UNIVERISDAD NACIONAL DEL LITORAL

FACULTAD DE INGENIERIA Y CIENCIAS HÍDRICAS

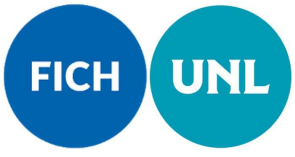
TRABAJO FINAL

MIGRACIÓN DE SOFTWARE PRIVATIVO A SOFTWARE LIBRE

baruja shen adonai elohim

Blanco Emanuel A.
Universidad Nacional Del Litoral.
Facultad de Ingeniería y Ciencias Hídricas.
Tecnicatura Universitaria en Software Libre.

julio, 2023



Índice general

Capítulo 1

Introducción

Este trabajo es el resultado de un largo periodo formativo, el cual concluyó en Diciembre del 2020. Lo desarrollado en esta página pretende cumplir con los requisitos del Trabajo Final de la Tecnicatura Universitaria en Software Libre.

El trabajo desarrollado tuvo lugar en la sala de informática perteneciente a la UNL, ubicada en el instituto de detención penal Nº 2 "Las Flores". El propósito de esta sala informática es posibilitar el acceso a personas privadas de su libertad a distintas trayectorias educativas/culturales/laborales, mediante el estudio a distancia a través de internet.

El siguiente texto describe la migración de Software Propietario a Software Libre realizada en el Aula Virtual de la UNL, en el penal de las flores.

Los objetivos propuestos y a concretar fueron los siguientes:

- Migrar los equipos informáticos de Software Privativos a Software Libre en todas sus dimensiones.
- Lograr el cambio de una filosofía a otra en la selección y organización de las tecnologías utilizadas en el aula.
- Posibilitar que los estudiantes modifiquen la visión que tienen respecto al Software Libre.

Problemas resueltos:

- Bajo rendimiento del hardware disponible.
- Incompatibilidad del software con el hardware disponible.
- Rendimiento deficitario de la red informática.
- Falta de autonomía tecnológica.

Para elaborar este trabajo fue necesario llevar adelante una investigación previa a la migración, la cual consistió en dos etapas:

- Investigar sobre otros procesos de migración o proyectos similares que hayan tenido éxito, indagar si podría tomarse como modelo alguno de ellos/continuarlo/mejorarlo.
- Diseñar una estrategia que permita alcanzar el objetivo planteado, junto con un cronograma especificando las acciones y tiempo estimado de inicio y finalización.

Todo lo realizado fue posible gracias al apoyo de la gran comunidad del Software Libre. Es el deseo de este estudiante activista que todo lo elaborado pueda servir para motivar y ayudar a otras personas en el proceso de migración de cualquier otra institución con similares características.

Movimiento Software Libre

En la década del '70', cuando la computación estaba en sus inicios era común que tanto los desarrolladores de software profesionales, como los aficionados, publicaran sus trabajos para que otros puedan utilizarlo, corregir errores y mejorarlo. A partir del avance de las industrias tecnologías sobre el software en la década del '80', las practicas colaborativas se vieron afectadas, y en consecuencia muchos desarrolladores de software decidieron dejar de compartir sus trabajos y solamente dejar que otras personas lo utilicen bajo ciertas condiciones, manifiestas en lo que se denomina "licencias restrictivas". Estas licencias no permiten que se pueda compartir el programa sin el consentimiento del desarrollador y mucho menos la posibilidad de poder hacerle modificaciones para corregir errores o agregar mejoras.

Un caso ejemplar lo constituye el episodio protagonizado por la empresa Microsoft, la cual le envió una carta a un grupo de programadores aficionados que utilizaban copias no autorizadas de su programa BASIC. En esta carta, Bill Gates, "General Partner", acusa a esos programadores de que le están robando su programa, argumentando que compartir el software es injusto, ya que su creador no recibe suficiente dinero a cambio. Esta forma de pensar atentaba contra el espíritu de cooperación, solidaridad y reciprocidad que existía en ese entonces en los grupos informáticos.

Para contrarrestar esta tendencia a no compartir el código fuente, surgió el movimiento Software Libre.

El Software Libre es un movimiento ético, político y social, que tiene por objetivo defender la libertad de las personas en un mundo donde las computadoras afectan cada vez más nuestra forma de vivir. Se lo considera como un movimiento político y social, dado que no solo implica defender las "cuatros libertades esenciales", sino que también, el no permitir que el capitalismo tome el poder del conocimiento. Es por ello, que centra su lucha en una mirada política sobre el conocimiento en general y las tecnologías en particular, ya que se cuestiona el concepto de "propiedad privada del conocimiento" y busca promover la libertad de los "usuarios de computadoras", para contribuir en la lucha por los derechos de los ciudadanos en el entorno digital. (traficante de sueños)

Los artefactos diseñados bajo la filosofía del Software Libre posibilita Satisfacer las necesidades tecnológicas de las comunidades y los individuos,ya que al poder modificarse se puede adaptar a las necesidades existentes.

Ademas, al poder ser redistribuido libremente, (sea la versión original o una con modificaciones) se aporta al desarrollo de la sociedad. De esta manera al compartir el programa y las ideas, se logra generar más conocimiento y el involucra miento de las personas en las decisiones sobre el desarrollo tecnológico de sus comunidades.

Capítulo 2

Motivación

Este trabajo fue motivado al reconocer las distintas problemáticas existentes en el aula informática de la UNL.

El bajo rendimiento de los ordenadores por el uso de software privativo, la degradación del rendimiento causada por virus o spyware, la falta de fondos para adquirir licencias de software, son algunos de los los problemas que no permitían un uso eficiente de los equipos informáticos, para que cubra las necesidades de los alumnos en ese lugar.

El interés por realizar este trabajo se centro en buscar alternativas viables para superar las limitaciones y mejorar las herramientas de trabajo en el marco de un proyecto que garantiza el acceso a la educación de una población vulnerable.

Capítulo 3

Contexto

El Programa “Educación Universitaria en Prisiones”. El cual consiste en alinea con aquellos intentos de transformar la herramienta educativa en un vehículo no ya de “corrección”, ni de “moralización”, sino de resistencia frente a la degradación cotidiana que el encierro supone. Se trata siempre de intentar construir espacios de libertad, gobernados por una lógica sustancialmente distinta de aquella que rige el penal.

El Programa comenzó a funcionar en el año 2004, a partir de la firma de un convenio entre la Universidad Nacional del Litoral y el entonces Ministerio de Gobierno, Justicia y Culto. En función de este convenio, se disponía la instalación de aulas virtuales en las Unidades Penitenciarias N º I de la Ciudad de Coronda y N º II “Las Flores” de la ciudad de Santa Fe.

Estas aulas se integrarían a la Red de Campus Virtuales a través de los cuales opera el Centro Multimedial de Educación a Distancia de la UNL.

Propuesta y actividades desarrolladas en las aulas virtuales

La oferta educativa del Programa está compuesta por carreras de pre-grado, denominadas Tecnicaturas, que brindan formación técnica vinculada con demandas del mercado laboral, tienen una duración que oscila entre 5 y 6 cuatrimestres y otorgan título universitario de validez nacional.

Capítulo 4

Importancia

La importancia de este trabajo radica en la mejora sustancial de las prestaciones del equipamiento informático del aula de la universidad como condición necesaria para garantizar el acceso a la educación superior de la población de la Unidad Penal N°2, “Las Flores”.

Capítulo 5

Destinatarios

Los destinatarios beneficiados por la concreción de este trabajo son alrededor de 30 internos, que se encuentran cursando diferente carreras de la oferta a distancia de la Universidad Nacional del Litoral.

Capítulo 6

Conceptos técnicos aplicados

Conocimientos técnicos aplicados en este trabajo:

- Administración de Sistemas Operativos GNU/Linux: Incluye la personalizacion, instalación y configuración de servicios.
- Administración de redes de datos: configuración de direcciones IP.
- Administración básica de Docker: Creación y Configuración de contenedores.

Capítulo 7

Licencias involucradas

7.1. Tipos de licencia

7.1.1. Licencia GPL

Una de las licencias más utilizadas es la Licencia Pública General de GNU (GNU GPL). El autor conserva los derechos de autor y permite la redistribución y modificación del software bajo ciertos términos diseñados para garantizar que todas las versiones modificadas del software permanezcan bajo la misma licencia GNU GPL. Esto significa que no es posible crear un producto que contenga partes no licenciadas bajo GPL. La licencia GNU GPL permite la modificación y redistribución del software, pero solo bajo la condición de utilizar la misma licencia.

7.1.2. Licencias LGPL

La Licencia Pública General Reducida de GNU, más conocida por su nombre en inglés GNU Lesser General Public License (LGPL), es una licencia creada por la Free Software Foundation (FSF) que garantiza la libertad de compartir y modificar el software cubierto por ella, asegurando que el software sea libre para todos sus usuarios. Esta licencia se aplica a cualquier programa o trabajo que incluya una nota del propietario de los derechos del trabajo, estableciendo que puede ser distribuido bajo los términos de la LGPL.

7.1.3. Licencia AGPL

La Licencia Pública General de Affero (AGPL) es una licencia copyleft derivada de la Licencia Pública General de GNU, diseñada específicamente para asegurar la cooperación con la comunidad en el caso de software que se ejecuta en servidores de red. Se encuentra dentro de las licencias que derivan de GNU y están destinadas a modificar los derechos de autor. La Affero GPL es en esencia una GNU GPL, pero con una cláusula adicional que impone la obligación de distribuir el software si se utiliza para ofrecer servicios a través de una red de ordenadores.

La novedad de la AGPL es que, además de incluir las cláusulas propias de una GNU GPL, también exige la distribución del software cuando se utilice como parte del desarrollo de un nuevo software destinado a ofrecer servicios a través de una red de ordenadores. Esto significa que si alguien utiliza software AGPL como componente de un nuevo proyecto, dicho proyecto estará obligado a ser distribuido libremente.

7.1.4. Licencia Estilo BSD

Se les llama así porque se utilizan en gran cantidad de software distribuido junto a los sistemas operativos BSD. Es una licencia permisiva que impone pocas condiciones sobre lo que un usuario puede hacer con el software. Bajo estas licencias, el autor mantiene la protección de copyright únicamente para renunciar a la garantía y para exigir la atribución adecuada de la autoría en trabajos derivados, pero permite la libre redistribución y modificación, incluso si esos trabajos tienen un propietario. Son muy permisivas,

tanto que se pueden mezclar fácilmente con la licencia GNU GPL, con la cual son compatibles. Además, BSD permite el cobro por la distribución de objetos binarios. Sin embargo, algunas opiniones señalan que este tipo de licencia no contribuye al desarrollo de más software libre. A menudo se utiliza la siguiente analogía: Üna licencia BSD es más libre que una GPL si y solo si se opina también que un país que permite la esclavitud es más libre que otro que no la permite.

7.1.5. Licencia PSFL

La Licencia de la Python Software Foundation (PSFL), anteriormente conocida como Python License, es una licencia permisiva de software libre, similar a la licencia BSD. Cumple con los requisitos de la Open Source Initiative (OSI) para ser considerada una licencia de software libre y también es compatible con la licencia GPL. A diferencia de la GPL y al igual que la mayoría de las licencias BSD, la PSFL no es una licencia copyleft, lo que significa que permite modificaciones del código fuente y la creación de trabajos derivados sin requerir que dichas modificaciones o derivados sean de código abierto. La licencia PSFL está incluida en las listas de licencias aprobadas tanto por la Free Software Foundation como por la Open Source Initiative.

7.1.6. Licencia MPL y derivadas

Esta licencia es de software libre y tiene un gran valor, ya que fue el instrumento utilizado por Netscape Communications Corp. para liberar su Netscape Communicator 4.0 y comenzar el proyecto tan importante para el mundo del software libre: Mozilla. Se utiliza en una gran cantidad de productos de software libre que se utilizan a diario en diversos sistemas operativos. La Licencia Pública de Mozilla (MPL) es considerada software libre y promueve eficazmente la colaboración, evitando el efecto "viral"de la GPL. Sin embargo, la MPL no es tan permisiva como las licencias tipo BSD. Estas licencias se conocen como çopyleft débil". La NPL (posteriormente la MPL) fue la primera licencia nueva en muchos años que abordó algunos puntos que no fueron considerados por las licencias BSD y GNU. En el espectro de las licencias de software libre, se puede considerar que la MPL es adyacente a la licencia estilo BSD, pero mejorada.

7.1.7. Licencia CDDL

La Licencia Común de Desarrollo y Distribución (CDDL), también conocida como Sun Public License (SPL) versión 2, es una licencia de código abierto y libre producida por Sun Microsystems. Está basada en la Licencia Pública de Mozilla (MPL), versión 1.1. La CDDL fue enviada para su aprobación al Open Source Initiative (OSI) el 1 de diciembre de 2004 y fue aprobada como una licencia de código abierto a mediados de enero de 2005. Según el primer borrador realizado por el comité de divulgación de licencias del OSI, la CDDL es una de las nueve licencias más populares, ampliamente utilizada a nivel mundial y con comunidades sólidas.

7.1.8. Licencia EPL

La Licencia Pública de Eclipse (EPL) es una licencia utilizada por la Fundación Eclipse para su software. Reemplaza a la Licencia Pública Común (CPL) y elimina ciertas condiciones relacionadas con litigios de patentes. La Licencia Pública de Eclipse está diseñada para ser una licencia de software favorable para los negocios y cuenta con disposiciones más flexibles que las licencias copyleft contemporáneas. Los destinatarios de programas con licencia EPL pueden utilizar, modificar, copiar y distribuir el trabajo y las versiones modificadas. En algunos casos, también pueden estar obligados a liberar sus propios cambios.

7.1.9. Licencia Apache

La Licencia Apache es una licencia de software libre creada por la Apache Software Foundation (ASF). La licencia tiene varias versiones, incluyendo la 1.0, 1.1 y 2.0. La licencia requiere que se conserve el aviso de copyright y el descargo de responsabilidad, pero no es una licencia copyleft, lo que significa que no exige la redistribución del código fuente cuando se distribuyen versiones modificadas. Tampoco requiere que las versiones modificadas se distribuyan como software libre o de código abierto. Sin embargo, la licencia Apache sí exige que se mantenga una notificación que informe a los receptores que se ha utilizado código con Licencia Apache en la distribución.

7.1.10. Licencia PHP

La licencia de PHP es la licencia bajo la cual se publica el lenguaje de programación PHP. Según la Free Software Foundation, es una licencia de software libre no copyleft y, según la Open Source Initiative, es una licencia de código abierto. Sin embargo, debido a restricciones en el uso del término ”PHP”, no es compatible con la licencia GPL.

Las continuas mejoras y avances en el lenguaje se deben a una gran comunidad de desarrolladores que contribuyen sin obtener beneficios comerciales. Estas contribuciones incluyen:

- Código fuente.
- Soporte a otros usuarios a través de listas de correo.
- Revisión del programa en busca de errores.
- Notificación de fallas de seguridad, entre otros.

Sobre esta base, se sostiene una licencia que garantiza la libertad del lenguaje y no permite que alguien obtenga beneficios comerciales exclusivos de PHP y se convierta en dueño del lenguaje. Este es el espíritu de la licencia.

Cuando se desarrolla una aplicación y se vende a terceros, el monto cobrado no es por el lenguaje de programación en sí, sino por la solución a un problema, el tiempo invertido en el desarrollo, el soporte u otros aspectos específicos.

7.1.11. Licencias Creative Commons

Las licencias de Creative Commons permiten a los usuarios utilizar obras protegidas por derechos de autor sin necesidad de solicitar permiso al autor de la obra. Estas licencias se crearon inicialmente en base a la legislación estadounidense y posteriormente se adaptaron rápidamente a las legislaciones de diferentes países alrededor del mundo.

Tipos de licencias de Creative Commons

Todas las licencias Creative Commons otorgan ciertos derechos básicos, como el derecho a reproducir y distribuir la obra de forma gratuita.

- **Atribución (BY):** El beneficiario de la licencia tiene el derecho de copiar, distribuir, exhibir y representar la obra, así como crear obras derivadas, siempre y cuando se reconozca y cite la obra de acuerdo a las especificaciones del autor o licenciante.
- **No Comercial (NC):** El beneficiario de la licencia tiene el derecho de copiar, distribuir, exhibir y representar la obra, así como crear obras derivadas, pero únicamente para fines no comerciales.
- **No Derivadas (ND):** El beneficiario de la licencia solo tiene el derecho de copiar, distribuir, exhibir y representar copias literales de la obra, sin poder crear obras derivadas.
- **Compartir Igual (SA):** El beneficiario de la licencia tiene el derecho de distribuir obras derivadas bajo una licencia idéntica a la que rige la obra original.

1

Capítulo 8

Documentación

Capítulo 9

Estado del arte

Tras realizar una exhaustiva investigación sobre trabajos relacionados con la migración a software libre, se tomó como referencia central Tesis de ”Migración a Software Libre”llevado a cabo por la Universidad Técnica de Manabí, específicamente el trabajo titulado Ïnstalación y Configuración de Equipos Informáticos bajo software Libre”.

El objetivo principal de este documento es ofrecer a la biblioteca de su facultad una alternativa al software propietario y lograr una migración exitosa de los entornos de escritorio hacia el software libre. Se trata de una guía de buenas prácticas que proporciona una visión clara de los pasos, procesos y elementos necesarios para llevar a cabo la migración de los equipos de la biblioteca, utilizando exclusivamente software libre.

Si bien este trabajo aborda parte de los aspectos planteados, una desventaja significativa es que no ha sido actualizado. Por lo tanto, se ha decidido realizar un nuevo trabajo que detalla la migración de una red de computadoras y los aplicativos de software que se deben utilizar. El objetivo es que este nuevo trabajo sirva como referencia para futuras migraciones en establecimientos con características similares.

La selección del software para esta migración se ha basado en el uso de licencias de Software Libre y las ventajas prácticas que ofrece en comparación con sus contrapartes propietarias. Se han tenido en cuenta tanto la disponibilidad de las licencias como las ventajas específicas que brinda cada aplicación de software libre seleccionada, en comparación con sus equivalentes privativos.

Capítulo 10

Objetivos

10.1. Generales

Como objetivo general, se pretende lograr la migración completa a software libre de una red de computadoras y optimizar el rendimiento de toda la red informática. Además, se busca que este trabajo pueda ser replicado en diferentes instituciones con características similares, brindando una referencia para futuras migraciones.

10.2. Específicos

Objetivos específicos:

- Mejorar la seguridad, funcionalidad y productividad de toda la red informática.
- Reducir los costos en la adquisición de hardware y licencias.
- Reutilizar el hardware desechado por no cumplir con los requisitos exigidos por los diferentes software privativos.
- Facilitar que la migración a software libre sea lo más transparente y fácil de asimilar para cualquier persona que intente implementar este trabajo.

Capítulo 11

Desarrollo

En el presente trabajo, se contempla la migración de la red informática del Aula Virtual, que consta de nueve estaciones de trabajo. Una de ellas se destinará para funcionar como un servidor de virtualización, en el cual se ejecutarán los servicios que se ofrecerán a los estudiantes del espacio tanto en el presente como en el futuro.

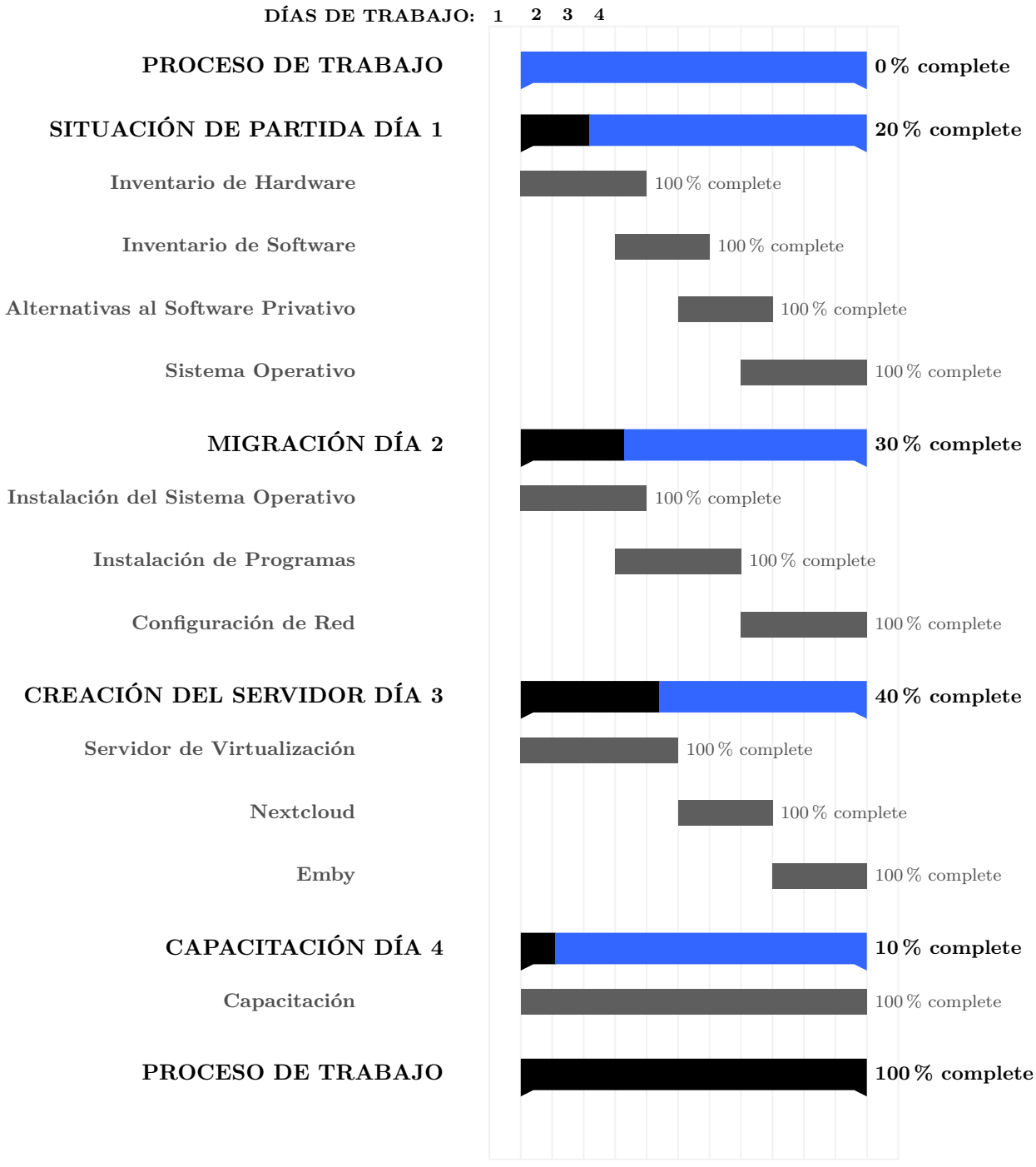
Para las demás estaciones de trabajo, se prevé la instalación de un Sistema Operativo GNU/Linux orientado a usuarios finales. En cada uno de estos equipos, se configurarán las aplicaciones necesarias para el trabajo diario, incluyendo un navegador de internet, cliente de correo electrónico, suite de oficina, cliente FTP y TexStudio.

Desde el inicio del proyecto, se ha tenido en cuenta la diversidad de formatos de archivos con el objetivo de facilitar la migración y evitar los desafíos comunes asociados con la transición hacia el software libre. Se ha seleccionado cuidadosamente aplicaciones de software libre que tienen la capacidad de soportar formatos privativos y permiten importarlos hacia formatos libres. Sin embargo, es importante señalar que al exportar documentos al formato libre, existe la posibilidad de que se pierdan algunas características específicas.

A pesar de esta consideración, la adopción del servidor de virtualización y el uso del software libre en las estaciones de trabajo promoverán una mayor autonomía y flexibilidad dentro del espacio educativo. Esto permitirá a los estudiantes trabajar con herramientas abiertas y transparentes, fomentando la colaboración y la compatibilidad con diferentes sistemas y formatos. Asimismo, la migración hacia el software libre contribuirá a fortalecer la seguridad y estabilidad de la infraestructura informática del Aula Virtual, beneficiando a toda la comunidad educativa.

Capítulo 12

Cronograma de trabajo



Capítulo 13

Proceso de migración

En este punto, se llevará a cabo un análisis exhaustivo de la situación de partida, con el propósito de obtener un conocimiento detallado de las arquitecturas de los hardware disponibles, la variedad de documentos, aplicaciones y tipos de archivos existentes, junto con otros aspectos relevantes. Este análisis tiene como objetivo evitar ajustes imprevistos durante el proceso de migración y establecer un plan de acción con la debida anticipación.

El propósito primordial de este análisis consiste en identificar los requisitos funcionales que el nuevo sistema operativo debe cumplir. Mediante este enfoque, se busca garantizar que el sistema operativo seleccionado sea capaz de satisfacer las necesidades y demandas específicas del entorno, asegurando así una transición exitosa y efectiva hacia el nuevo entorno informático.

13.1. Aspectos importantes

En el análisis de la situación de partida, se identificaron varios aspectos importantes que requieren una atención detallada para la migración exitosa. Entre ellos se encuentran:

- **Documentos y sus formatos:** Se ha recopilado información sobre los diferentes tipos de documentos utilizados en el entorno actual, así como sus formatos específicos. Este conocimiento será fundamental para asegurar que el nuevo sistema operativo pueda manejar y ser compatible con estos formatos, evitando la pérdida de información o la necesidad de conversiones complicadas.
- **Archivos de audio/videos y sus formatos:** Se ha evaluado la variedad de archivos de audio y video presentes en el sistema. Es esencial verificar que el nuevo sistema operativo cuente con las herramientas y códecs necesarios para reproducir y trabajar con estos archivos de manera efectiva.
- **Aplicaciones y sus interfaces:** Se ha analizado el conjunto de aplicaciones utilizadas actualmente y sus interfaces. Es crucial determinar si existen alternativas de software libre que puedan reemplazar estas aplicaciones y proporcionar una experiencia de usuario similar o mejor.
- **Bases de datos y estructura de datos:** Se ha estudiado la estructura y el contenido de las bases de datos utilizadas en el sistema actual. Este análisis permitirá planificar la migración de datos hacia el nuevo entorno y asegurarse de que la nueva plataforma sea compatible con las bases de datos existentes.
- **Disponibilidad de datos y aplicaciones:** Se ha verificado la disponibilidad de datos y aplicaciones críticas para el funcionamiento del sistema. Es fundamental asegurar que todos los datos y aplicaciones necesarios estarán disponibles en el nuevo entorno para garantizar la continuidad de las operaciones.
- **Hardware disponible y drivers necesarios:** Se ha realizado un inventario del hardware existente en el entorno actual. Es importante identificar si el nuevo sistema operativo es compatible con el hardware existente y si se requerirán controladores (drivers) adicionales para su correcto funcionamiento.

Considerando estos aspectos, se podrá establecer un plan de acción detallado para la migración hacia el nuevo sistema operativo, asegurando que se cubran todas las necesidades y se minimicen los inconvenientes durante el proceso de transición. La atención cuidadosa a estos aspectos clave permitirá lograr una migración exitosa y una adaptación fluida al nuevo entorno basado en software libre.

13.2. Inventario de Software

El objetivo del inventario de software es elaborar un listado completo de todos los programas, aplicaciones, servicios y configuraciones utilizados en los equipos que requieren migración. Este proceso tiene como propósito registrar de manera detallada todo el software presente en los sistemas, lo cual facilitará la planificación y ejecución de la migración de manera ordenada y estructurada.

El inventario de software contendrá información esencial, como el nombre de cada programa, su versión, el tipo de aplicación o servicio que representa y las configuraciones específicas realizadas en cada caso. A través de esta recopilación exhaustiva de datos, se obtendrá una visión clara y completa de los componentes del software existente, lo que resultará fundamental para llevar a cabo una migración efectiva y sin contratiempos.

Al contar con este inventario detallado, el equipo encargado de la migración podrá tomar decisiones informadas y estratégicas en cada etapa del proceso, asegurando la compatibilidad y disponibilidad de todas las aplicaciones y servicios críticos en el nuevo entorno. Asimismo, esta herramienta proporcionará una base sólida para realizar pruebas y verificar la funcionalidad de cada componente después de la migración, minimizando los riesgos y permitiendo una adaptación fluida al nuevo sistema basado en software libre.

13.2.1. Software Invetariado

- Windows 10 64 bits
- Navegador Chrome
- Microsoft Outlook,
- Adobe After Effect
- Adobe Acrobat
- Adobe Photoshop
- atube gatcher
- uTorrent
- Microsoft Office 2016
- Panda Cloud Antivirus
- Windows Media
- WinRAR
- CCleaner

13.3. Inventario de Hardware

El objetivo del inventario de hardware es obtener un conocimiento detallado de las características y especificaciones del hardware de los ordenadores que se planean migrar. Este proceso permite determinar si los equipos contarán con soporte nativo por parte de las distribuciones de software libre y también identificar la necesidad de actualizar algún componente o si existen posibles incidencias relacionadas con el soporte de hardware.

En el inventario de hardware se registran datos como el modelo y fabricante de cada equipo, la cantidad de memoria RAM, el tipo y capacidad del disco duro, la tarjeta gráfica, la tarjeta de red, entre

otros componentes relevantes. Además, se investiga la disponibilidad de los controladores necesarios para el correcto funcionamiento del hardware en las distribuciones de software libre consideradas para la migración.

Con este inventario detallado, se anticipan y abordan posibles desafíos relacionados con el hardware durante el proceso de migración, asegurando así una transición exitosa hacia el software libre.

13.3.1. Hardware Inventariado

Desktop

Cuatro equipos con procesador Intel:

Hardware	Modelo
Mother:	Asrock H55M-LE
Procesador:	Intel Core i3 3.0GHz.
Disco Duro	500GB.
Memoria RAM	DDR3 2GB
Arquitectura	64bits.

Tres equipos con procesador AMD:

Hardware	Modelo
Mother:	MSI
Procesador:	Amd athlon2 x2 3.00GHz.
Disco Duro	500GB.
Memoria RAM	DDR3 2GB
Arquitectura	64bits.

Servidor

Hardware	Modelo
Mother:	FM2
Procesador:	Amd 3.5ghz.
Disco Duro	500GB.
Memoria RAM	DDR3 8GB
Arquitectura	64bits.

13.4. Sistema Operativo

13.4.1. Distribución GNU/Linux

Las distribuciones GNU/Linux representan sistemas operativos basados en el núcleo Linux, que contienen una selección de paquetes de software. Estas distribuciones están mayormente compuestas por software libre, aunque en algunos casos también pueden incluir software propietario.

Adicionalmente al núcleo Linux, las distribuciones GNU/Linux típicamente integran las bibliotecas y herramientas del proyecto GNU, así como el sistema de ventanas X Window System. En función de su enfoque y el público objetivo al que van dirigidas, estas distribuciones pueden incluir una diversidad de aplicaciones y herramientas, como procesadores de texto, hojas de cálculo, reproductores multimedia y utilidades administrativas.

Es relevante enfatizar que cuando una distribución GNU/Linux incorpora paquetes del proyecto GNU, se le conoce como distribución GNU/Linux, en reconocimiento a la contribución del software libre desarrollado por el proyecto GNU dentro de la distribución.

13.4.2. Debian

Se seleccionó Debian como sistema operativo para el servidor de virtualización debido a sus características y ventajas. Debian es una distribución mantenida por la comunidad que ofrece una excelente estabilidad

y facilita las actualizaciones de paquetes y del propio sistema de manera sencilla.

Debian es conocido por su amplio soporte en diversas arquitecturas y dispositivos, además de ofrecer un soporte a largo plazo (LTS).

Entre las características destacadas de Debian se encuentran las siguientes:

- Es una distribución compuesta únicamente por software libre, y se compromete a mantener esa condición en todas sus versiones.
- Ofrece una gran estabilidad y seguridad, al estar basado en el núcleo Linux. Proporciona una configuración predeterminada para cada paquete y ofrece actualizaciones de seguridad de forma regular durante todo su ciclo de vida.
- Cuenta con un amplio soporte de hardware, siendo compatible con la mayoría de los dispositivos compatibles con el núcleo Linux. En caso de ser necesario, Debian cuenta con controladores adicionales que facilitan la compatibilidad con otros dispositivos.

13.5. Instalación del sistema operativo

Antes de iniciar la instalación del sistema operativo en los ordenadores, se llevará a cabo la tarea de respaldar toda la información de los alumnos en un disco externo. Esta acción es considerada esencial para preservar los datos de manera segura y evitar pérdidas.

Se presentan varias opciones para realizar el respaldo, pero siempre se recomienda utilizar un medio físico de almacenamiento adicional. Esto puede incluir discos duros externos, unidades USB, discos ópticos u otros dispositivos de respaldo adecuados. El objetivo es garantizar que los datos estén protegidos y disponibles para su posterior recuperación en caso de ser necesario.

13.5.1. Requisitos

Los requisitos mínimos necesarios para la instalación de **Debían Cinnamon** son los siguientes:

- 1 GB de RAM (se recomiendan 2 GB para un uso cómodo).
- 15 GB de espacio en disco (se recomiendan 20 GB).
- Resolución de 1024×768.
- Procesador 2Ghz, doble núcleo.

13.5.2. Proceso de Instalación

El proceso de instalación de Debian con el entorno de escritorio Cinnamon se lleva a cabo siguiendo los siguientes pasos:

- **Preparación del medio de instalación:** En primer lugar, se obtiene la imagen ISO de Debian con el entorno Cinnamon desde el sitio web oficial de Debian. Luego, se graba la imagen en un DVD o se crea un dispositivo USB de arranque.
- **Arranque del sistema desde el medio de instalación:** Una vez que el medio de instalación está listo, se inicia el ordenador desde este medio. En la mayoría de los casos, esto implica configurar la secuencia de arranque en la BIOS o UEFI para que el equipo inicie desde el DVD o USB.
- **Selección del idioma y configuración:** Al iniciar el sistema desde el medio de instalación, se presentará un menú de bienvenida donde se puede seleccionar el idioma preferido. A continuación, se eligen las configuraciones regionales adecuadas, como la zona horaria y el teclado.
- **Configuración de la red:** Si se necesita acceso a internet durante la instalación (por ejemplo, para descargar paquetes adicionales), se configura la conexión a la red.
- Particionado del disco: Luego, se procede a la configuración del particionado del disco duro. Aquí, se puede elegir entre diferentes opciones, como usar todo el disco para Debian o realizar un particionado personalizado. Es importante tener cuidado, ya que esta etapa puede borrar datos existentes si no se realiza con precaución.

- Selección de paquetes de software: En este paso, se elige el entorno de escritorio Cinnamon junto con otros paquetes adicionales que se deseen instalar. Además, se puede seleccionar el tipo de instalación, como una instalación estándar o personalizada.
- **Configuración del gestor de arranque:** Se selecciona el gestor de arranque que se utilizará para iniciar Debian. Por lo general, GRUB (Grand Unified Bootloader) es el gestor de arranque predeterminado y adecuado para la mayoría de las configuraciones.
- **Creación del usuario y contraseña:** Se crea una cuenta de usuario con un nombre de usuario y una contraseña para acceder al sistema.
- **Finalización de la instalación:** Una vez que se han realizado todas las selecciones y configuraciones, se confirma la instalación y el proceso de instalación de Debian con Cinnamon comienza. El sistema copiará todos los archivos necesarios y configurará los paquetes seleccionados.
- **Reinicio del sistema:** Al finalizar la instalación, se reinicia el sistema y se cargará Debian con el entorno Cinnamon. A partir de ese momento, el usuario podrá acceder a su cuenta y comenzar a utilizar Debian con Cinnamon como entorno de escritorio.

13.6. Repositorios

13.6.1. ¿Que es un repositorio?

Un repositorio es un lugar centralizado y organizado donde se almacenan y mantienen los archivos y paquetes de software de una aplicación, programa o sistema operativo. En el contexto de sistemas operativos basados en Linux, como Debian, Ubuntu, CentOS, entre otros, los repositorios son fundamentales para la instalación y actualización de software de manera sencilla y segura.

En un repositorio, se encuentran diversos tipos de software, como aplicaciones, bibliotecas, controladores y actualizaciones del sistema. Cada paquete de software está acompañado por metadatos que proporcionan información sobre su versión, autor, dependencias y otras características relevantes.

Los repositorios pueden ser oficiales, proporcionados por los desarrolladores del sistema operativo, o de terceros, creados y mantenidos por la comunidad o empresas externas. La utilización de repositorios oficiales y de confianza es esencial para garantizar la seguridad y estabilidad del sistema, ya que se asegura de que el software provenga de fuentes confiables y se someta a controles de calidad antes de su inclusión.

13.6.2. Configuración de repositorios Debian

La lista que requiere edición se encuentra alojada en */etc/apt/sources.list*. Para obtener información acerca de los repositorios disponibles proporcionados por Debian, se puede consultar una Wiki Debian que cuenta con información detallada sobre cada uno de los repositorios.

13.7. Instalación de Programas

La mayoría de los programas necesarios para los estudiante se instalan de forma automática, mientras que otros deberán ser instalados manualmente a través de la terminal.

A continuación, se detallan los programas que deben ser instalados en los equipos:

- **Brave:** Brave es un navegador totalmente gratuito y de código abierto para ordenadores o teléfonos móviles, que destaca por su enfoque en la privacidad y velocidad.
- **Chromium:** Chromium es una versión de código abierto de Google Chrome, pero sin los códecs exclusivos y otros elementos que Google utiliza para diferenciar Chrome de otros navegadores.
- **Cliente de correo Thunderbird:** Thunderbird es el cliente de correo electrónico desarrollado por la Fundación Mozilla, responsable también del navegador Firefox. Thunderbird se ha diseñado para cubrir las necesidades de aquellos que buscan un gestor de correo electrónico ligero y gratuito.
- **Evince:** Evince es un visor de documentos para el entorno de escritorio GNOME que permite visualizar archivos en formato PDF y PostScript.

- **Gimp:** GIMP (GNU Image Manipulation Program) es un programa de edición de imágenes de software libre, que forma parte del proyecto GNU y se distribuye bajo licencia pública y GNU Lesser General Public License.
- **Inkscape:** Inkscape es un editor de gráficos vectoriales que permite diseñar imágenes de calidad, tanto básicas como complejas. Ofrece herramientas para crear y editar diagramas, líneas, gráficos, logotipos, cómics, folletos, entre otros elementos.
- **HandBrake:** HandBrake es un programa de software libre que permite editar archivos de audio y video.
- **qBittorrent:** qBittorrent es un cliente P2P de software libre que se utiliza para la transferencia de archivos grandes.
- **FileZilla:** FileZilla es un programa de software libre que funciona a nivel cliente/servidor, permitiendo conectarse a un servidor para consultar, adquirir y manipular contenido del mismo.
- **VLC:** VLC es un reproductor y framework de video y música de software libre, compatible con una amplia gama de formatos multimedia. Es capaz de reproducir la mayoría de los códecs sin necesidad de descargar paquetes adicionales.
- **BleachBit:** BleachBit es una herramienta de software libre que se encarga de eliminar elementos como caché, cookies, archivos temporales, historiales, registros de chats, miniaturas y accesos directos inválidos.
- **TeXstudio:** TeXstudio es un editor de LaTeX multiplataforma de software libre, que ofrece funciones como marcadores, autocompletado de comandos, coloreado de sintaxis, soporte de arrastrar imágenes y asistente para la creación de tablas y fórmulas, entre otras características destacadas.

Para instalar el navegador Brave es necesario que agregue el repositorio donde se encuentra el navegador

```
1
2   #herramientas necesarias para agregar el repositorio.
3   apt install apt-transport-https curl
4
5   #repositorio de Brave
6   curl -fsSLo /usr/share/keyrings/brave-browser-archive-keyring.gpg https://brave-browser-
    apt-release.s3.brave.com/brave-browser-archive-keyring.gpg
7
8   echo "deb [signed-by=/usr/share/keyrings/brave-browser-archive-keyring.gpg arch=amd64]
    https://brave-browser-apt-release.s3.brave.com/ stable main"| tee /etc/apt/sources.list.d/
    brave-browser-release.list
9
10  #actualizar la lista de paquetes disponibles.
11  apt update
12
13  #instalar pogramas.
14  apt install brave-browser texstudio bleatbit vlc filezilla qbittorrent handbrake inkscape
    gimp evince thunderbird chromium
15
16
17
```

Código 13.1: Instalación de programas

Capítulo 14

Configuración red LAN

14.1. Definición de una red LAN

Una red de área local (LAN) es un conjunto de dispositivos, como impresoras, sistemas informáticos, dispositivos móviles y consolas de juegos, que están interconectados mediante alguna tecnología y comparten datos entre sí. Una LAN puede funcionar como una entidad autónoma e independiente o formar parte de redes más grandes y extensas.

14.2. Significado de una dirección IP

Una dirección IP (Protocolo de Internet, correspondiente al nivel de red del modelo TCP/IP) es un número que identifica de manera lógica y jerárquica una interfaz de red de un dispositivo. En un paquete IP, el encabezado contiene una dirección de 32 bits (IPv4) o 128 bits (IPv6). Este identificador es único y no se repite en ningún otro equipo en el mundo, y tiene la función de registrar el equipo en la red global. Las direcciones IP no solo son asignadas a equipos de cómputo, sino también a módems, routers, sitios web, entre otros.

En el mundo del direccionamiento IP, se distinguen dos tipos de direcciones IP: dinámicas y estáticas.

- Las direcciones IP dinámicas son variables y son entregadas y administradas por un servidor DHCP. Su funcionamiento se basa en el arrendamiento de la dirección por un tiempo específico, tras el cual la dirección se renovará y puede cambiar su sintaxis.
- Las direcciones IP estáticas, como su nombre indica, son direcciones fijas que no cambian. Se utilizan en servidores, máquinas de producción conectadas a la red y, en general, por usuarios que no necesitan que su dirección IP varíe, ya que otros servicios dependen de ellas.

14.3. Configuración de red

Para mantener la simplicidad en la red del aula, se ha decidido asignar direcciones IP dinámicas a los ordenadores de escritorio. Sin embargo, el servidor de virtualización recibirá una dirección IP estática para que las estaciones de trabajo puedan localizarlo en la red.

La red del aula utilizará la dirección IP 192.168.1.0/24 y se ha establecido la dirección IP 192.168.1.222 para el servidor de virtualización.

14.3.1. Diagrama de red

El diagrama de red es una representación visual que muestra los ordenadores que formarán parte de la red de trabajo y su interconexión.

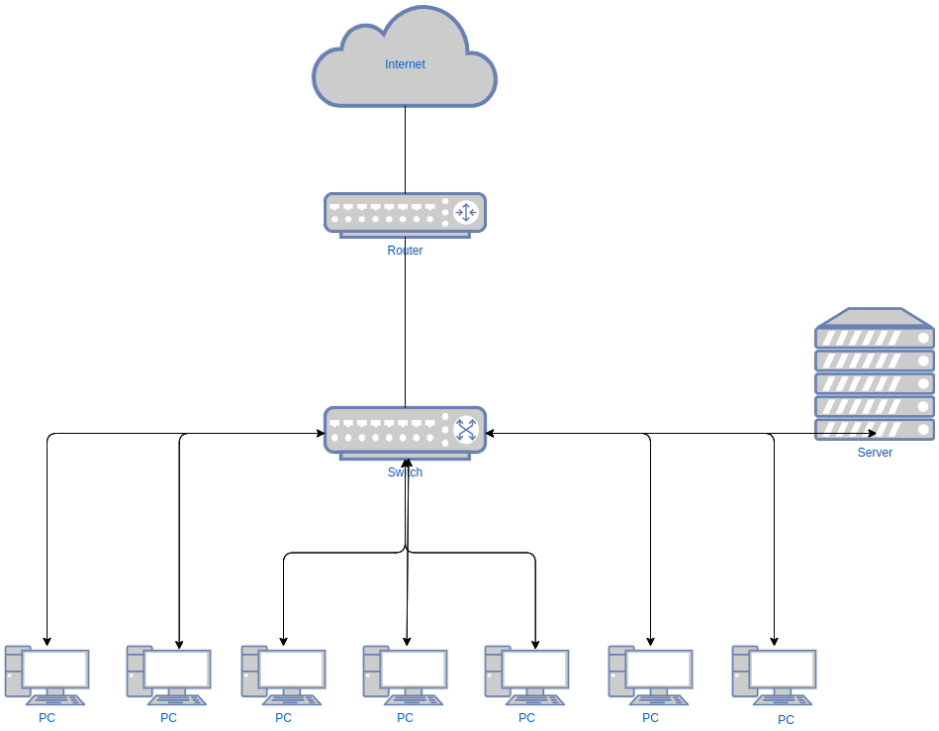


Figura 14.1: Diagrama de red

Capítulo 15

Servidor de Virtualizacion

15.1. ¿Que es un Servidor de Virtualizacion?

Un servidor de virtualización es un sistema de hardware o software que permite la creación y gestión de máquinas virtuales (VM, por sus siglas en inglés). Las máquinas virtuales son entornos de software aislados que se ejecutan dentro de un servidor físico, y cada una de ellas actúa como un sistema independiente con su propio sistema operativo y aplicaciones.

El servidor de virtualización proporciona una capa de abstracción entre el hardware físico y las máquinas virtuales, permitiendo que múltiples VMs se ejecuten en el mismo servidor físico de manera eficiente. Esto ayuda a maximizar la utilización de los recursos del servidor, ya que se pueden consolidar múltiples sistemas virtuales en una única máquina física.

Existen diferentes tecnologías de virtualización, como VMware, Microsoft Hyper-V, KVM (Kernel-based Virtual Machine), Xen, entre otras. Estas tecnologías permiten la creación, gestión y asignación de recursos a las máquinas virtuales, así como la migración de las mismas entre servidores físicos.

Los servidores de virtualización son ampliamente utilizados en entornos empresariales, centros de datos y proveedores de servicios en la nube, ya que ofrecen flexibilidad, eficiencia y escalabilidad en la gestión de los recursos computacionales. Además, facilitan la implementación y administración de aplicaciones, simplificando la infraestructura y reduciendo los costos operativos.

15.2. Usos del servidor de virtualización

El servidor de virtualización se utilizará para ofrecer servicios de alojamiento de archivos y streaming. Estos servicios permiten a los usuarios almacenar y acceder a archivos de forma remota, así como transmitir contenido multimedia a través de la red local. El servidor demuestra su capacidad para gestionar y distribuir eficientemente estos servicios, brindando a los usuarios la posibilidad de compartir y acceder a sus archivos y contenido multimedia de manera conveniente y segura. Además, el servidor se utilizará como laboratorio para realizar prácticas y experimentos, lo que proporciona un entorno controlado y flexible para el aprendizaje y desarrollo de nuevas habilidades en el ámbito de la virtualización y los servicios de red.

15.3. Sistema Operativo

Se ha elegido implementar el sistema operativo Debian debido a sus destacadas características y beneficios previamente mencionados. Esta distribución de Linux es reconocida por su sólida estabilidad, alto nivel de seguridad y el amplio respaldo que recibe de la comunidad.

15.4. Instalación del Sistema Operativo

Esta guía proporcionará los pasos necesarios para llevar a cabo una instalación exitosa de Debian, desde la configuración del medio de instalación hasta la personalización de las opciones de seguridad.

15.4.1. Preparación:

Antes de iniciar la instalación, es crucial contar con el medio de instalación adecuado, como un DVD o una unidad USB que contenga la imagen de Debian. Además, se debe seleccionar el idioma preferido y los ajustes regionales pertinentes para garantizar una experiencia personalizada.

15.4.2. Proceso de instalación:

1. **Arranque e inicio de instalación:** Al arrancar el sistema desde el medio de instalación, se presenta una pantalla de inicio donde se selecciona la opción "Instalar". Esto dará inicio al proceso de instalación de Debian en el disco duro del equipo.
2. **Configuración inicial:** Durante esta etapa, se establecen el idioma del sistema y la ubicación geográfica de acuerdo a las preferencias del usuario. Asimismo, si es necesario, se configura la red proporcionando los detalles de conexión correspondientes.
3. **Particionamiento:** Para tener un control total sobre la configuración de las particiones, se elige la opción "Manual" en la sección de particionamiento. A continuación, se selecciona el disco donde se desea instalar Debian y se procede a crear una partición primaria o extendida que abarque todo el espacio disponible en el disco. Esta partición será utilizada como contenedor de LVM (Logical Volume Manager).
4. **Configuración de volúmenes lógicos:** Dentro de la partición creada, se configura el gestor de volúmenes lógicos. Aquí, se crea un grupo de volúmenes asignándole un nombre y seleccionando la partición anteriormente creada como miembro del grupo. Posteriormente, se crea un único volumen lógico con el tamaño suficiente para completar la instalación.
5. **Detalles de configuración:** En esta etapa, se definen los detalles del volumen lógico, como asignarle un nombre y establecer un punto de montaje, como por ejemplo, el volumen raíz ("/"). Esto permitirá un correcto funcionamiento del sistema operativo.
6. **Configuración adicional:** Una vez completada la instalación básica, se continúa con la selección de componentes y paquetes deseados, así como la configuración del gestor de arranque según las necesidades del usuario. También se realizan configuraciones adicionales para adaptar el sistema a requerimientos específicos.
7. **Seguridad y usuarios:** Durante la instalación, se solicita al usuario que elija una contraseña para el usuario root y se ofrecen opciones para configurar otros usuarios y opciones de seguridad adicionales según sea necesario. Estas medidas garantizan la protección de los datos y la integridad del sistema.

15.5. LVM

15.5.1. ¿Que es LVM?

LVM (Logical Volume Manager) es una tecnología de administración de volúmenes lógicos que se utiliza en sistemas operativos como Linux. Proporciona una capa de abstracción entre los sistemas de archivos y los dispositivos físicos de almacenamiento, lo que permite una gestión más flexible y eficiente de los recursos de almacenamiento.

En lugar de manejar directamente las particiones físicas del disco duro, LVM organiza el espacio de almacenamiento en volúmenes físicos (Physical Volumes), que pueden ser particiones individuales, discos duros completos o incluso dispositivos de almacenamiento en red. Estos volúmenes físicos se agrupan en grupos de volúmenes (Volume Groups), que actúan como contenedores para los volúmenes lógicos.

Los volúmenes lógicos (Logical Volumes) son unidades de almacenamiento virtuales creadas a partir de los grupos de volúmenes. Se pueden asignar diferentes tamaños y configuraciones a los volúmenes lógicos

según las necesidades del usuario. Además, los volúmenes lógicos pueden ser redimensionados de forma dinámica, lo que facilita la administración del espacio de almacenamiento en tiempo real.

Una ventaja importante de LVM es la capacidad de agregar varios volúmenes físicos en un grupo de volúmenes y distribuir el espacio de manera más flexible entre los volúmenes lógicos. Esto permite combinar varios discos duros en un solo espacio de almacenamiento lógico, lo que mejora la capacidad de expansión y la tolerancia a fallos del sistema.

Además, LVM proporciona características avanzadas como instantáneas (snapshots), que permiten crear copias puntuales de los volúmenes lógicos para fines de respaldo o recuperación de datos.

En resumen, LVM es una tecnología de administración de volúmenes lógicos que ofrece flexibilidad, escalabilidad y características avanzadas de gestión de almacenamiento en sistemas operativos Linux. Permite una asignación más eficiente de los recursos de almacenamiento y facilita la administración y expansión del espacio de almacenamiento en tiempo real.

15.5.2. Gestión volúmenes lógicos

Se crearán los volúmenes lógicos correspondientes y se les asignarán los sistemas de archivos adecuados.

La sintaxis básica para crear volúmenes lógicos es la siguiente:

lvcreate **-L** *«tamaño»***G -n** *«nombre del volumen» «nombre del grupo»*

- **-L** Tamaño en GB o MB.
- **-n** Nombre para el (LV) y el nombre del VG con el que se trabajara.

```
1
2       lvcreate -L 2G -n home ema
3       lvcreate -L 100G -n ISOS ema
4       lvcreate -L 100G -n MV ema
5       lvcreate -L 5G -n opt ema
6       lvcreate -L 1G -n tmp ema
7       lvcreate -L 10G -n usr ema
8       lvcreate -L 1G -n var-tmp ema
9       lvcreate -L 3G -n var-log ema
10      lvcreate -L 2G -n swap ema
11
12
```

Código 15.1: Creación de LV

```
1
2       mkfs.ext4 /dev/mapper/ema-var--log
3       mkfs.ext4 /dev/mapper/ema-var--tmp
4       mkfs.ext4 /dev/mapper/ema-tmp
5       mkfs.ext4 /dev/mapper/ema-opt
6       mkfs.ext4 /dev/mapper/ema-home
7       mkfs.ext4 /dev/mapper/ISOS
8       mkfs.ext4 /dev/mapper/MV
9       mkfs.ext4 /dev/mapper/ema-usr
10      mkswap /dev/mapper/ema-swap
11
12
```

Código 15.2: Crear Sistema de Archivo

Una vez que se han creado los sistemas de archivos correspondientes, llega el momento de montar los volúmenes lógicos (LV) y mover el contenido de los directorios a sus respectivos LV.

El primer paso es montar cada LV en un punto de montaje adecuado utilizando el comando mount. Por ejemplo:

```
1
2     mv /mnt
3
4     mkdir {var-log,opt,home,usr,ISOS,MV}
5
6     mount /dev/mapper/ema-var--log var-log
```

```
    mount /dev/mapper/ema-opt opt
    mount /dev/mapper/ema-home home
    mount /dev/mapper/ema-usr usr


```

Código 15.3: Crear directorios y montar los (LVs)

.

Una vez montados los LV, se procede a mover el contenido de los directorios a los LV correspondientes. Esto se puede realizar utilizando comando mv o los comandos de copia, como cp o rsync.

```

    #Mover el contenido de los directorios
    mv -f var/log/* var-log
    mv opt/* opt
    mv -f home/* home
    mv usr/* usr
    mv srv/* srv
    #Eliminar archivos temporales
    rm -rf var/tmp/*
    rm -rf tmp/*


```

Código 15.4: Mover directorios

El comando anterior movera recursivamente todo el contenido del directorio original al LV correspondiente, preservando permisos y propiedades.

Es importante verificar que todos los archivos y directorios se hayan movido correctamente al LV antes de continuar.

15.6. fstab

El archivo fstab (File System Table) es un archivo de configuración en sistemas operativos basados en Linux que contiene información sobre los sistemas de archivos y sus opciones de montaje durante el arranque del sistema.

En el archivo fstab, se definen las particiones, dispositivos y volúmenes lógicos (LV) que deben montarse automáticamente en puntos específicos del sistema de archivos. Cada línea del archivo fstab corresponde a un sistema de archivos y sigue una estructura específica con campos separados por espacios o tabulaciones.

Los campos son los siguientes:

1. **Dispositivo:** Especifica el dispositivo de almacenamiento (como una partición o un LV) que se va a montar. Punto de montaje: Indica el directorio en el cual se montará el dispositivo.
2. **Tipo de sistema de archivos:** Especifica el tipo de sistema de archivos del dispositivo, como ext4, ntfs, xfs, entre otros.
3. **Opciones de montaje:** Define las opciones de montaje específicas para el dispositivo, como la configuración de permisos, la gestión de errores, el modo de acceso, entre otros.
4. **Opciones de respaldo:** Establece opciones de respaldo para el dispositivo, como la frecuencia de respaldo y otras opciones relacionadas con la integridad de los datos.
5. **Campo de comprobación:** Este campo se utiliza por herramientas de verificación de sistemas de archivos y generalmente se establece en 0 o 1.

La edición del archivo fstab permite configurar los sistemas de archivos y volúmenes lógicos para que se monten automáticamente durante el inicio del sistema, evitando la necesidad de realizar montajes manuales.

```
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
/dev/mapper/ema-raiz /               ext4    errors=remount-ro 0       1
# /boot/efi was on /dev/mmcblk0p1 during installation
UUID=2321-DD09  /boot/efi       vfat    umask=0077      0       1
/dev/mapper/ema-home /home ext4 rw,noexec,auto,nodev,nosuid,noatime,async,nouser  0  2
/dev/mapper/ema-opt        /opt ext4 rw,exec,auto,nodev,nosuid,noatime,async,nouseri 0 0
/dev/mapper/ema-ISOS        /mnt/ISOS        ext4        defaults        0        0
/dev/mapper/ema-MV            /mnt/MV            ext4        defaults        0        0
/dev/mapper/ema-var--log /var/log ext4 rw,noexec,auto,nodev,nosuid,noatime,async,nouser  0 0
/dev/mapper/ema-var--tmp /var/tmp ext4 rw,noexec,auto,nodev,nosuid,noatime,async,nouser  0 0
/dev/mapper/ema-tmp        /tmp ext4 rw,noexec,auto,nodev,nosuid,noatime,async,nouser 0 0
/dev/mapper/ema-usr        /usr        ext4        defaults        0        0
/dev/mapper/ema-swap none            swap    sw              0       0
```

15.7. Hypervisor

Un hypervisor, también conocido como monitor de máquina virtual o VMM (Virtual Machine Monitor), es un software o firmware que permite la virtualización y gestión de máquinas virtuales (VM) en un entorno de computación.

El hypervisor se ejecuta directamente sobre el hardware físico de un servidor y proporciona una capa de abstracción que permite crear y ejecutar múltiples máquinas virtuales, cada una de las cuales puede ejecutar su propio sistema operativo y aplicaciones de forma aislada. Esto permite que varios sistemas operativos y aplicaciones se ejecuten simultáneamente en un único servidor físico, lo que maximiza la utilización de recursos y proporciona una mayor flexibilidad y eficiencia.

Existen dos tipos principales de hypervisors:

- **Hypervisor de tipo 1 o "bare metal":** Este hypervisor se instala directamente en el hardware del servidor y gestiona directamente los recursos del sistema, sin necesidad de un sistema operativo adicional. Proporciona un rendimiento y una eficiencia óptimos y se utiliza comúnmente en entornos empresariales y de centros de datos.
- **Hypervisor de tipo 2 o "hosted":** Este hypervisor se ejecuta como una aplicación dentro de un sistema operativo existente. Requiere un sistema operativo base para funcionar y es más comúnmente utilizado en entornos de desarrollo y pruebas, así como en entornos de escritorio virtual.

Los hypervisors permiten la consolidación de servidores físicos, el aislamiento de recursos, la migración en vivo de máquinas virtuales, la creación rápida de entornos de prueba y muchas otras capacidades que facilitan la administración y el despliegue de infraestructuras de TI. Son fundamentales en la virtualización y son ampliamente utilizados en entornos de servidores y centros de datos modernos.

Es válido recalcar que se puede denominar al hypervisor como un "servidor de virtualización". Sin embargo, es importante tener en cuenta que el término "servidor de virtualización"puede ser interpretado de diferentes maneras según el contexto.

En un sentido, el hypervisor puede considerarse como el componente principal de un servidor de virtualización, ya que su función principal es administrar las máquinas virtuales y los recursos del sistema. De esta forma, se puede afirmar que el hypervisor desempeña el rol de un servidor de virtualización.

Por otro lado, el término "servidor de virtualización"también puede referirse al hardware físico en el cual se ejecutan las máquinas virtuales. En este caso, el servidor de virtualización sería el equipo físico que aloja y ejecuta las máquinas virtuales, y el hypervisor sería el software que posibilita la virtualización en dicho servidor.

15.7.1. KVM

KVM (Kernel-based Virtual Machine) es un hypervisor de código abierto de tipo 1, también conocido como "bare metal". Esto significa que se instala directamente en el hardware físico de un servidor y actúa como una capa de virtualización entre el hardware y los sistemas operativos invitados.

Como hypervisor de tipo 1, KVM no requiere un sistema operativo adicional para funcionar.

Permite convertir el kernel de Linux en un hypervisor, permitiendo la virtualización de servidores y la ejecución de múltiples sistemas operativos invitados en un solo host.

KVM utiliza las extensiones de virtualización de hardware presentes en los procesadores modernos (como Intel VT-x o AMD-V) para mejorar el rendimiento y la eficiencia de la virtualización. Proporciona una capa de abstracción entre el hardware físico y los sistemas operativos invitados, lo que permite que cada uno de ellos se ejecute de manera aislada y con sus propios recursos asignados.

Al ser parte del kernel de Linux, KVM ofrece una integración estrecha con el ecosistema Linux y aprovecha las características y funcionalidades del kernel. Permite la virtualización de sistemas operativos Linux y también es compatible con la virtualización de sistemas operativos Windows y otros sistemas operativos invitados.

KVM es ampliamente utilizado en entornos empresariales y en la nube para la consolidación de servidores, la creación de entornos de desarrollo y pruebas, la implementación de infraestructuras virtuales y otros casos de uso de virtualización. Además, cuenta con una gran comunidad de desarrollo y soporte, lo que garantiza su continua mejora y actualización.

15.7.2. QEMU/KVM

QEMU (Quick Emulator) es un software de emulación y virtualización de código abierto que se utiliza junto con KVM (Kernel-based Virtual Machine) para proporcionar capacidades de virtualización completas en sistemas Linux.

QEMU es un emulador de sistema completo que permite ejecutar sistemas operativos y aplicaciones de diferentes arquitecturas en un entorno virtualizado. Proporciona emulación de hardware a nivel de instrucción, lo que le permite simular diferentes arquitecturas de CPU y ejecutar sistemas operativos diseñados para esas arquitecturas en un sistema anfitrión diferente.

En combinación con KVM, QEMU permite la virtualización de sistemas completos y proporciona una capa de abstracción entre el hardware físico y los sistemas operativos invitados. KVM se encarga de la virtualización de hardware y proporciona un rendimiento óptimo, mientras que QEMU emula los dispositivos y brinda soporte para sistemas operativos invitados que no son nativos de la arquitectura del host.

QEMU también puede utilizarse de forma independiente para emular hardware y ejecutar sistemas operativos en un entorno puramente emulado, sin utilizar la virtualización de hardware de KVM. Esto puede ser útil para el desarrollo y la depuración de software, así como para ejecutar sistemas operativos antiguos o exóticos en un entorno moderno.

15.7.3. libvirt

Libvirt es una biblioteca y conjunto de herramientas de administración de virtualización de código abierto que brinda una capa de abstracción y una interfaz unificada para administrar diferentes tecnologías de virtualización, incluyendo QEMU/KVM.

En el contexto de QEMU/KVM, libvirt se utiliza como una interfaz de administración para gestionar las máquinas virtuales y los recursos del sistema. Proporciona una API y una interfaz de línea de comandos (CLI) que permiten realizar tareas como la creación, configuración, inicio, parada, migración y supervisión de las máquinas virtuales.

Libvirt se encarga de interactuar con el hypervisor KVM y el emulador QEMU para gestionar las operaciones de virtualización. Permite la gestión centralizada de múltiples hosts y proporciona una capa de abstracción que permite la interoperabilidad entre diferentes tecnologías de virtualización.

Además de la administración de máquinas virtuales, libvirt ofrece características como el control de almacenamiento, la configuración de redes virtuales, la gestión de dispositivos y la monitorización de recursos. También permite la integración con otras herramientas de administración de virtualización y orquestación, como virt-manager, oVirt, OpenStack y Kubernetes.

15.7.4. Instalación

En esta etapa, se instalaran las herramientas necesarias para configurar el servidor de virtualización. Algunas de las herramientas importantes que se deben instalar incluyen:

```
1
2       apt install qemu qemu-kvm qemu-system qemu-utils libvirt-clients libvirt-daemon-system
    virtinst virt-manager bridge-utils
3
4
```

Código 15.5: instalación de las herramientas

- **qemu:** Es un emulador de procesador utilizado en la virtualización de sistemas completos.
- **qemu-kvm:** Permite utilizar la virtualización basada en hardware KVM junto con QEMU.
- **qemu-system:** Proporciona la funcionalidad principal del sistema QEMU.
- **qemu-utils:** Incluye utilidades adicionales para QEMU.
- **libvirt-clients:** Contiene herramientas de línea de comandos para administrar libvirt y los servidores de virtualización.
- **libvirt-daemon-system:** Es el demonio principal de libvirt que permite la gestión y administración de los servidores de virtualización.
- **virtinst:** Proporciona herramientas para crear y administrar máquinas virtuales.
- **virt-manager:** Es una interfaz gráfica de usuario para administrar máquinas virtuales y servidores de virtualización.
- **bridge-utils:** Incluye herramientas para configurar y administrar puentes de red en el sistema.

15.8. Configuración de Red

Una vez completada la instalación, el servidor de virtualización estará listo para su funcionamiento. Sin embargo, es importante realizar la configuración de red adicional para garantizar la conectividad de las máquinas virtuales con la red local.

En primer lugar, se debe crear una interfaz llamada br0, que actuará como un puente entre la interfaz física del servidor y las máquinas virtuales. Esta configuración permitirá que las máquinas virtuales obtengan direcciones IP del mismo rango que las computadoras de la red local, en lugar de obtener IPs de una red virtual.

La creación de la interfaz br0 se puede realizar utilizando herramientas como ”bridge-utils”. A través de la configuración de este puente, se establecerá una conexión entre la interfaz física del servidor y las máquinas virtuales, lo que permitirá la comunicación fluida entre ellas y los dispositivos de la red local.

```
1    vi /etc/network/interfaces
2
```

Código 15.6: Interfaces

La configuración proporcionada indica la configuración de la interfaz de red principal en el archivo */etc/network/interfaces*.

```
1
2     The primary network interface
3     allow-hotplug eth1
4     iface eth1 inet static
5     auto br0
6     iface br0 inet static
7     address 192.168.1.222
8     netmask 255.255.255.0
9     gateway 192.168.1.1
10    bridge_ports eth1
11    up /usr/sbin/brctl std br0 on
12
```

```
13
```

Código 15.7: Configuración de red

Estas líneas definen la configuración para la interfaz física de red eth1. La directiva " allow-hotplug" permite que la interfaz se active automáticamente cuando se conecta en caliente. La opción "inet static" especifica que se utilizará una configuración de IP estática para la interfaz.

```
1     allow-hotplug eth1
2     iface eth1 inet static
3
```

Código 15.8: interfaz

Estas líneas definen la configuración para la interfaz de puente virtual br0, que se utiliza para conectar las máquinas virtuales a la red física. La directiva "auto" asegura que la interfaz de puente se active automáticamente durante el inicio del sistema.

Las líneas "address", "netmask" y "gateway" especifican la dirección IP, la máscara de red y la puerta de enlace para la interfaz de puente br0. Estos valores deben ajustarse según la configuración específica de la red.

La línea "bridge_ports" especifica la interfaz física (eth1) que se conectará al puente.

La directiva "up" ejecuta el comando especificado (/usr/sbin/brctl std br0 on) cuando se activa la interfaz. Este comando habilita la interfaz de puente.

Después de realizar estos cambios en el archivo */etc/network/interfaces*, se debe guardar y reiniciar el servicio de red para que la configuración tenga efecto.

```
1
2     auto br0
3     iface br0 inet static
4     address 192.168.1.222
5     netmask 255.255.255.0
6     gateway 192.168.1.1
7     bridge_ports eth1
8     up /usr/sbin/brctl std br0 on
9
10
```

Código 15.9: interfaz

15.8.1. iproute

Iproute2 es una suite de herramientas de red utilizada en sistemas Linux para administrar y configurar aspectos relacionados con el enrutamiento de paquetes IP y otras funcionalidades de red. Esta suite reemplazó a la antigua utilidad net-tools y ofrece un conjunto más avanzado y flexible de herramientas para el control y monitoreo de la red.

Algunas de las herramientas clave incluidas en Iproute2 son:

- **ip:** Esta herramienta es la más importante de la suite y se utiliza para administrar y configurar interfaces de red, direcciones IP, enrutamiento, túneles y otras funcionalidades de red. Proporciona una amplia gama de opciones para el control y monitoreo de la configuración de red.
- **tc:** Esta herramienta se utiliza para configurar y administrar la calidad de servicio (QoS) en el sistema. Permite controlar el ancho de banda, la prioridad y otras características de tráfico de red.
- **ss:** Esta herramienta se utiliza para mostrar estadísticas detalladas de sockets y conexiones de red en el sistema. Proporciona información sobre el estado de los sockets, las conexiones establecidas y los puertos utilizados.
- **bridge:** Esta herramienta se utiliza para configurar y administrar puentes de red (bridge) en el sistema. Permite crear puentes virtuales para interconectar diferentes interfaces de red y facilitar el tráfico entre ellas.
- **rtacct:** Esta herramienta se utiliza para recopilar estadísticas de enrutamiento en el sistema. Permite monitorear el tráfico de red, los caminos de enrutamiento y otras métricas relacionadas.

- **Iproute2** proporciona una mayor flexibilidad y control sobre la configuración de red en comparación con las herramientas más antiguas como ifconfig y route. Es ampliamente utilizado en entornos de red avanzados y sistemas Linux modernos para la gestión y el control precisos de la configuración de red.

Haciendo uso de IP, se creará un script que se ejecutará al inicio del sistema mediante crontab para configurar las tres interfaces asociadas a la interfaz virtual br0. Estas interfaces funcionarán como un puente entre la interfaz física y otras redes.

A continuación se presenta el contenido del script:

```
#! /bin/bash
echo "Configuración de las interfaces."
ip link add link br0 name dmz type vlan id 100
ip addr add 192.168.100.1/24 brd 192.168.100.255 dev dmz
ip link set dmz up
ip link add link br0 name lan1 type vlan id 101
ip addr add 192.168.101.1/24 brd 192.168.101.1 dev lan1
ip link set lan1 up
ip link add link br0 name lan2 type vlan id 102
ip addr add 192.168.102.1/24 brd 192.168.102.255 dev lan2
ip link set lan2 up
echo "Se han creado las interfaces de red correctamente."
```

Código 15.10: script de configuración de red

El siguiente código configura las interfaces de red en Linux:

- Se muestra el mensaje 'Configuración de las interfaces'.
- Se añade una interfaz virtual llamada "dmz" a la interfaz "br0" con el identificador de VLAN 100.
- Se asigna la dirección IP 192.168.100.1/24 a la interfaz "dmz" y se establece la dirección de broadcast.
- Se activa la interfaz "dmz".
- Se añade una interfaz virtual llamada "lan1" a la interfaz "br0" con el identificador de VLAN 101.
- Se asigna la dirección IP 192.168.101.1/24 a la interfaz "lan1" y se establece la dirección de broadcast.
- Se activa la interfaz "lan1".
- Se añade una interfaz virtual llamada "lan2" a la interfaz "br0" con el identificador de VLAN 102.
- Se asigna la dirección IP 192.168.102.1/24 a la interfaz "lan2" y se establece la dirección de broadcast.
- Se activa la interfaz "lan2".
- Se muestra el mensaje 'Se han creado las interfaces de red correctamente.'

Para programar la ejecución del script al inicio del sistema mediante crontab, se deben seguir los siguientes pasos:

- Abrir una terminal y ejecutar el siguiente comando para editar la lista de tareas cron:

```
crontab -e
```

Código 15.11: crontab

- En el archivo crontab, agregar la siguiente línea al final para programar la ejecución del script al inicio del sistema:

```
@reboot /ruta/al/script.sh
```

Código 15.12: ruta del archivo

Con esta configuración, el script se ejecutará automáticamente al inicio del sistema utilizando crontab. Las tres interfaces serán creadas y configuradas según las especificaciones dentro del script, y estarán asociadas a la interfaz virtual br0, lo que permitirá el puenteo entre la interfaz física y otras redes.

15.8.2. Servidor DNS

Es importante mencionar que, también es necesario configurar los servidores DNS. Para realizar esta configuración, se debe editar el archivo */etc/resolv.conf*.

Dentro de este archivo, se especifican los servidores DNS que el sistema utilizará para resolver las consultas de nombres de dominio. Para este caso en particular, se utilizará la dirección IP 1.1.1.1 como servidor DNS.

Al editar el archivo */etc/resolv.conf*, se debe agregar la siguiente línea:

```
nameserver 1.1.1.1
```

Código 15.13: resolv

- **La dirección IP "1.1.1.1"** corresponde a un servidor DNS público operado por Cloudflare. Este servidor ofrece una solución rápida y privada para la navegación por Internet. A diferencia de la mayoría de los servidores de DNS, 1.1.1.1 se distingue por su política de no vender los datos de los usuarios a los anunciantes. Esto implica que se compromete a mantener la privacidad de la información relacionada con las consultas DNS realizadas a través de su servicio.

15.8.3. NFS

NFS (Network File System) es un protocolo de red ampliamente utilizado que posibilita a los sistemas operativos Unix y Linux compartir archivos y directorios entre computadoras conectadas en una red. Este protocolo facilita la colaboración y el acceso compartido a datos de forma transparente, como si los archivos estuvieran almacenados localmente en cada sistema.

En la configuración implementada, se utilizará NFS para compartir directorios entre dos máquinas virtuales. La primera máquina virtual albergará el servidor multimedia Emby, que ofrecerá la capacidad de administrar y transmitir contenido multimedia a través de la red. Mediante NFS, se compartirá un directorio que contendrá el contenido multimedia centralizado, lo que permitirá a Emby acceder a los archivos de manera eficiente y proporcionar una experiencia de visualización fluida a los usuarios.

La segunda máquina virtual albergará el servidor Nextcloud, una plataforma de colaboración y almacenamiento en la nube. Nuevamente, se utilizará NFS para compartir otro directorio donde se almacenarán los datos de los usuarios que utilizarán el servidor Nextcloud. Esta configuración permitirá un acceso rápido y seguro a los datos de los usuarios, mejorando la eficiencia y el rendimiento del servidor Nextcloud.

La instalación y configuración de NFS para compartir los directorios con contenido multimedia para el servidor Emby y el almacenamiento de datos de los usuarios que utilizarán el servidor Nextcloud es una forma eficiente de centralizar y acceder a los archivos en una red. A continuación, se presentan los pasos generales para lograr esto:

- **Paso 1:** Instalar el paquete nfs-kernel-server

```
apt install nfs-kernel-server
```

Código 15.14: nfs

- **Paso 2:** Configurar el servidor NFS editando el archivo */etc/exports*

```
#Agrega las líneas siguientes al archivo para compartir los dos directorios:
/ruta/multimedia 192.168.100.2(rw,sync,no_subtree_check)
/ruta/datos 192.168.100.3(rw,sync,no_subtree_check)
```

Código 15.15: export

- **Paso 3:** Reiniciar el servidor NFS

  Una vez que se hayan configurado los directorios que se deseas compartir, se debe reinicia el servicio NFS para aplicar los cambios.

```
systemctl restart nfs-kernel-server
```

Código 15.16: export

Ahora, los dos directorios estarán compartidos mediante NFS y podrán ser montados en el servidor Emby y el servidor Nextcloud para acceder a los archivos de contenido multimedia y almacenar los datos de los usuarios, respectivamente.

15.8.4. IP forwarding

IP forwarding, o reenvío de IP, es una funcionalidad en los sistemas operativos que permite al sistema enrutar y reenviar paquetes IP entre diferentes interfaces de red. En otras palabras, cuando un paquete IP llega a una interfaz de red en un sistema con IP forwarding habilitado, el sistema puede determinar la mejor ruta para enviar el paquete a través de otra interfaz de red hacia su destino final.

El IP forwarding es una característica clave en el enrutamiento de redes, ya que permite que los paquetes IP atraviesen múltiples saltos o routers para llegar a su destino final. En un sistema con IP forwarding habilitado, actúa como un enrutador al tomar decisiones sobre la ruta óptima para reenviar los paquetes IP según las tablas de enrutamiento configuradas.

Cuando se habilita el IP forwarding, el sistema opera en el nivel de red (capa 3) del modelo OSI, permitiendo la conectividad entre diferentes redes y subredes.[4]

Habilitar IP forwarding

Para habilitar el IP forwarding (reenvío de IP) en un sistema Linux, se puede realizar a través de la modificación del archivo de configuración */etc/sysctl.conf*.

```
vi /etc/sysctl.conf
```

Código 15.17: IP forwarding

- descomentar net.ipv4.ip_forward y colocar el valor en 1 (por defecto, puede estar comentada).

Para aplicar los cambios realizados en el archivo */etc/sysctl.conf*, se debe ejecutar el siguiente comando que cargara la configuración:

```
sysctl -p
```

Código 15.18: Cargar Configuración

15.9. iptables

IPTABLES es una herramienta de administración de firewall en sistemas operativos Linux. Permite configurar reglas y políticas de filtrado de paquetes IP para controlar el tráfico de red entrante y saliente.

Mediante IPTABLES, se pueden establecer reglas para permitir o denegar el paso de paquetes según diversos criterios, como dirección IP de origen o destino, puerto de origen o destino, protocolo, estado de

la conexión, entre otros. También es posible realizar traducción de direcciones de red (NAT) y redirección de puertos (port forwarding).

La sintaxis para configurar reglas en IPTABLES puede variar según la versión del sistema operativo y las preferencias del administrador del sistema. Sin embargo, el proceso general implica especificar las condiciones de la regla (como dirección IP, puerto y protocolo) y la acción que se debe tomar (permitir, denegar, redirigir, etc.).

15.9.1. Configurar reglas

Una vez finalizadas las configuraciones de red, se procederá a definir una serie de reglas en IPTABLES con el objetivo de permitir el acceso a los servicios desde Internet, proteger las distintas redes y proporcionar salida a Internet para los hosts de cada red.

Para lograr esto, se configurarán las reglas en IPTABLES teniendo en cuenta los siguientes aspectos:

- **Acceso a servicios desde Internet:** Se establecerán reglas que permitan el acceso a los servicios específicos que se deseen exponer al tráfico proveniente de Internet. Estas reglas estarán basadas en los puertos y protocolos utilizados por los servicios.
- **Protección de redes:** Se crearán reglas en IPTABLES para proteger las distintas redes, bloqueando o limitando el acceso no autorizado desde Internet. Esto se logrará mediante reglas de filtrado de paquetes que permitan únicamente el tráfico necesario y bloqueen cualquier intento de acceso no autorizado.
- **Salida a Internet:** Se configurarán reglas que permitan a los hosts de cada red acceder a Internet, asegurando que el tráfico saliente esté correctamente encaminado y que se apliquen políticas de seguridad según sea necesario.

A continuación se presenta el contenido del script que se ejecutará en cada inicio del sistema, el cual contiene la definición de las reglas de IPTABLES:

```
1
2   ! /bin/bash
3   #
    ##############################################################################################################
4   #SERVIDOR DE VIRTUALIZACION
5   #br0 => red local => 192.198.1.222
6   #dmz => vlan => dentro del servidor => 192.168.100.0/24
7   #lan1 => vlan => dentro del servidor => 192.168.101.0/24
8   #lan2 => vlan => dentro del servidor => 192.168.102.0/24
9   echo "Comienzo de las Reglas"
10
11    ##############################################################################################
12  #VARIABLES
13  MAC_PC1="00:00:00:00:00:00"
14  MAC_NET="00:00:00:00:00:00"
15  MAC_CEL="00:00:00:00:00:00"
16  ##############################################################################################
17  #LIMPIAR REGLAS
18  /usr/sbin/iptables -F
19  /usr/sbin/iptables -X
20  /usr/sbin/iptables -Z
21  /usr/sbin/iptables -t nat -F
22
23  ##############################################################################################
24  #POLITICAS POR DEFECTO
25  /usr/sbin/iptables -P INPUT DROP
26  /usr/sbin/iptables -P FORWARD DROP
27  /usr/sbin/iptables -P OUTPUT DROP
28
29  ##############################################################################################
30  #LO QUE LLEGUE DE INTERNET
31  /usr/sbin/iptables -t nat -A PREROUTING -i br0 -p tcp --dport 2220 -j DNAT --to
    192.168.100.2:2222
32  /usr/sbin/iptables -t nat -A PREROUTING -i br0 -p tcp --dport 8096 -j DNAT --to
    192.168.100.2:8096
33  /usr/sbin/iptables -t nat -A PREROUTING -i br0 -p tcp --dport 2221 -j DNAT --to
    192.168.100.3:2222
```

```
34  /usr/sbin/iptables -t nat -A PREROUTING -i br0 -p tcp --dport 80 -j DNAT --to
    192.168.100.3:80
35  /usr/sbin/iptables -t nat -A PREROUTING -i br0 -p tcp --dport 443 -j DNAT --to
    192.168.100.3:443
36
37  ##############################################################################################
38  #RESPUESTA A LAS COMUNICACIONES YA ESTABLECIDAS
39  /usr/sbin/iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
40  /usr/sbin/iptables -A OUTPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
41  /usr/sbin/iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
42
43  ##############################################################################################
44  #PERMITEN LOS PING
45  /usr/sbin/iptables -A INPUT -i dmz -p icmp --icmp-type echo-request -j ACCEPT #PING
46  /usr/sbin/iptables -A INPUT -i lan1 -p icmp --icmp-type echo-request -j ACCEPT #PING
47  /usr/sbin/iptables -A INPUT -i lan2 -p icmp --icmp-type echo-request -j ACCEPT #PING
48  /usr/sbin/iptables -A OUTPUT -o dmz -p icmp --icmp-type echo-request -j ACCEPT #PING
49  /usr/sbin/iptables -A OUTPUT -o lan1 -p icmp --icmp-type echo-request -j ACCEPT #PING
50  /usr/sbin/iptables -A OUTPUT -o lan2 -p icmp --icmp-type echo-request -j ACCEPT #PING
51  #/usr/sbin/iptables -A FORWARD -i dmz -o br0 -p icmp --icmp-type echo-request -j ACCEPT
52  #/usr/sbin/iptables -A FORWARD -i lan1 -o br0 -p icmp --icmp-type echo-request -j ACCEPT
53  #/usr/sbin/iptables -A FORWARD -i lan2 -o br0 -p icmp --icmp-type echo-request -j ACCEPT
54
55  ##############################################################################################
56  #ADMINISTRAR FIREWALL DESDE MI RED LAN (NETBOOK,CELULAR,PC DESKTOP)
57
58  /usr/sbin/iptables -A INPUT -i br0 -m mac --mac-source $MAC_CEL -p tcp --dport 2222 -j
    ACCEPT #CELULAR
59  /usr/sbin/iptables -A INPUT -i br0 -m mac --mac-source $MAC_PC1 -p tcp --dport 2222 -j
    ACCEPT #MI PC
60  /usr/sbin/iptables -A INPUT -i br0 -m mac --mac-source $MAC_PC1 -p tcp --dport 5900:5920 -
    j ACCEPT #SPICE
61  /usr/sbin/iptables -A INPUT -i br0 -m mac --mac-source $MAC_NET -p tcp --dport 2222 -j
    ACCEPT #NET
62  /usr/sbin/iptables -A INPUT -i br0 -m mac --mac-source $MAC_NET -p tcp --dport 5900:5920 -
    j ACCEPT #SPICE
63
64  ##############################################################################################
65  #NFS => COMPARTIR DISCO CON CONTENIDO MULTIMEDIA
66  /usr/sbin/iptables -A INPUT -s 192.168.100.2 -p tcp --dport 2049 -j ACCEPT #NFS
67  /usr/sbin/iptables -A INPUT -s 192.168.100.2 -p udp --dport 2049 -j ACCEPT #NFS
68  /usr/sbin/iptables -A INPUT -s 192.168.100.3 -p tcp --dport 2049 -j ACCEPT #NFS
69  /usr/sbin/iptables -A INPUT -s 192.168.100.3 -p udp --dport 2049 -j ACCEPT #NFS
70
71  ##############################################################################################
72  #CONEXION SSH DESDE EL FIREWALL A LAS DEMAS REDES
73  /usr/sbin/iptables -A OUTPUT -o dmz -p tcp --dport 2222 -j ACCEPT #SSH
74  /usr/sbin/iptables -A OUTPUT -o lan1 -p tcp --dport 2222 -j ACCEPT #SSH
75  /usr/sbin/iptables -A OUTPUT -o lan2 -p tcp --dport 2222 -j ACCEPT #SSH
76  /usr/sbin/iptables -A OUTPUT -o br0 -p tcp --dport 2222 -j ACCEPT #SSH
77
78  ##############################################################################################
79  #PUERTOS QUE USA EL FIREWALL
80  /usr/sbin/iptables -A OUTPUT -o br0 -p tcp --dport 2222 -j ACCEPT #SSH
81  /usr/sbin/iptables -A OUTPUT -o br0 -p tcp --dport 80 -j ACCEPT #HTTP
82  /usr/sbin/iptables -A OUTPUT -o br0 -p tcp --dport 443 -j ACCEPT #HTTPS
83  /usr/sbin/iptables -A OUTPUT -o br0 -p tcp --dport 53 -j ACCEPT #DNS TCP
84  /usr/sbin/iptables -A OUTPUT -o br0 -p udp --dport 53 -j ACCEPT #DNS UDP
85
86  ##############################################################################################
87  #PUERTOS POR LOS QUE SE PODRA SALIR DE LA RED
88
89  #dmz
90  /usr/sbin/iptables -A FORWARD -i dmz -o br0 -p tcp --dport 80 -j ACCEPT
91  /usr/sbin/iptables -A FORWARD -i dmz -o br0 -p tcp --dport 443 -j ACCEPT
92  /usr/sbin/iptables -A FORWARD -i dmz -o br0 -p tcp --dport 53 -j ACCEPT
93  /usr/sbin/iptables -A FORWARD -i dmz -o br0 -p udp --dport 53 -j ACCEPT
94
95  #lan1
96  /usr/sbin/iptables -A FORWARD -i lan1 -o br0 -p tcp --dport 80 -j ACCEPT
97  /usr/sbin/iptables -A FORWARD -i lan1 -o br0 -p tcp --dport 443 -j ACCEPT
98  /usr/sbin/iptables -A FORWARD -i lan1 -o br0 -p tcp --dport 53 -j ACCEPT
99  /usr/sbin/iptables -A FORWARD -i lan1 -o br0 -p udp --dport 53 -j ACCEPT
```

```
100
101     #lan2
102     /usr/sbin/iptables -A FORWARD -i lan2 -o br0 -p tcp --dport 80 -j ACCEPT
103     /usr/sbin/iptables -A FORWARD -i lan2 -o br0 -p tcp --dport 443 -j ACCEPT
104     /usr/sbin/iptables -A FORWARD -i lan2 -o br0 -p tcp --dport 53 -j ACCEPT
105     /usr/sbin/iptables -A FORWARD -i lan2 -o br0 -p udp --dport 53 -j ACCEPT
106
107     ##############################################################################################
108     #LAS REDIRECCIONES QUE REALIZARA EL FIREWALL A LOS SERVIDORES
109     /usr/sbin/iptables -A FORWARD -i br0 -o dmz -p tcp --dport 8096 -j ACCEPT
110     /usr/sbin/iptables -A FORWARD -i br0 -o dmz -p tcp --dport 80 -j ACCEPT
111     /usr/sbin/iptables -A FORWARD -i br0 -o dmz -p tcp --dport 443 -j ACCEPT
112     /usr/sbin/iptables -A FORWARD -i br0 -o dmz -p tcp --dport 2222 -j ACCEPT
113
114     ##############################################################################################
115     #EL TRAFICO QUE SE ORIGINE EN LAS SIGUIENTES REDES SE ENMASCARA
116     /usr/sbin/iptables -t nat -A POSTROUTING -s 192.168.100.0/24 -j MASQUERADE
117     /usr/sbin/iptables -t nat -A POSTROUTING -s 192.168.101.0/24 -j MASQUERADE
118     /usr/sbin/iptables -t nat -A POSTROUTING -s 192.168.102.0/24 -j MASQUERADE
119
120
```

Código 15.19: iptables

Este script en bash contiene las reglas de iptables y se guardará en el mismo directorio donde se encuentra el script de configuraciones de red. De manera similar al script anterior, este script se ejecutará en cada inicio del sistema utilizando crontab.

15.10. Seguridad

En la configuración del servidor de virtualización, se procederá a instalar Fail2ban, una herramienta de seguridad que brinda protección contra ataques de fuerza bruta y otros intentos de intrusión. Además, se configurará un bot de Telegram para recibir notificaciones sobre las direcciones IP que son bloqueadas por Fail2ban.

Como medida adicional de seguridad, se implementará la generación de hashes para los archivos que se encuentran en directorios específicos considerados importantes. Esto permitirá realizar comparaciones de hashes en caso de sospechar cambios en el sistema, lo que facilitará la detección de modificaciones no autorizadas.

Al instalar Fail2ban y configurar el bot de Telegram, se fortalecerá la seguridad del servidor, ya que se estarán bloqueando las IP maliciosas y recibiendo notificaciones en tiempo real. Asimismo, mediante la generación de hashes de los archivos, se podrá realizar un seguimiento de su integridad y detectar cualquier alteración no autorizada.

Estas prácticas de seguridad contribuirán a mantener el servidor de virtualización protegido contra posibles amenazas y a garantizar la integridad del sistema en todo momento.

15.10.1. Fail2ban

Fail2Ban es una herramienta de seguridad utilizada para proteger servidores contra ataques de fuerza bruta y otros intentos de intrusión. Su objetivo principal es monitorear los registros del sistema en busca de actividades sospechosas y tomar medidas para mitigar dichos ataques.

Cuando Fail2Ban detecta comportamientos maliciosos, como múltiples intentos fallidos de inicio de sesión desde una misma dirección IP, toma medidas para bloquear temporalmente dicha IP y evitar que el atacante continúe sus intentos de acceso. Además del bloqueo de direcciones IP, Fail2Ban también puede enviar notificaciones al administrador del sistema sobre eventos sospechosos y realizar ajustes en la configuración de seguridad para fortalecer la protección del servidor.

Fail2Ban es altamente personalizable y configurable, lo que permite adaptarlo a las necesidades específicas del servidor y a los patrones de ataque más comunes. Es ampliamente utilizado en entornos de servidores para mejorar la seguridad y proteger los recursos de la red contra posibles amenazas.

Instalación

Para realizar la instalación de Fail2Ban, se ejecuta el comando:

```
1        apt install fail2ban -y
2
```

Código 15.20: Install

Una vez instalado, se inicia el servicio de Fail2Ban mediante el comando

```
1        systemctl start fail2ban
2
```

Código 15.21: iniciar servicio

Se verifica el estado actual del servicio de Fail2Ban ejecutando

```
1        systemctl status fail2ban
2
```

Código 15.22: Ver estado del servicio

Para comprobar la versión instalada de Fail2Ban, se utiliza el comando

```
1        fail2ban-client version
2
```

Código 15.23: Comprobar version

A fin de garantizar que Fail2Ban se ejecute automáticamente después de los reinicios del sistema, se habilita el servicio mediante

```
1        systemctl enable fail2ban.service
2
```

Código 15.24: habilitar servicio

Además, se reinicia el servicio rsyslog para aplicar los cambios realizados, a través del comando

```
1        systemctl restart rsyslog.service
2
```

Código 15.25: Reiniciar rsyslog

Al seguir estos pasos, Fail2Ban se encontrará instalado y configurado en el sistema. Es importante tener en cuenta que se pueden ajustar y personalizar las opciones de configuración según las necesidades particulares del sistema.

Configuración

En el contexto de la configuración de Fail2ban, se creará un archivo llamado ”telegram” dentro del directorio */etc/fail2ban/action.d*. Este archivo contendrá las definiciones necesarias para habilitar las notificaciones a través de Telegram. El propósito de este archivo es permitir una personalización detallada de los mensajes y acciones que Fail2ban ejecutará al enviar notificaciones mediante el servicio de mensajería Telegram cuando ocurran eventos de seguridad relevantes.

Junto con la creación del archivo ”telegram”, se desarrollará un script que incluirá el token de Telegram y contendrá los mensajes que se enviarán. Este script se encargará de interactuar con la API de Telegram y enviar las notificaciones correspondientes cuando sea requerido. Se configurará el archivo ”telegram” dentro de */etc/fail2ban/action.d* para que Fail2ban utilice este script como la acción a ejecutar para las notificaciones por Telegram.

Asimismo, se procederá a crear o modificar el archivo ”jail.local”, que contendrá la definición de la jaula (jail) específica para el servicio SSH. En dicho archivo, se establecerán las reglas y parámetros necesarios para detectar intentos fallidos de inicio de sesión y, en consecuencia, se llamará a Telegram como parte de las acciones de respuesta.

La incorporación de la notificación a través de Telegram en el archivo de la jaula SSH en ”jail.local” permitirá activar esta funcionalidad específicamente cuando se detecten intentos maliciosos de acceso a

través del servicio SSH. De esta manera, Fail2ban utilizará la configuración del archivo ”telegram” en el directorio */etc/fail2ban/action.d* para enviar notificaciones a través de Telegram cuando se produzcan eventos de seguridad relacionados con SSH, ayudando a mantener el servidor protegido y a mantener al administrador informado de posibles amenazas y ataques de fuerza bruta o intrusión no autorizada.

Script

Se ha desarrollado un script para enviar notificaciones a través del servicio de mensajería Telegram. El objetivo principal del script es mantener informado al administrador del sistema sobre eventos relacionados con la seguridad en Fail2ban.

El script se ha almacenado en un archivo llamado *send_telegram_notif.sh*. Este archivo contiene una serie de funciones y comandos que permiten la interacción con la API de Telegram para enviar mensajes de notificación a un grupo o chat específico.

Para poder utilizar el servicio de Telegram y enviar notificaciones, el script incluye dos parámetros esenciales que deben configurarse adecuadamente antes de su uso:

- El ”TOKEN” es un valor alfanumérico único que actúa como clave de acceso para el bot de Telegram. Este token permite la autenticación del bot para interactuar con la API de Telegram y enviar mensajes.
- El ”ID” del chat o grupo define el destino de las notificaciones. Debe proporcionarse el identificador numérico del chat o grupo de Telegram al que se desean enviar las notificaciones. Es importante tener en cuenta que, si el bot va a enviar notificaciones a un grupo, debe tener permisos de administrador para poder realizar esta acción.

El script incluye una función llamada *enviarMensajeAlBot()* que se encarga de enviar los mensajes a través de la API de Telegram. Esta función toma como parámetro el mensaje que se desea enviar y utiliza el comando **curl** para realizar una solicitud HTTP POST a la API de Telegram. El mensaje se enviará al chat o grupo definido previamente por medio del ”TOKEN” y ”ID”.

Además, el script acepta diferentes argumentos. Estos argumentos permiten especificar el tipo de notificación que se enviará, como el inicio o detención de Fail2ban, o cuando una dirección IP ha sido bloqueada o desbloqueada.

El script se encarga de analizar los argumentos pasados utilizando el comando **getopts**. Dependiendo de los argumentos proporcionados, el script selecciona el mensaje apropiado y lo envía a través de la función *enviarMensajeAlBot()*.

En caso de que no se pase ningún argumento o se pase un argumento no válido, el script mostrará un mensaje de uso para guiar al usuario sobre cómo utilizarlo correctamente. Además, finalizará la ejecución con un código de salida de error para evitar un uso incorrecto.

```
1
2       #!/bin/bash
3
4       #TOKEN
5       TOKEN='xxxxxxxxx:AAH9_Hjw-4P7xxxxxxxxxxLbEH0Ywx3pPUo'
6
7       # ID DEL CHAT O GRUPO
8       ID='-98xxxxxx' #ID DE MI GRUPO DE TELEGRAM O DEL CHAT
9       #para que el bot funcione en grupos debe tener permisos de administrador
10
11      function enviarMensajeAlBot() {
12
13        #$0 -> REPRESENTA EL NOMBRE DEL ARCHIVO
14
15        #$1 -> REPRESENTA EL PRIMER ARGUMENTO QUE SE PASA
16
17        #$2 -> REPRESENTA EL SEGUNDO ARGUMENTO QUE SE PASA
18
19        #${3,4,5,6,7,8,9} -> REPRESENTA ARGUMENTOS
20
21        message=$1
22
23        #ENVIA UN MENSAJE
```

```
24        curl -s -X POST https://api.telegram.org/bot${TOKEN}/sendMessage -d text="${message}"
    -d chat_id=${ID} > /dev/null 2>&1
25      }
26
27      #COMPARA LA CANTIDAD DE ARGUMENTOS PASADOS AL SCRIPT
28      # 0 -> no se pasó ningún argumento
29      #imprime en pantalla los parámetros que se deben pasar
30
31      if [ $# -eq 0 ]; then
32
33      #-a -> start o stop
34      #-b -> IP baneada
35      #-u -> IP desbaneada
36      #
37      echo "Usage $0 -a ( start || stop ) || -b \$IP || -u \$IP"
38
39      #FINALIZA EL SCRIPT
40      exit 1;
41      fi
42
43
44      #PARAMETROS ACEPTADOS
45      #-a, -n , -b , -u
46
47      while getopts "a:n:b:u:" opcion; do
48
49      case "$opcion" in
50      a)
51      accion=$OPTARG
52      ;;
53      n)
54      nombre_jaula=$OPTARG
55      ;;
56      b)
57      ban=y #se crea una variable con el caracter "y"
58      ip_add_ban=$OPTARG #se guarda la IP dentro de esta variable
59      ;;
60      u)
61      unban=y # se crea una variable con el caracter "y"
62      ip_add_unban=$OPTARG # la IP se guarda dentro de esta variable
63      ;;
64      \?)  #cualquier otro parámetro es inválido y termina el script
65      echo "OPCION INVALIDA. -$OPTARG"
66      exit 1
67      ;;
68      esac
69      done
70
71      # -z -> Verdadero si la longitud de la cadena es cero.
72      #si la variable string no está vacía devuelve false y ! la niega y cambia a true.
73
74      if [[ ! -z ${accion} ]]; then
75
76      case "${accion}" in
77
78      start) #si la variable "accion" contiene la cadena "start"
79
80      enviarMensajeAlBot "FAIL2BAN START"
81      ;;
82
83      stop) # si la variable "accion" contiene la cadena "stop"
84
85      enviarMensajeAlBot "FAIL2BAN STOP"
86      ;;
87
88      *) # si la variable "accion" contiene cualquier otra cadena
89
90      echo "OPCION INCORRECTA"
91
92      exit 1; #termina el script
93      ;;
94
95      esac
```

```
96
97      elif [[ ${ban} == "y" ]]; then
98
99      #se llama a la función con un solo argumento
100     #[nombre de la jaula] IP y el mensaje
101     enviarMensajeAlBot "[${nombre_jaula}] LA IP: ${ip_add_ban} FUE BANEADA"
102
103     exit 0;
104
105     elif [[ ${unban} == "y" ]]; then
106
107     #se llama a la función con un solo argumento que es nombre de la jaula, ip y mensaje
108     enviarMensajeAlBot "[${nombre_jaula}] LA IP: ${ip_add_unban} FUE DESBANEADA"
109
110     #finaliza el script correctamente
111     exit 0;
112
113     else
114
115     info
116     fi
117
118
119
```

Código 15.26: Script - send telegram notif

Action

Se procederá a crear el archivo ”telegram.conf” dentro del directorio */etc/fail2ban/action.d/* con el propósito de configurar las notificaciones a través del servicio de mensajería Telegram en Fail2ban.

El contenido del archivo ”telegram.conf” será el siguiente:

```
1
2   [Definition]
3   actionstart = /etc/fail2ban/scripts/send_telegram_notif.sh -a start
4   actionstop = /etc/fail2ban/scripts/send_telegram_notif.sh -a stop
5   actioncheck =
6   actionban = /etc/fail2ban/scripts/send_telegram_notif.sh -b <ip>
7   actionunban = /etc/fail2ban/scripts/send_telegram_notif.sh -u <ip>
8
9   [Init]
10  init = 123
11
12
13
```

Código 15.27: Action

Dentro de la configuración de Fail2ban en el archivo ”telegram.conf”, se han especificado una serie de acciones que el sistema ejecutará en respuesta a ciertos eventos relevantes para la seguridad. Estas acciones están definidas en la sección ”[Definition]” del archivo y tienen el siguiente propósito:

- **actionstart**: Es la acción que se realizará cuando Fail2ban comience. Se ejecutará el script *send_telegram_notif.sh* con el argumento ”-a start”, lo que enviará una notificación a través de Telegram indicando que Fail2ban ha iniciado.
- **actionstop**: Es la acción que se llevará a cabo cuando Fail2ban se detenga. Se ejecutará el script *send_telegram_notif.sh* con el argumento ”-a stop”, lo que enviará una notificación a través de Telegram indicando que Fail2ban se ha detenido.
- **actioncheck**: No se define ninguna acción específica para esta opción, lo que significa que no se realizará ninguna acción adicional al realizar la comprobación de los servicios de Fail2ban.
- **actionban**: Esta acción se llevará a cabo cuando una dirección IP sea bloqueada (baneada) por Fail2ban. Se ejecutará el script *send_telegram_notif.sh* con el argumento ”-b <ip>”, donde ”<ip>” representa la dirección IP bloqueada. Esto enviará una notificación a través de Telegram indicando que la IP ha sido bloqueada.

- **actionunban**: Es la acción que se llevará a cabo cuando una dirección IP sea desbloqueada (desbaneada) por Fail2ban. Se ejecutará el script "send_telegram_notif.sh" con el argumento "-u <ip>", donde "<ip>" representa la dirección IP desbloqueada. Esto enviará una notificación a través de Telegram indicando que la IP ha sido desbloqueada.
- **La sección [Init]** contiene una única línea que establece el valor "123" para la opción "init".

Una vez que el archivo "telegram.conf"se haya creado con la configuración adecuada, Fail2ban estará listo para enviar notificaciones a través de Telegram cuando ocurran eventos de seguridad relevantes, como el inicio o detención de Fail2ban, o cuando se bloquee o desbloquee una dirección IP debido a intentos de intrusión detectados. Esta integración mejorará significativamente la capacidad del administrador del sistema para supervisar y responder rápidamente a incidentes de seguridad en el servidor protegido por Fail2ban.

Jail

Para finalizar con la configurar Fail2ban, se debe crear un archivo llamado "jail.local" dentro del directorio */etc/fail2ban/*. Se recomienda crear este archivo debido a la diferencia en la gestión de los archivos de configuración de Fail2ban:

El archivo "jail.conf" es el archivo de configuración predeterminado proporcionado por Fail2ban. Contiene la configuración predeterminada y comentarios que explican cómo se puede personalizar cada opción. Este archivo se utiliza para definir todos los "jails" (jaulas) que deberían existir en el sistema. Sin embargo, se aconseja no modificar directamente este archivo. En cambio, se sugiere crear un archivo separado.

Por otro lado, el archivo "jail.local" es un archivo utilizado para realizar modificaciones personalizadas en la configuración de Fail2ban. Si existe un archivo llamado "jail.local", el programa leerá la configuración de ambos archivos, es decir, "jail.conf" y "jail.local". Sin embargo, las configuraciones en "jail.local" tienen prioridad sobre las de "jail.conf". Esto significa que cualquier cambio o adición realizada en "jail.local" no se verá afectado por actualizaciones futuras de Fail2ban, lo que garantiza que los ajustes personalizados no se sobrescriban y se mantengan a lo largo del tiempo.

```
1
2       [DEFAULT]
3       bantime.increment    = true
4       bantime.rndtime      = 30m
5       bantime.maxtime      = 60d
6       bantime.factor       = 2
7       bantime.formula      = ban.Time * math.exp(float(ban.Count+1)*banFactor)/math.exp(1*
    banFactor)
8       bantime.overalljails = true
9       maxretry = 3
10
11      [sshd]
12      enabled = true
13      action  = iptables[name=SSH, port=2222, protocol=tcp]
14            telegram
15
```

Código 15.28: Jaula ssh

En la sección "[DEFAULT]", se han definido varias opciones que afectarán el comportamiento general del sistema Fail2ban:

- **bantime.increment:** Esta opción está habilitada, lo que significa que el tiempo de bloqueo de las direcciones IP baneadas aumentará con cada intento fallido.
- **bantime.rndtime:** Está configurada en "30m", lo que indica que el tiempo de bloqueo de las direcciones IP baneadas tendrá un valor aleatorio entre 0 y 30 minutos.
- **bantime.maxtime:** Está configurada en "60d", lo que significa que el tiempo de bloqueo máximo para las direcciones IP baneadas será de 60 días.
- **bantime.factor:** Está configurada en "2", lo que determina el factor por el cual se multiplicará el tiempo de bloqueo para cada intento fallido adicional.
- **bantime.formula:** Esta opción define una fórmula matemática personalizada para calcular el tiempo de bloqueo de una dirección IP baneada, basada en el tiempo del baneo anterior y la cantidad

de intentos fallidos. La fórmula utiliza la variable "ban.Time"para representar el tiempo del baneo anterior, y "ban.Count"para la cantidad de intentos fallidos, además del valor de "bantime.factor". Esta configuración permite una personalización avanzada del tiempo de bloqueo.

- **bantime.overalljails:** Esta opción está habilitada, lo que indica que el tiempo de bloqueo será compartido entre todas las jaulas (jails) activas en el sistema.
- **maxretry:** Está configurada en "3", lo que significa que Fail2ban bloqueará una dirección IP después de 3 intentos fallidos.

A continuación, en la sección "[sshd]", se ha configurado una jaula (jail) específica para el servicio SSH:

- **enabled:** Esta opción está habilitada, lo que activa esta jaula para el servicio SSH.
- **action:** Se ha definido como "iptables[name=SSH, port=2222, protocol=tcp]", lo que indica que se utilizará la acción de iptables para bloquear la dirección IP de origen del intento de conexión SSH fallido. Además, se especifica el nombre "SSH", el puerto "2222" y el protocolo "tcp".
- **telegram:** Se menciona la opción "telegram" en esta jaula, lo que indica que también se ha configurado el archivo "telegram.conf" que permitirá enviar notificaciones a través del servicio de mensajería Telegram cuando ocurran eventos relacionados con esta jaula.

15.10.2. Controlar cambios en el sistema

En este caso, se utiliza la herramienta denominada comando md5sum para generar un hash MD5 de un archivo en el sistema. Al ejecutar este comando en un archivo específico, se realiza el cálculo de un hash único de 128 bits basado en el contenido del archivo.

El hash MD5 generado puede ser empleado para verificar la integridad del archivo y detectar cualquier modificación no autorizada o corrupción. Al comparar el hash actual con un hash previamente almacenado, es posible determinar si el archivo ha experimentado algún cambio.

Es importante destacar que, si bien el hash MD5 se utiliza ampliamente, se considera menos seguro en comparación con algoritmos más recientes debido a la posibilidad de colisiones. Las colisiones se producen cuando dos entradas diferentes generan el mismo hash. Por lo tanto, en entornos de seguridad más críticos, se recomienda el uso de algoritmos hash más robustos, como SHA-256. Estos algoritmos ofrecen una mayor resistencia a las colisiones y son más seguros en aplicaciones criptográficas y de seguridad.

Es relevante tener en cuenta que los ataques de fuerza bruta a hashes criptográficos como MD5 resultan computacionalmente costosos y, en muchos casos, impracticables debido a la longitud y complejidad de la entrada.

```
1  #! /bin/bash
2  workdir=$PWD/seguridad
3  echo "Crear Base de datos del HASH de cada archivo"
4  find /usr -type f -exec md5sum {} \; > $workdir/usr.txt
5  find /boot -type f -exec md5sum {} \; > $workdir/boot.txt
6  find /opt -type f -exec md5sum {} \; > $workdir/opt.txt
7  find /etc -type f -exec md5sum {} \; > $workdir/etc.txt
8  find /var -type f -not -path "/var/pool/*" -not -path "/var/log/*" -not -path "/var/tmp/*"
    -exec md5sum {} \; > $workdir/var.txt
9  echo "Fin..."
10
```

Código 15.29: Generar Hash

El siguiente script proporcionado está diseñado para generar una base de datos de hashes de archivos en directorios específicos, con el fin de verificar la integridad de los archivos y detectar cambios no autorizados en el sistema.

En primer lugar, el script crea una variable llamada **'workdir'** que almacena la ruta completa del directorio actual seguido de *'/seguridad'*. Luego, mediante el uso del comando **'find**, se realizan búsquedas en varios directorios clave, como **'/usr'**, **'/boot'**, **'/opt'**, **'/etc'** y **'/var'**.

Cada búsqueda se realiza en los archivos regulares del directorio correspondiente, y para cada archivo encontrado, se calcula el hash MD5 utilizando el comando "md5sum". Los resultados de cada búsqueda

se redirigen a archivos de texto específicos en el directorio ”seguridad” utilizando la variable ”workdir” previamente definida.

En resumen, este script demuestra una práctica de seguridad recomendada al generar y almacenar hashes de archivos en directorios importantes. Al comparar los hashes generados en un momento posterior con los hashes almacenados previamente, se puede detectar cualquier modificación o corrupción en los archivos del sistema.

Comparar Hash

Si es necesario validar la integridad de los archivos del sistema, se puede utilizar el siguiente script en bash:

```
#! /bin/bash
#crear directorio llamado "seguridad"
workdir=$PWD/seguridad
echo "Crear Base de datos HASH "
echo ""
find /usr -type f -exec md5sum {} \; > $workdir/usr.tmp
find /boot -type f -exec md5sum {} \; > $workdir/boot.tmp
find /opt -type f -exec md5sum {} \; > $workdir/opt.tmp
find /etc -type f -exec md5sum {} \; > $workdir/etc.tmp
find /var -type f -not -path "/var/pool/*" -not -path "/var/log/*" \
-not -path "/var/tmp/*" -exec md5sum {} \; > $workdir/var.tmp
echo ""
echo "Diferencias..."
diff $workdir/usr.txt $workdir/usr.tmp
diff $workdir/boot.txt $workdir/boot.tmp
diff $workdir/opt.txt $workdir/opt.tmp
diff $workdir/etc.txt $workdir/etc.tmp
diff $workdir/var.txt $workdir/var.tmp
echo ""
echo "Limpiar"
rm -f $workdir/usr.tmp $workdir/boot.tmp $workdir/opt.tmp \
$workdir/etc.tmp $workdir/etc.tmp $workdir/var.tmp
```

Código 15.30: Comparar Hash

15.11. Snapshot

Una vez que se haya completado la configuración del servidor de virtualización, se procederá a crear un snapshot que posibilitará obtener una imagen congelada del volumen lógico ”Raiz” en un punto determinado del tiempo. Este snapshot será utilizado con el propósito de recuperación en caso de que ocurra una instalación incorrecta o se detecte algún cambio indeseado en el sistema. La creación del snapshot permitirá revertir el estado del volumen a una versión previa, proporcionando una forma confiable de restaurar el sistema a un estado conocido y estable en caso de ser necesario. Esta funcionalidad resulta fundamental para mantener la integridad y la disponibilidad de los datos en el entorno de virtualización.

Para crear un snapshot de la partición raíz se deben seguir los siguientes pasos:

- **Paso 1:** Verificar el espacio disponible en el grupo de volúmenes LVM Antes de crear el snapshot, es importante asegurarse de que haya suficiente espacio libre en el grupo de volúmenes para alojar la nueva instantánea.

```
vgdisplay
```

Código 15.31: vgdisplay

- **Paso 2:** Para crear el snapshot, se utilizará el siguiente comando.

```
lvcreate -L20G -s -n snapshot-raiz /dev/ema/ema-raiz
```

```
2
```

Código 15.32: snapshot

15.12. Maquinas virtuales

Para la creación de máquinas virtuales, se utilizará la aplicación de interfaz gráfica llamada ”virt-manager”. Esta herramienta se basa en la biblioteca de administración de virtualización libvirt y ofrece una forma intuitiva de crear y administrar máquinas virtuales.

Virt-manager proporciona una interfaz gráfica fácil de usar que permite configurar y personalizar diferentes aspectos de las máquinas virtuales, como la asignación de recursos, la configuración de la red y la instalación del sistema operativo invitado.

Una característica destacada de virt-manager es la capacidad de utilizar el protocolo SSH con el parámetro -X para establecer conexiones gráficas remotas. Esto significa que se puede tunelizar el tráfico X11 a través de la conexión SSH, lo que permite acceder y administrar las máquinas virtuales de forma remota a través de una interfaz gráfica.

Al utilizar el protocolo SSH con el parámetro -X, se establece una conexión segura entre el cliente y el servidor, y se habilita el reenvío de las aplicaciones gráficas de virt-manager a través de la conexión SSH. Esto permite que la interfaz gráfica de virt-manager se ejecute en el servidor y se muestre en el cliente remoto, lo que facilita la administración de las máquinas virtuales de manera gráfica incluso cuando se trabaja de forma remota.

```
1
2   ssh root@192.168.1.222 -X
3
4   virt-manager
5
6
```

Código 15.33: ssh

Para cada equipo individual, se debe realizar la configuración de la interfaz de red de la misma manera que se hizo en el servidor. Esto implica seguir los pasos adecuados para establecer las direcciones IP, la configuración de la interfaz y cualquier otra configuración necesaria.

Cada equipo deberá tener una interfaz de red física conectada a la red correspondiente. Se deberá editar el archivo de configuración de red apropiado, que puede variar según el sistema operativo utilizado (como /etc/network/interfaces en sistemas basados en Debian).

Dentro del archivo de configuración de red, se deben agregar las líneas necesarias para establecer la dirección IP, la máscara de subred, la puerta de enlace y cualquier otra configuración requerida.

A continuación se presenta un ejemplo de una configuración correcta de interfaces de red:

```
1
2   #! /bin/bash
3   ip link add link enp1s0 name dmz type vlan id 100 #creo la vlan
4   ip addr add 192.168.100.2/24 brd 192.168.100.255 dev dmz #configuro la ip
5   ip link set dmz up #activo la interfaz
6   ip route add default via 192.168.100.1 #nueva ruta de ruteo
7   echo "nameserver 1.1.1.1" > /etc/resolv.conf # dns
8   ip addr del 192.168.1.124/24 dev enp1s0 #elimino la ip estatica anterio
9
10
```

Código 15.34: Configurar Interfaz

15.12.1. Configuración

Ahora, para finalizar con la configuración de QEMU/KVM, se debe abrir Virt-manager y comenzar el proceso de creación del ”silo de almacenamiento” y agregar el directorio de las imágenes ISO y las máquinas virtuales.

Luego, es necesario dirigirse a la sección "Detalles" y "Almacenamiento" dentro de Virt-manager. Para crear un "pool de almacenamiento" destinado a las imágenes ISO, se debe hacer clic en el botón "Agregar pool" o seleccionar el icono "+" en la pestaña "Almacenamiento". En el asistente para agregar el nuevo pool de almacenamiento, se selecciona el "tipo de pool": "dir: de sistema de archivos".

A continuación, se especifica la ubicación del directorio donde se almacenarán las imágenes ISO, en este caso, */mnt/ISOS*. Se proporciona un nombre para el pool de almacenamiento, como "Imágenes ISO", y se seleccionan las opciones de acceso según se necesite, ya sea lectura/escritura o solo lectura. Luego, se completa el proceso para crear el pool de almacenamiento.

Para agregar el directorio donde se almacenarán las máquinas virtuales, se sigue el mismo procedimiento que para las imágenes ISO. Se crea un nuevo "pool de almacenamiento" seleccionando el "tipo de pool": "dir: de sistema de archivos". A continuación, se especifica la ubicación del directorio deseado para las máquinas virtuales, en este caso, */mnt/vm*. También se asigna un nombre descriptivo al pool, como "Máquinas Virtuales", y se eligen las opciones de acceso adecuadas.

15.12.2. Servidor Docker

¿Que es Docker?

Docker es una plataforma de código abierto que permite a los desarrolladores crear, desplegar y ejecutar aplicaciones dentro de contenedores. Los contenedores son entornos de software livianos y portables que incluyen todo lo necesario para que una aplicación se ejecute, como código, bibliotecas y dependencias. Docker ofrece una solución para la virtualización a nivel de sistema operativo, lo que permite a las aplicaciones ejecutarse de manera consistente en cualquier entorno, ya sea en el desarrollo, pruebas o producción.

¿Qué es Docker Compose?

Docker Compose es una herramienta que simplifica la administración de aplicaciones multi-contenedor en Docker. Permite definir y gestionar la configuración de varios contenedores dentro de una aplicación, incluyendo sus relaciones, redes y volúmenes necesarios. Todo esto se logra a través de un archivo YAML llamado "docker-compose.yml".

Maquina Virtual

Una vez conectado al hipervisor QEMU/KVM, se debe acceder a la opción "Crear una nueva máquina virtual" a través del botón correspondiente o seleccionando "Archivo" >"Nueva máquina virtual" en la barra de menú superior.

A continuación, se abrirá un asistente para crear una nueva máquina virtual y se deben seguir los siguientes pasos:

- **Seleccionar la fuente de instalación:** La máquina virtual se puede instalar desde un archivo ISO, una imagen de disco existente o una red de instalación. En esta etapa, se debe elegir la opción que corresponda y proporcionar la ruta del archivo o imagen de disco.
- **Especificar el sistema operativo:** Se debe elegir el tipo de sistema operativo y su versión. Esto configurará automáticamente los recursos y ajustes recomendados para la máquina virtual.
- **Asignar recursos:** En este paso, se deben definir la cantidad de memoria RAM, el número de núcleos de CPU y el tamaño del disco duro para la máquina virtual.
- **Personalizar la configuración:** Aquí, es posible realizar ajustes adicionales, como agregar dispositivos de hardware, configurar la red, entre otros.
- **Finalizar la creación:** Aqui se debe revisar cuidadosamente la configuración y, si todo está correcto, completar el proceso para crear la máquina virtual.

Si es necesario, se procederá a la instalación del sistema operativo seleccionado. Si se eligió una fuente de instalación desde un archivo ISO o una red de instalación, Virt-manager iniciará el proceso de instalación del sistema operativo, donde se deberán seguir las instrucciones para completar dicho proceso.

Una vez creada la máquina virtual, se podrá iniciarla haciendo clic derecho en la máquina virtual y seleccionando la opción "Iniciar". A través de Virt-manager, también se podrán gestionar las máquinas virtuales en ejecución, detenerlas, pausarlas o realizar otras acciones según las necesidades específicas de configuración y uso.

Instalación

Las siguientes líneas se utilizan para configurar Docker en un sistema basado en Debian mediante la línea de comandos:

```
#! /bin/bash

sudo apt-get update
sudo apt-get install ca-certificates curl gnupg

sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg

echo \textbackslash\{\}
"deb [arch="\$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/debian \textbackslash\{\}
"\$(. /etc/os-release \&\& echo "\$VERSION\_CODENAME")" stable" | \textbackslash\{\}
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

DOCKER\_CONFIG=\$\{DOCKER\_CONFIG:-\$HOME/.docker\}
mkdir -p \$DOCKER\_CONFIG/cli-plugins
curl -SL https://github.com/docker/compose/releases/download/v2.20.0/docker-compose-linux-x86\_64 -o \$DOCKER\_CONFIG/cli-plugins/docker-compose

chmod +x \$DOCKER\_CONFIG/cli-plugins/docker-compose

docker compose version


```

Código 15.35: docker

- Actualiza la lista de paquetes disponibles en los repositorios de apt:

```
sudo apt update

```

Código 15.36: udpate

- Instala los paquetes necesarios para gestionar certificados, realizar solicitudes web y utilizar claves de seguridad:

```
sudo apt install ca-certificates curl gnupg

```

Código 15.37: install

- Crea el directorio /etc/apt/keyrings con permisos de lectura, escritura y ejecució:

```
sudo install -m 0755 -d /etc/apt/keyrings

```

Código 15.38: install

- Descarga la clave GPG de Docker y la almacena en /etc/apt/keyrings/docker.gpg:

```
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg

```

Código 15.39: gpg

- Asigna permisos de lectura a la clave GPG para que sea accesible por otros usuarios:

```
1     sudo chmod a+r /etc/apt/keyrings/docker.gpg
2
```

Código 15.40: permisos

- Añade una entrada al archivo /etc/apt/sources.list.d/docker.list con información sobre el repositorio de Docker para la versión de Debian instalada:

```
1     echo \ \ "deb [arch=\"\$(dpkg --print-architecture)\" signed-by=/etc/apt/keyrings/
    docker.gpg] https://download.docker.com/linux/debian \ \ \"\$(. /etc/os-release \&\&
     echo \"\$VERSION_CODENAME\")\" stable" | sudo tee /etc/apt/sources.list.d/docker.
    list > /dev/null
2
```

Código 15.41: repositorio

- Instala los paquetes de Docker y sus complementos:

```
1     sudo apt install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-
    compose-plugin
2
```

Código 15.42: docker compose

- Configura el directorio donde se almacenarán los archivos de configuración de Docker, en este caso, se establece el valor de la variable DOCKER_CONFIG como el directorio $HOME/.docker si no se ha definido previamente:

```
1     DOCKER_CONFIG=${DOCKER_CONFIG:-$HOME/.docker}
2
```

Código 15.43: variable

- Crea el directorio de complementos de la línea de comandos de Docker:

```
1     mkdir -p $DOCKER_CONFIG/cli-plugins
2
```

Código 15.44: directorio

- Descarga la última versión de Docker Compose y la almacena en el directorio de complementos de la línea de comandos de Docker:

```
1     curl -SL https://github.com/docker/compose/releases/download/v2.20.0/docker-compose
    -linux-x86_64 -o $DOCKER_CONFIG/cli-plugins/docker-compose
2
```

Código 15.45: descargar

- Asigna permisos de ejecución al archivo de Docker Compose:

```
1     chmod +x $DOCKER_CONFIG/cli-plugins/docker-compose
2
```

Código 15.46: permisos de ejecución

- Verifica la versión de Docker Compose instalada:

```
1     docker compose version
2
```

Código 15.47: version

Contenedor Emby

En el Aula virtual, uno de los problemas identificados es la velocidad de conexión a Internet, limitada a 2MB, lo que dificulta la reproducción adecuada de Vídeos Tutoriales y otro contenido multimedia recomendado por los docentes.

Para abordar este desafío, se ha implementado una solución mediante el uso de Emby. Ahora, el contenido multimedia se descarga previamente y se almacena de forma segura en un lugar controlable dentro del Servidor Local. A través de Emby y su capacidad de realizar streaming, este contenido multimedia se comparte eficientemente con todos los usuarios de la red LAN del Aula virtual.

Esta solución ha permitido optimizar la reproducción de los Vídeos Tutoriales y el contenido multimedia recomendado, ya que ahora los usuarios pueden acceder a estos recursos localmente, sin depender de la velocidad limitada de Internet. Además, al centralizar y controlar el almacenamiento de los archivos multimedia en el Servidor Local, se garantiza un acceso rápido y confiable para los estudiantes y docentes en el entorno educativo.

¿Que es Emby?

Emby es una plataforma de servidor de medios de comunicación de código abierto y gratuita que permite organizar, gestionar y transmitir contenido multimedia de manera eficiente. Similar a otras soluciones populares como Plex o Kodi, Emby está diseñado para brindar una experiencia completa de entretenimiento digital a los usuarios.

Basado en un servidor multimedia centralizado, Emby se instala en servidores, dispositivos de almacenamiento conectado a la red (NAS) o incluso en máquinas virtuales. Una vez configurado, los usuarios pueden acceder y disfrutar de sus bibliotecas de medios, que incluyen películas, programas de televisión, música, fotos y videos, desde una amplia variedad de dispositivos compatibles, como computadoras, teléfonos inteligentes, tabletas, televisores inteligentes y consolas de juegos.

La interfaz de usuario de Emby es elegante y de fácil navegación, lo que permite una experiencia intuitiva al explorar y buscar contenido multimedia. Además de las funcionalidades básicas de administración de medios, Emby ofrece características adicionales, como la capacidad de descargar y sincronizar contenido para visualización sin conexión, control de usuarios con diferentes niveles de acceso y transmisión remota fuera de la red local.

Emby se destaca por su flexibilidad y personalización, ya que admite el uso de complementos y extensiones de terceros que amplían sus funcionalidades y características. La integración de Emby con tecnologías de contenedores, como Docker y Docker Compose, facilita su implementación y administración en diferentes entornos y plataformas.

En la configuración proporcionada de Docker Compose, se define un servicio llamado "emby". Este servicio utiliza la imagen de Docker de Emby Server para ejecutar una aplicación de servidor de medios Emby.

```
    services:
    emby:
    image: emby/embyserver
    container_name: emby
    restart: unless-stopped
    ports:
    - 8096:8096
    environment:
    - UID=1000 # The UID to run emby as (default: 2)
    - GID=100 # The GID to run emby as (default 2)
    - GIDLIST=100 # A comma-separated list of additional GIDs to run emby as (default: 2)
    volumes:
    - ./config:/config
    - ./data:/data
    - ./media:/media


```

Código 15.48: docker compose

El servicio de Emby tiene los siguientes ajustes:

- **Imagen:** La imagen de Docker utilizada para el servicio de Emby es "emby/embyserver". Esta imagen contiene todos los componentes y configuraciones necesarios para ejecutar Emby Server.
- **Nombre del contenedor:** El nombre del contenedor que ejecuta el servicio de Emby se establece como "emby". Esto permite hacer referencia y gestionar el contenedor de manera sencilla.
- **Política de reinicio:** La política de reinicio se configura como "unless-stopped", lo que significa que el contenedor se reiniciará automáticamente a menos que sea detenido explícitamente por el usuario.
- **Puertos:** El servicio expone el puerto 8096 del contenedor y lo mapea al puerto 8096 de la máquina anfitriona. Esto permite acceder al servidor Emby a través del puerto 8096 de la máquina anfitriona.
- **Variables de entorno:** Se establecen varias variables de entorno para configurar el ID de usuario y de grupo del servidor Emby. Las variables son:
  - **UID:** El ID de usuario (UID) con el que se ejecutará Emby, establecido por defecto en 1000.
  - **GID:** El ID de grupo (GID) con el que se ejecutará Emby, establecido por defecto en 100.
  - **GIDLIST:** Una lista separada por comas de IDs de grupo adicionales (GIDs) con los que se ejecutará Emby, establecido por defecto en 100.
  - **Volúmenes:** Se montan tres directorios de la máquina anfitriona como volúmenes dentro del contenedor para persistir los cambios de configuración y los datos:
  - **./config:/config:** Esto monta el directorio "./config" de la máquina anfitriona en el directorio "/config" dentro del contenedor para almacenar los archivos de configuración de Emby.
  - **./data:/data:** Esto monta el directorio "./data" de la máquina anfitriona en el directorio "/data" dentro del contenedor para almacenar los archivos de datos de Emby.
  - **./media:/media:** Esto monta el directorio "./media" de la máquina anfitriona en el directorio "/media" dentro del contenedor, lo que permite a Emby acceder a los archivos de medios almacenados en la máquina anfitriona.

Configuración de red

Una vez finalizada la instalación y configuración de Docker, se procederá a configurar la interfaz de red y se colocará en la red DMZ del servidor de virtualización mediante el siguiente script:

```
#! /bin/bash
ip link add link enp1s0 name dmz type vlan id 100 #creo la vlan
ip addr add 192.168.100.2/24 brd 192.168.100.255 dev dmz #configuro la ip
ip link set dmz up #activo la interfaz
ip route add default via 192.168.100.1 #nueva ruta de ruteo
echo "nameserver 1.1.1.1" > /etc/resolv.conf # dns
ip addr del 192.168.1.124/24 dev enp1s0 #elimino la ip estatica anterio

```

Código 15.49: docker

Este script realizará los pasos necesarios para configurar la interfaz de red "dmz" con la dirección IP "192.168.100.2" en la VLAN 100, conectándola a la red DMZ del servidor de virtualización. Además, establecerá la puerta de enlace predeterminada a través de "192.168.100.1" y configurará el servidor DNS con la dirección "1.1.1.1". Por último, eliminará la dirección IP estática "192.168.1.124" que estaba configurada en la interfaz "enp1s0" anteriormente.

Configuración NFS

Para montar el directorio compartido a través de NFS en el servidor Emby,se deben seguir los siguientes pasos:

- Instalar nfs-common

```
        sudo apt install nfs-common

```

Código 15.50: nfs-common

- Montar el directorio compartido utilizando el comando ”mount” con la opción ”-t nfs” y la dirección IP del servidor NFS seguido del directorio compartido:

```
1         sudo mount -t nfs 192.168.100.1:/ruta/directorio /ruta/media
2
```

Código 15.51: mount

- Para que el directorio compartido se monte automáticamente cada vez que el servidor Emby se inicie, se debe agregar la línea correspondiente al archivo ”/etc/fstab”.

```
1       192.168.100.1:/ruta/directorio /ruta/media nfs defaults 0 0
2
```

Código 15.52: fstab

Personalizacion

La configuración de la plataforma Emby dependerá de las necesidades de cada establecimiento. Cada institución o administrador podrá personalizar y ajustar la configuración de Emby de acuerdo con los requerimientos y preferencias específicas de su entorno y usuarios. Emby ofrece una amplia gama de opciones y ajustes que permiten una adaptación versátil de la plataforma para ofrecer una experiencia de visualización multimedia satisfactoria y a medida.

15.12.3. Servidor Nextcloud

¿Que es Nextcloud?

Nextcloud es una plataforma de almacenamiento en la nube y colaboración de código abierto que permite a los usuarios almacenar, sincronizar y compartir archivos y datos en un servidor propio o en un proveedor de alojamiento. Es una alternativa popular a servicios comerciales de almacenamiento en la nube como Dropbox o Google Drive, pero con la diferencia de que Nextcloud permite a los usuarios tener un control completo sobre sus datos y la infraestructura utilizada.

La característica principal de Nextcloud es su capacidad para acceder y gestionar archivos desde cualquier lugar y dispositivo con conexión a Internet. Los usuarios pueden utilizar la interfaz web de Nextcloud o las aplicaciones móviles y de escritorio para acceder a sus datos, lo que facilita la colaboración y el intercambio de archivos con otros usuarios.

Además del almacenamiento y sincronización de archivos, Nextcloud ofrece una amplia gama de aplicaciones y extensiones que agregan funcionalidades adicionales. Estas incluyen la capacidad de compartir calendarios y contactos, editar documentos en línea de forma colaborativa, realizar videoconferencias y chatear con otros usuarios, administrar tareas y notas, entre otras opciones.

Nextcloud también se destaca por sus características de seguridad y privacidad. Proporciona opciones de cifrado de extremo a extremo y autenticación de dos factores para proteger los datos de los usuarios y garantizar su privacidad.

Al ser de código abierto, Nextcloud se beneficia de una comunidad activa de desarrolladores y usuarios que contribuyen con mejoras y nuevas características constantemente. Esto ha llevado a un ecosistema dinámico y en constante evolución, con una amplia variedad de complementos y aplicaciones de terceros disponibles para personalizar la experiencia de Nextcloud según las necesidades de cada usuario.

En resumen, Nextcloud es una solución de almacenamiento en la nube de código abierto que brinda a los usuarios un mayor control sobre sus datos, así como una serie de herramientas de colaboración para mejorar la productividad y la gestión de archivos en un entorno seguro y personalizable.

Maquina Virtual

Una vez conectado al hipervisor QEMU/KVM, se debe acceder a la opción ”Crear una nueva máquina virtual” a través del botón correspondiente o seleccionando ”Archivo” >”Nueva máquina virtual” en la barra de menú superior.

A continuación, se abrirá un asistente para crear una nueva máquina virtual y se deben seguir los siguientes pasos:

- **Seleccionar la fuente de instalación:** La máquina virtual se puede instalar desde un archivo ISO, una imagen de disco existente o una red de instalación. En esta etapa, se debe elegir la opción que corresponda y proporcionar la ruta del archivo o imagen de disco.
- **Especificar el sistema operativo:** Se debe elegir el tipo de sistema operativo y su versión. Esto configurará automáticamente los recursos y ajustes recomendados para la máquina virtual.
- **Asignar recursos:** En este paso, se deben definir la cantidad de memoria RAM, el número de núcleos de CPU y el tamaño del disco duro para la máquina virtual.
- **Personalizar la configuración:** Aquí, es posible realizar ajustes adicionales, como agregar dispositivos de hardware, configurar la red, entre otros.
- **Finalizar la creación:** Aqui se debe revisar cuidadosamente la configuración y, si todo está correcto, completar el proceso para crear la máquina virtual.

Si es necesario, se procederá a la instalación del sistema operativo seleccionado. Si se eligió una fuente de instalación desde un archivo ISO o una red de instalación, Virt-manager iniciará el proceso de instalación del sistema operativo, donde se deberán seguir las instrucciones para completar dicho proceso.

Una vez creada la máquina virtual, se podrá iniciarla haciendo clic derecho en la máquina virtual y seleccionando la opción ”Iniciar”. A través de Virt-manager, también se podrán gestionar las máquinas virtuales en ejecución, detenerlas, pausarlas o realizar otras acciones según las necesidades específicas de configuración y uso.

Instalación

1. Descargar Nextcloud
   - Se descarga Nextcloud siguiendo los siguientes pasos:

```
wget https://download.nextcloud.com/server/releases/latest.zip
```

Código 15.53: NextCloud

   - Descompresión del archivo zip:

```
unzip lates.zip
```

Código 15.54: Descomprimir

   - Movimiento del directorio Nextcloud a /opt:

```
sudo mv nextcloud /opt
```

Código 15.55: Mover contenido

   - Configuración de permisos y propiedad del directorio:

```
sudo chmod o-rwx /opt/nextcloud
sudo chown -R root:root /opt
```

Código 15.56: Dueño

   - Concesión de permisos al usuario ”www-data”:

```
sudo setfacl -R -m u:www-data:rwx /opt/nextcloud
```

Código 15.57: Permiso

2. Antes de iniciar la instalación de Nextcloud, es necesario abordar algunas dependencias mencionadas en su documentación. Se procede a instalar y configurar un servidor web con PHP y una base de datos MariaDB para satisfacer estos requisitos.

   En primer lugar, se lleva a cabo la instalación de las dependencias necesarias para el correcto funcionamiento de Nextcloud, según las indicaciones proporcionadas en la documentación oficial.

Luego, se procede a la configuración del servidor web y PHP para habilitar las extensiones y ajustes requeridos por Nextcloud. Además, se instala y configura una base de datos MariaDB para que Nextcloud pueda almacenar y administrar sus datos.

Con estas tareas completadas, se habrán resuelto las dependencias y se habrá preparado el entorno necesario para la instalación de Nextcloud.

```
sudo apt install php php-mysql php-mbstring php-json php7.4-common php7.4-xml php-zip
  php-gd curl php-curl php-pear php7.4-opcache php-intl mariadb-server

```

Código 15.58: Instalación de dependecias

- **php:** Es el paquete que instala el lenguaje de programación PHP, que es ampliamente utilizado en el desarrollo web.
- **php-mysql:** Este paquete proporciona soporte para la conexión a bases de datos MySQL/-MariaDB desde PHP.
- **php-mbstring:** Proporciona funciones para manipular cadenas de caracteres multibyte en PHP.
- **php-json:** Habilita el manejo de datos en formato JSON en PHP.
- **php7.4-common:** Contiene archivos compartidos y configuraciones comunes para PHP versión 7.4.
- **php7.4-xml:** Proporciona soporte para la manipulación de documentos XML en PHP.
- **php-zip:** Agrega soporte para la compresión y descompresión de archivos ZIP en PHP.
- **php-gd:** Habilita la manipulación de imágenes y la generación de gráficos en PHP mediante la biblioteca GD.
- **curl:** Es una herramienta y biblioteca para transferir datos con sintaxis URL. También es una dependencia común para muchas aplicaciones web y bibliotecas de PHP.
- **php-curl:** Extensión de PHP que permite realizar solicitudes HTTP y otras operaciones a través de cURL.
- **php-pear:** Gestor de paquetes para PHP, que permite instalar y administrar bibliotecas y extensiones.
- **php7.4-opcache:** Módulo de caché para PHP 7.4, que mejora el rendimiento y la velocidad de ejecución de los scripts PHP.
- **php-intl:** Proporciona funciones para la internacionalización y localización de aplicaciones PHP.
- **mariadb-server:** Paquete para instalar el servidor de base de datos MariaDB, una bifurcación de MySQL.

3. Una vez que la instalación de los programas ha finalizado, es necesario realizar algunas modificaciones en los valores predeterminados de PHP. Estas modificaciones son necesarias para asegurar que PHP esté correctamente configurado y pueda funcionar de manera óptima.

```
        #hacer backup del archivo original
        sudo cp /etc/php/7.4/cli/php.ini /etc/php/7.4/cli/php.ini.bk
        sudo nvim /etc/php/7.4/cli/php.ini


```

Código 15.59: Editar php

Dentro del documento se deben configurar los siguientes valores:

```
        post_max_size = 512M
         upload_max_lesize = 512M
        memory_limit = 512M

```

Código 15.60: Valores

- **post_max_size = 512M:** Se ajustó el límite máximo del tamaño de datos que pueden ser enviados mediante una petición POST al servidor a 512 megabytes (MB). Esto es relevante para aplicaciones que envían grandes cantidades de datos a través de formularios, por ejemplo.
- **upload_max_filesize = 512M:** Se modificó el tamaño máximo permitido para subir archivos al servidor a 512 megabytes (MB). Con esta configuración, las aplicaciones web podrán recibir y manejar archivos de hasta ese tamaño.
- **memory_limit = 512M:** Se aumentó el límite de memoria asignada a los scripts de PHP a 512 megabytes (MB). Esto permite que los scripts de PHP puedan utilizar una cantidad mayor de memoria durante su ejecución, lo que es útil para aplicaciones más complejas que requieren mayor capacidad de memoria.

Finalizada la configuracion se debe reiniciar apache:

```
1          systemctl restart apache2
2
```

Código 15.61: Reiniciar apache

4. MariaDB

Crear usuario y base de datos para Nextcloud

- Acceder a la consola de MySQL/MariaDB:

```
1        mysql -u root -p
2
```

Código 15.62: Mysql

- Crear la base de datos para Nextcloud:

```
1
2        CREATE DATABASE nextclouddb;
3
```

Código 15.63: Base de datos

- Crear un usuario para Nextcloud y asignarle una contraseña:

```
1        CREATE USER 'nextcloud'@localhost IDENTIFIED BY '123456';
2
```

Código 15.64: Usuario

- Conceder todos los privilegios al usuario sobre la base de datos creada:

```
1          GRANT ALL privileges ON nextclouddb.* TO 'nextcloud'@localhost;
2
```

Código 15.65: Otorgar Permisos

- Actualizar los privilegios para que los cambios tnegan efecto:

```
1          FLUSH PRIVILEGES;
2
```

Código 15.66: Actualizar privilegios

- Salir de la consola de MySQL/MariaDB:

```
1          EXIT;
2
```

Código 15.67: Salir

Con estos pasos, se habrá creado un usuario llamado ”nextcloud” con la contraseña ”123456” y una base de datos llamada ”nextclouddb”, los cuales podrán ser utilizados por Nextcloud para almacenar y gestionar sus datos.

5. VirtualHost "VirtualHost" (anfitrión virtual) es una característica que permite configurar múltiples sitios web para que puedan compartir el mismo servidor web físico.

   Abrir un editor de texto en el servidor:

```
sudo vim /etc/apache2/sites-available/nextcloud.conf

```

Código 15.68: VirtualHost

   Este VirtualHost está configurado para mostrar el sitio Nextcloud en el dominio "aulavirtual.duckdns.org" desde el directorio */opt/nextcloud/*, y permite el acceso completo a los archivos en ese directorio mientras deshabilita la compatibilidad con WebDAV para este sitio. Además, se habilita la posibilidad de usar archivos .htaccess para anular la configuración en el directorio.

```

<VirtualHost *:80>
DocumentRoot /opt/nextcloud/
ServerName aulavirtual.duckdns.org
ServerAlias aulavirtual.duckdns.org

<Directory /opt/nextcloud/>
Require all granted
AllowOverride All
Options FollowSymLinks MultiViews
<IfModule mod_dav.c>
Dav off
</IfModule>
</Directory>
</VirtualHost>

```

Código 15.69: VirtualHost

   Habilitar el sitio virtual creado mediante el comando a2ensite:

```
sudo a2ensite nextcloud.conf

```

Código 15.70: a2ensite

   Reiniciar el servicio de Apache para aplicar los cambios:

```
sudo systemctl restart apache2

```

Código 15.71: Reiniciar Apache

   Con estos pasos, se habrá creado y habilitado el archivo de configuración "nextcloud.conf" en el directorio */etc/apache2/sites-available/*, lo que permitirá que Apache muestre Nextcloud desde la URL "aulavirtual.duckdns.org" en el puerto 80. Además, se han configurado las opciones necesarias para que el servidor Apache funcione correctamente con Nextcloud, incluyendo la configuración de directorios, permisos y opciones de seguimiento de enlaces simbólicos.

6. Configuración adicional:

```
nvim /etc/php/7.4/apache2/conf.d/10-opcache.ini

```

Código 15.72: Configurar Adicional

```
zend_extension=opcache
opcache.enable=1
opcache.enable_cli=1
opcache.memory_consumption=128
opcache.interned_strings_buffer=8
opcache. max_accelerated_les=10000
opcache.revalidate_freq=1
opcache.save_comments=1
opcache.huge_code_pages=1
||
```

Código 15.73: Valores

En la configuración de PHP, se realizaron las siguientes modificaciones específicas para el OPCache:

- **zend_extension=opcache:** Se habilita la extensión del OPCache en PHP, lo que permite que PHP utilice la caché de código opcodificado para mejorar el rendimiento de las ejecuciones de los scripts.
- **opcache.enable=1:** Se activa el OPCache para que esté habilitado y funcione en el entorno de PHP.
- **opcache.enabl_cli=1:** Se habilita el OPCache también para la línea de comandos (CLI) de PHP, lo que permite que los scripts PHP ejecutados en la consola se beneficien de la caché opcodificada.
- **opcache.memory_consumption=128:** Se establece la cantidad de memoria que el OPCache puede utilizar para almacenar la caché de código opcodificado. En este caso, se asignan 128 megabytes (MB) de memoria para esta función.
- **opcache.interned_strings_buffer=8:** Se especifica la cantidad de memoria en megabytes (MB) que el OPCache utiliza para almacenar las cadenas internas, lo que puede mejorar la eficiencia y reducir la duplicación de cadenas en el código PHP.
- **opcache.max_accelerated_files=10000:** Se establece el número máximo de archivos que el OPCache puede almacenar en caché. En este caso, se configura para almacenar hasta 10,000 archivos en la caché opcodificada.
- **opcache.revalidate_freq=1:** Se define la frecuencia con la que el OPCache verificará si los archivos en caché han sido modificados en disco, en segundos. En este caso, se establece para que verifique cada segundo.
- **opcache.save_comments=1:** Se indica que el OPCache debe guardar los comentarios en el código opcodificado. Esto puede ser útil para propósitos de depuración.
- **opcache.huge_code_pages=1:** Habilita el uso de páginas de código grandes (huge code pages) si están disponibles en el sistema. Esto puede mejorar el rendimiento del OPCache en ciertas configuraciones.

Con estas configuraciones en OPCache, se optimiza el rendimiento del motor PHP, almacenando en caché el código opcodificado para reducir el tiempo de ejecución de los scripts y mejorar la eficiencia general del servidor PHP. Es importante tener en cuenta que estas configuraciones pueden variar según las necesidades y características del servidor y las aplicaciones web que se ejecuten.

```
1       systemctl restart apache2
2       a2dissite 000-default.conf
3       a2ensite nextcloud.conf
4       a2enmod rewrite
5       a2enmod headers
6       a2enmod env
7       a2enmod dir
8       a2enmod mime
9
```

Código 15.74: Habilitar

7. SSL - Certbot

Certbot es una herramienta de software desarrollada por la organización sin fines de lucro Internet Security Research Group (ISRG). Su objetivo principal es facilitar y automatizar el proceso de obtención, renovación e instalación de certificados SSL/TLS para sitios web, permitiendo así el uso de conexiones seguras (HTTPS) en los servidores web.

Los certificados SSL/TLS son fundamentales para garantizar la seguridad de las comunicaciones entre los usuarios y el servidor, ya que cifran los datos transmitidos y autentican la identidad del servidor. Esto ayuda a proteger la privacidad de los datos y garantizar que la información no sea interceptada o manipulada por terceros malintencionados.

Certbot es específicamente diseñado para funcionar con el servidor web Apache y el servidor web Nginx, aunque también es compatible con otros servidores web populares. La herramienta utiliza el

protocolo de autorización automática ACME (Protocolo de Gestión de Certificados de Autoridad), que permite validar y obtener certificados automáticamente a través de desafíos bien definidos.

La principal ventaja de Certbot es su facilidad de uso y automatización. Una vez configurado correctamente, puede realizar tareas como:

Generación de una solicitud de firma de certificado (CSR). Comunicarse con una Autoridad de Certificación (CA), como Let's Encrypt, para obtener un certificado SSL/TLS gratuito y confiable. Verificar la propiedad del dominio a través de diferentes métodos de desafío (por ejemplo, desafío HTTP o desafío DNS). Instalación automática del certificado en el servidor web configurado. Programación de tareas para renovar automáticamente los certificados antes de que expiren.

- Instalación de CertBot y el complemento para Apache:

```
1      apt install certbot python3-certbot-apache
2
```

Código 15.75: CertBot

- Configuración de CertBot para Apache: Una vez instalado, se configura CertBot para funcionar con Apache.

```
1      certbot --apache
2
```

Código 15.76: CertBot

El proceso de configuración de certificados SSL/TLS para sitios web alojados en Apache utilizando CertBot incluye un asistente interactivo. CertBot, de manera automática, detectará los dominios configurados en el servidor y presentará opciones para que se seleccione qué dominios requieren certificados SSL/TLS.

Siguiendo las indicaciones del asistente, se podrá decidir si se desea redireccionar todas las solicitudes HTTP a HTTPS, garantizando así conexiones seguras. Una vez completado el proceso de configuración, CertBot se encargará de obtener y configurar los certificados SSL/TLS necesarios para los dominios seleccionados.

CertBot también habilitará automáticamente la configuración de Apache para utilizar estos certificados, permitiendo ofrecer conexiones seguras a través de HTTPS. Con esta implementación, se asegura que los sitios web alojados en el servidor se carguen de manera segura mediante HTTPS y proporcionen conexiones cifradas para salvaguardar la privacidad y seguridad de los usuarios.

Editar el archivo */opt/nextcloud/config/config.php*:

```
1      vim /opt/nextcloud/ cong/ cong.php
2
```

Código 15.77: Directorio de trabajo NextCloud

Con esta configuración, Nextcloud solo permitirá el acceso desde los dominios y direcciones IP especificados en la lista de dominios confiables. Los usuarios podrán acceder a Nextcloud desde cualquiera de estos dominios o direcciones IP sin problemas de autenticación o seguridad. Cualquier otro intento de acceso desde un dominio no listado en 'trusted_domains' será bloqueado por Nextcloud.

```
1      'trusted_domains' =>
2      array (
3      0 => 'localhost',
4      1 => 'aulavirtual.duckdns.org',
5      2 => '192.168.100.3',
6      ),
7
```

Código 15.78: trusted_domains

- **trusted_domains:** se utiliza para especificar los nombres de dominio o direcciones IP desde las cuales se permite el acceso a Nextcloud. En este caso, se han establecido tres dominios confiables:

- **localhost:** Indica que el acceso a Nextcloud desde el propio servidor, a través de la dirección de loopback 'localhost' (127.0.0.1), está permitido.
- **aulavirtual.duckdns.org:** Especifica que el acceso a Nextcloud desde el dominio 'aulavirtual.duckdns.org' está permitido. Esto permitirá que Nextcloud sea accesible desde Internet a través de este dominio.
- **192.168.1.3:** Indica que el acceso a Nextcloud desde la dirección IP local '192.168.100.3' está permitido. Esto permitirá el acceso a Nextcloud desde la red local a través de esta dirección IP.

Reiniciar servidor web:

```
systemctl restart apache2

```

Código 15.79: Reiniciar Apache

Configuración de red

```
#! /bin/bash
ip link add link enp1s0 name dmz type vlan id 100 #creo la vlan
ip addr add 192.168.100.3/24 brd 192.168.100.255 dev dmz #configuro la ip
ip link set dmz up #activo la interfaz
ip route add default via 192.168.100.1 #nueva ruta de ruteo
echo "nameserver 1.1.1.1" > /etc/resolv.conf # dns
ip addr del 192.168.1.124/24 dev enp1s0 #elimino la ip estatica anterio


```

Código 15.80: Red

Este script realizará los pasos necesarios para configurar la interfaz de red "dmz" con la dirección IP "192.168.100.3" en la VLAN 100, conectándola a la red DMZ del servidor de virtualización. Además, establecerá la puerta de enlace predeterminada a través de "192.168.100.1" y configurará el servidor DNS con la dirección "1.1.1.1". Por último, eliminará la dirección IP estática "192.168.1.124" que estaba configurada en la interfaz "enp1s0" anteriormente.

Configuración NFS

Para montar el directorio compartido a través de NFS en el servidor Nextcloud,se deben seguir los siguientes pasos:

- Instalar nfs-common

```
        sudo apt install nfs-common

```

Código 15.81: nfs-common

- Montar el directorio compartido utilizando el comando "mount" con la opción "-t nfs" y la dirección IP del servidor NFS seguido del directorio compartido:

```
    sudo mount -t nfs 192.168.100.1:/ruta/directorio /ruta/datos

```

Código 15.82: mount

- Para que el directorio compartido se monte automáticamente cada vez que el servidor Emby se inicie, se debe agregar la línea correspondiente al archivo "/etc/fstab".

```
    192.168.100.1:/ruta/directorio /ruta/datos nfs defaults 0 0

```

Código 15.83: fstab

Personalización

La configuración de la plataforma Nextcloud dependerá de las necesidades de cada establecimiento. Cada institución o administrador tendrá la capacidad de personalizar y ajustar la configuración de Nextcloud según los requerimientos y preferencias específicas de su entorno y usuarios. Nextcloud ofrece una amplia gama de opciones y ajustes que permiten una adaptación versátil de la plataforma para satisfacer las necesidades particulares de cada implementación.

Capítulo 16

Conclusión

En este plan de migración se han considerado los programas privativos que se utilizan actualmente y sus respectivas alternativas en software libre. Según los resultados obtenidos a partir de la investigación y búsqueda de alternativas libres, se puede concluir que la migración a software libre es factible si los alumnos están de acuerdo.

Cabe destacar que uno de los factores más importantes en la búsqueda de información fue la comunidad de software libre, la cual está conformada en su mayoría por usuarios finales. Estos mismos usuarios sirvieron de guía y ayuda para enfocarse en la elaboración de una estrategia que permita afrontar el problema propuesto. Con la ayuda de sus recomendaciones, se llegó a la conclusión de que se requiere una capacitación y formación adecuada para trabajar en el nuevo entorno de trabajo. En el caso de la persona responsable de la migración e implementación, asumirá el tiempo asociado al entrenamiento, capacitación, formación y soporte a los alumnos.

Como trabajo futuro, se propone la ampliación de este proyecto, la generación de una política de seguridad, una administración centralizada y una planificación más detallada del proceso de migración.

Como señala Stallman, es importante que en escuelas y universidades se utilice software libre, ya que estas instituciones educativas están decidiendo el futuro de la sociedad. Por lo tanto, no se debe aceptar que en un espacio perteneciente a la universidad se utilice y enseñe a utilizar software privativo, sabiendo que el deber de una universidad es la creación y difusión del conocimiento. Los alumnos, como parte de ella, también deben defender la libertad de las personas para compartir el conocimiento y el software.

> «Las escuelas deben enseñar a sus alumnos a ser ciudadanos de una sociedad fuerte, capaz, independiente y libre».
>
> The Free Software Foundation.

Bibliografía

[1] By Santiago Becerra Carrillo. Lista de licencias con comentarios. January.

[2] Free Software Foundation. Historia de gnu. December.

[3] Free Software Foundation. Why Educational Institutions Should Use and Teach Free Software. page 1, June.

[4] William Henry Gates III. An Open Letter to Hobbyists. page 1, February.

[5] Nextcloud GmbH. Nextcloud Server Administration Manual. pages 1–351, Jul 09.

[6] KVM. Kvm - kernel-based virtual machine. `https://www.linux-kvm.org/`.

[7] Bailón Giler Cristina Lizeth, Delgado Castro Nexar Javier, Resabala Córdova Mayra Alejandra, and Resabala Córdova Olga Isabel. Instalación y Configuración de Equipos Informáticos bajo Software Libre para la Biblioteca de la Facultad de Ciencias Informáticas de la Universidad Técnica de Manabí.

[8] Emby LLC. Emby Server for Docker. 3 days ago.

[9] Alejandro Lopez. Administración GNU/Linux 2. page x.

[10] Debian org . Guía Debian GNU/Linux de instalación. pages 1–x, January 29.

[11] QEMU. Qemu. `https://www.qemu.org/`.

[12] By Fernando Ribeiro. Cómo Instalar un Servidor Linux con Debian 10 'Buster'. pages 1–7, April.

[13] By Richard Stallman. ¿qué es el software libre? June.

[14] Celeste Weidman. Redes. page x.