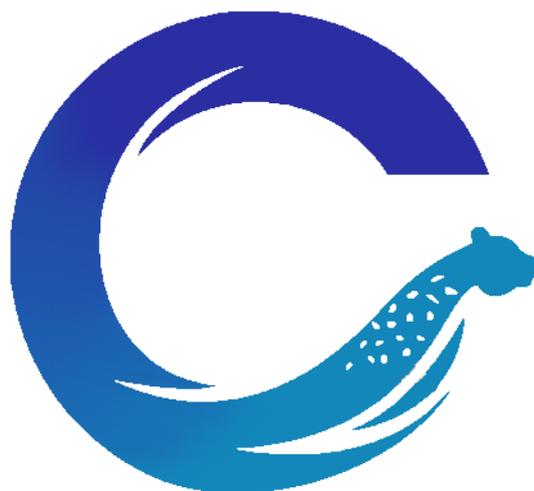




Institut Puig Castellar
Santa Coloma de Gramenet



Cheetah Cloud

Proyecto de Final de Ciclo

CFGS Administración de Sistemas Informáticos y Redes

Lidia Montero
Ivan Miralles
Juan Miguel Segura



Esta obra está sujeta a una licencia de [Reconocimiento-NoComercial-CompartirIgual 3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-sa/3.0/es/)

B) GNU Free Documentation License (GNU FDL)

Copyright © Juan Miguel Segura, Ivan Miralles Gomez, Lidia Montero gutierrez.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

C) Copyleft

© (l'autor/a)

Reservados todos los derechos. Está prohibida la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reproducción, el microfilm, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler o préstamo, sin la autorización escrita de los titulares o de los límites que autorice la Ley de Propiedad Intelectual.



Resumen del proyecto

El proyecto Cheetah Cloud se define como un panel web de hosting donde el cliente tendrá la oportunidad de contratar nuestros servicios y así crear su propio portal web. Los pilares básicos que forman Cheetah Cloud son Wordpress, Nextcloud y Collabora, todo ello creado sobre dos servidores, donde en uno de menor capacidad instalaremos Collabora Online, y en el otro servidor el resto del software sobre el que hemos creado la empresa mediante contenedores.

Por la contratación de un dominio público en la empresa se le ofrece al cliente un portal personal de almacenamiento en la nube sobre un Nextcloud con suite ofimática de Collabora Office.

El proyecto lo desarrollaremos usando software libre en su totalidad, y mediante virtualización con contenedores.

Palabras clave (entre 4 y 8)

Hosting

Nextcloud

Collabora Office

DNS

Dominio

Panel de hosting

Servidor



Project summary

Cheetah Cloud is an open source web hosting project, where users has the opportunity to hire our services and so create their own web portal. Most essential pillars in Cheetah Cloud are Wordpress, Nextcloud and Collabora softwares, created onto two servers where Collabora will be installed onto the less capacity one, and both the rest of the software onto the high capacity server, on which we have created the company through containers.

By contracting a public domain plan, we offer client a personal cloud storage portal on Nextcloud with office suite Collabora Office Online.

We will develop the project using free software in its whole, and through virtualization with containers.

Keywords (between 4 and 8)

Hosting
Nextcloud
Collabora Office
DNS
Dominio
Hosting panel
Server



Índice

1 Introducción	6
1.1 Contexto	6
1.2 Justificación	7
1.3 Objetivo general	7
1.3.1 Objetivos específicos	7
1.4 Estrategia y planificación del proyecto	8
1.5 Metodología de trabajo	8
1.6 Estudio económico y presupuestario	9
2 Descripción del proyecto	10
2.1 Análisis de requisitos	10
2.1.1 Requisitos funcionales	10
2.1.2 Requisitos no funcionales	10
2.2 Previsión de tareas de investigación	11
2.3 Tecnologías	11
2.3.1 Comparativa de las tecnologías valoradas	11
2.3.2 Tecnologías escogidas	12
2.4 Estructura del proyecto	13
2.5 Definición de las tareas	14
3 Desarrollo	15
3.1 Preparación del entorno	15
3.1.1 Instalación de Proxmox	15
3.1.2 Configuración de discos	16
3.1.3 Creación de contenedores	17
3.1.4 Configuración de un segundo servidor	19
3.1.5 Configuraciones estándar en los contenedores	19
3.2 Configuración del dominio	21
3.2.1 No-IP	21
3.2.2 Configurar el dominio	22
3.3 Implementación	22
3.3.1 Virtualmin	22
3.3.2 Servidores DNS	26
3.3.3 NextCloud	27
3.3.4 Collabora Office	28
3.3.4 Página web	29
4 Errores	30



5 Conclusions	30
5.1 Conclusions generals del projecte	30
5.2 Consecució dels objectius	31
5.3 Valoració de la metodologia i planificació	31
5.4 Visió de futur	31
7. Bibliografia	32



1 Introducción

Con la contratación de los servicios de Cheetah Cloud el cliente obtendrá la oportunidad de la creación de un dominio público propio y su personalización con el contenido libre que decida, junto con un espacio en la nube para usarlo como almacenamiento personal.

El panel de hosting de la empresa está configurado con Virtualmin, un conocido panel para gestionar portales web y alojamiento de dominios. El cliente tendrá acceso a un propio servicio Virtualmin mediante unas credenciales donde podrá administrar su portal web con las configuraciones deseadas. Además, dependiendo del plan contratado, el cliente tendrá la oportunidad de crear y gestionar cuentas de correo electrónico, así como redactar, enviar y recibir correos.

Wordpress es el programa gestor de contenidos para la creación de páginas web sobre el que está creada la interfaz web de Cheetah Cloud, donde se contratan los planes de la empresa y la vía pública donde darnos a conocer al mercado.

El espacio en la nube personal del cliente es un servicio Nextcloud, un programa donde almacenar ficheros y contenido en la nube. También permitimos al cliente tener un *file manager* a modo de suite donde crear documentos de texto, hojas de cálculo, presentaciones y carpetas, mediante el complemento Collabora.

El proyecto consta también de servidores DNS configurado mediante el software Bind para poder gestionar nuestros propios dominios y zonas DNS de forma personalizada.

Por último, cabe destacar que el proyecto lo desarrollaremos completamente usando software libre, ya que queremos que sea un sello de identidad diferencial a la competencia. Aprovechando así las ventajas de hacer uso de la virtualización de contenedores sobre uno de los servidores, para que los recursos del mismo se vean lo menos afectado posible y ofrecer una mayor experiencia al cliente final.

1.1 Contexto

La idea del proyecto nace debido a la falta de empresas de hosting que trabajan con software libre en su totalidad. Esto hace que los precios de los servicios que ofrecemos sean más asequibles en comparación al resto de mercado que el que podemos ofrecer nosotros y aun así sacando beneficios similares.

Desarrollando esta idea, posiblemente se evitan altos pagos mensualmente para servicios que realmente perjudican al bolsillo del usuario cuando con software libre, si tus necesidades no son muy caprichosas, puedes conseguir prácticamente lo mismo.



1.2 Justificación

La razón por la cual nos hemos decantado por este proyecto es para desarrollar un servicio de hosting de buena calidad con una cuota mensual baja que proporcione las mismas funcionalidades que la competencia.

Al tener un integrante del grupo trabajando en una empresa de hosting y explicar al resto de miembros el funcionamiento de trabajo de estas empresas, nos dimos cuenta de que podíamos crear la misma finalidad de la mano del software libre, dado que todos los componentes del grupo estamos a favor de la licencia GPL y apoyamos y apostamos siempre que podemos por el software Open Source.

1.3 Objetivo general

Nuestro objetivo principal es implementar una empresa de hosting el cual sea rápido, sencillo y potente. Todo esto utilizando software libre como se comenta anteriormente, para que el cliente final pueda disponer de su panel de hosting sin ningún tipo de coste adicional, tan solo pagando el mantenimiento del servidor en el caso de ser necesario.

1.3.1 Objetivos específicos

- Implementar y configurar un panel hosting donde el usuario final goce del control total de su dominio y pueda realizar las modificaciones deseadas, tanto en su alojamiento web como en las cuentas de correo
- Implementar y configurar una *suite* para que cada usuario tenga un espacio personal para alojar, crear y modificar ficheros y carpetas de forma intuitiva y con servicio en la nube. Queremos que ésta idea se lleve a cabo dado que queremos expandir nuestro conocimiento en la materia apostando por software sobre el que no tenemos experiencia previa, para así aprender conforme hacemos el proyecto mejorando nuestros conocimientos en software que conocemos y aumentando en conocimiento sobre otros servicios diferentes
- Conocer más sobre el software web de virtualización Proxmox VE, asignando, basándonos en los recursos del servidor físico, los necesarios en cada caso para que su rendimiento sea el más óptimo posible
- Saber mantener y configurar los diferentes servidores y contenedores necesarios para llevar a cabo el desarrollo de todo el proyecto, manteniendo la seguridad en los mismos con un óptimo rendimiento, ajustando cada uno de ellos a las necesidades y características del servidor
- Trabajar las oportunidades y dificultades a la vez que es trabajar con diferentes servicios y redes bajo una misma dirección IP, y aprender así más sobre el mantenimiento de las redes y las diferentes cosas que podemos hacer bajo recursos más reducidos en comparación con el gasto de dinero que precisa



una empresa

1.4 Estrategia y planificación del proyecto

La cotejación del desarrollo del proyecto se lleva a cabo entre diferentes estrategias, decidiendo finalmente la más adecuada en una relación basada en el tiempo ofrecido para realizarlo y la elasticidad que más se adapta al pensamiento general del grupo.

Así pues para implementar una empresa de hosting con todas las opciones que se ofrecen, podemos tomar las siguientes ideas para la elaboración:

- Comprar una licencia de un panel privativo y una suite, como por ejemplo cPanel y Google
- Desarrollar desde cero un panel y una suite a medida, personalizado y totalmente libre para que haga todas las funciones que queremos que tenga la empresa
- Implementar un panel y suite de software libre el cual solo haya que configurar con las necesidades básicas del proyecto

La segunda estrategia es la escogida, bajo una decisión unánime. Barajamos también la opción de hacerlo todo desde cero, junto con el panel de control del dominio, sin la necesidad de apoyarnos en software ya creado, pero resultó ser la opción más avariciosa y se acabó descartando de las posibles. Puede que con más tiempo hubiésemos tenido más responsabilidad y se hubiera podido plantear la opción de, por ejemplo, no toda la empresa, sino dedicarle más tiempo a un segmento del proyecto y hacerlo personalizado totalmente desde cero. Un ejemplo, el diseño de la página web.

Por último, la razón verdaderamente importante por la cual no hemos puesto tanto tiempo en el diseño y el desarrollo de aplicaciones es porque el ciclo que cursamos ya es de administración de sistemas, servicios y redes, por ello hemos querido centrarnos de forma definitiva en las áreas de implementación.

1.5 Metodología de trabajo

En el desarrollo del proyecto no vamos a implementar ninguna metodología de trabajo estándar, ya que al ser un proyecto para administradores de sistemas, todos los componentes del proyecto tocaremos todas las áreas de éste. Por este motivo hemos dividido el trabajo a partes iguales donde todos los integrantes del grupo haremos el trabajo tocando todos las mismas ramas de la informática de las que trata el proyecto.



1.6 Estudio económico y presupuestario

A continuación mostramos una tabla con los costes previstos aproximados que conlleva realizar el proyecto por completo, ya sea coste de hardware o de soporte de servicios.

Producto	Coste
Servidor (Intel Pentium, 120GB SSD, 300 HDD, 8GB RAM)	160€
Dominio .net (1 año)	10€ / año
Electricidad (3 meses)	7€ / mes = 21€
Red a internet (3 meses)	32€ / mes = 96€
Administrador de sistemas 1	1.400€ / mes = 4.200€
Administrador de sistemas 2	1.400€ / mes = 4.200€
Administrador de sistemas 3	1.400€ / mes = 4.200€
Coste total	12.887€

Como se puede observar es un proyecto el cual no requiere un gran gasto en dispositivos hardware, ya que la tarea más importante y grande es la de administrar el servidor para llevar a cabo toda la implementación.

Es clave mencionar que al utilizar software libre no tenemos ningún coste de licencia en todos los software que usamos e implementamos.



2 Descripción del proyecto

2.1 Análisis de requisitos

- Completar la inserción del Nextcloud en el panel de hosting
- Hacer de Nextcloud una *suite* con todo tipo de funcionalidades ofimáticas gracias a Collabora Office
- Configurar correctamente los servidores y zonas DNS para el almacenamiento de dominios, subdominios de la empresa y direcciones de correo
- Configurar el programa Proxmox VE como sistema de virtualización en los servidores
- Hacer funcionar correctamente los contenedores LXD con cada servicio correspondiente

2.1.1 Requisitos funcionales

- Servicio DNS
- Panel hosting
- NextCloud
- Collabora Office

2.1.2 Requisitos no funcionales

- Buen rendimiento en los servicios prestados
- Estabilidad en el funcionamiento de los servicios
- Seguridad en el servidor principal y contenedores
- Eficacia a la hora de cumplir los requisitos de los servicios prestados
- Una buena privacidad de la información de los clientes ficticios
- Optimización de los componentes hardware



2.2 Previsión de tareas de investigación

- Comprender el funcionamiento de Proxmox VE y sus capacidades
- Personalizar la interfaz y las funcionalidades Nextcloud para adaptarla a los servicios de éste que vamos a ofrecer al cliente
- Personalizar panel de hosting para poder acceder directamente a tu lugar en la nube de Nextcloud sin que el cliente deba hacer ningún esfuerzo
- Configuración del panel hosting Virtualmin

2.3 Tecnologías

2.3.1 Comparativa de las tecnologías valoradas

Cockpit vs Proxmox VE

En el grupo debatimos entre estos dos entornos de virtualización. Cockpit no tan enfocado a nuestro trabajo pero con las funcionalidades, y nos acabamos decantando por Proxmox VE ya que está más dedicado a ello y permite el trabajo con contenedores LXD. Esto último fue lo más determinante, ya que con contenedores el servidor consume menos recursos.

Cyber Panel vs Virtualmin vs IPSConfig

Virtualmin ofrece todas las funcionalidades que necesitamos, alternativa open source a Webmin, donde además ofrece una interfaz moderna y organizada.

Por otro lado las funcionalidades de Cyber Panel que necesitamos en la versión gratuita no las incluye, aunque como software gratuito es una buena opción.

Finalmente IPSConfig sí que ofrece las funcionalidades que precisamos pero su interfaz es muy anticuada.

Por estos motivos, hemos escogido Virtualmin como panel gestor de hosting que ofrece todas las funcionalidades que necesitamos con una interfaz moderna y visualmente bonita.

MV vs LXD

Hemos trabajado en nuestra mayoría de estudios en informática con virtualización sobre máquinas virtuales y tocando poca cantidad de software en contenedores.



Ambas opciones, están orientadas a la instalación y administración de sistemas, a diferencia de por ejemplo la herramienta *docker*.

Nos decantamos por contenedores LXD, porque el consumo de recursos es menor y la rapidez es mayor.

Fedora vs Ubuntu

En el grupo nos orientamos más a la idea de trabajar con Ubuntu, pese a que Fedora también es un sistema operativo fácil de manejar y con no demasiadas diferencias con Ubuntu.

Por un tema de simplemente gustos y preferencias, preferimos usar Debian en vez de Red Hat.

2.3.2 Tecnologías escogidas

Proxmox VE: es una herramienta libre, específica para lo que necesitábamos y te ofrece muchas más configuraciones y facilidades para las configuraciones de los contenedores

Virtualmin: ofrece todo lo que necesitamos en nuestro panel y es el más bonito e intuitivo a nivel de interfaz gráfica, entre las alternativas libres

LXD: permite tener un entorno de sistema con el menor consumo posible de los recursos del servidor físico

Nextcloud: es un sistema *cloud* libre, con la libertad de personalizar el panel del usuario y con variedad de plugins. Así nuestro proyecto junto con Collabora es lo más completo posible

Collabora CODE: herramienta ofimática por defecto en nuestra *suite* de Nextcloud. Alternativa open source a Collabora Online. Se implementa fácilmente con Nextcloud

Ubuntu: a lo largo de nuestra carrera estudiantil hemos utilizado con mayor frecuencia todo tipo de distribuciones basadas en Debian. Por ello se ha decidido utilizar como sistema operativo en los contenedores Ubuntu y por otro lado Debian en los servidores físicos. Esto es así porque Proxmox es sólo compatible con un Debian limpio, pero no nos supone ningún problema ni es un punto grave a destacar para el desarrollo

Let's Encrypt: el hecho de crear un proyecto con dominios y subdominios públicos, personalizados y privados, nos cuadraba más realizar certificados que no fuesen autofirmados y que los pudiésemos generar también de forma manual. En este caso de la mano de Certbot

Apache: escogemos Apache por el conocimiento medio que tenemos los integrantes del grupo en el servicio. Sabemos que existen mejores alternativas y más modernas



como Nginx, pero preferimos expandir conocimientos en otros software que nos interesan más abordar, a cambiar un programa del que vamos a tocar muy poca configuración.

MariaDB: en principio la opción por defecto de la pila LAMP es implementar MySQL como sistema gestor de base de datos, pero conocemos que en la versión 8 han cambiado muchas cosas y suelen haber problemas de compatibilidad y conflictos. No vamos a trastear mucho con las bases de datos, tan sólo explicaremos las funciones básicas para implementar una, pero por si acaso, preferimos instalar MariaDB, para quitarnos estas problemáticas de en medio y además, porque es en su totalidad software libre, a diferencia de MySQL.

2.4 Estructura del proyecto

El primer pilar de hardware y el más importante es el servidor físico, donde se encuentra instalado el sistema Debian 11. Allí se encuentra instalado **Proxmox VE**, nuestro entorno de virtualización.

En Proxmox VE hemos creado **dos contenedores LXD**, en el primero alojamos el panel de hosting Virtualmin y el servidor DNS primario, y en el segundo contenedor el servicio NextCloud para los clientes.

En segundo lugar tenemos un servidor con otros dos contenedores, donde en el primero se almacena un clúster DNS secundario (esclavo).

Y en tercer lugar, en el segundo contenedor del segundo servidor está almacenado el servicio Collabora Server CODE.

La estructura principal era tener un mismo servidor con tres contenedores, respectivos para cada dominio y subdominio, y en el servidor aparte tener el clúster DNS esclavo, pero debido a una problemática de redirección de puertos, debimos apartar Collabora de la estructura principal y dividirlo en cuatro contenedores en total.

A la hora de unir Nextcloud con Collabora, aunque especifiquemos el puerto de escucha del servicio, éste no funciona como es debido y tenemos problemas con los certificados, y por ende con el subdominio.

Nextcloud no encuentra ningún servidor con un CODE instalado en él porque al escribir el subdominio junto con el puerto, obvia el puerto y sólo lee el subdominio, pero la dirección IP es la misma que la del dominio oficial Cheetah Cloud.

Llegamos a la conclusión de que Collabora debía estar en un servidor aparte con una IP diferente a la del dominio oficial, y que así la implementación se hiciera efectiva, sin necesidad de especificar el puerto.



2.5 Definición de las tareas

	Hores Planificades
Projecte Integrat	191
Blocs de Fases prèvies al desenvolupament	19
Introduccion	8
Contexto	1
Justificacion	1
Objetivos	1
Alcance	1
Planificación	2
Presupuesto	2
Leyes y normativas	1
Analisis	11
Análisis de requisitos	1
Previsión de tareas de investigación	2
Tecnologías a usar	2
Definición de tareas	3
Definición de funcionalidades	3
Diseño	60
Estructura de los servidores	25
Diseño del entorno NextCloud	20
Diseño del Panel Hosting	15
Desarrollo	95
Configuración de contenedores LXD	50
Configurar Panel Hosting	35
Configurar NextCloud como una Suite	10
Pruebas	12
Functionals	6
Usabilidad	6
Lanzamiento	2
Web	2
Conclusiones	1
Biografía	2

[Enlace](#)



3 Desarrollo

3.1 Preparación del entorno

3.1.1 Instalación de Proxmox

Se implementa el sistema de virtualización [Proxmox](#) sobre el servidor de gran capacidad, para así crear los contenedores necesarios para desarrollar el proyecto de una forma más ágil y versátil.

Proxmox ofrece una gran variedad de opciones, pudiendo llegar a gestionar las máquinas desde la interfaz web, con también una terminal web, incluyendo gráficas que monitorizan sobre el consumo del procesador, memoria RAM y sobre la escritura y lectura de disco.

Una herramienta Open Source para facilitar la tareas de implementación de contenedores y/o máquinas virtuales a los administradores de sistemas.

Proxmox es un programa que trabaja sobre Debian, pudiendo realizar su instalación desde una imagen propia o sobre un sistema limpio ya instalado. En este paso se opta por instalarlo desde el repositorio oficial con Debian 11 Bullseye ya implementado y configurado adecuadamente.

Existe la posibilidad de seguir la [guía oficial](#) de instalación.

En primer lugar se configura una dirección IP estática asignada. Mediante la configuración del router se puede averiguar la última IP que ha proporcionado el servicio DHCP, y así escoger esa para asignarla al servidor. También se debe registrar la dirección junto al dominio en el archivo de configuración del servidor donde éstos se declaran para que la máquina pueda reconocerla.

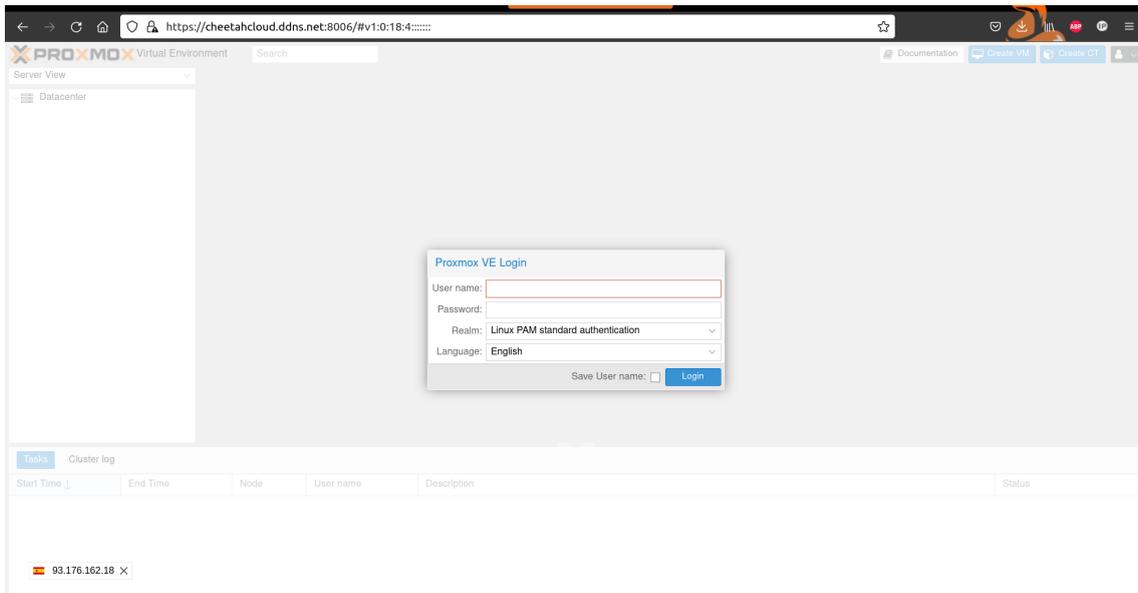
```
root@srvr:~# ip -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether e0:3f:49:52:f0:d4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.200/24 brd 192.168.1.255 scope global enp3s0
        valid_lft forever preferred_lft forever
    inet6 fe80::e23f:49ff:fe52:f0d4/64 scope link
        valid_lft forever preferred_lft forever
root@srvr:~#
```

```
root@srvr:~# hostname --ip
192.168.1.200
root@srvr:~#
```

Con los paquetes actualizados y el repositorio añadido en el sistema, se debe eliminar el paquete 'os-prober' para que no existan conflictos.



Como paso previo para la configuración gráfica, accedemos a ella mediante el navegador.



3.1.2 Configuración de discos

Se crea un almacenamiento llamado “local” una vez instalado Proxmox. En este almacenamiento por defecto se guardan las imágenes de los contenedores y las plantillas.

```
root@srvr:~# lsblk
NAME MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda   8:0    0 111,8G  0 disk
├─sda1 8:1    0 110,8G  0 part /
├─sda2 8:2    0     1K  0 part
└─sda5 8:5    0   975M  0 part [SWAP]
sdb   8:16   0 298,1G  0 disk
└─sdb1 8:17   0 298,1G  0 part /data
root@srvr:~#
```

El almacenamiento del servidor se compone de dos discos, donde *sda* contiene el sistema operativo y Proxmox, y es de 120 GB de espacio en formato SSD.

Por otro lado, *sdb* es el disco secundario del sistema y dispone de 300 GB en formato HDD, el cual permite recoger la información del sistema operativo en *sda* para poder crear los contenedores, y guardar los datos de éstos en el disco duro de mayor capacidad.

En el apartado Datacenter de la configuración se implementan estas opciones, de tal forma que los datos de los contenedores se almacenen en *sdb* (*/data*) y los datos del sistema operativo del que beben los contenedores en *sda* (*/local*)



Edit: Directory

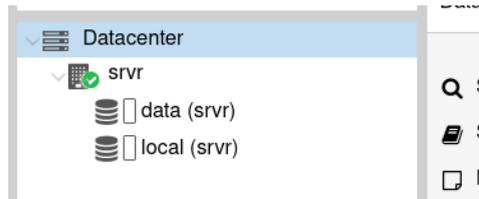
General Backup Retention

ID: data Nodes: srvr

Directory: /data Enable:

Content: Container Shared:

Help Advanced OK Reset



3.1.3 Creación de contenedores

Se descargan las plantillas del sistema operativo Ubuntu que precisan los contenedores para poder implementar los servicios correspondientes, y se almacena en */local*.

En la interfaz web, se crea un CT para configurar el dominio *cheetahcloud.net*



Create: LXC Container

General Template Disks CPU Memory Network DNS Confirm

Node: Resource Pool:

CT ID: Password:

Hostname: Confirm password:

Unprivileged container:

Nesting: SSH public key:

Create: LXC Container

General Template Disks CPU Memory Network DNS Confirm

Storage:

Template:

Create: LXC Container

General Template Disks CPU Memory Network DNS Confirm

Storage:

Disk size (GiB):

Create: LXC Container

General Template Disks CPU Memory Network DNS Confirm

Cores:

Create: LXC Container

General Template Disks CPU Memory Network DNS Confirm

Memory (MiB):

Swap (MiB):



The screenshot shows the 'Create: LXC Container' dialog box with the 'Network' tab selected. The configuration is as follows:

Field	Value	Field	Value
Name:	eth0	IPv4:	Static (selected), DHCP (unselected)
MAC address:	auto	IPv4/CIDR:	None
Bridge:	vmbr0	Gateway (IPv4):	
VLAN Tag:	no VLAN	IPv6:	Static (selected), DHCP (unselected), SLAAC (unselected)
Rate limit (MB/s):	unlimited	IPv6/CIDR:	None
Firewall:	<input type="checkbox"/>	Gateway (IPv6):	

Finalmente se repiten los pasos para los demás contenedores.

The screenshot shows the Proxmox VE interface for a container named 'srvr'. The 'Summary' page displays the following metrics:

Metric	Value
CPU usage	0.78% of 2 CPU(s)
Load average	1.09,0.43,0.22
RAM usage	15.10% (1.15 GiB of 7.65 GiB)
HD space	4.09% (4.44 GiB of 108.54 GiB)

3.1.4 Configuración de un segundo servidor

Llegados al punto de instalación y configuración de Collabora Office, es necesario configurar un VPS en una red distinta del servidor principal, ya que dicho software utiliza los puertos 80 y 443, los cuales están ocupados por el servidor de Virtualmin.

Por ello se ha instalado un nuevo servidor físico en una red distinta, para posteriormente importar el contenedor de Collabora. Así se podrán utilizar los puertos 80, 443 y 9980.

Además, también se utilizará dicha máquina para crear un nuevo contenedor donde estará el servidor DNS esclavo, el cual contendrá los registros del servidor DNS principal.

3.1.5 Configuraciones estándar en los contenedores

Se deben configurar direcciones IP internas diferentes para cada contenedor, apuntando todas a la misma dirección IP pública del servidor. Para consultar las direcciones disponibles es necesario hacer esta consulta al router de la red donde se encuentra el servidor principal:



DHCP

Enable

IPv4 Pool Start

IPv4 Pool End

IPv4 Lease Time in seconds

Restore Default DHCP Configuration

Add Reserved Address

MAC Address format is YY:YY:YY:YY:YY:YY (Y between 0 and 9 or between A and F)
IPv4 Address format is X.X.X.X (X between 0 and 255)

Enable	Device Name	MAC address	IPv4 Address	Options
<input checked="" type="checkbox"/>	Add MAC Address	E0:3F:49:52:F0:D4	192.168.1.200	✕
<input checked="" type="checkbox"/>	Add MAC Address	66:46:CC:BB:07:32	192.168.1.201	✕
<input checked="" type="checkbox"/>	Add MAC Address	AA:99:E1:1B:59:EE	192.168.1.203	✕
<input checked="" type="checkbox"/>	Add MAC Address	4E:CE:54:B1:52:1A	192.168.1.202	✕

Cancel

Apply

Al igual que con el servidor principal, es necesario configurar un nuevo dominio en No-IP, ya que el segundo servidor también se encuentra en una red doméstica donde la IP es dinámica. En este caso, el nombre escogido ha sido el siguiente:

A continuación se muestra un esquema de la red de contenedores virtualizados en ambos servidores.

Servidor principal (<https://cheetahcloud.net:8006>) - 192.168.1.200

- cheetahcloud.net -> 192.168.1.201
- nc.cheetahcloud.net -> 192.168.1.202

Servidor secundario (<https://coll.cheetahcloud.net:8006>) 192.168.1.201

- coll.cheetahcloud.net -> 192.168.1.206
- ns2.cheetahcloud.net -> 192.168.1.207



3.2 Configuración del dominio

El dominio público *cheetahcloud.net* se ha conseguido mediante contratar un plan de hosting en la empresa AFX Solutions, lo que permite que este proyecto se pueda llevar a cabo.

3.2.1 No-IP

El servidor está alojado en la red del hogar de un integrante del grupo, por lo tanto la IP pública no es estática sino dinámica. Por ello, vamos a utilizar el servicio de No-IP para configurar un subdominio junto a una IP dinámica.

Este servicio se encargará de cambiar la IP de forma automática e instantánea cada vez que nuestro router cambie de IP pública.

Dynamic DNS Client

Enable ON

Status Enabled

Provider No-IP

Username ivanmigo58

Password

Hostname cheetahcloud.ddns.net

Cancel Apply

Lamentablemente el servicio de No-IP no permite elegir tu propio dominio, sino que hay que escoger una sugerencia del mismo, dado que trabaja como un subdominio.

Nombre de host	Last Update	IP / Objetivo	Type
cheetahcloud.ddns.net Active	Mar 17, 2022 16:26 CET	93.176.162.18	A

La alternativa gratuita a este software, ya que no es en su totalidad gratis, fuerza a “regenerar” una vez al mes el nombre de dominio para que éste siga estando activo. Esto no supone un esfuerzo mayor, pero cabe comentar éste detalle para futuras lecturas.

Sin embargo, si a la larga en el tiempo supone algo tedioso, se puede aplicar un script, como este de GitHub <https://github.com/loblab/noip-renew>. Solo se necesita clonarlo en el servidor y seguir la configuración oficial que se muestra en la cuenta.



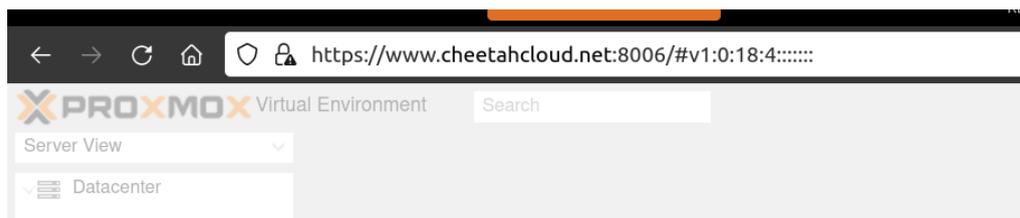
3.2.2 Configurar el dominio

Al adquirir el dominio *cheetahcloud.net* el proveedor ofrece gestionar las zonas DNS sobre el panel de gestión *cPanel*. Incluso ofrece, de forma gratuita, espacio para poder almacenar.

La idea principal es acceder siempre mediante el dominio y que resuelva obteniendo la IP de *cheetahcloud.ddns.net*. Todo esto configurado en la zona DNS.

www.cheetahcloud.net.	14400	CNAME	cheetahcloud.ddns.net	Editar	Eliminar
-----------------------	-------	-------	-----------------------	------------------------	--------------------------

```
root@srvr:~# host www.cheetahcloud.net
www.cheetahcloud.net is an alias for cheetahcloud.ddns.net.
cheetahcloud.ddns.net has address 93.176.162.18
root@srvr:~# host cheetahcloud.ddns.net
cheetahcloud.ddns.net has address 93.176.162.18
root@srvr:~#
```

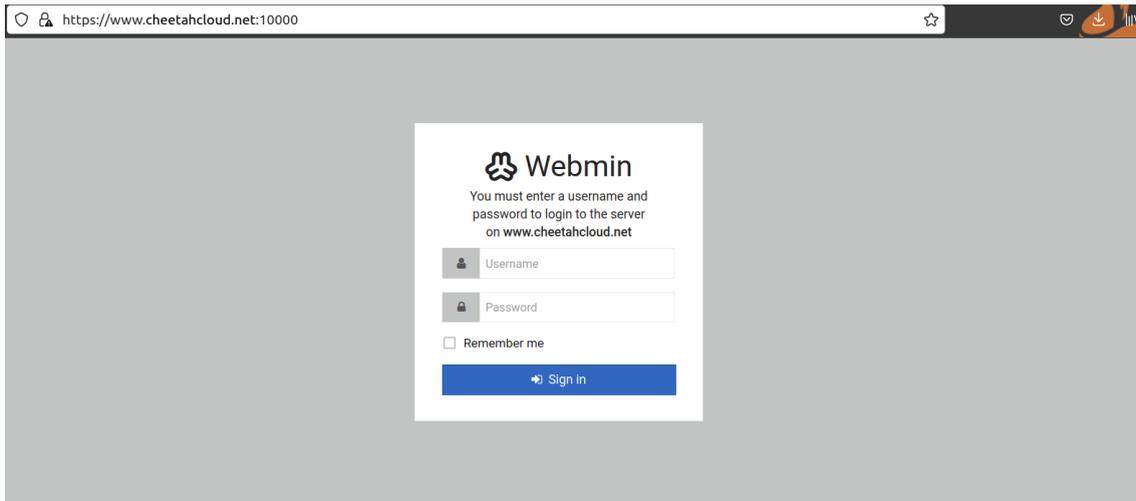


3.3 Implementación

3.3.1 Virtualmin

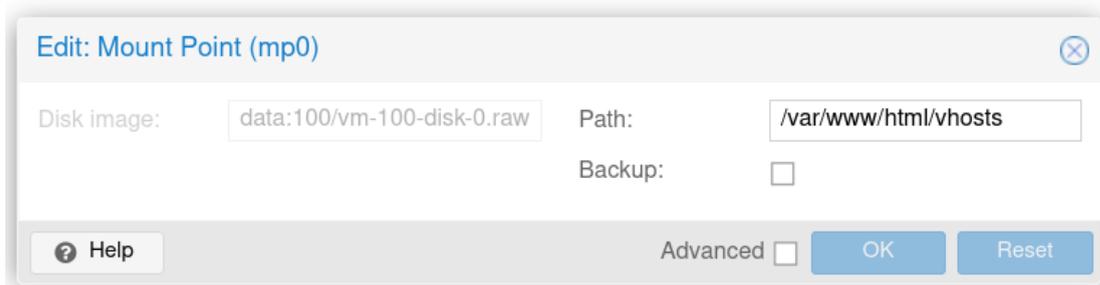
Para la instalación de Virtualmin se seguirá la guía oficial. Se empieza descargando y ejecutando el script de instalación

Y posteriormente se accede a la interfaz mediante el navegador, con la URL compuesta por el dominio y el puerto, y se accede mediante las credenciales descritas en la instalación.

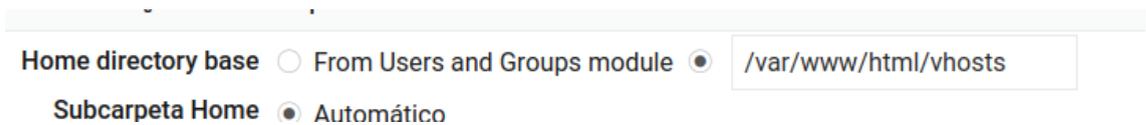


Una vez instalado, antes de empezar a configurar Virtualmin es imprescindible indicar que todo el contenido hosting esté alojado en el disco HDD y no en el SSD, donde está el sistema.

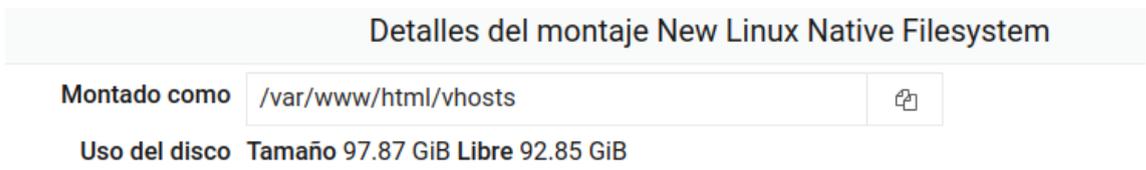
Para ello debemos de añadir el disco HDD en el contenedor y posteriormente montarlo donde deseemos.



Ahora configuramos que todos los usuarios creados se creen en esta ruta, desde el Virtualmin.



Más información.



Instalación de scripts

Además de la configuración principal, vamos a instalar diferentes scripts como WordPress y PhpMyadmin para poder acceder e instalar en tan solo un click. De esta forma, cualquier usuario podrá instalar de forma sencilla su el CMS



WordPress y aprovechar todas las funciones del mismo para crear o modificar su página web.

En referencia a PhpMyadmin, es muy útil para poder importar y exportar las bases de datos de cada alojamiento, además de realizar consultas SQL en su propia terminal.

Para poder instalar estos scripts, anteriormente debemos de configurar PHP en modo "Apache mod_php", de esta forma detectará todas las versiones de PHP de nuestro equipo.

Opciones de PHP para este dominio

Modo de ejecución de scripts de PHP

- Discapacitado
- Apache mod_php (ejecutar como usuario de Apache)
- FPM (ejecutar como propietario del servidor virtual)

Tiempo máximo de ejecución de script PHP

Ilimitado segundos

Información PHP

Versiones de PHP instaladas 7.4.3

Módulos para PHP 7.4	calendar	Core	ctype	curl	date	dom
exif	FFI	fileinfo	filter	ftp	gd	
gettext	hash	iconv	intl	json	libxml	
mbstring	mysqli	mysqlnd	openssl	pcntl	pcr	
PDO	pdo_mysql	Phar	posix	readline	Reflection	
session	shmop	SimpleXML	soap	sockets	sodium	
SPL	standard	sysmsg	syssem	sysvshm	tokenizer	
xml	xmlreader	xmlrpc	xmlwriter	xsl	zip	
zlib						

Módulos Pear para PHP 7 Archive_Tar Console_Getopt PEAR PEAR_Manpages Structures_Graph XML_Util

Guardar

Creación de planes

Algo muy importante en el proyecto, es crear y configurar los planes de hosting para poner limitaciones y demás configuración personalizada en cada uno de estos.

Los planes establecidos junto a sus características más relevantes son los siguientes

Nombre	Basic	Standard	Professional
Coste mensual	4.99€	9.99€	19.99€
Capacidad de disco HDD	15 GB	35 GB	65 GB
Cuentas de correo	5	10	Cuentas de correo ilimitadas
Base de datos	2	5	Base de datos ilimitadas

Como se puede observar, cada plan contiene unas limitaciones adecuadas al precio del mismo. Esto permite que cada cliente escoja el plan que más se adecue a sus necesidades.



Desde el panel de Virtualmin, podemos crear dichos planes, accediendo como usuario administrador del servidor, desde “Configuración del sistema” -> “Planes de cuenta”

Configuración de correo

Una de las principales funciones de crear una cuenta de hosting, es poder aprovechar y utilizar un correo con un dominio profesional, como es en nuestro caso.

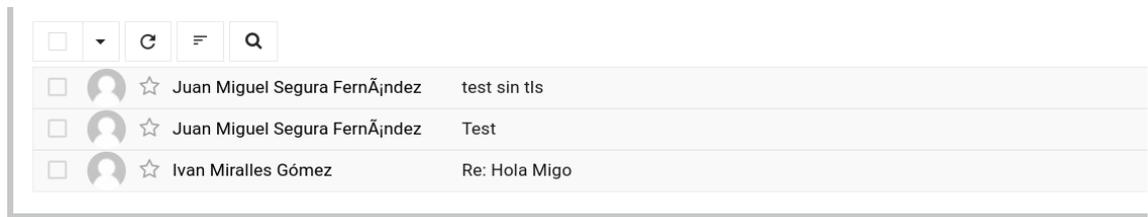
Por defecto, Webmin configura el servidor para utilizar como servicio de correo Postfix, no obstante, no nos ofrece una conexión cifrada en SSL y esto causa que algunos proveedores de correos como Gmail no puedan enviar mensajes.

Para solventar esto, debemos indicar el certificado a utilizar por Postfix, en nuestro caso utilizaremos el que ya ha generado Let 's Encrypt. Lo indicaremos de esta manera.

¿Habilitar el cifrado TLS? Si lo solicita el cliente

Archivo de certificado TLS	<input type="radio"/> Ninguno	<input checked="" type="radio"/> /etc/letsencrypt/live/cheetahcloud.net/cert.pem
Archivo de clave privada TLS	<input type="radio"/> Ninguno	<input checked="" type="radio"/> /etc/letsencrypt/live/cheetahcloud.net/privkey.pem
Archivo de autoridad de certificación TLS	<input type="radio"/> Ninguno	<input checked="" type="radio"/> /etc/letsencrypt/live/cheetahcloud.net/fullchain.pem

Si enviamos un correo a cualquier dirección @cheetahcloud.net, lo veremos entrando en el Usermin: cheetahcloud.net:20000.



Configuración de las copias de seguridad

En todo proyecto, es muy importante realizar copias de seguridad de todo el contenido de forma parcial y completa. Virtualmin, ofrece esta posibilidad utilizando como destino el propio servidor donde está alojado el proyecto, o bien otro diferente.

Para realizar una copia de seguridad automática, en virtualmin, debemos indicar la siguiente configuración.



▼ Servidores virtuales

Descripción de respaldo Backup

Servidores a guardar Todos los servidores virtuales Sólo los seleccionados ... Todos excepto ...

cheetahcloud.net

Incluir sub-servers de aquellos seleccionados

▼ Formato y destino

Destino de la Copia de Seguridad Archivo local

/backups/

Borrar backups antiguos Nunca Sí, después 7 días

Opciones adicionales de destino ¿Efectuar sustituciones de estilo de hora strftime en el archivo o nombre de directorio

Transferir cada servidor virtual luego del backup

Formato de Copia de Seguridad Archivo comprimido único Un archivo por servidor

Crear directorio de destino?

Acción en caso de error Detener la Copia de Seguridad inmediatamente Continuar con las otras prestaciones y servidores

Continuar, pero no eliminar backups antiguos

Formato de compresión de respaldo Valores por defecto tar+gzip tar+bzip2 tar zip

Nivel de backup Total (todos los archivos)

Incremental (solo los archivos modificados desde el último backup)

Ninguno (todos los archivos, y no actualizar incrementalmente)

Tal como hemos indicado anteriormente, los backups serán de la información que contienen todas las cuentas de hosting y se retendrán durante 7 días en un directorio del servidor local.

Además, por cada backup realizado se recibe un mensaje a la cuenta de correo indicada en la configuración del mismo, donde el contenido es parecido al siguiente. De esta manera, se puede asegurar que el backup se ha realizado correctamente.



Para restaurar una copia de seguridad, se debe iniciar sesión el Virtualmin y acceder a “Restaurar y backup” -> “Restaurar copia de seguridad”, para posteriormente restaurar el backup indicado en la configuración.

3.3.2 Servidores DNS

Dentro de nuestro servidor Virtualmin, tendremos un servidor DNS utilizando bind9, el cual resolverá todas las peticiones y será este el que pondremos como servidor DNS en los dominios que queramos configurar.

Además, crearemos un segundo servidor DNS esclavo, formando un cluster DNS, el cual estará alojado en una raspberry para mayor seguridad. Esto lo configuraremos con el dominio dinámico dns-cheetahcloud.ddns.net usando NO-IP.

Por último, como servidores DNS en todos nuestros dominios tendremos los siguientes:

- ns1-cheetahcloud.xarxes.site



- ns2-cheetahcloud.xarxes.site

Ambos serán un CNAME que apuntarán a los dominios dinámicos (cheetahcloud.ddns.net y dns-cheetahcloud.ddns.net) que apuntarán a los dos servidores DNS que tenemos en nuestra infraestructura. Cabe destacar que toda esta configuración debe ser así, debido a que todos los servidores que utilizamos los tenemos en una red doméstica con una IP dinámica, por lo tanto, no podemos apuntar los servidores a una única IP fija.

Podemos ver en una captura de pantalla un resumen de esto.

```
root@srvr:~# dig ns1-cheetahcloud.xarxes.site ns2-cheetahcloud.xarxes.site

; <<> DiG 9.16.27-Debian <<> ns1-cheetahcloud.xarxes.site ns2-cheetahcloud.xarxes.site
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 29033
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags;; udp: 512
;; QUESTION SECTION:
;ns1-cheetahcloud.xarxes.site. IN      A

;; ANSWER SECTION:
ns1-cheetahcloud.xarxes.site. 891 IN    CNAME   cheetahcloud.ddns.net.
cheetahcloud.ddns.net.      51  IN      A       212.230.156.98

;; Query time: 3 msec
;; SERVER: 212.230.135.2#53(212.230.135.2)
;; WHEN: Wed Apr 20 20:48:02 CEST 2022
;; MSG SIZE rcvd: 108

;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 44888
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags;; udp: 512
;; QUESTION SECTION:
;ns2-cheetahcloud.xarxes.site. IN      A

;; ANSWER SECTION:
ns2-cheetahcloud.xarxes.site. 900 IN    CNAME   dns-cheetahcloud.ddns.net.
dns-cheetahcloud.ddns.net.   60  IN      A       212.230.156.98

;; Query time: 55 msec
;; SERVER: 212.230.135.2#53(212.230.135.2)
;; WHEN: Wed Apr 20 20:48:02 CEST 2022
;; MSG SIZE rcvd: 112

root@srvr:~#
```

3.3.3 NextCloud

Se realiza la instalación del servidor Nextcloud mediante un contenedor y sobre el subdominio *nc.cheetahcloud.net*. Se empieza instalando una pila LAMP en el contenedor y todas las dependencias que el programa necesita. En el programa gestor de base de datos MariaDB instalado, se crea una base de datos para almacenar Nextcloud, un usuario con contraseña con rol *root*, un usuario normal con los privilegios necesarios para ejecutar la base de datos Nextcloud, y el mismo usuario con permisos para poderse conectar a la base de datos mediante fuera de la red (no localhost).



Se descarga el programa Nextcloud mediante la página web oficial del mismo en un archivo comprimido de la última versión. Una vez descomprimido se mueve la carpeta al directorio por defecto de Apache y se le da permisos con el usuario *www-data* para que Apache pueda visualizar Nextcloud en la máquina.

El siguiente paso es generar un fichero de configuración virtualhost en el directorio de Apache con la configuración del dominio de Nextcloud en modo HTTP.

Seguidamente se habilita el virtualhost y los mods necesarios para el correcto funcionamiento del sistema.

El paso importante de esta práctica es generar con éxito el certificado SSL para conseguir que nuestro dominio sea un lugar seguro en Internet, que se activa con el programa Certbot, creando un certificado firmado de forma manual.

Habiendo realizado todos estos pasos con éxito mediante una terminal de comandos, los últimos pasos son configurar Nextcloud desde la interfaz web: escribiendo el nombre de nuestro dominio en un navegador web, y accediendo mediante las credenciales creadas en MariaDB, junto con el nombre de la base de datos y la contraseña de acceso.

Se posponen los pasos finales a la completa instalación y configuración de Nextcloud para pasar al siguiente apartado de esta documentación, donde al acabar se retomará este último punto.

3.3.4 Collabora Office

Collabora Office es un software privativo con alternativa Open Source, así que hay que tener claro que el instalado en este punto es CODE junto al software COOLWSD, y no Collabora Office y LOOL, ya que pueden existir problemas de compatibilidad y de funcionalidades a medias en el proceso de instalación.

Se instalará el servicio CODE en un servidor diferente de donde tenemos todas las demás instalaciones hechas por una problemática que explicaremos más adelante. Por ahora existe una independencia de puertos, de direcciones IP y de dominios, lo ideal para continuar.

Con los repositorios actualizados, se importa la clave y se añade en el fichero de instalación de paquetería de la máquina lo necesario para instalar mediante comandos y paquetes el software COOLWSD, junto con una pila LAMP donde se declara un archivo VirtualHost para poder hacer las comprobaciones desde el navegador web y testear así que los certificados que se implementan estén en correcto funcionamiento, pese a que Collabora CODE no funcione mediante interfaz gráfica y no se haga uso de ella en este proyecto.

El funcionamiento de la configuración de Collabora es mediante un archivo XML, donde se registran todos los parámetros de forma manual; si no se quiere hacer de forma manual, mediante el comando *coolconfig set* se asignan del mismo modo las opciones para que Collabora no funcione con HTTP y sí lo haga mediante SSL. Este



paso es importante dado que si existen los dos parámetros activos, o HTTP activo y SSL apagado, Collabora Server no funcionará como es debido.

Seguidamente se le dice al programa que el cliente WOPI donde se gestionará la suite ofimática será nuestro dominio de Nextcloud. Por último se configura una contraseña para el servicio y reiniciamos éste.

De la misma forma que con Nextcloud, se implementa un certificado de firma manual con Certbot. Los certificados generados se suelen almacenar de forma automática en un fichero SSL VirtualHost en el directorio de Apache y así asignarse de forma automática al dominio, pero con Collabora no funciona así: en el archivo XML de configuración existe un apartado para escribir las rutas de los certificados; estos serán la ruta y nombres que deberán adaptar los certificados generados por Certbot, que suelen almacenarse en el directorio de Let's Encrypt, el programa que usa Certbot para generarlos.

Se retoma ahora el último punto para ambos apartados y conectamos los dos servidores. En la interfaz web de Nextcloud se instalan dos plugins, el actual 'Nextcloud Office' y el correspondiente dependiendo de la versión de la máquina para 'Collabora Server CODE'. De esta forma, el plugin de Office donde se declara el dominio Collabora junto con el puerto de escucha del servicio, pueda tener en cuenta que el servicio está en su modalidad Open Source, y pueda generar documentos ofimáticos correctamente.

3.3.4 Página web

Una vez desarrollado el proyecto al completo, se ha decidido crear una página web para promocionar el mismo, utilizando un hosting alojado en el servidor.

Esta web se ha realizado a modo informativo, para que el cliente conozca Cheetah Cloud y pueda ver los planes que se ofrecen para contratar el hosting. Desde la propia web se puede realizar el pedido del hosting o contactar de forma directa con los administradores del servicio.

Además de aprovechar el propio servidor de Virtualmin para alojar dicha web, se ha utilizado WordPress como CMS, ya que es uno de los más populares en internet, de esta forma se demuestra la buena compatibilidad y el gran rendimiento que tiene dicho CMS con el hosting en el que está alojado.

En enlace de la web es el siguiente: <https://www.cheetahcloud.net/>



4 Errores

Segmento Nextcloud y Collabora

Para la generación de certificados seguros para los dominios de Collabora y Nextcloud tuvimos muchos problemas

- El comando de Certbot precisa de muchos parámetros y tenemos que averiguar qué sintaxis es la correcta
- Todas las ayudas que encontramos por Internet afirman rotundamente que se debe desactivar el SSL de Collabora y activar el HTTP, para que así no haya conflicto con Apache y así pueda coger el servicio el dominio seguro por Apache y no por Collabora; se acaba llegando a la conclusión de que es en caso contrario: Collabora con SSL debe estar activada para que el servicio funcione con su SSL específico, y con Apache generar y declarar únicamente el dominio web con HTTP.
- De forma correlativa, acabamos indicando que los certificados estén en el directorio del servicio COOLWSD en vez de en Apache, con la ruta de Let's Encrypt y con el VirtualHost SSL correspondiente.
- Los errores anteriores se pudieron solventar gracias a darnos cuenta de que existe un pequeño error en los DNS de la red del instituto; por algún motivo de red que desconocemos, no encontraban los dominios públicos y los certificados no se podían generar. Solventamos esta problemática mediante forzar en nuestras máquinas el uso de los servidores DNS de Google.
- La infraestructura de diferentes contenedores para cada dominio bajo un mismo servidor con una única IP no pudo salir adelante debido a que la redirección de puertos no se hizo efectiva para el punto de generación de certificados. Como alternativa generamos el servidor Collabora en un servidor aparte con otra IP pública.

5 Conclusions

5.1 Conclusions generals del projecte

En general, el proyecto ha sido encarrilado desde el principio y se ha trabajado adecuadamente para conseguir los objetivos propuestos.

La conclusión general es óptima, ya que el resultado ha sido exitoso en el tiempo adecuado trabajando conjuntamente mano a mano todos los miembros del equipo.

Mediante la realización de este proyecto, se confirma que es posible gestionar un hosting junto a una suite informática, con distintas utilidades, utilizando servidores de baja calidad y software libre en todo momento.



5.2 Consecució dels objectius

Todos los objetivos estipulados al principio de la memoria, se han conseguido mediante el trabajo constante y la dedicación al proyecto.

En resumen, se ha instalado un panel de hosting gratuito el cual permite a un usuario gestionar su página web, ficheros, correo, etc. además de insertar en él una suite informática creada mediante software libre.

A partir de aquí, también se ha implementado una personalización en el código de Virtualmin y Nextcloud, además de la creación de la página web para promocionar los distintos planes de hosting.

5.3 Valoració de la metodologia i planificació

Al igual que en los anteriores puntos, la planificación y metodología escogida ha sido la adecuada. Dividiendo el trabajo de la creación y gestión de servidores, realizando cada uno una tarea distinta, el proyecto ha sido realizado de forma correcta.

En resumen, Juanmi ha instalado y configurado el panel de hosting, Lidia el nextcloud y colabora office, además de modificar el código de estos para personalizarlo al máximo. Iván se ha encargado de estructurar la red y el servidor principal, instalando Proxmox para poder virtualizar los contenedores correctamente, junto con la creación y personalización de la página web.

Con esta metodología, el trabajo ha sido repartido a partes iguales y los tiempos de desarrollo e implementación muy similares en los tres casos.

5.4 Visió de futur

Todos los objetivos planificados para el proyecto se han desarrollado y llevado a cabo correctamente, no obstante siempre es posible ampliarlo añadiendo nuevos servidores en la infraestructura o mejorando los actuales.

Al ser un proyecto de hosting, prácticamente no existen limitaciones para el mismo, siempre y cuando se utilicen tecnologías Open Source, ya que esta es la principal ideología de los integrantes del grupo y se debe de mantener en todo momento.



7. Bibliografía

Para la implementación y configuración de proxmox, se han utilizado los siguientes enlaces externos, en este caso todos ellos provienen de fuentes oficiales.

- [Instalación](#)
- [Creación de contenedores](#)
- [Configuración de red](#)

En Virtualmin.

- [Instalación](#)
- [Configuración de SSL](#)
- [Configuración de copias de seguridad](#)
- [Creación de cuentas](#)
- [Gestión de scripts](#)
- [Creación de paquetes](#)

Creación y configuración del servidor esclavo DNS

Nextcloud y Collabora

- Instalación e implementación del servicio COOLWSD

<https://www.linuxbabe.com/ubuntu/install-nextcloud-ubuntu-20-04-apache-lamp-stack>

<https://pydio.com/en/docs/v8/install-collabora-online>

<https://sdk.collaboraonline.com/docs/installation/Configuration.html>

<https://sdk.collaboraonline.com/CO-SDK-manual.pdf>

https://sdk.collaboraonline.com/docs/available_integrations.html#nextcloud-integration

<https://collaboraonline.github.io/post/build-code/>

https://sdk.collaboraonline.com/docs/installation/Proxy_settings.html#reverse-proxy-with-apache-2-webserver

<https://sdk.collaboraonline.com/docs/installation/Configuration.html#ssl-configuration>