

Análisis de despliegue de una IaaS utilizando Openstack

Brian Galarza¹, Gonzalo Zaccardi¹, Diego Encinas^{1,2}, Martín Morales¹

¹ Instituto de Ingeniería y Agronomía - Universidad Nacional Arturo Jauretche

² Instituto de Investigación en Informática LIDI – Facultad de Informática – Universidad Nacional de La Plata

briangalarza@hotmail.com, pcg.computers@hotmail.com, dencinas@unaj.edu.ar,
martin.morales@unaj.edu.ar

Resumen. El presente trabajo tiene como objetivo realizar un despliegue de una Infraestructure as a Service (IaaS). Para ello se utilizará software de código abierto (Openstack). Se demostrará un escenario con la arquitectura Legacy Networking (nova-network) en la cual se tendrán 2 nodos (controller y compute) para poder lanzar instancias virtualizadas. Se mostrarán los resultados obtenidos de la implementación realizada para, finalmente, concluir con un análisis de las ventajas que conllevarían la implementación de la misma en un ámbito universitario.

Palabras claves: Cloud-Computing, IaaS, Openstack, Node, Networking, Hypervisor.

1 Introducción

Cloud Computing es un paradigma que está en constante crecimiento durante estos últimos años, cada vez más compañías y grupos de investigación trabajan en conjunto con el fin de explotar las oportunidades ofrecidas por el mismo. Dicho paradigma ofrece muchas ventajas, tales como el bajo costo de implementación, ya que no se necesitan computadoras de última tecnología debido a que éstas trabajan conjuntamente (Clustering) con la posibilidad de escalar horizontalmente de manera sencilla. Además, hay software Open Source disponible para los nodos en el clúster como las infraestructuras Eucalyptus, OpenNebula, CloudStack u OpenStack integradas con GNU/Linux y compatibles, por ejemplo, con Amazon WebServices. Como objetivos principales de esta investigación y aplicación práctica/técnica se proponen:

- Mostrar las ventajas y desventajas de OpenStack como sistema de administración de nube (Cloud Computing) a partir de la infraestructura básica (un nodo controller y, al menos, un nodo compute).
- Proponer una infraestructura de red propia e independiente en el ámbito universitario.

1.1 Cloud Computing

Cloud computing [1] es un paradigma de computación en franco avance y tiene muchos servicios que aún no han sido desarrollados. La capacidad de ofrecer servicios a usuarios de manera sencilla y económica propendió que empresas como Google, Facebook, Outlook, Yahoo, Hotmail, etc. utilizaran sus recursos en hardware y software para desarrollar "nubes" que almacenan todos los datos de los usuarios ofreciendo distintos tipos de servicios:

IaaS (Infrastructure as a Service): Se contrata capacidad de proceso (CPU) y almacenamiento. En este caso sólo se paga por lo utilizado. Ejemplos: EC2 de Amazon y Azure de Microsoft.

PaaS (Platform as a Service): Se proporciona almacenamiento, un servidor de aplicaciones (en donde se ejecutarán las aplicaciones requeridas) y una base de datos. Ejemplo: Google App Engine.

SaaS (Software as a Service): Es una aplicación para el usuario final en donde se paga un alquiler por el uso de software. Muchas de estas aplicaciones son actualmente gratuitas. Ejemplos: Google Docs, Office 365, Dropbox.

HaaS (Hardware as a Service): Consiste en ofrecer hardware de manera que sea utilizada a través de internet logrando que el sistema utilice los recursos distribuidos geográficamente como si estuvieran localmente. Ofreciendo una mayor eficiencia de los mismos.

A su vez cada uno de ellos puede ser:

- Público: Una empresa ofrece IaaS a terceros, encargándose de toda la gestión del Cloud. El caso más conocido es Amazon Elastic Compute Cloud (EC2).
- Privado: Una organización configura sus propios recursos como IaaS para tener más flexibilidad y control total sobre sus recursos.
- Híbrido: Algunos servicios se gestionan en el cloud privado y otros se transfieren a uno público, normalmente utilizan una API común que permita una buena integración.

1.2 OpenStack

OpenStack [2] [3] es un software de código abierto que permite la implementación de una Infraestructure as a Service (IaaS) a través de múltiples servicios que, de manera coordinada, cumplen diferentes propósitos para lograr el correcto funcionamiento de dicha infraestructura. Cada servicio ofrece una interfaz de programación de aplicación (API) que facilita la integración de las mismas. Dependiendo de las necesidades de cada usuario, puede instalar algunos o todos los servicios disponibles. Estos servicios disponibles se describen en la Figura 1.

Servicio	Nombre del proyecto	Descripción
Dashboard	Horizon	Proporciona una interfaz gráfica web basado en los servicios subyacentes de OpenStack permitiendo el lanzamiento de una instancia, la asignación de direcciones IP y la configuración de los controles de acceso, etc.
Compute	Nova	Gestiona el ciclo de vida de las instancias de proceso en un ambiente de OpenStack. Las responsabilidades incluyen la descomposición, programación y el cierre definitivo de las máquinas en la demanda.
Networking	Neutron	Permite la conectividad de la red para uno o varios servicios de OpenStack, como OpenStack Compute. Proporciona una API para que los usuarios definan las redes y los archivos adjuntos en ellos. Tiene una arquitectura conectable que soporta varios proveedores y tecnologías de redes populares.
Almacenamiento		
Object Storage	Swift	Almacena y recupera los objetos de datos no estructurados de su elección mediante una REST, es una API basada en HTTP. Es muy tolerante a fallos con su réplica de datos y su posible escalabilidad. Su puesta en práctica no es como un servidor de archivos con los directorios montables.
Block Storage	Cinder	Proporciona almacenamiento persistente en bloques para las instancias en ejecución. Soporta dispositivos plug and play y permite gestionar, por lo tanto, todos aquellos dispositivos de almacenamiento en bloques.
Servicios compartidos		
Identity Service	Keystone	Proporciona un servicio de autenticación y autorización para otros servicios de OpenStack. Proporciona también un catálogo de endpoints para todos los servicios de OpenStack.
Image Service	Glance	Permite almacenar y recuperar imágenes de disco de máquinas virtuales. OpenStack Compute hace uso de esta instancia durante la instancia de provisión.
Telemetry	Ceilometer	Monitorea la nube de OpenStack para la evaluación comparativa, la escalabilidad y con fines estadísticos.
Servicios alto nivel		
Orchestration	Heat	Organiza múltiples aplicaciones en la nube de material compuesto utilizando ya sea el formato de la plantilla HOT nativo o el formato de la plantilla AWS, tanto a través de una API REST nativa de OpenStack nativo y una forma de nube compatible con la API de queries.

Figura 1. Servicios de OpenStack.

La arquitectura conceptual de OpenStack se puede ver en la Figura 2:

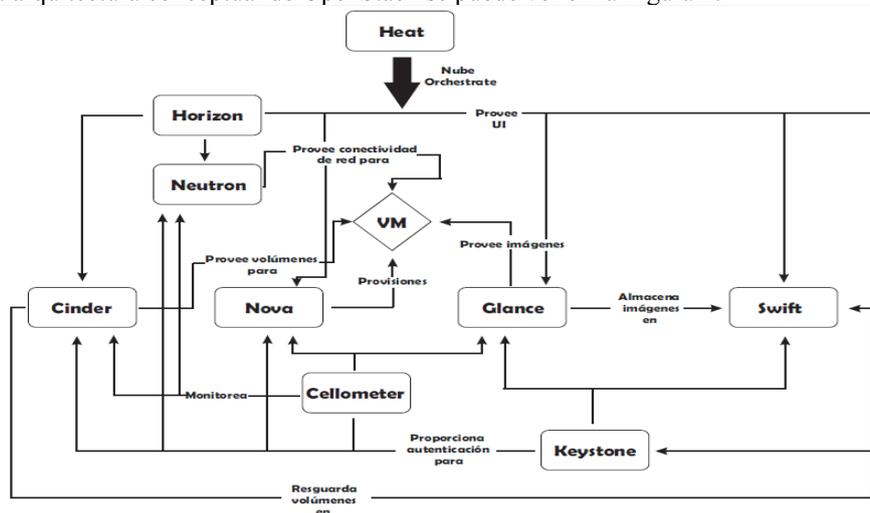


Figura 2. Arquitectura conceptual de OpenStack.

Varias compañías se han unido al proyecto, entre ellas, Red Hat, AT&T, Canonical, Cisco, Dell, Go Daddy, HP, IBM, Intel, Rackspace Hosting, Nexenta, AMD, Suse, VmWare, Oracle, Yahoo, entre otras.

2 Escenario

Para la implementación de la nube, se ha trabajado con dos equipos. Cada uno con una máquina virtual ejecutando Debian 8.1 [4] y en donde se instalaron los paquetes necesarios para que la infraestructura funcione correctamente.

Al utilizar la arquitectura legacy se tienen como mínimo dos nodos. Un nodo controller y un nodo compute.

Los servicios instalados en cada uno de los nodos son los indicados en la Figura 3.

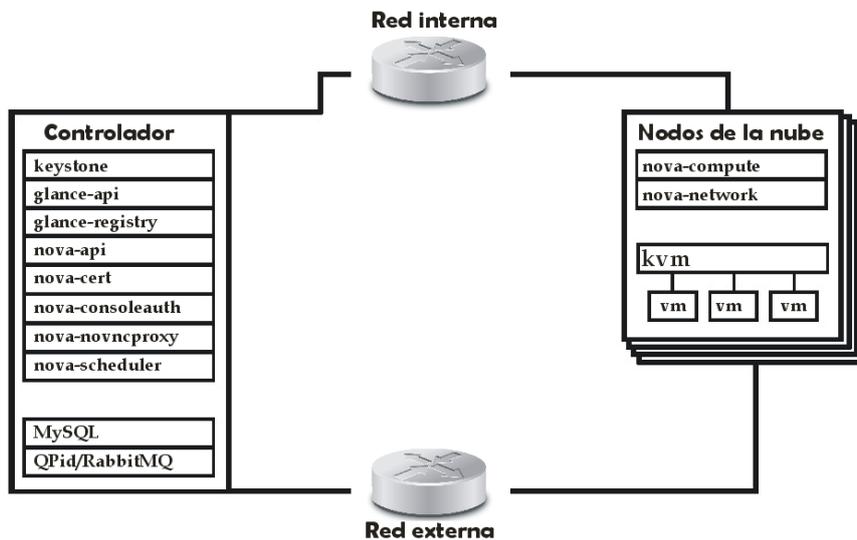


Figura 3. Legacy Networking.

El hipervisor seleccionado para el nodo compute depende del equipo utilizado, en caso de tener un equipo que soporte virtualización de tipo 1, se puede utilizar KVM o XEN, etc. En caso de no soportar, se puede utilizar QEMU [5].

Los dos equipos deben tener dos interfaces de red como mínimo. Una para la red externa y otra para la red interna. La red interna será utilizada para cuando se arme la red virtual legacy en la que se asigna un espacio de direcciones a los nodos que serán instanciados.

Los dos nodos se ejecutaron sobre máquinas virtuales utilizando Oracle virtualbox [6]. El utilizar máquinas virtuales reduce el rendimiento del nodo, sin embargo ofrece

las ventajas de tener todo virtualizado y poder realizar snapshots periódicamente lo cual permite retroceder fácilmente en caso de inconvenientes.

En las siguientes figuras (Figura 4 y Figura 5) se pueden observar cómo queda el esquema de los nodos.

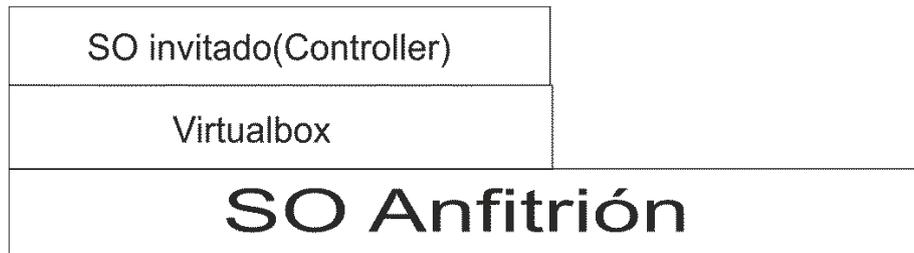


Figura 4. Controller node.

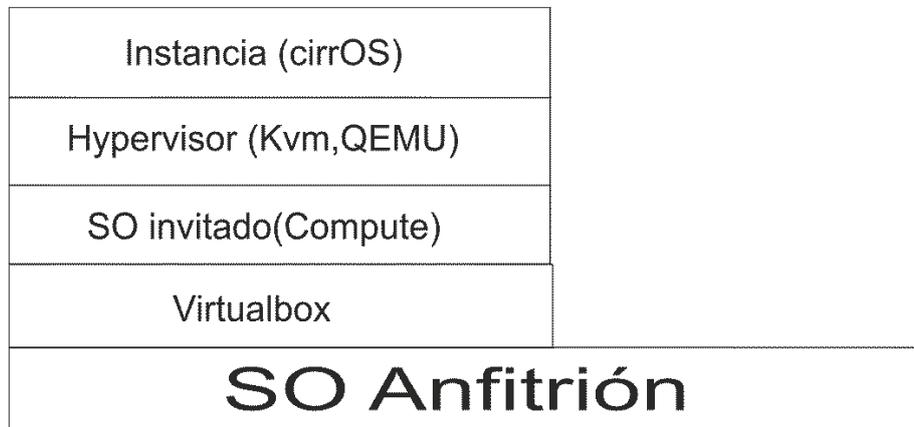


Figura 5. Compute node.

Openstack recomienda utilizar cirrOS [7] para entornos de prueba, sin embargo, es posible usar una imagen de cualquier distribución para el lanzamiento de instancias. Además es posible utilizar algún SO paravirtualizado [8] lo que optimizaría el rendimiento de las instancias lanzadas.

3 Resultados obtenidos

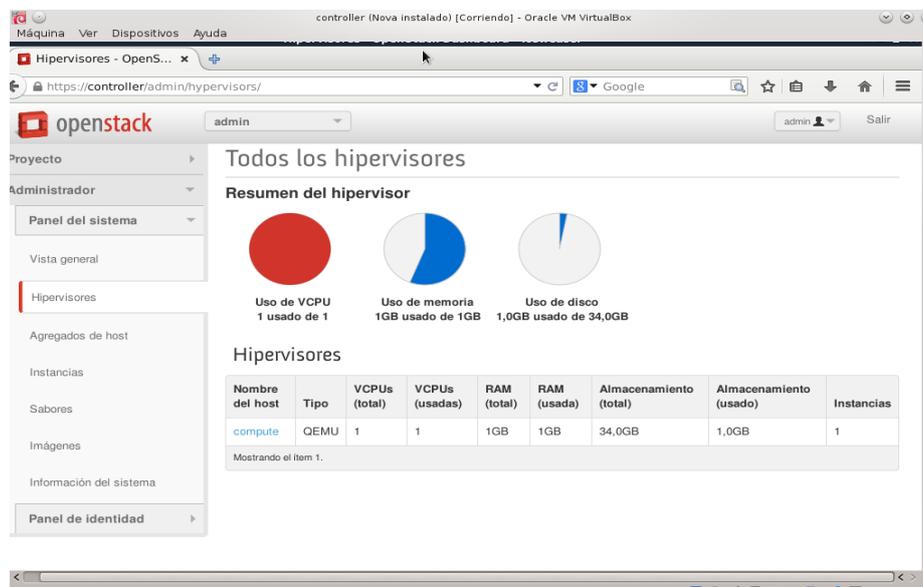


Figura 6. Hipervisor con una instancia lanzada.

En la Figura 6 puede observarse la interfaz que ofrece Openstack llamado Horizon en la cual se accede con el usuario y contraseña elegidos en la instalación y con los permisos de administración para poder realizar todas las operaciones. En ella se puede realizar todas las operaciones principales y observar todos los servicios instalados, nodos e instancias lanzadas. La principal ventaja que ofrece Dashboard consiste en la facilidad en usarlo para los usuarios sin elevado conocimiento en Linux, ya que la interfaz es muy intuitiva y ofrece todas las posibilidades que se pueden hacer por terminal.

Al dashboard se accede por medio de la dirección <https://controller> por lo que si todos los equipos de la red tienen configurada la resolución del host, pueden acceder remotamente [9]. También es posible configurar para acceder con la dirección IP del nodo controller, lo que permitiría acceder desde una red externa, sin embargo no es recomendable por razones de seguridad.

En la Figura 7 se puede observar como los servicios instalados en el controller se encuentran funcionando en el mismo (dirección IP usada 192.168.0.105)

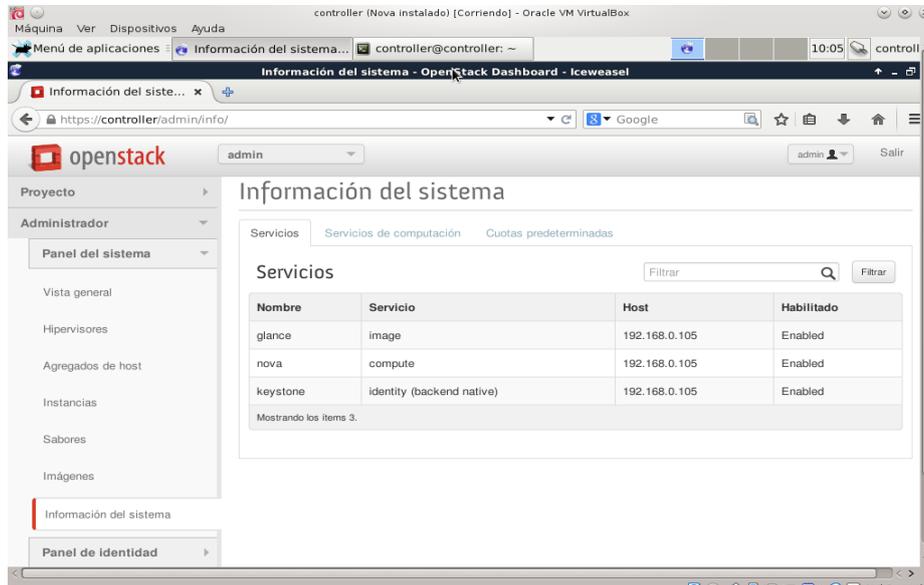


Figura 7. Servicios instalados y habilitados correctamente.

Estos servicios son los que necesita el nodo controller para poder funcionar correctamente. Al momento de lanzar una instancia se deben tener en cuenta la existencia de los sabores y de crear una red virtual. La red virtual es creada por el servicio Nova el cual otorgará direcciones IP flotantes a las instancias lanzadas y que ofrece configurar grupos de seguridad con reglas como el tipo de conexión (TCP, UDP) abrir un determinado puerto o un rango de ellos como también restringir las direcciones IPS con las que se permiten el tráfico, los nodos de un grupo de seguridad se podrán comunicar entre ellos y con otros grupos de seguridad (hay que indicarlo en las reglas de los mismos). Los sabores son configuraciones de los recursos que se tomarán al lanzar la instancia, estos son configurables por lo que es posible disponer de varias configuraciones según las necesidades de cada caso. Openstack ofrece algunos sabores predeterminados como el m1.tiny que cuenta con 1 VCPU, 512 MB RAM y 1 GB de disco como también el m1.xlarge que cuenta con 8 VCPUS, 16384 MB RAM y 160 GB de disco. En la Figura 8 se pueden observar los distintos sabores.

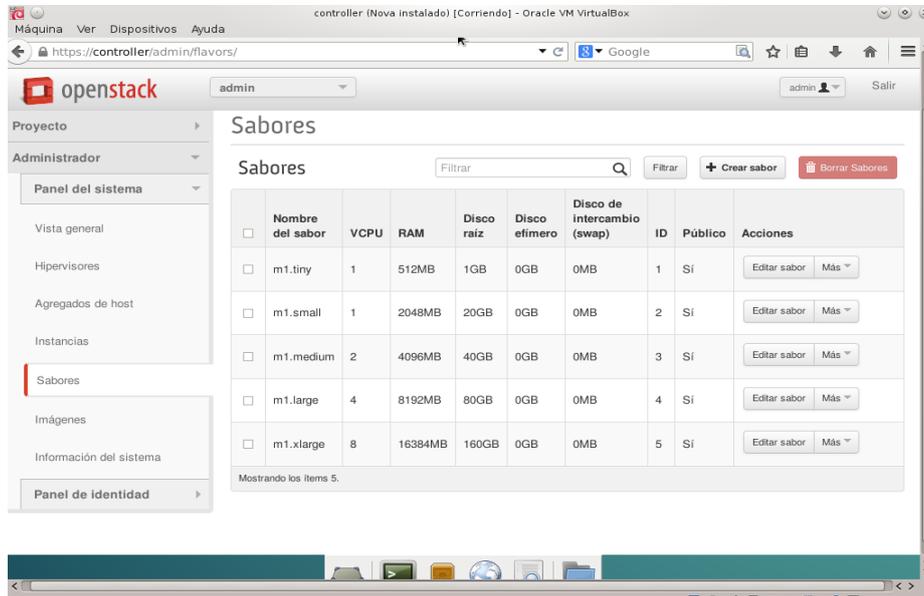


Figura 8. Sabores de openstack.

La principal problemática encontrada consiste en que los lanzamientos de Debian 8 y Openstack Kilo fueron muy cercanos (la salida de Debian 8 fue 5 días antes de la salida de Kilo), esto generó algunos problemas en los paquetes de instalación que se encuentran en los repositorios de Debian Jessie, además se remarca que aún no existe documentación sobre la instalación en Debian 8 por lo que es recomendable seguir con otra distribución de las que aparecen en la documentación (Ubuntu, CentOS, Fedora, etc.) o esperar a que Openstack se establezca en la nueva versión de Debian lanzada recientemente.

Conclusiones

Si bien OpenStack no es sencillo de implementar y hay que tener conocimientos básicos de redes una vez iniciada la expansión de la nube o clúster, es muy sencillo escalarlo. Se puede decir que puede hacerse de manera exponencial, ya que es un sistema muy modulado lo que permite tanto expandirse como reducirse de manera rápida y segura, una vez implementado correctamente los cimientos del sistema.

Finalmente, a partir de lo que se ha mostrado en el presente trabajo, puede decirse que OpenStack ofrece una gran variedad de servicios, los cuales, llevados a una buena práctica se traducen a un sin fin de posibilidades.

En línea con lo trabajado y expuesto, se busca:

Implementación de un IaaS encargado de realizar operaciones en procesamiento paralelo aumentando la eficiencia y reduciendo los costes generados.

Implementación de OpenStack Dashboard y de un sistema propio desarrollado para poder controlar/administrar de manera visual (WEB) y más básica cada uno de los servicios.

Implementación de una nube académica en la cual se puedan ofrecer distintos servicios personalizados y atender a las necesidades de cómputo y procesamiento de los distintos departamentos de la universidad.

Por último, cabe destacar que Openstack es una herramienta con una comunidad que periódicamente realizan actualizaciones con el fin de ofrecer mejores servicios y más variantes para poder adaptarse.

Referencias

1. Galarza, B.; Tuamá, C.; Zaccardi, G.; Encinas, D.; Morales, M. "Implementaciones de Cloud Computing y aplicaciones en el ámbito universitario". I Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaISI 2013). Ciudad de Córdoba, Argentina.
2. Encinas, D; Kunysz, E; Galarza, B; Zaccardi, G; Morales, M. "Performance de arquitecturas multiprocesador: técnicas de modelado y simulación, plataformas reconfigurables y cloud computing." XVII Workshop de Investigadores en Ciencias de la Computación (WICC 2015). Ciudad de Salta, Argentina.
3. OpenStack Cloud Software: Open source software for building private and public clouds. <http://www.openstack.org>. Agosto 2015.
4. Debian. The universal operating system. <https://www.debian.org> Agosto 2015
5. QEMU. Open source processor emulator. http://wiki.qemu.org/Main_Page. Agosto 2015
6. Oracle Virtualbox. <https://www.virtualbox.org/>. Agosto 2015
7. CirrOS. Tiny OS that specializes in running on a cloud. <https://launchpad.net/cirros>. Agosto 2015
8. Tanenbaum, Andrew S. 3rd Edition. Modern Operating Systems. Pearson
9. Kurose, James F; Ross, Keith W. 5th Edition. Computer Networking. Pearson