

**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE CUENCA**

**CARRERA DE INGENIERÍA DE SISTEMAS**

*Trabajo de titulación previo  
a la obtención del título de  
Ingeniero de Sistemas*

**PROYECTO TÉCNICO:**

**“DISEÑO Y ELABORACIÓN DE UN FRAMEWORK COMO  
HERRAMIENTA DE AUTOMATIZACIÓN PARA PROCESOS DE  
MIGRACIÓN DE MÁQUINAS VIRTUALES ENTRE UNA NUBE  
PROPIETARIA (VCLLOUD) Y NUBE OPENSOURCE (OPENSTACK)”**

**AUTORES:**

MARIO ANDRÉS LÓPEZ MERCHÁN  
CHRISTIAN LEÓNIDAS SUCO MARÍN

**TUTOR:**

REMIGIO ISMAEL HURTADO ORTIZ, PhD.

CUENCA - ECUADOR

2021

## CESIÓN DE DERECHOS DE AUTOR

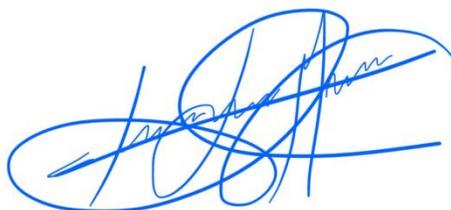
Nosotros, Mario Andrés López Merchán con documento de identificación N° 0105355143 y Christian Leónidas Suco Marín con documento de identificación N° 0105004014, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana, la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación: **“DISEÑO Y ELABORACIÓN DE UN FRAMEWORK COMO HERRAMIENTA DE AUTOMATIZACIÓN PARA PROCESOS DE MIGRACIÓN DE MÁQUINAS VIRTUALES ENTRE UNA NUBE PROPIETARIA (VPCLOUD) Y NUBE OPENSOURCE (OPENSTACK)”**, mismo que ha sido desarrollado para optar por el título de: *Ingeniero de Sistemas*, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores, nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, marzo del 2021.



Mario Andrés López Merchán  
C.I. 0105355143



Christian Leónidas Suco Marín  
C.I. 0105004014

## CERTIFICACIÓN

Yo, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **“DISEÑO Y ELABORACIÓN DE UN FRAMEWORK COMO HERRAMIENTA DE AUTOMATIZACIÓN PARA PROCESOS DE MIGRACIÓN DE MÁQUINAS VIRTUALES ENTRE UNA NUBE PROPIETARIA (VCLLOUD) Y NUBE OPENSOURCE (OPENSTACK)”**, realizado por Mario Andrés López Merchán y Christian Leónidas Suco Marín, obteniendo el *Proyecto Técnico*, que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, marzo del 2021.



Ing. Remigio Ismael Hurtado Ortiz, PhD.

C.I. 0104621388

## DECLARATORIA DE RESPONSABILIDAD

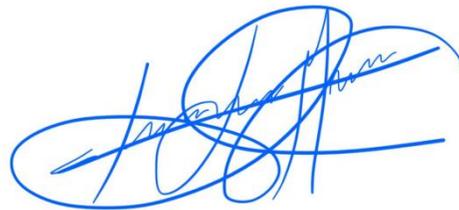
Nosotros, Mario Andrés López Merchán con documento de identificación N° 0105355143 y Christian Leónidas Suco Marín con documento de identificación N° 0105004014, autores del trabajo de titulación: **“DISEÑO Y ELABORACIÓN DE UN FRAMEWORK COMO HERRAMIENTA DE AUTOMATIZACIÓN PARA PROCESOS DE MIGRACIÓN DE MÁQUINAS VIRTUALES ENTRE UNA NUBE PROPIETARIA (VCLLOUD) Y NUBE OPENSOURCE (OPENSTACK)”**, certificamos que el total contenido del *Proyecto Técnico*, es de nuestra exclusiva responsabilidad y autoría.

Cuenca, marzo del 2021.



Mario Andrés López Merchán

C.I. 0105355143



Christian Leónidas Suco Marín

C.I.0105004014

## **DEDICATORIA**

*Queremos dedicar este trabajo de titulación, a nuestros maestros por brindarnos sus conocimientos y apoyo a lo largo de mi carrera universitaria, a nuestros compañeros de aula con quienes compartimos gratos momentos, a nuestros amigos del grupo de investigación Cloud Computing, Smart Cities and High Performance Computing por sus consejos que nos sirvieron de muchísima utilidad, y en especial a nuestros queridos padres y familiares quienes nunca dejaron de apoyarnos y creer en nosotros.*

***Mario Andrés López Merchán***

***Christian Leónidas Suco Marín***

*Quiero dedicar este trabajo de titulación a mis maestros por brindarme el conocimiento y el apoyo incondicional a lo largo de mi carrera y vida universitaria, a mis compañeros de aula, a mis amigos del grupo de investigación Cloud Computing, Smart Cities and High Performance Computing, y en especial familia, mi hijo y mis queridos padres quienes nunca dejaron de apoyarme y creer en mí.*

***Christian Leónidas Suco Marín***

*Quiero dedicar este trabajo de titulación, a mis queridos padres Benjamín y Carmen quienes nunca dejaron de apoyarme, aconsejarme y creer en mí, a mis hermanos Flavio, Nancy y Fanny quienes nunca dejaron de guiarme y ayudarme a fin de alcanzar esta meta, a mis maestros por brindarme sus conocimientos y apoyo a lo largo de mi carrera universitaria, a nuestros compañeros de aula con quienes compartimos gratos momentos, a nuestros amigos del grupo de investigación Cloud Computing, Smart Cities and High Performance Computing por sus consejos que sirvieron de muchísima utilidad.*

**Mario Andrés López Merchán**

## ÍNDICE DE CONTENIDO

<b>INDICE DE FIGURAS.....</b>	<b>12</b>
<b>INDICE DE TABLAS.....</b>	<b>15</b>
<b>RESUMEN.....</b>	<b>16</b>
<b>ABSTRACT.....</b>	<b>17</b>
<b>CAPITULO I INTRODUCCIÓN Y REVISIÓN DEL ESTADO DEL ARTE .....</b>	<b>18</b>
<b>1.1. INTRODUCCIÓN .....</b>	<b>18</b>
<b>1.2. OBJETIVOS.....</b>	<b>20</b>
1.2.1. OBJETIVO GENERAL .....	20
1.2.2. OBJETIVOS ESPECÍFICOS .....	20
<b>1.3. MARCO TEÓRICO .....</b>	<b>21</b>
1.3.1. VIRTUALIZACIÓN.....	21
1.3.1.1. TIPOS DE VIRTUALIZACIÓN.....	22
1.3.1.2. Ventajas y Desventajas de la Virtualización .....	23
1.3.2. CLOUD COMPUTING “COMPUTACIÓN EN LA NUBE” .....	24
1.3.2.1. Servicios en Cloud Computing.....	25
1.3.2.2. Despliegue de Cloud Computing.....	26
1.3.2.3. Características de Cloud Computing .....	27
1.3.3. PROYECTO OPENSTACK .....	28
1.3.3.1. Arquitectura de OpenStack.....	28
1.3.3.1.1. Nova.....	29
1.3.3.1.2. Glance .....	29
1.3.3.1.3. Swift.....	29
1.3.3.1.4. Neutron .....	30
1.3.3.1.5. Cinder .....	30
1.3.3.1.6. Heat.....	30
1.3.3.1.7. Ceilometer.....	30
1.3.3.1.8. Keystone .....	30
1.3.3.1.9. Horizon .....	31
1.3.4. VMWARE vCLOUD DIRECTOR .....	31
1.3.4.1. Estructura de VMWare VCloud .....	32
1.3.4.1.1. Organizaciones.....	32
1.3.4.1.2. Usuarios y grupos .....	32
1.3.4.1.3. Centros de datos virtuales .....	33
1.3.4.1.4. Redes de centros de datos virtuales de organización.....	33
1.3.4.1.5. Redes de vApp .....	33
1.3.4.1.6. Catálogos .....	34
1.3.5. MIGRACIÓN DE MÁQUINAS VIRTUALES .....	34
1.3.5.1. Proceso Migración de VMWare a OpenStack .....	34
1.3.6. TRABAJOS RELACIONADOS .....	35
<b>1.4. TOPOLOGÍA DE RED PROPUESTA.....</b>	<b>37</b>
1.4.1. COMPONENTES FUNCIONALES .....	37
1.4.1.1. DNS.....	37

1.4.1.2.	NFS .....	37
1.4.1.3.	Host ESXi.....	38
1.4.1.4.	Servidor vCenter.....	38
1.4.1.5.	vMotion Network .....	38
1.4.1.6.	Storage Network.....	38
1.4.2.	TOPOLOGÍA .....	38
1.4.3.	RECURSOS DE HARDWARE Y SOFTWARE UTILIZADOS PARA LA INSTALACIÓN DE OPENSTACK Y VCLLOUD .....	43
<b>1.5.</b>	<b>MARCO METODOLÓGICO .....</b>	<b>44</b>
1.5.1.	METODOLOGÍA.....	44
<b>CAPITULO II</b>	<b>ANÁLISIS DE REQUERIMIENTOS .....</b>	<b>45</b>
<b>2.1.</b>	<b>REQUERIMIENTOS DEL SOFTWARE.....</b>	<b>45</b>
<b>2.2.</b>	<b>ELICITACIÓN DE REQUERIMIENTOS .....</b>	<b>45</b>
<b>2.3.</b>	<b>ANÁLISIS DE REQUISITOS .....</b>	<b>46</b>
2.3.1.	REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES .....	46
2.3.1.1.	Requisitos Funcionales.....	46
2.3.1.2.	Requisitos No Funcionales .....	47
2.3.2.	ARQUITECTURA DEL SISTEMA.....	47
<b>CAPITULO III</b>	<b>DISEÑO Y DESARROLLO DEL SISTEMA.....</b>	<b>49</b>
<b>3.1.</b>	<b>DISEÑO DEL SISTEMA .....</b>	<b>49</b>
<b>3.2.</b>	<b>DESARROLLO DEL SISTEMA .....</b>	<b>50</b>
3.2.1.	ARQUITECTURA CLIENTE SERVIDOR .....	50
3.2.2.	ARQUITECTURA N-CAPAS.....	51
3.2.3.	DISEÑO DEL ESQUEMA DE FUNCIONAMIENTO .....	53
3.2.4.	PROCESO DE INTERACCIÓN CON EL FRAMEWORK.....	55
3.2.5.	CONFIGURACIÓN DEL ENTORNO DE DESARROLLO.....	56
3.2.5.1.	Configuración de Python.....	56
3.2.5.2.	Paquetes y Librerías .....	56
3.2.5.3.	IDE .....	57
3.2.5.4.	Desarrollo del Sistema de Migración de Instancias desde vCloud hacia Openstack .....	57
3.2.5.4.1.	Ventana Uno (Datos de las Nubes VCloud y OpenStack) .....	57
3.2.5.4.2.	Ventana Dos (Selección de Máquina Virtual a Migrar) .....	58
3.2.5.4.3.	Ventana Tres (Selección Redes e Inicio de Proceso Migratorio) .....	59
<b>CAPITULO IV</b>	<b>ENTORNOS DE COMPUTACIÓN EN LA NUBE.....</b>	<b>61</b>
<b>4.1.</b>	<b>ARQUITECTURAS DE COMPUTACIÓN EN LA NUBE.....</b>	<b>61</b>
4.1.1.	AMBIENTE DE MIGRACIÓN ENTRE NUBES .....	61
4.1.2.	ENTORNO DE NUBE DE LICENCIA PÚBLICA(OPENSTACK) .....	62
4.1.2.1.	Características de OpenStack .....	62
4.1.2.2.	Porque se eligió OpenStack.....	64
4.1.2.3.	Despliegue de Nube OpenSource .....	64
4.1.2.4.	Instalación de OpenStack .....	65
4.1.3.	ENTORNO DE NUBE PROPIETARIA (VCLLOUD DIRECTOR) .....	66

4.1.3.1.	Características de VMWare VCloud .....	67
4.1.3.2.	Por qué se eligió VMWare VCloud.....	67
4.1.3.3.	Despliegue de la Nube Propietaria (VMWare VCloud) .....	68
4.1.3.4.	Instalación de VMWare vCloud.....	70
4.1.4.	ENTORNO DE FRAMEWORK DE MIGRACIÓN .....	70
4.1.4.1.	Migración de Máquinas Virtuales con Framework Creado .....	70
<b>CAPITULO V PRUEBAS Y RESULTADOS.....</b>		<b>72</b>
<b>5.1.</b>	<b>ANÁLISIS DE ENTORNO Y HERRAMIENTAS EXISTENTES.....</b>	<b>72</b>
<b>5.2.</b>	<b>ARQUITECTURA DE VIRTUALIZACIÓN PARA PRUEBAS DE FUNCIONALIDAD .....</b>	<b>72</b>
<b>5.3.</b>	<b>MEDIDAS DE RENDIMIENTO .....</b>	<b>74</b>
5.3.1.	MÁQUINAS VIRTUALES A MIGRAR .....	74
<b>5.4.</b>	<b>HOST CLIENTE (FRAMEWORK).....</b>	<b>75</b>
5.4.1.	CONSUMO DE RECURSOS .....	76
5.4.1.1.	Proceso de migración 1: .....	76
5.4.1.2.	Proceso de migración 2: .....	78
5.4.1.3.	Proceso de migración 3: .....	81
<b>CAPITULO VI CONCLUSIONES Y RECOMENDACIONES .....</b>		<b>83</b>
<b>6.1.</b>	<b>CONCLUSIONES.....</b>	<b>83</b>
<b>6.2.</b>	<b>RECOMENDACIONES .....</b>	<b>87</b>
<b>GLOSARIO DE TÉRMINOS.....</b>		<b>89</b>
<b>REFERENCIAS.....</b>		<b>91</b>
<b>7.1.</b>	<b>ANEXO 1: INSTALACIÓN DE OPENSTACK.....</b>	<b>93</b>
7.1.1.	ESTABLECER IP ESTÁTICA EN EL SERVIDOR .....	93
7.1.2.	ACTUALIZACIÓN DE CENTOS .....	94
7.1.3.	AGREGAR REPOSITORIOS DE MARIADB .....	94
7.1.4.	DESHABILITAR SELINUX .....	95
7.1.5.	DESHABILITAR SERVICIOS DE FIREWALLD, POSTFIX, NETWORKMANAGER .....	95
7.1.6.	HABILITACIÓN E INICIO DE SERVICIO DE NETWORK .....	96
7.1.7.	INSTALACIÓN DE OPENSTACK TRAIN .....	97
<b>7.2.</b>	<b>ANEXO 2: INSTALACIÓN DE V CLOUD DIRECTOR 10.....</b>	<b>101</b>
7.2.1.	ASIGNACIÓN DE DIRECCIONES IP AL REGISTRO DE DNS LOCAL.....	101
7.2.2.	CONFIGURACIÓN DEL RECURSO COMPARTIDO (NFS).....	101
7.2.3.	DESPLÉGAMOS LA PLANTILLA DE V CLOUD DIRECTOR 10.....	103
7.2.4.	INTERFAZ DE ADMINISTRACIÓN.....	108
7.2.5.	ORGANIZACIONES DE V CLOUD .....	108
7.2.6.	ASIGNACIÓN DE USO DE CLOUD .....	109
7.2.7.	ACCESO A LA PLATAFORMA “USUARIOS ORGANIZATIVOS”.....	109

<b>7.3.</b>	<b>ANEXO 3: ARCHIVOS DE AUTENTICACIÓN PARA LA API DE OPENSTACK</b>	<b>112</b>
<b>7.4.</b>	<b>ANEXO 4: SCRIPT DE INSTALACIÓN DE DRIVERS PARA SISTEMAS LINUX</b>	<b>113</b>
<b>7.5.</b>	<b>ANEXO 5: CONFIGURACIÓN DEL ENTORNO VIRTUAL PARA DESARROLLO EN PYTHON.....</b>	<b>114</b>

## INDICE DE FIGURAS

FIGURA 1 ESQUEMA DE VIRTUALIZACIÓN.....	22
FIGURA 2 DASHBOARD DE OPENSTACK.....	28
FIGURA 3 ARQUITECTURA DE OPENSTACK.....	29
FIGURA 4 ESTRUCTURA DE VPCLOUD (VMWARE, 2015).....	32
FIGURA 5 TOPOLOGÍA SIMPLIFICADA DE IMPLEMENTACIÓN .....	40
FIGURA 6 TOPOLOGÍA DE PRUEBAS PARA LA MIGRACIÓN DE MÁQUINAS VIRTUALES ENTRE NUBES OPENSTACK Y VPCLOUD .....	41
FIGURA 7 TOPOLOGÍA DE RED IMPLEMENTADA EN VPCLOUD .....	42
FIGURA 8 TOPOLOGÍA DE RED IMPLEMENTADA EN OPENSTACK.....	42
FIGURA 9 ARQUITECTURA DE LA HERRAMIENTA DE SOFTWARE PARA MIGRAR INSTANCIAS.....	49
FIGURA 10 ARQUITECTURA CLIENTE SERVIDOR.....	50
FIGURA 11 ARQUITECTURA N-CAPAS .....	51
FIGURA 12 DIAGRAMA DE BLOQUES DE FRAMEWORK DE MIGRACIÓN .....	53
FIGURA 13 DIAGRAMA DE BLOQUES ENCRIPCIÓN DE CREDENCIALES POR SESIÓN.....	54
FIGURA 14 DIAGRAMA DE FLUJO-> FRAMEWORK.....	55
FIGURA 15 FRAMEWORK: VENTANA 1 .....	58
FIGURA 16 FRAMEWORK: VENTANA 2 .....	59
FIGURA 17 FRAMEWORK: VENTANA 3 .....	60
FIGURA 18 AMBIENTE DE MIGRACIÓN ENTRE NUBES .....	61
FIGURA 19 CARGAS DE TRABAJO EN OPENSTACK [18] .....	64
FIGURA 20 DIAGRAMA CONEXIÓN ENTRE VMWARE VPCLOUD Y OPENSTACK.....	73
FIGURA 21 INSTANCIA MIGRADA PROCESO DE MIGRACIÓN 1.....	77
FIGURA 22 CONSOLA INSTANCIA MIGRADA PROCESO DE MIGRACIÓN 1.....	77
FIGURA 23 CONSOLA DE INSTANCIA PROCESO DE MIGRACION 2 .....	80
FIGURA 24 CONSOLA RESULTANTE PROCESO DE MIGRACION 3 .....	82
FIGURA 25 COMANDO CONFIGURACIÓN DE ARCHIVO INTERFACE .....	93

FIGURA 26 ASIGNACIÓN DE LA IP ESTÁTICA .....	93
FIGURA 27 ACTUALIZACIÓN DE PAQUETES CENTOS.....	94
FIGURA 28 ACTUALIZACIÓN SISTEMA OPERATIVO .....	94
FIGURA 29 AGREGACIÓN DE REPOSITORIOS DE MARIADB .....	94
FIGURA 30 APERTURA DE ARCHIVO DE CONFIGURACIÓN DE SELINUX .....	95
FIGURA 31 MODIFICACIÓN DE ARCHIVO SELINUX .....	95
FIGURA 32 DETENER SERVICIO FIREWALLD.....	95
FIGURA 33 DESHABILITACIÓN SERVICIO FIREWALLD.....	96
FIGURA 34 DETENER SERVICIOS POSTFIX.....	96
FIGURA 35 DESHABILITACIÓN DE SERVICIO POSTFIX .....	96
FIGURA 36 DETENER SERVICIOS NETWORKMANAGER .....	96
FIGURA 37 DESHABILITACIÓN DEL SERVICIO NETWORKMANAGER.....	96
FIGURA 38 HABILITACIÓN DE SERVICIO NETWORK .....	97
FIGURA 39 INICIO DE SERVICIO NETWORK .....	97
FIGURA 40 CREACIÓN DE CERTIFICADO .....	97
FIGURA 41 CREACIÓN DEL KEY PARA EL CERTIFICADO CREADO.....	97
FIGURA 42 FIRMA DEL CERTIFICADO.....	97
FIGURA 43 INSTALACIÓN DE OPENSTACK TRAIN.....	98
FIGURA 44 INSTALACIÓN DE OPENSTACK PACKSTACK.....	98
FIGURA 45 CREACIÓN DE ARCHIVO DE CONFIGURACIÓN DE OPENSTACK .....	98
FIGURA 46 APERTURA DE ARCHIVO CON NANO .....	98
FIGURA 47 ARCHIVO /ROOT/ANSWER.TXT.....	99
FIGURA 48 ARRANQUE DE PACKSTACK .....	99
FIGURA 49 FRONTEND OPENSTACK.....	100
FIGURA 50 REGISTROS DNS TIPO A .....	101
FIGURA 51 ADICIÓN ROL SERVIDOR NFS .....	102
FIGURA 52 CONFIGURACIÓN NFS SERVER.....	102
FIGURA 53 SELECCIÓN PLANTILLA OVF.....	103
FIGURA 54 ASIGNACIÓN NOMBRE VM .....	104
FIGURA 55 SELECCIÓN POOL DE RECURSOS .....	104
FIGURA 56 VALIDACIÓN PLANTILLA OVF.....	104

FIGURA 57 SELECCIÓN TIPO CONFIGURACIÓN.....	105
FIGURA 58 SELECCIÓN RECURSO DE ALMACENAMIENTO.....	106
FIGURA 59 SELECCIÓN INTERFACES RED .....	106
FIGURA 60 CONFIGURACIÓN FINAL PARÁMETROS .....	107
FIGURA 61 INTERFAZ ADMINISTRACIÓN VCLLOUD DIRECTOR.....	108
FIGURA 62 ORGANIZACIONES VCLLOUD.....	109
FIGURA 63 VDC DE ORGANIZACIÓN VCLLOUD.....	109
FIGURA 64 INGRESO NOMBRE DE ORGANIZACIÓN .....	110
FIGURA 65 CENTRO DE DATOS VCLLOUD .....	110
FIGURA 66 MÁQUINAS VIRTUALES VCLLOUD .....	111
FIGURA 67 ARCHIVO DE ACCESO API OPENSTACK .....	112
SCRIPT 1 INSTALACIÓN DRIVERS SISTEMAS LINUX .....	113
FIGURA 68 CREACIÓN ENTORNO VIRTUAL PYTHON .....	114
FIGURA 69 ACTIVACIÓN ENTORNO VIRTUAL .....	114
FIGURA 70 INSTALACIÓN LIBRERÍAS MIGRACION .....	114

## INDICE DE TABLAS

TABLA 1 TIPOS DE SERVICIO PROPORCIONADOS EN CLOUD COMPUTING (INCIBE, 2013) .....	25
TABLA 2 MODELOS DE DESPLIEGUE DE CLOUD COMPUTING (INCIBE, 2013) .....	26
TABLA 3 CARACTERÍSTICAS DE SERVIDORES Y EQUIPOS FÍSICOS .....	43
TABLA 4 CUADRO DE CARACTERÍSTICAS QUE DISPONE OPENSTACK (OPENSTACK, 2020) .....	63
TABLA 5 CARACTERÍSTICAS DEL SERVIDOR OPENSTACK .....	65
TABLA 6 CARACTERÍSTICAS DE VCLLOUD (VMWARE, 2017).....	67
TABLA 7 CARACTERÍSTICAS DE VMWARE ESXI .....	69
TABLA 8 CARACTERÍSTICAS DE VMWARE VCENTER.....	69
TABLA 9 CARACTERÍSTICAS DE VMWARE CLOUD DIRECTOR .....	69
TABLA 10 CARACTERÍSTICAS DE MÁQUINA VIRTUAL A MIGRAR NUMERO 1 .....	74
TABLA 11 CARACTERÍSTICAS DE MÁQUINA VIRTUAL A MIGRAR NUMERO 2 .....	75
TABLA 12 CARACTERÍSTICAS DEL HOST DEL FRAMEWORK.....	75
TABLA 13 DISTRIBUCIÓN TAREAS- TIEMPOS PROCESO DE MIGRACIÓN 1.....	76
TABLA 14 CONSUMO DE RECURSOS PROCESO DE MIGRACION 1.....	78
TABLA 15 DISTRIBUCIÓN TAREAS- TIEMPOS PROCESO DE MIGRACIÓN 2.....	79
TABLA 16 CONSUMO DE RECURSOS PROCESO DE MIGRACIÓN 2.....	80
TABLA 17 DISTRIBUCIÓN TAREAS-TIEMPOS PROCESO DE MIGRACIÓN 3.....	81
TABLA 18 CONSUMO DE RECURSOS PROCESO MIGRACION 3 .....	82

## RESUMEN

En la actualidad la nube se ha convertido en el sistema de provisión de recursos de las empresas que requieren infraestructura y máquinas virtuales para el crecimiento de sus aplicaciones, bases de datos y procesos distribuidos, las empresa medianas y pequeñas no pueden incurrir en gastos de hardware para desarrollo de aplicaciones o procesos temporales y por estas razones valoran el uso de la nube. No obstante, una de las preocupaciones latentes para los administradores de infraestructura son los procesos de migración entre plataformas que ofrecen servicios de nube, más aún si se requiere migrar los servicios en producción desde una plataforma de pago como lo es vCloud hacia una plataforma open source como Openstack. En el presente documento se explicará el funcionamiento de un aplicativo de automatización (Framework) desarrollado bajo el lenguaje de programación Python para integrar y automatizar este proceso de migración y hacerlo transparente al usuario o administrador de infraestructura que opte por realizar el proceso de migración en frío, desde la nube propietaria hacia la nube código abierto antes mencionadas, siendo el aplicativo (Framework) desarrollado, capaz de interactuar entre las distintas plataformas transfiriendo entre sí las máquinas virtuales con sus almacenes de datos y adaptando las máquinas virtuales a los requisitos de compatibilidad de la nube opensource de destino, permitiendo obtener un software que es capaz de mover los recursos virtualizados obviando inversiones económicas considerables por uso o ejecución de aplicativos propietarios.

## **ABSTRACT**

Currently, the cloud has become the resource provision system of companies that require infrastructure and virtual machines for the growth of their applications, databases and distributed processes, medium and small companies cannot incur hardware expenses for application development or temporary processes and for these reasons they value the use of the cloud. However, one of the latent concerns for infrastructure administrators is the migration processes between platforms that offer cloud services, especially if it is required to migrate production services from a payment platform such as vCloud to an open source platform. like Openstack. This document will explain the operation of an automation application (Framework) developed under the Python programming language to integrate and automate this migration process and make it transparent to the user or infrastructure administrator who chooses to perform the cold migration process. from the proprietary cloud to the open source cloud, being the application developed, capable of interacting between the different platforms, transferring the virtual machines with their data warehouses among themselves and adapting the virtual machines to the compatibility requirements of the target opensource cloud, allowing to obtain a software that is capable of moving the virtualized resources avoiding considerable financial investments for the use or execution of proprietary applications.

# CAPITULO I

## INTRODUCCIÓN Y REVISIÓN DEL ESTADO DEL

### ARTE

#### 1.1. INTRODUCCIÓN

En la actualidad todos los centros de datos de las empresas varían en su dimensionamiento dependiendo de la capacidad económica de la empresa y del giro de negocio. En estos centros de datos se instalan y ejecutan diferentes tipos de servicios y aplicaciones informáticos, indispensables para el desarrollo diario de la empresa; en donde, los factores tecnológicos más relevantes son la virtualización, los tipos de hipervisores y el soporte para alta disponibilidad y redundancia. Sin embargo, para contar con estas características es necesario la inversión de onerosas cantidades de dinero que limitan a los administradores de red en sus labores cotidianas y sobre todo en el escalamiento de soluciones. Pero, qué sucede cuando una empresa no está en la capacidad de realizar inversiones para adquisición, puesta en marcha y crecimiento de servicios, ya sean éstas limitadas por costos de licenciamiento o infraestructura. Una alternativa a esta problemática podría ser el optar por el camino hacia soluciones de computación en la nube de código abierto.

En este entorno tecnológico, los servicios de computación en la nube persiguen un objetivo principal, el de permitir a los usuarios la migración de sus datos y aplicaciones informáticas sin que se genere un impacto en el rendimiento de sus servicios o aplicaciones. De esta manera se lograría reducir los costos de operación, al tiempo que se aumenta la escalabilidad y la disponibilidad, como una de las características más relevantes de una nube, al ser capaces de ajustarse a los requerimientos de los usuarios y demandando recursos de cómputo de forma

dinámica. Para optar por esta vía se necesita de software que permita mover las máquinas virtuales que contienen los servicios y aplicaciones informáticas en producción de la empresa. Este proceso denominado migración, actualmente, es posible de realizar de manera automatizada adquiriendo software licenciado, cuyo costo es muy elevado y sin disponer de licenciamiento perpetuo, sino con modalidad de suscripción mensual o anual. Entre estas aplicaciones de migración se puede tomar como ejemplo a “Coriolis-Cloud Migration as a Service” un servicio capaz de realizar esta operación de migración a precios variables que van desde los 5000 dólares en su versión base hasta los 25000 dólares anuales en su versión pro. Otra de las herramientas disponibles para estos procesos es “ATAMotion” de Deloitte misma que para una cotización requiere especificación de los recursos para el dimensionamiento económico, siendo cualquiera de las dos elecciones un camino hacia una inversión económica considerable.

## **1.2. OBJETIVOS**

### **1.2.1. OBJETIVO GENERAL**

Automatizar y simplificar los procesos de migración de Máquinas Virtuales entre una Nube Propietaria(vCloud) hacia una Nube OpenSource(OpenStack), generando un framework que permita garantizar una correcta movilidad de VMs entre las plataformas de cloud.

### **1.2.2. OBJETIVOS ESPECÍFICOS**

OE1. Definir los requerimientos, establecer el diseño y construir el software para migración entre las plataformas de virtualización.

OE2. Desplegar el ambiente de simulación de cloud propietaria (vCloud) y cloud opensource (Openstack).

OE3. Establecer un protocolo de pruebas para evaluar la funcionalidad del framework en un ambiente de producción simulado bajo condiciones de operación controladas.

OE4. Analizar resultados y establecer el manual de procesos para uso del framework desarrollado para la migración tecnológica a Computación en la Nube

### **1.3. MARCO TEÓRICO**

En este apartado se presenta nuestro análisis literario a los conceptos clave sobre la solución planteada en esta tesis.

“La infraestructura informática empresarial típica consistía en servidores potentes y muy caros. El sistema operativo era instalado directamente sobre el hardware del servidor, y la mayoría de los servidores alojan múltiples aplicaciones dentro del mismo sistema operativo sin proporcionar aislamiento físico o virtual” (Gorelik, 2013). Debido a que era difícil mover y balancear rápidamente la carga de las aplicaciones entre servidores, provocando que los recursos del servidor no se utilicen de la manera más eficiente; Todo este paradigma cambia con la aparición del Cloud Computing y con una de sus características más relevantes “la elasticidad”, al tener la capacidad de ajustarse a la demanda de los usuarios. A mayor demanda de requisitos de cómputo, el usuario puede solicitar mayor capacidad de cálculo, provocando el despliegue automático de nuevas instancias sobre la infraestructura física existente. Por el contrario, si la demanda de cómputo es baja, el usuario puede decidir prescindir de parte de su infraestructura, ajustándose de forma dinámica.

El Cambio importante que se dio para comenzar la era de cloud computing fue la introducción de Virtualización.

#### **1.3.1. VIRTUALIZACIÓN**

La virtualización es tanto la creación de muchos recursos virtuales a partir de un recurso físico, como la creación de un recurso virtual a partir de uno o más recursos físicos. Acorde a Virtualization 2.0 de Tutorials Point decimos que “La virtualización es la "capa" de tecnología que va entre el hardware físico de un dispositivo y el sistema operativo para crear una o más

copias del dispositivo. Esta virtualización mejora la utilización de los recursos y la eficiencia energética, lo que ayuda a reducir sustancialmente los gastos generales de mantenimiento del servidor y proporciona una recuperación rápida ante desastres y una alta disponibilidad. La virtualización ha sido muy importante para la computación en la nube, ya que aísla el software del hardware y, por lo tanto, proporciona un mecanismo para reasignar rápidamente las aplicaciones a través de los servidores en función de las demandas computacionales” (Tutorials Point, 2017).

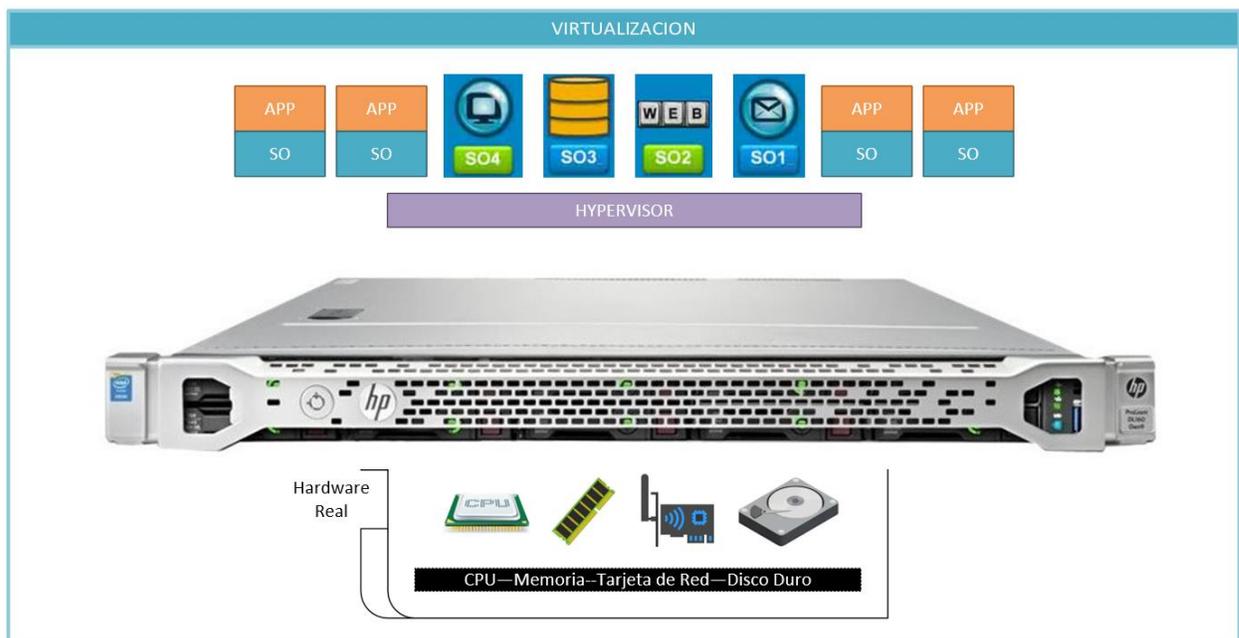


Figura 1 Esquema de Virtualización

### 1.3.1.1.TIPOS DE VIRTUALIZACIÓN

La virtualización es ampliamente aplicada en distintos tipos de conceptos como:

- *Virtualización de Servidor*

“Es virtualizar la infraestructura del servidor donde no tengas que usar más servidores físicos de diferentes propósitos” (Tutorials Point, 2017).

- *Virtualización de Cliente y Escritorio*

“Es la virtualización de escritorios en el sitio del usuario. Se cambia los escritorios de los clientes por clientes delgados utilizando recursos del centro de datos” (Tutorials Point, 2017).

- *Virtualización de Servicios y Aplicaciones*

“La virtualización aísla aplicaciones de su subyacente sistema operativo y de otras aplicaciones; a fin de incrementar la compatibilidad y la manejabilidad” (Tutorials Point, 2017). Como por ejemplo los dockers.

- *Virtualización de Red*

“Como parte de la infraestructura de virtualización se tiene que podemos virtualizar redes al partir de crear virtualmente switches, routers, vlans, etc” (Tutorials Point, 2017).

- *Virtualización de Almacenamiento*

“Esto se usa ampliamente en los centros de datos donde se cuenta con un gran almacenamiento y ayuda a crear, eliminar y asignar almacenamiento a diferentes reparticiones de hardware” (Tutorials Point, 2017).

### **1.3.1.2. Ventajas y Desventajas de la Virtualización**

Virtualizar trae ventajas y posibilidades como (Villar Fernández, 2013):

- Reducir costos en prácticamente todos los campos de acción de la administración de sistemas; desde la instalación y configuración de equipos hasta los procesos de respaldo, monitoreo, gestión y administración de la infraestructura.
- Disminuir la cantidad de servidores físicos necesarios y el porcentaje de desuso de los recursos disponibles para ellos, variando su eficiencia energética.
- Posibilita centralizar y automatizar procesos cuya administración normalmente consume mucho tiempo, pudiendo aprovisionar y migrar máquinas virtuales de manera rápida y confiable, alta calidad de servicio y bajo tiempo de respuesta antes de una caída.

Al virtualizar podemos presentar ciertas desventajas como (Villar Fernández, 2013):

- Pérdida de rendimiento. La ejecución de un sistema operativo y de aplicaciones en una máquina virtual nunca ofrecerá un rendimiento igual y mucho menos superior al obtenido con la ejecución directamente sobre el servidor físico.
- Compartición del servidor. Una de las principales ventajas del uso de la virtualización, es decir la redistribución de recursos físicos, puede llegar a ser una desventaja importante si no es gestionada y administrada correctamente.
- Anfitrión como único punto de fallo. Al estar todo centralizado dentro un solo recurso físico global como es el servidor, con una única falla en este anfitrión se nos caería múltiples servidores virtuales con sus respectivos servicios teniendo ello un gran impacto en la organización.

### **1.3.2. CLOUD COMPUTING “COMPUTACIÓN EN LA NUBE”**

El INCIBE (Instituto Nacional de Ciberseguridad) español en su guía sobre Cloud Computing (INCIBE, 2013), afirma que “El cloud computing, es un modelo de computación que permite al proveedor tecnológico ofrecer servicios informáticos a través de internet. De esta forma los recursos, es decir, el hardware, el software y los datos se pueden ofrecer a los clientes bajo demanda”, recalando en el artículo la sustitución de inversiones que permite el cloud computing al dejar de invertir en hardware o software para enfocar dicha inversión en servicios ofrecidos por proveedores de los cuales solamente se paga por lo que se usa.

### 1.3.2.1. Servicios en Cloud Computing

Los tipos de servicios que principalmente se proporcionan en cloud computing son los siguientes:

Tabla 1 Tipos de Servicio proporcionados en Cloud Computing (INCIBE, 2013)

SERVICIO	DESCRIPCIÓN	EJEMPLO
<i>SaaS</i>	Software como Servicio, los servicios que se proporcionan directamente para que el usuario final pueda consumirlos	Correo, escritorios remotos, etc.
<i>PaaS</i>	Plataforma como Servicio, el cual nos sirve para desplegar aplicaciones	Base de Datos, Servidor Web
<i>IaaS</i>	Un proveedor proporciona a los usuarios acceso a recursos en general como Máquinas Virtuales, Balanceadores de	Servicios Web de Amazon

	Carga, Equipos de Comunicación.	
--	---------------------------------	--

### 1.3.2.2.Despliegue de Cloud Computing

Los servicios de cloud computing son ofrecidos por parte de los proveedores en base a tres modelos de despliegue (INCIBE, 2013):

Tabla 2 Modelos de Despliegue de Cloud Computing (INCIBE, 2013)

<b>MODELO</b>	<b>DESCRIPCION</b>	<b>CARACTERISTICA</b>
<i>Publica</i>	Se comparten recursos (de almacenamiento, de proceso...) los cuales son ofrecidos como un servicio a muchos clientes por parte de un proveedor	Infraestructura disponible para el publico
<i>Privada</i>	Los recursos son entregados de forma exclusiva y privada al cliente, que tiene el control sobre el servicio	Administración de Infraestructura únicamente por parte de una organización
<i>Híbrida</i>	Combina servicios de nube privada y pública con una única administración, se gestiona desde un solo panel de gestión. Al mezclar servicios de nube privada y pública se consigue reducir los costes de nube privada	Formada por dos o más nubes privadas o públicas que interactúan entre si

### 1.3.2.3. Características de Cloud Computing

Cloud Computing tiene una amplia variedad de características, entre las cuales destacan:

- **Infraestructura Compartida.** “Utiliza un modelo de software virtualizado, que permite compartir servicios físicos, almacenamiento y capacidades de red. La infraestructura en la nube, independientemente del modelo de implementación, busca aprovechar al máximo la infraestructura disponible en varios usuarios” (Dialogic, 2017).
- **Aprovisionamiento Dinámico.** “Permite la prestación de servicios en función de los requisitos de demanda actuales. Esto se realiza automáticamente mediante la automatización del software, lo que permite la expansión y contracción de la capacidad del servicio, según sea necesario. Este escalado dinámico debe realizarse manteniendo altos niveles de confiabilidad y seguridad” (Dialogic, 2017).
- **Acceso de Red.** “Es necesario acceder a través de Internet desde una amplia gama de dispositivos, como PC, computadoras portátiles y dispositivos móviles, utilizando API basadas en estándares (por ejemplo, las basadas en HTTP)” (Vashist, 2015). “Las implementaciones de servicios en la nube incluyen todo, desde el uso de aplicaciones comerciales hasta la última aplicación en los teléfonos inteligentes más nuevos” (Dialogic, 2017).
- **Medición Gestionada.** “Utiliza la medición para administrar y optimizar el servicio y para proporcionar información de informes y facturación” (Dialogic, 2017). De esta manera, se factura a los consumidores por los servicios de acuerdo con la cantidad que realmente han utilizado durante el período de facturación.

### 1.3.3. PROYECTO OPENSTACK

OpenStack, es una plataforma de arquitectura modular capaz de administrar la infraestructura de cloud mediante componentes como Nova(Aportado por la NASA), Glance, Swift(Aportado por Rackspace), Horizon, Keystone, Neutron; en el artículo “*OpenStack: Toward an Open Source Solution for Cloud Computing*”, los investigadores Omar Sefraoui, Mohammed Aissaoui y Mohsine Eleuldj nos presentan el detalle del sistema funcional y arquitectónico de OpenStack con su respectivo estudio con otras soluciones en cuanto a servicios y características ofrecidas destacando su compatibilidad con Amazon EC2/S3 por medio de la interacción entre APIs; además mediante el hipervisor nativo KVM se obtienen beneficios como la aceleración en los procesos de virtualización lo cual es ventajoso pues la contextualización de las máquinas virtuales demanda menos tiempo (Sefraoui, Aissaoui, & Eleuldj, 2012).

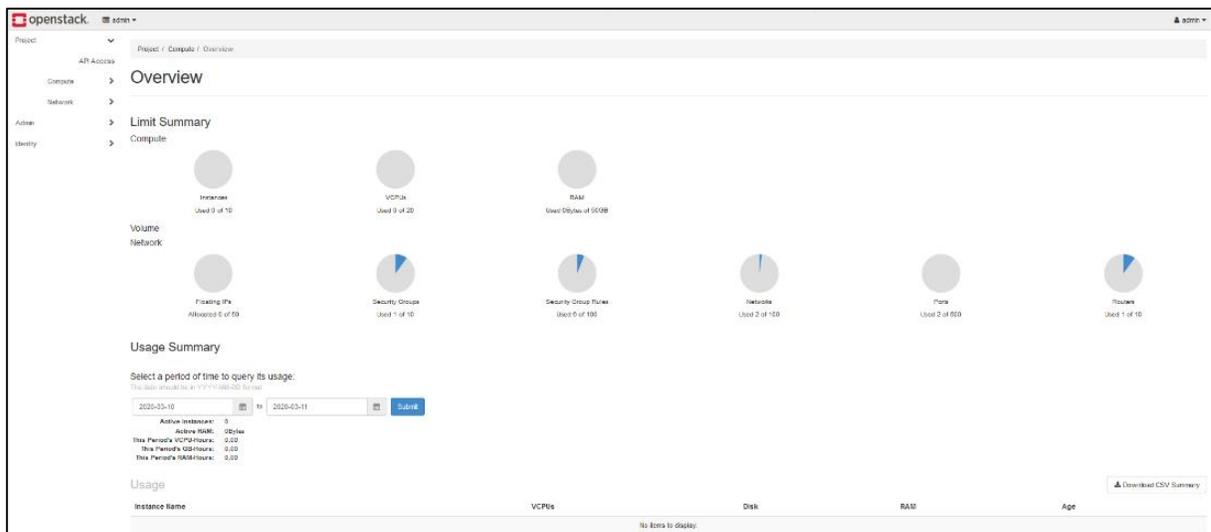


Figura 2 Dashboard de OpenStack

#### 1.3.3.1.Arquitectura de OpenStack

OpenStack está dado por distintos componentes, de los cuales podemos decir lo siguiente (Barrett, et al., 2018):

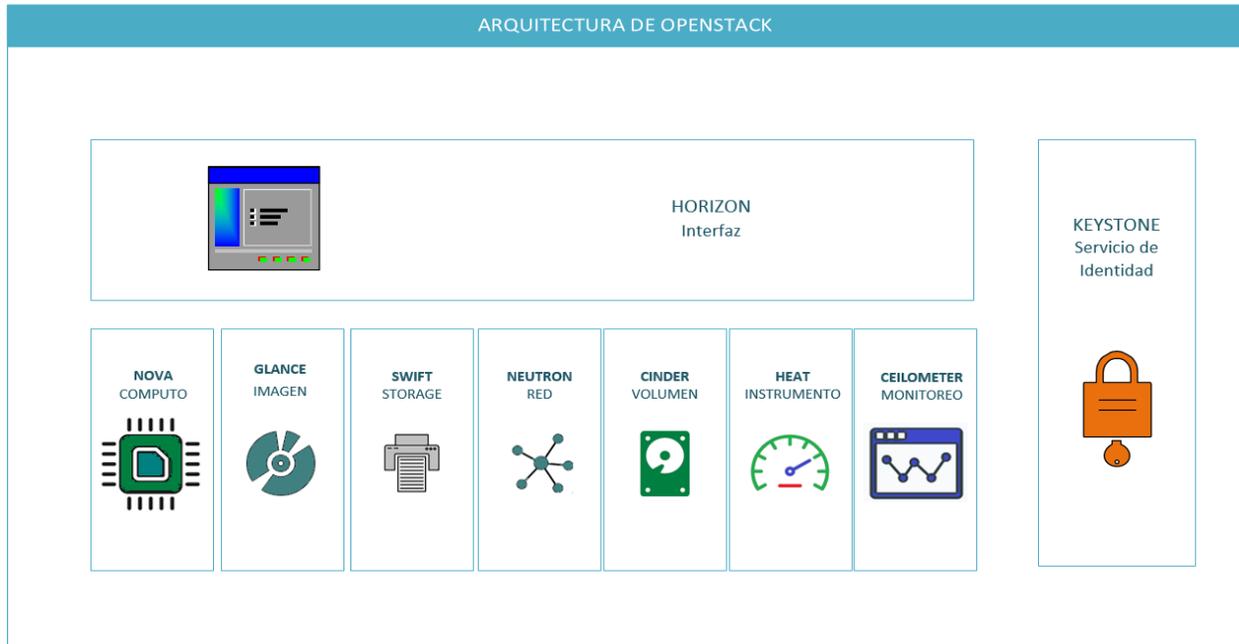


Figura 3 Arquitectura de OpenStack

#### 1.3.3.1.1. Nova

“Gestiona el ciclo de vida de las instancias informáticas en un entorno OpenStack. Las responsabilidades incluyen máquinas de desove, programación y desmantelamiento bajo demanda” (Red Hat, 2021).

#### 1.3.3.1.2. Glance

“Almacena y recupera imágenes de disco de máquina virtual. OpenStack Compute hace uso de esto durante el aprovisionamiento de instancias” (Red Hat, 2021).

#### 1.3.3.1.3. Swift

“Almacena y recupera objetos de datos no estructurados arbitrarios a través de una API RESTful basada en HTTP. Es altamente tolerante a fallas con su replicación de datos y

arquitectura escalable. Su implementación no es como un servidor de archivos con directorios montables” (Red Hat, 2021).

#### **1.3.3.1.4. Neutron**

“Permite la conectividad de red como un servicio para otros servicios de OpenStack, como OpenStack Compute. Tiene una arquitectura conectable que admite muchos proveedores y tecnologías de redes populares. Proporciona una API para que los usuarios definan redes y servicios avanzados como FWaaS, LBaaS” (Red Hat, 2021).

#### **1.3.3.1.5. Cinder**

“Proporciona almacenamiento en bloque persistente para instancias en ejecución. Su arquitectura de controlador conectable facilita la creación y gestión de dispositivos de almacenamiento en bloque” (Red Hat, 2021).

#### **1.3.3.1.6. Heat**

“Orquesta múltiples aplicaciones compuestas en la nube utilizando el formato nativo de plantilla de orquestación de calor (HOT) o el formato de plantilla AWS CloudFormation, a través del recurso nativo de OpenStack con API REST y de consultas de API compatibles con CloudFormation” (Red Hat, 2021).

#### **1.3.3.1.7. Ceilometer**

“Es un servicio de recopilación de datos que proporciona la capacidad de normalizar y transformar datos en todos los componentes principales actuales de OpenStack con trabajo en curso para admitir futuros componentes de OpenStack” (Red Hat, 2021).

#### **1.3.3.1.8. Keystone**

“Proporciona un servicio de autenticación y autorización para los servicios OpenStack. Proporciona un catálogo de puntos finales para todos los servicios de OpenStack” (Red Hat, 2021).

#### **1.3.3.1.9. Horizon**

“Proporciona un portal de autoservicio basado en la web para interactuar con los servicios subyacentes de OpenStack, como iniciar una instancia, asignar direcciones IP y configurar controles de acceso” (Red Hat, 2021).

#### **1.3.4. VMWare vCLOUD DIRECTOR**

VMWare vCloud Director es un proyecto de infraestructura de virtualización que provee acceso a los miembros de una respectiva organización a través de una interfaz web, misma que permite interactuar con los recursos de computación y red asignados a dicha organización, estos recursos serán gestionados a través de aplicaciones virtuales(vApp) que contienen instancias donde se pueden correr servicios virtualizados como servidores o aplicaciones.

“Antes de permitir el acceso hacia el sistema de cloud, un administrador del sistema de vCloud Director debe crear la organización, asignarle recursos y proporcionar la dirección URL para acceder a la consola web. Cada organización incluye uno o más administradores de organización, que finalizan la configuración de la organización al agregar miembros y al establecer las directivas y preferencias” (VMWare, 2015). Tras terminar de configurar la organización, cada usuario que no es un administrador puede iniciar sesión teniendo la posibilidad de crear, usar y administrar máquinas virtuales y las vApp.

### 1.3.4.1. Estructura de VMWare VCloud

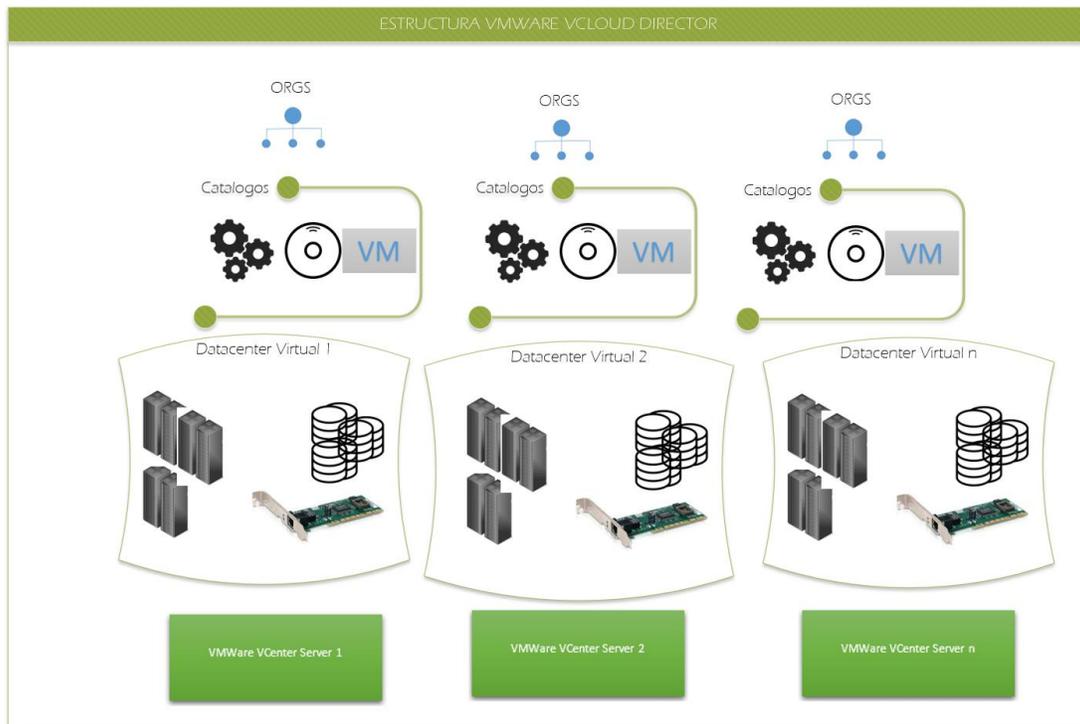


Figura 4 Estructura de VCloud (VMWare, 2015)

#### 1.3.4.1.1. Organizaciones

“Una organización es una unidad de administración para una colección de usuarios, grupos y recursos informáticos. Los usuarios se autentican a nivel de la organización, proporcionando credenciales establecidas por un administrador de la organización cuando el usuario fue creado o importado. Los administradores del sistema crean y aprovisionan organizaciones, mientras que los administradores de la organización administran los usuarios, grupos y catálogos de la organización” (VMWare, 2017).

#### 1.3.4.1.2. Usuarios y grupos

“Una organización puede contener un número arbitrario de usuarios y grupos. Los usuarios pueden ser creados localmente por el administrador de la organización o importados desde un

servicio de directorio como LDAP. Los grupos deben importarse desde el servicio de directorio” (VMWare, 2017). Los permisos dentro de una organización se controlan mediante la asignación de derechos y roles a usuarios y grupos.

#### **1.3.4.1.3. Centros de datos virtuales**

“El centro de datos virtual de una organización proporciona recursos a una organización. Los centros de datos virtuales proporcionan un entorno donde los sistemas virtuales se pueden almacenar, implementar y operar. También proporcionan almacenamiento para medios virtuales, como disquetes y CD ROM. Una organización puede tener múltiples centros de datos virtuales” (VMWare, 2017).

#### **1.3.4.1.4. Redes de centros de datos virtuales de organización**

Una red de centro de datos virtual de la organización está contenida dentro de un centro de datos virtual de la organización vCloud Director y está disponible para todas las vApps de la organización. “La red de un centro de datos virtual de una organización permite que las vApps dentro de una organización se comuniquen entre sí. La red de un centro de datos virtual de una organización se puede conectar a una red externa o aislada e interna a la organización. Solo los administradores del sistema pueden crear redes de centros de datos virtuales de la organización, pero los administradores de la organización pueden administrar las redes de centros de datos virtuales de la organización, incluidos los servicios de red que proporcionan” (VMWare, 2017).

#### **1.3.4.1.5. Redes de vApp**

“Una red de vApp está contenida dentro de una vApp y permite que las máquinas virtuales en la vApp se comuniquen entre sí. Puede conectar una red vApp a una red de centro de datos

virtual de la organización para permitir que la vApp se comunice con otras vApps en la organización y fuera de la organización, si la red del centro de datos virtual de la organización está conectada a una red externa” (VMWare, 2017).

#### **1.3.4.1.6. Catálogos**

“Las organizaciones usan catálogos para almacenar plantillas de vApp y archivos multimedia. Los miembros de una organización que tienen acceso a un catálogo pueden usar las plantillas vApp del catálogo y los archivos multimedia para crear sus propias vApps. Los administradores de organizaciones pueden copiar elementos de catálogos públicos a su catálogo de organización” (VMWare, 2017).

### **1.3.5. MIGRACIÓN DE MÁQUINAS VIRTUALES**

Acorde al artículo de Wang y Thalheim, se puede decir que la migración de datos (Thalheim & Wang, 2012) es el proceso de mover datos de fuentes de datos heredadas de un sistema heredado a nuevas fuentes de datos de un sistema de destino, en el que los sistemas heredados y nuevos tienen estructuras de datos diferentes.

La migración de aplicaciones existentes (Josyula, Orr, & Page, 2011; Raj, Mangal, Savitha, & Salvi S, 2020) a una nueva plataforma de servicios requiere una extensa investigación y planificación en relación no solo con la factibilidad técnica sino también con la factibilidad con respecto a las restricciones operativas. Estas restricciones de operación de funcionamiento pueden incluir al hipervisor base del sistema de virtualización, drivers, sistemas de virtualización de redes y almacenamiento, entre otros.

#### **1.3.5.1. Proceso Migración de VMWare a OpenStack**

- Primero, debe exportar su VM. La forma más sencilla de hacerlo es exportando un OVA usando vCenter UI (OVA es el formato estándar para empaquetar máquinas virtuales).
- En segundo lugar, “la VM necesita ser convertida KVM, esto significa convertir los volúmenes, los metadatos y reemplazar los controladores dentro de las VM invitadas” (Raj, Mangal, Savitha, & Salvi S, 2020). La conversión completa se puede hacer usando virt-v2v. “Virt-v2v convierte invitados de un hipervisor extranjero para que se ejecuten en KVM. Puede leer invitados de Linux y Windows que se ejecutan en VMware, Xen, Hyper-V y algunos otros hipervisores, y convertirlos a KVM administrado por OpenStack, o varios otros objetivos” (Raj, Mangal, Savitha, & Salvi S, 2020).
- “El resultado de la conversión es una imagen QCOW modificada, que luego se puede cargar en su implementación de OpenStack” (Raj, Mangal, Savitha, & Salvi S, 2020).
- “Más adelante podrá crear una máquina virtual utilizando esta imagen, conectarla a las redes relevantes, grupos de seguridad e IP flotante” (Raj, Mangal, Savitha, & Salvi S, 2020).

### **1.3.6. Trabajos Relacionados**

En (Jamshidi, Ahmad, & Pahl, 2014) se busca identificar, clasificar en un sentido general y comparar de una forma sistemática la investigación existente sobre la migración en la nube. En todo este estudio se da como resultado una base de conocimientos de soluciones para la migración a la nube identificando brechas y direcciones para futuras investigaciones. Esta investigación concluye que la investigación sobre migración a la nube está en sus primeras etapas de madurez, pero avanzando, también logra identificar la necesidad de un framework de migración para mejorar el proceso de migración, así como también la falta de herramientas de soporte para automatizar las tareas de migración.

Así mismo, según lo expuesto en (Gholami, Low, Beydoun, & Daneshggar, 2016), “Migrar Aplicaciones empresariales a entornos de nube es una tarea estratégica de TI importante y requiere un enfoque sistemático” (Gholami, Low, Beydoun, & Daneshggar, 2016). En este documento se intentó revisar y examinar los enfoques existentes de migración a la nube desde la perspectiva de los modelos de proceso. De tal manera que los investigadores proponen un marco de evaluación y lo utilizan para analizar y comparar los enfoques existentes para resaltar cada característica, similitud y diferencia clave. Este marco es una encuesta que extrae el estado del arte en la investigación sobre migración en la nube y hace un inventario de actividades importantes dándonos recomendaciones, técnicas y preocupaciones que frecuentemente están comprendidas dentro del proceso de migración. “Esto permite a los académicos y profesionales de la comunidad de computación en la nube obtener una visión general del proceso de migración a la nube. Además, la encuesta identifica una serie de desafíos que aún no han sido abordados por los enfoques existentes, desarrollando oportunidades para futuros esfuerzos de investigación” (Gholami, Low, Beydoun, & Daneshggar, 2016).

En el Artículo “Cloud Migration Metamodel” (Pamami, Jain, & Sharma, 2019), se presenta un modelo para la migración a la nube. “El propósito del documento es desarrollar un framework consolidado para el soporte de migración en la nube. El objetivo de esta investigación fue crear un proceso de referencia genérico para la migración a la nube” (Pamami, Jain, & Sharma, 2019).

Una de las temáticas más relevantes de la actualidad con respecto a nube es el concepto de “Nube Híbrida”, en el artículo “Data traffic management in a hybrid cloud composed of Openstack and Azure” (Arévalo-Cordero, et al., 2019), dando como resultado la evaluación de

la eficiencia y rendimiento en la convergencia entre una nube publica como Azure y una privada como Openstack.

## **1.4. TOPOLOGÍA DE RED PROPUESTA**

En la topología se utilizan los componentes funcionales que se detallan a continuación:

### **1.4.1. Componentes Funcionales**

En la arquitectura propuesta se encuentran los siguientes elementos:

#### **1.4.1.1.DNS**

Se instaló un Servidor de DNS (Domain Name System) sobre el sistema operativo Windows Server 2016 Standard, este servicio es utilizado en interconexiones para acceder mediante los nombres de dominio en lugar de las direcciones IP, además utiliza servidores para la resolución de dominios externos. En nuestra arquitectura cumple un rol fundamental para el correcto funcionamiento de diversos productos y servicios necesarios por las soluciones de VMware como vCenter, vCloud, NSX y la red interna general, mismas que requieren registros A de DNS y una resolución de búsqueda inversa previamente configuradas.

#### **1.4.1.2.NFS**

Los servicios de vCloud Director necesitan acceso a un recurso compartido (Servidor NFS), Network File System. este sistema puede ser configurado en cualquier sistema operativo adicionando los paquetes necesarios, para el ambiente de cloud privado hemos

implementado este servicio en una Máquina Virtual mediante roles de MS Windows en la versión 2016 Standard.

#### **1.4.1.3.Host ESXi**

VMWare ESXi, es la plataforma a nivel de centro de datos producida por VMWare para realizar virtualización. En nuestro caso usamos el hipervisor ESXi en primer lugar para desplegar Máquinas Virtuales con servicios y productos como vCenter, NSX, DNS, NFS para el posterior despliegue de vCloud Director 10.

#### **1.4.1.4.Servidor vCenter**

vCenter es la herramienta para administrar de manera centralizada tanto hosts ESXi, máquinas virtuales y recursos desde una ubicación centralizada. Nosotros desplegamos uno para manejar utilidades únicas de vCenter como movilidad entre hosts de máquinas virtuales y propiedades de red de vCenter.

#### **1.4.1.5.vMotion Network**

vMotion es una utilidad de vCenter que nos permite cambiar el recurso donde se ejecuta una máquina virtual. Nosotros configuramos una subred para esta funcionalidad pensando en nuestras máquinas virtuales.

#### **1.4.1.6.Storage Network**

Hemos configurado una subred para manejar los recursos de almacenamiento de nuestra infraestructura.

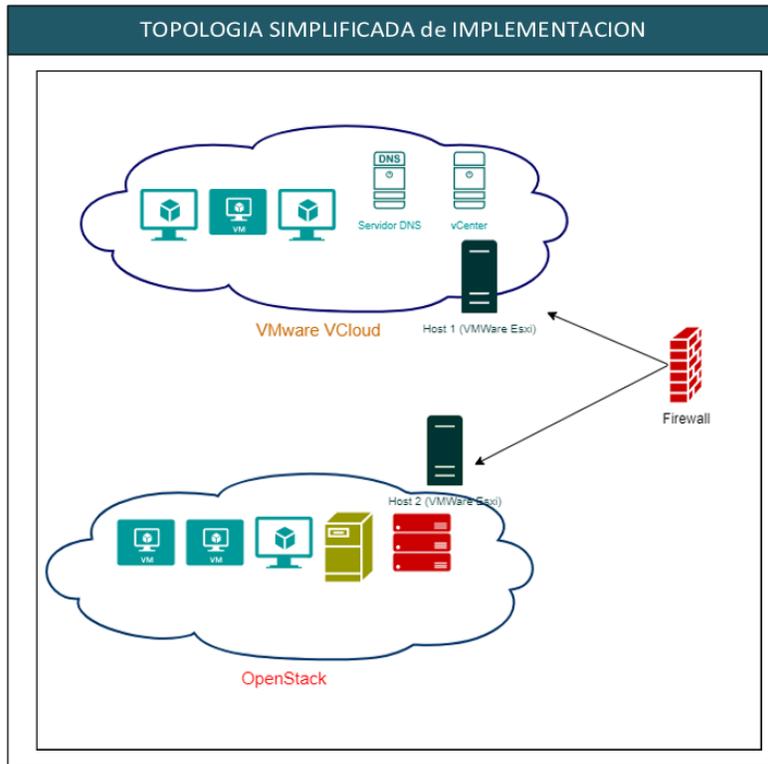
### **1.4.2. Topología**

La Topología de red propuesta para el ambiente de pruebas en nuestra temática de migración es una topología en bus (considerando que la Red del ISP no es controlable por nosotros), tanto para nuestra nube open-source Openstack como para nuestra nube propietaria vCloud. Como se ve en la Figura 6.

De manera que nuestra topología consta de tres elementos físicos: Un servidor ThinkServer de la marca Lenovo, Un servidor DL160 Gen9 de la Marca HP y un dispositivo en medio de los dos que hace las veces de enrutador de redes y firewall de la marca Mikrotik.

Así mismo consta de elementos virtualizados dentro de cada servidor:

- Dentro del Servidor de Lenovo ya mencionado de la marca HP se encuentra instalado a manera de hipervisor VMWare VSphere ESXi 6.7 y a su vez dentro del hipervisor está virtualizado Centos 7 con una integración de OpenStack de la distribución Train.
- Dentro del Servidor de HP se encuentra instalado como hipervisor VMWare VSphere ESXi 6.7 dentro de este se virtualiza los servidores de: DNS, NSX, VCenter y VCloud.



*Figura 5 Topología Simplificada de Implementación*

Podemos ver como distribuimos el tráfico de red usando diferentes subredes a fin de garantizar ciertas limitantes a la hora de seguridad, con el fin de que las subredes no puedan verse entre ellas. Así mismo, el framework deberá ubicarse en una máquina virtual independiente o en un computador externo. Véase en la Figura 6.

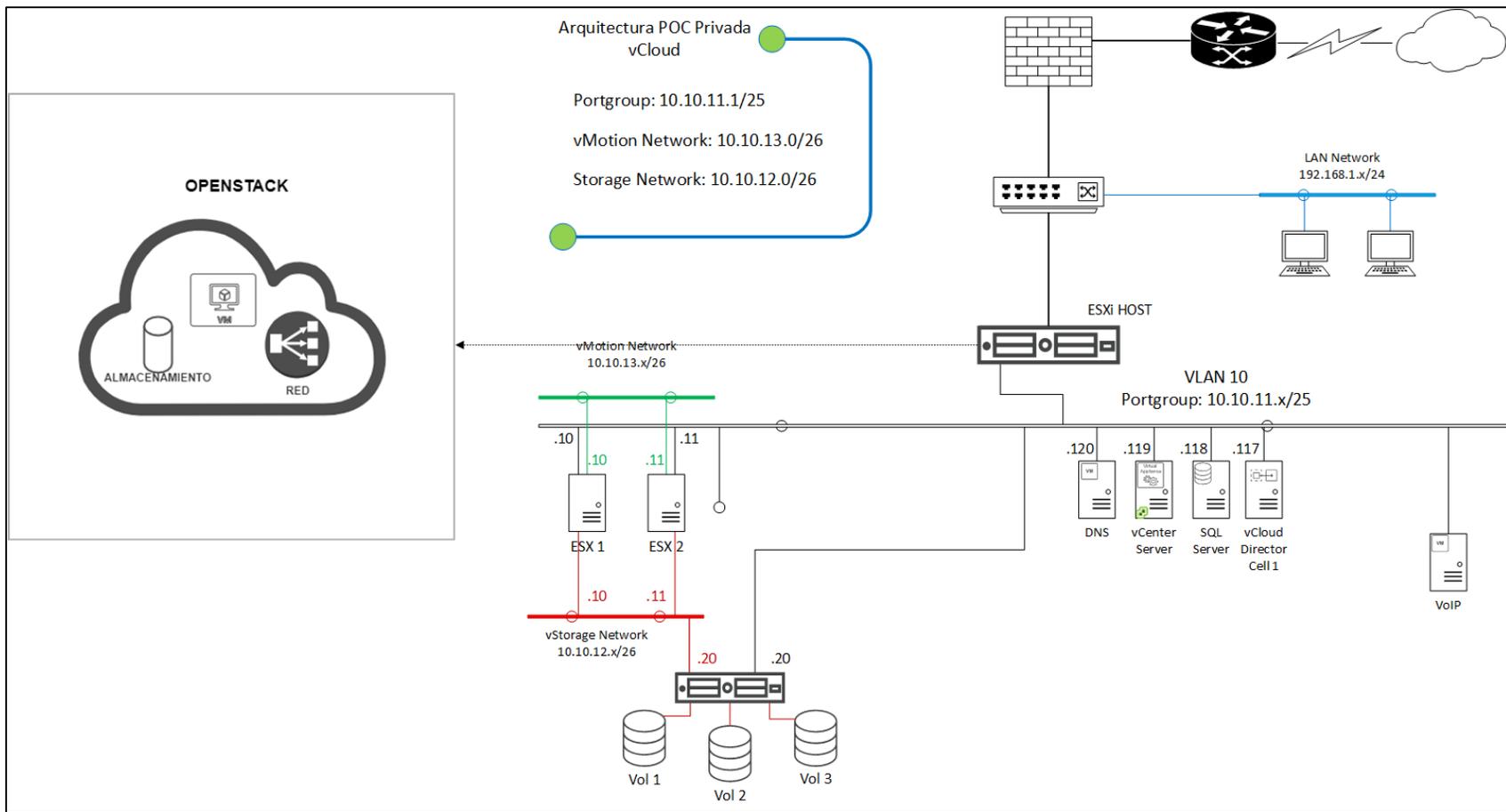


Figura 6 Topología de pruebas para la migración de máquinas virtuales entre nubes OpenStack y vCloud

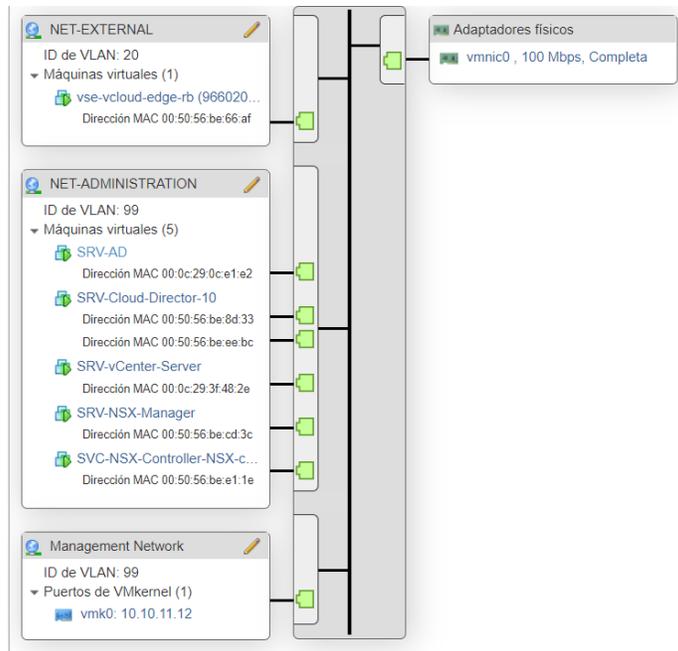


Figura 7 Topología de red implementada en vCloud

En la Figura 7, se puede observar los distintos servicios desplegados con su respectiva dirección IP a fin de poder realizar la implementación de VMWare VCloud dentro de los cuales están: DNS, NSF, NSX, VMWare VCenter.

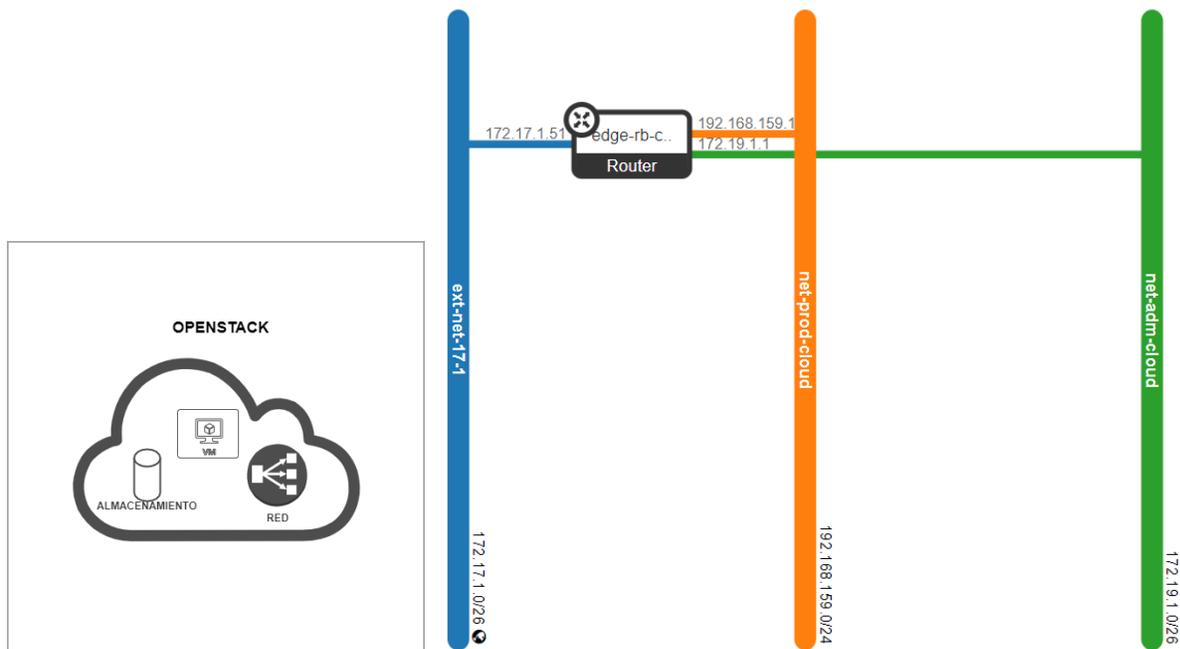


Figura 8 Topología de red implementada en OpenStack

En la Figura 8, podemos ver cómo está compuesta la topología de nuestra implementación básicamente son tres subredes que están divididas en: red externa, red de producción y red de administración.

Al terminar de explicar nuestra topología procedemos a indicar los recursos utilizados tanto en hardware como en software.

### 1.4.3. Recursos de Hardware y Software utilizados para la instalación de OpenStack y vCloud

Nuestras nubes tanto propietaria como de código abierto se alojarán dentro de equipos con las siguientes características:

Tabla 3 Características de Servidores y Equipos Físicos

<b>Marca/Modelo</b>	HP/Proliant DL160 Gen9
<b>Hipervisor</b>	VMware ESXi 6.7
<b>RAM</b>	32 GB
<b>Disco Duro</b>	2 TB
<b>Procesador</b>	Intel Xeon E5 – 2620
<b>Tipo de Procesador</b>	Número de Núcleos: 8 Velocidad de Reloj: 2.1GHz Cache: 20 MB Socket: LGA 2011-v3 Tipo de Memoria: DDR4-2133/1866/1600

## 1.5. MARCO METODOLÓGICO

### 1.5.1. Metodología

En base al trabajo a desarrollar y el alcance del mismo se ha escogido la metodología CASCADA, ya que la misma consta de 8 fases (concepción, iniciación, análisis, diseño, construcción, pruebas, despliegue y mantenimiento) que se llevan a cabo de manera sucesiva y a modo de cascada hasta su finalización.

Esta metodología nos propone como inicio plantear el tiempo adecuado para diseñar la solución del producto final debido a que en las primeras fases del proceso se puede evitar problemas que resultan en elevados costos cuando el proyecto ya estuviese en fase de desarrollo. Además de construir una documentación muy exhaustiva y si se une al equipo un nuevo colaborador, podrá comprender el proyecto leyendo la documentación.

Para poder llegar a realizar una implementación de esta índole, se tienen que realizar una serie de pasos en los cuales haremos hincapié en puntos relevantes, el mismo está detallado de la siguiente manera:

- Diagnóstico del entorno
- Selección de herramientas
- Evaluación de arquitecturas de red
- Instalación de nube propietario VCloud
- Instalación de nube publicada OpenStack
- Análisis de Requerimientos y Limitaciones
- Desarrollo de código en Python para realizar Migración entre las nubes escogidas
- Realizar Pruebas de Migración

## CAPITULO II

### ANALISIS DE REQUERIMIENTOS

#### 2.1. REQUERIMIENTOS DEL SOFTWARE

“Requisito es la disciplina de licitar, analizar, especificar, validar y gestionar los requisitos para el sistema a ser desarrollado” (Martinez, 2015). Este proceso es la forma en la que se intenta entender y solucionar un determinado problema presente en la vida real. Estas soluciones a menudo requieren generar estrategias o aplicaciones nuevas que permitan automatizar o complementar procesos preexistentes mediante correcciones o agregación de nuevas funcionalidades. El resultado de obtener los requerimientos permite validar las funciones, limitar las restricciones, establecer el alcance y adecuar funcionalidades con el usuario.

#### 2.2. ELICITACIÓN DE REQUERIMIENTOS

La solución del software permitirá solventar un determinado problema del usuario, este software deberá ser funcional y adecuado a la parte que lo requiere y para lograr esta meta se debe tener claro cuál es el problema a solventar y esto se logra a través de la comunicación con el mismo.

Para nuestro caso en particular la parte interesada en dar solución al problema son los administradores de red y específicamente nosotros quienes redactamos este documento, con esta información levantamos los requerimientos de acuerdo a las limitaciones de las plataformas de virtualización estudiadas y analizadas definimos que el software debe tener la capacidad de:

- Extraer los datos de una instancia (Máquina Virtual) del entorno de computación en la nube (vCLOUD).
- Desplegar la instancia extraída del proceso anterior en el entorno de computación en la nube (Openstack).

## **2.3. ANÁLISIS DE REQUISITOS**

Para poder establecer las especificaciones de los requerimientos es necesario realizar un análisis donde identificaremos los diferentes conflictos que puedan surgir en los requerimientos. Las clasificaciones de los requerimientos se especifican el SWEBOK, nosotros para nuestro propósito específico utilizaremos los especificados como Requerimientos Funcionales y no Funcionales.

### **2.3.1. REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES**

#### **2.3.1.1. Requisitos Funcionales**

- El sistema gestionará las credenciales proporcionadas de los dos sistemas de virtualización (Openstack y vCloud) proporcionando una conexión exitosa mediante solicitudes REST.
- El sistema generará solicitudes REST hacia las APIS de los sistemas de virtualización permitiendo obtener la información de los recursos alojados (Instancias) en vCloud Director.
- El sistema gestionará mediante solicitudes REST las instancias alojadas en vCloud director para la creación de Snapshots, reconfiguración de instancias, descarga de discos en formato vmdk, restauración de snapshots, conversión de discos.

- El sistema una vez convertido los discos enviará el resultado de los mismos en formato qcow2 hacia el sistema de Openstack, permitiendo crear volúmenes de manera automatizada mediante solicitudes REST.
- El sistema será capaz de enviar solicitudes REST crear instancias mediante los volúmenes previamente creados, asociando un puerto neutrón, un keypair, un grupo de seguridad y posterior al proceso deberá ejecutar la instancia.

### **2.3.1.2. Requisitos No Funcionales**

- El sistema debe ser desarrollado en el lenguaje de programación Python 3 para integrar las librerías existentes que integran los proyectos de virtualización.
- El sistema debe ser compatible con las plataformas Windows y Linux, en los cuales deberá poder ejecutarse de manera fluida y equilibrada.
- El sistema deberá ser capaz de migrar instancias desde vCloud hacia Openstack, las instancias podrán ejecutar cualquier servicio en los sistemas operativos como Ubuntu, Debian, Centos y Windows y podrán ejecutarse en Openstack de manera correcta.
- Se proporcionará una interfaz sencilla y simplificada que reduzca la tasa de errores cometidos por el usuario y se facilitará un manual de usuario técnico para el uso de la herramienta de migración.

### **2.3.2. ARQUITECTURA DEL SISTEMA**

Los componentes que integran el sistema se establecen en base a la funcionalidad del software, para lograr que el software cumpla con los requerimientos establecidos, se implantan los siguientes componentes:

- Clientes y Servidores: El código fuente que se ejecutará sobre un cliente o Host será el encargado de conectar a los servicios de los sistemas de cloud computing o computación en la nube mediante peticiones (API REST) a los servicios que están ejecutándose en el servidor, y estos a su vez generan una respuesta a estas peticiones. El cliente procesara toda la información extraída mediante las peticiones realizadas al servidor, se considera a este un Cliente Pesado.
- Interacciones: las llamadas a estos procesos serán independientes y podrán generar llamadas a subprocesos que tengan como funcionalidad diferentes acciones en los servicios que administran las instancias en los sistemas de cloud.

# CAPITULO III

## DISEÑO Y DESARROLLO DEL SISTEMA

### 3.1. Diseño del Sistema

Después de recopilar los datos necesarios y establecer los requerimientos del sistema, podemos establecer cuál será el esquema base para la operación del sistema a construir. Todos los componentes del mismo deberán integrarse siguiendo el esquema propuesto a continuación:

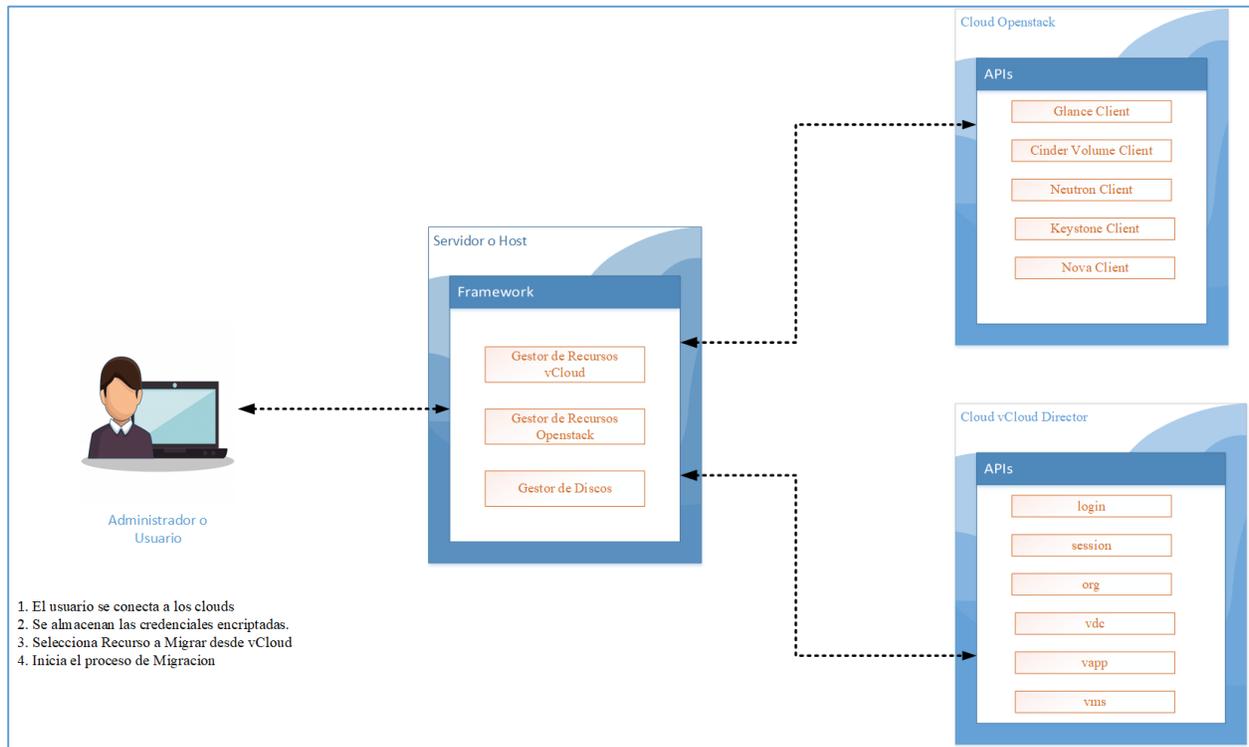


Figura 9 Arquitectura de la herramienta de software para migrar instancias

En la Figura 9 se establecen los componentes que integran la solución de software, como son el usuario o administrador de las instancias de cloud computing, este usuario debe tener permisos para poder gestionar dichas instancias, de preferencia debe ser un usuario con privilegios altos o administrativos mismo que se establece en las instancias de cloud respectivas. El sistema contara con la capacidad de administrar mediante solicitudes API REST

las distintas instancias sean estas encendido, apagado, creación de snapshots, habilitación y deshabilitación de descargas, exportación de instancias y discos, luego de obtenida los datos (Discos) el sistema tendrá las herramientas necesarias para tratar la información como conversión de discos, creación de directorios temporales, y borrado de archivos, el ultimo proceso de la herramienta será la autenticación y creación de una sesión, misma que permitirá el envío de los discos virtuales hacia el gestor de archivos glance en Openstack, mediante solicitudes API REST se creara de manera automática un volumen, un puerto de red, un keypair, un grupo de seguridad, una ip flotante y la instancia que será la encargada de asociar todos los componentes y ejecutarla en su nuevo hipervisor.

### 3.2. DESARROLLO DEL SISTEMA

Nuestro sistema de framework se basa inicialmente en una arquitectura de Cliente Servidor debido a que las clases, métodos y funciones establecidas se conectan directamente al servidor mediante solicitudes API REST para el envío y solicitud de información.

Para la validación de los métodos, funciones y clases establecemos otra arquitectura basada en Capas debido a que segmentamos el código inicial y operacional de la capa de presentación para hacer un sistema amigable al usuario.

#### 3.2.1. ARQUITECTURA CLIENTE SERVIDOR

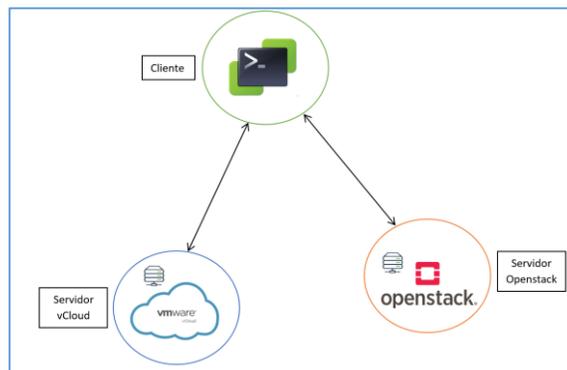


Figura 10 Arquitectura Cliente Servidor

En la Figura 10 se puede observar la distribución de los componentes en base a una arquitectura Cliente-Servidor misma que se establece para la construcción de la herramienta de migración(Framework).

- **Ciente:** Es un cliente pesado y procesa los datos de la información que recibe del servidor en este caso administra los discos virtuales entre los distintos formatos.
- **Servidor:** Esta capa tiene toda la lógica para gestionar las solicitudes API REST que genera el cliente, llama a los procedimientos necesarios para administrar las instancias.

### 3.2.2. ARQUITECTURA N-CAPAS

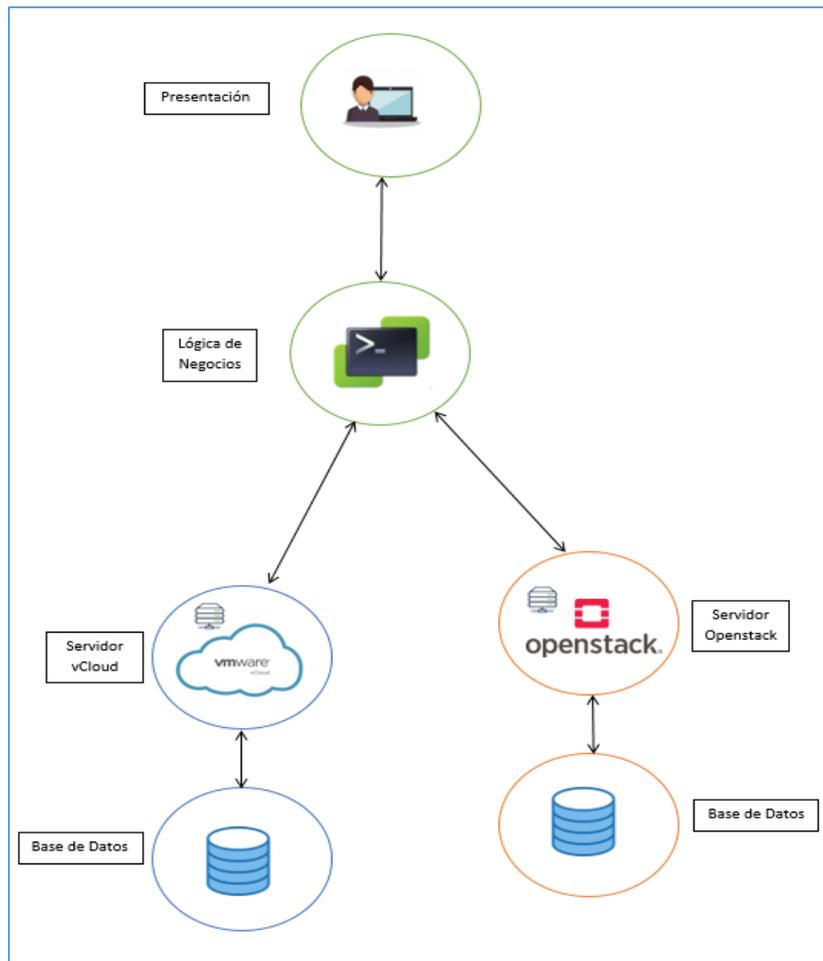


Figura 11 Arquitectura N-Capas

En la Figura 11 se puede observar, la distribución de componentes en base a una distribución N-Capas, y consta de lo siguiente:

- **Capa de Servidor:** Al igual que la descripción anterior de la Arquitectura Cliente servidor las funcionalidades de esta capa será la misma, gestionar instancias mediante las solicitudes que reciba de las capas superior.
- **Lógica de negocio:** Esta capa es la encargada de generar las reglas que permiten el funcionamiento correcto del sistema como validaciones, recibe parámetros trata la información y genera la estructura de las solicitudes que serán enviadas al servidor.
- **Entidades de negocio:** Esta entidad esta inhibida en la arquitectura de cada uno de los proyectos de cloud computing.
- **Presentación:** Es la capa que se encargará activamente de interactuar con el usuario para nuestra practica hemos generado una interfaz amigable y controlada que permitirá al usuario entender los datos de una manera simplificada.

### 3.2.3. DISEÑO DEL ESQUEMA DE FUNCIONAMIENTO

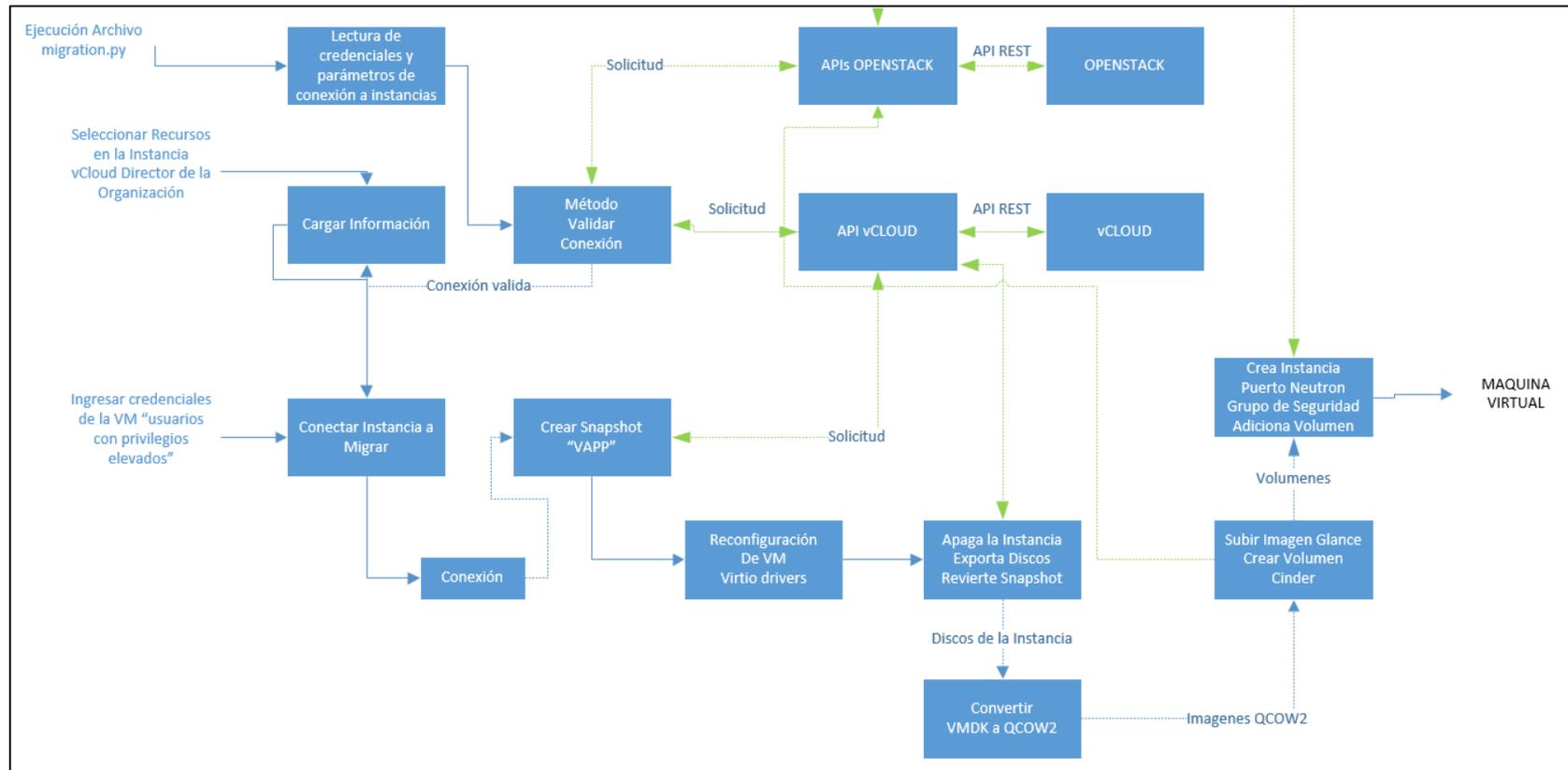


Figura 12 Diagrama de Bloques de Framework de Migración

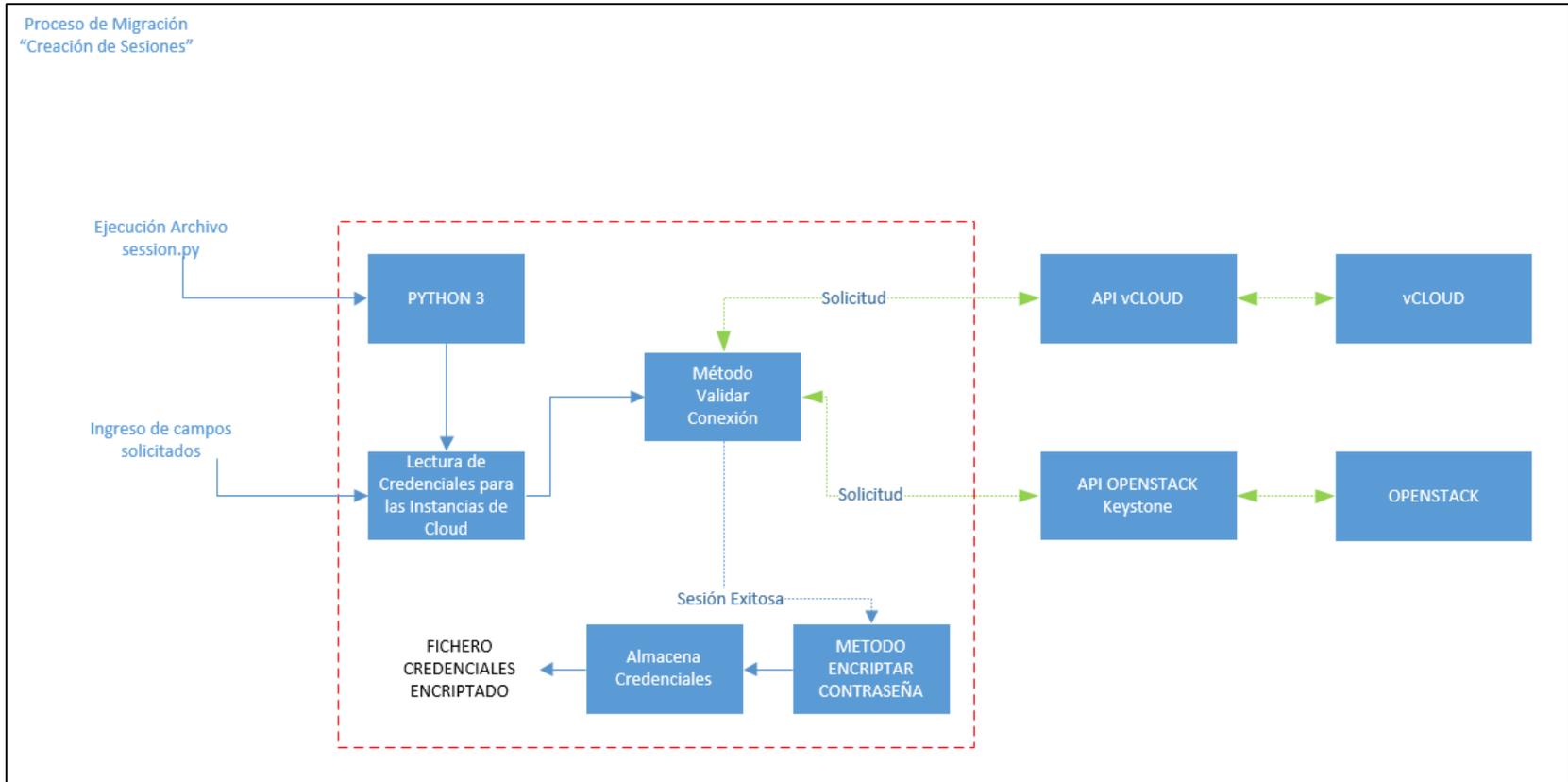


Figura 13 Diagrama de Bloques Encriptación de Credenciales por Sesión

### 3.2.4. PROCESO DE INTERACCIÓN CON EL FRAMEWORK

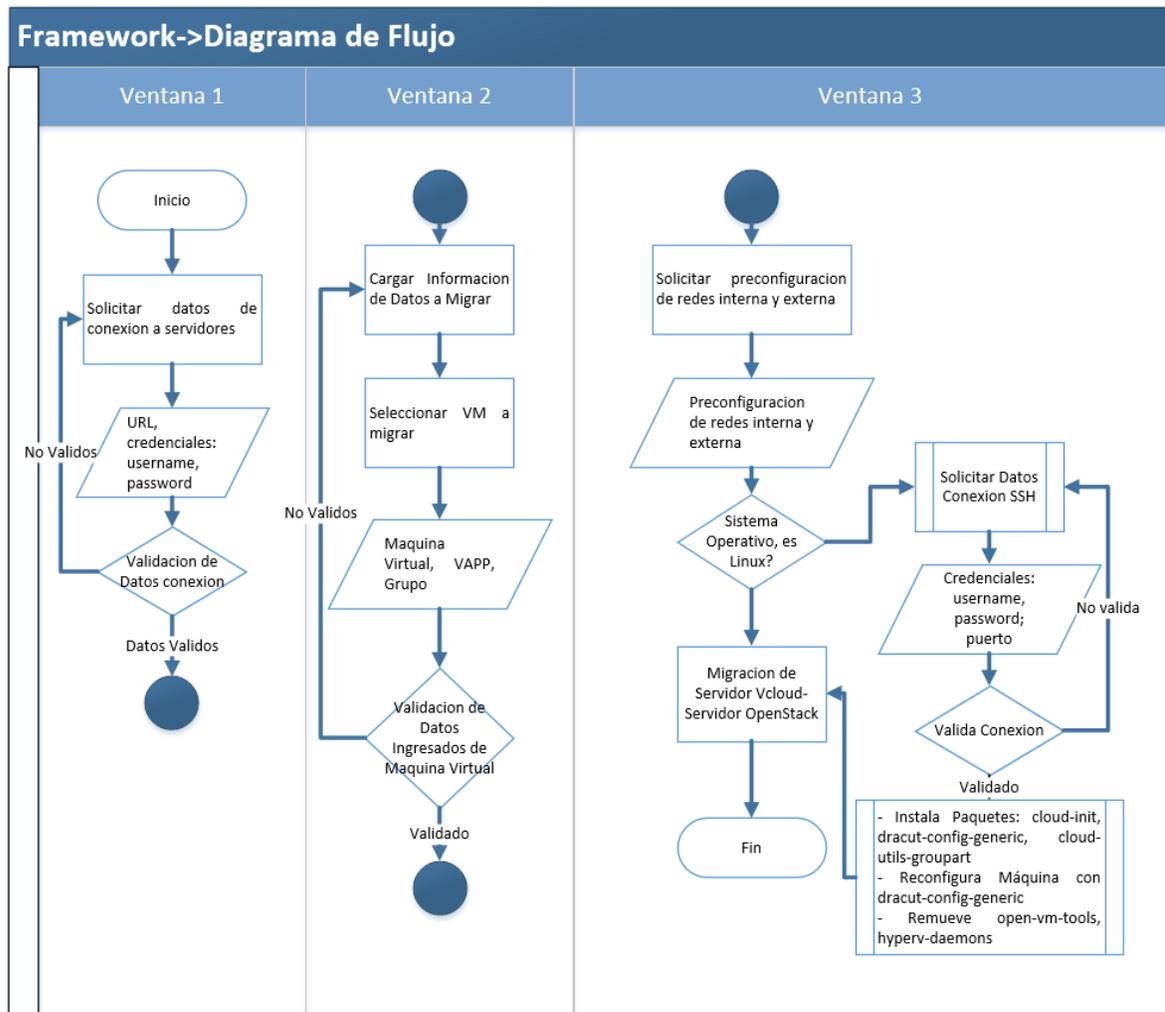


Figura 14 Diagrama de Flujo-> Framework

Para usar nuestro framework es necesario que un computador que funcione como middleware, tenga instalado Python en su versión 3.8. Este framework consta de tres ventanas donde se realizan pasos y comprobaciones para el proceso. Como se muestra en la Figura 14.

### **3.2.5. CONFIGURACIÓN DEL ENTORNO DE DESARROLLO**

La configuración del entorno de trabajo permitirá alcanzar cumplir con los más altos estándares de rendimiento de nuestro código, a continuación, especificaremos el entorno de operación donde se construirá el sistema y las herramientas para construir nuestro framework.

#### **3.2.5.1. Configuración de Python**

Para configurar el entorno de Python hemos elegido un entorno virtualizado, mismo que se detalla en el Anexo 5, y explica cómo crear este entorno de pruebas. El objetivo de esta práctica es asegurar que al desarrollar no se cometan errores que comprometan el correcto funcionamiento del sistema operativo anfitrión, ya que se requieren paquetes adicionales que complementaran la solución de software a desarrollar.

#### **3.2.5.2. Paquetes y Librerías**

Las bibliotecas preexistentes que permiten conexiones transparentes hacia los sistemas de cloud son varios, pero nos enfocaremos en los siguientes proyectos como “pyvcloud” para vCloud Director, mismo SDK que fue desarrollado en Python y es mantenido por VMware permite simplificar el proceso de obtención y envío de solicitudes REST y conectarse directamente a las APIs de vCloud Director. En el caso de Openstack existen múltiples librerías que realizan administración y gestión de este entorno de cloud pero de manera independiente y para el proyecto open source como es el caso de Openstack se cuenta con librerías como openstack-nova-client que se encarga de gestionar servicios de cómputo mediante solicitudes REST para la administración de instancias un recurso necesario para nuestro objetivo de migración, también se cuenta con una librería python-neutron-client una librería que se encarga de gestionar los recursos de red como puertos, ips flotantes necesarias para establecer

comunicación con la internet, openstack-cinder-client esta librería se enfoca en la gestión de volúmenes, la librería openstack-keystone-client es la que utilizaremos para generar sesiones mediante autenticación de usuario y contraseña, como último recurso elegido para el desarrollo del framework esta la librería openstack-glance-client la librería encargada de gestionar las imágenes en el proyecto todas estas librerías están licenciadas bajo Apache.

### **3.2.5.3. IDE**

Para codificar la herramienta y generar las clases y métodos para esta solución de software necesitaremos un IDE en nuestro caso la mejor herramienta fue el IDE Visual Studio Code de Windows adicionalmente usamos los plugin de Python.

### **3.2.5.4. Desarrollo del Sistema de Migración de Instancias desde vCloud hacia Openstack**

El administrador será quien gestione la interacción de la capa de presentación, validando que se cumplan las necesidades y que la lógica de negocio se cumpla.

#### **3.2.5.4.1. Ventana Uno (Datos de las Nubes VCloud y OpenStack)**

En la primera ventana de nuestro programa debemos ingresar los datos de conexión tanto de VMware VCloud como de OpenStack; para esto se presenta dos secciones la primera sección permitirá ingresar los datos de nuestro servidor VCloud y validarlos, una vez validados estos procedemos ingresando los datos del servidor OpenStack, los cuales podemos llenar con el archivo yaml, este archivo de configuración se puede descargar como lo indica el ANEXO 3 en la interfaz web del servidor Openstack, luego de cargar el fichero validamos la conexión este proceso generará un archivo encriptado para almacenar las credenciales de las instancias. En caso que la autenticación no tenga éxito el sistema alertará de un error y pedirá que vuelva a ingresar la contraseña.

La Interfaz de la ventana de validación de credenciales se puede ver en la Figura 15.

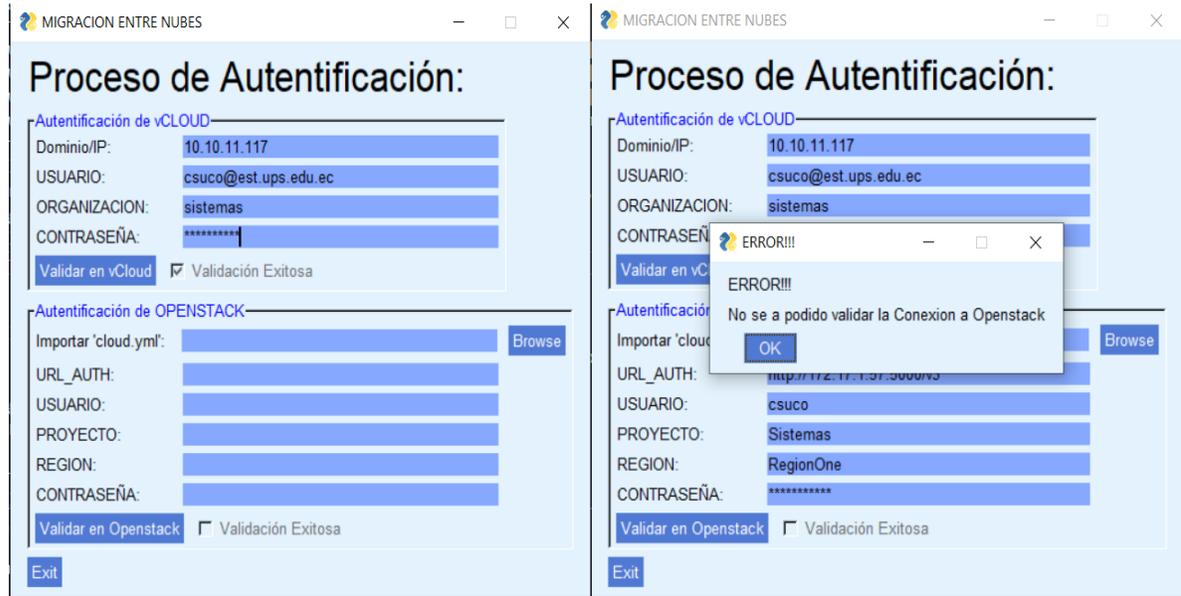


Figura 15 Framework: Ventana 1

#### 3.2.5.4.2. Ventana Dos (Selección de Máquina Virtual a Migrar)

En la segunda ventana del framework tenemos primeramente que cargar los datos, esto lo realizamos para que se nos habilite los campos con las respectivas opciones para buscar la máquina virtual que deseamos migrar. Seguido debemos seleccionar la organización, el sistema mostrará únicamente los datos a los que el usuario autenticado tiene acceso, luego seleccionaremos la vApp donde se ejecuta nuestra instancia y finalmente seleccionamos nuestra máquina virtual. Una vez seleccionados correctamente nuestros parámetros damos clic en Migrar para llegar hacia nuestra ventana final. La Interfaz de la ventana se puede ver en la Figura 16.

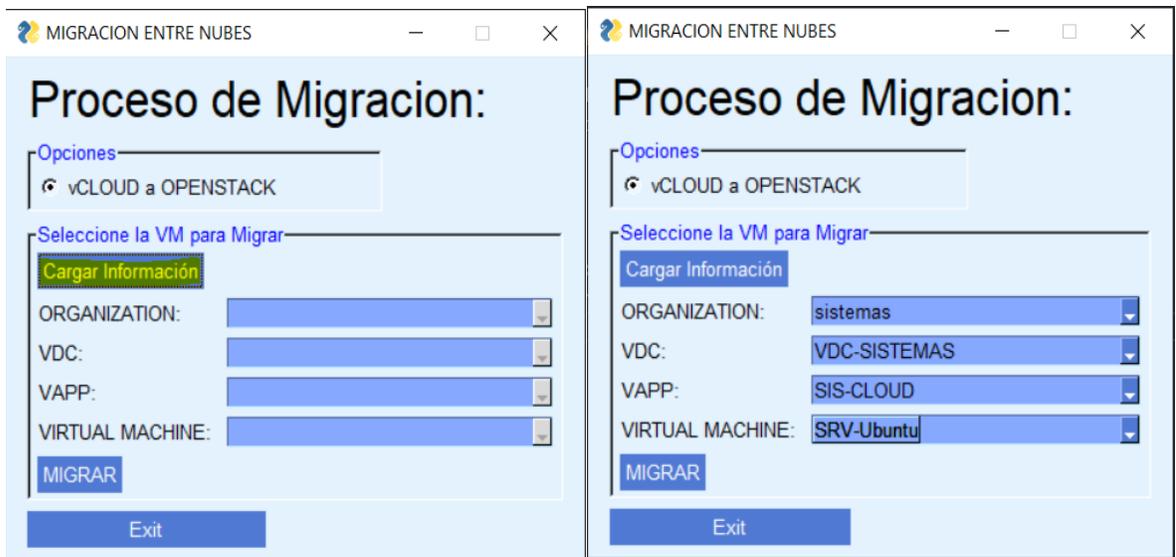


Figura 16 Framework: Ventana 2

### 3.2.5.4.3. Ventana Tres (Selección Redes e Inicio de Proceso Migratorio)

En la ventana final de nuestro framework solamente debemos seleccionar las redes del destino ya sea OpenStack o VMWare VCloud; hecho ya esto debemos iniciar el proceso dando clic en Run. De esta manera se da comienzo la migración y veremos el procedimiento de la misma en la ventana. La Interfaz de la Ventana se puede ver en la Figura 17.

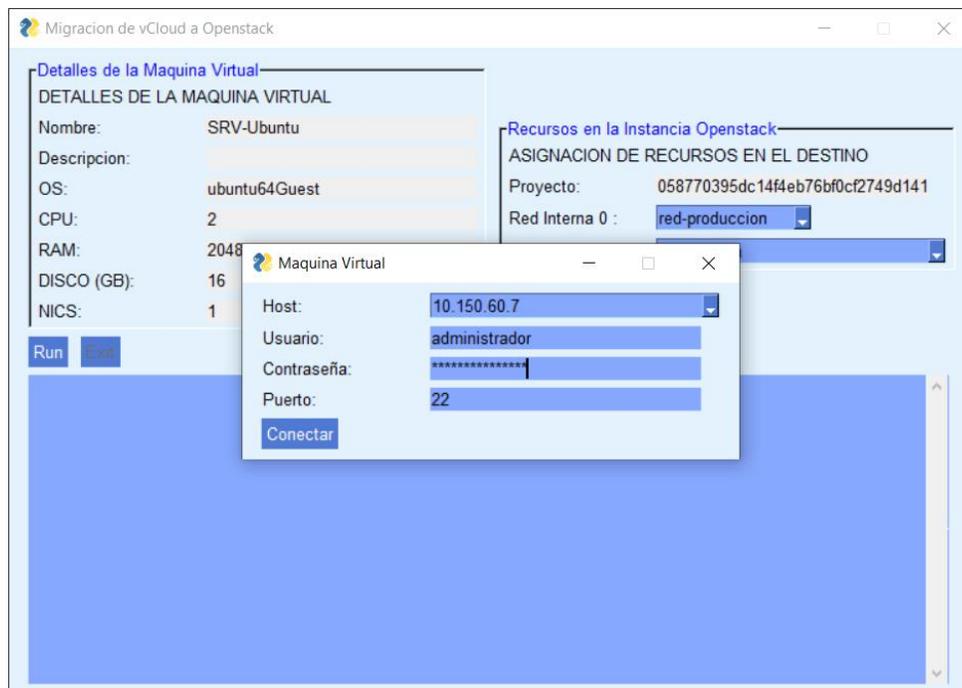




Figura 17 Framework: Ventana 3

## CAPITULO IV

# ENTORNOS DE COMPUTACIÓN EN LA NUBE

### 4.1. ARQUITECTURAS DE COMPUTACIÓN EN LA NUBE

En este apartado, se enseñan los distintos parámetros técnicos que fueron usados para el despliegue de la arquitectura de solución propuesta, obteniendo una guía de los distintos procesos que tuvieron que realizarse a fin de obtener los resultados deseados.

#### 4.1.1. Ambiente de Migración entre Nubes

Los administradores de los servidores interactúan con nuestro framework, teniendo acceso al mismo donde deben ingresar los datos de servidores y máquinas virtuales para realizar los procesos requeridos de migración.

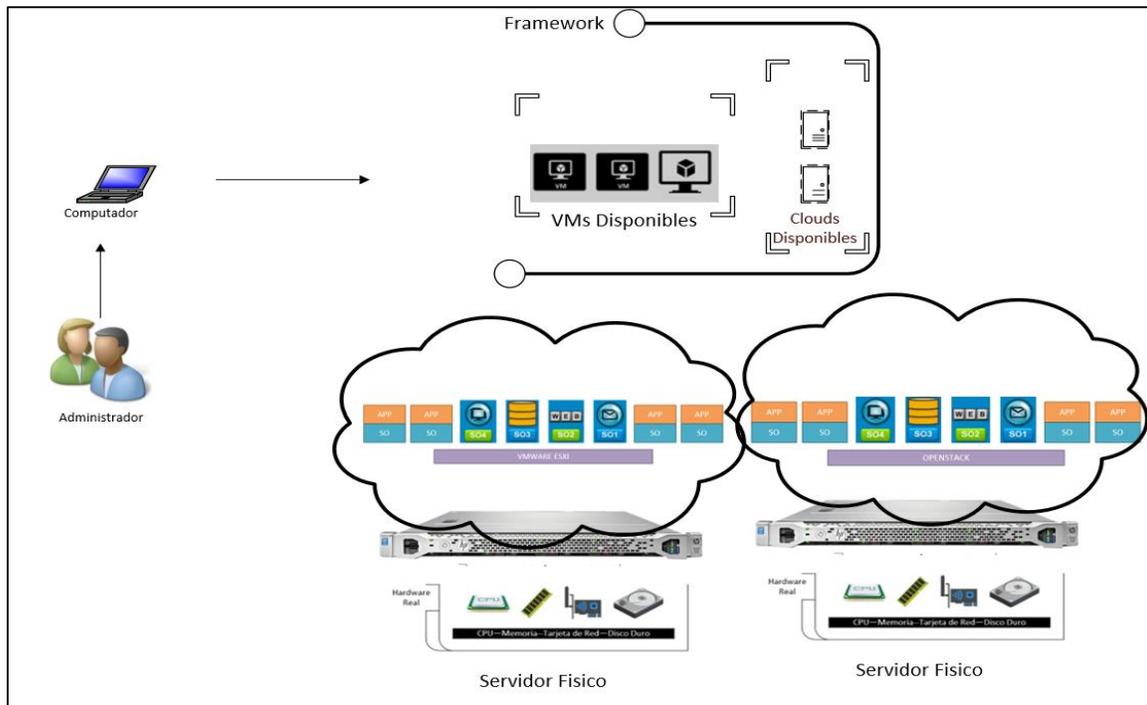


Figura 18 Ambiente de Migración entre nubes

A partir de la Figura 18, podemos describir el funcionamiento de la siguiente manera,

- El administrador de red/servidores tiene acceso al framework elaborado para poder realizar las migraciones requeridas o revisar la situación del entorno con el que cuenta.
- El computador es el medio a través del cual se puede acceder hacia el framework.
- Servidor Físico, es el lugar donde físicamente se tendrá alojado recursos que son de memoria, procesador, almacenamiento, red entre otros.
- Hipervisores (OpenStack, VCloud), son software que nos permiten realizar una virtualización de hardware a fin de desperdiciar lo menos posibles recursos de este tipo, es donde encontraremos las máquinas virtuales que deseamos migrar entre nubes.
- Framework, aquí encontraremos el intermediario que nos permitirá gestionar la migración que vamos a ejecutar entre las nubes con las que trabajamos (VCloud, OpenStack)

#### **4.1.2. Entorno de Nube de Licencia Pública(OpenStack)**

Es una plataforma de software para computo en la nube que tiene por principal atractivo el ser gratuito y de código abierto(open-source) licenciado bajo apache versión 2. (Gerald, 2018). La mayoría de las implementaciones en OpenStack son aquellas que se refieren a infraestructura con recursos virtuales como enrutadores, servidores, contenedores; todo esto implementado acorde a las necesidades del usuario.

##### **4.1.2.1. Características de OpenStack**

Acorde a la Tabla 4, tomada de la Documentación de OpenStack, decimos que disponemos de herramientas, características y utilidades que nos permiten un manejo coordinado de recursos, así mismo los usuarios y administradores podrán usar esto a su favor. Estas características con sus beneficios se detallan a continuación.

Tabla 4 Cuadro de Características que dispone OpenStack (OpenStack, 2020)

Características	Beneficios
Aprovecha el hardware básico	Sin embargo, el precio más bajo por GigaByte.
Independiente de la falla de nodo HDD	La redundancia de datos confiable y con recuperación automática protege contra fallas.
Almacén desmedido	Espacio de nombres grande y plano, acceso de lectura / escritura altamente escalable, capaz de servir contenido directamente desde el sistema de almacenamiento.
Escalabilidad multidimensional	Arquitectura de escalamiento horizontal: escale el almacenamiento distribuido vertical y horizontalmente. Realiza copias de seguridad y archiva grandes cantidades de datos con rendimiento lineal.
Estructura de cuenta / contenedor / objeto.	Sin anidamiento, no es un sistema de archivos tradicional: optimizado para escala, escala a múltiples petabytes y miles de millones de objetos.
Replicación incorporada 3 veces más redundancia de datos (en comparación con 2 veces en RAID)	Un número configurable de cuentas, contenedores y copias de objetos para alta disponibilidad.
Agregue capacidad fácilmente (a diferencia del cambio de tamaño de RAID)	Escalado de datos elásticos con facilidad.
Sin base de datos central	Mayor rendimiento, sin cuellos de botella.
RAID no requerido	Maneje muchas lecturas y escrituras pequeñas y aleatorias de manera eficiente.
Utilidades de administración integradas	Administración de cuentas: cree, agregue, verifique y elimine usuarios; Gestión de contenedores: carga, descarga y verificación; Supervisión: capacidad, host, red, arrastre de registros y estado del clúster.
Auditoría de unidades	Detecte fallas en la unidad y evite la corrupción de datos.
Objetos expirados	Los usuarios pueden establecer un tiempo de vencimiento o un TTL en un objeto para controlar el acceso.
Acceso directo a objetos	Habilite el acceso directo del navegador al contenido, como un panel de control.
Visibilidad en tiempo real de las solicitudes de los clientes	Sepa lo que solicitan los usuarios.
Soporta API S3	Utilice herramientas diseñadas para la popular API S3.
Restringir contenedores por cuenta	Limite el acceso para controlar el uso por parte del usuario.

**Nota:** Recuperado de (OpenStack, 2020)

#### 4.1.2.2. Porque se eligió OpenStack

Acorde al artículo del Analista Alan Waite (Waite, 2016), se escoge OpenStack bajo una simple premisa es la "plataforma de IaaS en la nube de código abierto más exitoso". Además, es el proyecto open-source de más rápido crecimiento en la historia con más de 20 millones de líneas de código (Lundquist, 2016).

OpenStack puede manejar pequeños o grandes pools de computo, almacenamiento y recursos de redes en un centro de datos y puede hacer estos recursos disponibles y escalables através de un tablero centralizado (Lundquist, 2016).

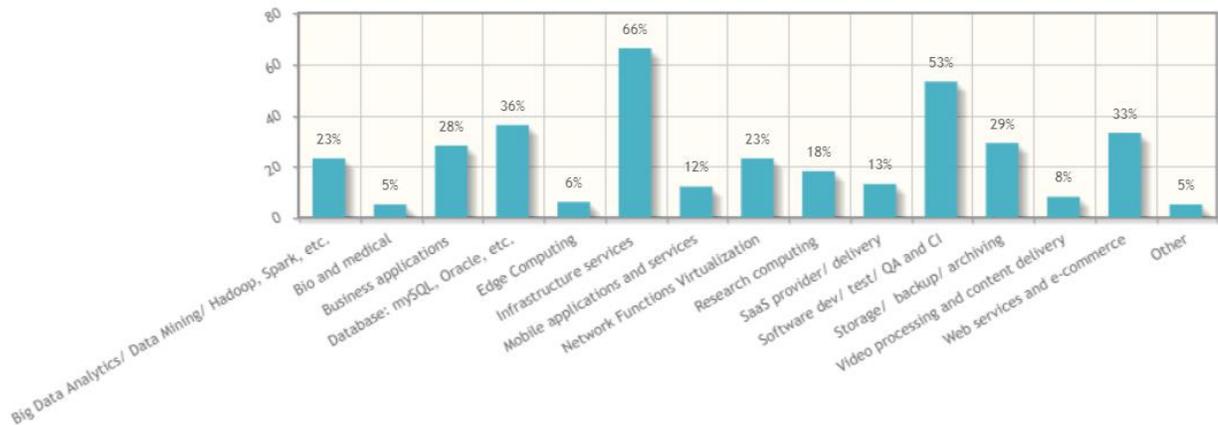


Figura 19 Cargas de trabajo en openstack [18]

Acorde al informe del OpenStack Survey del año 2019, donde se pregunta a los usuario de OpenStack para que usan esta plataforma más frecuentemente, en su mayoría la respuesta es Servicios de Infraestructura, como se muestra en la Figura 19, con lo que podemos decir (Openstack, 2019) que estamos usando una de las plataformas con más respaldo.

#### 4.1.2.3. Despliegue de Nube OpenSource

Para el despliegue de la plataforma OpenStack se utilizó un solo servidor que cumplirá con un rol de “all-in-one” (Red Hat, 2020). Para dicha instalación se usó TripleO (set de herramientas

para despliegue de OpenStack) para implementar OpenStack y los servicios relativos a este en un ambiente simple de un solo nodo. Este tipo de instalación es realizada con la finalidad de tener habilitada opciones como proof-of-concept (prueba de concepto), development(desarrollo) y test deployments (despliegue de pruebas) sin seguimiento de operaciones.

Las características de OpenStack usadas en este despliegue son:

*Tabla 5 Características del Servidor OpenStack*

Sistema Operativo	CentOS 7 x64
Memoria RAM	32 GB
Disco Duro	100 GB Sistema Storage Cinder: 1 Terabyte Raid 0 1 Terabyte Raid 0 Storage RAW 1 Terabyte
Procesador	8 vcpus
Adaptadores de Red	1 NIC “Administración” 1 NIC “Red Externa”

#### **4.1.2.4. Instalación de OpenStack**

Para realizar la instalación de la plataforma de OpenStack es necesario realizar un procedimiento consistente en una serie de pasos que pueden resumirse en (ANEXO 1):

- Actualizar Sistema

- Agregar Repositorios de las Base de Batos
- Deshabilitar SELinux
- Parar e inhabilitar servicios postfix, firewalld, networkManager
- Habilitar e iniciar servicio network
- Instalar OpenStack desde su repositorio oficial
- Instalar Certificados
- Configurar red, almacenamiento

#### **4.1.3. Entorno de Nube Propietaria (vCloud Director)**

“VMWare VCloud es despliegue, automatización y administración de software para recursos de infraestructura virtual en ambientes de nube multi-tenant. VCloud Director habilita a los proveedores de servicio en nube a convertir centros de datos físicos en centros de datos virtuales altamente elásticos. Alcanza esto al convertir los recursos de centros de datos físicos como red almacenamiento y computo en recursos de centro de datos virtual, con lo que el proveedor de servicio luego dispone de servicios basados en catalogo para usuarios internos” (TechTarget, 2021).

#### 4.1.3.1. Características de VMWare VCloud

En la Tabla 6, que fue tomada del Guía del Usuario de VMWare VCloud, se muestran las características con las que VMWare describe a VCloud.

Tabla 6 Características de VCloud (VMWare, 2017)

<b>Características</b>	<b>Descripción</b>
Centros de datos virtuales seguros elásticos	Un enfoque basado en políticas para computación, almacenamiento, redes y seguridad garantiza que los inquilinos tengan recursos virtuales aislados de forma segura, autenticación independiente basada en roles y control detallado de sus servicios de nube pública.
Gestión de múltiples sitios	Supervisión de recursos desde un panel único intuitivo con vistas agregadas de varios sitios.
Próspero ecosistema de ISV	Proveedores de software líderes en Cloud Native, Backup como Dell Data Protection y Veeam, DRaaS, y muchos más, integrados de forma nativa con VMware Cloud Director utilizando nuestra extensibilidad abierta marco de referencia.
Protección de cargas de trabajo y migración a la nube	Habilite la migración y protección de VM simples y seguras con el complemento de disponibilidad de VMware Cloud Director para Cloud Director.
Visibilidad e información operativa	Un panel actualizado y un panel único proporcionan vistas centralizadas de administración en la nube de múltiples inquilinos.
Desarrollo y aplicaciones nativas de la nube	Los desarrolladores tienen acceso a cientos de aplicaciones probadas y seguras de VMware Cloud Marketplace en un catálogo fácil de consumir.
Automatización	VMware Cloud Director permite a los proveedores de la nube automatizar flujos de trabajo complejos de infraestructura como código y de interfaz de usuario de mosaico para implementar X-as-a-Service, todo mientras se mantiene el control de acceso y la visibilidad.
Plataforma de aplicaciones como servicio	Ahora entregue sus propias aplicaciones basadas en catálogos o aplicaciones en la nube de terceros certificadas por VMware Cloud Marketplace, y aplicaciones de catálogos personalizados directamente a los clientes a través de una interfaz de catálogo simple desde el complemento App Launchpad VMware Cloud Director.

**Nota:** Recuperado de (VMWare, 2017)

#### 4.1.3.2. Por qué se eligió VMWare VCloud

Debido a que VCloud está desarrollada por VMWare, es una solución con bastante respaldo (VMWare, 2011), a su vez está tienes ciertos parámetros que la hacen destacar por encima del resto de soluciones como el hecho de estar construida sobre vSphere y proporciona todos los componentes necesarios para transformar un centro de datos interno en una nube privada segura con extensibilidad a nubes públicas. “Al integrarse y ampliar las capacidades de vSphere, las soluciones de infraestructura en la nube de VMware brindan todos los beneficios de la computación en la nube:

- Experiencia de negocio incrementada a través de rápido autoservicio de recursos bajo demanda
- Reducción de costos a través de una eficiente entrega de recursos y consolidación de infraestructura
- Seguridad Mejorada
- Portabilidad de Aplicación e interoperabilidad entre nubes privadas y públicas” (VMWare, 2011)

#### **4.1.3.3. Despliegue de la Nube Propietaria (VMWare VCloud)**

Para el despliegue e instalación de VMWare VCloud es necesario tener instancias físicas o virtuales en ejecución de VMWare ESXi y VMWare VCenter. Para tener estos elementos se prefirió tener un solo servidor físico para simplificar los temas de instalación y despliegue. ESXi es el encargado de proporcionar las capacidades de almacenamiento, cómputo y también redes. VCenter es la plataforma donde se centraliza la administración uno o más instancias ESXi. Cada instancia de VCenter es agregada a VCloud.

Los recursos de las instancias de VMware ESXi usados son:

*Tabla 7 Características de VMWare ESXi*

Sistema Operativo	VMware Sphere ESXi 6.7
Memoria RAM	4 GB
Disco Duro	500 GB
Procesador	1 cpu 1 socket 1 núcleo x socket

Los recursos del servidor de vCenter usados son:

*Tabla 8 Características de VMWare vCenter*

Sistema Operativo	vCenter Server Appliance 6.7
Memoria RAM	16 GB
Disco Duro	250 GB
Procesador	2 vcpus

Las características del servidor de vCloud son:

*Tabla 9 Características de VMware Cloud Director*

Sistema Operativo	VMware Cloud Director 9.7
-------------------	---------------------------

Memoria RAM	8 GB
Disco Duro	100 GB
Procesador	2 vcpus

#### 4.1.3.4. Instalación de VMWare vCloud

En esta sección resumimos el procedimiento a fin de instalar VMWare VCloud Director en nuestra infraestructura (VMWare, 2017). Para un detalle más amplio de la instalación de vCloud Director 10 revisar el ANEXO 2, donde se detalla el proceso de despliegue de la solución.

- Configurar Servidor DNS
- Instalar sobre servidores físicos instancias de VMWare ESXi
- Configurar Red y Conectividad de Servidores ESXi
- Realizar despliegue de VMWare VCenter con su respectiva configuración de red y administrativa.
- Instalar y Configurar Servidor NFS
- Instalar y Configurar NSX
- Desplegar Máquina Virtual de VMWare VCloud Director

#### 4.1.4. Entorno de Framework de Migración

El Framework en el que hemos trabajado es un software que usa las APIs de VMWare y OpenStack que están en el lenguaje de programación Python para realizar el proceso de migración de Máquinas Virtuales entre nubes selectas en este caso VCloud y OpenStack.

##### 4.1.4.1. Migración de Máquinas Virtuales con Framework Creado

En esta sección se presenta el procedimiento que realiza el framework para migrar una máquina virtual desde vCloud hacia OpenStack. Véase Figura 12:

- Inyecta controladores VirtIO
- Expande las particiones de disco
- Customiza la máquina virtual que será migrada
- Crear los volúmenes Cinder en OpenStack
- Convierte el disco duro de VMDK a Ceph
- Crear el puerto Neutron
- Crear e Inicia la instancia en OpenStack

## CAPITULO V

### **PRUEBAS Y RESULTADOS**

Una vez concluida la codificación del sistema de software, mismo que fue desarrollado bajo el lenguaje de programación Python 3 e integrada a una interfaz de fácil uso para el usuario se procedió a realizar el tratamiento correspondiente de los datos obtenidos como resultado del proceso de migración de las instancias desde el sistema de cloud computing vCloud de licenciamiento privado de VMware hacia la plataforma de cloud computing Openstack, mismo que es distribuido bajo la licencia Apache. Estos datos mostrarán la eficiencia y funcionalidad del sistema de migración e indicarán las conclusiones respectivas a las cuales llega la presente investigación.

#### **5.1. ANÁLISIS DE ENTORNO Y HERRAMIENTAS EXISTENTES**

Los proyectos de virtualización tanto vCLOUD propiedad de VMWare y de licenciamiento pagado y Openstack un proyecto open source licenciado bajo Apache, cuentan con bibliotecas de APIs definidas que permiten realizar la gestión y administración de los recursos de cómputo, red y almacenamiento mediante procesos automatizados o definiendo llamadas a subprocesos definidos que realizan acciones determinadas en el sistema.

#### **5.2. ARQUITECTURA DE VIRTUALIZACIÓN PARA PRUEBAS DE FUNCIONALIDAD**

Las pruebas de funcionamiento se realizan en escenarios virtualizados haciendo uso de tecnologías de hardware HPE (ProLiant DL160 Gen 9) donde se ejecuta un Hipervisor ESXi 6.7 Update 2, misma infraestructura virtualizada donde se ejecutan las instancias de vCenter

6.7, NSX 6.5, vCloud Director 10 y un servidor de DNS. a su vez la plataforma de virtualización Open Source se ejecuta sobre una máquina virtual con Centos 7 la cual se encuentra desplegada sobre una infraestructura de hardware de la familia Lenovo (ThinkServer TS150) como se detalla en el capítulo 7 “Topología de Red Propuesta”, a esto se le suma un cliente que desempeña los roles de intermediador entre las plataformas de cloud permitiendo gestionar los procesos necesarios para la migración de las máquinas virtuales.

En la siguiente ilustración se puede visualizar un detalle general de cómo se hizo la distribución de componentes para realizar las pruebas de validación de herramienta de software, se determina la ubicación del host con el agente de migración (framework), en una ubicación de la intranet de vCLOUD pero esta elección es opcional y podría distribuirse en cualquier punto de red siempre y cuando el host con el agente (framework) pueda establecer una comunicación con las plataformas de cloud mediante consultas REST, se debe considerar esto a la hora de configurar los firewalls de borde o cualquier otro firewall que proteja la red en cualquiera de las dos arquitecturas permitiendo establecer solicitudes API REST a través de HTTP.

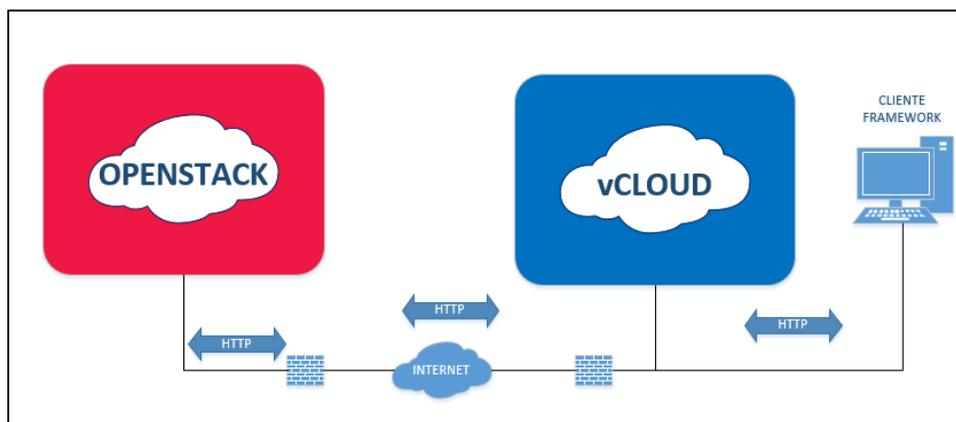


Figura 20 Diagrama Conexión entre VMWare VCloud y OpenStack

### 5.3. MEDIDAS DE RENDIMIENTO

Estas métricas se toman en base a los tiempos que el framework concluye una tarea específica misma que se obtiene en base a la hora de inicio y la hora de fin de la tarea

De esta forma los parámetros que tomaremos en cuenta es:

- El sistema con el que opera la máquina virtual que deseamos migrar.
- El espacio de almacenamiento que ocupa la máquina virtual.
- El consumo de recursos necesario para la migración.
- El tiempo que toma el proceso.

#### 5.3.1. Máquinas Virtuales a Migrar

Para realizar estas pruebas se despliegan máquinas virtuales sobre la plataforma de vCloud Director 10 a fin de ser migradas hacia la nube de OpenStack:

- Un servidor simple de Ubuntu, basado en la distribución de Linux Debian, con el sistema operativo de Ubuntu y con características como son detalladas en la Tabla 10.
- Un servidor windows, basado en la distribución de Windows Server 2016 Standard, con servicios de DNS corriendo y con características como son detalladas en la Tabla 11.

Tabla 10 Características de Máquina Virtual a Migrar Numero 1

<b>Nombre de equipo:</b>	SRV-Ubuntu
<b>Descripción:</b>	-
<b>Sistema Operativo:</b>	Ubuntu Linux (64 bit)
<b>Memoria:</b>	2 GB
<b>Almacenamiento:</b>	12 GB
<b>CPUs:</b>	2

Tabla 11 Características de Máquina Virtual a Migrar Numero 2

<b>Nombre de equipo:</b>	SRV-Windows
<b>Descripción:</b>	-
<b>Sistema Operativo:</b>	Windows (64 bit)
<b>Memoria:</b>	4 GB
<b>Almacenamiento:</b>	40 GB
<b>CPUs:</b>	4

#### 5.4. HOST CLIENTE (FRAMEWORK)

Para alojar el Framework, se usó un laptop de uso doméstico, la misma que contaba con las características que se detallan en la Tabla 12.

Tabla 12 Características del Host del Framework

<b>Modelo:</b>	81LK
<b>Marca:</b>	Lenovo
<b>Memoria:</b>	8 GB
<b>Almacenamiento:</b>	512 GB
<b>Procesador:</b>	Intel Core i5-9300H
<b>CPUs:</b>	4 físico, 8 lógicos

### 5.4.1. Consumo de Recursos

En el host del Framework se consumen recursos de hardware durante el proceso de migración los mismos que afectan principalmente a la parte de red, memoria, disco, tarjeta gráfica y procesador, como puede verse en la Tabla 14, 16 y 18.

#### 5.4.1.1. Proceso de migración 1:

Proceso de Migración de una instancia haciendo uso de certificados para la seguridad en la transmisión de la información.

Tabla 13 Distribución Tareas- Tiempos Proceso de Migración 1

Instancia Migrada: SRV-Ubuntu	
ID: f5af9831-1edc-4004-b81d-02cdf52403f	
Tarea	Hora
Creación de Snapshot	2021-02-24T08:54:23.932272
Reconfiguración de Instancia	2021-02-24T08:54:43.885408
Exportación de Discos	2021-02-24T09:06:47.080716
Conversión de Discos	2021-02-24T09:22:16.195656
Envío de Imagen QCOW2 a Glance	2021-02-24T09:23:34.381133
Creación de Volumen Cinder	2021-02-24T09:56:42.174998
Creación Keypair	2021-02-24T09:57:39.037343
Creación de Puerto Neutron	2021-02-24T09:57:59.037343
Creación de Grupo de Seguridad	2021-02-24T09:58:50.983858

Creación de Instancia	2021-02-24T10:05:00.685825
Asignación de IP Flotante	2021-02-24T10:49:10.142278
Asignación de Grupo de Seguridad	2021-02-24T10:49:14.517511
Renombrar Instancia	2021-02-24T10:49:18.397499
<b>Total:</b>	<b>1:55:37.997948</b>

## Resultado del proceso de migración 1:

Proyecto / Compute / Instancias / SRV-Ubuntu

### SRV-Ubuntu

Vista general Interfaces Log Consola Registro de acciones

**Nombre** SRV-Ubuntu  
**ID** b80b9296-51db-47dd-bcc5-940faebdd05  
**Descripción** mgworker-ins-f5af8831-1edc-4004-b81d-02cdf652403f  
**ID del proyecto** 76ac60b1f0a46468a5787c31a17aece  
**Estado** Activo  
**Bloqueada** False  
**Zona de Disponibilidad** nova  
**Creada** 24 de Febrero de 2021 a las 15:49  
**Age** 5 minutos  
**Host** opslab.ops.local

**Especificaciones**

**Nombre del sabor** m1.small  
**ID del sabor** 2  
**RAM** 2GB  
**VCPU** 1 VCPU  
**Disco** 20GB

**Direcciones IP**

**net-prod-cloud** 192.168.159.178, 172.17.1.37

**Grupos de seguridad**

**mgworker-seg-ba0b5f2...** PERMITIR IPv4 445/tcp from 0.0.0.0/0

Figura 21 Instancia Migrada Proceso de Migración 1

```

Ubuntu 18.04.5 LTS ubuntu tty1
ubuntu login: [ 140.6289s] cloud-init[903]: Cloud-init v. 20.4.1-ubuntu18.04.1 running 'modules:final' at Wed, 24 Feb 2021
15:53:20 +0000. Up 139.78 seconds.
[ 149.31607s] cloud-init[945]: Cloud-init v. 20.4.1-ubuntu18.04.1 running 'modules:final' at Wed, 24 Feb 2021 15:53:32 +0000
. Up 141.04 seconds.
[ 149.33279s] cloud-init[945]: Cloud-init v. 20.4.1-ubuntu18.04.1 finished at Wed, 24 Feb 2021 15:53:34 +0000. DataSource: Ba
tsSourceNone. Up 149.31 seconds
[ 149.94148s] cloud-init[945]: 2021-02-24 15:53:34.863 - cc_final_message.py[WARN]: Used failBack: datasource

Ubuntu 18.04.5 LTS ubuntu tty1
ubuntu login: administrador
Password:
Last login: Wed Feb 24 13:21:58 UTC 2021 on tty1
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-136-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:   https://landscape.canonical.com
 * Support:      https://ubuntu.com/support

System information disabled due to load higher than 1.0

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
0 packages can be updated.
0 of these updates are security updates.

New release '20.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

administrador@ubuntu:~$

```

Figura 22 Consola Instancia Migrada Proceso de Migración 1

Como resultado de este proceso de migración se puede verificar que la máquina virtual ha migrado exitosamente hacia el sistema de virtualización de Openstack este proceso se puede validar mediante los IDs de la instancia que se está ejecutando.

Tabla de Métricas para verificar el consumo de recursos generado por el proceso de migración 1 en el HOST donde se ejecuta el framework de migración, mismas que fueron obtenidas mediante el sistema de monitoreo Zabbix un proyecto open source.

Tabla 14 Consumo de Recursos Proceso de Migracion 1

CONSUMO DE RECURSOS			
CPU	MÍNIMO	MEDIA	MÁXIMO
	1.165%	7.3074%	38.234%
MEMORIA	MÍNIMO	MEDIA	MÁXIMO
Utilización:	88.274%	95.6127%	98.0965%
RED	MÍNIMO	MEDIA	MÁXIMO
Envío:	274.48 Kbps	15.83 Mbps	48.76 Mbps
Recepción:	927.68 Kbps	32.67 Mbps	73.12 Mbps
DISCO	MÍNIMO	MEDIA	MÁXIMO
Rear rate:	1.2123 r/s	131.2252 r/s	1723.8866 r/s
Write rate:	46.7109 w/s	38.8186 w/s	329.2287 w/s

#### 5.4.1.2. Proceso de migración 2:

Este proceso se realiza haciendo uso de la herramienta sin ningún tipo de seguridad y los datos se envían a través de la red sin ningún tipo de protección.

Tabla 15 Distribución Tareas- Tiempos Proceso de Migración 2

Instancia Migrada: SRV-Ubuntu	
ID: 5a11c7e3-ac26-4831-9d96-252c1decb688	
<b>Tarea</b>	<b>Tiempo</b>
Creación de Snapshot	2021-03-02T22:49:52.894479
Reconfiguración de Instancia	2021-03-02T22:50:13.90.0353
Exportación de Discos	2021-03-02T22:56:19.980602
Conversión de Discos	2021-03-02T23:06:45.868637
Envío de Imagen QCOW2 a Glance	2021-03-02T23:07:49.706127
Creación de Volumen Cinder	2021-03-02T23:17:54.671704
Creación Keypair	2021-03-02T23:18:06.801378
Creación de Puerto Neutron	2021-03-02T23:18:09.829312
Creación de Grupo de Seguridad	2021-03-02T23:18:11.648289
Creación de Instancia	2021-03-02T23:18:14.334343
Asignación de IP Flotante	2021-03-02T23:53:09.405619
Asignación de Grupo de Seguridad	2021-03-02T23:53:12.500051
Renombrar Instancia	2021-03-02T23:53:26.286278
Tiempo:	1:03:55.284.158

Tabla de Métricas para verificar el consumo de recursos en el HOST donde se ejecuta el framework de migración, mismas que fueron obtenidas mediante el sistema de monitoreo Zabbix un proyecto open source.

Tabla 16 Consumo de Recursos Proceso de Migración 2

CONSUMO DE RECURSOS			
CPU	MÍNIMO	MEDIA	MÁXIMO
	0.772%	4.9725%	19.234%
MEMORIA	MÍNIMO	MEDIA	MÁXIMO
	Utilización:	87.4197%	89.026%
RED	MÍNIMO	MEDIA	MÁXIMO
	Envío: Recepción:	222.12 Kbps 49.62 Kbps	13.52 Mbps 27.8 Mbp
DISCO	MÍNIMO	MEDIA	MÁXIMO
	Rear rate: Write rate:	0.8127 r/s 7.1394 w/s	146.1657 r/s 47.9623 w/s

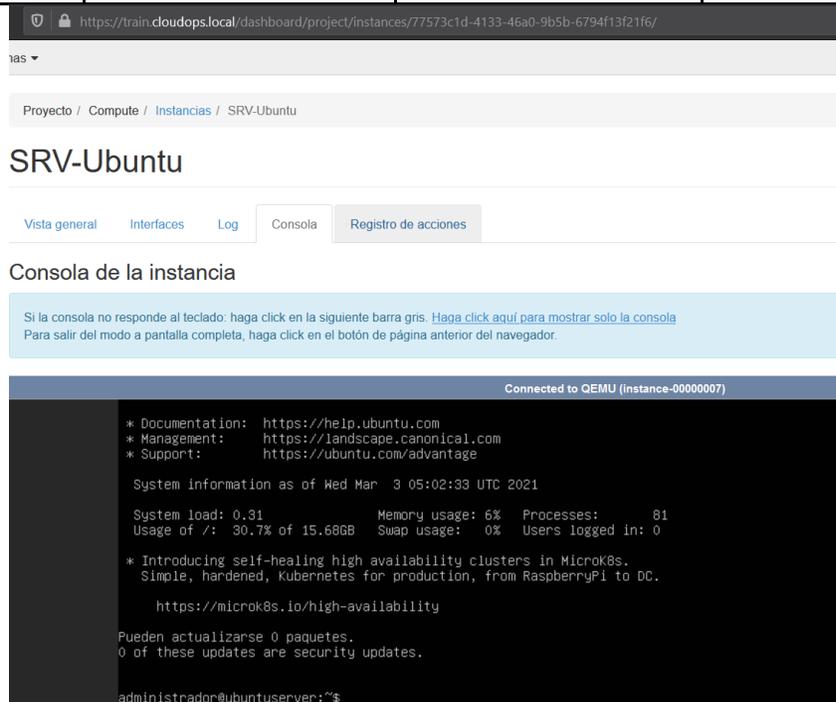


Figura 23 Consola de Instancia Proceso de Migración 2

### 5.4.1.3. Proceso de migración 3:

Para este proceso de validación de la herramienta se elige un servidor de la familia Windows, mismo que se caracteriza y se especifica a detalle en la Tabla 17.

Tabla 17 Distribución Tareas-Tiempos Proceso de Migración 3

Instancia Migrada: SRV-Windows	
ID: 500c3090-9e06-4cb4-aa8e-874e3bed6502	
Tarea	Tiempo
Creación de Snapshot	2021-03-03T01:49:24.504824
Reconfiguración de Instancia	2021-03-03T01:49:43.942049
Exportación de Discos	2021-03-03T01:49:44.965772
Conversión de Discos	2021-03-03T02:18:30.478216
Envío de Imagen QCOW2 a Glance	2021-03-03T02:22:30.748689
Creación de Volumen Cinder	2021-03-03T02:36:37.936447
Creación Keypair	2021-03-03T02:36:51.006101
Creación de Puerto Neutron	2021-03-03T02:36:57.293858
Creación de Grupo de Seguridad	2021-03-03T02:36:59.997796
Creación de Instancia	2021-03-03T02:37:11.335002
Asignación de IP Flotante	2021-03-03T03:58:09.368890
Asignación de Grupo de Seguridad	2021-03-03T03:58:54.505493
Renombrar Instancia	2021-03-03T03:58:57.764587
Tiempo:	2:09:37.655815

Tabla de Métricas para verificar el consumo de recursos en el HOST donde se ejecuta el framework de migración, mismas que fueron obtenidas mediante el sistema de monitoreo Zabbix un proyecto open source.

Tabla 18 Consumo de Recursos Proceso Migracion 3

CONSUMO DE RECURSOS			
CPU	MÍNIMO	MEDIA	MÁXIMO
	0.7769%	4.3204%	20.4312%
MEMORIA	MÍNIMO	MEDIA	MÁXIMO
Utilización:	85.0326%	84.6966%	91.752%
RED	MÍNIMO	MEDIA	MÁXIMO
Recepción: Envío:	72.9 Kbps 20.98 Kbps	17.04 Mbps 32.83 Mbps	51.49 Mbps 98.32 Mbps
DISCO	MÍNIMO	MEDIA	MÁXIMO
Rear rate: Write rate:	0.6311 r/s 7.522 w/s	47.8612 r/s 14.1508 w/s	1274.4711 r/s 237.639 w/s

Resultado del proceso de migración 3:

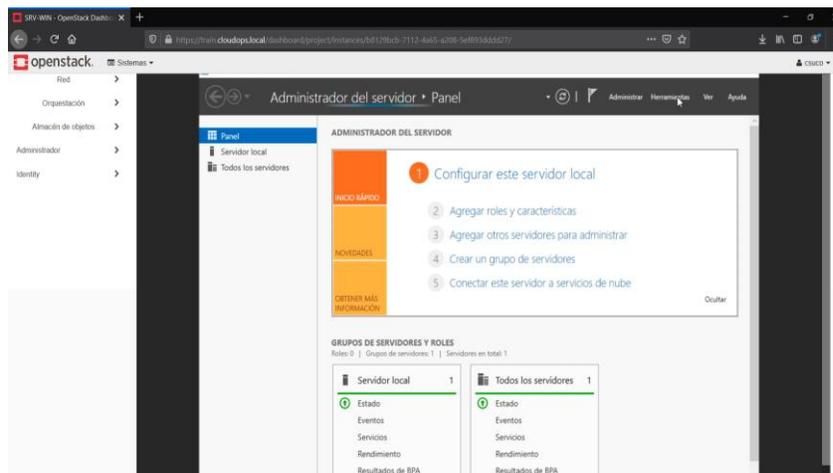


Figura 24 Consola Resultante Proceso de Migracion 3

## CAPITULO VI

### CONCLUSIONES Y RECOMENDACIONES

#### 6.1. CONCLUSIONES

El desarrollo de este trabajo de investigación ha demostrado que se cumplió los propósitos propuestos en cada una de sus etapas dejando como conclusiones los siguientes puntos:

- La infraestructura de vCloud Director es una solución que debe ser diseñada a medida de la empresa o de los servicios que esta fuera a ofrecer bajo su ambiente de producción ya que requiere de complementos o soluciones adicionales y complejas al hipervisor de tipo 1 más comúnmente conocido como ESXi o vSphere, adicional a esto se deben desplegar los servicios de DNS y NFS(Almacén de registros de la base de datos de vCloud) bajo un sistema operativo a elección del administrador sea esta Windows o Linux, misma que permitirá simplificar la gestión bajo resoluciones FQDN o nombres de dominios, la segunda solución a desplegar es vCenter en una versión 6.7 esta es la versión que se ajusta completamente con la solución NSX 6.5 la cual es un requerimiento obligatorio para este ambiente de Cloud y es la que se encargará de gestionar las Redes definidas por software que se crearán a medida que vCloud agregue Routers y Redes a las organizaciones para segmentar el tráfico entre las mismas y permitir accesos hacia la internet.
- Como agregado adicional a lo propuesto para el desarrollo de este proyecto se decidió agregar una solución de seguridad mediante SSL, misma que se descarta en esta implementación debido a que los tiempos de migración de una máquina virtual llegan a ser hasta 82.5% más lento que el envío de datos sin encriptar como se indica en las Tablas 13 y 15 por esta razón se descarta esta solución, en conclusión a este problema recomendamos

desplegar el HOST (Framework) detrás de un Firewall o Router de borde como se indica en la Figura 5 y establecer un túnel IPSec entre los routers de borde de las soluciones de Virtualización, así garantizamos que los datos viajan seguros a través de la internet, optimizando los tiempos y recursos del sistema donde corre el Framework.

- Los recursos de Hardware sobre los cuales se realice el despliegue de las soluciones de Cloud Computing deben ser adecuados para obtener el mayor rendimiento del Framework y a su vez de las mismas infraestructura de virtualización, para nuestro ambiente de pruebas se usaron discos HDD de 7200 rpm para el caso de la solución requerida por VMware para vCloud Director 10 y discos de 5200 rpm para la solución de Openstack lo cual representa un cuello de botella si nos enfocamos en velocidades de lectura y escritura.
- Se establece que el código fuente base de la solución que hemos desarrollado en este proyecto es simplificado y funcional ha sido validado de manera independiente método a método permitiendo únicamente recibir variables necesarias para la correcta operación de los mismos, todas las validaciones y restricciones para el usuario se realizan en una clase separada, misma que se enlaza con las clases de la interfaz del usuario y donde se validan parámetros y se trata la información que el usuario ingrese, a fin de minimizar la incurrencia de errores del usuario se implementa la solución de elección múltiple en las opciones que el sistema mostrará al usuario. En conclusión, esta arquitectura permitirá realizar cambios en la capa de presentación sin comprometer la capa de negocios donde están los métodos base.
- De acuerdo a la pruebas realizadas las máquinas virtuales con sistemas operativos de la familia Linux o Windows que se requiere migrar deben estar encendidas, Adicionalmente las instancias que tengan linux como sistema operativo deberán tener una conexión a

internet y deberán poder ser capaces de instalar paquetes desde la misma este proceso se validará antes de comenzar la migración de manera automática, todos los paquetes que se instalarán y las modificaciones al sistema se realizarán después de haberse creado un snapshot de la instancia y no será perceptible al usuario, una vez concluya el proceso de extracción de discos la instancia será restaurada a su estado original y todas las modificaciones de la VM serán removidas, en conclusión el sistema garantiza que la máquina virtual de origen quede íntegra en caso de que se genere algún percance el proceso pueda repetirse o la máquina original pueda volver a producción sin alteraciones.

- Para que la migración de Instancias desde el ambiente de vCloud hacia Openstack sea correcta y funcional en el destino se requieren que las mismas tengan instalado los drivers virtio, estos paquetes son necesarios y deberán ser instalados antes de iniciar la migración, en conclusión para solventar este inconveniente hemos desarrollado un script que automatiza este proceso mismo que puede se encuentra en la sección ANEXO 4 y ha sido probado en estas versiones de la familia de Linux (Ubuntu desde la versión 14.04 hasta la versión actual 20.04 , Centos 7 y 8, y Debian 8,9 y 10), para los sistemas operativos de la familia Windows a partir de la versión windows server 2008 se proporciona un instalador en la carpeta Recursos/drivers/virtio en el paquete del framework, mismo que a sido obtenido de los repositorios de fedora y deberá ser instalado manualmente antes de iniciar la migración, el proceso no es perjudicial ni compromete la funcionalidad del servidor ya que al ser un software empaquetado en un instalador .msi puede ser removido a voluntad si es necesario.
- Los tiempos de migración son variables como se establece en las tablas 11, 12, todo esta variación de tiempos se debe a la cantidad de datos que tendrá el o los discos virtuales, no

es lo mismo migrar un disco de 40 GB con espacio utilizado de 30 GB que un disco de 100GB con un espacio utilizado de 8 GB, discos con aprovisionamiento en “think” o grueso, o desde y hacia sistemas de almacenamiento de bajo rendimiento, para los casos de discos con espacio utilizado de consideración los tiempos de proceso pueden ser de hasta 4 horas, mientras que para los discos pequeños el tiempo será variable y estaría en el rango de 30 minutos a 1 hora con 30 minutos en todo este proceso el Host anfitrión del framework deberá estar ejecutándose por lo que la mejor ubicación del mismo sería una máquina virtual con python que ejecute el framework en un servidor dedicado bajo un hipervisor de tipo 1, que un computador personal.

- En cuanto de a qué solución de nube optar, si VMWare VCloud o OpenStack, podríamos decir que tomando como principal parámetro la economía, la decisión es evidente “OpenStack”, sin embargo hay que tomar en cuenta los costos de mantenimiento para soporte, también nos será necesario un nivel de conocimientos mayor relacionado con el sistema y networking requerido para trabajar con OpenStack, este nivel de conocimiento es mucho más elevado que con VCloud, necesitando de profesionales más experimentados. En síntesis, en un inicio OpenStack será más hostil con respecto a su mantenimiento, no obstante, los casos de uso de la nube tienden al crecimiento y, a medida que las personas adquieran más experiencia con OpenStack, los costos administrativos serán inclusive menores.

## 6.2. RECOMENDACIONES

- La tecnología de almacenamiento a la fecha presente ha mejorado circunstancialmente y hoy en día en sistemas de producción críticos puede ser un punto muy importante y de consideración a la hora de diseñar o dimensionar un sistema de cloud computing por lo que el administrador de red debe buscar la mejor alternativa para el almacenamiento debiendo optar de ser necesario por utilizar soluciones con tecnología ALL FLASH o SAS de 10k o 15k.
- Los recursos de cómputo para el sistema de vCloud Director deberían estar distribuidos en múltiples Host para optimizar los mismos ya que para un correcto funcionamiento de los componentes se requiere para vCenter 12 GB RAM y 4 vCPUs, NSX 16 GB RAM y 4 vCPUs, DNS 4 GB RAM y 2 vCPUs, adicional a esto se requieren recursos adicionales para el funcionamiento de Controladores 2 vCPUs/cu y 4GB RAM/cu, Routers 2 vCPUs/cu y 4GB RAM/cu y Switches 2 vCPUs/cu y 4GB RAM/cu. Adicionalmente se deberá asignar cuotas de consumo por organizaciones para optimizar y garantizar la correcta operación de las instancias.
- Se recomienda que las instalaciones de Openstack para sistemas en producción se instalen en sistemas operativos Centos 7 para evitar problemas de dependencias o soporte a largo plazo, o a su vez buscar una alternativa en Ubuntu que cumpla con todos los requerimientos de la solución Openstack, además es recomendable que la solución se base en una arquitectura distribuida separando el cómputo, almacenamiento, la red permitiendo que pueda ser concurrente y que a futuro sea un sistema escalable.
- Se recomienda usar entornos virtualizados de python para ejecutar el sistema de migración ya que requiere paquetes adicionales para su correcto funcionamiento y si el Host no es un

entorno aislado y de pruebas podría incurrir en errores al sistema anfitrión, lo que en entornos virtualizados se puede evitar ya que al ser aislado se puede generar y borrar cuantas veces sea necesario.

- Se recomienda ubicar el Host o la máquina anfitrión donde se ejecutará el sistema de migración en un punto en la red donde pueda conectarse y enviar solicitudes API REST hacia los dos sistemas de cloud computing y se deberá garantizar esta conectividad a través de apertura de puertos entre firewalls o a su vez mediante VPNs.
- Se recomienda que toda máquina virtual a ser migrada esté encendida al momento de iniciar el proceso de migración y no se recomienda interrumpirlo forzando o terminando el proceso, una vez haya inicializado el proceso el código fuente deshabilita los botones de “Terminar” y “Salir” para evitar que este sea interrumpido, el código habilitará estos botones una vez haya concluido el proceso en su totalidad.

## GLOSARIO DE TÉRMINOS

Cloud	Nube (Internet).
Cloud Computing	Computación en la nube.
Framework	Entorno de Trabajo.
Hipervisor	Tecnología que permite el uso de diferentes sistemas operativos o máquinas virtuales sobre un único host
IaaS	Infrastructure as a Service, Infraestructura como Servicio.
OpenStack	Software de Cloud Computing para proporcionar IaaS.
PaaS	Platform as a Service, Plataforma como Servicio.
PYMES	Empresas pequeñas y medianas.
SaaS	Software as a Service, Software como Servicio.
vCloud	Software de nube privada empresarial propiedad de VMware.

VMs	Máquinas Virtuales
vRealize	Plataforma de gestión de nube híbrida propiedad de VMware.
vSphere	Plataforma de virtualización de servidores propiedad de VMware.
FWaaS	Firewall as a Service
LBaaS	Load Balancing as a Service
TripleO	Set de herramientas para el despliegue y administración de OpenStack
all-in-one	Ambiente de un solo nodo en instalación de OpenStack

## REFERENCIAS

- Arévalo-Cordero, S. J., Gallegos-Segovia, P. L., Vintimilla-Tapia, P. E., Bravo-Torres, J. F., Cedillo-Elias, E. J., & Larios-Rosillo, V. M. (2019). *Data traffic management in a hybrid cloud composed of Openstack and Azure*. Barranquilla: IEEE Colombian Conference on Communications and Computing (COLCOM).
- Barrett, C., Britten, T., Cacciatore, K., Chadwick, P., Phipps, P., Prüßmann, G., . . . Wu, S. (2018). *OpenStack: The Path to Cloud Considerations and recommendations for businesses adopting cloud technology*. OpenStack Foundation.
- Dialogic. (2017). *Introduction to Cloud Computing*. Parsippany: White Paper.
- Gerald, M. (2018). *Design and Implementation of OpenStack*. Helsinki: Metropolia University of Applied Sciences.
- Gholami, M., Low, G., Beydoun, G., & Daneshggar, F. (2016). *Cloud migration process-A survey, evaluation framework, and open challenges*. Journal of Systems and Software.
- Gorelik, E. (2013). *Cloud Computing Models*. Massachusetts: Composite Information Systems Laboratory (CISL), Massachusetts Institute of Technology.
- INCIBE. (2013). *Cloud Computing, Una guía de aproximación para el Empresario*. Leon: Instituto Nacional de Ciberseguridad.
- Jamshidi, P., Ahmad, A., & Pahl, C. (2014). *Cloud Migration Research: A Systematic Review*. IEEE Transactions on Cloud Computing.
- Josyula, V., Orr, M., & Page, G. (2011). *Cloud Computing, Automating the Virtualized Data Center*. Cisco Press.
- Lundquist, L. (2016, Abril 14). *Why Choose OpenStack?* IBM Bluemix Private Cloud Help. Retrieved Enero 19, 2021, from <https://ibm-blue-box-help.github.io/help->



# ANEXOS

## 7.1. ANEXO 1: INSTALACIÓN DE OPENSTACK

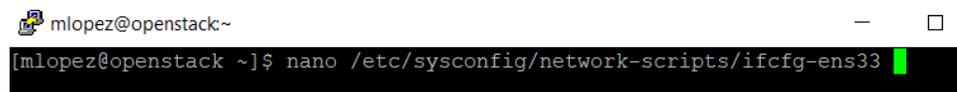
Para el despliegue de nuestro Servidor de Nube opensource, escogimos la opción de despliegue “all-in-one”. Para cumplir nuestra instalación debemos hacer lo siguiente:

### 7.1.1. Establecer IP estática en el Servidor

Para proceder con la respectiva instalación de los paquetes con los respectivos repositorios de OpenStack se debe establecer una IP estática, para esto debemos configurar el archivo de la interfaz con la información requerida.

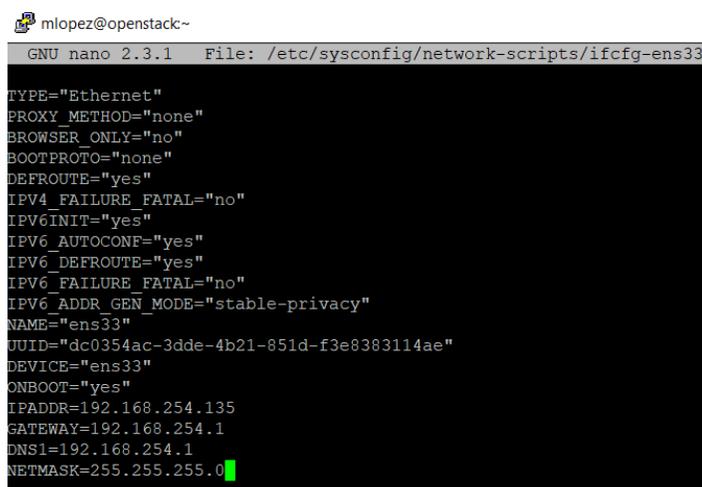
#### Comando:

```
nano /etc/sysconfig/network-scripts/ifcfg-ens33
```



```
mlopez@openstack:~  
[mlopez@openstack ~]$ nano /etc/sysconfig/network-scripts/ifcfg-ens33
```

Figura 25 Comando Configuración de Archivo Interface



```
mlopez@openstack:~  
GNU nano 2.3.1 File: /etc/sysconfig/network-scripts/ifcfg-ens33  
TYPE="Ethernet"  
PROXY_METHOD="none"  
BROWSER_ONLY="no"  
BOOTPROTO="none"  
DEFROUTE="yes"  
IPV4_FAILURE_FATAL="no"  
IPV6INIT="yes"  
IPV6_AUTOCONF="yes"  
IPV6_DEFROUTE="yes"  
IPV6_FAILURE_FATAL="no"  
IPV6_ADDR_GEN_MODE="stable-privacy"  
NAME="ens33"  
UUID="dc0354ac-3dde-4b21-851d-f3e8383114ae"  
DEVICE="ens33"  
ONBOOT="yes"  
IPADDR=192.168.254.135  
GATEWAY=192.168.254.1  
DNS1=192.168.254.1  
NETMASK=255.255.255.0
```

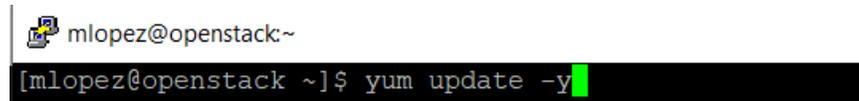
Figura 26 Asignación de la IP Estática

### 7.1.2. Actualización de Centos

Previo a iniciar con nuestra instalación de OpenStack, debemos actualizar nuestro sistema Centos a fin de tener los paquetes necesarios en sus versiones más recientes.

#### Comando:

```
yum update -y
```



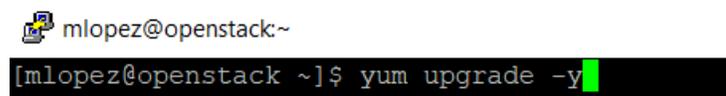
```
mlopez@openstack:~  
[mlopez@openstack ~]$ yum update -y
```

Figura 27 Actualización de Paquetes Centos

Tras la ejecución del comando anterior se actualiza la lista de paquetes de CentOS a sus versiones más recientes, procedemos ejecutando un upgrade para tener el sistema operativo actualizado.

#### Comando:

```
yum upgrade -y
```



```
mlopez@openstack:~  
[mlopez@openstack ~]$ yum upgrade -y
```

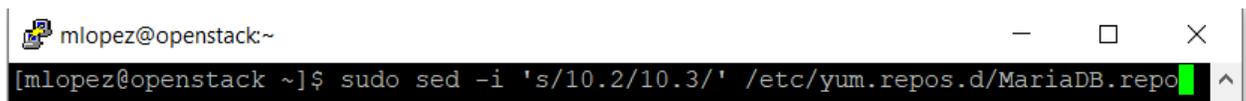
Figura 28 Actualización Sistema Operativo

### 7.1.3. Agregar repositorios de MariaDB

Debemos agregar los repositorios de MariaDB para poder continuar con nuestro procedimiento de instalación.

#### Comando:

```
sudo set -i 's/10.2/10.3/' /etc/yum.repos.d/MariaDB.repo
```



```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo sed -i 's/10.2/10.3/' /etc/yum.repos.d/MariaDB.repo
```

Figura 29 Agregación de Repositorios de MariaDB

#### 7.1.4. Deshabilitar SELinux

Es preferible deshabilitar la configuración de SELinux.

**Comando:**

```
nano /etc/selinux/config
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ nano /etc/selinux/config
```

Figura 30 Apertura de Archivo de Configuración de SELinux

```
root@openstack:~  
GNU nano 2.3.1 File: /etc/selinux/config  
# This file controls the state of SELinux on the system.  
# SELINUX= can take one of these three values:  
#   enforcing - SELinux security policy is enforced.  
#   permissive - SELinux prints warnings instead of enforcing.  
#   disabled - No SELinux policy is loaded.  
SELINUX=disabled  
# SELINUXTYPE= can take one of three two values:  
#   targeted - Targeted processes are protected,  
#   minimum - Modification of targeted policy. Only selected processes are pr$  
#   mls - Multi Level Security protection.  
SELINUXTYPE=targeted
```

Figura 31 Modificación de Archivo SELinux

#### 7.1.5. Deshabilitar servicios de Firewall, Postfix, NetworkManager

Para evitar inconvenientes a la hora de instalar OpenStack es recomendable detener y deshabilitar el servicio de firewall de CentOS.

**Comando:**

```
systemctl stop firewalld
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ systemctl stop firewalld
```

Figura 32 Detener Servicio Firewall

**Comando:**

```
systemctl disable firewalld
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo systemctl disable firewalld
```

Figura 33 Deshabilitación servicio firewalld

Posteriormente procedemos con la Deshabilitación del servicio de correo por defecto postfix.

**Comando:**

```
sudo systemctl stop postfix
```

---

```
mlopez@openstack:~  
[mlopez@openstack ~]$ systemctl stop postfix
```

Figura 34 Detener Servicios Postfix

**Comando:**

```
sudo systemctl stop postfix
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo systemctl disable postfix
```

Figura 35 Deshabilitación de Servicio Postfix

El último servicio que debemos deshabilitar es el servicio de Network Manager.

**Comando:**

```
sudo systemctl stop NetworkManager
```

---

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo systemctl stop NetworkManager
```

Figura 36 Detener Servicios NetworkManager

**Comando:**

```
sudo systemctl disable NetworkManager
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo systemctl disable NetworkManager
```

Figura 37 Deshabilitación del Servicio NetworkManager

### 7.1.6. Habilitación e Inicio de Servicio de Network

OpenStack usa el servicio Network así que debemos habilitarlo e Inicialo.

**Comando:**

```
sudo systemctl enable network
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo systemctl enable network
```

*Figura 38 Habilitación de Servicio Network*

**Comando:**

```
sudo systemctl start network
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo systemctl start network
```

*Figura 39 Inicio de Servicio Network*

### 7.1.7. Instalación de OpenStack Train

Cumplidos los pasos previos debemos proceder instalando OpenStack Stein con su respectivo comando.

Como paso principal debemos adicionar un certificado para que nuestra instancia de Openstack sea segura y la conexión se realice a través de HTTPS.

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo openssl genrsa -out /etc/pki/tls/private/openstack.key 1024
```

*Figura 40 Creación de Certificado*

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo openssl req -new -key /etc/pki/tls/private/openstack.key -out /etc/pki/tls/openstack.csr
```

*Figura 41 Creación del key para el certificado creado*

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo cp /etc/pki/tls/openstack.csr root@172.17.1.37:/home/certificadosXfirmar/openstack.csr
```

*Figura 42 Firma del Certificado*

**Comando:**

```
sudo yum install centos-release-openstack-train -y
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo yum install -y centos-release-openstack-train_
```

Figura 43 Instalación de OpenStack Train

Tenemos que instalar Packstack para tener ciertas partes de OpenStack específicas para nuestras utilidades.

### Comando:

```
sudo yum install openstack-packstack -y
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo yum install -y openstack-packstack_
```

Figura 44 Instalación de OpenStack Packstack

Necesitamos generar el respectivo archivo de configuración que será usado para instalar OpenStack.

### Comando:

```
sudo packstack --get-answer-file=/root/answer.txt
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo packstack --gen-answer-file=/root/answer.txt  
Packstack changed given value to required value /root/.ssh/id_rsa.pub  
Additional information:  
* Parameter CONFIG_NEUTRON_L2_AGENT: You have chosen OVN Neutron backend. Note  
that this backend does not support the VPNaaS or FWaaS services. Geneve will be  
used as the encapsulation method for tenant networks
```

Figura 45 Creación de Archivo de Configuración de OpenStack

Abrimos el archivo generado y procedemos modificando los campos requeridos.

### Comando:

```
sudo nano /root/answer.txt
```

```
mlopez@openstack:~  
[mlopez@openstack ~]$ sudo nano /root/answer.txt
```

Figura 46 Apertura de Archivo con nano

### Modificaciones:

```
CONFIG_PROVISION_DEMO=n  
CONFIG_KEYSTONE_ADMIN_PW=xxx
```

```

CONFIG_HORIZON_SSL=y
CONFIG_NEUTRON_OVS_BRIDGE_MAPPINGS=extnet:br-ex
CONFIG_NEUTRON_OVS_BRIDGE_IFACES=br-ex:ens33
-----
mlopez@openstack:~
-----
GNU nano 2.3.1 File: /root/answer.txt

[general]

# Path to a public key to install on servers. If a usable key has not
# been installed on the remote servers, the user is prompted for a
# password and this key is installed so the password will not be
# required again.
CONFIG_SSH_KEY=/root/.ssh/id_rsa.pub

# Default password to be used everywhere (overridden by passwords set
# for individual services or users).
CONFIG_DEFAULT_PASSWORD=

# The amount of service workers/threads to use for each service.
# Useful to tweak when you have memory constraints. Defaults to the
# amount of cores on the system.
CONFIG_SERVICE_WORKERS=%{::processorcount}

# Specify 'y' to install MariaDB. ['y', 'n']
CONFIG_MARIADB_INSTALL=y
-----

```

*Figura 47 Archivo /root/answer.txt*

Luego de tener el archivo de configuración con los parámetros definidos procedemos a ejecutar la instalación mediante packstack.

### **Comando:**

```
sudo packstack -answer-file=/root/answer.txt
```

```

mlopez@openstack:~
-----
[mlopez@openstack ~]$ sudo packstack --answer-file=/root/answer.txt █
-----

```

*Figura 48 Arranque de Packstack*

Después de completar la instalación se podrá acceder al frontend de Openstack mediante el usuario y la contraseña proporcionadas al momento de crear el archivo de configuración packstack.

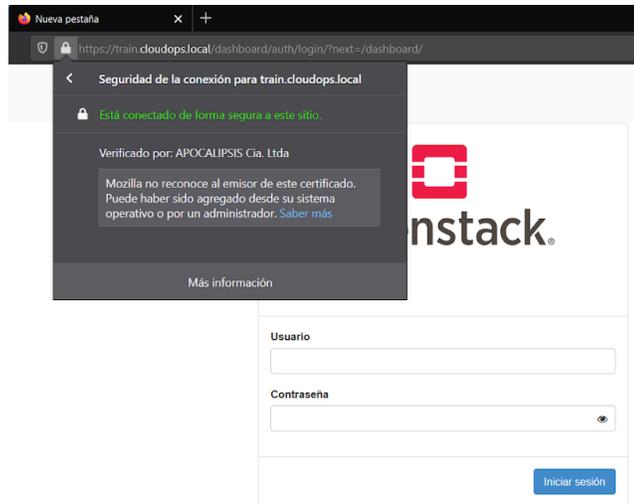


Figura 49 Frontend OpenStack

## 7.2. ANEXO 2: INSTALACIÓN DE VCLLOUD DIRECTOR 10

### 7.2.1. Asignación de direcciones IP al registro de DNS Local.

La arquitectura de vCloud requiere dos direcciones IP, la primera se usará para procesos y conexiones como son la base de datos y demás recursos de consumo interno, y la segunda dirección IP permitirá el acceso al dashboard de administración para los usuarios de las organizaciones y el administrador de vCloud Director.

Los registros DNS deben ser de tipo A como se indica a continuación:

 intcloudapp	Host (A)	10.10.11.116	
 cloudapp	Host (A)	10.10.11.117	static

*Figura 50 Registros DNS Tipo A*

### 7.2.2. Configuración del Recurso Compartido (NFS)

Desde la versión de vCloud 9.5 se requiere un sistema que brinde los servicios NFS, este recurso es importante en el despliegue de la infraestructura ya que almacenara los registros de la base de datos de vCloud. En nuestra infraestructura hemos decidido optar por un recurso de NFS sobre el sistema operativo Windows mismo en el que se ejecuta el DNS, pero la elección de distribución o localización de este sistema es opcional y a elección del administrador de red. Mediante la instalación de Roles y Características agregamos los roles de NFS en nuestro servidor.

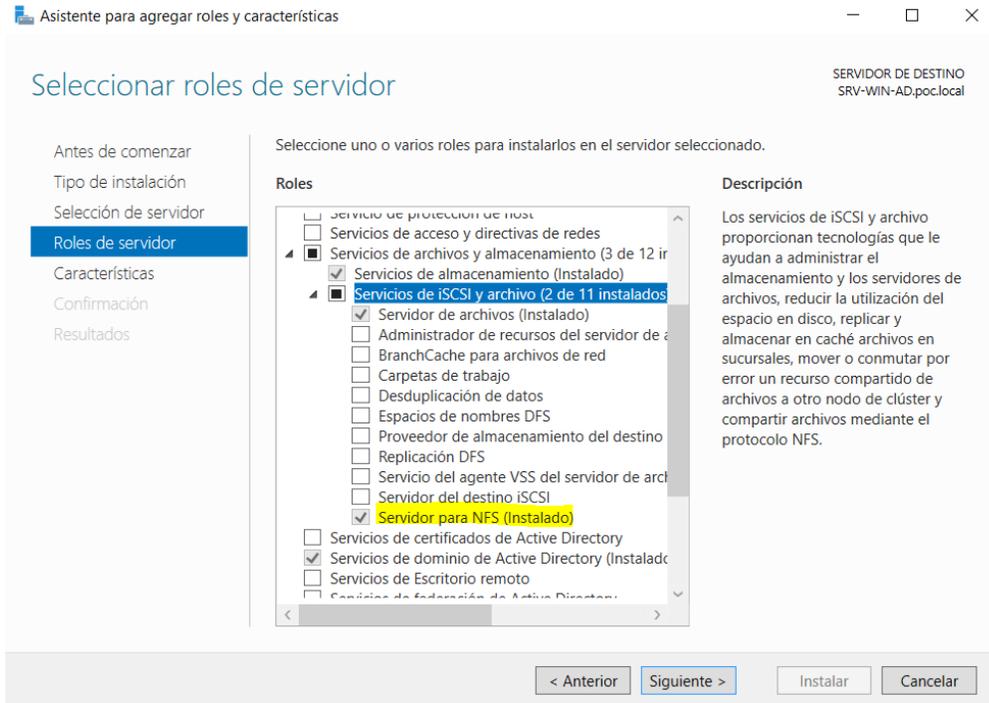


Figura 51 Adición Rol Servidor NFS

Elegimos una carpeta de preferencia en una partición independiente con una capacidad mínima de 20 GB y la compartimos mediante NFS de Windows.

Nota: podemos asignar los permisos únicamente para que el Nodo vCloud pueda escribir agregando las dos IPs de vCloud

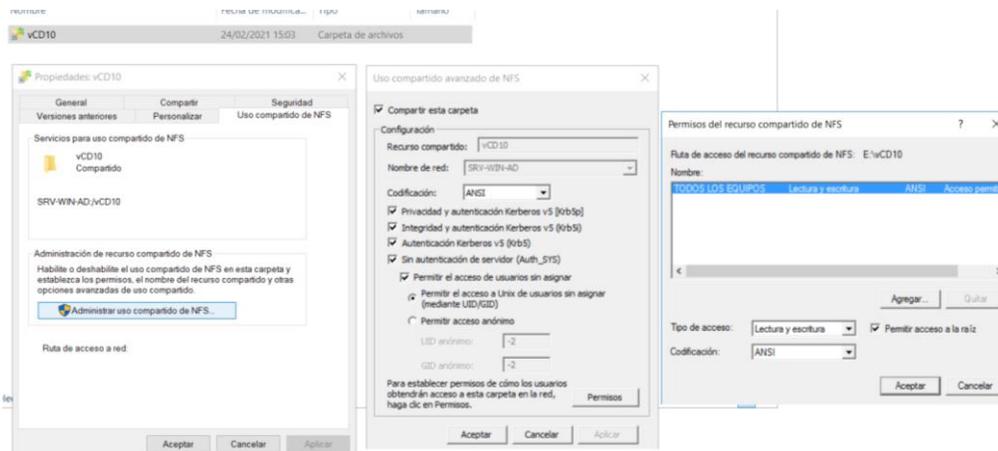


Figura 52 Configuración NFS Server

### 7.2.3. Desplegamos la plantilla de vCloud Director 10

vCloud Director tiene varias opciones de despliegue, pero la más simplificada para nuestra arquitectura y la más eficiente es la versión de ovf misma que puede ser usada inclusive en entornos de producción.

Este despliegue se realiza mediante la interfaz de administración del vCenter y el proceso es común a una plantilla estándar, siendo necesario los parámetros como IPs, ruta NFS, credenciales para el root, el administrador y rutas para las conexiones de red. A continuación, expondremos las configuraciones que se adaptan a nuestra arquitectura.

#### 7.2.3.1. Obtenemos la ruta del archivo ovf

Este archivo esta en formato ova u ovf y se lo puede descargar de las páginas de VMWare para nuestro caso usamos la versión 10 de la solución.

Desplegamos la Máquina Virtual como Plantilla:

Implementar plantilla de OVF

- 1 Seleccione una plantilla d...
- 2 Seleccionar un nombre y ...
- 3 Seleccionar un recurso in...
- 4 Revisar detalles
- 5 Seleccionar almacenamie...
- 6 Listo para completar

Seleccione una plantilla de archivo OVF

Seleccione una plantilla de archivo OVF en una URL remota o un sistema de archivos local

Introduzca una URL para descargar e instalar el paquete de OVF desde Internet o desplácese hasta una ubicación accesible desde el equipo, como un disco duro local, un recurso de red compartido o una unidad de CD/DVD.

URL

http | https://remoteserver-address/filetodeploy.ovf | .ova

Archivo local

Elegir archivos VMware\_Clou...253\_OVF10.ova

Figura 53 Selección Plantilla OVF

Asignamos un nombre para la instancia o máquina virtual:

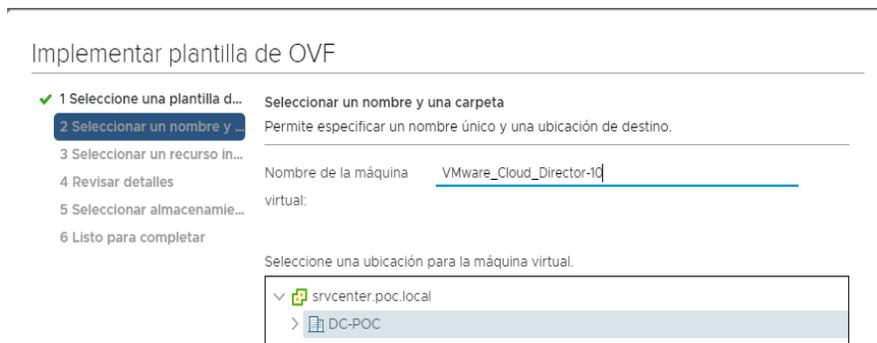


Figura 54 Asignación Nombre VM

### 7.2.3.2. Seleccionamos un Recurso en el Clúster

Podemos seleccionar un recurso si es el caso o simplemente asignarle al clúster:



Figura 55 Selección Pool de Recursos

### 7.2.3.3. Validamos la Configuración

La siguiente pantalla permitirá validar los detalles de la plantilla:



Figura 56 Validación Plantilla OVF

#### 7.2.3.4. Seleccionamos el tipo de configuración

Realizamos esta selección en base al dimensionamiento para el cual será adaptado la solución de vCloud Director. Podemos seleccionar entre las distintas opciones en nuestro caso al ser un ambiente de pruebas seleccionamos la configuración Primaria “small”.



Figura 57 Selección Tipo Configuración

#### 7.2.3.5. Seleccionamos un recurso de Almacenamiento

En nuestro caso el único recurso de almacenamiento disponible es el de HOST, por lo que seleccionamos este recurso, pero se debe tener en cuenta que tipo de discos conforman este storage para evitar cuellos de botella en procesos de lectura y escritura de disco, en caso de tener un sistema de almacenamiento de alto rendimiento recomendamos usar siempre el que mejor resultados brinde. Es muy importante elegir la directiva de almacenamiento para nuestro caso elegimos la opción Fino (Tiny).

## Implementar plantilla de OVF

1 Seleccione una plantilla d... Selecciónar almacenamiento  
2 Seleccione un nombre y ... Selecciónar el almacenamiento para los archivos de configuración y de disco  
3 Seleccione un recurso in...  
4 Revisar detalles  Cifrar esta máquina virtual (Requiere un servidor de administración de claves)  
5 Contratos de licencia Seleccione el formato de disco virtual: **Aprovisionamiento fino**  
6 Configuración Directiva de almacenamiento de máquina virtual: Valor predeterminado de almacené...  
**7 Selecciónar almacenamie...**  
8 Selecciónar redes  
9 Personalizar plantilla  
10 Listo para completar

Nombre	Capacidad	Aprovisionado	Libre	Ti
Storage	2 TB	1,2 TB	1,81 TB	VI
sysesx1	4,5 GB	1,41 GB	3,09 GB	VI

Figura 58 Selección Recurso de Almacenamiento

### 7.2.3.6. Seleccionamos las interfaces de Red

Como ya especificamos anteriormente el sistema de vCloud Director requiere de dos interfaces la primera para procesos internos y la segunda para acceso a la interfaz de administración.

## Implementar plantilla de OVF

1 Seleccione una plantilla d... Selecciónar redes  
2 Seleccione un nombre y ... Permite seleccionar una red de destino para cada red de origen.  
3 Seleccione un recurso in...  
4 Revisar detalles  
5 Contratos de licencia  
6 Configuración  
7 Selecciónar almacenamie...  
**8 Selecciónar redes**  
9 Personalizar plantilla  
10 Listo para completar

Red de origen	Red de destino
eth1 Network	NET ADMINISTRATION
eth0 Network	NET ADMINISTRATION

2 items

Configuración de asignación de IP

Asignación de IP: Estático: manual

Protocolo IP: IPv4

Figura 59 Selección Interfaces Red

### 7.2.3.7. Configuración de parámetros

La plantilla requiere que se asignen contraseñas y direcciones IP para los adaptadores de red, además de rutas y dominios, a continuación, se puede observar la configuración asignada a este proyecto:

## Implementar plantilla de OVF

1 Seleccione una plantilla d... **Personalizar plantilla**  
 2 Seleccionar un nombre y ... Personaliza las propiedades de implementación de esta solución de software.  
 3 Seleccionar un recurso in...  
 4 Revisar detalles  
 5 Contratos de licencia  
 6 Configuración  
 7 Seleccionar almacenamie...  
 8 Seleccionar redes  
 9 **Personalizar plantilla**  
 10 Listo para completar

Todas las propiedades tienen valores válidos X

> VCD Appliance Settings	5 settings
> VCD Configure - Required only for "primary" appliances	8 settings
> Additional Networking Properties	2 settings
> Networking Properties	8 settings

## Implementar plantilla de OVF

1 Seleccione una plantilla d...  
 2 Seleccionar un nombre y ...  
 3 Seleccionar un recurso in...  
 4 Revisar detalles  
 5 Contratos de licencia  
 6 Configuración  
 7 Seleccionar almacenamie...  
 8 Seleccionar redes  
 9 Personalizar plantilla  
 10 **Listo para completar**

Tipo de aprovisionamiento	Implementar desde plantilla
Nombre	VMware_Cloud_Director-10
Nombre de plantilla	VMware_Cloud_Director-10.1.0.4837-15967253_OVF10
Tamaño de descarga	1,1 GB
Tamaño en disco	1,9 GB
Carpeta	DC-POC
Recurso	RGRP-SERVICIOS
Asignación de almacenamiento	1
Todos los discos	Almacén de datos: Storage; formato: Aprovisionamiento fino
Asignación de red	2
eth1 Network	NET ADMINISTRATION
eth0 Network	NET ADMINISTRATION
Configuración de asignación de IP	
Protocolo IP	IPV4
Asignación de IP	Estático: manual

CANCEL
BACK
FINISH

Figura 60 Configuración Final Parámetros

## 7.2.4. Interfaz de Administración

Para el acceso a la interfaz de vCloud como usuario administrador debemos usar el siguiente enlace ya sea por nombre de dominio o ip.

<https://IPoDNS/provider/cloud>

Esta URL nos dirigirá a una interfaz como la siguiente en donde deberemos configurar la conexión a NSX y vCenter para poder integrar los servicios y levantar la infraestructura de vCloud Director 10.

En la pestaña de Recursos de vSphere deberemos agregar la instancia de vCenter y NSX-Manager, únicamente necesitaremos las IPS o DNS, URLs, usuario y la contraseña de las soluciones.

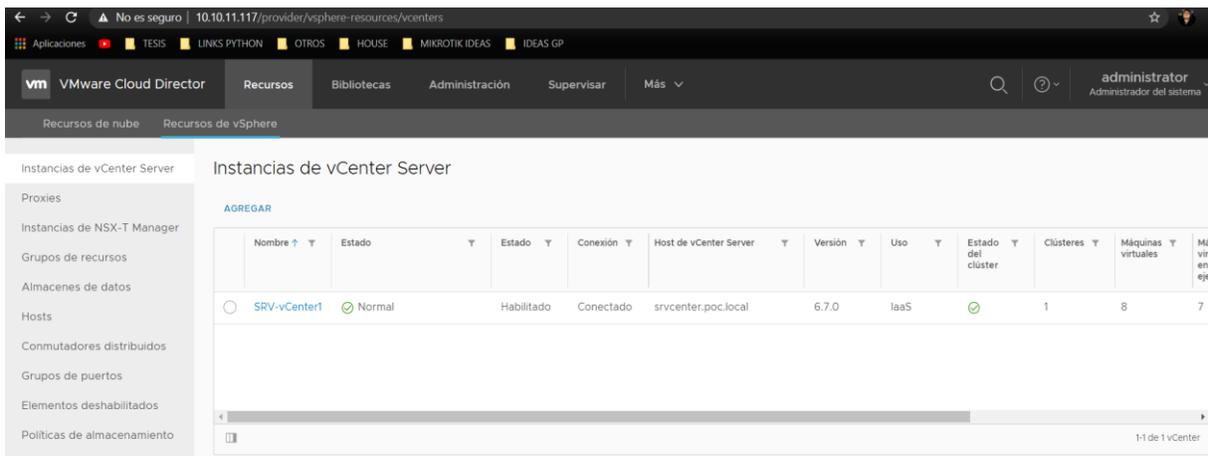


Figura 61 Interfaz Administración VCloud Director

## 7.2.5. Organizaciones de vCloud

Las organizaciones de vCloud permiten distribuir los recursos segmentando los centros de datos de manera virtual. Estas organizaciones serán a las que los usuarios que consuman los recursos de cloud computing accederán y gestionaran sus centros de datos de manera organizada, segura y centralizada.

Para nuestro laboratorio únicamente crearemos una organización y a esta asignaremos los recursos necesarios para su correcto funcionamiento.

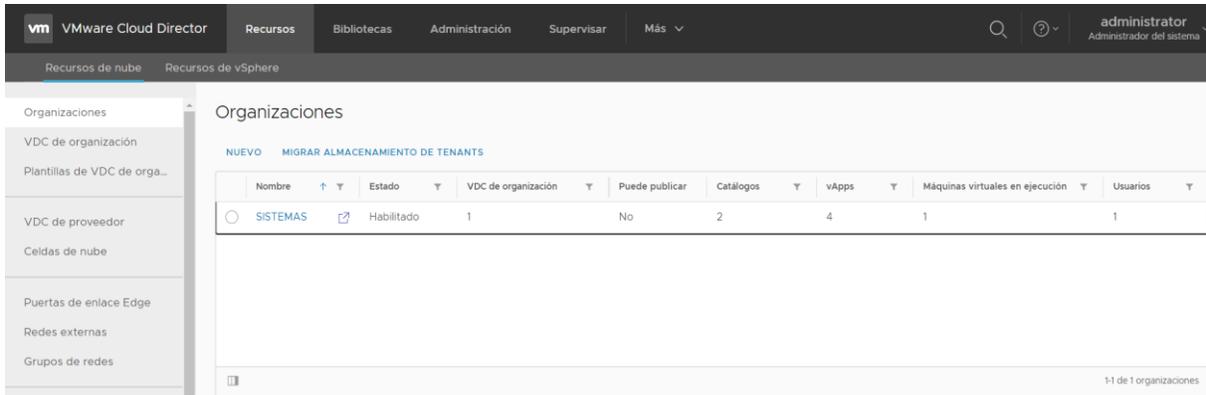


Figura 62 Organizaciones VCloud

## 7.2.6. Asignación de uso de Cloud

Para la solución de prueba hemos elegido un centro de datos virtual para la organización de prueba, misma que será gestionada bajo la normativa pago por uso. No hemos asignado limitaciones, pero el administrador deberá elegir la opción que mejor se adapte al servicio que ofrecerá.

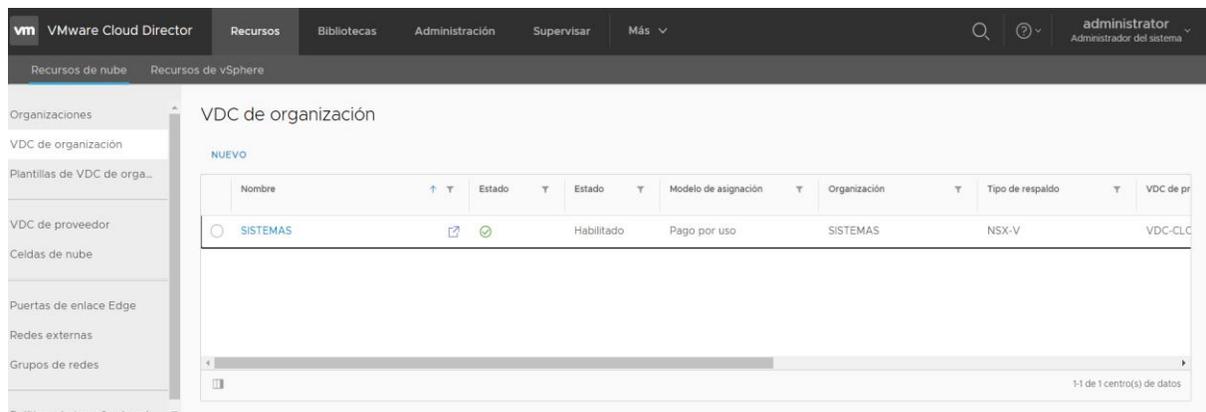


Figura 63 VDC de Organización VCloud

## 7.2.7. Acceso a la plataforma “Usuarios Organizativos”

Para que un usuario pueda acceder a su organización lo deberá hacer mediante la siguiente url:

<https://IPoDNS/tenant>, esta ruta le llevara a la interfaz de autenticación del usuario y le dara acceso a su Organización. En donde podrá visualizar los recursos de computo que tiene ejecutándose.



Figura 64 Ingreso Nombre de Organización

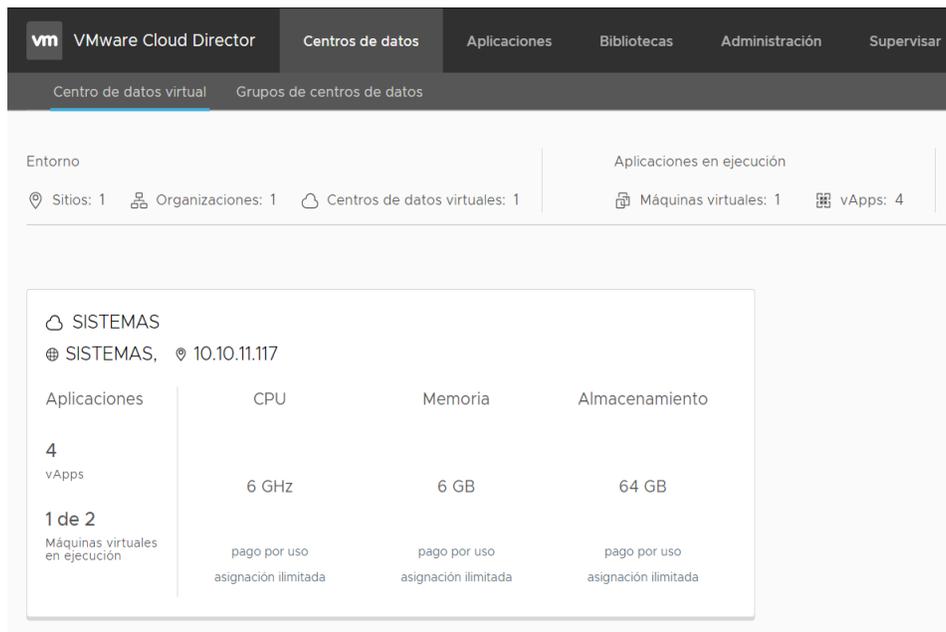


Figura 65 Centro de Datos VCloud

vm VMware Cloud Director    Centros de datos    Aplicaciones    Bibliotecas    Administración    Supervisar

← Todos los centros de datos virtuales    Sitio: 10.10.11.117    Organización: SISTEMAS    Centro de datos: SISTEMAS

### Máquinas virtuales

Buscar por: Nombre    FILTRADO AVANZADO

2 máquinas virtuales    Caducado: false    Borrar todos los filtros

[NUEVA MÁQUINA VIRTUAL](#)

#### SRV-AD

**Encendido**  
Consola de máquina virtual

Concesión	Nunca caduca
Creada el	23/02/2021 08:48:11 p. m.
Propietario	csuco@est.ups.edu.ec
vApp	CLOUD-SERVICIOS
SO	Microsoft Windows Server 201...

CPU	Almacenam...	Memoria	Redes
4	44 GB	4 GB	①

[INSIGNIAS](#)

[ACCIONES](#)    [DETALLES](#)

#### SRV-Ubuntu

**Apagado**  
Consola de máquina virtual

Concesión	Nunca caduca
Creada el	03/02/2021 04:29:18 p. m.
Propietario	csuco@est.ups.edu.ec
vApp	VAPP-GIHP4C
SO	Ubuntu Linux (64-bit)

CPU	Almacenam...	Memoria	Redes
2	12 GB	2 GB	①

[INSIGNIAS](#)

[ACCIONES](#)    [DETALLES](#)

Figura 66 Máquinas Virtuales VCloud

### 7.3. Anexo 3: ARCHIVOS DE AUTENTICACIÓN PARA LA API DE OPENSTACK

Este archivo puede ser usado para conectar las APIS de Openstack y se encuentra en la siguiente ubicación en la interfaz web del proyecto de Openstack:

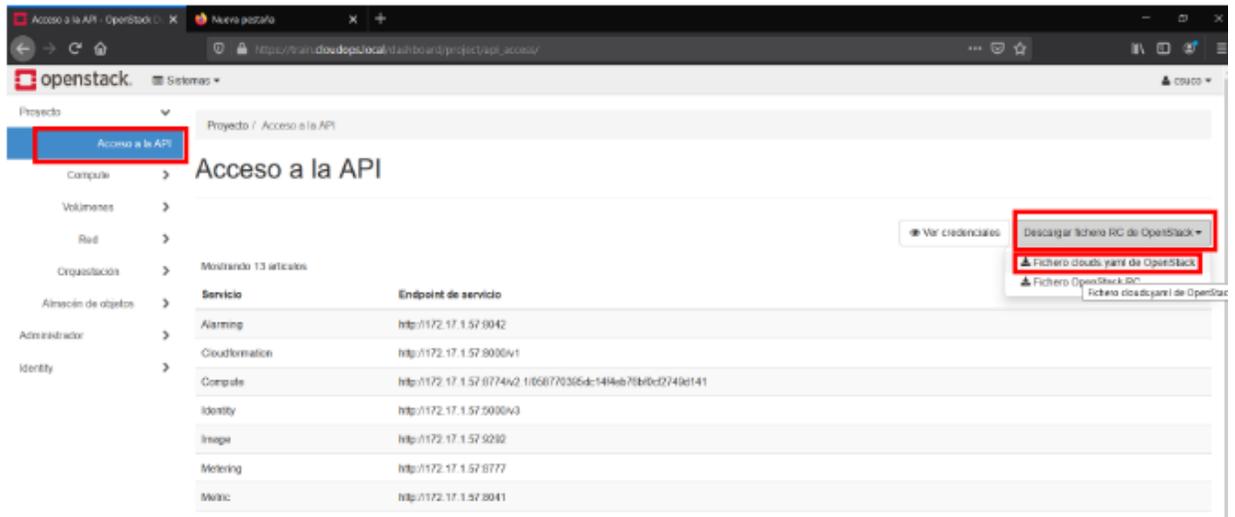


Figura 67 Archivo de Acceso API OpenStack

## 7.4. Anexo 4: SCRIPT DE INSTALACIÓN DE DRIVERS PARA SISTEMAS LINUX

El siguiente script se desarrolló para permitir instalar y configurar los drivers virtio en los sistemas operativos de la familia linux sean estas distribuciones Debian, Ubuntu o Centos sistemas en los cuales el código fue probado y es funcional.

```
#!/bin/bash
DNF_CMD=$(which dnf)
YUM_CMD=$(which yum)
APT_GET_CMD=$(which apt)
if [[ ! -z $DNF_CMD ]]; then
    dnf update -y
    dnf install -y cloud-init dracut-config-generic cloud-utils cloud-utils-growpart
    dnf remove -y open-vm-tools hyperv-daemons
    mkinitrd -f --allow-missing --with=xen-blkfront --preload=xen-blkfront --
with=virtio_blk --preload=virtio_blk --with=virtio_pci --preload=virtio_pci --
with=virtio_console --preload=virtio_console /boot/initramfs-$(uname -r).img $(uname -
r)
    dracut --force --no-hostonly
elif [[ ! -z $YUM_CMD ]]; then
    yum update -y
    yum install -y cloud-init dracut-config-generic cloud-utils cloud-utils-growpart
    yum remove -y open-vm-tools hyperv-daemons
    mkinitrd -f --allow-missing --with=xen-blkfront --preload=xen-blkfront --
with=virtio_blk --preload=virtio_blk --with=virtio_pci --preload=virtio_pci --
with=virtio_console --preload=virtio_console /boot/initramfs-$(uname -r).img $(uname -r)
    dracut --force --no-hostonly
elif [[ ! -z $APT_GET_CMD ]]; then
    apt update -y
    apt-get install -y cloud-init dracut-config-generic cloud-utils
    apt-get remove -y open-vm-tools hyperv-daemons
    echo -e "xen-blkfrontvirtio_blkvirtio_pci" >> /etc/initramfs-
tools/modules
    mkinitramfs -o /boot/initrd.img-$(uname -r)
    dracut --force --no-hostonly
else
    exit 1;
fi
```

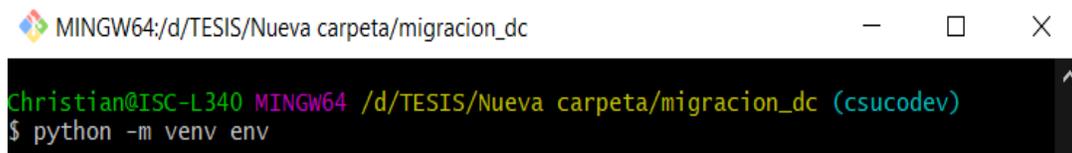
*Script 1 Instalación Drivers Sistemas Linux*

## 7.5. Anexo 5: CONFIGURACIÓN DEL ENTORNO VIRTUAL PARA DESARROLLO EN PYTHON

Para crear un entorno virtualizado de nuestro Python, debemos ubicarnos en la carpeta o directorio donde queramos desplegar nuestro entorno virtual y seguimos los siguientes pasos:

- Comando 1:

Comando para crear el entorno virtual: `python -m venv env`



```
MINGW64:/d/TESIS/Nueva carpeta/migracion_dc
Christian@ISC-L340 MINGW64 /d/TESIS/Nueva carpeta/migracion_dc (csucodev)
$ python -m venv env
```

Figura 68 Creación Entorno Virtual Python

- Comando 2:

Comando para activar el entorno virtual: `. env/Scripts/activate`



```
Christian@ISC-L340 MINGW64 /d/TESIS/Nueva carpeta/migracion_dc (csucodev)
$ . env/Scripts/activate|
```

Figura 69 Activación Entorno Virtual

- Comando 3:

Comando para instalar las librerías necesarias: `pip install -r requeriments.txt`



```
Christian@ISC-L340 MINGW64 /d/TESIS/Nueva carpeta/migracion_dc (csucodev)
$ pip install -r requeriments.txt|
```

Figura 70 Instalación Librerías Migracion