

# Despliegue de Arquitectura de nube con OpenStack

PROYECTO FINAL

Ciclo Formación Profesional Superior de  
Administración de Sistemas informáticos en Red

Pedro Vicente López Bañón

Tutor: Francisco Javier López Mota



Proyecto final Ciclo Formación Profesional Superior de  
Administración de Sistemas informáticos en Red

# Despliegue de Arquitectura de nube con OpenStack

Pedro Vicente López Bañón <sup>1 2</sup>

Junio 2018

<sup>1</sup>Tutor: Francisco Javier López Mota

<sup>2</sup>I.E.S. José Luís Castillo Puche

**PROYECTO FINAL DE CICLO DE FORMACIÓN PROFESIONAL SUPERIOR DE ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN RED.**

Este Proyecto final fue realizado bajo la supervisión del profesor Francisco Javier López Mota y el soporte del I.E.S. José Luís Castillo Puche de Yecla durante marzo, abril, mayo y junio de 2018.

*Edición única, Junio 2018*



[https://github.com/pvlopez/Proyecto\\_Final\\_FPS\\_ASIR\\_Pedro\\_V\\_Lopez\\_Banon](https://github.com/pvlopez/Proyecto_Final_FPS_ASIR_Pedro_V_Lopez_Banon)

©2018 \* PEDRO VICENTE LÓPEZ BAÑÓN

pedrovicenteloba@gmail.com

ESTA OBRA ESTÁ SUJETA A LA LICENCIA RECONOCIMIENTO 4.0 INTERNACIONAL DE CREATIVE COMMONS.



PARA VER UNA COPIA DE ESTA LICENCIA, VISITE: <https://creativecommons.org/licenses/by-sa/4.0/legalcode>

*"La única manera de hacer  
un trabajo genial es amar lo  
que haces."*

---

*Steve Jobs*



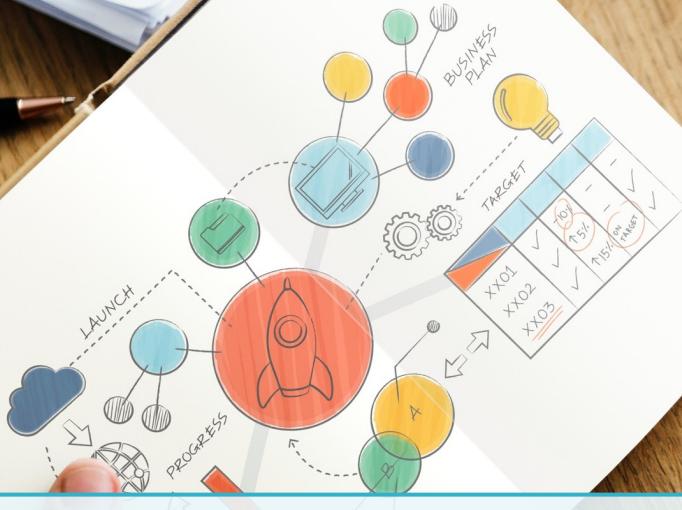
---

## Sinópsis

---

*Esto es el resumen.*





# Índice general

<b>Glosario</b>	11
<b>Siglas</b>	13
<b>Índice de figuras</b>	15
<b>Índice de cuadros</b>	17
<b>1 Introducción</b>	19
1.1 Motivación	19
1.2 La computación en la nube	20
1.3 Tipos de nube	21
1.3.1 Tipos de servicio	21
1.4 Objetivo del proyecto	22
1.5 Porqué OpenStack	23
1.5.1 Historia y alcance de OpenStack	23
<b>2 Punto de partida y planificación inicial</b>	25
2.1 Situación de partida	25
2.2 Recursos disponibles	25
2.2.1 Recursos de Hardware	25
2.2.2 Recursos de software	26
2.2.3 Subvenciones y ayudas	26
2.3 Entorno de desarrollo	26
2.3.1 Oracle VirtualBox	26
2.3.2 CentOS 7	27

2.3.3	Python .....	28
2.3.4	Git y Github .....	28
2.3.5	IDEs y editores .....	29
2.3.6	Control de proyectos .....	29
2.3.7	Documentación .....	30
<b>3</b>	<b>OpenStack .....</b>	<b>31</b>
<b>3.1</b>	<b>Introducción a OpenStack</b>	<b>31</b>
<b>3.2</b>	<b>Componentes básicos de OpenStack</b>	<b>31</b>
3.2.1	Servicio de identidad (Keystone) .....	31
3.2.2	Servicio de cómputo (Nova) .....	33
3.2.3	Servicio de red (Neutron) .....	34
3.2.4	Servicio de imagen (Glance) .....	34
3.2.5	Servicio de almacenamiento de objetos (Swift) .....	35
3.2.6	Servicio de almacenamiento en bloque (Cinder) .....	36
3.2.7	OpenStack dashboard (Horizon) .....	36
<b>3.3</b>	<b>Componentes adicionales de OpenStack</b>	<b>37</b>
3.3.1	Servicio de orquestación (Heat) .....	37
3.3.2	Servicio de telemetría (Ceilometer) .....	38
3.3.3	Servicio de bases de datos (Trove) .....	38
3.3.4	Servicio de procesamiento de datos (Sahara) .....	38
3.3.5	Otros componentes .....	39
<b>3.4</b>	<b>OpenStack en entornos de producción</b>	<b>40</b>
3.4.1	RDO .....	40
3.4.2	Triple-O .....	40
3.4.3	Otros .....	41
<b>4</b>	<b>Seguridad y Notas Legales .....</b>	<b>43</b>
<b>5</b>	<b>Escenario y configuración inicial .....</b>	<b>45</b>
<b>5.1</b>	<b>Configuración de red</b>	<b>45</b>
<b>5.2</b>	<b>Preconfiguración de los equipos</b>	<b>45</b>
5.2.1	Configurar autenticación sin contraseña en el nodo controller .....	46
<b>6</b>	<b>Instalación de OpenStack .....</b>	<b>47</b>
<b>6.1</b>		<b>47</b>
6.1.1	.....	47
6.1.2	.....	47
<b>6.2</b>		<b>47</b>
<b>6.3</b>		<b>47</b>
<b>7</b>	<b>Inicialización y puesta en marcha del sistema .....</b>	<b>49</b>
<b>7.1</b>	<b>Login</b>	<b>49</b>
<b>7.2</b>	<b>Personalización</b>	<b>49</b>

---

<b>7.3</b>	<b>Usuarios y grupos</b>	<b>49</b>
7.3.1	Usuarios .....	49
7.3.2	Grupos .....	49
<b>7.4</b>	<b>proyectos</b>	<b>49</b>
7.4.1	Instancias .....	49
7.4.2	imágenes .....	49
7.4.3	volumenes .....	49
7.4.4	imágenes .....	49
7.4.5	redes .....	49
<b>8</b>	<b>Contratiempos, Soluciones y línea temporal</b>	<b>51</b>
8.1	Contratiempos y soluciones	51
8.2	Línea temporal	51
<b>9</b>	<b>Recursos utilizados, Costes y presupuesto</b>	<b>53</b>
9.1	Recursos utilizados	53
9.2	Costes	53
9.3	Presupuesto	53
<b>10</b>	<b>Epílogo</b>	<b>55</b>
10.1	Ideas para el futuro	55
10.2	Palabras finales	55
<b>11</b>	<b>Anexos</b>	<b>61</b>
11.1	Anexo A	61
11.2	Anexo B	64



# Glosario

**Big Data** Big data, macrodatos, datos masivos, inteligencia de datos o datos a gran escala es un concepto que hace referencia a un conjuntos de datos tan grandes que aplicaciones informáticas tradicionales de procesamiento de datos no son suficientes para tratar con ellos y los procedimientos usados para encontrar patrones repetitivos dentro de esos datos. Los textos científicos en español con frecuencia se usa directamente el término en inglés big data, tal como aparece en el ensayo de Viktor Schönberger: La revolución de los datos masivos.[22]. 19

**Business Intelligence** Se denomina inteligencia empresarial, inteligencia de negocios o BI (del inglés business intelligence), al conjunto de estrategias, aplicaciones, datos, productos, tecnologías y arquitectura técnicas, los cuales están enfocados a la administración y creación de conocimiento sobre el medio, a través del análisis de los datos existentes en una organización o empresa.[21]. 19

**CentOS** Es un sistema operativo de código abierto, basado en la distribución Red Hat Enterprise Linux, operándose de manera similar, y cuyo objetivo es ofrecer al usuario un software de "clase empresarial" gratuito. Se define como robusto, estable y fácil de instalar y utilizar. Desde la versión 5, cada lanzamiento recibe soporte durante diez años, por lo que la actual versión 7 recibirá actualizaciones de seguridad hasta el 30 de junio de 2024.[6]. 22

**Ciencia de datos** La ciencia de datos es un campo interdisciplinario que involucra métodos científicos, procesos y sistemas para extraer conocimiento o un mejor entendimiento de datos en sus diferentes formas, ya sea estructurados o no estructurados,<sup>1</sup> lo cual es una continuación de algunos campos de análisis de datos como la estadística, la minería de datos, el aprendizaje automático y la analítica predictiva.[7]. 19

**fully qualified domain name** Un FQDN (sigla en inglés de fully qualified domain name) es un nombre que incluye el nombre de la computadora y el nombre de dominio asociado a ese equipo. Por ejemplo, dada la computadora llamada «serv1» y el nombre de dominio «bar.com.», el FQDN será «serv1.bar.com.»; a su vez, un FQDN asociado a serv1 podría ser «post.serv1.bar.com.». En los sistemas de nombre de dominio de zonas, y más especialmente en los FQDN, los nombres de dominio se especificarán con un punto al final del nombre, aunque se puede omitir.. 45

**Instituto de Fomento de la Región de Murcia** El info es más que una agencia de desarrollo

de la Comunidad Autónoma de la Región de Murcia. En 30 años cumplidos de historia, hemos sumado talento y experiencia al espíritu emprendedor de los ciudadanos para avanzar juntos en los nuevos retos de un entorno en constante evolución.. 26

**Machine Learning** El aprendizaje automático o aprendizaje de máquinas (del inglés, "Machine Learning") es el subcampo de las ciencias de la computación y una rama de la inteligencia artificial cuyo objetivo es desarrollar técnicas que permitan a las computadoras aprender. De forma más concreta, se trata de crear programas capaces de generalizar comportamientos a partir de una información suministrada en forma de ejemplos. Es, por lo tanto, un proceso de inducción del conocimiento. En muchas ocasiones el campo de actuación del aprendizaje automático se solapa con el de la estadística computacional, ya que las dos disciplinas se basan en el análisis de datos. Sin embargo, el aprendizaje automático también se centra en el estudio de la complejidad computacional de los problemas. Muchos problemas son de clase NP-hard, por lo que gran parte de la investigación realizada en aprendizaje automático está enfocada al diseño de soluciones factibles a esos problemas. El aprendizaje automático puede ser visto como un intento de automatizar algunas partes del método científico mediante métodos matemáticos.[4]. 19

**OpenStack** OpenStack es una plataforma de software de código abierto para nubes privadas y públicas. OpenStack, una plataforma que ofrece infraestructura como servicio (IaaS), permite a las empresas agregar servidores, y componentes de redes y de almacenamiento de manera fácil y eficiente a su nube.. 22, 23

**Red Hat Enterprise Linux** Red Hat Enterprise Linux también conocido por sus siglas RHEL es una distribución comercial de GNU/Linux desarrollada por Red Hat. Es la versión comercial basada en Fedora que a su vez está basada en el anterior Red Hat Linux, de forma similar a como Novell SUSE Enterprise (SUSE Linux Enterprise Desktop y SLE Server) lo es respecto de OpenSUSE o Mandriva Corporate respecto de Mandriva Linux One.[28]. 22

**Replicación** La replicación es el proceso de copiar y mantener actualizados los datos en varios modos de bases de datos ya sean estos persistentes o no. Éste usa un concepto donde existe un nodo amo o maestro (master) y otros sirvientes o esclavos (slaves).[1]. 19

**Virtualización** virtualización es la creación a través de software de una versión virtual de algún recurso tecnológico, como puede ser una plataforma de hardware, un sistema operativo, un dispositivo de almacenamiento u otros recursos de red. Dicho de otra manera, se refiere a la abstracción de los recursos de una computadora, llamada Hypervisor o VMM (Virtual Machine Monitor) que crea una capa de abstracción entre el hardware de la máquina física (host) y el sistema operativo de la máquina virtual (virtual machine, guest), dividiéndose el recurso en uno o más entornos de ejecución.[wiki-virtualizacion]. 19

---

# Siglas

---

**aaaS** as a Service. 19  
**AMQP** Advanced Message Queuing Protocol. 31  
**AWS** Amazon Web Services. 26

**CMDB** configuration management database. 36

**DHCP** Dinamic Host Configuration Protocol. 19

**FTP** File Transfer Protocol. 19

**IaaS** Infrastructure. 20, 21, 23  
**IoT** Internet of Things. 20

**PaaS** Platform as a Service. 20, 21

**REST** Representation State Transfer. 31

**SaaS** Software as a Service. 20, 21

**Triple-O** OpenStack On OpenStack. 41



# Índice de figuras

1.1	Cloud Computing. . . . .	22
1.2	Objetivo: Implementar OpenStack sobre CentOS. . . . .	22
1.3	Hadoop y su simpático logo, permite a las aplicaciones trabajar con miles de nodos y petabytes de datos. <a href="http://hadoop.apache.org/">http://hadoop.apache.org/</a> . . . . .	23
1.4	OpenStack, <a href="https://www.openstack.org/">https://www.openstack.org/</a> . . . . .	23
1.5	Logos de NASA y Rackspace. . . . .	24
2.1	Oracle VirtualBox, <a href="https://www.virtualbox.org/">https://www.virtualbox.org/</a> . . . . .	26
2.2	The CentOS Project Linux, <a href="https://www.centos.org/">https://www.centos.org/</a> . . . . .	27
2.3	Python, <a href="https://www.python.org/">https://www.python.org/</a> . . . . .	28
2.4	Git y Github, <a href="https://github.com/">https://github.com/</a> . . . . .	28
2.5	Trello, <a href="https://trello.com/">https://trello.com/</a> . . . . .	29
2.6	Captura de pantalla del código L <sup>A</sup> T <sub>E</sub> X en Overleaf de este documento. . . . .	30
3.1	Diagrama OpenStack. . . . .	32
3.2	OpenStack Keystone . . . . .	32
3.3	Flujo de peticiones de RabbitQM. . . . .	33
3.4	Diagrama de Neutron. . . . .	34
3.5	Estados de una imagen de Glance. . . . .	35
3.7	Heat. . . . .	37
3.8	Esquema de Heat. . . . .	37
3.9	Sahara: Big Data en OpenStack . . . . .	38
3.10	Esquema de OpenStack Sahara. . . . .	39
3.11	Generaciones de OpenStack y sus servicios. funte: Wikipedia. . . . .	40
3.12	Mapa conceptual de servicios de OpenStack. funte: <a href="https://openstack.org">openstack.org</a> . . . . .	41
3.13	RDO es la distribución rpm de OpenStack (RPM Distribution of OpenStack). . . . .	41
3.14	Despliegue conceptual del Overcloud desde el Undercloud. . . . .	42
6.1	In this . . . . .	47
7.1	In this . . . . .	49
7.2	The. . . . .	49

7.3 Illu . . . . .	49
7.4 All . . . . .	49

---

## Índice de cuadros

---

2.1 Especificación de los PCs prestados por el I.E.S. Castillo Puche y mi ordenador personal. . . . .	26
5.1 Especificaciones de los valores de red en los dispositivos del proyecto. . . . .	45





# 1. Introducción

## 1.1 Motivación

Uno de los grandes retos, sin duda, a los que se enfrentan en la actualidad los profesionales de las tecnologías de la información, es la dificultad de implementar entornos de producción de forma rápida y efectiva, adquiriendo equipos o infraestructuras, ajustando presupuestos de material y personal, resolviendo problemas de conectividad o incompatibilidad de software o hardware y lo más importante, quitando tiempo al desarrollo en sí del núcleo del proyecto. En adición, con la generación tan inmensa de datos que se generan cada día y la necesidad que existe por analizarla a tiempo real con la ambición de dar una mejor experiencia al usuario, la dificultad de implementar plataformas capaces de utilizar la ciencia de datos [7], el Machine Learning o la Replicación y almacenamiento masivo en lagos de datos requiere de una gran infraestructura que, físicamente, es difícil implementar.

La necesidad de abordar estos problemas hizo surgir el concepto de nube, la capacidad de separar la infraestructura y configuración "física" de la arquitectura conceptual, dando alta disponibilidad multiplataforma y dando a los desarrolladores una gran ventaja: dejar de preocuparse tanto por la implementación física y utilizando el tiempo para el núcleo del proyecto. Sin duda la Virtualización es uno de los pilares de este desarrollo, ofreciendo abstracción y orquestación de servicios, haciendo que con unas pocas líneas de código sea posible implementar cientos o miles de máquinas virtuales, con sus propias redes virtualizadas (switches, routers, IP's flotantes, SO, librerías preinstaladas, servicios de servidor web, File Transfer Protocol (FTP), Dinamic Host Configuration Protocol (DHCP) y un largo etcétera), consiguiendo en cuestión de minutos tener una arquitectura base para cualquier proyecto.

Hacía ya un tiempo que me había interesado en el tema de la implementación de nubes, infraestructura de clusters, bare metal, contenedores y los distintos conceptos de as a Service (aaS), y sin duda mis conocimientos matemáticos junto con los informáticos me llevaron inevitablemente a introducirme en el mundo del Big Data, Ciencia de datos, Machine Learning y Business Intelligence. Por ello tomé la decisión de sumergirme de cabeza en el mundo de la implementación de arquitectura de nubes y Virtualización, eligiendo este tema inmenso, arduo y sin duda maravilloso.

## 1.2 La computación en la nube

Según [14] la Compuación en la Nube (Cloud computing) es el desarrollo y la utilización de capacidad de procesamiento computacional basado en Internet, denominado habitualmente por "la nube.<sup>es</sup>, dicho de otra forma, tomar los equipos que están conectados a una red (ordenadores, servidores, smartphones, dispositivos Internet of Things (IoT), etc.), en este caso la mayor de todas Internet", y hacer que trabajen todos juntos como un mismo organismo que, independientemente de sus partes, trabaja dinámicamente y desde cualquier lugar.

Según IBM [31], la computación en la nube proporciona una serie de ventajas:

- *Flexibilidad:* Los usuarios pueden escalar los servicios para ajustarlos a sus necesidades, personalizar las aplicaciones y acceder a los servicios cloud desde cualquier sitio, con una conexión a Internet.
  - *Escalabilidad:* La infraestructura cloud se escala on demand para dar soporte a las cargas de trabajo fluctuantes.
  - *Opciones de almacenamiento:* Los usuarios pueden elegir soluciones de almacenamiento público, privado o híbrido, en función de las necesidades de seguridad y otras consideraciones.
  - *Opciones de control:* Las organizaciones pueden determinar su nivel de control con opciones como servicio, las cuales incluyen Software as a Service (SaaS), Platform as a Service (PaaS) e Infrastructure (IaaS).
  - *Selección de herramientas:* Los usuarios pueden seleccionar entre un menú de herramientas y características predefinidas para crear una solución que se adapte a sus necesidades específicas.
  - *Características de seguridad:* Cloud privado virtual, cifrado y claves de API ayudan a proteger los datos.
- *Eficiencia:* Los usuarios empresariales pueden lanzar sus aplicaciones al mercado rápidamente, sin preocuparse por los costes de infraestructura subyacentes o el mantenimiento.
  - *Accesibilidad:* Las aplicaciones basadas en cloud y los datos son accesibles desde prácticamente cualquier dispositivo conectado a Internet.
  - *Acelerar la comercialización:* El desarrollo en cloud permite a los usuarios lanzar sus aplicaciones al mercado rápidamente.
  - *Seguridad de datos:* Los fallos de hardware no provocan la pérdida de datos gracias a las copias de seguridad en red.
  - *Ahorro en el equipo:* Cloud computing utiliza recursos remotos, de esta manera las organizaciones se ahorran el coste de los servidores y otros equipos.
  - *Estructura de pago:* Una estructura de pago “utility” significa que los usuarios solo pagan por los recursos que utilizan.

- *Valor estratégico:* Los servicios cloud dan a las empresas una ventaja competitiva a través de la tecnología más innovadora disponible.
  - *Trabajo optimizado:* Los proveedores de servicios cloud (CSP) gestionan la infraestructura subyacente, permitiendo a las organizaciones centrarse en el desarrollo de aplicaciones y otras prioridades.
  - *Actualizaciones regulares:* Los proveedores de servicios actualizan las soluciones con regularidad para ofrecer a los usuarios la tecnología más actualizada.
  - *Colaboración:* El acceso mundial significa que los equipos pueden colaborar desde muy diversas ubicaciones.
  - *Ventaja competitiva:* Las organizaciones se mueven con mayor agilidad que la competencia, que tiene que destinar los recursos de TI a la gestión de la infraestructura.

Debido a la gran cantidad de opciones que se pueden construir con el cloud computing, es habitual clasificar las nubes según los servicios o el dominio para el que tienen fin, esto lo vemos en el apartado ??

## 1.3 Tipos de nube

Los distintos modelos de cloud computing pueden dividirse en dos subgrupos bien por tipo de servicio o por otra parte por disponibilidad y privacidad.

### 1.3.1 Tipos de servicio

Los proveedores de Cloud ofrecen una serie de servicios adicionales que ajusta la oferta a las necesidades del cliente. En función de lo completa que sea la solución que se ofrece tenemos Infraestructura como Servicio, Plataforma como Servicio o Software como Servicio:

- *SaaS:* Podemos hablar de este sistema cuando el usuario se encuentra con todas las herramientas para implementar todos los procesos necesarios para la empresa.
- *IaaS:* No existe valor añadido, el proveedor ofrece capacidad de almacenamiento y proceso en bruto , siendo el usuario el que construye las aplicaciones que necesita desde cero.
- *PaaS:* En este caso, se ofrece herramientas y utilidades para facilitar la construcción de aplicaciones , como puede ser base de datos o entornos de programación.

según [30].

También se suelen clasificar por:

- *Nube Pública:* Un servicio de Nube Pública es cuando el proveedor proporciona sus recursos de forma abierta a todas las entidades que lo deseen, desde particulares a grandes corporaciones. Este tipo de servicios son los que ofrecen Amazon, Azure de Microsoft o Google Engine.
- *Nube Privada:* En la otra parte de la balanza se encuentra el Cloud Privado, que es una

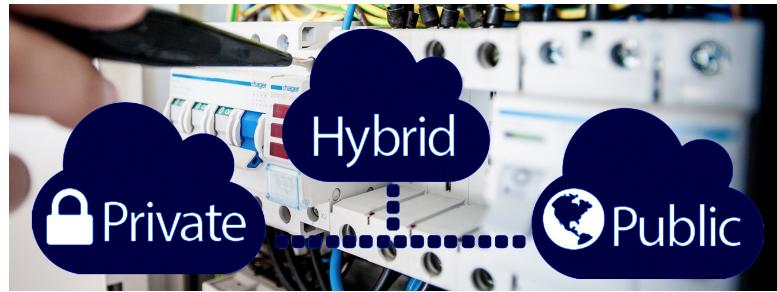


Figura 1.1: Cloud Computing.

cuento el proveedor realiza la implementación y administración del sistema para la entidad que forma parte de ella. Las entidades que optan por este tipo de sistemas son aquellas que tienen un alto nivel de complejidad y necesitan centralizar sus recursos, como pueden ser grandes corporaciones o administraciones públicas. El sistema que más está destacando para este tipo de servicios es Openstack, solución OpenSource.

- *Nube Híbrida:* Como su propio nombre indica esta solución esta compuesta por las dos anteriores , donde una parte de los servicios y la información se ofrece de manera pública y otra de manera privada. Este tipo de soluciones tienen mucho potencial, ya que permiten hacer crecer tu sistema contratando a terceros lo que vayas necesitando. Este tipo de servicios también se pueden realizar con Openstack.

#### 1.4 Objetivo del proyecto

El objetivo de este proyecto es el realizar la implantación de una arquitectura de Nube, concretamente de OpenStack, que es sin duda kit de soluciones de cluod computing más expandido a nivel mundial. Dicha implementación se decide realizarla a través del sistema operativo CentOS en su versión 7, perteneciente a la familia de distribuciones de Red Hat Enterprise Linux.

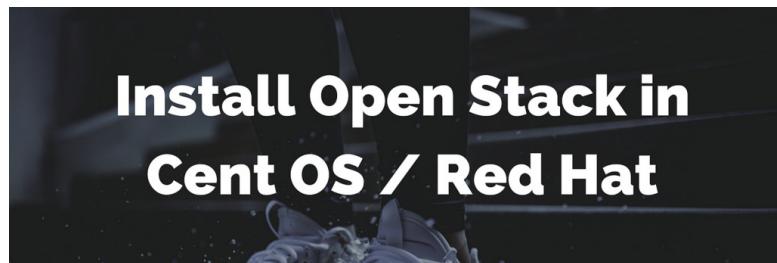


Figura 1.2: Objetivo: Implementar OpenStack sobre CentOS.

La instalación y configuración debe ser completamente funcional y permitir el levantamiento de instancias, redes y todos los recursos básicos de openstack.

Como segundo objetivo, limitado por el tiempo, se quiere implementar el servicio de análisis de datos de openstack SAHARA, para su utilización en desarrollo de cluster de Hadoop.

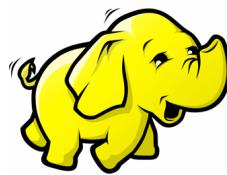


Figura 1.3: Hadoop y su simpático logo, permite a las aplicaciones trabajar con miles de nodos y petabytes de datos.<http://hadoop.apache.org/>

## 1.5 Porqué OpenStack



Figura 1.4: OpenStack, <https://www.openstack.org/>

Según Rackspace (uno de los fundadores del proyecto) en [27], OpenStack es:

*®OpenStack es una plataforma de software de código abierto para nubes privadas y públicas. OpenStack, una plataforma que ofrece IaaS, permite a las empresas agregar servidores, y componentes de redes y de almacenamiento de manera fácil y eficiente a su nube.*

Elegí OpenStack precisamente por ser de código abierto, lo que ha convertido al proyecto en uno de los que mayor desarrollo tienen a nivel mundial, con una inmensa cantidad de contribuidores, tanto particulares como empresas u organismos públicos. La documentación sobre el proyecto es extensa, y tal es la capacidad de desarrollo que sale una versión nueva cada seis meses. Puedes implementarlo sobre casi cualquier sistema con las distintas distribuciones y soportes que existen, o directamente en Bare metal a través de herramientas de orquestación e implementación multinodo.

### 1.5.1 Historia y alcance de OpenStack

Rackspace y la NASA lanzaron juntos OpenStack como un proyecto de software de nube de código abierto en 2010. Hoy en día, OpenStack es administrado por la organización sin fines de lucro OpenStack Foundation. Rackspace es un socio platino de la organización y está muy involucrado en la comunidad OpenStack.

OpenStack es la plataforma de cómputo en la nube predilecta de las mayores empresas del mundo. Esta plataforma impulsa el sitio de comercio electrónico de Walmart, que recibe más de 80 millones de visitas cada mes. CERN, una organización europea de investigación, también utiliza OpenStack para su nube privada: la mayor nube de OpenStack del mundo, que cuenta con casi 200,000 núcleos.



Figura 1.5: Logos de NASA y Rackspace.



## 2. Punto de partida y planificación inicial



### 2.1 Situación de partida

El comienzo del proyecto fue un tanto desbordante. Hay una gran cantidad de documentación y recursos sobre OpenStack en la nube, como no podía ser de otra forma. Buscar entre tanta cantidad de información y aclararme las ideas sobre qué era OpenStanck implicó muchas horas de estudio, leyendo documentación en inglés y aprendiendo nuevos conceptos. En definitiva, OpenStack se presentó ante mí como todo un universo lleno de posibilidades.

Una vez hube investigado lo suficiente - máquinas necesarias, redes, presupuesto, recursos de todo tipo - comenté la disponibilidad de recursos con los que podía contar por parte del centro educativo I.E.S. José Luís Castillo Puche, pensando en hacer un desarrollo multinodo necesitaba por lo menos un par de nodos de computación (ordenadores), y algún dispositivo de interconexión. Gracias a la ayuda de los profesores y de mi tutor conseguí un par de equipos y un router, con los que hacer las pruebas de desarrollo necesarias.

Mis recursos personales, por otra parte, no resultaban significativos para el desarrollo de este proyecto, pudiendo aportar un viejo portátil como nodo de la nube.

De entre todos los sistemas que valoré utilizar (Ubuntu, debian, fedora, windows, mac, Archlinux..) al final me decidí a utilizar CentOS Linux en su versión 7, puesto que tiene un muy buen soporte de OpenStack a través de RDO.

### 2.2 Recursos disponibles

#### 2.2.1 Recursos de Hardware

Dentro de los recursos de hardware que tenía a mi disposición contaba con dos equipos del instituto con la disposición que se encuentra en 2.1 junto con n viejo portátil.

También se me procuró un router Linksys que se utilizará durante el proyecto como swicht y como puerta de enlace.

	<b>PC-1</b>	<b>PC-2</b>	<b>portátil</b>
<b>CPU</b>	Intel-i5	Intel-i5	Intel Core 2
<b>RAM</b>	12Gb	4Gb	2 Gb
<b>DISCO</b>	1TB	1TB	120 GB

Cuadro 2.1: Especificación de los PCs prestados por el I.E.S. Castillo Puche y mi ordenador personal.

Para las conexiones a través de cable Ethernet, utilicé un rollo de cable de cat.6 S/FTP, y crimpé personalmente todos los cables de conexión de la red.

### 2.2.2 Recursos de software

Este proyecto está basado en OpenSource, por tanto todos los recursos de software a utilizar son aquellos que estén disponibles para su libre distribución.

Tanto OpenStack, CentOS, RDO y demás software utilizado para el proyecto está sacada directamente de fuentes de libre distribución.

También se cuenta como recurso Amazon Web Services (AWS) en donde tengo un dominio y suelo utilizar instancias ec2. Debido a que es un coste adicional, se intentará no utilizar mucho, solo en pruebas puntuales si fuera necesario.

### 2.2.3 Subvenciones y ayudas

Dentro de este apartado no puedo especificar mucho más allá de la ayuda aportada por el centro. A pesar de buscar ayudas o subvenciones para el proyecto en plataformas como el Instituto de Fomento de la Región de Murcia, distintos ministerios y/o Consejerías, la gran mayoría no encajaba con las necesidades del proyecto, estando entre los criterios de concesión, criterios tales como: "ser autónomo"; "ser desempleado"; ñvestigardorü otras en las que no encajaba mi perfil. Otras por el contrario, tenían un calendario no compatible con el desarrollo de este proyecto.

## 2.3 Entorno de desarrollo

Durante el desarrollo del proyecto he utilizado una serie de herramientas transversales para llevar a cabo cada una de las partes. Se especifican a continuación:

### 2.3.1 Oracle VirtualBox



Figura 2.1: Oracle VirtualBox, <https://www.virtualbox.org/>

Oracle VM VirtualBox es un software de virtualización para arquitecturas x86/amd64. Actualmente es desarrollado por Oracle Corporation como parte de su familia de productos de virtualización. Por medio de esta aplicación es posible instalar sistemas operativos adicionales, conocidos como «sistemas invitados», dentro de otro sistema operativo «anfitrión», cada uno con su propio ambiente virtual. Entre los sistemas operativos soportados (en modo anfitrión) se encuentran GNU/Linux, Mac OS X, OS/2 Warp, Microsoft Windows, y Solaris/OpenSolaris, y dentro de ellos es posible virtualizar los sistemas operativos FreeBSD, GNU/Linux, OpenBSD, OS/2 Warp, Windows, Solaris, MS-DOS y muchos otros.

He utilizado este sistema para simular el desarrollo del proyecto, creando máquinas virtuales sobre un host. Requiere demasiados recursos como para tratarlo fuera de simples pruebas.



Otra alternativa podría ser VMware, QEMU, Xen o PROXMOX

### 2.3.2 CentOS 7



Figura 2.2: The CentOS Project Linux, <https://www.centos.org/>

CentOS (Community ENTerprise Operating System) es una bifurcación a nivel binario de la distribución Linux Red Hat Enterprise Linux RHEL, compilado por voluntarios a partir del código fuente publicado por Red Hat, siendo la principal diferencia con este la eliminación de todas las referencias a las marcas y logos propiedad de Red Hat.

Es un sistema operativo de código abierto, basado en la distribución Red Hat Enterprise Linux, operándose de manera similar, y cuyo objetivo es ofrecer al usuario un software de clase empresarial "gratuito". Se define como robusto, estable y fácil de instalar y utilizar. Desde la versión 5, cada lanzamiento recibe soporte durante diez años, por lo que la actual versión 7 recibirá actualizaciones de seguridad hasta el 30 de junio de 2024.

Red Hat Enterprise Linux se compone de software libre y código abierto, pero su compilación se distribuye a través de medios (CD-ROM o DVD-ROM) solamente a suscriptores de pago. Red Hat libera todo el código fuente del producto de forma pública bajo los términos de la Licencia pública general de GNU y otras licencias. Los desarrolladores de CentOS usan ese código fuente

para crear un producto final que es compatible a nivel binario y funcional con Red Hat Enterprise Linux y está libremente disponible para ser descargado, utilizado y redistribuido libremente por el público, pero sin que sea mantenido directamente por Red Hat. CentOS es una de las distribuciones consideradas clones de Red Hat Enterprise Linux [6].

### 2.3.3 Python



Figura 2.3: Python, <https://www.python.org/>

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.

Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License, que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores.[26]

### 2.3.4 Git y Github



Figura 2.4: Git y Github, <https://github.com/>

- *Git:*

Git (pronunciado "guit") es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.

Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o front end como Cogito o StGIT. Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena. Hay algunos proyectos de mucha relevancia que ya usan Git, en particular, el grupo de programación del núcleo Linux.

El mantenimiento del software Git está actualmente (2009) supervisado por Junio Hamano, quien recibe contribuciones al código de alrededor

de 280 programadores. En cuanto a derechos de autor Git es un software libre distribuible bajo los términos de la versión 2 de la Licencia Pública General de GNU. [19]

- **GitHub:**

GitHub es una forja (plataforma de desarrollo colaborativo) para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de computadora. El software que opera GitHub fue escrito en Ruby on Rails. Desde enero de 2010, GitHub opera bajo el nombre de GitHub, Inc. Anteriormente era conocida como Logical Awesome LLC. El código de los proyectos alojados en GitHub se almacena típicamente de forma pública, aunque utilizando una cuenta de pago, también permite hospedar repositorios privados.

El 4 de junio de 2018, Microsoft compró GitHub por un monto de 7.500 millones de dólares, lo que hizo que muchos desarrolladores empezaran a migrar al competidor de código abierto GitLab. [20]

### 2.3.5 IDEs y editores

A la hora de manejar códigos de todo tipo en distintos lenguajes de programación (Shell Script, python, YAML, txt, etc.) me gusta utilizar el editor Atom. A pesar de ello cuando trabajas en terminal los recursos gráficos pasan a segundo plano y llegas a utilizar otros editores de terminal:

- **nano:**

Este es un editor muy sencillo, bastante fácil de manejar si eres novato en terminal, pero con la práctica puede volverse una herramienta fundamental.

- **Vim:**

Vim sin duda es una verdadera "máquina" de teclear código en terminal. Tiene una curva de aprendizaje que no deja indiferente a nadie, pero su capacidad para adjuntarle plugins o sus capacidades avanzadas de manejo de código hacen a este editor una verdadera navaja suiza.

### 2.3.6 Control de proyectos

Todo proyecto debe llevar un control de las actividades que se realizan, bien por tener en cuenta todo lo que queda por hacer o simplemente por logística. Personalmente creo que trabajar con herramientas de control de proyectos hace que el trabajo en equipo sea mucho más fluido y que en todo momento sea mucho más sencillo determinar el estado actual de desarrollo.



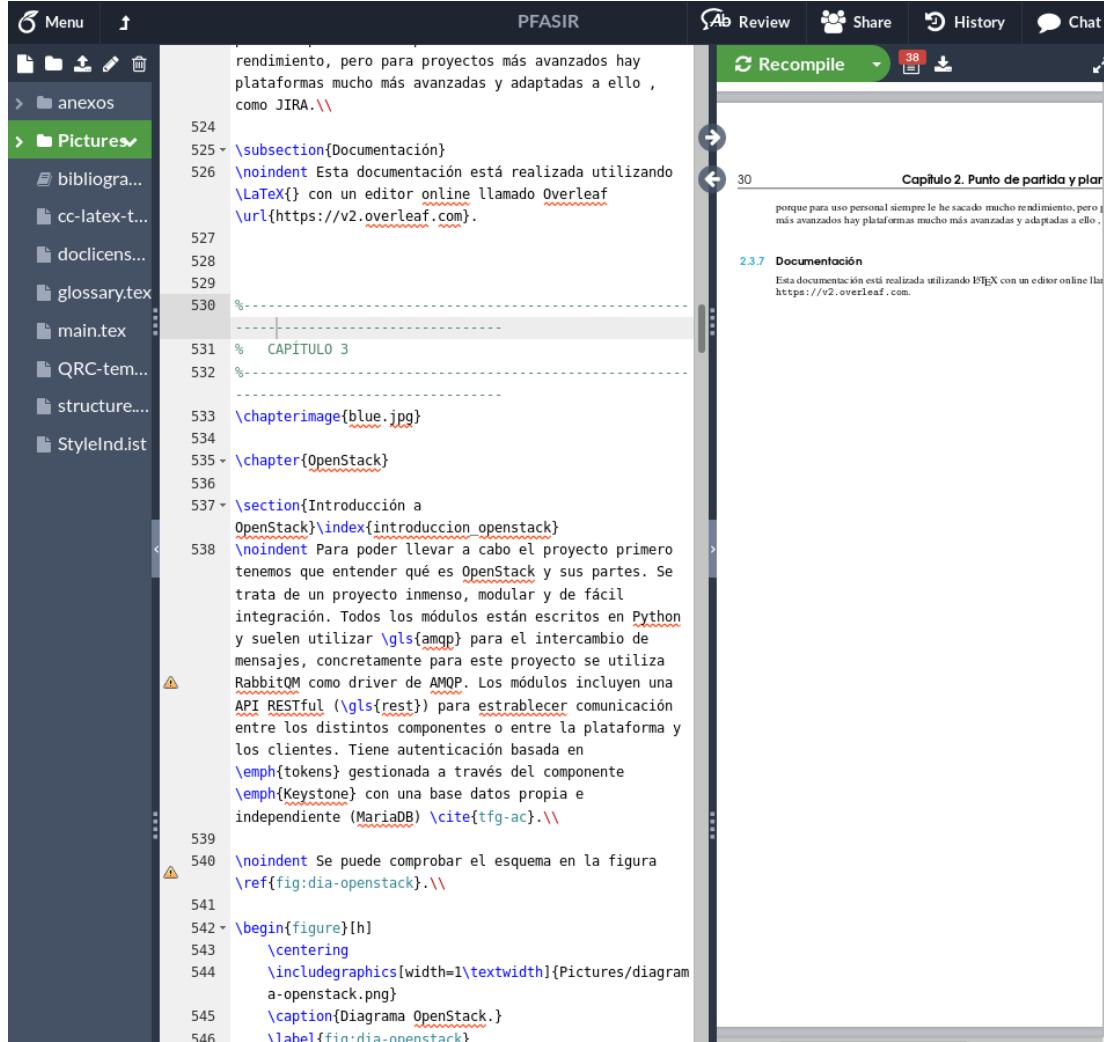
Figura 2.5: Trello, <https://trello.com/>

Particularmente en este proyecto he utilizado Trello como herramienta de control para el proyecto. Es muy sencillo crear tareas y agregar comentarios, pasando de un estado a otro de una tarea con simplemente arrastrar en un panel gráfico. He utilizado esta herramienta

porque para uso personal siempre le he sacado mucho rendimiento, pero para proyectos más avanzados hay plataformas mucho más avanzadas y adaptadas a ello , como JIRA.

### 2.3.7 Documentación

Esta documentación está realizada utilizando L<sup>A</sup>T<sub>E</sub>X con un editor online llamado Overleaf <https://v2.overleaf.com> tal como se puede observar en la figura 2.6.



PFASIR

Review Share History Chat

Recompile 38

30 Capítulo 2. Punto de partida y planificación inicial

porque para uso personal siempre le he sacado mucho rendimiento, pero para proyectos más avanzados hay plataformas mucho más avanzadas y adaptadas a ello , como JIRA.\\"

\subsection{Documentación}

\noindent Esta documentación está realizada utilizando \LaTeX{} con un editor online llamado Overleaf <https://v2.overleaf.com>.

\chapterimage{blue.jpg}

\chapter{OpenStack}

\section{Introducción a OpenStack}\index{introducción openstack}

\noindent Para poder llevar a cabo el proyecto primero tenemos que entender qué es OpenStack y sus partes. Se trata de un proyecto inmenso, modular y de fácil integración. Todos los módulos están escritos en Python y suelen utilizar \gls{amqp} para el intercambio de mensajes, concretamente para este proyecto se utiliza RabbitMQ como driver de AMQP. Los módulos incluyen una API RESTful (\gls{rest}) para establecer comunicación entre los distintos componentes o entre la plataforma y los clientes. Tiene autenticación basada en \emph{tokens} gestionada a través del componente \emph{Keystone} con una base datos propia e independiente (MySQL) \cite{tfg-ac}.\\"

\noindent Se puede comprobar el esquema en la figura \ref{fig:dia-openstack}.\\"

\begin{figure}[h]

\centering

\includegraphics[width=1\textwidth]{Pictures/diagrama-a-openstack.png}

\caption{Diagrama OpenStack.}

\label{fig:dia-openstack}

Figura 2.6: Captura de pantalla del código L<sup>A</sup>T<sub>E</sub>X en Overleaf de este documento.

# 3. OpenStack

## 3.1 Introducción a OpenStack

Para poder llevar a cabo el proyecto primero tenemos que entender qué es OpenStack y sus partes. Se trata de un proyecto inmenso, modular y de fácil integración. Todos los módulos están escritos en Python y suelen utilizar Advanced Message Queuing Protocol (AMQP) para el intercambio de mensajes, concretamente para este proyecto se utiliza RabbitMQ como driver de AMQP. Los módulos incluyen una API RESTful (Representation State Transfer (REST)) para establecer comunicación entre los distintos componentes o entre la plataforma y los clientes. Tiene autenticación basada en *tokens* gestionada a través del componente *Keystone* con una base datos propia e independiente (MariaDB) [14].

Se puede comprobar el esquema en la figura 3.1.

## 3.2 Componentes básicos de OpenStack

### 3.2.1 Servicio de identidad (Keystone)

KeyStone es el servicio de identidad que utiliza OpenStack para la autenticación, dicho sistema puede ser integrado con ldap, además puede ser usado a través de nombre de usuario y contraseña estándar, sistemas basados en tokens e inicios de sesión. Dispone de una API de identidad (actualmente la versión 3.0) la cual puede ser usada por herramientas de terceros para determinar la accesibilidad de los recursos. KeyStone nos va a permitir la autenticación y autorización entre los servicios, siendo instalado en el nodo Controlador, el cual suele usar como motor de base de datos tanto MySQL como MariaDB. [23] 3.2

Para entender cómo funciona KeyStone o Servicio de identidad debemos tener en cuenta los siguientes componentes:

- **Usuario:**

Pueden ser personas, sistemas o servicios que usen la nube de OpenStack, KeyStone será el encargado de validar las peticiones de éste. Los usuarios disponen de un login.

- **Credenciales:**

Es el dato que va a usar el usuario para su autenticación, puede ser un login y contraseña, un token, una clave de la API.

- **Autenticación:**

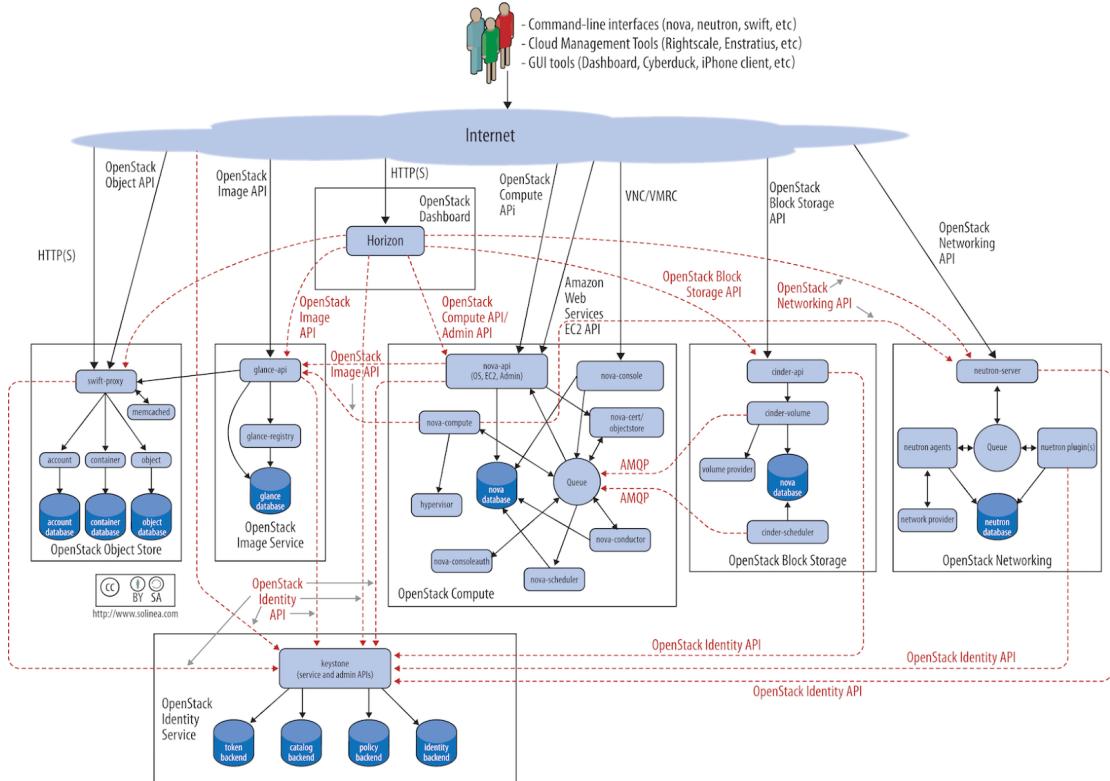


Figura 3.1: Diagrama OpenStack.



Figura 3.2: OpenStack Keystone

Es el acto de confirmar la identidad del usuario usando las credenciales provistas. Una vez validada la autenticación se provee de un token para ser usado en las peticiones del usuario, de esta forma no necesita autenticarse de nuevo.

■ **Token:**

Es un bit de texto usado para acceder a los recursos, es generado cuando se valida la autenticación, cada token tiene un alcance el cual nos permite saber a qué recursos puede acceder un usuario.

■ **Tenant:**

Es el contenedor usado para agrupar y aislar los recursos. Los tenants pueden ser clientes, organizaciones, cuentas o proyectos.

■ **Servicio:**

Es cada uno de los servicios que conforman OpenStack, como pueden ser Computo (Nova), Almacenamiento (Swift) o Servicio de imágenes (Glance), nos va a proveer de los endpoints o URL para que los usuarios puedan acceder a los recursos.

■ **Endpoint:**

Es el direccionamiento IP o URL para acceder a los servicios.

- **Roles:**

Para especificar o limitar las operaciones de usuarios KeyStone usa roles, los cuales contienen un grupo de privilegios para determinar el alcance del usuario.

### 3.2.2 Servicio de cómputo (Nova)

Nova es el componente principal del IaaS (infraestructura como servicio) y es el encargado de administrar los pools de recursos que tenemos disponibles, hay que tener en cuenta que Nova NO es un hypervisor, sino el gestor de recursos de hipervisores tales como: Xen, KVM o VMware. []

La arquitectura del nodo de cómputo esta compuesta de los siguientes servicios:

- **Servidor de API:**

Nos va a permitir la comunicación de los hipervisores con nuestro entorno, gracias a él los distintos fabricantes pueden integrar sus productos de forma eficaz.

- **Colas de mensajes (rabbit queues):**

Se encarga de la comunicación entre componentes (ya sean nodos de cómputos, componentes de red o API de fabricantes) a la hora de provisionar recursos.

- **Worker:**

Es el administrador de instancias en los hosts que nos va a permitir, crear, terminar, reiniciar instancias, añadir o quitar volúmenes.

- **Controlador de red:**

Podremos provisionar recursos de networking, como por ejemplo IP, vLAN o routers.

En la imagen 3.3 podéis observar como se realiza el flujo de peticiones:

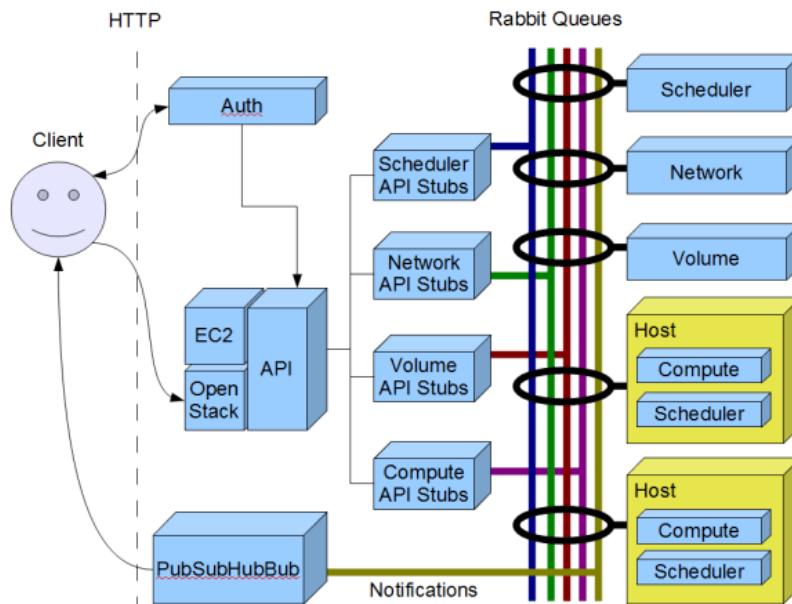


Figura 3.3: Flujo de peticiones de RabbitQM.

### 3.2.3 Servicio de red (Neutron)

El componente de administración de red en OpenStack es uno de los más complejos debido a la cantidad de posibilidades de configuración que tenemos. Neutron nos permite hacer uso de su API a través de plugins y agentes capaces de conectar/desconectar puertos, provisionar redes y subredes, etc, todo ello vinculado al proveedor que vayamos a usar en nuestra nube, como por ejemplo switches físicos o virtuales de Cisco, productos de NEC OpenFlow, Open vSwitch, Linux bridging, Ryu Network Operating System o VMware NSX.

Los tres principales componentes de neutron son:

- **Neutron L3 agent.**
- **DHCP agent.**
- **Neutron plugin agent.**

En el esquema 3.4 podemos ver de forma representativa la comunicación que se realiza entre el servicio de administración de red y los nodos de una arquitectura de OpenStack. Hay que tener en cuenta que en el esquema anterior, se usa un nodo independiente para la administración de red.

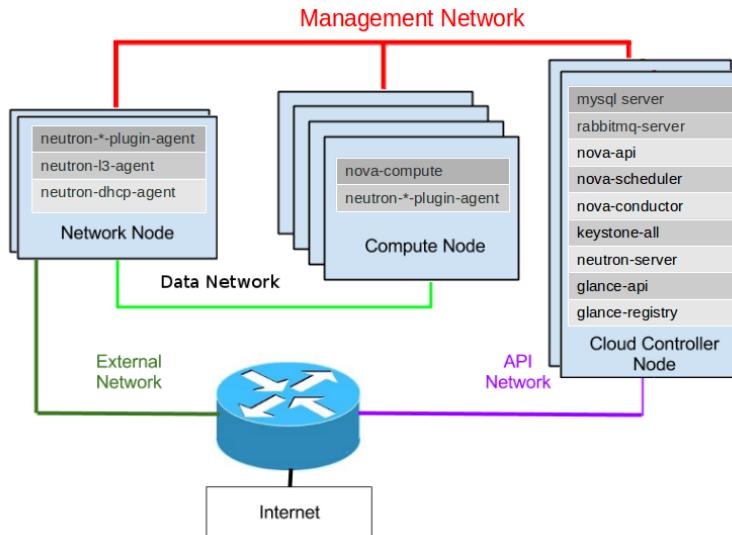


Figura 3.4: Diagrama de Neutron.

### 3.2.4 Servicio de imagen (Glance)

Glance es el servicio de OpenStack el cual permite registrar, descubrir y recuperar imágenes de máquinas virtuales para ser usadas en nuestro entorno de OpenStack. Dichas imágenes son almacenadas tanto en simples como en distribuidos filesystems y son identificadas de la siguiente forma:

```
<Glance Server Location>/v1/images/<ID>
```

Cuando usamos imágenes en Glance, ya sean para crear nuevas o eliminarlas, éstas pueden pasar por los siguientes estados:

- **Encolada - queued:** Quiere decir que hemos registrado una imagen nueva, asignando un identificador pero aún no ha sido subida a Glance.
- **Guardando - saving:** La imagen está siendo subida a Glance.
- **Activa - active:** La imagen está totalmente disponible para ser usada en Glance.

- **Abortada - killed:** Se produce cuando existe un error durante la subida de una imagen y ésta no es legible.
- **Borrada - deleted:** En este estado aún disponemos de la información de la imagen pero no podemos usarla ya que se encuentra eliminada.
- **Pendiente de borrado - pending\_delete:** Estamos a punto de eliminar la imagen pero aún podemos recuperarla.

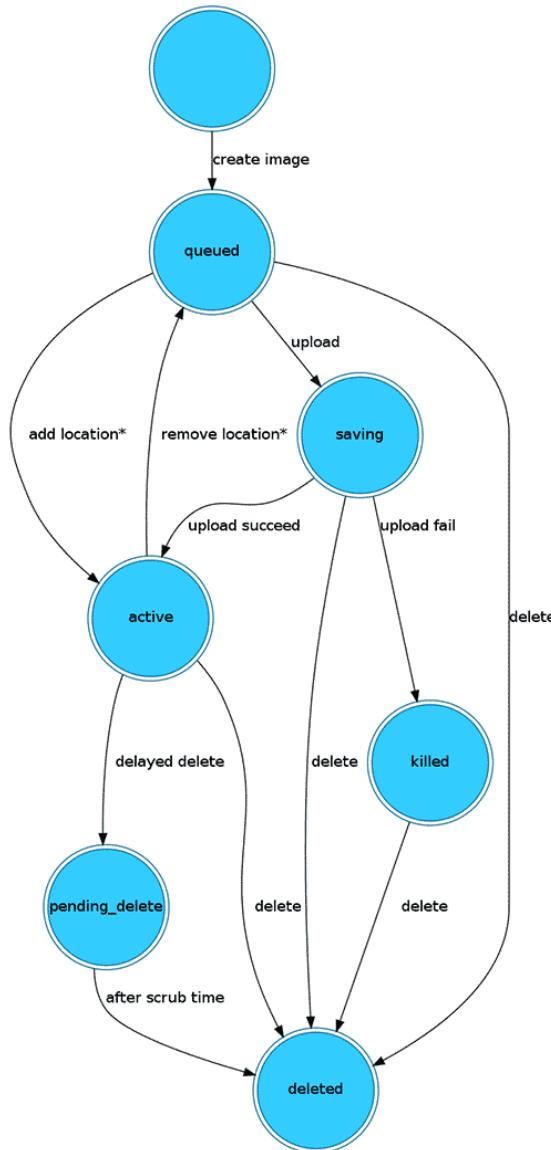


Figura 3.5: Estados de una imagen de Glance.

### 3.2.5 Servicio de almacenamiento de objetos (Swift)

OpenStack Object Storage (Swift) es un sistema de almacenamiento redundante y escalable. Los objetos y los archivos se escriben en varias unidades de disco repartidos por los servidores del centro de datos, con el software OpenStack responsable de asegurar la replicación y la integridad de los datos en el clúster. Agrupaciones de almacenamiento escalar horizontalmente simplemente añadiendo nuevos servidores. En caso de que un servidor o disco duro falla, OpenStack replica su contenido desde otros nodos activos a nuevas ubicaciones en el clúster. Debido a que OpenStack utiliza la lógica del software para asegurar la replicación de datos y la distribución a través de diferentes dispositivos,

discos duros y servidores de bajo costo pueden ser utilizados.

En agosto de 2009, Rackspace comenzó el desarrollo del precursor de OpenStack Object Storage, como un reemplazo completo para el producto Cloud Files. El equipo de desarrollo inicial consistió en nueve desarrolladores. SwiftStack, una compañía de software de almacenamiento de objetos, es actualmente el líder en el desarrollo de Swift.

### 3.2.6 Servicio de almacenamiento en bloque (Cinder)

OpenStack Block Storage (Cinder) proporciona dispositivos de almacenamiento a nivel de bloque persistentes para usar con instancias de OpenStack Compute. El sistema de almacenamiento de bloques gestiona la creación, aplicación y el desprendimiento de los dispositivos de bloque a los servidores. Volúmenes de almacenamiento de bloque se integran plenamente en OpenStack Compute y el Dashboard que permite a los usuarios en la nube gestionar sus propias necesidades de almacenamiento. Además del almacenamiento del servidor local de Linux, puede utilizar las plataformas de almacenamiento incluyendo Ceph, CloudByte, Coraid, EMC (VMAX y VNX), GlusterFS, Hitachi Data Systems, IBM Storage (familia Storwize, controlador de volumen SAN, XIV Storage System, y GPFS), Linux LIO, NetApp, Nexenta, Scality, SolidFire, HP (StoreVirtual y 3PAR StoreServ familias) y almacenamiento puro. El almacenamiento de bloques es apropiado para escenarios donde el rendimiento es sensible, tales como el almacenamiento de base de datos, sistemas de archivos expandibles, o la prestación de un servidor con acceso al almacenamiento a nivel de bloque en bruto. La gestión Snapshot ofrece una potente funcionalidad para realizar copias de seguridad de los datos guardados en volúmenes de almacenamiento en bloque. Las instantáneas pueden restaurarse y utilizarse para crear nuevos volúmenes de almacenamiento en bloque.

### 3.2.7 OpenStack dashboard (Horizon)

Horizon es una interfaz web que permite a los usuarios administrar los servicios que ofrece OpenStack incluyendo Nova, Swift, Keystone.



Figura 3.6: Horizon.

<https://www.dbigcloud.com/cloud-computing/187-openstack-desde-cero-horizon.html>

Al igual que cualquier componente de OpenStack dispone de su propia API, la cual puede ser accedida de forma sencilla, una utilidad que se le puede dar al uso de las API de OpenStack, es la integración con tu propia configuration management database (CMDB) CMDB, cuando usuarios desplieguen instancias, volúmenes, etc. estos datos pueden ser capturados para la integración con la base de datos.

### 3.3 Componentes adicionales de OpenStack

#### 3.3.1 Servicio de orquestación (Heat)

Heat es el orquestador de Openstack, y sirve para poder orquestar lanzamientos de varias cosas a la vez escribiendo una serie de código. Para hacerse una idea: podemos lanzar un proyecto entero de cloud (con sus redes, sus instancias, sus volúmenes, sus imágenes, etc.) con sólo ejecutar una pila (stack), es decir, con “ejecutar” un solo fichero.



Figura 3.7: Heat.

Cómo funciona:

En las plantillas de Heat, se definen los recursos necesarios para lanzar la aplicación, estos son tomados de los servicios disponibles de Openstack. Heat maneja el ciclo de vida de la app y se proveen características como auto-escalabilidad. Dentro de la misma plantilla se puede definir que aplicaciones lanzar directamente con shell scripting o dejar que herramientas de administración, tales como Ansible o Puppet, se encarguen de ello.

Podemos observar un esquema de heat en la figura 3.8.

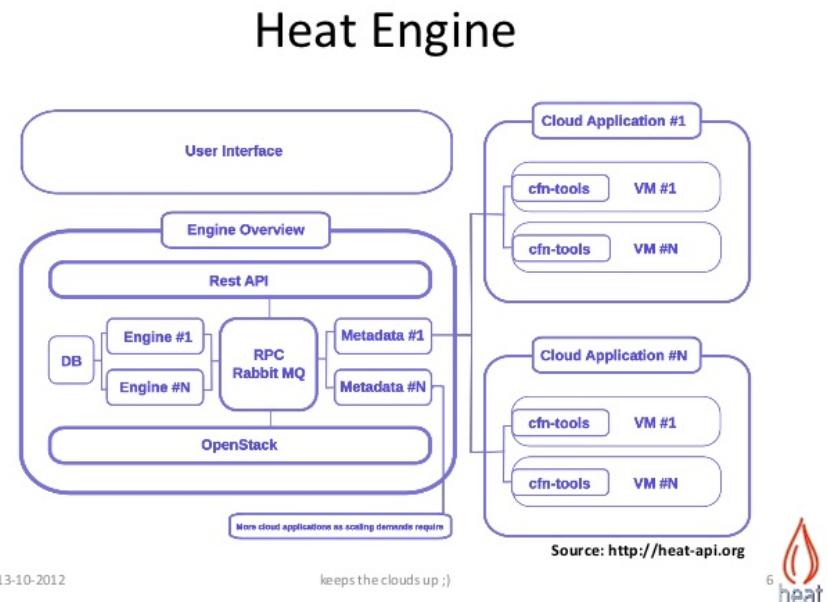


Figura 3.8: Esquema de Heat.

### 3.3.2 Servicio de telemetría (Ceilometer)

Ofrece servicios de telemetría con los que podremos monitorizar el uso de cada usuario en nuestra infraestructura, así como facturar individualmente por dicho uso.

### 3.3.3 Servicio de bases de datos (Trove)

Trove sienta las bases para el aprovisionamiento y gestión de bases de datos SQL y NoSQL. (MySQL, Percona Server, MariaDB, PostgreSQL, MongoDB, Cassandra, Couchbase y Redis), es decir, ejerce como director de bases de datos como servicio (DBaaS). [2]

El uso de DBaaS se sustenta en 5 pilares básicos:

- **Reducción de Costes:** Gracias a la estandarización se reducen los costes asociados a la gestión y mantenimiento de las diferentes plataformas. Proporcionando de forma integrada capacidades adicionales (backup, actualizaciones automáticas y gestión de parches, auto-reparación) que ahondan aún más esta reducción de costes.
- **Agilidad:** mayor velocidad de aprovisionamiento y por tanto menos esperas a la hora de entregar nuevos servicios. Tendremos un nuevo servicio de bbdd a tan solo unos clicks de distancia.
- **Facilitar el acceso a nuevas tecnologías:** NoSQL y NewSQL son el futuro de muchos de los entornos que a día de hoy trabajan con bbdd relacionales. El disponer de un entorno preconfigurado, estable y robusto puede ayudar a la adopción de nuevas herramientas que permitan un mayor crecimiento y una mejor prestación de servicios.
- **Mayor poder de innovación:** muy ligado al punto anterior tanto en cuanto va asociado a nuevas tecnologías y nuevas formas de afrontar la problemática del almacenamiento y sobre todo el acceso a la información.
- **Reducción de tiempos y esfuerzos en los pasos a producción:** la posibilidad de autoescalar y dotar de funcionalidades asociadas a la plataforma sobre la que se está trabajando permitirá una mayor agilidad a la hora de transformar proyectos de desarrollo a entornos productivos, a la vez que escalamos en capacidad y disponibilidad.

Más allá del aprovisionamiento básico, Trove automatiza la gestión del ciclo de vida de las instancias de base de datos. Cuando llega el momento de lanzar una copia de seguridad, este inicia el proceso y almacena la copia de seguridad en el almacén de objetos asociado a nuestra plataforma OpenStack. Un usuario más adelante podría lanzar una nueva instancia de base de datos a partir de esa copia de seguridad realizada previamente. Cuando un usuario desea cambiar el tamaño de la base de datos, ya sea cambiando los recursos de almacenamiento o de cómputo que se le asignen, Trove organiza todo el proceso, incluida la migración de datos.

### 3.3.4 Servicio de procesamiento de datos (Sahara)



Figura 3.9: Sahara: Big Data en OpenStack

OpenStack Sahara es un proyecto integrado en OpenStack cuyo objetivo es hacer elástico Hadoop bajo demanda. Hadoop es prácticamente el estándar cuando hablamos de Big Data, las organizaciones típicamente comienzan montando un cluster. Incluso con las distribuciones comerciales este proceso es complejo.

Sahara está preparado para configurar, autodesplegar y escalar clusters Hadoop sobre OpenStack. Sahara ofrece por tanto capacidades Elastic Data Processing (EDP) sobre Hadoop al estilo de Amazon Elastic MapReduce. Sahara permite:

- Crear clusters desde UI o bien integrar Sahara con tu aplicación vía el API.
- legir entre diversas distribuciones, incluidas Hortonworks Data Platform (HDP) y Cloudera Hadoop Distribution (CHD).
- Soporte de trabajos (jobs) de Apache Spark.
- Configurar HDFS de forma sencilla.

Sahara está integrado con los servicios core OpenStack como Keystone, Glance, Horizon y Nova, además soporta las APIS nativa OpenStack lo que permite a los usuarios provisionar entornos Hadoop desde el Horizon GUI. Referencia [3]

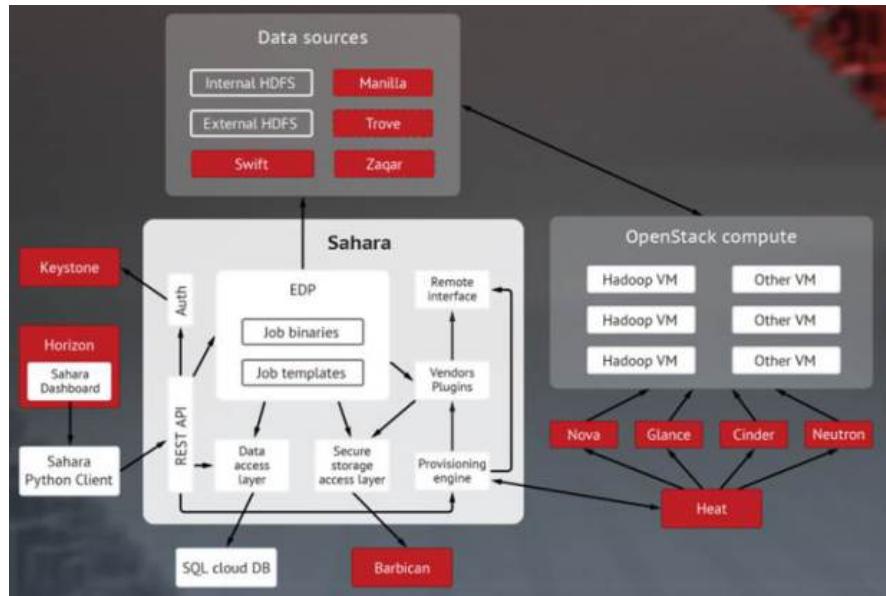


Figura 3.10: Esquema de OpenStack Sahara.

### 3.3.5 Otros componentes

Hay una infinidad más de componentes y módulos de servicio para OpenStack (figura 3.12) que no veremos en este documento, pero hay que destacar que crecen con cada nueva versión como se puede ver en la imagen 3.11.

Para más información sobre módulos se puede consultar: <https://www.openstack.org/software/project-navigator>

Historia de Versiones [editar]

Nombre de la Versión	Fecha de la Versión	Códigos de los componentes incluidos <sup>9</sup>
Austin	21 de octubre de 2010 <sup>10 11</sup>	Nova, Swift
Bexar	3 de febrero de 2011 <sup>12</sup>	Nova, Glance, Swift
Cactus	15 de abril de 2011 <sup>13</sup>	Nova, Glance, Swift
Diablo	22 de septiembre de 2011 <sup>14</sup>	Nova, Glance, Swift
Essex	5 de abril de 2012 <sup>15</sup>	Nova, Glance, Swift, Horizon, Keystone
Folsom	27 de septiembre de 2012 <sup>16</sup>	Nova, Glance, Swift, Horizon, Keystone, Quantum, Cinder
Grizzly	4 de abril de 2013 <sup>17</sup>	Nova, Glance, Swift, Horizon, Keystone, Quantum, Cinder
Havana	17 de octubre de 2013 <sup>18</sup>	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer
Icehouse	17 de abril de 2014 <sup>19</sup>	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer, Trove
Juno	16 de octubre de 2014 <sup>20</sup>	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer, Trove, Sahara
Kilo	30 de abril de 2015 <sup>21</sup>	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer, Trove, Sahara, Ironic
Liberty	16 de octubre de 2015 <sup>22</sup>	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer, Trove, Sahara, Ironic, Zaqar, Manila, Designate, Barbican
Mitaka	7 de abril de 2016	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer, Trove, Sahara, Ironic, Zaqar, Manila, Designate, Barbican
Newton	6 de octubre de 2016	Barbican, Searchlight, Magnum, aodh, cloudkitty, congress, freezer, mistral, monasca-api, monasca-log-api, murano, panko, senlin, solum, tacker, vitrage, watcher
Ocata	22 de febrero de 2017	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer, Trove, Sahara, Ironic, Zaqar, Manila, Designate, Barbican, Searchlight, Magnum, aodh, cloudkitty, congress, freezer, mistral, monasca-api, monasca-log-api, murano, panko, senlin, solum, tacker, vitrage, watcher
Pike	30 de agosto de 2017	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Heat, Ceilometer, Trove, Sahara, Ironic, Zaqar, Manila, Designate, Barbican, Searchlight, Magnum, aodh, cloudkitty, congress, freezer, mistral, monasca-api, monasca-log-api, murano, panko, senlin, solum, tacker, vitrage, Watcher

Figura 3.11: Generaciones de OpenStack y sus servicios. fuente: Wikipedia.

## 3.4 OpenStack en entornos de producción

### 3.4.1 RDO

RDO es una comunidad de personas que usan e implementan OpenStack en CentOS, Fedora y Red Hat Enterprise Linux. Desarrollan documentación para ayudar a comenzar, listas de correo donde puede conectarse con otros usuarios y paquetes compatibles con la comunidad de las versiones de OpenStack más actualizadas disponibles para su descarga. Imagen 3.13

OpenStack se basa fundamentalmente en el sistema operativo y el hipervisor, y ¿qué mejor sistema operativo para construir que el sistema operativo líder de la industria? La comunidad RDO es el sitio comunitario integral para todo lo relacionado con el uso de OpenStack en las plataformas basadas en Red Hat.

El proyecto actual se ha desarrollado siguiendo en gran parte esta herramienta y la documentación relacionada que se puede encontrar en <https://www.rdoproject.org/use/> disponible en inglés.

### 3.4.2 Triple-O

Durante el proceso de diseño de una arquitectura de OpenStack uno de los puntos más importantes a tener en cuenta es el paso hacia la instalación del producto. Para ese paso en la realización del proyecto se utiliza Packstack, de RDO ??, el cual funciona perfectamente para una prueba de concepto y para empezar a conocer el uso y funcionamiento de OpenStack, pero a la hora de trasladar nuestra solución a un entorno más empresarial la primera pregunta que debemos plantearnos es: si yo estoy ofreciendo escalado horizontal a mis clientes de nube, ¿cómo puedo yo escalar horizontalmente mi infraestructura? (En otras palabras: ¿cómo puedo crecer mi capacidad de cómputo, almacenamiento y disponibilidad?).

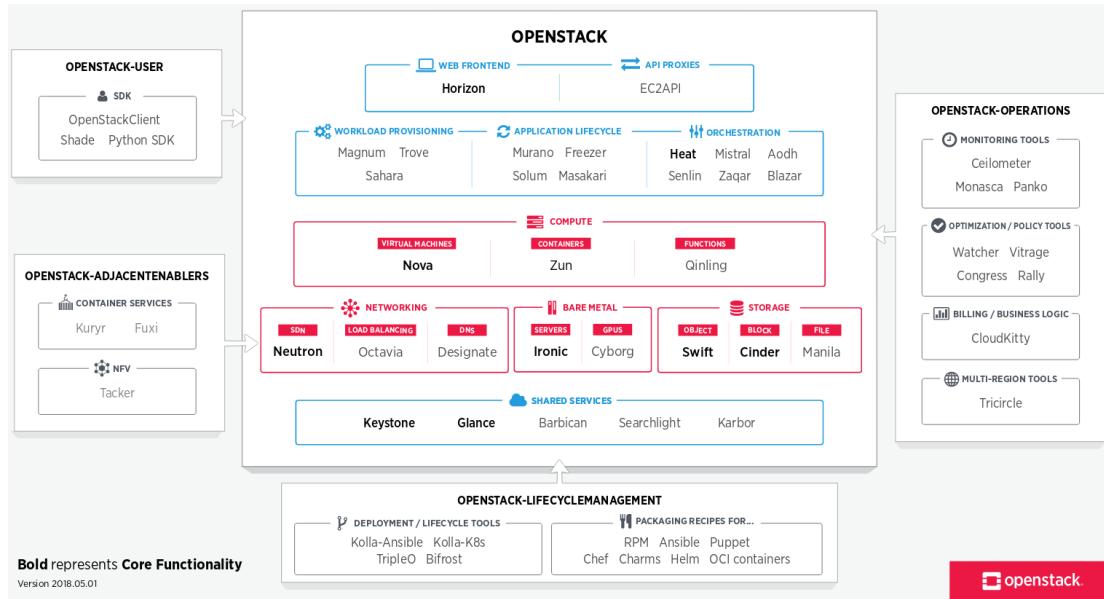


Figura 3.12: Mapa conceptual de servicios de OpenStack. fuente: [openstack.org](http://openstack.org)



Figura 3.13: RDO es la distribución rpm de OpenStack (RPM Distribution of OpenStack).

Para resolver esta cuestión nació el proyecto OpenStack On OpenStack (Triple-O), una plataforma para instalar y administrar OpenStack desde una implementación de OpenStack directamente en baremetal. Es decir, implementar y gestionar una plataforma de producción OpenStack desde otra plataforma OpenStack (más sencilla) que actuará como controlador. El concepto puede resultar confuso al principio.

*"Triple-O es la utilización de una infraestructura OpenStack auto-gestionada, es decir, OpenStack baremetal (nova y cinder) junto con Heat, diskimage-builder más un orquestador de imágenes como Chef o Puppet para instalarlo, mantenerlo y actualizarlo."*

La estrategia utilizada para llevar a producción dichas implementaciones con Triple-O/Director es la siguiente:

- Implementar una máquina física/virtual que se encargara del despliegue de toda nuestra infraestructura de producción (con Packstack, por ejemplo). A este entorno más sencillo se le llama **undercloud**.
- Desplegar nuestro entorno de producción (llamado **Overcloud**) desde el Undercloud.

A este proceso se le denomina *desplegar la Overcloud desde la UnderCloud*. En la figura 3.14 se puede observar. Simple, ¿Cíerto?

### 3.4.3 Otros

OpenStack tiene muchos sabores, por así decirlo, debido a que es Open Souce y cualquiera puede tomar el código fuente del sistema y hacer su propia implementación.

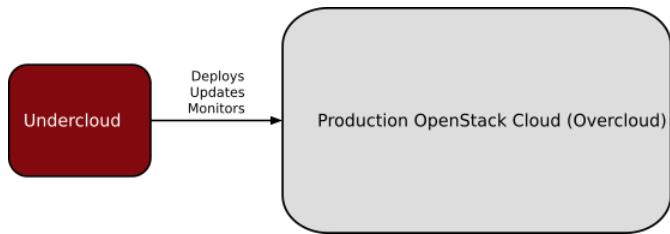
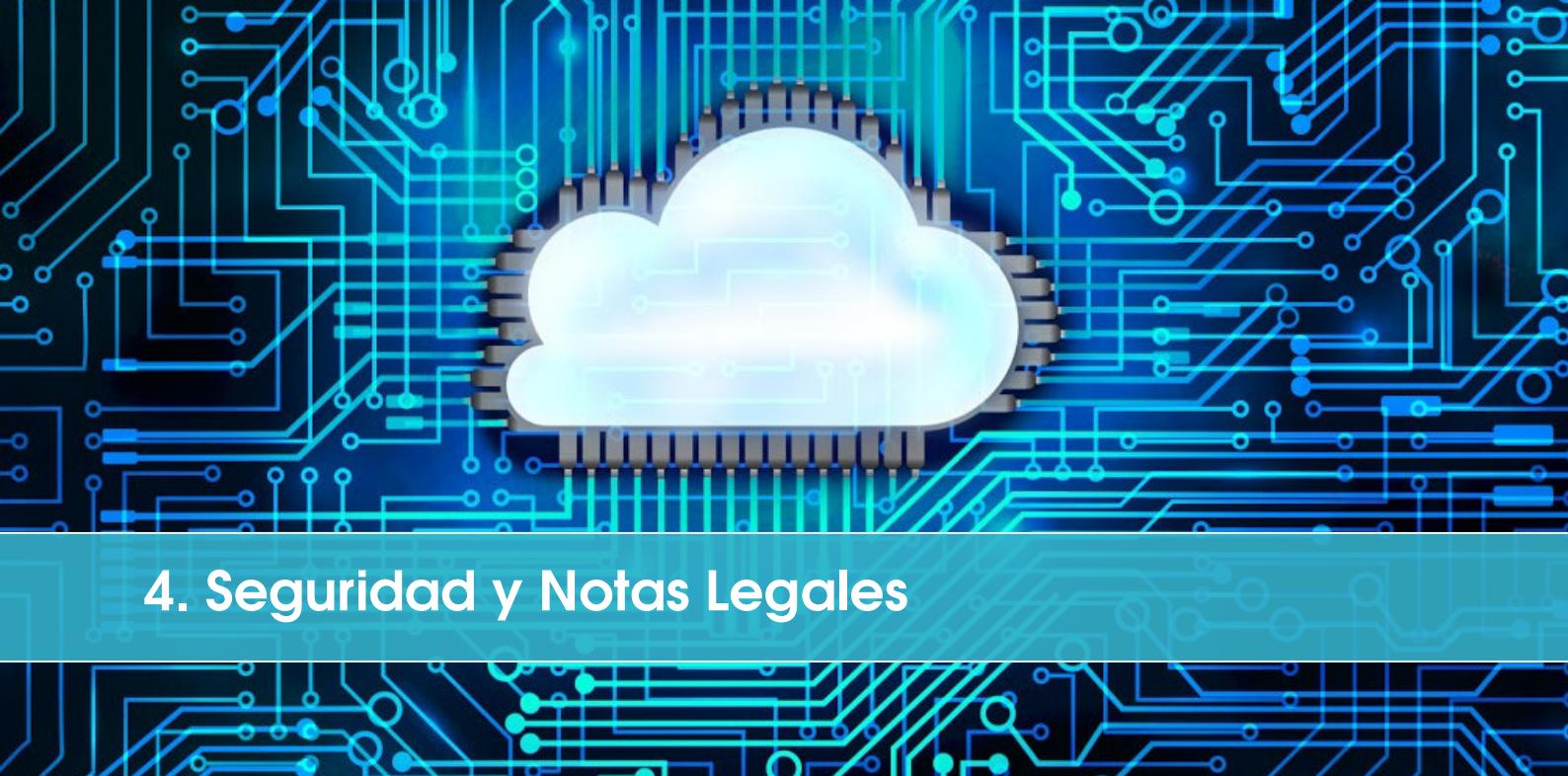


Figura 3.14: Despliegue conceptual del Overcloud desde el Undercloud.

La causalidad lleva por tanto a que surjan distintas distribuciones de OpenStack, está Conjure-up junto con MAAS de Ubuntu, RDO de Red Hat, Mirantis, DevStack, VM-ware OpenStack, distribuciones para Debian y un lista inmensa de implementaciones. Si tienes curiosidad en implementar OpenStack sobre un sistema determinado o simplemente te gustaría conocer las opciones que existen, te sugiero que eches un vistazo al marketplace de openstack desde donde podrás ver multitud de implementaciones: <https://www.openstack.org/marketplace/distros/>



## 4. Seguridad y Notas Legales

La seguridad, el aislamiento y la implementación de políticas de firewall, proxy y de evasión de malware son temas troncales de la administración de OpenStack. La configuración de redes debe ser independiente y las instancias deben poder aislarse efectivamente unas de otras para asegurar la privacidad de los servicios dentro de la infraestructura.

Para este proyecto se han tenido en cuenta las especificaciones básicas de seguridad como la utilización del protocolo de Secure Shell con certificado ssl para la comunicación entre el servidor de despliegue y las máquinas clientes. Asegurar que los puertos del router no permiten las conexiones externas con internet, haciendo que la implementación del proyecto se constituya en una nube privada.

En OpenStack existen todo tipo de entidades como usuarios, grupos, áreas de disponibilidad, claves de seguridad, Roles de permisos y multitud de configuraciones destinadas a la seguridad y la privacidad. Destaco que buscando una implementación hacia la analítica de datos, es fundamental tratar este tema en profundidad en el futuro.

OpenStack implementa muchas de estas reglas de seguridad automáticamente y no permite controlar la configuración desde fuera de los elementos de openstack. Prueba de ello es la gran cantidad de cortafuegos y reglas de firewall con las que estuve varios días buscando configuraciones y no conseguí manualmente abrir los puertos para conseguir enrutar el acceso http para acceder al panel de control desde fuera del router. Seguiré intentándolo en el futuro cuando sean más profundos mis conocimientos sobre la arquitectura lógica de OpenStack.

Otra parte fundamental es llevar unas buenas prácticas en la configuración y almacenamiento de contraseñas y pares de claves, siendo necesario asegurar que no se encuentran al alcance de un ataque de cualquier tipo.

Para más información sobre la seguridad en OpenStack se puede encontrar más información en OpenStack Security Guide (HAVANA) [17] o en OpenStack Security guide Official [8].





## 5. Escenario y configuración inicial

### 5.1 Configuración de red

La red se divide en tres nodos u ordenadores, con un router haciendo de switch de red interna, con red ip 172.21.1.0/24. El router hace de punto de acceso y servidor dns. No es necesario configurar DHCP puesto que se va a trabajar con IP estática (aunque por seguridad se emplea DHCP y se hace reserva de IP para cada una de las máquinas dentro del rango). La configuración se puede observar en la tabla 5.1.

	IP	FQDN
<b>PC1</b>	172.21.1.1	viernes
<b>PC1</b>	172.21.1.51	controller.acendata.com
<b>PC2</b>	172.21.1.53	compute.acendata.com
<b>Portatil</b>	172.21.1.52	network.acendata.com

Cuadro 5.1: Especificaciones de los valores de red en los dispositivos del proyecto.

Los nombres Fully qualified domain name utilizados para la resolución DNS de la red (configuración aconsejable de packstack) están directamente relacionados con el propósito al que va dirigido el nodo en cuestión.

Se utiliza un nombre de dominio no existente de propia ocurrencia.

### 5.2 Preconfiguración de los equipos

Todos los equipos son primariamente formateados con el sistema CentOS 7 y se configuran las interfaces de red y nombres de hosts con los parámetros de la tabla 5.1.

Para ver los nombres de host tecleamos en terminal:

```
1 $ hostnamectl status  
2  
3 $ hostnamectl set-hostname <nombre>
```

En cada uno de los tres hosts a configurar entrar en el archivo */etc/hosts* e introducimos los siguiente parámetros:

```

1 172.21.1.51    controller.acendata.com controller
2 172.21.1.52    network.acendata.com   network
3 172.21.1.53    compute.acendata.com  compute

```

y guardamos.

Actualizamos el sistema y los repositorios.

```
1 $ yum -y update ; reboot
```

Tras el reinicio de todos los nodos. Deshabilitamos Selinux en los tres equipos. Para ello ejecutamos el comando:

```
1 $ setenforce 0
```

Si queremos que este cambio sea permanente tendremos que acceder al fichero */etc/sysconfig/selinux* con un editor y asegurar que el parámetro "SELINUX=" está con el valor "disabled".

Tras ello deshabilitamos NetworkManager y Firewalld, en los tres equipos. Con

```

1 $ systemctl stop NetworkManager
2
3 $ systemctl disable NetworkManager
4
5 $ systemctl stop firewalld
6
7 $ systemctl disable firewalld

```

### 5.2.1 Configurar autenticación sin contraseña en el nodo controller

Vamos a configurar la autenticación sin contraseñas a través de claves ssh desde el controller hacia los nodos compute y network.

Para ello nos situamos en el nodo controller únicamente, y ejecutamos:

```

1 ssh-keygen
2
3 ssh-copy-id -i /root/.ssh/id_rsa.pub root@172.21.1.52
4 ssh-copy-id -i /root/.ssh/id_rsa.pub root@172.21.1.53

```

Tras ello si no hay ningún problema, podremos hacer ssh a los dos nodos y no nos pedirá contraseña.

```

1 ssh compute
2
3 ssh network

```

para asegurarnos pedimos el hostname dentro de la sesión ssh para comprobar que estamos dentro de las otras máquinas. Con esto ya tendríamos la preconfiguración de los equipos preparada y podríamos pasar a la instalación.



# OPENSTACK POWERED PLANET

## 6. Instalación de OpenStack

6.1

6.1.1

6.1.2

6.2

6.3

Figura 6.1: In this



# 7. Inicialización y puesta en marcha del sistema

## 7.1 Login

## 7.2 Personalización

## 7.3 Usuarios y grupos

### 7.3.1 Usuarios

### 7.3.2 Grupos

## 7.4 proyectos

### 7.4.1 Instancias

### 7.4.2 imágenes

### 7.4.3 volúmenes

Figura 7.1: In this

### 7.4.4 imágenes

Figura 7.2: The.

### 7.4.5 redes

Figura 7.3: Illu

Figura 7.4: All



## **8. Contratiempos, Soluciones y línea temporal**

**8.1 Contratiempos y soluciones**

**8.2 Línea temporal**



## **9. Recursos utilizados, Costes y presupuesto**

**9.1 Recursos utilizados**

**9.2 Costes**

**9.3 Presupuesto**



# 10. Epílogo

## 10.1 Ideas para el futuro

## 10.2 Palabras finales

Aquí bajamos el la imagen de debian [12] Tenemos este libro bonito [32] También tenemos a [14] También tenemos a [11] También tenemos a [24] También uno nuevo [29] También tenemos a [15] También tenemos a [25] también [16] También [18] También [10] [9] [5] [27] [13]

*Hola mundo*

---

Pedro V. López Bañón



## Bibliografía

- [1] URL: [https://es.wikipedia.org/wiki/Replicaci%C3%B3n\\_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Replicaci%C3%B3n_(inform%C3%A1tica)) (véase página 12).
- [2] URL: <http://blog.opencloud.es/database-as-a-service/> (véase página 38).
- [3] URL: <https://unpocodejava.com/2017/05/30/que-es-openstack-sahara/> (véase página 39).
- [4] *Aprendizaje Automático*. URL: [https://es.wikipedia.org/wiki/Aprendizaje\\_autom%C3%A1tico](https://es.wikipedia.org/wiki/Aprendizaje_autom%C3%A1tico) (véase página 12).
- [5] Rack Blogger. *OpenStack y virtualización, ¿en qué se diferencian?* URL: <https://blog.rackspace.com/es/openstack-y-virtualizacion-en-que-se-diferencian> (véase página 55).
- [6] *CentOS*. URL: <https://es.wikipedia.org/wiki/CentOS> (véanse páginas 11, 28).
- [7] *Ciencia de datos*. URL: [https://es.wikipedia.org/wiki/Ciencia\\_de\\_datos](https://es.wikipedia.org/wiki/Ciencia_de_datos) (véanse páginas 11, 19).
- [8] OpenStack Contributors. *OpenStack Security Guide*. openstack.org, 2018. URL: <https://docs.openstack.org/security-guide/> (véase página 43).
- [9] OpenStack Contributors. URL: <https://docs.openstack.org/sahara/queens/admin/configuration-guide.html> (véase página 55).
- [10] OpenStack Contributors. *Sahara Installation Guide*. URL: <https://docs.openstack.org/sahara/queens/install/installation-guide.html#to-install-with-rdo> (véase página 55).
- [11] OpenStack contributors. *Virtual Machine Image Guide*. OpenStack, 2018 (véase página 55).
- [12] Debian. *Debian Image for OpenStack*. URL: <https://cdimage.debian.org/cdimage/openstack/current/> (véase página 55).

- [13] *Documentation for Queens*. Mar. de 2018. URL: <https://docs.openstack.org/queens/> (véase página 55).
- [14] Antonio Cobos Domínguez. *Despliegue de arquitectura cloud basada en OpenStack y su integración con Chef sobre CentOS*. Sevilla: Universidad de Sevilla, 2014 (véanse páginas 20, 31, 55).
- [15] FLOSSYSTEMS. “Clouds con OpenStack”. En: Sevilla, jun. de 2012. URL: <http://www.gonzalonazareno.org/cloud/formacion/openstack.pdf> (véase página 55).
- [16] Alejandro Carlos Osuna Fontan. *Creación de sistema Cloud con OpenStack*. Valencia: Universidad Politécnica de Valencia, 2016 (véase página 55).
- [17] OpenStack Foundation. *OpenStack Security Guide*. Havana, 2013 (véase página 43).
- [18] Everett Toews Joe Topjian; Lorin Hochstein; Jonathan Proulx; Tom Fifield; Diano Fleming; Anne Gentle. *OpenStack operations Guide*. Sebastopol: O'Reilly, 2014. ISBN: 978-1-491-94695-4 (véase página 55).
- [19] *Git*. URL: <https://es.wikipedia.org/wiki/Git> (véase página 29).
- [20] *Git Hub*. URL: <https://es.wikipedia.org/wiki/GitHub> (véase página 29).
- [21] *Inteligencia empresarial*. URL: [https://es.wikipedia.org/wiki/Inteligencia\\_empresarial](https://es.wikipedia.org/wiki/Inteligencia_empresarial) (véase página 11).
- [22] *Macrodatos*. URL: <https://es.wikipedia.org/wiki/Macrodatos> (véase página 11).
- [23] *OpenStack desde cero - KeyStone*. URL: <https://www.dbigcloud.com/cloud-computing/170-openstack-desde-cero-keystone.html> (véase página 31).
- [24] Manuel Parra. *Instalación y despliegue de OpenStack*. URL: <https://github.com/manuparra/openstack> (véase página 55).
- [25] *Proyecto Cloud Computing*. Sevilla: I.E.S. Gonzalo Nazareno. URL: <http://www.gonzalonazareno.org/cloud/> (véase página 55).
- [26] *Python*. URL: <https://es.wikipedia.org/wiki/Python> (véase página 28).
- [27] rackspace. *¿Qué es OpenStack?* URL: <https://www.rackspace.com/es/library/what-is-openstack> (véanse páginas 23, 55).
- [28] *Red Hat Enterprise Linux*. URL: [https://es.wikipedia.org/wiki/Red\\_Hat\\_Enterprise\\_Linux](https://es.wikipedia.org/wiki/Red_Hat_Enterprise_Linux) (véase página 12).
- [29] Benjamin Ruland; Dr. Yih Leong Sun Thomas Bludau; William Genovese; Meenakshi Kaushik; Dr. Peter Meitz. *OpenStack Workload Reference Arquitecture: Big Dta*. URL: <https://www.openstack.org/assets/software/mitaka/OpenStack-WorkloadRefBigData-v4.pdf> (véase página 55).
- [30] *Tipos de Cloud Computing*. Open Webinars. URL: <https://openwebinars.net/blog/tipos-de-cloud-computing/> (véase página 21).

- [31] *Ventajas del cloud Computing.* IBM. URL: <https://www.ibm.com/cloud-computing/es-es/learn-more/benefits-of-cloud-computing/> (véase página 20).
- [32] Luís Manuel Guijarro Verdura. *Despliegue de una Nube de Computación Privada OpenStack en un entorno académico.* Leganés, Madrid: Universidad Carlos III, jun. de 2014 (véase página 55).



# 11. Anexos

## 11.1 Anexo A

Fichero de configuración de Packstack sin comentarios

```
1 [general]
2 CONFIG_SSH_KEY=/root/.ssh/id_rsa.pub
3 CONFIG_DEFAULT_PASSWORD=
4 CONFIG_SERVICE_WORKERS=%{::processorcount}
5 CONFIG_MARIADB_INSTALL=y
6 CONFIG_GLANCE_INSTALL=y
7 CONFIG_CINDER_INSTALL=y
8 CONFIG_MANILA_INSTALL=n
9 CONFIG_NOVA_INSTALL=y
10 CONFIG_NEUTRON_INSTALL=y
11 CONFIG_HORIZON_INSTALL=y
12 CONFIG_SWIFT_INSTALL=y
13 CONFIG_CEILOMETER_INSTALL=y
14 CONFIG_AUDIT_INSTALL=y
15 CONFIG_PANKO_INSTALL=n
16 CONFIG_SAHARA_INSTALL=n
17 CONFIG_HEAT_INSTALL=n
18 CONFIG_MAGNUM_INSTALL=n
19 CONFIG_TROVE_INSTALL=n
20 CONFIG_IRONIC_INSTALL=n
21 CONFIG_CLIENT_INSTALL=y
22 CONFIG_NTP_SERVERS=
23 EXCLUDE_SERVERS=
24 CONFIG_DEBUG_MODE=n
25 CONFIG_CONTROLLER_HOST=172.21.1.51
26 CONFIG_COMPUTE_HOSTS=172.21.1.53
27 CONFIG_NETWORK_HOSTS=172.21.1.52
28 CONFIG_VMWARE_BACKEND=n
29 CONFIG_UNSUPPORTED=n
30 CONFIG_USE_SUBNETS=n
31 CONFIG_VCENTER_HOST=
32 CONFIG_VCENTER_USER=
33 CONFIG_VCENTER_PASSWORD=
34 CONFIG_VCENTER_CLUSTER_NAMES=
35 CONFIG_STORAGE_HOST=172.21.1.52
36 CONFIG_SAHARA_HOST=172.21.1.53
37 CONFIG_REPO=
38 CONFIG_ENABLE_RDO_TESTING=n
39 CONFIG_RH_USER=
40 CONFIG_SATELLITE_URL=
41 CONFIG_RH_SAT6_SERVER=
42 CONFIG_RH_PW=
43 CONFIG_RH_OPTIONAL=y
44 CONFIG_RH_PROXY=
45 CONFIG_RH_SAT6_ORG=
46 CONFIG_RH_SAT6_KEY=
47 CONFIG_RH_PROXY_PORT=
48 CONFIG_RH_PROXY_USER=
49 CONFIG_RH_PROXY_PW=
50 CONFIG_SATELLITE_USER=
51 CONFIG_SATELLITE_PW=
52 CONFIG_SATELLITE_AKEY=
53 CONFIG_SATELLITE_CACERT=
54 CONFIG_SATELLITE_PROFILE=
55 CONFIG_SATELLITE_FLAGS=
56 CONFIG_SATELLITE_PROXY=
```

```

57   CONFIG_SATELLITE_PROXY_USER=
58   CONFIG_SATELLITE_PROXY_PW=
59   CONFIG_SSL_CACERT_FILE=/etc/pki/tls/certs/selfcert.crt
60   CONFIG_SSL_CACERT_KEY_FILE=/etc/pki/tls/private/selfkey.key
61   CONFIG_SSL_CERT_DIR=/packstackca/
62   CONFIG_SSL_CACERT_SELFSIGN=y
63   CONFIG_SSL_CERT_SUBJECT_C=-
64   CONFIG_SSL_CERT_SUBJECT_ST=State
65   CONFIG_SSL_CERT_SUBJECT_L=City
66   CONFIG_SSL_CERT_SUBJECT_O=openstack
67   CONFIG_SSL_CERT_SUBJECT_OU=packstack
68   CONFIG_SSL_CERT SUBJECT_CN=central.acadata.com
69   CONFIG_SSL_CERT SUBJECT_MAIL=admin@central.acadata.com
70   CONFIG_AMQP_BACKEND=rabbitmq
71   CONFIG_AMQP_HOST=172.21.1.51
72   CONFIG_AMQP_ENABLE_SSL=n
73   CONFIG_AMQP_ENABLE_AUTH=n
74   CONFIG_AMQP_NSS_CERTDB_PW=PW_PLACEHOLDER
75   CONFIG_AMQP_AUTH_USER=amqp_user
76   CONFIG_AMQP_AUTH_PASSWORD=PW_PLACEHOLDER
77   CONFIG_MARIADB_HOST=172.21.1.51
78   CONFIG_MARIADB_USER=root
79   CONFIG_MARIADB_PW=380df20842ee47f7
80   CONFIG_KEYSTONE_DB_PW=9baa7cfe87740cb
81   CONFIG_KEYSTONE_DB_PURGE_ENABLE=True
82   CONFIG_KEYSTONE_REGION=RegionOne
83   CONFIG_KEYSTONE_ADMIN_TOKEN=0d7478be89504701a78d42a952ed3380
84   CONFIG_KEYSTONE_ADMIN_EMAIL=root@localhost
85   CONFIG_KEYSTONE_ADMIN_USERNAME=admin
86   CONFIG_KEYSTONE_ADMIN_PW=3d4bb514ea2f4dba
87   CONFIG_KEYSTONE_DEMO_PW=7bbf083c0d9f4227
88   CONFIG_KEYSTONE_API_VERSION=v3
89   CONFIG_KEYSTONE_TOKEN_FORMAT=FERNET
90   CONFIG_KEYSTONE_IDENTITY_BACKEND=mysql
91   CONFIG_KEYSTONE_LDAP_URL=ldap://172.21.1.51
92   CONFIG_KEYSTONE_LDAP_USER_DN=
93   CONFIG_KEYSTONE_LDAP_USER_PASSWORD=
94   CONFIG_KEYSTONE_LDAP_SUFFIX=
95   CONFIG_KEYSTONE_LDAP_QUERY_SCOPE=one
96   CONFIG_KEYSTONE_LDAP_PAGE_SIZE=-1
97   CONFIG_KEYSTONE_LDAP_USER_SUBTREE=
98   CONFIG_KEYSTONE_LDAP_USER_FILTER=
99   CONFIG_KEYSTONE_LDAP_USER_OBJECTCLASS=
100  CONFIG_KEYSTONE_LDAP_USER_ID_ATTRIBUTE=
101  CONFIG_KEYSTONE_LDAP_USER_NAME_ATTRIBUTE=
102  CONFIG_KEYSTONE_LDAP_USER_MAIL_ATTRIBUTE=
103  CONFIG_KEYSTONE_LDAP_USER_ENABLED_ATTRIBUTE=
104  CONFIG_KEYSTONE_LDAP_USER_ENABLED_MASK=-1
105  CONFIG_KEYSTONE_LDAP_USER_ENABLED_DEFAULT=True
106  CONFIG_KEYSTONE_LDAP_USER_ENABLED_INVERT=n
107  CONFIG_KEYSTONE_LDAP_USER_ATTRIBUTE_IGNORE=
108  CONFIG_KEYSTONE_LDAP_USER_DEFAULT_PROJECT_ID_ATTRIBUTE=
109  CONFIG_KEYSTONE_LDAP_USER_ALLOW_CREATE=n
110  CONFIG_KEYSTONE_LDAP_USER_ALLOW_UPDATE=n
111  CONFIG_KEYSTONE_LDAP_USER_ALLOW_DELETE=n
112  CONFIG_KEYSTONE_LDAP_USER_PASS_ATTRIBUTE=
113  CONFIG_KEYSTONE_LDAP_USER_ENABLED_EMULATION_DN=
114  CONFIG_KEYSTONE_LDAP_USER_ADDITIONAL_ATTRIBUTE_MAPPING=
115  CONFIG_KEYSTONE_LDAP_GROUP_SUBTREE=
116  CONFIG_KEYSTONE_LDAP_GROUP_FILTER=
117  CONFIG_KEYSTONE_LDAP_GROUP_OBJECTCLASS=
118  CONFIG_KEYSTONE_LDAP_GROUP_ID_ATTRIBUTE=
119  CONFIG_KEYSTONE_LDAP_GROUP_NAME_ATTRIBUTE=
120  CONFIG_KEYSTONE_LDAP_GROUP_MEMBER_ATTRIBUTE=
121  CONFIG_KEYSTONE_LDAP_GROUP_DESC_ATTRIBUTE=
122  CONFIG_KEYSTONE_LDAP_GROUP_ATTRIBUTE_IGNORE=
123  CONFIG_KEYSTONE_LDAP_GROUP_ALLOW_CREATE=n
124  CONFIG_KEYSTONE_LDAP_GROUP_ALLOW_UPDATE=n
125  CONFIG_KEYSTONE_LDAP_GROUP_ALLOW_DELETE=n
126  CONFIG_KEYSTONE_LDAP_GROUP_ADDITIONAL_ATTRIBUTE_MAPPING=
127  CONFIG_KEYSTONE_LDAP_USE_TLS=n
128  CONFIG_KEYSTONE_LDAP_TLS_CACERTDIR=
129  CONFIG_KEYSTONE_LDAP_TLS_CACERTFILE=
130  CONFIG_KEYSTONE_LDAP_TLS_REQ_CERT=demand
131  CONFIG_GLANCE_DB_PW=bccb6d1705f73400a
132  CONFIG_GLANCE_KS_PW=b78c41fdea6d48b1
133  CONFIG_GLANCE_BACKEND=file
134  CONFIG_CINDER_DB_PW=53a8317f100842b5
135  CONFIG_CINDER_DB_PURGE_ENABLE=True
136  CONFIG_CINDER_KS_PW=ac91ada1f99c4adc
137  CONFIG_CINDER_BACKEND=vm
138  CONFIG_CINDER_VOLUMES_CREATE=y
139  CONFIG_CINDER_VOLUME_NAME=cinder-volumes
140  CONFIG_CINDER_VOLUMES_SIZE=200
141  CONFIG_CINDER_GLUSTER_MOUNTS=
142  CONFIG_CINDER_NFS_MOUNTS=
143  CONFIG_CINDER_NETAPP_LOGIN=
144  CONFIG_CINDER_NETAPP_PASSWORD=
145  CONFIG_CINDER_NETAPP_HOSTNAME=
146  CONFIG_CINDER_NETAPP_SERVER_PORT=80
147  CONFIG_CINDER_NETAPP_STORAGE_FAMILY=ontap_cluster
148  CONFIG_CINDER_NETAPP_TRANSPORT_TYPE=http
149  CONFIG_CINDER_NETAPP_STORAGE_PROTOCOL=nfs
150  CONFIG_CINDER_NETAPP_SIZE_MULTIPLIER=1.0
151  CONFIG_CINDER_NETAPP_EXPIRY_THRES_MINUTES=720

```

```

152 CONFIG_CINDER_NETAPP_THRES_AVL_SIZE_PERC_START=20
153 CONFIG_CINDER_NETAPP_THRES_AVL_SIZE_PERC_STOP=60
154 CONFIG_CINDER_NETAPP_NFS_SHARES=
155 CONFIG_CINDER_NETAPP_NFS_SHARES_CONFIG=/etc/cinder/shares.conf
156 CONFIG_CINDER_NETAPP_VOLUME_LIST=
157 CONFIG_CINDER_NETAPP_VFILER=
158 CONFIG_CINDER_NETAPP_PARTNER_BACKEND_NAME=
159 CONFIG_CINDER_NETAPP_VSERVER=
160 CONFIG_CINDER_NETAPP_CONTROLLER_IPS=
161 CONFIG_CINDER_NETAPP_SA_PASSWORD=
162 CONFIG_CINDER_NETAPP_ESERIES_HOST_TYPE=linux_dm_mp
163 CONFIG_CINDER_NETAPP_WEBSERVICE_PATH=/devmgr/v2
164 CONFIG_CINDER_NETAPP_STORAGE_POOLS=
165 CONFIG_CINDER_SOLIDFIRE_LOGIN=
166 CONFIG_CINDER_SOLIDFIRE_PASSWORD=
167 CONFIG_CINDER_SOLIDFIRE_HOSTNAME=
168 CONFIG_IRONIC_DB_PW=PW_PLACEHOLDER
169 CONFIG_IRONIC_KS_PW=PW_PLACEHOLDER
170 CONFIG_NOVA_DB_PURGE_ENABLE=True
171 CONFIG_NOVA_DB_PW=59add9394e0f4090
172 CONFIG_NOVA_KS_PW=01d5d3f2db56409a
173 CONFIG_NOVA_MANAGE_FLAVORS=y
174 CONFIG_NOVA_SCHED_CPU_ALLOC_RATIO=16.0
175 CONFIG_NOVA_SCHED_RAM_ALLOC_RATIO=1.5
176 CONFIG_NOVA_COMPUTE_MIGRATE_PROTOCOL=ssh
177 CONFIG_VNC_SSL_CERT=
178 CONFIG_VNC_SSL_KEY=
179 CONFIG_NOVA_PCI_ALIAS=
180 CONFIG_NOVA_PCI_PASSTHROUGH_WHITELIST=
181 CONFIG_NOVA_LIBVIRT_VIRT_TYPE=%{::default_hypervisor}
182 CONFIG_NEUTRON_KS_PW=023dc9a0652f4c08
183 CONFIG_NEUTRON_DB_PW=0bf2ac106c04a60
184 CONFIG_NEUTRON_L3_EXT_BRIDGE=br-ex
185 CONFIG_NEUTRON_METADATA_PW=ba5d9825c2774eb6
186 CONFIG_LBaaS_INSTALL=n
187 CONFIG_NEUTRON_METERING_AGENT_INSTALL=y
188 CONFIG_NEUTRON_FWAAS=n
189 CONFIG_NEUTRON_VPNAAS=n
190 CONFIG_NEUTRON_ML2_TYPE_DRIVERS=vxlan,flat
191 CONFIG_NEUTRON_ML2_TENANT_NETWORK_TYPES=vxlan
192 CONFIG_NEUTRON_ML2_MECHANISM_DRIVERS=openvswitch
193 CONFIG_NEUTRON_ML2_FLAT_NETWORKS=*
194 CONFIG_NEUTRON_ML2_VLAN_RANGES=
195 CONFIG_NEUTRON_ML2_TUNNEL_ID_RANGES=
196 CONFIG_NEUTRON_ML2_VXLAN_GROUP=
197 CONFIG_NEUTRON_ML2_VNI_RANGES=10:100
198 CONFIG_NEUTRON_L2_AGENT=openvswitch
199 CONFIG_NEUTRON_ML2_SUPPORTED_PCI_VENDOR_DEVS=['15b3:1004', '8086:10ca']
200 CONFIG_NEUTRON_ML2_SRIOV_INTERFACE_MAPPINGS=
201 CONFIG_NEUTRON_LB_INTERFACE_MAPPINGS=
202 CONFIG_NEUTRON_OVS_BRIDGE_MAPPINGS=extnet:br-ex
203 CONFIG_NEUTRON_OVS_BRIDGE_IFACES=br-ex:eth0
204 CONFIG_NEUTRON_OVS_BRIDGES_COMPUTE=
205 CONFIG_NEUTRON_OVS_EXTERNAL_PHYSNET=extnet
206 CONFIG_NEUTRON_OVS_TUNNEL_IF=
207 CONFIG_NEUTRON_OVS_TUNNEL_SUBNETS=
208 CONFIG_NEUTRON_OVS_VXLAN_UDP_PORT=4789
209 CONFIG_NEUTRON_OVN_BRIDGE_MAPPINGS=extnet:br-ex
210 CONFIG_NEUTRON_OVN_BRIDGE_IFACES=
211 CONFIG_NEUTRON_OVN_BRIDGES_COMPUTE=
212 CONFIG_NEUTRON_OVN_EXTERNAL_PHYSNET=extnet
213 CONFIG_NEUTRON_OVN_TUNNEL_IF=
214 CONFIG_NEUTRON_OVN_TUNNEL_SUBNETS=
215 CONFIG_MANILA_DB_PW=PW_PLACEHOLDER
216 CONFIG_MANILA_KS_PW=PW_PLACEHOLDER
217 CONFIG_MANILA_BACKEND=generic
218 CONFIG_MANILA_NETAPP_DRV_HANDLES_SHARE_SERVERS=false
219 CONFIG_MANILA_NETAPP_TRANSPORT_TYPE=https
220 CONFIG_MANILA_NETAPP_LOGIN=admin
221 CONFIG_MANILA_NETAPP_PASSWORD=
222 CONFIG_MANILA_NETAPP_SERVER_HOSTNAME=
223 CONFIG_MANILA_NETAPP_STORAGE_FAMILY=ontap_cluster
224 CONFIG_MANILA_NETAPP_SERVER_PORT=443
225 CONFIG_MANILA_NETAPP_AGGREGATE_NAME_SEARCH_PATTERN=(.*)
226 CONFIG_MANILA_NETAPP_ROOT_VOLUME_AGGREGATE=
227 CONFIG_MANILA_NETAPP_ROOT_VOLUME_NAME=root
228 CONFIG_MANILA_NETAPP_VSERVER=
229 CONFIG_MANILA_GENERIC_DRV_HANDLES_SHARE_SERVERS=true
230 CONFIG_MANILA_GENERIC_VOLUME_NAME_TEMPLATE=manila-share-%s
231 CONFIG_MANILA_GENERIC_SHARE_MOUNT_PATH=/shares
232 CONFIG_MANILA_SERVICE_IMAGE_LOCATION=https://www.dropbox.com/s/vi5oeih10q1qkckh/ubuntu_1204_nfs_cifs.qcow2
233 CONFIG_MANILA_SERVICE_INSTANCE_USER=ubuntu
234 CONFIG_MANILA_SERVICE_INSTANCE_PASSWORD=ubuntu
235 CONFIG_MANILA_NETWORK_TYPE=neutron
236 CONFIG_MANILA_NETWORK_STANDALONE_GATEWAY=
237 CONFIG_MANILA_NETWORK_STANDALONE_NETMASK=
238 CONFIG_MANILA_NETWORK_STANDALONE_SEG_ID=
239 CONFIG_MANILA_NETWORK_STANDALONE_IP_RANGE=
240 CONFIG_MANILA_NETWORK_STANDALONE_IP_VERSION=4
241 CONFIG_MANILA_GLUSTERFS_SERVERS=
242 CONFIG_MANILA_GLUSTERFS_NATIVE_PATH_TO_PRIVATE_KEY=
243 CONFIG_MANILA_GLUSTERFS_VOLUME_PATTERN=
244 CONFIG_MANILA_GLUSTERFS_TARGET=
245 CONFIG_MANILA_GLUSTERFS_MOUNT_POINT_BASE=
246 CONFIG_MANILA_GLUSTERFS_NFS_SERVER_TYPE=gluster

```

```

247 CONFIG_MANILA_GLUSTERFS_PATH_TO_PRIVATE_KEY=
248 CONFIG_MANILA_GLISTERFS_GANESHA_SERVER_IP=
249 CONFIG_HORIZON_SSL=n
250 CONFIG_HORIZON_SECRET_KEY=9cfbe907dd374eefb64e21f9e6cac5cd
251 CONFIG_HORIZON_SSL_CERT=
252 CONFIG_HORIZON_SSL_KEY=
253 CONFIG_HORIZON_SSL_CACERT=
254 CONFIG_SWIFT_KS_PW=7c5fd54a2dd74228
255 CONFIG_SWIFT_STORAGES=
256 CONFIG_SWIFT_STORAGE_ZONES=1
257 CONFIG_SWIFT_STORAGE_REPLICAS=1
258 CONFIG_SWIFT_STORAGE_FSTYPE=ext4
259 CONFIG_SWIFT_HASH=8fa2633bcd2e47f9
260 CONFIG_SWIFT_STORAGE_SIZE=2G
261 CONFIG_HEAT_DB_PW=PW_PLACEHOLDER
262 CONFIG_HEAT_AUTH_ENC_KEY=1206c2bded914d64
263 CONFIG_HEAT_KS_PW=PW_PLACEHOLDER
264 CONFIG_HEAT_CFN_INSTALL=y
265 CONFIG_HEAT_DOMAIN=heat
266 CONFIG_HEAT_DOMAIN_ADMIN=heat_admin
267 CONFIG_HEAT_DOMAIN_PASSWORD=PW_PLACEHOLDER
268 CONFIG_PROVISION_DEMO=n
269 CONFIG_PROVISION_TEMPEST=n
270 CONFIG_PROVISION_DEMO_FLOATRANGE=172.24.4.0/24
271 CONFIG_PROVISION_DEMO_ALLOCATION_POOLS=[]
272 CONFIG_PROVISION_IMAGE_NAME=cirros
273 CONFIG_PROVISION_IMAGE_URL=http://download.cirros-cloud.net/0.3.5/cirros-0.3.5-x86_64-disk.img
274 CONFIG_PROVISION_IMAGE_FORMAT=qcow2
275 CONFIG_PROVISION_IMAGE_PROPERTIES=
276 CONFIG_PROVISION_IMAGE_SSH_USER=cirros
277 CONFIG_PROVISION_UEC_IMAGE_NAME=cirros-uec
278 CONFIG_PROVISION_UEC_IMAGE_KERNEL_URL=http://download.cirros-cloud.net/0.3.5/cirros-0.3.5-x86_64-kernel
279 CONFIG_PROVISION_UEC_IMAGE_RAMDISK_URL=http://download.cirros-cloud.net/0.3.5/cirros-0.3.5-x86_64-initramfs
280 CONFIG_PROVISION_UEC_IMAGE_DISK_URL=http://download.cirros-cloud.net/0.3.5/cirros-0.3.5-x86_64-disk.img
281 CONFIG_TEMPEST_HOST=
282 CONFIG_PROVISION_TEMPEST_USER=
283 CONFIG_PROVISION_TEMPEST_USER_PW=PW_PLACEHOLDER
284 CONFIG_PROVISION_TEMPEST_FLOATRANGE=172.24.4.0/24
285 CONFIG_PROVISION_TEMPEST_FLAVOR_NAME=m1.nano
286 CONFIG_PROVISION_TEMPEST_FLAVOR_DISK=0
287 CONFIG_PROVISION_TEMPEST_FLAVOR_RAM=128
288 CONFIG_PROVISION_TEMPEST_FLAVOR_VCPUS=1
289 CONFIG_PROVISION_TEMPEST_FLAVOR_ALT_NAME=m1.micro
290 CONFIG_PROVISION_TEMPEST_FLAVOR_ALT_DISK=0
291 CONFIG_PROVISION_TEMPEST_FLAVOR_ALT_RAM=128
292 CONFIG_PROVISION_TEMPEST_FLAVOR_ALT_VCPUS=1
293 CONFIG_RUN_TEMPEST=n
294 CONFIG_RUN_TEMPEST_TESTS=smoke
295 CONFIG_PROVISION_OVS_BRIDGE=y
296 CONFIG_GNOCCHI_DB_PW=90604bcff541432b
297 CONFIG_GNOCCHI_KS_PW=09af45482a3f4d455
298 CONFIG_CEILOMETER_SECRET=27333ef32ad54a07
299 CONFIG_CEILOMETER_KS_PW=3d36d546d6c4f27e
300 CONFIG_CEILOMETER_SERVICE_NAME=httpd
301 CONFIG_CEILOMETER_COORDINATION_BACKEND=redis
302 CONFIG_ENABLE_CEILOMETER_MIDDLEWARE=n
303 CONFIG_REDIS_HOST=172.21.1.51
304 CONFIG_REDIS_PORT=6379
305 CONFIG_AODH_KS_PW=84cc83566d3e44b8
306 CONFIG_AODH_DB_PW=d9b2a168bcd8483d
307 CONFIG_PANKO_DB_PW=PW_PLACEHOLDER
308 CONFIG_PANKO_KS_PW=PW_PLACEHOLDER
309 CONFIG_TROVE_DB_PW=PW_PLACEHOLDER
310 CONFIG_TROVE_KS_PW=PW_PLACEHOLDER
311 CONFIG_TROVE_NOVA_USER=trove
312 CONFIG_TROVE_NOVA_TENANT=services
313 CONFIG_TROVE_NOVA_PW=PW_PLACEHOLDER
314 CONFIG_SAHLA_DB_PW=PW_PLACEHOLDER
315 CONFIG_SAHLA_KS_PW=PW_PLACEHOLDER
316 CONFIG_MAGNUM_DB_PW=PW_PLACEHOLDER
317 CONFIG_MAGNUM_KS_PW=PW_PLACEHOLDER

```

## 11.2 Anexo B

### Comandos CLI OpenStack

1	+-----+-----+-----+-----+-----+	-15-	openstack.cli	command list	
2	Command Group	Commands	16	module list	
3	+-----+-----+-----+-----+-----+	-17-	+-----+		
4	openstack.alarming.v2	alarm create	18	openstack.common	availability zone list
5		alarm delete	19		configuration show
6		alarm list	20		extension list
7		alarm show	21		extension show
8		alarm state get	22		limits show
9		alarm state set	23		project purge
10		alarm update	24		quota list
11		alarm-history search	25		quota set
12		alarm-history show	26		quota show
13		alarming capabilities list	27	+-----+	
14	+-----+-----+-----+-----+-----+	-28-	openstack.compute.v2	aggregate add host	

29	aggregate create	124	recordset list
30	aggregate delete	125	recordset set
31	aggregate list	126	recordset show
32	aggregate remove host	127	tld create
33	aggregate set	128	tld delete
34	aggregate show	129	tld list
35	aggregate unset	130	tld set
36	compute agent create	131	tld show
37	compute agent delete	132	tsigkey create
38	compute agent list	133	tsigkey delete
39	compute agent set	134	tsigkey list
40	compute service delete	135	tsigkey set
41	compute service list	136	tsigkey show
42	compute service set	137	zone abandon
43	console log show	138	zone axfr
44	console url show	139	zone blacklist create
45	flavor create	140	zone blacklist delete
46	flavor delete	141	zone blacklist list
47	flavor list	142	zone blacklist set
48	flavor set	143	zone blacklist show
49	flavor show	144	zone create
50	flavor unset	145	zone delete
51	host list	146	zone export create
52	host set	147	zone export delete
53	host show	148	zone export list
54	hypervisor list	149	zone export show
55	hypervisor show	150	zone export showfile
56	hypervisor stats show	151	zone import create
57	ip fixed add	152	zone import delete
58	ip fixed remove	153	zone import list
59	ip floating add	154	zone import show
60	ip floating remove	155	zone list
61	keypair create	156	zone set
62	keypair delete	157	zone show
63	keypair list	158	zone transfer accept list
64	keypair show	159	zone transfer accept request
65	server add fixed ip	160	zone transfer accept show
66	server add floating ip	161	zone transfer request create
67	server add network	162	zone transfer request delete
68	server add port	163	zone transfer request list
69	server add security group	164	zone transfer request set
70	server add volume	165	zone transfer request show
71	server backup create	166	+-----+-----+
72	server create	167	openstack.identity.v3
73	server delete	168	access token create
74	server dump create	169	catalog list
75	server event list	170	catalog show
76	server event show	171	consumer create
77	server group create	172	consumer delete
78	server group delete	173	consumer list
79	server group list	174	consumer set
80	server group show	175	consumer show
81	server image create	176	credential create
82	server list	177	credential delete
83	server lock	178	credential list
84	server migrate	179	credential set
85	server pause	180	credential show
86	server reboot	181	domain create
87	server rebuild	182	domain delete
88	server remove fixed ip	183	domain list
89	server remove floating ip	184	domain set
90	server remove network	185	domain show
91	server remove port	186	ec2 credentials create
92	server remove security group	187	ec2 credentials delete
93	server remove volume	188	ec2 credentials list
94	server rescue	189	ec2 credentials show
95	server resize	190	endpoint add project
96	server restore	191	endpoint create
97	server resume	192	endpoint delete
98	server set	193	endpoint list
99	server shelve	194	endpoint remove project
100	server show	195	endpoint set
101	server ssh	196	endpoint show
102	server start	197	federation domain list
103	server stop	198	federation project list
104	server suspend	199	federation protocol create
105	server unlock	200	federation protocol delete
106	server unpause	201	federation protocol list
107	server unrescue	202	federation protocol set
108	server unset	203	federation protocol show
109	server unshelve	204	group add user
110	usage list	205	group contains user
111	usage show	206	group create
112 +-----+	+-----+-----+	207 +-----+	group delete
113   openstack.dns.v2	dns quota list	208	group list
114	dns quota reset	209	group remove user
115	dns quota set	210	group set
116	dns service list	211	group show
117	dns service show	212	identity provider create
118	ptr record list	213	identity provider delete
119	ptr record set	214	identity provider list
120	ptr record show	215	identity provider set
121	ptr record unset	216	identity provider show
122	recordset create	217	implied role create
123	recordset delete	218	implied role delete
			implied role list

219	mapping create	314	metric benchmark measures show
220	mapping delete	315	metric benchmark metric create
221	mapping list	316	metric benchmark metric show
222	mapping set	317	metric capabilities list
223	mapping show	318	metric create
224	policy create	319	metric delete
225	policy delete	320	metric list
226	policy list	321	metric measures add
227	policy set	322	metric measures aggregation
228	policy show	323	metric measures batch-metrics
229	project create	324	metric measures \
230	project delete	325	batch-resources-metrics
231	project list	326	metric measures show
232	project set	327	metric metric create
233	project show	328	metric metric delete
234	region create	329	metric metric list
235	region delete	330	metric metric show
236	region list	331	metric resource batch delete
237	region set	332	metric resource create
238	region show	333	metric resource delete
239	request token authorize	334	metric resource history
240	request token create	335	metric resource list
241	role add	336	metric resource search
242	role assignment list	337	metric resource show
243	role create	338	metric resource update
244	role delete	339	metric resource-type create
245	role list	340	metric resource-type delete
246	role remove	341	metric resource-type list
247	role set	342	metric resource-type show
248	role show	343	metric resource-type update
249	service create	344	metric server version
250	service delete	345	metric show
251	service list	346	metric status
252	service provider create	347	+-----+-----+
253	service provider delete	348	openstack.network.v2
254	service provider list	349	address scope create
255	service provider set	350	address scope delete
256	service provider show	351	address scope list
257	service set	352	address scope set
258	service show	353	address scope show
259	token issue	354	floating ip create
260	token revoke	355	floating ip delete
261	trust create	356	floating ip list
262	trust delete	357	floating ip pool list
263	trust list	358	floating ip set
264	trust show	359	floating ip show
265	user create	360	floating ip unset
266	user delete	361	ip availability list
267	user list	362	ip availability show
268	user password set	363	ip floating create
269	user set	364	ip floating delete
270	user show	365	ip floating list
271	+-----+-----+-----+-----+	366+	ip floating pool list
272	+-----+-----+-----+-----+	366+	ip floating show
273	openstack.image.v2	367	network agent add network
274	image add project	368	network agent add router
275	image create	369	network agent delete
276	image delete	370	network agent list
277	image list	371	network agent remove network
278	image remove project	371	network agent remove router
279	image save	372	network agent set
280	image set	373	network agent show
281	image show	374	network auto allocated \
282	image unset	375	topology create
283	+-----+-----+-----+-----+	376+	network auto allocated \
284	openstack.key_manager.v1	377	topology delete
285	acl delete	377	network create
286	acl get	378	network delete
287	acl submit	379	network flavor add profile
288	acl user add	380	network flavor create
289	acl user remove	381	network flavor delete
290	ca get	382	network flavor list
291	ca list	383	network flavor profile create
292	secret container create	384	network flavor profile delete
293	secret container delete	385	network flavor profile list
294	secret container get	386	network flavor profile set
295	secret container list	387	network flavor profile show
296	secret delete	388	network flavor remove profile
297	secret get	389	network flavor set
298	secret list	390	network flavor show
299	secret order create	391	network list
300	secret order delete	392	network meter create
301	secret order get	393	network meter delete
302	secret order list	394	network meter list
303	secret store	395	network meter rule create
304	secret update	396	network meter rule delete
305	+-----+-----+-----+-----+	397+	network meter rule list
306	openstack.metric.v1	398	network meter rule show
307	metric aggregates	398	network meter show
308	metric archive-policy create	399	network qos policy create
309	metric archive-policy delete	400	network qos policy delete
310	metric archive-policy list	401	network qos policy list
311	metric archive-policy show	402	network qos policy set
312	metric archive-policy update	403	network qos policy show
313	metric archive-policy-rule create	404	network qos rule create
314	metric archive-policy-rule delete	405	network qos rule delete
315	metric archive-policy-rule list	406	
316	metric archive-policy-rule show	407	
317	metric benchmark measures add	408	

409	network qos rule list	504	firewall group create
410	network qos rule set	505	firewall group delete
411	network qos rule show	506	firewall group list
412	network qos rule type list	507	firewall group policy add rule
413	network qos rule type show	508	firewall group policy create
414	network rbac create	509	firewall group policy delete
415	network rbac delete	510	firewall group policy list
416	network rbac list	511	firewall group policy remove rule
417	network rbac set	512	firewall group policy set
418	network rbac show	513	firewall group policy show
419	network segment create	514	firewall group policy unset
420	network segment delete	515	firewall group rule create
421	network segment list	516	firewall group rule delete
422	network segment set	517	firewall group rule list
423	network segment show	518	firewall group rule set
424	network service provider list	519	firewall group rule show
425	network set	520	firewall group rule unset
426	network show	521	firewall group set
427	network unset	522	firewall group show
428	port create	523	firewall group unset
429	port delete	524	network log create
430	port list	525	network log delete
431	port set	526	network log list
432	port show	527	network log set
433	port unset	528	network log show
434	router add port	529	network loggable resources list
435	router add subnet	530	network subport list
436	router create	531	network trunk create
437	router delete	532	network trunk delete
438	router list	533	network trunk list
439	router remove port	534	network trunk set
440	router remove subnet	535	network trunk show
441	router set	536	network trunk unset
442	router show	537	sfc flow classifier create
443	router unset	538	sfc flow classifier delete
444	security group create	539	sfc flow classifier list
445	security group delete	540	sfc flow classifier set
446	security group list	541	sfc flow classifier show
447	security group rule create	542	sfc port chain create
448	security group rule delete	543	sfc port chain delete
449	security group rule list	544	sfc port chain list
450	security group rule show	545	sfc port chain set
451	security group set	546	sfc port chain show
452	security group show	547	sfc port chain unset
453	subnet create	548	sfc port pair create
454	subnet delete	549	sfc port pair delete
455	subnet list	550	sfc port pair group create
456	subnet pool create	551	sfc port pair group delete
457	subnet pool delete	552	sfc port pair group list
458	subnet pool list	553	sfc port pair group set
459	subnet pool set	554	sfc port pair group show
460	subnet pool show	555	sfc port pair group unset
461	subnet pool unset	556	sfc port pair list
462	subnet set	557	sfc port pair set
463	subnet show	558	sfc port pair show
464	subnet unset	559	sfc service graph create
465	+-----	560+	sfc service graph delete
466   openstack.neutronclient.v2	bgp dragnet add speaker	561	sfc service graph list
467	bgp dragnet remove speaker	562	sfc service graph set
468	bgp peer create	563	sfc service graph show
469	bgp peer delete	564	vpn endpoint group create
470	bgp peer list	565	vpn endpoint group delete
471	bgp peer set	566	vpn endpoint group list
472	bgp peer show	567	vpn endpoint group set
473	bgp speaker add network	568	vpn endpoint group show
474	bgp speaker add peer	569	vpn ike policy create
475	bgp speaker create	570	vpn ike policy delete
476	bgp speaker delete	571	vpn ike policy list
477	bgp speaker list	572	vpn ike policy set
478	bgp speaker list advertised routes	573	vpn ike policy show
479	bgp speaker remove network	574	vpn ipsec policy create
480	bgp speaker remove peer	575	vpn ipsec policy delete
481	bgp speaker set	576	vpn ipsec policy list
482	bgp speaker show	577	vpn ipsec policy set
483	bgp speaker show dragents	578	vpn ipsec policy show
484	bgpvpn create	579	vpn ipsec site connection create
485	bgpvpn delete	580	vpn ipsec site connection delete
486	bgpvpn list	581	vpn ipsec site connection list
487	bgpvpn network association create	582	vpn ipsec site connection set
488	bgpvpn network association delete	583	vpn ipsec site connection show
489	bgpvpn network association list	584	vpn service create
490	bgpvpn network association show	585	vpn service delete
491	bgpvpn port association create	586	vpn service list
492	bgpvpn port association delete	587	vpn service set
493	bgpvpn port association list	588	vpn service show
494	bgpvpn port association set	589	+-----
495	bgpvpn port association show	590	openstack.object_store.v1
496	bgpvpn port association unset	591	container create
497	bgpvpn router association create	592	container delete
498	bgpvpn router association delete	593	container list
499	bgpvpn router association list	594	container save
500	bgpvpn router association show	595	container set
501	bgpvpn set	596	container show
502	bgpvpn show	597	container unset
503	bgpvpn unset	598	object create
			object delete

599	object list	636		volume create
600	object save	637		volume delete
601	object set	638		volume host failover
602	object show	639		volume host set
603	object store account set	640		volume list
604	object store account show	641		volume migrate
605	object store account unset	642		volume qos associate
606	object unset	643		volume qos create
607 +-----		644 +-----		volume qos delete
608   openstack.volume.v2	backup create	645		volume qos disassociate
609	backup delete	646		volume qos list
610	backup list	647		volume qos set
611	backup restore	648		volume qos show
612	backup show	649		volume qos unset
613	consistency group add volume	650		volume service list
614	consistency group create	651		volume service set
615	consistency group delete	652		volume set
616	consistency group list	653		volume show
617	consistency group remove volume	654		volume snapshot create
618	consistency group set	655		volume snapshot delete
619	consistency group show	656		volume snapshot list
620	consistency group snapshot create	657		volume snapshot set
621	consistency group snapshot delete	658		volume snapshot show
622	consistency group snapshot list	659		volume snapshot unset
623	consistency group snapshot show	660		volume transfer request accept
624	snapshot create	661		volume transfer request create
625	snapshot delete	662		volume transfer request delete
626	snapshot list	663		volume transfer request list
627	snapshot set	664		volume transfer request show
628	snapshot show	665		volume type create
629	snapshot unset	666		volume type delete
630	volume backup create	667		volume type list
631	volume backup delete	668		volume type set
632	volume backup list	669		volume type show
633	volume backup restore	670		volume type unset
634	volume backup set	671		volume unset
635	volume backup show	672	+-----+-----+	