UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE INGENIERÍA DE SISTEMAS

Tesis previa a la obtención del título de Ingeniero de Sistemas.

TÍTULO

"Análisis, Diseño e Implementación de Cloud Computing para una Red de Voz sobre IP".

Autores:

Adriana Marcela Cornejo Orellana

Christian Fernando Díaz Escalante

Director:

Ing. Pablo Gallegos

CUENCA, 2015

ECUADOR

CERTIFICACIÓN

Ing. Pablo Gallegos Segovia.

Certifico:

Haber dirigido y revisado prolijamente cada uno de los capítulos de la tesis: ***"Análisis, Diseño e Implementación de Cloud Computing para una Red de Voz sobre IP"***, realizado por os estudiantes Adriana Marcela Cornejo Orellana y Christian Fernando Díaz Escalante y por cumplir los requisitos autorizo su presentación.

Cuenca, Marzo del 2015

Atentamente

Ing. Pablo Gallegos Segovia

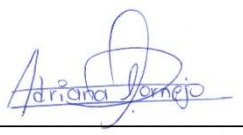
DIRECTOR DE LA TESIS

DECLARATORIA DE RESPONSABILIDAD

Nosotros, Adriana Marcela Cornejo Orellana y Christian Fernando Díaz Escalante, declaramos bajo juramento que el informe de tesis presentado a continuación es de nuestra autoría, en donde hemos consultado las referencias bibliográficas incluidas en el presente documento.

A través de la presente declaración autorizamos los derechos de propiedad intelectual, a la Universidad Politécnica Salesiana según la ley de Propiedad Intelectual, por su Reglamento y por la Normativa Institucional Vigente.

Cuenca, Marzo del 2015.



Adriana Marcela Cornejo Orellana

Christian Fernando Díaz Escalante

C.I: 0103962056 C.I: 0104774393

DEDICATORIA

Quiero dedicar esta tesis, a todos aquellos que me han brindado su apoyo a lo largo de mi carrera universitaria, a mi director de tesis por el apoyo brindado, a mis compañeros de clases, a mis amigos que han sido soporte en tiempos difíciles, a mis maestros que fueron fomentando, de a poco lo necesario para generar un gran profesional, y en especial a mi familia que durante todo este tiempo siempre han estado ahí apoyándome y creyendo en mí.

Christian Fernando Díaz Escalante

Quiero agradecer a Dios por regalarme la vida y permitirme compartir este logro con mi familia y las personas que me conocen, por no dejarme ni un solo momento sin su protección y por ser mi compañero fiel en todos los momentos de mi vida.

A mi familia en especial a mis padres: Luis y Anita, hermanos: Valeria y Byron y a mi abuelita Alejandrina, que durante todos estos años me han apoyado para cumplir con todas las metas que me he propuesto y me han guiado para no dejarme vencer ante cualquier obstáculo.

A mis maestros, director de tesis, compañeros y amigos que durante toda mi formación han aportado con conocimientos que me ayudarán tanto en mi desarrollo profesional como en la vida misma.

Adriana Marcela Cornejo Orellana.

AGRADECIMIENTOS

Quiero agradecer a todos los que han sido parte de esta formación académica, a mi director de tesis, a mis maestros, a mis amigos que han estado ahí apoyándome, y a mi familia que siempre supo darme fuerza y alentarme para seguir adelante, en mi carrera universitaria.

Christian Fernando Díaz Escalante.

Agradezco a Dios, mi familia, maestros, director de tesis, compañeros y amigos, y a la Universidad Politécnica Salesiana por permitirme en sus aulas adquirir conocimientos tanto profesionales como para la vida.

Adriana Marcela Cornejo Orellana.

INDICE

Contenido

ÍNDICE DE IMÁGENES

Índice de Tablas

CAPÍTULO 1

1. CLOUD COMPUTING

1.1. Concepto de Cloud Computing

Cloud Computing o Computación en la nube se refiere a un paradigma nuevo dentro de las redes de comunicaciones, ante este nuevo término se analizarán algunos conceptos definidos por organizaciones de estándares de tecnologías como la IEEE[1] y la NIST[2].

- Según la IEEE Computer Society define al Cloud Computing como: "*Un paradigma en el cual la información permanente es almacenada en servidores en Internet y colocada ("cache") temporalmente en clientes que incluyen computadoras de escritorio, centros de entrenamiento, tablets, notebooks, laptops, y dispositivos portátiles, etc.*" [1] y según el autor: *Cloud computing is an information-processing model in which centrally administered computing capabilities are delivered as services, on an as-needed basis, across the network to a variety of user-facing devices",* en español: *La computación en nube es un modelo de procesamiento de información en el que las capacidades de computación administrados centralmente son entregados como servicios, en una función de las necesidades, a través de la red a una variedad de dispositivos de cara al usuario* [1].
- Según el Instituto Nacional de Estándares y Tecnología de los Estados Unidos (NIST) la define como: *"Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.".* [2] En español: *"Un modelo que permite el acceso conveniente a una red bajo demanda que comparte un conjunto de recursos informáticos configurables (por ejemplo redes, servidores, aplicaciones y servicios) que pueden ser rápidamente almacenados, provistos y lanzados con el mínimo esfuerzo de administración o interacción con el proveedor de servicios.*" [2].
- Los autores Erl, Mahmood y Puttini, en el libro: "Cloud Computing Concepts, Technology & Architecture" definen Cloud Computing como: "*Cloud computing is a specialized form of configurable Computing that introduces utilization models for remotely provisioning scalable and measured resources" en español:"La computación en la nube es una forma especializada de Informática configurable que presenta modelos de utilización para el aprovisionamiento de forma remota de los recursos escalables y medidos"* [3].

[1]IEEE: Institute of Electrical and Electronics Engineers (Instituto de Ingenieros Eléctricos y Electrónicos)
[2]NIST: National Institute of Standards and Technology (Instituto Nacional de Estándares y Tecnologías de los Estados Unidos)

Por tanto "La Computación en la Nube es un modelo remoto de gestión de servicios que busca el manejo de información en una red bajo demanda, que puede brindar tanto servicios, como también infraestructura al cliente, minimizando el trabajo del mismo y facilitando la administración de las aplicaciones y recursos, a través de las tecnologías actuales y desde cualquier ubicación, sin restricción alguna de dispositivos o tiempo".

La Computación en la Nube busca incrementar el número de servicios que se pueden ofrecer en una red, con la mayor rapidez y eficiencia, reduciendo costos en los proveedores como para los clientes; lo que se busca es garantizar el funcionamiento de varios servicios simultáneos como voz, video y datos; para a través de la flexibilidad, la escalabilidad y sobre todo la reducción de los costos de implementación que brinda la Computación en la nube, cumplir con la demanda de servicios que actualmente se requieren en el mercado [4].

El principal objetivo de Cloud Computing es resolver problemas computacionales de gran escala, a través de la optimización de recursos distribuidos que combinados logran un mejor rendimiento de la red; ya que su implementación representa a un sistema distribuido que a través de la virtualización permite cumplir con los estándares de rendimiento establecidos [5].

Actualmente las grandes empresas de TI han destinado sus recursos para la implementación de la computación en el nube, debido a la gran demanda tanto de servicios de hardware como de software que esta tecnología representa, algunas empresas son HP[3], Intel, Microsoft, Oracle, IBM[4] [6].

1.2. Características de Cloud Computing

Según el NYST, el Cloud Computing debe cumplir con cinco características principales que consideran la implementación como un servicio Cloud. Estas características son: auto-servicio bajo demanda, acceso a la red extensa (www), agrupación de recursos, rápida elasticidad y servicio medido [2].

- **Auto Servicio Bajo Demanda:** El cliente puede solicitar y recibir acceso al servicio ofrecido, sin necesidad de un administrador que le de soporte. Se emplea en la aplicación de (API[5]s) de usuario como interfaces, se deben tener en cuenta que estos servicios requieren control de usuario y al no ser completamente automáticos se deben analizar los procesos anteriores antes de su uso.

[3] HP: Hewlett Packard, Empresa líder en ventas de equipos computacionales y mantenimiento http://www8.hp.com/ec/es/home.html

[4] IBM: International Business Machines Corp, es una empresa multinacional estadounidense de tecnología y consultoría. http://www.ibm.com/

[5] API: Application Programming Interface o Interfaz de Programación de Aplicaciones son las funciones de una biblioteca sobre otro software.

- **Acceso a la Red Extensa:** Con esta característica se permite el acceso desde cualquier lugar a la red, en donde los usuarios debería contar con una conexión sencilla a internet para conectarse a aplicaciones y servicios. Para la conexión de usuarios existe gran cantidad de dispositivos que en la actualidad, reemplazan tanto a laptops y equipos de escritorio, debido a que ahora se utilizan tablets, smartphones, y dispositivos mucho más rápidos.

- **Compartición de Recursos:** Permite la reducción de costos y da flexibilidad a los usuarios de acceder a los recursos disponibles, gracias a la virtualización, que permite establecer múltiples sesiones. Los usuarios pueden acceder a los servicios sin la necesidad de conocer su ubicación física, simplemente los recursos están concentrados de acuerdo a la infraestructura aplicada.

- **Las aplicaciones son independientes del hardware:** Las aplicaciones pueden ser usadas simultáneamente entre varios usuarios, sin tomar en cuenta los equipos usados, muchas veces los servicios son levantados sobre máquinas virtuales con características muy parecidas a las físicas.

- **Rapidez y Elasticidad:** Permite el crecimiento rápido de la red para satisfacer la demanda, esto incluye recursos como computadoras, discos duros, entre otros. El objetivo es cumplir con los requerimientos solicitados por los usuarios, en el menor tiempo posible.

- **Servicio Ponderado:** Los servicios Cloud se pueden medir en su implementación, gracias a esto se puede usar varios parámetros de análisis como: ancho de banda, tiempo de uso, uso de datos, etc.

Según los conceptos analizados la Computación en la Nube ofrece varias características que la diferencian de otros modelos de TI que existen en la actualidad. Entre las cuales están:

- **Usuarios pueden acceder a los servicios de la nube sin importar el lugar o el dispositivo en el que esté.**

Esto indica la facilidad con la que los usuarios pueden acceder a la información, sin importar la ubicación, pudiendo ser esta en la propia empresa como también desde cualquier parte del mundo, dependen las restricciones establecida por el administrador de la nube, así como el dispositivo empleado, como computadoras personales, laptops, smartphones, que permiten acceder de forma flexible y económica a los servicios brindados por el Cloud Computing.

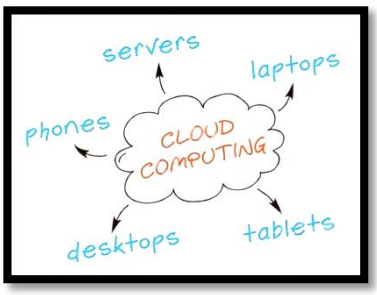


Imagen 1. 1: Clientes de a nube [7]

- **La infraestructura es suministrada por terceros y se accede a través de internet a los servicios provistos.**

El Cloud Computing maneja un modelo de negocio mediante el pago por consumo de servicios, este es administrado por un tercero, el cual se encarga de únicamente de los servicios prestados y en donde los clientes pueden acceder a ellos desde cualquier lugar con una conexión a internet [7].

Imagen 1. 2: Acceso a través de terceros [9]

- **Reducción de costos debido que el cliente no adquiere la infraestructura, paga por uso de la infraestructura provista por terceras personas.**

El cliente paga solo por los servicios que usa, es decir, por la infraestructura esto reduce los costos notablemente. La virtualización representa un papel importante, ya que se reducen las compras de servidores, y por consiguiente el mantenimiento del mismo lo que implica un ahorro bastante rentable para empresas que planean administrar sus servicios a través de una nube sea esta pública o privada.

Imagen 1. 3: Reducción de costos de mantenimiento

- **Los conocimientos informáticos requeridos por el cliente son mínimos.**

El usuario no debe tener conocimientos avanzados de cómo funciona internamente la nube, ya que solamente usa el servicio y se limita a conocer cómo usarlo. Claramente en esta característica se manifiesta que el cliente final solamente requiere conocimientos mínimos de cómo funciona la nube internamente, sino el funcionamiento el servicio final que está desplegando sobre la nube, como por ejemplo si se instala un servidor de voz, el cliente debería saber cómo configurar y administrar este servicio para posteriormente analizar su comportamiento dentro de la nube.

Imagen 1. 4: Fácil acceso para los clientes

- **Las aplicaciones que brinda la nube pueden ser accedidas por cualquier usuario sin la necesidad de instalarlas en sus computadores.**

Los usuarios accederán a los servicios a través de una conexión de internet y mediante un navegador web, por lo tanto no es necesario que instale ningún aplicativo extra en sus ordenadores. Esta es una de las características más beneficiosas, ya que de ocurrir algún inconveniente, el administrador y encargado de la nube podría acceder desde cualquier lugar siempre y cuando las opciones de acceso estén debidamente configuradas y no comprometan a la empresa en ningún sentido.

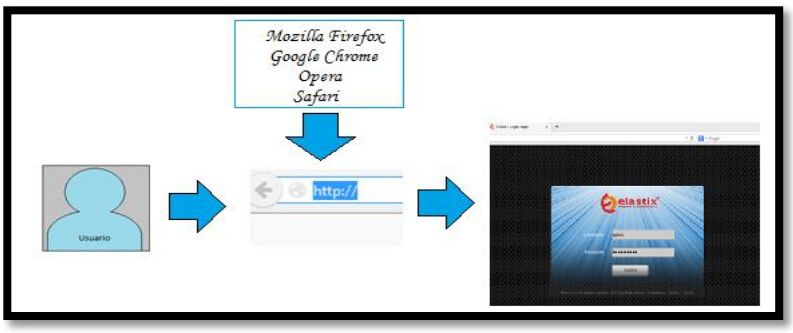


Imagen 1. 5: Fácil acceso de usuarios a la aplicaciones

- **El mantenimiento es flexible y fácil de realizar.**

El mantenimiento no representa gastos adicionales ya que anteriormente se necesitaban servidores físicos, por lo que el mantenimiento tenía un costo adicional pero en la actualidad, debido a que toda la infraestructura está virtualizada, y se ubica en un solo espacio físico, en donde el administrador puede administrarla fácilmente, además que la virtualización brinda flexibilidad en la administración de la infraestructura.

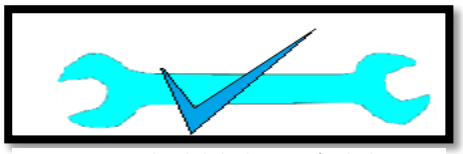


Imagen 1. 6: Flexibilidad a través de la virtualización [10]

1.3. Modelos de Servicio de Cloud Computing

Usar un modelo de servicio hace referencia a la utilización de componentes a los que se puede acceder desde el internet, un modelo ofrece escalabilidad y permite el compartir recursos entre varios usuarios debido a la autonomía de hardware, que es accedido por los usuarios. Los clientes o usuarios finales son los encargados de elegir la mejor opción en cuanto al modelo aplicado dentro de la organización ya que en la implementación se requieren recursos tanto físicos como de inversión y que van de acuerdo a los servicios que serán aplicados. Según el NIST [2] los modelos de servicio son tres representados en la ilustración. 1.7 [8].

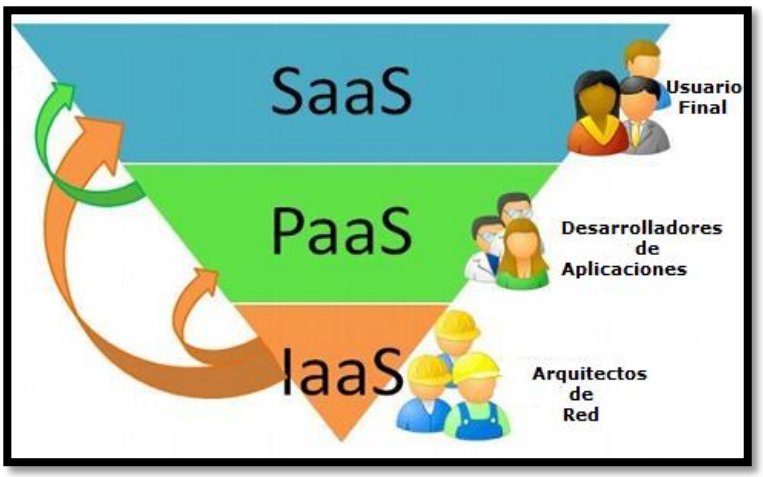


Imagen 1. 7: Modelos de servicio de la nube [12]

1.3.1. Servicios por Software

SaaS (Software as a Service) o Software como Servicio, este se refiere al modelo en el cual un proveedor ofrece software como servicio hacia una o varias organizaciones, para lo cual distribuye costos a únicamente las licencias que permiten el uso de su software como servicios bajo demanda. El cliente administra las aplicaciones sin preocuparse de la infraestructura sobre cual está desarrollado el software, ya sean servidores o sistemas operativos; las aplicaciones de software pueden ser accedidas desde un navegador mediante una conexión a internet. Es el modelo original de servicios de Cloud [5]. Las aplicaciones SaaS sirven para brindar servicios a clientes empresariales ya que reducen los costos tanto de instalación como de mantenimiento [6].

El modelo SaaS es parecido a los servidores de aplicación en donde se brindan tan solo los servicios finales como puede ser: un ASP[6], debido a que el software es instalado en computadores de usuarios y en muchas implementaciones le conocen como: "software como producto" [9].

En este modelo se accede a las aplicaciones de varias formas como navegadores web o enlaces, que apuntan a los servidores en donde están las aplicaciones listas para ser usadas por el usuario, algunos son GoogleDrive, Dropbox, Gmail, y aplicaciones ERP[7], CRM[8] online.

[6] ASP: Application Service Provider o Proveedor de Servicios de Aplicación, es una empresa que ofrece servicios a través de internet.

[7] ERP Enterprise Resource Planning o Sistemas de Planificación de Recursos Empresariales permite la administración de recursos en una organización.

[8] CRM Customer Relationship Management o Administración Basada en la Relación con los Clientes permite la adminsitracion de las actividades de los clientes.

Características Saas [6]

- El acceso a la red es libre, a través de internet el usuario puede acceder de forma remota a los servicios.
- La distribución de servicios permite actuar como modelo Cliente-Servidor, debido que se aplica el modelo de un servidor con varios clientes.
- Las actualizaciones están centralizadas y permiten que los sistemas no requieran muchos cambios cuando exista una actualización o algún cambio

Este modelo de servicio define varios tipos de clientes para ofrecer sus servicios [4]:

- Particulares
    - Ofimática bajo demanda.
    - Web 2.0 (Aplicaciones WEB)
- Profesionales
    - ERP online
    - CRM online

Imagen 1. 8: Ejemplo de software Saas [14]

1.3.2. Servicios por Plataforma

PaaS (Plataform as a Service) este modelo consiste en el desarrollo de aplicaciones propias para la organización, el cliente cuenta con los recursos físicos de acceso a una plataforma en donde él se encarga de desarrollar aplicaciones y administrar los servicios de acuerdo a sus requerimientos, sin necesidad de adquirir el hardware.

La nube cuenta con los recursos necesarios para crear aplicaciones sin disponer de una infraestructura física o virtual, que es ofrecida por el proveedor quien se encarga de administrar la plataforma y las herramientas [6]. El cliente únicamente dispone de las herramientas ofrecidas y se encarga de desarrollar una aplicación o servicio, en donde puede utilizar alguna de las plataformas como: .NET, Java, Python, PHP, Ajax, Ruby onRails[9], etc [9].

Características de PAAS [6]**:**

- Permite la administración de Servicios WEB que se desarrollan, prueban, despliegan y alojan servidores WEB.
- La Integración de Servicios genera flexibilidad ya que combina servicios Web y Bases de Datos.
- La compartición de recursos, permite participar con código entre varios desarrolladores.

Los desarrolladores de las aplicaciones se deben preocupar de que estas funcionen de forma optimizada con el fin de consumir menos recursos, incluye los siguientes servicios: desarrollo, prueba, despliegue, alojamiento [4].

El usuario tiene todo el control sobre los servicios que creó, más no sobre la infraestructura sobre el que fue desplegado [10].

Un ejemplo de esta aplicación de modelo es Google App Engine, que permite desarrollar aplicaciones en java y python.

[9]. NET, Java, Python, PHP, Ajax, Ruby onRails: Lenguajes de Programación para la creación de servicios provistos por la nube.

Imagen 1. 9: Ejemplo de PaaS -> Google Engine

\

1.3.3. Servicios por Infraestructura

IaaS (Infraestructure as a Service) el proveedor brinda al cliente la infraestructura para el uso de uno o más servicios, el cliente solo se encarga de administrar el procesamiento, almacenamiento, asignación de recursos, y equipos en donde se encuentran desplegados los servicios [11].

Este modelo es usado para reducir los costos que implican gastos de tecnologías, como son: servidores, routers, switchs, equipo, debido a esto los costos referentes a este modelo están dados en base al consumo de recursos que el cliente realice con el fin de reemplazar el funcionamiento de Datacenters [6].

Un ejemplo de infraestructura como servicio es Amazon Web Services, que permite administrar máquinas virtuales en la nube, es decir, darles las características de acuerdo a los requerimientos sin importar el hardware, ya que este es manejado de forma virtual para mayor facilidad de administración [12].

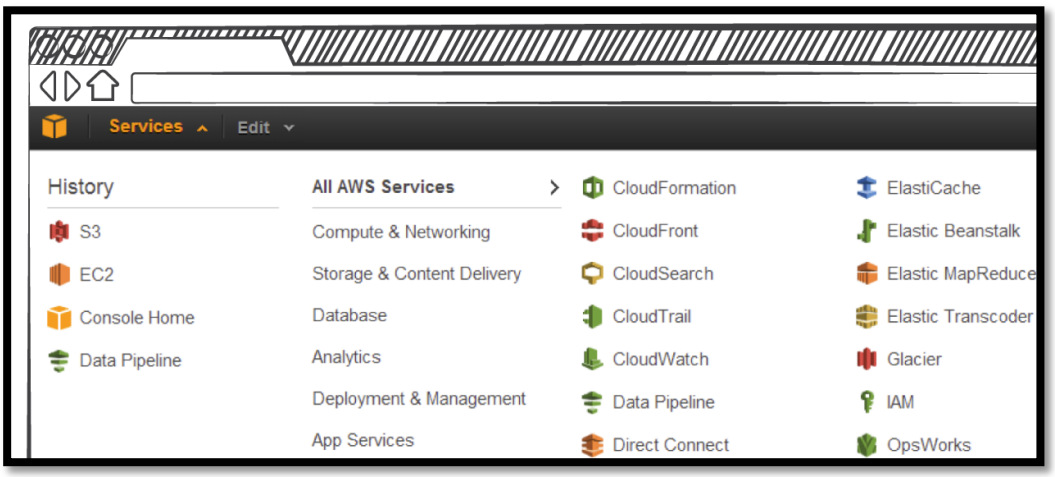


Imagen 1. 10: Ejemplo de servicio [18]

En este modelo de servicio encontramos a Openstack, que nos brinda servicios de infraestructura, en donde se pueden tener los servicios que se deseen sobre plataformas propias creadas por el administrador.

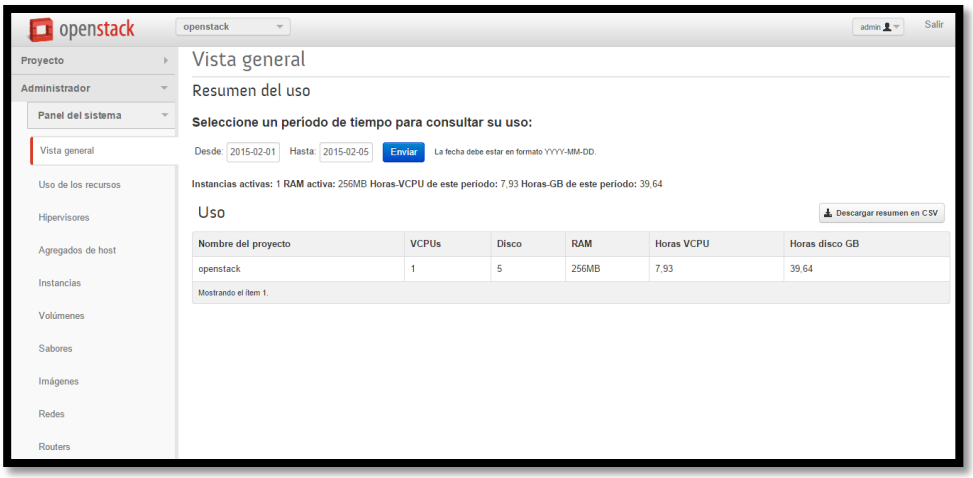


Imagen 1. 11: Ejemplo IaaS -> Openstack [19]

1.4. Modelos de Despliegue de Cloud Computing

Los modelos de despliegue de una nube se caracterizan por la manera como brinda el acceso a un nodo privado, o acceso a grandes cantidades de usuarios que pueden acceder a la información a través de acceso enlaces directos o remotos, para lo cual se han definido 4 modelos de despliegue [2] [9] [11] :

Imagen 1. 12: Modelos de despliegue de la nube [20]

1.4.1. Cloud Privada

Una nube privada es la infraestructura que solamente una empresa maneja o administra, puede estar tanto dentro de la empresa o fuera de ella, generalmente está ubicada dentro de las instalaciones de la empresa.

Una de sus mayores ventajas, es la que permite ofrecer solamente los servicios requeridos por la organización propietaria, además cuenta con las características de una nube pública, pero lo más importante, es que se cuenta con el control completo sobre los datos, accesos, procesos y aplicaciones que se desplieguen sobre la misma [9] [11].

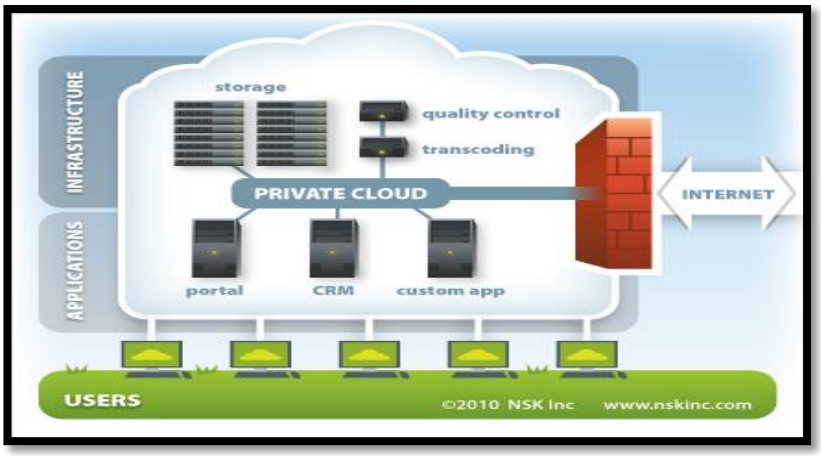


Imagen 1. 13: Nube privada [21]

Su infraestructura a diferencia de una nube pública no debe ser compartida con ninguna otra empresa, y de ahí el nombre de privada.

Su despliegue es muy parecido a un Datacenter, pues se cuenta con la infraestructura, máquinas propias, y su crecimiento está en base a la demanda [11]. Las nubes privadas están diseñadas para alta concentración de recursos y sistemas tecnológicos, por ejemplo es empleado en: entidades bancarias, administración pública, entornos de investigación y desarrollo, tecnológicas o de negocio, etc [13].

Imagen 1. 14: Características de una nube privada [20]

Este modelo presenta varias ventajas y desventajas, como [11]:

- **Ventajas**
  - Debido a que la infraestructura se encuentra dentro de la empresa, no se debe tomar a la seguridad como un gasto adicional.
  - Los controles de seguridad son mucho más concretos y no tienen ninguna restricción, ya que toda la infraestructura es propiedad de la empresa.
  - La organización mantiene un control total sobre el Datacenter.
- **Desventajas**
  - El costo de implementación es alto.
  - Se debe adquirir tanto hardware como software.

1.4.2. Cloud Pública

Una nube pública, es un modelo de implementación de uso general, que busca que todos tengan acceso a la información, servicios o aplicaciones, y pertenece generalmente a una organización que brinda este servicio [14].

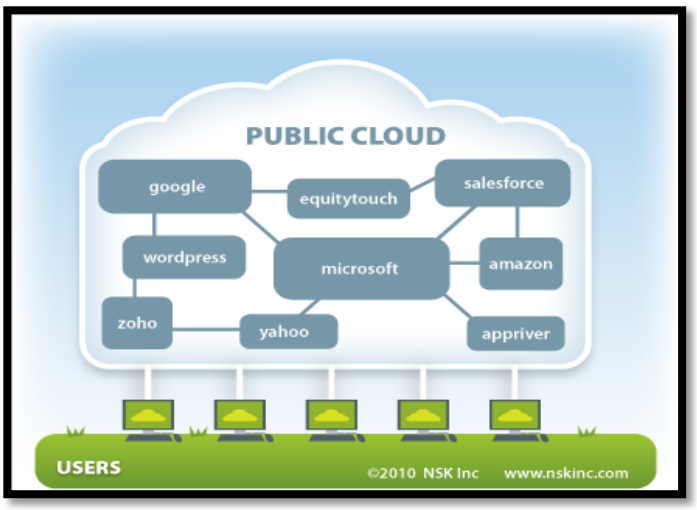


Imagen 1. 15: Nube pública [23]

La infraestructura es compartida entre varios clientes a través de internet o conexiones VPN[10], se aplica el modelo de pago por consumo en donde se paga la infraestructura entre estos mismos clientes [4].

Las plataformas son propias y pueden crecer y disminuir de acuerdo a las necesidades del cliente, pagando solamente los recursos utilizados, y sin necesidad de re-dimensionar los recursos [13].

Imagen 1. 16: Características de una nube [20]

[10] VPN: Virtual Private Network o Red Privada Virtual ayuda a las organizaciones a ampliar la conectividad de forma segura y económica y a mejorar la velocidad.

La infraestructura está alojada dentro de un Datacenter propiedad de una organización proveedora, a la cual acceden varios usuarios, por lo que representa un modelo escalable y el costo es bajo demanda [15].

Algunos servicios de Cloud público son:

- Cloud Hosting, aplicación de VPS[11] para atender alta demanda.
- Cloud on Demand, los recursos se ajustan a la demanda.
- Cloud Pool, permite un entorno con una disponibilidad alta.

Este modelo presenta varias ventajas y desventajas, como [11]:

- **Ventajas**
  - Bajo costo de inversión.
  - Se paga por lo que se usa.
  - Se puede realizar un escalado de las aplicaciones y podrían correr sobre cualquier sistema operativo.
  - Ofrece un entorno de pruebas para aplicaciones en inminente desarrollo.
- **Desventajas**
  - La compartición de recursos que brinda una nube pública, puede con llevar a varios problemas de seguridad.
  - Las organizaciones no desean tener limitaciones en el manejo de la infraestructura.

1.4.3. Cloud Híbrida

Una Nube Hibrida, consiste en la combinación de aplicaciones de una nube pública con las de una nube privada, con esto se puede mantener el control de sus aplicaciones [9].

Este tipo de Cloud se aplica debido a la necesidad de clientes que contando con infraestructura propia buscan aprovechar las ventajas de un proveedor externo que también implementa ciertos servicios que serán ofrecidos a otras empresas u organizaciones.

11 VPS: Virtual Private Server o Servidor Privado Virtual es un método de particionar un servidor físico en varios servidores

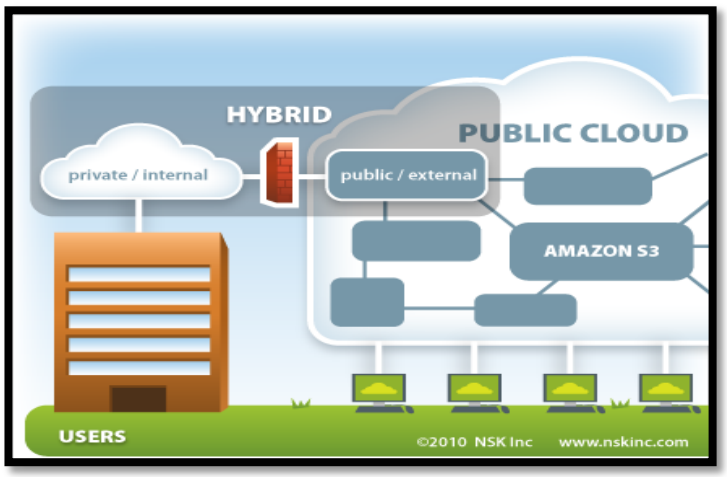


Imagen 1. 17: Nube Híbrida [25]

Una nube hibrida reúne características de las nubes pública y privada, por lo cual presenta una serie de ventajas y desventajas [11]:

- **Ventajas**
  - Ofrece flexibilidad ya que permite que la información de la empresa se mantenga dentro de las instalaciones, mientras que lo menos relevante puede ser alojado en una nube pública.
  - Es una solución aplicable si no se cuenta con los recursos necesarios para tener exclusivamente una nube privada con todos los servicios que se piensan brindar a los usuarios.
- **Desventajas**
  - La implementación de la seguridad, debido a los protocolos que se maneja entre las mismas, deben combinarse y administrarse por dos empresas diferentes.

1.5. Arquitectura Básica de Cloud Computing

La arquitectura básica del Cloud Computing distingue varias capas, que se pueden implementar en nubes públicas, privadas e hibridas. La arquitectura está basada en la arquitectura de red, debido a que Cloud Computing utiliza los mismos protocolos, por ejemplo en nubes privadas que no tienen conexión con la WEB, como en nubes públicas que cuentan con salidas a internet.

Se caracterizan porque al desplegar en redes propias como arrendadas tienen una arquitectura genérica, que cuenta con cinco capas principales [16].

Imagen 1. 18: Capas básicas del cloud computing [6]

La arquitectura genérica para Cloud Computing, tienen las siguientes capas de abajo hacia arriba:

- **Recursos físicos:** Esta capa está compuesta por elementos físicos como servidores, almacenamiento y red, es decir, todo el hardware que interviene en la nube. Estos elementos pueden ser:
    - **CPU, discos duros, memorias:** Estos elementos se encargan del procesamiento de la información, por lo que representan la parte más importante de Cloud Computing.
    - **Redes:** Son los elementos de red que se encargan del transporte hacia los medios de almacenamiento.
    - **Equipos de Enfriamiento:** Se encargan de mantener los elementos que intervienen en la nube, a una temperatura considerable, para evitar fallas o recalentamiento por uso.
    - **Redundancia:** Aquí intervienen los respaldos y recuperación ante desastres, o por fallas como cortes de luz, caídas de servidores o sobrecarga de datos.
- **Virtualización:** Esta capa se encarga de la infraestructura virtual como un servicio, aquí están los virtualizadores que nos ayudan con la emulación de los recursos físicos.
- **Infraestructura:** Esta capa es la encargada de administrar el software de plataforma como servicio, como puede ser en el caso de la aplicación Openstack, Cloudstack.[12]
- **Plataforma:** En esta capa están los componentes de aplicación como servicio, aquí estarían los módulos o componentes en donde se despliegan las aplicaciones,

[12] Cloudstack: Software para crear una Infraestructura de Cloud Computing.

por ejemplo en el caso de Openstack, Keystone, que es el módulo de autenticación, Dashboard que es la interfaz de usuario, también conocido como Horizon.

- **Aplicación:** En esta capa se incluyen los servicios basados en web y software como servicio, en el caso de la tesis, la aplicación que se está administrando es Asterisk[13], que es un servidor de VoIP que nos permitirá administrar telefonía IP, dentro de la nube [7].

La ilustración 1.19, detalla algunos componentes que están dentro de las diferentes capas y que permiten la gestión de las diferentes aplicaciones desplegadas dentro de la nube.

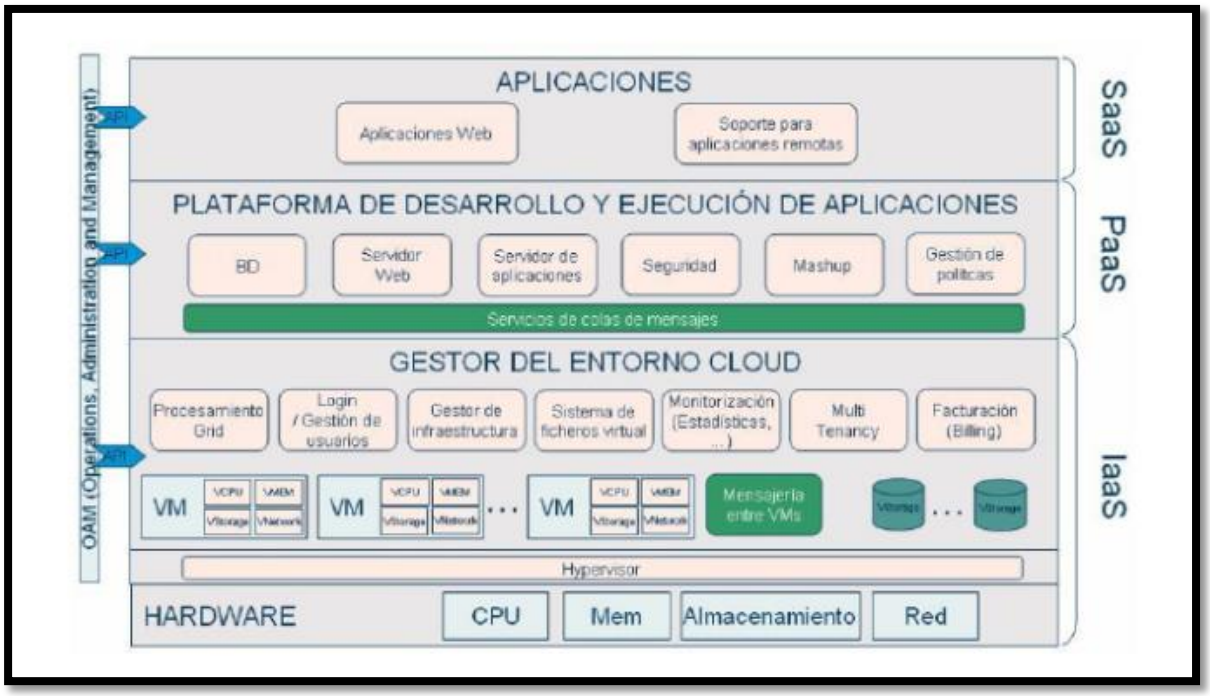


Imagen 1. 19: Arquitectura genérica de cloud computing [27]

1.6. Ventajas y Desventajas de Cloud Computing

El uso del Cloud Computing brinda unas serie de ventajas y desventajas, las cuales nos darán la mejor noción de si elegir una solución de Cloud, Así mismo, el tipo de nube implica un importante parámetro al momento de conocer cuál es la más aplicable dentro de la organización, que recurrirá a esta implementación para satisfacer necesidades propias de la actividad a la que se dedican.

Las principales ventajas son un extracto de los beneficios en general que se lograrían alcanzar aplicando los diferentes niveles de servicios y los tipos de nubes específicos.

[13] Asterisk: Es un Servidor de Voz sobre IP que permite la creación, administración y control del sistema de voz.

Ventajas [17] [9]**:**

- **Reducción de costos:** Los costos se reducen, debido a la virtualización de infraestructura.
- **Escalabilidad:** No existe limite en el crecimiento de la infraestructura, debido a que todos los recursos se virtualizarán.
- **Mejora la gestión de las TI:** Al poseer una infraestructura centralizada, permitirá tener acceso más sencillo a su administración.
- **Permite lanzar nuevas aplicaciones:** Al tener la infraestructura lista, se reduce el tiempo de lanzamiento y pruebas.
- **Control directo de infraestructura:** El administrador se encargará de tener un control de todos los elementos desplegados y los servicios utilizados.
- **Asegurar la calidad del servicio:** Al tener control total de la infraestructura podemos asegurar el mayor nivel posible de QoS.
- **Eliminar defectos por mala configuración:** Al ir desarrollando y configurando módulos, se pueden ir corrigiendo errores inmediatamente.
- **Movilidad:** Acceder desde cualquier parte respecta un gran beneficio en cuanto a movilidad y configuraciones.
- **Flexibilidad**: La implementación permite perfectamente ir creciendo de acuerdo a las necesidades de la organización [8].

Desventajas [17] [9]**:**

- **Seguridad:** Se debe aplicar controles más estrictos de seguridad para prevenir ataques.
- **Privacidad**: Dependiendo del modelo de despliegue la privacidad de los datos pueden estar comprometida.
- **Conectividad:** Se necesitan conexiones de alta velocidad para brindar algunos de los servicios que el cliente necesita.
- **Ciertos dispositivos puede o no acceder a los servicios:** Algunos de los servicios que se ofrecen no pueden ser accedidos por todos los dispositivos en el mercado.
- **Dependencia a la conectividad:** Para poder acceder a los servicios de la nube, debemos estar conectados a ella para poder usarlos.

- **Se necesitan conexiones de alta velocidad:** Para ofrecer ciertos servicios se necesitan tener conexiones de alta velocidad, pero actualmente esto no podría no ser un limitante dadas las altas velocidades que los ISP [14]ofrecen [8].

.

1.6.1. Disponibilidad

En el caso de Cloud Computing, la disponibilidad, es la representación del servicio que garantiza el cumplimiento del mismo, para esto es necesario un plan en donde se detalle el nivel de servicio de las aplicaciones para cumplir con los acuerdos establecidos.

Para alcanzar una disponibilidad de alto nivel es necesario un gran ancho de banda, en el caso del Cloud Computing, los servicios son monitoreados lo que da una ventaja superior a las tecnologías anteriores. La disponibilidad en Cloud Computing analiza algunos parámetros como: el tiempo de respuesta, analizado en cada una de las aplicaciones desplegadas en la nube, el rendimiento, que permite analizar los recursos están bien distribuidos y la fiabilidad, que demuestra que tan confiables son los datos que están siendo manejados [8].

La disponibilidad en Cloud Computing puede analizarse en algunos elementos como:

- **Servidores:** Estos elementos deben ser seguros y cumplir con los estándares que garantizan, que se cumplirá con los niveles de disponibilidad requeridos por las organizaciones.
- **Redes:** Los enlaces y las configuraciones de red también deben estar debidamente estandarizados, para garantizar que los servicios se mantendrán estables durante el funcionamiento de las aplicaciones.

- **Respaldos:** De la misma manera que los elementos anteriores se debe contar con estándares de respaldos y acciones previas y posteriores en casos de fallos.

Imagen 1. 20: Disponibilidad de cloud computing

[14] ISP: Internet Service Provider o Proveedor de Servicios de Internet provee de conexión a inernet a las organizaciones.

1.6.2. Integridad

La integridad, garantiza que los datos no sean alterados y se mantengan idénticos al realizar consultas sobre los mismos, además habla mucho acerca de la reputación de la empresa, debido a que si se mantienen los principios, que guardan información confidencial, esto hace mejorar como los clientes consideran a la organización que presta el servicio.

Para lograr que los datos no se vean afectados bajo ninguna circunstancia, se deben tener dentro de la organización unas políticas claras, en donde se detalle que los accesos y modificaciones vayan de acuerdo a lo estipulado para el funcionamiento de la organización pero sin presentar problemas que les impliquen pérdidas o caídas en el servicio. [8]

Imagen 1. 21: Integridad en cloud computing

1.6.3. Confidencialidad

La confidencialidad debe garantizar que personas no autorizadas accedan a los datos para esto existen una serie de métodos que se deben analizar y aplicar, para evitar violaciones de identidad, y accesos no deseados.

Al hablar de confidencialidad se debe tomar en cuenta la seguridad de la información que se piensa ofrecer dentro de la nube, esta información debe ser considerada como un activo para la empresa, constituyendo unos de los más importantes que poseen.

Existen algunos métodos o procedimientos, entre estos se pueden usar para asegurar la confidencialidad de la información y algunos son:

- Políticas de usuarios, aquí se administran permisos así como capacitación para que los usuarios se sientan informados ante cualquier atentado que podría presentarse.

- Se pueden implementar servidores de autenticación, para garantizar que solamente los usuarios permitidos puedan acceder a información y manejarla de manera conveniente.
- Debe existir un alto nivel de autenticación entre los componentes de las aplicaciones para que solamente quien tiene acceso autorizado a estos datos puedan obtenerlos [8].

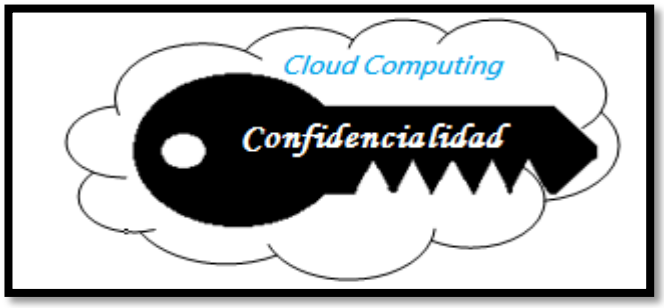


Imagen 1. 22: Confidencialidad en cloud computing

1.7. VoIP Conceptos Básicos

VoIP (Voice over IP, Voz sobre IP) Es un servicio que se inició en 1995, como una implementación casera de un grupo de jóvenes, y más adelante se contó con el apoyo de Vocaltec[15], se crea el primer equipo que no cumplió con los requerimientos necesarios para cumplir con calidad de transferencia de voz. De esta manera en 1997 se da el primer congreso VON[16] en donde se comienzan a establecer normas que provocaron el desarrollo rápido de esta tecnología, para que en 1998 se comiencen a crear los primeros gateway que interconectaban redes y a partir de esto se desarrollaron las tecnologías actuales [18] [19].

VoIP es una tecnología de señalización y procesamiento de llamadas en tiempo real, [20] mediante la red IP, para la comunicación de voz, esta tecnología está dirigida por algunas organizaciones de estandarización como: IETF [17] y la UIT[18] con el fin de lograr la convergencia de tecnologías para las comunicaciones IP [21].

[15] Vocaltec: Empresa líder en tecnología de voz sobre IP.
[16] VON: Congreso y Feria para la industria de voz sobre IP. http://www.voip-info.org/wiki/view/VON
[17] IETF: Internet Engineering Task Force o Grupo de Trabajo de Ingeniería de Internet, es una organización de normalización de diversas áreas tecnológicas.
[18] UIT: Unión Internacional de Telecomunicaciones, es el organismo especializado de las Naciones Unidas para las tecnologías de la información y la comunicación.

VoIP cuenta con elementos básicos como: los servidores encargados de la comunicación, los clientes que hacen uso de los servicios y los equipos que sirven como emisores y receptores, es decir, el hardware que interviene en la comunicación [20], que lo que hace es encargarse de convertir la señal analógica en digital, mediante el muestreo, cuantificación y codificación de señales [22].

En la ilustración 1.23 se detallan las capas con los respectivos protocolos que intervienen en la comunicación de telefonía IP, esta estructura está relacionada con el modelo OSI[19], excepto los protocolos que intervienen en las capas de aplicación y transporte.

En la capa de aplicación están los protocolos que son específicos para voz como: SIP, H.323 e IAX, se mencionarán algunas características específicas de cada uno. Y en la capa de transporte se emplea el protocolo RTP, que es un protocolo de Tiempo Real para video conferencias, es decir, tráfico de voz y video dentro de la VoIP.

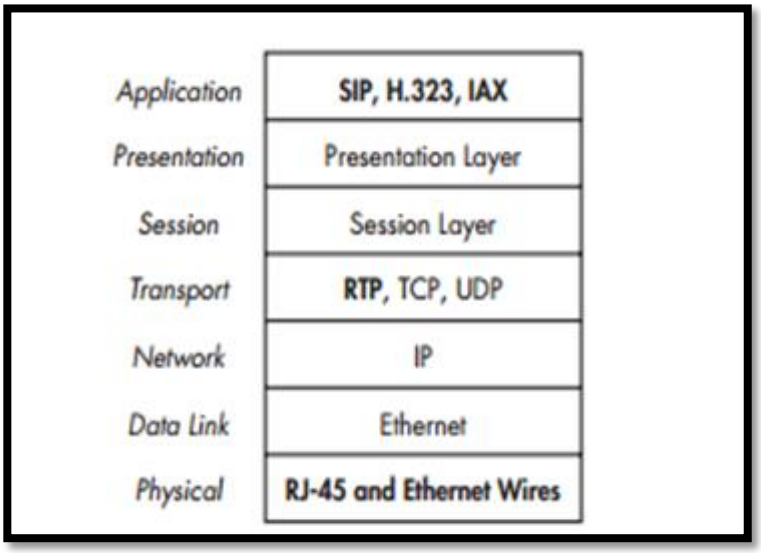


Imagen 1. 23: Modelo de referencia VoIP [34]

1.8. Protocolos empleados en VoIP

Los principales protocolos que usa VoIP son H.323[20] y SIP[21] en la capa de aplicación, y son usados para establecer la llamada, RTP[22] en cambio se encarga de manejar el

[19] OSI: Open System Interconnection o Modelo de Interconexión de Sistemas Abiertos, es un estándar de referencia de interconexión de red.
[20] H.323: Protocolo empleado en VoIP para la comunicación
[21] SIP: Session Initiation Protocol o Protocolo de Inicio de Sesión para voz sobre IP
[22] RTP: Real Time Protocol o Protocolo en Tiempo Real

transporte de la llamada, IAX[23] por su parte es el único protocolo que maneja tanto el transporte y la sesión de la llamada.

A continuación se detallarán algunos protocolos empleados por VoIP para la transferencia de paquetes, así como sus características específicas de acuerdo al papel que desempeñan dentro de la red de voz, que es muy similar a la red TCP/IP[24] [21].

1.8.1. Protocolos de Señalización

Los protocolos de señalización en una llamada se encargan establecimiento, cancelación y manejo de las llamadas, de acuerdo al estándar usando, a continuación se explicarán los protocolos más usados en VoIP [23].

- **H.323:**

Es un protocolo de señalización creado por la ITU, que permite que los paquetes de voz y señalización se comuniquen por vías independientes; los paquetes de señalización se comunican con los usuarios de las llamadas y los equipos que se encargan de la comunicación de las llamadas (teléfonos, celulares), y los paquetes de voz se comunican directamente entre los usuarios, en donde, ambos tipos de paquetes pueden tener caminos distintos" [21]. El tráfico se realiza mediante UDP/IP[25].

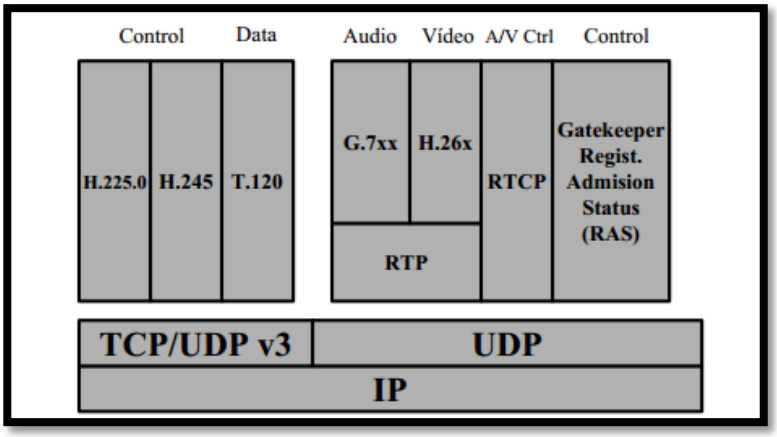


Imagen 1. 24: Protocolo H3.23 [33]

H.323 se encarga de algunas funciones principales como [24]*:*

- Permite la interoperabilidad entre distintos fabricantes
- Permite la comunicación entre distintas plataformas y aplicaciones

[23] IAX :Inter-Asterisk eXchange protocol es uno de los protocolos utilizado por Asterisk
[24] TCP/IP: Es un modelo de Referencia de Protocolos de Red para la comunicación.
[25] UDP/IP User Datagram Protocol, es un protocolo de transporte no orientado a la conexión sobre la red IP

- Permite monitorear el ancho de banda
- Permite alcanzar un nivel alto de seguridad en la comunicación

El protocolo H.323 está compuesto por los siguientes elementos: [23]

- Terminal: Solo los teléfonos o terminales de usuarios.
- Gateway: Son dispositivos que enrutan las peticiones hacia las diferentes redes como PSTN.
- Gatekeeper: Es un software opcional que sirve para proveer servicios como plan de marcado, traducción de direcciones, etc.
- Unidad de Control Multipunto: Permite establecer las multiconferencias entre gateways.

- **SIP**

Es un protocolo inicio de sesión de la IETF, está en la capa de aplicación, y sirve establecer las sesiones como pueden ser: llamadas, videos en tiempo real, etc.

SIP es un protocolo mucho más sencillo que H.323, debido a que H.323 cuenta con protocolos adicionales para control de las llamadas lo que hace de la comunicación más segura lo que hace más complicada su implementación [24].

SIP emplea servidores proxy para enrutar peticiones, autenticar y autorizar servicios, y puede viajar por cualquier protocolo de transporte, cuenta con algunas funciones principales como [22]:

- Identificar el host para la conexión.
- Analizar si está disponible el que recibe la llamada.
- Establece el intercambio de datos durante la llamada.
- Establece la sesión entre el que llama y el que recibe la llamada.

Una red SIP se compone básicamente de los siguientes elementos principales [23]:

- Agente usuario: Es el usuario que hace o recibe las llamadas.
- Servidor: Encargado de establecer y mantener las llamadas.

- **IAX**

Es un protocolo usado en servidores Asterisk para la transmisión de voz sobre IP, con el fin de realizar y manejar conexiones entre clientes, se usa IAX2 debido a su rendimiento

y facilidad en relación a otros protocolos. Utiliza UDP para la comunicación entre terminales. IAX2[26], debido a esto no es detectable por los firewall y permite trabajar dentro de redes internas [22].

El protocolo IAX2 creado por Mark Spencer, permite la señalización de VoIP debido a que crea sesiones internas que pueden utilizar cualquier códec por lo que es flexible y puede ser utilizado con cualquier tipo de datos como voz, vídeo y ambos a la vez.

IAX ofrece soporte para trunking y multiplexación de los canales de transmisión, por ello las metas por las cuales IAX fue diseñado son [25]:

- Reducción del ancho de banda
- Soporte para NAT[27]
- Facilidad de uso sin tener problemas con un firewall.

Una red IAX cuenta con los siguientes elementos que permiten la comunicación [25]:

- **Servidor IAX:** Es el encargado de permitir la comunicación entre los terminales IAX, aquí se configuran los usuarios, los servicios y la seguridad de la red.
- **Terminales IAX:** Son los dispositivos que permiten la comunicación entre usuarios, por ejemplo: teléfonos IP, smartphones y software phones.
- **Equipos intermedios:** Como routers o switchs, que permiten la interconexión entre dispositivos.

Comparación entre AIX y SIP

En el caso de la implementación que vamos a realizar, se empleará el protocolo SIP, debido a que Asterisk, trabaja directamente con este protocolo y este es mucho más sencillo y no es de mayor complejidad para administrar y gestionar los servicios de voz actuales.

[26] IAX2: Segunda versión del Protocolo IAX con funcionalidades adicionales.
[27] NAT: Network Access Translation o Traducción de nombres de Red, su función es permitir el intercambio de paquetes entre dos redes.

Característica	AIX	SIP
Ancho de Banda	Menos ancho de banda por codificación binaria. Reducción de cabeceras.	Usa mayor ancho de banda y cabeceras más grandes.
NAT	La señalización y los datos viajan juntos.	Señalización y datos viajan separados. Requiere servidor adicional
Estandarización	No se encuentra estandarizado.	Estandarizado por IETF
Puertos	Utiliza el puerto 4569, para señalización y datos	SIP usa el puerto 5060, para señalización y 2 puertos RTP por cada conexión de audio.
Flujo de Audio	Los datos pasan directamente por el servidor, lo que genera más ancho de banda.	Los datos no pasan directamente por el servidor lo que genera menos tráfico en el servidor.

Tabla 1.1: Comparación entre protocolos SIP y AIX [36] [37]

1.8.2. Protocolos de Encaminamiento

VoIP emplea protocolos ya sea dentro de una empresa o fuera de ella, debido a esto se usan algunos protocolos tanto estáticos como dinámicos para enrutar las llamadas. Los protocolos de enrutamiento dinámico son mucho más empleados, debido a que permiten la actualización rápida a cambios que se puedan dar en la infraestructura de la red [21].

Los protocolos de enrutamiento dinámico se caracterizan porque contienen tablas con las rutas hacia los dispositivos con los que están interconectados, y cuanto se presenta algún cambio en la red, los protocolos actualizan sus tablas con los datos nuevos lo que facilita hacerlo manualmente.

Los protocolos de enrutamiento pueden ser clasificados como:

- **Protocolo de Gateway Interior (IGP)** [21]:

Estos protocolos son empleados para enrutamiento interno, es decir, solo para comunicaciones dentro de una organización. Los protocolos que pertenecen a esta clasificación son:

- **RIP (Routing Information Protocol o Protocolo d Información de Enrutamiento)*:*** Es un protocolo de enrutamiento estático, que emplea vector distancia para calcular el camino más corto, esto se lo hace por medio de la cuenta del número de saltos en donde cuyo mayor valor es de 15, y conocido como la métrica.

- **EIGRP (Enhanced Interior Gateway Routing Protocolo o Protocolo de Enrutamiento de Gateway Interior Mejorado):** Este protocolo es propietario de Cisco, está basado en el protocolo de estado de enlace, permite la detección de vecinos y es muy usado por las organizaciones por su fácil configuración.
- **OSPF (Open Shortest Path First o Primero el Camino más Corto):** Es un protocolo de enrutamiento dinámico, el COSTO como métrica, su funcionamiento se basa en el ancho de banda y la congestión del enlace, usa MD5[28], que es una herramienta de seguridad, y emplea un protocolo llamado "Hello" para determinar los vecinos que se encuentran conectados, y actualizar la tabla de enrutamiento.

- **Protocolo de Gateway Exterior (EGP)** [21]:

  - **BGP (Border Gateway Protocol o Protocolo de Gateway de Borde):** Es un protocolo empleado por internet en los ISP's, para compartir las tablas de enrutamiento, entre ISP's y así permitir la interconexión entre los diferentes ISP.

1.8.3. Protocolos de Transporte de Voz

Son los encargados de realizar la transmisión de los paquetes de voz digitalizados, de un lugar a otro, a través de los flujos de voz, y luego puedan ser reconstruidos en el orden correcto.

Para este propósito se usa **RTP (Real Time Protocol),** el cual está diseñado para transporte de paquetes en tiempo real, como es el caso de la voz. RTP encapsula UDP para la trasmisión, ya que ofrece una entrega rápida, con esto RTP agrega fiabilidad al ofrecer una secuencia numérica y reordenamiento de paquetes [21]:

1.9. Tipos de Diseños a Aplicar en VoIP

Diseñar una red para la utilización de VoIP es un procedimiento complejo en donde se deben cumplir ciertos criterios para garantizar el cumplimiento de normas y estándares para el nivel de tráfico al brindar este servicio. Se debe analizar el tráfico de la red, los

[28] MD5: Message-Digest Algorithm 5 o Algoritmo de Resumen del Mensaje 5, es un algoritmo de encriptación de 128 bits.

usuarios que tendrán acceso, los equipos a emplear, así como el tipo de red que va a ser aplicada esta puede ser: LAN, MAN, WAN [21] [23].

Los diseños de VoIP, presentan ciertas características que darán una calidad de servicio aceptable al ser analizadas correctamente [21]:

- **Retraso:** Es el tiempo que necesita un paquete para llegar de una dispositivo transmisor hasta un dispositivo receptor.
- **Jitter:** Es un factor que se puede analizar en transmisiones multimedia como audio, video y voz, debido al retraso de paquetes.
- **Paquetes perdidos y erróneos:** Este problema se presenta cuando un paquete no llega al destino, o los que llegan a su destino sufren algún cambio en el trayecto.
- **QoS[29]:** O calidad de servicio se encarga de brindar el mejor servicio posible, va de acuerdo a la tasa de bits y al BER[30] mínimo.
- **Tasa de bits:** Es la cantidad de bits que se transmiten por el canal de voz.

1.9.1. Diseño de una LAN

Diseñar una LAN es una parte muy importante en el diseño de una red para VoIP, ya que los usuarios finales son los que están conectados a ella, y permite la conexión hacia la red WAN. El esquema se detallará con 3 niveles [21]:

- **Nivel de Acceso:** Permite la conexión de los dispositivos de usuario final, o de los servidores.
- **Nivel de Distribución:** Constituye las políticas para entrega de tráfico, seguridad, QoS, balanceo de carga.
- **Nivel de Núcleo:** Ofrece un backbone de conexión a internet debe ser seguro ya que es el área crítica de la red, y adaptarse a cambios.

[29] QoS: Quality of Service o Calidad de Servicio, es un factor que garantiza el nivel de servicio de una red de telefonía o de computadores.
[30] BER: Bit Error Ratio o Tasa de Error Binario se refiere a la cantidad de bits recibidos erróneamente.

1.9.2. Diseño de una WAN

Para el diseño de una red WAN[31] se debe tener en cuenta principalmente el medio a través del cual se quiere transmitir, así como el lugar por la cual el enlace se realizará, existen varios tipos de tecnología WAN, a continuación se muestran algunas de las tecnologías que existen actualmente.

Tecnologías Aplicables a una Red WAN

Tipo	Tecnología WAN
Red de Acceso	ADSL[32]
	Cable módem
	Wifi
Red Troncal	Metro Ethernet
	SDH[33]
	Fibra Oscura
	DWDM[34]

Tabla 1.2: Tecnologías de acceso y troncal en una red WAN [32]

En el caso de diseñar una red WAN se deben analizar parámetros importantes como la elección del medio troncal, que es el que garantizará la comunicación entre sucursales, depende las distancias a cubrir y el tipo de servicio a prestar.

1.9.3. Diseño de una Red Privada

Una red privada garantiza que los datos y las aplicaciones y servicios con administradas por la misma organización, y por lo tanto se tiene un control absoluto de toda la información. Para la conexión fuera de la red privada se emplean las VPN o Virtual Private Network, que nos permiten la conexión remota entre dos redes a través de internet.

[31] WAN: Wide Area Network o Red de Area Extensa abarca redes amplias e gran distancia como por ejemplo un país.
[32] ADSL: Asymmetric Digital Subscriber Line o Línea de Abonado Digital Asimétrica tecnología de transmisión analógica de datos digitales.
[33] SDH: Synchronous Digital Hierarchy o Jerarquía Digital Síncrona o Tecnología de transmisión de alta capacidad.
[34] DWDM: Dense Wavelength Division Multiplexing multiplexado compacto por división en longitudes de onda es una técnica de tranmision de alta velocidad mediante fibra óptica.

Las VPN permiten la conexión de usuarios de forma remota, así como de varias redes mediante una IP pública, y mediante esta el usuario puede tener acceso a los servicios que la red ofrece como transmisiones de archivos, o uso de aplicaciones, etc [21].

1.9.4. Diseño de VoIP en la Nube

En este diseño la aplicación requiere de una estructura basada en un modelo compuesto que incluye los diseños mencionados anteriormente para alcanzar la convergencia de redes.

La red completa sobre VoIP consta de una serie de niveles los cuales se encargan de realizar funciones específicas como:

- **Núcleo:** Es el nivel encargado de brindar la conectividad backbone a la red de distribución.
- **Distribución:** Se encargan de conectar a los usuarios con las redes DMZ[35], las VPN, la WAN corporativa y los Gateway de Voz.
- **Borde Corporativo:** Se conecta de manera directa con la red de acceso.
- **Acceso:** Aquí se dan conexión a internet, a una WAN y a la RTC[36].
- **Equipos Terminales:** Permiten la conexión a los servicios y puedes ser: computadores, teléfonos inteligentes, tablets, centro de datos o sucursales corporativas las cuales se conectan mediante enlaces WAN o VPN. [21]

[35] DMZ: Demilitarized zone o zona desmilitarizada, es una red local intermedia que está entre la red interna y la red externa.
[36] RTC: Red Telefónica Conmutada son todos los elementos que enlazan una red mediante un circuito físico.

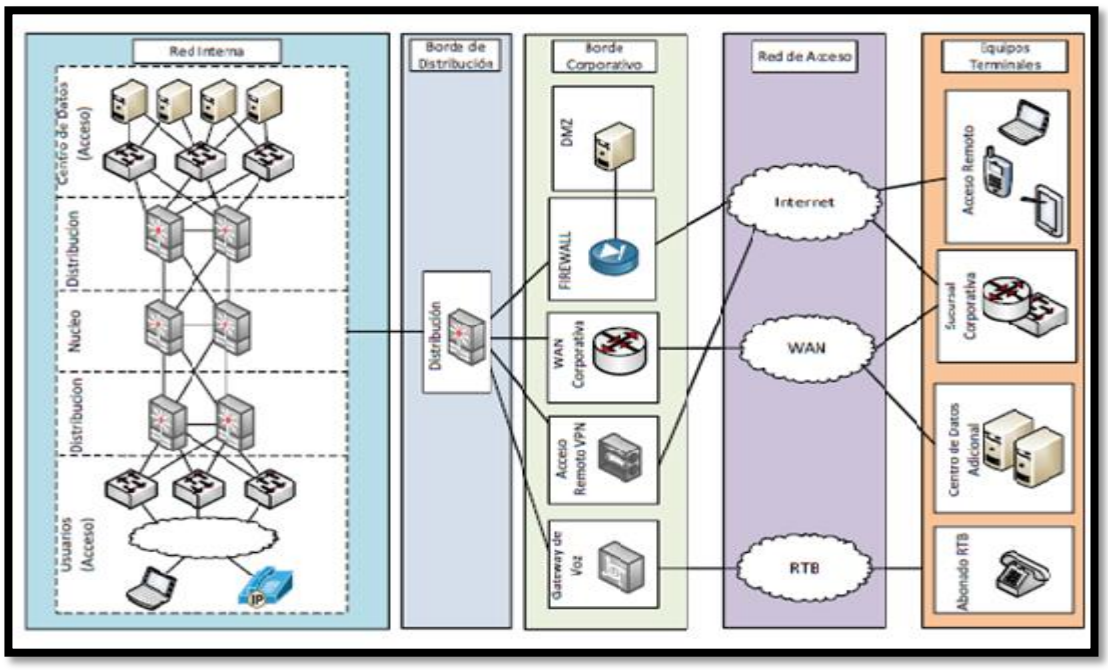


Imagen 1. 25: Diseño básico de una nube par VoIP [32]

1.9.4.1. VoIP sobre Nube Pública

En este diseño completo sobre VoIP pública los usuarios pueden conectarse a la DMZ por una VPN, dependiendo del servicio.

Así también se puede implementar Bases de Datos de facturación o clientes, cuando se implemente un firewall antes de las mismas, con el fin de mantener la confidencialidad en caso de accesos de los usuarios a la red.

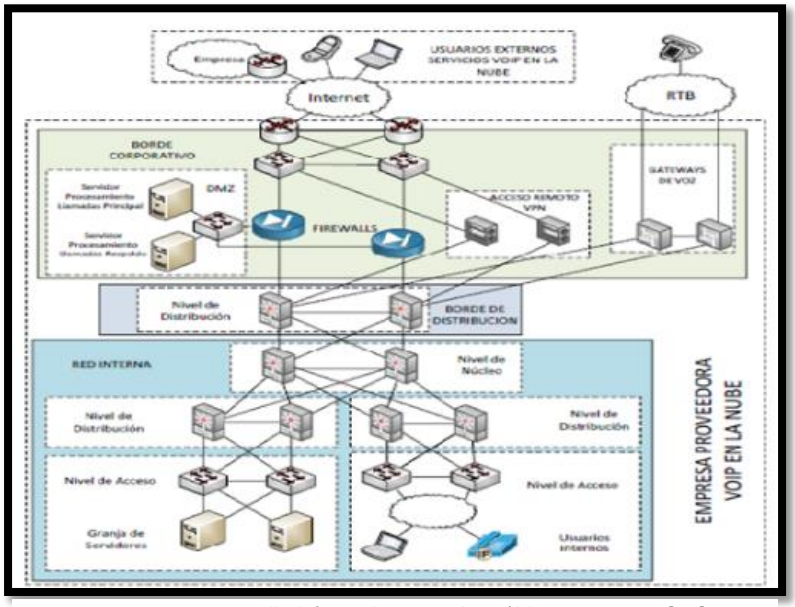


Imagen 1. 26: Diseño básico de una nube pública para VoIP [32]

1.9.4.2. VoIP sobre Nube Privada

Es implementada por empresas para brindar servicios internos a usuarios, en este diseño las redes de acceso, permiten que los usuarios corporativos se conecten a internet o enlaces dedicados, que no requieren gran seguridad como las redes públicas o hibridas.

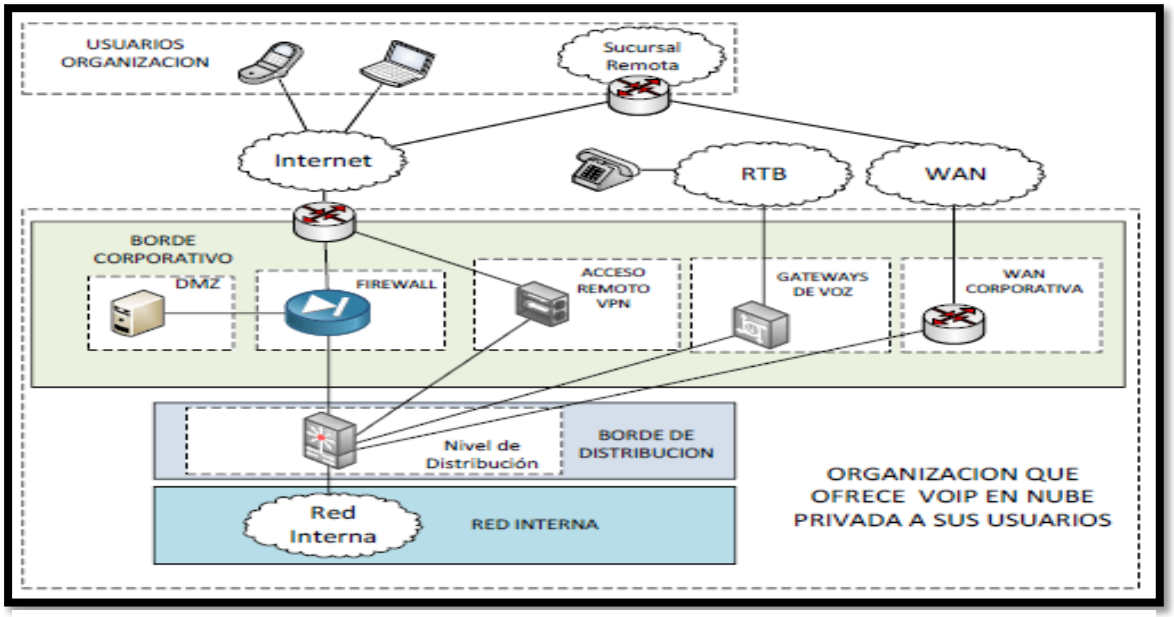


Imagen 1.2: Diseño básico para una nube par VoIP privada [32]

1.9.4.3. VoIP sobre Nube Híbrida

Es una combinación entre nube privada y pública, en donde una empresa consta de una WAN para interconectar sucursales y que por requerimientos necesita conectarse al proveedor de servicios en la nube.

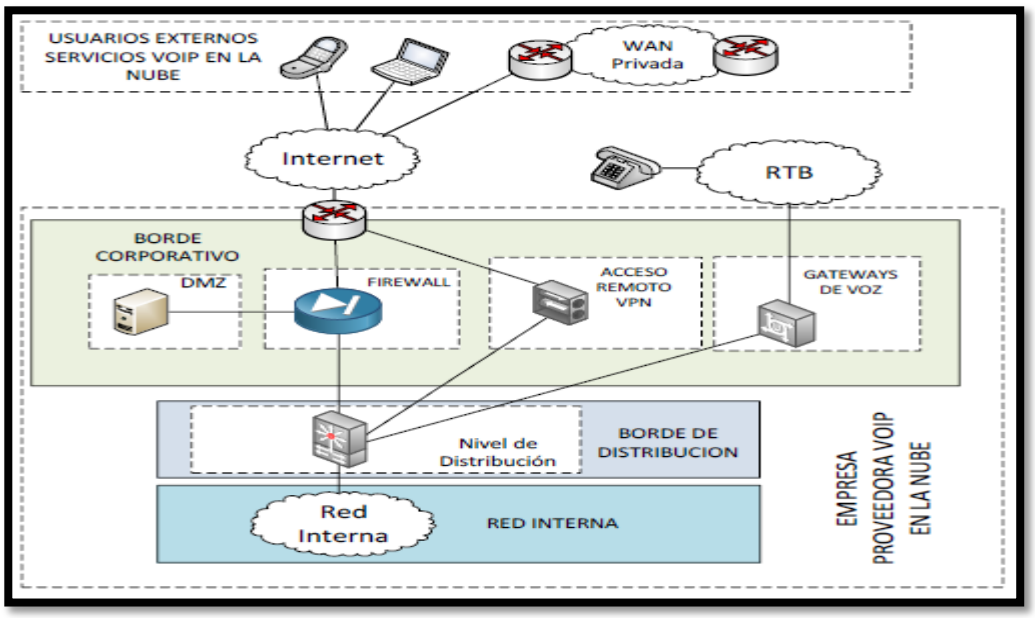


Imagen 1.48: Diseño básico de una nube para VoIP [32]

En las implementaciones de VoIP existen aplicaciones de tipo SaaS en donde los proveedores de servicio instalan y operan los servicios, pero los clientes no tienen gestión sobre los mismos.

Mientras que también por otra parte están las aplicaciones IaaS en donde los clientes gestionan los equipos a fin de personalizar aplicaciones pero con un menor costo de gestión. [21]

CAPÍTULO 2

2. Requerimientos de Diseño

2.1. Virtualización Xen-Server

El término virtualización hace referencia a un despliegue que permite la simulación de recursos físicos en varios recursos virtuales, que están alojados en un sistema anfitrión y se ejecutan a través de un hipervisor [6].

La virtualización lo que hace es simular la ejecución de varias máquinas virtuales con diferentes sistemas operativos de forma independiente pero en la misma máquina. Cada máquina virtual posee su propio hardware virtual como la RAM, CPU, NIC[37], Disco. Al crear una máquina virtual se crean discos de respaldo que son diseñados para ser utilizados rápidamente en el caso que se presenten fallos, brindando escalabilidad, rapidez, seguridad y administración [11].

Algunos ejemplos de software de virtualización: VMwarevSphere, Microsoft Hyper-V y Citrix XenServer.

Imagen 2. 1: Ejemplos de Virtualizadores [10]

La virtualización, es empleada en varios campos pero sobre todo en las TI[38], con el fin de aprovechar los recursos de forma eficiente, además brindan las siguientes ventajas como [11]:

- La consolidación del hardware facilita la capacidad de que un servidor físico pueda abarcar más servidores dentro de sí mismo en donde se encuentran las diferentes aplicaciones.
- La virtualización permite agilidad y flexibilidad para lograr la implementación y mantenimiento de sistemas y aplicaciones tanto en servidores físicos como virtuales.

[37] NIC: Network Interface Card o Tarjeta de Interfaz de Red es un dspositivo que permite la comunicación entre máquinas interconectadas.
[38] TI: Tecnología Informática, es un término empleado para agrupar a las tecnologías de comunicaciones.

- La Optimización de Recursos, garantiza el mejor uso de los servidores físicos, con la ayuda de la virtualización las organizaciones aprovechan de mejor manera los equipos.
- Los gastos de funcionamiento, se simplifican gracias a la gestión de sistemas mediante virtualización y se reducen también gastos de energía y equipamiento.

2.1.1. Características

Citrix XenServer utiliza el hipervisor de Project Xen, es de código abierto y está diseñado para administrar de forma eficiente las máquinas virtuales (VM) de Windows y Linux con el fin de brindar una opción rentable de consolidación de servidores y continuidad del negocio.

XenServer utiliza el hipervisor Xen Project como un componente principal de su arquitectura para proporcionar una abstracción de la infraestructura [26].

XenServer posee varias características ya que proporciona una ruta sin inconvenientes para que los clientes migren sus cargas de trabajo más importantes a un entorno de nube, a continuación se listarán las características más importantes:

- Permite la administración de rendimiento, como CPU, memoria, disco, red. Además administra alertas por medio de parámetros antes mencionados de acuerdo a la configuración.
- Escalabilidad en componentes virtuales, que pueden crecer según las necesidades de los servicios desplegados en la nube.
- Mantenimiento sencillo, pues al tener un solo servidor encargado de gestionar las máquinas no hay mayor inconveniente.
- Disponibilidad de recursos, el administrador puede disponer de los datos y realizar las configuraciones necesarias de acuerdo a los requerimientos *[27]*.

Algunos ejemplos de complementos que se pueden administrar junto con el servidor Xen son:

- **XenMotion**

Permite el mover una máquina virtual de un lugar a otro, esto por mantenimiento o actualización del host, con esto se eliminan los tiempos de inactivada de una máquina virtual [28].

- **Xen Storage Motion**

Permite mover una máquina virtual o un VDI[39], de un entorno de desarrollo a uno de producción, permite migrar una máquina virtual cuyos discos están en el almacenamiento local, o incluso para migrar discos de una VM[40], de un repositorio de almacenamiento a otro, a la vez que la máquina virtual está en ejecución.

2.1.2. Requerimientos

Antes de definir los requerimientos es importante diferenciar dos tipos de aplicaciones, la una sé que se instalará directamente en el hardware donde estará XenServer y la otra llamada XenCenter, la cual es una administración remota y debe instalarse en una maquina con Windows. A continuación se especificarán ambos tipos de aplicaciones.

- **XenServer**

Es una plataforma de virtualización de código abierto muy utilizada para administrar las infraestructuras virtuales de la nube, como servidores y máquinas de escritorio. XenServer es gratuito y permite la virtualización de las cargas de trabajo grandes y automatiza la administración, para hacer más rápido y flexible en funcionamiento de las tecnologías de la Información, tiene acceso al hardware, maneja los dominios invitados, el administrador se encarga de la gestión de los recursos y también se encarga de la sincronización del uso del CPU por parte de los invitados [29].

Imagen 2. 2: Virtualizador Xen [41]

Los componentes básicos de una infraestructura XenServer son [30]:

- Los servidores físicos XenServer (hypervisores[41]) que proporcionan los recursos. de procesador y disco a las máquinas virtuales.

[39] VDI: Virtual Drive Interface o Interfaz de Unidad Virtual permite gestionar a través de un escritorio las máquinasvirtuales creadas por el cliente.
[40] VM: Virtual Machine son las abreviaturas de Maquina Virtual en español.
[41] Hipervisor: es una plataforma para desplegar los sistemas operativos.

- Una estación de trabajo que funciona como la consola XenCenter desde la que se administra el entorno.
- Espacio en disco (local o remoto) donde ubicar las máquinas virtuales (storage repositories).

Entorno XenServer = Servidores XenServer + Consola XenCenter + Espacio en disco

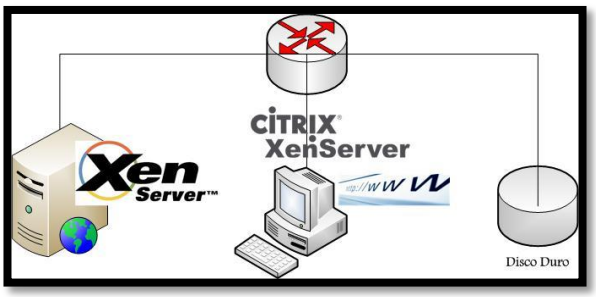


Imagen 2. 3: Entorno Xen Server [41]

XenServer debe administrar la carga de recursos necesarios para cada uno de los invitados, por lo tanto cumplir con al menos las siguientes características, las cuales fueron tomadas de las página de Xen [31].

Características de Xen:

- 64-bit x86 Tipo de Arquitectura del Sistema
- CPU: 1.5 GHz mínimo - 2 GHz o más rápido y multi-core recomendado
- Intel VT o AMD-V necesarios para el apoyo de los invitados Windows
- 2 GigaByte a 1 TeraByte de Memoria Física
- Hasta 64 Procesadores Lógicos
- Hasta 16 tarjetas de red
- Alta disponibilidad

- **XenCenter**

XenCenter es la interfaz de administración de XenServer la cual corre en sistemas operativos Windows, permite a los usuarios la configuración de nuevas máquinas virtuales, configuración, administración, etc. Permite también la configuración del almacenamiento remoto y manejo de la red, VLAN, y redes internas.

Los requerimientos que el equipo donde será instalado debe tener fueron tomados del libro Implementation Citrix Xen Server [31]:

- XenCenterClient
- x86: Arquitectura mínima
- Soporte para Microsoft Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008, Windows Vista, or Windows 7
- Velocidad de CPU: 750 MHz mínina
- RAM[42]: 512 MB mínimo
- Espacio en Disco: 100 MB mínimo

2.1.3. Ventajas y Desventajas

El uso de XenServer nos brinda varias ventajas y desventajas nombradas a continuación:

Ventajas [28]**:**

- Puede ser instalado tanto en máquinas de escritorio como en Laptops, siempre y cuando cumplan con los requerimientos básicos.
- No hay límite en la creación de máquinas virtuales alojadas en el host.
- Permite la creación de hasta 7 NICs por cada invitado.
- Permite la creación de sistemas invitados tanto de 32 como de 64 bits.
- Ofrece seguridad en la comunicación de XenServer y XenCenter, a través de encriptación 256-bits AES SSL [43]

Desventajas [32]

- Si el host se daña se pueden perder todos los datos y máquinas virtuales instaladas.
- Es costosa la adquisición del hardware.
- Su principal desventaja es que trabaja con sistemas operativos que han sido modificados para su uso sobre XenServer.
- Su instalación toma un poco más de trabajo que otros virtualizadores.

[42] RAM: Random Access Memory o Memoria de Acceso Aleatorio, es un dispositivo para almacenamiento en donde se cargan las instrucciones para el procesador.
[43] 256-bits AES SSL: 256 bits Advanced Encryption Standard o Estándar de Encriptación Avanzada y SSL Socket Secure Layer o Capa de Conexion Segura, técnica que sirve para el aseguramiento de la red.

- La documentación que ofrece su página oficial no es completa.

Comparación entre Virtualizadores

Características	**XEN**	**XCP**	**VirtualBox**
Compañía	xen.org	xen+complementos	Oracle
Plataforma	Libre	Libre	Libre
Conocimientos de administración	Alto	Alto	Bajo
Integración de video	Bajo	Bajo	Alto
Para-Virtualización	Si	Si	No
Aplicaciones	Máquinas virtuales para servidores de pruebas	Máquinas para servidores	Virtualización de máquinas Windows

Tabla 2. 1: Comparativa de virtualización aplicable [45]

Según la tabla 3 se puede observar que Xen es una buena opción para implementar en sistemas cloud, pues es libre, permite virtualización por hardware lo que facilita las pruebas antes de ponerlo en producción con los servidores de servicios. En el anexo se procede con la instalación de XenServer [33].

Anexo 1 Instalación de XEN

2.2. Solución de OpenStack

OpenStack es una solución opensource de software que permite el despliegue de nubes públicas y privadas, "es una colección de proyectos de software libre mantenidos por la comunidad" [34].

OpenStack fue un proyecto desarrollado por la NASA y Rackspace y en donde, Rackspace contribuyó con la parte de almacenamiento (Cinder), mientras que la NASA contribuyó con código para procesamiento Compute. Actualmente cuenta con varias empresas y organizaciones que contribuyen con el desarrollo de este software de infraestructura. Actualmente es una herramienta muy conocida para crear entornos IaaS, tanto públicos como privados. Ofrece servicios de procesamiento, almacenamiento y gestión de redes [35].

Para comprender su implementación se deberán conocer algunos conceptos que se aplicarán dentro de la nube privada y que son de conocimiento general para su despliegue, entre estos están [35]:

- **Imagen:** Son los sistemas operativos que se encuentran dentro de la nube como plantillas para su posterior creación. El servicio que se encarga de administrar las imágenes es el Glance, mientras que las máquinas se ejecutan en el nodo de cómputo y la gestión es realizada por el servicio Nova.

- **Flavor:** Es un plantilla con características específicas de acuerdo a las necesidades de despliegue, tiene como parámetros el número de CPU's, la memoria RAM, puede constar con un disco o no, todos estos elementos constan de plantillas por defecto.

- **Volumen:** Es la porción de disco en donde se desplegaran las instancias. Existen dos tipos de almacenamiento: el temporal que no guarda los estados de las máquinas, es decir, cada que se reinicie la instancia tendrá los mismos valores y el almacenamiento por volumen, en donde, se crean discos que guardan estados de las máquinas como discos extraíbles que al momento de su implementación son capturas de los sistemas operativos en funcionamiento. El módulo que se encarga de la gestión de volúmenes, es el Cinder y su configuración se realiza en un nodo adicional con gran capacidad en disco para el despliegue de las máquinas virtuales.

- **Instancias:** Son las máquinas virtuales creadas a partir de las imágenes creadas en Glance. Las máquinas que se crean son independientes de la imagen que se agregó al sistema y requieren de un flavor, que es una plantilla y recursos adicionales como los volúmenes y la conexión a la red para lograr lanzar las instancias. La gestión de las instancias es manejada por nova-scheduler y nova-api, que son desplegadas en el nodo controller.

Openstack contiene varios componentes, entre los más importantes están:

- Compute Service - Servicio de Computo (Nova).
- Storage Service - Servicio de Almacenamiento (Cinder o Swift).
- Image Service - Servicio de Imágenes (Glance).
- Identity Service - Servicio de Autenticación (Keystone).
- UI Service - Servicio de Interfaz de Usuario (Horizon).

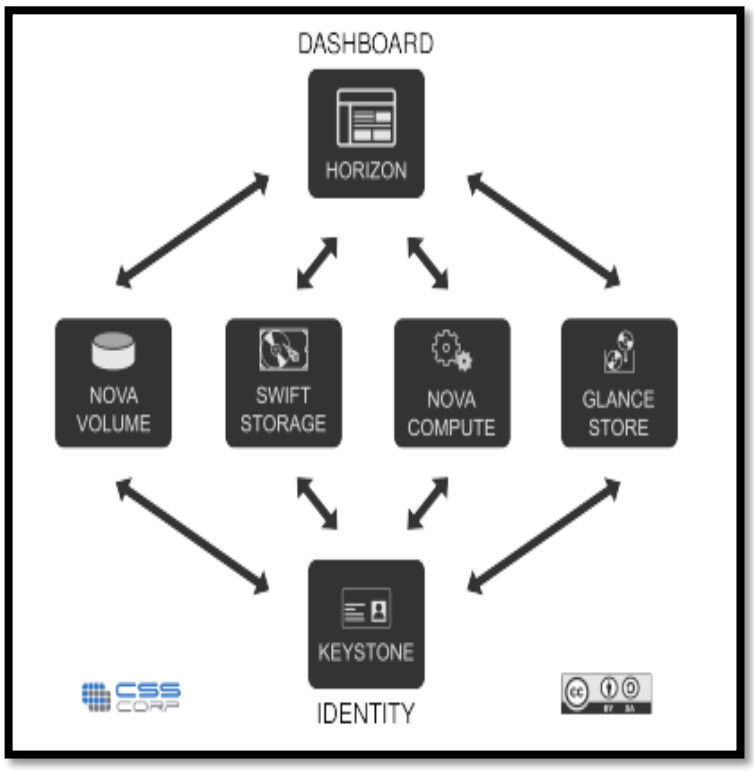


Imagen 2. 4: Comunicación entre módulos [47]

La ilustración 2.4 describe la interacción entre los módulos, en donde se encuentra Horizon como interfaz de comunicación con los módulos de Cómputo (Nova), Almacenamiento (Cinder o Swift), Imágenes (Glance), Autenticación (Keystone).

2.2.1. Características

Openstack posee algunas características, a continuación se listarán las características más importantes que se indican en su página oficial [35]:

- **Administración de los recursos del servidor:** como el CPU, memoria, disco, interfaces de red, estos recursos deben ser manejados de una forma rápida a través de la virtualización de servidores para su implementación para lograr que los recursos sean optimizados y manejados como maquinas físicas.
- **Administración de la LAN:** Permite manejar redes tanto públicas como privadas, pero al tratarse de una nube privada.
- **Administración de APIs para gestión de velocidad y servicios de autenticación:** Openstack cuenta con API´s que permiten realizar controles de telemetría, para el análisis de tráfico y carga en los servidores, también permiten

un nivel de seguridad a través de autenticación para los accesos a los servicios que se están brindando.

- **Administración de Arquitectura:** Este tipo de arquitecturas son soportadas en Openstack para aplicaciones distribuidas en donde las cargas son administradas mediante software especializado que realiza balanceo de carga y brinda escalabilidad de la nube.
- **Administración de las Imágenes:** Permite mantener un control de imágenes, las cuales representan Sistemas Operativos, en donde se desplegaran los servicios, y posteriormente la información de la organización.
- **Administración de Seguridad:** Permite a través de llaves de acceso, tener asegurada la infraestructura creada por el administrador. Y a través de estas llaves se asegura la disponibilidad e integridad en los servidores desplegados.
- **Administración de Roles:** Para el servicio de integridad, se crean roles los cuales según la jerarquía, están creados de acuerdo a las políticas de la empresa para cumplir con los estándares de seguridad de la información.
- **Administración de Almacenamiento a través de APIs:** Al igual que los servicios antes mencionados, el almacenamiento es uno de los más importantes pues aquí están los activos de la organización y en el caso que estos se vean afectados, pondrían en peligro la estabilidad de la empresa.

2.2.2. Requerimientos

El despliegue de Openstack define tres nodos cada uno tiene sus propias características según el manual de instalación de Openstack Juno:

Características de Nodos de Openstack

Nodo	N° de Procesadores	RAM	Disco
Controlador	1	2 GB	8 GB
Red	1	512 MB	8GB
Cómputo	1	2GB	10GB

Tabla 2. 2: Características de los nodos de despliegue [48]

2.2.3. Ventajas y Desventajas

Ventajas [17]

- No hay que pagar costos por las aplicaciones empleadas en Openstack ya que es software libre, excepto si se implementan aplicaciones que tienen costo.
- Permite el modificar y mejorar el diseño de software, ya que es software libre, modificarle en el sentido de, cambios que garanticen la integridad y la mejora de servicios.
- Permite lograr escalabilidad, pues al momento de agregar un servicio este puede ser rápidamente levantado y con conocimientos adicionales configurado para su funcionamiento.
- Ofrece flexibilidad en la administración y gestión de aplicaciones, gracias a los servicios implementados la administración es mucho más sencilla y fácil de entender, lo que beneficia al usuario final, que tan solo administrará los servicios que está brindando para su empresa.

Desventajas [17]

- Se requiere contratar a terceros para configuración del software para dar soporte a las aplicaciones que corren sobre la plataforma. En el caso que se implementen aplicaciones no conocidas, es necesario contratar a terceros, para administrar de una manera correcta.
- Falta de conocimientos sobre estándares a aplicar debido a que es una herramienta poco conocida. Al momento de desplegar aplicaciones, se podrían desconocer estándares de seguridad como de correcto funcionamiento, de ahí que se requieren conocimientos adicionales de gestión de aplicaciones.

2.3. Identificación de Módulos de OpenStack

OpenStack ofrece una gran cantidad de módulos, que permiten la administración y manejo de la nube. A continuación se detallarán los módulos más importantes. [36]

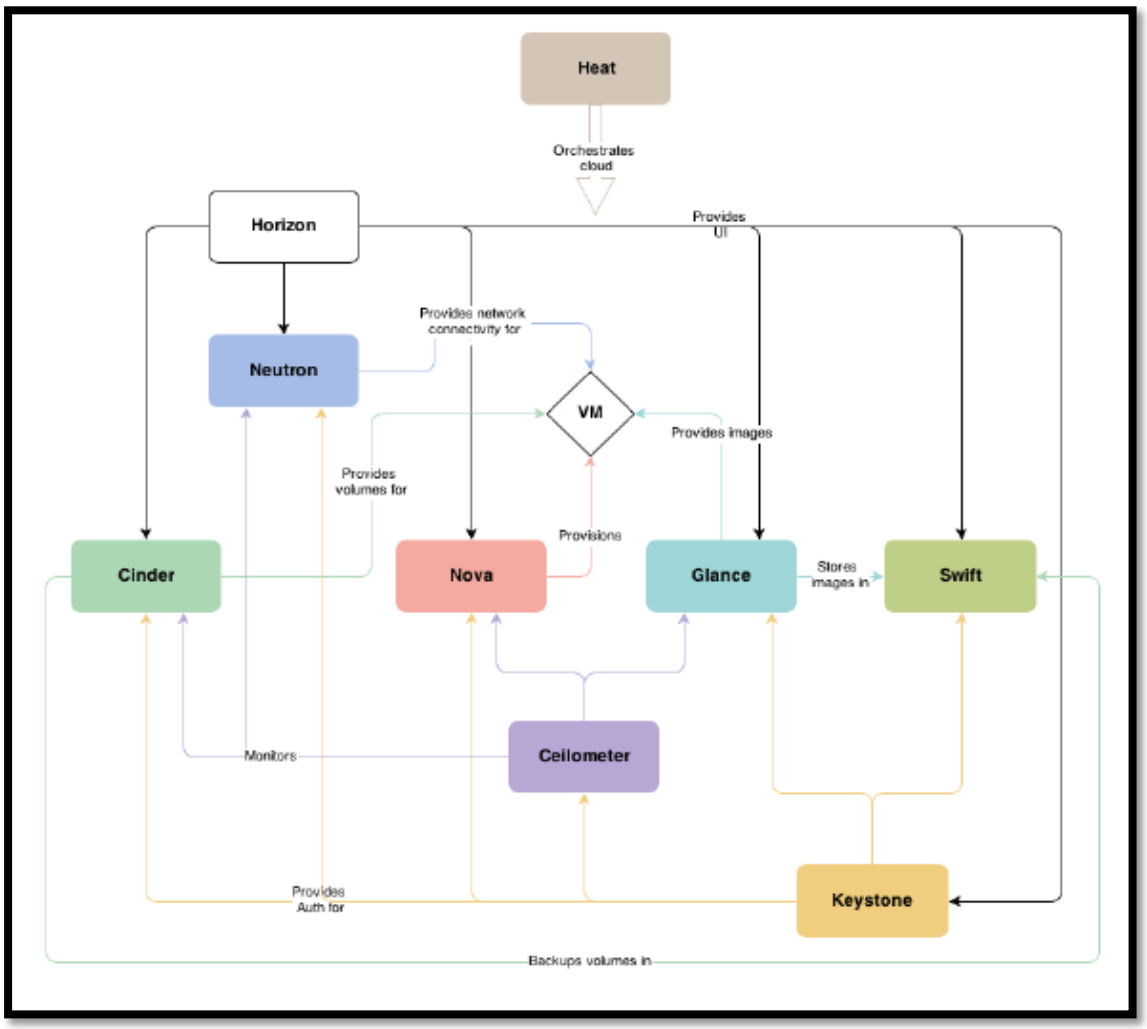


Imagen 2. 5: Despliegue de la nube [48]

La ilustración 2.5 detalla el despliegue y la forma como se interconectan los módulos:

- Principalmente los nodos se autentican y son administrados con el Horizon.
- Glance se encarga de la creación de máquinas virtuales.
- Nova se relaciona con todos los componentes para la gestión y termino de las máquinas virtuales.
- Neutron se encarga de la creación y gestión de las redes.
- Cinder brinda almacenamiento de volúmenes para las instancias.
- Glance almacena discos virtuales en Swift.

Arquitectura de Openstack

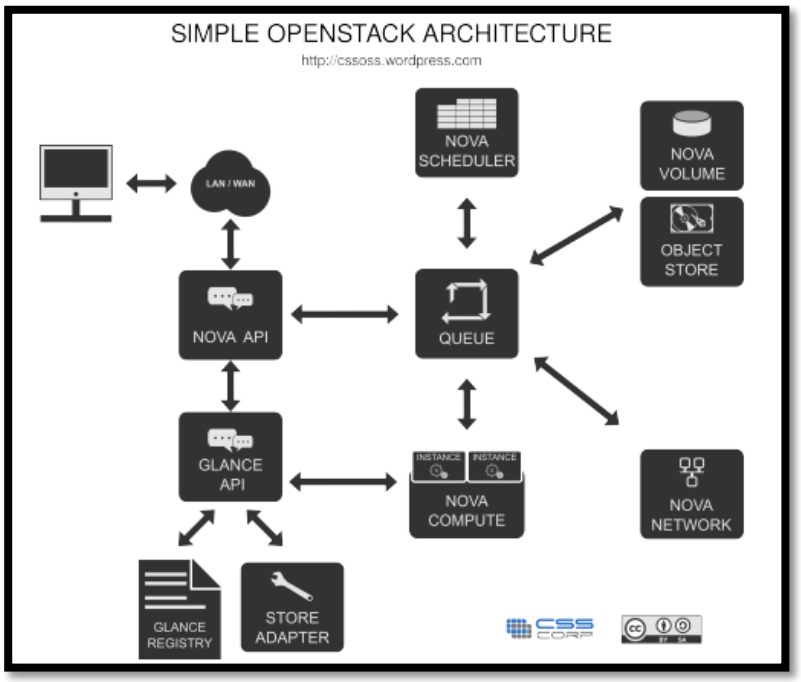


Imagen 2. 6: Comunicación entre módulos [46]

La ilustración 2.6 describe la comunicación de los servicios para el lanzamiento de una imagen comenzando con un cliente, que accede a la red privada (LAN), en donde a través del nova-api que se comunica con el Glance api, permite la elección de la imagen. Posteriormente la imagen es administrada mediante nova-compute en donde se gestiona el almacenamiento en Cinder así como la conexión a la red con Neutron.

2.3.1. Dashboard (Horizon)

Dashboard u Horizon es una interfaz web de OpenStack desarrollado en Django[44], que nos brinda algunas funcionalidades de manejo de la nube, también proporciona a los administradores y usuarios una interfaz gráfica para el acceso, gestión y sirve para automatizar los recursos basados en la nube [37].

[44] Django: es un framework web de Python de alto nivel que fomenta el rápido desarrollo y diseño de aplicaciones.

Horizon es una interfaz web de tipo WSGI[45], creada sobre apache, y que permite ejecutar aplicaciones a través de python. Por una parte permite la administración de aplicaciones y por otra facilita la intervención del cliente en el despliegue de la nube.

El acceso de los clientes se realiza por medio de una interfaz WEB, a través del protocolo (HTTP), para acceder a los servicios de Openstack.

Con Horizon se realizan operaciones como administración y lanzamiento de instancias, creación y asignación de redes y sobre todo autenticación de usuarios.

Este diseño facilita conectar y exponer los productos y servicios de terceros, tales como la facturación, seguimiento y herramientas de gestión adicionales [38].

2.3.2. Nova

Nova es un componente principal del proyecto OpenStack llamado también Compute, su función principal es permitir el control del cómputo, y representa la parte principal de los sistemas del modelo de despliegue IaaS. Se encarga de administrar el ciclo de vida de las instancias en la nube, a través de la ejecución de demonios los cuales se detallan a continuación.

Nova está compuesta por varios servicios o demonios *[34]*:

- **nova-api:** Permite aceptar las peticiones o llamadas de los servicios que envían los usuarios o los componentes de OpenStack mediante un API que permite interactuar con la infraestructura a través de las colas de mensajes como RabbitMQ. Se comunica directamente con nova-compute y nova-volume para la creación de máquinas y la asignación de la red.
- **RabbitMQ:** Permite comunicación entre componentes como scheduler, compute, etc.
- **nova-scheduler:** Permite planificar la ejecución de las instancias en los diferentes nodos, a través de un algoritmo que está basado en diferentes parámetros como carga, memoria, distancia entre zonas, tipo de arquitectura.
- **nova-compute:** Este servicio permite ejecutar una instancia sobre un hipervisor. Lo que hace este módulo es que recibe instrucciones y las ejecuta, estas acciones están comunicadas por una cola de mensajes gestionada por el paquete RabbitMQ, mediante AMQP[46].

[45] WSGI: Web Server Gateway Interface o Servidor Web Gateway Interface es una interfaz de servidor Web par alas aplicaciones de Python.
[46] AMQP: Advanced Message Queuing Protocol o Protocolo Avanzado de Cola de Mensajes, permite la comunicación a través de mensajes de confirmación entre cliente y servidor.

Compute al gestionar las instancias de imágenes, requiere de los hypervisores que deben ejecutarse para posteriormente contar con el acceso mediante consola. Openstack soporta diferentes hypervisores (KVM o Xen):

- KVM/QEMU[47]
- Xen
- Hyper-V[48]
- VMware[49]
- LXC[50]
- VirtualBox

- **nova-network:** Se encarga de la conexión de las máquinas virtuales, VLANs, grupos de seguridad y todo lo referente a comunicación en la nube. Aquí se configura la red tanto para las máquinas de la red interna como para la salida hacia la red externa.
- **nova-volume:** Permite administrar los volúmenes LVM, desde su creación. Aquí se crearán los volúmenes como unidades de disco que sirven como dispositivos de almacenamiento de las máquinas virtuales para su posterior configuración.

2.3.3. Neutron

Este módulo está encargado de la configuración de redes virtuales, su nombre inicialmente fue quantum, pero se lo cambió por Neutron desde la versión IceHouse de Openstack. Cuenta con complementos adicionales: OpenvSwitch, Cisco, Nicira, etc. Gestiona la red a través de direcciones IP, también permite administrar múltiples redes de forma independiente gracias al uso de Linux network name spaces, que permite que compartir un solo dispositivo de red físico en varias redes independientes. También se encarga de la gestión de los grupos de seguridad y de las IPs flotantes [37].

Los usuarios pueden crear sus redes, controlar el tráfico y conectar los servidores y los dispositivos a una o más redes, para esto permite la extensión de servicios de red como sistemas de detección de intrusos (IDS[51]), balanceo de carga, cortafuegos y redes privadas virtuales (VPN).

[47]KVM/QEMU: es un módulo del kernel de Linux que permite que un programa de espacio de usuario para utilizar las características de virtualización de hardware de varios procesadores.
[48] Hyper-V: programa de virtualización basado en un hipervisor para los sistemas de 64-bits con procesadores ADM o Intel.
[49] VMWare: Empresa desarrolladora de software de virtualización apra arquitecturas x86.
[50]LXC o Linux Containers es una tecnología de virtualización en el nivel de sistema operativo para Linux.
[51] IDS: Intrusion Detection System o Sistema de Deteccion de intrusos, es un software que permite protección ante posibles ataques de intrusión.

2.3.4. Cinder

Los volúmenes son dispositivos de bloques que al momento de su creación, esta se hace de forma independiente de la instancia pero pueden asociarse con cuando sea necesario. Los volúmenes se crean como LVM (Logical Volume Manager), en el nodo de almacenamiento y se conectan a la instancia mediante un protocolo de redes de almacenamiento como por ejemplo: iSCSI, que es el más utilizado. Cinder cuenta con controladores para gran variedad de dispositivos de almacenamiento del mercado [34].

Los volúmenes son considerados como discos externos que se conectan o desconectan de las instancias. Y ofrece las siguientes características:

- Alta disponibilidad
- Tolerancia a fallos
- Recuperable
- Uso de estándares abiertos
- Compatibilidad con APIs.

2.3.5. Glance

Gestiona las plantillas de imágenes de los sistemas, y también las instantáneas de las instancias. Provee servicios de búsqueda, registro y recuperación de las imágenes de discos virtuales, posee un api RESTFULL[52] que permite la consulta de los metadatos de las imágenes. Puede utilizar algunos formatos de almacenamiento de imágenes como *[35]*:

- raw
- qcow2 (Qemu/KVM)
- vhd (Hyper-V)
- ami (Amazon)
- vdi (VirtualBox)
- vmdk (VMware)

Glance consta con demonios que son los encargados de la creación de las imágenes de las máquinas virtuales y estos son:

- Glance-api: acepta las peticiones de los usuarios para la creación de las máquinas.

[52] RESTFUL: Representation State Transfer o Transferencia de Estado Representacional, es una técnica que describe el uso de XML y HTTP en interfaces WEB.

- glance-registry: se encarga de almacenar metadatos de las máquinas virtuales.
- Consta de una base de datos en Mysql donde se guardan las imágenes como plantillas.

2.3.6. Swift

Es un componente de OpenStack, que es independiente, y se encarga de almacenar objetos de forma redundante y escalable. Es empleado en aplicaciones Web, cuenta con una base de datos de objetos distribuidos que brinda alta disponibilidad, está relacionado con glance para gestionar imágenes y para la gestión de grandes cantidades de datos, este elemento permite levantar una réplica desde otros nodos activos. Tiene algunas características como [37]:

- Almacena objetos de gran tamaño
- Redundancia de Datos
- Almacenamiento seguro
- Maneja copias de seguridad.
- Permite escalabilidad

2.3.7. Keystone

Este módulo se encarga de la autenticación de usuarios y políticas de servicios para el uso específico de los módulos de Openstack, es el primer componente que se recomienda instalar y se parece a un directorio en donde se almacena toda la información de los proyectos *[37]*:

Se identifican dos funciones principales que desempeña el módulo Keystone:

- **Gestión de Usuarios:** dentro de esta tarea se administran algunos elementos como:
    - Usuarios: Clientes que administrarán la nube.
    - Proyectos (tenants): Es un perfil en donde se despliegan los servicios a emplear en la nube.
    - Roles: Son los permisos encargados de dar privilegios depende los requerimientos.

- **Registro de Servicios:** Permite el despliegue o administración de los servicios.

- Servicios y sus endpoints: Cinder, glance, telemetry, etc.

2.4. Administración de la Red de Usuarios

Para la administración de la Red los usuarios se identificarán dos grupos:

- **Administradores:** Encargados de la gestión de las aplicaciones desplegadas en la plataforma, requieren de un conocimiento detallado de la herramienta, así como de las aplicaciones que están ejecutándose, para realizar un correcto mantenimiento de los servicios.
- **Clientes:** Son los usuarios a los cuales están destinadas las aplicaciones y por lo tanto solo requieren una capacitación en cuanto a manejo de las aplicaciones mas no de la infraestructura desplegada. Los clientes se dedican propiamente a la explotación de los recursos de software y no deben conocer como están desplegados los servidores ni las redes configuradas.

2.4.1. Funcionamiento de Openstack

A continuación se describirá una explicación breve de cómo se representa el funcionamiento de la nube de forma sencilla [37]:

1. Un usuario interactúa con la API de nova (por medio del Horizon) para ejecutar una instancia, nova-api le pedirá que ingrese mediante el módulo Keystone.
2. Una vez autenticado le mostrará las imágenes disponibles en glance.
3. Cuando selecciona la imagen y las características para la instancia, se enviará la petición a nova-scheduler para lanzar la instancia, nova-scheduler determina en que nodo debe ejecutarse la instancia.
4. Nova-compute se encarga de ejecutar la instancia sobre el hipervisor sobre el que se esté implementando.
5. Nova-network se encarga de las configuraciones de la red para la comunicación interna y externa.
6. Una vez lanzada la instancia se puede entrar a esta mediante consola o interfaz gráfica.

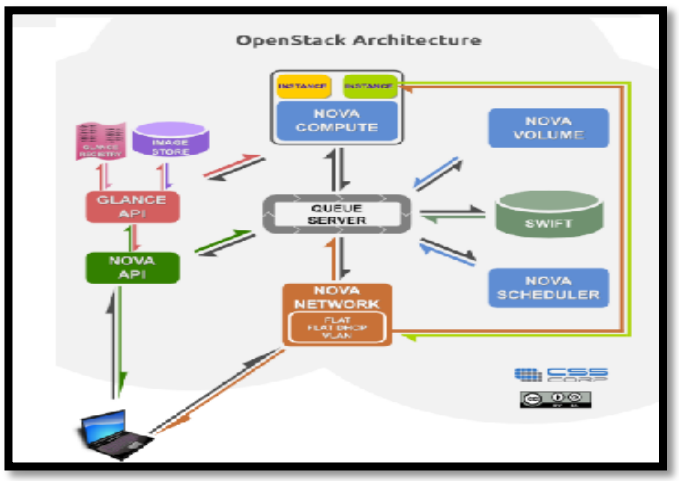


Imagen 2. 7: Funcionamiento de OpenStack [49]

CAPITULO 3

3. Diseño del Sistema

3.1. Arquitectura del Sistema

Openstack ofrece una gran flexibilidad en cuanto a la administración de computadores, redes y almacenamiento. Ofrece dos tipos de arquitecturas básicas:

3.1.1. Two-node Architecture with Legacy Networking (nova-network) o Arquitectura dos nodos con sistema heredado.

Tiene dos nodos que contienen diferentes servicios [36].

1. **Nodo Controlador (Controller Node)** se encarga del Soporte de Servicios, control de servicios básicos, y servicios opcionales.

    - **Soporte de Servicios:** Tiene conexión con las bases de datos, a través de un intermediario de mensajes como es: RabbitMQ o Qpid.

    - **Servicios Básicos:** Aquí están los servicios de Identificación (Keystone), servicios de Imágenes (Glance), servicios de computo (Nova Management) y servicios de interfaz de usuarios (Horizon).

    - **Servicios Opcionales:** Contiene bloques de almacenamiento (Cinder), objetos de almacenamiento (Swift), servicio de base de datos (Database Service), Orchestation, es un término que automatiza, coordina y gestiona sistemas complejos de computación, middleware y servicios (Heat), y sistemas de telemetría relacionada con la transferencia de datos y su monitorización (Celiometer).

2. **Nodo de Cómputo (Compute Node):** Se encarga de administrar servicios básicos, servicios opcionales [36].

    - **Servicios básicos** están los módulos de cómputo como Nova hipervisor, KVM o Quemu, y Nova Networking.

    - **Servicios Opcionales:** Emplea servicios como Cinder, Swift para almacenamiento, Orchestation para platillas de despliegue y telemetry para monitorización de la red.

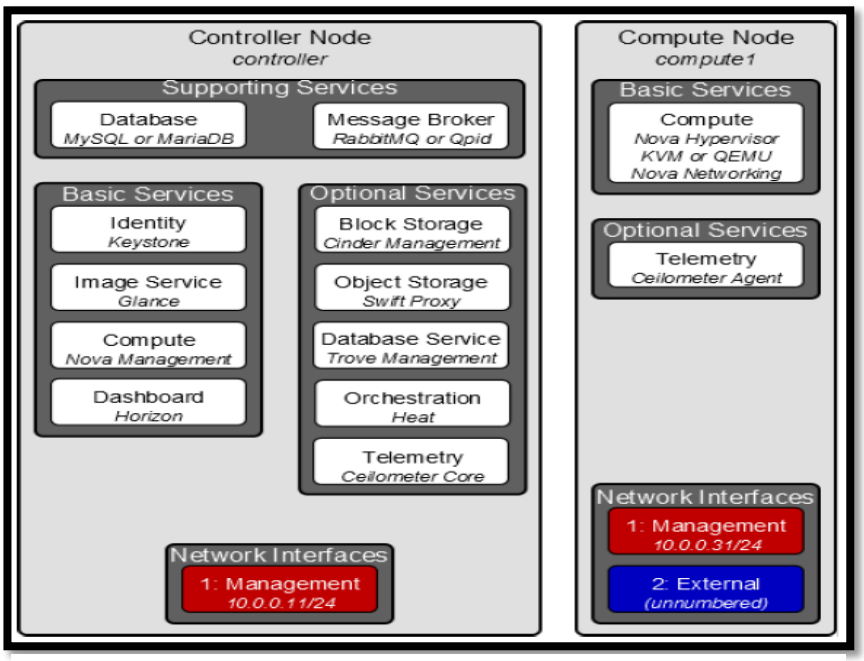


Imagen 3. 1: Despliegue de dos nodos

3.1.2. Three-node Architecture with OpenStack Networking (neutron) o Arquitectura tres nodos [36].

La arquitectura de tres nodos contiene diferentes servicios:

- **Nodo Controlador (Controller Node):** Este nodo está basado en la arquitectura de dos nodos y se encarga del Soporte de Servicios, control de servicios básicos, y servicios opcionales.
- **Nodo de Red:** Contiene servicios básicos como Agentes capa 2, agentes capa 3, agentes DHCP.
- **Nodo de Cómputo (Compute Node):** Permite administrar servicios básicos, servicios opcionales. En donde divide el único módulo de red en: módulo de cómputo y módulo de red.

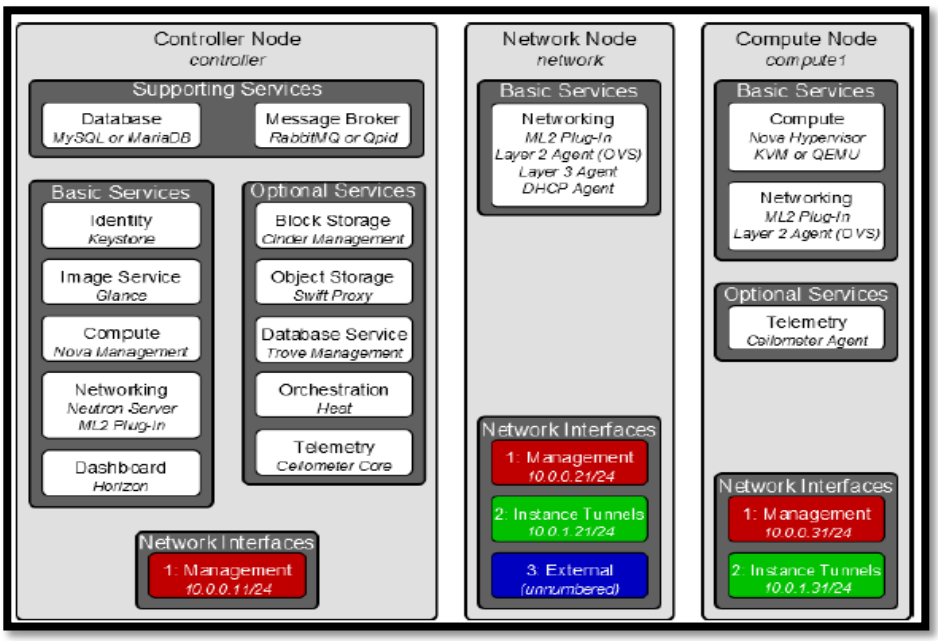


Imagen 3. 2: Despliegue en 3 nodos [48]

3.2. Diagramas de la Red

La red está dividida en tres máquinas virtuales que contienen las siguientes especificaciones para comunicación y gestión de la nube privada.

La estructura básica representada en la ilustración 3.3, demuestra gráficamente como están divididos los módulos en la nube, en donde compute se encarga de las instancias, que estarán corriendo dentro de la nube, el nodo network se encarga de conectar todos los elementos de la red y por último controller gestiona que todos los componentes interactúen correctamente en la nube.

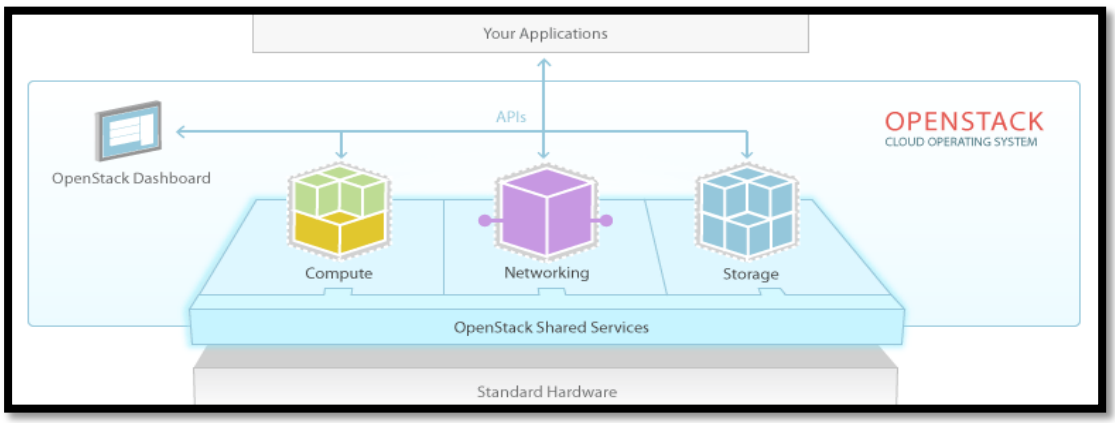


Imagen 3. 3: Despliegue de nodos [51]

La ilustración. 3.4 detalla la estructura, de la nube privada con las respectivas IP's de los nodos y la conexión de salida mediante un router con el fin de brindar el servicio de voz, a los clientes que se conectarán mediante teléfonos IP o smartphones al router y este a su vez al servidor Asterisk, el cual está dentro de la nube como una instancia, que de acuerdo a las configuraciones, permitirá que los usuarios puedan establecer las conexiones de voz dentro de la misma red.

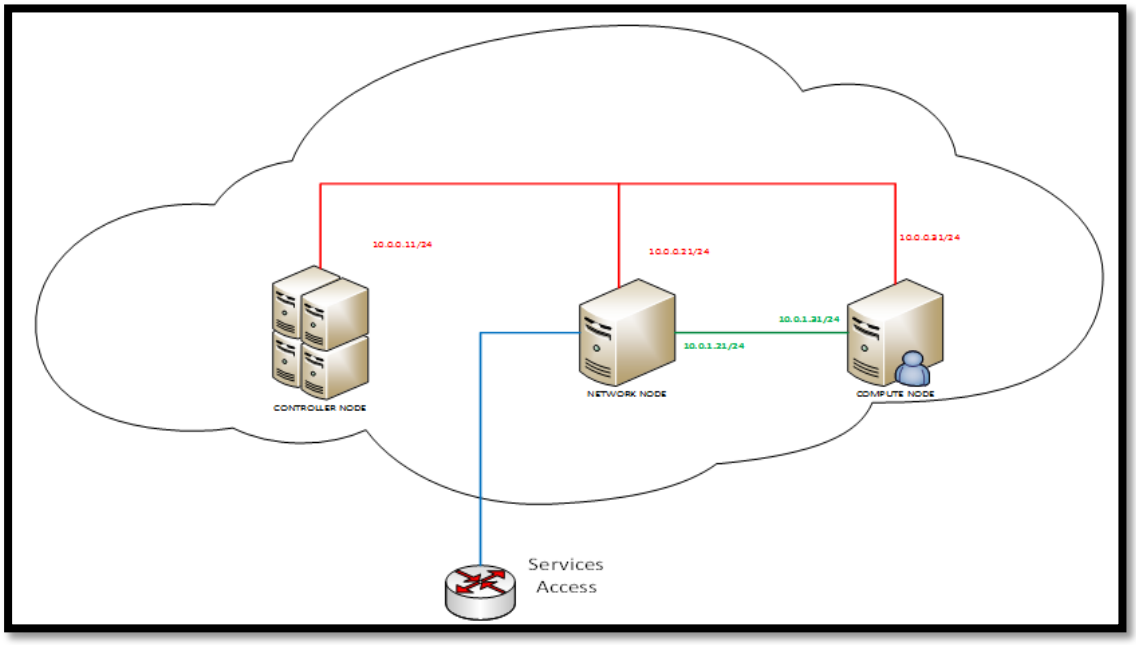


Imagen 3. 4: Diseño básico de conectividad de la solución propuesta

3.3. Herramientas de Virtualización

Existen varias herramientas que ofrecen virtualización, entre las más conocidas están VmWare, VirtualBox, XenServer.

De entre estas opciones la que más se utiliza por las prestaciones que brinda es XenServer, a través de Citrix XenServer.

3.3.1. VMware

Es un software de virtualización disponible para ordenadores compatibles con la arquitectura X86. Entre este software está VMware Workstation, que es pagado y los gratuitos son VMware Server y VMware Player. El software de VMware puede funcionar en Windows, Linux, y en la plataforma MacOS que corre en procesadores INTEL, bajo el nombre de VMware Fusion [39].

Algunas características de VMWare son [39]:

- VMware proporciona una forma conveniente de establecer conexiones remotas a hosts en Workstation, VMware vSphere y VMware vCenter.
- Los usuarios pueden compartir las máquinas virtuales con el equipo de trabajo, para probar las aplicaciones rápidamente en un entorno de producción.
- La interfaz de usuario VMware cuenta con menús sencillos, imágenes de las máquinas en miniatura, y se pueden buscar y tener acceso a una biblioteca de máquinas virtuales *[40]*.

3.3.2. Virtual Box

Es un software de virtualización, en donde el usuario puede cargar múltiples sistemas operativos invitados en un solo sistema operativo anfitrión. El usuario puede configurar, iniciar, pausar o apagar las máquinas creadas de forma independiente [41]. Permite el despliegue de servidores sin costo ya que existen versiones libres, por lo que es una solución conveniente al momento de emplearla dentro de las soluciones presentadas.

Imagen 3. 5: VirtualBox [54]

Algunas características de VirtualBox son:

- Puede ser instalado en diversos S.O. de 32 y 64 bits, como Windows, GNU/Linux, Mac OS X y Solaris.
- Puede virtualizar múltiples S.O. de 32 y 64 bits, como Windows, Debian, Ubuntu, OpenSuSe, OS/2, Mac OSX, Solaris, etc.
- Es libre ya que usa la licencia GPLv2, aunque algunos componentes son gratuitos con licencia PUEL[53].

[53] PUEL: Personal Use Evatuation License o Licencia Personasl de Evaluación y Uso de VirtualBox para uso académico.

- Es portable debido a que se pueden crear Máquinas Virtuales, en Windows y luego ejecutarla en GNU/Linux, pueden ser fácilmente importadas y exportadas, incluso se puede importar con un software de virtualización diferente.
- Se aplican las funciones de hardware actual como es la integración de virtualización por hardware como Intel VT-x [54]para los microprocesadores de Intel o AMD-V [55]para los micros de AMD *[42]*.

3.3.3. XEN Server

Citrix XenServer es una plataforma de virtualización de código abierto, que sirve para administrar servidores virtuales de nube, servidores sencillos y escritorios virtuales. XenServer se puede instalar de forma gratuita para virtualizar y automatizar procesos de administración, lo que aumenta la agilidad y flexibilidad de TI y reduce los costos.

Algunas características son [42]:

- XenServer es la mejor plataforma de virtualización de servidores para nubes públicas y privadas, fue creada con la idea de mantener la capacidad de ampliación, seguridad y con la aplicación multiempresarial, permite aún más flexibilidad y eficacia de costos.
- Tiene una plataforma que permite construir rápidamente una nube, diseñada para ejecutar carga de trabajo de múltiples aplicaciones.
- Al ser una plataforma de nueva generación las cargas de trabajo de aplicaciones, ofrecen disponibilidad, almacenamiento y redes al estilo Amazon.
- Está basado en Apache CloudStack de código abierto, lo que hace muy flexible y abierta , aprovechando las API de servicio web Amazon que son estándares que sirven para ampliar e integrar las soluciones y servicios añadidos.

3.4. Interfaces de Conexión

El modelo de red que estamos usando presenta 3 nodos principales, en donde se contarán con interfaces de conexión entre los nodos que componen OpenStack, y poder brindar la conexión con los usuarios para que consuman los servicios que se ofrecen. A continuación se listan las interfaces y las redes que intervienen en la conexión para el manejo y control de las aplicaciones que se vayan a desplegar dentro de la nube privada.

[54] Intel VT-x Intel Virtualization Technologies, tecnologías de virtualización de Intel.
[55] AMD-V: AMD virtualization es una tenología de virtualización aplicada a los procesadores AMD.

Diseño Planteado

Las interfaces que intervienen están divididas en tres grupos explicados a continuación:

- **Gestión o Red Privada**

Esta red nos permitirá la de los 3 nodos de la arquitectura. Su IP de red es la siguiente 10.0.0.021 con máscara de red 255.255.255.0; a cada uno de los nodos se le asignará una dirección dentro de este rango de red, al nodo de Controlador se le asignará la ip 10.0.0.11/24, al nodo de red se le asignará la ip 10.0.0.21/24, y al nodo de Cómputo se le asignará la ip 10.0.0.31/24.

- **Túnel**

Esta red permitirá la conexión entre el nodo de Cómputo y el nodo de Red, esta conexión se realizará a través de un túnel con la siguiente dirección de red 10.0.1.0 con máscara de red 255.255.255.0; al nodo de red se le asignará la ip 10.0.1.21/24, y a nodo de cómputo recibirá la dirección ip 10.0.1.31/24.

- **Externa o NAT o Red Pública**

Esta red permitirá al usuario final consumir servicios que se ofrecen en la nube privada, a través de Openstack. Para lo cual se le asignará una dirección de la red 172.16.26.0 con máscara 255.255.255.0.

El esquema de los módulos que intervienen con las respectivas interfaces:

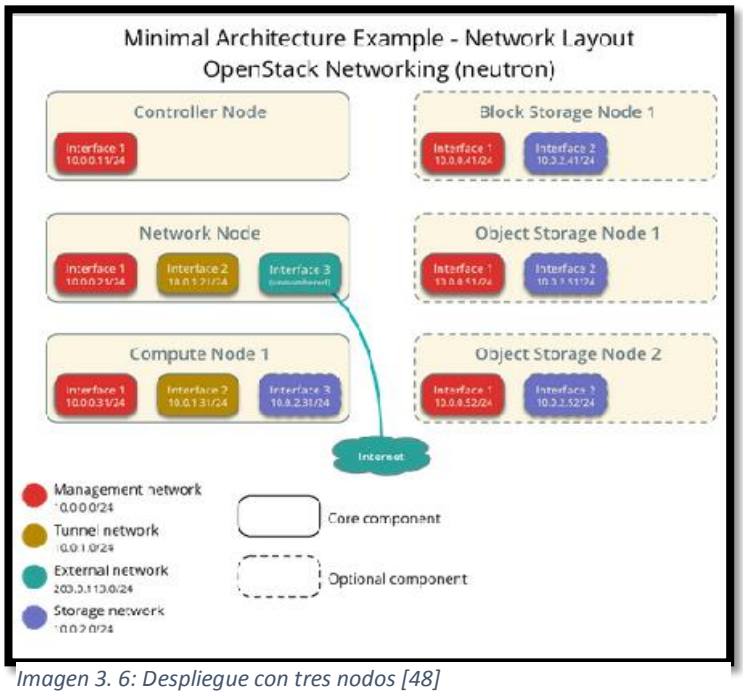


Imagen 3. 6: Despliegue con tres nodos [48]

Diseño Aplicado

En el diseño que se aplicó como solución al inconveniente con los sistemas virtualizadores, se cuenta con un solo servidor que cuenta con dos interfaces principales.

Las interfaces aplicadas son:

- **Interfaz de Administración o Gestión**

Esta interfaz sirve para para la creación de los servicios y los respectivos endpoint con el fin de administrar cada uno de acuerdo a los parámetros configurados. Además de la creación de servicios se encarga de la gestión de seguridad, de la creación de red interna que asigna las ips a las instancias desplegadas.

La IP de administración está en la red 192.168.2.0/24.

- **Interfaz de Conexión a la Red Externa**

Esta interfaz que esta sobre la red 172.31.3.0/24, sirve para permitir la conexión a los dispositivos externos, es decir, la conexión desde la instancia de Asterisk hacia los dispositivos externos como teléfonos IP o Smartphone. Las direcciones IP que se asocian con esta interfaz son conocidas como IP flotantes y su fin es establecer conexión con la red externa asociando las máquinas de la red interna a la red externa.

3.5. Software Implementado

Para la solución de Voz sobre IP, se instalará Asterisk sobre las máquinas virtuales plantilla que se crearon.

Asterisk es un servidor de voz, de código abierto que es muy utilizado debido a su facilidad de administración y empleo. Puede ser desplegado en Linux, Windows y MAC, tiene soporte para los protocolos SIP y IAX, por lo que es compatible con la mayoría de teléfonos, softphones y smartphones.

Tiene algunas ventajas como:

- Permitir la conexión de líneas de telefonía tradicional, mediante interfaces FXO o Interfaces de Conexión a la PSTN (Red Telefónica Pública Conmutada).
- Tiene soporte para troncales
- Soporta extensiones: SIP, IAX, H323
- Ofrece funciones básicas para los usuarios como: llamadas en espera, conferencias, transferencias, buzón de voz [43].

3.6. Seguridad en la Nube

En la implementación de Openstack para VoIP se puede implementar métodos de seguridad, incluyendo contraseñas, políticas, y el cifrado. La aplicación de seguridad en la nube privada estará principalmente contará con las contraseñas en cada servicio utilizado, posteriormente se aplicarán las políticas como el cifrado.

Igualmente al momento de la creación de las instancias se lanza un par de claves las cuales sirven para identificar que, la máquina que está corriendo en la nube es realmente la que se creó y no otra, que se pudo conectar remotamente por alguna falla de seguridad.

Las políticas de seguridad, se deben detallar en un documento, que deben conocer tanto los administradores como los usuarios que están yendo a brindar el servicio, aquí deben estar políticas como las de contraseñas, accesos físicos y lógicos, respaldos.

Además se pueden agregar mecanismos de protección como IPS e IDS, con el fin de detectar cualquier amenaza de acceso a la nube, también se podrían implementar analizadores de tráfico, o mantener monitorizados los equipos conectados para detectar intrusos dentro de la red.

En la implementación se detallarán políticas de accesos para lo que son los ingresos a infraestructura de la nube, posteriormente se aplicarán algunas políticas para detectar intrusos, de contraseñas, de usuarios, y finalmente se aplicarán algunas técnicas como monitores de tráfico para analizar intrusos y garantizar el aseguramiento de la red [5] [17].

Capítulo 4

4. Implementación de Sistema

Para la implementación del modelo IASS, aplicado al servicio de voz sobre IP, se instalarán módulos como el Keystone para autenticación de usuarios, glance para la creación de imágenes o mejor detallado como repositorio de máquinas virtuales, el módulo de nova se encarga de integrar los módulos para lograr el funcionamiento de la nube, neutron se encarga de la parte de Networking y permite crear conexiones entre las instancias creadas asignándoles ips dentro del rango creado y así mismo permite la conexión hacia la red externa o salida a internet, Dashboard en cambio se encarga de reunir todos los sistemas en una interfaz gráfica de administración a la que posteriormente se le agregan con Cinder los volúmenes para el arranque y funcionamiento de las instancias.

Los módulos adicionales para garantizar disponibilidad y calidad de servicio, son Orchestation que se encarga de mediante plantillas la creación más sencilla de volúmenes, redes, seguridad y usuarios. Otro módulo es el de Telemetría que sirve para el análisis de tráfico y características adicionales como estadísticas de las instancias desplegadas.

Análisis de Servicios

Los servicios principales implementados son los detallados en los manuales de instalación de OpenStack, dentro del diseño de la red se detallan los tres principales que contendrán las características adicionales para la implementación completa de la red:

- **Controller o Nodo de Control**, que es el encargado de comunicar los diferentes nodos para alcanzar la convergencia de nodos para gestionar la infraestructura y comunicarla con los clientes. Requiere de unas características más complejas que los demás nodos, pues aquí se realiza el procesamiento de todo el sistema y todos los servicios. En este nodo se instalan componentes como:
    - Keystone
    - Nova-api, nova-cert, nova-conductor, nova-consoleauth, nova-scheduler, novaclient
    - Plugins de OpenvSwitch
    - Glance-Api
    - Cinder-Api

- **Network o Nodo de Red**, Se encarga de la conexión física o virtual entre los nodos como levantamiento de interfaces, conexión de redes y subredes. Permite que las máquinas creadas en el nodo de cómputo se conecten a través de la instalación de paquetes de red. Aquí se instalan paquete como:
    - Neutron-Api
    - El agente DHCP
    - Paquetes de OpenvSwitch

- **Compute o Nodo de Cómputo**, este se encarga de gestionar el ciclo de vida de instancias que están siendo implementadas en OpenStack. Las responsabilidades incluyen la implementación, la programación y la eliminación de las máquinas virtuales. Recibe las órdenes del nodo controller y procesa la gestión de las máquinas virtuales En este nodo se instalan los paquetes de:
    - Nova-Compute
    - OpenvSwitch

Esquema de Red Propuesto

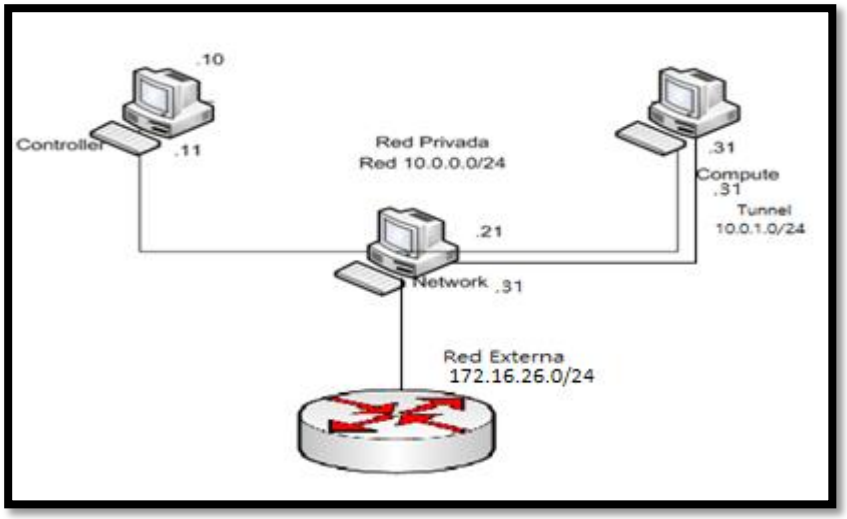


Imagen 4. 1: Esquema de red de la implementación IaaS

De acuerdo al esquema planteado para la implementación se presentan principalmente la red privada, está representada con el color rojo y se encarga de conectar los tres nodos bajo la red 10.0.0.0/24, y sirve para la interconexión entre nodos y la gestión de las colas de mensajes de RabbitMQ.

Los túneles entre los nodos de red y cómputo, representados de color marrón dentro de la red 10.0.1.0/24 representan la conexión entre las máquinas virtuales.

La conexión a la red externa representada por el color turquesa dentro de la red 172.16.26.0/24 nos sirve para conectarnos hacia internet y acceder a la WEB, para ciertos servicios.

Diseño de la Implementación

Esta arquitectura de implementación consta de 3 nodos en donde se instalarán los servicios básicos de acuerdo a la imagen, y también contendrá servicios adicionales como almacenamiento, coordinación (orchestation) y telemetría.

Identificación de Nodos

Nodo	IP Administración	Red Privada
Controller	172.16.26.5	10.0.0.11/24
Red	172.16.26.6	10.0.0.21/24
Cómputo	172.16.26.7	10.0.0.31/24

Tabla 4. 1: Asignación de IPs

Detalle de Redes que intervienen en la Nube privada

- **Red Privada:** Es una red que se usa exclusivamente para la comunicación interna de los nodos de la nube.
- **Red Externa:** Red externa hace referencia a la red real a la cual está conectado nuestros servidores, y a través de la cual podremos acceder a las instancias que se van a desplegar en nuestra nube.
- **Túnel:** Este túnel se utiliza para comunicar solamente el nodo de Red y el nodo de Cómputo.

Esquema de Red Aplicado

Diseño Aplicado

Para la solución aplicada el diseño de red cuenta con un servidor el cual está conectado tanto a la administración como a la red externa.
La interfaz de administración con la IP 192.168.2.103
La interfaz externa con la IP dentro de la red 172.31.3.1

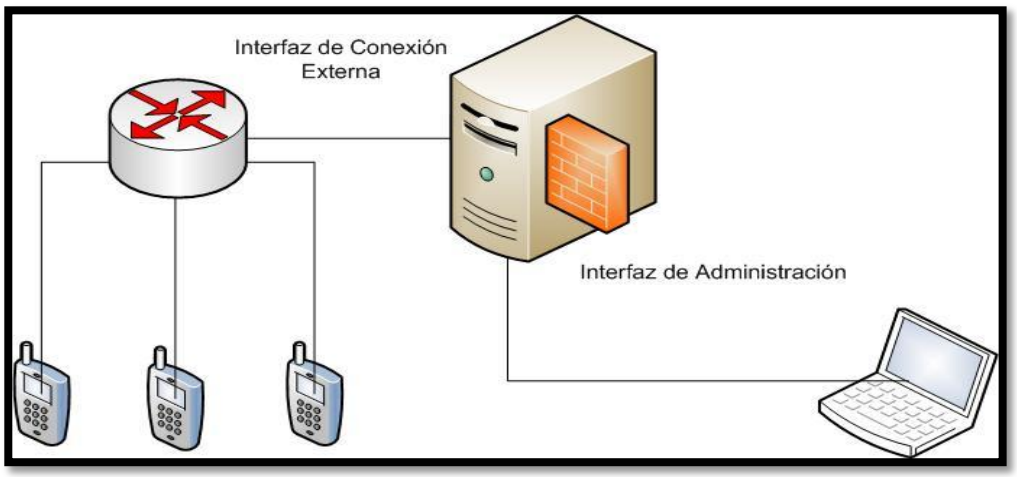


Imagen 4. 2: Esquema de red de a implementación aplicada

Análisis de Servicios

Existe un solo nodo controlador se encarga de administrar todos los servicios desplegados y la comunicación entre módulos.

- **Controlador:** Aquí se despliegan todos los elementos para la creación de las instancias y su almacenamiento. Así como su monitoreo y gestión de usuarios.

  Los servicios que encontramos aquí son:

  - Keystone
  - Nova-api, nova-cert, nova-conductor, nova-consoleauth, nova-scheduler, novaclient
  - Plugins de OpenvSwitch
  - Glance-Api
  - Neutron
  - Cinder

Características del Servidor de la Solución Aplicada

Para la implementación de un solo nodo son necesarias ciertas características que se detallarán a continuación como solución al inconveniente de virtualización de servidores. Las características básicas son:

Nodo	CPU	RAM	DISCO	N° NICS	Virtualización
Controlador	Core Dos Duo	4096	250	2	Intel-VX

Tabla 4. 2: Hardware del servidor aplicado como solución

4.1. Virtualización de Servidores

La implementación de los nodos o servidores necesarios para el despliegue de la nube, se los realizará con el virtualizador Xen server en su última versión, a continuación se detallará los requerimientos mínimos para el despliegue de cada nodo:

A continuación se detallara paso a paso la instalación de cada uno de los nodos.

Nodo	CPU	RAM	DISCO	N° NICS	Virtualización
Controlador	1	4096	60	2	Xen
Red	1	2048	60	3	Xen
Cómputo	1	4096	200	3	Xen

Tabla 4. 3: Hardware recomendado por nodo (Ideal)

Ahora se proseguirá con la instalación detallada en el Anexo 2, que se realizó sobre equipos propietarios de la universidad con las siguientes características. Para esto se emplearon los tutoriales Openstack versión Juno e IceHouse para Ubuntu 14.04 [36].

Para la instalación se realiza sobre una máquina con las siguientes características: 4 GB de RAM, 500 disco, 2 NICS, Core Dos Dúo de procesador y está instalado el Sistema Operativo Ubuntu 14.04, la instalación de los servicios se detallará en el Anexo 2, y que servirán para cualquier configuración estándar pues se montan para cualquier tipo de servidores. Anexo 2 Instalación Openstack 3 Nodos

Implementación Aplicada un Solo Nodo

La implementación Aplicada en un solo nodo no requiere de virtualizadores pues se realiza en un servidor con las características de la tabla 6.

Cuenta con dos interfaces una externa y una interna, todos los servicios se encuentran dentro del mismo nodo. Para detalles de la configuración se debe revisar el Anexo 3 Instalación Openstack Icehouse un solo nodo.

4.2. Configuración de Complementos

La instalación básica de un sistema de cloud desarrollado con Openstack consta de tres nodos básicos: Nodo Controlador, Nodo de Red, Nodo de Cómputo. Sin embargo existe varios servicios adicionales a los que ya se han nombrado anteriormente, los cuales nos permiten realizar varias funciones extras de la nube; a continuación se explicarán para que sirven estos servicios adicionales [36].

Storage (Cinder)

El servicio de Storage permite ofrecer a los usuarios almacenamiento persistente para las instancias que se podrán desplegar, ya que cada instancia puede desplegar un volumen, el cual estará ubicado en este nodo opcional. Consta de cuatro componentes que son:

- cinder -api: enruta las peticiones al cinder-volume.
- cinder-volume: se comunica con el block-storage a través de una cola de mensajes con el fin de almacenar estados de las instancias.
- cinder-scheduler: Almacena datos en el nodo más óptimo de acuerdo a las configuraciones de almacenamiento.
- messaging quenue: se encarga de la cola de mensajes entre procesos de almacenamiento.

Orchestation

Este módulo contiene un conjunto de plantillas para crear ciertos elementos en Openstack como por ejemplo: IPs flotantes, volúmenes, grupos de seguridad. Lo que hace es crear plantillas para grandes aplicaciones que requieren gran cantidad de recursos y al momento de implementar se pueden crear scripts con las plantillas previamente creadas. Este módulo puede ser implementado directamente o a través de otros complementos de Openstack [36].

Este módulo cuenta con 4 elementos [36]:

- heat command-line: Permite al cliente conectarse con el heat-api y realizar actividades como desarrollador con la ayuda de orchestation.
- heat-api-component: Permite que a través de RPC (Llamadas a Procedimientos Remotos) se procesen las solicitudes al heat-engine.
- heat-api-cfn: Procesa las solicitudes api a través de las RPC.

- heat-engine: Procesa plantillas y eventos ante solicitudes.

Telemetry

Este módulo se encarga de recoger información de rendimiento de los CPU, la red, a través de notificaciones enviadas por los servicios. También se pueden obtener datos desde plugins, cuenta con firmas digitales que no puede negar transacciones, contiene cinco componentes como [36]:

- ceilometer-agent-compute: se ejecuta en el nodo de cómputo y genera estadísticas sobre recursos del sistema.
- ceilometer-agent-central: Se ejecuta en un servidor de administración central para obtener estadísticas de utilización de recursos
- ceilometer-collector: Se ejecuta en el servidor controller para supervisar las colas de mensajes procedentes del agente. Los mensajes de notificación son mensajes de medición y se guardan en el almacén de datos sin modificación.
- ceilometer-alarm-notifier: Se ejecuta en el servidor controller para permitir que las alarmas se creen en base a una colección de muestras.
- data store: Es una base de datos que maneja escrituras concurrentes.
- ceilometer-api: Se ejecuta en el controller para proporcionar acceso a los datos.

4.3. Configuración de Seguridad

Se emplea los paquetes de OpenSSL, con el fin de generar tokens para determinadas acciones como autenticación entre máquinas, ingreso de usuarios y conexiones con la plataforma [5] [57].

- Se generan los token para autenticación: Se crean archivos para autenticación en keystone y con estos archivos que setean los token que permiten crear usuarios, roles, tenants (proyectos) y servicios.
- Se generan los token para identificación entre máquinas: Al momento de crear las instancias de las máquinas virtuales a desplegar se crea un token de seguridad que permite enlazar la maquina administradora con la instancia.
- Se generan los token para acceso a máquinas: Al generar los token en la creación se puede tener acceso mediante telnet o ssh a las instancias. En el caso de no crear los token no se accede a las maquinas instanciadas.
- Seguridad para base de datos: La máquina administradora contiene MYSQL como base de datos con el comando mysql_secure_installation, que elimina usuarios por defecto y quita acceso remoto a la base de datos.

- Seguridad para Usuarios: cada usuario tendrá su respectiva contraseña, así como el proyecto asignado para su administración con el fin de evitar fallas de servicios de acuerdo a los perfiles asignados.

4.4. Implementación alternativa

Dado que la implementación en máquinas virtualizadas, se ha decidido realizar la implementación de un entorno de prueba en un solo nodo, llamada instalación AIO (All In One), o Todo en Uno, la cual nos brindará un ambiente de prueba, para poder realizar la implementación del servidor VoIP Asterisk, y poder brindar el servicio de telefonía IP a los clientes.

Luego de realizar la instalación y prueba de varios entornos de prueba como son devStack, Mirantis y Cisco, la instalación que mostró mejores resultados fue el despliegue de la solución desde los repositorios de Cisco.

Para la implementación o despliegue de la nube de pruebas se siguió un tutorial, proporcionado por la página de Cisco Systems, el cual lo podemos encontrar en: http://docwiki.cisco.com/w/index.php?title=OpenStack:_Icehouse_All-in-One&oldid=60411, para el despliegue de un ambiente AIO.

El despliegue fue basado en este tutorial, se ingresó en los archivos de configuración para editar las redes internas, y las ip flotantes para acceso externo.

Una vez realizada la instalación nos dirigimos al ingreso web, a través de la dirección http://192.168.2.103 para acceso administrativo, para el acceso a las instancias, nuestra red externa debe ser 172.31.3.0/24, ya que las ip flotantes que serán asignadas a nuestras instancias están dentro de esta red.

Una vez que realicemos el ingreso a Openstack, podemos observar los módulos necesarios, para el lanzamiento de las instancias. Entre los más importantes podemos encontrar el Keystone o módulo de autenticación, nova que es el encargado del cómputo, neutrón encargado del Networking de la solución de Cloud computing, volume encargado del almacenamiento que se va a asignar a cada una de las instancias desplegadas, y glance que es el servicio de almacenamiento de Imágenes de disco, que serán usadas para el despliegue de instancias dentro de nuestra nube.

Cabe recalcar que existen varias imágenes predefinidas que deberemos subir a la nube, para su posterior lanzamiento, estas imágenes prediseñadas pueden ser varios tipos de sistemas operativos los cuales en ocasiones son de menores características que un Sistema operativo real, pero al estar restringidos en cuanto a las imágenes que podemos subir, nos servirán para realizar las pruebas de nuestro servidor de voz.

Desafortunadamente Elastix no es una imagen soportada por este servidor AIO, se tuvo que optar por la instalación de una servidor Asterisk sobre un Ubuntu server que se desplego dentro de la nube, dado que el Ubuntu que se subió a la nube no necesita instalación, solamente necesita ser lanzado, realizar la asignación de la ip flotante para el acceso externo y actualización e instalación de paquetes.

Una vez ingresado dentro de la maquina realizado el update del Sistema operativo, procederemos a la instalación del servidor VoIP Asterisk.

Ver anexo Asterisk

Capítulo 5

5. Pruebas

5.1. Herramientas de Pruebas de Servidores

Debido al crecimiento y complejidad de las aplicaciones, los servicios implementados en las empresas requieren una administración y mantenimiento en tiempo real, debido a este tipo de requerimientos se debe tener en cuenta que las pruebas tanto en disponibilidad de servicios, integridad de datos y autenticación, que deben seguir estándares para garantizar el correcto funcionamiento de la nube.

Un reto en estas pruebas de Cloud Computing es el de mantener la seguridad, los servicios y las aplicaciones de forma optimizada, ya que como esta tecnología es nueva, y al no existir gran cantidad de estándares como para las TI anteriores, se requieren una serie de políticas creadas por los administradores en donde se detallen los accesos, a los servicios, a los usuarios, a las instancias, a la comunicación de la red, y a la infraestructura creada para que no existan inconvenientes en su funcionamiento.

Imagen 5. 1: Pruebas en la nube [57]

Para es testeo en la nube se requieren analizar algunos factores como [44]:

- Las pruebas deben ser planificadas, debido a que se deben realizar periódicamente garantizando así la integridad de los servicios que se están ofreciendo.
- Las pruebas no deben afectar los datos sensibles, no deben representar mayor impacto, pero son necesarias para analizar tráfico inusual, usuarios sin conocimientos, administradores descuidados, que representan alertas para el funcionamiento de la nube.
- Las pruebas más recomendadas son las que se deben realizar sobre aplicaciones móviles y web, así también existen pruebas para sistemas operativos y actualizaciones, navegadores y versiones, diferentes tipos de hardware y un gran número de usuarios simultáneos en tiempo real.

- Los entornos de pruebas que se aplican, son complejas ya que consumen capital y recurso.

Estás pruebas se encuentran frente a varios retos, que al superarlos podrán brindar resultados mucho más confiables, estos retos son [44]:

- En la actualidad no existen estándares para integración de nubes públicas y privadas por lo que no hay aún interoperabilidad completa entre nubes.
- Algunos proveedores que rentan las plataformas no permiten realizar ciertas configuraciones, tecnologías, servidores, almacenamiento, redes y anchos de banda, por lo que es complicado crear entornos de pruebas para estas implementaciones.
- La falta de conocimientos es un factor que puede causar que los entornos de pruebas si no son controlados, generen gastos, por esta razón hay que conocer bien como se deben realizar estas pruebas de forma segura.
- Los entornos de pruebas deben estar bien estructurados de acuerdo a cronogramas en donde se detallen las actividades, así como los gastos de encriptación de datos, y consumo de recursos como CPU y la memoria.

Acciones para garantizar estabilidad en la nube al momento de realizar pruebas *[44]*:

- Se deben preparar los escenarios para garantizar un ahorro que beneficiaría a la empresa.
- Se deben construir proyectos de pruebas, en donde, los desarrolladores y testeadores deben detallar las pruebas de rendimiento de los recursos de la nube.
- Se debe tener en cuenta los entornos de prueba de forma cuidadosa, ya que al seleccionar las herramientas y aplicaciones de prueba y determinar cuánto tiempo van a requerir los testeadores, pueden surgir inconvenientes que deben ser debidamente planificados.
- Para la ejecución de las pruebas, estás se deben realizar de acuerdo a las estrategias formuladas a fin de tener una utilización óptima de la infraestructura.
- Estas pruebas sirven para analizar el comportamiento y tomar acciones tanto preventivas como correctivas en la nube para garantizar su funcionamiento en tiempo real.

Ventajas de las Pruebas en la Nube *[44]*

- Para reducir los costos se debería planificar de acuerdo a los factores mencionados, los cuales si se aplican correctamente presentan en beneficios económicos para la organización.
- Los requerimientos necesarios en cada una de las pruebas puede ser modificado, reduciendo el tiempo.
- Las pruebas se las pueden programar para horas específicas, en donde no es necesaria la presencia del testeador pero siempre se debe informar de las actividades a realizar en los horarios establecidos.
- Las pruebas pueden hacerse tanto dentro de la red como fuera siempre y cuando se tengan configuradas VPN, con los respectivos permisos.

Tipos de Pruebas en la Nube

Estas pruebas deben ser confiables y garantizar el funcionamiento de aplicaciones e infraestructura, para analizar las pruebas a aplicar en la nube primero se analizaran los tipos de pruebas *[45]*:

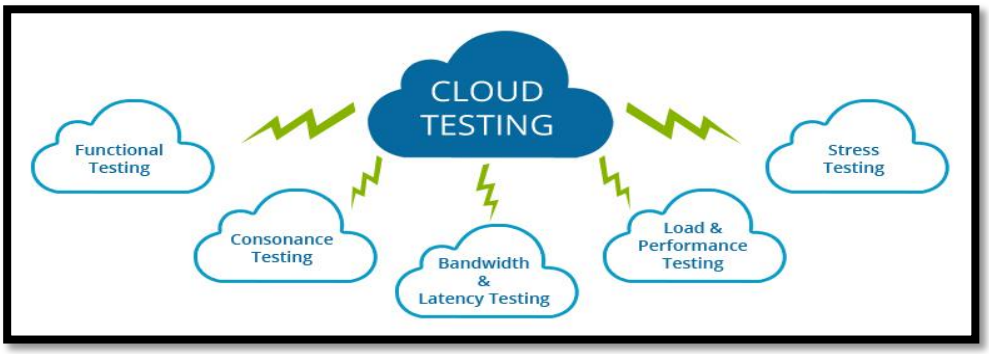


Imagen 5. 2: Tipos de Pruebas

- **Pruebas de disponibilidad:** Estas pruebas garantizan la disponibilidad 7x24.
- **Pruebas de seguridad:** Estas pruebas se encargan de la integridad de los datos y el acceso seguro a la nube.
- **Pruebas de interoperabilidad:** Estas pruebas garantizan el funcionamiento de las aplicaciones en las diferentes plataformas.
- **Pruebas de rendimiento:** Estas pruebas se encargan de probar las cargas con el fin de analizar la demanda de los servicios de la nube y el tiempo de respuesta y peticiones para encontrar el punto más congestionado en la nube.

- **Pruebas Funcionales**, Estas pruebas se encargan de determinar el cumplimiento de los requerimientos del cliente.

5.2. Metodología de Pruebas

Para realizar las pruebas se empleó la metodología llamada, el Círculo de Deming o más conocida como PDCA, debido a los pasos con los que se deben cumplir para realizar una mejora continua de calidad en SGC (Sistemas de Gestión de Calidad) y SGSI (Sistemas de Gestión de Seguridad de la Información), consta de cuatro fases que se detallarán a continuación *[46]*:

- **Plan o Planear:** Permite identificar el problema y definir qué es lo que se va a realizar para cumplir el objetivo deseado. Se puede realizar acciones como:
    - Decidir los objetivos
    - Escoger las metodologías
- **Do o Hacer:** Permite detallar las acciones a cumplir de acuerdo a los objetivos propuestos en el Plan. Se puede cumplir acciones como:
    - Realizar adiestramiento y formación
    - Cumplir con el trabajo propuesto
- **Check o Revisar:** Permite llevar un registro de cumplimiento de actividades e ir revisando las acciones que se están realizando.
    - Si los objetivos no son cumplidos volver a planificar.
- **Act o Actuar:** Permite controlar e ir corrigiendo fallas en las acciones que se están realizando.
    - Corregir las fallas que se detectaron.

Antes de cumplir con las fases de la metodología se deben responder algunas preguntas básicas con las que se arrancará con el empleo de la metodología y las preguntas son:

- **Que**
- **Porqué**
- **Cómo**
- **Dónde**
- **Cuándo**
- **Quiénes**

Esta metodología es continua y pretende mantenerse en constante control y acción para garantizar que se están cumpliendo con los objetivos planteados y en una mejora continua y evita el desperdicio de recursos.

Ventajas

- Planificación de Estrategias
- Permite prevenir y corregir rápidamente
- Las acciones son rápidas

Aplicación de la Metodología

En la tesis propuesta se detallarán las preguntas para arrancar con la implementación de la metodología aplicando cada fase para cumplir con los objetivos propuestos:

- **Que:** Aquí se describen los objetivos que se intentan alcanzar.
    - Analizar el tráfico del servidor
    - Monitorear los Servidores
    - Realizar Pruebas de seguridad y tráfico

- **Porqué:** Justificación de objetivos a cumplir
    - Porque se desea monitorear y administrarla a conveniencia de acuerdo a los parámetros establecidos.

- **Cómo:** Detalla que elementos se emplearan para cumplir con el objetivo establecido
    - Instalar los complementos de escaneo para pruebas.
    - Configurando los elementos que intervienen como servidores, redes, clientes, etc.

- **Dónde:** Es el lugar donde se aplicará la implementación.
    - Principalmente se presentó una solución propia que contaría con medio propios pero en vista de la necesidad de servidores de características especiales se analizó la implementación en una red privada pero al no resultar la virtualización como solución se realizó la instalación en un servidor físico.

- **Cuándo:** Es la fecha desde la que se comienza con la implementación.
    - Según cronograma la implementación se la realizó desde el mes de octubre, en donde se realizaron pruebas con equipos que no eran suficientes por lo que se pidió el apoyo de la universidad y se comenzó la implementación desde el mes de diciembre (16) finalmente no contando con los resultados esperados se procedió en el mes de enero con la instalación en un servidor físico.

- **Quiénes:** Son las personas que intervienen en el desarrollo de la tesis.
    - Estudiantes: Adriana Cornejo y Christian Díaz
    - Tutor de Tesis: Ing. Pablo Gallegos

PLAN o Planear

En esta etapa se detallan los objetivos descritos en el ¿QUÉ?

Objetivos

Objetivo	Detalle
Instalar el servidor de VoIP Asterisk	Una vez con todos los elementos se procede con la instalación del servidor de voz.
Configurar el servidor con todos los requerimientos	Se realizan las configuraciones de las cuentas para las llamadas y los contestadores.
Analizar el tráfico del servidor	Se instalarán algunas herramientas que sirven para realizar el análisis de los servidores.
Monitorear los Servidores	Se procede con el escaneo de ips y servicios para hacer pruebas de monitoreo.
Realizar Pruebas de seguridad y tráfico	Se realizarán pruebas de seguridad y tráfico con software libre.

Tabla 5. 1: Tabla de objetivos

DO o Hacer

En esta fase se cumplirán con todas las actividades detalladas en el cronograma y en los tiempos determinados pero siempre suelen haber inconvenientes que estarán contemplados dentro del plan de cumplimiento.

Imagen A6. 1: Cronograma

CHECK o Revisar

Se realizaron una lista con las tareas de cumplimiento en donde se detalla, los elementos, las actividades, los controles, etc.

Tabla A5. 1: Actividades realizadas.

ACT o Actuar

Aquí están las personas que cumpliremos con determinadas tareas para esto en el cronograma también se describen tareas pero se ha realizado un cuadro de tareas en resumen de las actividades cumplidas y el tiempo que se requirió.

Para proseguir con las pruebas planteadas se analizarán una serie de testeadores de software que permitirán conocer el estado de la infraestructura así como de los servicios desplegados. Existe gran variedad de software para pruebas en la nube pero solo se analizarán para tres tipos de servicios como son conexiones, monitorización y seguridad.

5.2.1. Pruebas de Conexión

Para realizar las pruebas de conexión principalmente se procederá con la utilización de PING, que es una herramienta muy utilizada en los sistemas operativos para probar conectividad en un red tanto local como remota, a través del envío de paquetes ICMP, trabaja en la capa de red y permite probar la conectividad entre host de una red IP [47].

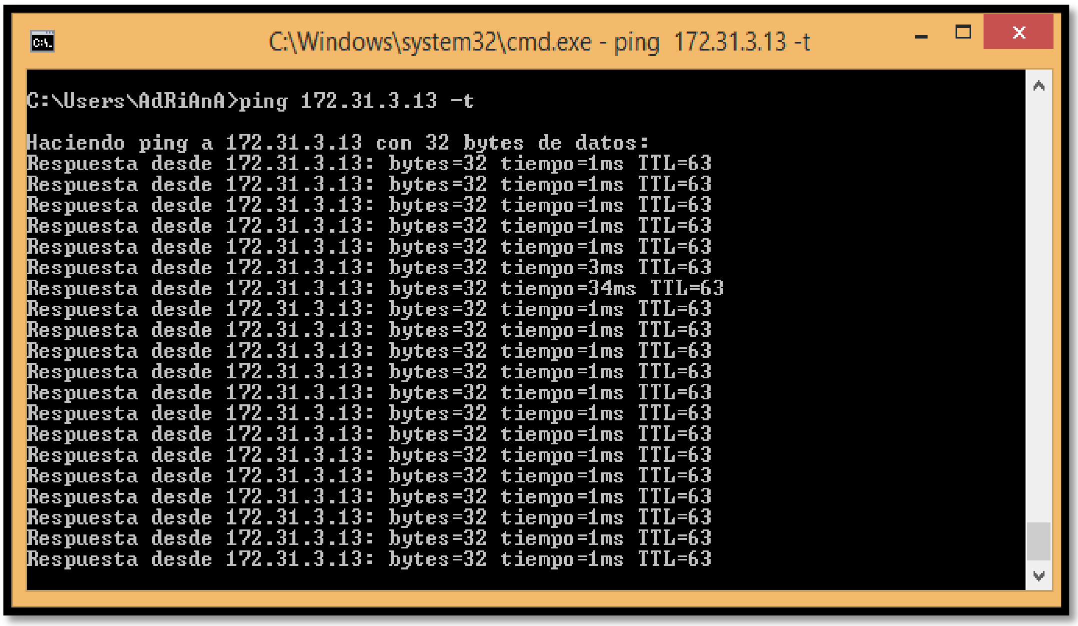


Imagen 5. 3: Pruebas de conectividad a la infraestructura de acceso

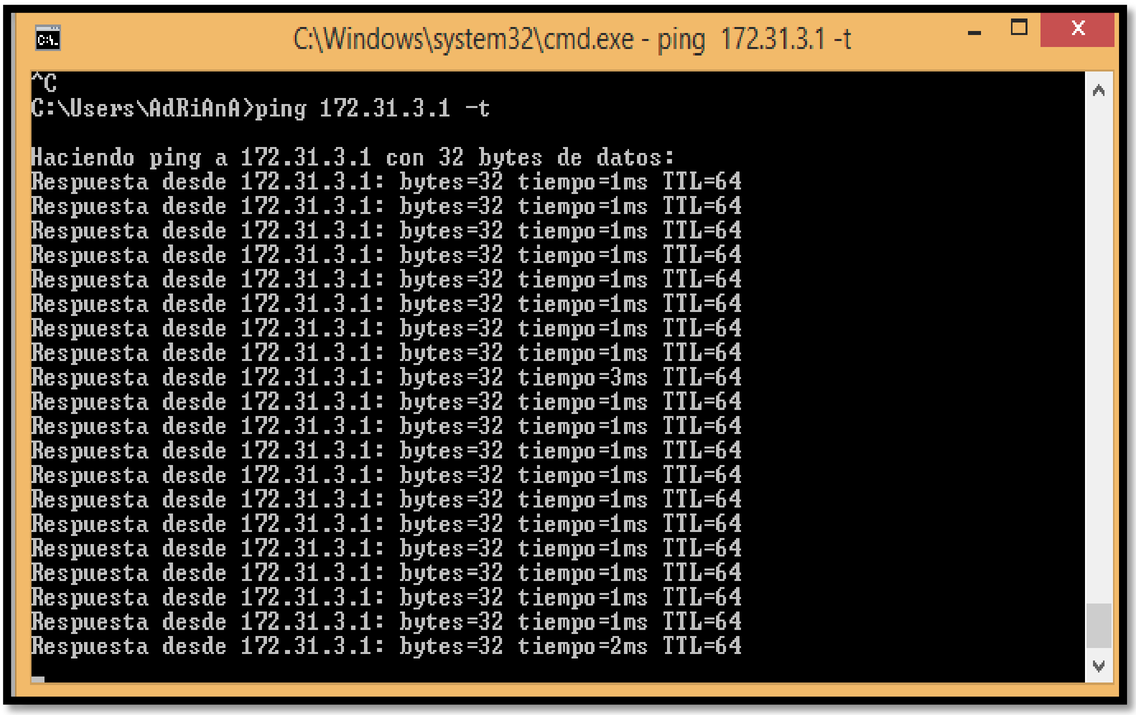


Imagen 5. 4: Prueba de conectividad al router

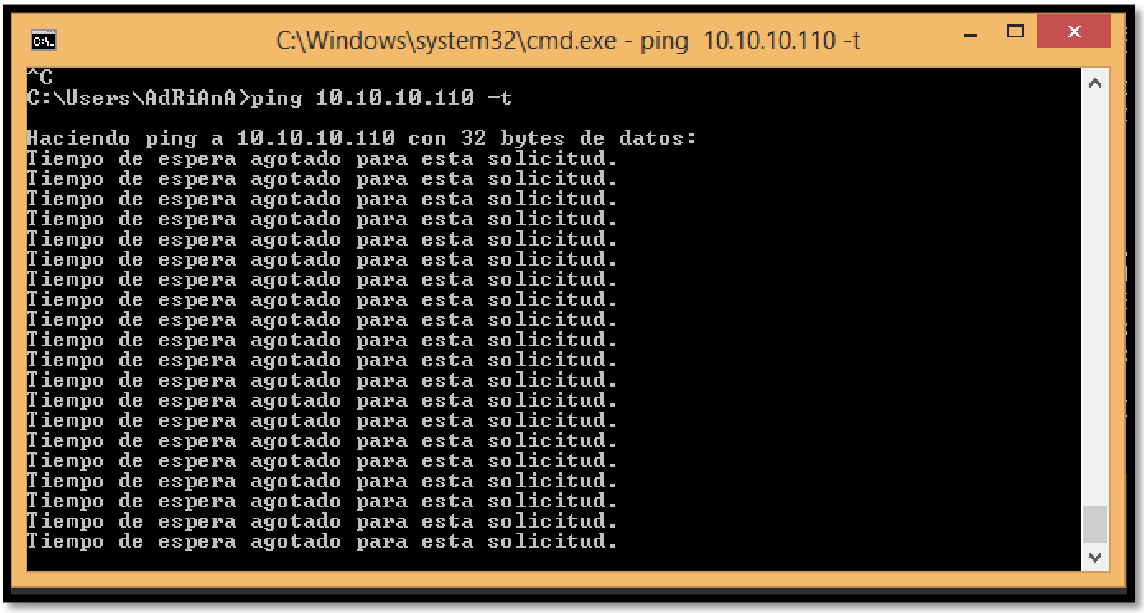


Imagen 5. 5: Prueba de conectividad al servidor de voz

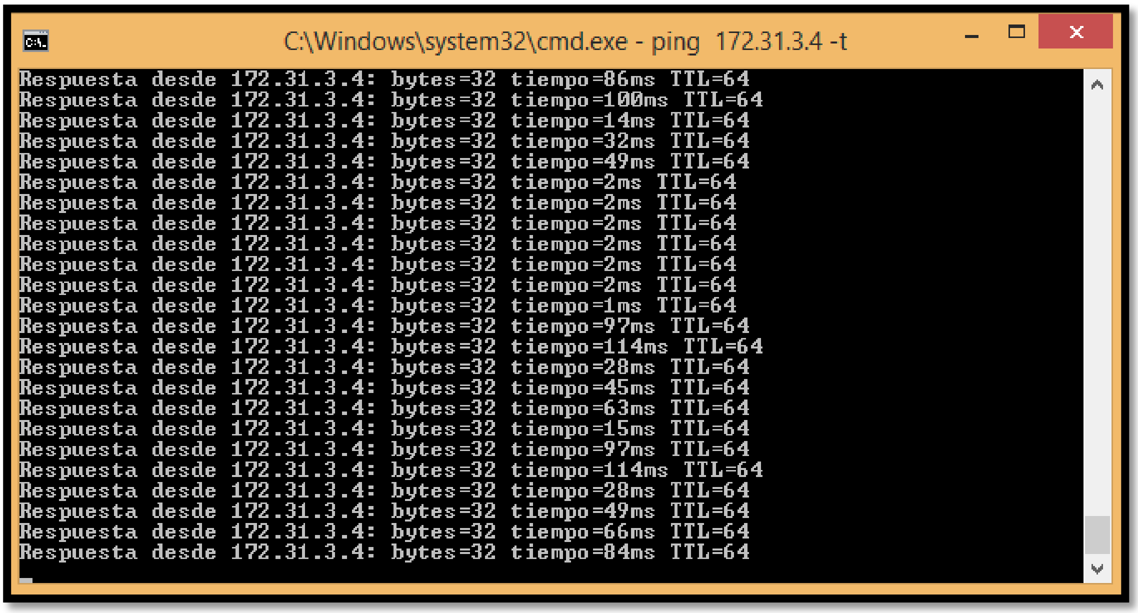


Imagen 5. 6: Prueba de conectividad a usuarios finales

5.2.2. Pruebas de Monitorización

Se realizaran estas pruebas con los módulos de OpenStack como es telemetry que nos presenta estadísticas de uso de servidores, CPU, RAM, volúmenes etc

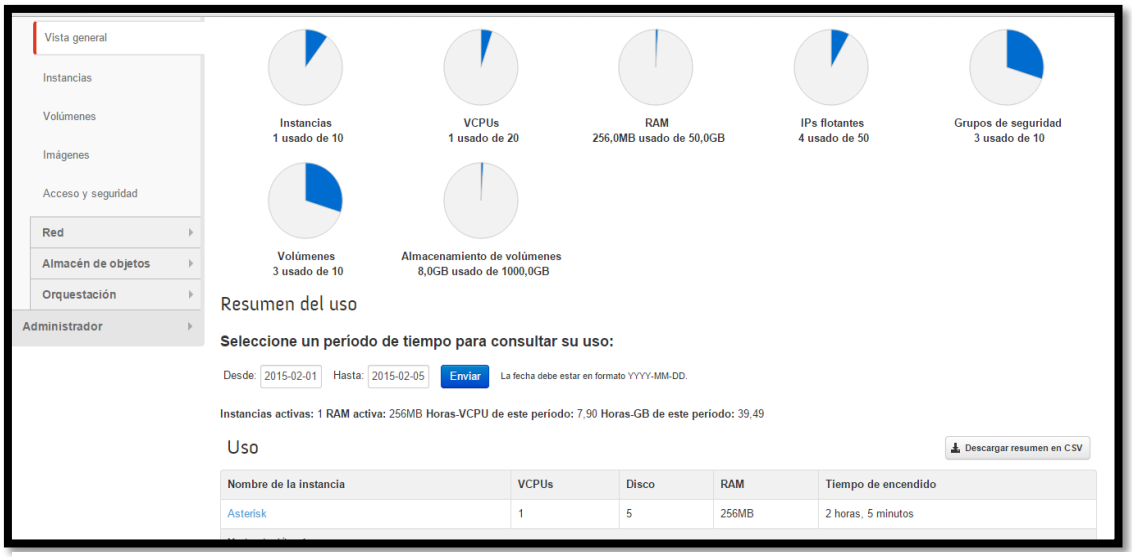


Imagen 5. 7: Monitorización de recursos de la nube

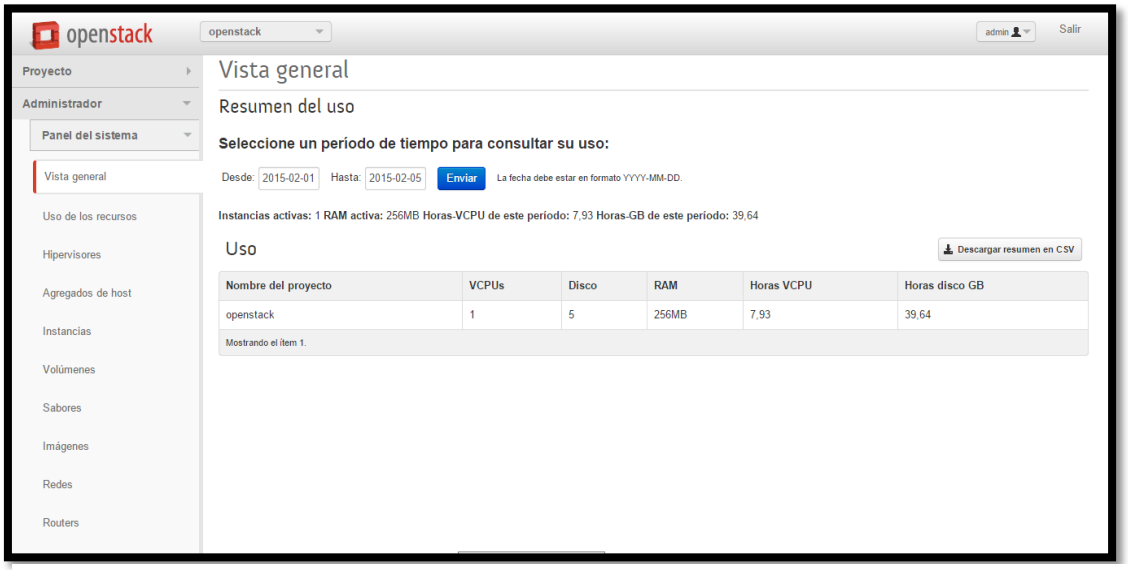


Imagen 5. 8: Generación de reporte de rendimiento

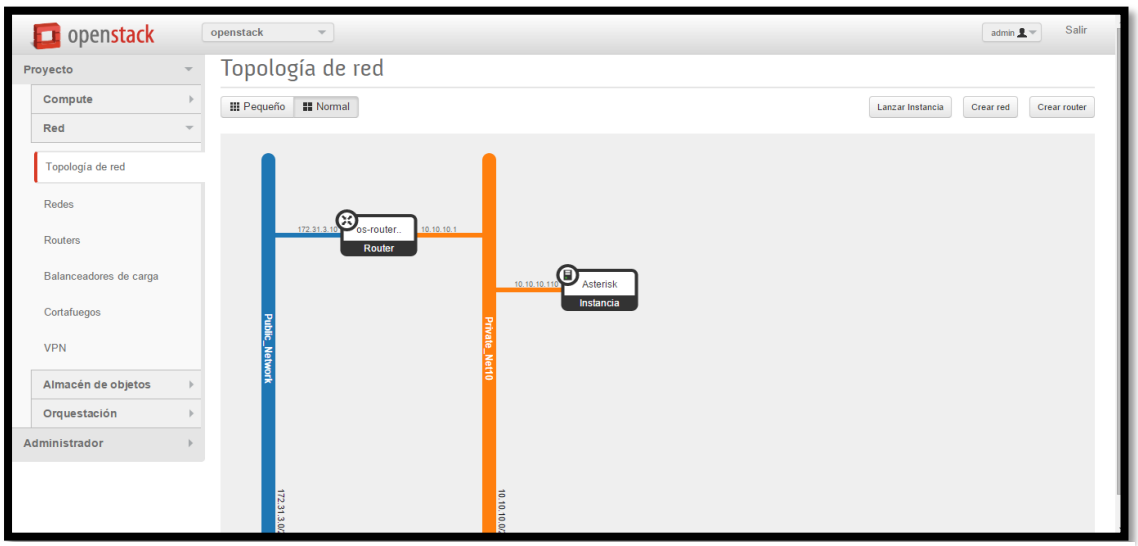
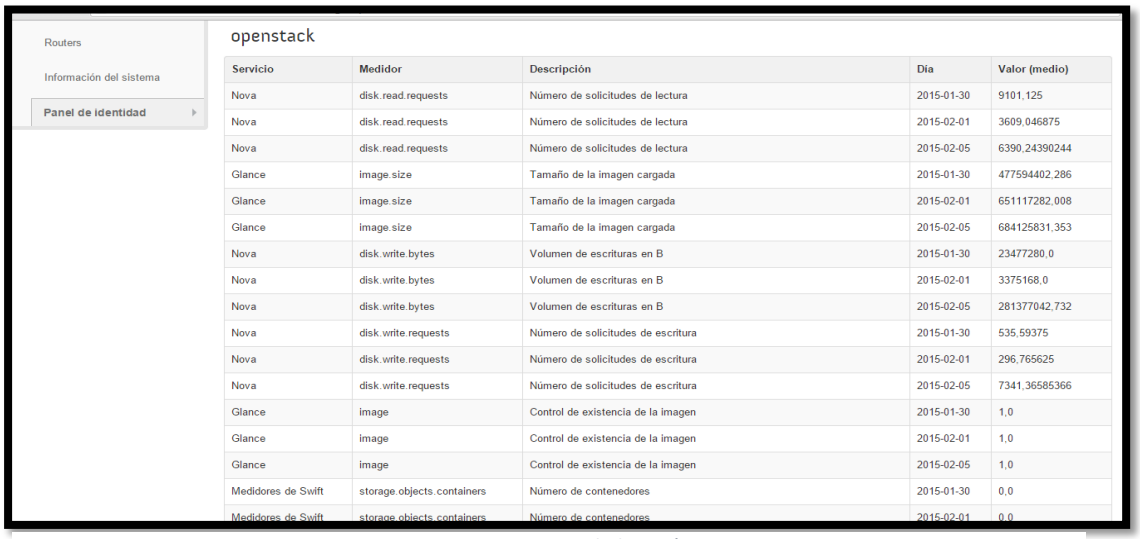


Imagen 5. 9: Topología de red de las instacias lanzadas

Imagen 5. 10: Gráfica del uso de una instancia



Servicio	Medidor	Descripción	Día	Valor (medio)
Nova	disk.read.requests	Número de solicitudes de lectura	2015-01-30	9101,125
Nova	disk.read.requests	Número de solicitudes de lectura	2015-02-01	3609,046875
Nova	disk.read.requests	Número de solicitudes de lectura	2015-02-05	6390,24390244
Glance	image.size	Tamaño de la imagen cargada	2015-01-30	477594402,286
Glance	image.size	Tamaño de la imagen cargada	2015-02-01	651117282,008
Glance	image.size	Tamaño de la imagen cargada	2015-02-05	684125831,353
Nova	disk.write.bytes	Volumen de escrituras en B	2015-01-30	23477280,0
Nova	disk.write.bytes	Volumen de escrituras en B	2015-02-01	3375168,0
Nova	disk.write.bytes	Volumen de escrituras en B	2015-02-05	281377042,732
Nova	disk.write.requests	Número de solicitudes de escritura	2015-01-30	535,59375
Nova	disk.write.requests	Número de solicitudes de escritura	2015-02-01	296,765625
Nova	disk.write.requests	Número de solicitudes de escritura	2015-02-05	7341,36585366
Glance	image	Control de existencia de la imagen	2015-01-30	1,0
Glance	image	Control de existencia de la imagen	2015-02-01	1,0
Glance	image	Control de existencia de la imagen	2015-02-05	1,0
Medidores de Swift	storage.objects.containers	Número de contenedores	2015-01-30	0,0
Medidores de Swift	storage.objects.containers	Número de contenedores	2015-02-01	0,0

Imagen 5. 11: Reporte de las métricas

Imagen 5. 12: Reporte de métricas

Imagen 5. 13: Gráfica del uso de la red de la nube

- Monitorización del servidor VoIP Asterisk instalado

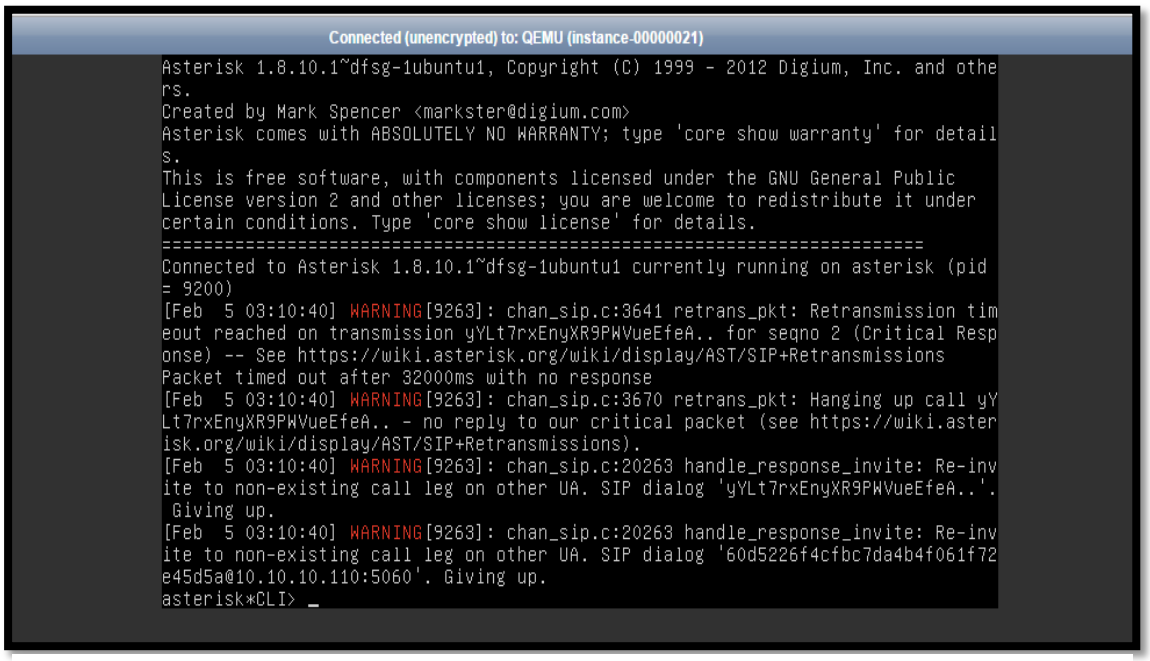


Imagen 5. 14: Errores en establecimiento de la llamada

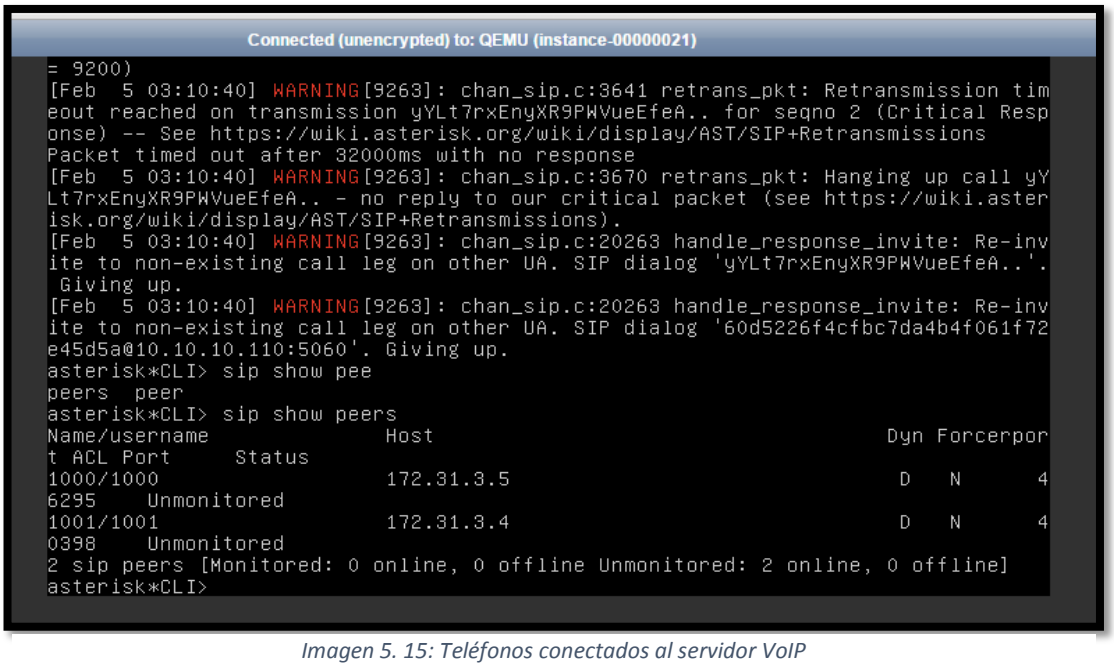


Imagen 5. 15: Teléfonos conectados al servidor VoIP

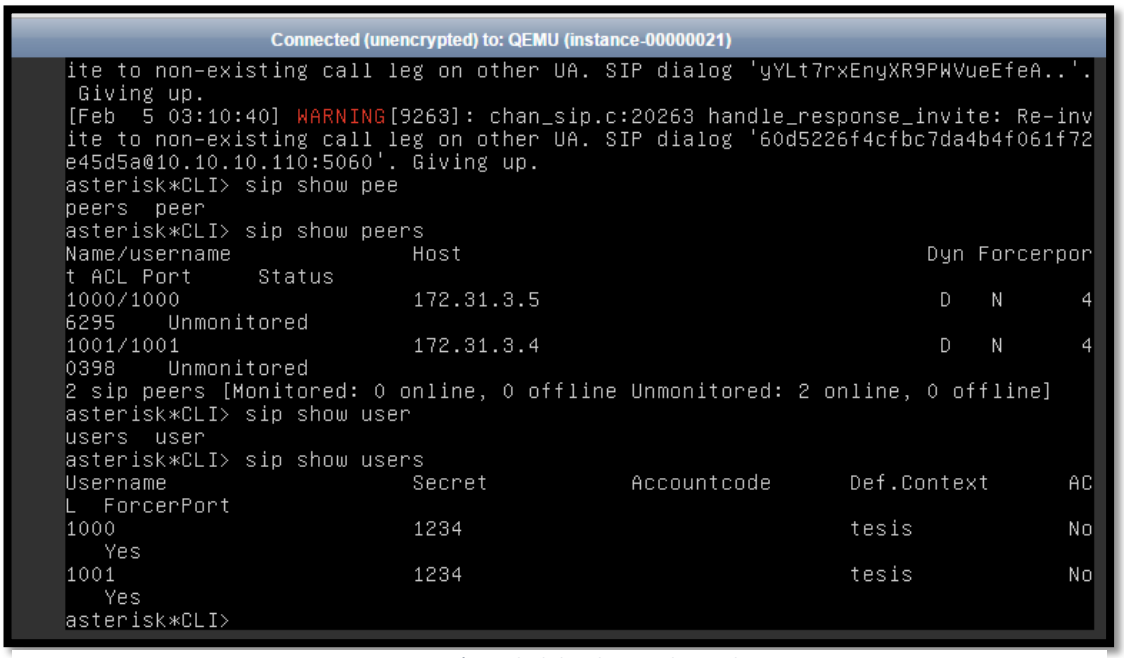


Imagen 5. 16: Líneas habilitadas en el servidor VoIP

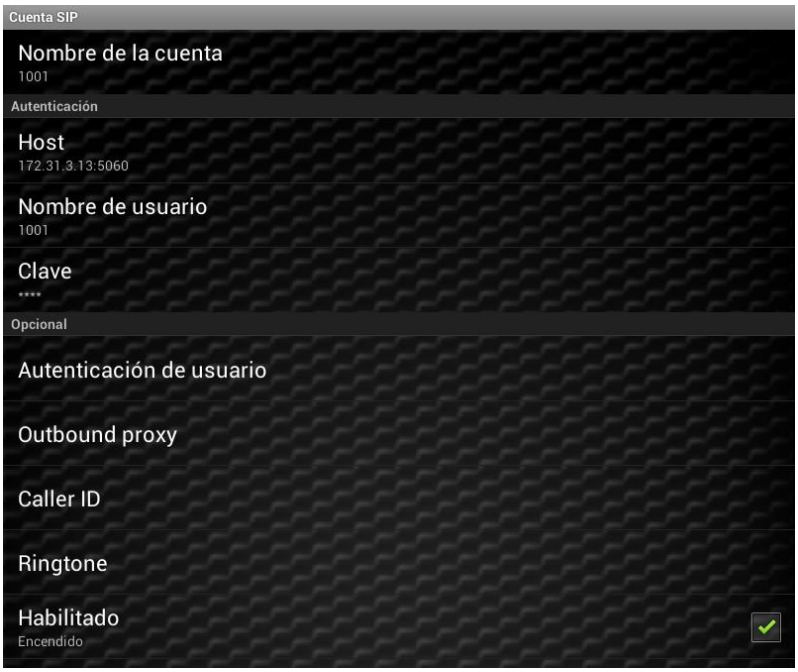


Imagen 5. 17: Log in de cuenta en el dispositívo movil

5.2.3. Pruebas de Seguridad

- Revisar los token para autenticación

  Para la generación de los tokens de autenticación de los servicios de OpenStack se genera aleatoriamente, a través del comando:

```
# openssl rand -hex 10
```

Imagen 5. 18: Generación de clave aleatória

- Revisar los token para identificación entre máquinas

  De la misma forma, para la creación de los tokens de autenticación entre máquinas se los genera aleatoriamente a través del comando explicado en la Ilustracion 62.

- Seguridad para base de datos

  La seguridad de la base de datos, se logra a través de la generación de claves seguras, que cumplan los siguientes requisitos:

  - Tener entre 8 y 16 caracteres
  - Ser fácil de recordar.
  - Contener mayúsculas, minúsculas, números y caracteres especiales.
  - Compartir esta contraseña solo con el personal autorizado.

- En caso de anotarla, no hacerlo en lugares de fácil acceso a cualquier persona

Luego de tener definida una contraseña debemos eliminar a los usuarios que se instalan por defecto, como son el usuarios anonymus.

- Seguridad para Usuarios [5] [57]
  - La capacitación es un punto importante al momento de aplicar la seguridad para accesos por parte de usuarios.
  - Las claves deben ser renovadas durante los tiempos establecidos por la organización.
  - Las contraseñas deben estar basadas en procedimientos establecidos por la organización como: mínimo de 8 caracteres, mayúsculas, caracteres alfanuméricos, números.
  - No entregar claves a usuarios desconocidos ni mantenerlas a la vista de personal que no esté autorizado.

Capítulo 6

6. Análisis de Costos del Sistema

6.1. Análisis de Costos de Servidores de Virtualización

Al no aplicar la opción de servidores de virtualización debido a los inconvenientes de compatibilidad de módulos se procedió a la implementación por hardware.

6.2. Análisis de Costos de Servidores de Cloud Computing

Se implementó un solo nodo con las características antes mencionadas su costo aproximado es de 700,00 dólares de acuerdo a las proformas detalladas a continuación.

Servidor	Router	Teléfonos IP	Smartphone	Total
$1600	$38	$150	$100	$1888

Tabla 6. 1: Costo de equipos para implementación

6.3. Análisis de Costos de Implementación de Servidores

Para la implementación se ocupó un tiempo de 5 meses las tareas se distribuyen de la siguiente manera:

Integrantes	N° de Horas Totales	Valor Hora	Total
Adriana Cornejo	200	5.00	1000.00
Christian Díaz	275	5.00	1375.00
Implementación de Software			2375.00

Tabla 6. 2: Costos de mano de obra de la implementación.

La implementación que se plantea para la empresa, incorpora teléfonos IP genéricos, lo cual reduce mucho el costo de implementación frnte a marca como Cisco, además de que esta solución permite la integración de la solución con las redes de comunicación que ya existen en la empresa sin necesidad de cambio de equipos específicos.

Nombre	Marca	Costo
Cisco SPA 303 3-Line IP Phone	**Cisco**	**$68,50**
Grandstream GXP1400 Small-Medium Business HD IP Phone	**GrandStream**	**$39**
Cisco SPA 504G 4-Line IP Phone	**Cisco**	**$97**
Grandstream GXP1100 Simple HD IP Phone for Small Business or Home Office	**GrandStream**	**$36**

Tabla 6. 3: Comparativa teléfonos VoIP

Como se puede apreciar en Tabla 6.3, los costos en equipos de usuario final disminuyen notablemente, y si esto se apica a implementaciones de gran escala, veremos el ahorro significativo que será para la empresa, y si a esto se agrega que la implementación ofrecida es completamente OenSource, el cosot de implementación se reducirá aun más frente a implementaciones licenciadas.

Frente a estos factores, podemos indicar que existe una reducción significativa en cuanto al costo de la implementación, observando las tablas se evidencia la diferencia en la inversión, para el despliegue de la solución de CloudComputing ofrecida, y aquellas que ya se encuentran en el mercado

CONCLUSIONES

En el marco de este trabajo de investigación se han llegado a las siguientes conclusiones:

- De acuerdo con los objetivos planteados inicialmente, se ha logrado el despliegue de una nube privada, para brindar el servicio de VoIP.
- La solución de Cloud Computing fue desarrollada y desplegada con OpenStack, sobre un Sistema operativo Ubuntu Server, cave recalcar que existe soporte para sistemas operativos OpenSUSE y Centos, siendo Ubuntu Server el más usado a nivel mundial.
- El servidor VoIP que se planteó inicialmente fue Elastix, el cual está basado en Asterisk, sin embargo este no era soportado por la versión de Openstack, sin embargo al estar basado en Asterisk, se pudo realizar la instalación de este, alcanzando de esta manera el objetivo planteado.
- La instalación de OpenStack realizada, es una instalación prototipo, por los recursos limitados con los que se contaba, y debido al costo de implementación de la solución a gran escala, por esto el rendimiento que la solución ofrece, puede verse disminuido, tanto en el procesamiento, almacenamiento, trabajo de memoria interna de cada una de las instancias lanzadas, así como el impedimento que se tiene al lanzar simultáneamente más de 2 instancias, que en un solución a gran escala no se darán.
- Hoy en día el despliegue de servicios virtualizados e ha convertido en la tendencia número 1 en la optimización de recursos, y en el despliegue de servicios, reduciendo así los costos de implementación del usuario final, así la administración de la infraestructura que soporta los servicios consumidos.
- Los conceptos teóricos sobre la instalación deben estar muy claros y se debe realizar un planteamiento adecuado de la infraestructura que se va a usar e la instalación, siendo indispensable la correcta planificación de los componentes que contendrá la solución, para así corregir oportunamente los errores que se puedan ir generando dentro de la simulación.
- Los componentes de red de OpenStack corresponde a un nuevo paradigma de red por los que se debe tener una visión puntual y concisa de los mismos, de esta forma podemos realizar las conexiones desde y hacia las instancias.
- OpenStack utiliza una dualización de red por lo que podemos contar con redes internas y externas siendo las instancias internas las que administran las conexiones entre nodos y la externa no permite tener conexión con los clientes o redes que no pertenezcan a las instancias de OpenStack.

- La telefonía ip es una herramienta útil para las empresas que permiten el ahorro de costos, más aun si se encuentran en la nube, lo que garantiza la comunicación entre los empleados y sus respectivas labores.

- La implementación de la telefonía IP con servicio en la nube si es posible en un entorno de nube híbrida lo que nos permitió cumplir con nuestros objetivos, y al investigar en un ambiente de cloud y un servicio como VoIP podemos concluir que las redes hibridas son el futuro de los Datacenters y crecimiento de centros de datos.

RECOMENDACIONES

- Para futuras implementaciones de Openstack, como solución para una nube privada se recomienda el despliegue en servidores físicos debido a que la virtualización no es una solución viable, pues presenta problemas al momento de levantar los servicios de neutrón y cinder, dando problemas de compatibilidad en los plugin usados por estos.

- Openstack puede ser desplegado sobre algunos Sistemas Operativos Linux como Centos, Ubuntu y OpenSUSE, sin embargo es altamente recomendable realizar el despliegue sobre Ubuntu, debido a la gran cantidad de soporte que existe de parte de la comunidad.
- Al realizar una implementación de un servidor de VoIP opensource como Asterisk, reducimos costos, y al ser altamente configurable se puede dar solución a los problemas que se vayan presentando durante su uso.
- OpenStack ofrece varios servicios adicionales los cuales nos permitirán tener un mejor desempeño de nuestra nube, beneficiando así a los usuarios.
- Durante el despliegue de Asterisk pueden surgir problemas en la calidad del servicio, ya sea en las llamadas o en la conexión de los usuarios, se debe primero verificar la calidad de la conexión entre el servidor y los cientes, luego se debe mejorar la calidad de los paquetes para que no existan perdida de ls mismo, así garantizando que la llamada podrá ser realizada.
- Se recomienda revisar el encapsulamiento de los paquetes, para darnos cuenta de si están saliendo por donde deben, en caso de no hacerlo se deberá agregar las líneas de comando que indiquen cual es la dirección IP externa del servidor, así como indicar cuál es la red interna, es decir la red que comunica a los nodos internamente, así asegurando que las rutas de entrada y salida sean las correctas.
- Se debe habilitar el comando QUALIFY, permitiendo así visualizar la calidad de la transmisión, y junto a TOS=0X04, permitirán tener una tasa de error muy baja, realizando así la comunicación y el establecimiento de la llamada.
- Se recomienda obtener claves seguras, y ser muy cauteloso con las mismas, para evitar futuros fraudes o intrusiones al servidor.

GLOSARIO DE TÉRMINOS

- **IEEE:** Institute of Electrical and Electronics Engineers (Instituto de Ingenieros Eléctricos y Electrónicos)
- **NIST:** National Institute of Standards and Technology (Instituto Nacional de Estándares y Tecnologías de los Estados Unidos)
- **HP:** Hewlett Packard, Empresa líder en ventas de equipos computacionales y mantenimiento http://www8.hp.com/ec/es/home.html
- **IBM:** International Business Machines Corp, es una empresa multinacional estadounidense de tecnología y consultoría. http://www.ibm.com/
- **API:** Application Programming Interface o Interfaz de Programación de Aplicaciones son las funciones de una biblioteca sobre otro software.
- **ASP:** Application Service Provider o Proveedor de Servicios de Aplicación, es una empresa que ofrece servicios a través de internet.
- **ERP:** Enterprise Resource Planning o Sistemas de Planificación de Recursos Empresariales permite la administración de recursos en una organización.
- **CRM:** Customer Relationship Management o Administración Basada en la Relación con los Clientes permite la administración de las actividades de los clientes
- **.NET, Java, Python, PHP, Ajax, Ruby onRails:** Lenguajes de Programación para la creación de servicios provistos por la nube.
- **VPN:** Virtual Private Network o Red Privada Virtual ayuda a las organizaciones a ampliar la conectividad de forma segura y económica y a mejorar la velocidad.
- **VPS:** Virtual Private Server o Servidor Privado Virtual es un método de particionar un servidor físico en varios servidores
- **Cloudstack:** Software para crear una Infraestructura de Cloud Computing.
- **Asterisk:** Es un Servidor de Voz sobre IP que permite la creación, administración y control del sistema de voz.
- **ISP:** Internet Service Provider o Proveedor de Servicios de Internet provee de conexión a internet a las organizaciones.
- **Vocaltec:** Empresa líder en tecnología de voz sobre IP.
- **VON:** Congreso y Feria para la industria de voz sobre IP. http://www.voip-info.org/wiki/view/VON
- **IETF:** Internet Engineering Task Force o Grupo de Trabajo de Ingeniería de Internet, es una organización de normalización de diversas áreas tecnológicas.
- **UIT:** Unión Internacional de Telecomunicaciones, es el organismo especializado de las Naciones Unidas para las tecnologías de la información y la comunicación
- **OSI:** Open System Interconnection o Modelo de Interconexión de Sistemas Abiertos, es un estándar de referencia de interconexión de red.
- **H.323:** Protocolo empleado en VoIP para la comunicación
- **SIP:** Session Initiation Protocol o Protocolo de Inicio de Sesión para voz sobre IP
- **RTP:** Real Time Protocol o Protocolo en Tiempo Real

- **IAX:** Inter-Asterisk eXchange protocol es uno de los protocolos utilizado por Asterisk
- **TCP/IP:** Es un modelo de Referencia de Protocolos de Red para la comunicación.
- **UDP/IP** :User Datagram Protocol, es un protocolo de transporte no orientado a la conexión sobre la red IP
- **IAX2:** Segunda versión del Protocolo IAX con funcionalidades adicionales.
- **NAT:** Network Access Translation o Traducción de nombres de Red, su función es permitir el intercambio de paquetes entre dos redes.
- **MD5:** Message-Digest Algorithm 5 o Algoritmo de Resumen del Mensaje 5, es un algoritmo de encriptación de 128 bits.
- **QoS:** Quality of Service o Calidad de Servicio, es un factor que garantiza el nivel de servicio de una red de telefonía o de computadores.
- **BER:** Bit Error Ratio o Tasa de Error Binario se refiere a la cantidad de bits recibidos erróneamente.
- **WAN:** Wide Area Network o Red de Area Extensa abarca redes amplias e gran distancia como por ejemplo un país.
- **ADSL:** Asymmetric Digital Subscriber Line o Línea de Abonado Digital Asimétrica tecnología de transmisión analógica de datos digitales.
- **SDH:** Synchronous Digital Hierarchy o Jerarquía Digital Síncrona o Tecnología de transmisión de alta capacidad.
- **DWDM:** Dense Wavelength Division Multiplexing multiplexado compacto por división en longitudes de onda es una técnica de tranmision de alta velocidad mediante fibra óptica.
- **DMZ:** Demilitarized zone o zona desmilitarizada, es una red local intermedia que está entre la red interna y la red externa.
- **RTC:** Red Telefónica Conmutada son todos los elementos que enlazan una red mediante un circuito físico.
- **NIC:** Network Interface Card o Tarjeta de Interfaz de Red es un dispositivo que permite la comunicación entre máquinas interconectadas.
- **TI:** Tecnología Informática, es un término empleado para agrupar a las tecnologías de comunicaciones
- **VDI:** Virtual Drive Interface o Interfaz de Unidad Virtual permite gestionar a través de un escritorio las máquinas virtuales creadas por el cliente
- **VM:** Virtual Machine son las abreviaturas de Máquina Virtual en español.
- **Hipervisor:** es una plataforma para desplegar los sistemas operativos.
- **RAM:** Random Access Memory o Memoria de Acceso Aleatorio, es un dispositivo para almacenamiento en donde se cargan las instrucciones para el procesador.
- **256-bits AES SSL:** 256 bits Advanced Encryption Standard o Estándar de Encriptación Avanzada y SSL Socket Secure Layer o Capa de Conexión Segura, técnica que sirve para el aseguramiento de la red.

BIBLIOGRAFÍA

[1] B. CHEE y C. FRANKLIN, Cloud Computing Technologies and Strategies of the Ubiquitous Data Center, New York: CRC Press, 2010.

[2] P. MELL y T. GRANCE, The NIST Definition of Cloud Computing, Gaithersburg: NIST (National Indtitute Standards and Technology, 2011.

[3] T. ERL, Z. MAHMOOD y R. PUTTINI, Cloud Computing Concepts, Technology & Architecture, Massachusetts: Prentice Hall, 2014.

[4] A. URUEÑA, A. FERRARIE, D. BLANCO y E. VALDESACA, «Cloud Computing Retos y Oportunidades,» Observatorio Nacional de Telecomunicaciones y de la SI, Madrid, 2012.

[5] D. ROUNTREE y I. CASTRILLO, The Basics of Cloud Computing, Massachusetts: Syngress, 2013.

[6] D. NÚÑEZ, *Implementación de un Prototpo de Software como Servicio (SAAS) para pequeñas y medianas empresas,* Quito: Escuela Politécnica Nacional, 2013.

[7] E. QUELIN, J. NAVARRO y R. RODRIGUES, «Cloud Computing,» Universidad Nacional Federico Villareal, Lima, 2011.

[8] L. JOYANES, «Computación en la Nube,» *REVISTA DEL INSTITUTO ESPAÑOL DE ESTUDIOS ESTRATÉGICOS,* vol. 1, nº 00, pp. 87-110, 2012.

[9] M. BUENAVENTURA, *Desarrollo de una Nube Estudiantil "Cloud-IT" Libre y Portable,* Guayaquil: Universidad Internacional del Ecuador, 2013.

[10] E. MENA, A. GUERRERO y I. BERNAL, «Implementación de un prototipo de Cloud Computing de modelo privado para ofrecer Infraestructura como Servicio (IaaS),» Escuela Politécnnica Nacional, Quito, 2013.

[11] E. CABRERA, *Esrudio para Implementación de Servicios de Data Center basados en el Modelo Cloud Computing,* Cuenca: Universidad de Cuenca, 2013.

[12] Amazon Web Servces, «Amazon Web Services (AWS),» Amazon, [En línea]. Available: http://aws.amazon.com/es/. [Último acceso: 24 Diciembre 2014].

[13] Nexica, «Nexica Critical Cloud & Hosting,» www.nexica.com, [En línea]. Available: http://www.nexica.com/es/modelos-cloud. [Último acceso: 24 Diciembre 2014].

[14] N. MATIZ, *Implementación de Cloud Security en un sistema basado en XEN Cloud Plataform,* Sangolquí: Escuela Politécnica del Ejército, 2013.

[15] G. ULLAURI, *Servicio de Virtualización de infraestructura tecnológica basado en Clod Computing,* Guayaquil: Universidad Politécnica Salesiana, 2013.

[16] M. REVELO, C. ROMERO y R. GORDILLO, «Diseño e Implementación de una Red ede servicios basados en los conceptos de Cloud Computing,» Escuela Politécnica del Ejército, Sangolquí, 2013.

[17] H. HINOJOSA y M. ULLOA, *Diseño e Implementación de un Sistema de Nube Computacional basados en la Plataforma de codigo abierto Openstack,* Sangolquí: ESPE Universidad de las Fuerzas Armadas, 2014.

[18] C. PUGA, *Servicio de Call Center e Interconexión de la Central del Diario Independiente y sus cuatro sucursales con la utilización de Software Libre,* Ibarra: Universidad Técnica del Norte, 2013.

[19] A. LAVARIEGA, *Diseño y Desarrollo de un softphone para Telefonía IP utilizado el Protocolo IAX,* Huajuapan de León : Universidad Tencológica de la Mixteca, 2007.

[20] M. CUEVA, *Diseño de una Red de Voz sobre IP para el Hospital Provincial General "Isidro Ayora Loja" con calidad de Servicios en un Ambiente Open Source,* Loja: Universidad de Cuenca, 2010.

[21] L. BLACIO, *Diseño de una Red para Voz sobre IP en la Nube y posible implementación con HTML5,* Guayaquil: Universidad Politécnica Superior del Litoral, 2013.

[22] O. BAQUE, *Diseño y Desarrollo de un prototipo para el levantamiento de Servicios sobre IP para la red de Fibra Optica de la ULEAM,* Manabí: Universidad Laica Eloy Alfaro de Manabí, 2009.

[23] F. SAAVEDRA, *Diseño e Implementación de un enlace telefónico VoIP H323 entre dos sitios remotos utilizando servidores de comunicación,* Quito: Universidad Tecnológica América, 2011.

[24] D. AMÁN y V. ARDILA, *Análisis y Diseño de una Red de telefonía IP para la escuela Héroes del Cenepa de la ESPE,* Sangolquí: Universidad Politécnica del Ejército, 2012.

[25] Elastix, «Protocolo IAX,» WordPress, [En línea]. Available: http://elastixtech.com/protocolo-iax/. [Último acceso: 28 Diciembre 2014].

[26] Citrix.com, «XenServer-Información Técnica,» Citrix.com, 2014. [En línea]. Available: http://lac.citrix.com/products/xenserver/tech-info.html. [Último acceso: 28 Diciembre 2014].

[27] Citrix Technical Support, «Citrix Support,» Citrix Systems, Junio 2013. [En línea]. Available: http://support.citrix.com/article/CTX137826. [Último acceso: 28 Diciembre 2014].

[28] Citrix Technical Suport, «FAQ XenMotion, Live Migration,» Citrix Systems, 26 Marzo 2014. [En línea]. Available: http://support.citrix.com/article/CTX115813. [Último acceso: 29 Diciembre 2014].

[29] info@TxMQ.com, «WebSphere Cast Iron Hypervisor Delivers Xen Support,» WordPress, 30 Julio 2014. [En línea]. Available: http://www.txmq.com/websphere-cast-iron-hypervisor-delivers-xen-support/. [Último acceso: 28 Diciembre 2014].

[30] http://www.miniacademia.es/author/ebhum/, «Miniacademia,» WordPress, 10 Marzo 2014. [En línea]. Available: http://www.miniacademia.es/manual-citrix-xenserver/. [Último acceso: 29 Diciembre 2014].

[31] G. AHMED, Implementing Citrix Xen Server Quickstarter, Birmingham: Packt Publishing, 2013.

[32] Citrix Technical Support, «XenServer Technical FAQ,» Citrix Systems, 11 Febrero 2010. [En línea]. Available: http://support.citrix.com/article/CTX123996. [Último acceso: 29 Diciembre 2014].

[33] H. HERRERO , «Instalación, configuración y administración de Citrix XenServer,» www.bujarra.com, [En línea]. Available: http://www.bujarra.com/ProcedimientoCitrixXenServer.html#instalar. [Último acceso: 29 Diciembre 2014].

[34] C. ALVAREZ, M. IBAÑEZ, A. MOLINA , J. MORENO, J. MUÑOZ, J. PEDRAJAS, C. REINALDOS y A. ROCA, «Administración de OpenStack Essex: Instalación, configuración y explotación,» Gobierno de España, Murcia, 2012.

[35] J. ATUL, J. D, K. MURARI, R. MURTHY, C. VIVEK y G. YOGESH, «OpenStack Beginner's Guide,» CSS Corp, 2012.

[36] Openstack.org, «OpenStack Installation Guide for Ubuntu 14.04,» Openstack Fundation, 2014.

[37] A. MOLINA y J. MUÑOZ, «Componentes de Openstack,» Openstack, 2013. [En línea]. Available: http://iesgn.github.io/cloud/curso/u5/componentes_openstack.html#/. [Último acceso: 29 Diciembre 2014].

[38] OpenStack, «OpenStack Cloud Administrator Guide,» OpenStack Fundation, 2014.

[39] VMWARE, «VMWARE,» VMWARE, 2014. [En línea]. Available: http://www.vmware.com/products/vcenter-server. [Último acceso: 29 Diciembre 2014].

[40] VMWARE, «VMWare Presentation,» VMWARE, 2014. [En línea]. Available: http://www.vmware.com/latam/company/news/releases/20111409-vmw-workstation8#sthash.dFQUrAGG.dpuf. [Último acceso: 29 Diciembre 2014].

[41] Oracle Virtual Box, «VirtualBox,» Oracle VirtualBox, 2012. [En línea]. Available: https://www.virtualbox.org/. [Último acceso: 29 Diciembre 2014].

[42] F. PERIAÑEZ, «Características de Virtual Box,» I.E.S, 2013. [En línea]. Available: http://fpg.hol.es/VirtualBox/caractersticas_de_virtualbox.html. [Último acceso: 29 Diciembre 2014].

[43] Quarea, «quarea.com,» quarea, 31 enero 2015. [En línea]. Available: http://www.quarea.com/sites/quarea.com/files/files/imce/AsteriskA4-e.pdf. [Último acceso: 03 febrero 2015].

[44] Cognizant reports , *Taking Testing to the Cloud,* New Jersey: Cognizant , 2013.

[45] F. GARCIA, «Una Perspectiva de las Pruebas de Software en la Nube,» 2013.

[46] iesmachado.org, «Metodología PDCA,» 10 Marzo 2012. [En línea]. Available: http://iesmachado.org/web%20insti/depart/electr/apuntes/files/sti/diurno/curso1/calidad/METODOLOGIA_PDCA.pdf. [Último acceso: 30 Diciembre 2014].

[47] Cisco Systems, «Practica de Laboratorio,» 16 Noviembre 2012. [En línea]. Available: http://webcache.googleusercontent.com/search?q=cache:http://cisco.monagas.udo.edu.ve/curriculum/RyS_mod1/course/files/8.3.2.7%2520Lab%2520-%2520Testing%2520Network%2520Connectivity%2520with%2520Ping%2520and%2520Traceroute.pdf. [Último acceso: 30 Diciembre 2014].

[48] . I. BELMONTE, «Unix, tecnología y algunas cosas sobre mí,» 18 Septiembre 2006. [En línea]. Available: http://ivanhq.net/2006/09/18/asterisk-y-el-maldito-sip-nat/. [Último acceso: 03 Marzo 2015].

[49] E. GARCIA, «Cloud computing-Marcelo GC-IBP-Tarapoto,» 16 09 2014. [En línea]. Available: http://cloudcomputingblaisepascal.blogspot.com/. [Último acceso: 23 12 2014].

[50] Soluciones de Bolsilo S.A., «Soluciones de Bolsillo,» Soluciones de Bolsillo, 2000. [En línea]. Available: http://www.solucionesdebolsillo.com/cloud-computing/. [Último acceso: 24 Diciembre 2014].

[51] J. SANCHEZ, «blogjordisanchez,» wordpress, 23 Noviembre 2011. [En línea]. Available: https://blogjordisanchez.wordpress.com/2011/11/23/tipos-de-nube/. [Último acceso: 24 Diciembre 2014].

[52] R. González, «Dollibarr,» Dolibarr España-Latinoamérica, [En línea]. Available: http://www.dolibarr.es/index.php/doliCloud. [Último acceso: 24 Diciembre 2014].

[53] Google , «Google Cloud Plataform,» Gloogle, [En línea]. Available: https://cloud.google.com/appengine/docs. [Último acceso: 25 Diciembre 2014].

[54] Openstack, «Openstack,» Openstack, [En línea]. Available: http://www.openstack.org/. [Último acceso: 25 Diciembre 2014].

[55] Nisk inc, «Nisk inc,» [En línea]. Available: http://www.nskinc.com/wp-content/uploads/2010/03/cloud-diagrams-private-cloud.png. [Último acceso: 25 Diciembre 2014].

[56] Nsk inc, «Nsk inc,» Nsk inc, [En línea]. Available: http://www.nskinc.com/wp-content/uploads/2010/11/cloud-diagrams.png. [Último acceso: 25 Diciembre 2014].

[57] Nsk inc, «Nsk inc,» [En línea]. Available: http://www.nskinc.com/wp-content/uploads/2010/11/cloud-diagrams-hybrid.png. [Último acceso: 25 Diciembre 2014].

[58] C. ZOU, H. DENG y Q. QIU, «Design and Implementation of Hybrid Cloud Computing Architecture Based on Cloud Bus,» de *2013 IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks*, Guangzhou, 2013.

[59] J. INOSTROZA y N. INZUNZA, *Evaluación Técnico-Económica de servicios de Cloud Computing para su Implementación en PYMES,* Valparaíso: Universidad Tecnológica de Chile, 2010.

[60] H. DWIVEDI, *Hacking VoIP,* San Francisco: No Starch Press, 2009.

[61] Blogger Team, «Sistemas Operativos para Red,» 23 Febrero 2004. [En línea]. Available: http://sosred.blogspot.com/2010/04/virtualizadores.html. [Último acceso: 27 Diciembre 2014].

[62] Openstack , «Openstack,» Openstack Fundation, Diciembre 2014. [En línea]. Available: http://www.openstack.org/software/. [Último acceso: 29 Diciembre 2014].

[63] knowledge sharing team, «Future of Cloud Testing,» knowledge sharing team, 10 Mayo 2013. [En línea]. Available: http://www.thecloudcomputingaustralia.com/2013/05/10/future-of-Cloud-testing-testing-as-a-service-taas/. [Último acceso: 29 Diciembre 2014].

[64] B. ERNST, «Top 3 Trends in Mobile and Software QA Testing,» MyCroud, 1 Agosto 2014. [En línea]. Available: https://mycrowd.com/2014-mobile-software-qa-testing-update-trends/. [Último acceso: 30 Diciembre 2014].

ANEXOS

Anexo1

Instalación XEN

1. Boot de Xen

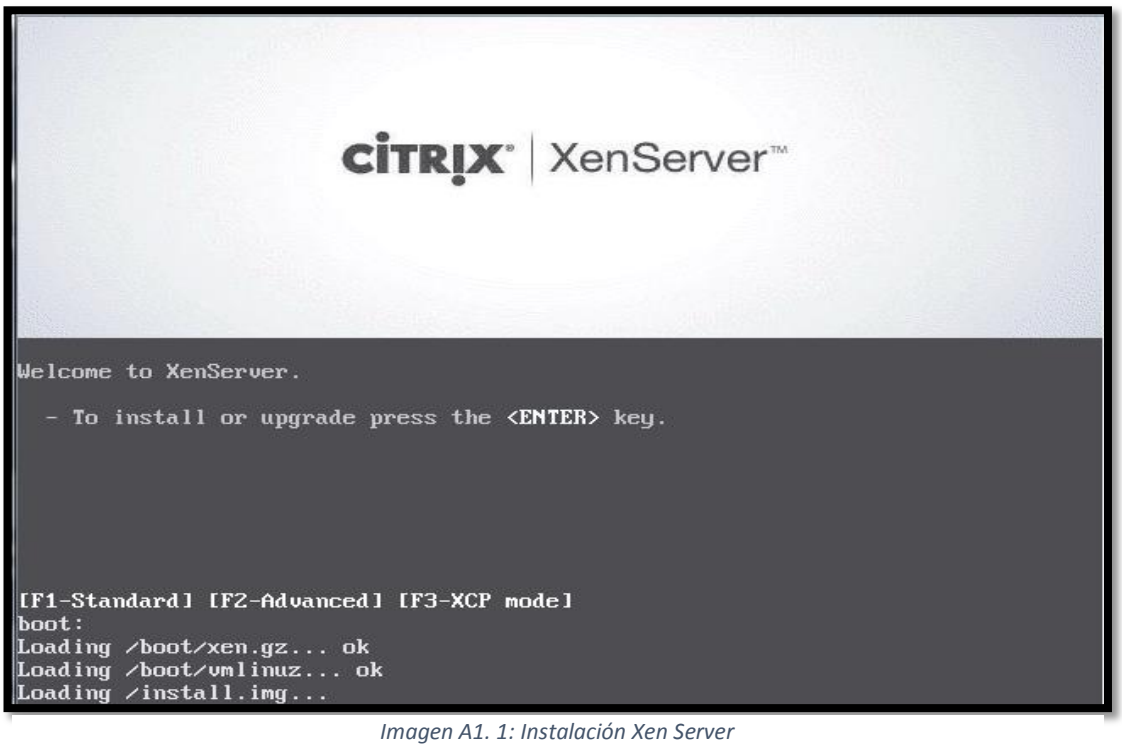


Imagen A1. 1: Instalación Xen Server

2. **Elegir el Teclado**

Imagen A1. 2: Distribución teclado

3. Continuar con la instalación

Imagen A1. 3: Indicaciones básicas

4. Aceptar acuerdo y continuar

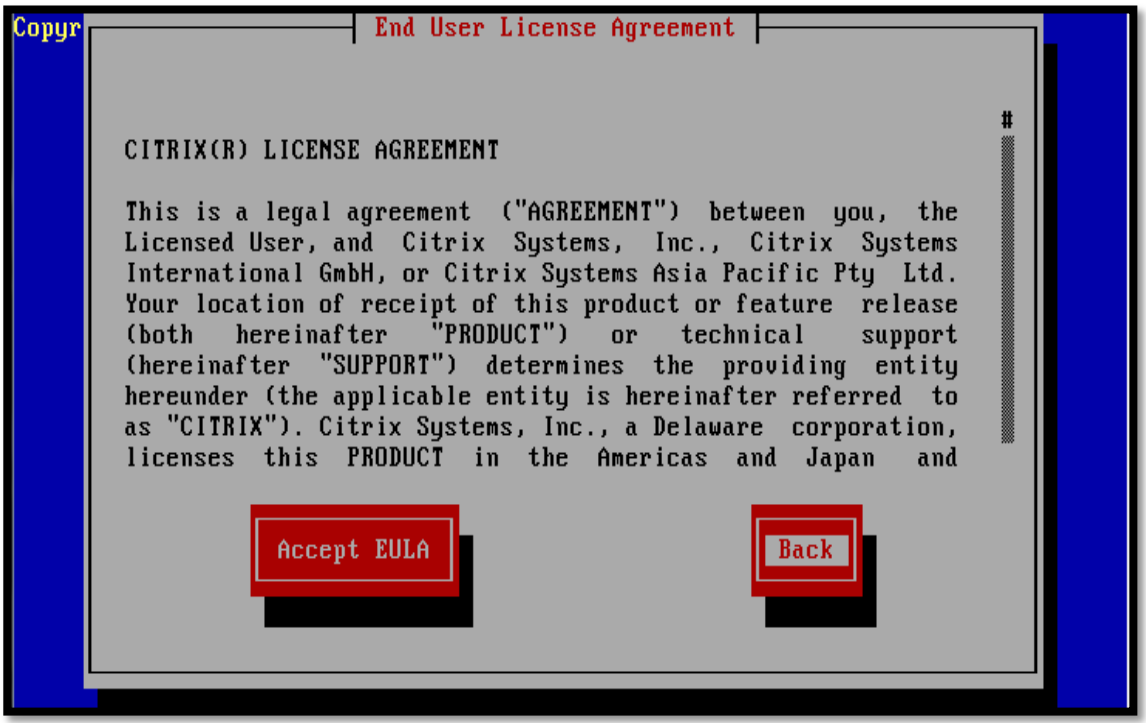


Imagen A1. 4: Contrato de servicios

5. **Elegir el disco donde será instalado**

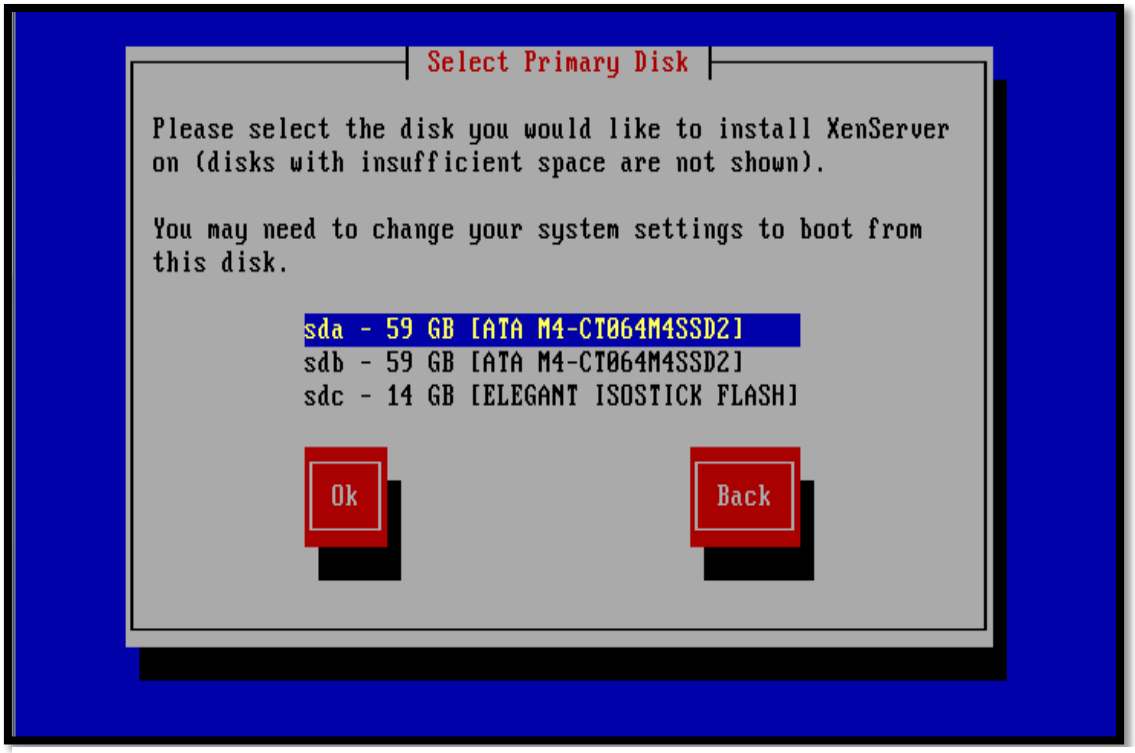


Imagen A1. 5: Elección almacenamiento

6. **Elegir almacenamiento de máquinas virtuales**

Imagen A1. 6: Elección Storage para virtualización

7. Elegir fuente de instalación

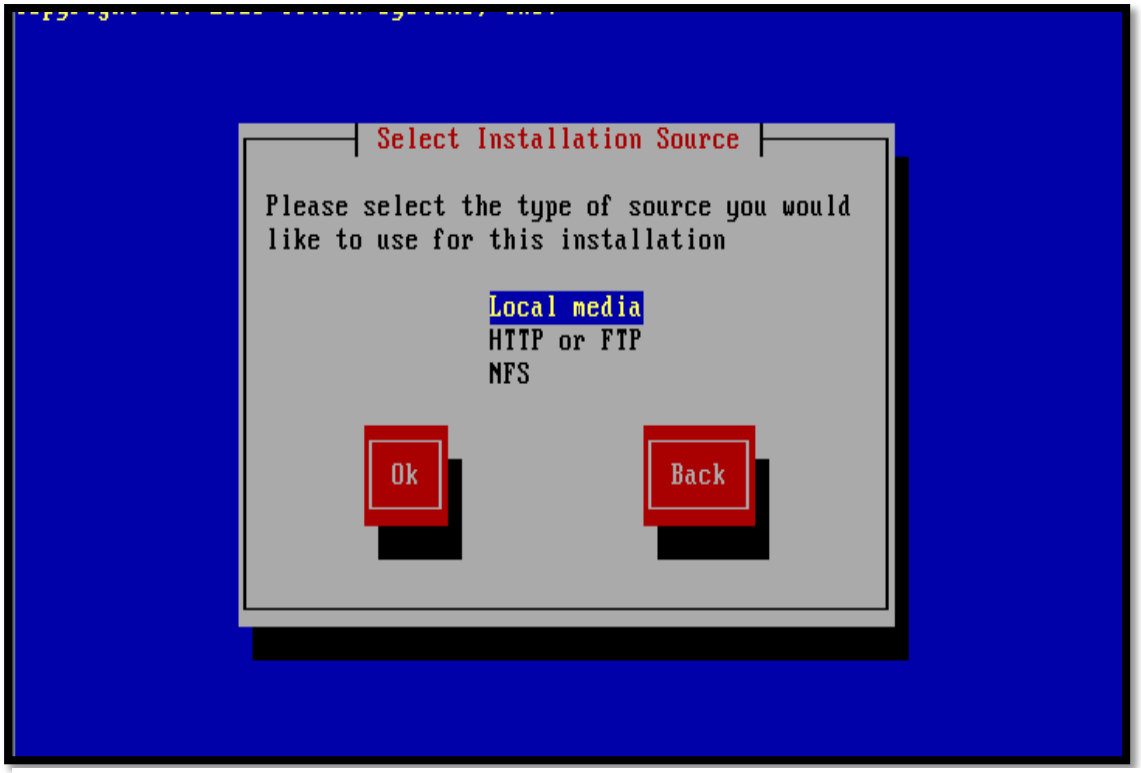


Imagen A1. 7: Descarga de Fuentes externas

8. Instalar Paquetes Adicionales

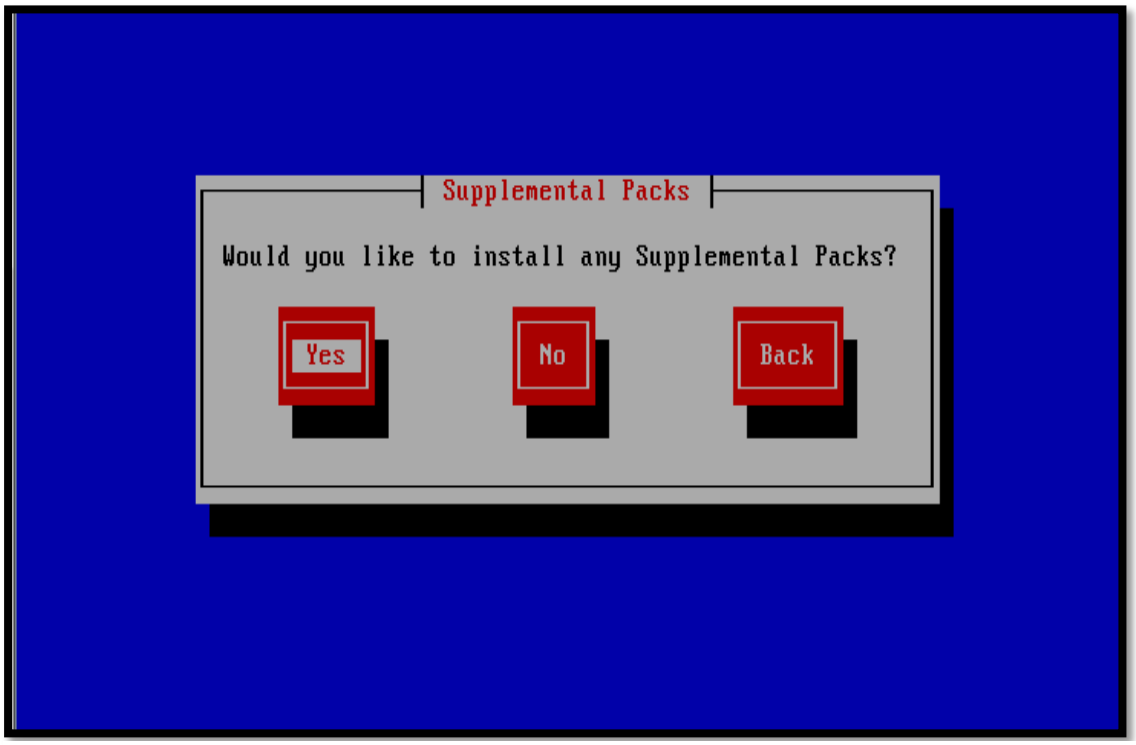


Imagen A1. 8: Confirmación de paquetes adicionales

9. Verificar medio de instalación

Imagen A1. 9: Verificación de instalación

10. Escribir contraseña

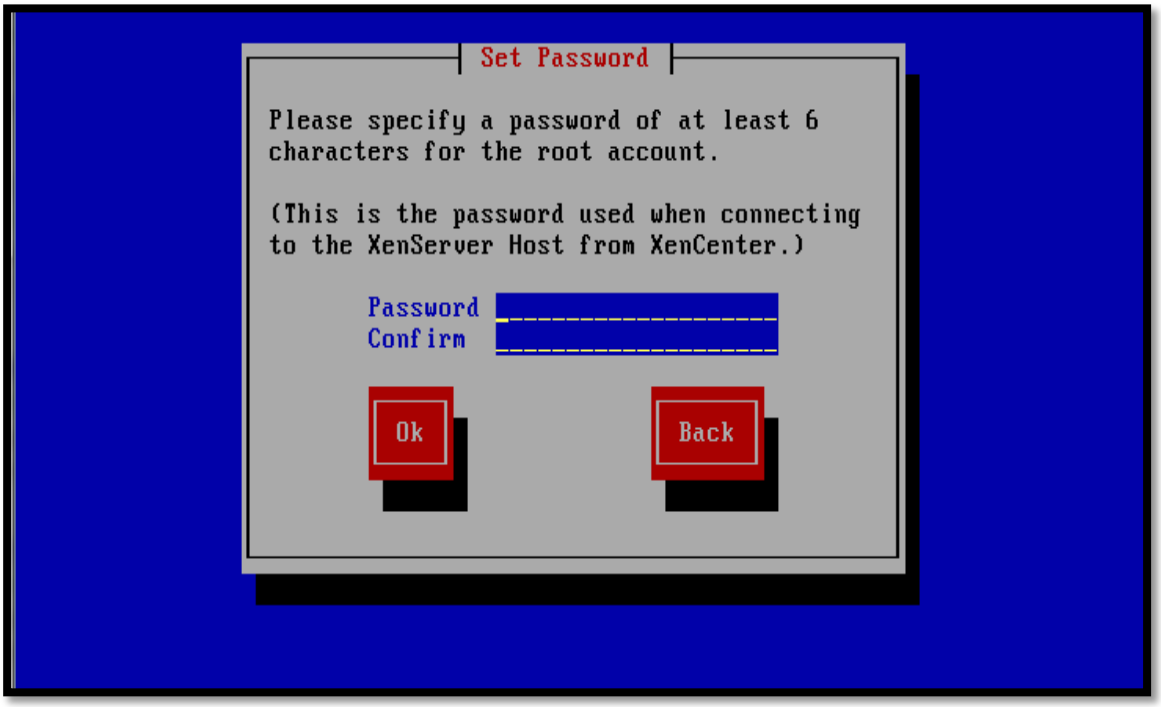


Imagen A1. 10: Seguridad de Xen Server

11. Seleccionamos la interfaz de red

Imagen A1. 11: Selección de NIC

12. Configurar la interfaz de red escogida

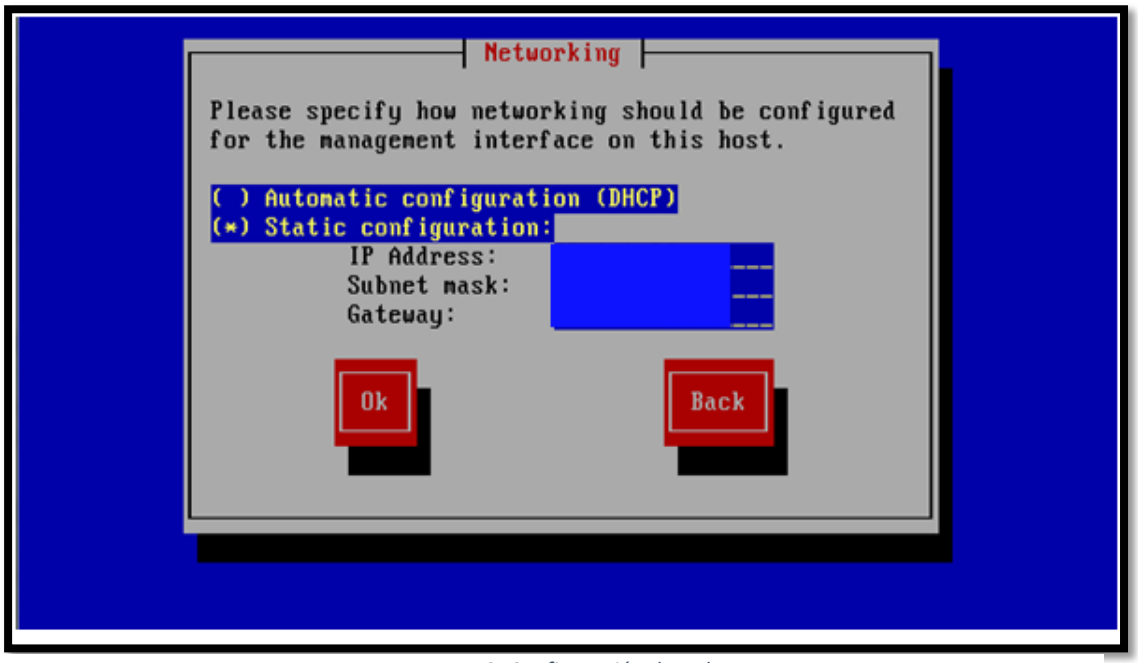


Imagen A1. 12: Configuración de red

13. Configurar el host y los DNS

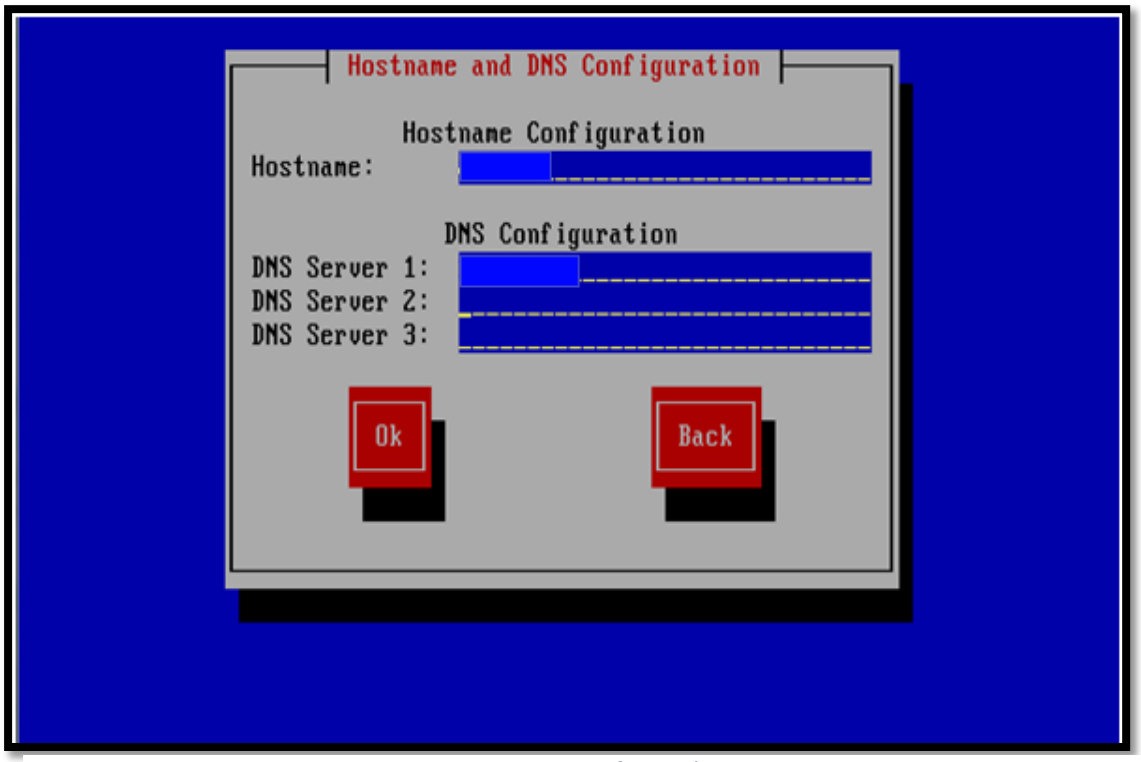


Imagen A1. 13: Configuración DNS

14. Seleccionar ubicación geográfica

Imagen A1. 14: Ubicación Geográica

15. Configuración de fecha y hora con NTP

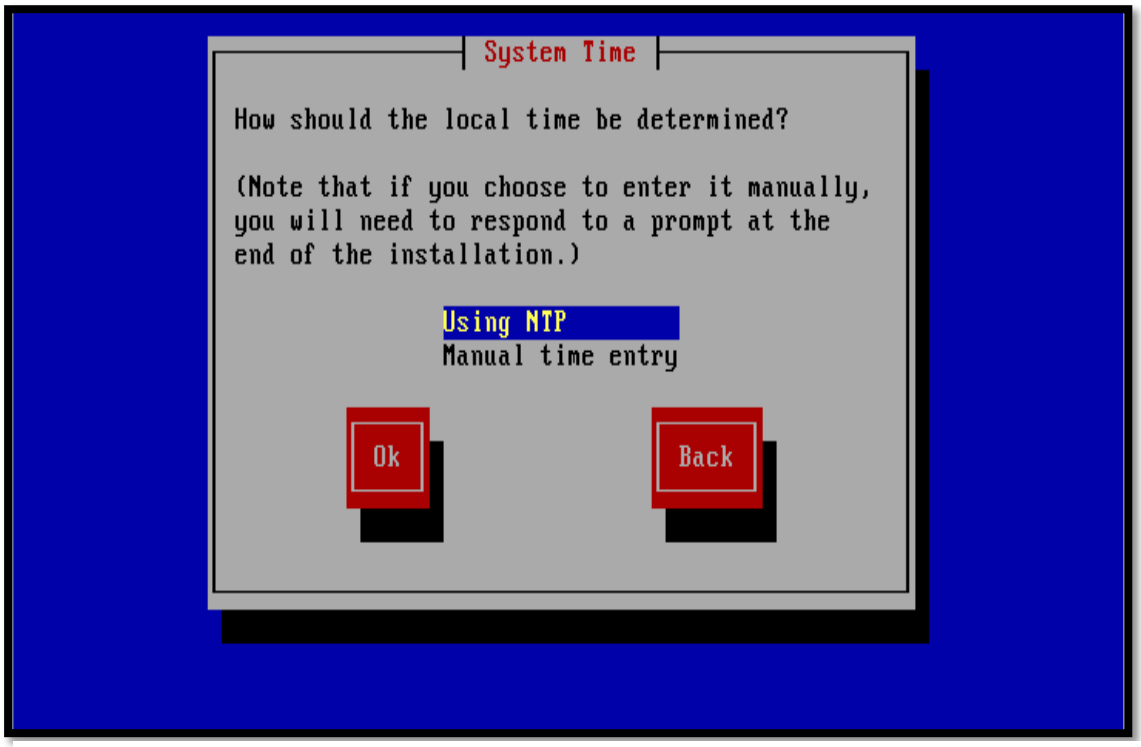


Imagen A1. 15: Network Time Protocol

16. Seleccionar Install XenServer

Imagen A1. 16: Finalización de instalación

17. **Una vez terminada la instalación la página inicial detalla la configuraciones de red para acceder con la IP a citrix**

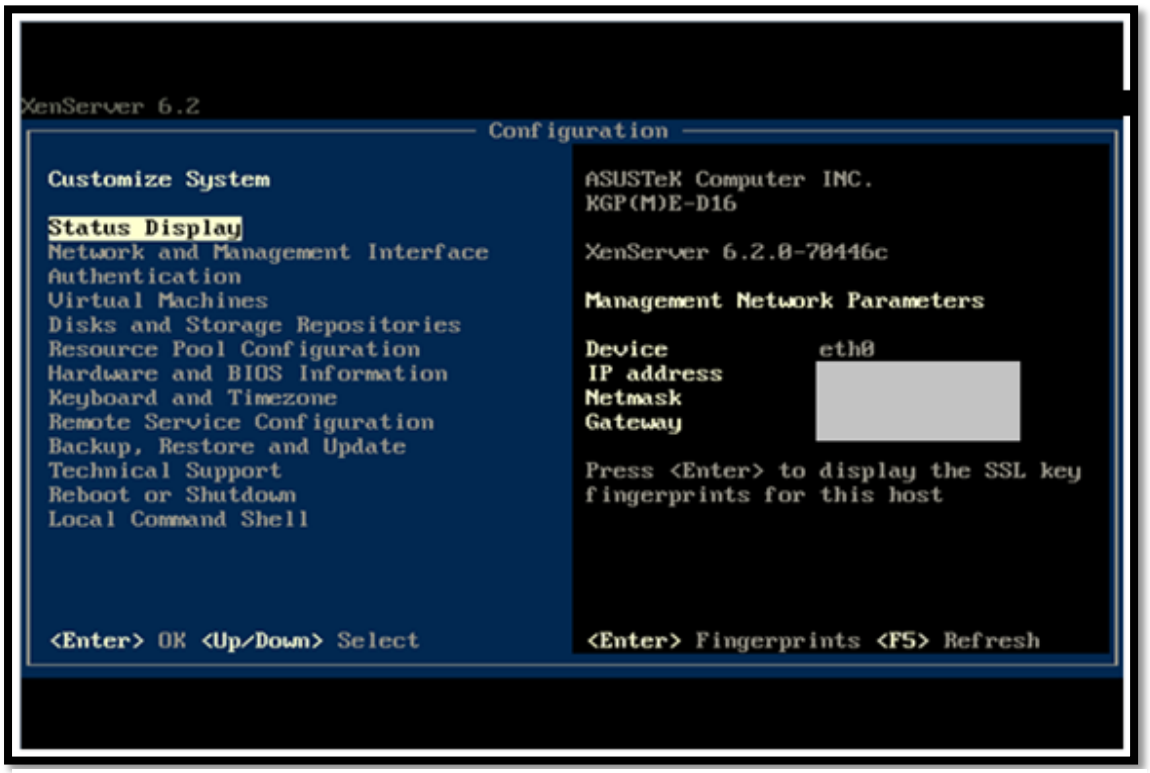


Imagen A1. 17: Acceso local a Xen Server

18. **Acceso remoto a Xen Server**

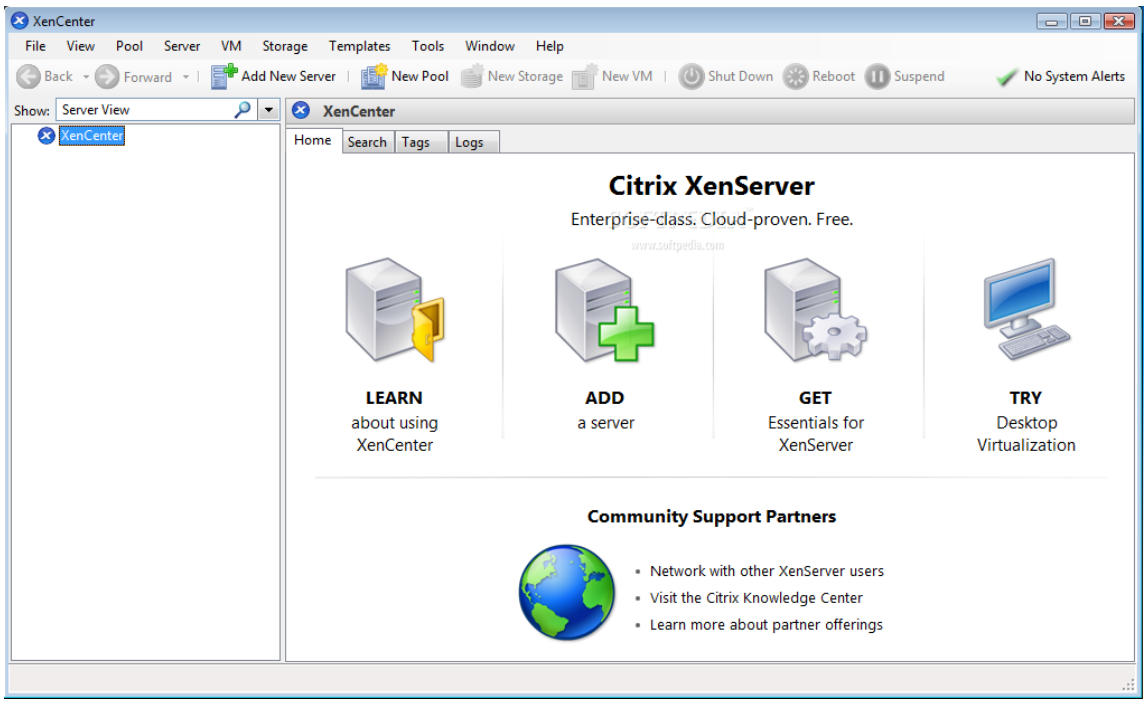


Imagen A1. 18: Acceso Remoto XenServer

Anexo 2

Instalación Openstack 3 Nodos (Neutron) versión JUNO para Ubuntu 14.04 en XEN Server

1. **Configuración de la Red**

1.1. Editar los archivos /etc/network/interfaces de cada nodo

Nodo Controller

```
# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
auto eth0
iface eth0 inet dhcp
auto eth1
iface eth1 inet static
    address 10.0.0.11
    netmask 255.255.255.0
    network 10.0.0.0
    broadcast 10.0.0.255
```

Nodo Compute

```
# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
auto eth0
iface eth0 inet dhcp
auto eth1
iface eth1 inet static
    address 10.0.0.31
    netmask 255.255.255.0
    network 10.0.0.0
    broadcast 10.0.0.255
auto eth2
iface eth2 inet static
    address 10.0.1.31
    netmask 255.255.255.0
    network 10.0.1.0
    broadcast 10.0.1.255
```

Nodo Network

```
# The loopback network interface
```

```
auto lo
iface lo inet loopback
# The primary network interface
auto eth0
iface eth0 inet dhcp
auto eth1
iface eth1 inet static
    address 10.0.0.21
    netmask 255.255.255.0
    network 10.0.0.0
    broadcast 10.0.0.255
auto eth2
iface eth2 inet static
    address 10.0.1.21
    netmask 255.255.255.0
    network 10.0.1.0
    broadcast 10.0.1.255
```

1.2.Configurar direcciones de host (Todos los nodos)

Editar el Archivo: /etc/hosts

```
# controller
10.0.0.11       controller
# network
10.0.0.21       compute1
# compute1
1.0.0.31       network1
```

1.3.Actualizaciones del sistema (Todos los nodos)

```
#sudo apt-get update && sudo apt-get update --fix-missing && sudo apt-get upgrade -y && sudo apt-get dist-upgrade -y
#reboot
```

1.4.Instalación de NTP en el controlador

```
#apt-get install ntp
```

1.5. Editar el Archivo: /etc/ntp.conf

```
server controller iburst
restrict -4 default kod notrap nomodify
restrict -6 default kod notrap nomodify
```

1.6. Configurar Otros Nodos Agregar al archivo /etc/ntp.conf

Agregar al archivo /etc/ntp.conf

```
server controller iburst
```

1.7.Instalación de Mysql en el Controlador

#apt-get install mariadb-server python-mysqldb

1.8. Editar el archivo : /etc/mysql/my.cnf

default-storage-engine = innodb
innodb_file_per_table
collation-server = utf8_general_ci
init-connect = 'SET NAMES utf8'
character-set-server = utf8
bind-address = 0.0.0.0

1.9.Instalacion de Paquetes de Juno

Instala el paquete python-software-properties para facilitar la administración del repositorio:

#apt-get install python-software-properties

Habilitar el depósito de archivos de Ubuntu nube:

#add-apt-repository cloud-archive:juno

Actualizar los paquetes en su sistema:

#apt-get update && apt-get dist-upgrade

1.10. Instalar RABBITMQ Servidor de Mensajería en el Nodo Controller

#apt-get install rabbitmq-server
#rabbitmqctl change_password guest 12345

2. Instalar Keystone en el Nodo Controller para Autenticación

#apt-get install keystone python-keystoneclient

2.1.Crear Base de Datos

#mysql -u root -p

CREATE DATABASE keystone;
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'localhost' IDENTIFIED BY '12345';
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'controller' IDENTIFIED BY '12345';
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'%' IDENTIFIED BY '12345';

2.2.Editar archivo /etc/keystone/keystone.conf

[DEFAULT]
admin_token = ADMIN_TOKEN
verbose = True
[database]

connection = mysql://keystone:12345@controller/keystone

2.3.Correr la base y reiniciar

Llenar base

#su -s /bin/sh -c "keystone-manage db_sync" keystone

#rm -f /var/lib/keystone/keystone.db

Reiniciar

#service keystone restart

#(crontab -l -u keystone 2>&1 | grep -q token_flush) || echo '@hourly /usr/bin/keystone-manage token_flush >/var/log/keystone/keystone-tokenflush.log 2>&1' >> /var/spool/cron/crontabs/keystone

2.4.Creación de usuarios y roles

Setear variables de entorno

#export OS_SERVICE_TOKEN=12345

#export OS_SERVICE_ENDPOINT=http://controller:35357/v2.0

Creación de usuarios, roles, tenants

#keystone tenant-create --name admin --description "Admin Tenant"

#keystone user-create --name admin --pass 12345 --email yulong@cloudcraft.cn

#keystone role-create --name admin

#keystone user-role-add --tenant admin --user admin --role admin

#keystone role-create --name _member_

#keystone user-role-add --tenant admin --user admin --role _member_

#keystone tenant-create --name demo --description "Demo Tenant"

#keystone user-create --name demo --pass 12345 --email yulong@cloudcraft.cn

#keystone user-role-add --tenant demo --user demo --role _member_

#keystone tenant-create --name service --description "Service Tenant"

#keystone service-create --name keystone --type identity --description "OpenStack Identity"

2.5.Crear endpoint

#keystone endpoint-create --service-id $(keystone service-list | awk '/ identity / {print $2}') --publicurl http://controller:5000/v2.0 --internalurl http://controller:5000/v2.0 --adminurl http://controller:35357/v2.0 --region regionOne

2.6.Verificar Conexión

Salir de variables de entorno

#unset OS_SERVICE_TOKEN OS_SERVICE_ENDPOINT

Probar creación correcta

#keystone --os-tenant-name admin --os-username admin --os-password 12345 --os-auth-url http://controller:35357/v2.0 token-get

#keystone --os-tenant-name admin --os-username admin --os-password 12345 --os-auth-url http://controller:35357/v2.0 tenant-list

3. Instalar Glance en el Nodo Controller para Imágenes

3.1.Crear Base de Datos

#mysql -u root -p

CREATE DATABASE glance;

GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'localhost' IDENTIFIED BY '12345';

GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'controller' IDENTIFIED BY '12345';

GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'%' IDENTIFIED BY '12345';

3.2.Creación de Usuarios y Contraseñas

Setear variables de entorno

#export OS_SERVICE_TOKEN=12345

#export OS_SERVICE_ENDPOINT=http://controller:35357/v2.0

Creación de usuarios, roles, tenants

#keystone user-create --name glance --pass 12345

#keystone user-role-add --user glance --tenant service --role admin

#keystone service-create --name glance --type image --description "OpenStack Image Service"

#keystone endpoint-create --service-id $(keystone service-list | awk '/ image / {print $2}') --publicurl http://controller:9292 --internalurl http://controller:9292 --adminurl http://controller:9292 --region regionOne

3.3.Instalación de Glance

Instalación de Paquetes

#apt-get install glance python-glanceclient

Editar el archivo y agregar /etc/glance/glance-api.conf

[DEFAULT]

verbose = True

[database]

connection = mysql://glance:12345@controller/glance

[keystone_authtoken]

auth_uri = http://controller:5000/v2.0

identity_uri = http://controller:35357

admin_tenant_name = service

admin_user = glance

admin_password = 12345
[paste_deploy]
flavor = keystone

Editar el archivo /etc/glance/glance-registry.conf

[DEFAULT]
verbose = True
[database]
connection = mysql://glance:12345@controller/glance
[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = glance
admin_password = 12345
[paste_deploy]
flavor = keystone

3.4.Correr base y reiniciar

Llenar datos en glance

#su -s /bin/sh -c "glance-manage db_sync" glance

Borrar enlace a sqlite

#rm -f /var/lib/glance/glance.sqlite

Reiniciar servicios

#service glance-registry restart
#service glance-api restart

3.5.Verificamos Glance

Crear archivo keystonerc.sh

#nano keystonerc.sh

Editar archive y agregar en keystonerc.sh

export OS_TENANT_NAME=admin
export OS_USERNAME=admin
export OS_PASSWORD=12345
export OS_AUTH_URL=http://controller:35357/v2.0

3.6.Descargar la imagen

#curl -O http://cdn.download.cirros-cloud.net/0.3.3/cirros-0.3.3-x86_64-disk.img
glance image-create --name "cirros-0.3.3-x86_64" --file cirros-0.3.3-x86_64-disk.img --disk-format qcow2 --container-format bare --is-public True –progress

4. Instalar Nova en el Nodo Controller

4.1.Creamos la Base de Datos

#mysql -u root -p

CREATE DATABASE nova;

GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'localhost' IDENTIFIED BY '12345';

GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'controller' IDENTIFIED BY '12345';

GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'%' IDENTIFIED BY '12345';

4.2.Crear usuarios y roles

#keystone user-create --name nova --pass 12345

#keystone user-role-add --user nova --tenant service --role admin

#keystone service-create --name nova --type compute --description "OpenStack Compute"

#keystone endpoint-create --service-id $(keystone service-list | awk '/ compute / {print $2}') --publicurl http://controller:8774/v2/%\(tenant_id\)s --internalurl http://controller:8774/v2/%\(tenant_id\)s --adminurl http://controller:8774/v2/%\(tenant_id\)s --region regionOne

4.3.Instalar paquetes de Nova

#apt-get install nova-api nova-cert nova-conductor nova-consoleauth nova-novncproxy nova-scheduler python-novaclient

4.4.Editar el archive /etc/nova/nova.conf

[database]

connection = mysql://nova:12345@controller/nova

[DEFAULT]

verbose = True

my_ip = 10.0.0.11

rpc_backend = rabbit

rabbit_host = controller

rabbit_password = 12345

vncserver_listen = 10.0.0.11

vncserver_proxyclient_address = 10.0.0.11

[keystone_authtoken]

auth_uri = http://controller:5000/v2.0

identity_uri = http://controller:35357

admin_tenant_name = service

admin_user = nova

admin_password = 12345

[glance]
host = controller

4.5.Correr Base y Reiniciar Nova

Llenar base

#su -s /bin/sh -c "nova-manage db sync" nova

Borrar enlace con sqlite

#rm -f /var/lib/nova/nova.sqlite

4.6.Reiniciar Servicios

#service nova-api restart
#service nova-cert restart
#service nova-consoleauth restart
#service nova-scheduler restart
#service nova-conductor restart
#service nova-novncproxy restart

4.7.Instalar Nova en el Nodo Compute

#apt-get install nova-compute

4.8.Configurar el archivo /etc/nova/nova.conf

[DEFAULT]
verbose = True
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = 12345
my_ip = 10.0.0.11
vnc_enabled = True
vncserver_listen = 0.0.0.0
vncserver_proxyclient_address = 10.0.0.11
novncproxy_base_url = http://10.0.0.11:6080/vnc_auto.html

[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = nova
admin_password = 12345

[glance]
host = controller

4.9.Editar el archivo /etc/nova/nova-compute.conf

[libvirt]
virt_type = qemu/kvm

4.10. Reiniciar los servicios de Nova

#service nova-compute restart

4.11. Borrar enlace con sqlite

#rm -f /var/lib/nova/nova.sqlite

4.12. Verificar en el Nodo Controlador

#nova service-list
#nova image-list

5. Instalar Neutron en el Nodo Controlador

5.1.Crear Base de Datos

#mysql -u root -p

CREATE DATABASE neutron;
GRANT ALL PRIVILEGES ON neutron.* TO 'neutron'@'localhost' IDENTIFIED BY '12345';
GRANT ALL PRIVILEGES ON neutron.* TO 'neutron'@'controller' IDENTIFIED BY '12345';
GRANT ALL PRIVILEGES ON neutron.* TO 'neutron'@'%' IDENTIFIED BY '12345';

5.2.Crear Usuarios y Roles

#keystone user-create --name neutron --pass 12345
#keystone user-role-add --user neutron --tenant service --role admin
#keystone service-create --name neutron --type network --description "OpenStack Networking"
#keystone endpoint-create --service-id $(keystone service-list | awk '/ network / {print $2}') --publicurl http://controller:9696 --adminurl http://controller:9696 --internalurl http://controller:9696 --region regionOne

5.3.Instalar Paquetes Neutron

#apt-get install neutron-server neutron-plugin-ml2 python-neutronclient

5.4.Configurar el Archivo /etc/neutron/neutron.conf

[DEFAULT]
verbose = True
auth_strategy = keystone
rpc_backend = rabbit
rabbit_host = controller

rabbit_password = 12345
core_plugin = ml2
service_plugins = router
allow_overlapping_ips = True
notify_nova_on_port_status_changes = True
notify_nova_on_port_data_changes = True
nova_url = http://controller:8774/v2
nova_admin_auth_url = http://controller:35357/v2.0
nova_region_name = regionOne
nova_admin_username = nova
nova_admin_tenant_id = SERVICE_TENANT_ID
nova_admin_password = 12345

[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = neutron
admin_password = 12345

[database]
connection = mysql://neutron:12345@controller/neutron

5.5.Configurar Plugin ML2 /etc/neutron/plugins/ml2/ml2_conf.ini

[ml2]
tenant_network_types = vlan
type_drivers = local,flat,vlan,gre,vxlan
mechanism_drivers = openvswitch,linuxbridge
[ml2_type_vlan]
network_vlan_ranges = physnet2:1000:2999
[securitygroup]
enable_security_group = True
enable_ipset = True
firewall_driver =
neutron.agent.linux.iptables_firewall.OVSHybridIptablesFirewallDriver

5.6.Configurar Nova /etc/nova/nova.conf

[DEFAULT]
network_api_class = nova.network.neutronv2.api.API
security_group_api = neutron
linuxnet_interface_driver = nova.network.linux_net.LinuxOVSInterfaceDriver
firewall_driver = nova.virt.firewall.NoopFirewallDriver

[neutron]

url = http://controller:9696
auth_strategy = keystone
admin_auth_url = http://controller:35357/v2.0
admin_tenant_name = service
admin_username = neutron
admin_password = 12345

5.7.Correr la Base

#su -s /bin/sh -c "neutron-db-manage --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/ml2/ml2_conf.ini upgrade juno" neutron

5.8.Reiniciar Servicios Nova

#service nova-api restart
#service nova-scheduler restart
#service nova-conductor restart

5.9.Reiniciar Servicios Neutron

#service neutron-server restart

5.10. Verificar Operación

#source keystonerc
#neutron ext-list

6. Configurar Neutron en el Nodo Network

6.1.Editar el archivo /etc/sysctl.conf

net.ipv4.ip_forward=1
net.ipv4.conf.all.rp_filter=0
net.ipv4.conf.default.rp_filter=0

Aplicar Cambios

#sysctl –p

6.2.Instalar Paquetes

#apt-get install neutron-plugin-ml2 neutron-plugin-openvswitch-agent neutron-l3-agent neutron-dhcp-agent ipset

6.3.Editar el Archivo /etc/neutron/neutron.conf

Borrar de [database] las conexiones

[DEFAULT]
verbose = True
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = 12345
auth_strategy = keystone
core_plugin = ml2

service_plugins = router
allow_overlapping_ips = True

[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = neutron
admin_password = 12345

6.4.Editar el archivo /etc/neutron/plugins/ml2/ml2_conf.ini

[ml2]
tenant_network_types = vlan
type_drivers = local,flat,vlan,gre,vxlan
mechanism_drivers = openvswitch,linuxbridge
[ml2_type_vlan]
network_vlan_ranges = physnet2:1000:2999
[securitygroup]
enable_security_group = True
enable_ipset = True
firewall_driver =
neutron.agent.linux.iptables_firewall.OVSHybridIptablesFirewallDriver
[ovs]
local_ip = 10.0.1.21
tenant_network_type = vlan
integration_bridge = br-int
network_vlan_ranges = physnet2:1000:2999
bridge_mappings = physnet2:br-eth2

6.5.Configurar el Agente Capa3 L3 /etc/neutron/l3_agent.ini

[DEFAULT]
verbose = True
interface_driver = neutron.agent.linux.interface.OVSInterfaceDriver
use_namespaces = True
external_network_bridge = br-ex
Configurar el Agente DHCP /etc/neutron/dhcp_agent.ini
[DEFAULT]
verbose = True
interface_driver = neutron.agent.linux.interface.OVSInterfaceDriver
dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq
use_namespaces = True
Configurar el Agente Metadata /etc/neutron/metadata_agent.ini
[DEFAULT]
verbose = True

auth_url = http://controller:5000/v2.0
auth_region = regionOne
admin_tenant_name = service
admin_user = neutron
admin_password = 12345
nova_metadata_ip = controller
metadata_proxy_shared_secret = 12345

6.6.En el nodo Controller agregar al archivo /etc/nova/nova.conf

[neutron]
service_metadata_proxy = True
metadata_proxy_shared_secret = 12345

6.7.Reinicia el servicio

#service nova-api restart

6.8.Configurar OpenvSwitch Reiniciando el servicio

#service openvswitch-switch restart

6.9.Agregar puertos

ovs-vsctl add-br br-ex
ovs-vsctl add-port br-ex eth1
ovs-vsctl add-br br-eth2
ovs-vsctl add-port br-eth2 eth2

```
auto eth1
iface eth1 inet manual
    up ip address add 0/0 dev $IFACE
    up ip link set $IFACE up
    down ip link set $IFACE down

auto br-ex
iface br-ex inet static
    address 10.0.0.11
    netmask 255.255.255.0
    network 10.0.0.0
    broadcast 10.0.0.255

auto eth2
iface eth2 inet manual
    up ip address add 0/0 dev $IFACE
    up ip link set $IFACE up
    down ip link set $IFACE down

auto br-eth2
```

```
iface br-eth2 inet static
    address 10.0.1.21
    netmask 255.255.255.0
    network 10.0.1.0
    broadcast 10.0.1.255
```

6.10. Reiniciar los servicios Neutron

```
#service neutron-plugin-openvswitch-agent restart
#service neutron-l3-agent restart
#service neutron-dhcp-agent restart
#service neutron-metadata-agent restart
```

6.11. Listamos los servicios

```
#neutron agent-list
```

7. Instalar Neutron en el Nodo de Computo

7.1.Editar el archivo /etc/sysctl.conf

```
net.ipv4.conf.all.rp_filter=0
net.ipv4.conf.default.rp_filter=0
```

Comprobar los cambios

```
#sysctl –p
```

7.2.Instalar los servicios de Neutron

```
#apt-get install neutron-plugin-ml2 neutron-plugin-openvswitch-agent ipset
```

7.3.Editar el archivo /etc/neutrón/neutrón.conf

```
[DEFAULT]
verbose = True
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = 12345
auth_strategy = keystone
core_plugin = ml2
service_plugins = router
allow_overlapping_ips = True

[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = neutron
admin_password = 12345
```

7.4.Editar el archivo /etc/neutron/plugins/ml2/ml2_conf.ini

```
[ml2]
tenant_network_types = vlan
type_drivers = local,flat,vlan,gre,vxlan
mechanism_drivers = openvswitch,linuxbridge
[ml2_type_vlan]
network_vlan_ranges = physnet2:1000:2999
[securitygroup]
enable_security_group = True
enable_ipset = True
firewall_driver =
neutron.agent.linux.iptables_firewall.OVSHybridIptablesFirewallDriver
[ovs]
local_ip = 10.0.1.31
tenant_network_type = vlan
integration_bridge = br-int
network_vlan_ranges = physnet2:1000:2999
bridge_mappings = physnet2:br-eth2
```

7.5.Configurar OpenvSwitch

```
#service openvswitch-switch restart
#ovs-vsctl add-br br-eth2
#ovs-vsctl add-port br-eth2 eth2
        auto eth2
        iface eth2 inet manual
            up ip address add 0/0 dev $IFACE
            up ip link set $IFACE up
            down ip link set $IFACE down
        auto br-eth2
        iface br-eth2 inet static
            address 10.0.1.31
            netmask 255255.255.0
            network 10.0.1.0
            broadcast 10.0.1.255
```

7.6.Editar el archivo /etc/nova/nova.conf

```
[DEFAULT]
network_api_class = nova.network.neutronv2.api.API
security_group_api = neutron
linuxnet_interface_driver = nova.network.linux_net.LinuxOVSInterfaceDriver
firewall_driver = nova.virt.firewall.NoopFirewallDriver
[neutron]
url = http://controller:9696
auth_strategy = keystone
```

admin_auth_url = http://controller:35357/v2.0
admin_tenant_name = service
admin_username = neutron
admin_password = 12345

7.7.Reiniciar Servicios

#service nova-compute restart
#service neutron-plugin-openvswitch-agent restart

7.8.Verificar Servicios

#neutron agent-list

8. Instalación de Dashboard (Horizon)

8.1.Instalación de Paquetes

#apt-get install openstack-dashboard apache2 libapache2-mod-wsgi memcached python-memcache

8.2.Editar el archivo /etc/openstack-dashboard/local_settings.py

```
OPENSTACK_HOST = "controller"
ALLOWED_HOSTS = ['*']
CACHES = {
  'default': {
     'BACKEND': 'django.core.cache.backends.memcached.MemcachedCache',
     'LOCATION': '127.0.0.1:11211',
  }
}
        TIME_ZONE = UTP
```

8.3.Reiniciar los servicios

#service apache2 restart
#service memcached restart

9. Verificamos los siguientes servicios

#source keystonerc
#keystone user-list
#keystone role-list
#keystone tenant-list
#keystone service-list
#keystone endpoint-list
#sudo nova-manage service list
#neutron agent-list

10. Instalación de cinder en el Nodo Controller

10.1. Crear Base de Datos

#mysql -u root -p

CREATE DATABASE cinder;

GRANT ALL PRIVILEGES ON cinder.* TO 'cinder'@'localhost' IDENTIFIED BY '12345';

GRANT ALL PRIVILEGES ON cinder.* TO 'cinder'@'controller' IDENTIFIED BY '12345';

GRANT ALL PRIVILEGES ON cinder.* TO 'cinder'@'%' IDENTIFIED BY '12345';

10.2. Crear usuarios, roles, tenants en Keystone

#keystone user-create --name cinder --pass 12345

#keystone user-role-add --user cinder --tenant service --role admin

#keystone service-create --name cinder --type volume --description "OpenStack Block Storage"

#keystone service-create --name cinderv2 --type volumev2 --description "OpenStack Block Storage"

#keystone endpoint-create --service-id $(keystone service-list | awk '/ volume / {print $2}') --publicurl http://controller:8776/v1/%\(tenant_id\)s --internalurl http://controller:8776/v1/%\(tenant_id\)s --adminurl http://controller:8776/v1/%\(tenant_id\)s --region regionOne

#keystone endpoint-create --service-id $(keystone service-list | awk '/ volumev2 / {print $2}') --publicurl http://controller:8776/v2/%\(tenant_id\)s --internalurl http://controller:8776/v2/%\(tenant_id\)s --adminurl http://controller:8776/v2/%\(tenant_id\)s --region regionOne

10.3. Instalar Paquetes de Cinder

#apt-get install cinder-api cinder-scheduler python-cinderclient

10.4. Editar el archivo /etc/cinder/cinder.conf

```
[DEFAULT]
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = 12345
auth_strategy = keystone
my_ip = 10.0.0.11
verbose = True

[database]
connection = mysql://cinder:12345@controller/cinder

[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = cinder
admin_password = 12345
```

10.5. Llenar Base de Datos

#su -s /bin/sh -c "cinder-manage db sync" cinder

Borrar enlace a sqlite

#rm -f /var/lib/cinder/cinder.sqlite

10.6. Reiniciara Servicios de cinder

#service cinder-api restart
#service cinder-scheduler restart

11. Crear Nodo de Almacenamiento Cinder

11.1. Instalar Paquetes

#apt-get install lvm2

11.2. Creación de Volúmenes Físicos LVM

#pvcreate /dev/sdb

11.3. Crear el grupo de volúmenes Cinder

#vgcreate cinder-volumes /dev/sdb

11.4. Instalar paquetes de cinder-volume

#apt-get install cinder-volume python-mysqldb

11.5. Editar el archivo /etc/cinder/cinder.conf

```
[DEFAULT]
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = 12345
auth_strategy = keystone
my_ip = 10.0.0.11
glance_host = controller
verbose = True

[database]
connection = mysql://cinder:12345@controller/cinder

[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = cinder
admin_password = 12345
```

11.6. Borrar enlace a Sqlite

#rm -f /var/lib/cinder/cinder.sqlite

11.7. Reiniciar Servicios de Cinder

#rm -f /var/lib/cinder/cinder.sqliteservice tgt restart
#service cinder-volume restart

11.8. Verificar con los comandos

#cinder service-list
#cinder create --display-name demo-volume1 1
#cinder list

Anexo 3

Implementación Aplicada en un solo nodo

1. Una vez instalado el Sistema Operativo Ubuntu en la versión 14.04 server proceder con la actualización de repositorios y servicios mediante los comandos:

```
#apt-get update && apt-get upgrade
```

2. Se debe configurar las interfaces de red para las conexiones de administración (interfaz em1) y la red pública (interfaz p5p1), para esto se edita el archivo:

```
# nano   /etc/network/interfaces

auto lo
iface lo inet loopback
#Primary interface
auto em1
iface em1 inet static
address 192.168.2.103
netmask 255.255.255.0
network 192.168.2.0
broadcast 192.168.2.255
gateway 192.168.2.1
dns-nameservers 8.8.8.8 8.8.4.4

auto p5p1
iface p5p1 inet manual
up ifconfig $IFACE 0.0.0.0 up
up ip link set $IFACE promisc on
down ifconfig $IFACE 0.0.0.0 down
```

3. Para aplicar los cambios reiniciar el servidor con el comando:

```
#reboot
```

4. Proceder con la descarga del repositorio git y descarga de repositorios de la versión icehouse de openstack.

```
$ sudo su
# apt-get install -y git
# git clone -b icehouse https://github.com/CiscoSystems/puppet_openstack
_builder
# cd puppet_openstack_builder
#git checkout i.0
```

5. Una vez con los repositorios descargados se procede con la configuración del archivo install.sh que se crea en la carpeta puppet_openstack_builder

```
# nano puppet_openstack_builder/install-scripts/install.sh
```

```
export vendor=cisco
export vendor_name="${vendor:-cisco}"
export default_interface="${default_interface:-em1}"
export external_interface="${external_interface:-p5p1}"
export scenario="${scenario:-all_in_one}"
```

6. Luego se edita el archivo install.sh con los respectivos cambios

```
# nano puppet_openstack_builder/install-scripts/install.sh
```

7. Luego se procede con la instalación de los módulos empleados en Opentack Icehouse Cisco

```
# cd puppet_openstack_builder/install-scripts && ./install.sh
```

8. Una vez con los repositorios descargados se procede a la configuración de la nube a través de la interfaz web de Openstack. Para esto ingresar a cualquier navegador Web y digitar la dirección IP destinada para la administración que se creó anteriormente.

9. Ingresamos a la interfaz con el usuario admin y la contraseña configurada en él ./install.sh

10. Se procede con la creación de la red interna para la conexión de máquinas virtuales

Imagen A3. 1: Creación de una red

Imagen A3. 2: Creación de una red

Imagen A3. 3: Creación de una red

12. Configuración de la red externa para conexión de máquinas virtuales a la red externa.

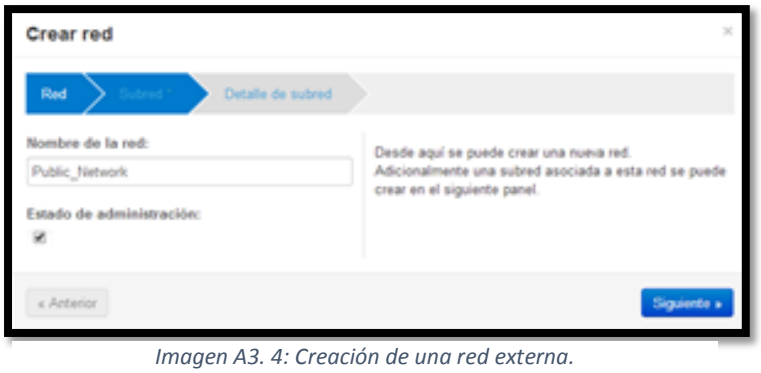


Imagen A3. 4: Creación de una red externa.

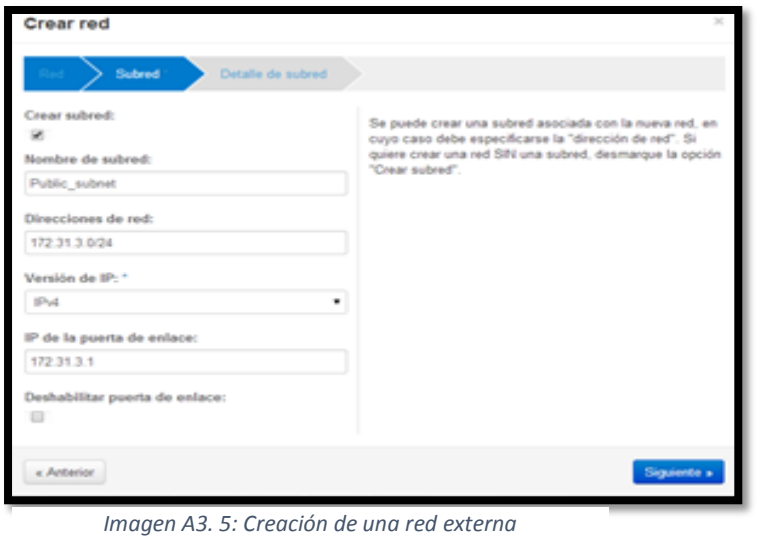


Imagen A3. 5: Creación de una red externa

13. Creación del router virtual

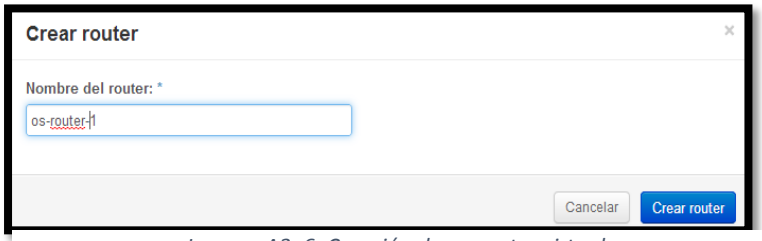


Imagen A3. 6: Creación de un router virtual

14. Para añadir la interfaz se da clic en el router y se edita los siguientes parámetros

Imagen A3. 7: ANade interfaz al router virtual

14. Luego editar la puerta de enlace

Imagen A3. 8: Definición de Gateway

15. Lanzamiento de Instancias

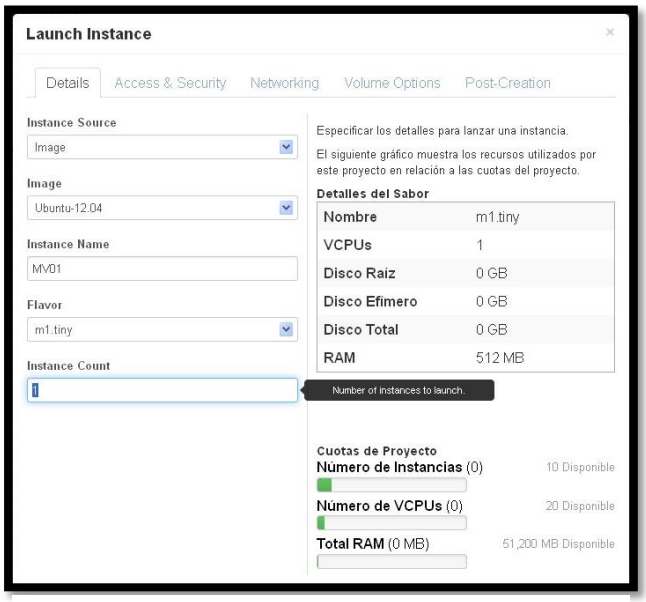


Imagen A3. 9: Lanzamiento de instancias

Anexo 4

Instalación Asterisk

Para la instalación requerimos los paquetes de Asterisk

1. Para actualizar el sistema ejecutar el comando:

    #apt-get update && apt-get upgrade

2. Para instalar los repositorios de asterisk ejecutamos el comando:

    #apt-get install asterisk

Configuración de Asterisk

Una vez instalado Asterisk versión 13.1.1 procedemos con la configuración de las cuentas de usuarios para la comunicación de usuarios que accederán a la red. [48]

1. Para la configuración inicial de asterisk editar el archivo con los siguientes parámetros:

    #nano /etc/asterisk/sip.conf

```
[general]
context=unauthenticated
allowguest=no
srvlookup=yes
udpbindaddr=0.0.0.0
nat=yes
tcpenable=no
tos=0x18
rtptimeout=60
encryption=no
externip=172.31.3.21
localnet= 10.10.10.0/255.255.255.0

tos_sip=0x04
cos_sip=3

[office-phone](!)
type=friend
context=LocalSets
host=dynamic
nat=yes
secret=1234
dtmfmode=auto
externip=172.31.3.21
localnet=10.10.10.0/255.255.255.0
```

disallow=all
allow=alaw
allow=ullaw
allow=gsm
qualify=yes
tos =0x04

[1000](office-phone)
[1001](office-phone)
[1002](office-phone)
[1003](office-phone)
[1004](office-phone)
[1005](office-phone)

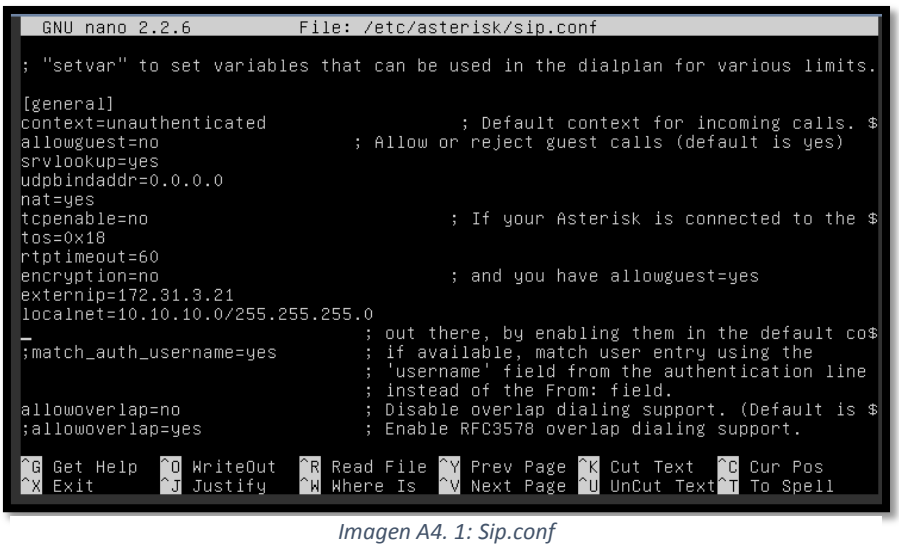


Imagen A4. 1: Sip.conf

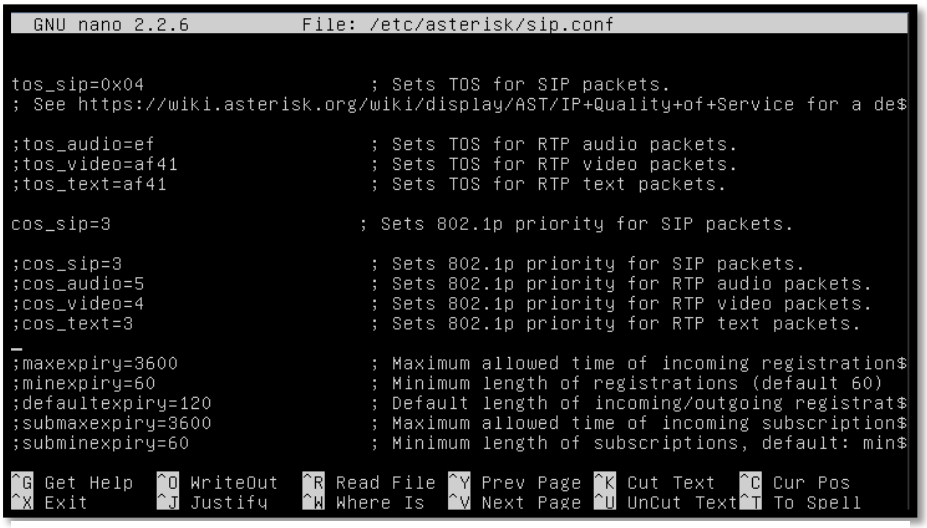


Imagen A4. 2: Sip.conf

2. Aplicar cambios en la central SIP

CLI> sip reload

CLI> module reload chan_sip.so

3. Para editar las extensiones editar el archivo con las siguientes configuraciones:

#nano /etc/asterisk/extensions.conf

```
[LocalSets]
exten => 1000,1,Dial(SIP/1000)
exten => 1001,1,Dial(SIP/1001)
exten => 1002,1,Dial(SIP/1002)
exten => 1003,1,Dial(SIP/1003)
exten => 1004,1,Dial(SIP/1004)
exten => 1005,1,Dial(SIP/1005)

exten => 123,1,Answer()
same => n,Playback(hello-world)
same => n,Hangup()
```

4. Una vez configurados los archivos ingresar a asterisk mediante el comando:

#asterisk –r

5. Aplicar cambios en las extensiones

CLI> sip reload

CLI> dialplan reload

Anexo 5

Check de Actividades Realizadas

Actividad	Septiembre	Octubre	Noviembre	Diciembre	Enero	Febrero	Revisión
Desarrollo de la teoría							
Capitulo 1							x
Capitulo 2							x
Análisis de la Plataforma							x
Diseño de la Plataforma							
Capitulo 3							x
Implementación de Openstack							
Capitulo 4							x
Pruebas en Openstack							
Capitulo 5							x
Analisis de Costos							
Capitulo 6							x

Tabla A5. 1: Actividades realizadas

Anexo 6

Cronograma de actividades

Id		Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1			Capítulo 1: Cloud Computing	32 días	lun 08/04/14	mar 09/16/14		
2			Concepto de Cloud	2 días	lun 08/04/14	mar 08/05/1		
3			Características de C	2 días	mié 08/06/1	jue 08/07/14	2	
4			*Modelos de Servicio de Cloud Computing*	4 días	vie 08/08/14	mié 08/13/14	3	
5			Servicios por Sof	2 días	vie 08/08/14	lun 08/11/14		
6	✓		Servicios por Plat	1 día	mar 08/12/1	mar 08/12/1		
7	✓		*Servicios por Infr*	1 día	mié 08/13/1	mié 08/13/1		
8			*Modelos de Despliegue de Cloud Computing*	24 días	jue 08/14/14	mar 09/16/14	4	
9			Cloud Privada	2 días	jue 08/14/14	vie 08/15/14		
10	✓		Cloud Pública	1 día	lun 08/18/14	lun 08/18/14		
11	✓		Cloud Híbrida	1 día	mar 08/19/1	mar 08/19/1		
12			Arquitectura Bás	2 días	mié 08/20/1	jue 08/21/14		
13			Ventajas y Desve	6 días	vie 08/22/14	vie 08/29/14		
14			Disponibilidad	2 días	vie 08/22/14	lun 08/25/14		
15	✓		Integridad	1 día	mar 08/26/1	mar 08/26/1		
16	✓		Confidencialid	1 día	mié 08/27/1	mié 08/27/1		
17	✓		VoIP Conceptos	2 días	jue 08/28/14	vie 08/29/14		
18			Protocolos empl	4 días	lun 09/01/1	jue 09/04/1		
19			Protocolos de	2 días	lun 09/01/14	mar 09/02/1		
20	✓		Protocolos de	1 día	mié 09/03/1	mié 09/03/1		
21	✓		Protocolos de	1 día	jue 09/04/14	jue 09/04/14		
22			Tipos de diseños	8 días	vie 09/05/14	mar 09/16/1		
23			Diseño de una	2 días	vie 09/05/14	lun 09/08/14		
24	✓		Diseño de una	1 día	mar 09/09/1	mar 09/09/1		
25	✓		Diseño de una	1 día	mié 09/10/1	mié 09/10/1		
26			Diseño de Vol	4 días	jue 09/11/14	mar 09/16/1		

Imagen A6. 1: Cronograma

Id		Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
27			**Capítulo 2: Requerimientos de Diseño**	**25 días**	**mié 09/17/14**	**mar 10/21/14**	**1**	
28	✓		*Virtualización Xen-Server*	**3 días**	**mié 09/17/14**	**vie 09/19/14**	**1**	
29	✓		Características	1 día	mié 09/17/14	mié 09/17/14		
30	✓		Requerimientos	1 día	jue 09/18/14	jue 09/18/14		
31	✓		Ventajas y Desve	1 día	vie 09/19/14	vie 09/19/14		
32			*Solución de OpenStack*	**6 días**	**lun 09/22/14**	**lun 09/29/14**	**28**	
33			Características	2 días	lun 09/22/14	mar 09/23/1		
34			Requerimientos	3 días	mié 09/24/14	vie 09/26/14		
35	✓		Ventajas y Desve	1 día	lun 09/29/14	lun 09/29/14		
36			*Identificación de Módulos de OpenStack*	**14 días**	**mar 09/30/14**	**vie 10/17/14**	**32**	
37			Horizon o Dashb	2 días	mar 09/30/1	mié 10/01/14		
38			Nova	2 días	jue 10/02/14	vie 10/03/14		
39			Quantum	2 días	lun 10/06/14	mar 10/07/1		
40			Cinder	2 días	mié 10/08/14	jue 10/09/14		
41			Glace	2 días	vie 10/10/14	lun 10/13/14		
42			Swift	2 días	mar 10/14/1	mié 10/15/14		
43			Keystone	2 días	jue 10/16/14	vie 10/17/14		
44			*Administración de I*	2 días	lun 10/20/14	mar 10/21/1	36	
45			**Capítulo 3: Diseño del Sistema**	**14 días**	**mié 10/22/14**	**lun 11/10/14**	**27**	
46			*Arquitectura del Sis*	3 días	mié 10/22/14	vie 10/24/14	27	
47			*Diagramas de la Re*	2 días	lun 10/27/14	mar 10/28/1	46	
48			*Herramientas de Vi*	2 días	mié 10/29/14	jue 10/30/14	47	
49			*Interfaces de Conex*	3 días	vie 10/31/14	mar 11/04/1	48	
50			*Seguridad en la Nu*	4 días	mié 11/05/14	lun 11/10/14	49	

Imagen A6. 2: Cronograma

Id		Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
51			**Capítulo 4: Implementación de Sistema**	**35 días**	**mar 11/11/14**	**lun 12/29/14**	**45**	
52			*Virtualización de Se*	5 días	mar 11/11/1	lun 11/17/14	45	
53			*Configuraciones de*	15 días	mar 11/18/1	lun 12/08/14	52	
54			*Configuraciones de*	15 días	mar 12/09/1	lun 12/29/14	53	
55			**Capítulo 5: Pruebas**	**16 días**	**lun 01/05/15**	**lun 01/26/15**	**51**	
56			*Herramientas de Pr*	5 días	lun 01/05/15	vie 01/09/15	51	
57			*Metodología de Pruebas*	**11 días**	**lun 01/12/15**	**lun 01/26/15**	**56**	
58			Pruebas de Cone	3 días	lun 01/12/15	mié 01/14/1	56	
59			Pruebas de Mon	3 días	jue 01/15/15	lun 01/19/15	58	
60			Pruebas de Segu	5 días	mar 01/20/1	lun 01/26/15	59	
61			**Capítulo 6: Análisis de Costos del Sistema**	**10 días**	**mar 01/27/15**	**lun 02/09/15**	**55**	
62			*Análisis de Costos d*	3 días	mar 01/27/1	jue 01/29/15	55	
63			*Análisis de Costos d*	3 días	vie 01/30/15	mar 02/03/1	62	
64			*Análisis de Costos d*	4 días	mié 02/04/1	lun 02/09/15	63	

Imagen A6. 3: Cronograma