



UNIVERSIDAD POLITÉCNICA SALESIANA

Sede Guayaquil

Facultad de Ingenierías

Carrera de Ingeniería Electrónica.

Tesis previa a la obtención del título de ingeniero en electrónica
con mención en sistemas computacionales.

Tema:

Servicio de virtualización de infraestructura tecnológica basado
en cloud computing.

Autor:

Gabriel Ullauri García

Tutor:

Ing. Adrián Arce

Guayaquil, Enero del 2013

DECLARATORIA DE RESPONSABILIDAD

A través de la siguiente declaratoria certifico que los conceptos desarrollados en este proyecto tanto en su análisis y en las conclusiones aquí descritas no han sido previamente presentados por ningún grado o calificación profesional y son de exclusiva responsabilidad de mi autoría.

Guayaquil, Enero del 2013

Gabriel Ullauri García

AGRADECIMIENTOS

Mis agradecimientos a Dios por brindarme sabiduría para poder realizar este proyecto y por tener la opción de haber terminado mis estudios superiores en la Universidad Politécnica Salesiana que me orientó a no solo ser un profesional sino también a ver la parte humana de esta; y sobre todo a los excelentes docentes que la hacen sobresalir de las demás.

A mi director de tesis, Ing. Adrian Arce por su esfuerzo y dedicación, quien con sus conocimientos, su experiencia, su paciencia y su motivación ha hecho posible que pueda terminar este proyecto con éxito.

A todas las personas que de alguna forma me han ayudado con sus consejos, apoyo, y ánimo, mis más sinceros agradecimientos.

DEDICATORIA

A mi madre por haberme apoyado en todo momento, por sus consejos, sus valores, por la motivación constante que me ha permitido ser una persona de bien, pero más que nada, por su amor.

Índice General

Capítulo 1: Planteamiento del problema.	1
1.1 Información general.	1
1.2 Planteamiento del problema.	6
1.3 Objetivo General.	6
1.4 Objetivos específicos.	7
1.5 Justificación.	7
1.6 Variables e indicadores.	8
1.7 Hipótesis.	8
Capítulo 2 Marco Teórico.	9
2.1 ¿Qué es el Cloud Computing?	9
2.1.1 Cloud Computing bajo demanda.	12
2.1.2 Pago por uso.	14
2.1.3 Taxonomía.	15
2.1.4 Consumidor de servicio (Service Consumer).	16
2.1.5 Proveedor del servicio (Service Provider)...	17
2.1.6 Desarrollador de servicio (Service Developer).	19
2.1.7 Actualización y mantenimiento.	19
2.2 Modelos de servicios de cloud computing.	20
2.2.1 IaaS, Infrastructure as a Service (Infraestructura como servicio).	21
2.2.2 PaaS, Platform as a Service (Plataforma como servicio).	22
2.2.3 SaaS, Software as a Service (Software como servicio).	22
2.3 Modelos de implementación de cloud computing.	23
2.3.1 Las nubes privadas.	24
2.3.2 Las nubes públicas.	24
2.3.3 Las nubes híbridas.	25
2.4 Aspectos tecnológico.	25
2.4.1 Virtualización.	26
2.4.2 La virtualización completa.	29
2.4.3 Paravirtualización.	29
2.4.4 La virtualización asistida por hardware.	29
2.4.5 Las alternativas actuales en el mercado de la computación en nube.	30

2.5	Aspectos Económicos.....	36
2.6	Limitaciones de cloud computing.	37
2.6.1	Disponibilidad de servicios.	38
2.6.2	Abuso de privilegios.	38
2.6.3	Escala de recursos.	39
2.6.4	Los datos de seguridad y confidencialidad.	39
2.6.5	Eliminación de los datos.	40
2.6.6	Recuperación y copia de seguridad.	41
Capítulo 3 Estudio de Openstack para realizar nubes privadas.		42
3.1	Los componentes de Openstack.	42
3.1.1	Openstack.	42
3.1.2	Arquitectura Contextual de Openstack.	43
3.2	Introducción a Openstack.	44
3.2.1	Hipervisores.	45
3.2.1.1	Kvm.	45
3.2.1.2	Qemu.	46
3.2.1.3	Xen.	47
3.2.1.4	Lxc Linux Containers.	47
3.2.1.5	Vmware Esx/Esxi.	48
3.3	Administradores e usuarios.	48
3.4	Imágenes e instancias... ..	49
3.5	Arquitectura del sistema de Openstack.	49
3.6	Arquitectura técnica de Openstack.....	50
3.6.1	Nodo simple.	51
3.6.2	Nodo doble.	51
3.6.3	Multi nodo.	52
3.7	Diseño del sistema del sistema y configuración de red.	53
3.7.1	Diseño de la red.	54
3.8	Herramientas de gestión.	57
3.8.1	Hybridfox.	57
3.8.2	Euca2ools.	57

Capítulo 4 Implementación de una nube privada con Openstack.	58
4.1 Requerimiento del sistema.	58
4.1.2 Requerimiento del Nodo Controlador.	58
4.1.3 Requerimiento del Nodo Compute (Almacenamiento).	59
4.1.4 Requerimiento del cliente.	59
4.2 Utilización y análisis de Stackops.	60
4.2.1 Que es Stackops.	60
4.2.2 Stackops Openstack distro 0.5	60
4.2.3 Instalacion bajo consola Openstack.	61
4.3 Instalación del sistema operativo Stackops.	71
4.3.1 Nodo Controlador.	71
4.3.2 Seleccione el idioma para la instalación.	72
4.3.3 Seleccione el lenguaje de su región.	73
4.3.4 Seleccione el lenguaje de su región.	73
4.3.5 Configuración de la interfaz de red.	74
4.3.6 Dirección ip mascara, y puerta de enlace del controlador.	74
4.3.7 Sincronización de las configuraciones.	75
4.3.8 Particionamiento de los discos.	75
4.3.9 Confirmación de la partición.	76
4.3.10 Particiones del disco básico.	76
4.3.11 Instalación finalizada.	77
4.3.12 Conectar con Smart Installer.	78
4.4 Configuración del Nodo controlador con Smart Installer.	79
4.4.1 Registro de datos.	79
4.4.2 Selección de la arquitectura a implementar.	80
4.4.3 Arquitectura Nodo Doble.	80
4.4.4 Comprobación de los requisitos del software.	81
4.4.5 Topología de la red.	81
4.4.6 Configuración del volumen.	82
4.4.7 Creación de una zona.	82
4.4.8 Finalización de la Instalación.	83
4.5 Configuración del Nodo de almacenamiento (Compute) con Smart Installer	84
4.5.1 Configuración del hipervisor.	84

4.5.2	Finalización de la instalación.	85
4.6	Herramienta de gestión para la verificación de la conexión entre los nodos.	86
4.6.1	Herramienta Putty.	86
4.6.1.1	Conectividad nodo controlador con nodo de almacenamiento.	87
4.6.1.2	Conectividad nodo de almacenamiento con nodo controlador.	88
4.7	Configuración de los componentes de Openstack.	89
4.8	Descargar de las imágenes virtuales.	90
4.9	Interfaz web, horizon dashboard.	91
4.10	Creación de Instancias.	94
4.11	Conexión con la instancia.	96
	Cronograma.	99
	Presupuesto.	100
	Conclusiones.	100
	Recomendaciones.	102
	Bibliografía.	103

Índice de tablas

Tabla 1.0 Direcciones ip dentro de la red. Fuente [Autor].	56
Tabla 2.0 Requerimiento del nodo controlador. Fuente [Autor].	59
Tabla 3.0 Requerimiento del nodo controlador. Fuente [Autor].	59
Tabla 4.0 Requerimiento del cliente. Fuente [Autor].	59

Índice de gráficos

2.1 Conceptos de Cloud Computing. Fuente [35].	11
2.2 Cloud Computing bajo demanda. Fuente [36].	13
2.3 Gastos fijos y variables. Fuente: Autor.	14
2.4 Taxonomía de Cloud Computing. Fuente [37].	15
2.5 Modelo de Cloud Computing. Fuente: [Autor].	20
2.6 Infraestructura como Servicio. Fuente: [38].	21
2.7 Plataforma como Servicio. Fuente: [38].	22
2.8 Responsabilidades de los modelos de servicios de cloud computing Fuente: [39].	23
2.9 Capas de virtualización. Fuente: [40].	26
3.0 Diagrama de virtualización de un hipervisor OpenStack. Fuente: [41].	27
3.1 Tecnologías con virtualización (Intel). Fuente: Autor.	30
3.2 Amazon web services. Fuente: [23].	30
3.3 Rackspace. Fuente: [42].	31
3.4 GoGrid. Fuente: [43].	32
3.5 Salesforce. Fuente: [44].	32
3.6 Google App Engine. Fuente: [45].	33
3.7 Microsoft Windows Azure. Fuente: [46].	34
3.8 Oracle. Fuente: [47].	34
3.9 Eucalyptus. Fuente: [48].	35
4.0 Openstack. Fuente: [49].	35
4.1 Arquitectura Contextual. Fuente: [Autor].	44

4.2 Kvm Fuente: [50].	45
4.3 Qemu Fuente: [31].	46
4.4 Xen Fuente: [51].	47
4.5 Lxc Linux Containers Fuente: [52].	47
4.6 Vmware Fuente: [53] Fuente [Autor].	48
4.7 Nodo Simple [54] Fuente [Autor].	51
4.8 Nodo Doble [55] Fuente [Autor].	51
4.9 Nodo Multi nodo [56] Fuente [Autor].	52
5.0 Esquema del diseño de los servidores. Fuente [Autor].	53
5.1 Esquema de red de la nube privada con Openstack. Fuente [Autor].	54
5.2. Stackops Openstack distro 0.5. Fuente [57].	60
5.3. Instalación del nodo controlador. Fuente [Autor].	72
5.4. Registro de datos. Fuente [Autor].	79
5.5. Selección de la arquitectura a implementar. Fuente [Autor].	80
5.6. Requerimiento del hardware. Fuente [Autor].	80
5.7. Comprobación de los requisitos del software Fuente [Autor].	81
5.8. Configuración de la topología de la red. Fuente [Autor].	81
5.9. Configuración del volumen. Fuente [Autor].	82
6.0. Creación de zonas. Fuente [Autor].	82
6.1 Finalización de la instalación.	83
6.2. Configuración del nodo de almacenamiento (Compute). Fuente [Autor].	84
6.3. Configuración del hipervisor. Fuente [Autor].	85
6.4. Finalización de la instalación del nodo de almacenamiento Fuente [Autor].	86
6.5 Conectividad nodo controlador con nodo de almacenamiento.	87
6.6 Conectividad nodo de almacenamiento con nodo controlador.	88
6.7 Servicios disponibles.	89

Abstracto

AÑO	ALUMNO	DIRECTOR DE TESIS	TEMA DE TESIS
2013	GABRIEL DARIO ULLAURI GARCÍA.	ING. ADRIAN ARCE	“SERVICIO DE VIRTUALIZACIÓN DE INFRAESTRUCTURA TECNOLÓGICA BASADO EN CLOUD COMPUTING”

Este documento describe a cloud computing, y todo el aporte que pueden generar a las empresas en la actualidad, y se compromete a cambiar la forma de utilizar los recursos informáticos TI. Lo que se ha realizado en este proyecto es un prototipo de una infraestructura de cloud computing, utilizando dos servidores emulando un front-end y un nodo de almacenamiento, donde el primero se encuentran los componentes que serán administrados por el proveedor del servicio (Autor), y el nodo de almacenamiento será donde se obtendrán los recursos. Además se ha realizado la virtualización de un panel de control donde se podrá configurar el espacio virtual que se utilizara para la administración de los usuarios como un modelo de aprovisionamiento rápido de recursos.

Palabras claves

Cloud computing. Grid Computing. Virtualización. Tecnología de la información. Plataforma como servicio. Infraestructura como servicio. Software como Servicio. Nube privada, Nube pública. Hipervisor. Kvm. Qemu. Openstack.

Capítulo 1: Planteamiento del problema

1.1 Información general

Cloud computing es, sin lugar a duda, una nueva área dentro del campo de la tecnología de la información (TI). Permitiendo así el uso del almacenamiento, procesamiento, y elementos de mayor nivel, tales como los sistemas operativos o aplicaciones de software, sin tener la parte física presente. Cabe mencionar que, debido a que la información pertenece a los clientes esto no implica que los equipos también les pertenezcan; estos recursos simplemente se utilizaran como un servicio.

El término Cloud computing genera confusión debido a los múltiples aspectos del servicio que se pueden incluir. Desde el punto de vista genérico, se podría decir que cloud computing es una especie de computación escalable, adaptable y elástica con capacidades de TI, que se proporcionan como un servicio a varios usuarios.

La utilización de esta tecnología en estos tiempos significa un gran avance, ya que toda la información que mantienen los usuarios, se alojan en uno ó varios servidores en función del software, y acceden a ellos a través del servicio de internet sin necesidad de un disco duro, memoria o la utilización de un procesador, permitiendo al usuario final la configuración del espacio virtual como si tuviera físicamente los equipos.

Existen conceptos en los que el usuario es más que un soporte para los equipos, ya que es él quien utiliza y administra los sistemas y aplicaciones, con él que a diario se enfrentan, pero son las máquinas la que le ayudan a procesar las tareas y les dan las facilidades de manejar dichos sistemas, y dan solución efectiva a sus requerimientos en el menor tiempo posible.

Con este concepto surgen las demandas de nuevos servicios que van de la mano con las tecnologías en el mercado global; es aquí donde aparece el nombre de virtualización, dando así un gran aporte al conocimiento. Los sistemas de

información tradicionales fueron ejecutados principalmente en máquinas físicas que se encontraban en las empresas y ahora con la aparición de cloud computing se permite el desarrollo de los sistemas de información en máquinas virtuales que se alojan en proveedores que ofrecen este servicio. Es cuando cloud computing tiene una oportunidad muy atractiva con las pequeñas empresas y sobre todo para las empresas cuyo negocio principal no es la computación, ya que representa una oportunidad para ahorrar costos en el desarrollo de sus sistemas de información, y les permiten respaldar sus operaciones de negocios más importantes.

La adopción de cloud computing también dispara los cambios de las actividades en el proceso de desarrollo de sistemas de información. A medida que más funciones de negocios se basan en servicios en la nube, el papel de los desarrolladores de TI se someterá a una transformación significativa.

Cloud computing se encuentra en una etapa de crecimiento, sin embargo, ya está revolucionando el mundo de la tecnología ya que los recursos de TI se comparten entre un gran número de consumidores que permiten, por un menor costo, se les administren toda su información. Además ofrece recursos bajo demanda de forma dinámica, que permiten a las empresas cambiar fundamentalmente sus estrategias de tecnología.

Para algunas empresas, la decisión principal para empezar a utilizar cloud computing inicia desde lo más básico haciéndose ciertas preguntas. ¿Mi personal de TI es muy grande?, ¿Debo pagar tanto dinero por la energía que consume mi data center?, ¿Podría pagar por todas las licencias que se aplicarían a mis servicios?, ¿Por qué no dejar que alguien más administre todos estos problemas?. Estas son algunas de las tantas razones para utilizar cloud computing y dar soluciones a los usuarios, convirtiendo a esta tecnología en una herramienta tan solicitada en estos últimos años. [1].

Una vez que las empresas toman la decisión de obtener este servicio, el cual se encuentra en cualquier parte del mundo, podrán gestionar y administrar dependiendo

sus necesidades, toda su información de una forma segura, esto se definirá como virtualización. La virtualización no es más que el proceso de desvinculación del hardware con el sistema operativo en una máquina física. Una máquina virtual (VM) es la representación de un equipo físico que se ejecuta y se mantiene en un host con un monitor de máquina virtual o un hipervisor.

“Un hipervisor (*hypervisor*) o monitor de máquina virtual (*virtual machine monitor*) es una plataforma que permite aplicar diversas técnicas de control de virtualización para su utilización, al mismo tiempo, diferentes tipos de sistemas operativos (sin modificar en el caso de paravirtualización) son permitidos en una misma computadora. Es una extensión de un término anterior, “supervisor”, que se aplicaba al kernel del sistema operativo.”[2].

En la actualidad, el uso del internet cada día va en aumento; las aplicaciones siguen evolucionando, la vida de las personas y sus trabajos están tan integrados a la red informática. Ahora es muy fácil para todos llevar a cabo negocios en línea; la web no es sólo un medio de comunicación sino también una plataforma para empresas y las sociedades. Para llegar a la gran cantidad de usuarios o posibles usuarios, las empresas tienen que ofrecer sus contenidos y procesos empresariales en línea.

En general, la aparición de cloud computing ha proporcionado a las empresas la oportunidad de revisar sus infraestructuras de información tecnológica TI. Algunos proveedores de cloud computing son capaces de ofrecer una infraestructura de TI basados en suscripciones con un modelo de servicios que es considerado de utilidad para las futuras empresas.

En esta nueva tecnología de sistemas se incluyen elementos tales como el procedimiento, la parte de la red y su almacenamiento. Por lo tanto, la arquitectura de la nube puede ser visto desde tres diferentes modelos: infraestructura, plataforma y aplicación. En función de éstos, los servicios en la nube se pueden dividir en diferentes categorías: infraestructura como servicio (*IaaS*), plataforma como servicio (*PaaS*), y software como servicio (*SaaS*). En el modelo de *IaaS*, los recursos de

hardware como el servidor de datos y de cálculo (por ejemplo, CPU) se ofrecen como servicios a los clientes. En lugar de invertir dinero en los servidores dedicados y equipos de redes, los clientes pueden alquilar el servicio de “*IaaS*. Los proveedores de *IaaS* pueden asignar más espacio y recursos de hardware a las aplicaciones según sea necesario [3]. Con esta flexibilidad, las empresas pueden aumentar la eficiencia del uso de los recursos de TI y reducir así sus costos.

En el modelo de *PaaS*, los proveedores de las nubes brindan las facilidades para apoyar el ciclo de vida completo del desarrollo de las aplicaciones, incluyendo el diseño, implementación, depuración, pruebas y despliegue de aplicaciones web. El consumidor no tiene que administrar o controlar la infraestructura de la nube, redes, servidores, ni sistemas operativos, o de almacenamiento, pero el consumidor si tiene control sobre las aplicaciones implementadas.

En el modelo *SaaS*, las aplicaciones de software se ofrecen como servicios en la web en vez de paquetes de software para ser comprados por clientes individuales además permiten a los proveedores, desarrollar, alojar y operar el software para uso y facilidad del cliente.

Cloud computing está ganando popularidad debido a que confronta el problema de muchas empresas tradicionales con respecto a la computación, en la que las empresa dedica tiempo y dinero en implementar una infraestructura, ya sea con personal o en un data center. El modelo tradicional de la computación por lo general conduce a la subutilización y el despilfarro de los recursos de TI, debido a la fragmentación de recursos y la distribución desigual de la carga de trabajo. Cloud computing permite a las empresas virtualizar su infraestructura de TI, y abordar cualquier tipo de problema que esto conlleve.

Definitivamente las empresas y sus empleados en la actualidad, viven rodeados con esta nueva tecnología desde el momento que acceden a sus cuentas de correo ya son parte de cloud computing. Se puede citar algunas compañías que ya están siendo parte de la cloud computing. Se puede citar algunas compañías que ya están siendo

parte de la nube por ejemplo google apps [5], Microsoft Azure, [4], Ebay [6], Amazon Elastic Cloud Computing [7], Salesforce [8].

La compañía que se ha mencionado Amazon Elastic Cloud, ofrece el servicio de la nube denominado “Amazon Web Services” el cual brinda a las compañías, almacenamiento de datos y a su vez les permiten crear una infraestructura para la creación de dicha tecnología.

En resumen lo que se desea realizar en este proyecto es la creación de un prototipo de una infraestructura de cloud computing virtualizando los mecanismo de una nube privada utilizando openstack [9], este es un software de código abierto que las empresas y proveedores de servicios pueden utilizar para configurar y ejecutar su nube e infraestructura de almacenamiento; para la implementación de este prototipo de la nube privada se decidió tomar esta aplicación. Se virtualizarán los recursos de hardware para crear las maquinas virtuales, con la cual el usuario final ingresará, y podrá crear otras aplicaciones.

En concreto, se diseñara un prototipo de infraestructura como servicio con tamaño de porción variable para facilitar el uso y para realizar las creaciones de imágenes y espacio virtual se ha decidido utilizar una herramienta de gran ayuda como un hipervisor suficientemente compatible con “stackops” que es Qemu. Se Utiliza un sistema de credenciales para la autenticidad de los archivos y de las imágenes que serán generadas desde el nodo controlador, además no serán permitidos protocolos más que los necesarios para poder tener el control de flujo de las aplicaciones que el usuario requiera. Las imágenes serán generadas en el nodo controlador que será el front-end y desde allí se podrá realizar pruebas al momento de tener el espacio virtual.

1.2 Planteamiento del problema.

La aparición de cloud computing cambia la forma de ver los sistemas convencionales de las empresas, los datos ya no estarán físicamente almacenados en un conjunto de equipos y discos específico, más bien el concepto de este sistema varía ya que el lugar de los datos se han convertido en algo difuso y están distribuidas geográficamente en lugares desconocidas por las empresas. La deducción lógica para las empresas es tener toda su información bajo control: Así que, como en la mayoría de ellas es importante mantener la preocupación de que exista compañías que puedan ofrecer servicios tecnológicos, que sean capaces de brindar confiabilidad y sobre todo seguridad en el momento de empezar a prestar el servicio de cloud computing.

Lo primero que se debe ver es la limitante de mantener toda la información en un data center y lo que ello conlleve, además se debe ver desde varias perspectivas técnicas las ventajas y desventajas de invertir en recursos físicos y humanos para obtener los mejores beneficios de disponibilidad y rendimiento de la información.

Esta tesis se referirá especialmente a la realización de un prototipo de una infraestructura tecnológica basada en cloud computing utilizando openstack y poder ser orientada a su utilización, quien ofrece sus servicios a través de Amazon EC2/S3 compatible con API y por lo tanto se puede utilizar esta herramienta para una implementación real de cloud computing [10]. Al final se desea dar otra alternativa a lo tradicional, al momento de que se requiera contratar un servicio de almacenamiento de datos, con lo cual queda el precedente de un problema en la actualidad para las compañías.

1.3 Objetivo General.

Implementar un prototipo de infraestructura como servicio y obtener el control de toda la información alojada en ella, accediendo desde cualquier lugar y en cualquier momento que se requiera por medio de la red.

1.4 Objetivos específicos.

- Implementar un prototipo de nube privada para ofrecer infraestructura como servicio mediante el sistema operativo Linux (Openstack).
- Analizar la tecnología de cloud computing.
- Sugerir a futuro la implementación de esta tecnología a la Universidad Politécnica Salesiana.

1.5 Justificación.

El éxito del cloud computing está relacionado con el servicio que ofrecen y las oportunidades de negocio que se activan. El objetivo de esta tesis será de investigar y demostrar que mediante la implementación de un prototipo de una infraestructura como servicio es capaz de ser administrada dependiendo a los requerimientos de cada empresa.

Esta tesis se centro en examinar la tecnología de cloud computing que representa la evolución de la virtualización de los datos y toda la información, esto permitirá la interacción entre las empresas y los proveedores que ofrecen este servicio bajo demanda. Se justifica la realización de este proyecto al cambio constante que la tecnología realiza, y en la actualidad simplifica muchos los recursos en función de sus necesidades al disponer de cloud computing; ampliando así la gran capacidad de procesamiento y almacenamiento de datos sin necesidad de instalar más infraestructura, ayudando a las empresas a escalar rápidamente, sin tener que agregar software ni personal extra.

1.6 Variables e indicadores.

Cloud computing se basa en un nuevo modelo de entrega y consumo de servicios de IT a las empresas: Este modelo permiten independizarse de las tecnologías utilizadas en la actualidad, presentando así mejores beneficios a las compañías que adopten esta nueva tecnología. Las variables a tomar en consideración para utilizar un modelo de negocio de cloud computing serán las siguientes:

Confiabilidad.- Se dará esta variable en el momento de implementar el prototipo de la nube privada al manejar información de los clientes que se brindan con este servicio.

Consultas.- Se consultara sobre las ventajas y desventajas y la diferencia de la implementación de un centro de datos y una nube privada, a empresas que en la actualidad están transfiriendo sus datos a una centro de datos.

Recurso Físico.- El recurso físico para la implementación será un variable importante para la estructura del prototipo, para crear las instancias es necesario mantener los requisitos recomendados.

1.7 Hipótesis.

Cloud computing gestiona de mejor forma los recursos de las empresas capaces de disminuir y simplificar los problemas que generan los centro de datos. Por lo tanto, nuestra hipótesis es:

H1.- Al utilizar la infraestructura como servicio de cloud computing las pequeñas y medianas empresas de Guayaquil disminuirán dramáticamente sus costos de implementación de TI, al no tener que realizar instalación, mantenimiento y administración de equipos físicos.

Capítulo 2 Marco Teórico.

2.1 ¿Qué es el Cloud Computing?

“Cloud computing es fiable, precisa y fundamentalmente puede mantener los datos y aplicaciones bien seguros utilizando los servidores virtualizados por medio de internet. Permitiendo a los consumidores y a las empresas utilizar las aplicaciones sin necesidad de instalar y acceder a sus archivos personales en cualquier computadora, con acceso a Internet. Esta tecnología permite a las empresas ser más eficiente, en almacenamiento, en memoria, en procesamiento y el ancho de banda”. [11].

En la última década la informática ha realizado un cambio significativo, pasando a ser un modelo de servicios de consumo masivo, ofreciendo una tecnología bajo el esquema de pago bajo demanda. En este nuevo modelo, se están desarrollando millones de aplicaciones las cuales los usuarios pueden acceder a través de internet y no mediante máquinas físicas. Cloud computing no es más que una forma donde las aplicaciones son expuestas en internet como servicios a los cuales cualquier persona con autorización puede consumirlos y aprovechar todas las ventajas que ella brinda.

Desde mucho tiempo se ha utilizado el término “Cloud” para indicar una infraestructura por la cual los usuarios tienen acceso a varias aplicaciones desde cualquier lugar del mundo bajo demanda, lo que conocemos más comúnmente como internet. Hay conceptos que nos indican que la nube es una seria alternativa de cómo confiar nuestros datos a una infraestructura abstracta y compleja donde la información fluye de un lado a otro sin importar lo que existe en medio. Aun cuando existe una estrecha relación entre esta metáfora bien conocida por todos y el verdadero significado del término cloud computing, es importante hacer énfasis en una definición precisa y certera de lo que se quiere expresar.

“Podríamos definir dos conceptos precisos para el término cloud que serían: Abstracción y virtualización. La abstracción nos hace olvidar los detalles de la implementación por parte de los usuarios y los desarrolladores de TI, tomando este concepto desde un enfoque en donde las aplicaciones se ejecutan sobre una máquina

física que no está especificada, los datos y nuestros archivos son almacenados en ubicaciones desconocidas, la administración de los sistemas está bajo responsabilidad de una tercera persona y finalmente los usuarios tienen acceso a esta infraestructura desde cualquier otro lugar, con tan solo tener acceso a la red. En cuanto a la virtualización se refiere a la habilidad del sistema para crear sistemas que parezcan independientes ante los usuarios a través de mecanismos de compartir y asignar periodos de uso a los recursos que cada unidad necesite”. [12].

Esta nueva tecnología de la nube de computación es la descripción básica de compartir recursos físicos y hacer una presentación ante el usuario final como recursos independientes a través de la virtualización. Cuando mencionamos cloud computing se hace un punto de partida a los servicios y a todas las aplicaciones que se procesan en una red distribuida usando recursos virtualizados y que se puede acceder a través de protocolos comunes de internet y estándares de comunicación en redes. Como usuario lo más importante es poder administrar los datos en la nube y no la infraestructura física que hay detrás, y suponemos que los recursos con los que podemos contar en la nube son ilimitados y podemos acceder en cualquier lugar que se necesite de ellos.

“Para entrar en una descripción más detallada tenemos dos modelos que definen cloud computing: a partir del lugar donde esté ubicada y la manera cómo se administra la infraestructura (Modelo de Despliegue), a partir del tipo de servicios a los que se puede acceder en la plataforma (Modelo de Servicio). A partir de estos dos modelos es que NIST (*National Institute of Standards and Technology*) da una definición formal y que puede ser resumida en la siguiente figura, teniendo en cuenta que esta definición se está moviendo hacia una interacción de componentes basados en estándares como SOA (*Service Oriented Architecture*) por lo que en un futuro incluirá nuevos componentes” [13].

Todas las personas en esta industria desde los expertos hasta los proveedores de la nube, tienen su propio concepto de lo que es computación en la nube. Hoy por hoy no existe un concepto de lo qué es exactamente, y lo que esto significa a largo y corto plazo. Examinar algunas de las definiciones existentes ayuda a clarificar el

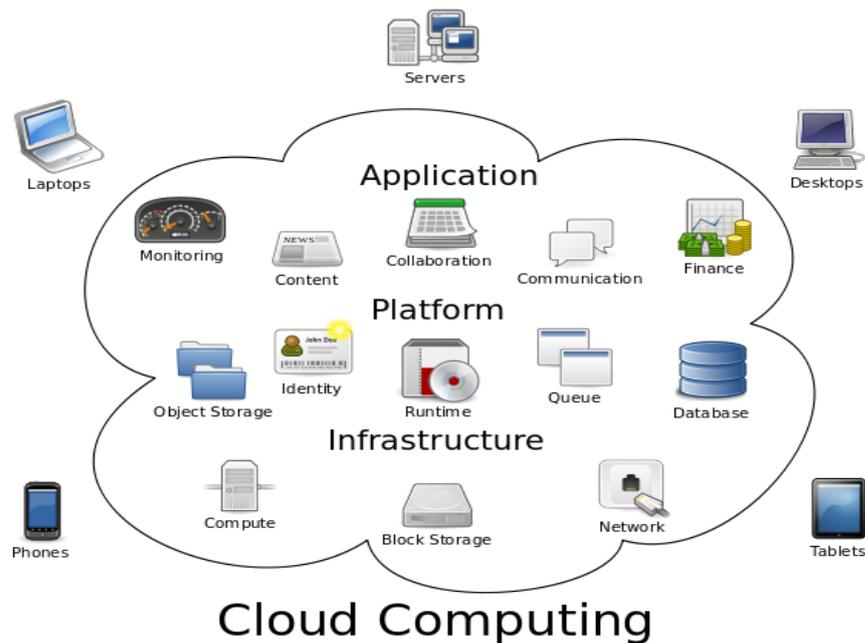
Existen varias tecnologías principales que han conducido al desarrollo del paradigma de cloud computing, una de ellas es la virtualización y la otra que complementa, y son partes principales es *Grid Computing*. La primera será el cual nos dedicaremos a analizar e implementar en este proyecto así que nada más se añadirá en esta investigación. Esta última, *grid computing*, se refiere a la aplicación de los recursos de muchos ordenadores en una red, pero que resuelve varios problemas al mismo tiempo, quien lo introdujo por primera vez *Foster y Kesselman* al principios de los noventa y presentada formalmente por ellos en un libro en 1999.

La *grid computing* se utiliza normalmente para hacer frente a problemas científicos o técnicos que requieren un gran número de ciclos de procesamiento de las computadoras, o que involucran grandes cantidades de datos.

La diferencia entre este paradigma y el cloud computing es que los sistemas de la red han sido diseñados de manera que el usuario puede solicitar y consumir grandes fracciones, del grupo de recursos totales, mientras que en los sistemas de las nubes las peticiones de los usuarios individuales están limitadas a pequeñas fracciones de la capacidad total del sistema. Como consecuencia de los usuarios con fracciones muy pequeñas de la capacidad total, el cloud computing se ha centrado en la ampliación de manejar un gran número de usuarios.

2.1.1 Cloud computing bajo demanda.

Todos los proveedores de cloud computing deben cumplir como condición básica la facilidad de otorgar los recursos informáticos almacenados cada vez que el cliente lo solicite. Desde el punto de vista del cliente los recursos informáticos disponibles en la nube son casi infinitos es decir, el cliente no limita el conjunto de servidores ubicados en un sitio, y es total responsabilidad del proveedor de cloud computing de contar con recursos suficientes para satisfacer las necesidades de todos sus los clientes, lo que el cliente pagara por el servicio que va a utilizar.



Cloud Computing

2.2 Cloud Computing bajo demanda.

Fuente: <http://es.wikipedia.org/wiki/PuTTY> [36]

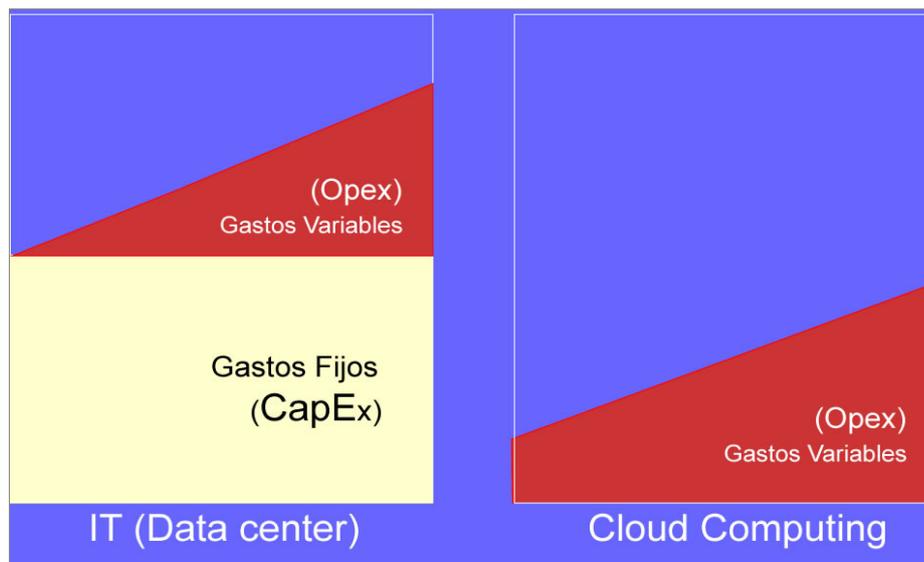
La adquisición de un espacio virtual de cloud computing, es una de las capacidades más deseada para un gran número de compañías actualmente, ya que elimina la necesidad de planificar el futuro, la compra e instalación de los recursos que se requiere en algún momento en el futuro. Esto permite al cliente evitar hacer una inversión inicial innecesaria comprando servidores. Al comparar un espacio en la nube con el modelo tradicional de propiedad de los servidores, la computación en la nube le ayuda a evitar los costos de tener recursos subutilizados y el proveedor de cloud computing hace lo que las empresas deberían hacer.

Muchas de las característica de los recursos de la informática bajo demanda inducen a las empresas a una entrada de algunos modelos de negocio, como proveedores de software puedan desarrollar aplicaciones sin preocuparse de antemano de los suministros para un número determinado de clientes y teniendo con el riesgo de un mayor éxito de lo previsto, lo que el servicio no está disponible o, peor aún, tener muy pocos usuarios, y un gasto de capital causado por la compra de recursos que son muy poco utilizados.

2.1.2 Pago por uso

Algo que llama mucho la atención de esta tecnología es la aplicación de un modelo de uso de facturación esto quiere decir que el cliente sólo paga por el uso a corto plazo de los procesadores y almacenamiento si este fuera el caso, por dar un ejemplo, este uso puede ser medido en incrementos de horas o días, convirtiéndolo así en capital (CAPEX¹), y no en gastos operativos (OPEX²).

Gran parte del concepto de cloud computing se relaciona con la de *utility computing*³. En estos dos casos los recursos almacenados se están proporcionando bajo demanda, como lo es la energía eléctrica, el agua potable, televisión por cable. A diferencia de un contrato de alquiler común, donde estos recursos se encuentran físicamente en las localidades del cliente, en el caso de cloud computing los recursos están simplemente en la nube y no nos debe importar el lugar ni la ubicación.



2.3 Gastos Fijos y Variables.

Fuente: Autor

¹ **Capex.**- Son erogaciones o inversiones de capital que crean beneficios

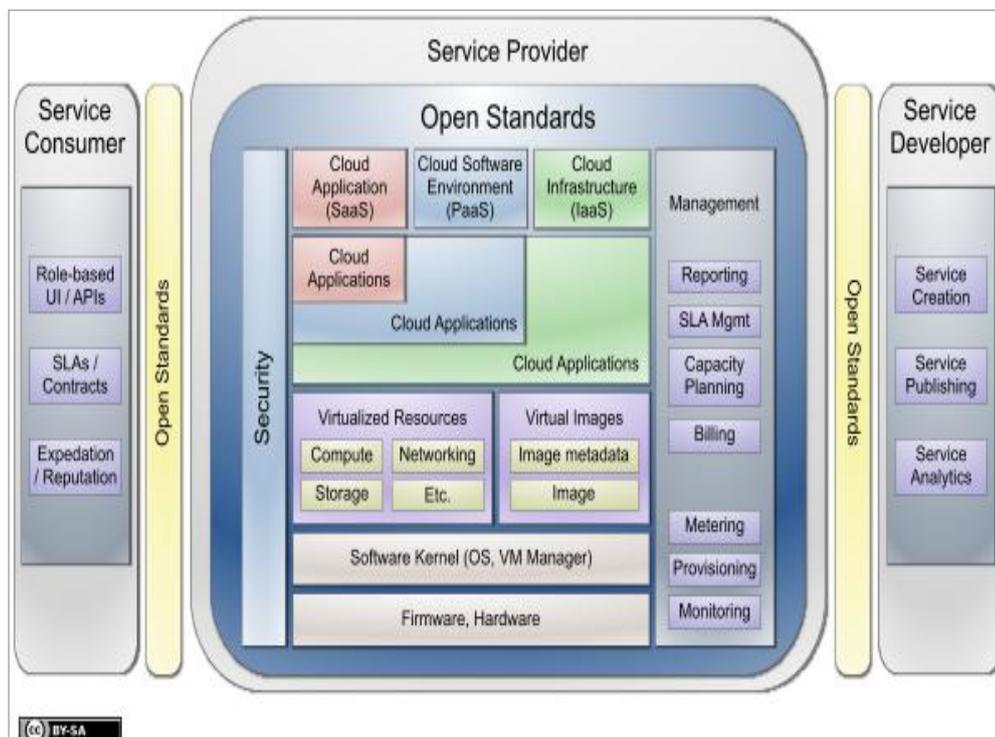
² **Opex.**- Es una herramienta para el cálculo de gastos operativos para la gerencia de servicios técnicos

³ **Utility Computing** se define como el suministro de recursos computacionales, como puede ser el procesamiento y almacenamiento, como un servicio medido similar a las utilidades públicas tradicionales (como la electricidad, el agua, el gas natural o el teléfono).

En la tabla de la gráfica 2.3 se la realizo en base a un sondeo a la empresa IBM en el Ecuador que a cierto modo presta servicios de consultoría y nos da una guía clara de los gastos fijos y variables al querer implementar una nube privada, también se realizó consultas al banco de Guayaquil que actualmente está en el proyecto de implementación de pasar sus servidores al data center que la empresa TELCONET está ofreciendo a las miles de empresas en el país.

2.1.3 Taxonomía

La taxonomía para proporcionar una visión general a este proyecto de computación en la nube es en base a un desarrollo estándar que todas las empresas proveedoras de esta tecnología emplean, y la definiríamos de la siguiente manera:



2.4 Taxonomía de Cloud Computing.

Fuente: <http://outsourcendo.blogspot.com/2011/05/jornada-clud-computing-en-esic-madrid.html> [37]

Sin embargo, el enfoque de lo que se plantea realizar en este proyecto es definir el escenario de las nubes y los casos de uso basado en aplicaciones en el mundo real.

2.1.4 Consumidor de servicio (*Service Consumer*).

Este concepto de consumidor del servicio no es más que la empresa o el cliente que en realidad utiliza el servicio de la nube, ya sea de software, plataforma o infraestructura como un servicio como tal. Dependiendo del tipo de servicio que se requiere y de su papel, el consumidor funciona con diferentes interfaces de usuario e interfaces de programación. El cliente no necesita saber acerca de esta tecnología, ya que utilizan las aplicaciones y otras interfaces parecidas a la de su máquina de escritorio. Pudiendo realizar tareas tales como iniciar y detener las máquinas virtuales o la gestión de almacenamiento en la nube.



Taxonomía de Cloud Computing.

Fuente: <http://outsourcando.blogspot.com/2011/05/jornada-clud-computing-en-esic-madrid.html> [37]

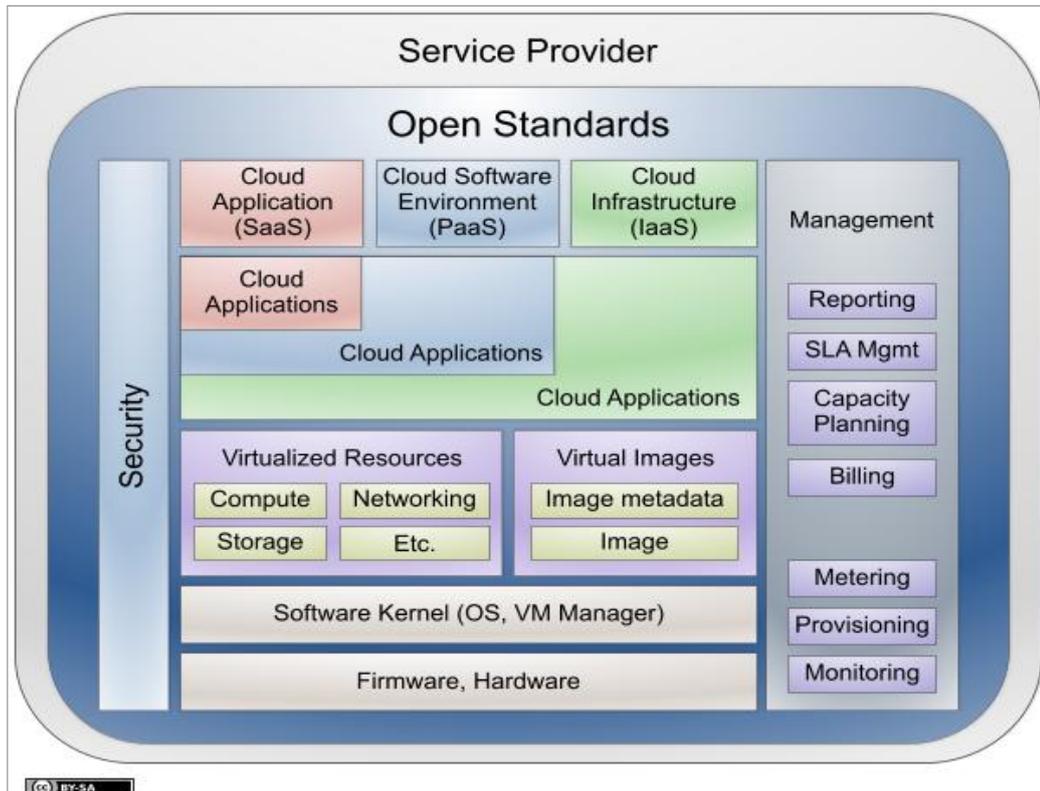
Los consumidores trabajan con los SLAs⁴ y sus contratos, normalmente estos se negocian a través de la intervención humana entre el consumidor y el proveedor de la nube. Las expectativas del consumidor y la reputación del proveedor son una de las partes más importantes entre las negociaciones.

El desarrollador del servicio de la nube es el encargado de crear, editar y supervisar los servicios que ofrecen. En la mayoría de los casos, las aplicaciones que se entregan directamente a los clientes finales a través del modelo de servicio “*SaaS*”. Las aplicaciones escritas en IaaS (*Infrastructure as a Service*), y los niveles de PaaS (*Platform as a Service*) serán posteriormente utilizados por los desarrolladores y proveedores de *SaaS* en la Nube. [18]

⁴ SLA (Service Level Agreement) es un documento habitualmente anexo al contrato de prestación de servicios.

2.1.5 Proveedor del servicio (Service Provider).

Software como servicio (SaaS), el proveedor instala, administra y mantiene el software. El proveedor no es necesariamente la propia infraestructura física en la que se ejecuta el software. En cualquier caso, el consumidor no tiene acceso a la infraestructura, sólo tiene acceso a la aplicación requerida.



Taxonomía de Cloud Computing.

Fuente: <http://outsourcendo.blogspot.com/2011/05/jornada-clud-computing-en-esic-madrid.html> [37]

Plataforma como servicio (*PaaS*), el proveedor se encarga de gestionar la infraestructura para la plataforma en la nube, por lo general un framework⁵ para un tipo específico de aplicación. La aplicación para el consumidor nunca podrá ingresar a la infraestructura por debajo de la plataforma.

⁵ Framework (marco de aplicación) es un conjunto de bibliotecas orientadas a la reutilización.

Infraestructura como un servicio (*IaaS*), el proveedor mantiene firme el almacenamiento, las base de datos, middleware⁶, y el entorno de alojamiento de las máquinas virtuales. El consumidor solo se encarga de utilizarlo sin necesidad de acceder a la infraestructura que lo aloja.

En este diagrama, la última capa está el firmware y hardware quien es la base para todo lo demás. Por encima, esta el núcleo del software (Kernel⁷), ya sea el sistema operativo o gestor de la máquina virtual que aloja la infraestructura por debajo de la nube. Los recursos virtuales y las imágenes son los servicios básicos de computación en la nube como la potencia de procesamiento, almacenamiento y middleware. Las imágenes virtuales controladas por openstack incluyen tanto las imágenes de sí mismos como la de los usuarios.

La capa de gestión, es fundamentalmente crucial para realizar las operaciones del proveedor de servicios. Esta capa de gestión se encarga de realizar la mediciones para determinar quién utiliza los servicios y hasta qué punto, el aprovisionamiento para determinar la asignación de recursos para los consumidores, y la vigilancia para el seguimiento del estado del sistema y sus recursos.

La gestión de facturación se encuentra en un nivel superior, el que implica recuperar los costos, la planificación de capacidad para asegurar que las demandas de los consumidores sean establecidas, gestión de SLA para asegurarse de que los términos de servicio acordado por el proveedor y el consumidor se cumplan, y los informes para los administradores sean creados según los requerimientos.

⁶ Middleware es un software de conectividad entre la capa de aplicaciones y las capas inferiores (Sistema operativo y red), que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.

⁷ El kernel ó núcleo se puede definir como el corazón de un sistema operativo. Es el encargado de que el software y el hardware de tu ordenador puedan trabajar juntos.

Con lo que a seguridad se refiere se debe ejecutar todos los recursos que el proveedor de la nube nos ofrezca para tener todos los datos seguros.

2.1.6 Desarrollador de Servicio (*Service Developer*).

En este nivel el desarrollador de servicios se encarga de crear, editar y supervisar la nube de servicio. Estas aplicaciones son entregadas directamente a los usuarios finales a través del modelo *SaaS*. Los entornos del desarrollo para la creación de servicios no siempre son fijos tienden a variar. Si los desarrolladores a crear una aplicación *SaaS*, están escribiendo su código es más probable que para un entorno hospedado sea realizado por un proveedor de la nube. En este caso, la publicación del servicio es el despliegue de la infraestructura del proveedor de la nube durante la creación de servicios, y el previo análisis de involucrar a la depuración remota para poner a prueba el servicio antes de su publicación para el consumidor. Una vez que el servicio es publicado, el análisis permite a los desarrolladores supervisar el rendimiento de su servicio y hacer los cambios necesarios.



Taxonomía de Cloud Computing.

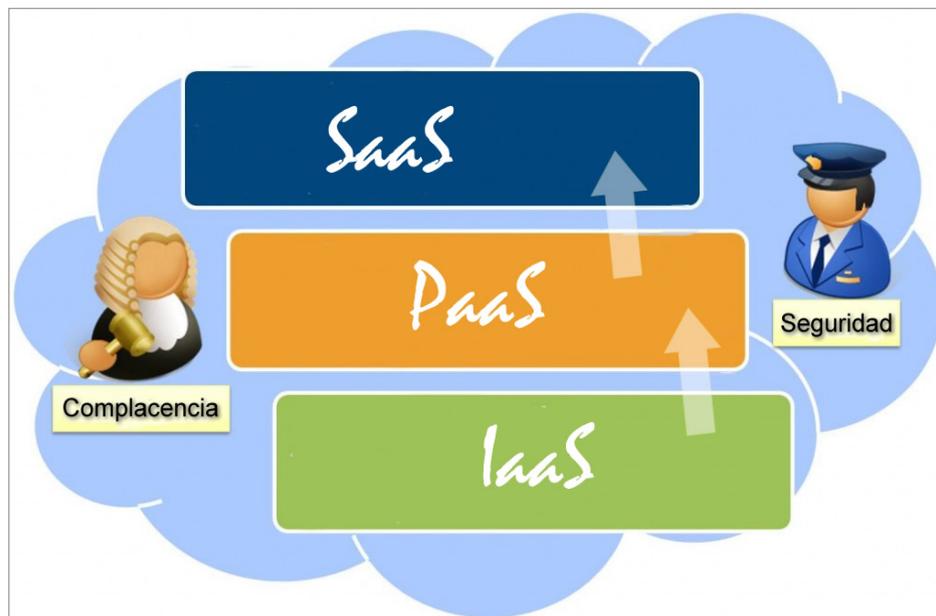
Fuente: <http://outsourcando.blogspot.com/2011/05/jornada-clud-computing-en-esic-madrid.html> [37]

2.1.7 Actualización y Mantenimiento.

El proveedor de la nube es quien mantiene los recursos tanto de hardware como de software actualizado y otorgando más recursos físicos: Además de darle mantenimiento a la infraestructura y a los equipos donde el usuario mantiene todos sus datos, existe una externalización eficaz de las tareas de estos mantenimientos por lo tanto, todas las reparaciones y la sustitución de los recursos de hardware subyacentes son transparentes para el cliente no son afectados al momento de realizar las actualizaciones.

Si bien esto puede ser cierto en el caso ideal, existe la posibilidad de existir intervalos cortos de retraso al momento de migrar desde la plataforma de hardware a otra con el fin de realizar tareas de mantenimiento o reparación de una plataforma física determinada, en este corto período de tiempo el cliente no reflejara algún tipo de molestia en sus trabajos ya que existe al momento de la migración registros de respaldos asociados a las imágenes disponible que mantiene el usuario final.

2.2 Modelos de servicios de Cloud Computing.



2.5 Modelo de Cloud Computing.
Fuente: [Autor].

Todas las empresas pueden beneficiarse de la nube en diferentes formas. A continuación se detallara las diferentes formas en que las compañías pueden utilizar los diferentes servicios, así como la forma en que algunas organizaciones actualmente están haciendo justamente eso.

Normalmente en las todos los libros y documentación en la web primero se habla de estos servicios desde la capa más alta que es *SaaS* por su importancia desde la aplicación hasta su infraestructura, sin embargo en esta documentación se empezara a hablar desde la capa base que es *IaaS* por ser el servicio al que vamos a analizar e implementar.

Es importante tener en cuenta, como se muestra en la Figura 2.5, que estos modelos de servicios están basados en las siguientes capas: *SaaS*, *PaaS*, y el *IaaS*. Por lo tanto, no se trata de un enfoque de exclusión de la clasificación, sino que se refiere al nivel del servicio prestado, cada uno de estos modelos de servicio se describe a continuación.

2.2.1 *IaaS* (Infraestructura como Servicio)



2.6 Infraestructura como Servicio.

Fuente: <http://manuelvieda.com/2011/07/cloud-computing-que-es/> [38].

Es la capacidad de entregar al cliente los recursos, el espacio de almacenamiento, o la red con la que el cliente puede ejecutar un sistema operativo, aplicaciones, o cualquier software que ellos deseen. El cliente no será capaz de controlar ni manipular la distribución del software de una plataforma de hardware específica o cambiar los parámetros en la infraestructura subyacente a lo requerido, pero sí podrá gestionar y personalizar sus aplicaciones.

Es decir, se utiliza un servidor virtualizado y este a su vez ejecuta el software del usuario final. Uno de los más frecuentes en prestar este servicio es la de amazon elastic compute cloud (EC2). [19].

2.2.2 PaaS (Plataforma como Servicio)

En el caso de PaaS, el proveedor de la nube no sólo ofrece el hardware, sino que también proporcionan un conjunto de herramientas y una serie de lenguajes de programación soportados para crear servicios de mayor nivel (por ejemplo, aplicaciones de software que están disponibles como parte de una plataforma específica) que puede ser desarrollada y al mismo tiempo alojada. Los usuarios de *PaaS* son típicamente los desarrolladores de software que son aplicables a la plataforma y las ofrecen a los usuarios finales.



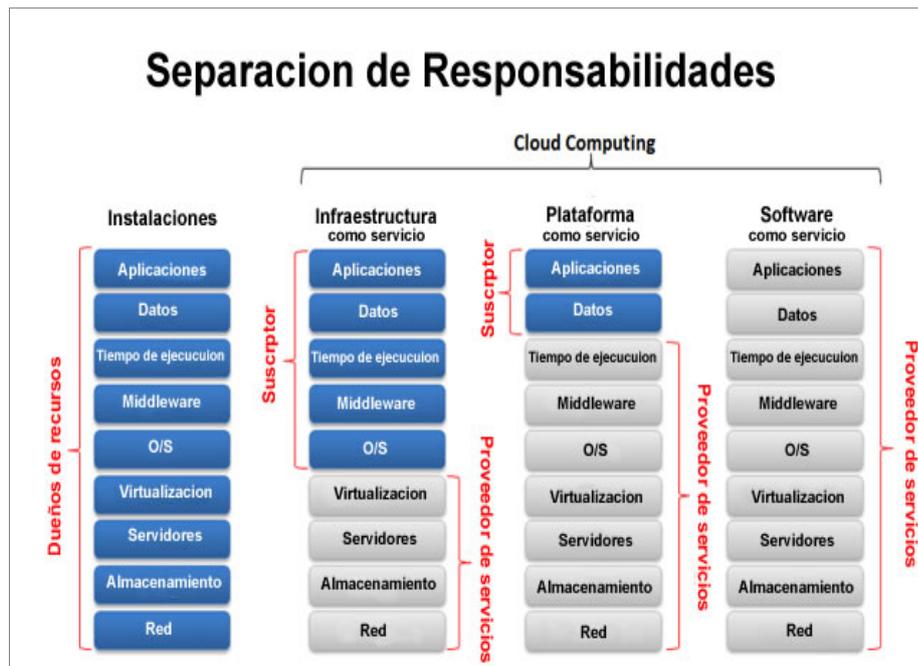
2.7 Plataforma como Servicio.

Fuente: <http://manuelvieda.com/2011/07/cloud-computing-que-es/> [38].

2.2.3 SaaS (Software as a Service)

El cliente *SaaS* es un usuario final de aplicaciones completas que se ejecutan en una infraestructura de la nube y se ofrece para la plataforma bajo demanda. Las aplicaciones son generalmente accesibles a través de una interfaces amistosas, como un navegador web. El cliente tiene control de los parámetros de las aplicaciones para su configuración específica y administración constante. Este servicio está relacionado con las demás compañías el hecho de compartir la misma infraestructura con las demás organizaciones, compartiendo el valor de la licencia entre todos de manera equitativa para su uso.

La figura 2.8 muestra cada una de las responsabilidades de los modelos de servicios de cloud computing, los tres bloques de servicio que ofrece la nube de computación al cliente les asigna control al momento de realizar la contratación. El primer bloque, el *IaaS* se encarga de dar soporte hasta los sistemas virtualizados, el soporte se ejecuta a todas las aplicaciones del suscriptor.



2.8 Responsabilidades de los modelos de servicios de cloud computing
Fuente: <http://blogs.cisco.com/security/penetration-testing-in-the-cloud/> [39].

2.3 Modelos de implementación de Cloud Computing.

Cloud computing también puede ser clasificada en base al modelo de despliegue de la infraestructura subyacente, como públicas, privadas, o híbridas. Los modelos de las diferentes infraestructuras de implementación se distinguen por su arquitectura, la ubicación del centro de datos donde se realiza la nube, y las necesidades de los clientes. Se pueden mencionar los tres modelos básicos a considerar, los cuales pueden ser de origen propietario, o basados en software libre.

2.3.1 Las nubes privadas

Las nubes privadas son creadas exclusivamente para el uso de un usuario final, quien es propietario maneja completamente esta nube. Además, hay variaciones de este en términos de propiedad, operación, y configuración. Se determina que la nube sea usada por un solo usuario en particular para que sea distintiva de cualquier nube privada.

Una nube privada puede ser propiedad del cliente, pero se construye, se instala y se gestiona por una tercera persona y no por el cliente. Los servidores físicos pueden estar ubicados en las instalaciones del cliente o ubicada en un centro de datos que por lo general el cliente nunca sabrá donde están físicamente los equipos.

Una alternativa recientemente introducida por los proveedores de este servicio es crear una nube privada virtual. Una nube privada virtual está constituida por una nube privada dentro de la infraestructura física de una nube pública. Debido a la asignación de recursos específicos dentro de la nube, el cliente asegura todos sus datos almacenados en la nube privada y solo realiza las tareas en los servidores dedicados o nube virtual (es decir estos servidores no se comparten con ningún otro cliente del proveedor de la nube).

2.3.2 Las nubes públicas

Es la infraestructura física como tal, una nube pública es propiedad de un proveedor de servicios cloud. Como una nube ejecuta las aplicaciones de varios clientes que comparten la misma infraestructura deben pagar por su utilización de los recursos más no por la administración de los datos. Las empresas pueden utilizar la funcionalidad total de sus aplicaciones respectivamente y ofrecer sus propios servicios a los usuarios finales. Proporcionar al usuario la capacidad real para aprovechar las características de la nube para sus propios fines permitiendo también a otras empresas externalizar sus servicios a los proveedores de la nube, reduciendo así los costos y el esfuerzo para construir su propia infraestructura.

2.3.3 Las nubes híbridas

Por último, toda la composición de las nubes, ya sean públicas o privadas, podría formar una nube híbrida y manejar una sola entidad, siempre que exista coincidencia suficiente entre el estándar utilizados por las nubes constituidas.

Al permitir a las empresas externalizar las nubes públicas y privadas y gran parte de su gran infraestructura los proveedores de la nube perderían el control sobre los recursos y la gestión de la distribución de código y datos. En muchos casos, esto no es deseado por las organizaciones que brindan estos servicios de cloud computing.

Las nubes híbridas realizan un trabajo de interacción de las infraestructuras de cloud públicas y cloud privadas a fin de llegar a una gran reducción de costos, mediante la contratación externa, mientras que mantener el mismo grado de control sobre los datos más importantes y confidenciales mediante el empleo de las nubes privadas locales.

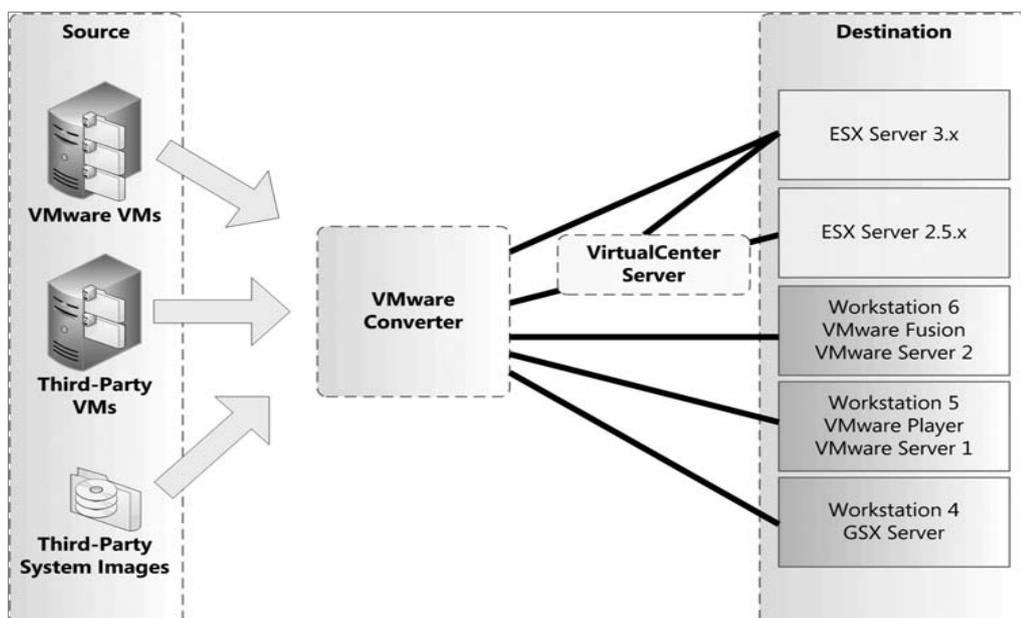
2.4 Aspecto tecnológico.

La característica más importante que puede ser distinguida en cloud computing, aparte de los modelos de servicios anteriores mencionado es la de ser multiusuario teniendo la capacidad de alojar a diferentes usuarios permitiéndole operar en las mismas características físicas de hardware software ya que las realiza utilizando la tecnología de virtualización.

Utilizar la Virtualización para ejecutar múltiples clientes de cloud computing es un retorno a lo que IBM hizo en 1972 con su sistema VM/370, la introducción al mercado del compartimiento en un ordenador central el Sistema 370 de IBM fue capaz de proporcionar a sus usuarios múltiples tareas al mismo momento de ejecución [20].

2.4.1 La Virtualización

Lo que realmente da rentabilidad a las nubes es la demanda de las organizaciones al adquirir un espacio para alojar toda su información en una forma dinámica de máquinas virtuales. Esto permite al proveedor de servicios de la nube compartir la infraestructura ubicada en un centro de datos entre varios clientes. El nivel de la Virtualización dependerá de lo que las compañías requieran y lo que el proveedor haya ofrecido y esto dependerá de cuál de los tres modelos de servicio (*SaaS*, *PaaS* o *IaaS*) requiera el usuario o compañía. La Virtualización se refiere estrictamente a la optimización de los recursos de la máquina real usando máquinas virtuales; las implementaciones del software de las máquinas virtuales ejecutan programas como si no se hubiera separado las máquinas físicas. La Virtualización permite que múltiples sistemas operativos se ejecuten simultáneamente en la misma máquina física. La virtualización y la migración dinámica de las máquinas virtuales permiten a la nube de computación hacer un uso más eficiente de los recursos físicos disponibles. La virtualización se logra mediante la adición de una capa por debajo del sistema operativo, entre el OS y el hardware. Esta capa adicional que hace posible ejecutar varias instancias de sistemas operativos en la parte superior de los mismos recursos subyacentes. Existen dos procesos para la realización de la virtualización que son:



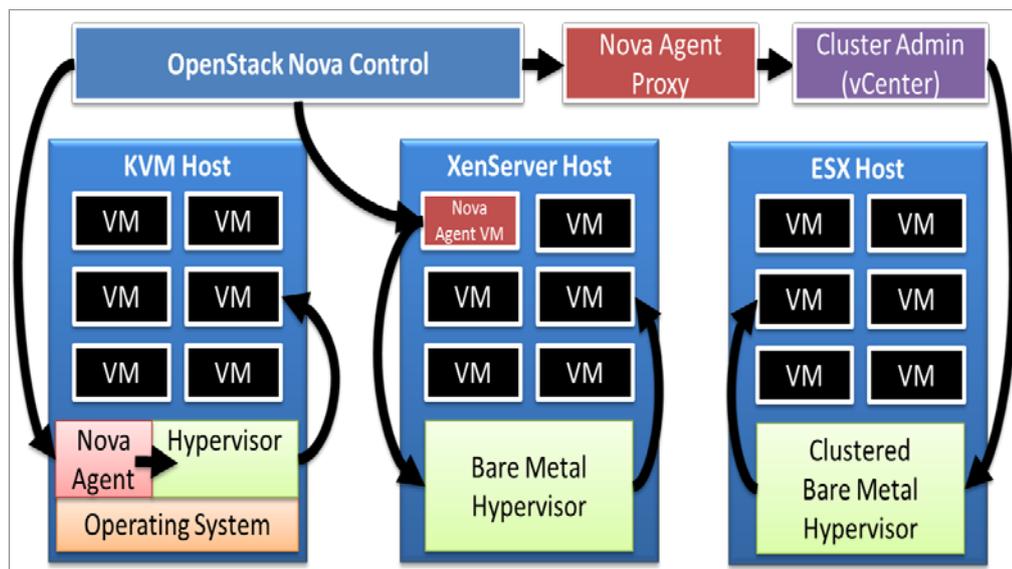
2.9 Capas de virtualización.

Fuente: *Virtualization: A Beginner's Guide* [40].

Fuente: En esta parte se encuentran la parte del sistema de lo que se requiere virtualizar que tienen como nombre imágenes. Se instala directamente en el sistema, que tiene nombre de hipervisor y tendrá acceso directo al hardware. Por esta razón, es más rápido el trabajo de la virtualización debido a que aloja los requerimientos de las capacidades a ofrecer.

Destino: Las capacidades y arquitectura requeridas se colocan en la parte superior de un sistema operativo siendo esto virtualización, ya sea para un host o para un servidor. En cualquier caso, la capa de virtualización gestiona todas las máquinas virtuales, el lanzamiento de un monitor de la máquina virtual (VMM) para cada uno.

La técnica de un hipervisor es el utilizado en todos los centros de datos cloud computing, ya que es la opción más eficiente en términos de utilización del hardware. Véase la figura 3.0.



3.0 Diagrama de virtualización de un hipervisor OpenStack.

Fuente: <http://en.community.dell.com/techcenter/b/techcenter/archive/2011/06/01/virtualizing-approaches-for-openstack-nova-looking-at-the-many-ways-to-skin-the-cactus-kvm-v-xenserver-v-esx.aspx> [41].

Todas las plataformas de virtualización son tan necesarias al momento de realizar trabajos de abstracción de recursos tanto así que openstack siente la necesidad de recurrir a esta tecnología. Por lo general, cuando se habla o se menciona de virtualización existe un estándar de la forma en realizarlo pero existen variaciones

significativas relacionadas ante a estas plataformas de hipervisores que actualmente se han desplegado. Es muy importante tener claro la diferencia, con ello se podrá tomar mejores decisiones al momento de utilizar ciertas plataformas de virtualización. Muchas veces se menciona a los hipervisores de openstack por ser uno de las plataformas más utilizadas en el mercado por lo cual se pueda dar una breve descripción de cada una de las tres opciones:

“SO + Hipervisor (KVM), la arquitectura del hipervisor se despliega hacia una aplicación gratuita en la parte superior de un sistema operativo (SO). En este modelo, el proveedor de servicios gestiona el sistema operativo y el hipervisor de forma independiente. Esto significa que el sistema operativo debe ser mantenido, también permite que el sistema operativo se ha mejorado para la gestión de la nube, añadiendo otras funciones de almacenamiento. Debido a que son hipervisores gratuitos, permanentemente conducen a la innovación de la virtualización.” [21].

“Hipervisor Xenserver, su arquitectura integra el hipervisor, y el sistema operativo como una sola unidad. En este modelo, el proveedor de servicios gestiona el hipervisor como una sola unidad. Esto hace que sea más fácil de sostener y mantener el hipervisor ya que la plataforma puede ser estrictamente controlada, sin embargo, limita la capacidad del operador para extender consultas múltiples en el servidor. En este modelo, los operadores pueden agregar agentes directamente al hipervisor, pero no realizar cambios en el sistema operativo subyacente o a la asignación de recursos.” [21].

“Hipervisor vCenter, su arquitectura integra múltiples servidores en un único hipervisor. En este modelo, el proveedor de servicios no logra que el hipervisor opere el medio ambiente a través del grupo. Esto hace que sea más fácil realizar el equilibrio de los recursos y la tolerancia a fallos en el dominio de la agrupación, sin embargo, el operador debe contar con el supervisor, porque el manejo directo del sistema crea un problema con varias cabeceras. La falta de gestión directa mejora la compatibilidad en el costo de la flexibilidad, la escala es también un reto para los hipervisores agrupados debido a que su ámbito de control se limita a las fronteras de

recursos prácticos: esto significa que grandes nubes añaden complejidad a medida que interactúan con varios clústeres.”. [21].

2.4.2 La virtualización completa

La virtualización completa, utiliza una combinación de la ejecución directa y la traducción binaria. Traducción binaria se refiere a la utilización de las instrucciones no virtualizable mediante la sustitución de estas instrucciones con las de otros que realizan el mismo efecto sobre el hardware virtualizado.

Este enfoque se denomina virtualización completa, ya que los resúmenes son completos en el sistema operativo del hardware subyacente. El hipervisor secundario traduce todas las instrucciones especiales y las reemplaza con una nueva secuencia de instrucciones (que se almacenan en caché para su uso futuro). A nivel de usuario las instrucciones se ejecutan directamente en el hardware, por lo tanto, la virtualización completa no tiene ningún impacto en su velocidad de ejecución. [22].

2.4.3 Paravirtualización

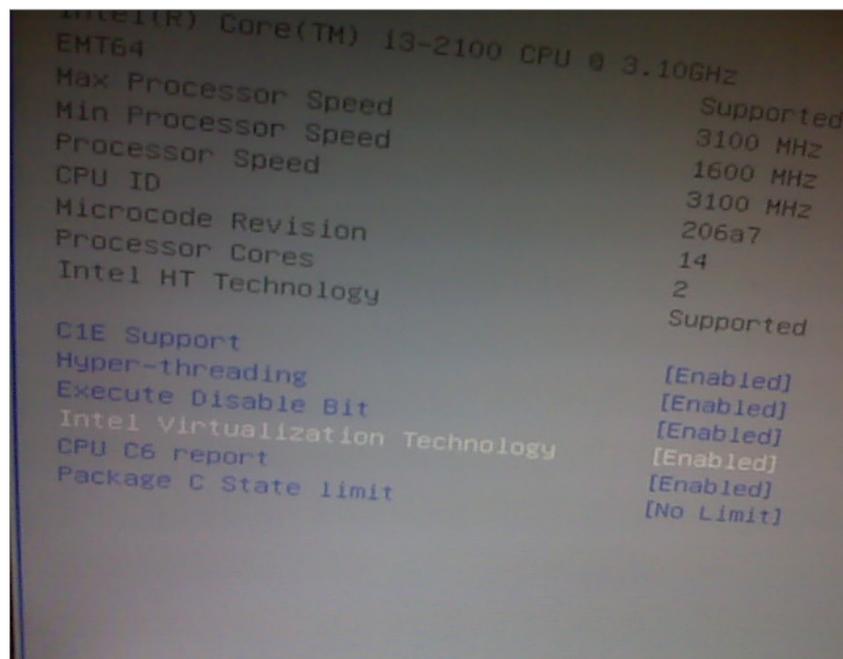
Paravirtualización implica la modificación de un sistema operativo invitado. Hoy en día este método sólo es compatible con sistemas operativos de código abierto, lo que limita su aplicabilidad. Sin embargo, la paravirtualización ofrece un mayor rendimiento que la virtualización completa en el rendimiento, ya que no necesita capturar y convertir las peticiones de un sistema operativo.

2.4.4 La virtualización asistida por hardware

Virtualización asistida por hardware es un enfoque alternativo. Los proveedores han añadido el soporte de virtualización para sus procesadores, debido a la utilización generalizada de la virtualización.

La virtualización asistida por hardware está disponible en los productos que emplean

Intel VT-x y las tecnologías de AMD AMD-V. Esta solución de hardware se basa en un nuevo modo de ejecución de la cpu que permite que el vmw, para peticiones de sistemas operativos automáticamente atrapados por el hipervisor, por lo que no hay necesidad de paravirtualización o la traducción binaria requerida en la virtualización completa.



3.1 Tecnologías con virtualización (Intel).
Fuente: Autor.

2.4.5 Las alternativas actuales en el mercado de la computación en nube

Las alternativas actuales que presenta en la actualidad la nube de computación se pueden distinguir basándose en el nivel de abstracción (es decir, el nivel de servicio), y el nivel de gestión de los recursos.



3.2 Amazon web services.
Fuente: <http://aws.amazon.com/ec2/> [23].

Una de los principales compañías que presta este servicio es amazon web services (AWS) AWS [23] y se refiere a los servicios que ofrece Amazon para cubrir todo el espectro de servicio completo. Amazon es el único proveedor a la fecha con los productos de las tres clases. AWS incluye una serie de componentes que se define a continuación:

- Amazon Elastic Compute Cloud (EC2): El producto IaaS de Amazon es el líder en su clase. Se suministra a clientes con un “*pay-as-you-go*” de los recursos que pueden incluir el almacenamiento o la computación. EC2 tiene una interfaz web para solicitar las máquinas virtuales del servidor de los casos. Una instancia EC2 parece hardware físico y su nivel relativamente bajo de la abstracción (es decir, por definición, IaaS tienen bajos niveles de abstracción, en comparación a PaaS o SaaS. Permite la configuración del control de los clientes de casi toda la pila de software. Los clientes tienen la posibilidad de aumentar o disminuir el número de servidores casos, a continuación, AWS reacciona al calcular el número de casos arriba o hacia abajo.
- Amazon Simple Storage Service (S3) implementa un servicio de almacenamiento dinámicamente escalable que puede ser utilizado para alojar aplicaciones que posteriormente se le ofrecerá a los usuarios finales.

Rackspace



3.3 Rackspace.

Fuente: <http://www.rackspacecloud.com/> [42].

Rackspace [42] ofrece una infraestructura como un servicio, llamado Cloud servers, o una plataforma como un servicio cloud sites, para alojar aplicaciones web con las necesidades de esta. Rackspace también proporciona cloud files, un servicio de almacenamiento, que pueden ser combinados con una red de distribución de

contenidos (CDN⁸) de servicio. Este último servicio compite directamente con el CDN de Amazon, llamado cloud front, pero Rackspace, a diferencia de Amazon, no se cobra por el consumo de ancho de banda entre el servicio de almacenamiento y el CDN.

GoGrid



3.4 GoGrid.

Fuente: <http://www.gogrid.com/cloud-hosting/> [43].

GoGrid [43] proporciona una infraestructura como un servicio, de pie como un competidor directo a Amazon o Rackspace. GoGrid ofrece un servicio competitivo que consiste en servidores dedicados alojados en las instalaciones de su nube. Así que es un proveedor de infraestructura virtual y física en la demanda, a diferencia de Amazon. Además, GoGrid complementa la oferta de la infraestructura dedicada a un entorno híbrido que permite a los usuarios de su servicio de hosting dedicado solicitar los recursos virtuales para manejar los picos de uso.

Salesforce



3.5 Salesforce.

Fuente: <http://www.salesforce.com/> [44].

⁸ CDN es una red superpuesta de computadoras que contienen copias de datos, colocados en varios puntos de una red con el fin de maximizar el ancho de banda para el acceso a los datos a través de la red.

Salesforce [44] es uno de los pioneros de la computación en la nube. Salesforce ha centrado en los clientes empresariales y ha añadido nuevas aplicaciones en la parte superior de su CRM⁹. Mientras que Salesforce antes sólo ofrecía productos SaaS, en la actualidad salesforce se desplazó hacia el mercado del PaaS con el lanzamiento de su plataforma Force.com, que permite a los desarrolladores, aplicaciones que se ejecutan de forma nativa de la plataforma de salesforce o ser integrado con servicios de terceros. En el caso de Force.com, Salesforce es responsable de la escala hacia arriba o debajo de la plataforma, según sea necesario. APEX está estrechamente relacionados con C# y Java. Force.com ofrece también los no-programadores con tutorías y modelos para que puedan crear aplicaciones web de negocios de una manera visual.

Google App Engine

El producto de Google PaaS [45] es una plataforma para desarrollar y alojar aplicaciones web en los servidores de Google. El usuario puede aprovechar de Google sistemas de archivos distribuidos y escalables, junto con las tecnologías utilizadas por gran variedad de aplicaciones de google web (por ejemplo, gmail, google docs, google reader, maps, earth, o youtube).



3.6 Google App Engine.

Fuente: <http://code.google.com/appengine/> [45].

⁹ CRM hace referencia a la estrategia de negocio focalizada hacia el cliente

Actualmente también hay soporte para Java, y está previsto que otros lenguajes de programación se permitirá en el futuro. En un movimiento hacia la conexión de dos nubes, Google y Salesforce han proporcionado recientemente bibliotecas que permiten a los desarrolladores acceder a la web de otros servicios de programación de las aplicaciones

Microsoft Windows Azure

El servicio de Microsoft PaaS se llama Windows Azure [46]. Esta a la venta a partir de febrero de 2010 y ofrece una plataforma en la nube los desarrolladores de computación bajo demanda y de almacenamiento para hospedar, escalar y gestionar aplicaciones web



3.7 Microsoft Windows Azure.

Fuente: <http://www.microsoft.com/windowsazure/windowsazure/> [46].

La plataforma de servicios azure en la actualidad sólo se ejecuta en net framework, pero microsoft ha indicado que se apoyara en una amplia gama de idiomas. De hecho, dos kits de desarrollo de software (SDK) ya han sido puestos a disposición para la interoperabilidad con la plataforma de servicios de Microsoft azure.

Sun Cloud



3.8 Oracle.

Fuente: <http://www.oracle.com/us/technologies/cloud/index.htm>. [47].

Sun Microsystems (ahora Oracle) en marzo de 2009 introdujo un servicio de nube para competir contra Amazon EC2 en el campo de la IaaS [47]. Hoy en día el servicio de esta aplicación va en aumento, trayendo beneficios a las empresas con las que se contrata. Tras la fusión con Oracle, se anunció que el servicio de Sun Cloud se ya no estará disponible, pero no está claro si otro producto de la nube se dará a conocer en su lugar.

Eucalipto



3.9 Eucalyptus.

Fuente: <http://open.eucalyptus.com/> [48].

Eucalipto [48] no es comparable en tamaño o la capacidad con la oferta anterior, pero vale la pena, incluso a causa de su propósito distintivo. La misión inicial de eucalipto fue, y sigue siendo, permitir a los académicos realizar investigaciones en el campo de cloud computing. Además de la investigación de mercado, también se ha posicionado como una nube que ofrece el sistema privado. Esta iniciativa combina el apoyo al desarrollo abierto con el objetivo de ser fácil de instalar y mantener.

Openstack



4.0 Openstack.

Fuente: <http://openstack.org/> [49]

Openstack [49] tiene el mejor ámbito de aplicación específica para el modelo de IaaS, donde también es totalmente compatible con Amazon EC2, como amazon

también utiliza las mismo API AWS. OpenStack es de código abierto diseñado para suministrar y gestionar grandes redes de máquinas virtuales, la creación de una plataforma de cloud computing redundante y escalable. Le da el software, los paneles de control, y las API necesaria para personalizar una nube privada, incluyendo las instancias en ejecución, las redes de gestión y acceso de control a través de los usuarios y proyectos. Esta plataforma se implementó con el fin de realizar nuestras propias pruebas de rendimiento en esta tesis.

2.5 Aspectos Económicos.

Cloud computing promete optimizar la economía de las compañías pero antes de enfrentar este reto los usuarios deben estar convencido que las nubes es la mejor opción para confiar sus información y contenido y poderlas administrar desde la comodidad de cualquier parte del mundo.

Los siguientes aspectos ayudarían considerablemente al momento de considerar tomar decisiones en favor de las economías de las empresas:

Reducción de costos.- Debería ser la principal preocupaciones de las compañías para construir un sistema de nubes privadas que pueda cambiar el comportamiento del consumidor y reducir los costos de mantenimiento de la infraestructura. Escalabilidad y pago por uso son aspectos esenciales de este problema. En general, la creación de un sistema de nubes normalmente implica costos extras ya sea mediante la adaptación de la lógica de negocio a las interfaces de host específicos de las nubes.

Mejora del tiempo.- Este aspecto en el mercado es esencial, especialmente para pequeñas y medianas empresas que desean vender sus servicios de forma rápida y sencilla con pequeños retrasos causados por la adquisición y el establecimiento de la infraestructura, en particular es un ámbito compatible y competitivo con las grandes industrias. Las grandes empresas deben ser capaces de publicar nuevas capacidades con pocos gastos indirectos para seguir siendo competitivos.

Retorno de la inversión (ROI).- Es sumamente esencial para todos los inversores y no siempre puede ser garantizada de que el retorno será más que la inversión de hecho, algunos sistemas de nubes en la actualidad no garantizan este aspecto. El empleo de un sistema de nubes debe asegurar que el costo y el esfuerzo recaen en él, se ve compensado por los beneficios que para ser comercialmente viables. Ello puede conllevar directa (por ejemplo, más clientes) e indirectos (por ejemplo, beneficios de la publicidad) retorno de la inversión.

CAPEX y OPEX.- Es un implícito y mucho argumentó este aspecto lo pudimos revisar con más detalles en el índice (2.1.2 Pago por uso): como el costo-beneficio real (retorno de la inversión) no siempre está claro (ver Figura 2.3 Gastos Fijos y Variables). Los gastos de capital (CAPEX) se requiere para construir una infraestructura local, pero con la internalización de recursos informáticos a los sistemas de nubes en la demanda y escalable, en una empresa realmente se gastan los gastos operativos (OPEX) para el aprovisionamiento de sus capacidades, ya que adquieren y utilizan los recursos de acuerdo a sus necesidades operativas.

2.6 Limitaciones cloud computing.

La mayoría de las empresas ya tienen conocimiento de cloud computing, en la actualidad está siendo reconocida como un revolucionario concepto de TI y con diferentes servicios ya que está satisfaciendo las necesidades de los clientes muy diversos, que van desde las pequeñas empresas hasta las más grandes, y sobre todo a los usuarios finales.

Algunas aplicaciones basadas en la nube como gmail, han tenido gran éxito; Pero como la diversidad de la oferta crece, también las decisiones de renunciar a confiar en los datos que no estarán en servidores físicos dentro de las mismas compañías.

Esto se observa fácilmente en las empresas cuando toman las decisiones en los departamentos de tecnología de la información de las empresas y las organizaciones siguen rechazando el movimiento a la nube. En la actualidad la mayoría de las

organizaciones sólo están dispuestas a externalizar aplicaciones que requieren menos información sensible [24].

2.6.1 Disponibilidad de servicios

Las interrupciones en un servicio de las nubes pueden convertirse en una preocupación importante cuando los clientes han depositado toda su información en la nube y lo requieran necesitar en cualquier momento.

Teniendo en cuenta que las interfaces de gestión para los clientes de las nubes públicas y privadas son accesibles a través de internet, hay un mayor riesgo de fracaso en comparación con los servicios tradicionales, ya que hay puntos más débiles de la cadena de los elementos necesarios para acceder a la información o aplicación.

Por ejemplo, las vulnerabilidades del navegador web pueden llevar a fallas en la prestación de servicios. A los medios posibles para obtener un alto grado de disponibilidad sería el uso de múltiples proveedores de cloud computing.

Proveedores de la nube son muy conscientes de estos riesgos y en la actualidad tiene más información acerca del estado actual del sistema, ya que esto es algo que demandan los usuarios.

2.6.2 Abuso de privilegios

El abuso por parte del proveedor podrían tener un impacto y daños a la confianza que de a poco se van ganando estas organizaciones, la reputación, o directamente el daño al cliente. Se debe tener en consideración que estos mismos tipos de ataques pueden ser llevadas a cabo por los empleados internos en una infraestructura de computación tradicionales (es decir, no de las nubes).

Las empresas por lo general antes de empezar a utilizar las nubes deben realizar una evaluación completa de cualquier proveedor potenciales, se especifican los requisitos de recursos humanos (es decir, indicando que tendrán acceso a sus datos y qué nivel de acceso se tiene) y que requieren medidas de transparencia.

2.6.3 Escala de recursos

La capacidad de los recursos de la escala hacia arriba o hacia abajo para cumplir con la carga de trabajo es una de las ventajas de cloud computing. Sin embargo, esta ventaja puede llevar a fallas en el servicio si no está bien implementada, o si en un tiempo máximo de respuesta no se ha acordado albergar su servicio en una nube, puede ver cómo el tiempo de respuesta aumenta de manera constante cuando el uso de la aplicación aumenta también porque la nube no se puede ampliar hasta los recursos con la suficiente rapidez.

El cliente puede y tiene la potestad de elegir el servicio que ellos no querían. Los acuerdos de nivel de servicio determinar calidad del servicio, pero no en términos de tiempo de respuesta a las variaciones de la carga de trabajo. Hay soluciones propuestas en los acuerdos de nivel de servicio (SLA¹⁰) para la escalabilidad implementada a través de la máquina de aprendizaje estadístico.

2.6.4 Los datos de seguridad y confidencialidad

La confidencialidad de los datos deben estar seguro de estar almacenados en la nube o en tránsito (es decir, hacia y desde la nube). Una manera de proporcionar confianza y seguridad de que los datos de los usuarios son íntegros y confidenciales, es la de un entorno de ejecución, esto quiere decir que los datos son verificados por su

¹⁰ SLA.- Es un contrato escrito entre un proveedor de servicio y su cliente con objeto de fijar el nivel acordado para la calidad de dicho servicio.

propietario. Este aspecto distribuido del modelo de la nube implica necesariamente más tránsito de datos a través de las redes, creando así nuevos riesgos de seguridad desafiante. Se toma en cuenta la encriptación de los datos que sean de forma segura en el almacenamiento en la nube, sin embargo esto crea un problema y se debe a que por lo general cuando la información pasa a la nube, estos datos y aplicaciones que se ejecutan no deben estar encriptado por el motivo que consume más recursos de computación.

Hay recientes medidas para enfrentar la cuestión de confianza en la nube [25], cuyo objetivo es aplicar el modelo de computación (desarrollado en 2003 por Intel, AMD, HP e IBM) a la nube. Sin embargo, el alcance de esta iniciativa es para proteger contra la información privilegiada maliciosos, dentro de la organización proveedor de la nube. Otro de los proyectos del Grupo de criptografía es un mecanismo de búsqueda de cifrado introducido por *Kamara y Lauter* [40].

El proceso que subyace en este sistema es basado en una aplicación local, instalado en la máquina del usuario, compuesto por tres módulos: un procesador de datos, un verificador de datos, y un generador de símbolos.

El usuario encripta los datos antes de subirlo a la nube. Cuando algunos datos son necesarios, el usuario utiliza el generador de símbolos para generar un símbolo y una clave de descifrado. El token se envía a la nube, el archivo seleccionado y el cifrado se descarga luego, estos archivos se verifican a nivel local y el descifrado comparte la activación mediante el envío de la señal y la clave a otro usuario que desea interactuar en la misma infraestructura. La versión empresarial de la solución consiste en añadir un generador de credenciales para simplificar el proceso de interacción y tener más tiempo de respuestas.

2.6.5 Eliminación de los datos

En estrecha relación con los problemas de aislamiento que la arquitectura multiusuario puede conllevar al hecho de que el usuario puede borrar los datos a

petición. Un usuario de una nube pública puede exigir que sus datos sean eliminados, es decir, retirarse completamente de la nube. Ya que esto sólo puede ser completamente realizado por el proveedor, en varias ocasiones volver a escribir los sectores del disco con datos aleatorios. Y posiblemente, el formateo del disco duro del servidor, esto podría llegar a ser imposible de hacer en el entorno del proveedor de servicios.

2.6.6 Recuperación y copia de seguridad

Los proveedores de la nube deben tener un plan establecido para los datos de back-up en caso de situaciones de desastre. Esto puede lograrse mediante la replicación de datos a través de diferentes lugares y el plan debe ser abordado de acuerdo al nivel de servicio. Esto crea un problema mayor si el usuario se mueve y la calidad de la conexión puede cambiar, por lo tanto, en algunas situaciones, depender de un proveedor de servicios de internet no es simplemente una opción.

Este modelo se basa en el hecho de que la aplicación de software más utilizado actualmente es el navegador web y que hoy en día las solicitudes completas pueden ser entregadas como un servicio a través de Internet y todos los de la interacción del usuario final puede ocurrir a través de un navegador web.

Una conclusión obvia es construir un sistema operativo basado en web. En este enfoque, el navegador web actúa como interfaz con el resto del sistema y de hardware, como discos duros o los procesadores de gran alcance, no sería necesaria un nivel local.

En cambio, una laptop o de otro tipo de equipo ligero como un procesador de bajo consumo de energía sería suficiente, siempre que la mayoría de los cálculos se lleven a cabo en la nube y que todos los datos se almacenen allí. Un modelo parecido es el que está llevando a cabo google con su sistema operativo chrome [27]. Además, otros proveedores de software independientes están desarrollando ofertas web de escritorio, tales como el eyeOS [28].

Capítulo 3 Estudio de Openstack para realizar nubes privadas

3.1 Los componentes de Openstack

3.1.1 OpenStack

Openstack y su plataforma de tecnología de código abierto basado en el proyecto de infraestructura proporcionado por rackspace en conjunto con las NASA, de tal forma estandariza la nube evitando así el monopolio o privatización que ofrece las nubes de computación. La plataforma comprende básicamente del software que utiliza actualmente rackspace permitiendo a las organizaciones crear y ofrecer cloud computing. Esta plataforma se compone de dos aspectos fundamentales: La computación y su almacenamiento. Openstack es el software más grande en la actualidad para crear y gestionar grandes grupos de servidores virtuales y surge debido a la necesidad de manejar grandes cantidades de datos. [29]

El proyecto Openstack tiene como objetivo crear una nube de código abierto para la plataforma de computación, nubes públicas y privadas destinadas a la escalabilidad sin complejidad. En un principio se centra en infraestructura como Servicio (IaaS), la plataforma Openstack abarca tres componentes que se mencionara a continuación.

- OpenStack Cálculo: Es una aplicación para poder organizar, administrar y ofrecer a las máquinas virtuales un software al cual le llamamos "Nova". En este componente lo utilizamos para poner en marcha instancias virtuales, ya sea para un usuario o un grupo. También se utiliza para configurar la red para cada instancia o proyecto que contiene varias instancias de un proyecto en particular.
- Objetos de openstack: Software para el almacenamiento redundante de objetos estáticos. Este aplicativo se llama "Swift" sin embargo omitiremos este proceso en nuestro proyecto.
- OpenStack Imagen: Ofrece servicios de consulta y almacenamiento de imágenes de disco virtuales. El software para esto se llama "Glance".

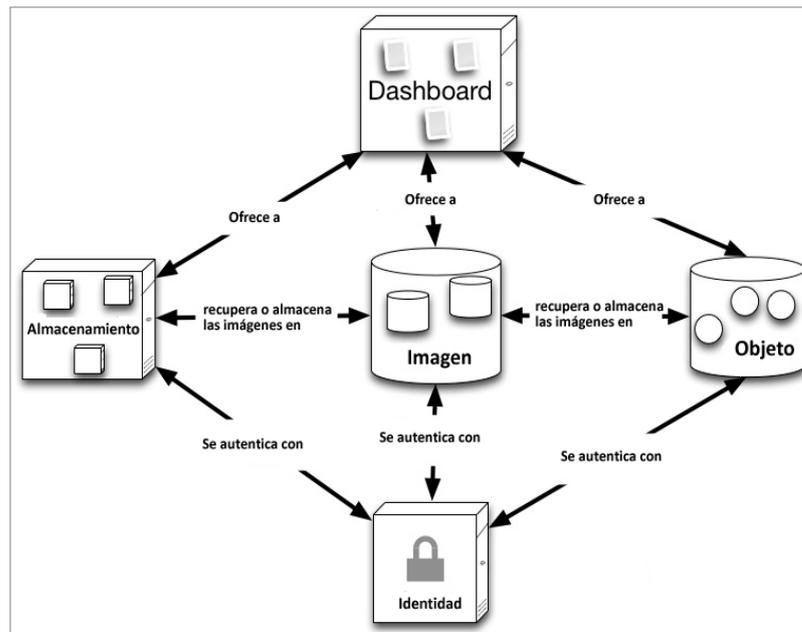
Las imágenes se la pueden configurar de tres maneras: con el objeto de openstack para almacenar las imágenes, utilizando solución de almacenamiento simple de Amazon (S3) de almacenamiento directamente, o mediante el almacenamiento S3 con el objeto como intermedio para el acceso a S3.

3.1.2 Arquitectura Contextual de Openstack.

La arquitectura de OpenStack está diseñada para crear un sistema operativo cloud masivamente estable y escalable. Para lograr esto, cada uno de los servicios que lo constituyente están diseñados para trabajar en conjunto y proporcionar una completa infraestructura como servicio *IaaS*.

Esta integración se ve facilitada a través de interfaces de aplicaciones públicas de programación API's que ofrece cada servicio. Si bien estas API permiten cualquier otro servicio, también permite que un ejecutor pueda cambiar cualquier otro servicio, siempre y cuando mantengan las API en este mismo diseño de Amazon web. Conceptualmente, se pueden ver las relaciones entre los servicios de la siguiente forma:

La siguiente grafica muestra un esquema simplificado de la arquitectura, suponiendo que se esté utilizando todos los servicios juntos en la configuración más común que es de un nodo simple. Los bloques en la figura 4.1, muestra el tablero de instrumentos, el almacenamiento, el objeto, y por último el bloque de identidad aquí es donde se autentican las credenciales y los datos.



4.1 Arquitectura Contextual.

Fuente: [Autor]

El tablero de instrumentos es quien nos proporciona una interfaz web. El nodo del almacenamiento recupera los discos virtuales “Imágenes” con todos los datos asociados en el bloque de imágenes. La imagen almacena los archivos reales del disco virtual en el objeto. Todos estos servicios son autenticados en el bloque de identidad con sus claves que son generadas en el momento de configurar la nube privada.

3.2 Introducción a Openstack.

Openstack ofrece una herramienta para organizar la creación de una nube incluyendo las instancias en ejecución, las redes de gestión, y de control de acceso a la nube a través de los usuarios y proyectos. El nombre del proyecto de base de código abierto es nova, y proporciona el software que puede controlar una infraestructura como servicio (IaaS) o una plataforma de cloud computing. Se trata de un alcance similar al de Amazon EC2 y los servidores de rackspace cloud. Openstack compute no incluye ningún software de virtualización, sino que define los conductores que interactúan con los mecanismos subyacentes de virtualización que se ejecutan en el sistema operativo host, y expone la funcionalidad en un API basado en la web. Para

la ejecución y diseño de una nube se deben tomar en cuenta los siguientes componentes que son parte del esquema final:

3.2.1 Hipervisores.

El proceso de selección de un hipervisor por lo general significa establecer prioridades y tomar decisiones basadas en el presupuesto y las limitaciones de los recursos, así como la lista inevitable de las funciones y las especificaciones técnicas requeridas. La implementación de esta nube se hace con los hipervisores kvm y gemu a continuación se detalla el concepto básico de estos emuladores de maquinas virtuales que nos ayudan mucho a la hora de la implementación de la virtualización.

3.2.1.1 Kvm



4.2 KVM

Fuente: http://www.linux-kvm.org/page/Main_Page [50]

Está basada en el Kernel Virtual Machine (KVM) [31] es una solución completa de virtualización para linux. Se basa en las extensiones de virtualización de la CPU (es decir, amplía el conjunto de instrucciones de la CPU con nuevas instrucciones que permiten las escritura simples de monitores de máquina virtual).

KVM es un subsistema de linux (el componente del núcleo de KVM está incluido en la línea principal del núcleo de linux) que se aprovecha de estas extensiones para añadir un monitor de máquina virtual con la capacidad para Linux. Utilizando KVM, se puede crear y ejecutar múltiples máquinas virtuales que aparecen como procesos normales de linux y se integra con el resto del sistema que trabaja en la arquitectura x86 y soporta tecnologías de virtualización de hardware como Intel. KVM tiene soporte para la migración en vivo, lo que se refiere a la capacidad de migrar una

máquina virtual de un host a otro sin interrupción del servicio. Esta migración se realiza de forma transparente para el usuario final, sin desactivar las conexiones de red o el cierre de las aplicaciones en ejecución en la máquina virtual.

3.2.1.2 Qemu



4.3 Qemu

Fuente: <http://wiki.qemu.org/Index.html> [31].

Qemu al igual que los hipervisores mencionados anteriormente es de código abierto increíblemente poderoso emulador desarrollado por *Fabrice Bellard* y un sin número de otros que pueden emular una gran variedad de arquitecturas, procesadores y periféricos de hardware relacionados.

Qemu virtualiza un entorno de hardware completo para cada uno de estos procesadores y arquitecturas, incluyendo la CPU y los dispositivos periféricos tales como una o más puertos seriales, interfaces de red, un bus PCI, adaptadores de gráficos VGA, el hardware de sonido, unidades de disco USB componentes, y mucho más. Esto le permite ejecutar sistemas operativos invitados no modificados por cualquiera de estos procesadores y arquitecturas de QEMU.

La emulación de hardware QEMU es tan excepcional que una parte significativa de QEMU se utiliza para soportar las conexiones de serie, hardware de gráficos, múltiples particiones e imágenes de disco, y así sucesivamente en las implementaciones de virtualización completas de los paquetes de virtualización como Xen y KVM.

Se analizo los dos más importantes hipervisores para la emulación de maquinas virtuales sin embargo existe más tipos de emuladores estándares de virtualización

que pueden ser utilizados con otras plataformas las cuales se menciona a continuación:

3.2.1.3 XEN



4.4 Xen

Fuente: <http://xen.org/products/xenhyp.html> [51]

El hipervisor Xen se inserta entre el hardware del servidor y el sistema operativo. Esto proporciona una capa de abstracción que permite a cada servidor físico ejecutar uno o más "servidores virtuales", desde del sistema operativo y sus aplicaciones del servidor físico subyacente. Xen literalmente define las interfaces de máquinas virtuales y su interacción con el invitado (Usuario). El hipervisor Xen se posa sobre el hardware físico y presenta los dominios invitados con un hardware de interfaz virtual, de esta manera, se define la máquina virtual que los dominios de los huéspedes ven en lugar del hardware físico.

La utilización del hipervisor Xen es técnicamente administrable debido a que maneja menos de 150.000 líneas de código. Eso se traduce en sobrecarga extremadamente bajo y un rendimiento casi nativo para los huéspedes. Xen re-utiliza los controladores de dispositivos existentes (tanto de código cerrado y abierto) de linux, por lo que su utilización es sumamente fácil. [50].

3.2.1.4 LXC LINUX CONTAINERS



4.5 Lxc Linux containers

Fuente: <http://lxc.sourceforge.net/> [52].

“LXC (Linux Containers) es una tecnología de virtualización en el nivel de sistema operativo (SO) para linux. OpenVZ permite que un servidor físico ejecute múltiples instancias de sistemas operativos aislados, conocidos como servidores privados

virtuales (SPV o VPS en inglés) o Entornos Virtuales (EV). LXC no provee de una máquina virtual, más bien provee un entorno virtual que tiene su propio espacio de procesos y redes.” [52].

3.2.1.5 VMWARE ESX/ESXI



4.6 VMware

Fuente: <http://www.vmware.com/products/vsphere-hypervisor/support.html> [53].

“El hipervisor vmware vsphere es la forma más simple y más fácil de comenzar con la virtualización de forma gratuita. Este hipervisor totalmente funcional le permite virtualizar los servidores y ejecutar sus aplicaciones en máquinas virtuales en cuestión de minutos. Este hipervisor está basado en vmware, la arquitectura de hipervisor es el que establece el estándar de la industria por su fiabilidad, rendimiento y soporte de ecosistema. Consolidar las aplicaciones en menos servidores y comenzar a ahorrar dinero a través de hardware reducido, energía, refrigeración y los costos de administración es la misión de este hipervisor vmware vSphere. [53]

A continuación los esquemas principales de este hipervisor:

- Ejecutar múltiples aplicaciones en un solo servidor.
- Ejecutar una virtualización más ecológica y reducir los costes energéticos.
- Realizar copias de seguridad y recuperar aplicaciones con mayor facilidad.
- Virtualizar incluso aplicaciones críticas de negocio.

3.3 Administradores e usuarios.

El sistema de OpenStack está diseñado para ser utilizado por muchos usuarios que no están familiarizados con cloud computing, fundamentalmente las personas que nunca han participado de asignaciones basadas en roles de accesos compartidos. Las

funciones de control de acceso que se le permite a un usuario para poder administrar su espacio virtual no requiere un papel particular, pero esto es configurable a partir el momento que se le asigna el usuario y contraseña y se le garantiza el seguro acceso a sus recursos virtuales.

Los controles que pueden ser previamente solicitados por el usuario son:

- Número de volúmenes que pueden ser creados
- El tamaño total de todos los volúmenes medidos en GB
- Número de núcleos de procesador que podrá ser utilizado
- Accesibles al público, las direcciones IP

3.4 Imágenes e instancias.

Una imagen es un archivo que contiene información acerca de un disco virtual que se replica por completo toda la información acerca de un equipo de trabajo en un punto en el tiempo, incluyendo información del sistema operativo y la información del sistema de archivos. Puede utilizar la administración de certificados para descifrar las imágenes agrupadas. Con el sistema de identificación en desuso, puede utilizar los euca2ools distribuidas por el equipo de Openstack para la inclusión, la agrupación, y borrar imágenes. Al utilizar el servicio de la imagen (Keystone), puede utilizar (Nova) para trabajar con imágenes y el (Glance) para subir las imágenes.

3.5 Arquitectura del sistema de Openstack.

Openstack compute consta de varios elementos principales. Un "nodo controlador de la nube" que contiene muchos componentes, y representa el estado global e interactúa con todos los demás componentes. Un API de servidor actúa como el interfaz web llamado (Dashboard) quien da el servicios para el controlador de la nube. El controlador proporciona los recursos informáticos del servidor y por lo general contiene el servicio de cómputo, el componente del objeto, opcionalmente proporciona servicios de almacenamiento. Un gestor de autenticación proporciona

servicios de autenticación y la autorización cuando se utiliza el sistema de almacenamiento, o puede usar el servicio de identidad (Keystone) como un servicio de autenticación independiente.

Un controlador de volumen proporciona un servicio rápido y permanente a nivel de bloque de almacenamiento para el servidor de almacenamiento. Un controlador de red proporciona redes virtuales para permitir servidores de cómputo de interactuar entre sí y con la red pública. Un programador de computación selecciona el controlador más adecuado para acoger una instancia.

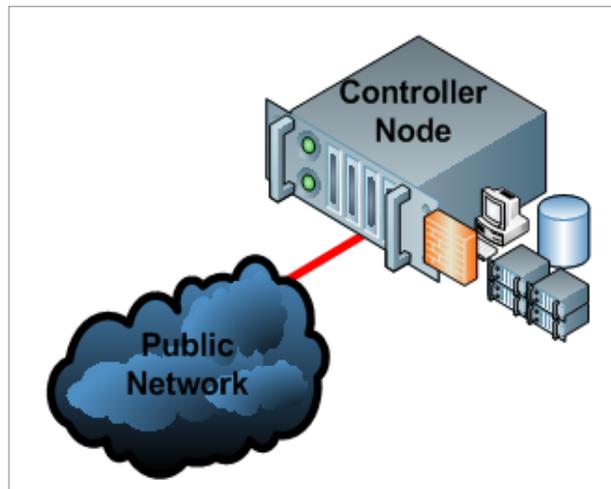
Openstack se basa en una responsabilidad compartida basada en arquitecturas. Puede ejecutar la totalidad de los principales componentes en varios servidores, incluyendo un controlador de almacenamiento, control de volumen, controlador de red, y el almacén de objetos (o servicio de la imagen). Un controlador de la nube se comunica con el almacén de objetos internos a través de HTTP (HyperText Transfer Protocol), pero se comunica con un planificador, controlador de red, y el controlador de volumen a través de AMQP (Protocolo de resolución de problemas de Avanzada).

3.6 Arquitectura técnica de Openstack.

Las instalaciones de Openstack con (Stackops) reduce en gran medida la complejidad de la plataforma. Por defecto se establece los valores para tener una plataforma totalmente funcional, sin que sea necesaria conocer los detalles de cada uno de los parámetros. El propósito de realizar una nube privada utilizando esta plataforma es apropiado para las empresas a dejar los centros de datos e innovar en esta nueva tecnología.

En Stackops tenemos diferentes formas de configura las arquitecturas en base a las necesidades de los clientes desde una instalación simple sobre una máquina virtual, hasta el despliegue de una solución multinodo con alta disponibilidad de los componentes.

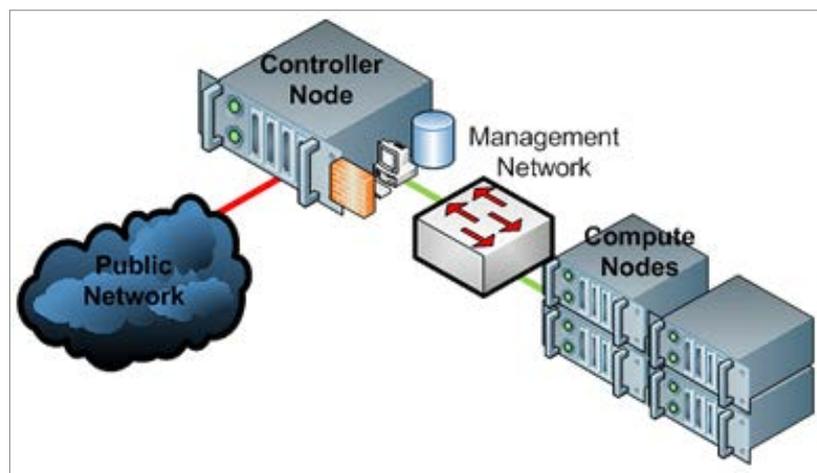
3.6.1 Nodo Simple.



4.7 Nodo Simple.
Fuente: Autor [54].

Esta es la instalación más básica, si sólo se desea probar la infraestructura. Todos los componentes de openstack como el controlador de red, almacenamiento y procesamiento, más otros, están instalados en un solo nodo. Además, el sistema puede ser instalado en una máquina virtual y luego se utiliza el emulador QEMU en lugar de KVM como hipervisor predeterminado. Sólo se necesita una tarjeta de red para hacer funcionar su sistema.

3.6.2 Nodo Doble.



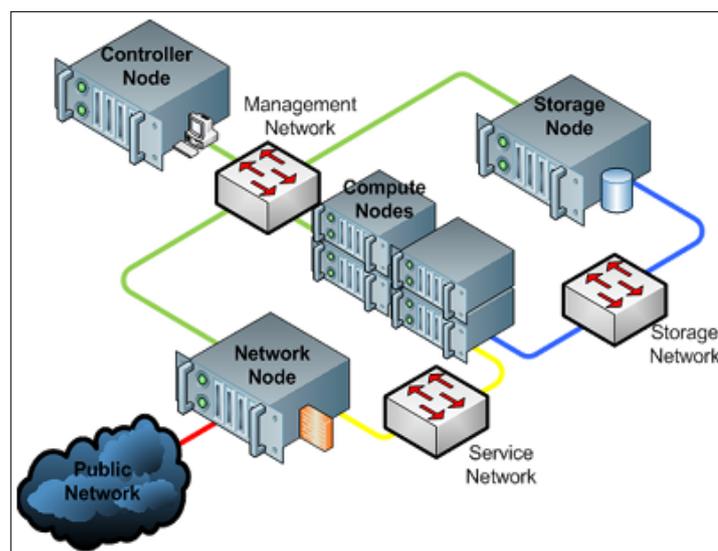
4.8 Nodo Doble.
Fuente: Autor [55].

Una arquitectura de nodo doble tiene todos los componentes a excepción de nova compute que es el nodo de almacenamiento que se ejecuta en otro, lo veremos más adelante, además de uno o más nodos de computación en este diseño solo aplicaremos uno solo para la demostración. La configuración del nodo controlador contiene lo siguiente: La configuración del controlador, de volumen y componentes de red.

Esta es la instalación que se realizara para la demostración de producción. Todos los componentes de la solución openstack más otros están instalados en un controlador del nodo, y el nodo de cálculo para el almacenamiento está instalado en el nodo diferente. Ambos nodos pueden funcionar en un entorno virtualizado, pero es recomendable ejecutar el nodo de cálculo en un servidor físico. Lo bueno de esta instalación doble es que se pueden añadir más nodos para la parte del nodo de almacenamiento.

Para la implementación de este prototipo de cloud computing se debe tener dos tarjetas de red en cada nodo para ejecutar la solución con éxito. Dado que los requerimientos de este entorno son más altos que la arquitectura de un solo nodo.

3.6.3 Multi Nodo.

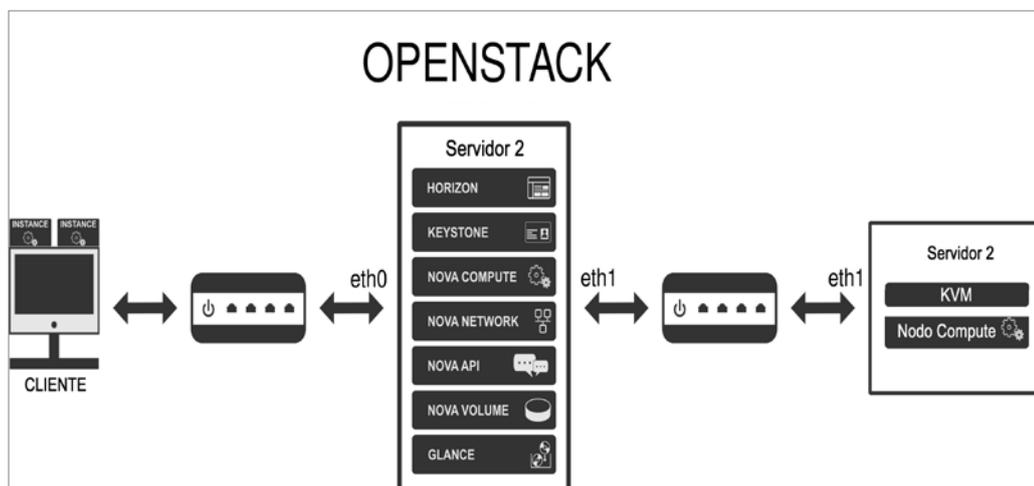


4.9 Nodo Multi nodo.
Fuente: Autor [56].

Esta es la instalación más completa que se podría realizar. Los componentes se dividen alrededor de los diferentes nodos y un mínimo se recomienda 4 servidores físicos. También se pueden añadir más nodos de computación en un momento dado. El controlador de red, y los nodos de cálculo y de volumen necesitan un servidor dedicado con los requisitos del sistema muy específicos. Esta arquitectura añade complejidad adicional a nivel de red. Esta implementación podría ser considerada para trabajos futuros y continuación de esta investigación.

3.7 Diseño del sistema Openstack y configuración de red

El diseño a implementar en este proyecto para la creación de una nube privada con Openstack es la de nodo doble. Esta configuración está diseñada para entornos medios, los requisitos de la red son muy similares a los de la configuración de un solo nodo, con una diferencia significativa: En esta configuración la parte de almacenamiento esta en otro servidor para dar los recursos de una manera independiente.

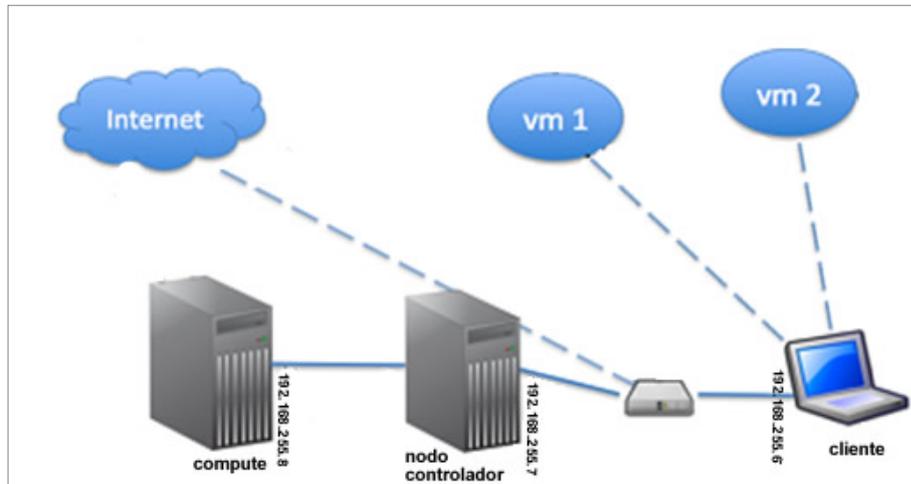


5.0 Esquema del diseño de los servidores.

Fuente [Autor]

Usted puede agregar nodos de almacenamientos, como que sea necesario. Estos nodos de se puede configurar con los parámetros previamente introducidos en la configuraciones del nodo controlador, disminuyendo las posibilidades de error.

3.7.1 Diseño de la red



5.1 Esquema de red de la nube privada con OpenStack.

Fuente [Autor]

En este esquema tenemos dos servidores (nodo controlador y el nodo compute marcados en la figura) ambos servidores tienen 2 tarjetas de red y estos a su vez se interconectan utilizando un cable directo conectados a un switch, creando una red privada.



Esquema de red de la nube privada con OpenStack.

Fuente [Autor]

El nodo controlador contiene dos interfaces de red de modo que se pueda acceder simultáneamente a la red privada y la red de área local (red pública).



Nodo controlador.
Fuente [Autor]

Esta red pública, a su vez tiene acceso a internet, pero es importante destacar que el propio nodo no afecta directamente el acceso a internet. El nodo controlador dirige todo su tráfico hacia y desde internet a través del mismo.



Nodo de almacenamiento.
Fuente [Autor]



Conexiones.
Fuente [Autor]



Infraestructura como servicio.
Fuente [Autor]

A continuación se muestra una tabla de las direcciones ip's asignadas dentro de la red y las ip's que los componentes de Openstack asignaría a las maquinas virtuales.

COMPUTADORA	eth0	eth1	
Nodo Controlador	192.168.255.7	x	
Nodo compute	192.168.255.8	x	
Cliente	192.168.255.6	x	
MV1	10.0.0.2	x	
MV2	10.0.0.3	x	

Tabla 1.0 Direcciones ip dentro de la red.
Fuente [Autor]

3.8 Herramientas de gestión

3.8.1 Hybridfox

Hybridfox es una extensión de código abierto para el navegador web mozilla firefox que ayuda a gestionar tanto Amazon EC2 como cuentas de usuario de openstack y también para eucalipto desde una interfaz única. Como alternativa para un usuario de la nube a las herramientas de línea de comandos [32].

Las principales capacidades de Hybridfox son los siguientes:

- Creación de instancias de una máquina virtual con una dirección IP privada.
- Interfaces para la administración de las maquinas virtuales.
- Lista de las instancias en ejecución

3.8.2 Euca2ools

Euca2ools es la versión de código abierto del conjunto de utilidades de gestión y de línea de comandos son herramientas utilizadas con EC2 de Amazon S3¹¹ y sus servicios, llamado Amazon EC2 con las herramientas de API. Ellos ponen en práctica una larga lista de gestión de la imágenes para las maquinas virtuales. A continuación se mencionan algunas de características que incluyen: [32].

- Consulta las zonas de disponibilidad, llamados "clusters" en nuestro proyecto estará dentro del Nodo controlador.
- Gestión de claves SSH (añadir, lista, eliminar)
- Gestión de las Maquinas Virtuales (inicio, detener, reiniciar)
- Seguridad de la gestión.
- Gestión de imágenes.

¹¹ Amazon S3 proporciona una interfaz de servicios web sencilla que se puede utilizar para almacenar y recuperar cualquier cantidad de datos en cualquier momento, desde cualquier lugar de la web.

Capítulo 4 Implementación de una nube privada con Openstack.

En este capítulo realizaremos la parte implementaría del proyecto tratando de llevar a cabo la parte de virtualización brindando un servicio útil que apoyen las necesidades de las empresas es así que reduciría el numero de servidores físicos y reducir considerablemente costos, incluso poner en práctica soluciones de alta disponibilidad que en algún tiempo iban más allá de sus presupuestos en el cual reducirá tiempo y esfuerzos, a continuación describiremos la parte organizacional de cada uno de los equipos a utilizar a demás se realizar paso a paso la configuración y los componentes de stackops en base a las funciones requeridas.

4.1 Requerimiento del sistema

Básicamente existe un requerimiento fundamental para poner en funcionamiento un prototipo de una nube privada que es elemental y cada servidor tiene sus propias características y necesita sus recursos físicos independiente. Como lo mencionamos anteriormente en la sección 3.7.2 vamos a realizar este método para nuestra nube privada.

4.1.2 Requerimiento del Nodo Controlador.

Para esta configuración necesitamos los siguientes requerimientos:

- Mínimo: El sistema puede tener los recursos y características básicas, los rendimientos serán iguales que un entorno de recursos recomendados y la virtualización puede ser suficiente para desplegar esta configuración.
- Recomendado: La configuración recomendada es apta para el despliegue de un sistema de producción y de virtualización. Se recomienda sea asignados los recursos detallados en la siguiente tabla.

Servidor	Requerimiento Mínimo	Requerimiento Recomendado
Controlador/ Red/ Almacenamiento.	64 bits x86 2 GB Memoria RAM 30 GB de disco duro 1 - 100/100 NIC	2 x 64 bits x86 32 GB Memoria RAM 100 GB de disco duro 2 - 100/100 NIC

Tabla 2.0 Requerimiento del nodo controlador.
Fuente [Autor]

4.1.3 Requerimiento del Nodo Compute (Almacenamiento).

Servidor	Requerimiento Mínimo	Requerimiento Recomendado
Procesamiento	64 bits x86 2 GB Memoria RAM 30 GB de disco duro 2 - 100/100 NIC	2 x 64 bit x86 - Intel VT 32 GB Memoria RAM 2T de disco duro 2 - 100100 NIC

Tabla 3.0 Requerimiento del nodo controlador.
Fuente [Autor]

4.1.4 Requerimiento del cliente

Servidor	Requerimiento Mínimo	Requerimiento Recomendado
Cliente	64 bits x86 1 GB Memoria RAM 30 GB de disco duro 2 100/100 NIC	64 bits x86 4 GB Memoria RAM 320 de disco duro 2 100100 NIC

Tabla 4.0 Requerimiento del cliente.
Fuente [Autor]

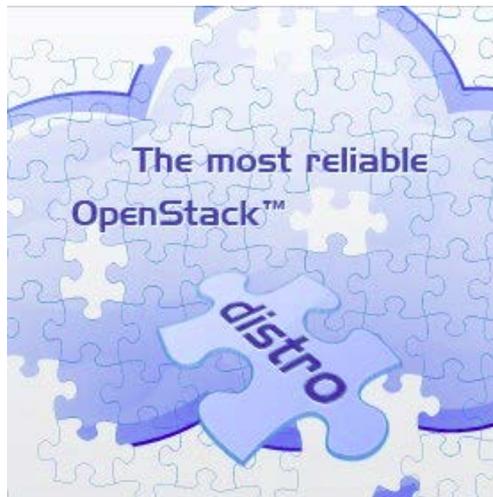
Vale indicar que los 2 GB de memoria RAM sólo permiten 1 instancia m1.small o 2 m1.tiny sin tener problemas de memoria, básicamente sólo para entornos de prueba se utilizara una instancia básica, sin duda será la forma demostrativa para la creación de virtualización.

4.2 Utilización y análisis de Stackops.

4.2.1 Que es Stackops.

“Openstack es una colección de tecnologías de código abierto que ofrece un sistema operativo cloud masivamente escalable. Con el respaldo de rackspace, la NASA, Dell, citrix, cisco, canonical y más de 50 otras organizaciones, openstack se ha convertido en una comunidad global de software de desarrolladores, técnicos, investigadores y de empresas”. [33]

4.2.2 Stackops Openstack Distro 0.5



5.2. Stackops openstack distro 0.5.

Fuente [57]

En un intento de crear un método más amigable en el momento de crear una nube de computación stackops desarrollo un software llamado distro en su versión más actual la 0.5 es una aplicación más fiable, diseñado para reducir el tiempo de evaluación de Openstack.

La instalación tiene un asistente que guía al administrador al momento de crear la nube, la persona será capaz de evaluar si esta solución se adapta a sus necesidades en un tiempo corto. Dependiendo la arquitectura de la nube de computación al ser diseñada, el sistema de stackops puede desplegar un único nodo, o el nodo doble o un multi-nodo impulsado por OpenStack. [56].

4.2.3 Instalacion bajo consola Openstack



Openstack
Fuente [<http://www.openstack.org/>]

Openstack es un software de plataforma de IaaS de código abierto, creado en el lenguaje de programación Py-thon. A continuación se describirá en detalle la instalación, configuración y el uso de esta infraestructura de cloud computing.

Lo primero que se realizara es actualizar las maquinas en ambos nodos con los siguientes comandos:

```
root@nova-controller:~# sudo apt-get update
```

```
root@nova-controller:~# sudo apt-get upgrade
```

Instalamos bridge-utils:

```
root@nova-controller:~# sudo apt-get install bridge-utils
```

Arquitectura

Vamos a instalar este software en dos servidores físicos.

Nodo 1 (Nodo Controlador)

En el nodo controlador se ejecutaran los siguientes componentes nova-scheduler, nova-network, glance, horizon, MySQL, y RabbitMQ)

Nodo 2 (Nodo de Almacenamiento)

En el nodo de almacenamiento se va a ejecutar nova-compute and Qemu. Esto significa que una instancia se puede crear en el nodo 1 o el nodo 2. Significa también que el nodo 1 puede funcionar en modo autónomo.

Configuración del Nodo 1.

Configuramos las interfaces a utilizar (eth1).

```
root@nova-controller:~# nano /etc/network/interfaces
```

Resultado:

```
root@nova-controller:~# ifconfig
eth0      Link encap:Ethernet  HWaddr d0:27:88:55:00:c1
          inet addr:172.18.18.5  Bcast:172.18.18.255  Mask:255.255.255.0
          inet6 addr: fe80::d227:88ff:fe55:c1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:632 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1566 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:73313 (73.3 KB)  TX bytes:197524 (197.5 KB)
          Interrupt:43 Base address:0xc000
```

Configuración del Nodo 2.

```
# The primary network interface
auto eth0
iface eth0 inet static
    post-up ifconfig eth1 0.0.0.0
    address 172.18.18.6
    netmask 255.255.255.0
    network 172.18.18.0
    broadcast 172.18.18.255
    gateway 172.18.18.1
    # dns-* options are implemented by the resolvconf package, if installed
    dns-nameservers 172.18.18.1
    dns-search stackops.org
```

Reiniciamos el servicio.

Luego de la instalación, se debe reiniciar las conexiones con el siguiente comando.

```
root@nova-controller:~# /etc/init.d/networking restart
```

```
root@nova-controller:~# /etc/init.d/networking restart
* Running /etc/init.d/networking restart is deprecated because it may not enable a
gain some interfaces
* Reconfiguring network interfaces...
ssh stop/waiting
ssh start/running, process 2561
[ OK ]
```

Configuración del NTP.

Para mantener todos los servicios sincronizados (fecha y hora) es necesario instalar un servidor NTP (Network Time Protocol).

```
root@nova-controller:~# apt-get install ntp
```

Se realizara el siguiente cambio.

```
root@nova-controller:~# nano /etc/ntp.conf
```

```
server ntp.ubuntu.com
server 127.127.1.0
fudge 127.127.1.0 stratum 10
```

Reinicio

```
root@nova-controller:~# /etc/init.d/ntp restart
```

```
* Stopping NTP server ntpd [ OK ]
* Starting NTP server ntpd [ OK ]
root@nova-controller:~#
```

Glance

El componente de OpenStack encargado de la gestión de instancias es Glance que inicialmente almacena las imágenes en el directorio `/var/lib/glance/images`, aunque es posible utilizar también un componente de almacenamiento de objetos como OpenStack

Instalación

```
root@nova-controller:~# sudo apt-get install glance
```

Se puede encontrar el archivo de configuración de Glance en la siguiente ruta: `/etc/glance/glance.conf`. Los principales componentes son `Glance-api.conf` y `Glance-registry.conf`.

```
root@nova-controller:/etc/glance# ls
glance-api.conf          glance-cache-paste.ini    glance-scrubber.conf
glance-api-paste.ini    glance-registry.conf     glance-scrubber-paste.ini
glance-cache.conf       glance-registry-paste.ini policy.json
```

MySQL Server

Todos los componentes de OpenStack (incluso Glance que por defecto utilizan sqlite) guardarán sus datos en bases de datos MySQL. El servidor de bases de datos se instala en el nodo controlador:

```
root@nova-controller:~# apt-get install mysql-server
```

Configuración de Mysql server

El fichero de la configuración se la encuentra en la siguiente ruta /etc/nova/nova.conf.

```
root@nova-controller:~# nano /etc/nova/nova.conf
```

```

# This is an automatically generated file by stackops
# Please DO NOT CHANGE the parameters manually
--enabled_apis=ec2,osapi_compute,osapi_volume,metadata
--verbose
--ec2_dmz_host=172.18.18.5
--ec2_port=8773
--sql_connection=mysql://root:nova@172.18.18.5:3306/nova?charset=utf8
--quota_volumes=8192
--use_deprecated_auth=false
--lock_path=/tmp
--glance_api_servers=172.18.18.5:9292
--notification_topics=notifications,monitor
--auth_driver=nova.auth.dbdriver.DbDriver
--osapi_compute_listen_port=8774
--root_helper=sudo nova-rootwrap
--allow_admin_api
--quota_metadata_items=128
--quota_ram=20971520
--quota_max_injected_files=5
--quota_max_injected_file_content_bytes=10240
--nodaemon
--quota_max_injected_file_path_bytes=255
--s3_port=80
--api_paste_config=/etc/nova/api-paste.ini
--rabbit_host=172.18.18.5
--my_ip=172.18.18.5
--quota_floating_ips=254
--quota_gigabytes=20480
--state_path=/var/lib/nova
--quota_instances=4096
--image_service=nova.image.glance.GlanceImageService
--ec2_host=172.18.18.5
--logdir=/var/log/nova
--quota_cores=8192
--notification_driver=nova.notifier.rabbit_notifier
--s3_host=172.18.18.5
--s3_dmz=172.18.18.5
--use_project_ca
--keystone_ec2_url=http://172.18.18.5:5000/v2.0/ec2tokens

```

Luego se debe cambiar la dirección del enlace de 127.0.0.1 a 0.0.0.0 en /etc/mysql/my.cnf

```
bind-address = 0.0.0.0
```

Se reiniciara el servidor MySQL para asegurarse de que se inicien todas las interfaces.

Nova

Instalar el servidor de nova, RabbitMQ y varios componentes más de esta aplicación.

```
root@nova-controller:~# sudo apt-get install -y rabbitmq-server nova-common nova-doc python-nova nova-api nova-network nova-volume nova-objectstore nova-scheduler nova-compute
```

Instalación de euca2ools.

Instalar el paquete euca2ools para interactuar con nova.

```
root@nova-controller:~# sudo apt-get install -y euca2ools
```

Instalación de unzip.

Instalar la herramienta de unzip para extraer los archivos.

```
root@nova-controller:~# apt-get install unzip
```

Reinicio de los componentes.

```
root@nova-controller:~# /etc/init.d/libvirt-bin restart; sudo restart nova-network; sudo restart nova-compute; sudo restart nova-api; sudo restart nova-objectstore; sudo restart nova-scheduler; sudo restart nova-volume; sudo restart glance-api; sudo restart glance-registry
```

Creación de tablas en BD

Se debe crear las tablas necesarias en la base de datos, para ello ejecutamos el siguiente comando.

```
root@nova-controller:/var/lib/nova# nova-manage db sync
```

Creación de ip privadas virtuales para las instancias.

```
root@nova-controller:/bin# nova-manage network create 10.0.0.0/24 1 255
```

Resultado:

```
root@nova-controller:/bin# nova-manage network list
id      IPv4      IPv6      start address  DNS1      DNS2
VlanID  project  uuid
2013-01-27 22:10:34 DEBUG nova.utils [req-abfbb287-fe75-4b90-bc98-bc8f24162166 None None] backend <module 'nova.db.sqlalchemy.api' from '/usr/lib/python2.7/dist-packages/nova/db/sqlalchemy/api.pyc'> from (pid=3871) __get_backend /usr/lib/python2.7/dist-packages/nova/utils.py:658
1       10.0.0.0/24  None     10.0.0.2      8.8.8.8    8.8.4.4
None    None     0d44b42d-41f1-48a7-8975-2e9296fc45e8
```

Creación de ip privadas reales.

Se creara direcciones IP privadas para la asociación de la red LAN con las ip virtuales será a partir de 172.18.18.64/28.

```
root@nova-controller:/bin# nova-manage floating create 1172.18.18.64/28
```

Reinicio de los servicios

```
root@nova-controller:~# /etc/init.d/libvirt-bin restart; sudo restart nova-network; sudo restart nova-compute;sudo restart nova-api; sudo restart nova-objectstore; sudo restart nova-scheduler; sudo restart nova-volume; sudo restart glance-api; sudo restart glance-registry
```

Status servicios de Nova.

Se verificara que todos los servicios estén levantados.

```
root@nova-controller:~# cd /var/lib/nova/instances/nova-manager service list
```

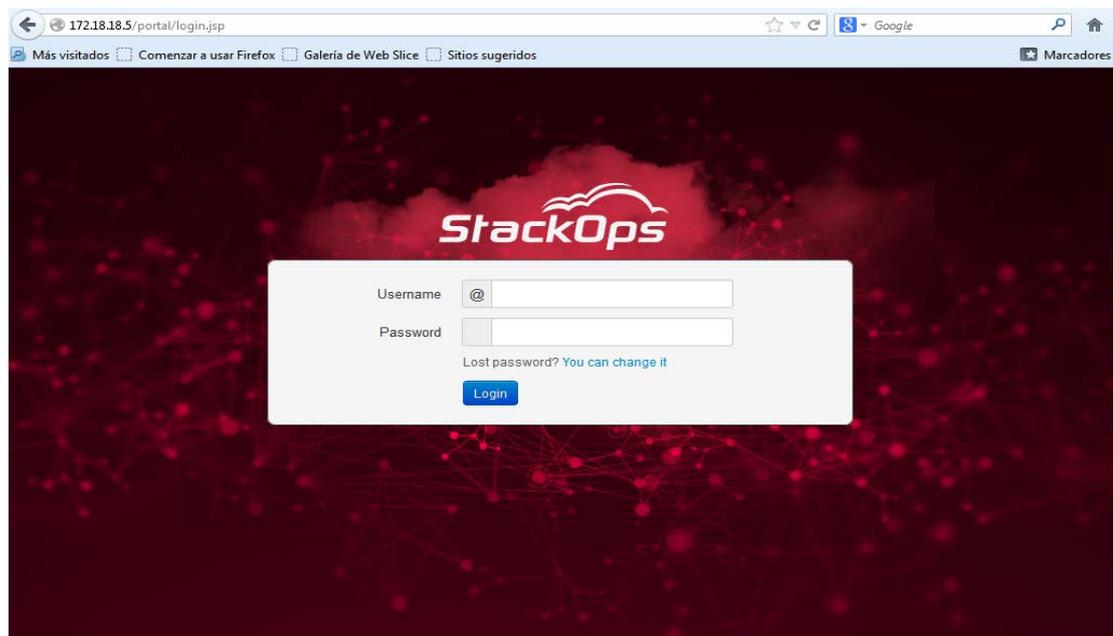
Resultado:

```
root@nova-controller:/var/lib/nova/instances# nova-manage service list
2013-01-27 23:04:23 DEBUG nova.utils [req-8e3efd00-7a09-4c3b-b21c-e8df77b49ea3 None None] backend <module 'nova.db.sqlalchemy.api' from '/usr/lib/python2.7/dist-packages/nova/db/sqlalchemy/api.pyc'> from (pid=2135) __get_backend /usr/lib/python2.7/dist-packages/nova/utils.py:658
Binary      Host                Zone      Status  State Updated_At
nova-scheduler nova-controller     nova      enabled :-) 2013-01-27 23:04:23
nova-consoleauth nova-controller     nova      enabled :-) 2013-01-27 23:04:23
nova-network  nova-controller     nova      enabled :-) 2013-01-27 23:04:15
```

Podemos observar una carita feliz que significa que los servicios están corriendo correctamente, al contrario cuando aparecen una X es porque los servicios se encuentran todavía desactivados.

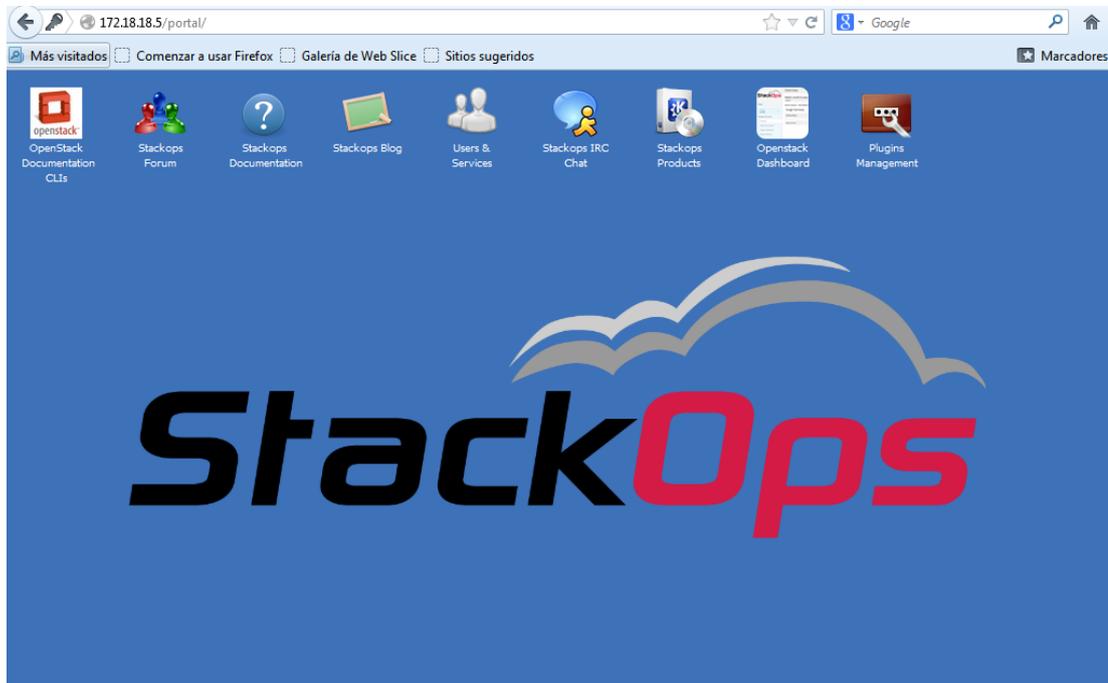
Openstack Dashboard

Ingresamos a la dirección ip no será necesario ningún puerto y nos aparecerá una elegante interfaz que será ya el panel de control de la virtualización el cual nos pedirá la validación para su ingreso.



Interfaz Stackops.
Fuente [Autor]

Una vez que se ingresa al panel de control de la instancia creada tendremos la siguiente interfaz, tal cual fuera un sistema operativo donde se podrá administrar y configurar las aplicaciones requerida por el usuario final.



Dashboard Stackops.
Fuente [Autor]

OpenStack es una colección de tecnologías open source que permitió sea posible realizar esta pequeña demostración de una infraestructura de cloud computing como servicio. Básicamente se ha desplegado una interfaz en la cual se puede configurar y administrar el espacio virtual solicitado por el usuario final.

Se brinda el servicio con todas las seguridades para que las pequeñas y grandes empresas se sientan seguras en depositar toda su información de TI.

El propósito de este proyecto sin lugar a duda, es ofrecer una alternativa a lo convencional como es un data center, y tener el acceso desde cual lugar que los administradores de la información lo requieran y así aprovechar los beneficio que esta tecnología nos ofrece.

4.3 Instalación del sistema operativo Stackops.

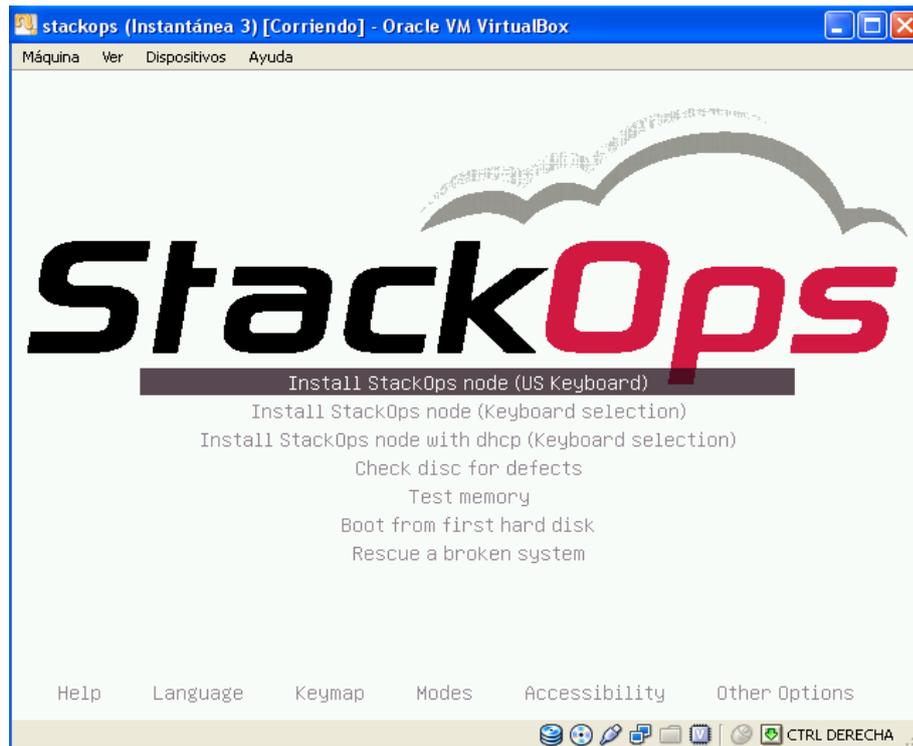
Los paquetes para poder instalar distro openstack se lo puede descargar en un solo disco donde encontramos todos los paquetes necesarios, podemos descargar la distribución actual del siguiente link <http://sourceforge.net/projects/stackops/files/>.

Luego de la descarga debemos quemar la imagen ISO en un CD o de un pendrive e iniciar el proceso de instalación al igual que cualquier sistema operativo. Para el nodo controlador, el nodo de almacenamiento, y para el cliente se utilizara una versión de 64 bits.

4.3.1 Nodo Controlador

En la pantalla principal de la instalación se muestran todas las opciones de configuración, se debe seleccionar el tipo de componente que se requiere. En nuestro caso instalaremos un nodo controlador con los siguientes componentes, controlador, red y volumen, el procesamiento (Compute) se lo realiza en otro servidor.

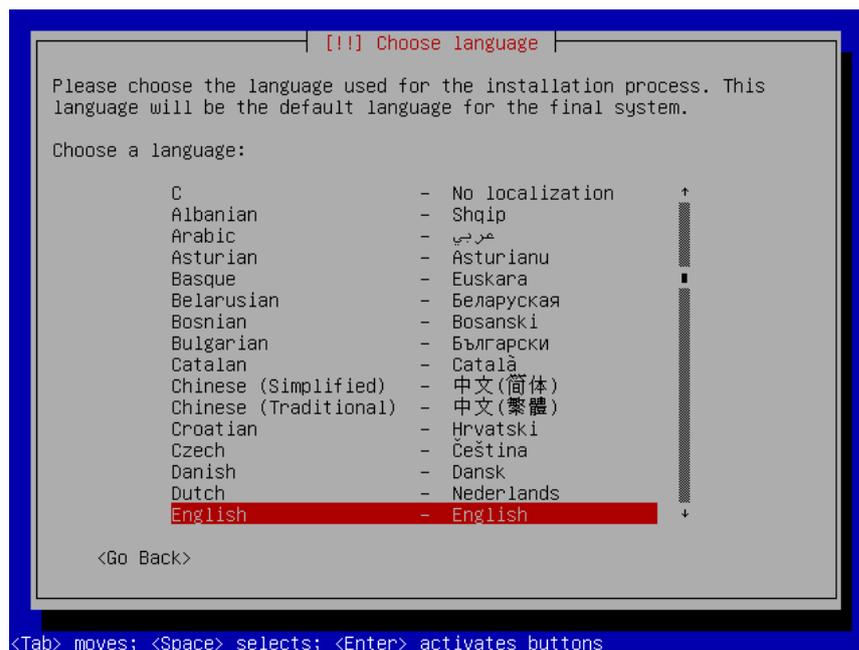
Como implementaremos un nodo doble, escogeré la primera opción de la pantalla como se muestra en la figura 5.3, pero si queremos el teclado español tendremos que seleccionar la segunda opción, si tenemos la configuración de DHCP en la red podremos elegir la tercera opción, y el resto son herramientas adicionales con otros tipos de configuraciones.



5.3. Instalación del Nodo Controlador.
Fuente [Autor]

4.3.2 Seleccione el idioma para la instalación.

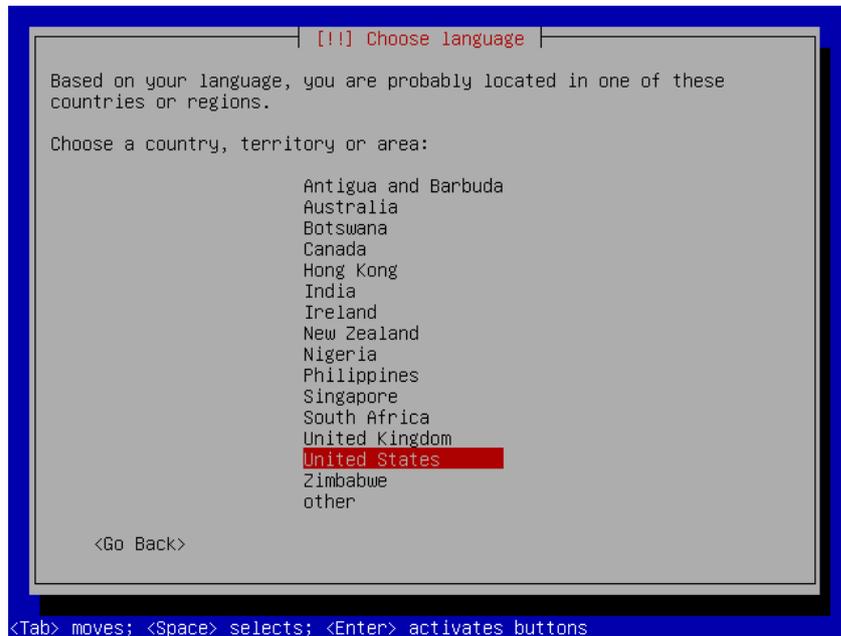
Seleccionaremos el idioma y la ubicación para la configuración.



Configuración del idioma.
Fuente [Autor]

4.3.3 Seleccione el lenguaje de su región.

El sistema nos sugiere elegir el idioma de su región. Seleccionaremos nuestra ubicación geográfica, dándole click en la opción deseada.

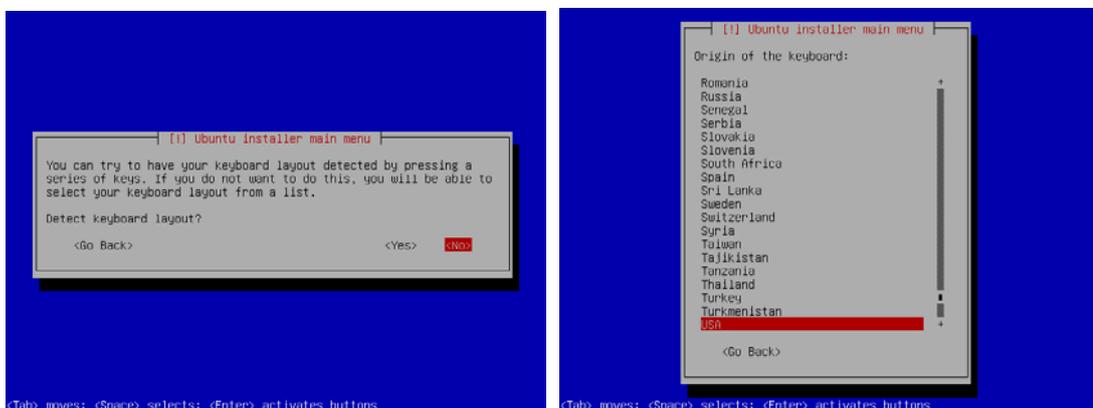


Configuración del lenguaje.

Fuente [Autor]

4.3.4 Elegimos el modelo del Teclado

El asistente nos ayuda a encontrar la configuración del teclado, omitimos presionando no y seleccionamos la configuración de teclado manualmente.

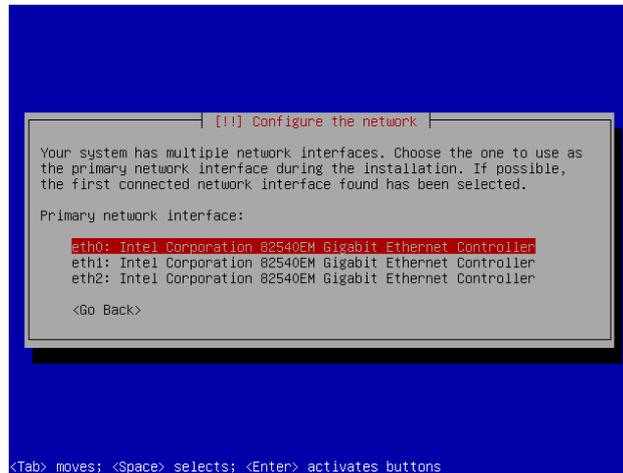


Configuración del teclado.

Fuente [Autor]

4.3.5 Configuración de la interfaz de red.

Nuestro nodo controlador tiene dos interfaz de red, seleccionaremos la interfaz eth0 como principal para la gestión de openstack.

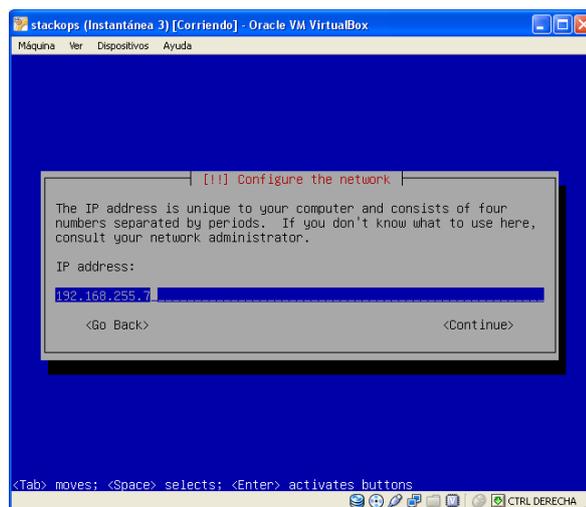


Configuración de la interfaz de red.

Fuente [Autor]

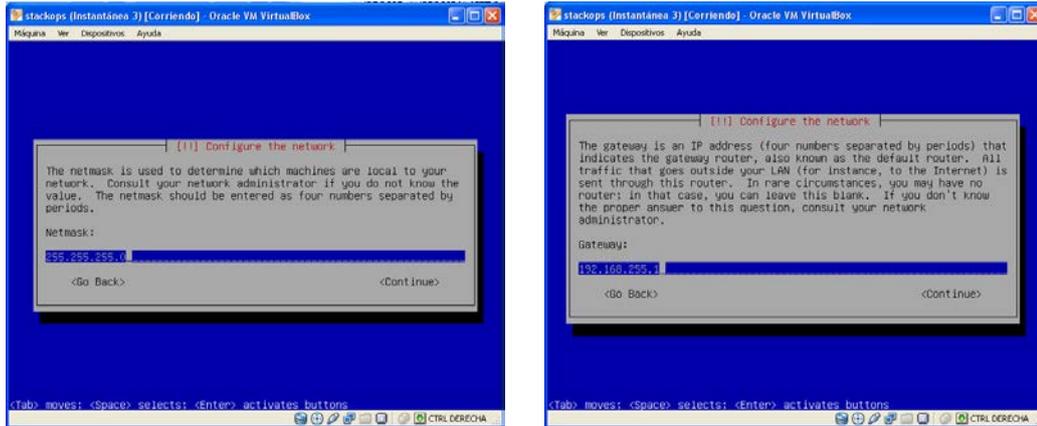
4.3.6 Dirección ip, mascara, y puerta de enlace del controlador.

Aquí configuramos la dirección IP de la red de gestión. Solo el administrador tendrá acceso a este sistema.



Configuración de la dirección Ip.

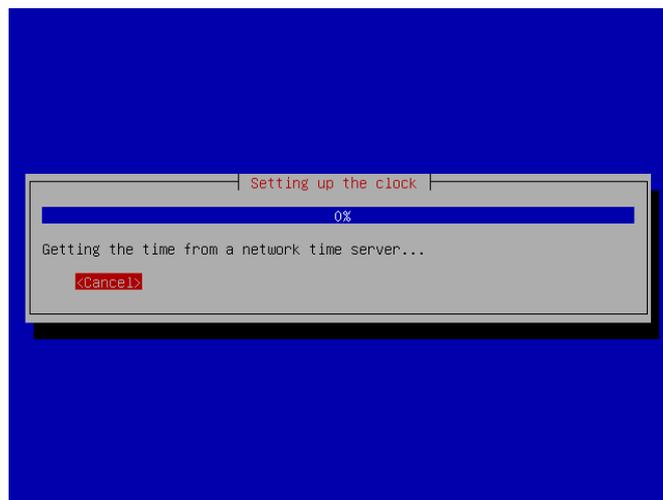
Fuente [Autor]



Configuración de la máscara de red y del gateway.
Fuente [Autor]

4.3.7 Sincronización de las configuraciones.

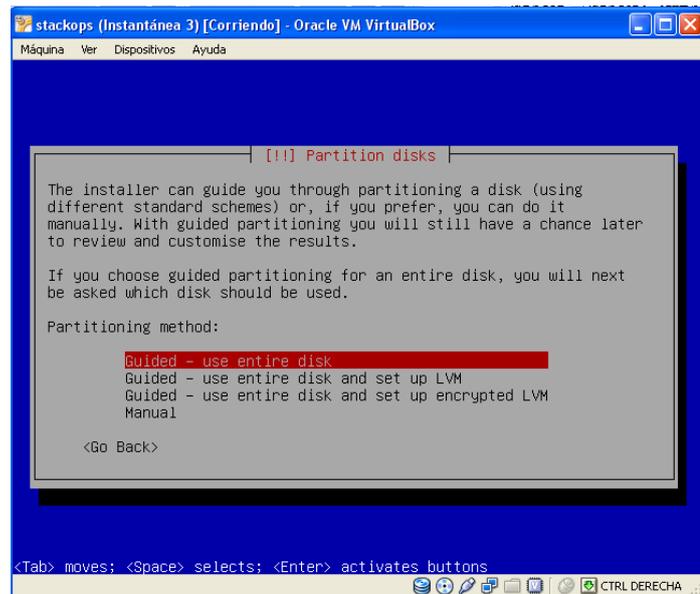
Tenemos que verificar que cada nodo este sincronizado con el resto de los nodos.



Sincronización de las configuraciones.
Fuente [Autor]

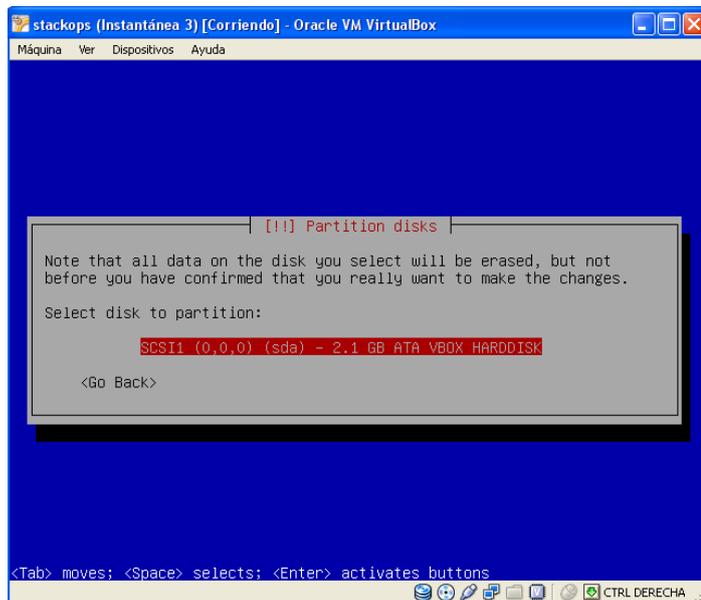
4.3.8 Particionamiento de los discos.

Se debe realizar las particiones de los discos, por defecto, se escoge la segunda opción sin embargo en pruebas realizadas tengo un mejor rendimiento eligiendo todo el disco duro sin realizar particiones de raíz.



Partición de discos.
Fuente [Autor]

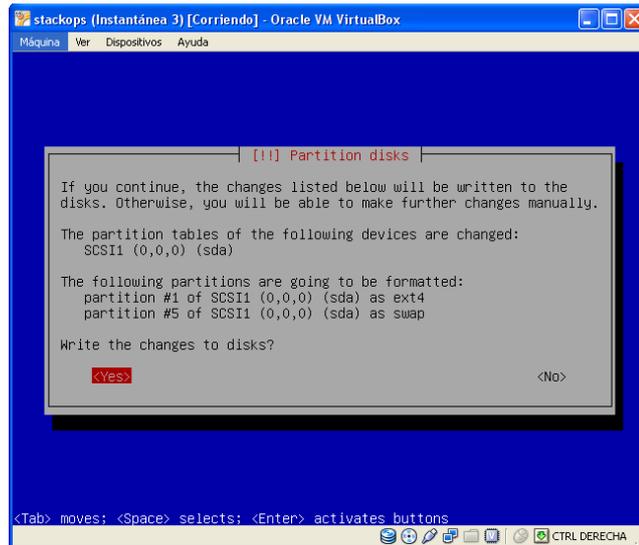
4.3.9 Confirmación de la partición.



Confirmación de los particionamiento de disco.
Fuente [Autor]

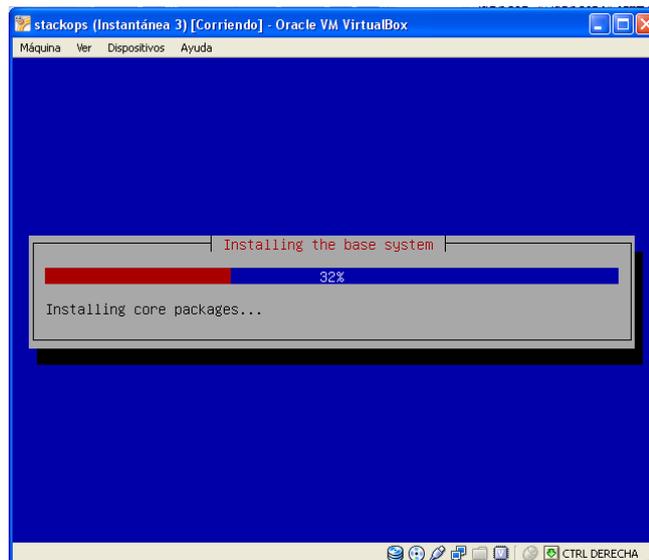
4.3.10 Particiones del disco básico

Por defecto, el sistema emitirá una partición del disco con una partición raíz y una partición swap.



Partición del disco básico.
Fuente [Autor]

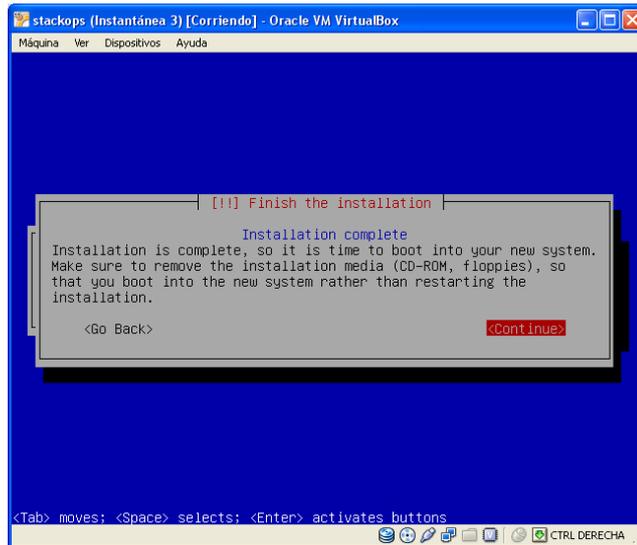
Configuración de los paquetes adicionales para Openstack



Configuración de componentes adicionales.
Fuente [Autor]

4.3.11 Instalación finalizada.

Se acaba de instalar un nodo controlador y es necesario continuar con la configuración de cada uno de los componentes del nodo de almacenamiento (Compute) al final se debe reiniciar el servidor, y se debe continuar con la configuración del smart installer.

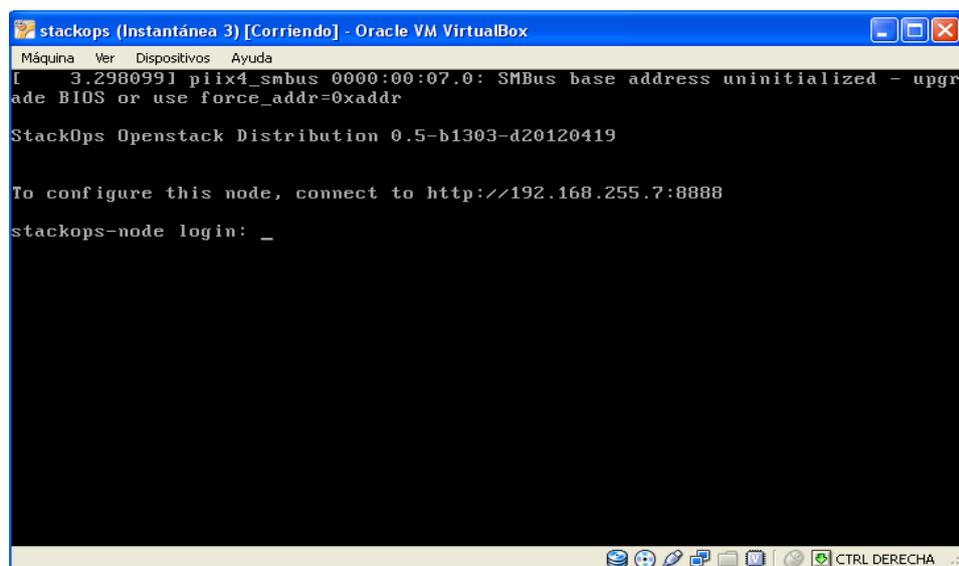


Finalización de la instalación.
Fuente [Autor]

4.3.12 Conectar con Smart Installer

Continuando con la configuración del nodo controlador, necesitamos conectarnos a la aplicación de instalación smart web, por defecto, para ingresar tenemos que ingresar con el usuario: “root” y contraseña “password”.

Así se validara el ingreso por primera vez, luego de esto hay que cambiar estas configuración con una nueva contraseña para la administración de las instancias que se crearan mas adelante.



Conexión al fron-end.
Fuente [Autor]

Probamos el estado del proceso *stackops* con el siguiente comando.

```
root@nova-controller:~# status stackops
stackops start/running, process 770
root@nova-controller:~#
```

4.4 Configuración del Nodo controlador con Smart Installer

4.4.1 Registro de datos.

El primer paso para la configuración es conectarse a la url que nos muestra en la pantalla del nodo, luego de esto llenamos los campos que nos solicitan en la viñeta “registro” y volvemos a la pestaña “login” e iniciamos sesión con los datos creados.

Los datos a ingresar son a considerar por la compañía de stackops pero son de registros tradicionales además la licencia para poder experimentar una prueba de entrenamiento es de sesenta días.

StackOps is a bare-metal distribution verified, tested and designed to reach as many users as possible thanks to a new and simple installation process. You only need to download the ISO image with the distro and install it on one or more servers. In a few minutes you will be able to enjoy the power of the Cloud for your own!

Stackops uses Openstack plus many other open source technologies. Openstack is a collection of open source technologies delivering a massively scalable cloud operating system. Backed by Rackspace, NASA, Dell, Citrix, Cisco, Canonical and over 50 other organizations, Openstack, has grown to be a global software community of developers, technologists, researchers and corporations.

Thanks to the "Smart Cloud Installer" you will be able to deploy different Cloud architectures assisted by our automated assistant. **We strongly recommend to register before starting with the installation process.** The "Smart Cloud Installer" will keep track of your configuration and will save you from the hassle of a manual configuration of the solution. If you decide to install a Cloud Node without registering, be aware of the fact that the application will not remember your name and will not guarantee the consistency of the data entered. Click [here](#) if you want to proceed with the anonymous installation.

The assistant will report information about how the installation and configuration process is going. If the assistant detects a problem in the hardware or software configuration of the base system, it will be reported before continuing.

So if you are ready, please register or log in to continue.

No thanks, please [continue](#)

© 2011 Stackops Technologies. All rights reserved. [Terms & Conditions](#)

5.4. Registro de datos.

Fuente [Autor]

4.4.2 Selección de la arquitectura a implementar.

Ahora usted puede conectarse y comenzar el proceso de configuración, en el siguiente paso elegiremos el tipo de arquitectura a implementar, en nuestro caso ya habíamos elegido la de del Nodo Doble.

Select your deployment architecture

Single Node | **Dual Node** | Multi Node | StackOps Reference Architecture (Advanced Multi Node)

This is the minimum installation for production. All the components of the Openstack solution plus others are installed in Node Controller, but the Compute node is installed in different node. Both nodes can run in a virtualized environment, but it's recommended to run the Compute Node in a physical server. You can add more Compute Nodes at any given time.

You need TWO Network cards on each server to run the solution successfully. Since the requirements for this environment are higher than the Single Node architecture, we recommend to review carefully the Global System Requirements. Please visit the Dual node deployment documentation for further details.

Start deployment

5.5. Selección de la arquitectura a implementar.

Fuente [Autor]

4.4.3 Arquitectura Nodo Doble

En las pestañas del tipo de configuración de la nube seleccionaremos la opción “*Dual Node*” y daremos click el botón “*Start deployment*”, luego el asistente nos indicara si el hardware y el software del equipo nos permiten instalar openstack.

StackOps | openstack

Zones List | Support | Documentation | Bugs/Improvements

Welcome back gabriel! | Logout

hardware

Name	Speed	Cores
Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz	1000	1
Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz	1000	1
Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz	1000	1
Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz	1000	1

Virtualization	Value
Virtualization	true
RAM Size	4066136064

Device	Size	Used	Mount Point
/dev/sda	500107862016	-1	
/dev/sda1	247390999488	-1	
/dev/sda2	1024	-1	
/dev/sda5	11911823360	-1	

Before continuing with the installation process, you must verify that your hardware can support the architecture selected for this node. The following parameters will be checked by the Smart Installer:

- **Virtualization:** if Virtualization extensions are not enabled or not exist, then only QEMU emulator will be available. Keep in mind that there is a huge penalization in performance compared to KVM, the default hypervisor supported by the Stackops Distribution. If you think your servers have virtualization extensions but the Installer cannot detect it, please review the System BIOS of your servers.
- **RAM Size:** different architectures and nodes have different needs of RAM memory. Please check that you have enough RAM to run the solution.
- **Space in disk:** different node roles have different needs of disk space. Please check that you have enough space in your disk.
- **Network interfaces:** again the network configuration can differ in the nodes. Please check that you have enough NICs.

5.6. Requerimiento del hardware.

Fuente [Autor]

4.4.4 Comprobación de los requisitos del Software.

En este paso el smart installer del nodo comprueba si el sistema operativo nos permite instalar openstack. Además nos muestra los datos de la tarjeta de red y cuantas tenemos físicamente con la ip ya anteriormente configurada.

 You are not using the latest version of the StackOps distro (0.5-b1312-d20120510), please download it from [here](#).

software

Operating System Linux/stackops-node/2.6.32-28-server/#55-Ubuntu SMP Mon Jan 10 23:57:16 UTC 2011/x86_64/0.5-b1303-d20120419 (uname)

Hostname	nova-controller
DNS 1	8.8.8.8
DNS 2	8.8.4.4

Name	Address	Netmask	Gateway	DHCP Client
eth0	192.168.255.7	255.255.255.0	192.168.255.1	false

[Back](#) [Continue](#)

The underlying operating system has been prepared to run Openstack smoothly. There are some parameters that must be verified before continuing:

- **Operating System uname:** We trace this information (only if you have registered previously) to troubleshoot different issues in different versions.
- **Hostname:** By default the hostname is 'openstack-node' because it still has no role assigned. If there is a different hostname, probably the node is not ready for a new deployment.
- **DNSx:** By default we use the Google DNS.
- **Network Interfaces:** The TCP/IP network configuration of the system.

5.7. Comprobación de los requisitos del Software.

Fuente [Autor]

4.4.5 Topología de la red

El controlador sólo utiliza la red de gestión openstack. Optaremos por la tarjeta de red que se utilizará para esta red.

[Zones List](#) [Support](#) [Documentation](#) [Bugs/Improvements](#) Welcome back **gabriel!** | [Logout](#)

Network Topology (controller,network,volume)

management eth0=192.168.255.7
service eth1=null
public eth0=192.168.255.7

[Back](#) [Continue](#)

Stackops Distro allows different network configurations for Openstack Nova. So you need to know the three different networks available:

- **Management:** this network connects all the components of the Openstack architecture. It's mandatory and you should think of this network as the wire that connects everything. As a rule of thumb, use the first ethernet NIC available in ALL nodes.
- **Service:** it connects all the Compute Nodes and Network Node. This network is used by the guest VMs, so it's the network that the Cloud User will see. Due to the chosen network type used (FlatDHCPManager), you need to leave an unconfigured NIC on each node to this network. Again, use the SECOND ethernet NIC available in compute and network nodes in ALL nodes.
- **Public:** it connects the Network Node to the rest of the world (Internet or intranet). This NIC cannot be the same used in Service. Management and Public can share the same Network interface and the same IP addressing, but **do not share the Service Network interface with any other service**, it won't work.

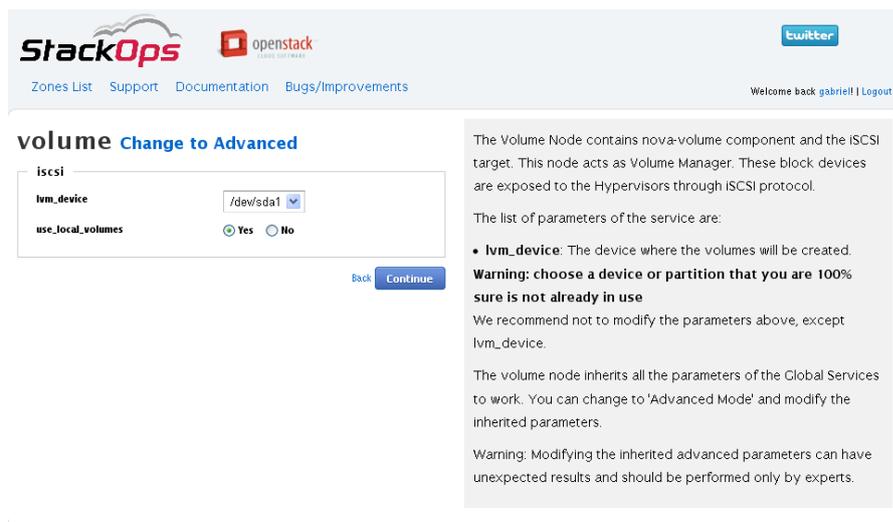
© Transnenc

5.8. Configuración de la topología de la red.

Fuente [Autor]

4.4.6 Configuración del volumen.

En la configuración de volúmenes, elegiremos el disco total sin partición como se lo menciono en la sección 4.3.8. Esta configuración se la realizo debido a que, en el momento de terminar la instalación creaba conflictos y se determino que era necesario realizar la partición debido a la función del nodo controlador y al final se realizo la instalación con resultados positivos.



The screenshot shows the StackOps interface for configuring a volume. The page title is "volume Change to Advanced". Under the "iscsi" section, there is a form with the following fields:

- lvm_device**: /dev/sda1
- use_local_volumes**: Yes (selected), No

A "Continue" button is visible. To the right of the form, there is explanatory text:

The Volume Node contains nova-volume component and the iSCSI target. This node acts as Volume Manager. These block devices are exposed to the Hypervisors through iSCSI protocol.

The list of parameters of the service are:

- lvm_device**: The device where the volumes will be created. **Warning: choose a device or partition that you are 100% sure is not already in use**

We recommend not to modify the parameters above, except lvm_device.

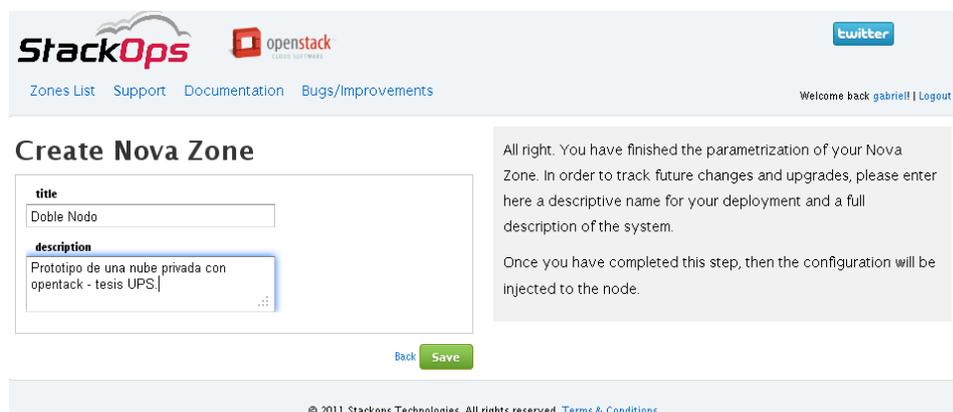
The volume node inherits all the parameters of the Global Services to work. You can change to 'Advanced Mode' and modify the inherited parameters.

Warning: Modifying the inherited advanced parameters can have unexpected results and should be performed only by experts.

5.9. Configuración del Volumen.
Fuente [Autor]

4.4.7 Creación de una zona.

Introduciremos aquí un nombre descriptivo para su implementación y una descripción completa del sistema. Una vez que haya completado este paso, la configuración será colocada en el nodo.



The screenshot shows the StackOps interface for creating a Nova Zone. The page title is "Create Nova Zone". There is a form with the following fields:

- title**: Doble Nodo
- description**: Prototipo de una nube privada con openstack - tesis UPS.

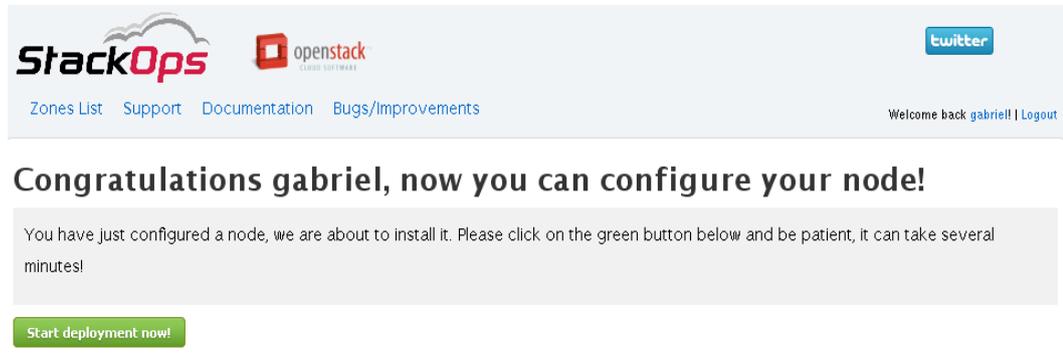
A "Save" button is visible. To the right of the form, there is explanatory text:

All right. You have finished the parametrization of your Nova Zone. In order to track future changes and upgrades, please enter here a descriptive name for your deployment and a full description of the system.

Once you have completed this step, then the configuration will be injected to the node.

6.0. Creación de zonas.
Fuente [Autor]

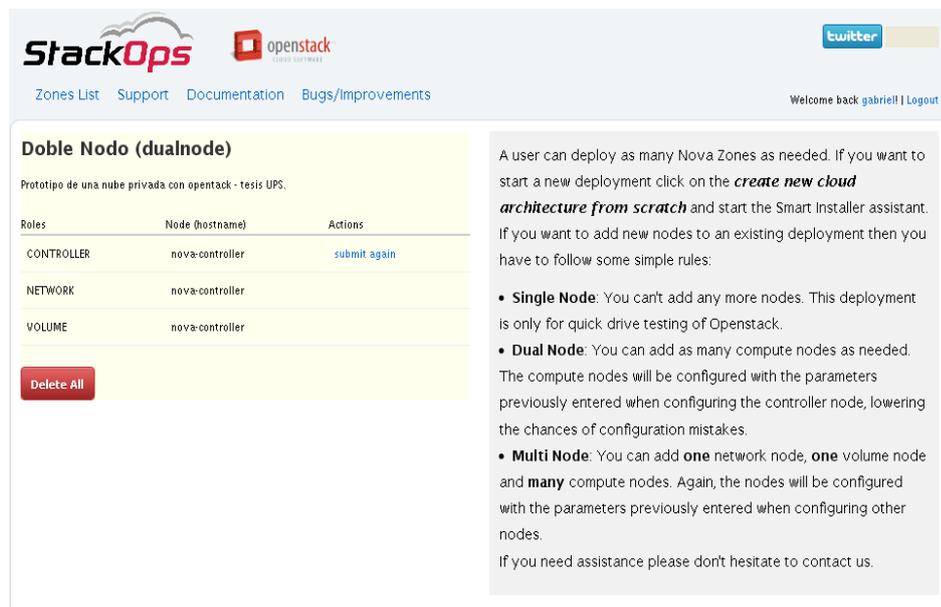
Por último nos aparece en la pantalla la confirmación y presionaremos “Start deployment now”.



Confirmación de la configuración de Stackops.
Fuente [Autor]

4.4.8 Finalización de la Instalación

Podemos observar que la configuración que realizamos fue exitosa. En el nodo controlador nos aparecerá configurado la red el controlador y el volumen. Las partes más importantes del almacenamiento se las realizara en el servidor del nodo de compute.



6.1. Finalización de la Instalación
Fuente [Autor]

4.5 Configuración del Nodo de almacenamiento (Compute) con Smart Installer

La configuración del nodo de almacenamiento es igual a los pasos mencionados anteriormente desde la sección 4.4 lo único que cambia que la parte de procesamiento (compute) aparece al final de la instalación como se muestra en la figura 5.9.

Roles	Node (hostname)	Actions
CONTROLLER	nova-controller	reconfigure
NETWORK	nova-controller	
VOLUME	nova-controller	
COMPUTE		add compute

6.2. Configuración del Nodo de almacenamiento (Compute).
Fuente [Autor]

A user can deploy as many Nova Zones as needed. If you want to start a new deployment click on the **create new cloud architecture from scratch** and start the Smart Installer assistant. If you want to add new nodes to an existing deployment then you have to follow some simple rules:

- **Single Node:** You can't add any more nodes. This deployment is only for quick drive testing of Openstack.
- **Dual Node:** You can add as many compute nodes as needed. The compute nodes will be configured with the parameters previously entered when configuring the controller node, lowering the chances of configuration mistakes.
- **Multi Node:** You can add **one** network node, **one** volume node and **many** compute nodes. Again, the nodes will be configured with the parameters previously entered when configuring other nodes.

If you need assistance please don't hesitate to contact us.

4.5.1 Configuración del hipervisor.

Luego al ingresar, podremos ver la arquitectura que hemos creado para esto el nodo controlador y el nodo de computo lo debemos colocar en una misma red, luego en el botón de “*add compute*” le damos click y los siguientes pasos es igual a la configuración nodo del controlador sin embargo al final tendremos que en la configuración del nodo de computo se debe elegir el tipo de hipervisor más recomendable que es “Qemu” como se muestra en la imagen 6.0.

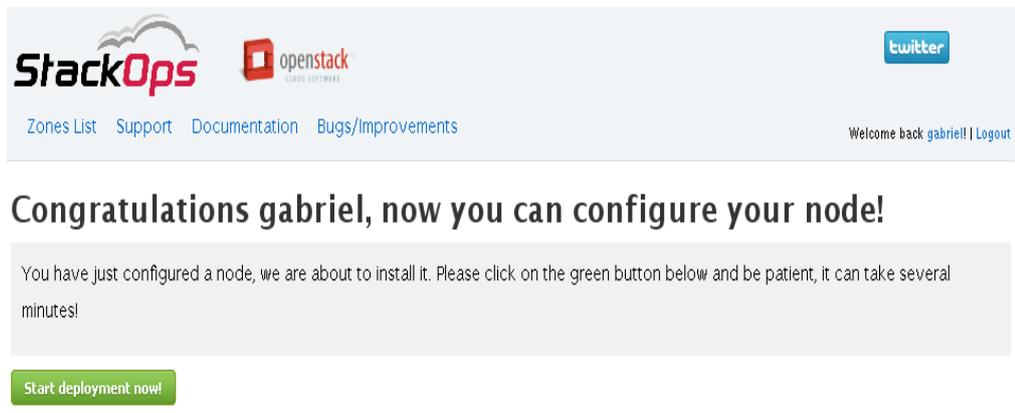
Sin embargo se probó la instalación con el hipervisor kvm y los resultados fueron los mismos pero para llevar el mismo proceso que sugieren stackops se utilizo el hipervisor mencionado.

The screenshot shows the 'compute' configuration page in StackOps. The 'libvirt' section has a 'type' dropdown menu with 'kvm' selected and 'qemu' highlighted. The 'interfaces' section has a 'flat_interface' field with 'eth1'. The 'iscsi' section has 'ip_prefix' set to '192.168.255.8', 'num_targets' set to '100', and 'storage-hostname' set to 'nova-controller'. A 'Continue' button is at the bottom right. A text box on the right explains the Compute Node components and provides a warning about modifying inherited parameters.

6.3. Configuración del hipervisor.
Fuente [Autor]

4.5.2 Finalización de la instalación.

Al final nos aparece la confirmación de la correcta instalación.



Configuración del nodo compute.
Fuente [Autor]

Luego, nos aparecerá la sincronización entre el nodo controlador y el nodo de almacenamiento (Compute) y podremos empezar a realizar la parte de la virtualización ya teniendo los dos servidores conectados entre sí.

StackOps **openstack** [twitter](#)

[Zones List](#) [Support](#) [Documentation](#) [Bugs/Improvements](#) Welcome back gabriel! | [Logout](#)

fgh (dualnode)

fgh

Roles	Node (hostname)	Actions
CONTROLLER	nova-controller	submit again
NETWORK	nova-controller	
VOLUME	nova-controller	
COMPUTE	nova-compute-1	submit again delete

[Delete All](#)

A user can deploy as many Nova Zones as needed. If you want to start a new deployment click on the **create new cloud architecture from scratch** and start the Smart Installer assistant. If you want to add new nodes to an existing deployment then you have to follow some simple rules:

- **Single Node:** You can't add any more nodes. This deployment is only for quick drive testing of Openstack.
- **Dual Node:** You can add as many compute nodes as needed. The compute nodes will be configured with the parameters previously entered when configuring the controller node, lowering the chances of configuration mistakes.
- **Multi Node:** You can add **one** network node, **one** volume node and **many** compute nodes. Again, the nodes will be configured with the parameters previously entered when configuring other nodes.

If you need assistance please don't hesitate to contact us.

6.4. Finalización de la instalación del nodo de almacenamiento.

Fuente [Autor].

4.6 Herramienta de gestión para la verificación de la conexión entre los nodos.

4.6.1 Herramienta Putty.

“Putty es un cliente de red que soporta los protocolos SSH, telnet y demás y sirve principalmente para iniciar una sesión remota con otra máquina o servidor. A pesar de su sencillez es muy funcional y configurable.

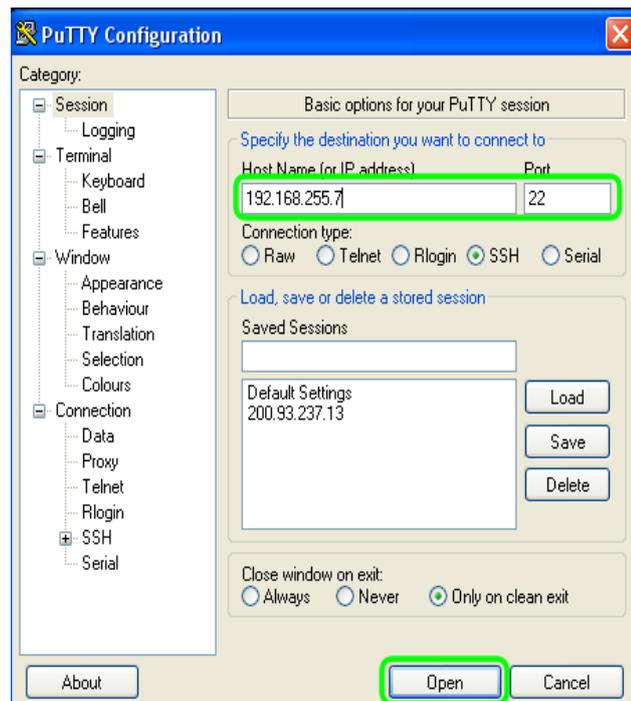
Se puede conseguir Putty en la siguiente dirección:
<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

Una vez descargado, comprobaremos que es un solo archivo ejecutable, que se puede hacer doble clic para ponerlo en marcha. Podemos copiar el archivo en cualquier directorio de nuestra computadora y hacer los accesos directos que deseemos anualmente para tenerlo a mano.

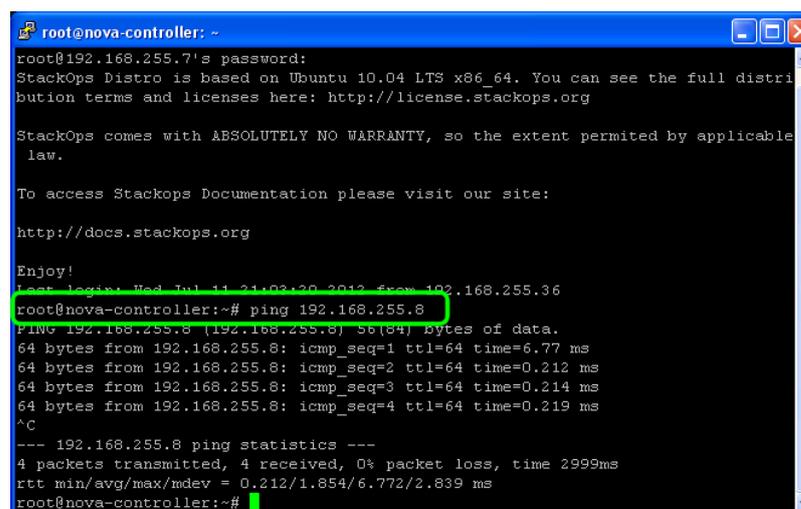
Al ejecutarlo accedemos a una ventana de administración de las conexiones, desde la que podemos configurar accesos a servidores remotos y guardarlos para entrar más tarde sin necesidad de introducir de nuevo los datos de acceso”. [34]

4.6.1.1 Conectividad nodo controlador con nodo de almacenamiento.

Una vez descargado ponemos la dirección IP del nodo controlador (192.168.255.7) y accedemos con el protocolo por defecto, luego iniciaremos en el botón “Open”. Luego verificaremos la conectividad entre los dos nodos con el comando ping como nos muestra la siguiente grafica de la ventana del D.O.S.

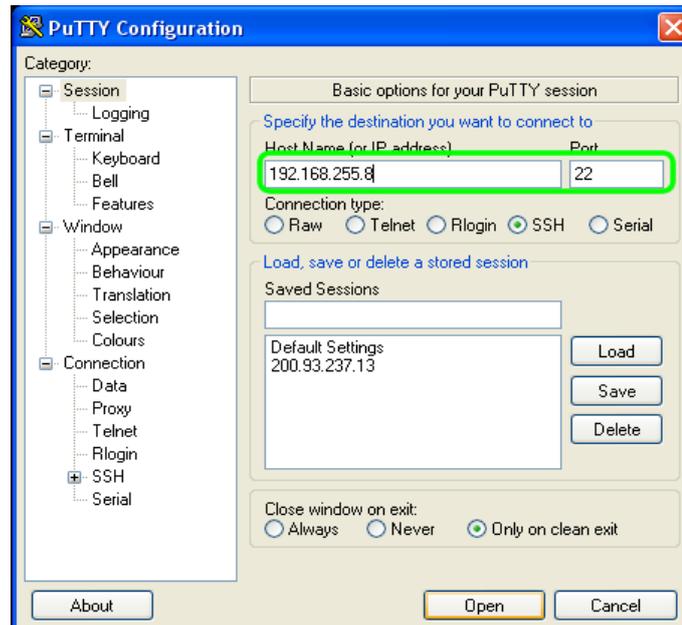


6.5 Conectividad nodo controlador con nodo de almacenamiento.
Fuente [Autor]



Conectividad nodo controlador desde consola.
Fuente [Autor].

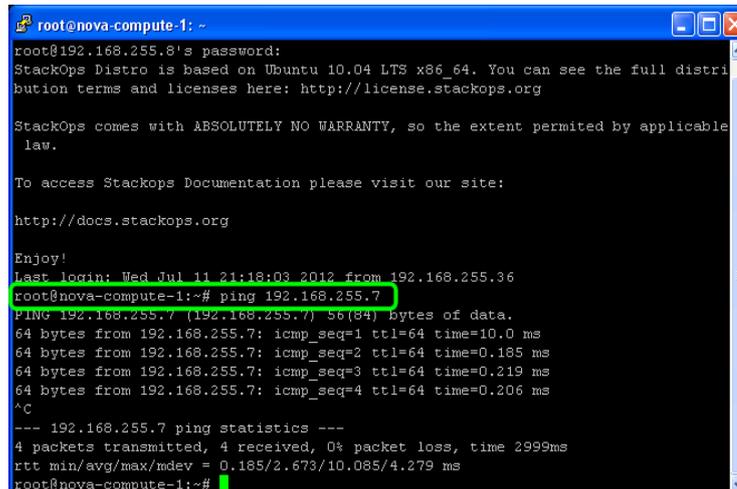
4.6.1.2 Conectividad nodo de almacenamiento con nodo controlador.



6.6 Conectividad nodo de almacenamiento con nodo controlador.

Fuente [Autor].

En este caso hacemos ping al nodo controlador obteniendo resultados positivos. Hasta este paso tenemos ya agregados y sincronizados los dos servidores.

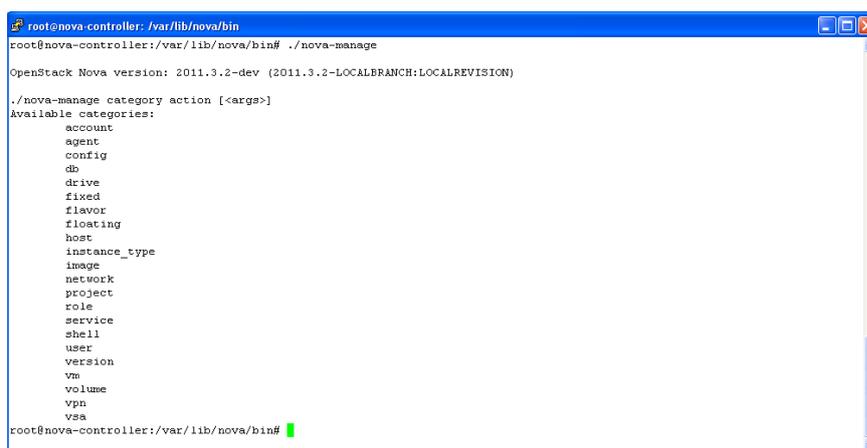


Conectividad nodo de almacenamiento desde consola.

Fuente [Autor].

4.7 Configuración de los componentes de Openstack

A partir de aquí hemos realizado todos los pasos necesarios que nos garantizan poder realizar la nube privada. Lo primero que haremos será ahora es verificar y comprobar el estado básico de nuestra plataforma. Con el comando **nova-manage** podemos hacer muchas cosas.

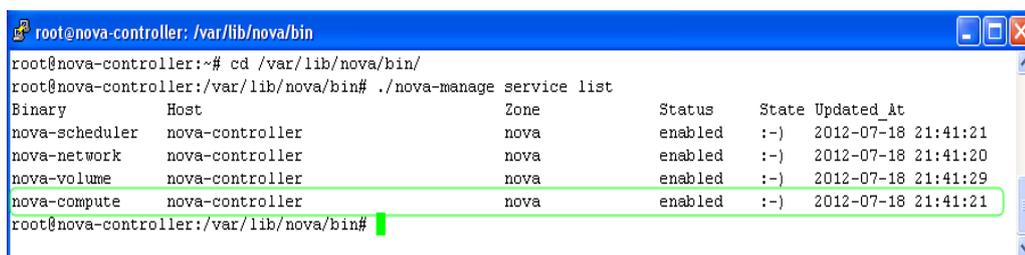


```
root@nova-controller: /var/lib/nova/bin
root@nova-controller: /var/lib/nova/bin# ./nova-manage
OpenStack Nova version: 2011.3.2-dev (2011.3.2-LOCALBRANCH:LOCALREVISION)

./nova-manage category action [<args>]
Available categories:
  account
  agent
  config
  db
  drive
  fixed
  flavor
  floating
  host
  instance_type
  image
  network
  project
  role
  service
  shell
  user
  version
  vm
  volume
  vpn
  vsa
root@nova-controller: /var/lib/nova/bin#
```

Componentes de Openstack..
Fuente [Autor].

Podremos ver los servicios activos y disponibles para el funcionamiento de la nube utilizando el comando: **nova-manage service list** como se muestra en la figura 6.7

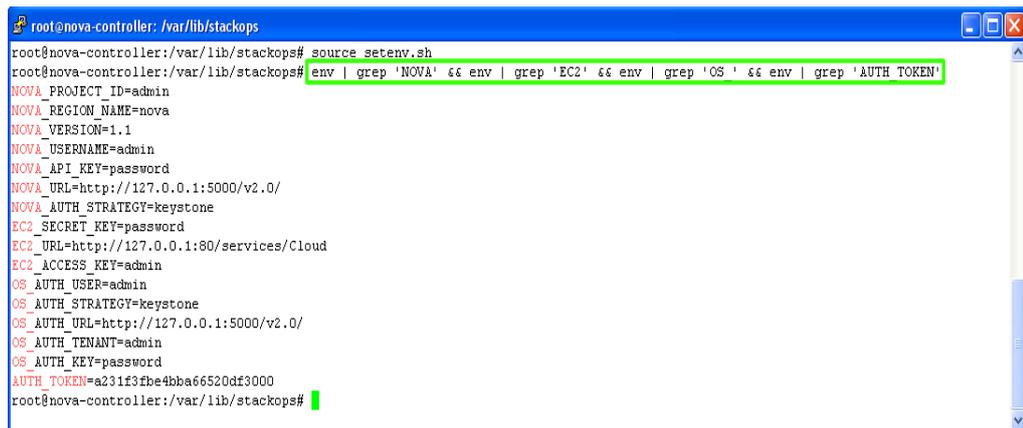


```
root@nova-controller: /var/lib/nova/bin
root@nova-controller: ~# cd /var/lib/nova/bin/
root@nova-controller: /var/lib/nova/bin# ./nova-manage service list
Binary      Host          Zone    Status   State Updated At
nova-scheduler nova-controller nova     enabled  :-      2012-07-18 21:41:21
nova-network  nova-controller nova     enabled  :-      2012-07-18 21:41:20
nova-volume   nova-controller nova     enabled  :-      2012-07-18 21:41:29
nova-compute  nova-controller nova     enabled  :-      2012-07-18 21:41:21
root@nova-controller: /var/lib/nova/bin#
```

6.7 Servicios disponibles.
Fuente [Autor].

Como podemos ver en el estado una cara feliz significa que el componente está en funcionamiento y conectado a la plataforma. Una "x" significa que el componente no está funcionando o no está conectado a la plataforma. En nuestro caso todos los servicios están funcionando perfectamente.

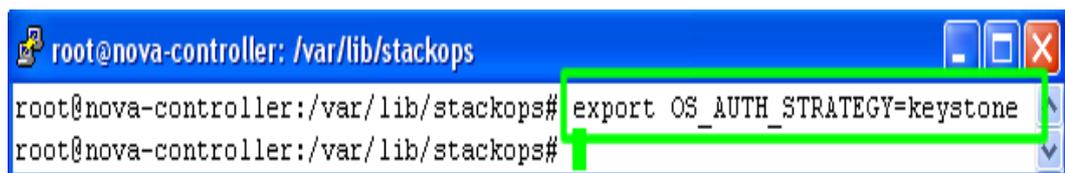
Por defecto, existe el usuarios para ingresar a la interfaz web (Dashboard) en nuestro nodo controlador que es "admin" con contraseña "password" al momento de acceder tenemos administración de las credenciales EC2. Antes debemos ejecutar un script para administrar este proyecto con la API de EC2 en la siguiente ruta / var / lib / stackops / setenv.sh. Este script creará todas las variables de "admin" y todo su entorno necesario para el usuario.



```
root@nova-controller: /var/lib/stackops
root@nova-controller:/var/lib/stackops# source setenv.sh
root@nova-controller:/var/lib/stackops# env | grep 'NOVA' && env | grep 'EC2' && env | grep 'OS' && env | grep 'AUTH TOKEN'
NOVA_PROJECT_ID=admin
NOVA_REGION_NAME=nova
NOVA_VERSION=1.1
NOVA_USERNAME=admin
NOVA_API_KEY=password
NOVA_URL=http://127.0.0.1:5000/v2.0/
NOVA_AUTH_STRATEGY=keystone
EC2_SECRET_KEY=password
EC2_URL=http://127.0.0.1:80/services/Cloud
EC2_ACCESS_KEY=admin
OS_AUTH_USER=admin
OS_AUTH_STRATEGY=keystone
OS_AUTH_URL=http://127.0.0.1:5000/v2.0/
OS_AUTH_TENANT=admin
OS_AUTH_KEY=password
AUTH_TOKEN=a231f3f3e4bba66520df3000
root@nova-controller:/var/lib/stackops#
```

Ejecución de script.
Fuente [Autor].

Existe un error en la escritura que no se establece de forma predeterminada, sin embargo hay un bug para solucionarlo y no tener más errores en el futuro



```
root@nova-controller: /var/lib/stackops
root@nova-controller:/var/lib/stackops# export OS_AUTH_STRATEGY=keystone
root@nova-controller:/var/lib/stackops#
```

Bug de error..
Fuente [Autor].

4.8 Descargar de las imágenes virtuales

Para realizar pruebas en la plataforma es necesario descargar una imagen de prueba y registrarla, la podemos descargar desde la ruta / var / lib / stackops / pubimages.sh al descargar la imagen se la podrá ver en el (Dashboard) ingresando en la interfaz web poder desplegar las instancias, para este caso nos descargamos la imagen de ttylinux.sh.

```
root@nova-controller: /var/lib/stackops
root@nova-controller:/var/lib/stackops# ./pub
pubcentos57.sh  pubcentos62.sh  pubdebian6.sh  pubttylinux.sh  pububuntu1004.sh
root@nova-controller:/var/lib/stackops# ./pubttylinux.sh
Downloading images...
--2012-07-19 14:57:26-- http://stackops.s3.amazonaws.com/images/ttylinux-uec-amd64-12.1_2.6.35-22_1.tar.gz
Resolving stackops.s3.amazonaws.com... 178.236.6.33
Connecting to stackops.s3.amazonaws.com [178.236.6.33]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24346106 (23M) [application/x-gzip]
Saving to: `'/tmp/ttylinux-uec-amd64-12.1_2.6.35-22_1.tar.gz'

100%[=====] 24,346,106 678K/s in 2m 22s

2012-07-19 14:59:54 (167 KB/s) - `'/tmp/ttylinux-uec-amd64-12.1_2.6.35-22_1.tar.gz' saved [24346106/24346106]

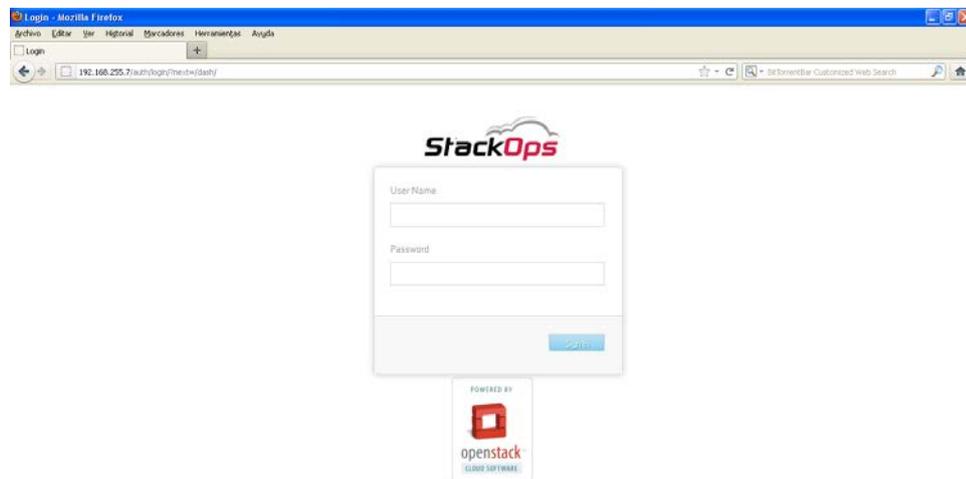
Added new image with ID: 2
root@nova-controller:/var/lib/stackops# █
```

Ejecución de script.
Fuente [Autor].

La imagen está lista para ser usada, nos vamos ahora al entorno web.

4.9 Interfaz web, Horizon Dashboard.

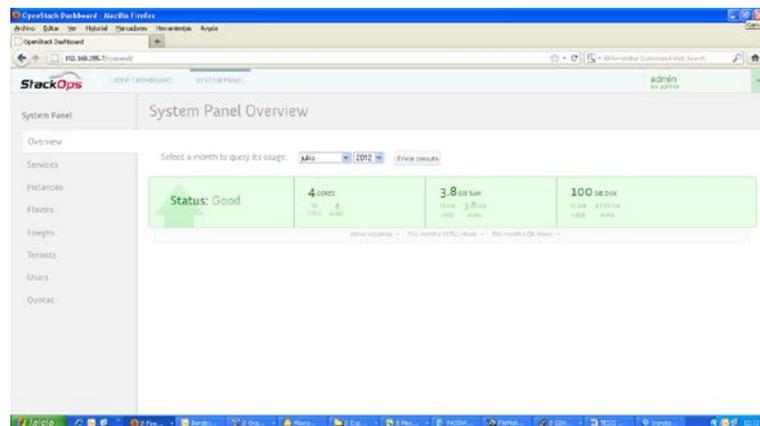
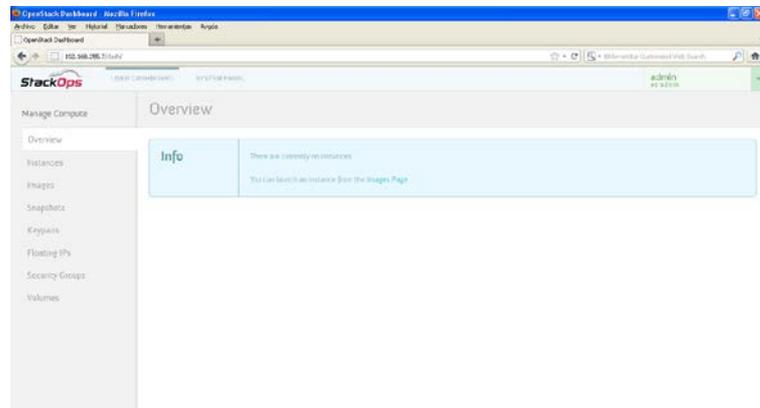
Para ingresar al “*Dashboard*” por medio de la web debemos introducir solo la dirección IP sin puerto (192.168.255.7), y nos aparecerá un formulario donde ingresaremos el usuario y las contraseñas ya mencionadas en la sección 4.7.



Ingreso interfaz web.
Fuente [Autor].

En la siguiente figura, tenemos el menú principal que es, “*User Dashboard*” y “*System Panel*” el cual nos muestra un esquema de los recursos para administrar la

nube privada, por ejemplo nos muestra las imágenes, instancias activas, volúmenes, grupo de seguridad etc. Cuanto más compute y volumen tengamos más recursos nos permitirá crear la ventana.



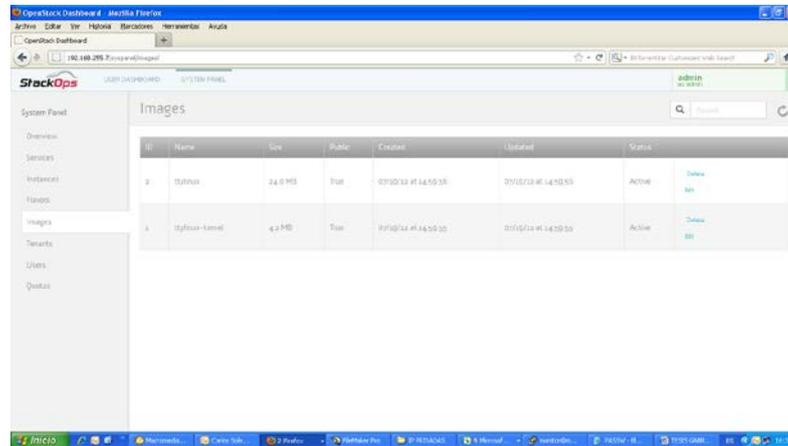
Dashboard.
Fuente [Autor].

La parte de servicios es un resumen en tiempo real de los servicios activos y su estado.



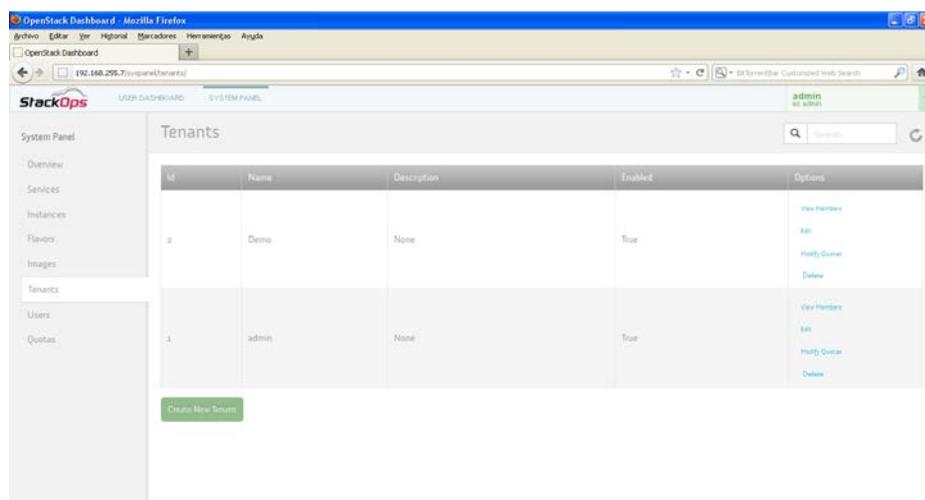
Servicios Activos.
Fuente [Autor].

En la parte de imagen, podemos ver las imágenes que ya hemos descargado como se lo menciona en la sección 4.8. En nuestro caso las dejamos como están. Cuando queramos subir una nueva imagen será aquí donde se las podrá ver y administrar ya sea editándola o eliminándola de la interfaz.



Alojamiento de imágenes.
Fuente [Autor].

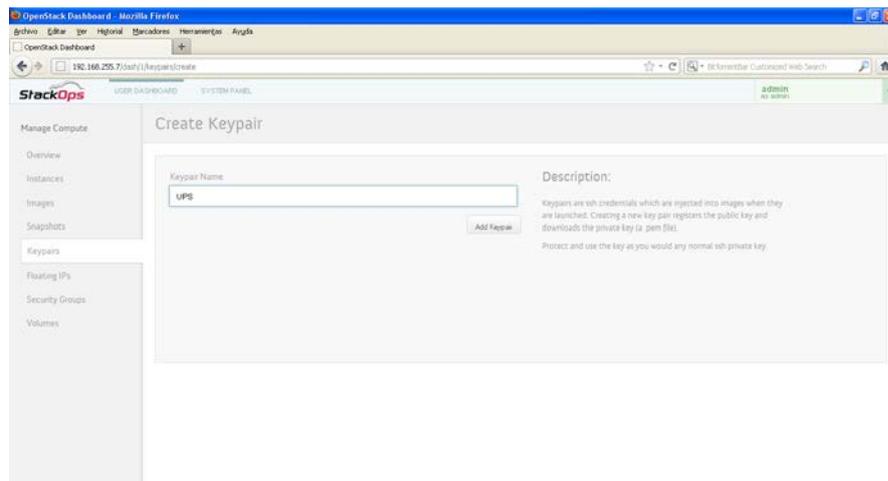
En el menú de “Tenants y Users” crearemos los usuarios para el proyecto para que sea el que defina los diferentes recursos según sus requerimientos esto se lo realiza dando un click en “Create new tenants”, por ejemplo crearemos vCPU, vRAM, instancias, y los core’s, todo con los permisos ya establecidos.



Creación de Usuarios.
Fuente [Autor].

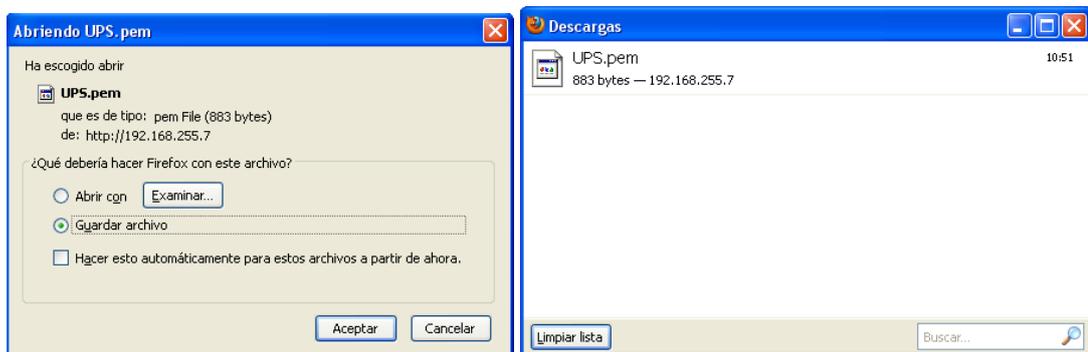
4.10 Creación de Instancias.

Para la administración de las instancias debemos crear un certificado y así configurarlas en el “Dashboard” luego ir al menú de “Keypairs” y generamos el certificado que necesitamos haciendo un click en “Add Keypair” y procederemos a guardarlo. Es muy importante crear certificados al momento de crear las instancias esto nos permitirá tener un control de acceso a las imágenes.



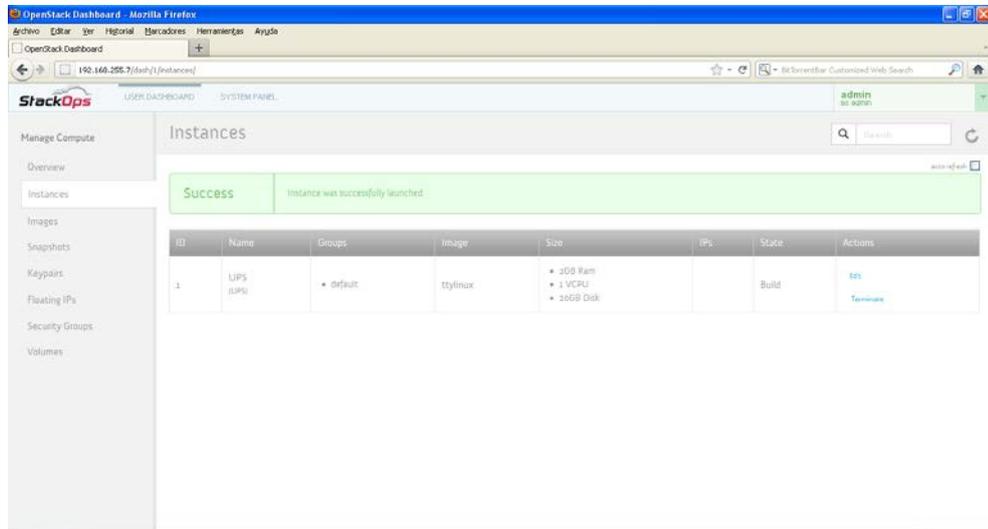
Ingreso interfaz web.
Fuente [Autor].

Pedirá guardarlo, y se lo enviara el archivo en un lugar donde se lo pueda recordar ya que lo utilizara más adelante



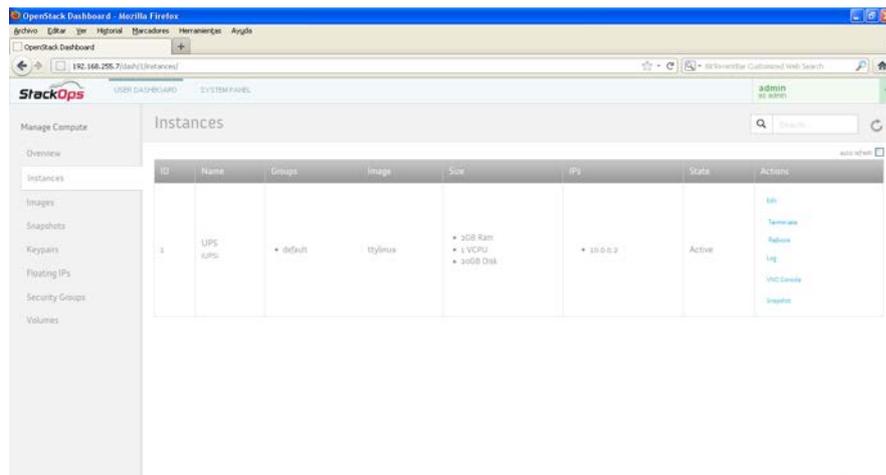
Credenciales.
Fuente [Autor].

En el menú de imágenes del Panel de control hacemos click en launch, y elegiremos una descripción para la instancia y un tamaño para ella, en nuestro caso seleccionaremos m1.small y de keypair nuestro certificado “ups”.



Configuración de instancias.
Fuente [Autor].

Hasta este paso se ha creado la instancia, si refrescamos la pagina nos daremos cuenta que la imagen con los recursos configurados están todos establecidos como nos muestra la siguiente figura.



Componentes de la instancia.
Fuente [Autor].

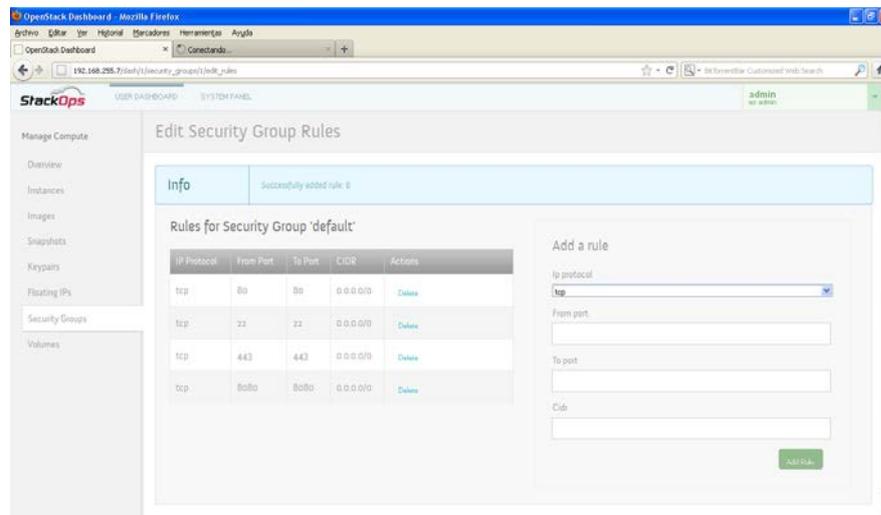
En el grupo de seguridades aplicaremos algunos filtros para la denegación o aprobación del flujo de comunicación o datos. Se habilitaran los siguientes puertos:

TCP/22.- Podremos tener acceso a ssh, scp, y a sftp.

TCP/80.- Utilizada para la navegación a internet. (HTTP).

TCP/443.- Es usada para la transferencia segura de páginas web (HTTPS, SSL)

TCP/8080.- Este puerto se lo dejara abierto para escuchar peticiones.

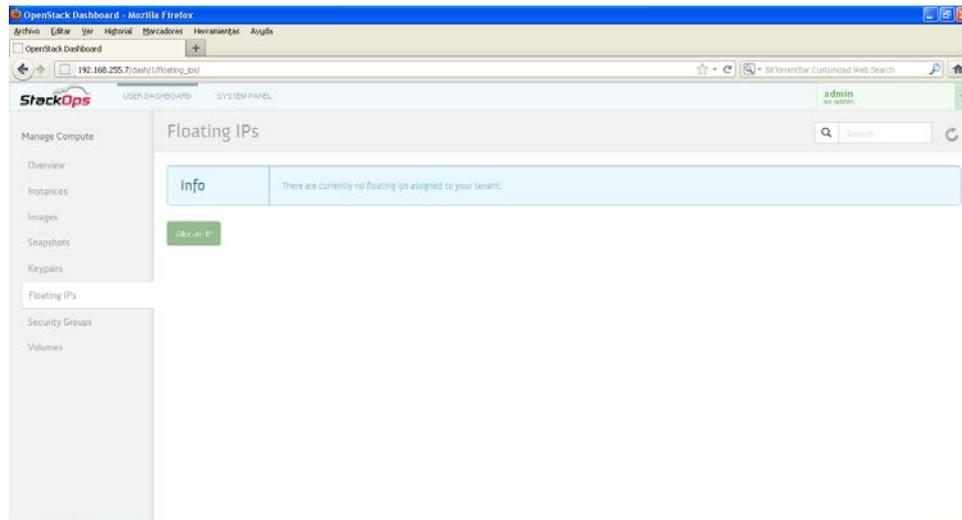


Seguridades.
Fuente [Autor].

4.11 Conexión con la instancia.

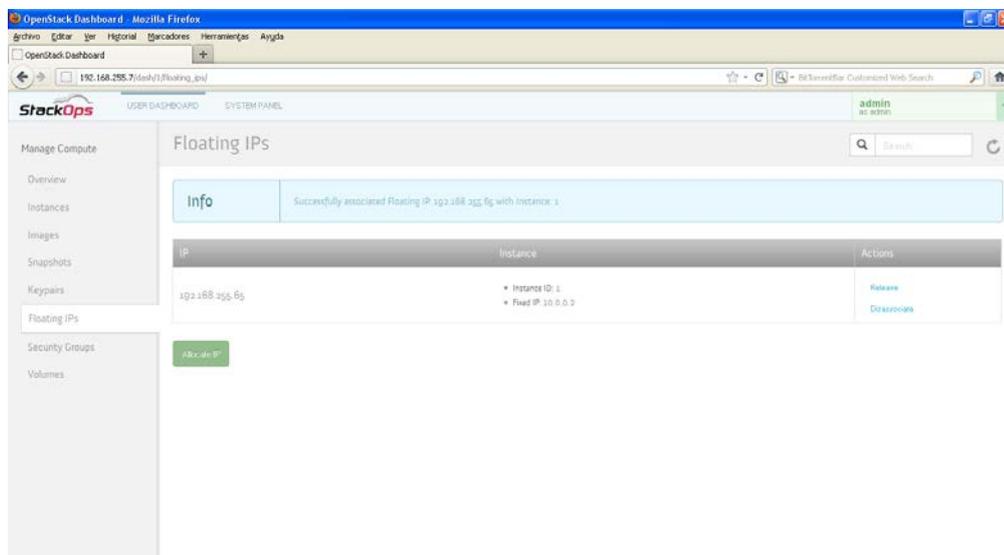
Para empezar a realizar la conexión de la instancia tendremos que colocar una ip de nuestro rango de la red privada ya que la ip de la instancia (10.0.0.2) no podría establecer una conexión, en nuestro caso el rango a cambiar seria 192.168.255.X

Es necesario en el momento de la instalación cuando se deba configurar la red privada colocar un rango de ip virtuales superior a las ip reales, así no se creara conflictos al momento se asociarlas.



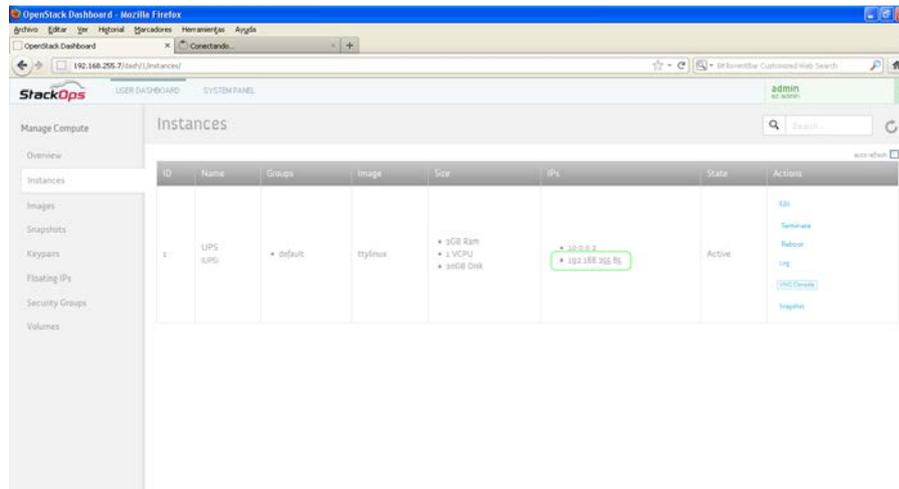
Asignación de ip real.
Fuente [Autor].

En el menú de “*Floating*” damos click en “*Allocate IP*” esto nos devolverá una ip del mismo pool de nuestra red privada que tengamos disponibles al final pulsaremos “*Associate to instance*” para finalmente asociarla a la instancia ya anteriormente creada.



Asociación de la ip con la instancia.
Fuente [Autor].

Nos conectaremos a la ip a la cual queremos asociar seleccionándola y pulsaremos en el link “*Associate IP*”, luego de esto veremos cómo quedo la ip y la instancia a la cual asociamos conectados todo esto en el menú “*Instances*”.



Ip real con ip de la instancia.
Fuente [Autor].

Luego de haber realizado todas estas configuraciones en el Front-End podremos tener acceso a la nube de computación y realizar lo que el cliente desee instalar para levantar sus servicios y aplicaciones.



Acceso al front-end.
Fuente [Autor].

Al igual que con los primeros pasos de la adopción de cualquier nueva tecnología, los clientes y las empresas que se trasladan a la nube comienzan a vivir una etapa de acoplamiento en su mayoría las empresas pymes. Pero a medida que la computación

en la nube vaya dejando en mano la administración de todas sus aplicaciones a las empresas esto va cambiando dejando a un lado otros requisitos de desempeño.

Las ofertas actuales de esta tecnología suelen ofrecer varias opciones de almacenamiento pero para este prototipo que se ha realizado nos permite verificar con pocos recursos la disponibilidad de espacio en la nube de computación y lo que podemos realizar y administrar con maquinas virtuales y un acceso rápido a nuestros datos.

Es así que se demostró con un pequeño prototipo la parte funcional de una infraestructura de cloud computing, que brinda servicios de alojamiento de datos mediante internet y que actualmente está revolucionando el mundo de TI.

Cronograma.

	Actividad 2012-2013						
		AGO	SEP	OCT	NOV	DIC	ENE
1	Procesamiento de información	■	■	■			
2	Análisis de requerimientos	■	■	■			
3	Instalación de los equipos		■	■			
4	Instalación del software		■	■			
5	Configuración de los componentes			■	■		
6	Pruebas del prototipo			■	■		
7	Creación de la documentación	■	■	■	■	■	■
8	Correcciones del prototipo				■	■	
9	Presentación del proyecto al tutor				■		
10	Depuración del prototipo				■	■	
11	Pruebas finales					■	■
12	Depurar documento						■
13	Entrega de documento corregido						■
14	Fecha de sustentación						■

Presupuesto.

Equipo	Características	Precio
Nodo Controlador	Core i3 3,3 Ghz, 4Gb, 1T HD, (64bit)	600
Nodo almacenamiento	Intel Pentium 2,1 Ghz, 2Gb, 500 HD, (64bit)	450
Switch 8 puertos	CNet	25
Lector Cd Externo	Lector Cd	40
Monitor 19"	BenQ	100
Switch Kvm De 4 Puertos Usb D-link	Dkvm-4u	75
	Total	1290
	IVA	154,8
	TOTAL	1444,8

Conclusiones.

El objetivo de esta tesis ha sido investigar y demostrar que si es factible crear un prototipo de cloud computing y ser capaz de poderla administrar como si fuera una maquina real con los mismos recursos y lo que implica su utilidad en algunos campos.

Esta tesis se centro en examinar la tecnología de cloud computing que representa la evolución de la virtualización de los datos y toda la información, esto ha permitido la interoperabilidad entre varias plataformas y la oferta de servicio bajo demanda. En la tesis se puede comprobar que se ejecuta una instancia como nuestro espacio virtual donde se tendrá disponibilidad y su respectivo acceso.

Con una ip pública se podría hacer la demostración de una nube publica y acceder y tener el control desde cualquier lugar que se requieran por parte de las compañías. Al final de este proyecto se puede concluir que la implementación de cloud computing es rentable comparado a la implementación de un data center dados en las pruebas del prototipo creado.

- Cuando se realizan las configuraciones en el front-end que es el nodo controlador no se debe tener la misma configuración del “dual node” ya que se crearía una inconsistencia, siempre se debe configurar al nodo compute como el storage de la forma que permita dar el espacio de la sesión.
- Para las conexiones de ssh de las instancias se deben habilitar los puertos 22, 8080 y el protocolo icmp. Estos protocolos se los aplica en el momento de configurar las seguridades.
- Si se quiere acceder a la ip de la instancia virtual debe estar asociada a la misma red donde se vaya a configurar, de no ser así en el momento de la asignación de la red LAN en la configuración del nodo se debe poner una subred de la misma red LAN para que en el momento que se genere la ip de acceso no se tenga validación denegada.
- Se debe configurar los volúmenes con dos particiones y no en su totalidad del disco ya que en el momento de asignar el espacio para la virtualización se genera un error en la configuración de la aplicación de la nube.
- Para solucionar posibles problemas en la ejecución de las instancias es recomendable revisar los registros logs ya que estos poseen toda la información necesaria para verificar la causa del problema.

La implementación de esta tesis se la pudo realizar con el esfuerzo y los mecanismos de investigación con lo cual se ha podido implementar en un pequeño prototipo lo que nos brinda las nubes de computación y sus servicios y queda una gran posibilidad para futuras investigaciones sobre el rendimiento de esta nueva tecnología la cual no es tarea fácil sobre todo por los diferentes tipos de aplicaciones que se podría aplicar. Un trabajo de seguimiento de la disponibilidad y rendimiento daría lugar a un futuro estudio de las diferentes aplicaciones donde el acuerdo de nivel de servicio sea parte del contrato donde las empresas se sientan más seguras del compromiso de los proveedores todo esto como parte de una investigación a futuro.

Recomendaciones.

Cada empresa es muy diferente a las demás, así también los proveedores de cloud computing con respecto a los tipos de servicios que ofrecen, ya sea soporte, soluciones, y costos. Por tal motivo es importante consideren ciertas recomendaciones al momento de evaluar la decisión de contratar servicios de un determinado proveedor de cloud computing.

Un punto muy importante y recomendable es hacer un análisis del costo a largo plazo y ver si realmente le resulta favorable acceder a la utilización de esta tecnología con las tarifas de cada proveedor de cloud computing.

Otro punto muy importante es ver la disponibilidad del servicio y el tiempo de respuesta en caso de falla, que ofrezca el proveedor al momento de firmar contrato. La seguridad y privacidad de los datos ofrecidas por el proveedor se deben adaptar a las necesidades de cada empresa.

Es importante tomar en cuenta estas recomendaciones para que las compañías se sientan con la seguridad de tener sus datos alojados en manos de personas serias y responsables y poder acceder a su información, en el momento y en el lugar que se desee.

Bibliografía

[1] BRIAN J,S, Chees y CURTIS Franklin Jr, “*Cloud Computing -- Technologies and Strategies of the Ubiquitous Data Center*”, página 17.

Recuperado en: Diciembre, 2010.

[2] Wikipedia, “Hipervisor”.

Tomado de: <http://es.wikipedia.org/wiki/Hipervisor>

Recuperado en: Febrero, 2011.

[3] MOTAHARI-NEZHAD, H. R. STEPHENSOND SINGHAL, S. “*Outsourcing Business to Cloud Computing Services: Opportunities and Challenges*”.

Tomado de: <http://www.hpl.hp.com/techreports/2009/HPL-2009-23.pdf>

Recuperado en: Febrero, 2011.

[4] Microsoft azure, “Windows azure”.

Tomado de: <http://www.microsoft.com/windowsazure/>

Recuperado en: Marzo, 2011.

[5] Google app, “Google app for bussines”

Tomado de: <http://www.google.com/enterprise/apps/business/>

Recuperado en: Marzo, 2011.

[6] Ebay, “cloud computing and eBay”

Tomado de: <http://www.cloudcomputingzone.com/2010/05/06/cloud-computing-and-ebay/>

Recuperado en: Marzo, 2011.

[7] Amazon elastic cloud computing, “Amazon Elastic Compute Cloud (Amazon 2)”

Tomado de: <http://aws.amazon.com/es/ec2/>

Recuperado en: Junio, 2011.

[8] Salesforce, “What is cloud computing”

Tomado de: <http://www.salesforce.com/cloudcomputing/>

Recuperado en: Septiembre, 2011.

[9] Openstack, “Software de código abierto para la creación de nubes privadas y públicas.”. Tomado de: <http://www.openstack.org/>

Recuperado en: Enero, 2012.

[10] Stackops, “Stackops”

Tomado de: <http://www.stackops.com/>

Recuperado en: Marzo, 2012.

[11] INFOBRAND Digital – Revista, “*Llega Virtualización, un nuevo servicio de FiberCorp*”. Tomado de: de marketing branding y comunicación. Página 7.

Recuperado en: Julio, 2012.

[12] A. Eric And LOZANO Marks Bob, “*Executive's Guide to Cloud Computing.*” página. 45. Recuperado en: Julio, 2012.

[13] NIST (National Institute of Standards and Technology)

Tomado de: <http://www.nist.gov/index.html>

Recuperado en: Agosto, 2012.

[14] NIST (National Institute of Standards and Technology)

Tomado de: <http://www.nist.gov/itl/csd/>

Recuperado en: Agosto, 2012.

[15] STATEN James, “*Is cloud computing ready for the enterprise?*”

Tomado de: *Forrester Research*,

Recuperado en: Septiembre, 2012.

[16] PLUMMER Dc, BITTMAN Tj, AUSTIN T., CLEARLEY D., and SMITH Dm, “*Cloud Computing: Defining and describing and emerging phenomenon*”.

Tomado de: Gartner, Inc.

Recuperado en: Agosto, 2012

[17] Buyya R., Yeo C.S., and Venugopal S. “*Market-oriented cloud computing*”: *Vision, hype, and reality for delivering it services as computing utilities*. In *High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on*, pages 5–13. IEEE, 2008.

[18] REESE George. “*Cloud Application Architecture. Building application and infrastructure in the Cloud*”. O'Reilly, 2009.

[19] VELTE Anthony T. and VELTE Toby J, Ph. D. and ELSEN PETER Robert “*Cloud Computing: A Practical Approach*”, página 69. 2010.

[20] CREASY R.J.. “*The origin of the VM/370 time-sharing system*”.

Tomado de: IBM Journal of Research and Development, February 18, 1981

[21] DELL Rob H, “*Virtualizing Approaches for OpenStack Nova: looking at the many ways to skin the Cactus (KVM v XenServer v ESX)*”.

Tomado

de:

<http://en.community.dell.com/techcenter/b/techcenter/archive/2011/06/01/virtualizing-approaches-for-openstack-nova-looking-at-the-many-ways-to-skin-the-cactus-kvm-v-xenserver-v-esx.aspx>

Recuperado en: Noviembre, 2012

[22] RUEST Danielle and RUEST Nelson, “*Virtualization: A Beginner's Guide*”, página 25. 2009

- [23] Amazon EC2. “Amazon Elastic Compute Cloud (Amazon EC2)”.
Tomado de: <http://aws.amazon.com/ec2/>
Recuperado en: Noviembre, 2012
- [24] PEPPLE Ken Deploying Openstack . “*Creating Open Source Clouds*”, página 1.
2011.
- [24] RESEARCH Kelton. Survey: “*Global Study: Cloud computing provides real business benefits, but Fear of Security and Control Slowing Adoption*”.
Tomado de: <http://tv.sys-con.com/node/852659>
Recuperado en: Diciembre, 2012
- [25] SANTOS Nuno, GUMMADI Krishna. P. and RODRIGUEZ Rodrigo.
“*Towards trusted cloud computing*”, página 3, 2009.
- [26] KAMARA S. and Lauter K. Cryptographic cloud storage. “*Financial Cryptography and Data Security*”, páginas 136–138, February 12-16, 2007.
- [27] Google. The Chromium projects: Chromium OS.
Tomado de: <http://www.chromium.org/chromium-os>,
Recuperado en: Diciembre, 2012
- [28] EyeOS. “*Cloud computing operating system*”.
Tomado de: <http://eyeos.org/>.
Recuperado en: Diciembre, 2012
- [29] WARTALA Ramón. “*Linux Magazine*” edición N° 73. “*El framework de computación en la nube Openstack*”, paginas 30–36.
- [30] Rackspace Cloud. “The rackspace cloud powered by Openstack”
Tomado de: <http://www.rackspacecloud.com/>
Recuperado en: Noviembre, 2012

[31] QEMU.

Tomado de: <http://wiki.qemu.org/Index.html>.

Recuperado en: Diciembre, 2012

[32] Hybridfox

Tomado de: <http://code.google.com/p/hybridfox/>

Recuperado en: Enero, 2013

[32] euca2ools

Tomado de: <http://information-technology.web.cern.ch/node/4414>

Recuperado en: Enero, 2013

[33] Stackops

Tomado de: <http://www.stackops.org/>

Recuperado en: Diciembre, 2012

[34] Putty

Tomado de: <http://es.wikipedia.org/wiki/PuTTY>

Recuperado en: Diciembre, 2012

[35] DE MIGUEL CUEVAS Eduardo. “*Aspectos jurídicos del cloud computing*”.

CompusSoft Comunidad de software y tecnología. Julio 21, 2011

Tomado de: <http://www.comusoft.com/aspectos-juridicos-del-cloud-computing>

Recuperado en: Agosto, 2012

[36] Wikipedia.

Tomado de: <http://es.wikipedia.org/wiki/PuTTY>

Recuperado en: Diciembre, 2012

[37] outsourceand.

Tomado de: <http://outsourceando.blogspot.com/2011/05/jornada-clud-computing-en-esic-madrid.html>

Recuperado en: Enero, 2013

[38] VIEDA Manuel, “*Que es Cloud Computing?*”.

Tomado de: <http://manuelvieda.com/2011/07/cloud-computing-que-es/>

Recuperado en: Octubre, 2012

[39] BRYSON Aaron, “*Penetration Testing in the Cloud*”.

Tomado de: <http://blogs.cisco.com/security/penetration-testing-in-the-cloud/>

Recuperado en: Octubre, 2012

[40] RUEST Danielle and RUEST Nelson, “*Virtualization: A Beginner’s Guide*”,
página 228. 2009.

[41] DELL Rob H, “*Virtualizing Approaches for Openstack Nova: looking at the
many ways to skin the Cactus (KVM v XenServer v ESX)*”.

Tomado de: [http://en.community.dell.com/techcenter/b/techcenter/archive/2011/06/01/virtualizin](http://en.community.dell.com/techcenter/b/techcenter/archive/2011/06/01/virtualizing-approaches-for-openstack-nova-looking-at-the-many-ways-to-skin-the-cactus-kvm-v-xenserver-v-esx.aspx)

[g-approaches-for-openstack-nova-looking-at-the-many-ways-to-skin-the-cactus-
kvm-v-xenserver-v-esx.aspx](http://en.community.dell.com/techcenter/b/techcenter/archive/2011/06/01/virtualizing-approaches-for-openstack-nova-looking-at-the-many-ways-to-skin-the-cactus-kvm-v-xenserver-v-esx.aspx)

Recuperado en: Noviembre, 2012

[42] Rackspace Cloud. “The rackspace cloud powered by Openstack”

Tomado de: <http://www.rackspacecloud.com/>

Recuperado en: Noviembre, 2012

[43] GoGrid. Cloud hosting. “*Demanda de infraestructura a su alcance*”

Tomado de: <http://www.gogrid.com/cloud-hosting/>

Recuperado en: Noviembre, 2012

[44] Salesforce.

Tomado de: <http://www.salesforce.com/>

Recuperado en: Diciembre, 2012

[45] Google App Engine.

Tomado de: <http://code.google.com/appengine/>

Recuperado en: Diciembre, 2012

[46] Microsoft Windows Azure.

Tomado de: <http://www.microsoft.com/windowsazure/windowsazure/>

Recuperado en: Diciembre, 2012

[47] Oracle Cloud Computing.

Tomado de: <http://www.oracle.com/us/technologies/cloud/index.htm>.

Recuperado en: Diciembre, 2012

[48] Eucalyptus Cloud Platform.

Tomado de: <http://open.eucalyptus.com/>

Recuperado en: Diciembre, 2012

[49] Openstack

Tomado de: <http://openstack.org/>

Recuperado en: Diciembre, 2012

[50] KVM.

Tomado de: http://www.linux-kvm.org/page/Main_Page

Recuperado en: Diciembre, 2012

[51] XEN.

Tomado de: <http://xen.org/products/xenhyp.html>

Recuperado en: Diciembre, 2012.

[52] LXC LINUX CONTAINERS.

<http://lxc.sourceforge.net/>

Recuperado en: Diciembre, 2012

[53] VMWARE ESX/ESXI

Tomado de: <http://www.vmware.com/products/vsphere-hypervisor/support.html>

Recuperado en: Diciembre, 2012

[54] Stackops Distro Community Edition

Tomado de: <http://www.stackops.com/products/stackops-distro-community-edition/>

Recuperado en: Diciembre, 2012