

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS



**DESARROLLO DE PROTOTIPO ENFOCADO EN LA INTEGRACIÓN DE
REGIONES EN UNA INFRAESTRUCTURA DE NUBE**

PRESENTADO POR:

VANESSA CRISTINA ALFARO COREAS

GERARDO WILFREDO PERALTA PINTO

FRANCISCO ALEXANDER SANTANA MÉNDEZ

DANIEL ENRIQUE TENNANT ALVARENGA

PARA OPTAR AL TÍTULO DE:

INGENIERO DE SISTEMAS INFORMÁTICOS

CIUDAD UNIVERSITARIA, NOVIEMBRE 2023

UNIVERSIDAD DE EL SALVADOR

RECTOR:

MSC. JUAN ROSA QUINTANILLA

SECRETARIO GENERAL:

LIC. PEDRO ROSALÍO ESCOBAR CASTANEDA

FACULTAD DE INGENIERÍA Y ARQUITECTURA

DECANO:

ING. LUIS SALVADOR BARRERA MANCÍA

SECRETARIO:

ARQ. RAUL ALEXANDER FABIAM ORELLANA

ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

DIRECTOR:

ING. CÉSAR AUGUSTO GONZÁLEZ

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO DE SISTEMAS INFORMÁTICOS

Título:

**DESARROLLO DE PROTOTIPO ENFOCADO EN LA INTEGRACIÓN DE
REGIONES EN UNA INFRAESTRUCTURA DE NUBE**

Presentado por:

VANESSA CRISTINA ALFARO COREAS

GERARDO WILFREDO PERALTA PINTO

FRANCISCO ALEXANDER SANTANA MÉNDEZ

DANIEL ENRIQUE TENNANT ALVARENGA

Trabajo de Graduación Aprobado por:

Docente Asesor:

MSc. DAMIÁN MORALES

CIUDAD UNIVERSITARIA, NOVIEMBRE 2023

Trabajo de Graduación Aprobado por:

Docente Asesor:

Msc. Damián Morales

Agradecimientos

En principio, deseo expresar mi sincero agradecimiento a Dios por la constancia, paciencia y comprensión que me ha otorgado a lo largo de mi carrera. Asimismo, extendo mi agradecimiento a mis padres, quienes han sido ese pilar fundamental en mi desarrollo personal, espiritual y profesional, quienes me han brindado su apoyo ante las adversidades que se presentaron a lo largo de este trayecto.

Agradezco de manera especial a mi tía, quien ha representado un apoyo incondicional, ofreciendo sabios consejos y alentándome a avanzar y ser mejor persona en todos los aspectos.

Por último, pero no menos importante, agradezco a los demás miembros de mi familia, así como a mis amigos y compañeros, cuyo constante respaldo y motivación fueron elementos fundamentales en este camino. Donde en compañía de ellos, compartimos experiencias significativas, risas, lágrimas y valiosas lecciones que perdurarán en nuestros recuerdos.

Vanessa Cristina Alfaro Coreas

A Dios y a la Virgen de Guadalupe, a quienes encomendé mi carrera y me han permitido culminarla, por darme la fuerza y la sabiduría para superar cada reto al que me he enfrentado y por poner en mi camino a personas espectaculares que me han acompañado a lo largo de esta etapa de mi vida.

A mis papás, Salvador Peralta y Teresa de Jesús, mi más profundo agradecimiento, ya que sin su apoyo y amor incondicional, este logro no habría sido posible. Me han guiado a lo largo de la vida, enseñándome que con disciplina y esfuerzo todo es posible. Su aliento y sacrificios innumerables me han permitido culminar mi carrera. A mi hermana Alma Peralta, que ha sido mi inspiración, ya que con su ejemplo he aprendido a superar los obstáculos y buscar siempre la excelencia. A Liliana Sánchez, por inspirarme valor en momentos cruciales, recordándome que incluso en la adversidad, hay una luz que nos impulsa a seguir adelante.

Y, por último, y no menos importante, a mis compañeros de trabajo de grado y a los muchos otros que a lo largo de la carrera he tenido el gusto de conocer. Gracias por las experiencias vividas, los desafíos superados juntos, por trabajar codo a codo para lograr nuestros objetivos, sin egoísmos, entendiendo que esta experiencia es un viaje, no una competencia.

Gerardo Wilfredo Peralta Pinto

Expresar mi agradecimiento a Dios, quien junto a las personas como mi familia y amigos que me han apoyado a lo largo de este trayecto académico, han impulsado mi ánimo para lograr llegar a uno de mis sueños, ya que han sido los principales motivos de mi inspiración lo cual agradezco de todo corazón.

Agradecer profundamente a mi madre que desde mis inicios académicos hasta este momento importante en mi alma mater, ha estado a mi lado, ofreciéndome en todo momento su apoyo, su constante ánimo y dedicación hacia mí, han sido esenciales para mi desarrollo y crecimiento.

Reconocer especialmente a mis compañeros de universidad que juntos hemos compartido desafíos, noches enteras dedicadas al estudio con el propósito de aprender, los cuales nos dejaron conocimientos para contribuir al bienestar de nuestra comunidad.

Francisco Alexander Santana Méndez

Ante todo, le doy gracias a Dios por permitirme culminar esta gran aventura, sin la ayuda de Él no sería la persona que soy ahora, por la paciencia y perseverancia brindada que me ha dado para emprender éste maravilloso viaje lleno de obstáculos y sacrificios.

Agradezco a mi madre que es el motor y que me impulsa a seguir siempre adelante, la cual siempre ha estado pendiente desde mis inicios académicos hasta culminar mi carrera.

A mis abuelitos y familia que han sido un apoyo incondicional y por haberme demostrado que somos capaces de conseguir lo que anhelamos.

Además, a mis compañeros y amigos con los cuales crecí tanto personal como profesionalmente, donde cada noche de desvelo, sacrificio y esfuerzo con ellos ha valido la pena.

Daniel Enrique Tennant Alvarenga

Índice

Agradecimientos	4
1. Introducción	1
2. Definición Del Proyecto	3
2.1. Antecedentes	3
2.2. Pregunta de Investigación	5
2.3. Planteamiento del Problema	5
2.4. Objetivos	6
2.4.1. Objetivo General	6
2.4.2. Objetivos Específicos	6
2.5. Justificación	8
2.6. Alcances	9
2.7. Limitaciones	9
3. Marco Teórico	10
3.1. Virtualización	10
3.1.1. Hipervisores	10
3.1.1.1. Tipos de hipervisores	11
3.1.1.1.1. Hipervisor tipo 1 o Bare-Metal	11
3.1.1.1.2. Hipervisor tipo 2 o Alojado	12
3.2. Cloud Computing	13
3.2.1. Modelos de despliegue de servicios cloud	14
3.2.1.1. Nube Pública	14
3.2.1.2. Nube Privada	15
3.2.1.3. Nube Comunitaria	16
3.2.1.4. Nube Híbrida	17
3.2.2. Modelo de servicio cloud	17
3.2.2.1. Infraestructura como servicio (IaaS)	17
3.2.2.1.1. Características de un IaaS según Nayan Ruparelia (Ruparelia, 2015)	19
3.2.2.1.2. Componentes de un IaaS según Nayan Ruparelia (Ruparelia, 2015)	20
3.2.2.1.3. Ventajas y desventajas de un IaaS según Sandeep Bhowmik (Bhowmik, 2017)	21
3.2.3. Gestión de Acceso y Control en la Nube	23
3.2.3.1. Regiones	23
3.2.3.2. Zonas de Disponibilidad (AZ)	23
3.2.3.3. Relación entre regiones y zonas de disponibilidad	23
3.2.3.4. Importancia de las regiones en una nube comunitaria	24
3.2.3.4.1. ¿Cómo las regiones contribuyen a la colaboración en una nube comunitaria?	26
3.2.3.4.2. Escalabilidad y redundancia a través de regiones	26
3.2.3.5. Herramientas y recursos útiles para gestionar regiones	28
3.2.3.6. Ventajas estratégicas de distribuir servicios en regiones	29
3.2.3.7. Dominio	30
3.2.3.8. Proyecto	31

3.2.3.9. Usuarios	31
3.2.3.10. Roles	31
3.2.3.11. Control de Acceso Basado en Roles (RBAC)	31
3.3. Estrategias de Implementación en la Nube	32
3.3.1. Openstack	32
3.3.1.1. Componentes principales	32
3.3.1.1.1. Nova	33
3.3.1.1.2. Neutron	33
3.3.1.1.3. Keystone	33
3.3.1.1.4. Cinder	34
3.3.1.1.5. Swift	34
3.3.1.1.6. Placement	34
3.3.1.1.7. Glance	35
3.3.1.1.8. Horizon	35
3.3.2. OpenNebula	36
3.3.3. VMware vSphere	37
3.4. Almacenamiento en la nube	37
3.4.1. Almacenamiento en bloques	37
3.4.2. Almacenamiento de objetos	38
3.4.3. Almacenamiento distribuido	39
4. Metodología de Investigación	41
4.1. Método cuantitativo	41
4.2. Población y muestreo	41
4.3. Instrumentos de investigación	42
4.3.1 Encuesta	43
5. Resultados	44
5.1. Resultados cuantitativos	44
5.2. Conclusión de los resultados	49
6. Factibilidad del Proyecto	50
6.1. Factibilidad económica	53
7. Desarrollo del Prototipo	57
7.1. Herramientas utilizadas	57
7.1.1. Kernel-based Virtual Machine (KVM)	57
7.1.2. Ubuntu Server	57
7.1.3. Ceph	58
7.1.4. OpenStack	58
7.1.5. Ansible	58
7.1.6. Tailscale (VPN)	58
7.1.7. Pfsense	58
7.2. Diseño	59
7.3. Construcción	60
7.3.1. Infraestructura	60
7.3.2. Regiones	61
7.4. Pruebas	64

7.4.1. Funcionamiento de OpenStack	64
7.4.2. Funcionamiento de Ceph	70
8. Caso de Uso para el Prototipo de Región	74
9. Conclusiones	76
10. Recomendaciones	77
11. Referencias Bibliográficas	78
12. Anexos	82
Anexo 1. Encuesta	83
Anexo 2. Diseño de Bajo Nivel	86

1. Introducción

El presente trabajo aborda los antecedentes que contextualizan la investigación, se plantea la pregunta de investigación y se expone el problema que motiva este proyecto. Se definen los objetivos generales y específicos que se persiguen, y se proporciona una justificación para su realización. Se delimitan los alcances y se identifican las limitaciones que enmarcan el estudio.

En el marco teórico se exploran conceptos cruciales para la comprensión del proyecto. Se examinan temas relacionados con la virtualización, la nube y los modelos de despliegue de servicios cloud. Se profundiza en la infraestructura como servicio (IaaS) y sus componentes, ventajas y desventajas. Se explora la importancia de las regiones y zonas de disponibilidad en la infraestructura cloud, destacando su papel en una nube comunitaria. Se analizan herramientas y recursos para gestionar regiones y se resaltan las ventajas estratégicas de distribuir servicios en diferentes ubicaciones. Por último, se revisan estrategias de implementación en la nube, como OpenStack, OpenNebula y VMware vSphere, así como los conceptos de almacenamiento en bloques, almacenamiento de objetos y almacenamiento distribuido.

En la metodología de investigación se aborda el enfoque cuantitativo el cual se usa para comprender la adopción de la nube comunitaria en la educación. Se detalla el enfoque de encuestas estructuradas para recopilar datos cuantitativos de docentes en la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador. Se describe la población y el muestreo, destacando la importancia de una muestra representativa. Se discuten los instrumentos de investigación, centrándose en la encuesta por correo electrónico y sus ventajas y desventajas. También se exploran en detalle los resultados obtenidos de los instrumentos de investigación, los cuales son analizados para obtener las respectivas conclusiones.

Además, se lleva a cabo una evaluación de la factibilidad del proyecto, para determinar cuál es la mejor herramienta a utilizar en la construcción del prototipo tomando

en cuenta diversos factores. También se explica en detalle el proceso de desarrollo del prototipo, las herramientas y tecnologías utilizadas, el diseño y construcción.

Posteriormente, se llevan a cabo pruebas rigurosas del prototipo, evaluando tanto el funcionamiento de OpenStack como el de Ceph.

Un elemento crucial de este proyecto es el caso de uso práctico, donde se presenta una aplicación real del prototipo desarrollado. Se destaca cómo este puede resolver problemas concretos y satisfacer necesidades específicas en un contexto práctico y realista.

Se brindan conclusiones concisas basadas en los hallazgos de la investigación y desarrollo del prototipo, discutiendo las implicaciones de los resultados obtenidos y proporcionando observaciones finales sobre el proyecto en su totalidad. Por último, se incluyen recomendaciones hacia los implicados, identificando áreas de mejora y ofreciendo orientación para investigaciones posteriores.

2. Definición Del Proyecto

El enfoque de este proyecto se centra en la implementación de una Nube Comunitaria basada en OpenStack, con el objetivo de establecer una infraestructura robusta que permita el despliegue de regiones. Estas regiones se integran en la Nube Comunitaria, ofreciendo servicios esenciales, que incluyen almacenamiento (Storage), redes (Network) y capacidad de cómputo (Compute). Además, se realiza una implementación de infraestructura básica con los elementos mínimos requeridos, con el fin de evaluar el proceso de integración y todas las configuraciones relacionadas.

Un punto importante a considerar en este proyecto es su enfoque práctico en el cual se propone un caso de uso en el que se demuestra la importancia de las regiones, donde cada escuela de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador podría ser una región diferente dentro de la nube, y por motivos de recursos limitados se implementaron dos regiones que representan a la Escuela de Ingeniería Civil e Ingeniería de Sistemas Informáticos, con el propósito de evidenciar cómo la implementación de una nube comunitaria basada en regiones, puede contribuir en el mejoramiento de la colaboración y comunicación.

2.1. Antecedentes

La gestión del almacenamiento de datos en hospitales, universidades y farmacias ha sido un aspecto crítico en la operación de estas instituciones. Antes de la adopción generalizada de tecnologías de almacenamiento en la nube, estas organizaciones se vieron confrontadas con diversos problemas técnicos y operativos.

Un desafío clave residía en la limitación del espacio físico. Los hospitales, por ejemplo, acumulaban grandes cantidades de registros médicos, imágenes, diagnósticos y datos de pacientes. Al carecer de sistemas de almacenamiento en la nube, la única opción era invertir en servidores locales y dispositivos de almacenamiento físico. Estos requerían infraestructura dedicada, lo que no solo ocupaba valioso espacio, sino también generaba un aumento en los costos operativos asociados con la adquisición, el mantenimiento y el

consumo de energía de estos sistemas (*¿Qué Pasaría Si La Nube No Existiera? – Artículos Y Tendencias Sobre Soluciones Tecnológicas*, 2017).

La accesibilidad a los datos también planteaba un problema técnico significativo. Sin soluciones en la nube, el acceso remoto a la información era limitado, lo que dificultaba la colaboración eficiente entre profesionales de diferentes ubicaciones. La dependencia de sistemas locales también implicaba que el acceso a los datos estaba restringido por la disponibilidad de la red local, lo que afectaba la agilidad de la institución.

La seguridad y la integridad de los datos eran preocupaciones tecnológicas críticas. Gestionar medidas de seguridad efectivas y garantizar copias de seguridad regulares era una tarea complicada y que requería una inversión considerable en recursos humanos y tecnológicos. La pérdida de datos críticos debido a fallas en las copias de seguridad o vulnerabilidades de seguridad representaba una amenaza real.

Además, la escalabilidad era un aspecto técnico desafiante. A medida que las instituciones crecían o requerían más espacio de almacenamiento, la expansión de los recursos de almacenamiento local no era un proceso trivial. La adición de nuevos servidores y sistemas de almacenamiento requería planificación, inversión y tiempo.

Para abordar estos desafíos técnicos, las instituciones implementaban soluciones como sistemas de almacenamiento en red (NAS) para consolidar datos, la externalización de servicios de almacenamiento a proveedores externos para reducir costos, y la mejora de la eficiencia mediante técnicas de compresión y de duplicación de datos. Sin embargo, estas soluciones no eliminaban por completo los desafíos técnicos, y en algunos casos, introducían preocupaciones adicionales en términos de latencia de red, velocidad de acceso y compatibilidad (Ruiz, 2016).

A medida que las organizaciones buscaban el equilibrio entre los beneficios de las nubes públicas y privadas, emergió la nube comunitaria. Estas nubes surgieron para atender necesidades específicas de conjuntos de organizaciones con intereses y requisitos afines. Las nubes comunitarias favorecen el intercambio de recursos y control al tiempo que mantienen la personalización y la seguridad. La subdivisión en regiones al interior de estas

nubes comunitarias añade un nivel adicional de eficiencia y adaptabilidad (*Servidores En La Nube | Qué Son, Tipos Y Ventajas Para Las Empresas*, n.d.).

2.2. Pregunta de Investigación

¿Es posible crear una región utilizando OpenStack, que pueda integrarse de manera efectiva en una infraestructura externa de nube comunitaria, permitiendo una colaboración más amplia entre regiones y eficiente en la gestión de recursos?

2.3. Planteamiento del Problema

Antes de la adopción de una infraestructura de nube comunitaria para conectar las diferentes escuelas de una facultad, la institución puede enfrentar diversos obstáculos y desafíos que afectan su eficiencia y flexibilidad en la gestión de operaciones académicas y administrativas. Estos obstáculos incluyen:

- **Comunicación Descentralizada:** En una facultad sin una infraestructura de nube, la comunicación entre escuelas puede ser dispersa y fragmentada. La falta de una plataforma unificada para compartir información, colaborar en proyectos y gestionar comunicaciones puede resultar en retrasos y falta de eficiencia en las operaciones de la facultad.
- **Dificultades en el Acceso Remoto:** En situaciones en las que es necesario acceder a recursos y datos de la facultad desde ubicaciones remotas, como el trabajo desde casa o la enseñanza en línea, la falta de una nube comunitaria puede dificultar el acceso a herramientas y documentos de manera eficiente y segura.
- **Problemas de Almacenamiento y Acceso a Datos:** La gestión de datos puede ser un desafío, ya que cada usuario puede tener su propio sistema de almacenamiento y acceso a datos. Esto dificulta la disponibilidad y la integridad de la información, lo que puede generar pérdida de datos y problemas de colaboración entre las distintas escuelas de la facultad.

- Falta de Escalabilidad y Flexibilidad: La escalabilidad puede ser un reto ya que cada escuela por separado podría optar por diferentes formas que dificultarían la colaboración entre las mismas.
- Problemas de Seguridad y Cumplimiento: Cada escuela debe implementar medidas de seguridad por separado en cada caso, además no se tendría un estándar de privacidad y protección de datos.
- Falta de Colaboración Eficiente: La colaboración entre escuelas y la gestión de proyectos pueden ser menos eficientes debido a la falta de una plataforma centralizada que permita compartir documentos, recursos y comunicación en tiempo real.

2.4. Objetivos

2.4.1. Objetivo General

Diseñar y probar una región en la nube comunitaria basada en la plataforma OpenStack, que ofrezca servicios de almacenamiento, red y cómputo, y se integre con la infraestructura existente.

2.4.2. Objetivos Específicos

- Configurar y desplegar una región, asegurando que las instancias creadas en esta, puedan ser accedidas desde una interfaz común entre regiones.
- Establecer la configuración de red de la región, definiendo políticas de seguridad para garantizar la privacidad y la conectividad adecuada entre los recursos.
- Asegurar la asignación eficiente de recursos y la capacidad de escalar horizontalmente según la demanda en la región.
- Validar la interoperabilidad de la región con la infraestructura existente y garantizar que los servicios de almacenamiento, red y cómputo puedan ser consumidos de manera coherente y confiable.

- Facilitar el acceso remoto a las instancias virtuales de la región mediante el protocolo SSH y que garantice la autenticación y autorización de usuarios de manera óptima.
- Establecer un sistema de almacenamiento centralizado en la región con el objetivo de proporcionar una solución robusta que asegure la disponibilidad y la integridad de los datos para fortalecer la continuidad del servicio.
- Crear instancias virtuales a partir de imágenes de sistemas operativos almacenados en la región que posean acceso a internet y conectividad con instancias tanto de la misma región como aquellas desplegadas en regiones diferentes.
- Emplear la migración en vivo de las instancias virtuales para asegurar la continuidad de los servicios que estas brindan, asegurando la tolerancia a fallos.

2.5. Justificación

Las soluciones en la nube han mejorado la eficiencia y la colaboración en instituciones educativas, entidades gubernamentales y empresas de diversos tamaños. El enfoque de Infraestructura en la Nube se ha convertido en un pilar fundamental para optimizar el uso de recursos y abordar los desafíos tecnológicos en la actualidad. En este contexto, la nube comunitaria surge como una respuesta precisa y crucial a las necesidades cambiantes de la sociedad y la industria. (Carr, 2008)

La importancia de esta solución se deriva de la demanda creciente de colaboración, escalabilidad y optimización de recursos en diversos contextos. Por ejemplo, en instituciones educativas, la implementación de una nube comunitaria basada en regiones potencia el aprendizaje colaborativo y la investigación, al mismo tiempo que maximiza la eficiencia en el uso de recursos. En el ámbito gubernamental, la solución aborda los desafíos de seguridad y el acceso a datos sensibles, mejorando la eficacia operativa y fomentando la colaboración entre diferentes departamentos. En el entorno empresarial, esta implementación brinda oportunidades para la agilidad empresarial, permitiendo la adaptación rápida a las cambiantes demandas del mercado. El desarrollo de esta solución aborda cuestiones cruciales y desafíos inherentes en estos entornos.

El despliegue de regiones en la nube presenta soluciones tecnológicas sólidas para la escalabilidad, el uso eficiente de recursos, la colaboración entre los equipos y la seguridad de datos. De este modo, al integrarse de manera transparente en la infraestructura comunitaria y ofrecer servicios de Storage, Network y Cómputo a través de regiones, la solución fomenta la interoperabilidad y la eficacia. (Marinescu, 2017)

La implementación de regiones en la nube, respaldada por OpenStack y con servicios que son clave para su realización, ofrece una solución tecnológica adaptable que optimiza la colaboración, la escalabilidad y la eficiencia en un panorama tecnológico en constante cambio.

2.6. Alcances

- Diseñar el prototipo de una región para una nube comunitaria basada en OpenStack que tenga la capacidad de lanzar instancias con acceso a internet y comunicarse con otras de la misma región y aquellas desplegadas en regiones diferentes, facilitando la colaboración entre los miembros de las diferentes regiones al proveerles un almacenamiento centralizado, normas de seguridad estandarizadas, acceso remoto y tolerancia a fallos.
- Implementar Vlans para la gestión de redes internas de la región de la nube comunitaria basada en OpenStack.
- Distribuir en diferentes zonas geográficas las regiones y la infraestructura para formar una misma topología.

2.7. Limitaciones

- El diseño, la construcción y las pruebas de una región en la nube comunitaria está limitada por los recursos disponibles de cada miembro del grupo, incluyendo capacidad de almacenamiento, potencia de cómputo y ancho de banda de red. Estas limitaciones pueden afectar la escala y la capacidad de la región.
- La disponibilidad y calidad del ancho de banda de Internet de cada miembro del equipo puede ser una limitación significativa en el desarrollo de la infraestructura de la región en la nube comunitaria. Dado que el proyecto involucra el despliegue de servicios de almacenamiento, red y cómputo, la transferencia de datos y la comunicación constante con la infraestructura central y entre las regiones podrían requerir una cantidad sustancial de ancho de banda.
- No se realiza una gestión de usuarios de forma óptima debido a que el servicio de LDAP no ha sido implementado puesto que, el prototipo se centra en la implementación de la región así como la administración de las instancias virtuales creadas en esta.

3. Marco Teórico

3.1. Virtualización

La virtualización es una técnica que permite compartir una sola instancia física de una aplicación o recurso entre múltiples organizaciones o inquilinos (clientes). Lo hace asignando un nombre lógico a un recurso físico y proporcionando un puntero a ese recurso físico según la demanda.

La creación de una máquina virtual sobre un sistema operativo y hardware existentes se conoce como "Virtualización de Hardware". Las Máquinas Virtuales proporcionan un entorno que está lógicamente separado del hardware subyacente. La máquina en la que se crea la máquina virtual se conoce como máquina anfitriona (host), y la máquina virtual se denomina máquina invitada. Esta máquina virtual es gestionada por un software o firmware, que se conoce como hipervisor (Marinescu, 2017).

La virtualización aporta muchos beneficios a las organizaciones tales como:

- Utilización eficiente de los recursos.
- Administración automatizada de las TI.
- Recuperación de desastres más rápida.

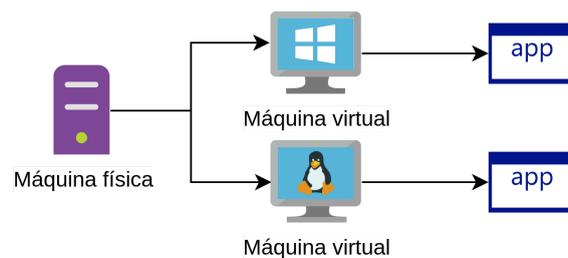


Figura 1: Conceptualización de virtualización.

3.1.1. Hipervisores

La virtualización implica la utilización de un software especializado conocido como hipervisor, el cual tiene la capacidad de crear múltiples instancias en la nube o máquinas virtuales en un solo equipo físico. En este contexto, se explorarán dos aspectos clave de la virtualización los cuales son:

- **Instancias en la nube o máquinas virtuales:** Estos elementos representan una parte fundamental de la virtualización. Después de instalar el software de virtualización en la computadora, se habilita la creación de una o más máquinas virtuales. Estas máquinas virtuales operan de manera similar a otras aplicaciones en la computadora y pueden tener su propio sistema operativo, el cual puede ser diferente o igual al sistema operativo del host.
- **Hipervisores:** Por otro lado, el hipervisor, que es el software de virtualización instalado en la máquina física, desempeña un papel esencial en este proceso. Funciona como un intermediario entre las máquinas virtuales y el hardware subyacente o el sistema operativo del host, coordinando eficazmente el acceso a los recursos físicos asignados a cada máquina virtual (Amazon, n.d.).

3.1.1.1. Tipos de hipervisores

3.1.1.1.1. Hipervisor tipo 1 o Bare-Metal

Un hipervisor tipo 1, conocido como hipervisor de bare metal, se instala directamente en un servidor físico y se utiliza principalmente en servidores. Este componente esencial en la virtualización actúa como un sistema operativo principal programado para permitir el acceso a los componentes de hardware subyacentes, como si tuvieran acceso directo al hardware. Esto permite dividir el sistema operativo en su infraestructura física y brinda la capacidad de operar múltiples unidades virtuales en un solo servidor, lo que permite una implementación más densa a un costo relativamente alto, pero con limitaciones en la capacidad de utilizar todas las características del equipo.

Cada cliente recibe una copia de su propio servidor dedicado, aunque los recursos virtuales del servidor, como ciclos de CPU, memoria y ancho de banda de red, se comparten entre todos los clientes en el servidor. Este tipo de hipervisor generalmente se encuentra en un centro de datos empresarial o en otros entornos basados en servidores. Algunos ejemplos de hipervisores de tipo 1 son: KVM, Microsoft Hyper-V, VMware vSphere, entre otros (Red Hat, 2023).

El hipervisor de tipo 1 ofrece flexibilidad al permitir un uso más eficiente del hardware, especialmente cuando no se utilizan todos los recursos físicos disponibles. No requiere un sistema operativo básico y se utiliza comúnmente en productos relacionados con centros de datos, donde se instala directamente sobre el servidor físico.

Las ventajas de los hipervisores de bare metal (tipo 1) incluyen una mayor seguridad, una mayor densidad de hardware y acceso directo al hardware. Sin embargo, tienen desventajas como la necesidad de componentes de hardware específicos, requisitos estrictos de hardware y un costo elevado (Red Hat, 2023).

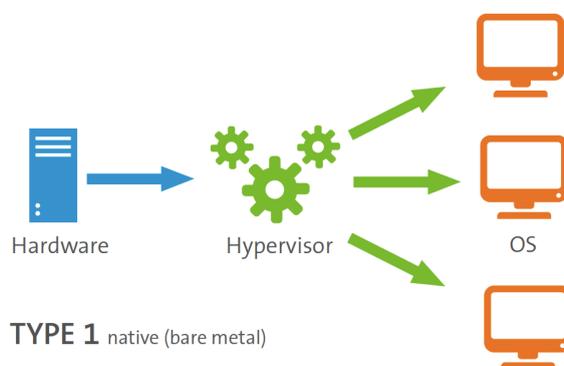


Figura 2: Arquitectura de un hipervisor tipo 1.

3.1.1.1.2. Hipervisor tipo 2 o Alojado

Un hipervisor tipo 2, conocido como hipervisor alojado, se ejecuta en un sistema operativo alojado que también proporciona la gestión de E/S y memoria. Estos hipervisores alojados se sitúan entre el hardware y las máquinas virtuales. Básicamente, este tipo de hipervisor tipo 2 se coloca por encima de los sistemas operativos y no por debajo de los sistemas operativos o las máquinas virtuales, lo que permite que otra máquina virtual se ejecute en la máquina virtual existente.

Los hipervisores alojados son gestionados y actúan como controladores de corrección de errores visuales. Su funcionalidad incorporada permite realizar cualquier tarea. Para crear y mantener un entorno virtual, no es necesario instalar un software separado en otra máquina. Cuando no se puede encontrar ninguna otra aplicación dentro

del sistema operativo, se utiliza el hipervisor tipo 2. Se pueden realizar tareas como resúmenes, herramientas de modificación, importación o exportación, etc., a través del hipervisor tipo 2.

Los hipervisores alojados pueden ejecutar más sistemas operativos que cualquier otro sistema operativo. Por ejemplo, si alguien con un sistema operativo Windows desea utilizar una aplicación diseñada para un sistema operativo Linux, puede usar el sistema operativo Linux en un entorno virtual sobre la máquina virtual de Windows. El hipervisor de tipo 2 es mejor para los usuarios individuales que buscan ejecutar varios sistemas operativos en una computadora personal. Algunos ejemplos de hipervisores de tipo 2 son: VMware Workstation y Oracle VirtualBox (Red Hat, 2023).

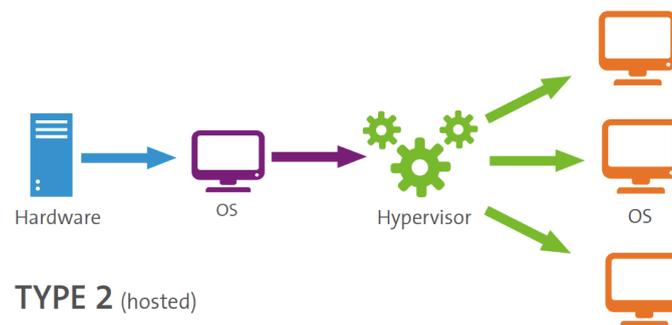


Figura 3: Arquitectura de un hipervisor tipo 2.

3.2. Cloud Computing

El cloud computing es un modelo que permite un acceso fácil y en tiempo real a recursos informáticos compartidos, como redes, servidores, almacenamiento y aplicaciones, las cuales pueden ser provisionados y liberados de manera rápida y con poco esfuerzo de gestión (Ruparelia, 2015).

Para entenderlo mejor, se debe de imaginar que se tiene un servicio de almacenamiento en la nube, como Dropbox, Google Drive o cualquier otro servicio. En lugar de almacenar los archivos en la computadora o teléfono personal, estos se guardan en la nube, donde es posible acceder a ellos desde cualquier lugar con conexión a internet y compartirlos fácilmente con otras personas. En otras palabras, los datos están almacenados

en alguna parte de Internet o en muchas partes, y que detrás de todo eso hay una red de servidores que se encargan de buscar y entregar lo que se necesita. El cloud computing hace posible esta flexibilidad y accesibilidad a recursos informáticos.

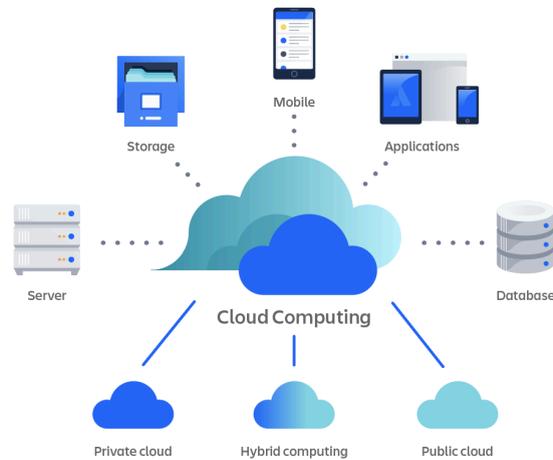


Figura 4: Cloud Computing. De (Yalalov, 2022)

3.2.1. Modelos de despliegue de servicios cloud

Las organizaciones se enfrentan a una variedad de alternativas al decidir cómo migrar sus sistemas, aplicaciones, software y datos hacia la nube. Se destacan cuatro modelos clave para implementar la nube: nube privada, nube pública, nube comunitaria y nube híbrida (Marinescu, 2017). La elección de uno de estos modelos tiene un impacto significativo en aspectos cruciales como la capacidad de crecimiento, la confiabilidad, la seguridad y el costo asociado a la transición hacia la nube. Además de estos modelos, es importante destacar que las organizaciones también deben considerar factores como la ubicación de los servidores en la nube, la gestión de datos y la adaptación de su infraestructura existente para aprovechar al máximo los beneficios de la nube. Por lo tanto, la toma de decisiones en la implementación de la nube es un proceso estratégico que requiere una cuidadosa evaluación de las necesidades y objetivos específicos de cada organización.

3.2.1.1. Nube Pública

Como su nombre indica, está disponible para el público en general. En este contexto, el público puede ser un consumidor o una organización que desea utilizar

servicios en la nube. El acceso a los servicios de la nube pública suele ser a través de Internet en lugar de una red privada o restringida. La nube pública es el modelo de implementación con el que la mayoría de las personas están familiarizadas. Tiene aplicabilidad en varios niveles de abstracción, lo que significa que se pueden encontrar nubes públicas que ofrecen infraestructura, plataforma, software, información o procesos empresariales como servicio. A diferencia de los servicios en la nube de otros modelos de implementación, que pueden accederse a través de redes locales, extensas o a nivel mundial (Internet), el modelo de nube pública casi siempre solo está disponible a través de Internet y con un cargo mensual por su uso. Algunos ejemplos de servicios de nube pública incluyen Google Cloud Platform (GCP), Amazon Web Services (AWS), Microsoft Azure, Alibaba Cloud, entre otros. Todos ellos comparten un modelo de precio basado en gastos operativos mensuales y tienen costos de inversión inicial bajos o nulos para el consumidor (Ruparelia, 2015).

3.2.1.2. Nube Privada

Tiene como alcance una organización, una unidad de negocios o incluso una sola persona. Una empresa puede contar con su propia nube privada que ofrece servicios a través de su red de área amplia (WAN). Se puede pensar en una WAN como una especie de Internet de toda la empresa que restringe el acceso a personas ajenas mediante dispositivos de seguridad como firewalls. Por otro lado, una red de área local (LAN) es similar a una WAN, pero su alcance geográfico es mucho más pequeño, generalmente limitado a un lugar específico, como un hogar o un local comercial. Incluso una persona podría disponer de su propia nube privada para consumir servicios a través de una LAN.

Para ilustrar lo anterior, se debe de imaginar que un hogar podría establecer su propia nube privada y que esta cumple varias funciones las cuales son:

- Conectar un servidor de transmisión a un decodificador de vídeo, permitiendo ver, grabar o reproducir videos desde cualquier lugar de la casa.

- Proporcionar un servidor de respaldo para almacenar archivos de forma centralizada.
- Ofrecer un servicio de sincronización que sincroniza datos entre dispositivos (computadoras portátiles, teléfonos móviles, tabletas, etc.) utilizando una LAN inalámbrica.

Una nube privada es aquella que brinda servicios a través de una LAN o una WAN y limita el consumo de esos servicios a un grupo selecto de usuarios. En circunstancias limitadas, los servicios de nube privada pueden entregarse a través de Internet, pero con restricciones de acceso para que solo entidades privadas puedan acceder a esos servicios.

En la práctica, problemas de integridad y seguridad de datos pueden dificultar la entrega de servicios a través de Internet para las nubes privadas. Por lo general, las nubes privadas presentan un componente no recurrente y también incluyen un componente de gastos operativos (Red Hat, 2023).

3.2.1.3. Nube Comunitaria

La nube comunitaria es una versión más amplia de la nube privada. Esta nube brinda soporte a una comunidad que comparte intereses comunes o preocupaciones compartidas, como requisitos de seguridad, privacidad de datos, entorno regulatorio, modelo de negocio y necesidades de los usuarios finales. La nube comunitaria puede incluso tener una región geográfica como su alcance, por ejemplo, una nube comunitaria compuesta por varias universidades mediante algún convenio. El concepto de comercio es interesante porque se puede extender a cualquier número de industrias o grupos empresariales: una nube de la industria papelera, una nube de publicaciones, una nube de regulación bancaria, una nube de la industria de la salud específica para un país (por ejemplo, una Nube Comunitaria del Ministerio de Salud) o para el sector de la salud a nivel mundial (por ejemplo, una Nube Comunitaria de Salud Mundial). Esta nube puede tener como participantes a organismos reguladores, proveedores de servicios de salud, profesionales de la salud y consumidores, o cualquier combinación de estos. A diferencia de

una nube privada, los servicios de la nube comunitaria generalmente se entregan a través de Internet y suelen tener un modelo de precio basado en gastos operativos (Ruparella, 2015).

3.2.1.4. Nube Híbrida

Se define como una combinación de al menos dos entornos informáticos que comparten información y ejecutan aplicaciones uniformes para empresas o negocios (Red Hat, 2023). Estos entornos pueden incluir lo siguiente:

- Al menos una nube privada y una nube pública.
- Dos o más nubes privadas.
- Dos o más nubes públicas.
- Un servidor dedicado (bare metal) o un entorno virtual conectado a al menos una nube.

El término "nube" se refiere a una red de recursos informáticos accesibles a través de Internet según sea necesario. Este entorno facilita la provisión y el intercambio de servicios informáticos, como el almacenamiento de datos, aplicaciones de software, análisis de sistemas y herramientas de inteligencia (Red Hat, 2023).

Cuando se combinan estas nubes, sin importar su ubicación física, se crea un nuevo entorno que se conoce como una nube híbrida. La estrategia de la nube híbrida promueve la portabilidad de las cargas de trabajo, lo que permite que las aplicaciones funcionen de manera consistente en diferentes entornos y se comuniquen entre sí.

3.2.2. Modelo de servicio cloud

3.2.2.1. Infraestructura como servicio (IaaS)

La informática en la nube permite el acceso a recursos informáticos en un entorno virtualizado conocido popularmente como 'la nube'. La Infraestructura como Servicio (IaaS, por sus siglas en inglés) proporciona recursos virtualizados de hardware (no físicos, sino simulados mediante software) a los consumidores, conocidos como recursos virtuales o componentes virtuales (Marinescu, 2017). Ofrece la posibilidad de utilizar de forma remota

procesadores virtuales, memoria, almacenamiento y recursos de red virtuales para los consumidores. Estos recursos virtuales pueden utilizarse de la misma manera que los recursos físicos (hardware) para construir cualquier configuración informática, como máquinas virtuales o redes virtuales. Por esta razón, la IaaS también se conoce como Hardware como Servicio (HaaS).

Los consumidores ya no necesitan gestionar ni controlar la infraestructura informática subyacente que consumen como parte de la Infraestructura como Servicio (IaaS). La IaaS es la capa más baja del modelo de servicio de informática en la nube. Se trata de una solución informática en la que las complejidades y gastos de gestionar el hardware subyacente se externalizan a algunos proveedores de servicios en la nube.

Aquí, el significado de externalización es diferente de su interpretación convencional. En un sistema tradicional, la externalización del hardware informático significa que alguna otra parte proporcionará y gestionará los recursos de hardware según los requisitos del usuario. Los usuarios pueden acceder directamente a ese hardware y utilizar esos recursos instalando el software necesario en ellos. En el modelo de IaaS de la informática en la nube, el proveedor organiza y gestiona los recursos de hardware para los usuarios, pero los usuarios no pueden acceder directamente a esos recursos de hardware. Los recursos de hardware se representan como componentes de software simulados (implementados mediante técnicas de virtualización de recursos) y se entregan a los consumidores a través de Internet (o una red) mediante servicios web (Marinescu, 2017). Los consumidores pueden utilizar esos componentes simulados de la misma manera que los dispositivos de hardware reales y pueden construir sistemas informáticos con la capacidad de procesamiento, memoria, almacenamiento y facilidades de red necesarios.

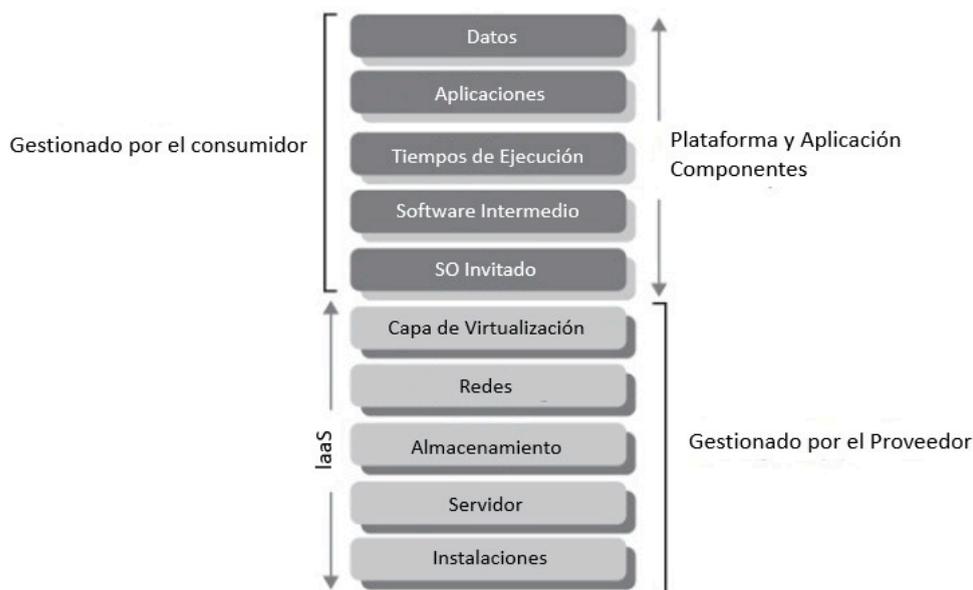


Figura 5: Componentes de un IaaS. De (Cloud Computing, Sandeep Bhowmik)

3.2.2.1.1. Características de un IaaS según Nayan Ruparelia (Ruparelia, 2015)

- **Recursos Virtuales:** Proporciona recursos de hardware virtualizados en lugar de recursos físicos. Estos recursos incluyen procesadores virtuales, memoria, almacenamiento y recursos de red.
- **Acceso Remoto:** Los consumidores pueden acceder a estos recursos virtualizados de forma remota desde cualquier ubicación a través de Internet.
- **Flexibilidad:** Los recursos virtuales pueden utilizarse de manera similar a los recursos físicos para crear configuraciones informáticas personalizadas, como máquinas virtuales o redes virtuales.
- **Outsourcing de Hardware:** La IaaS subcontrata la gestión y la complejidad de la infraestructura de hardware subyacente a proveedores de servicios en la nube. Los usuarios no tienen que preocuparse por la gestión de hardware físico.
- **Representación Virtual:** Los recursos de hardware se representan como componentes de software simulados mediante técnicas de virtualización de recursos.
- **Servicios por Internet:** Estos recursos virtuales se entregan a los consumidores a través de Internet o redes utilizando servicios web.

- **Personalización:** Además de ofrecer componentes de recursos por separado, los proveedores de IaaS suelen ofrecer máquinas virtuales personalizadas que se crean a partir de esos componentes virtuales.
- **Los tres modelos de servicio (SaaS-PaaS-IaaS):** juntos se conocen como el modelo SPI de la nube. IaaS proporciona infraestructura como servidores, almacenamiento y facilidades de red; PaaS ofrece entornos de desarrollo y despliegue de aplicaciones, y SaaS entrega aplicaciones de software.

Tanto en los sistemas de computación tradicionales (no en la nube) como en los de computación en la nube, existen tres capas de instalaciones informáticas, donde la aplicación se encuentra en la parte superior de la pila para operar sobre los datos. Descendiendo en la pila, las dos capas subyacentes son las instalaciones de plataforma e infraestructura.

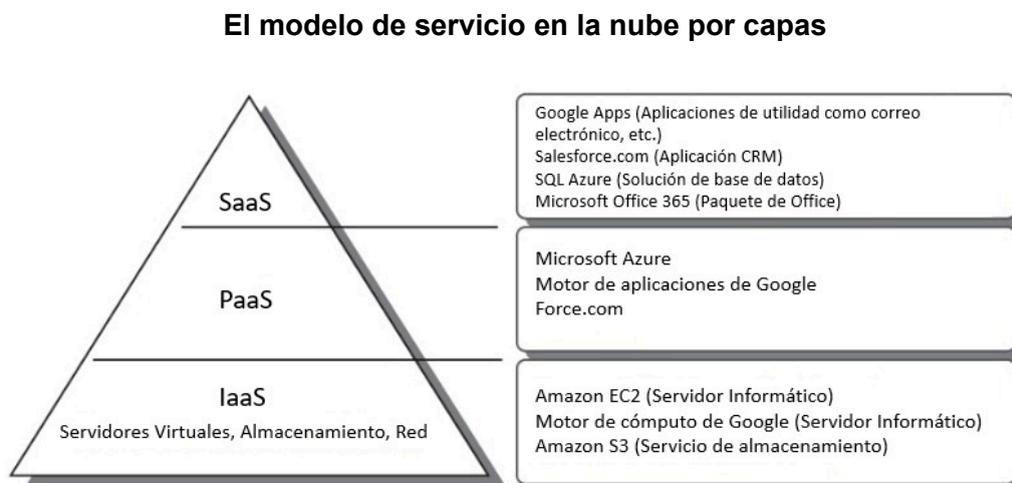


Figura 6: Diferentes tipos de servicios de cloud. De (Cloud Computing, Sandeep Bhowmik)

3.2.2.1.2. Componentes de un IaaS según Nayan Ruparelia (Ruparelia, 2015)

IaaS se divide en tres componentes principales: cómputo, red y storage. Con estas ofertas, los usuarios tienen los elementos básicos que necesitan para crear sus sistemas personalizados, tan complejos o potentes como lo requieran, y la capacidad de escalar hacia arriba o hacia abajo según sus necesidades.

Cómputo: En la infraestructura como servicio (IaaS) se basa en servidores físicos que pueden dividirse en máquinas virtuales (VM) mediante un hipervisor. Estas VM pueden ejecutar su propio sistema operativo y aplicaciones de manera independiente. Los proveedores de IaaS mantienen centros de datos con servidores bare-metal y ofrecen opciones de escalabilidad vertical y horizontal. Los desarrolladores pueden utilizar herramientas como Kubernetes para contener su código y gestionar de manera eficiente la escalabilidad y el equilibrio de carga. Los usuarios pueden seleccionar sistemas operativos e iniciar VM fácilmente

Storage: En la infraestructura como servicio (IaaS) se divide en tres categorías principales: almacenamiento de archivos, almacenamiento de objetos y almacenamiento de bloques. El almacenamiento de archivos organiza los datos de manera jerárquica, similar a las computadoras personales, mientras que el almacenamiento de objetos se centra en la simplicidad y la eficiencia para manejar grandes volúmenes de datos no estructurados. Por otro lado, el almacenamiento de bloques ofrece velocidad y eficiencia en la transferencia de datos, pero puede ser menos amigable para el usuario. La elección entre estas opciones es crucial para garantizar el rendimiento y la escalabilidad de las aplicaciones en la nube, y depende de las necesidades específicas de cada proyecto.

Red: Esta se comunica con la función de almacenamiento, otras máquinas virtuales, contenedores, otros servidores, internet, intranet y otros componentes. Es la forma en que la información se transfiere a través de la arquitectura, independientemente de los puntos finales. Los usuarios necesitarán diferentes anchos de banda de red según la cantidad de datos transmitidos entre los recursos informáticos.

3.2.2.1.3. Ventajas y desventajas de un IaaS según Sandeep Bhowmik (Bhowmik, 2017)

Ventajas de IaaS:

- Escalabilidad: Es posible aumentar o disminuir los recursos de infraestructura según las necesidades del negocio de manera rápida y sencilla, lo que permite gestionar picos de carga sin invertir en hardware adicional.

- Costos reducidos: Se elimina la necesidad de comprar y mantener hardware físico costoso. En su lugar, se paga por los recursos que realmente se utilizan, lo que puede reducir significativamente los gastos operativos.
- Flexibilidad: Existe la libertad de elegir entre una variedad de sistemas operativos y configuraciones de hardware, adaptándose a necesidades específicas.
- Tiempo de implementación más rápido: Es posible implementar servidores y recursos en minutos en lugar de semanas o meses que llevaría adquirir y configurar hardware físico.
- Mantenimiento simplificado: Los proveedores de IaaS se encargan del mantenimiento físico de la infraestructura, incluyendo la gestión de hardware, actualizaciones y parches de seguridad.
- Acceso global: Se pueden desplegar recursos en múltiples ubicaciones geográficas para ofrecer una mejor experiencia a los usuarios finales en diferentes regiones.

Desventajas de IaaS:

- Dependencia del proveedor: Al utilizar IaaS, se confía en el proveedor de servicios en la nube para la disponibilidad y la seguridad de la infraestructura. Cualquier interrupción de sus servicios podría afectar la operación.
- Costos impredecibles: Aunque IaaS puede reducir los costos operativos, si no se gestiona adecuadamente, los gastos pueden aumentar si no se monitorizan los recursos y se ajusta la capacidad según sea necesario.
- Complejidad inicial: Configurar y gestionar una infraestructura en la nube puede ser complejo, especialmente si no se tiene experiencia previa en la nube. Requiere un tiempo de aprendizaje y habilidades adicionales.
- Seguridad y cumplimiento: La gestión de la seguridad y el cumplimiento de normativas pueden ser desafiantes en un entorno de nube, y es responsabilidad del usuario garantizar que sus aplicaciones y datos sean seguros.

- Latencia: En algunas aplicaciones sensibles a la latencia, como juegos en línea o procesamiento de transacciones financieras, la latencia de la red en la nube puede ser un problema si no se elige una ubicación de centro de datos cercana.
- Posible bloqueo de proveedor: Si se decide cambiar de proveedor de IaaS en el futuro, la migración de aplicaciones y datos puede ser costosa y complicada.

3.2.3. Gestión de Acceso y Control en la Nube

3.2.3.1. Regiones

Una región se refiere a un área geográfica específica donde un proveedor de servicios en la nube tiene centros de datos y recursos informáticos. Cada región puede contener uno o varios centros de datos y suele estar ubicada en un lugar estratégico para ofrecer servicios a una determinada área geográfica. Estas regiones se utilizan para distribuir recursos y servicios de nube a los usuarios y empresas de manera eficiente y escalable. La existencia de múltiples regiones permite a los proveedores de nube distribuir recursos y servicios de manera estratégica para satisfacer las necesidades de los usuarios en diferentes partes del mundo, ofreciendo baja latencia y redundancia geográfica (Amazon, n.d.).

3.2.3.2. Zonas de Disponibilidad (AZ)

Las Zonas de Disponibilidad, también conocidas como Availability Zones, son subdivisiones dentro de una región en la infraestructura de nube. Cada zona de disponibilidad consta de uno o varios centros de datos físicamente separados pero cercanos entre sí. Estas zonas ofrecen una alta disponibilidad y redundancia a nivel local dentro de una región. En caso de que ocurra un problema en una zona, los servicios pueden redirigirse automáticamente a otra zona dentro de la misma región (Amazon, n.d.).

3.2.3.3. Relación entre regiones y zonas de disponibilidad

En la infraestructura de nube, las regiones y las zonas de disponibilidad están estrechamente vinculadas para garantizar la disponibilidad y la redundancia de los servicios. En este contexto, una región se puede comparar con una "zona geográfica" en la que se

ubican múltiples zonas de disponibilidad, según Google (2023) estos interactúan de la siguiente manera:

- **Alta Disponibilidad:** La principal función de las Zonas de Disponibilidad es proporcionar alta disponibilidad a nivel local dentro de una región. Esto significa que si un problema afecta a una zona de disponibilidad, como un fallo de hardware o una interrupción de servicio, los recursos y servicios pueden redirigirse automáticamente a otras zonas de disponibilidad dentro de la misma región. Esto evita interrupciones significativas en los servicios y garantiza que los usuarios puedan acceder a ellos de manera continua (Google, 2023).
- **Redundancia Geográfica:** Además de la alta disponibilidad local, las regiones en sí mismas también son componentes críticos para la redundancia geográfica. Las regiones representan ubicaciones geográficas separadas en las que los proveedores de servicios en la nube tienen centros de datos y recursos duplicados. Si una región completa se ve afectada por un evento inesperado, como un desastre natural, los servicios pueden conmutar a otra región, lo que garantiza la continuidad del servicio (Google, 2023).

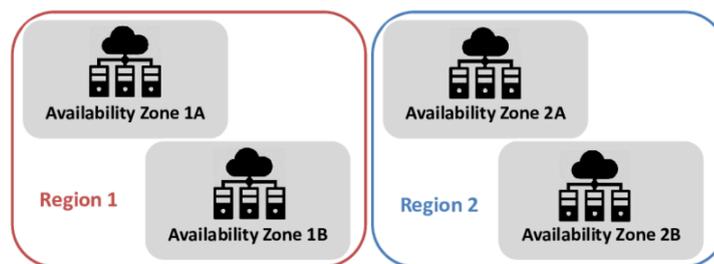


Figura 7: Infraestructura cloud con dos regiones y sus diferentes zonas de disponibilidad. De (Alroobaea & Krichen, 2020).

3.2.3.4. Importancia de las regiones en una nube comunitaria

Haciendo énfasis sobre la nube comunitaria, dado que es un enfoque de colaboración en la nube en el que múltiples organizaciones comparten recursos de TI ya sea en una infraestructura privada o pública, diseñada para satisfacer las necesidades específicas de un grupo de usuarios, la colaboración suele ser impulsada por interés

compartido, como requisitos de cumplimiento o la necesidad de compartir aplicaciones y datos de manera segura.

Las regiones en la nube son ubicaciones físicas donde los proveedores de servicios en la nube tienen centros de datos con servidores, almacenamiento y redes, es por eso que la importancia de la región en una nube comunitaria permite a las organizaciones, o por ende a sus usuarios reducir costos compartiendo recursos de infraestructura y servicios de TI, a su vez fomenta la colaboración y el compartir conocimientos entre sus usuarios, lo cual permite y da paso al impulso de la innovación y eficiencia. (Marinescu, 2017)

Según Marinescu (2017) algunos puntos fuertes de la región en la nube son:

- **Flexibilidad y Redundancia:** detalla que la presencia de múltiples regiones en una nube comunitaria garantiza que su operatividad es continua, con esto se refiere a que si una región se vuelve inaccesible debido a desastres naturales en sus servidores físicos, o algunas fallas de hardware o ciberataques, los usuarios de la nube comunitaria pueden actuar rápidamente para cambiar a otras regiones y evitar interrupciones graves en el servicio.
- **Latencia y Rendimiento:** puede mejorar notablemente el rendimiento de las aplicaciones al reducir la latencia, sus usuarios pueden alojar sus servicios en la región más cercana para minimizar los tiempos de respuesta y mejorar la experiencia.

Ejemplo práctico:

Redes Sociales y Distribución Global

Una de las aplicaciones más conocidas que se beneficia de la gestión de regiones en la nube es una red social líder, cómo Facebook, ya que opera en todo el mundo y atiende a miles de millones de usuarios y para garantizar un rendimiento óptimo y la disponibilidad de sus servicios, utiliza una estrategia de nube global con regiones distribuidas. Además, la estrategia de regiones de Facebook garantiza la resiliencia y la

redundancia. Si un centro de datos en una región específica experimenta problemas técnicos o una interrupción, Facebook puede redirigir el tráfico a otras regiones para mantener la disponibilidad de sus servicios sin interrupciones significativas.

3.2.3.4.1. ¿Cómo las regiones contribuyen a la colaboración en una nube comunitaria?

Las contribuciones de las regiones en la nube comunitaria son aspectos clave, tales como:

- **Distribución de datos y recursos:** permite la distribución de datos y recursos informáticos, lo que significa que pueden alojar datos y aplicaciones, esta distribución facilita la colaboración al mejorar los accesos a los recursos compartidos, reducir la latencia y acelerar la transferencia de datos.
- **Acceso remoto y colaboración en tiempo real:** la persistencia de regiones en una nube comunitaria permite acceder de manera eficiente a recursos y aplicaciones, esto es fundamental para la colaboración en tiempo.
- **Seguridad y cumplimiento:** también ayudan a garantizar la seguridad y el cumplimiento de datos, se puede optar por ubicar datos sensibles o que requieren cumplimiento en regiones específicas.
- **Conexiones de redes privadas:** pueden establecer conexiones de red privada entre las regiones de la nube, esto permite una colaboración segura y eficiente, ya que los datos pueden fluir de manera rápida y segura entre las redes y los recursos en la nube.
- **Economía de escala:** al compartir recursos de una infraestructura y servicios en la nube, se puede reducir los costos, lo que a su vez fomenta la colaboración al hacer que las soluciones sean más accesibles.

3.2.3.4.2. Escalabilidad y redundancia a través de regiones

La escalabilidad y la redundancia son aspectos críticos que se deben de tomar en cuenta para regiones en una nube comunitaria.

Definición de escalabilidad: en el contexto de la computación en la nube, la escalabilidad se refiere a la capacidad de aumentar o disminuir recursos de TI, como servicios, almacenamiento y capacidad de red, de manera dinámica para satisfacer las demandas cambiantes de las aplicaciones y servicios.

Importancia de la escalabilidad: En una nube comunitaria donde varios usuarios comparten recursos, la capacidad de escalar recursos de manera eficiente garantiza que cada usuario pueda satisfacer sus necesidades operativas sin afectar el rendimiento de los demás. La escalabilidad mejora la eficiencia al garantizar que los recursos se utilicen de manera óptima, pueden provisionar automáticamente recursos adicionales cuando sea necesario y liberarlos cuando ya no los requieran, evitando así el desperdicio de recursos infrutilizados. La escalabilidad también está directamente relacionada con la experiencia del usuario, dentro de este contexto entra el término de elasticidad, que es una característica de la escalabilidad que permite que los recursos se ajustan automáticamente según las condiciones, esto es especialmente valioso en una nube comunitaria, donde múltiples organizaciones pueden compartir recursos, ya que garantiza que cada organización obtenga los recursos que necesita cuando los necesita. (Amazon, n.d.)

Definición de redundancia: en el contexto de la computación en la nube, se refiere a la duplicación de componentes críticos, como servidores o datos, esto para garantizar la disponibilidad continua en caso de falla, por lo que la redundancia es fundamental para la continuidad del negocio y la minimización de tiempos de inactividad.

Importancia de la redundancia: al tener múltiples regiones y copias de datos críticos, se puede mantener las operaciones incluso en situaciones de falla en una nube, la redundancia aumenta la tolerancia a fallos, en caso de una interrupción permite una recuperación más rápida, debido a que los sistemas redundantes pueden asumir automáticamente la carga de trabajo y los datos, minimizando el tiempo de inactividad y el impacto en las operaciones. (Amazon, n.d.)

En resumen, la escalabilidad y la redundancia son fundamentales en las regiones de nube para garantizar la eficiencia, la continuidad operativa y la flexibilidad, por lo que estos

elementos permiten compartir recursos de manera efectiva y mantienen sus operaciones en funcionamiento incluso en condiciones adversas.

3.2.3.5. Herramientas y recursos útiles para gestionar regiones

La gestión de regiones en una nube comunitaria es un aspecto crítico en la administración de recursos de tecnología de la información en un entorno compartido, por lo que la gestión de regiones implica la planificación, configuración, monitoreo y optimización de recursos, en este contexto, es fundamental comprender cómo las herramientas, estrategias y mejores prácticas pueden facilitar la administración eficiente de regiones en una nube comunitaria. Este enfoque no solo garantiza la disponibilidad y el rendimiento continuo de los servicios, sino que también contribuye a la seguridad de datos, la escalabilidad y la protección en situaciones de falla. (Amazon, n.d.)

Consolas de Administración de proveedores de nube: los principales proveedores de nube, como Amazon Web Services (AWS), Microsoft Azure y Google Cloud, ofrecen consolas de administración que permiten configurar y gestionar recursos en diferentes regiones. Estas consolas proporcionan interfaces de usuario intuitivas para administrar servidores, bases de datos, redes y más en una nube comunitaria.

Herramientas de Orquestación y Automatización: Herramientas como Terraform, Ansible y Kubernetes son útiles para orquestar y automatizar la implementación y gestión de recursos en múltiples regiones. Estas herramientas permiten definir infraestructuras como código y desplegarlas de manera consistente en diferentes ubicaciones.

Gestión de Configuración: Utiliza herramientas como Chef, Puppet o AWS Config para gestionar y rastrear la configuración de recursos en varias regiones. Esto garantiza que la infraestructura se mantenga coherente y cumpla con los estándares de seguridad y cumplimiento.

Monitorización y Gestión de Rendimiento: Plataformas de monitoreo como Prometheus, Grafana, y las soluciones nativas de monitoreo de proveedores de nube

permiten supervisar el rendimiento y la disponibilidad de recursos en todas las regiones. Esto es esencial para la detección temprana de problemas y la optimización de recursos.

Redes Definidas por Software (DNS): La implementación de soluciones DNS, como VMware, KVM o soluciones nativas de proveedores de nube, facilita la gestión de redes en múltiples regiones al permitir la creación y configuración de redes virtuales de manera eficiente.

3.2.3.6. Ventajas estratégicas de distribuir servicios en regiones

La distribución estratégica de servicios en regiones en una nube comunitaria es un enfoque que ofrece beneficios significativos para las organizaciones que comparten recursos en este modelo de implementación. A medida que las nubes comunitarias continúan ganando relevancia en el panorama tecnológico, es esencial comprender cómo la distribución en múltiples regiones puede aportar ventajas estratégicas clave.

Tolerancia a fallos: es una característica crítica en la distribución de servicios en regiones en la nube. Significa que, incluso en el caso de un fallo inesperado en un componente o región, los servicios continúan funcionando sin interrupciones significativas. Esto se logra mediante la redundancia y la capacidad de conmutación por error.

Redundancia: implica tener duplicados o réplicas de componentes críticos, como servidores, bases de datos o servicios, en diferentes regiones. Si un componente falla, el tráfico o las solicitudes se pueden redirigir automáticamente a las réplicas activas en otras regiones, lo que evita la interrupción del servicio.

Conmutación por Error: la capacidad de cambiar automáticamente a un sistema de respaldo o a una región alternativa en caso de fallo. Esto garantiza que, si se detecta un problema en una región, el servicio se restablezca de manera inmediata en otra región operativa.

La tolerancia a fallos es esencial para garantizar la disponibilidad continua de servicios críticos y minimizar el impacto de problemas imprevistos en la experiencia del usuario.

Gestión de Datos Distribuida: implica el almacenamiento y la administración de datos en múltiples ubicaciones geográficas o regiones. Esto se hace para mejorar la disponibilidad, la velocidad de acceso y la redundancia de los datos. Algunos aspectos clave de la gestión de datos distribuida son:

Replicación de Datos: los datos se replican en diferentes regiones, lo que significa que existen copias idénticas en ubicaciones separadas. Esto asegura que los datos estén disponibles incluso si una región se vuelve inaccesible.

Sincronización en Tiempo Real: para garantizar que los datos sean coherentes en todas las ubicaciones, se utiliza la sincronización en tiempo real. Cualquier cambio en un conjunto de datos se replica de manera instantánea en todas las ubicaciones.

Balanceo de Cargas de Datos: los sistemas de gestión de datos distribuidos pueden equilibrar automáticamente las cargas de datos entre regiones para garantizar un rendimiento óptimo y evitar la congestión en una ubicación específica.

Acceso a Datos de Proximidad: al almacenar datos en regiones cercanas a los usuarios finales, se reduce la latencia de acceso a los datos, lo que mejora la velocidad de las aplicaciones y la experiencia del usuario.

La gestión de datos distribuida es fundamental en la nube para garantizar la disponibilidad y la consistencia de los datos, así como para mitigar los riesgos de pérdida de datos en caso de fallos.

3.2.3.7. Dominio

Se refiere a una unidad de aislamiento que puede utilizarse para separar y organizar recursos y usuarios. Se puede pensar en un dominio como una entidad de nivel superior que puede representar, por ejemplo, una organización, un departamento o una entidad que controla un conjunto específico de recursos en la nube. Los dominios permiten una segregación lógica y la asignación de recursos a diferentes grupos de usuarios (OpenStack, 2023).

3.2.3.8. Proyecto

Se utiliza para agrupar y gestionar recursos relacionados dentro de un dominio. Un proyecto podría representar un equipo, un proyecto de desarrollo o cualquier otra entidad que necesite su propio conjunto de recursos en la nube. Los proyectos permiten a los usuarios gestionar y controlar recursos, como máquinas virtuales, redes y almacenamiento, de manera independiente dentro de un dominio (Red Hat, n.d.).

3.2.3.9. Usuarios

Son individuos que interactúan con los servicios y recursos de la nube. Estos usuarios pueden ser empleados de una organización, clientes, socios o cualquier entidad autorizada para acceder y utilizar los servicios en la nube. La gestión adecuada de usuarios es esencial para garantizar la seguridad y la eficiencia de los servicios en la nube (Law Insider, n.d.).

3.2.3.10. Roles

Se refiere a un conjunto de permisos o derechos que se asignan a un usuario o entidad para realizar acciones específicas en recursos de la nube. Los roles son fundamentales para establecer la seguridad y la gestión de acceso en la nube, asegurando que los usuarios tengan los privilegios adecuados para realizar sus tareas sin comprometer la seguridad (Google, n.d.).

3.2.3.11. Control de Acceso Basado en Roles (RBAC)

Es un modelo de seguridad que permite controlar y gestionar el acceso a recursos en la nube asignando roles específicos a usuarios y entidades. Cada rol tiene permisos predefinidos que determinan las acciones que un usuario puede realizar en los recursos de la nube. RBAC es una parte esencial de la gestión de acceso en la nube para garantizar la seguridad y la administración adecuada (Microsoft Azure, 2023).

3.3. Estrategias de Implementación en la Nube

Las soluciones de despliegue en la nube varían en enfoque y funcionalidad, abarcando desde la virtualización de servidores hasta la gestión de contenedores. La elección de la solución depende de las necesidades de la organización, como escalabilidad, seguridad y compatibilidad. Entre las soluciones populares se encuentra OpenStack, OpenNebula y VMware vSphere, cada uno con características únicas para adaptarse a las necesidades específicas de las organizaciones.

3.3.1. Openstack

OpenStack es una plataforma de código abierto que utiliza recursos virtuales para crear y gestionar nubes, tanto para uso público como privado. Los componentes clave de OpenStack se llaman "proyectos" y se ocupan de tareas esenciales como el procesamiento de datos, la gestión de redes, el almacenamiento de información, la autenticación de usuarios y la administración de imágenes. Además, se pueden combinar más de una docena de proyectos opcionales para personalizar y configurar nubes específicas según las necesidades.

En términos de virtualización, OpenStack toma recursos como el espacio de almacenamiento, la potencia de procesamiento (CPU) y la memoria RAM de diferentes fuentes y los divide utilizando una tecnología llamada "hipervisor" para luego distribuirlos según lo que se requiera. OpenStack utiliza un conjunto de reglas comunes llamadas "interfaces de programación de aplicaciones" (API) para acceder a recursos virtuales adicionales, que se organizan en grupos específicos para mejorar las capacidades de las herramientas estándar que utilizan administradores y usuarios para trabajar en la nube (Red Hat, 2021).

3.3.1.1. Componentes principales

Al hablar de Openstack, es necesario examinar y subsanar cualquier deficiencia en la comprensión o falta de conocimiento potencial, para tener una visión más sólida de los conceptos fundamentales y el uso de cada uno de sus componentes principales. Para

comprender su funcionamiento, es beneficioso empezar analizando de manera concisa los elementos que lo hacen posible y preguntarse cuál es el propósito de cada uno de estos.

En el marco de OpenStack, se encuentran servicios básicos que gestionan aspectos como la informática, las conexiones de red, el almacenamiento, la identidad y las imágenes. Los servicios principales constituyen la infraestructura que permite administrar funciones como los paneles de control, la coordinación, el aprovisionamiento de hardware sin sistema operativo, la mensajería, los contenedores y la gobernabilidad.

3.3.1.1.1. Nova

Nova es el proyecto de OpenStack que brinda una manera de crear instancias de cómputo, también conocidas como servidores virtuales. Para llevar a cabo este servicio, Nova opera como un conjunto de programas que se ejecutan en servidores Linux preexistentes. Nova tiene la capacidad de crear máquinas virtuales y servidores físicos (mediante el uso de Ironic), y aunque su compatibilidad con contenedores de sistema es limitada, desempeña un papel fundamental en la infraestructura de OpenStack (Red Hat, 2021).

3.3.1.1.2. Neutron

Neutron es un proyecto de OpenStack diseñado para proporcionar "conectividad de red como servicio (NaaS)". La plataforma OpenStack Networking (Neutron) ofrece a los usuarios una API que les permite crear configuraciones de red complejas, definir conectividad de red, establecer políticas avanzadas de red y gestionar el direccionamiento en el entorno de la nube. Su función principal es supervisar la creación y administración de una infraestructura de red virtual, que incluye elementos como redes, switches, subredes y routers para los dispositivos que son gestionados por el servicio OpenStack Compute (Nova) (OpenStack, 2023).

3.3.1.1.3. Keystone

Keystone es el servicio en OpenStack que se encarga de gestionar la identificación de usuarios y la autorización para acceder a recursos. Actualmente, utiliza autenticación

basada en tokens y autorización de servicios de usuario. Recientemente, se ha renovado para permitir futuras expansiones, lo que significa que podrá trabajar con servicios externos y usar métodos de autenticación y autorización como OAuth, SAML y OpenID en versiones posteriores (OpenStack, 2023).

3.3.1.1.4. Cinder

Cinder es un servicio de almacenamiento en bloques para OpenStack. Su objetivo es ofrecer recursos de almacenamiento a los usuarios finales que pueden ser utilizados por el Proyecto de Cómputo de OpenStack (Nova). Esto se logra a través de una implementación de referencia (LVM) o controladores de complemento para otros sistemas de almacenamiento por ejemplo Ceph (OpenStack, 2023).

3.3.1.1.5. Swift

El proyecto de Almacenamiento de Objetos de OpenStack, también conocido como Swift, proporciona software de almacenamiento en la nube que permite a los usuarios almacenar y recuperar grandes cantidades de datos de manera fácil a través de una API sencilla. Está diseñado para funcionar en entornos de gran escala y se ha optimizado para garantizar la durabilidad, disponibilidad y acceso simultáneo de datos en todo el conjunto de información. Swift es especialmente adecuado para almacenar datos no estructurados que pueden crecer de manera ilimitada (OpenStack, 2023).

3.3.1.1.6. Placement

Placement es un servicio dentro de OpenStack que brinda a los demás servicios una API HTTP para monitorizar los inventarios y usos de recursos en la nube, lo que resulta fundamental para permitir que otros servicios dentro de la plataforma gestionen y asignen sus recursos de manera efectiva. En otras palabras, Placement actúa como un sistema de control y monitoreo que ayuda a optimizar la asignación de recursos en la nube de OpenStack, lo que a su vez contribuye a una gestión más eficiente y un mejor rendimiento en todo el ecosistema de servicios de la plataforma (OpenStack, 2023).

3.3.1.1.7. Glance

El servicio de imágenes, también conocido como "Glance", brinda a los usuarios la capacidad de buscar, registrar y obtener imágenes de máquinas virtuales. A través de su API REST, permite acceder a información sobre metadatos de imágenes de máquinas virtuales y descargar las propias imágenes. Además, proporciona la flexibilidad de almacenar estas imágenes en diversas ubicaciones, desde sistemas de archivos simples hasta sistemas de almacenamiento de objetos, como OpenStack Object Storage, facilitando así la gestión y disponibilidad de las imágenes virtuales en entornos de nube (OpenStack, 2023).

3.3.1.1.8. Horizon

Horizon representa la implementación estándar del panel de control de OpenStack, proporciona una interfaz de usuario web para acceder y administrar los servicios que ofrece OpenStack, tales como Nova, Swift, Keystone, entre otros (OpenStack, 2023).

A continuación se presenta una vista general de la arquitectura de openstack y cómo se interconectan sus distintos componentes:

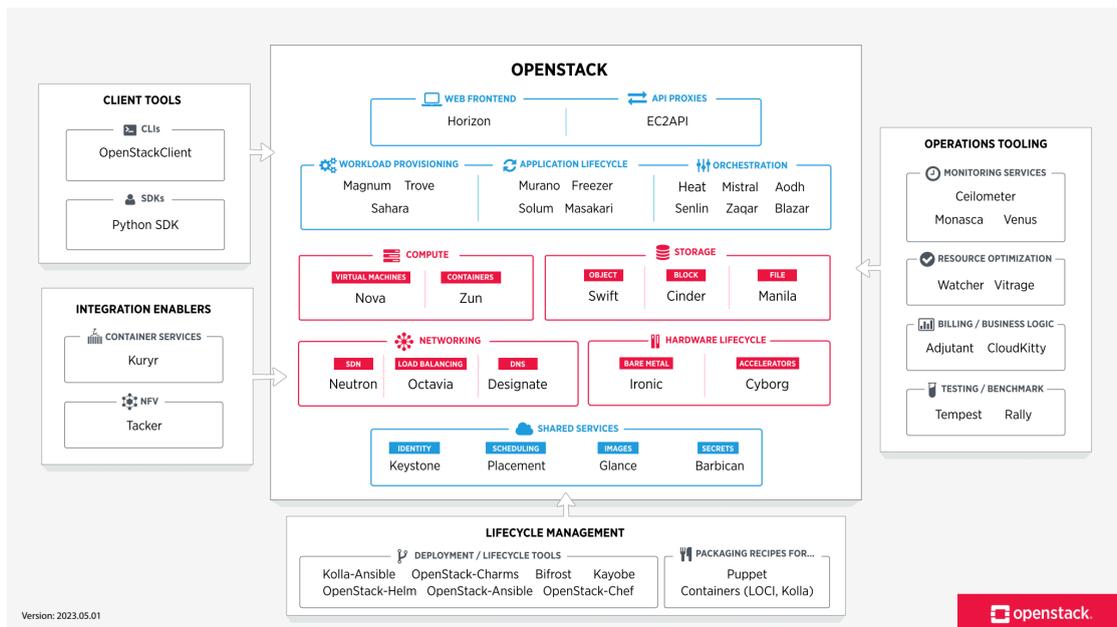


Figura 8: Vista general de Openstack con sus componentes individuales.

(Openstack, 2023)

3.3.2. OpenNebula

OpenNebula representa una plataforma de computación en la nube que se utiliza para gestionar infraestructuras de centros de datos heterogéneas y distribuidas. Esta plataforma se encarga de administrar la infraestructura virtual de un centro de datos, permitiendo la creación de implementaciones de infraestructura como servicio en entornos privados, públicos e híbridos. Las dos aplicaciones principales de OpenNebula son la virtualización de centros de datos y la implementación de infraestructura en la nube. Además, la plataforma puede proporcionar la infraestructura necesaria para operar una nube, complementando las soluciones de gestión de infraestructura existentes (Wikipedia, n.d.).

Los componentes principales de OpenNebula son:

- **OpenNebula Daemon (oned):** Es el corazón de la plataforma que se encarga de gestionar recursos, usuarios y máquinas virtuales, y permite la comunicación con otros servicios.
- **Base de Datos:** Almacena la información sobre la nube y necesita ser ajustada para un buen rendimiento.
- **Planificador (Scheduler):** Decide dónde ejecutar las Máquinas Virtuales en los servidores disponibles.
- **Provisión de Clúster Edge:** Crea clústeres en nubes públicas usando herramientas de automatización como Terraform y Ansible.
- **Monitoreo:** Reúne datos sobre el estado y el rendimiento de los servidores y las Máquinas Virtuales.
- **OneFlow:** Coordina servicios con varias Máquinas Virtuales, las controla y permite ajustar su tamaño automáticamente.
- **OneGate:** Permite que las Máquinas Virtuales envíen y reciban información desde/hacia OpenNebula para monitoreo y control.

En conjunto, estos componentes hacen que la gestión de una infraestructura de nube sea más fácil y eficiente (Opennebula, 2022).

3.3.3. VMware vSphere

VMware vSphere es la plataforma de virtualización de VMware, la cual transforma los centros de datos en infraestructuras de computación agregadas que incluyen recursos de redes, CPU y almacenamiento. vSphere administra estas infraestructuras como un entorno operativo unificado y le proporciona las herramientas para administrar los centros de datos que participan en dicho entorno. Los dos componentes principales de vSphere son ESXi y vCenter Server. ESXi es la plataforma de virtualización en la que se crean y se ejecutan las máquinas virtuales y los dispositivos virtuales. vCenter Server es el servicio mediante el cual se administran varios hosts conectados en una red y se agrupan recursos del host (VMware, n.d.).

3.4. Almacenamiento en la nube

El almacenamiento en la nube es un enfoque de computación en la nube en el que la información, como archivos, datos y recursos digitales, se guarda en servidores remotos en Internet. Estos servidores son mantenidos y administrados por un proveedor de servicios de computación en la nube. Los usuarios pueden acceder a sus datos almacenados en la nube desde cualquier dispositivo con conexión a Internet, lo que les permite almacenar, gestionar y recuperar información de manera conveniente y sin depender de recursos de almacenamiento locales en sus propios dispositivos (Harvey, 2020).

3.4.1. Almacenamiento en bloques

El almacenamiento de bloques implica dividir los datos en bloques de datos fijos y almacenarlos por separado con identificadores únicos. Estos bloques se pueden almacenar en diferentes entornos y sistemas. Cuando un usuario recupera un bloque, el sistema de almacenamiento reensambla los bloques en una unidad única. El almacenamiento de bloques se utiliza comúnmente para datos que se actualizan con frecuencia y es el tipo de

almacenamiento predeterminado para unidades de disco duro y datos en constante cambio (Harvey, 2020).

Ventajas segun Emily Freeman & Nathen Harvey:

- Alta velocidad y baja latencia cuando los bloques se almacenan localmente o cercanos entre sí.
- Confiabilidad, ya que los bloques se almacenan como unidades autocontenidas.
- Facilidad para realizar modificaciones sin necesidad de crear un nuevo bloque.

Desventajas segun Emily Freeman & Nathen Harvey:

- Falta de metadatos, lo que limita su utilidad para el almacenamiento de datos no estructurados.
- Dificultad para buscar y gestionar grandes volúmenes de datos de bloque.
- Costoso, especialmente a gran escala.

Usos Comunes:

- Bases de datos transaccionales.
- Servidores de correo electrónico.
- Almacenamiento de sistemas de archivos de máquinas virtuales (VMFS).

3.4.2. Almacenamiento de objetos

El almacenamiento de objetos es un sistema que organiza los datos en unidades separadas y autocontenidas llamadas objetos. Estos objetos se almacenan en un entorno plano sin jerarquías de carpetas ni subdirectorios, y cada objeto tiene un identificador único en lugar de un nombre de archivo y una ruta (Harvey, 2020, #). Cada objeto también contiene metadatos que proporcionan información sobre el archivo y los usuarios pueden establecer metadatos fijos o personalizados asociados a un objeto. Para acceder y gestionar objetos en el almacenamiento de objetos, se requiere el uso de una interfaz de programación de aplicaciones (API).

Ventajas segun Emily Freeman & Nathen Harvey:

- Manejo eficiente de grandes cantidades de datos no estructurados.
- Modelo de consumo asequible, donde solo se paga por la cantidad de almacenamiento de objetos que se necesita.
- Escalabilidad ilimitada, permitiendo agregar más almacenamiento según sea necesario.
- Utiliza metadatos para facilitar la búsqueda y recuperación de objetos.
- Ofrece capacidades avanzadas de búsqueda basadas en metadatos.

Desventajas segun Emily Freeman & Nathen Harvey:

- No permite bloquear archivos, lo que significa que cualquier usuario con acceso puede acceder a los objetos almacenados.
- Rendimiento más lento en comparación con otros tipos de almacenamiento, ya que los objetos requieren más tiempo de procesamiento.
- No es posible modificar una porción única de un archivo; solo se puede crear un nuevo objeto.

Usos Comunes:

- Gestión de datos de IoT.
- Almacenamiento de correos electrónicos para cumplir con requisitos de cumplimiento.
- Almacenamiento de copias de seguridad y recuperación.
- Almacenamiento de grabaciones de video en vigilancia.

3.4.3. Almacenamiento distribuido

El almacenamiento distribuido es un sistema de almacenamiento basado en software que posibilita el acceso a los datos en cualquier momento y lugar, restringiendo el

acceso solo a usuarios autorizados. Este sistema de gestión de volúmenes lógicos está diseñado para manejar la expansión y la disponibilidad constante de datos en un entorno de alta disponibilidad (HA). Además, posee la capacidad de identificar y responder a fallos y ciberataques. En lugar de seguir la arquitectura tradicional de tres capas, se adopta un enfoque de sistema de archivos distribuido, donde los datos se almacenan en clústeres de nodos de almacenamiento distribuidos geográficamente. El sistema de almacenamiento también incorpora funciones que aseguran la sincronización y coordinación de datos entre los nodos del clúster (*¿Qué Es El Almacenamiento Distribuido? Explore La Cloud Distribuida*, n.d.). Algunos ejemplos de soluciones de almacenamiento distribuido son:

- **Cephfs:** es una completa solución de almacenamiento que dispone de un sistema de archivos propio, el Ceph File System (CephFS). Con Ceph pueden almacenarse datos distribuidos en diversos componentes de la propia red. Además, los datos pueden guardarse en diferentes soportes físicos de almacenamiento. Ceph garantiza una gran flexibilidad en la elección del soporte de almacenamiento, así como una gran escalabilidad (Ionos, 2020).
- **GlusterFS** es un sistema de archivos distribuido y escalable conectado a la red que combina el espacio de almacenamiento de varios servidores en un solo sistema. Funciona bajo un modelo de cliente-servidor, donde los servidores albergan "bricks" de almacenamiento y ejecutan el Daemon glusterfsd (Ionos, 2020).
- **Amazon S3:** Amazon Simple Storage Service (Amazon S3) es un servicio de almacenamiento de objetos que ofrece escalabilidad, disponibilidad de datos, seguridad y rendimiento, puede utilizarse para proteger y almacenar datos, sin importar su cantidad. Además, es de gran utilidad para casos de uso como sitios web, copia de seguridad, entre otros (Amazon, n.d.).
- **Microsoft Azure Blob Storage:** es la solución de almacenamiento de Microsoft, la cual está optimizada para almacenar cantidades masivas de datos, es ideal tanto para el acceso distribuido así como también para copias de seguridad y restauración, recuperación ante desastres y archivado (Microsoft, 2023).

4. Metodología de Investigación

La metodología de investigación es un proceso organizado y detallado que busca generar conocimiento a través de la resolución de problemas. Surge como respuesta a las necesidades que requieren de un método para dirigir eficientemente los procesos y obtener resultados que ayuden a interpretación y análisis de casos de estudio, además que su enfoque en investigación guía la búsqueda de resultados significativos. (Ortero-Ortega, 2018).

La finalidad está centrada en medir los conocimientos acerca de la nube de los docentes de las diferentes escuelas de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador en el contexto actual, para determinar la factibilidad de desarrollar un prototipo de nube comunitaria basada en regiones, dicha medición se realizó a partir de los resultados obtenidos de implementar un método cuantitativo, esto nos permite obtener información descriptiva expresada en términos numéricos con el objetivo de crear una nube comunitaria basada en regiones.

4.1. Método cuantitativo

La metodología cuantitativa involucra la medición y la cuantificación de variables para establecer patrones, identificar relaciones y generalizar resultados (Creswell, 2014).

En el contexto de la investigación se optó por el método cuantitativo debido a su capacidad para asignar valores a las variables clave en los modelos de recolección de datos utilizados. Esta elección facilitó la obtención de información descriptiva expresada en términos numéricos, crucial para el caso de estudio en cuestión.

4.2. Población y muestreo

El muestreo se define como el procedimiento por el cual, de un conjunto de unidades que forman parte del objeto de estudio (Población), se elige un número reducido de unidades (Muestra) aplicando criterios que permitan generalizar los resultados obtenidos del estudio de la muestra de toda la población (Corbetta, 2007).

Por tanto, para el desarrollo de la investigación se determinó la población a considerar en este caso, que incluye a todos los docentes de las diferentes escuelas de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador el cual asciende a 180 docentes (Ministerio de Hacienda, El Salvador, 2023), ya que son quienes forman parte del grupo de interés en esta investigación.

Para la selección de la muestra representativa, se optó por elegir a doce docentes de manera aleatoria entre aquellos de diversas escuelas que impartieron clases al equipo de investigación. Estos docentes representativos formarán parte de la muestra sobre la cual se basará el análisis para determinar la viabilidad del prototipo de nube comunitaria en la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador.

4.3. Instrumentos de investigación

Para llevar a cabo un estudio de diagnóstico integral de una población específica en el ámbito de la infraestructura en la nube, se optó por la aplicación de instrumentos de investigación, destacando la encuesta como una elección estratégica sobre el cuestionario, fundamentada en consideraciones retrospectivas.

La preferencia por la encuesta radicó en su capacidad para recopilar datos de manera eficiente y abarcativa. Al desplegarse en modelos cuantitativos, la encuesta proporcionó una base numérica sólida, esencial para el análisis detallado y la identificación de patrones sobre conocimientos en la infraestructura en la nube.

La encuesta, como instrumento principal de investigación destacó por su capacidad para capturar datos cuantificables de la muestra, permitiendo generalizaciones y ofreciendo una perspectiva representativa de la población en cuestión. En este contexto y sobre otros instrumentos de investigación, conceptualización como metodologías y estrategias, la encuesta se eligió con miras a la consecución de objetivos específicos en la resolución del caso de estudio.

4.3.1 Encuesta

La encuesta presentada se enfocó en temas relacionados a nuestra facultad y su potencial uso de la tecnología de la nube. A través de una serie de preguntas, se buscó recopilar información esencial que permitiera una mejor comprensión de las necesidades y preocupaciones de la comunidad académica en las escuelas de la facultad en relación con la nube y sus regiones.

Las preguntas se diseñaron cuidadosamente para ser claras y directas, evitando sesgos y asegurando que los encuestados pudieran comprender fácilmente el contenido. El objetivo era obtener respuestas sinceras y precisas, y la secuencia de preguntas se planificó estratégicamente para seguir un flujo lógico de temas relacionados. Esto permitió explorar la percepción de los encuestados sobre la tecnología de la nube, su nivel de confianza en ciertos aspectos y las posibles barreras que podrían encontrar en su adopción.

Además, se optó por llevar a cabo esta encuesta por correo electrónico con formularios de Microsoft de forma anónima, una opción económica y conveniente para llegar de una forma más sencilla al grupo de participantes y obtener respuestas sinceras debido al anonimato. Sin embargo, esto conlleva ciertos desafíos, como la desconfianza hacia las encuestas en línea y la posibilidad de una alta tasa de no respuesta. Por lo tanto, se incluyeron recomendaciones para mitigar estos problemas, como el envío de un correo presentando la intención de la encuesta y la importancia de esta para nuestro proyecto.

5. Resultados

5.1. Resultados cuantitativos

1. ¿Está familiarizado con el término "la nube"?



Se puede observar que el 100% de los encuestados están familiarizados con el término de la nube, es decir, que tienen una base de conocimiento general sobre el tema.

2. Si su respuesta anterior fue "Sí", ¿Sabe usted qué es la "Nube comunitaria"?



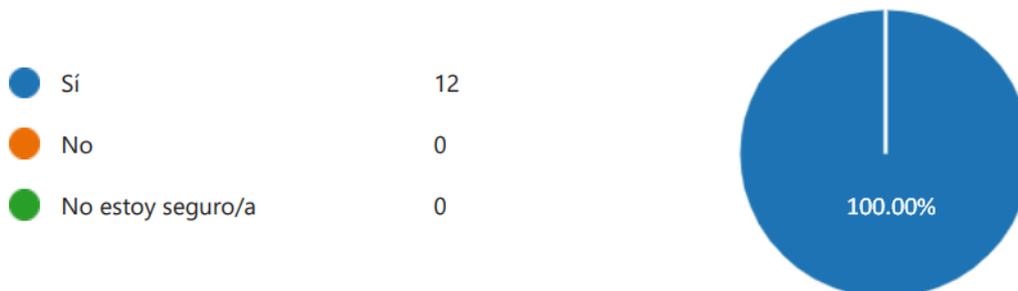
Se observa que la mitad de los encuestados están familiarizados con el concepto de "Nube comunitaria", mientras que un tercio tiene ciertas dudas al respecto y un pequeño porcentaje no tiene conocimiento completo sobre el tema. Esto indica la necesidad de educación o clarificación sobre este término específico en futuras comunicaciones o materiales informativos.

3. ¿Sabe usted dónde se almacena la información en "la nube"?



Los resultados indican que hay cierta confusión entre los encuestados sobre dónde se almacena la información en la nube. La mitad comprende que se almacena en servidores virtuales, mientras que un 33.33% cree que está en servidores físicos y un 16.66% no lo sabe, reflejando una falta de entendimiento y la necesidad de educar a las personas sobre la naturaleza virtual de la infraestructura de la nube, para mejorar la comprensión general sobre el almacenamiento de datos en línea.

4. ¿Sabe usted qué es una máquina virtual?



Se puede observar que todos los encuestados están familiarizados con el concepto de máquina virtual. Esto sugiere que los ingenieros de las diferentes escuelas de la facultad de ingeniería y arquitectura tienen conocimiento acerca de las máquinas virtuales.

5. Si es así, ¿Cuál de las siguientes opciones describe mejor su nivel de confianza en la seguridad de las máquinas virtuales?



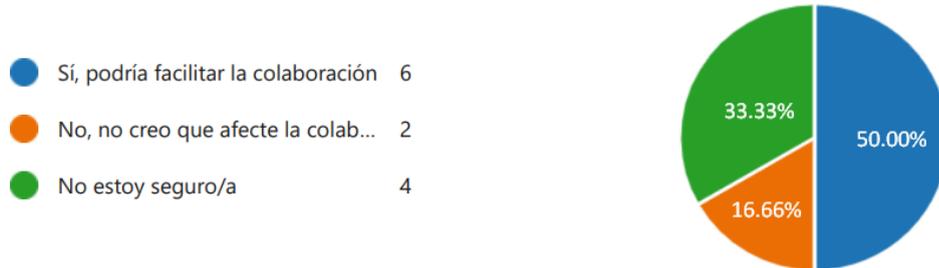
Se puede observar que el 83% de los encuestados, siendo esto una gran mayoría, confía en las máquinas virtuales, por lo tanto podríamos determinar que estarían dispuestos a usarlas, y esto no sería un impedimento para llevar a cabo el proyecto.

6. ¿Está familiarizado con el término "región" cuando se habla de "la nube"?



Se puede observar que la mitad de los encuestados conocen que es una región, esto sugiere que hay un nivel variable de conocimiento sobre este concepto en particular entre los ingenieros encuestados, se puede determinar que aunque el 33% no está seguro del término, pueden tener una noción por lo que la mayoría de los encuestados no desconoce el término, esto facilita la puesta en marcha del proyecto.

7. Si su respuesta anterior fue "sí", responda: ¿Cree usted que si cada facultad es una región dentro de una nube comunitaria podría beneficiar la colaboración entre escuelas de la facultad?



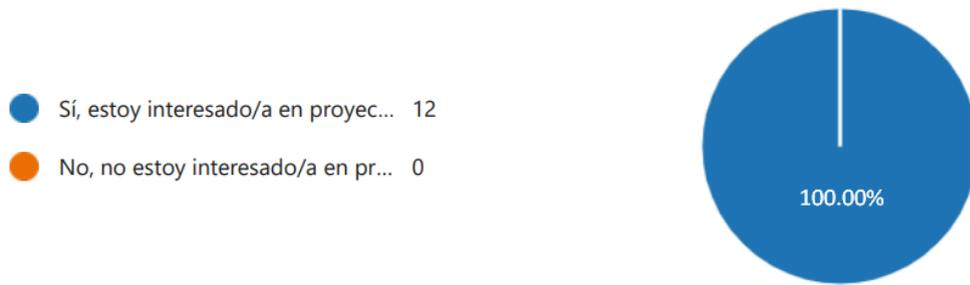
Se observa que la mitad de los encuestados creen que si cada facultad es una región dentro de una nube comunitaria puede beneficiar la colaboración entre escuelas, mientras que un 16.66% de los encuestados no creen que afecte en la colaboración y un 33.33% creen que no afecta en la colaboración.

8. ¿Considera usted que la curva de aprendizaje sobre el manejo de máquinas virtuales es un impedimento para realizar su trabajo del día a día?



Se puede observar que el 16.66% de los encuestados considera que la curva de aprendizaje sobre el manejo de máquinas virtuales es un impedimento para realizar el trabajo cotidiano, mientras que un 83.33% considera que la curva de aprendizaje en el manejo de máquinas virtuales no es un impedimento para su trabajo cotidiano.

9. ¿Existen proyectos que podría realizar en conjunto con otras escuelas de la facultad?



Se observa que el 100% de los encuestados están interesados en proyectos que se podrían realizar en conjunto con otras escuelas de la facultad.

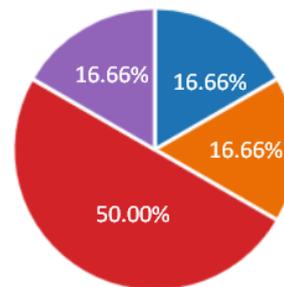
10. ¿La conexión a Internet y la electricidad son estables en la facultad?



Se puede concluir que un 33.33% de los encuestados reportaron que tanto la conexión a Internet como la electricidad son estables en la facultad. Otro 33.33% indicaron que la conexión a Internet es estable, pero la electricidad no lo es. Mientras que un 33.33% restante opinaron que ni la conexión a Internet ni la electricidad son estables en la facultad. Por lo tanto, existe una distribución equitativa de respuestas en cuanto a la estabilidad de la conexión a Internet y la electricidad en la facultad.

11. ¿Cree que algunas de las siguientes opciones serían un impedimento para utilizar regiones en la nube para la facultad?

Falta de presupuesto	2
Falta de capacitación en tecnolo...	2
Preocupaciones de seguridad d...	0
Todas las anteriores	6
Ninguna de las anteriores	2



Se puede concluir que un 16.66% de los encuestados consideran que la falta de presupuesto sería un impedimento para utilizar regiones en la nube en la facultad. Otro 16.66% mencionan que la falta de capacitación en tecnología de la nube sería un obstáculo. Un 50% opinaron que todas las opciones anteriores serían un impedimento para utilizar regiones en la nube para la facultad, y el 16.66% restante creen que ninguna de las opciones mencionadas sería un obstáculo.

5.2. Conclusión de los resultados

Se ha identificado que el 100% de los encuestados posee un conocimiento general sobre la nube, lo cual constituye una base sólida y positiva para futuros proyectos. No obstante, se ha detectado cierta confusión en relación con conceptos específicos, como la "nube comunitaria" y la ubicación del almacenamiento en la nube, lo que sugiere la necesidad de brindar información adicional y educación en estos aspectos.

La mayoría de los encuestados confía en las máquinas virtuales y muestra un alto interés en colaborar en proyectos con otras escuelas. Además, la percepción de la estabilidad de la conexión a Internet y la electricidad en la facultad se distribuye equitativamente. Aunque existen preocupaciones relacionadas con el presupuesto y la capacitación en tecnología de la nube, a pesar de esto, los resultados proporcionan una base sólida y un panorama favorable para la implementación de proyectos en la nube.

6. Factibilidad del Proyecto

La evaluación de factibilidad desempeña un papel fundamental en el proceso de toma de decisiones estratégicas cuando se trata de la construcción de una infraestructura en la nube. En este contexto, se está llevando a cabo una comparación entre OpenStack y otras soluciones líderes en el mercado, como AWS, Azure y Google Cloud, para determinar cuál de estas ofrece la mejor combinación de características y beneficios para su implementación.

Para lograrlo, es esencial analizar los componentes específicos de OpenStack diseñados para una infraestructura en la nube comunitaria. Asimismo, se deben comparar estos componentes con las alternativas disponibles en AWS, Azure y Google Cloud. El proceso de evaluación involucra considerar factores clave como escalabilidad, seguridad, usabilidad, flexibilidad, entre otros. Cada uno de estos aspectos desempeña un papel crucial en la determinación de la plataforma más adecuada para garantizar el éxito y la eficiencia de la infraestructura en la nube.

Tabla 6.1

Principales componentes de los diferentes proveedores de nube

Servicio	OpenStack	AWS (Amazon Web Services)	Azure (Microsoft Azure)	GCP (Google Cloud Platform)
Servicio de Computación	Nova	Amazon EC2	Azure Virtual Machines	Google Compute Engine
Redes	Neutron	Amazon VPC	Azure Virtual Network	Google VPC
Gestión de Identidad	Keystone	AWS Identity and Access Management (IAM)	Azure Active Directory	Google Cloud Identity and Access Management (IAM)
Colocación de Recursos	Placement	AWS Resource Groups	Azure Resource Manager	Google Cloud Resource Manager
Almacenamiento de Bloques	Cinder	Amazon EBS	Azure Disks	Google Persistent Disks

Almacenamiento de Objetos	Swift	Amazon S3	Azure Blob Storage	Google Cloud Storage
Panel de Control	Horizon	AWS Management Console	Azure Portal	Google Cloud Console
Servicio de Imágenes	Glance	Amazon Machine Images (AMIs)	Azure Managed Images	Google Cloud Images

Además, cuando se trata de elegir una plataforma de infraestructura en la nube, es necesario considerar una serie de factores cruciales. OpenStack, AWS (Amazon Web Services), Azure de Microsoft y Google Cloud son opciones populares en el mundo de la informática en la nube. Cada una de estas plataformas tiene sus propias características y ventajas únicas, lo que hace que la elección sea una decisión importante para las empresas y organizaciones.

A continuación, se presenta una comparación detallada de varios aspectos clave de estas plataformas, incluyendo su modelo de despliegue, flexibilidad, usabilidad, escalabilidad, seguridad y comunidad de soporte.

Tabla 6.2
Aspectos generales de los diferentes proveedores de nube

Aspecto	OpenStack	AWS	Azure	Google Cloud
Modelo de Despliegue	Código abierto, desplegable en la infraestructura local o en la nube pública.	Nube pública con opción de entornos híbridos y privados.	Nube pública con enfoque en entornos híbridos y soluciones empresariales.	Nube pública con opciones de despliegue en múltiples regiones y modelos.
Flexibilidad	Altamente personalizable y adecuado para casos de uso específicos, pero requiere más configuración y gestión.	Menos personalizable en comparación con OpenStack, pero más rápido de implementar.	Personalizable y apto para casos de uso específicos.	Ofrece un equilibrio entre la personalización y facilidad de uso.

Usabilidad	Mayor conocimiento técnico y experiencia en la gestión de la infraestructura.	Interfaz de usuario intuitiva y amplia documentación.	Interfaz de usuario amigable y herramientas de administración robustas.	Interfaz de usuario intuitiva y documentación detallada.
Escalabilidad	Limitada por recursos disponibles localmente.	Prácticamente ilimitada en la nube pública.	Prácticamente ilimitada en la nube pública.	Prácticamente ilimitada en la nube pública.
Seguridad	Depende de la implementación y la configuración local.	Ofrece sólidas medidas de seguridad y cumplimiento.	Ofrece sólidas medidas de seguridad y cumplimiento.	Ofrece sólidas medidas de seguridad y cumplimiento.
Comunidad y Soporte	Comunidad activa de usuarios y desarrolladores de código abierto, con opciones de soporte comercial.	Amplia comunidad de usuarios y sólido soporte técnico.	Amplia comunidad de usuarios y sólido soporte técnico.	Amplia comunidad de usuarios y sólido soporte técnico.

OpenStack ofrece un modelo de despliegue altamente personalizable, lo que significa que las comunidades pueden adaptar la infraestructura para satisfacer sus requisitos exactos. Al permitir la integración de componentes específicos y la configuración detallada, OpenStack brinda a las comunidades la capacidad de diseñar soluciones de nube que sean óptimas para sus necesidades particulares. Esto se traduce en un mayor control y autonomía sobre la infraestructura de nube, lo cual es esencial para una nube comunitaria.

Otro aspecto relevante es la escalabilidad. En una infraestructura de nube comunitaria, es vital contar con la capacidad de escalar recursos según la demanda. OpenStack, aunque puede requerir más esfuerzo de configuración inicial en comparación con los proveedores de nube pública, ofrece la flexibilidad para expandir los recursos de manera eficiente. Esto se vuelve especialmente importante a medida que las comunidades crecen y evolucionan con el tiempo.

La seguridad es otro factor crítico en la elección de una plataforma de nube comunitaria basada en regiones. Dado que la infraestructura de nube comunitaria almacena y procesa datos sensibles, la seguridad es una prioridad. OpenStack, al ser altamente personalizable, permite a las comunidades implementar medidas de seguridad específicas que se adapten a sus requisitos y políticas de seguridad. Esto garantiza un mayor grado de confidencialidad e integridad de los datos.

Además, OpenStack se beneficia de una comunidad activa de usuarios y desarrolladores de código abierto. Esto proporciona acceso a un vasto conocimiento colectivo y recursos compartidos, lo que puede ser invaluable para las comunidades que buscan soluciones en constante mejora y desarrollo.

6.1. Factibilidad económica

Los costos asociados con servicios en la nube pueden experimentar variaciones considerables, dependiendo de diversos factores. Entre ellos se encuentran:

- La especificación y cantidad de servidores que se pretenden implementar.
- El volumen de datos que se requiere almacenar y transferir.
- La elección del sistema operativo y software a ser implementados.
- La modalidad de pago, ya sea por minuto, hora o mes.
- La elección entre pago por uso, instancias reservadas o contratos a largo plazo.
- La ubicación del centro de datos en la nube necesario.

Estos son solo algunos aspectos superficiales, ya que proveedores destacados como AWS, Azure y Google Cloud Platform ofrecen una amplia gama de servicios, cada uno con su propio esquema de precios y diversas opciones de configuración que influyen en el costo.

Para facilitar la comprensión de los costos asociados con la implementación en la nube, hemos comparado a los principales proveedores de nube, como AWS, Azure y Google Cloud Platform.

Para comparar el precio de implementar la región en nubes públicas se ha utilizado la página web www.cloudorado.com la cual nos permite seleccionar especificaciones como potencia del CPU, memoria RAM y almacenamiento para las instancias, según las necesidades descritas en la siguiente tabla:

Tabla 6.3

Costo estimado mensual para la elaboración del prototipo para una región en los diferentes proveedores de nube

Nodo	Cantidad	CPU	RAM (MB)	Disco (GB)	Google Cloud	Amazon Web Services	Microsoft Azure
Pfsense	1	1	700	1 x 20	\$13.75	\$33.68	\$12.96
Deployment	1	4	2048	1 x 25	\$73.47	\$127.40	\$114.93
Control Plane	1	12	16896	1 x 75	\$240.49	\$499.60	\$552.96
Compute	2	8	5120	1 x 75	\$311.36	\$509.60	\$481.32
Ceph	3	1	700	1 x 50	\$44.85	\$110.04	\$42.93
				1 x 100	\$50.85	\$125.04	\$49.68
Total					\$734.77	\$1,405.36	\$1,254.78

Una vez conocidos los costos de implementar el prototipo en la nube pública, se procede a estimar los costos para implementar una región en servidores on-promises partiendo de la base que todos los servidores serán nuevos y que la universidad ya cuenta con espacios para colocarlos. Se necesitarán servidores con las características descritas en la tabla 6.4.

Un servidor será utilizado para instalar el firewall pfSense, el siguiente será para el nodo deployment el cual servirá para desplegar OpenStack Ansible (OSA), el control plane ejecutará los servicios de infraestructura de OpenStack es por esto que es el más exigente en poder de cómputo y memoria RAM, dos servidores estarán destinados a los nodos

compute los cuales serán instalados en diferentes lugares físicos, ya que funcionarán para las zonas de disponibilidad y por último, tres servidores destinados para el cluster Ceph el cual proveerá Block Storage.

Tabla 6.4

Costo estimado para la elaboración del prototipo para una región

Nodo	Cantidad	CPU	RAM (GB)	Disco	Modelo	Precio
Pfsense	1	4	8	1 TB	PowerEdge R250	\$2,037.90
Deployment	1	4	8	1 TB	PowerEdge R250	\$2,037.90
Control Plane	1	12	16	480 GB	PowerEdge R350	\$4,251.00
Compute	2	12	16	480 GB	PowerEdge R350	\$8,502.00
Ceph	3	4	8	1 TB 1.92 TB	PowerEdge R350	\$20,548.68
Total						\$37,377.48

En base a los costos mensuales necesarios para lanzar el prototipo de una región en nubes públicas y el costo para lanzarlo con servidores propios, se procede a realizar el cálculo del costo del prototipo para 5 años asumiendo que las mensualidades de las nubes públicas no cambia y no se necesita de más servicios, también se asume la contratación de una persona que se encargará de administrar los servidores físicos en \$661.50 mensuales (Costo estimado de Ministerio de Hacienda, El Salvador, 2023 y de ofertas de empleo para puestos similares).

Tabla 6.5

Comparativa de precios del prototipo para una región

Proveedor	Año 1	Año 2	Año 3	Año 4	Año 5
Google Cloud Platform	\$8,817.24	\$17,634.48	\$26,451.72	\$35,268.96	\$44,086.20
Amazon Web Services	\$16,864.32	\$33,728.64	\$50,592.96	\$67,457.28	\$84,321.60

Microsoft Azure	\$15,057.36	\$30,114.72	\$45,172.08	\$60,229.44	\$75,286.80
Servidores físicos	\$45,315.48	\$53,253.48	\$61,191.48	\$69,129.48	\$77,067.48

Teniendo en cuenta lo planteado en la tabla 6.5 se puede observar que en las nubes públicas desde el año 1 hasta el año 5 son notablemente más económicas que los servidores físicos a excepción de Amazon Web Services, siendo Google Cloud Platform la nube pública más económica para desplegar el prototipo de una región, sin embargo, se debe tomar en cuenta las ventajas que ofrecen los servidores físicos a cambio de su costo elevado, como por ejemplo:

- Control y personalización: ofrecen un mayor nivel de control y personalización sobre la infraestructura en comparación con los proveedores de nubes públicas, en el caso que se necesiten requisitos específicos o configuraciones para adaptarse a las necesidades de manera precisa.
- Independencia del proveedor: no se está atado a un proveedor específico lo cual puede ser beneficioso en términos de evitar posibles bloqueos de proveedores.
- Seguridad y privacidad: Algunas organizaciones tienen requisitos estrictos en cuanto a la seguridad y privacidad de los datos. También utilizar servidores físicos puede permitir un mayor control sobre la seguridad y el manejo de datos sensibles.

7. Desarrollo del Prototipo

7.1. Herramientas utilizadas

Con el propósito de establecer un entorno en la nube, se ha identificado una alternativa destacada con OpenStack que se compone de diversos servicios fundamentales instalados de forma independiente, los servicios operan de manera conjunta, lo que les permite adaptarse a los requerimientos específicos en la nube, los cuales incluyen funciones tales como Cómputo, Redes, Imágenes, Almacenamiento y Base de datos.

7.1.1. *Kernel-based Virtual Machine (KVM)*

KVM en su infraestructura incluye:

La arquitectura de KVM y sus componentes en conjunto se adaptan a las demandas del prototipo, además que ofrecen un entorno de virtualización robusto y eficiente.

Extensión del kernel: Permite una integración directa con el kernel del sistema operativo Linux, el cual tiene y proporciona una base sólida para la virtualización, dando como resultado un rendimiento eficiente y seguro.

Libvirt: Esta API actúa como un medio de comunicación entre el sistema operativo anfitrión y las máquinas virtuales, facilitando la gestión y su control.

Qemu: Emulador para gestionar las máquinas virtuales utilizadas en la infraestructura diseñada, asegurando la ejecución eficiente y eficaz.

VirtIO Drivers: Se integran con el hipervisor, permitiendo la virtualización y mejorando la eficiencia del rendimiento de las máquinas virtuales, garantizando la flexibilidad y rendimiento necesarios para la implementación exitosa de una infraestructura en la nube comunitaria con regiones.

7.1.2. *Ubuntu Server*

En el marco de este proyecto, la adopción de KVM es fundamental para realizar el despliegue y la ejecución de máquinas virtuales, por lo tanto, se optó por la versión 20.04 LTS de Ubuntu como sistema operativo anfitrión para las máquinas virtuales porque ofrece una alta compatibilidad con OpenStack Ansible.

7.1.3. Ceph

Ceph se escogió dado que permite el almacenamiento de datos en diversos medios físicos, brindando así una flexibilidad en la selección de soportes de almacenamiento, junto con su capacidad de escalabilidad.

Para que Ceph funcione de manera óptima, se han utilizado múltiples ordenadores que forman un cluster, cada uno de ellos denominado nodo. En nuestro prototipo no solo proporciona una solución robusta para el almacenamiento distribuido, sino que también garantiza la gestión eficaz de instancias, volúmenes e imágenes.

7.1.4. OpenStack

Abarca todos los aspectos cruciales para la construcción eficiente del prototipo de región para la nube comunitaria, ya que facilita la orquestación de recursos al permitir la definición y automatización de flujos de trabajo para gestionar máquinas virtuales.

7.1.5. Ansible

Facilita la orquestación de los diversos componentes de la infraestructura en el prototipo de nube comunitaria, contribuye a mejorar el despliegue y la escalabilidad reduciendo así los tiempos de configuración y provee la capacidad de adaptarse rápidamente a cambios en la infraestructura.

7.1.6. Tailscale (VPN)

Como servicio de VPN (Red Privada Virtual) proporciona una conexión segura entre los diferentes componentes utilizados en la infraestructura, facilitando el acceso remoto lo cual es esencial para equipos distribuidos y operaciones descentralizadas.

7.1.7. Pfsense

Es una sólida solución de firewall, que se ha incorporado a la región de la nube para proporcionar una capa de seguridad adicional. La elección de PfSense se fundamenta en su capacidad para gestionar el tráfico de red de manera segura, segmentar la red y ofrecer una protección robusta. Gracias a sus funciones de VPN, se han establecido conexiones

seguras entre los distintos componentes de la infraestructura, asegurando así la confidencialidad e integridad de las comunicaciones en el entorno de nube comunitaria. La integración de PfSense con OpenStack permitió aprovechar de manera conjunta las capacidades inherentes de ambas plataformas.

7.2. Diseño

Se ha diseñado un prototipo que tiene como objetivo probar una región en la nube comunitaria basada en la plataforma OpenStack, que ofrezca servicios de almacenamiento, red y cómputo, y se integre con la infraestructura existente. En este diseño, se emplean máquinas virtuales que son creadas y ejecutadas mediante el hipervisor KVM, y utilizan un sistema operativo basado en la distribución GNU/Linux Ubuntu 20.04. En estas máquinas virtuales se ha instalado y configurado OpenStack, una plataforma de tecnología de código abierto que aprovecha recursos virtuales para la creación y gestión de nubes tanto privadas como públicas. OpenStack consta de varios componentes que ofrecen interfaces de programación de aplicaciones (API) para acceder a los recursos de la infraestructura.

Algunos de los servicios implementados a través de estos componentes incluyen:

- **Keystone:** Este componente se encarga de autenticar y autorizar todos los servicios dentro de OpenStack.
- **Placement:** Este componente se encarga de gestionar la asignación y seguimiento de recursos de cómputo, garantizando una distribución eficiente de recursos en las máquinas virtuales y su óptimo rendimiento.
- **Glance:** Su tarea principal consiste en almacenar y recuperar imágenes de disco de máquinas virtuales desde diversas ubicaciones, facilitando la gestión de imágenes y su disponibilidad.
- **Neutron:** Su función principal es facilitar la conectividad de redes entre los diferentes servicios, asegurando una comunicación eficiente.

- **Nova:** Este servicio se despliega como una herramienta integral de gestión y acceso para los recursos de cómputo. Se encarga de supervisar la planificación, creación y eliminación de recursos.
- **Cinder:** Se utiliza para ofrecer almacenamiento persistente de bloques, permitiendo la gestión de almacenamiento de datos de manera eficaz.

En el siguiente diagrama se presentan de forma gráfica los servicios necesarios para la creación de una región.

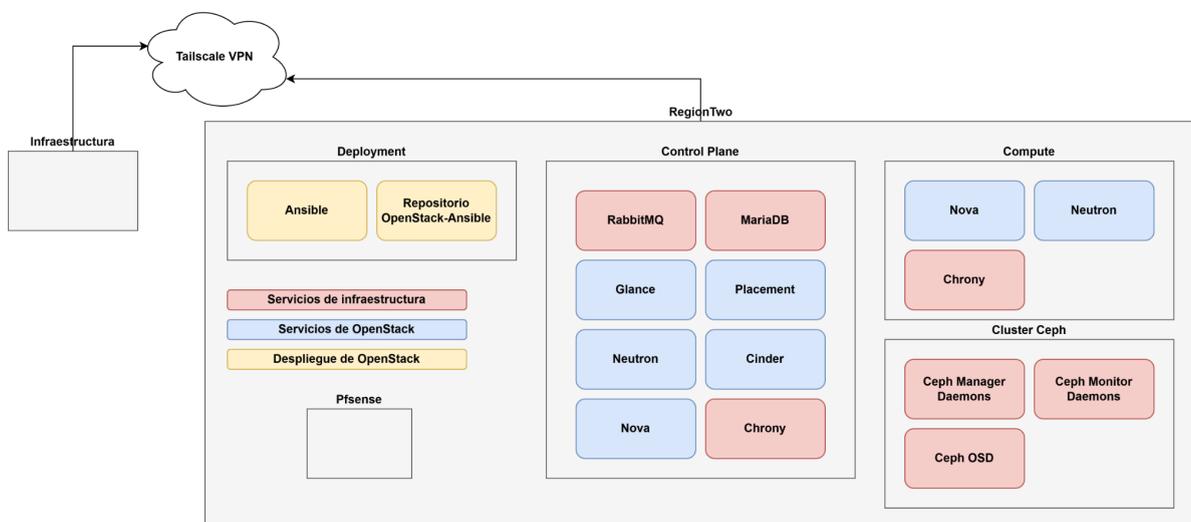


Figura 9: Diagrama de región con sus componentes y servicios.

7.3. Construcción

7.3.1. Infraestructura

El despliegue de la infraestructura requiere únicamente de un nodo con las siguientes características:

Tabla 7.1

Recursos de hardware para la máquina virtual de la infraestructura

Nodo	CPU	RAM (MB)	Disco (GB)	NIC	OS
Controller	4	8192	1 x 100	1	Ubuntu Server 20.04

Los componentes instalados en el nodo Controller son los siguientes:

- Keystone
- MariaDB
- RabbitMQ
- Memcached
- Horizon

7.3.2. Regiones

Para la construcción del prototipo se utilizaron ocho máquinas virtuales, las cuales están distribuidas de la siguiente manera:

- **Nodo pfSense:** Se utiliza como un firewall y enrutador para las redes internas de OpenStack.
- **Nodo Deployment:** Se utiliza para desplegar OpenStack Ansible (OSA).
- **Nodo Control Plane:** Se encarga de la administración, el monitoreo y la toma de decisiones para dirigir el tráfico y las operaciones en la red. Es responsable de coordinar las acciones de otros nodos.
- **Dos Nodos Compute:** Se utiliza para el procesamiento de datos y tareas de cómputo, además de ejecutar aplicaciones y cargas de trabajo específicas, como máquinas virtuales.
- **Tres Nodos de Ceph (mon, mgr, osd):**
 - **Mon (Monitor):** Supervisan y administran el clúster Ceph, manteniendo un registro de los estados.
 - **Mgr (Manager):** Proporcionan servicios de gestión y monitoreo avanzados para el clúster Ceph.
 - **OSD (Object Storage Daemon):** Almacenan los datos reales y administran su distribución y replicación en el clúster Ceph. Son esenciales para el almacenamiento y la recuperación de datos de manera confiable.

Tabla 7.2

Recursos de hardware para las máquinas virtuales de la región del prototipo

Nodo	CPU	RAM (MB)	Disco (GB)	NIC	OS
Pfsense	1	700	1 x 20	2	FreeBSD 13.1
Deployment	4	2048	1 x 25	1	Ubuntu Server 20.04
Control Plane	12	16896	1 x 75	2	Ubuntu Server 20.04
Compute	8	5120	1 x 75	2	Ubuntu Server 20.04
Ceph	1	700	1 x 50 1 x 100	2	Ubuntu Server 20.04

Los servicios instalados necesarios en cada uno de los nodos para el correcto funcionamiento del prototipo son los siguientes:

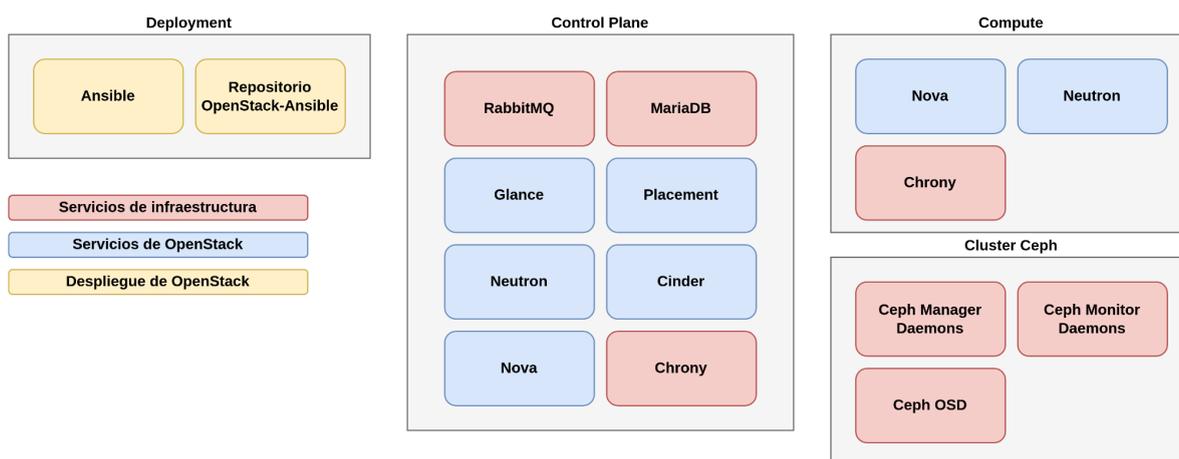


Figura 10: Nodos a utilizar para el despliegue y funcionamiento de una región.

A continuación, se definen las redes utilizadas para el funcionamiento del prototipo:

- **VLAN para la administración de OpenStack:** se encarga de la administración y la comunicación entre la infraestructura y los servicios de OpenStack que se ejecutan en contenedores o en hardware físico y es la que provee el acceso a internet en las máquinas virtuales.
- **VLAN para el tunnel VXLAN:** proporciona conectividad entre los hosts con el propósito de encapsular el tráfico usando VXLAN.

- **VLAN para storage:** ofrece acceso segregado al almacenamiento de bloques desde los servicios de OpenStack, como Cinder y Glance.
- **VLAN para red interna de las instancias en OpenStack:** ofrece una red interna para las instancias de OpenStack.
- **Red para comunicación entre máquina host y máquina virtual:** red proveída por el router de nuestro hogar. Se utiliza únicamente como método de acceso a las máquinas virtuales, su configuración es opcional.

Tabla 7.3

Configuración de redes empleadas para el despliegue de regiones con OpenStack Ansible (OSA)

Nombre de la red	Tipo	Red		ID
		Región Tres	Región Dos	
Management	VLAN	172.29.236.0/22	172.30.236.0/22	10
Tunnel	VLAN	172.29.240.0/22	172.30.240.0/22	30
Storage	VLAN	172.29.244.0/22	172.30.244.0/22	20
VLAN (provider)	VLAN	172.29.248.0/22	172.30.248.0/22	40
NAT (provider)	FLAT	192.168.123.0/22	192.168.122.0/22	-
LAN	FLAT	192.168.1.0/22	192.168.100.0/22	-

La asignación de direcciones IP es un paso fundamental en la configuración de una red, ya que desempeña un papel esencial en la comunicación y la identificación de dispositivos, las direcciones ip asignadas a la región tres son las siguientes:

Tabla 7.4

Direccionamiento de direcciones ip empleadas en la región tres.

Nodo	Management	Tunnel	Storage	VLAN
Pfsense	172.29.236.1	172.29.240.1	172.29.244.1	172.29.248.1
Deployment	172.29.236.2	-	-	-
Control Plane	172.29.236.11	172.29.240.11	172.29.244.11	172.29.248.11
Compute1	172.29.236.21	172.29.240.21	172.29.244.21	172.29.248.21
Compute2	172.29.236.22	172.29.240.22	172.29.244.22	172.29.248.22

Ceph1	172.29.236.18	-	172.29.244.18	-
Ceph2	172.29.236.19	-	172.29.244.19	-
Ceph3	172.29.236.20	-	172.29.244.20	-

La configuración detallada del nodo es tratada en la sección [Diseño de Bajo Nivel](#) de los Anexos.

7.4. Pruebas

Para el caso de estudio se tiene una infraestructura con dos regiones las cuales representan dos diferentes escuelas de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador, como demostración la región tres representa la Escuela de Ingeniería Civil, mientras que la región dos representa la Escuela de Ingeniería de Sistemas Informáticos, se ha creado el dominio “hidráulica-civil” el cuál representa el Departamento de Hidráulica de la Escuela de Ingeniería Civil, dicho departamento desea crear un proyecto de “pagina_web_hidraulica_civil” en colaboración con la Escuela de Ingeniería de Sistemas Informáticos. Para dicha colaboración se ha establecido que una instancia será lanzada en el dominio de “desarrollo-sistemas” en la región de la Escuela de Ingeniería de Sistemas Informáticos y ésta contendrá la base de datos de la página web, mientras que la segunda instancia que contendrá el sitio web en Wordpress será lanzada en el dominio de “hidráulica-civil” en la región de la Escuela de Ingeniería Civil, ambas tendrán que tener acceso dos usuarios, uno de la Escuela de Ingeniería Civil y otro de la Escuela de Sistemas Informáticos, éstos podrán acceder vía SSH o por medio de VNC.

7.4.1. Funcionamiento de OpenStack

- Login de Dashboard Horizon de OpenStack con campos de entrada como el usuario, contraseña y dominio, para este caso se debe de ingresar con el usuario administrador.

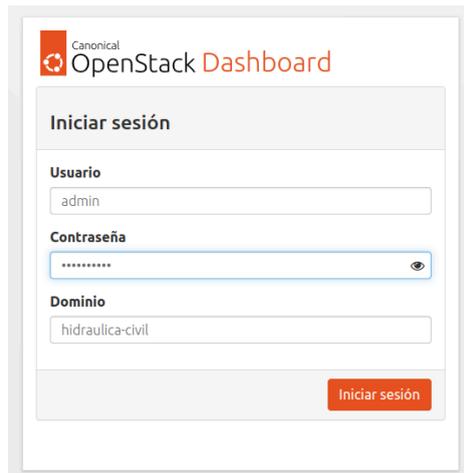


Figura 11: Vista general del dashboard de Openstack.

- Vista general del Dashboard Horizon para el usuario administrador, la cual muestra el nombre del dominio, proyecto y región en la que se encuentra el usuario, además de un vistazo a los recursos como el almacenamiento, instancias y redes dentro del proyecto de la región.

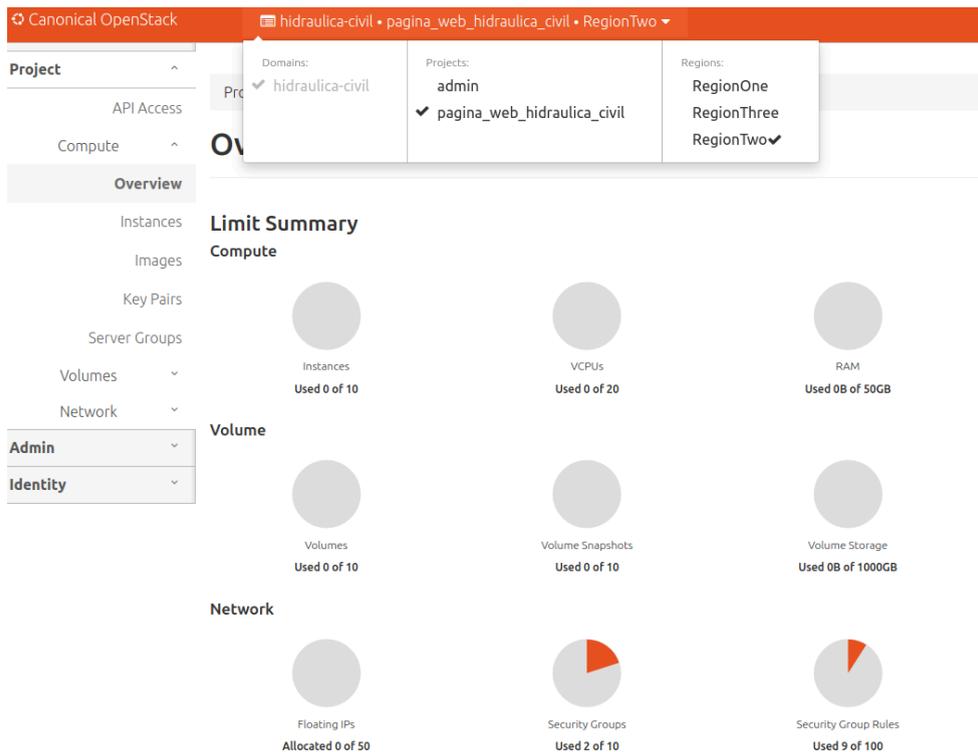


Figura 12: Resumen de la región del usuario administrador.

- Vista general de los usuarios asociados al proyecto llamado pagina_web_hidraulica_civil dentro de la región en la que se encuentra el usuario.

pagina_web_hidraulica_civil

Overview Users **Groups**

Displaying 5 items

User Name	Description	Email	User ID
sm16021	-		0b9f89b1b31d4ecf9
ac17026	-		3246bfdc8be14d6dl
admin	-		b18551467b954a0c
ta17002	-		c4ec876ce9cc4d0ba
pp16011	-		e108a5c192ec40b18

Displaying 5 items

Figura 13: Vista general de los usuarios pertenecientes al proyecto.

- Vista de la región dos, en donde se proporciona información como el propietario de las imágenes creadas, el tipo y el estado actual.

Images

Key Pairs

Server Groups

Images

Work

Displaying 3 items

<input type="checkbox"/>	Owner	Name ^	Type	Status
<input type="checkbox"/>	> admin	Cirros RegionTwo	Image	Active
<input type="checkbox"/>	> pagina_web_hidraulica_civil	Debian 12	Image	Active
<input type="checkbox"/>	> pagina_web_hidraulica_civil	MariaDB Template	Snapshot	Active

Displaying 3 items

Figura 14: Vista general de las imágenes de Openstack.

- Sabores que han sido agregados a la región dos, la cual muestra la cantidad de VCPUs, RAM y tamaño del disco de cada sabor disponible en la región.

Flavors

Displaying 2 items

<input type="checkbox"/>	Flavor Name	VCPUs	RAM	Root Disk	Ephemeral Disk
<input type="checkbox"/>	m1.tiny	1	512MB	1GB	0GB
<input type="checkbox"/>	m1.tiny-debian	1	512MB	5GB	0GB

Displaying 2 items

Figura 15: Vista general de los sabores Openstack.

- Redes asociadas al proyecto, incluyendo detalles específicos como el nombre de la red, subred configurada con una dirección IP.

Network ^

Networks

Routers

Floating IPs

RBAC Policies

Displaying 1 item

<input type="checkbox"/>	Project	Network Name	Subnets Associated	DHCP Agents
<input type="checkbox"/>	-	provider-flat	provider-subnet-flat 192.168.122.0/24	1

Displaying 1 item

Figura 16: Vista general de las redes de Openstack.

- En el apartado de Instancias, se muestran las que pertenecen al proyecto, detallando el nombre de la instancia, nombre de la imagen, la dirección IP y el sabor asociado a dicha instancia.

Instances

Images

Key Pairs

Server Groups

imes v

work v

Displaying 1 item

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Flavor
<input type="checkbox"/>	mariadb	MariaDB Template	192.168.122.84	m1.tiny-debian

Displaying 1 item

Figura 17: Vista general de las instancias Openstack.

- Verificación de forma visual para validar la configuración y el estado de la instancia por medio de VNC.

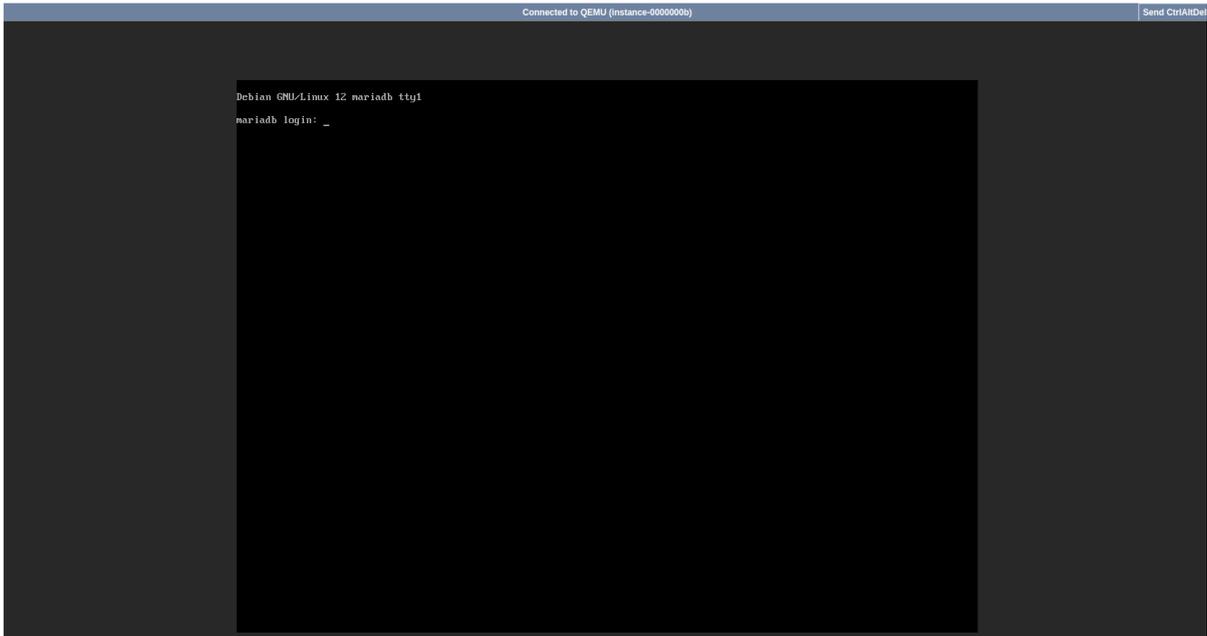


Figura 18: Vista general de la instancia en el VNC.

- Ejecución de prueba de conexión a internet desde la instancia en el VNC, lo cual garantiza la conectividad adecuada y la accesibilidad a recursos externos desde la instancia.

```
Debian GNU/Linux 12 mariadb tty1
mariadb login: debian
Password:
Linux mariadb 6.1.0-13-and64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.55-1 (2023-09-29) x86_64

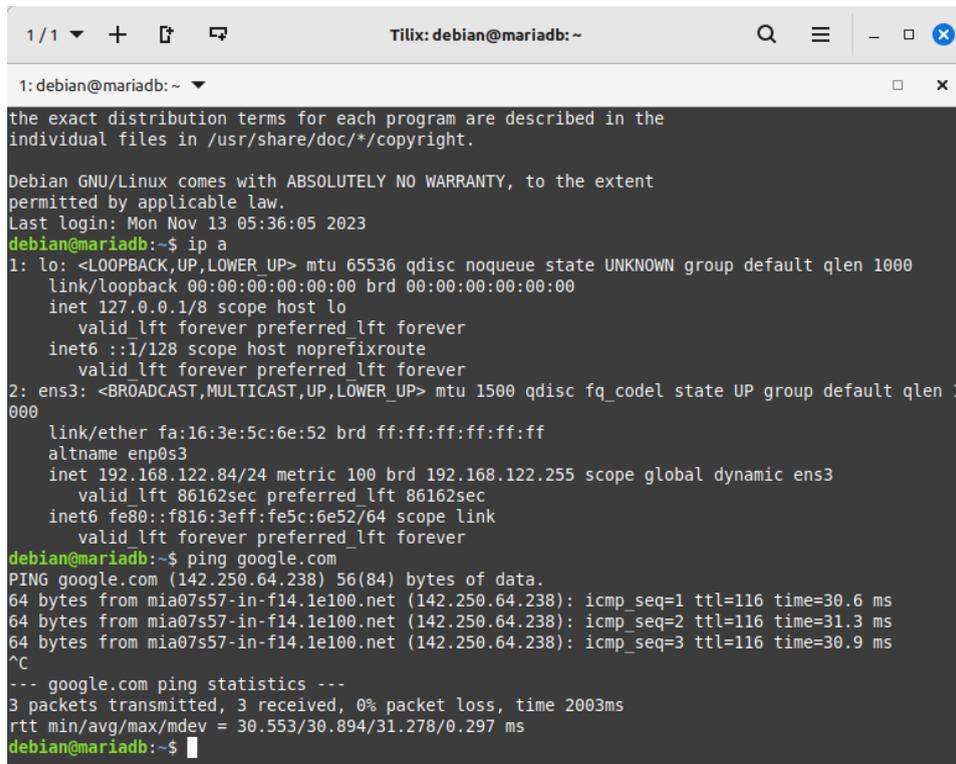
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Nov 3 23:18:48 UTC 2023 from 192.168.122.1 on pts/0
debian@mariadb:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether fa:16:3e:5c:6e:52 brd ff:ff:ff:ff:ff:ff
    altname emp0s3
    inet 192.168.122.84/24 metric 100 brd 192.168.122.255 scope global dynamic ens3
        valid_lft 86266sec preferred_lft 86266sec
    inet6 fe80::f816:3eff:fe5c:6e52/64 scope link
        valid_lft forever preferred_lft forever
debian@mariadb:~$ ping google.com
PING google.com (142.250.64.238) 56(84) bytes of data:
64 bytes from mia07s57-in-f14.1e100.net (142.250.64.238): icmp_seq=1 ttl=116 time=31.3 ms
64 bytes from mia07s57-in-f14.1e100.net (142.250.64.238): icmp_seq=2 ttl=116 time=31.2 ms
64 bytes from mia07s57-in-f14.1e100.net (142.250.64.238): icmp_seq=3 ttl=116 time=30.9 ms
64 bytes from mia07s57-in-f14.1e100.net (142.250.64.238): icmp_seq=4 ttl=116 time=30.7 ms
^C
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 30.743/31.053/31.321/0.231 ms
debian@mariadb:~$ _
```

Figura 19: Conexión a internet desde el VNC.

- Se realiza una prueba de conectividad a la instancia por medio del protocolo SSH desde la máquina host, utilizando la IP 192.168.122.84 de la máquina

remota. Además se visualiza las interfaces de red que posee la instancia y se verifica la conectividad hacia internet, mediante un ping a google.com.



```
1/1 + [ ] [ ] Tilix: debian@mariadb: ~
1: debian@mariadb: ~
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Nov 13 05:36:05 2023
debian@mariadb:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid lft forever preferred lft forever
    inet6 ::1/128 scope host noprefixroute
        valid lft forever preferred lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether fa:16:3e:5c:6e:52 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 192.168.122.84/24 metric 100 brd 192.168.122.255 scope global dynamic ens3
        valid lft 86162sec preferred lft 86162sec
    inet6 fe80::f816:3eff:fe5c:6e52/64 scope link
        valid lft forever preferred_lft forever
debian@mariadb:~$ ping google.com
PING google.com (142.250.64.238) 56(84) bytes of data:
64 bytes from mia07s57-in-f14.1e100.net (142.250.64.238): icmp_seq=1 ttl=116 time=30.6 ms
64 bytes from mia07s57-in-f14.1e100.net (142.250.64.238): icmp_seq=2 ttl=116 time=31.3 ms
64 bytes from mia07s57-in-f14.1e100.net (142.250.64.238): icmp_seq=3 ttl=116 time=30.9 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 30.553/30.894/31.278/0.297 ms
debian@mariadb:~$
```

Figura 20: Conexión a la instancia mediante SSH.

- Visualización de las zonas de disponibilidad creadas para el desarrollo del prototipo: nova (zona creada por defecto), planta alta y planta baja. Además de los host aggregates creados por zona de disponibilidad y los compute que están utilizando cada una de estas.

Host Aggregates Filter

Displaying 2 items

<input type="checkbox"/> Name	Availability Zone	Hosts	Metadata
<input type="checkbox"/> host-aggregate-planta-alta	planta-alta	• compute2	• availability_zone = planta-alta
<input type="checkbox"/> host-aggregate-planta-baja	planta-baja	• compute	• availability_zone = planta-baja

Displaying 2 items

Availability Zones

Displaying 4 items

Availability Zone Name	Hosts
internal	• infra1-nova-api-container-f435cf7c (Services Up)
nova	
planta-alta	• compute2 (Services Up)
planta-baja	• compute (Services Up)

Displaying 4 items

Figura 21: Zonas de disponibilidad y host aggregates.

- Se muestra la sección para realizar migración en vivo y poder reubicar una instancia de una zona de disponibilidad a otra, en caso de cualquier intervención o fallo.

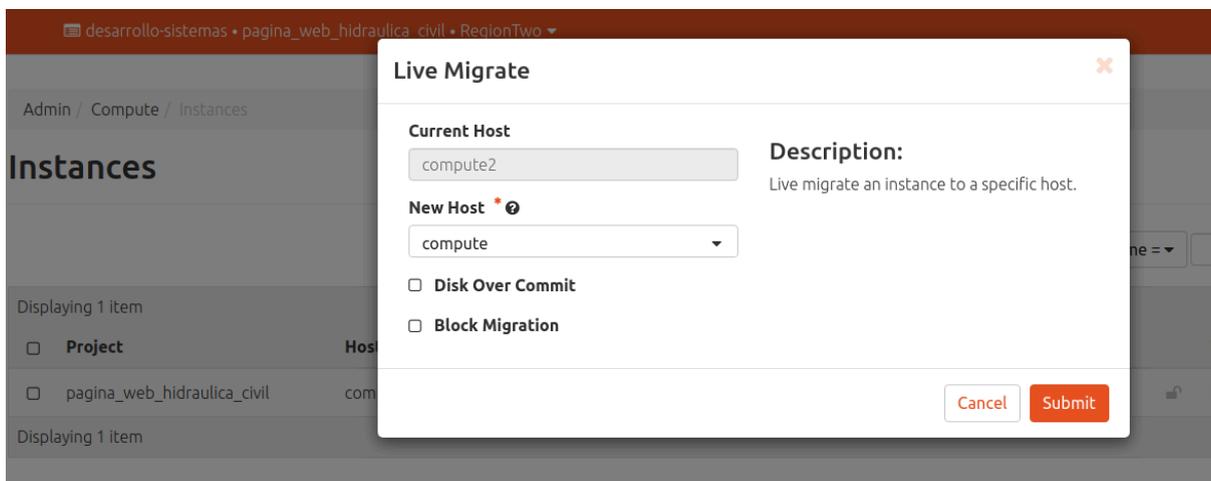


Figura 22: Ventana de migración en vivo.

7.4.2. Funcionamiento de Ceph

- Inicio de Sesión para ingresar al Dashboard de Ceph



Figura 23: Pantalla de inicio de sesión de Ceph.

- Se muestra el Dashboard de Ceph, donde aparece información y estadísticas sobre el clúster de Ceph y un monitoreo sobre su salud.

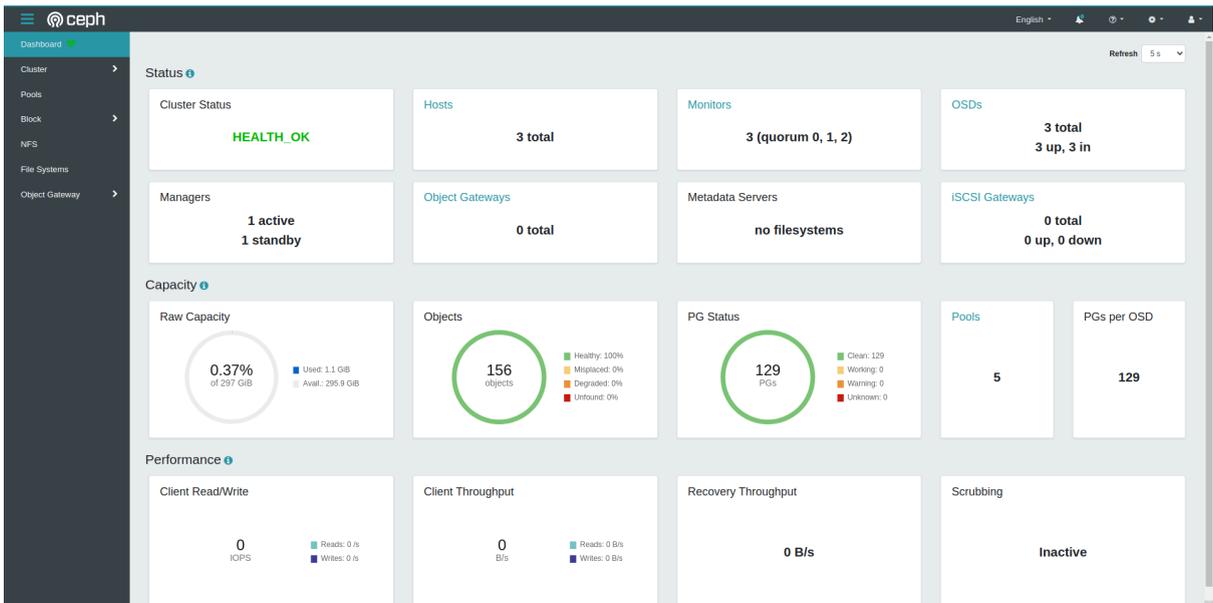


Figura 24: Dashboard de Ceph.

- Listado de los diferentes OSDs que conforman el cluster Ceph

Cluster > OSDs

OSDs List Overall Performance

+ Create - Cluster-wide configuration

ID	Host	Status	Device class	PGs	Size	Flags	Usage	Read bytes	Write bytes	Read ops	Write ops
> 0	ceph1	In Up	hdd	129	99 GB		0%	0 /s	0 /s
> 1	ceph2	In Up	hdd	129	99 GB		0%	0 /s	0 /s
> 2	ceph3	In Up	hdd	129	99 GB		0%	0 /s	0 /s

0 selected / 3 total

Figura 25: Listado de OSDs.

- Listado de los diferentes Pools utilizados para el desarrollo del prototipo

Cluster > Pools

Pools List Overall Performance

+ Create -

Name	Data Protection	Applications	PG Status	Usage	Read bytes	Write bytes	Read ops	Write ops
> backups	replica:3	ibfd	32 active+clean	0%	0 /s	0 /s
> device_health_metrics	replica:3	mgr, devicehealth	1 active+clean	0%	0 /s	0 /s
> images	replica:3	ibfd	32 active+clean	0.00%	0 /s	0 /s
> vms	replica:3	ibfd	32 active+clean	0.00%	0 /s	0 /s
> volumes	replica:3	ibfd	32 active+clean	0%	0 /s	0 /s

0 selected / 5 total

Figura 26: Listado de pools.

- Se muestran las diferentes imágenes e instancias creadas en una región dentro de Openstack y que son guardadas automáticamente en el cluster Ceph

Name	Pool	Namespace	Size	Objects	Object size	Provisioned	Total provisioned	Parent	Mirroring
> 27a3a1ae-9332-4bfc-81b3-4b01ce0d48b5	images		12.1 MiB	2	8 MiB	0 B	12.1 MiB	-	journal
> 9994cad5-bbea-4064-b6e1-e5cd3a1cbe1_disk	vms		1 GiB	256	4 MiB	532 MiB	532 MiB	-	journal
> 9994cad5-bbea-4064-b6e1-e5cd3a1cbe1_disk.co	vms		482 KiB	1	4 MiB	482 KiB	482 KiB	-	journal
> bde8787c-50b0-4882-800b-76f19539e0ee_disk	vms		1 GiB	256	4 MiB	532 MiB	532 MiB	-	journal
> bde8787c-50b0-4882-800b-76f19539e0ee_disk.co	vms		482 KiB	1	4 MiB	482 KiB	482 KiB	-	journal
> volume-41bad577-76c4-4bbe-ae99-3cc2628bc46c	volumes		1 GiB	128	8 MiB	0 B	0 B	-	journal

0 selected / 6 found / 6 total

Figura 27: Imágenes e instancias.

8. Caso de Uso para el Prototipo de Región

Prototipo de Regiones para la formación de una Nube Comunitaria para las Escuelas De La Facultad De Ingeniería Y Arquitectura de la Universidad de El Salvador

Se planteó la necesidad de crear una infraestructura en la nube basada en OpenStack para satisfacer las necesidades de computación, almacenamiento y red de las escuelas de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador. Existen siete escuelas: Escuela de Ingeniería en Sistemas Informáticos, Escuela de Ingeniería Industrial, Escuela de Ingeniería Civil, Escuela de Ingeniería Mecánica, Escuela de Ingeniería Eléctrica, Escuela de Arquitectura y Escuela de Ingeniería Química e Ingeniería de Alimentos donde cada una representa una región, y por cuestiones de recursos solo se implementaron dos regiones (Escuela de Ingeniería en Sistemas Informáticos y Escuela de Ingeniería Civil).

El objetivo principal de este caso de uso es proporcionar a la facultad un prototipo de nube comunitaria basada en regiones, que pueda ser implementado en un futuro en una infraestructura centralizada, para que los docentes que estén interesados en realizar proyectos conjuntamente entre distintas escuelas ya sea para compartir información, recursos y herramientas como WordPress, MySQL, entre otros, y que estos puedan ser accedidos de forma remota a través de un protocolo seguro como SSH.

Para la autenticación en el prototipo se hace uso del servicio Keystone el cual es proveído por OpenStack, en el cual se valida la identidad de los usuarios, garantizando que solo aquellos con las credenciales adecuadas accedan a los recursos. La autorización, por su parte, se encarga de definir y administrar los roles, estableciendo quién puede acceder a qué funcionalidades y datos. Además se cuenta con una capa extra de seguridad con el firewall de Pfsense para brindar seguridad a la red.

Cada región se conecta a una única infraestructura central a través de una API, para que pueda brindar recursos como máquinas virtuales, redes y almacenamiento. Las regiones se implementan utilizando OpenStack Ansible (OSA) y se integran servicios de almacenamiento con Ceph para garantizar la gestión segura de los datos.

Además, se integra la capacidad de realizar migraciones en vivo lo que ofrece ventajas significativas en términos de flexibilidad y eficiencia operativa. Con la capacidad de trasladar máquinas virtuales de un host a otro sin afectar su disponibilidad, logrando así la posibilidad de llevar a cabo actualizaciones de hardware, mantenimientos planificados o incluso ajustes de carga de trabajo en tiempo real, sin afectar la experiencia de usuario, además de brindar tolerancia ante cualquier imprevisto o fallo.

Al asignar recursos de manera independiente a cada escuela, se optimiza el rendimiento de la infraestructura en su conjunto. Esta solución integral se adapta no solo a las necesidades específicas de cada escuela de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador sino que también puede ser implementada en cualquier entorno que se desee poner en marcha una nube comunitaria con múltiples regiones, como los ministerios gubernamentales, hospitales, convenio de universidades, empresas, entre otros.

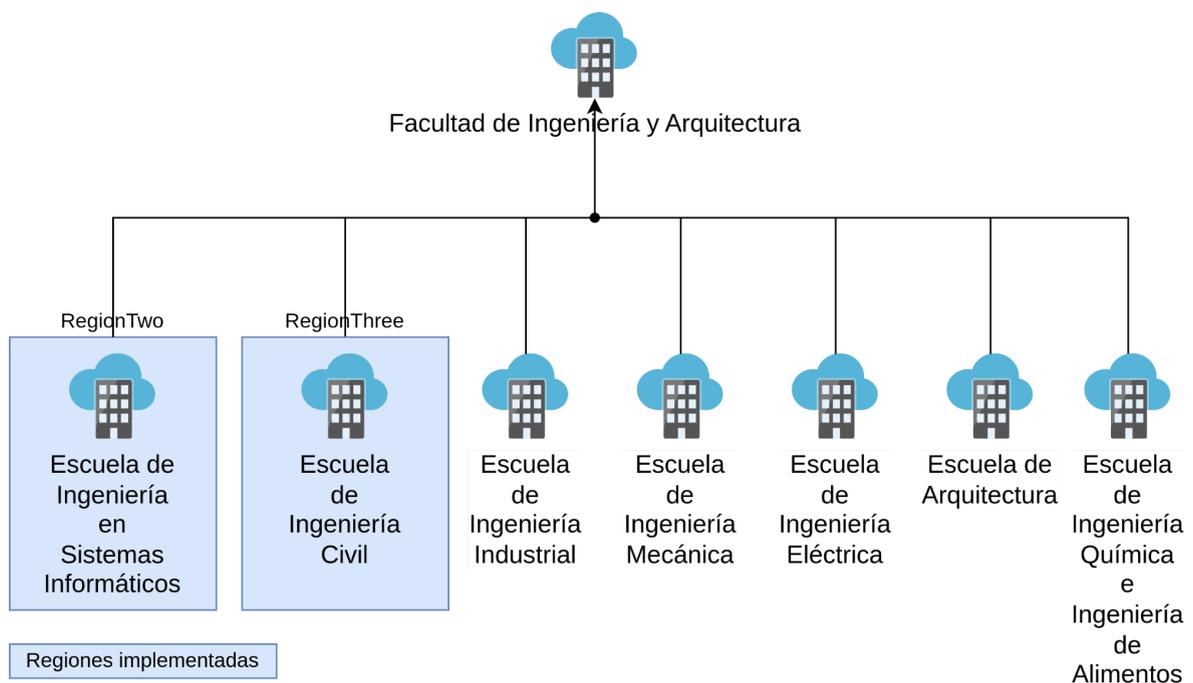


Figura 28: Diagrama de regiones para la Facultad de Ingeniería y Arquitectura.

9. Conclusiones

- La implementación de una nube comunitaria por regiones plantea desafíos técnicos para las instituciones que deseen adoptar esta solución, entre estos se encuentran la necesidad de establecer una región sólida y segura, la gestión eficiente de datos a gran escala y la garantía de la compatibilidad con sistemas existentes. Sin embargo, a pesar de estos obstáculos, las instituciones que se unen a esta nube comunitaria experimentan una serie de beneficios notables, estos incluyen un acceso compartido a recursos informáticos y servicios, lo que reduce costos operativos, impulsa la colaboración y fomenta la innovación.
- Al concentrar los servicios en una nube compartida dentro de una región específica, se facilita la cooperación entre instituciones y usuarios. Por lo que resulta en un acceso más rápido y confiable a los datos, y también fomenta el desarrollo conjunto de soluciones adaptadas a las necesidades particulares. Además, al limitar la infraestructura a una región, se pueden abordar preocupaciones de privacidad de manera más efectiva.
- El despliegue de dos regiones utilizando OpenStack Ansible (OSA) con Ceph integrado en cada una, así como su agregación a la infraestructura, resalta la viabilidad y la mejora de la eficiencia de los servicios en entornos multi-región.
- Al implementar una nube comunitaria basada en regiones, se garantiza la accesibilidad uniforme a las instancias a través de una interfaz común entre regiones, proporcionando una conectividad eficiente y segura. La configuración de red, respaldada por políticas de seguridad sólidas, garantiza la privacidad y la conectividad adecuada entre los recursos. Además, la asignación eficiente de recursos y la capacidad de escalar horizontalmente cumple con las necesidades que pueden presentarse en un entorno en constante evolución.

10. Recomendaciones

- Los empleados deben recibir formación sobre conceptos clave relacionados con la nube, prácticas de seguridad, gestión de datos y resolución de problemas técnicos. Si la capacitación es insuficiente o inadecuada, las instituciones corren el riesgo de enfrentar problemas de seguridad de datos, ineficiencias operativas y la resistencia al cambio por parte del personal. Además, la falta de conocimiento puede dar lugar a errores costosos y a un aprovechamiento insuficiente de los beneficios de la región unida a la nube comunitaria.
- Dada la constante evolución de la nube, se recomienda encarecidamente evaluar la compatibilidad antes de optar por nuevas versiones de las herramientas utilizadas en este proyecto. Garantizar la compatibilidad con las versiones utilizadas es esencial para aprovechar al máximo las capacidades de gestión y escalabilidad, al tiempo que se garantiza la seguridad y el rendimiento óptimo en entornos de nube regionales.
- Como organización se deben establecer políticas internas claras sobre el uso de la nube comunitaria. Es importante definir quién tiene acceso a qué datos y establecer medidas de seguridad internas para proteger la información sensible. En base a lo anterior, se recomienda implementar prácticas de seguridad sólidas para las contraseñas, es esencial que estas sean fuertes y que se cambien regularmente. Además, se deben gestionar cuidadosamente los permisos de acceso, brindando solo a las personas que realmente lo necesitan y asegurarse de revocar los privilegios de inmediato cuando ya no sean necesarios.

11. Referencias Bibliográficas

- Amazon. (s.f.). *¿Qué es Amazon S3? - Amazon Simple Storage Service*. Recuperado el 10 de noviembre de 2023, de https://docs.aws.amazon.com/es_es/AmazonS3/latest/userguide/Welcome.html
- Amazon. (s.f.). *¿Qué es la virtualización? - Explicación de la virtualización de la computación en la nube - AWS*. Amazon AWS. Recuperado el 13 de septiembre de 2023, de <https://aws.amazon.com/es/what-is/virtualization/>
- Amazon. (s.f.). *Regiones y zonas de disponibilidad de la infraestructura global*. Amazon AWS. Recuperado el 9 de septiembre de 2023, de https://aws.amazon.com/es/about-aws/global-infrastructure/regions_az/
- Atlassian. (s.f.). *¿Qué es la computación en la nube? Visión general de la nube*. Atlassian. Recuperado el 13 de septiembre de 2023, de <https://www.atlassian.com/es/microservices/cloud-computing>
- Bhowmik, S. (2017). *Cloud Computing*. Cambridge University Press.
- Carr, N. C. N. (2008). *The Big Switch: Rewiring the World, from Edison to Google*.
- Google. (n.d.). *Roles and permissions | IAM Documentation*. Google Cloud. Recuperado el 24 de septiembre de 2023, de <https://cloud.google.com/iam/docs/roles-overview>
- Google. (2023, 4 de agosto). *Geografía y regiones | Documentación*. Google Cloud. Recuperado el 9 de septiembre de 2023, de <https://cloud.google.com/docs/geography-and-regions?hl=es-419>
- Harvey, N. (2020). *97 Things Every Cloud Engineer Should Know: Collective Wisdom from the Experts* (E. Freeman & N. Harvey, Eds.). O'Reilly.
- IBM. (n.d.). *Despliegue de GlusterFS*. IBM. Recuperado el 14 de septiembre de 2023, de <https://www.ibm.com/docs/es/cloud-private/3.2.x?topic=private-glusterfs>
- Ionos. (2020, 29 de julio). *GlusterFS | Descripción del sistema de archivos distribuido*. IONOS. Recuperado el 14 de septiembre de 2023, de <https://www.ionos.es/digitalguide/servidores/know-how/que-es-glusterfs/>

Ionos. (2020, 30 de julio). *Ceph – Una práctica solución de almacenamiento para empresas de cualquier tamaño*. IONOS. Recuperado el 14 de septiembre de 2023, de <https://www.ionos.es/digitalguide/servidores/know-how/que-es-ceph/>

Law Insider. (n.d.). *Cloud User Definition*. Law Insider. Recuperado el 24 de septiembre de 2023, de <https://www.lawinsider.com/dictionary/cloud-user>

Marinescu, D. C. (2017). *Cloud Computing: Theory and Practice*. Elsevier Science.

Microsoft. (n.d.). *Global Infrastructure*. Microsoft Azure. Recuperado el 9 de septiembre de 2023, de <https://azure.microsoft.com/en-us/explore/global-infrastructure/>

Microsoft. (2023, 15 de marzo). *Introducción a Azure Storage: almacenamiento en la nube en Azure*. Microsoft Learn. Recuperado el 10 de noviembre de 2023, de <https://learn.microsoft.com/es-es/azure/storage/common/storage-introduction>

Microsoft Azure. (2023, 1 de junio). *¿Qué es el control de acceso basado en rol de Azure (RBAC)?* Microsoft Learn. Recuperado el 24 de septiembre de 2023, de <https://learn.microsoft.com/es-es/azure/role-based-access-control/overview>

Ministerio de Hacienda, El Salvador. (2023, 22 de septiembre). *Gasto de Estado*. Portal de Transparencia Fiscal de El Salvador. Recuperado el 12 de noviembre de 2023, de <https://www.transparenciafiscal.gob.sv/downloads/pdf/SalPro2024ID185.pdf>

Opennebula. (2022, 2 de octubre). *Conceptos*. Recuperado el 14 de septiembre de 2023, de https://docs.opennebula.io/6.0/overview/opennebula_concepts/opennebula_overview.html#the-opennebula-model-for-cloud-infrastructure-deployment

OpenStack. (2023, 25 de julio). *Dominios — Documentación de la Guía de Seguridad*. OpenStack Documentation. Recuperado el 24 de septiembre de 2023, de <https://docs.openstack.org/security-guide/identity/domains.html>

¿Qué es el almacenamiento distribuido? Explore la cloud distribuida. (n.d.). Nutanix. Recuperado el 14 de septiembre de 2023, de <https://www.nutanix.com/es/info/distributed-storage>

¿Qué pasaría si la nube no existiera? – Artículos y tendencias sobre soluciones tecnológicas. (2017, 22 de mayo). *Artículos y tendencias sobre soluciones*

- tecnológicas. Recuperado el 10 de noviembre de 2023, de <https://itblogsogeti.com/2017/05/22/que-pasaria-si-la-nube-no-existiera/>
- Red Hat. (n.d.). *Chapter 6. Managing projects Red Hat OpenStack Platform 16.2*. Red Hat Customer Portal. Recuperado el 24 de septiembre de 2023, de https://access.redhat.com/documentation/es-es/red_hat_openstack_platform/16.2/html/users_and_identity_management_guide/assembly_managing-projects
- Red Hat. (2021, 3 de octubre). *Temas El concepto de OpenStack. Red Hat*. Recuperado el 9 de septiembre de 2023, de <https://www.redhat.com/es/topics/openstack>
- Red Hat. (2023, 31 de enero). *¿Qué es cloud computing o computación en la nube? Red Hat*. Recuperado el 8 de septiembre de 2023, de <https://www.redhat.com/es/topics/cloud>
- Red Hat. (2023, 17 de marzo). *¿Qué es una nube híbrida? Hybrid cloud. Red Hat*. Recuperado el 8 de septiembre de 2023, de <https://www.redhat.com/es/topics/cloud-computing/what-is-hybrid-cloud>
- Red Hat. (2023, 23 de marzo). *¿Qué es un hipervisor? Hypervisor. Red Hat*. Recuperado el 13 de septiembre de 2023, de <https://www.redhat.com/es/topics/virtualization/what-is-a-hypervisor>
- Ruiz, F. J. (2016, 17 de junio). *Servidores On Premise problemas frente solución Cloud. Blog de Dataprius*. Recuperado el 10 de noviembre de 2023, de <https://blog.dataprius.com/index.php/2016/06/17/on-premise-servidores-problemas-solucion-cloud/>
- Ruparelia, N. B. (2015). *Cloud Computing*. MIT Press.
- Servidores en la nube | Qué son, tipos y ventajas para las empresas*. (n.d.). Esystems. Recuperado el 10 de noviembre de 2023, de <https://esystems.com.co/servidores-en-la-nube-ventajas/>
- VMware. (n.d.). *Documentación de VMware vSphere*. VMware Docs. Recuperado el 14 de septiembre de 2023, de <https://docs.vmware.com/es/VMware-vSphere/index.html>

What Does a Hypervisor Do? (2014, 5 de febrero). *Flexiant*. Recuperado el 13 de septiembre de 2023, de

<https://www.flexiant.com/2014/02/05/what-does-a-hypervisor-do/>

Wikipedia. (n.d.). *Opennebula*. Wikipedia. Recuperado el 14 de septiembre de 2023, de

<https://es.wikipedia.org/wiki/Opennebula>

12. Anexos

- Anexo 1. Encuesta
- Anexo 2. Diseño de Bajo Nivel

Anexo 1. Encuesta

Universidad de El Salvador

Facultad de Ingeniería y Arquitectura

Escuela de Ingeniería de Sistemas Informáticos.

Trabajo final del Curso de Especialización en Infraestructura en la Nube.

DESARROLLO DE UN PROTOTIPO ENFOCADO EN LA INTEGRACIÓN DE REGIONES EN UNA INFRAESTRUCTURA DE NUBE

Esta encuesta tiene como propósito recopilar información sobre el conocimiento de diferentes miembros de las escuelas de la Facultad de Ingeniería y Arquitectura en relación con el tema de la nube, así como para entender su interés y preocupaciones con respecto a la formación de una nube comunitaria para la facultad.

Agradecemos su participación en esta encuesta, ya que sus respuestas nos ayudarán a obtener una visión más clara de las necesidades y perspectivas de nuestra comunidad académica en cuanto a la tecnología de la nube y su potencial impacto en la colaboración entre las escuelas de la facultad. Por favor, responda las siguientes preguntas:

1. ¿Está familiarizado con el término "la nube"?
 - a. Sí
 - b. No
2. Si su respuesta anterior fue "sí", ¿Sabe usted qué es la "Nube comunitaria"?
 - a. Sí
 - b. No
 - c. No estoy seguro/a
3. ¿Sabe usted dónde se almacena la información en "la nube"?
 - a. En servidores físicos
 - b. En servidores virtuales

- c. No lo sé
4. ¿Sabe usted qué es una máquina virtual?
- a. Sí
 - b. No
 - c. No estoy seguro/a
5. Si su respuesta anterior fue "sí", ¿Cuál de las siguientes opciones describe mejor su nivel de confianza en la seguridad de las máquinas virtuales?
- a. Tengo plena confianza en la seguridad de las máquinas virtuales
 - b. No estoy seguro/a de la seguridad de las máquinas virtuales
 - c. No tengo confianza en la seguridad de las máquinas virtuales
6. ¿Está familiarizado con el término "región" cuando se habla de "la nube"?
- a. Sí
 - b. No
 - c. No estoy seguro/a
7. Si su respuesta anterior fue "sí", responda: ¿Cree usted que si cada facultad es una región dentro de una nube comunitaria podría beneficiar la colaboración entre escuelas de la facultad?
- a. Sí, podría facilitar la colaboración
 - b. No, no creo que facilite la colaboración
 - c. No estoy seguro/a
8. ¿Considera usted que la curva de aprendizaje sobre el manejo de máquinas virtuales es un impedimento para realizar su trabajo del día a día?
- a. Sí, es un impedimento
 - b. No, no es un impedimento
9. ¿Existen proyectos que podría realizar en conjunto con otras escuelas de la facultad?
- a. Sí, estoy interesado/a en proyectos colaborativos

- b. No, no estoy interesado/a en proyectos colaborativos
10. ¿La conexión a Internet y la electricidad son estables en la facultad?
- a. Sí, ambas son estables
 - b. La conexión a Internet es estable, pero la electricidad no lo es
 - c. La electricidad es estable, pero la conexión a Internet no lo es
 - d. No son estables, ni la conexión a Internet, ni la electricidad
11. ¿Cree que algunas de las siguientes opciones serían un impedimento para utilizar regiones en la nube para la facultad?
- a. Falta de presupuesto
 - b. Falta de capacitación en tecnología de la nube
 - c. Preocupaciones de seguridad de datos
 - d. Todas las anteriores
 - e. Ninguna de las anteriores

Anexo 2. Diseño de Bajo Nivel

**UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS**



**DISEÑO DE BAJO NIVEL PARA EL DESARROLLO DE PROTOTIPO ENFOCADO
EN LA INTEGRACIÓN DE REGIONES EN UNA INFRAESTRUCTURA DE NUBE**

PRESENTADO POR:

**VANESSA CRISTINA ALFARO COREAS
GERARDO WILFREDO PERALTA PINTO
FRANCISCO ALEXANDER SANTANA MÉNDEZ
DANIEL ENRIQUE TENNANT ALVARENGA**

PARA OPTAR AL TÍTULO DE:

INGENIERO DE SISTEMAS INFORMÁTICOS

índice

1. Instalación de KVM	1
2. Imágenes de Sistema Operativo a usar	3
2.1 Ubuntu Server 20.04 (Focal Fosa - LTS)	3
2.2 PfSense	3
3. Instalación y configuración de máquinas virtuales en KVM	4
3.1 Máquinas virtuales con PfSense	4
3.1.1 Creando y configurando VLANs	15
3.1.2. Reglas Firewall para VLANs	17
3.1.2.1. Reglas para Management	17
3.1.2.2. Reglas para las otras VLANs	18
3.1.3. Configuración de Tailscale	18
3.2. Máquinas virtuales con Ubuntu Server	20
4. Instalación y Configuración del Clúster de Ceph	27
4.1. Requerimientos	27
4.2. Preparando los nodos	27
4.3. Instalando cephadm	29
4.4. Iniciar un nuevo cluster	29
4.5. Habilitando ceph CLI	30
4.6. Agregando hosts adicionales al cluster	30
4.7. Agregando los OSDs	31
4.7.1. Inflando el cluster	31
4.7.2. Agregando mon daemons	31
4.8. Configuración de Pools y Usuarios en Ceph	32
4.8.1. Creación de pools	32
4.8.2. Inicialización de los pools	32
4.8.3. Generando los usuarios y keyrings	32
5. Instalación y Configuración de la región de OpenStack	34
5.1. Requerimientos	34
5.2. Configuración de redes empleadas	34
5.3. Asinacion de Ip	34
5.4. Configuración de netplan	35
5.4.1. Nodo deployment	35
5.4.2. Nodo control-plane y nodos compute	35
5.5. Preparando nodos	36
5.5.1. Configurando llaves SSH	36
5.5.2. Instalando dependencias	37
5.5.2.1. Nodo deployment	37
5.5.2.2. Nodos control plane y compute	37
5.5.2.3. Ejecutando script	37
5.6. Configurando despliegue de OpenStack Ansible	37
5.6.1. Copiando Keyrings de Ceph	40
5.6.2. Configurando la integración con Ceph	40
5.6.2.1. Personalización del entorno	40

5.6.2.2. Configuración de user_variables.yml	41
5.7. Ejecución de playbooks	42
6. Integración de las regiones a la infraestructura de nube	43
6.1. Configurando Keystone para la administración de varias regiones	43
6.2. Servicio de glance	43
6.3. Servicio de placement	43
6.4. Servicio de nova	44
6.5. Servicio de neutron	44
6.6. Servicio de cinderv3	44
6.7. Servicio Glance	45
6.8. Servicio Placement	45
6.9. Servicio Nova	45
6.9.1. Nodo Control-Plane	45
6.9.2. Nodos compute	46
6.10. Servicio Neutron	46
6.11. Servicio Cinder	46

1. Instalación de KVM

Antes de comenzar a instalar KVM en Ubuntu 20.04 LTS, es importante comprobar que el hardware sea compatible, para ello debemos ejecutar el siguiente comando desde la terminal de ubuntu:

```
egrep -c '(vmx|svm)' /proc/cpuinfo
```

Para comprobar que el hardware es compatible con KVM, el resultado debe de ser mayor a 0. Ahora bien, para comprobar que el sistema soporta KVM, se debe de ejecutar el siguiente comando:

```
sudo kvm-ok
```

Si al ejecutar el comando anterior, muestra un mensaje como “sudo: kvm-ok: orden no encontrada”, se debe a que esta utilidad no está instalada, y para poder instalarla se debe ejecutar el siguiente comando:

```
sudo apt install cpu-checker
```

Una vez ejecutado el comando se debe de presionar “S” para confirmar la descarga e instalación de esta utilidad. Una vez terminada la instalación, ejecutamos nuevamente “sudo kvm-ok” para comprobar que la utilidad ya se encuentra disponible. Después de comprobar que la utilidad está instalada y disponible, debemos de instalar KVM y sus dependencias asociadas como virt-manager y bridge-utils, para poder lograrlo se debe ejecutar el siguiente comando:

```
sudo apt install -y qemu qemu-kvm libvirt-daemon libvirt-clients bridge-utils virt-manager
```

Terminada la instalación es importante comprobar que libvirtd-daemon está corriendo con normalidad y se debe de ejecutar el siguiente comando:

```
sudo systemctl status libvirtd
```

Luego de comprobar que libvirt está activo y en ejecución, se debe de habilitar el servicio con el arranque de Ubuntu 20.04, por lo cual se debe de ejecutar el siguiente comando:

```
sudo systemctl enable --now libvirt
```

Y por último, comprobar que los módulos de KVM están ejecutándose correctamente, con el siguiente comando:

```
lsmod | grep -l kvm
```

Hasta el momento tenemos instalado KVM en Ubuntu 20.04, por lo que es importante conocer que podemos crear máquinas virtuales en modo gráfico, y a través de línea de comandos. Para este caso es necesario el modo gráfico de KVM para tener un mejor control de las máquinas virtuales creadas, por lo que se debe de ejecutar el siguiente comando:

```
sudo apt install uvtool
```

Una vez terminada la descarga, podemos acceder a la interfaz gráfica ejecutando el comando “virt-manager” desde la terminal de Ubuntu, el cual nos desplegará lo siguiente:

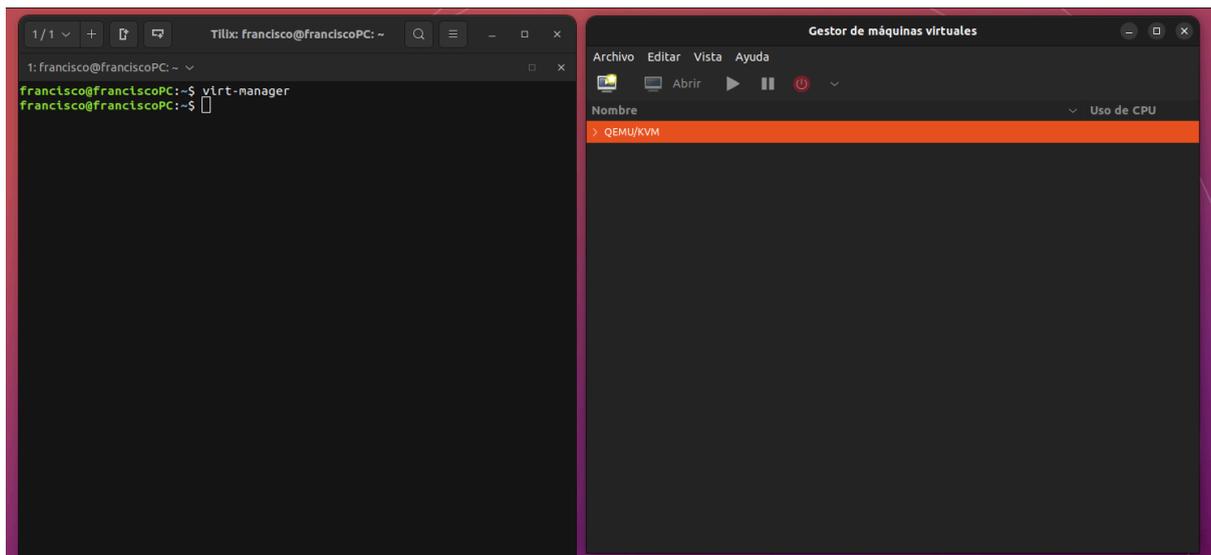


Figura 1: Interfaz de KVM.

2. Imágenes de Sistema Operativo a usar

2.1 Ubuntu Server 20.04 (Focal Fosa - LTS)

La descarga de Ubuntu 20.04 (Focal Fosa) se realiza a través del sitio web oficial

<https://releases.ubuntu.com/focal/>.



Figura 2: Opción para descargar Ubuntu Server mediante ISO.

2.2 PfSense

Se debe de visitar el sitio web oficial de PfSense <https://www.pfsense.org/download/>

.A continuación se recomiendan las siguientes opciones para su descarga:

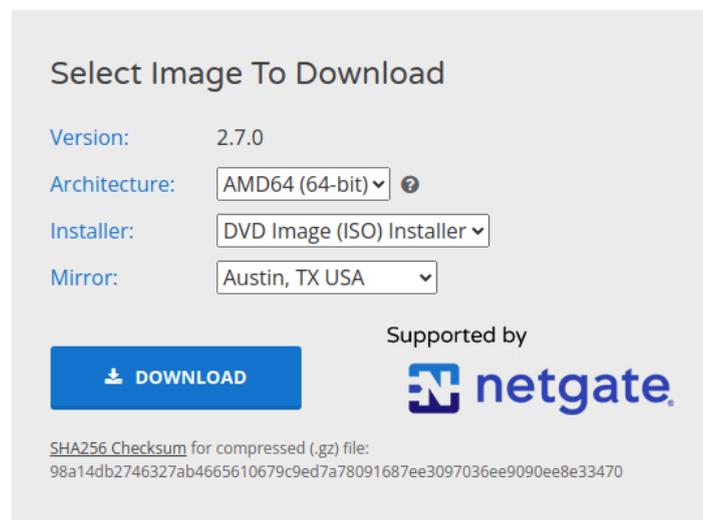


Figura 3: Opciones recomendadas para descargar PfSense.

Una vez descargada, es necesario descomprimir la imagen. Para hacerlo, se utiliza la herramienta gzip y se ejecuta el siguiente comando:

```
gzip -d pfsense-CE-2.7.0-RELEASE-amd64.iso.gz
```

3. Instalación y configuración de máquinas virtuales en KVM

3.1 Máquinas virtuales con PfSense

Es importante aclarar que la red NAT Y LAN pueden ser diferentes dependiendo del proveedor de internet. Para iniciar la creación de una nueva máquina virtual en KVM, se debe abrir la herramienta KVM. Luego, se debe hacer clic en el botón "Crear nueva máquina virtual".

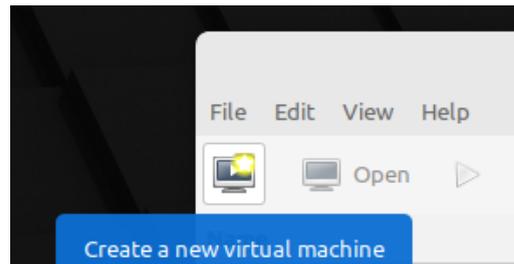


Figura 4: Opción para crear una nueva máquina virtual en KVM.

Dentro de KVM, se procede a crear una nueva máquina virtual. En las opciones disponibles, se selecciona "Imagen ISO" como la fuente de instalación.

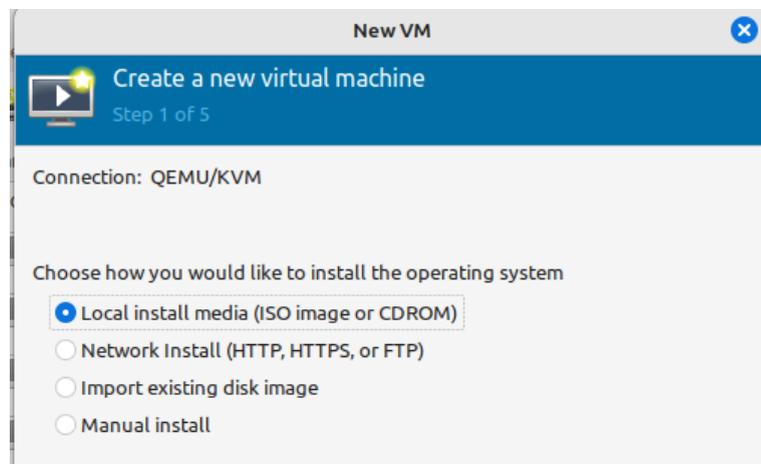


Figura 5: Opciones para crear una máquina virtual.

A continuación, se selecciona el archivo ISO previamente descomprimido para utilizarlo como la imagen de instalación. Se desactiva la detección automática y se especifica "FreeBSD 13.1" como el sistema operativo que se desea instalar.

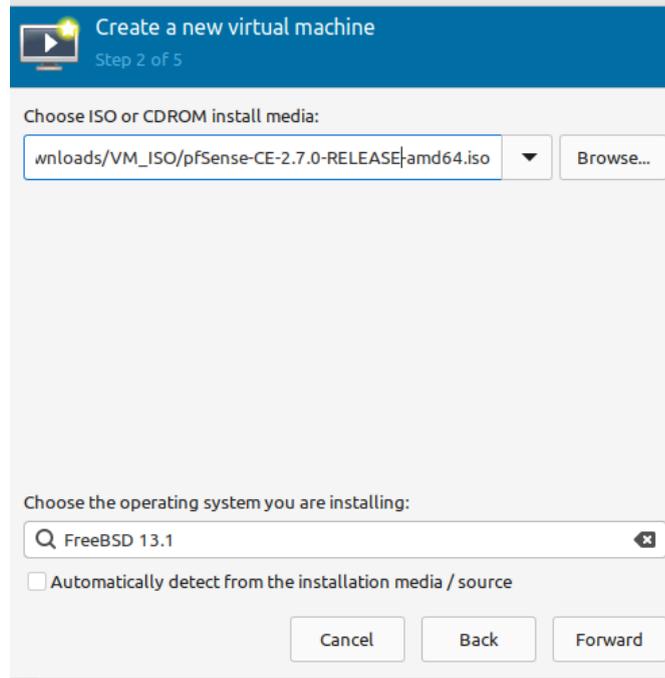


Figura 6: Formulario para seleccionar imagen en KVM.

Se especifica la cantidad de memoria RAM y la cantidad de núcleos de CPU que se asignan a la máquina virtual.

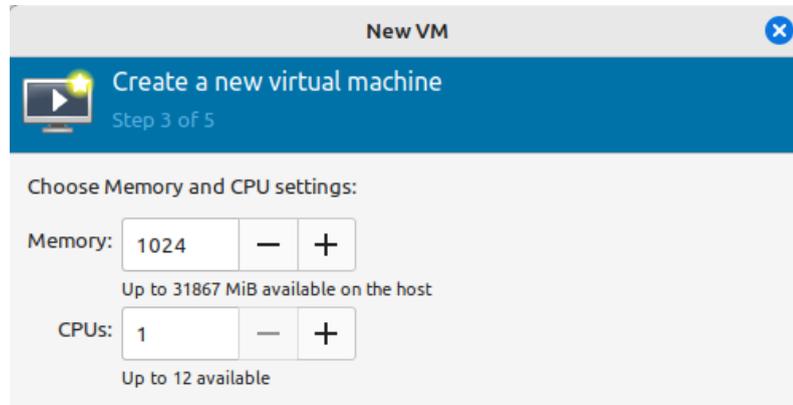


Figura 7: Opciones de memoria y CPU de la maquina virtual.

Se define el tamaño del disco que se asignará a la máquina virtual en el formulario correspondiente.

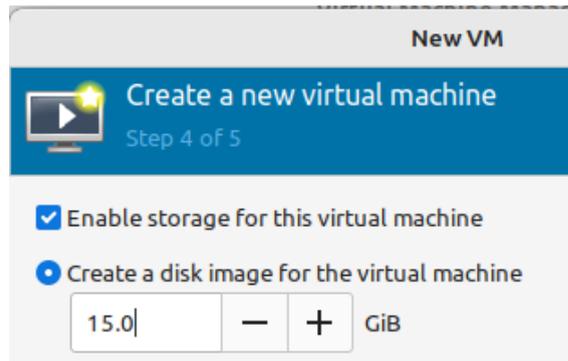


Figura 8: Tamaño de la máquina virtual.

En este punto, se elige la opción que permite personalizar la configuración antes de iniciar la instalación.

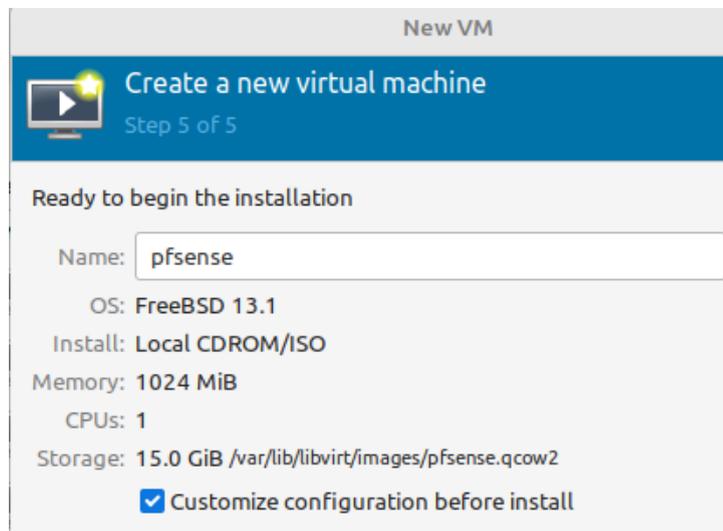


Figura 9: Formulario para el nombre de la máquina virtual.

Luego, se agrega nuevo hardware haciendo clic en el botón correspondiente.

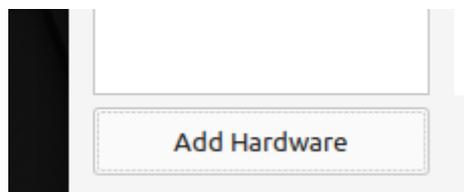


Figura 10: Opción para agregar nuevo hardware a la máquina virtual.

Se agrega una nueva interfaz de red con la configuración de red LAN, utilizando un dispositivo puente.

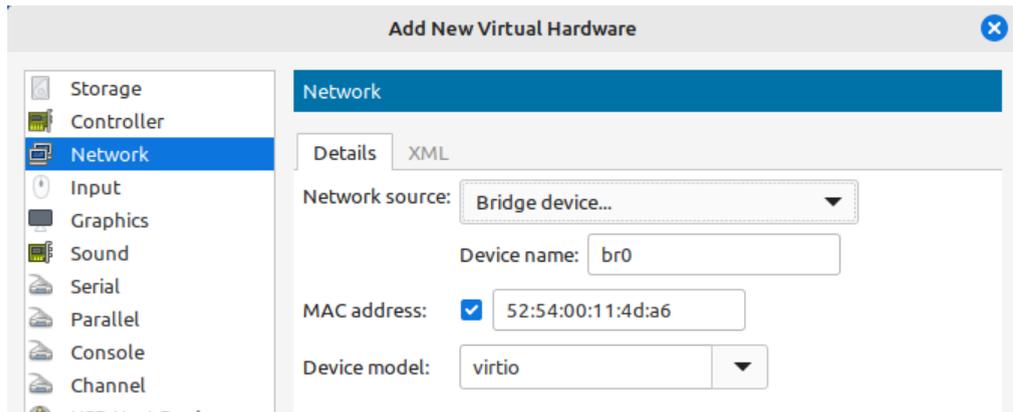


Figura 11: Opciones de configuración para la nueva interfaz de red.

Por último, se inicia el proceso de instalación haciendo clic en el botón "Iniciar instalación" y se espera unos segundos para que se realice el autoboot.

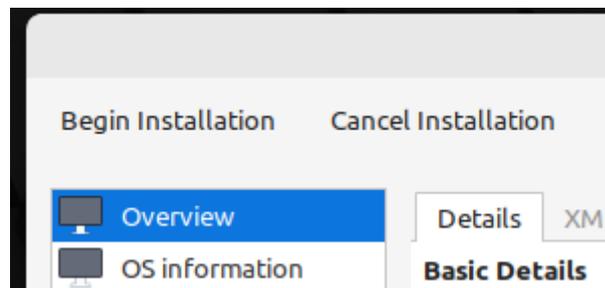


Figura 12: Opción para iniciar instalación.

Se muestra una pantalla con los derechos reservados de PfSense en la cual los aceptamos seleccionando la opción "Enter".

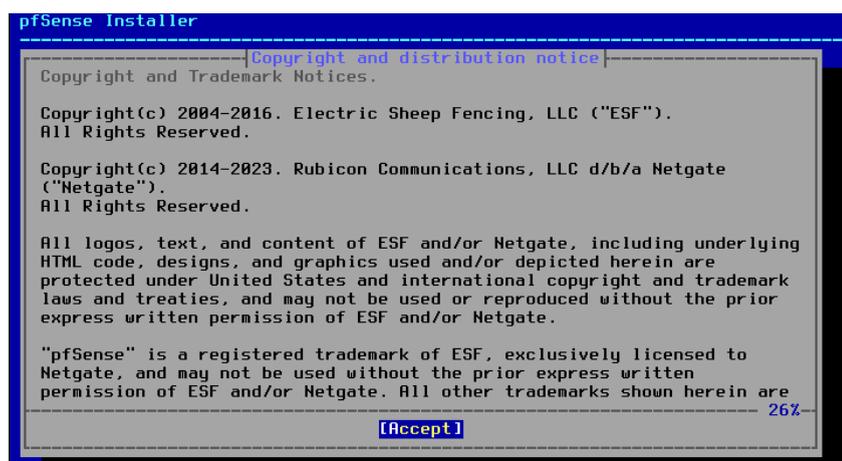


Figura 13: Derechos reservados de pfSense

Se muestran unas opciones y se selecciona la opción marcada por defecto "Install pfSense" y le damos Enter.



Figura 14: Opciones del instalador de pfSense.

Se muestra una pantalla donde se pregunta la partición de disco que se quiere utilizar, se selecciona la opción marcada por defecto “Auto (ZFS)” y damos Enter.



Figura 15: Opciones de partición de disco.

Se muestran las configuraciones de ZFS y se selecciona la opción marcada por defecto “Install” y damos Enter.

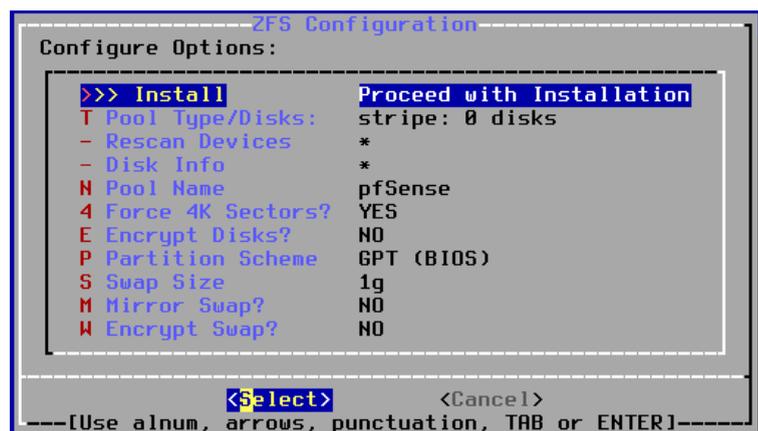


Figura 16: Opciones de configuración de ZFS.

Se muestra el tipo de dispositivo virtual que se desea utilizar de las configuraciones ZFS, se selecciona la opción marcada por defecto “stripe” y damos Enter.

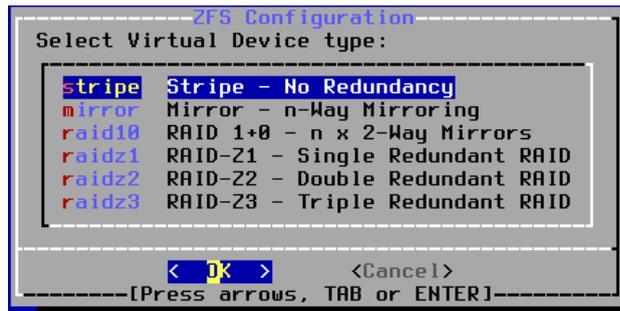


Figura 17: Opciones del tipo de dispositivo virtual.

Se muestra la lista con los dispositivos de almacenamiento de la máquina virtual, en éste caso como solo es uno aparece como "vtbd0" el cual se selecciona con la barra espaciadora y damos Enter.



Figura 18: Opciones de disco.

Se muestra una pantalla donde se pregunta si se quiere proceder con las configuraciones, se selecciona la opción por defecto "YES" y damos Enter y comenzará el proceso de instalación de pfSense. Una vez terminado nos pedirá reiniciar la VM para lo cual se selecciona la opción por defecto "Reboot" y damos Enter.

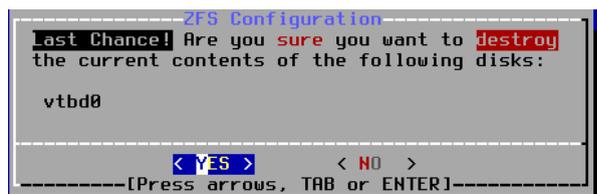


Figura 19: Pantalla de confirmación para la instalación de pfSense.

Una vez reiniciado pfSense nos pedirá si queremos configurar VLANS, procedimiento que se realizará después, por lo cual se selecciona "N" y damos Enter.

```
Do VLANs need to be set up first?
If VLANs will not be used, or only for optional interfaces, it is typical to
say no here and use the webConfigurator to configure VLANs later, if required.

Should VLANs be set up now [y|n]? 2023-10-29T23:18:09.889069+00:00 - php-fpm 377
- - /rc.linkup: Ignoring link event during boot sequence.
2023-10-29T23:18:09.889617+00:00 - php-fpm 377 - - /rc.linkup: Ignoring link eve
nt during boot sequence.
N
```

Figura 20: Pantalla de consola para la creación de VLANs.

Se muestran las opciones para configurar las interfaces de red (WAN y LAN) para la WAN se selecciona vtnet0 y para la LAN vtnet1 y se selecciona la opción “y” indicando que se desea proceder y damos Enter.

```
If the names of the interfaces are not known, auto-detection can
be used instead. To use auto-detection, please disconnect all
interfaces before pressing 'a' to begin the process.

Enter the WAN interface name or 'a' for auto-detection
(vtnet0 vtnet1 or a): vtnet0

Enter the LAN interface name or 'a' for auto-detection
NOTE: this enables full Firewalling/NAT mode.
(vtnet1 a or nothing if finished): vtnet1

The interfaces will be assigned as follows:

WAN -> vtnet0
LAN -> vtnet1

Do you want to proceed [y|n]? y
```

Figura 21: Pantalla de consola para la configuración de las interfaces.

Una vez completado el proceso de configuración de las interfaces se muestra la siguiente pantalla con las respectivas ip de cada interfaz.

```
Starting syslog...done.
Starting CRON... done.
pfSense 2.7.0-RELEASE amd64 Wed Jun 28 03:53:34 UTC 2023
Bootup complete

FreeBSD/amd64 (pfSense.home.arpa) (ttyv0)
QEMU Guest - Netgate Device ID: 007afb97a9d2ac0f86f0

*** Welcome to pfSense 2.7.0-RELEASE (amd64) on pfSense ***

WAN (wan)      -> vtnet0      -> v4/DHCP4: 192.168.1.20/24
LAN (lan)      -> vtnet1      -> v4: 192.168.1.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults   13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                 15) Restore recent configuration
7) Ping host                   16) Restart PHP-FPM
8) Shell

Enter an option: █
```

Figura 22: Pantalla principal de opciones de consola de pfSense

Se asigna una dirección ip estática a la interfaz LAN seleccionando la opción 2, nos pregunta la interfaz a configurarse, se desactiva el DHCP de dicha interfaz con la opción “n” y asignamos la ip estática 192.168.1.122 y posteriormente la máscara de subred “24”, la parte de IPv4 upstream gateway address se configura vacía presionando Enter, se desactiva el DHCP6 para la LAN y la ipv6 se deja vacía, por último se selecciona la opción “y” para revertir el webConfigurator protocol.

```
Available interfaces:
1 - WAN (vtnet0 - dhcp, dhcp6)
2 - LAN (vtnet1 - static)

Enter the number of the interface you wish to configure: 2
Configure IPv4 address LAN interface via DHCP? (y/n) n
Enter the new LAN IPv4 address. Press <ENTER> for none:
> 192.168.1.122

Subnet masks are entered as bit counts (as in CIDR notation) in pfSense.
e.g. 255.255.255.0 = 24
     255.255.0.0   = 16
     255.0.0.0    = 8

Enter the new LAN IPv4 subnet bit count (1 to 32):
> 24

For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
> █
```

Figura 23: Pantalla de configuración para interfaz LAN de manera estática.

Con la ip asignada se puede acceder al dashboard desde un navegador. Para iniciar sesión se ingresa el nombre de usuario y contraseña, por defecto el usuario es *admin* y la contraseña es *pfsense*.

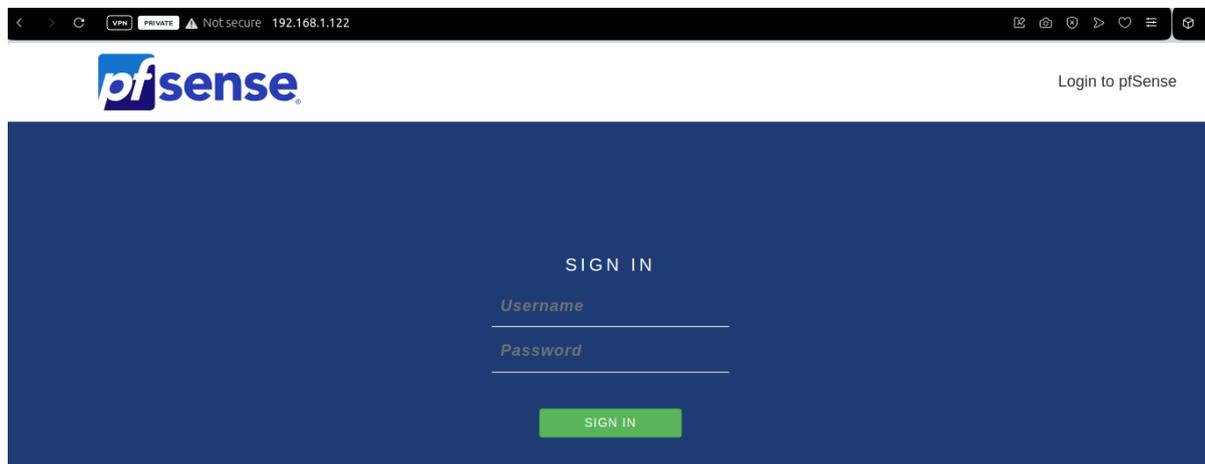


Figura 24: Dashboard de pfSense

Muestra una pantalla de bienvenida donde se procede a configurar pfSense, se le da siguiente hasta llegar al paso 2, en el cual se va a reescribir los servidores DNS primario (8.8.8.8) y secundario (8.8.4.4) o los que estén configurados de acuerdo a su ambiente.

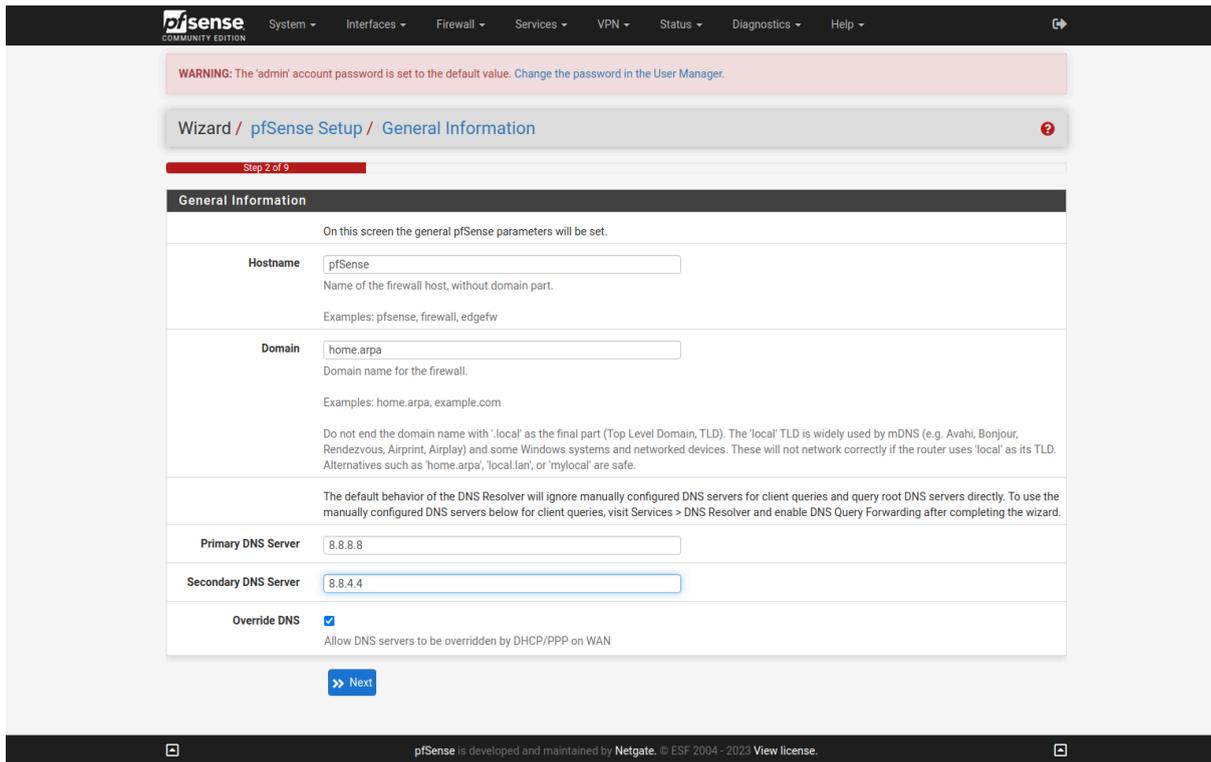


Figura 25: Pantalla de configuraciones generales del wizard.

En el paso 3 se selecciona la zona horaria deseada

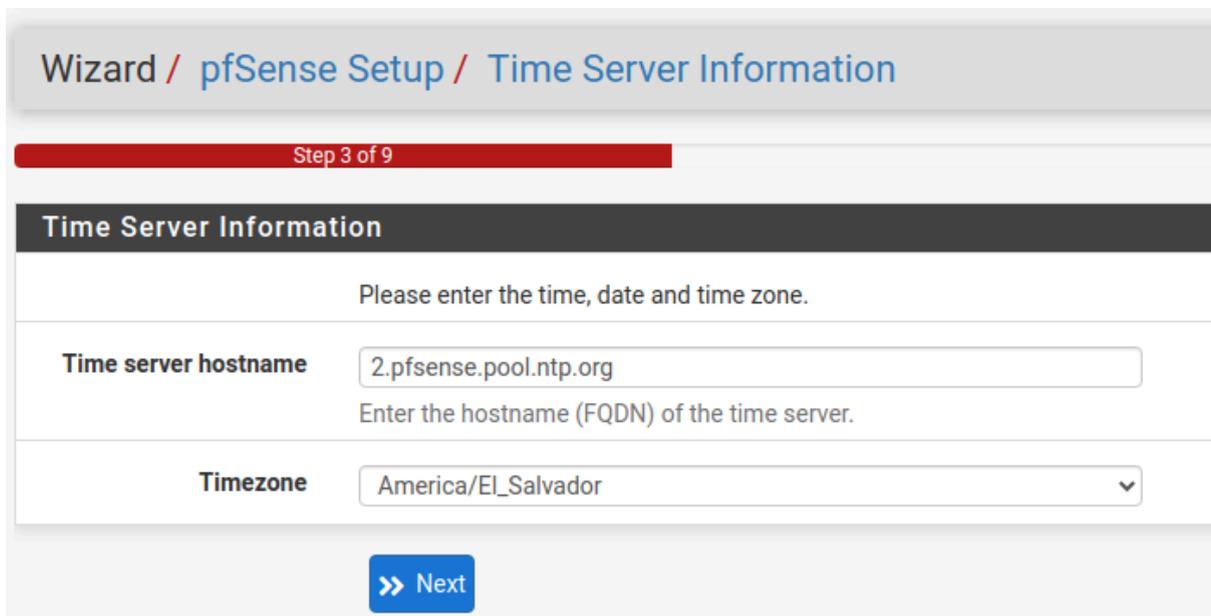


Figura 26: Pantalla de configuración del NTP.

El paso 4 sobre la configuración de la interfaz WAN se deja con los valores por defecto y se le da siguiente.

Wizard / pfSense Setup / Configure WAN Interface ?

Step 4 of 9

Configure WAN Interface

On this screen the Wide Area Network information will be configured.

SelectedType:

General configuration

MAC Address

This field can be used to modify ("spoof") the MAC address of the WAN interface (may be required with some cable connections). Enter a MAC address in the following format: xx:xx:xx:xx:xx:xx or leave blank.

MTU

Set the MTU of the WAN interface. If this field is left blank, an MTU of 1492 bytes for PPPoE and 1500 bytes for all other connection types will be assumed.

MSS

If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 (TCP/IP header size) will be in effect. If this field is left blank, an MSS of 1492 bytes for PPPoE and 1500 bytes for all other connection types will be assumed. This should match the above MTU value in most all cases.

Static IP Configuration

Figura 27: Pantalla de configuración de la interfaz WAN.

El paso 5 sobre la configuración de la interfaz LAN se deja con los valores por defecto, asumiendo que se configuró manualmente la ip.

Wizard / pfSense Setup / Configure LAN Interface

Step 5 of 9

Configure LAN Interface

On this screen the Local Area Network information will be configured.

LAN IP Address

Type dhcp if this interface uses DHCP to obtain its IP address.

Subnet Mask

[» Next](#)

Figura 28: Pantalla de configuración de la interfaz LAN.

El paso 6 muestra una pantalla para cambiar la contraseña por defecto del usuario admin.

Wizard / pfSense Setup / Set Admin WebGUI Password

Step 6 of 9

Set Admin WebGUI Password

On this screen the admin password will be set, which is used to access the WebGUI and also SSH services if enabled.

Admin Password

Admin Password AGAIN

Figura 29: Pantalla de cambio de contraseña del usuario admin.

Posteriormente se recarga el dashboard y podemos ver que el wizard ha sido completado con éxito.

3.1.1 Creando y configurando VLANs

En la barra de navegación se selecciona la opción “Interfaces” y seguido de la opción “Assignments” dentro de esa pantalla en la tab se selecciona la opción “VLANs” y se selecciona el botón “Add”.

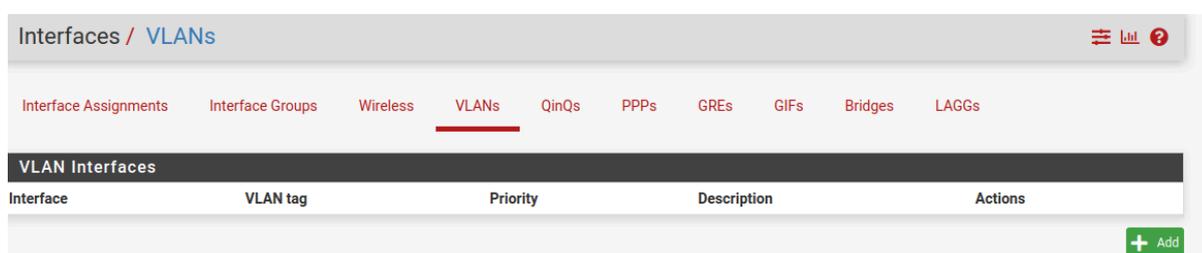


Figura 30: Pantalla para creación de VLANs.

Se selecciona la interfaz padre (parent interface) como la LAN, en el VLAN Tag se le asigna el TAG de acuerdo a la tabla de VLANs que se utilizarán, opcionalmente se agrega una descripción.

VLAN Configuration	
Parent Interface	<input type="text" value="vtnet1 (52:54:00:5e:e5:8a) - lan"/> <small>Only VLAN capable interfaces will be shown.</small>
VLAN Tag	<input type="text" value="10"/> <small>802.1Q VLAN tag (between 1 and 4094).</small>
VLAN Priority	<input type="text" value="0"/> <small>802.1Q VLAN Priority (between 0 and 7).</small>
Description	<input type="text" value="Management"/> <small>A group description may be entered here for administrative reference (not parsed).</small>

Figura 31: Pantalla de configuración de VLANs.

Se agregan las interfaces VLANs en la opción “Interface Assignments” seleccionando la interfaz deseada y haciendo click en el botón “Add”

Interface	Network port	
WAN	vtnet0 (52:54:00:c3:45:35)	
LAN	vtnet1 (52:54:00:5e:e5:8a)	Delete
OPT1	VLAN 10 on vtnet1 - lan (Management)	Delete
OPT2	VLAN 30 on vtnet1 - lan (Tunnel)	Delete
OPT3	VLAN 20 on vtnet1 - lan (Storage)	Delete
OPT4	VLAN 40 on vtnet1 - lan (VXLAN)	Delete

Save

Figura 32: Pantalla de VLANs asignadas.

Posteriormente se selecciona la interfaz en el nombre por defecto asignado, para este caso OPT1, etc. Se habilita la interfaz, se cambia el nombre de la interfaz, se asigna el tipo de configuración IPv4 a “static” y se asigna la ip estática de la respectiva VLAN”, guardamos y aplicamos los cambios. Este paso se debe de realizar para cada VLAN.

Interfaces / OPT1 (vtnet1.10)

General Configuration

Enable Enable interface

Description Management
Enter a description (name) for the interface here.

IPv4 Configuration Type Static IPv4

IPv6 Configuration Type None

MAC Address xxxxxxxxxxxxxx
The MAC address of a VLAN interface must be set on its parent interface

MTU
If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.

MSS
If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 for IPv4 (TCP/IPv4 header minus 60 for IPv6 (TCP/IPv6 header size) will be in effect.

Speed and Duplex Default (no preference, typically autoselect)
Explicitly set speed and duplex mode for this interface.
WARNING: MUST be set to autoselect (automatically negotiate speed) unless the port this interface connects to has its speed and duplex mode set.

Static IPv4 Configuration

IPv4 Address 172.236.29.1 / 22

Figura 33: Pantalla para la configuración general de VLANs.

3.1.2. Reglas Firewall para VLANs

En la sección “Firewall” se selecciona la opción “Rules”, ahí se selecciona las interfaces VLAN creadas anteriormente y se selecciona el botón “Add” para agregar una nueva regla.

3.1.2.1. Reglas para Management

- Para acceso a internet.

The screenshot shows the 'Edit Firewall Rule' configuration page. The breadcrumb navigation is 'Firewall / Rules / Edit'. The main title is 'Edit Firewall Rule'. The configuration fields are as follows:

- Action:** Pass. Below it, a note states: 'Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.'
- Disabled:** Disable this rule. Below it, a note states: 'Set this option to disable this rule without removing it from the list.'
- Interface:** MANAGEMENT. Below it, a note states: 'Choose the interface from which packets must come to match this rule.'
- Address Family:** IPv4. Below it, a note states: 'Select the Internet Protocol version this rule applies to.'
- Protocol:** Any. Below it, a note states: 'Choose which IP protocol this rule should match.'

Figura 34: Pantalla para regla con acceso a internet.

- Para ICMP

The screenshot shows the 'Edit Firewall Rule' configuration page for ICMP. The breadcrumb navigation is 'Edit Firewall Rule'. The configuration fields are as follows:

- Action:** Pass. Below it, a note states: 'Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.'
- Disabled:** Disable this rule. Below it, a note states: 'Set this option to disable this rule without removing it from the list.'
- Interface:** MANAGEMENT. Below it, a note states: 'Choose the interface from which packets must come to match this rule.'
- Address Family:** IPv4. Below it, a note states: 'Select the Internet Protocol version this rule applies to.'
- Protocol:** ICMP. Below it, a note states: 'Choose which IP protocol this rule should match.'
- ICMP Subtypes:** any. Below it, a list of subtypes is shown: Alternate Host, Datagram conversion error, Echo reply. Below the list, a note states: 'For ICMP rules on IPv4, one or more of these ICMP subtypes may be specified.'
- Source:** Invert match. Source Address: any.
- Destination:** Invert match. Destination Address: any.

Figura 35: Pantalla para regla ICMP.

3.1.2.2. Reglas para las otras VLANs

Las VLANs restantes tendrán únicamente una regla, la cual es ICMP, la cual utiliza la misma configuración de la VLAN Management como se describe en la figura 34 , solamente se debe de cambiar la interfaz “Management” por la interfaz deseada.

3.1.3. Configuración de Tailscale

Para poder usar la VPN es necesario instalar el paquete, por lo cual en la barra de navegación se debe seleccionar “System” y seleccionar la opción “Package Manager” y seleccionar el tab “Available Packages”. Para iniciar su búsqueda simplemente se escribe *tailscale* y se presiona enter y para instalarlo damos click en el botón “install”.

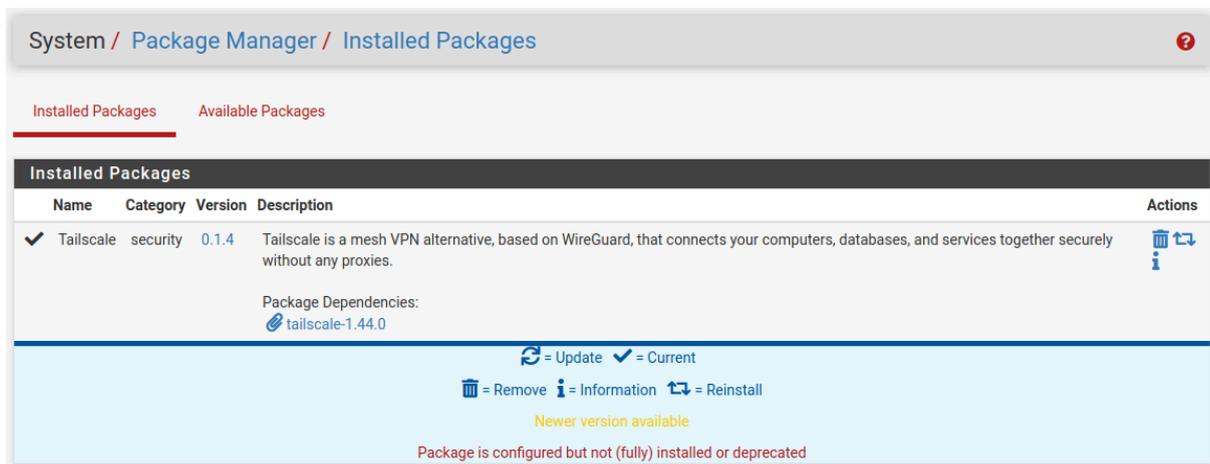


Figura 36: Pantalla de paquetes instalados.

Una vez instalado el paquete en la barra de navegación se selecciona la opción “Firewall” y seguido de la opción “NAT” dentro de esa pantalla en la tab se selecciona la opción “Outbound”, ahí se selecciona el modo “Hybrid Outbound NAT rule generation”, y en la parte de Mappings se agrega lo siguiente:

Firewall / NAT / Outbound / Edit

Edit Advanced Outbound NAT Entry

Disabled Disable this rule

Do not NAT Enabling this option will disable NAT for traffic matching this rule and stop processing Outbound NAT rules
In most cases this option is not required.

Interface
The interface on which traffic is matched as it exits the firewall. In most cases this is "WAN" or another externally-connected interface.

Address Family
Select the Internet Protocol version this rule applies to.

Protocol
Choose which protocol this rule should match. In most cases "any" is specified.

Source /
Type Source network for the outbound NAT mapping. Port or Range

Destination /
Type Destination network for the outbound NAT mapping. Port or Range

Not
Invert the sense of the destination match.

Figura 37: Pantalla de configuración de Mappings.

Una vez guardado el mapping debería salir lo siguiente:

Firewall / NAT / Outbound

Port Forward 1:1 **Outbound** NPt

Outbound NAT Mode

Mode

- Automatic outbound NAT rule generation. (IPsec passthrough included)
- Hybrid Outbound NAT rule generation. (Automatic Outbound NAT + rules below)
- Manual Outbound NAT rule generation. (AON - Advanced Outbound NAT)
- Disable Outbound NAT rule generation. (No Outbound NAT rules)

Mappings

<input type="checkbox"/>	Interface	Source	Source Port	Destination	Destination Port	NAT Address	NAT Port	Static Port	Description	Actions
<input checked="" type="checkbox"/>	Tailscale	any	*	*	*	Tailscale address	*	↔		<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>

Figura 38: Pantalla general de Outbound.

Una vez configurado el Mapping en la barra de navegación se selecciona la opción "VPN" seguido de la opción "Tailscale" dentro de esa pantalla en la tab se selecciona la opción "Settings" y se pone la siguiente configuración:

Settings													
Enable	<input checked="" type="checkbox"/> Enable Tailscale												
Listen Port	<input type="text" value="41641"/> UDP port to listen on for WireGuard and peer-to-peer traffic.												
State Directory	<input type="text" value="/usr/local/pkg/tailscale/state"/> Path to directory for storage of config state, certificates, and incoming files. WARNING: Changing this value will not move an existing configuration and will require reauthentication with the control server.												
Keep Configuration	<input checked="" type="checkbox"/> Enable With 'Keep Configuration' enabled (default), all package settings and the local Tailscale state cache will persist on install/de-install.												
DNS													
Accept DNS	<input checked="" type="checkbox"/> Accept DNS configuration from the control server.												
Routing													
Advertise Exit Node	<input type="checkbox"/> Offer to be an exit node for outbound internet traffic from the Tailscale network.												
Accept Subnet Routes	<input checked="" type="checkbox"/> Accept subnet routes that other nodes advertise.												
Notice	Routes will be transformed into proper subnet start boundaries prior to validating and saving.												
Advertised Routes	<table border="1"> <tr> <td><input type="text" value="172.29.236.0/22"/></td> <td><input type="text"/></td> <td><input type="button" value="Delete"/></td> </tr> <tr> <td><input type="text" value="172.29.240.0/22"/></td> <td><input type="text"/></td> <td><input type="button" value="Delete"/></td> </tr> <tr> <td><input type="text" value="172.29.244.0/22"/></td> <td><input type="text"/></td> <td><input type="button" value="Delete"/></td> </tr> <tr> <td><input type="text" value="172.29.248.0/22"/></td> <td><input type="text"/></td> <td><input type="button" value="Delete"/></td> </tr> </table>	<input type="text" value="172.29.236.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>	<input type="text" value="172.29.240.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>	<input type="text" value="172.29.244.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>	<input type="text" value="172.29.248.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>
<input type="text" value="172.29.236.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>											
<input type="text" value="172.29.240.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>											
<input type="text" value="172.29.244.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>											
<input type="text" value="172.29.248.0/22"/>	<input type="text"/>	<input type="button" value="Delete"/>											
	Subnet expressed using CIDR notation Administrative description (not parsed)												
Add	<input type="button" value="+ Add"/>												
Logging													
Syslog Logging	<input checked="" type="checkbox"/> Enable syslog output												
Syslog Settings	<input type="text" value="NOTICE"/> <input type="text" value="LOG_DAEMON"/> <small>Set the syslog logging priority. Set the syslog logging facility.</small>												

Figura 39: Pantalla de configuración de Tailscale.

Se selecciona el tab de Authentication y en el "Pre-authentication Key" se pone el key generado en Tailscale, se guardan los cambios y con eso pfSense ya está configurado con la VPN.

Authentication	
Login Server	<input type="text" value="https://controlplane.tailscale.com"/> Base URL of login (control) server.
Pre-authentication Key	<input type="text" value="tskey-auth-kQhpCq3CNTRL-w1UWTxg48yR3pAijgmKh1yRQRvgDjXKGHr"/> Set the machine authorization key.
Recommendation	Use a non-reusable auth key and disable key expiry for trusted machines via the provider admin console.
Logout and Clean	
	<input type="button" value="Logout and Clean"/> Disconnect local machine from login server (if connected), expire the current log in, and flush the local state cache.
	<input type="button" value="Save"/>

Figura 40: Pantalla de autenticación de Tailscale.

3.2. Máquinas virtuales con Ubuntu Server

Es importante aclarar que las redes pueden ser diferentes dependiendo del proveedor de internet. Para iniciar la creación de una nueva máquina virtual con Ubuntu

Server en KVM, se deben realizar los mismos pasos utilizados para la creación de pfSense, solo que esta vez se debe de seleccionar la opción “Detectar automáticamente el medio de instalación” y se debe agregar la cantidad de memoria RAM, CPU, el almacenamiento necesario según lo que se requiera y de igual manera las interfaces de red.

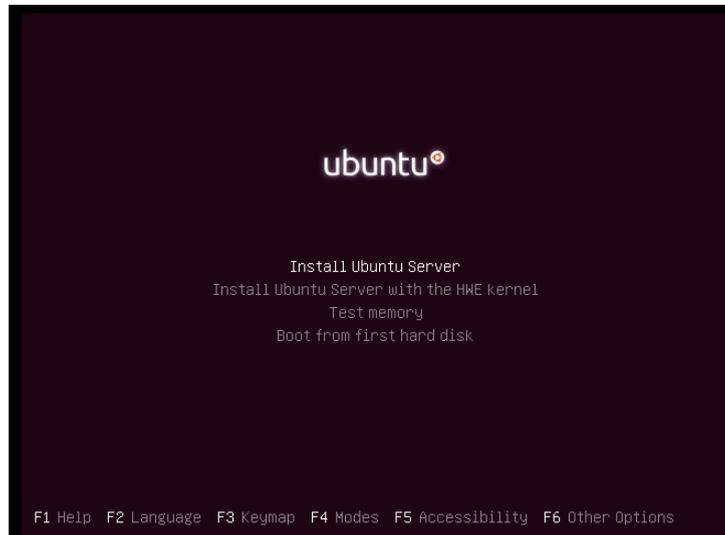


Figura 41: Pantalla principal con las opciones de instalación.

En el instalador busca el idioma deseado y se selecciona “siguiente”.

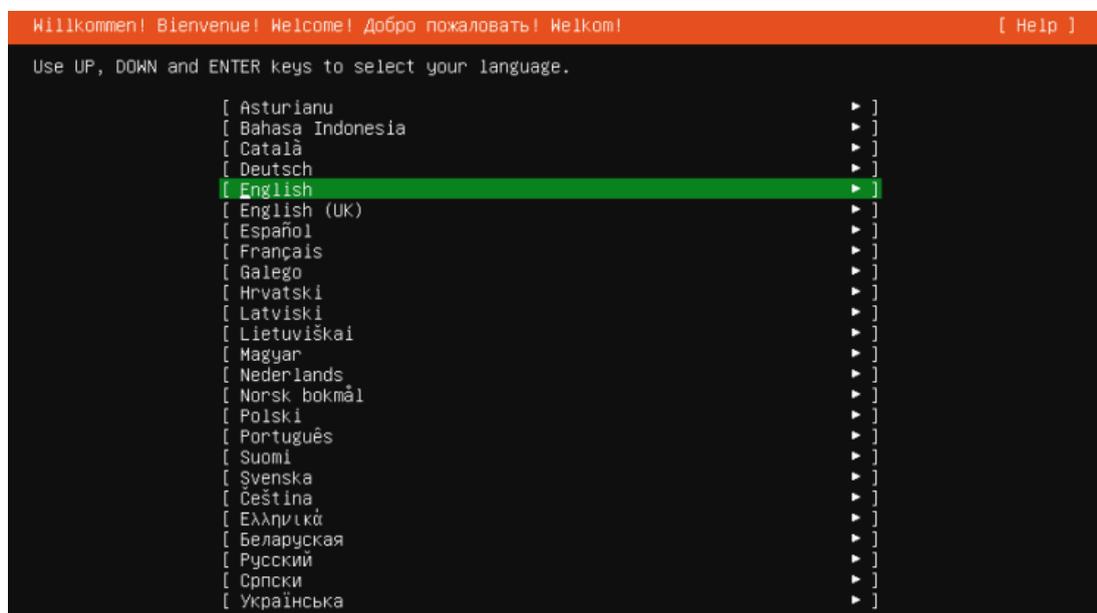


Figura 42: Pantalla para seleccionar idioma del instalador.

Si aparece una opción que indica que se debe actualizar, se debe seleccionar “Continuar sin actualizar”.

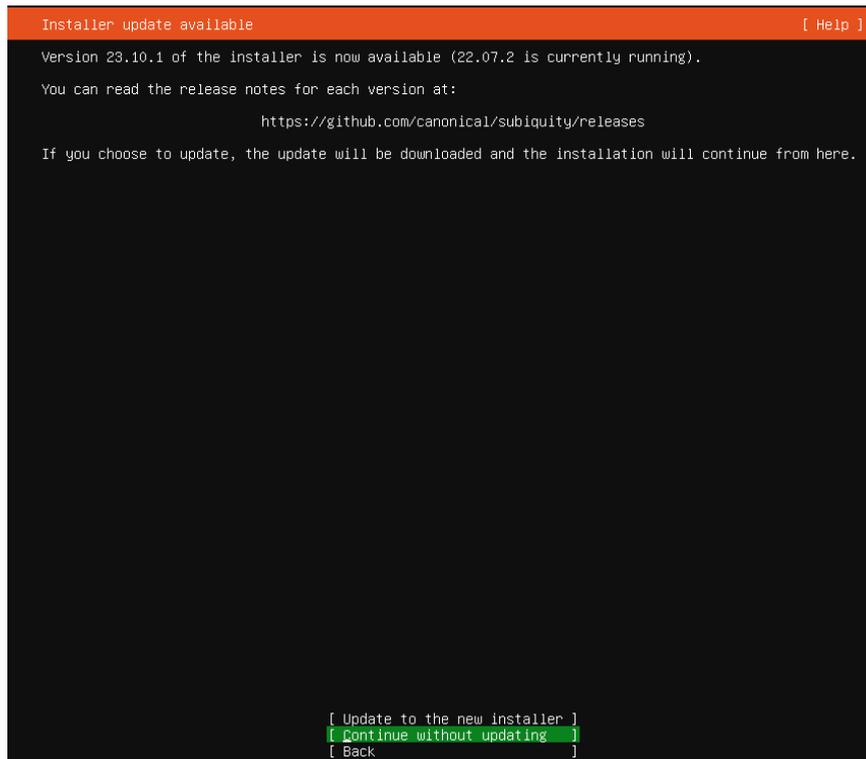


Figura 43: Pantalla para actualizar instalador.

Se configura la distribución del teclado deseada y se selecciona “siguiente”.

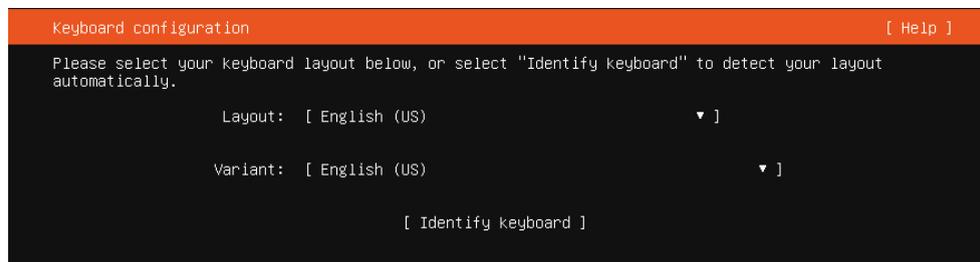


Figura 44: Pantalla para seleccionar la configuración del teclado.

Se configuran las redes deseadas y dar en siguiente. Se puede dejar la configuración por defecto.

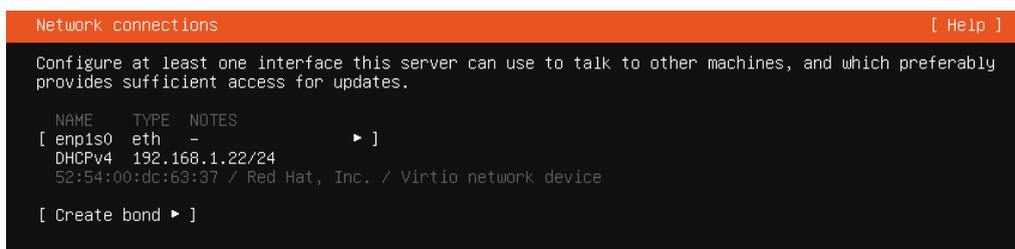


Figura 45: Pantalla para configurar las redes.

La configuración del proxy y del mirror se deben dejar con las configuraciones por defecto y dar en siguiente.

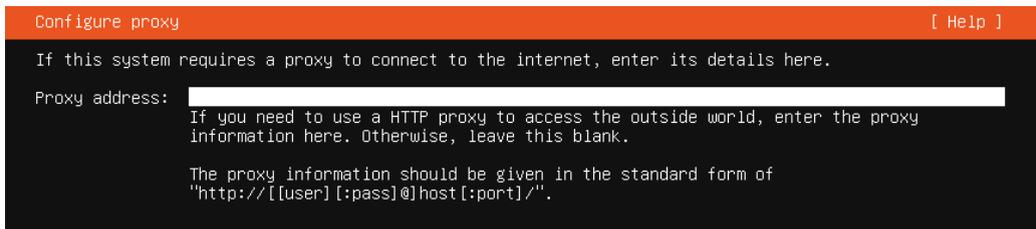


Figura 46: Pantalla para configuración del proxy.

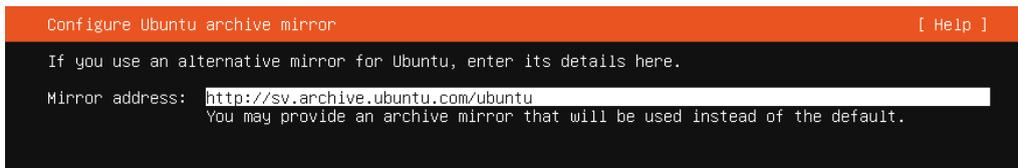


Figura 47: Pantalla para configuración de mirror.

Se muestran las opciones de configuración de almacenamiento, se dejan las opciones que aparecen por defecto y se selecciona “siguiente”.

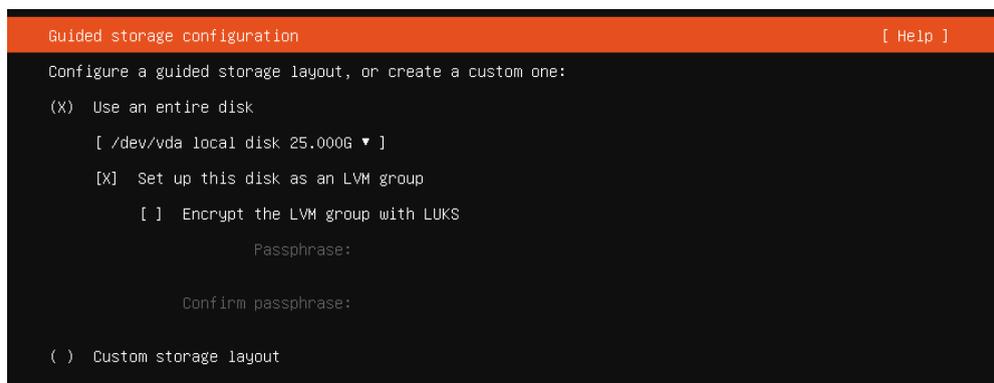


Figura 48: Pantalla para configuración de almacenamiento.

Se muestra una configuración más detallada del almacenamiento, se dejan las opciones por defecto y se selecciona “siguiente”.

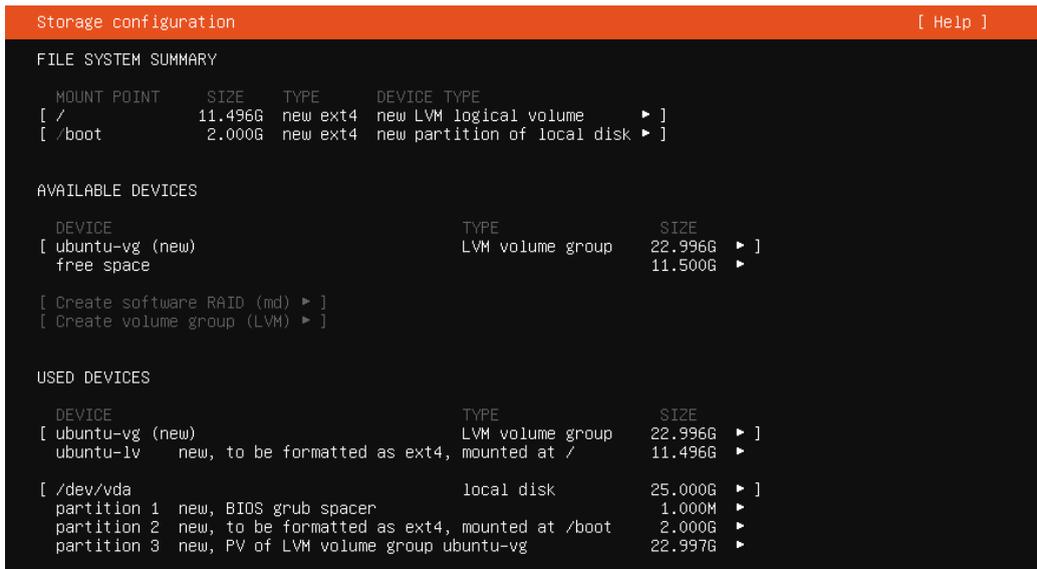


Figura 49: Pantalla para configuración de almacenamiento avanzado.

Se muestra una pantalla donde se pregunta si se quiere proceder con la configuración se selecciona “siguiente”.

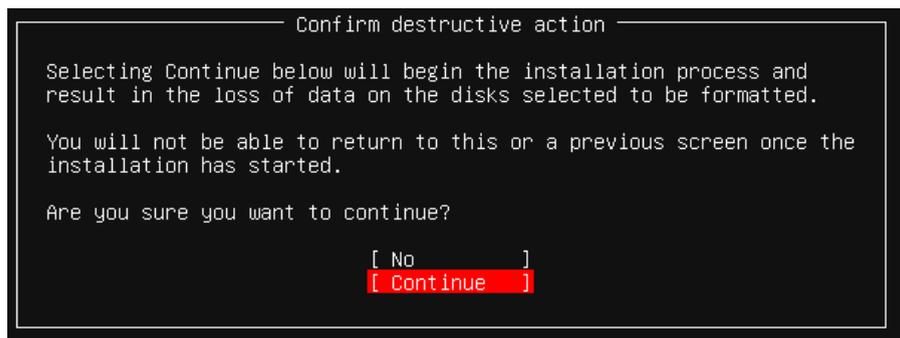


Figura 50: Pantalla de confirmación de opciones de almacenamiento.

Se introduce el nombre de usuario y contraseña deseados y se selecciona “siguiente”.

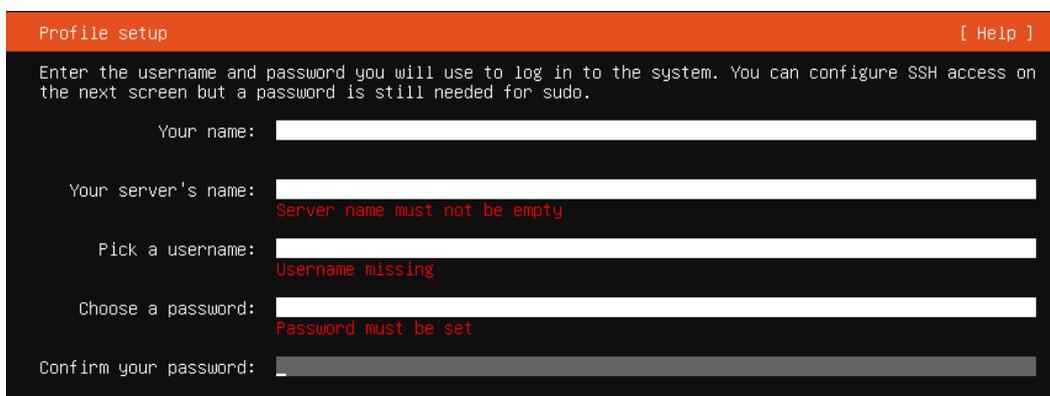


Figura 51: Pantalla para configurar usuario y contraseña.

En la configuración del SSH se selecciona que instale el servidor OpenSSH y se selecciona “siguiente”.

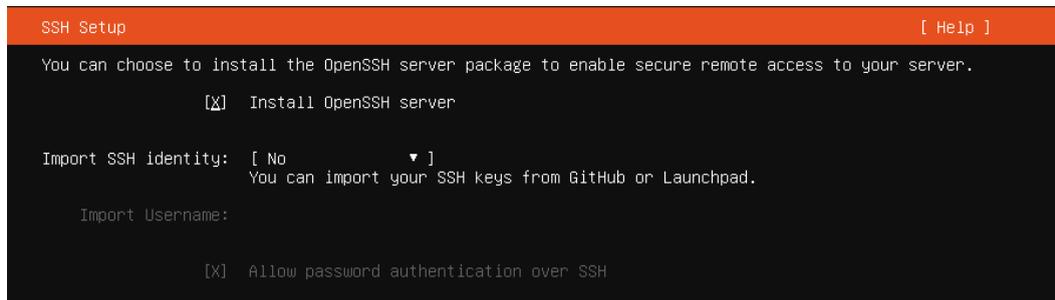


Figura 52: Pantalla para configurar OpenSSH.

En las opciones extra que se pueden instalar no se selecciona ninguna y se selecciona “siguiente”.

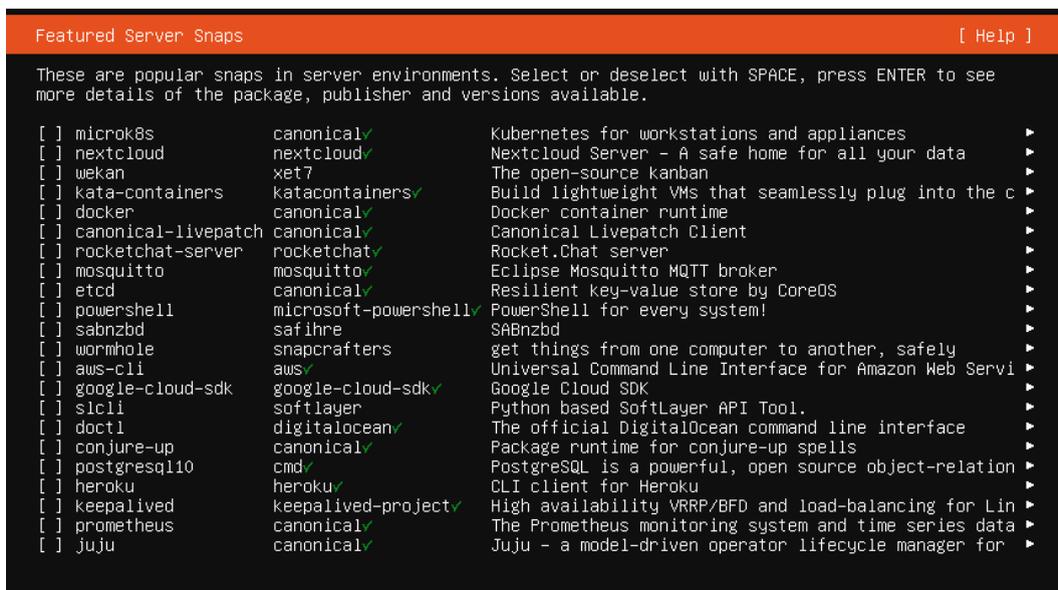


Figura 53: Pantalla para seleccionar paquetes extra.

Se muestra la instalación de Ubuntu Server, la cual una vez finalizada se selecciona “Reiniciar ahora”.

```
Install complete! [ Help ]

configuring format: format-1
configuring mount: mount-1
configuring mount: mount-0
writing install sources to disk
running 'curtin extract'
curtin command extract
  acquiring and extracting image from cp:///tmp/tmp6vcjxiz/mount
configuring installed system
running 'mount --bind /cdrom /target/cdrom'
running 'curtin curthooks'
curtin command curthooks
  configuring apt configuring apt
  installing missing packages
  configuring iscsi service
  configuring raid (mdadm) service
  installing kernel
  setting up swap
  apply networking config
  writing etc/fstab
  configuring multipath
  updating packages on target system
  configuring pollinate user-agent on target
  updating initramfs configuration
  configuring target system bootloader
  installing grub to target devices
finalizing installation
running 'curtin hook'
  curtin command hook
  executing late commands
final system configuration
  configuring cloud-init
  calculating extra packages to install
  installing openssh-server
  curtin command system-install
  downloading and installing security updates
  curtin command in-target
  restoring apt configuration
  curtin command in-target
subiquity/Late/run

[ View full log ]
[ Reboot Now ]
```

Figura 54: Pantalla de instalación de Ubuntu Server.

4. Instalación y Configuración del Clúster de Ceph

4.1. Requerimientos

- Ubuntu Server 20.04 LTS
- Tres nodos ceph1, ceph2, ceph3 (mon,mgr,osd)

Host name	NIC1 (NAT)	NIC2 (Bridge)		Disco (GB)	RAM (MB)	NIC
		Management	Storage			
ceph1	192.168.5.18	172.29.236.18	172.29.244.18	1 x 50 1 x 100	1024	2
ceph2	192.168.5.19	172.29.236.19	172.29.244.19	1 x 50 1 x 100	1024	2
ceph3	192.168.5.20	172.29.236.20	172.29.244.20	1 x 50 1 x 100	1024	2

Nota: Los tres nodos de ceph actuarán como monitores y como osd

4.2. Preparando los nodos

Se debe cambiar el hostname en los tres nodos con el siguiente comando:

```
hostnamectl set-hostname ceph1  
exec bash
```

Nota: se debe cambiar ceph1 por ceph2 y ceph3 en sus respectivos nodos.

Una vez cambiado el hostname se debe modificar el archivo /etc/hosts en cada uno de los nodos.

```
172.29.236.18 ceph1  
172.29.236.19 ceph2  
172.29.236.20 ceph3
```

Para poder usar Vlans se debe de instalar un paquete llamado vlan y activarlo (se debe realizar en cada Nodo).

```
sudo apt-get install vlan  
sudo modprobe 8021q  
lsmod | grep 8021q
```

Después de instalar el paquete de vlan y activarlo se configura el netplan para el nodo ceph1 con lo siguiente:

```

# This is the network config written by 'subiquity'
network:
  ethernets:
    enp1s0:
      dhcp4: false
      addresses: [192.168.5.18/24]
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
      routes:
        - to: default
          via: 192.168.5.1
          metric: 1000
    enp2s0:
      dhcp4: false
      dhcp6: false
  vlans:
    enp2s0.10:
      id: 10
      link: enp2s0
    enp2s0.20:
      id: 20
      link: enp2s0
  bridges:
    br-mgmt:
      addresses: [172.29.236.18/22]
      interfaces: [enp2s0.10]
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
      routes:
        - to: default
          via: 172.29.236.1
          metric: 100
    br-storage:
      interfaces: [enp2s0.20]
      addresses: [172.29.244.18/22]
  version: 2

```

Nota: Realizar lo mismo para los otros nodos con sus respectivas ip.

Para instalar Docker se crea un script el cual agrega e instala Docker, esto se debe de realizar en cada uno de los nodos. Para lo cual se ejecuta lo siguiente **nano** **dockerInstall.sh** y se pone el siguiente contenido:

```

#!/bin/sh

# Add Docker's official GPG key:
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/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg

```

```
# Add the repository to Apt sources:
echo \
  "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/ubuntu \
  "$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update

# Install the latest version
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin
```

Se cambian los permisos y se ejecuta el script como usuario root

```
chmod u+x dockerInstall.sh
./dockerInstall.sh
```

4.3. Instalando cephadm

Para instalar cephadm se crea un script el cual instalará cephadm, éste proceso sólo se realiza en el nodo de ceph1. Se ejecuta **nano installCeph.sh** y se pone el siguiente contenido:

```
#!/bin/bash

# Install cephadm
curl --silent --remote-name --location
https://github.com/ceph/ceph/raw/pacific/src/cephadm/cephadm
ls
chmod +x cephadm

# Add repo
./cephadm add-repo --release pacific
./cephadm install
which cephadm
```

Se cambian los permisos y se ejecuta el script como usuario root

```
chmod u+x installCeph.sh
./installCeph.sh
```

4.4. Iniciar un nuevo cluster

Para iniciar el clúster se debe ejecutar el comando **cephadm bootstrap** en el primer host del clúster Ceph.

```
sudo cephadm bootstrap --mon-ip 172.29.244.18 --allow-fqdn-hostname
```

Nota: Una vez finalizado el comando anterior, aparecerá el nombre de usuario/contraseña del dashboard.

4.5. Habilitando ceph CLI

Habilitando ceph CLI, esto se debe de realizar solo en ceph1

```
sudo cephadm install ceph-common
```

4.6. Agregando hosts adicionales al cluster

Para agregar hosts adicionales al cluster se debe poner el monitor como unmanaged, la opción "--unmanaged" se utiliza para indicar que el monitor se configurará como "no administrado". Esto significa que no será gestionado automáticamente por la herramienta de administración de Ceph, llamada Cephadm. Al establecer un monitor como "no administrado," se evita que Cephadm realice despliegues automáticos de monitores en ese nodo. De lo contrario, Cephadm desplegará automáticamente un monitor en ceph2.

En el nodo ceph1 se debe ejecutar lo siguiente:

```
sudo ceph orch apply mon --unmanaged
```

Para agregar el nuevo host al cluster se debe pasar el SSH Key al nuevo host (ceph2).

Ejecutar en el nodo ceph1

```
sudo ssh-copy-id -f -i /etc/ceph/ceph.pub root@172.29.236.19  
sudo ssh-copy-id -f -i /etc/ceph/ceph.pub root@172.29.236.20
```

Se le dice a ceph que el nuevo nodo es parte del cluster y se agregan etiquetas a cada uno de los nodos.

```
sudo ceph orch host add ceph2 172.29.236.19 --labels _admin  
sudo ceph orch host add ceph3 172.29.236.20 --labels _admin
```

Una vez ejecutado lo anterior, en los nuevos nodos se iniciará la descarga y ejecución de los contenedores en Docker, éste proceso puede tomar entre 5 a 15 minutos, por eso al ejecutar *docker ps* la primera vez en los otros nodos no muestra ninguna salida.

4.7. Agregando los OSDs

4.7.1. Inflando el cluster

Para agregar los OSDs se crean discos LVM en los nodos ceph1, ceph2 y ceph3.

Listando los discos disponibles en el host

```
sudo fdisk -l
```

De los discos disponibles se asume que la dirección */dev/vdb* será el disco de 100GB extra de cada nodo OSD. Creando el volumen LVM *vg01*

```
vgcreate vg01 /dev/vdb
```

Creando el volumen de disco *lv01* en el grupo *lvm vg01*

```
lvcreate -L 99G -n lv01 vg01
```

Agregando los OSDs (ejecutar en ceph1)

```
sudo ceph orch daemon add osd ceph1:vg01/lv01
sudo ceph orch daemon add osd ceph2:vg01/lv01
sudo ceph orch daemon add osd ceph3:vg01/lv01
```

4.7.2. Agregando mon daemons

Los nodos ceph2 y ceph3 también actuarán como monitores por lo tanto se ejecuta lo siguiente solamente en el ceph1

```
sudo ceph orch daemon add mon ceph2:172.29.236.19
sudo ceph orch daemon add mon ceph3:172.29.236.20
```

Ejecutar en el nodo ceph1

```
sudo ceph orch apply mon --placement="ceph1,ceph2,ceph3"
```

Una vez completado estos pasos tendremos listo podremos usar el cluster de ceph donde se requiera.

- Dashboard

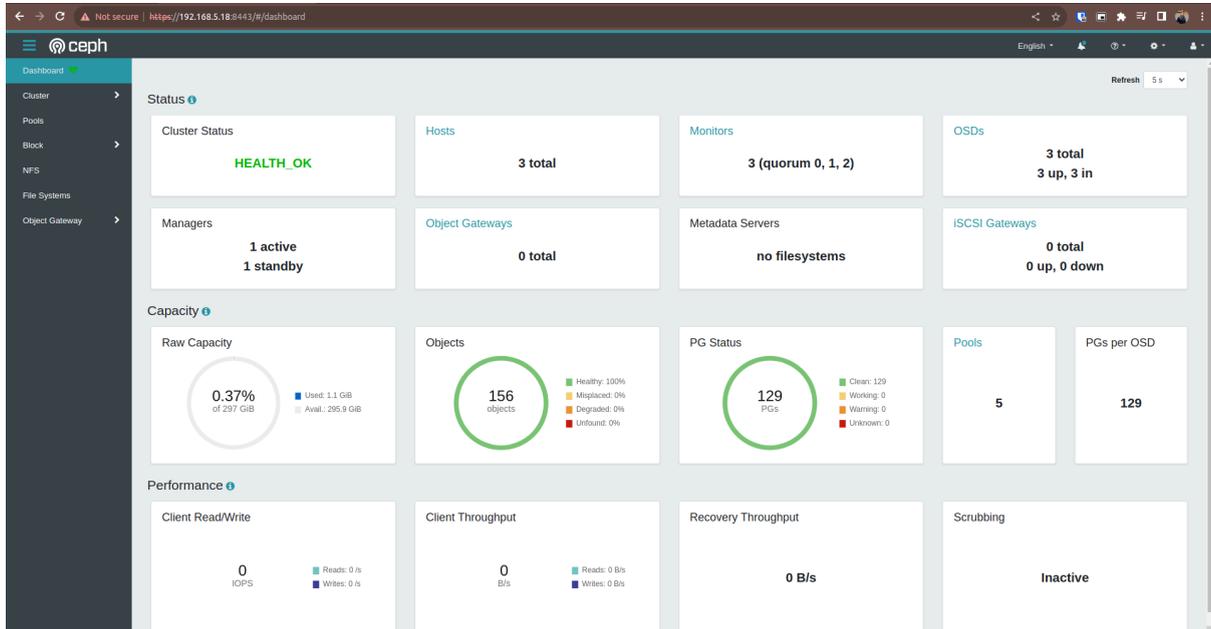


Figura 55: Dashboard principal en Ceph

4.8. Configuración de Pools y Usuarios en Ceph

4.8.1. Creación de pools

```
ceph osd pool create volumes
ceph osd pool create images
ceph osd pool create backups
ceph osd pool create vms
```

4.8.2. Inicialización de los pools

```
rbd pool init volumes
rbd pool init images
rbd pool init backups
rbd pool init vms
```

4.8.3. Generando los usuarios y keyrings

```
ceph auth get-or-create client.glance mon 'profile rbd' osd 'profile rbd pool=images' mgr 'profile rbd pool=images' -o /etc/ceph/glance.keyring
```

```
ceph auth get-or-create client.cinder mon 'profile rbd' osd 'profile rbd pool=volumes, profile rbd pool=vms, profile rbd-read-only pool=images' mgr 'profile rbd pool=volumes, profile rbd pool=vms' -o /etc/ceph/cinder.keyring
```

```
ceph auth get-or-create client.cinder-backup mon 'profile rbd' osd 'profile rbd  
pool=backups' mgr 'profile rbd pool=backups' -o /etc/ceph/cinder-backup.keyring
```

5. Instalación y Configuración de la región de OpenStack

5.1. Requerimientos

Nodo	CPU	RAM (MB)	Disco (GB)	NIC	OS
Pfsense	1	700	1 x 20	2	FreeBSD 13.1
Deployment	4	2048	1 x 25	1	Ubuntu Server 20.04
Control Plane	12	16896	1 x 75	2	Ubuntu Server 20.04
Compute	8	5120	1 x 75	2	Ubuntu Server 20.04
Ceph	1	700	1 x 50 1 x 100	2	Ubuntu Server 20.04

5.2. Configuración de redes empleadas

Nombre de la red	Tipo	Red		ID
		Región Uno	Región Dos	
Management	VLAN	172.29.236.0/22	172.30.236.0/22	10
Tunnel	VLAN	172.29.240.0/22	172.30.240.0/22	30
Storage	VLAN	172.29.244.0/22	172.30.244.0/22	20
VLAN (provider)	VLAN	172.29.248.0/22	172.30.248.0/22	40
NAT (provider)	FLAT	192.168.123.0/22	192.168.122.0/22	-
LAN	FLAT	192.168.1.0/22	192.168.100.0/22	-

5.3. Asinacion de Ip

Nodo	Management	Tunnel	Storage	VLAN
Pfsense	172.29.236.1	172.29.240.1	172.29.244.1	172.29.248.1
Deployment	172.29.236.2	-	-	-
Control Plane	172.29.236.11	172.29.240.11	172.29.244.11	172.29.248.11
Compute1	172.29.236.21	172.29.240.21	172.29.244.21	172.29.248.21
Compute2	172.29.236.22	172.29.240.22	172.29.244.22	172.29.248.22

5.4. Configuración de netplan

5.4.1. Nodo deployment

```
network:
ethernets:
enp1s0:
addresses: [192.168.16.175/24]
nameservers:
addresses: [8.8.8.8]
routes:
- to: default
via: 192.168.16.1
metric: 1000
vlans:
enp1s0.10:
id: 10
link: enp1s0
macaddress: 00:11:00:00:bb:aa
bridges:
br-mgmt:
addresses: [172.29.236.2/22]
interfaces: [enp1s0.10]
macaddress: 60:f4:12:dd:70:cc
nameservers:
addresses: [8.8.8.8]
routes:
- to: default
via: 172.29.236.1
metric: 100
version: 2
```

5.4.2. Nodo control-plane y nodos compute

```
network:
ethernets:
enp10s0:
dhcp4: false
dhcp6: false
enp1s0:
addresses: [192.168.16.145/24]
nameservers:
addresses: [8.8.8.8]
routes:
- to: default
via: 192.168.16.1
metric: 1000
enp9s0:
dhcp4: false
dhcp6: false
vlans:
enp10s0.10:
id: 10
```

```
link: enp10s0macaddress: 00:01:00:00:00:10
enp10s0.20:
id: 20
link: enp10s0
macaddress: 00:02:00:00:00:20
enp10s0.30:
id: 30
link: enp10s0
macaddress: 00:01:00:00:00:30
bridges:
br-mgmt:
addresses: [172.29.236.11/22]
interfaces: [enp10s0.10]
macaddress: 60:f4:12:ac:32:aa
nameservers:
addresses: [8.8.8.8]
routes:
- to: default
via: 172.29.236.1
metric: 100
br-storage:
interfaces: [enp10s0.20]
addresses: [172.29.244.11/22]
macaddress: 60:f4:12:bb:ff:55
br-vxlan:
interfaces: [enp10s0.30]
addresses: [172.29.240.11/22]
macaddress: 60:f4:12:ac:34:89
br-vlan:
interfaces: [enp10s0]
macaddress: 60:f4:12:ac:34:52
version: 2
```

Nota: Para los nodos compute realizar el mismo procedimiento con sus respectivas direcciones ip.

5.5. Preparando nodos

5.5.1. Configurando llaves SSH

Será necesario transferir la llave a los target host para tener privilegios de superusuario durante el despliegue.

```
ssh-copy-id root@172.29.236.11
ssh-copy-id root@172.29.236.21
ssh-copy-id root@172.29.236.31
ssh-copy-id root@172.29.236.18
```

5.5.2. Instalando dependencias

Se debe crear script.sh en todos los nodos, su finalidad es agilizar el proceso, pero se debe ejecutar antes de la ejecución de los playbooks más adelante.

5.5.2.1. Nodo deployment

```
#!/bin/bash
apt update
apt dist-upgrade
apt install build-essential git chrony openssh-server python3-dev sudo
git clone -b 23.4.3 https://opendev.org/openstack/openstack-ansible
/opt/openstack-ansible
cd /opt/openstack-ansible
scripts/bootstrap-ansible.sh
```

5.5.2.2. Nodos control plane y compute

```
#!/bin/bash
apt update
apt dist-upgrade
apt install bridge-utils debootstrap openssh-server tcpdump vlan python3
apt install linux-modules-extra-$(uname -r)
```

Para ambos archivos ejecutar el comando para brindarles permisos de ejecución, pero aún no ejecutar.

```
chmod u+x script.sh
```

5.5.2.3. Ejecutando script

Procederemos a ejecutar los script que creamos anteriormente en el nodo deployment primeramente y posteriormente en los demás nodos, dicha ejecución será con el siguiente comando.

```
./script.sh
```

5.6. Configurando despliegue de OpenStack Ansible

En el nodo deployment procederemos a crear en la ruta /etc/ el openstack_deploy copiándolo de openstack-ansible, además dentro crearemos el archivo openstack_user_config.yml donde configuraremos nuestro despliegue.

```
cp -r /opt/openstack-ansible/etc/openstack_deploy /etc/  
cd /etc/openstack_deploy  
cp openstack_user_config.yml.example openstack_user_config.yml
```

Dentro del archivo openstack_deploy configuraremos:

```
---  
cidr_networks:  
  container: 172.29.236.0/22  
  tunnel: 172.29.240.0/22  
  storage: 172.29.244.0/22  
  
used_ips:  
- "172.29.236.1,172.29.236.50"  
- "172.29.240.1,172.29.240.50"  
- "172.29.244.1,172.29.244.50"  
- "172.29.248.1,172.29.248.50"  
  
global_overrides:  
  external_lb_vip_address: 192.168.16.145  
  internal_lb_vip_address: 172.29.236.11  
  management_bridge: "br-mgmt"  
  
provider_networks:  
- network:  
  group_binds:  
    - all_containers  
    - hosts  
  type: "raw"  
  container_bridge: "br-mgmt"  
  container_interface: "eth1"  
  container_type: "veth"  
  ip_from_q: "container"  
  is_container_address: true  
- network:  
  group_binds:  
    - glance_api  
    - cinder_api  
    - cinder_volume  
    - nova_compute  
  type: "raw"  
  container_bridge: "br-storage"  
  container_type: "veth"  
  container_interface: "eth2"  
  ip_from_q: "storage"  
- network:  
  group_binds:  
    - neutron_linuxbridge_agent  
  container_bridge: "br-vxlan"  
  container_type: "veth"  
  container_interface: "eth10"  
  ip_from_q: "tunnel"  
  type: "vxlan"
```

```

    range: "1:1000"
    net_name: "vxlan"
- network:
  group_binds:
    - neutron_linuxbridge_agent
  container_bridge: "br-vlan"
  container_type: "veth"
  container_interface: "eth11"
  type: "vlan"
  range: "1:4094"
  net_name: "vlan"
- network:
  group_binds:
    - neutron_linuxbridge_agent
  container_bridge: "br-vlan"
  container_type: "veth"
  container_interface: "eth12"
  host_bind_override: "enp9s0"
  type: "flat"
  net_name: "flat"

###
### Infrastructure
###

_infrastructure_hosts: &infrastructure_hosts
infra1:
  ip: 172.29.236.11

# nova hypervisors
compute_hosts: &compute_hosts
compute1:
  ip: 172.29.236.21

# galera, memcache, rabbitmq, utility
shared-infra_hosts: *infrastructure_hosts

# repository (apt cache, python packages, etc)
repo-infra_hosts: *infrastructure_hosts

# load balancer
# Ideally the load balancer should not use the Infrastructure hosts.
# Dedicated hardware is best for improved performance and security.
haproxy_hosts: *infrastructure_hosts

###
### OpenStack
###

# Hosts on which to deploy the glance API, nova API, heat API, and horizon.
os-infra_hosts: *infrastructure_hosts

# keystone
identity_hosts: *infrastructure_hosts

```

```
# neutron server, agents (L3, etc)
network_hosts: *infrastructure_hosts

# cinder api services
storage-infra_hosts: *infrastructure_hosts

# cinder volume hosts (Ceph RBD-backed)
storage_hosts: *infrastructure_hosts
```

Colocar el nombre de la región, para este caso será RegionOne, cabe destacar que si es la RegionOne no es necesario configurar este archivo pero si para las otras regiones cuyo nombre no sea RegionOne.

```
sudo nano user_variables.yml
service_region: RegionOne
```

Procederemos a configurar las credenciales

```
cd /opt/openstack-ansible
./scripts/pw-token-gen.py --file /etc/openstack_deploy/user_secrets.yml
```

5.6.1. Copiando Keyrings de Ceph

En el nodo deployment se copian los archivos de configuración del cluster de Ceph, se debe ejecutar lo siguiente:

```
scp -r root@172.29.236.18:/etc/ceph/ /etc/openstack_deploy
mv ceph ceph-keyrings
```

5.6.2. Configurando la integración con Ceph

5.6.2.1. Personalización del entorno

Para los entornos de ceph, se debe ejecutar el cinder-volume en un contenedor. Por defecto cinder-volume se ejecuta en el host. Para lograrlo se debe de crear un archivo con el siguiente comando **nano /etc/openstack_deploy/env.d/cinder.yml** donde el contenido será el siguiente:

```
---
# This file contains an example to show how to set
# the cinder-volume service to run in a container.
#
```

```

# Important note:
# When using LVM or any iSCSI-based cinder backends, such as NetApp with
# iSCSI protocol, the cinder-volume service *must* run on metal.
# Reference: https://bugs.launchpad.net/ubuntu/+source/lxc/+bug/1226855

container_skel:
  cinder_volumes_container:
    properties:
      is_metal: false

```

5.6.2.2. Configuración de user_variables.yml

En el archivo `/etc/openstack_deploy/user_variables.yml` poner lo siguiente:

```

# OSA options for using an existing Ceph deployment. This example can be used
# if all configuration needs to come from OSA configuration files instead of
# the Ceph MONs.

# Directory containing the Ceph keyring files with access credentials.
ceph_keyrings_dir: /etc/openstack_deploy/ceph-keyrings

# General Ceph configuration file containing the information for Ceph clients
# to connect to the Ceph cluster.
ceph_conf_file: |
  [global]
  mon initial members = ceph1,ceph2,ceph3
  ## Ceph clusters starting with the Nautilus release can support the v2 wire protocol
  mon host =
[v2:172.29.244.18:3300,v1:172.29.244.18:6789],[v2:172.29.244.19:3300,v1:172.29.244.1
9:6789],[v2:172.29.244.20:3300,v1:172.29.244.20:6789]
  ## for a Ceph cluster not supporting the v2 wire protocol (before Nautilus release)
  # mon host = [v1:172.29.244.18:6789],[v1:172.29.244.19:6789],[v1:172.29.244.20:6789]

# For configuring the Ceph backend for Glance to store images in Ceph.
glance_ceph_client: glance
glance_default_store: rbd
glance_rbd_store_pool: images

# For configuring a backend in Cinder to store volumes in Ceph. This
# configuration will be used for Nova compute and libvirt to access volumes.
cinder_ceph_client: cinder

cinder_backends:
  rbd:
    volume_driver: cinder.volume.drivers.rbd.RBDDriver
    rbd_pool: volumes
    rbd_ceph_conf: /etc/ceph/ceph.conf
    rbd_store_chunk_size: 8
    volume_backend_name: rbd
    rbd_user: "{{ cinder_ceph_client }}"
    rbd_secret_uuid: "{{ cinder_ceph_client_uuid }}"
    report_discard_supported: true

```

```
# Configuration for Nova compute and libvirt to store ephemeral discs in Ceph.  
nova_libvirt_images_rbd_pool: vms
```

5.7. Ejecución de playbooks

El siguiente paso es ejecutar los playbooks, el tiempo de ejecución dependerá de diferentes factores como la conectividad entre los compute, la conexión a internet y los recursos de los nodos.

Nota: El tiempo que se experimentó durante el despliegue de este proyecto fue entre 1 hora y 1 hora con 30 minutos.

```
cd /opt/openstack-ansible/playbooks  
openstack-ansible setup-infrastructure.yml --syntax-check  
openstack-ansible setup-hosts.yml  
openstack-ansible setup-infrastructure.yml  
openstack-ansible setup-openstack.ym
```

6. Integración de las regiones a la infraestructura de nube

6.1. Configurando Keystone para la administración de varias regiones

En el nodo controller de la infraestructura ejecutar lo siguiente:

```
keystone-manage bootstrap --bootstrap-password icc1152023 \  
--bootstrap-admin-url http://10.0.0.11:5000/v3/ \  
--bootstrap-internal-url http://10.0.0.11:5000/v3/ \  
--bootstrap-public-url http://10.0.0.11:5000/v3/ \  
--bootstrap-region-id RegionTwo
```

```
keystone-manage bootstrap --bootstrap-password icc1152023 \  
--bootstrap-admin-url http://10.0.0.11:5000/v3/ \  
--bootstrap-internal-url http://10.0.0.11:5000/v3/ \  
--bootstrap-public-url http://10.0.0.11:5000/v3/ \  
--bootstrap-region-id RegionThree
```

Siempre en el nodo controller de la infraestructura se deberá integrar uno por uno los diferentes servicios de openstack de la siguiente forma.

6.2. Servicio de glance

```
openstack endpoint create --region RegionTwo image public https://192.168.100.212:9292  
openstack endpoint create --region RegionTwo image admin http://172.30.236.12:9292  
openstack endpoint create --region RegionTwo image internal http://172.30.236.12:9292  
  
openstack endpoint create --region RegionThree image public https://192.168.1.211:9292  
openstack endpoint create --region RegionThree image admin http://172.29.236.11:9292  
openstack endpoint create --region RegionThree image internal http://172.29.236.11:9292
```

6.3. Servicio de placement

```
openstack endpoint create --region RegionTwo placement public  
https://192.168.100.212:8780  
openstack endpoint create --region RegionTwo placement admin  
http://172.30.236.12:8780  
openstack endpoint create --region RegionTwo placement internal  
http://172.30.236.12:8780  
  
openstack endpoint create --region RegionThree placement public  
https://192.168.1.211:8780  
openstack endpoint create --region RegionThree placement admin  
http://172.29.236.11:8780  
openstack endpoint create --region RegionThree placement internal  
http://172.29.236.11:8780
```

6.4. Servicio de nova

```
openstack endpoint create --region RegionTwo nova public
https://192.168.100.212:8774/v2.1
openstack endpoint create --region RegionTwo nova admin http://172.30.236.12:8774/v2.1
openstack endpoint create --region RegionTwo nova internal
http://172.30.236.12:8774/v2.1

openstack endpoint create --region RegionThree nova public
https://192.168.1.211:8774/v2.1
openstack endpoint create --region RegionThree nova admin
http://172.29.236.11:8774/v2.1
openstack endpoint create --region RegionThree nova internal
http://172.29.236.11:8774/v2.1
```

6.5. Servicio de neutron

```
openstack endpoint create --region RegionTwo network public
https://192.168.100.212:9696
openstack endpoint create --region RegionTwo network admin http://172.30.236.12:9696
openstack endpoint create --region RegionTwo network internal http://172.30.236.12:9696

openstack endpoint create --region RegionThree network public
https://192.168.1.211:9696
openstack endpoint create --region RegionThree network admin http://172.29.236.11:9696
openstack endpoint create --region RegionThree network internal
http://172.29.236.11:9696
```

6.6. Servicio de cinderv3

```
openstack endpoint create --region RegionTwo volumev3 public
https://192.168.100.212:8776/v3/%(tenant_id)s
openstack endpoint create --region RegionTwo volumev3 admin
http://172.30.236.12:8776/v3/%(tenant_id)s
openstack endpoint create --region RegionTwo volumev3 internal
http://172.30.236.12:8776/v3/%(tenant_id)s

openstack endpoint create --region RegionThree volumev3 public
https://192.168.1.211:8776/v3/%(tenant_id)s
openstack endpoint create --region RegionThree volumev3 admin
http://172.29.236.11:8776/v3/%(tenant_id)s
openstack endpoint create --region RegionThree volumev3 internal
http://172.29.236.11:8776/v3/%(tenant_id)s
```

En los nodos de la región, en el control plane se configuraran los diferentes servicios, tomar en cuenta que estos son contenedores por lo que entraremos en cada uno a configurarlo, dentro de los archivos de configuración buscaremos diferentes secciones para configurarlos y que apunten a la infraestructura, para el proyecto la infraestructura estaba en

la ip: 10.0.0.11 y la contraseña icc1152023, por lo que se procederá a explicar los archivos y las secciones que se deben modificar en cada uno.

6.7. Servicio Glance

```
# lxc-attach -n infra1_glance_container-2905931d
# nano /etc/glance/glance-api.conf

[keystone_authtoken]
auth_url = http://10.0.0.11:5000/v3
www_authenticate_uri = http://10.0.0.11:5000
password = icc1152023
memcached_servers = 10.0.0.11:11211
```

6.8. Servicio Placement

```
# lxc-attach -n infra1_placement_container-13c9a979
# nano /etc/placement/placement.conf

[keystone_authtoken]
auth_url = http://10.0.0.11:5000
www_authenticate_uri = http://10.0.0.11:5000
password = icc1152023
memcached_servers = 10.0.0.11:11211
```

6.9. Servicio Nova

6.9.1. Nodo Control-Plane

```
# lxc-attach -n infra1_nova_api_container-f435cf7c
# nano /etc/nova/nova.conf

[DEFAULT]
#reserved_host_memory_mb = 2048

[neutron]
password = icc1152023
auth_url = http://10.0.0.11:5000/v3

[placement]
password = icc1152023
auth_url = http://10.0.0.11:5000/v3

[keystone_authtoken]
auth_url = http://10.0.0.11:5000
www_authenticate_uri = http://10.0.0.11:5000
password = icc1152023
memcached_servers = 10.0.0.11:11211
```

6.9.2. Nodos compute

En los nodo compute, se deberá configurar únicamente el archivo nova.conf

```
sudo nano /etc/nova/nova.conf

[DEFAULT]
#reserved_host_memory_mb = 2048

[neutron]
password = icc1152023
auth_url = http://10.0.0.11:5000/v3

[placement]
password = icc1152023
auth_url = http://10.0.0.11:5000/v3

[keystone_authtoken]
auth_url = http://10.0.0.11:5000
www_authenticate_uri = http://10.0.0.11:5000
password = icc1152023
memcached_servers = 10.0.0.11:11211
```

6.10. Servicio Neutron

```
# lxc-attach -n infra1_neutron_server_container-aa137241
# nano /etc/neutron/neutron.conf

[placement]
password = icc1152023
auth_url = http://10.0.0.11:5000/v3

[nova]
password = icc1152023
auth_url = http://10.0.0.11:5000/v3
#endpoint_type = internal

[keystone_authtoken]
auth_url = http://10.0.0.11:5000
www_authenticate_uri = http://10.0.0.11:5000
password = icc1152023
memcached_servers = 10.0.0.11:11211
```

6.11. Servicio Cinder

Nota: Este servicio se divide en 2 contenedores

```
# lxc-attach -n infra1_cinder_api_container-2f4031a1
# nano /etc/cinder/cinder.conf

[keystone_authtoken]
auth_url = http://10.0.0.11:5000
www_authenticate_uri = http://10.0.0.11:5000
```

```
password = icc1152023  
memcached_servers = 10.0.0.11:11211
```

```
[nova]  
password = icc1152023  
auth_url = http://10.0.0.11:5000/v3
```

```
# lxc-attach -n infra1_cinder_volumes_container-72d38cb6  
# nano /etc/cinder/cinder.conf
```

```
[keystone_authtoken]  
auth_url = http://10.0.0.11:5000  
www_authenticate_uri = http://10.0.0.11:5000  
password = icc1152023  
memcached_servers = 10.0.0.11:11211
```

```
[nova]  
password = icc1152023  
auth_url = http://10.0.0.11:5000/v3
```