RID | UNAJ
Repositorio Institucional Digital UNAJ

Universidad Nacional
ARTURO JAURETCHE

Galarza, Brian Ivan

Implementación y despliegue de un Cloud Privado por medio de OpenStack

2016

Instituto de Ingeniería y Agronomía

Ingeniería en Informática

Esta obra está bajo una Licencia Creative Commons Internacional.
Atribución - No Comercial - Compartir Igual 4.0
https://creativecommons.org/licenses/by-nc/4.0/

Documento descargado de RID - UNAJ Repositorio Institucional Digital de la Universidad Nacional Arturo Jauretche

Cita recomendada:
Galarza, B. I. (2016). Implementación y despliegue de un Cloud Privado por medio de OpenStack [Informe final]. Universidad Nacional Arturo Jauretche.
Disponible en RID - UNAJ Repositorio Institucional Digital UNAJ https://biblioteca.unaj.edu.ar/rid-unaj-repositorio-institucional-digital-unaj

Práctica Profesional Supervisada

Implementación y despliegue de un Cloud Privado por medio de OpenStack

Alumno: Galarza Brian Ivan

Director UNAJ: Morales Daniel Martin

Director Institución: Encinas Diego Omar

Instituto de Ingeniería y Agronomía

Ingeniería en Informática

Noviembre de 2016

Universidad Nacional ARTURO JAURETCHE

Índice

Capítulo 1 3
Introducción 3
Antecedentes 3
Objetivos 3
Cloud 4
Capítulo 2 7
Herramientas 7
Fuel 7
PXE 7
OpenStack 9
Capítulo 3 13
Desarrollo 13
Openstack Dashboard 17
Sabores 17
Imagenes 19
Instancias 20
Resultados 23
Capítulo 4 25
Conclusiones 25
Agradecimientos 27
Referencias 27
Anexo 28
Publicaciones 28
Convocatorias 28
Exposiciones 28

Universidad Nacional ARTURO JAURETCHE

Capítulo 1

Introducción

En los últimos años sucedieron cambios en el ámbito de la informática, entre las que se destaca el ofrecer productos, ya sean hardware o software como un servicio. Por ello mismo cabe destacar el crecimiento que ha tenido Cloud en estos ultimos años. A raiz de esto se ha trabajado en el despliegue de un Cloud privado, en el que se ha implementado y desplegado una Infraestructura como un servicio (Infraestructure as a Service, IaaS) utilizando las herramientas Fuel y Openstack.

La estructura de este trabajo consta de 4 Capítulos:

Capítulo 1: Introducción al trabajo y explicación del concepto de Cloud Computing.

Capítulo 2: Se mencionan y describen las herramientas utilizadas para el desarrollo de las prácticas.

Capítulo 3: Desarrollo de las prácticas y resultados obtenidos.

Capítulo 4: Conclusiones acerca de lo trabajado, líneas a seguir a futuro.

Antecedentes

Esta Práctica Profesional Supervisada es resultado de una sublínea de investigación en la UNAJ y cuenta con varias publicaciones en congresos durante los últimos tres años como también de proyectos del programa Universidad, Diseño y Desarrollo Productivo.
En el anexo se encuentran las publicaciones, convocatorias y exposiciones.

Objetivos

Mediante esta práctica se ha buscado cumplir los siguientes objetivos:

- Implementar los desarrollos realizados durante los trabajos de investigación.
- Ofrecer soluciones eficientes por medio de las instancias lanzadas.
- Configuración de instancias personalizadas que puedan ofrecer distintos servicios predeterminados.

- Poder otorgar servicios mediante la infraestructura implementada.
- Posibilidad de escalar el sistema mediante el uso de PXE
- Ofrecer HPC (High Perfomance Computing) debido al uso eficiente de los recursos.
- Posibilidad de seguir trabajando en la infraestructura implementada tanto en el ámbito de producción como de investigación.

Cloud

Cloud Computing[1] o computación en la nube es un paradigma en Ciencias Informáticas el cual consiste en el ofrecimiento de servicios a traves de un espacio denominado nube, como puede verse en la Figura 1 estos servicios se pueden clasificar en:

IaaS: Ofrece una infraestructura en la cual se pueden lanzar instancias que pueden utilizarse en un campo amplio de funcionalidades (por ejemplo instancias con algun sistema operativo en particular, con servers, instancias orientadas a testing, a seguridad, etc.).

PaaS: Ofrece una plataforma como servicio, ideal para empresas en las que por cuestiones de tiempo o de licencias es conveniente pedir un entorno ya preparado para el desarrollo, testing, control, etc.

SaaS: Ofrece software como servicio, se suelen pagar por un servicio premium mientras que se pueden obtener un servicio básico de manera gratuita, por ejemplo dropbox, gmail, etc.

HaaS: Ofrece Hardware el cual es reconocido a través de internet logrando el uso de los mismos sin la necesidad de tenerlos fisicamente en el lugar de uso.

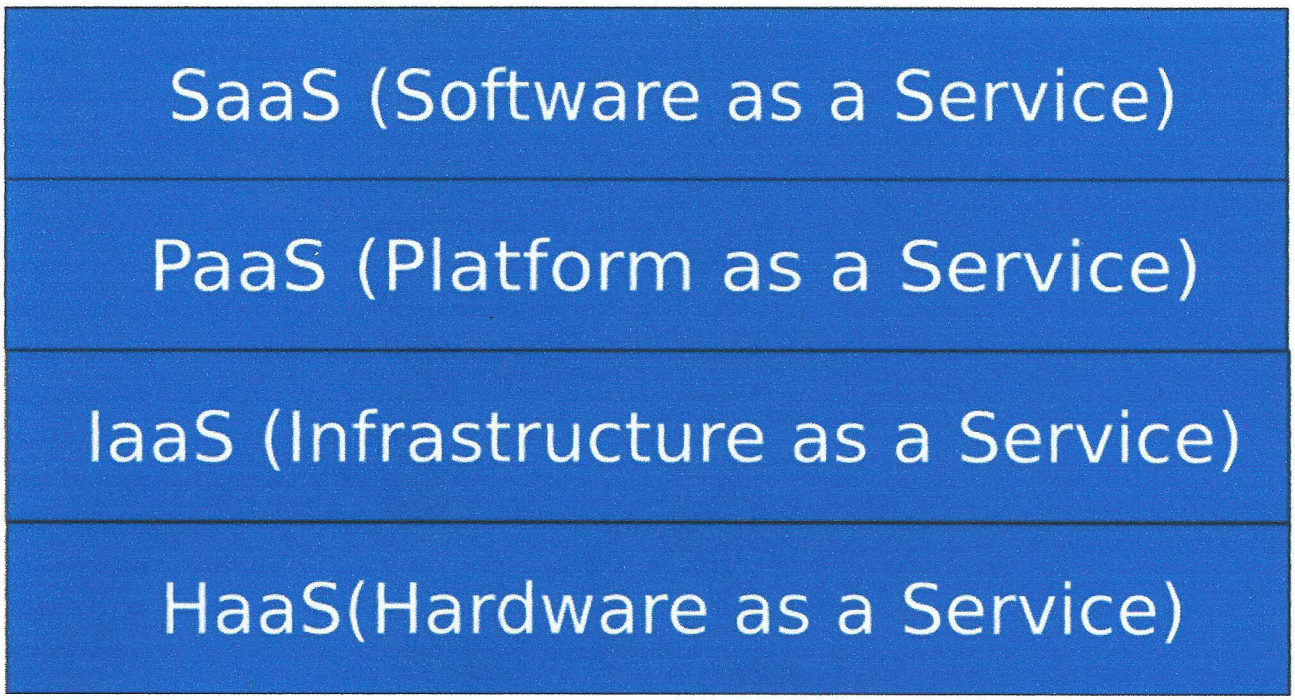


Figura 1. Vista en capas de Cloud.

Asimismo, se pueden clasificar según el ámbito en el que se desempeñaran, es decir quienes serán los que van a utilizar los servicios disponibles (Figura 2). Por lo cual la clasificación dada es la siguiente:

Pública: Cualquier usuario tiene acceso a los servicios de cloud, en cuestiones de seguridad es la peor debido a su gran alcance y por lo que en caso de utilizar una cloud pública, es conveniente tomar precaución con los datos que se vayan a utilizar.

Privada: Limita el acceso a usuarios de la misma red, ideal para empresas o institutos en los cuales se desea tener un cloud propio. En esta PPS se ha optado por realizar este tipo de cloud.

Hibrida: Comparte elementos del cloud publico como del cloud privado, por ejemplo que un usuario pueda acceder a través de internet a una instancia lanzada en la empresa en la que se desempeña mediante el uso de un usuario y contraseña, de manera que solamente ciertos usuarios puedan acceder a servicios especificos.

Figura 2. Cloud Privado, Hibrido y Público.

Capítulo 2

Herramientas

Fuel[2]

Figura 3. Fuel logo.

Es una herramienta creada por Mirantis[3] la cual ofrece un sistema encargado de administrar IaaS, por el medio de Fuel el cual a traves de PXE permite la integración de nuevos equipos para que formen parte de una infraestructura.

Fuel se encarga de la parte más compleja en la instalación de OpenStack en los equipos en donde se le indica previamente que servicios se les instalarán.

PXE[4][5]

PXE (Preboot Execution Environment) es en un protocolo desarrollado por Intel y SystemSoft.[6]

Este protocolo funciona de la siguiente manera: un cliente PXE se comunica con un servidor PXE

(si existe) en la red, de manera que al encontrarlo se comuniquen y se pueda iniciar la instalación del sistema operativo correspondiente (Figura 4).

En Fuel, el uso de PXE es vital en aspectos de escalabilidad porque de manera sencilla se puede colocar un equipo configurandolo para que inicie con PXE y de esta manera se comunique con el servidor PXE (Fuel) y se instale el sistema operativo junto a OpenStack. Además, de configurarle la red y los servicios para que sea funcional al momento de finalizar la instalación.

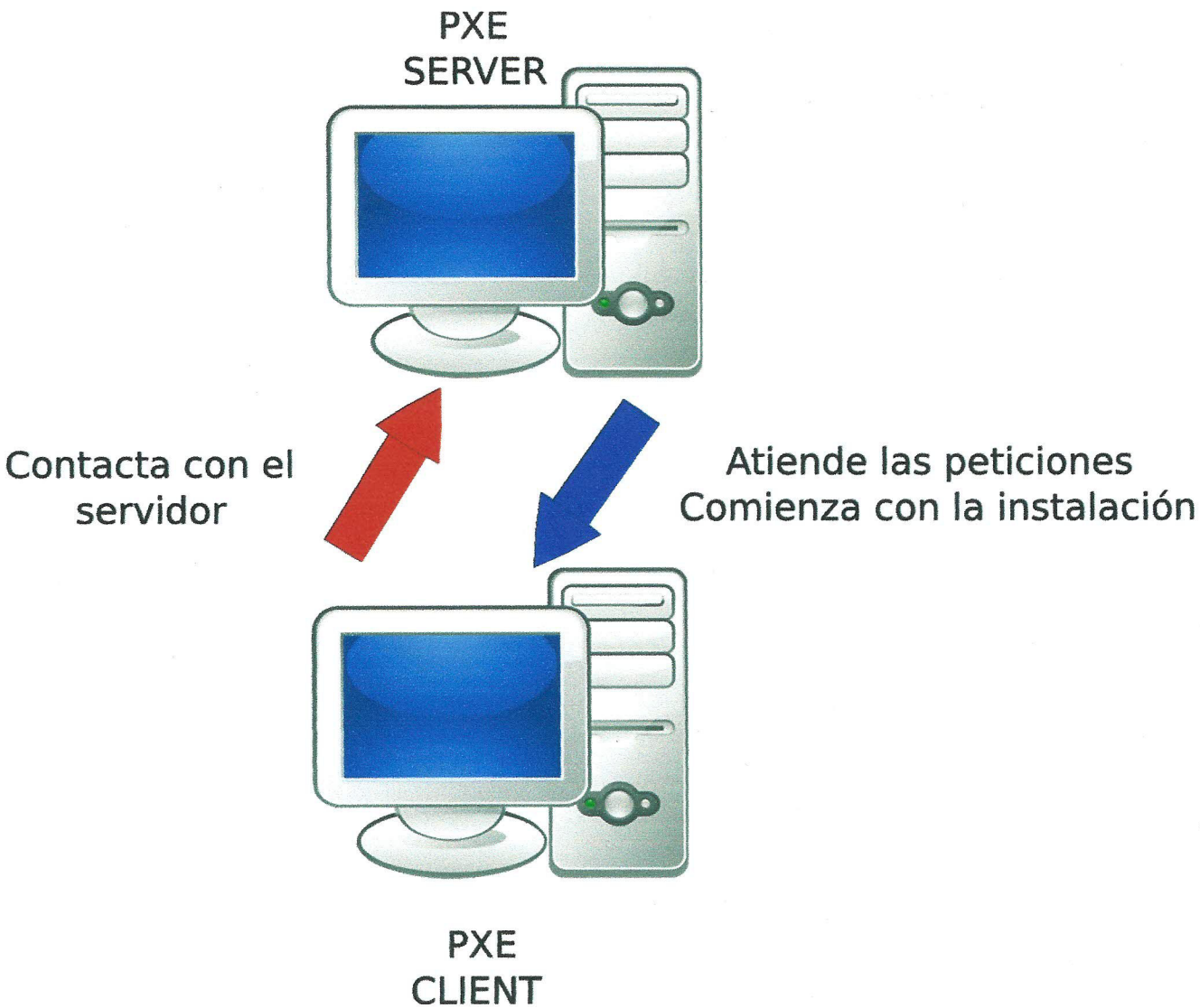


Figura 4. PXE.

OpenStack[7][8]

Figura 5. OpenStack logo.

OpenStack (Figura 5) es una herramienta creada por OpenStack Foundation en cooperación con varias empresas que se encontraron interesadas en el desarrollo de la herramienta.
OpenStack ofrece a los usuarios la posibilidad de crear una IaaS privada de manera gratuita con un alto nivel de escalabilidad y de personalización con el fin de poder ofrecer a las empresas e instituciones un alto grado en el manejo del Cloud y en el mantenimiento de la misma.

Esta herramienta al ser Open Source contó con un gran soporte por parte de la comunidad como de las empresas que la patrocinan (Figura 6). Actualmente es de las más utilizadas a nivel mundial gracias a su fuerte apoyo y constante crecimiento sostenido.

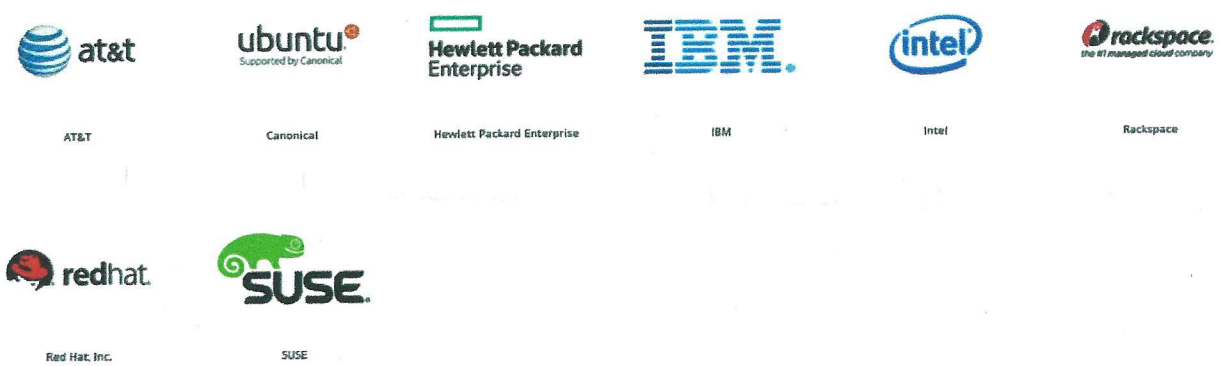


Figura 6. Patrocinadores OpenStack.

Servicios

OpenStack cuenta con varios servicios (Figura 7) coordinados entre si de manera que se pueda desplegar una cloud privada. Estos servicios son:

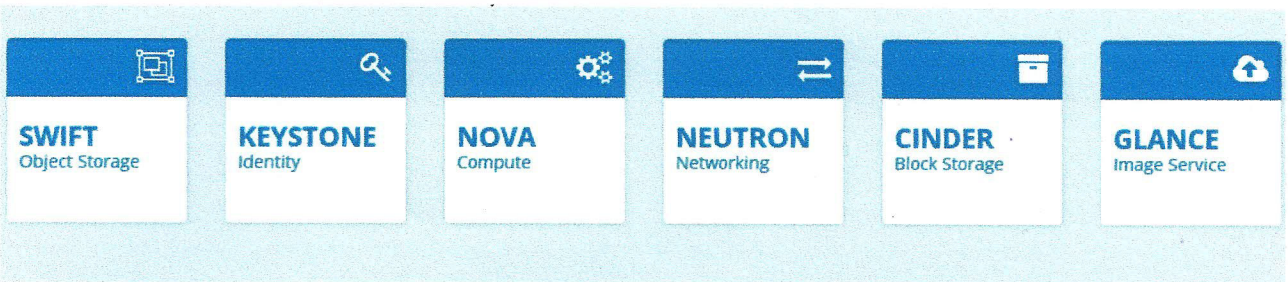


Figura 7. Core Services.

- **Glance:** Servicio que se encarga de tomar imagenes ya sea .iso, .qcow, .qcow2, etc. Para luego ser utilizadas al momento de lanzar una instancia. Por ejemplo, Glance permite que se puedan lanzar instancias con Windows, Debian, Ubuntu, etc. Incluso permite lanzar instantaneas y de esta manera contar con imagenes personalizadas para el uso como puede ser un Windows 7 con las herramientas de desarrollo de visual studio integradas para que cualquier usuario lance la instancia y pueda acceder a las herramientas directamente.

- **Keystone:** Es el servicio más importante, se comunica con los demás servicios y permite la autenticación de los mismos. Es necesario que todos los servicios instalados en la infraestructura se puedan comunicar y autenticar con Keystone para un correcto funcionamiento del Cloud. Los servicios se autentican por medio de tokens, los cuales son utilizados por los servicios una vez que hayan sido autenticados y autorizados por Keystone.

- **Nova:** Servicio que se encarga del ciclo de vida de las instancias. Por lo que se encarga de lanzarlas, de mantenerlas y de finalizarlas. Este servicio es clave en el funcionamiento de OpenStack debido a que el usuario interactua con las instancias lanzadas por lo que el correcto funcionamiento de este servicio permite que el usuario pueda acceder a una instancia y utilizarla. Para la virtualización en el lanzamiento de las intancias ofrece KVM y QEMU entre otras.

- **Neutron:** Servicio de networking el cual es util para la conexion entre los distintos servicios

que integren el cloud (estos servicios se pueden ubicar en distintos equipos de manera que es vital la conexion entre ellos). Neutron gestiona direcciones IP y ofrece las direcciones de manera estática o por DHCP a las instancias generadas por Nova. Por lo que es posible la administración y configuración de la red acorde a lo requerido.

- **Swift:** Provee almacenamiento por objetos. Este servicio está pensado para situaciones en la que se dispongan datos inestructurados que creceran en gran escala. Por ello Swift ofrece escalabilidad y se encuentra optimizado para durabilidad, disponibilidad y concurrencia. Al ser por objetos es posible la replicación de los mismos.

- **Cinder:** Provee almacenamiento por bloques persistentes. La utilización de bloques permite que los usuarios puedan acceder a los mismos utilizando las API ofrecidas por OpenStack sin la necesidad de saber en que dispositivo se encuentran esos recursos.

Estos servicios son básicamente los más importantes, sin embargo OpenStack ofrece servicios adicionales (Figura 8) para permitir un despliegue que pueda adaptarse a las necesidades de la empresa, universidad, instituto, etc.

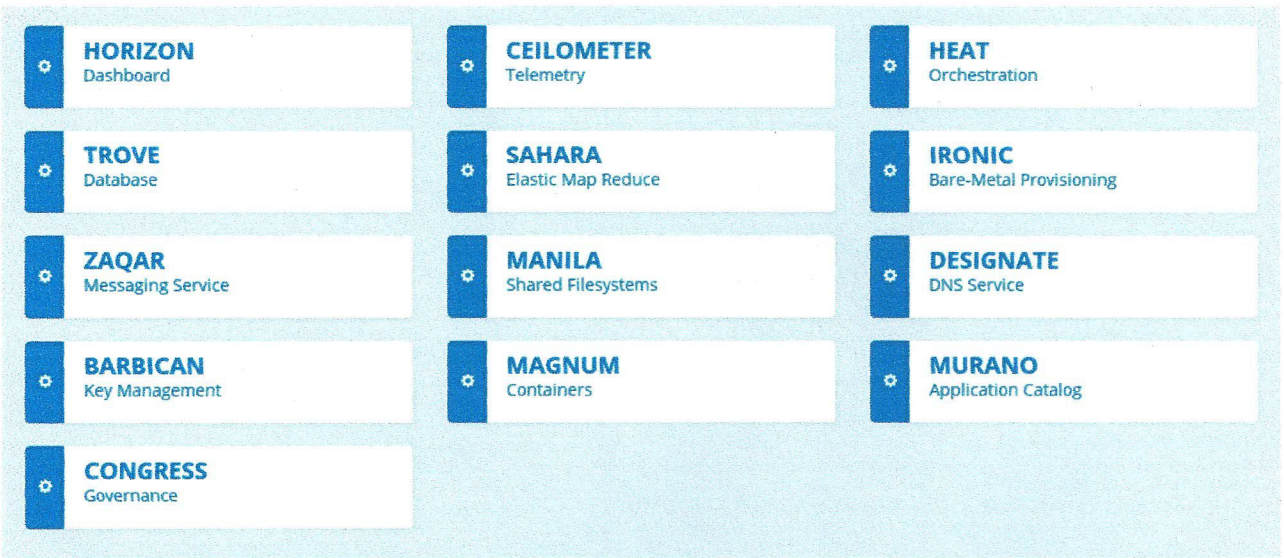


Figura 8. Optional Services.

- **Horizon:** Es el dashboard, el front-end de OpenStack, es decir con lo que interactuará el

usuario, cuenta con una interfaz amigable de manera que cualquier usuario sin la necesidad de tener conocimientos especificos en cloud pueda acceder y utilizar una instancia lanzada (o lanzar una instancia para posteriormente utilizarla)

- **Telemetry:** Ofrece herramientas para monitoreo, benchmarking, analisis del rendimiento, etc. Ideal para el seguimiento del Cloud y para poder evaluar el funcionamiento del mismo a través del tiempo.

- **Trove:** Ofrece correr motores de bases de datos ya sean relacionales (SQL) como no relacionales (MongoDB).

Capítulo 3

Desarrollo

Al momento de realizar la propuesta para la PPS se contaba con un servidor hp ml360 proliant G5 con un procesador Intel Xeon E5450 3.0Ghz QuadCore, 12 GB de RAM (4X2GB + 4X1GB) DDR2 667Mhz. 3 HDD SAS X 146GB 10000RPM + 3 HDD SAS 72GB 10000 RPM.

Sin embargo por cuestiones de arquitectura y del estado del servidor la instalación siempre demoraba demasiado, generando un error de timeout por lo que era imposible la instalación de OpenStack. Luego, se optó por utilizar un equipo con un procesador socket 1155 Intel Quad Core I7 3770 3.4 Ghz, 2 Memorias RAM DDR3 1333 Mhz Kingston 4GB, Disco rígido SATA 3 7200 rpm Western Digital Caviar Blue 1 Tb 64Mb, Motherboard Gigabyte GA-H61M-DS2 y una placa de red PCI Gigabit TP-Link TG-3468.

El primer paso para el despliegue consiste en configurar la red para obtener un correcto funcionamiento y evitar problemas a futuro, para ello se contó con un servidor DHCP el cual está conectado a la red de la universidad, la implementación de este servidor permitió obtener un rango de direcciones IP sin generar conflictos con las direcciones que disponga la universidad.

Una vez configurada la red, en el equipo se instaló inicialmente Debian 8 en una partición de 320GB de disco, una vez instalado se procedió a la instalación de los programas básicos (virtualbox con el extension pack entre otros) necesarios para poder realizar la instalación de Fuel (Figura 9).

La instalación de Fuel contaba con un script que otorga Mirantis en el cual se le configura el rango de direcciones IP a usar, el gateway, como también la cantidad de núcleos, la memoria RAM y el espacio de disco.
Al ejecutar el script, automaticamente se generan 3 maquinas virtuales con la información dada al script, es decir, cada maquina se encuentra configurada con la cantidad de RAM, de disco y de nucleos indicados previamente en el script de Mirantis.
La primera maquina virtual consiste en el nodo master al cual se le instala Fuel, este nodo contará con un dashboard de fuel (Figura 10) en el que se podrá armar y desplegar distintas infraestructuras,

además se le podrá indicar que servicios opcionales se desean instalar y en que nodos se harán. La forma en que reconoce los nodos a los cuales instalará es por medio de PXE, ya que el nodo master funcionará como un servidor PXE mientras los dos nodos slaves con los que se cuentan se comunicarán con el nodo master. Una vez conectados y reconocidos por el nodo master, se les indicará cuales serán sus funcionalidades.

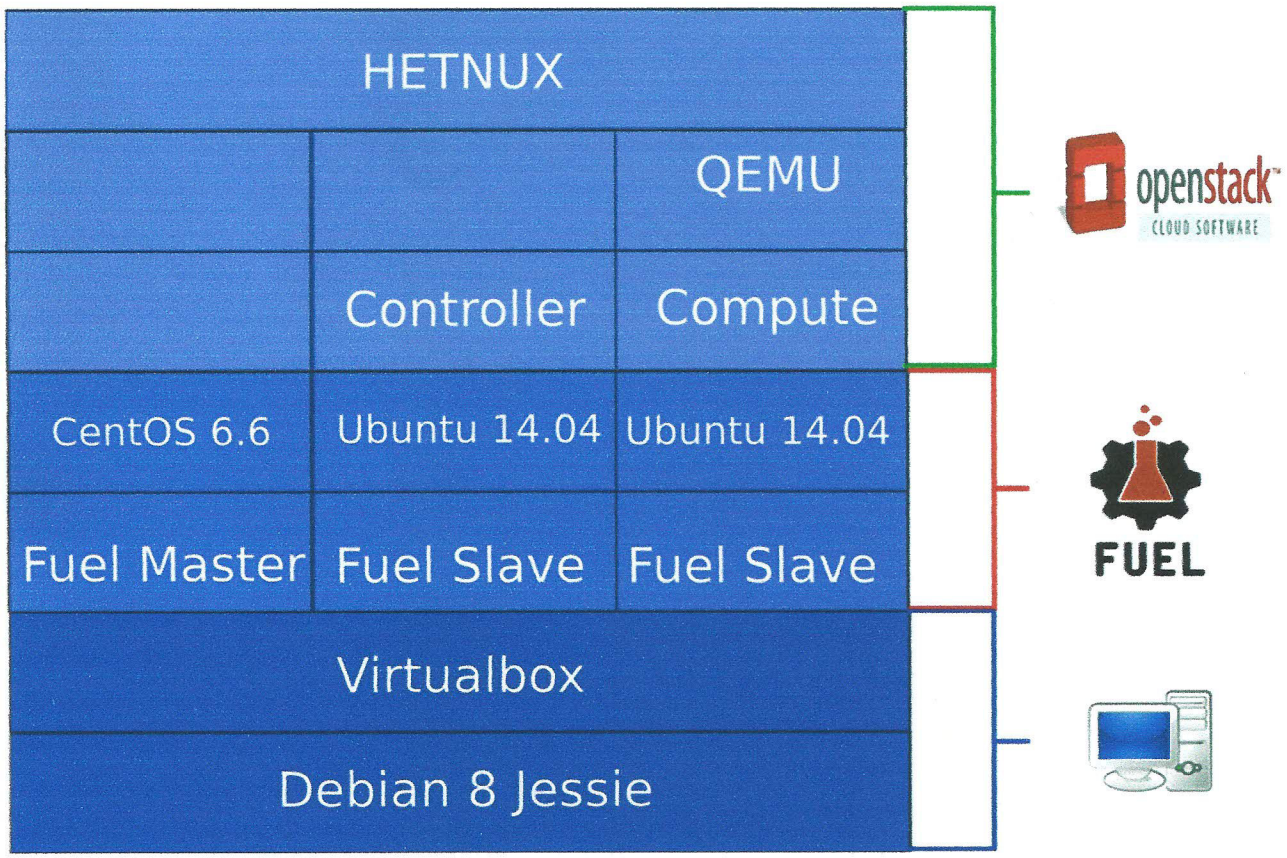


Figura 9. Vista en capas de la IaaS.

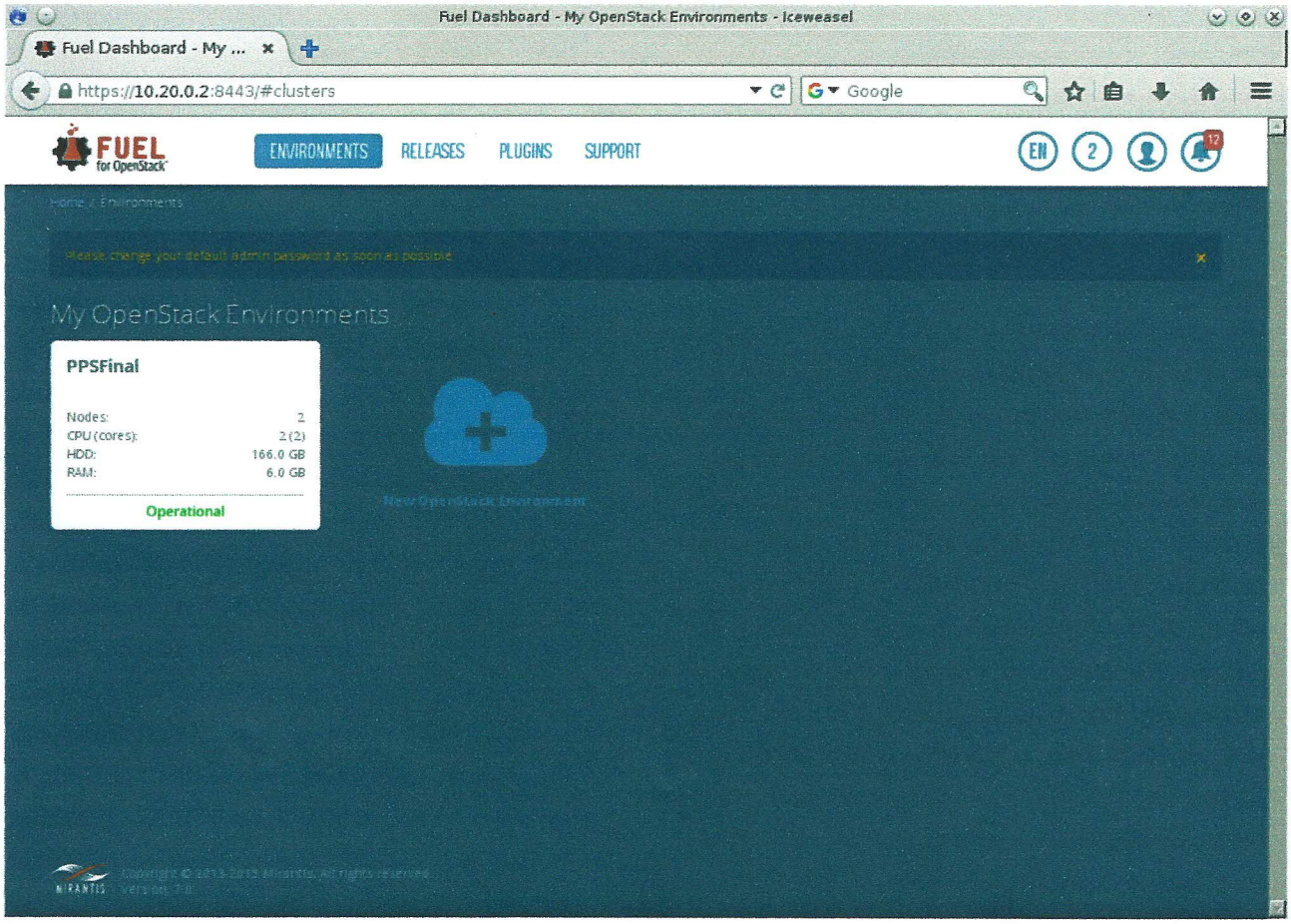


Figura 10. Dashboard de Fuel.

Para el funcionamiento de OpenStack es necesario un despliegue mínimo de dos nodos:

- **Controller Node:** Contará con los servicios Nova, Keystone, Glance, Neutron, Horizon entre otros. Este nodo es el encargado de administrar a los nodos Compute de la infraestructura, puede existir más de un nodo Controller para no centralizar la actividad en un solo equipo. Gracias a estos servicios, el Controller se encargará de la autentificación de los servicios tanto del Controller como del Compute, de ofrecer distintas imagenes para el lanzamiento de instancias, de lanzar las intancias y de la configuración de la red para las mismas.

- **Compute Node:** Contará con nova-compute para poder comunicarse con el nodo Controller y de esta manera poder ser utilizado para el lanzamiento de instancias, cuenta además con QEMU como hypervisor para la virtualización y lanzamiento de la instancia.

Además se ha elegido telemetry para poder contar con algún tipo de control y metrica sobre la infraestructura. Para el lanzamiento de la instancia no se ha optado por ningun tipo de Storage, es decir que no se utilizarán Swift y Cinder.

Una vez configurado y aceptado todos los cambios, Fuel comenzará con la instalación remota, para ello instalará en primer lugar ubuntu en los dos nodos. Una vez finalizada la instalación de ubuntu, el agente de fuel (nailgun) se encarga de la instalación remota de cada uno de los servicios de OpenStack, esta instalación automatizada es una gran ventaja debido a la elevada complejidad que resulta la instalación de OpenStack y el correcto funcionamiento de los servicios (cada uno debe ser instalado en el orden adecuado para un correcto funcionamiento además de requerir una configuración de redes avanzada).

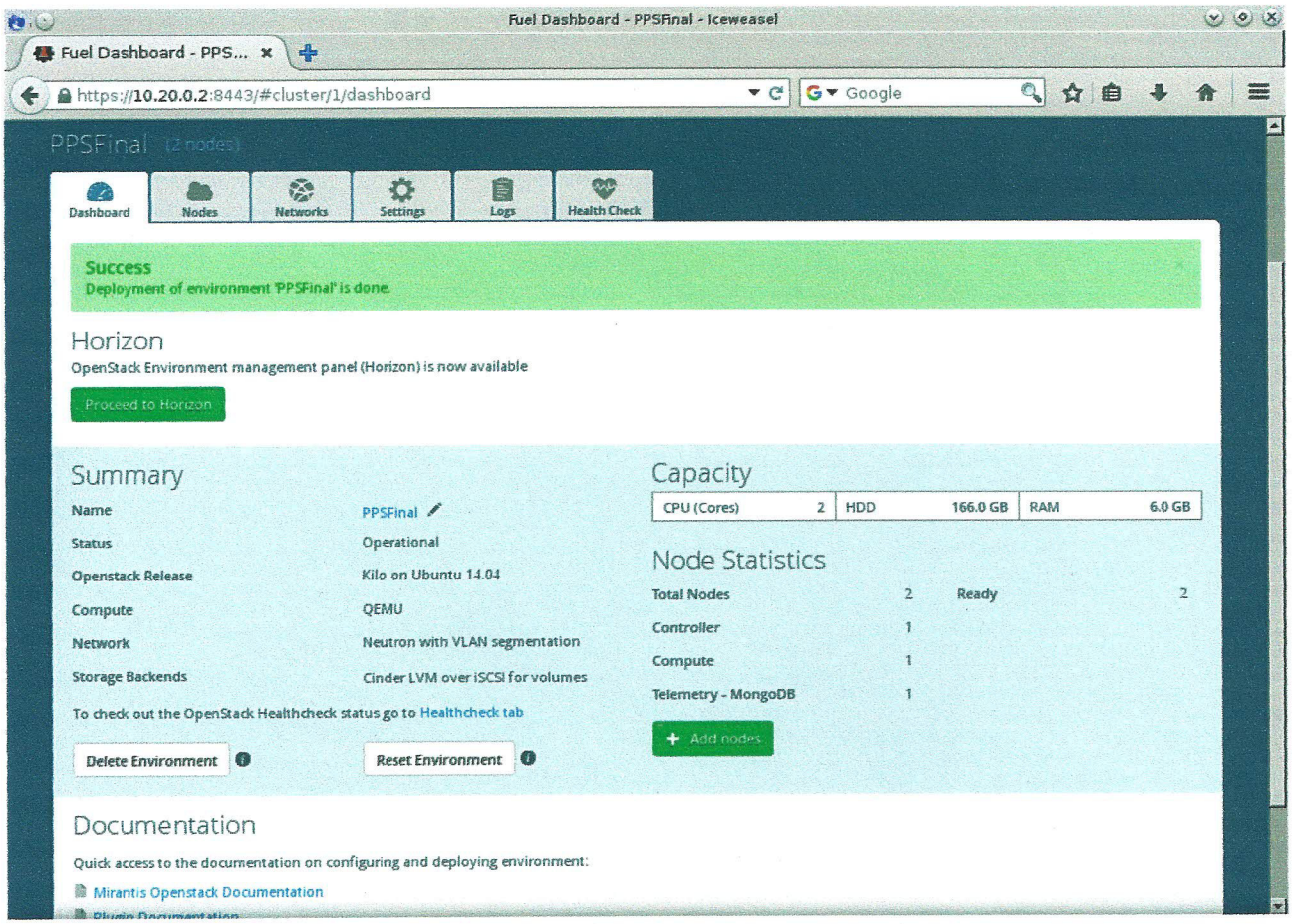


Figura 11. Despliegue del Cloud.

Al finalizar la instalación de OpenStack, es posible acceder al dashboard (Horizon) como se puede ver en la Figura 11.

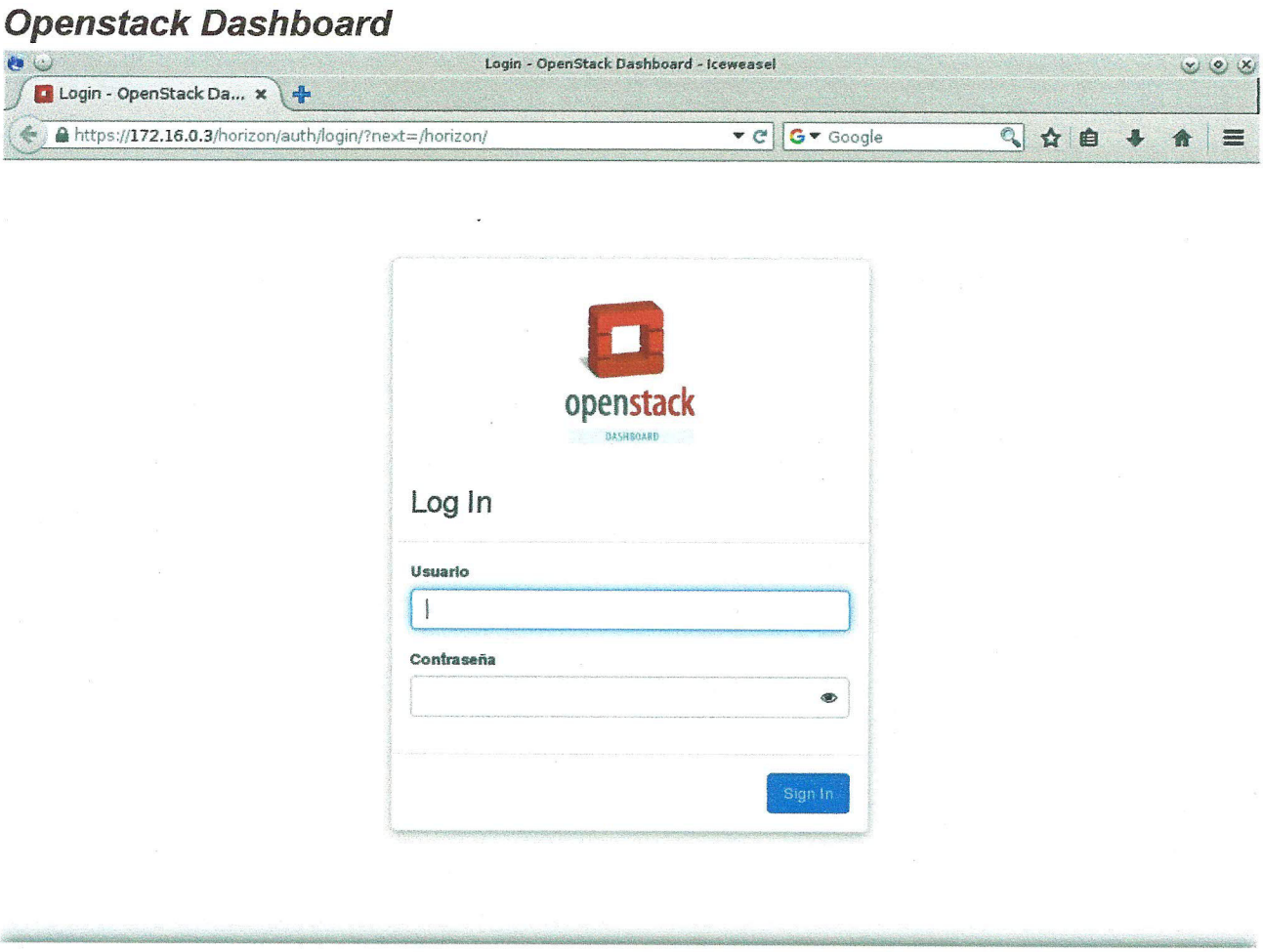


Figura 12. Dashboard de OpenStack.

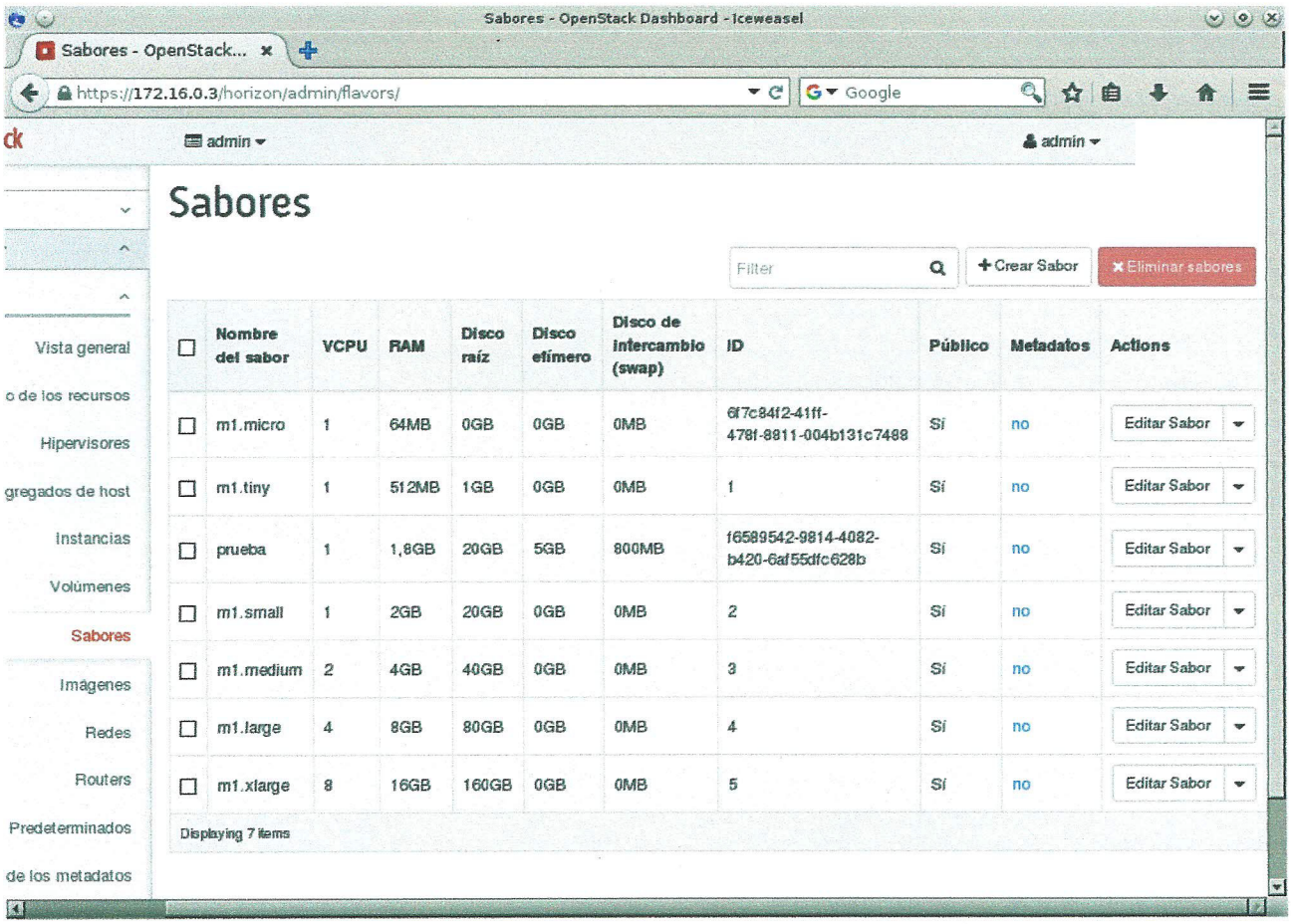
El dashboard (Figura 12) ofrece una interfaz muy amigable e intuitiva para la administración y configuración de la nube. Desde aquí se podrá acceder con el usuario de administrador para poder realizar todas las tareas necesarias para el despliegue de una instancia sin la necesidad de utilizar una terminal y lineas de comandos.

Sabores

Los sabores (Figura 13) son configuraciones preestablecidas en las cuales se indican unas prestaciones particulares para ser usadas según el caso requerido. Estos sabores se suelen clasificar por una silaba, acompañado de la palabra small, tiny, large, entre otros. Estos sabores indican cuanto procesamiento, RAM y espacio en el disco serán reservados y utilizados por la instancia por

lo que es conveniente el uso adecuado de los sabores para el aprovechamiento eficiente de los recursos.

En este caso, para el funcionamiento de HETNUX se decidió crear un sabor específico para el correcto funcionamiento y debido a las limitaciones de Hardware actuales.



Nombre del sabor	VCPU	RAM	Disco raíz	Disco efímero	Disco de intercambio (swap)	ID	Público	Metadatos	Actions
m1.micro	1	64MB	0GB	0GB	0MB	6f7c84f2-41ff-478f-8811-004b131c7488	Sí	no	Editar Sabor
m1.tiny	1	512MB	1GB	0GB	0MB	1	Sí	no	Editar Sabor
prueba	1	1,8GB	20GB	5GB	800MB	f6589542-9814-4082-b420-6af55dfc628b	Sí	no	Editar Sabor
m1.small	1	2GB	20GB	0GB	0MB	2	Sí	no	Editar Sabor
m1.medium	2	4GB	40GB	0GB	0MB	3	Sí	no	Editar Sabor
m1.large	4	8GB	80GB	0GB	0MB	4	Sí	no	Editar Sabor
m1.xlarge	8	16GB	160GB	0GB	0MB	5	Sí	no	Editar Sabor

Figura 13. Sabores

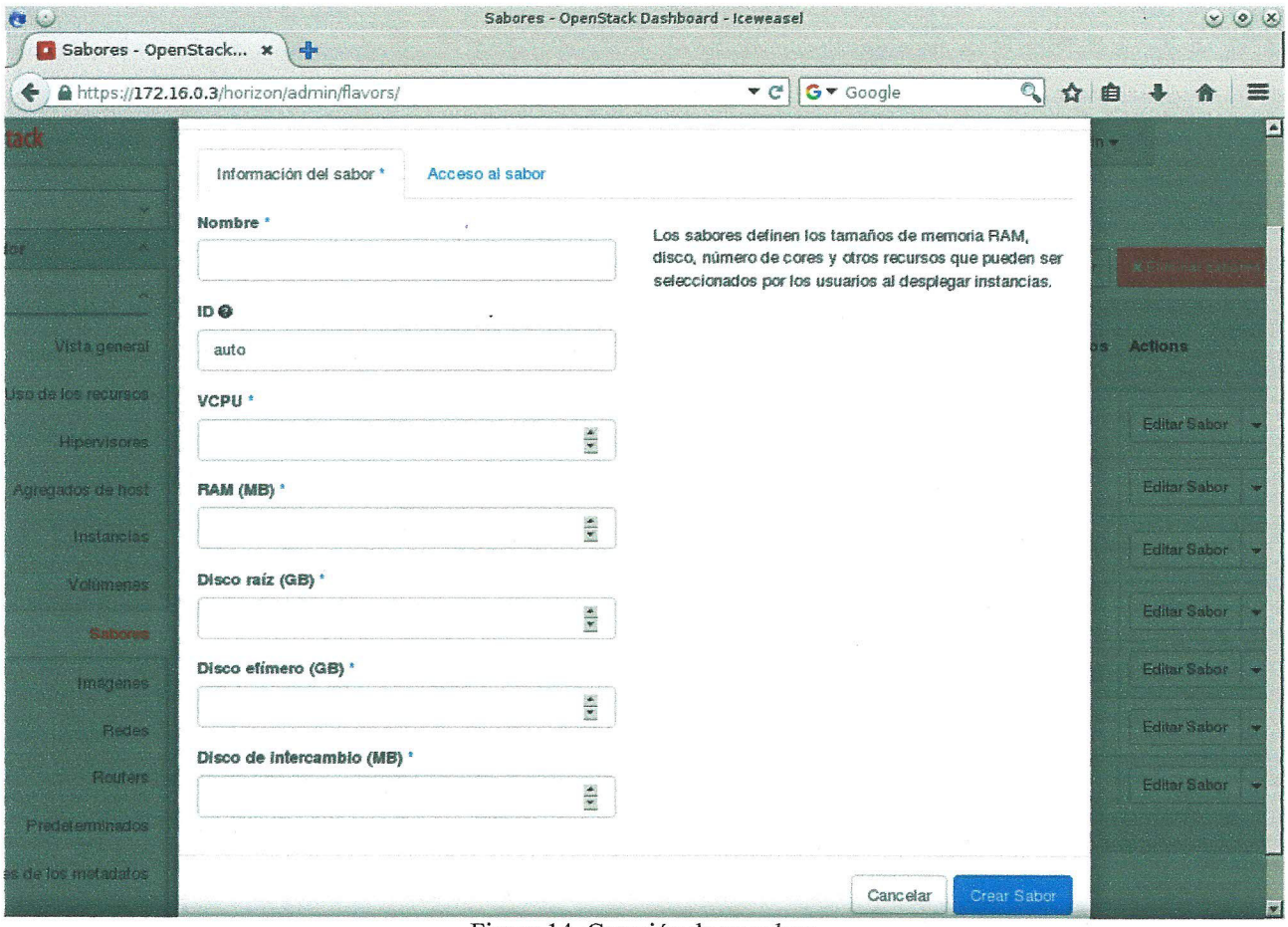


Figura 14. Creación de un sabor.

En la Figura 14 se puede observar la pantalla de creación de un sabor en donde se le indica la cantidad de disco, de VCPU y RAM, entre otros.

Imagenes

OpenStack por medio de Glance y del dashboard permite al usuario la subida de imágenes (Figura 15) de varios tipos de formato. Para subir una imagen solamente se le debe ser indicada al Dashboard la ubicación y que sea compatible con alguno de los formatos, preferiblemente las imagenes deben estar en formato qcow2. Debido a que este tipo de formato está pensado para el uso con el hypervisor QEMU.

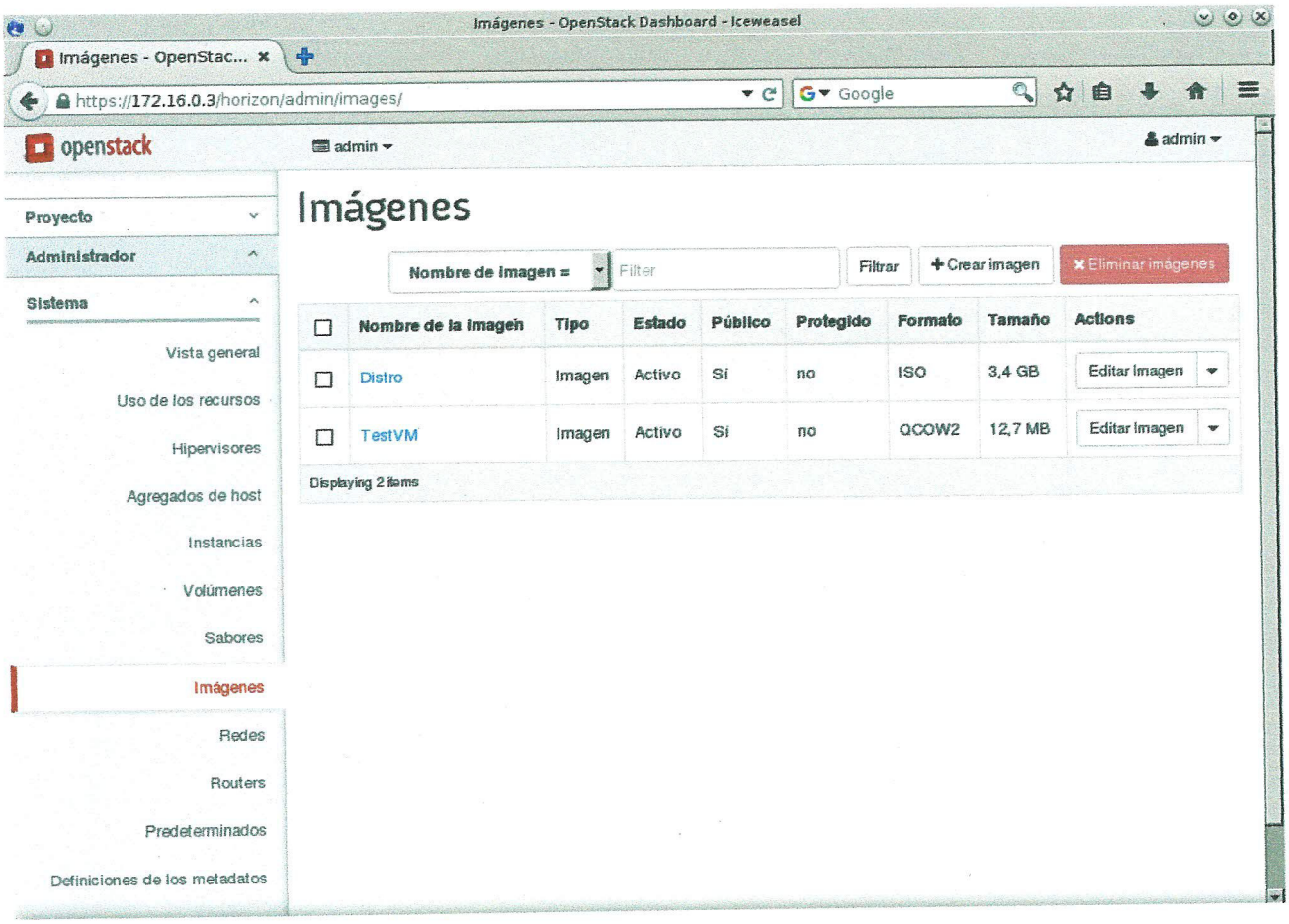


Figura 15. Imagenes.

Instancias

Sección en la que el usuario podrá lanzar una instancia. Al momento de lanzar la instancia se deben indicar varios elementos configurados anteriormente (Figura 16):

La imagen: Se selecciona que imagen se desea utilizar para la instancia a lanzar, estas imagenes son subidas a OpenStack por medio del servicio Glance. En el caso de este trabajo se utilizará una imagen llamada Distro.

El sabor: Se coloca el sabor configurado previamente para el funcionamiento de la instancia.

La red: Es necesario seleccionar una red ofrecida por Neutron para otorgarle una dirección IP a la instancia.

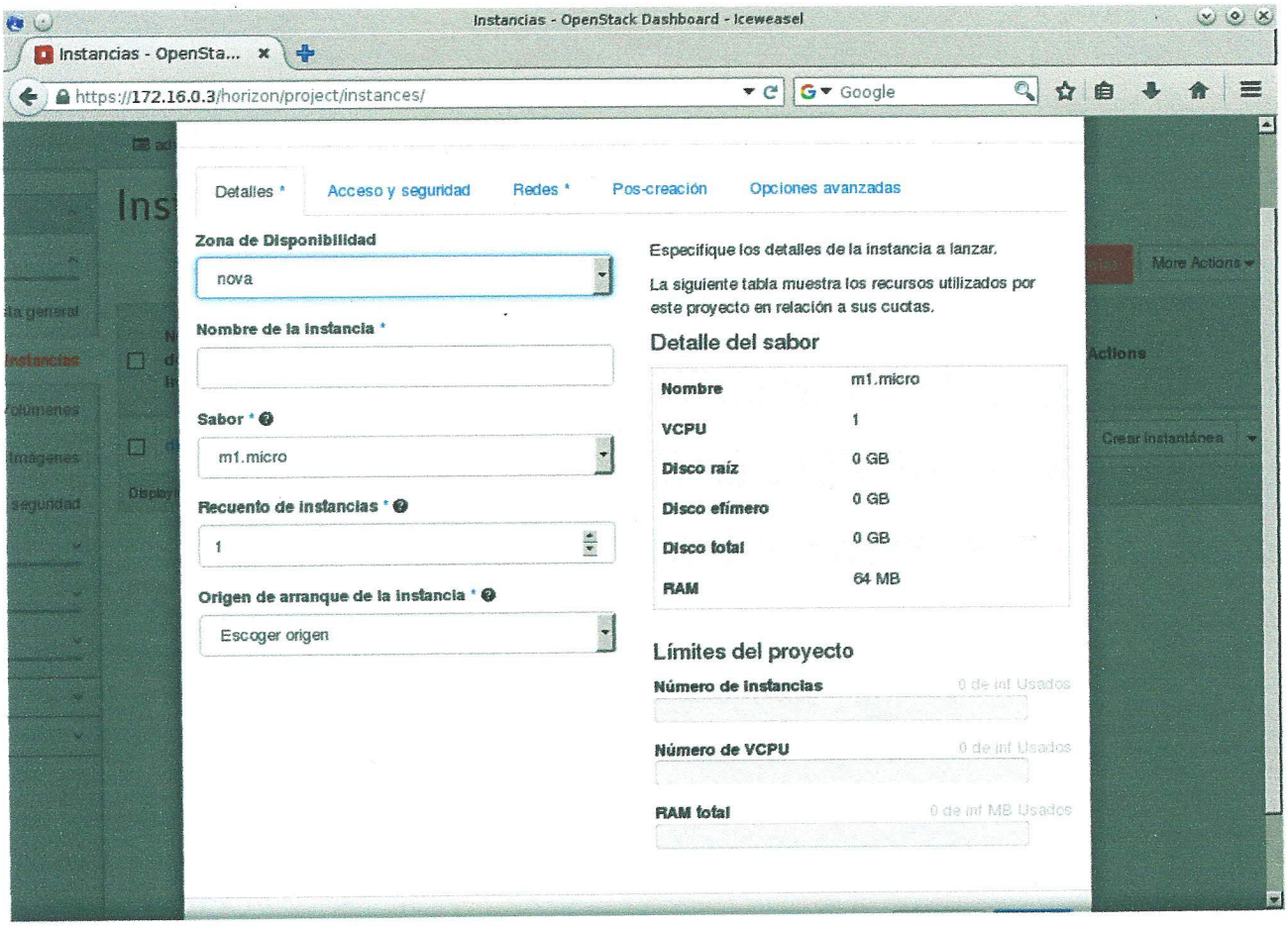


Figura 16. Lanzamiento de una instancia.

Una vez seleccionado todo se puede lanzar la instancia y en unos minutos aparecerá en el dashboard una terminal para poder acceder de manera remota. Además, OpenStack ofrece un link para que el usuario pueda acceder a esa instancia de manera gráfica permitiendole el uso y manejo de la instancia. Este link viene acompañado de un token de manera que sin el mismo sea imposible acceder a la instancia (Keystone). Una de las caracteristicas de HETNUX consiste en tener un modo live para ejecutar el Sistema Operativo sin la necesidad de realizar una instalación. Entonces, OpenStack puede correr la instancia sin la necesidad de la instalación por lo que se debe generar la instancia para luego HETNUX se ejecute en modo live y así poder utilizarlo (Figura 17).

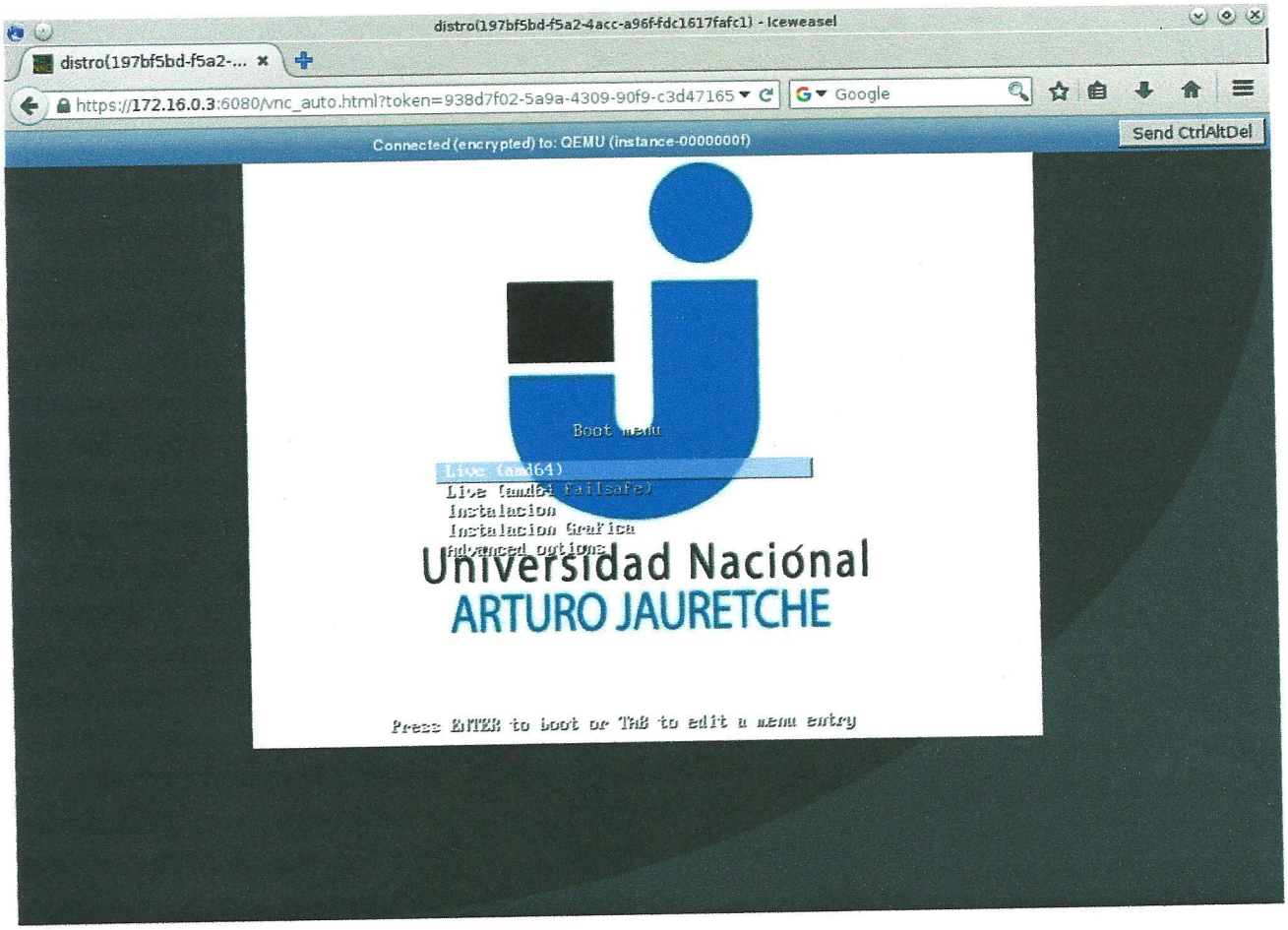


Figura 17. Hetnux lanzado.

Resultados

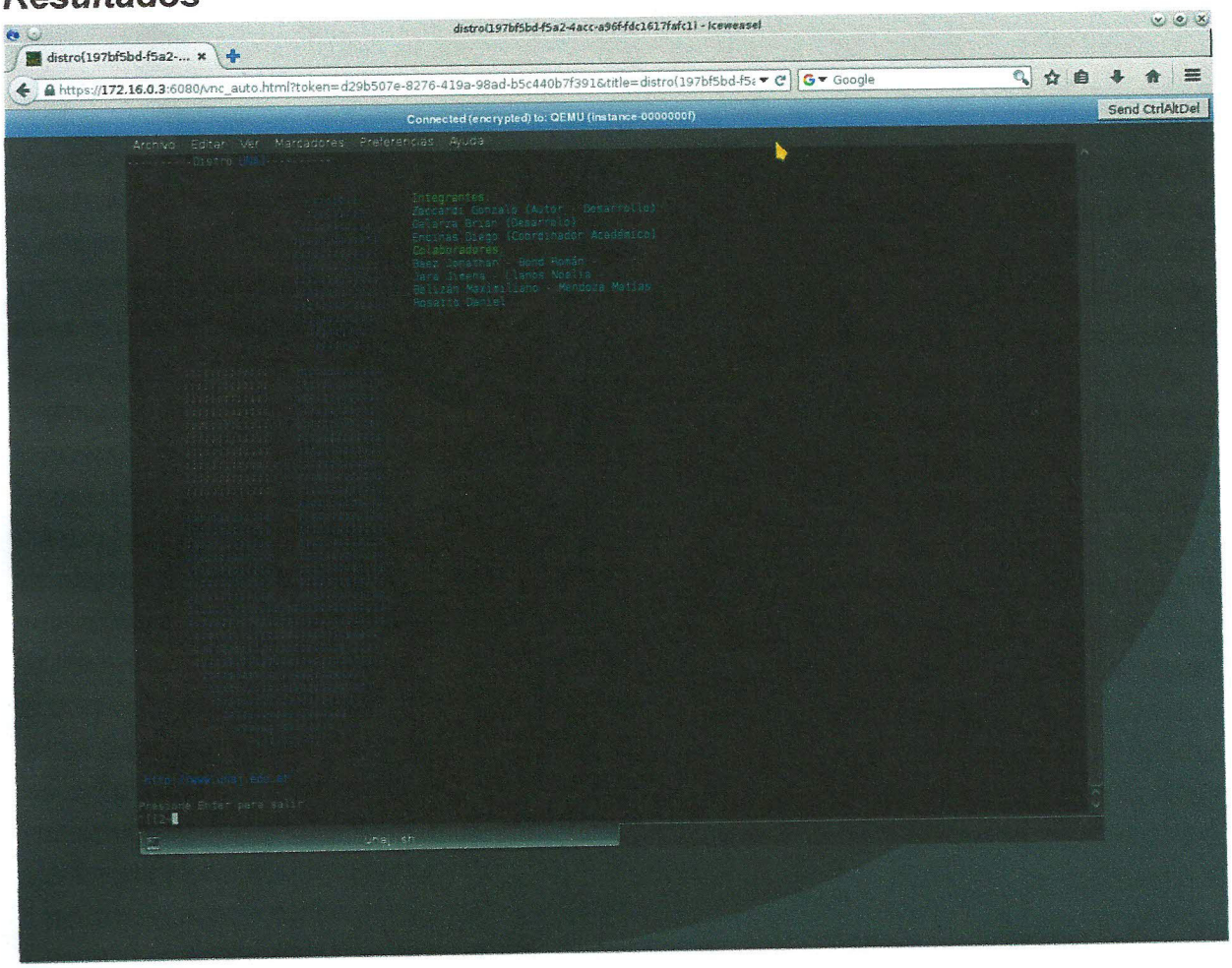


Figura 18. Instancia con HETNUX ejecutandose.

Se ha podido lanzar una instancia (Figura 18) la cual cuenta con la distribución linux desarrollada en el marco del programa Universidad, Diseño y Desarrollo Productivo 2013.

La distribución al contar con un modo live permite la ejecución del sistema operativo en la nube sin necesidad de tener que realizar la instalación, lo que ofrece un mayor dinamismo frente a otros sistemas operativos. Además cuenta con diversos programas utilizados durante la carrera lo que permite que cualquier alumno que quiera acceder a un equipo y trabajar con programas especificos, lo pueda hacer sin la necesidad de tener que instalar ni configurar en absoluto.

Al tener la distribución disponible en la nube, cualquier alumno podria acceder mediante un usuario con permisos de lanzar instancias y de esta manera poder utilizar la distro libremente. Una vez finalizado el uso, puede terminar la instancia liberando sus recursos de manera que pueda ser

utilizable por otro usuario.

Lamentablemente por cuestiones de Hardware, solo es posible lanzar una sola instancia con la distro y la cual no tiene un rendimiento óptimo. Una solución sería realizar una escalado horizotal como vertical de manera que, mediante PXE, Fuel pueda reconocer nuevos equipos los cuales se les instalará como Compute Node y de esta manera ser utilizables por la infraestructura con la que se cuenta.

Capítulo 4

Conclusiones

De acuerdo a los objetivos propuestos se puede concluir que se ha conseguido:

- **Implementar los desarrollos realizados durante los trabajos de investigación.** El funcionamiento correcto de OpenStack no es una tarea sencilla, durante estos años se ha encontrado con una elevada cantidad de errores y bugs que debido a los cambios y diferentes configuraciones disponibles, no podían solucionarse. Sin embargo con la utilización de Fuel se han facilitado el tratamiento de varios de esos problemas debido a la automatización de la instalación.

- **Ofrecer soluciones eficientes por medio de las instancias lanzadas.** El lanzamiento de las instancias pueden ofrecer distintos servicios (por medio de sistemas operativos personalizados, snapshots, etc.) las cuales aprovecharán los recursos disponibles gracias a los sabores.

- **Configuración de instancias personalizadas que puedan ofrecer distintos servicios predeterminados.** HETNUX es una prueba de ello, al lanzar una instancia con ella, cualquier usuario va a disponer inmediatamente de un conjunto de herramientas ya configuradas para su uso.

- **Poder otorgar servicios mediante la infraestructura implementada.** La implementación de una IaaS permite la implementación de PaaS como de SaaS. Estas plataformas pueden ofrecerse a través de instancias personalizadas mientras que SaaS pueden ofrecerse por medio de un desarrollo que luego utilicen el procesamiento y storage del Cloud.

- **Posibilidad de escalar el sistema mediante el uso de PXE.** Gracias a PXE y Fuel es posible conectar un nuevo nodo al cluster y por medio del dashboard de Fuel poder configurarlo como compute o controller

- **Ofrecer HPC (High Perfomance Computing) debido al uso eficiente de los recursos.** Mediante Nova y el dashboard de OpenStack es posible crear, mantener y eliminar instancias, por lo que se podrán terminar las instancias que no se usen más y de esta manera liberar recursos para otras instancias que se encuentro corriendo.

- **Posibilidad de seguir trabajando en la infraestructura implementada tanto en el ámbito de producción como de investigación.** Es posible ofrecer servicios mediante las instancias personalizadas, otorgándoles a los usuarios procesamiento, memoria y almacenamiento. Además se pueden realizar investigaciones al respecto del funcionamiento del Cloud y las diferencias en rendimiento en comparación a servicios de pago como Amazon Web Services o Microsoft Azure.

El paradigma de cloud (de ofrecer hardware y software como servicio) es un mercado novedoso el cual ofrece a los usuarios una abstracción de los elementos a utilizar para que el mismo se enfoque en utilizar los recursos para lo necesitado. Esto trae una solución elegante a los usuarios sin tanto conocimiento. Además Cloud aprovecha las ventajas de un Cluster como de conceptos muy importantes como HPC (High Perfomance Computing) y HA (High Availabity). De cara al futuro Cloud junto a Internet de las cosas son los paradigmas que posiblemente abrirán nuevos mercados y dirigirán el rumbo de la informática hacia los servicios y la conectividad de las cosas.

Agradecimientos

Quiero agradecer a mi mamá y a mis hermanos, por estar siempre y acompañarme en todo momento durante toda la carrera.

A mi papá, sé que ahora estas mejor. Te extraño.

A mis abuelos quienes también estuvieron acompañando en la carrera.

A mis amigos y compañeros con los que compartí varias cursadas, Gonzalo, Leandro, Leonel, David y a todos los que estuvieron desde que inicié la carrera hasta finalizarla.

A Jenny, você é uma pessoa muito boa. Sei que vai poder fazer tudo o que você se proponha. Saudades de você.

A los docentes de la UNAJ, principalmente a Diego y a Martín, quienes fueron guiandome y acompañando desde mis primeros años hasta el final de la carrera.

Referencias

[1]Galarza Brian, Zaccardi Gonzalo , Tuamá Cristian, Morales Martín, Encinas Diego. *"Implementaciones de Cloud Computing y aplicaciones en el ámbito universitario"*, CoNaIISI 2013

[2]https://wiki.openstack.org/wiki/Fuel. Fuel Openstack. Junio de 2016.

[3]https://www.mirantis.com/. Mirantis. The pure Openstack Company. Junio de 2016

[4]https://technet.microsoft.com/es-es/library/cc732649%28v=ws.10%29.aspx. Como funciona el servidor PXE. Microsoft

[5]http://www.syslinux.org/wiki/index.php?title=PXELINUX. PXELINUX.

[6]http://www.pix.net/software/pxeboot/archive/pxespec.pdf. The Preboot Execution Environment specification v2.1. Intel y Systemsoft.

[7]https://www.openstack.org/. OpenStack Open Source Cloud Computing Software

[8]http://docs.openstack.org/

Universidad Nacional ARTURO JAURETCHE

Anexo

Publicaciones

- Galarza Brian, Zaccardi Gonzalo , Tuamá Cristian, Morales Martín, Encinas Diego. *"Implementaciones de Cloud Computing y aplicaciones en el ámbito universitario"*, CoNaIISI 2013.
- *Galarza Brian, Zaccardi Gonzalo , Morales Martín, Encinas Diego."Implementación de Cloud Computing utilizando OpenStack"*, CoNaIISI 2014.
- *Encinas Diego, Kunysz Eduardo, Galarza Brian, Zaccardi Gonzalo, Morales Martín."Performance de arquitecturas multiprocesador: técnicas de modelado y simulación, plataformas reconfigurables y cloud computing"*, WICC 2015.
- *Galarza Brian, Zaccardi Gonzalo, Encinas Diego, Morales Martín. "Análisis de despliegue de una IaaS utilizando Openstack"*, CACIC 2015.
- *Galarza Brian, Zaccardi Gonzalo, Rosatto Daniel, Bond Román, Morales Martín, Encinas Diego. "Procesamiento Distribuido y Paralelo"*, WICC 2016.
- *Zaccardi Gonzalo, Galarza Brian, Morales Martín, Encinas Diego. "Despliegue y ejecución de un cloud privado"*, CoNaIISI 2016.

Convocatorias

- Universidad Diseño y Desarrollo Productivo,"Desarrollo Distribucción Linux", 2013.
- Universidad Diseño y Desarrollo Productivo,"Implementación Nubes Virtuales", 2015.

Exposiciones

- Participación y exposición en el stand "yo quiero estudiar" en Tecnopolis. 26 de Septiembre de 2014
- Encuentro de prototipos del programa Universidad Diseño y Desarrollo Productivo en la Universidad Nacional de Lanús (UNLA). Octubre de 2014.

Además cabe destacar que el proyecto de la convocatoria Universidad Diseño y Desarrollo Productivo "Desarrollo Distribucción Linux" se encuentra registrado con el nombre de **HETNUX**.

Morales
Ingeniería en Informática
de Ingeniería y Agronomía
U.N.A.J.

DIEGO ENCINAS