9Ks6CcM0mSQpgkhI0VugegCEbKWd+tMyLu3WcjnZU
-----END RSA PRIVATE KEY-----
</key>

## Server.conf

```
#########################################################
# Sample OpenVPN 2.0 config file for            #
# multi-client server.                          #
#                                               #
# This file is for the server side              #
```

```
# of a many-clients <-> one-server              #
# OpenVPN configuration.                        #
#                                               #
# OpenVPN also supports                         #
# single-machine <-> single-machine             #
# configurations (See the Examples page         #
# on the web site for more info).               #
#                                               #
# This config should work on Windows            #
# or Linux/BSD systems.  Remember on            #
# Windows to quote pathnames and use            #
# double backslashes, e.g.:                     #
# "C:\\Program Files\\OpenVPN\\config\\foo.key" #
#                                               #
# Comments are preceded with '#' or ';'         #
#################################################

# Which local IP address should OpenVPN
# listen on? (optional)
;local a.b.c.d

# Which TCP/UDP port should OpenVPN listen on?
# If you want to run multiple OpenVPN instances
# on the same machine, use a different port
# number for each one.  You will need to
# open up this port on your firewall.
port 1194

# TCP or UDP server?
;proto tcp
proto udp

# "dev tun" will create a routed IP tunnel,
# "dev tap" will create an ethernet tunnel.
# Use "dev tap0" if you are ethernet bridging
# and have precreated a tap0 virtual interface
# and bridged it with your ethernet interface.
# If you want to control access policies
# over the VPN, you must create firewall
# rules for the the TUN/TAP interface.
# On non-Windows systems, you can give
# an explicit unit number, such as tun0.
# On Windows, use "dev-node" for this.
# On most systems, the VPN will not function
# unless you partially or fully disable
# the firewall for the TUN/TAP interface.
;dev tap
dev tun

# Windows needs the TAP-Win32 adapter name
```

```
# from the Network Connections panel if you
# have more than one.  On XP SP2 or higher,
# you may need to selectively disable the
# Windows firewall for the TAP adapter.
# Non-Windows systems usually don't need this.
;dev-node MyTap

# SSL/TLS root certificate (ca), certificate
# (cert), and private key (key).  Each client
# and the server must have their own cert and
# key file.  The server and all clients will
# use the same ca file.
#
# See the "easy-rsa" directory for a series
# of scripts for generating RSA certificates
# and private keys.  Remember to use
# a unique Common Name for the server
# and each of the client certificates.
#
# Any X509 key management system can be used.
# OpenVPN can also use a PKCS #12 formatted key file
# (see "pkcs12" directive in man page).
ca ca.crt
cert server.crt
key server.key  # This file should be kept secret

# Diffie hellman parameters.
# Generate your own with:
#   openssl dhparam -out dh2048.pem 2048
dh dh2048.pem

# Network topology
# Should be subnet (addressing via IP)
# unless Windows clients v2.0.9 and lower have to
# be supported (then net30, i.e. a /30 per client)
# Defaults to net30 (not recommended)
;topology subnet

# Configure server mode and supply a VPN subnet
# for OpenVPN to draw client addresses from.
# The server will take 10.8.0.1 for itself,
# the rest will be made available to clients.
# Each client will be able to reach the server
# on 10.8.0.1. Comment this line out if you are
# ethernet bridging. See the man page for more info.
server 10.8.0.0 255.255.255.0
;server 192.168.1.242 255.255.255.0
# Maintain a record of client <-> virtual IP address
# associations in this file.  If OpenVPN goes down or
# is restarted, reconnecting clients can be assigned
```

```
# the same virtual IP address from the pool that was
# previously assigned.
ifconfig-pool-persist ipp.txt

# Configure server mode for ethernet bridging.
# You must first use your OS's bridging capability
# to bridge the TAP interface with the ethernet
# NIC interface.  Then you must manually set the
# IP/netmask on the bridge interface, here we
# assume 10.8.0.4/255.255.255.0.  Finally we
# must set aside an IP range in this subnet
# (start=10.8.0.50 end=10.8.0.100) to allocate
# to connecting clients.  Leave this line commented
# out unless you are ethernet bridging.
;server-bridge 10.8.0.4 255.255.255.0 10.8.0.50 10.8.0.100
;server-bridge 192.168.1.240 255.255.255.0 192.168.1.245 192.168.1.250
# Configure server mode for ethernet bridging
# using a DHCP-proxy, where clients talk
# to the OpenVPN server-side DHCP server
# to receive their IP address allocation
# and DNS server addresses.  You must first use
# your OS's bridging capability to bridge the TAP
# interface with the ethernet NIC interface.
# Note: this mode only works on clients (such as
# Windows), where the client-side TAP adapter is
# bound to a DHCP client.
server-bridge

# Push routes to the client to allow it
# to reach other private subnets behind
# the server.  Remember that these
# private subnets will also need
# to know to route the OpenVPN client
# address pool (10.8.0.0/255.255.255.0)
# back to the OpenVPN server.
push "route 192.168.1.0 255.255.255.0"
;push "route 192.168.20.0 255.255.255.0"

# To assign specific IP addresses to specific
# clients or if a connecting client has a private
# subnet behind it that should also have VPN access,
# use the subdirectory "ccd" for client-specific
# configuration files (see man page for more info).

# EXAMPLE: Suppose the client
# having the certificate common name "Thelonious"
# also has a small subnet behind his connecting
# machine, such as 192.168.40.128/255.255.255.248.
# First, uncomment out these lines:
;client-config-dir ccd
```

```
;route 192.168.40.128 255.255.255.248
# Then create a file ccd/Thelonious with this line:
#   iroute 192.168.40.128 255.255.255.248
# This will allow Thelonious' private subnet to
# access the VPN.  This example will only work
# if you are routing, not bridging, i.e. you are
# using "dev tun" and "server" directives.

# EXAMPLE: Suppose you want to give
# Thelonious a fixed VPN IP address of 10.9.0.1.
# First uncomment out these lines:
;client-config-dir ccd
;route 10.9.0.0 255.255.255.252
# Then add this line to ccd/Thelonious:
#   ifconfig-push 10.9.0.1 10.9.0.2

# Suppose that you want to enable different
# firewall access policies for different groups
# of clients.  There are two methods:
# (1) Run multiple OpenVPN daemons, one for each
#     group, and firewall the TUN/TAP interface
#     for each group/daemon appropriately.
# (2) (Advanced) Create a script to dynamically
#     modify the firewall in response to access
#     from different clients.  See man
#     page for more info on learn-address script.
;learn-address ./script

# If enabled, this directive will configure
# all clients to redirect their default
# network gateway through the VPN, causing
# all IP traffic such as web browsing and
# and DNS lookups to go through the VPN
# (The OpenVPN server machine may need to NAT
# or bridge the TUN/TAP interface to the internet
# in order for this to work properly).
push "redirect-gateway def1 bypass-dhcp"

# Certain Windows-specific network settings
# can be pushed to clients, such as DNS
# or WINS server addresses.  CAVEAT:
# http://openvpn.net/faq.html#dhcpcaveats
# The addresses below refer to the public
# DNS servers provided by opendns.com.
push "dhcp-option DNS 80.58.61.250"
push "dhcp-option DNS 80.58.61.254"

# Uncomment this directive to allow different
# clients to be able to "see" each other.
# By default, clients will only see the server.
```

```
# To force clients to only see the server, you
# will also need to appropriately firewall the
# server's TUN/TAP interface.
client-to-client

# Uncomment this directive if multiple clients
# might connect with the same certificate/key
# files or common names.  This is recommended
# only for testing purposes.  For production use,
# each client should have its own certificate/key
# pair.
#
# IF YOU HAVE NOT GENERATED INDIVIDUAL
# CERTIFICATE/KEY PAIRS FOR EACH CLIENT,
# EACH HAVING ITS OWN UNIQUE "COMMON NAME",
# UNCOMMENT THIS LINE OUT.
duplicate-cn

# The keepalive directive causes ping-like
# messages to be sent back and forth over
# the link so that each side knows when
# the other side has gone down.
# Ping every 10 seconds, assume that remote
# peer is down if no ping received during
# a 120 second time period.
keepalive 10 120

# For extra security beyond that provided
# by SSL/TLS, create an "HMAC firewall"
# to help block DoS attacks and UDP port flooding.
#
# Generate with:
#   openvpn --genkey --secret ta.key
#
# The server and each client must have
# a copy of this key.
# The second parameter should be '0'
# on the server and '1' on the clients.
;tls-auth ta.key 0 # This file is secret

# Select a cryptographic cipher.
# This config item must be copied to
# the client config file as well.
;cipher BF-CBC        # Blowfish (default)
cipher AES-128-CBC   # AES
;cipher DES-EDE3-CBC  # Triple-DES

# Enable compression on the VPN link.
# If you enable it here, you must also
# enable it in the client config file.
```

```
comp-lzo

# The maximum number of concurrently connected
# clients we want to allow.
;max-clients 100

# It's a good idea to reduce the OpenVPN
# daemon's privileges after initialization.
#
# You can uncomment this out on
# non-Windows systems.
user nobody
group nobody

# The persist options will try to avoid
# accessing certain resources on restart
# that may no longer be accessible because
# of the privilege downgrade.
persist-key
persist-tun

# Output a short status file showing
# current connections, truncated
# and rewritten every minute.
status openvpn-status.log

# By default, log messages will go to the syslog (or
# on Windows, if running as a service, they will go to
# the "\Program Files\OpenVPN\log" directory).
# Use log or log-append to override this default.
# "log" will truncate the log file on OpenVPN startup,
# while "log-append" will append to it.  Use one
# or the other (but not both).
;log         openvpn.log
;log-append  openvpn.log

# Set the appropriate level of log
# file verbosity.
#
# 0 is silent, except for fatal errors
# 4 is reasonable for general usage
# 5 and 6 can help to debug connection problems
# 9 is extremely verbose
verb 3

# Silence repeating messages.  At most 20
# sequential messages of the same message
# category will be output to the log.
;mute 20
```

Server.crt

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=ES, ST=Barcelona, L=Barcelona, O=upc,
OU=MyOrganizationalUnit,
CN=Openvpnserver/name=upckey/emailAddress=kcxarokkc@gmail.com
        Validity
            Not Before: Dec 18 14:10:24 2015 GMT
            Not After : Dec 15 14:10:24 2025 GMT
        Subject: C=ES, ST=Barcelona, L=Barcelona, O=upc,
OU=MyOrganizationalUnit,
CN=Server/name=Openvpnserver/emailAddress=me@myhost.mydomain
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:e4:14:a9:e8:a4:61:80:d0:21:73:82:8d:49:bd:
                    6d:b6:e0:18:11:e6:93:8f:c7:4f:87:22:78:e0:17:
                    50:84:49:22:93:6d:a6:2d:53:07:4a:6a:6c:78:c7:
                    a2:51:7b:da:1e:1e:d4:90:30:25:a3:37:92:84:d3:
                    b8:f7:42:82:73:3a:7a:ef:71:e0:0b:67:da:fc:50:
                    4f:27:76:c9:8d:c1:00:83:c2:c3:e1:4c:23:94:6d:
                    08:59:d7:08:07:d3:b3:76:e5:26:9b:25:87:f8:3d:
                    02:2c:07:a2:6e:c7:67:ee:25:61:b9:9e:ae:0a:f8:
                    9b:32:c2:ca:d2:d0:b9:c2:4b:a0:d8:ba:68:7e:d2:
                    a9:85:a2:0d:7f:36:2f:37:69:bf:c4:91:87:3f:91:
                    9a:b0:93:02:83:d2:b0:2a:76:b7:20:c0:3e:a0:52:
                    00:91:61:3a:89:17:11:d3:ec:20:b4:6f:01:f4:67:
                    78:8a:1a:1c:ba:2f:4b:a5:9e:19:35:51:8c:c0:09:
                    20:31:de:d8:e4:08:4f:f0:eb:e7:70:85:21:9f:e8:
                    c4:58:b8:33:10:a6:f6:05:14:52:40:1b:50:2e:bc:
                    f7:be:0c:73:45:9c:16:08:61:31:2a:6f:e1:dd:47:
                    68:43:fd:b6:87:ef:af:7d:3c:dc:8e:75:57:98:0b:
                    7d:0f
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Cert Type:
                SSL Server
            Netscape Comment:
                Easy-RSA Generated Server Certificate
            X509v3 Subject Key Identifier:
```

```
            23:61:F6:4D:FA:E6:C4:5C:78:C1:36:97:59:FF:86:2E:9A:DB:1C:06
        X509v3 Authority Key Identifier:

keyid:F4:BE:1E:AE:C6:FA:40:72:19:5D:D2:C8:98:03:68:80:D9:F9:47:9E

DirName:/C=ES/ST=Barcelona/L=Barcelona/O=upc/OU=MyOrganizationalUnit/
CN=Openvpnserver/name=upckey/emailAddress=kcxarokkc@gmail.com
            serial:9C:99:02:7A:BF:41:9E:85

        X509v3 Extended Key Usage:
            TLS Web Server Authentication
        X509v3 Key Usage:
            Digital Signature, Key Encipherment
    Signature Algorithm: sha256WithRSAEncryption
        30:24:95:51:5b:9c:76:82:40:a2:06:44:35:fe:da:44:be:ad:
        9a:af:cd:00:7c:12:54:9a:a9:97:86:e4:9b:45:3f:52:b6:51:
        d7:a0:6b:fc:b2:21:2f:91:e3:2b:94:da:8d:33:c0:d9:8a:6b:
        99:21:5c:ee:4c:a8:3f:ab:19:69:98:38:78:d5:a7:7b:3e:3f:
        1d:08:57:3d:11:43:32:46:87:7d:88:7e:c7:6d:0a:94:95:59:
        2c:32:a7:42:73:68:f9:6b:d7:ea:d5:af:47:6f:ac:98:3b:5e:
        d5:94:5a:61:64:a2:e6:31:1a:85:a5:99:55:68:6f:90:1d:8e:
        af:2f:89:1f:83:23:8c:81:11:5b:ac:15:2c:00:85:eb:a6:93:
        99:f0:b5:2b:c9:cb:b2:d7:0b:cd:00:76:7b:f3:88:61:13:75:
        9a:0e:9c:98:9e:d5:be:6c:c8:b8:ba:92:a3:65:6b:b5:ae:db:
        4d:1d:a4:3e:bc:2f:60:30:19:f7:cb:8f:f0:d1:04:69:0d:2d:
        fb:0d:e3:ab:66:8f:9b:98:d5:a3:9e:5b:14:34:4d:d1:32:b9:
        ff:f3:66:57:68:3d:8d:7b:c3:f0:9a:99:c0:b2:f5:59:ea:ca:
        73:c7:e2:ac:9b:c6:8b:2f:8e:3a:da:fa:bb:91:5f:fe:24:6c:
        ea:2b:b2:8a
-----BEGIN CERTIFICATE-----
MIIFXDCCBESgAwIBAgIBATANBgkqhkiG9w0BAQsFADCBrzELMAkGA1UEBh
MCRVMx
EjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb25hMQw
wCgYDVQQK
EwN1cGMxHTAbBgNVBAsTFE15T3JnYW5pemF0aW9uYWxVbml0MRYwFAY
DVQQDEw1P
cGVudnBuc2VydmVyMQ8wDQYDVQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w0BCQ
EWE2tj
eGFyb2trY0BnbWFpbC5jb20wHhcNMTUxMjE4MTQxMDI0WhcNMjUxMjE1MT
QxMDI0
WjCBrjELMAkGA1UEBhMCRVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBA
GA1UEBxMJ
QmFyY2Vsb25hMQwwCgYDVQQKEwN1cGMxHTAbBgNVBAsTFE15T3JnYW
5pemF0aW9u
YWxVbml0MQ8wDQYDVQQDEwZTZXJ2ZXIxFjAUBgNVBCkTDU9wZW52cG5
zZXJ2ZXIx
ITAfBgkqhkiG9w0BCQEWEm1lQG15aG9zdC5teWRvbWFpbjCCASIwDQYJKo
ZIhvcN
AQEBBQADggEPADCCAQoCggEBAOQUqeikYYDQIXOCjUm9bbbgGBHmk4/
HT4cieOAX
```

UIRJIpNtpi1TB0pqbHjHolF72h4e1JAwJaM3koTTuPdCgnM6eu9x4Atn2vxQTyd2
yY3BAIPCw+FMI5RtCFnXCAfTs3blJpslh/g9AiwHom7HZ+4lYbmergr4mzLCytLQ
ucJLoNi6aH7SqYWiDX82Lzdpv8SRhz+RmrCTAoPSsCp2tyDAPqBSAJFhOokXEdPs
ILRvAfRneIoaHLovS6WeGTVRjMAJIDHe2OQIT/Dr53CFIZ/oxFi4MxCm9gUUUkAb
UC68974Mc0WcFghhMSpv4d1HaEP9tofvr3083I51V5gLfQ8CAwEAAaOCAYAwggF8
MAkGA1UdEwQCMAAwEQYJYIZIAYb4QgEBBAQDAgZAMDQGCWCGSAGG+EIBDQQnFiVF
YXN5LVJTQSBHZW5lcmF0ZWQgU2VydmVyIENlcnRpZmljYXRlMB0GA1UdDgQWBBQj
YfZN+ubEXHjBNpdZ/4YumtscBjCB5AYDVR0jBIHcMIHZgBT0vh6uxvpAchld0siY
A2iA2flHnqGBtaSBsjCBrzELMAkGA1UEBhMCRVMxEjAQBgNVBAgTCUJhcmNlbG9u
YTESMBAGA1UEBxMJQmFyY2Vsb25hMQwwCgYDVQQKEwN1cGMxHTAbBgNVBAsTFE15
T3JnYW5pemF0aW9uYWxVbml0MRYwFAYDVQQDEw1PcGVudnBuc2VydmVyMQ8wDQYD
VQQpEwZ1cGNrZXkxIjAgBgkqhkiG9w0BCQEWE2tjeGFyb2trY0BnbWFpbC5jb22C
CQCcmQJ6v0GehTATBgNVHSUEDDAKBggrBgEFBQcDATALBgNVHQ8EBAMCBaAwDQYJ
KoZIhvcNAQELBQADggEBADAklVFbnHaCQKIGRDX+2kS+rZqvzQB8ElSaqZeG5JtF
P1K2Udega/yyIS+R4yuU2o0zwNmKa5khXO5MqD+rGWmYOHjVp3s+Px0IVz0RQzJG
h32IfsdtCpSVWSwyp0JzaPlr1+rVr0dvrJg7XtWUWmFkouYxGoWlmVVob5Adjq8v
iR+DI4yBEVusFSwAheumk5nwtSvJy7LXC80AdnvziGETdZoOnJie1b5syLi6kqNl
a7Wu200dpD68L2AwGffLj/DRBGkNLfsN46tmj5uY1aOeWxQ0TdEyuf/zZldoPY17
w/CamcCy9VnqynPH4qybxosvjjra+ruRX/4kbOorsoo=
-----END CERTIFICATE-----

## Server.key

-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA5BSp6KRhgNAhc4KNSb1ttuAYEeaTj8dPhyJ44BdQhEkik22m
LVMHSmpseMeiUXvaHh7UkDAlozeShNO490KCczp673HgC2fa/FBPJ3bJjcEAg8LD

4UwjlG0IWdcIB9OzduUmmyWH+D0CLAeibsdn7iVhuZ6uCvibMsLK0tC5wkug2Lpo
ftKphaINfzYvN2m/xJGHP5GasJMCg9KwKna3IMA+oFIAkWE6iRcR0+wgtG8B9Gd4
ihocui9LpZ4ZNVGMwAkgMd7Y5AhP8OvncIUhn+jEWLgzEKb2BRRSQBtQLrz3vgxz
RZwWCGExKm/h3UdoQ/22h++vfTzcjnVXmAt9DwIDAQABAoIBAG33gIxIZTJkJxy4
P0LkcDeSCEjpMRBLeo6fEkSJOSru0Brg7RRBSeyZLz90kVBUNK/9EWOLsznkm2Ir
aqzchmTkN02nS+xz4GNynRdb8IXGHmdoymSf/y+a+kHGqXqMmaSji1+WUt5fG+L6
s1hSEEwD/gAu/58OilWJEZ1fQZDpWNgpRF6/vTUptxs4AA05daUrYtIWLtZh2Nm8
tEGLSluMjWPWigmhgeuvKgtD2UL5IwLPS+AoBbmXTzOer1SqpBum4KTO7+LzQrJ8
ENIgIbf4q0vYBoELr0qy8JTQogBzN4x0gp6JxAh1lhp3oMvT3YcIxAZqk+K2kc/l
5SrclRECgYEA9C2nCM66IMzR6u+cGySrhvdO2LZG0jfOdMFIOrW159xpuZCJSCpM
6WVls9FGvXXC/UMvm6ARlhcLUAYVBk4eD359xPFodNhnKJMOuulhdGssV29YtKfQ
uNfNsp8Q9MKnqWyFy4Z6tb0ZfeYDPi/b5hDnYD/dYFVhBj3t2DdeOO0CgYEA7x9/
UUyKUNhBtBwkfr9dDu086dRSmhez84mvqYhEE1CpWEFy04ap7ZM6tfWzRRVVXKEg
Js6NirlEKCB301eCGkhG7SrZrjnp7UNEisQnfFXSECiGFnzIK+q+AEC7ng8ELcSm
2/ZRCtnUW9oyLtJf+hUXk30ukNWfO0qVv4FUOmsCgYEA7rKaMQ/2AA55vxLRJe+p
Bl1IZK17ehE38AbqVwo/cxOS/uX+bd98JEUNMMWKQ7eVEAPXzGikCHcqheYMM4e1
9yuOQzJDDMZnwdBQfAOdUzNZu3eL3b6XWbsMyGqbJsoLIKKIALfLyVG8U8jArtAW
RUeNCBgmjv+nw3RSyCUIJNkCgYA6wwm9h8qhrVMQGK5bO1l59pC2gYe++D4vhdFW
Hvdjq7nBx2uZUYlmqXyPBI8GVgTBG5NUFgQwZ/C4z6nClMmS/dn4JESUDQVQWex5
EVUGt+xELkEtO4LRUAFa/I1efSAwwxeiDDjEKt2FqOwWF3qgcRh3FRchXm2f22dL
RVPWewKBgQDSemSWbmy9YM8dFhvrkU/1oCAQGPLI82sHPvPABGDB6GovQhy/Wf7r
ISPeJoEhnGG7Y9LRlMiFeMrs+kPwJ1hOLPXW/dyGJDehan4+Q8HYZjiILkM0vGFN
6cAbo812Z5MaWorE+Mzi8qQvIQlWBs3wY2i3HhgvkkEtSJGFzeh0ow==
-----END RSA PRIVATE KEY-----