

▪ Trabajo Fin de Grado ▪

Ingeniería de Computadores

---

Implantación de una solución de escritorios virtuales con  
OpenStack

Autor: Jonathan Fernández Ramos

Directores de Proyecto: Alexander Mendiburu Alberro

José Miguel Alonso

Septiembre 2018



# RESUMEN

---

Con el propósito de proporcionar a los alumnos, profesorado o cualquier otra persona autorizada de la Facultad de Informática de San Sebastián/Donostia un espacio de trabajo sobre el que disponer y almacenar sus documentos, aplicaciones y cualquier otro servicio que estimen oportuno, ha surgido la idea de este proyecto, en donde la virtualización de escritorio será donde fundamentemos nuestro estudio.

Para alcanzar esta meta, se ha llevado a un cabo un estudio minucioso de las diferentes soluciones de virtualización de escritorio que proporciona el mercado actual, buscando aquella o aquellas que mejor se adecuen a nuestras exigencias. Esto implicará tener que distinguir las diferentes formas en las que se puede presentar esa solución de virtualización de escritorio, teniendo que saber seleccionar aquella que se considere más adecuada en base a los objetivos que se hayan marcado. Una vez hemos logrado alcanzar una posible solución, tendremos que verificar por medio de un conjunto de pruebas que efectivamente es una resolución válida. Además, esto nos permitirá conocer más de cerca la plataforma junto con la administración que requiere, entre otros aspectos de igual importancia.

Este proyecto ha sido realizado para la titulación de Ingeniería Informática en la especialidad de computadores, en la Facultad de Informática de la Universidad UPV/EHU de Donostia/San Sebastián.

# INDICE

---

RESUMEN .....	ii
LISTA DE FIGURAS Y TABLAS .....	vi
AGRADECIMIENTOS .....	xi
INTRODUCCIÓN .....	1
1.1. Visión general .....	2
1.2. Objetivos .....	3
PLANIFICACIÓN .....	5
2.1. Estructura de descomposición del trabajo .....	6
2.2. Tareas del proyecto .....	8
2.3. Características e identificación de entregables.....	9
2.4. Planificación .....	10
2.4.1. Diagrama de tiempo.....	10
2.4.2. Dedicación.....	11
2.4.3. Desviaciones.....	12
2.5. Metodología de trabajo.....	13
2.6. Plan de comunicación.....	14
2.7. Gestión de riesgos .....	14
CONCEPTOS .....	17
3.1. Introducción .....	18
3.2. Virtualización.....	18
3.3. Escritorio virtual .....	19
3.3.1. Desplegar escritorios virtuales .....	21
3.3.2. Recomendaciones .....	22
3.4. Facultad de Informática de San Sebastián .....	23
3.4.1. Introducción .....	23
3.4.2. Infraestructura .....	23
3.5. Necesidades prácticas .....	25
ESTUDIO DE ALTERNATIVAS .....	27
4.1. Introducción .....	28
4.2. Horizon 7 - VMware.....	30
4.3. VirtualCable – UDS Enterprise .....	31
4.4. Casos de éxito.....	34
4.4. 1. Introducción .....	34
4.4.2. Universidad de A Coruña.....	34
4.4.3. Universidad de Sevilla .....	34

4.4.4. Universidad de Murcia .....	35
4.4.5. Universidad Autónoma de Madrid .....	35
4.5. Conclusiones .....	36
OPENSTACK.....	37
5.1. OpenStack .....	38
5.2. Componentes .....	39
5.3. Distribuciones.....	41
5.4. Decisión .....	45
PRUEBAS REALIZADAS.....	47
6.1. Introducción .....	48
6.2. Prueba 1: Instanciación de la imagen y aplicación del protocolo SSH .....	48
6.2.1. Secure Shell (SSH) .....	48
6.2.2. Procedimiento.....	49
6.3. Prueba 2: Instanciación de la imagen y aplicación de escritorio VNC.....	52
6.3.1. VNC .....	52
6.3.2. VNC en OpenStack .....	53
6.3.3. Procedimiento.....	54
OTROS ASPECTOS.....	59
7.1. LDAP .....	60
7.2. Coste.....	60
CONCLUSIONES Y TRABAJO FUTURO .....	63
8.1. Valoración del resultado alcanzado .....	64
8.2. Valoración personal .....	65
8.3. Lecciones aprendidas .....	65
8.4. Posibles líneas de trabajo .....	66
BIBLIOGRAFÍA .....	67
9.1. Glosario .....	68
9.2. Referencias .....	72
9.3. Vídeos.....	77
APÉNDICE .....	79
A. Cloud computing.....	80
B. Planificación del tiempo .....	82
B.1. Introducción.....	82
B.2. Estimación de tiempo .....	83
C. Soluciones de virtualización .....	90
C.1. Introducción.....	90
C.2. Soluciones VDI .....	90
C.3. Comercialización VDI.....	94

C.4. Conclusiones .....	101
D. Manual de instalación de OpenStack PackStack en el Sistema Operativo CentOS .....	102
D.1. Introducción.....	102
D.2. Instalación.....	103
D.3. Soporte de ayuda.....	128
E. Administración básica de la Plataforma OpenStack.....	133
E.1. Introducción.....	133
E.2. Conceptos de Administración .....	135
E.3. Descripción de los Componentes .....	139
E.4. Instrucciones de administración.....	146
F. Instanciar y conexión remota SSH .....	157
F.1. Introducción .....	157
F.2. Administración – Fedora .....	157
F.3. Otras cuestiones .....	179
G. Instanciación y aplicación de escritorio VNC.....	184
G.1. Introducción .....	184
G.2. Conceptos previos .....	184
G.3. OpenStack – VNC.....	185
G.4. Dificultades.....	199
H. LDAP.....	202
H.1. Introducción .....	202
H.2. Conceptos previos .....	203
H.3. Instalación.....	204
H.4. Conclusiones.....	213

# LISTA DE FIGURAS Y TABLAS

---

## ILUSTRACIONES:

Ilustración 1: Desglose del trabajo .....	7
Ilustración 2: Metodología de trabajo.....	13
Ilustración 3: Escritorio Virtual por cada usuario .....	19
Ilustración 4: Arquitectura UDS Enterprise .....	31
Ilustración 5: Infraestructura UDS Enterprise .....	33
Ilustración 6: Arquitectura OpenStack .....	38
Ilustración 7: Componentes OpenStack.....	39
Ilustración 8: Infraestructura VMware Integrated OpenStack .....	41
Ilustración 9: Infraestructura Red Hat OpenStack Platform .....	42
Ilustración 10: Mirantis Inc. OpenStack .....	43
Ilustración 11: Estructura - Ubuntu OpenStack.....	44
Ilustración 12: Procedimiento SSH - Topología de red .....	49
Ilustración 13: Procedimiento SSH - Resultado reconocimiento de la instancia .....	51
Ilustración 14: Procedimiento SSH - Resultado conexión.....	51
Ilustración 15: Sistema de funcionamiento - OpenStack VNC Proxy .....	53
Ilustración 16: Procedimiento VNC - Nueva regla grupo de seguridad .....	55
Ilustración 17: Procedimiento VNC – Instancia .....	55
Ilustración 18: Procedimiento VNC - Resultado reconocimiento instancia .....	56
Ilustración 19: Procedimiento VNC - Nueva conexión VNC Client.....	56
Ilustración 20: Procedimiento VNC - Conexión alcanzada.....	57
Ilustración 21: Modelos de servicio en la nube .....	80
Ilustración 22: Cloud Computing .....	81
Ilustración 23: Arquitectura flexVDI .....	92
Ilustración 24: Microsoft VDI - Estructura RDS.....	93
Ilustración 25: Arquitectura Qindel Group –QVD.....	95
Ilustración 26: Cloud Desktop Infraestructure .....	96
Ilustración 27: Arquitectura OVH .....	97
Ilustración 28: VirtualBox - Creación Nueva Máquina Virtual .....	104
Ilustración 29: VirtualBox - Nombre y Sistema Operativo .....	104
Ilustración 30: VirtualBox - Espacio de memoria (RAM) .....	105
Ilustración 31: VirtualBox - Asignación disco virtual .....	105
Ilustración 32: VirtualBox - Tipo de archivo de disco duro virtual.....	106
Ilustración 33: VirtualBox - Almacenamiento en unidad de disco duro físico .....	106
Ilustración 34: VirtualBox - Ubicación del archivo y tamaño .....	107
Ilustración 35: VirtualBox - Máquina creada.....	107
Ilustración 36: VirtualBox - Configuración.....	108
Ilustración 37: VirtualBox - Configuración Red.....	108
Ilustración 38: VirtualBox - Configuración de almacenamiento .....	109
Ilustración 39: VirtualBox - Iniciar máquina virtual .....	109
Ilustración 40: CentOS - Selección idioma .....	110
Ilustración 41: CentOS - Destino de la instalación.....	110
Ilustración 42: CentOS - Opciones de almacenamiento.....	111
Ilustración 43: CentOS - Punto de montaje "/boot" .....	111
Ilustración 44: CentOS - Punto de montaje "swap" .....	112
Ilustración 45: Punto de montaje "/" .....	112
Ilustración 46: CentOS - Aceptación cambios de punto de montaje.....	113

Ilustración 47: CentOS - Opción KDUMP.....	113
Ilustración 48: CentOS - Deshabilitar KDUMP .....	114
Ilustración 49: CentOS - Seleccionar Red & Nombre de Equipo.....	114
Ilustración 50: CentOS - Activación Ethernet .....	115
Ilustración 51: CentOS - Ajustes de IPv4 .....	115
Ilustración 52: CentOS - Cambios tarjeta de red y nombre de máquina .....	116
Ilustración 53: CentOS - Nombre máquina .....	116
Ilustración 54: CentOS - Inicio instalación.....	116
Ilustración 55: CentOS - Contraseña root y creación nuevo usuario.....	117
Ilustración 56: CentOS - Usuario Root.....	117
Ilustración 57: CentOS - Comprobación dirección IP .....	118
Ilustración 58: PuTTY - Comprobación.....	118
Ilustración 59: PuTTY - Deshabilitar firewall y Network Manager .....	119
Ilustración 60: PuTTY - Editar fichero "selinux" .....	119
Ilustración 61: PuTTY - Deshabilitar "selinux" .....	119
Ilustración 62: PuTTY - Reinicio máquina.....	119
Ilustración 63: PuTTY - Restablecer conexión .....	120
Ilustración 64: PuTTY - Comprobación "selinux" deshabilitado .....	120
Ilustración 65: OpenStack - Instalación versión Ocata .....	121
Ilustración 66: OpenStack - Actualizar sistema .....	121
Ilustración 67: OpenStack - Reinicio sistema .....	121
Ilustración 68: OpenStack - Instalación OpenStack PackStack .....	121
Ilustración 69: OpenStack - Configuración e Instalación final OpenStack PackStack.....	122
Ilustración 70: OpenStack - Instalación completada satisfactoriamente.....	122
Ilustración 71: Configuración - Tarjetas de red .....	123
Ilustración 72: Configuración - Fichero de configuración tarjeta "enp0s3".....	123
Ilustración 73: Configuración - Datos tarjeta "enp0s3".....	123
Ilustración 74: Configuración - Fichero de configuración tarjeta "br-ex" .....	124
Ilustración 75: Configuración - Datos tarjeta "br-ex" .....	124
Ilustración 76: Configuración - Reinicio del servicio de red .....	124
Ilustración 77: Configuración - Comprobar tarjetas de red.....	125
Ilustración 78: Acceso - Fichero Usuarios .....	126
Ilustración 79: Acceso - Usuario y Contraseña Dashboard.....	126
Ilustración 80: Acceso - CLI y comando OpenStack.....	126
Ilustración 81: Acceso - Dashboard.....	127
Ilustración 82: Ayuda - Estado de la máquina .....	128
Ilustración 83: Ayuda - Problema PuTTY .....	129
Ilustración 84: Ayuda - Ficheros de configuración SSH .....	129
Ilustración 85: Ayuda - Ficheros de configuración ssh_config.....	129
Ilustración 86: Ayuda - Ficheros de configuración sshd_config .....	129
Ilustración 87: Ayuda - Opción instantánea .....	130
Ilustración 88: Ayuda - Tomar instantáneas .....	131
Ilustración 89: Ayuda - Nombre y descripción de instantánea.....	131
Ilustración 90: Ayuda - Opciones instantáneas .....	132
Ilustración 91: Dashboard/Servicio Horizon - Apartado "Sistema" .....	136
Ilustración 92: Dashboard/Servicio Horizon - Apartado "Identity".....	136
Ilustración 93: Cargar credenciales de administrador .....	137
Ilustración 94: Arquitectura de servicio de OpenStack .....	139
Ilustración 95: Conexión Servicio Nova - Servicio Neutron .....	145
Ilustración 96: Conexión Servicio Nova - Servicio Neutron (2).....	145
Ilustración 97: Servicio Keystone - Inicio sesión administrador .....	157
Ilustración 98: Servicio Keystone - Proyectos iniciales disponibles .....	158



Ilustración 99: Servicio Keystone - Usuarios iniciales disponibles .....	158
Ilustración 100: Servicio Keystone - Roles iniciales disponibles .....	159
Ilustración 101: Servicio Keystone - Creación de usuario.....	159
Ilustración 102: Servicio Keystone - Creación de proyecto .....	159
Ilustración 103: Servicio Keystone - Creación de rol .....	160
Ilustración 104: Servicio Keystone - Comprobación rol, usuario, proyecto.....	160
Ilustración 105: Servicio Keystone - Asignar rol usuario en proyecto.....	161
Ilustración 106: Servicio Keystone - Comprobación rol asignado.....	161
Ilustración 107: Servicio Keystone - Acceso Servicio Horizon .....	161
Ilustración 108: Servicio KeyStone - Menú usuario no administrador .....	162
Ilustración 109: Servicio Neutron - Comprobación inicial.....	163
Ilustración 110: Servicio Neutron - Creación de red interna.....	163
Ilustración 111: Servicio Neutron - Creación subred de red interna.....	163
Ilustración 112: Servicio Neutron - Creación de router .....	164
Ilustración 113: Servicio Neutron - Crear red .....	164
Ilustración 114: Servicio Neutron - Red externa .....	164
Ilustración 115: Servicio Neutron - Subred externa (1).....	165
Ilustración 116: Servicio Neutron - Subred externa (2).....	165
Ilustración 117: Servicio Neutron - Conexión router con red externa.....	166
Ilustración 118: Servicio Neutron - Conexión router con red interna .....	166
Ilustración 119: Servicio Neutron - Comprobación redes .....	166
Ilustración 120: Servicio Glance - Imágenes disponibles inicialmente .....	167
Ilustración 121: Servicio Glance - Estado activo.....	168
Ilustración 122: Servicio Cinder - Opción "Crear Volumen" .....	168
Ilustración 123: Servicio Cinder - Crear volumen.....	169
Ilustración 124: Servicio Cinder - Comprobación volumen creado .....	169
Ilustración 125: Claves - Importar claves .....	170
Ilustración 126: Claves – Creación .....	170
Ilustración 127: Claves – Comprobación.....	171
Ilustración 128: Claves – PuTTYgen.....	171
Ilustración 129: Servicio Nova - Opción "Lanzar instancia" .....	172
Ilustración 130: Servicio Nova - Detalles instancia.....	173
Ilustración 131: Servicio Nova - Origen instancia.....	173
Ilustración 132: Servicio Nova - Sabor instancia .....	174
Ilustración 133: Servicio Nova - Red interna de la instancia.....	174
Ilustración 134: Servicio Nova - Grupo de seguridad de la instancia .....	175
Ilustración 135: Servicio Nova - Claves instancias.....	175
Ilustración 136: Servicio Nova - Comprobación instancia .....	176
Ilustración 137: Comprobación SSH – Ping.....	176
Ilustración 138: Comprobación SSH - Dirección IP en PuTTY .....	177
Ilustración 139: Comprobación SSH - Importar clave privada PuTTY .....	177
Ilustración 140: Comprobación SSH – PuTTY .....	178
Ilustración 141: Inconvenientes SSH - Carpeta raíz.....	180
Ilustración 142: Inconvenientes SSH - Archivo "admin" .....	180
Ilustración 143: Inconvenientes SSH - Asignación a proyecto por defecto.....	180
Ilustración 144: Inconvenientes SSH - Modo adaptador puente.....	181
Ilustración 145: Inconvenientes SSH - Administrar red anfitrión (1) .....	182
Ilustración 146: Inconvenientes SSH - Administrar red anfitrión (2) .....	182
Ilustración 147: Inconvenientes SSH - Creación red anfitrión .....	183
Ilustración 148: Inconvenientes SSH - Modo red solo anfitrión .....	183
Ilustración 149: QEMU - Lanzar imagen.....	186
Ilustración 150: QEMU - Lanzar VNC Server Script .....	186

Ilustración 151: QEMU - Comrpobación conexión VNC .....	187
Ilustración 152: QEMU - Guardar cambios Tiny Core.....	187
Ilustración 153: Configuración VNC - Activación proxy .....	188
Ilustración 154: Configuración VNC - Ubicación del proxy .....	188
Ilustración 155: Configuración VNC - Dirección IP instancia .....	188
Ilustración 156: Configuración VNC - Dirección IP cliente.....	188
Ilustración 157: Configuración VNC - Ubicación consola nova xvp .....	189
Ilustración 158: Configuración VNC - Puerto solicitudes entrantes .....	189
Ilustración 159: Configuración VNC - Host peticiones entrantes .....	189
Ilustración 160: Tiny Core - Subir imagen .....	189
Ilustración 161: Tiny Core - Grupo de seguridad VNC.....	190
Ilustración 162: Tiny Core - Grupos de seguridad disponible.....	190
Ilustración 163: Tiny Core - Instanciar imagen.....	191
Ilustración 164: Tiny Core - Topología de red.....	191
Ilustración 165: Tiny Core - Comprobación sistema local .....	191
Ilustración 166: Conexión VNC - PuTTY Tiny Core Linux.....	192
Ilustración 167: Conexión VNC - Lanzar servidor VNC .....	192
Ilustración 168: Conexión VNC - Nueva conexión VNC Client .....	193
Ilustración 169: Conexión VNC - Acceso VNC Client .....	193
Ilustración 170: Conexión VNC - Conexión alcanzada .....	194
Ilustración 171: VirtualBox Kali - VNC Viewer nueva conexión .....	195
Ilustración 172: VirtualBox Kali - Cifrado VNC Server .....	196
Ilustración 173: VirtualBox - Autenticación VNC Viewer.....	196
Ilustración 174: VirtualBox - Kali Linux VNC Viewer .....	197
Ilustración 175: VBoxManage - Instrucción "clonehd" .....	197
Ilustración 176: VBoxManage - Archivo generado .....	198
Ilustración 177: Error VirtualBox – Advertencia.....	199
Ilustración 178: Error VirtualBox - Extensión no disponible.....	199
Ilustración 179: Error VirtualBox - Extensión disponible.....	200
Ilustración 180: Escenario OpenStack – LDAP.....	202

## TABLAS:

Tabla 1: Dedicación .....	11
Tabla 2: Desplegar escritorios virtuales .....	22
Tabla 3: Proveedores de virtualización de escritorio.....	28
Tabla 4: Alternativas archivo "nova.conf" .....	54
Tabla 5: Mes de enero .....	83
Tabla 6: Mes de febrero .....	84
Tabla 7: Mes de marzo.....	85
Tabla 8: Mes de abril.....	86
Tabla 9: Mes de mayo .....	87
Tabla 10: Mes de junio.....	88
Tabla 11: Mes de julio.....	89
Tabla 12: Resultado proveedores.....	101
Tabla 13: OpenStackCLI - Componentes principales .....	137
Tabla 14: KeyStone - Lista de comandos .....	147
Tabla 15: Glance - Lista de comandos.....	147
Tabla 16: Nova - Claves SSH .....	148
Tabla 17: Nova - Grupo Seguridad .....	149
Tabla 18: Nova – Información .....	149
Tabla 19: Nova - Creación instancia .....	149
Tabla 20: Nova - IP flotante.....	149
Tabla 21: Nova - Acceder instancia .....	149
Tabla 22: Nova - Operaciones en las instancias.....	150
Tabla 23: Cinder - Asociar volumen a instancia.....	151
Tabla 24: Cinder - Creación de instancia con el volumen.....	151
Tabla 25: Cinder - Instantánea de un volumen .....	152
Tabla 26: Cinder - Redimensionar volumen .....	152
Tabla 27: Cinder - Transferencia de volumen .....	153
Tabla 28: Cinder - Crear copia de seguridad .....	153
Tabla 29: Cinder - Restauración copia de seguridad .....	154
Tabla 30: Swift - Lista de comandos .....	155
Tabla 31: Neutron - Creación red externa.....	155
Tabla 32: Neutron -Red interna asociada a una instancia .....	156
Tabla 33: Tiny Core – Requisitos .....	185
Tabla 34: Instalación LDAP - Fichero del usuario root .....	205
Tabla 35: Instalación LDAP - Fichero de dominio .....	205
Tabla 36: Instalación LDAP - Fichero del árbol de dominio .....	206
Tabla 37: Instalación LDAP - Instrucciones.....	206
Tabla 38: Instalación TLS - Creación certificado .....	207
Tabla 39: Instalación TLS - Fichero SSL.....	207
Tabla 40: Instalación TLS - Fichero slapd.....	208
Tabla 41: Instalación TLS - Configuración servidor.....	208
Tabla 42: Configuración OpenStack - Fichero "local_settings" .....	209
Tabla 43: Configuración OpenStack - Fichero "keystone_mydomai.conf".....	210
Tabla 44: Configuración OpenStack - Instrucciones .....	212

# AGRADECIMIENTOS

---

Quisiera agradecer a mi familia por toda la confianza que han depositado en mí. En especial a mi madre por ser quien ha aguantado todos y cada uno de mis momentos ya sean buenos o malos haciéndome ver que nada es imposible si se intenta. También a mi Niña por escucharme y alentarme, aún sin saber de lo que hablaba; siempre sacándome una sonrisa. Dar las gracias a toda mi familia por todos los valores que me han inculcado, los cuales son responsables del esfuerzo y dedicación constante que se requieren para poder superar retos como el que estoy concluyendo. A mi hermano por estar pendiente de mí.

Me gustaría reconocer también la ayuda ofrecida por mi compañero Mikel Mesanza. Sin su ayuda y comprensión no hubiera sido posible la realización de las diferentes pruebas así como de conocer el funcionamiento real de la plataforma.

El proyecto se ha elaborado bajo la dirección de Alexander Mendiburu Alberro y José Miguel Alonso, profesores de la Facultad de Informática de San Sebastián, a los que me gustaría reconocer por el trabajo y tiempo invertidos, y agradecerles la confianza puesta en mí. También a la Facultad por proporcionarme el material necesario para poder alcanzar el resultado deseado.

A todos y a todas, gracias.

*1*

---

---

INTRODUCCIÓN

## 1.1. Visión general

La observación que se llevó a cabo sobre el funcionamiento y la gestión de los laboratorios de la Facultad de Informática de San Sebastián (FISS) permitió detectar diferentes dificultades que detallo a continuación:

- ❖ Se realiza una gestión compleja de los diferentes sistemas operativos, software, etc; que obligaban a la creación de imágenes ad-hoc por cada laboratorio y asignatura. La creación de imágenes ad-hoc es un término que se utiliza para referirse a la manera de trabajo, en la que se busca únicamente lograr un desarrollo que dé respuesta al problema en el que se está trabajando, sin dotar al progreso de la necesaria modularidad que permita reutilizar sus componentes en el futuro.
- ❖ El/la estudiante no tiene posibilidad de disponer de una máquina que guarde el estado de su trabajo. Es decir, salvo que guarde su trabajo en un dispositivo externo (plataformas en la nube, USB,...); tendrá que reiniciar todo su trabajo desde el principio en cada inicio de sesión.

Esta forma de trabajar plantea una serie de incoherencias:

- ❖ No se trata de una solución escalable<sup>13</sup> ya que no podemos reutilizar sus componentes en el futuro.
- ❖ La solución puede no ser óptima<sup>22</sup> con respecto al tiempo de ejecución, uso de memoria, espacio en disco, ancho de banda, consumo de energía, etc.
- ❖ El rendimiento<sup>26</sup> de un sistema operativo<sup>27</sup>, programa o dispositivo puede verse mermado por una falta de optimización<sup>22</sup>.
- ❖ Esta manera de trabajo puede no ser eficiente<sup>12</sup>.

Por otro lado, no estamos facilitando la tarea al alumno/a. La Facultad de Informática de San Sebastián debería disponer de los mecanismos necesarios para que cada estudiante tuviera su espacio de trabajo donde poder guardar constantemente toda la labor que ha realizado, pudiendo acceder a la misma en cualquier momento desde cualquier lugar. De la misma forma sucede con el profesorado o cualquier otra persona autorizada de la Facultad de Informática de San Sebastián/Donostia.

Es por estas dificultades que surge este proyecto en donde nuestra misión será solventar todos y cada uno de estos inconvenientes buscando la solución que mejor se adecue a las circunstancias.

## 1.2. Objetivos

El objetivo de este proyecto es poder ofrecer a los alumnos, profesorado y cualquier otra persona autorizada de la Facultad de Informática de San Sebastián (FISS) una mejor experiencia para el desarrollo de su trabajo. Por ejemplo, en el caso de un estudiante, para la realización de prácticas. Esto va a consistir en facilitarles un escritorio unificado independientemente del ordenador físico en el que realicen su trabajo, por lo que se precisará realizar una revisión de las diferentes soluciones de escritorio virtual.

Para alcanzar este propósito se deberán analizar las necesidades prácticas que requieran y estudiar las diferentes soluciones de virtualización de escritorio escogiendo una solución e implementar un prototipo sobre el mismo. A continuación, se deberá evaluar el resultado alcanzado mediante diferentes pruebas de rendimiento<sup>26</sup>. En caso de cumplir con los requisitos se aceptará como posible solución y en caso contrario, se rechazará y se buscará otra posible salida.

Los requisitos serán los siguientes:

- ❖ Buscar preferentemente soluciones con un coste razonable (incluso nulo) en licencias.
- ❖ La resolución debe ser escalable<sup>13</sup> a un gran número de alumnos.
- ❖ Debe ofrecer un buen nivel de servicio, teniendo en cuenta la infraestructura de comunicaciones de la Facultad de Informática de San Sebastián (FISS) y el acceso remoto. Alternativamente, si existe cualquier complicación con la infraestructura de comunicación, habrá que tomar medidas que permitan su resolución.

Además, antes de planificar una virtualización de escritorios e implantar una infraestructura de escritorios virtuales (VDI), debemos tener en consideración los siguientes aspectos que forman parte de la planificación para implementar la solución VDI:

- ❖ Red LAN, WAN, ancho de banda, modo de conexión de los usuarios.
- ❖ Capacidad de almacenamiento, rendimiento<sup>26</sup> que deberá tener, espacio para las máquinas virtuales.
- ❖ Hardware: Número de servidores, alta disponibilidad, clientes ligeros.
- ❖ Tipos de aplicaciones de usuarios existentes para el futuro VDI.
- ❖ Formación del administrador de la empresa que hará uso de la solución VDI y de todas sus herramientas de gestión.

Habrá que tener en consideración que pueden existir diferentes soluciones que cumplan con nuestras necesidades. Es por esa razón, que se deberán tener presentes y bien documentados todos los resultados que se alcancen de aquellas soluciones de virtualización.





# 2

---

---

## PLANIFICACIÓN

## 2.1. Estructura de descomposición del trabajo

Para facilitar el desarrollo, la redacción y la futura lectura-comprensión del proyecto se decidió dividir la estructura de desglose de trabajo en cuatro bloques los cuales se han considerado esenciales:

- ❖ **Gestión:** Se refiere a la planificación de todo el trabajo que se quiere llevar a cabo y su respectivo seguimiento y control. También hacemos mención de las reuniones que se realicen con mis directores de proyecto, Alexander Mendiburu Alberro y José Miguel Alonso, así como, el seguimiento y control que se lleve a cabo del proyecto.
- ❖ **Aprendizaje:** Por medio de este bloque se pretende representar la formación y el aprendizaje necesarios para llevar a cabo el proyecto.
- ❖ **Desarrollo:** Puesta en marcha de los objetivos del trabajo en base al estudio realizado anteriormente.
- ❖ **Documentación:** Este último bloque está constituido de la memoria, así como del material necesario para la defensa del proyecto. De esta forma, reflejaremos todo lo relacionado (objetivos alcanzados, problemas encontrados, etc) de nuestro trabajo.

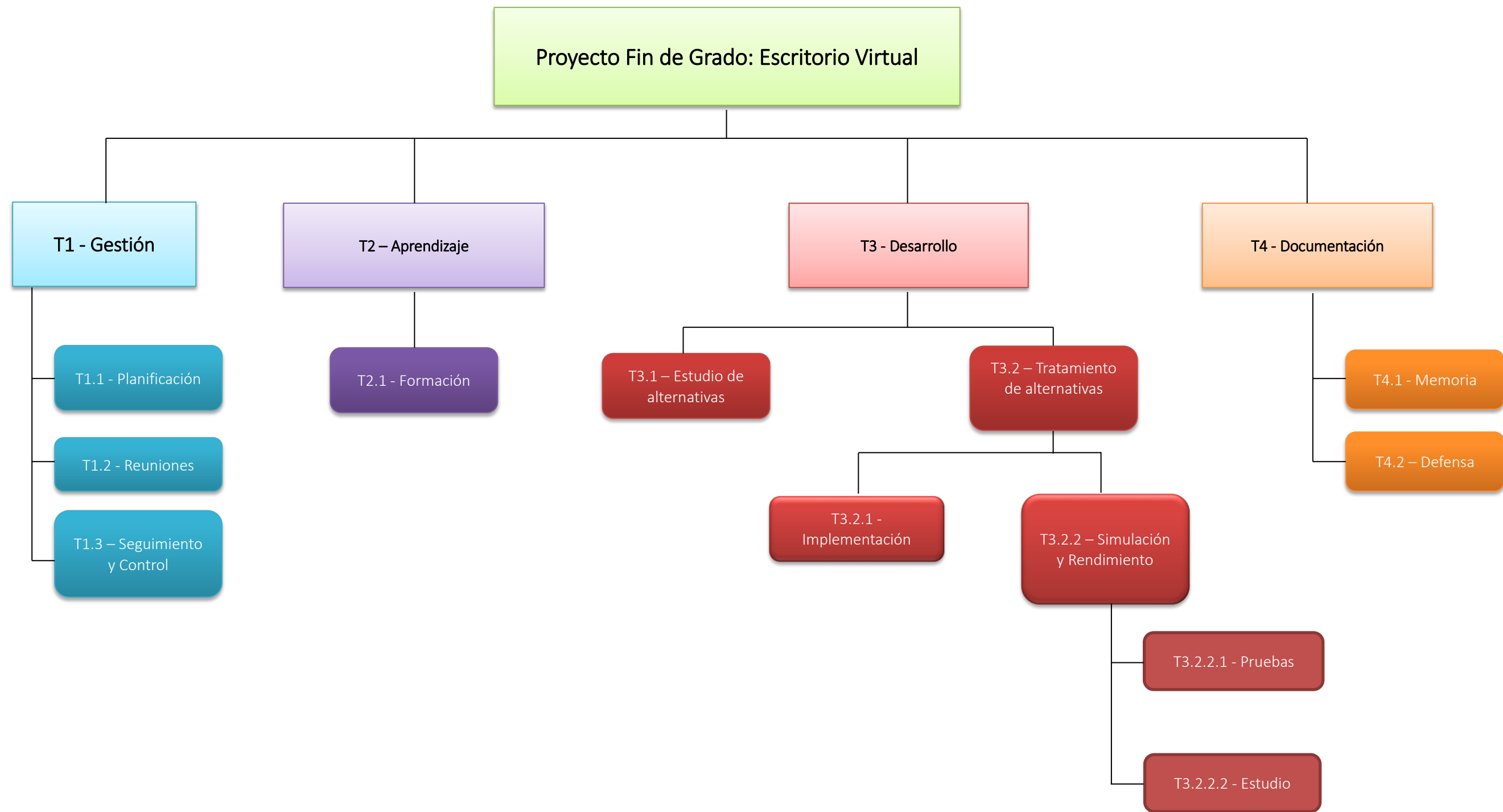


Ilustración 1: Desglose del trabajo

## 2.2. Tareas del proyecto

Por medio del siguiente apartado daremos a conocer las distintas tareas que encontramos a partir de la estructura de desglose de trabajo comentada anteriormente.

- ❖ T1 – Gestión
  - T1.1 – Planificación.
    - T1.1.1 – Definición de la estructura de desglose del trabajo.
    - T1.1.2 – Definición de las tareas,
    - T1.1.3 – Identificación y definición de los entregables del proyecto.
    - T1.1.4 – Planificar el tiempo.
      - Diagrama de tiempo.
      - Estimar la dedicación.
    - T1.1.5 – Determinar la metodología de trabajo.
    - T1.1.6 – Definir un plan de comunicación.
    - T1.1.7 – Estimar un plan de riesgo para todo el proyecto.
  - T1.2 – Reunión. Reuniones periódicas que lleve a cabo con mis directores de proyecto.
  - T1.3 – Seguimiento y Control.
    - T1.3.1 – Representar las horas reales invertidas.
    - T1.3.2 – Identificar las diferencias entre las horas reales invertidas y las horas estimadas.
    - T1.3.3 – Sacar conclusiones de la variación de horas.
    - T1.3.4 – Verificar el correcto desarrollo del proyecto.
- ❖ T2 – Aprendizaje
  - T2.1 – Formación. Aprendizaje de los diferentes temas que se tratarán en el proyecto.
- ❖ T3 – Desarrollo
  - T3.1 – Estudio de alternativas. Estudio de las diferentes tecnologías.
  - T3.2 – Tratamiento de alternativas.
    - T3.2.1 – Implementación. Instauración de cada mecanismo dado por viable en el estudio anterior.
    - T3.2.2 – Simulación y Rendimiento. Estudio, comparación y valoración de los resultados obtenidos de cada una de las herramientas utilizadas.
      - T3.2.2.1 – Pruebas. Definición del tipo de pruebas.
      - T3.2.2.2 – Estudio. Análisis de los resultados alcanzados de las pruebas anteriores.
- ❖ T4 – Documentación
  - T4.1 – Memoria.
  - T4.2 – Defensa. Hace referencia al material que es necesario elaborar para la defensa del proyecto.

## 2.3. Características e identificación de entregables

Realicemos a continuación la identificación de los entregables que tendrá el proyecto, así como una breve descripción de sus características:

- ❖ La memoria: Documento en el que se reflejará todo el estudio realizado, así como los resultados alcanzados.

La memoria estará constituida por un conjunto de anexos en donde cabe destacar los siguientes:

- Anexo donde reflejo la planificación inicial del tiempo.
  - Anexo con las diferentes soluciones de virtualización de escritorio estudiadas.
  - Anexo de la instalación de OpenStack PackStack sobre el sistema operativo<sup>27</sup> CentOS.
  - Anexo para la administración básica de la plataforma OpenStack.
  - Anexo de la prueba de administración realizada haciendo uso del protocolo SSH.
  - Anexo de la prueba de administración en donde se ha utilizado la aplicación VNC.
  - Anexo donde se explica cómo integrar la aplicación LDAP en la plataforma OpenStack.
- ❖ El material para la defensa: Este material deberá representar la forma en la que se ha realizado el proyecto, así como los resultados alcanzados de una manera clara y precisa. La defensa se verá apoyada en un conjunto de diapositivas.

## 2.4. Planificación

### 2.4.1. Diagrama de tiempo

La planificación del tiempo se verá apoyada en un conjunto de diagramas que se encontrarán anexos a esta documentación, más en concreto en el anexo B. Por medio de estos diagramas se pretende conocer cómo voy a gestionar mi proyecto en base al conjunto de horas y meses que tengo disponibles. Por ese motivo, se ha realizado un estudio exhaustivo de cada una de las tareas que tenemos que realizar con el fin de organizar cuando llevar a cabo cada una, teniendo así en consideración todos los objetivos propuestos.

La única etapa transversal a todo el proyecto es la gestión del proyecto, por razones que considero obvias. Las reuniones con los tutores de la universidad están previstas con una frecuencia semanal (a excepción de épocas de exámenes y festividades), y el seguimiento y control se ejecutará todas las semanas que haya previstas tareas. La primera semana se espera que se dediquen a la planificación, es decir, se decidirá la estrategia para concluir satisfactoriamente el proyecto.

El objetivo de la etapa de “Aprendizaje” será ampliar nuestros conocimientos sobre los diferentes temas de los que se fundamenta el proyecto. Para ello se ha estimado dedicar otra semana completa.

La tercera de las etapas hace referencia al estudio de las diferentes tecnologías que permiten alcanzar el objetivo de este proyecto, así como la puesta en marcha de aquella o aquellas soluciones que mejor cumplen con todas las exigencias junto con diferentes pruebas sobre las mismas. Esto conllevará a alcanzar unos resultados en donde veremos cuál es la opción más viable. Esta será la etapa donde dedicaré mayor tiempo y recursos, debiendo de ser capaz de lograr el propósito de este trabajo.

Finalmente, la cuarta y última etapa se refiere al desarrollo de la documentación. La mayor parte de ese periodo se dedicará a llevar a cabo la memoria aunque también se espera preparar el material para la defensa. Aunque estas tareas se realizarán en la última parte, durante todo el transcurso del proyecto se irá apuntando las decisiones tomadas, los problemas encontrados, entre otros aspectos.

## 2.4.2. Dedicación

Una vez conocida la planificación del tiempo, habrá que estimar una dedicación a cada una de las tareas realizando para ello la siguiente tabla donde reflejamos el tiempo estimado de dedicación, el tiempo real invertido y su influencia en el proyecto. Este tiempo reflejará el conjunto de horas que se han estimado o se pretenden dedicar en la elaboración del Trabajo Fin de Grado.

TAREA	DEDICACIÓN ESTIMADA	DEDICACIÓN REAL	DEDICACIÓN ESTIMADA (%)	DEDICACIÓN REAL (%)
<b>T1 - Gestión</b>	40	29	13%	7%
T1.1 - Planificación	15	13	5%	3%
T1.2 - Reunión	15	12	5%	3%
T1.3 -Seguimiento y Control	10	5	3%	1%
<b>T2 - Comienzo</b>	20	58	7%	14%
T2.1 – Formación	20	58	7%	14%
<b>T3 - Desarrollo</b>	160	183	53%	44%
T3.1 - Estudio de Alternativas	60	46	20%	11%
T3.2 - Tratamiento de alternativas	100	137	33%	33%
T3.2.1 - Implementación	50	37	17%	9%
T3.2.2 -Simulación y Rendimiento	50	100	17%	24%
T3.2.2.1 - Pruebas	35	56	12%	13%
T3.2.2.2 - Estudio	15	44	5%	11%
<b>T4 - Documentación</b>	80	137	27%	35%
T4.1 - Memoria	70	134	23%	32%
T4.2 - Material para la defensa	10	13	3%	3%
<b>TOTAL</b>	300	418	100%	100%

Tabla 1: Dedicación

### 2.4.3. Desviaciones

Durante el desarrollo del proyecto nos hemos encontrados con diferentes dificultades que han originado que se hayan aumentado el tiempo de dedicación a alguna tarea. De la misma manera que en algunos casos se han ampliado los plazos de realización, en otros se han disminuido. Esto ha originado a su vez que este Trabajo Fin de Grado se ha llevado a cabo en un plazo superior al planificado. Expliquemos y justifiquemos las razones de estas desviaciones:

- ❖ Inicialmente se había estipulado un tiempo de dedicación al estudio y comprensión de los diferentes conceptos que se consideran importantes dentro de este proyecto. Sin embargo, con el análisis realizado de la plataforma OpenStack así como la administración en la que se basa, ha originado que esta duración se haya aumentado considerablemente.
- ❖ La tarea de implementación ha tenido un periodo de tiempo menor, al haberse puesto exclusivamente en marcha la plataforma OpenStack.
- ❖ Debido a los problemas encontrados en la plataforma OpenStack se ha dedicado un mayor tiempo que el planificado a la tarea de “simulación y rendimiento”. Se ha requerido un análisis y puesta en funcionamiento de diferentes alternativas, con el objetivo de solventar los problemas. Todo ello ha servido además para conocer más de cerca la plataforma y todo lo que es capaz de ofrecernos.
- ❖ El tiempo para las reuniones ha sido menor que el esperado. El tiempo de dedicación por cada reunión ha sido aproximadamente de 30' por lo general y se han llevado a cabo los miércoles en vez de los viernes. Se han solventado muchas dudas por medio electrónico aunque presencialmente ha servido para conocer la situación donde nos encontrábamos y hacía donde queríamos dirigirnos.
- ❖ También se ha requerido realizar una documentación más robusta con el fin de reflejar todo el estudio, inconvenientes, metas alcanzadas, etc; del trabajo.

Finalmente aclarar que el espacio de tiempo utilizado para la realización de este proyecto ha sido ligeramente mayor que el planificado, pero gracias a ello se ha podido llevar a cabo el mismo.



## 2.5. Metodología de trabajo

Inicialmente se empezará elaborando la planificación. Dado que se desconoce el tiempo exacto de duración, se hará una estimación temporal determinada.

Seguidamente se continuará con la fase de “Comienzo” en donde se realizará un aprendizaje de los diferentes temas de los que se constituye el proyecto.

Después de la fase de “Comienzo” se empezará con la parte del desarrollo. Se estudiará en primer lugar las diferentes herramientas que permiten alcanzar nuestro objetivo. A continuación, se llevará a cabo por cada mecanismo que cumpla con los requisitos establecidos su implementación, es decir, su puesta en marcha. Habiendo puesto ya en funcionamiento la herramienta se le realizarán un conjunto de pruebas debiendo analizar los resultados alcanzados. Esta fase deberá finalizar con un conjunto de conclusiones sobre que material o materiales se adecuan mejor a las exigencias del proyecto. Una vez hemos dado por finalizado esta fase de desarrollo, se dará fin a la redacción de la memoria debiendo posteriormente elaborar el material para la defensa del Trabajo Fin de Grado.

Cada semana se realizará una reunión con los directores de proyecto de la universidad siempre y cuando se pueda, así como deberá verse reflejado en la memoria todo el progreso alcanzado.

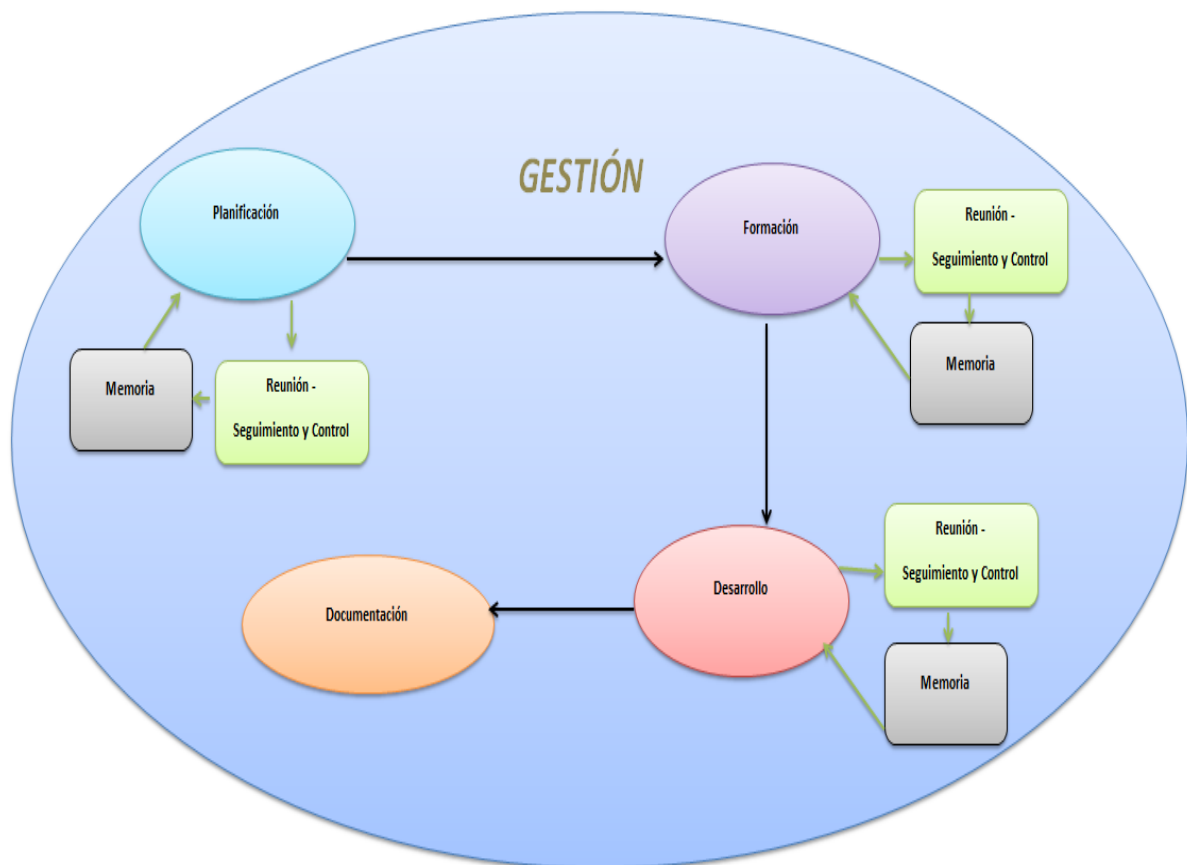


Ilustración 2: Metodología de trabajo

## 2.6. Plan de comunicación

El plan de comunicación con mis directores de proyecto se fundamentará en dos medios diferentes:

- ❖ Mediante reuniones periódicas presenciales.
- ❖ Haciendo uso del correo electrónico facilitado por la Universidad del País Vasco UPV/EHU.

Se ha decidido realizar cada semana una reunión para estudiar el progreso de mi proyecto, así como dirigir la memoria. En caso de existir cualquier complicación (dudas, sugerencias,...) se podrá concretar una reunión. De igual manera, si se requiere mayor tiempo para alcanzar un objetivo propuesto en un instante dado, puede no realizarse la reunión posponiéndose a una fecha más adecuada.

El trabajo de planificación, aprendizaje, análisis y documentación se realizarán desde la Universidad de Vitoria o desde mi alojamiento. Por otra parte, las reuniones que realice con mis directores de proyecto se realizarán en la Facultad de Informática UPV/EHU de San Sebastián/Donostia.

El correo proporcionado por la Universidad se utilizarán cuando se requiera concretar una reunión o incluso para solventar dudas entre otras posibles situaciones.

## 2.7. Gestión de riesgos

Es conveniente definir un plan de riesgo donde identifiquemos cada uno de ellos, con la intención de tomar las medidas necesarias para que no se den lugar, evitando de esta forma posibles cambios significativos que pongan en peligro el desarrollo del proyecto. Por cada uno de ellos conoceremos la probabilidad de que suceda así como la influencia que tiene dentro del trabajo.

- ❖ Pérdida de la documentación realizada.

La pérdida de la documentación, ya sea la memoria o el material para la defensa, podría ocasionar graves alteraciones en el desarrollo del proyecto. Esto ha originado que se tomen diferentes medidas como disponer de distintas copias de seguridad ubicadas en diferentes dispositivos (USB, Drive, etc).

La probabilidad de que suceda es pequeña, aunque su influencia en el proyecto es muy grande.

- ❖ Incompatibilidad a la hora de hacer reuniones.

Durante el desarrollo del proyecto, pueden surgir incompatibilidades de horarios con alguno de los directos de proyecto. Por ese motivo, se ha decidido acordar las reuniones previamente con suficiente antelación. Además, en caso de no estar presente alguno de los directores, estaría en manos del otro o de la mía, el notificar y por tanto mantener informado al primero.

La probabilidad de que sucede es media al igual que la influencia que tiene en el proyecto.

❖ No disposición del material.

Podemos encontrarnos en una situación en donde para realizar una prueba necesitemos de cierto material que no tengamos disponible. Ello puede provocar desviaciones en el tiempo al tener que solicitarlo o buscar otro medio para su desarrollo.

La probabilidad de que suceda es media y su influencia en el proyecto es alta.

❖ Realización de Pruebas

La realización de las primeras pruebas sobre la plataforma OpenStack así como su propia instalación, estará condicionada a un equipo disponible bajo préstamo, que es el que me proporciona los recursos para poder llevarlo a cabo. El encontrarse más o menos limitado de tiempo, ocupado, puede afectar considerablemente al desarrollo y progreso del proyecto.

La probabilidad de que sucede es alta y su influencia en el proyecto es muy alta.

Sin embargo las últimas pruebas se basarán en un equipo proporcionado por la Facultad en donde la orientación y apoyo de mis directores de proyecto será fundamental. Serán en estos casos donde deberemos tener presente el riesgo de incompatibilidad para hacer reuniones comentado anteriormente.

La probabilidad de que suceda es media y su influencia en el proyecto es muy alta.



# 3

---

---

## CONCEPTOS

## 3.1. Introducción

A partir de este apartado asentaremos las bases sobre las que se cimenta nuestro proyecto. Se realizará una explicación breve de cada aspecto con el objetivo de poder ser entendido indistintamente de los conocimientos previos que se dispongan.

## 3.2. Virtualización

Demos una primera definición de lo que se entiende en informática por virtualización:

- ✓ Virtualización es la creación a través de software de una versión virtual de algún recurso tecnológico, como puede ser una plataforma de hardware, un sistema operativo<sup>27</sup>, un dispositivo de almacenamiento, una red u otros recursos.

El nuevo software, que va desde los sistemas operativos hasta las aplicaciones, demanda constantemente cada vez más datos, más potencial de procesamiento y más memoria. La virtualización hace que una sola máquina física actúe como muchas máquinas y disminuye el costo de servidores y estaciones de trabajos adicionales. Entonces, ¿qué es la virtualización?

La virtualización es una tecnología que permite crear múltiples entornos simulados o recursos dedicados desde un solo sistema de hardware físico. Por ejemplo, en una virtualización de ordenadores el software llamado “hipervisor” se conecta directamente con el hardware y permite dividir un sistema en entornos separados, diferentes y seguros, los cuales se denominan “máquinas virtuales” (VM). Estas “máquinas virtuales” dependen de la capacidad del “hipervisor” para separar los recursos de la máquina del hardware y distribuirlos adecuadamente.

La máquina física original equipada con el hipervisor se denomina “host”, y las máquinas virtuales que utilizan estos recursos se llaman “guests”. Estos “guests” tratan a los recursos informáticos, como la CPU, la memoria y el almacenamiento, como un almacén de recursos que pueden hacer uso con facilidad. Los operadores pueden controlar las instancias virtuales de la CPU, la memoria, el almacenamiento y otros recursos para que los “guests” reciban los recursos que necesitan cuando así lo estimen oportuno.

En resumen, la virtualización crea un entorno informático simulado, o virtual, en lugar de un entorno físico. Esto permite a las organizaciones particionar un equipo o servidor físico en varias máquinas virtuales. Cada máquina virtual puede interactuar de forma independiente y ejecutar sistemas operativos o aplicaciones diferentes mientras comparten los recursos de una sola máquina host. Al crear varios recursos a partir de un único equipo o servidor, la virtualización mejora la escalabilidad y las formas de procesar las cargas de trabajo, al tiempo que permite usar menos servidores y reducir el consumo de energía, los costos de infraestructura y el mantenimiento.

La virtualización se divide en distintas categorías distinguiendo para este proyecto las siguientes:

- ❖ Virtualización de escritorio: Permite que un servidor centralizado ofrezca y administre escritorios individualizados.
- ❖ Virtualización de red: Diseñada para dividir el ancho de banda de una red en canales independientes que se asignan a servidores o dispositivos específicos.
- ❖ Virtualización de software: Separa las aplicaciones del hardware y el sistema operativo<sup>27</sup>.
- ❖ Virtualización de almacenamiento: Combina varios recursos de almacenamiento en red en un solo dispositivo de almacenamiento accesible por varios usuarios.
- ❖ Virtualización de equipos completos.

### 3.3. Escritorio virtual

La virtualización de los puestos de trabajo es una opción cada vez más utilizada por grandes compañías y entidades, obligadas a trabajar con una gran cantidad de empleados en entornos cuya gestión puede convertirse en un gran problema si no se realiza de la manera adecuada. El uso de escritorios virtuales ofrece importantes ventajas y beneficios, no quedando restringido su uso al ámbito empresarial. De este modo, estas tecnologías son aplicadas con éxito por ejemplo en otros entornos como el educativo. Pero, ¿qué se conoce por escritorio virtual?

La virtualización de escritorio es un término relativamente nuevo que describe el proceso de separación entre el escritorio, que engloba los datos y programas que utilizan los usuarios para trabajar, de la máquina física. El escritorio “virtualizado” es almacenado remotamente en un servidor central en lugar de en el disco duro del ordenador personal. Esto significa que cuando los usuarios trabajan en su escritorio desde su portátil u ordenador personal, todos sus programas, aplicaciones, procesos y datos se almacenan y se ejecutan en recursos remotos, permitiendo a los usuarios acceder a sus escritorios desde cualquier dispositivo capaz de conectarse vía Internet o red corporativa al escritorio, tales como un portátil, computadora (PC), Tablet, teléfono inteligente o cliente ligero<sup>28</sup>.

Entonces la idea de escritorio virtual se centra en que el usuario inicie sesión con sus credenciales, y acceda a un escritorio con las aplicaciones y programas instalados como si estuviera sentado frente a ese computador.

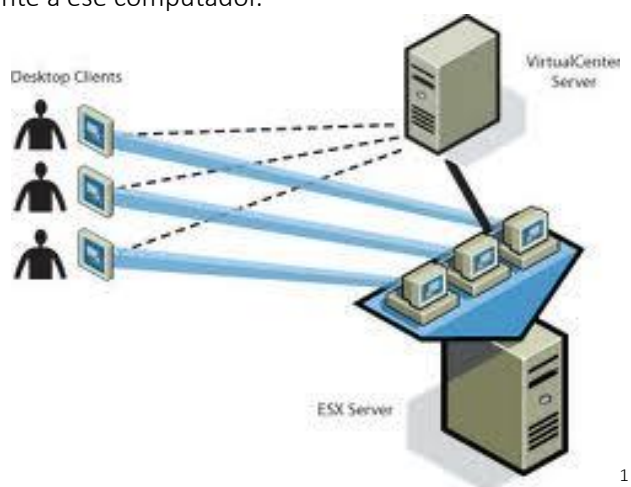


Ilustración 3: Escritorio Virtual por cada usuario

<sup>1</sup> <http://www.brainlabs.com.ar/novedad/images//images5.jpg>

Al igual que cualquier otra tecnología, la virtualización de escritorios proporciona una serie de beneficios originando que esta tecnología sea elegida por un gran número de ellos:

- ❖ Reducción del coste general de los equipos

Utilizar escritorios virtuales no requiere de equipos de última generación o gran capacidad de memoria, ya que las aplicaciones son ejecutadas realmente en los servidores de la plataforma, donde también se guardan los datos y archivos. Esto supone una reducción de costes y a su vez una prolongación de la vida de los equipos informáticos, además de posibilitar la utilización de otros dispositivos de menor coste como tabletas o “thin client”<sup>28</sup>. Asimismo, el uso de esta tecnología puede suponer un destacado ahorro en el consumo de energía.

Esta reducción de coste se da lugar por tanto en el lado del cliente, ya que el servidor por su parte tendrá que invertir una gran cantidad de recursos para poder proporcionar estos servicios.

- ❖ Disminución de soporte y mantenimiento técnico

Los costes de soporte y mantenimiento de los puestos de trabajo pueden verse reducidos drásticamente, gracias a la posibilidad de administración remota de los puestos de usuario.

Además, el tiempo de respuesta ante una avería en un equipo cliente puede quedar minimizado, ya que basta con llevar a cabo la sustitución del equipo para que este vuelva a tener disponible su entorno de trabajo personal de forma inmediata.

- ❖ Asegura la continuidad del negocio

Un estudio reciente indica que la gran mayoría de las organizaciones no tiene planes de continuidad del negocio actualmente. El estudio se basa en una encuesta realizada a 100 administradores de IT de Pymes del Reino Unido (empresas con 30-500 empleados) llevada a cabo por ICM Investigación. La encuesta sugiere que ante la posibilidad de un fallo en el transporte público o desastre en el lugar de trabajo, por ejemplo, muchas empresas no estarían en condiciones de reabrir el negocio. De ahí la necesidad de uso de una virtualización de escritorio.

- ❖ Mejora la seguridad de los datos

La virtualización de escritorio hace que todos los datos de los usuarios de los escritorios, y por tanto de las organizaciones, se almacenen centralmente en los servidores. Absolutamente nada se almacena a nivel local. Esto provoca que en el caso de que un empleado pierda su dispositivo, los datos no se extravíen ni se corrompan y que tampoco pasen a manos peligrosas.



Conocidas las ventajas que proporciona una virtualización de escritorio, cabría destacar los inconvenientes. Estos problemas están especialmente orientados al servidor que proporciona esa infraestructura de escritorio virtual:

- ❖ Por cada usuario seguramente se requiera una imagen única que tenga que guardar las configuraciones personales o simplemente que dispongan de la libertad de instalar sus propias instalaciones. Esto rápidamente multiplica el consumo de almacenamiento en el servidor que proporciona la infraestructura.
- ❖ La solución VDI requerirá una importante inversión en servidores y posiblemente en almacenamiento e infraestructura de red. El coste total de todo este hardware podrá superar al de un PC tradicional.
- ❖ En la situación de que el servidor deje de estar operativo, todos los usuarios que confían en esa máquina van a ser incapaces de trabajar. Es decir, cualquier problema que afecte al servidor afectará a múltiples usuarios. Por esa razón es una buena idea configurar servidores redundantes como mecanismo de seguridad.

Finalmente destacar, que también el rendimiento<sup>26</sup> de un sistema operativo<sup>27</sup> virtualizado será inferior, no podrá alcanzar las mismas cotas que si estuviera directamente instalado.

### 3.3.1. Desplegar escritorios virtuales

El impacto positivo de la virtualización de escritorios como palanca de la transformación digital de las empresas está claro. Existe una amplia oferta de software de virtualización para desplegar una infraestructura de escritorios virtuales VDI en las propias instalaciones de la empresa denominadas estas como VDI on-premise. Pero el paradigma “Cloud Computing” también permite disfrutar de esta tecnología a través de soluciones Cloud.

- ❖ En las soluciones VDI on-premise, todas las piezas que configuran una infraestructura de virtualización del escritorio forma parte del centro de datos corporativo. Entre estas destacan el almacenamiento y la red; de su calidad y eficiencia<sup>12</sup> va a depender en un tanto por ciento muy alto el éxito de la implantación. Son dos recursos vitales y si no se invierte adecuadamente en ellos, se sufrirán diferentes problemas que pueden originar el fracaso del proyecto.
- ❖ Por su parte, en las soluciones Cloud, la solución VDI es provista por un proveedor de servicios Cloud. De esta manera, las empresas trasladan a un modelo IaaS (Infrastructure as a Service)<sup>18</sup> de pago por suscripción su necesidad de virtualización de escritorios. El proveedor Cloud es responsable de los recursos TI<sup>29</sup>, de su actualización y de su seguridad. Existe además la posibilidad de que el propio proveedor se ocupe también de la gestión de los escritorios o que sea competencia del cliente, delegándola, si así lo considera.

Elegir entre VDI on-premise o VDI en la Nube, va a depender de las circunstancias de cada organización. Es verdad que el mundo Cloud permite desentenderse de la adquisición y del mantenimiento informático, pero todavía genera reticencias en cuanto a seguridad y pérdida de control. Además, no siempre las licencias de los proveedores son atractivas para las empresas.

De nuevo, es importante recordar que las soluciones on-premise y las Cloud no son incompatibles, y que lo más práctico es oscilar entre unas y otras mediante la implantación de infraestructuras de Nube híbrida. Una estrategia de escritorios virtuales híbridos permite complementar los desplegados en el CPD corporativo con otros bajo modelo DaaS (Desktop as a

Service) para responder a necesidades concretas de forma ágil. De esta manera, pueden consumirse como servicio para perfiles que no sean críticos, picos puntuales de demanda, etc.

Llegados a este punto es fundamental distinguir si se van a instalar servidores específicos en centro de datos de la propia empresa atendidos por personal técnico propio, o bien si se va a tratar de un servicio remoto, ofrecido por un proveedor en la nube:

Servidor	On premise	Los servidores se instalan en los centros de datos de la organización.
	Cloud	Los servidores son del proveedor.
	Híbrido	Hay servidores locales y remotos.

Tabla 2: Desplegar escritorios virtuales

### 3.3.2. Recomendaciones

Antes de analizar las opciones del mercado conviene reflexionar sobre el enfoque que la organización desea darle a las soluciones VDI:

- ❖ Enfoque de servicio gestionado: Que todo lo realice el proveedor a cambio de una cuota mensual.
- ❖ Enfoque de asimilación de tecnología: El proveedor aporta las piezas de tecnología para que el área TI<sup>29</sup> de la empresa asimile los conocimientos y gestione el servicio.

Entre estos dos enfoques hay numerosas posiciones intermedias. En todo caso, en este proyecto se seguirá el segundo planteamiento, haciendo uso del siguiente itinerario:

1. Documentarse acerca de la tecnología.
2. Elegir los casos de uso para una primera implantación estudiando los resultados que se alcancen.

A continuación se dan a conocer las recomendaciones que se consideran oportunas para abordar cualquier proyecto VDI (Virtual Desktop Infrastructure):

- ❖ Costes del proyecto: Tener en consideración tanto los costes directos como los indirectos a fin de obtener correctamente el coste total del proyecto.
- ❖ Simplicidad: Estudiar el grado de sencillez/complejidad de la solución VDI que se está considerando.
- ❖ Escalabilidad: Planificar de entrada cómo se puede ampliar la solución al cabo del tiempo buscando una solución que pueda ir creciendo progresivamente por etapas ajustándose a las diferentes necesidades, pudiendo evitar el sobredimensionamiento.
- ❖ Alta disponibilidad.
- ❖ Experiencia del usuario: Analizar las cargas de trabajo y evaluar de qué modo una solución determinada se ajustará a las necesidades de cada uno de los usuarios.
- ❖ Grado de apertura: Observar si la solución VDI es lo bastante abierta para ofrecer opciones y flexibilidad en materia de protocolos y tecnologías de virtualización.

En este proyecto, no se han tenido en cuenta estos aspectos al haber realizado pruebas en un equipo local con recursos limitados.

## 3.4. Facultad de Informática de San Sebastián

### 3.4.1. Introducción

La Facultad de Informática de Donostia/ San Sebastián, dentro de la Universidad del País Vasco, se erige como un punto de referencia para la docencia y conocimiento técnico/científico en informática. Se trata de una universidad pública, investigadora, enraizada en la sociedad vasca y abierta al mundo. Ofrece un Grado en Ingeniería Informática basado en tres especialidades (Ingeniería de Computadores, Computación, Ingeniería de Software), seis másteres de postgrado y dos programas de doctorado. Para ello dispone de un equipo docente de hasta 97 miembros (57 profesores y 40 profesoras).

### 3.4.2. Infraestructura

En el caso de la FISS podemos mencionar la siguiente infraestructura que proporciona a su alumnado, profesorado y cualquier empleado de la misma:

- ❖ 21 laboratorios docentes, con laboratorios específicos de redes, robots móviles, microprocesadores, gráficos, ...
- ❖ Sala informática 24h.
- ❖ Sala de estudio y sala de Trabajos Fin de Grado.
- ❖ Servicio de copistería.
- ❖ Taberna y comedor.
- ❖ Taquillas para poder guardar en un instante dado nuestro material.
- ❖ WIFI en todo el campus. Proporciona una conexión segura debido a que tendremos que registrarnos para poder hacer uso de este servicio.

Si nos centramos ahora en las necesidades de los laboratorios docentes, estos hacen uso de imágenes de software, entendiendo por imagen un archivo con todo el contenido de un disco duro. Se pueden copiar en un disco vacío, y el resultado es un ordenador que arranca con un sistema operativo<sup>27</sup> y un software ya instalado. En el caso de la FISS, serán duales. Es decir, podremos trabajar indistintamente con dos tipos de sistemas operativos: Windows y Linux. Estas imágenes entonces son proporcionadas a los diferentes equipos informáticos que dispone la Facultad, y cada usuario podrá hacer uso de uno u otro dependiendo de las necesidades que tenga. La imagen seleccionada se cargará por red y además, se encontrará instalada en la misma aquellos programas los cuales se han considerado más necesarios para poder cumplir con todas las exigencias de cada una de las asignaturas.

El problema lo podemos encontrar en el mantenimiento de la instalación. Por ejemplo, el tamaño en bytes de una imagen de Windows es muy elevado, y podrá suponer un problema para poder virtualizar todo, en lo que respecta a tiempo de transferencia y grabación. Además, Microsoft de manera muy periódica realiza actualizaciones de su sistema, lo que puede hacer que aumente el tamaño de la imagen.

Otro inconveniente podemos encontrarlo en guardar el estado de trabajo de un alumno. Dada la situación actual, en el instante que el alumno finaliza sesión pierde todo su trabajo en caso de no haberlo guardado en algún dispositivo externo. No hay un sistema de ficheros en red eficiente<sup>12</sup> que permita su uso automático con cualquier sistema operativo<sup>27</sup>.

Ante esta situación, los técnicos de la Facultad han iniciado una prueba con un conjunto de usuarios de una asignatura concreta, en donde la idea es proporcionar un escritorio personalizado por cada usuario. Esta persona tendrá a disposición una interfaz donde inicia sesión y a continuación se le mostrará su escritorio en particular, con todo el conjunto de programas que requiera. En esta plataforma podrá guardar el estado de su trabajo pudiendo recuperarlo cada vez que acceda a su escritorio. Para alcanzar este resultado han utilizado la solución de VMware que recibe el nombre de vCenter<sup>33</sup>.

Aparentemente, la solución propuesta por los técnicos de la Facultad resulta, por cómo se ha argumentado, equivalente a una solución de virtualización de escritorio. Sin embargo, podemos distinguir diferentes limitaciones:

- ❖ No soporta el sistema de autenticación de la UPV/EHU, LDAP, debiendo de crear una nueva cuenta por cada usuario.
- ❖ La gestión de las direcciones IP de las máquinas virtuales se realiza manualmente, originando que cada máquina virtual tenga una dirección IP estática.
- ❖ No dispone de una interfaz gráfica que permita su gestión. Para poder hacer uso de ella se deberá utilizar otra herramienta adicional que es de pago. Esto origina, por ejemplo, que toda la creación y eliminación de máquinas virtuales deba llevarse a cabo vía línea de comandos.

Además, la solución VMware vCenter<sup>33</sup> no ha sido diseñada para proporcionar escritorios virtuales, sino para ofrecer servicios virtualizados.

Por lo tanto nuestro objetivo, como se ha dicho al inicio de esta memoria, será buscar una solución de virtualización de escritorio que además de cumplir con todas las especificaciones comentadas anteriormente, soporte el sistema de autenticación utilizado por la universidad, LDAP, teniendo que solventar todas las cuestiones e inconvenientes encontrados en la prueba realizada por los técnicos de la Facultad.

### 3.5. Necesidades prácticas

En este apartado se pretende dar a conocer las necesidades que todo estudiante de la Facultad de Informática de San Sebastián requiere para la correcta realización de su trabajo, de sus prácticas; y que serán tenidas en cuenta durante la realización de esta Trabajo Final de Grado.

#### ❖ Aplicaciones

Podemos destacar desde aplicaciones comunes para todas las asignaturas (navegador web, procesamiento de textos,...); hasta utilidades propias de la asignatura. Pongamos un ejemplo de esto último: Si cursamos la asignatura de Tecnologías e Infraestructuras de red (TIR) nos requerirán para la correcta realización de las prácticas el programa GNS3 en la versión que el profesor haya estimado oportuna.

#### ❖ Sistemas operativos

En base a la asignatura que estemos cursando podemos encontrarnos con la situación de requerir un sistema operativo<sup>27</sup> como es Windows u algún otro como son los de UNIX. Por ejemplo: Si cursamos la asignatura de Administración de Sistemas y Redes (ASR) nos requieren la utilización del sistema operativo<sup>27</sup> Ubuntu.

La utilización de un sistema operativo<sup>27</sup> u otro será especificada por el docente de la asignatura en cuestión.

#### ❖ Almacenar archivos

Todo el conjunto de trabajos que lleve a cabo cada alumno/a ocupará un espacio de memoria determinado así como el conjunto de aplicaciones que hace uso para la creación o alteración de las mismas. Por tanto habrá que tener presente la capacidad de almacenamiento suficiente que requiere un alumno/a, para que pueda guardar toda la labor que ha realizado de manera correcta pudiendo acceder a ella si así lo viera necesario.

Es por todo ello, que no solamente habrá que atender a las necesidades de los alumnos/as sino también a las del profesorado, así como cualquier persona con autoridad dentro de la Facultad.



# **4**

---

---

## **ESTUDIO DE ALTERNATIVAS**

## 4.1. Introducción

Si se realiza una búsqueda en Internet del concepto VDI (Infraestructura de escritorio virtual) obtendremos numerosas referencias entre las que destacan Citrix<sup>1</sup>, VMware y Microsoft. Otras grandes empresas tecnológicas también tienen sus productos: Telefónica y Huawei por ejemplo. Por otro lado aparecen proveedores como UDS y muchos otros que ofrecen unos servicios determinados de escritorio virtual. Todo ello conlleva a tener que realizar una clasificación de los distintos tipos de solución que dan los diferentes proveedores y que quedan reflejados en la siguiente tabla:

Tipo de solución	Servidor	Proveedores
Servicio de escritorio remoto (SBC)	On premise	- Solutia IT - EvaOS
Virtualización de aplicaciones	On premise	- Citrix XenApp <sup>17</sup> - Horizon Apps
Infraestructura de escritorio virtual	On premise	- QVD Qindel Group - Nologin FlexVDI - VMware Horizon 7 - Citrix XenDesktop <sup>18</sup> - VirtualCable UDS Enterprise
Escritorio como Servicio (DaaS)	Cloud	- Flexxible Desktop Cloud - OVH – Innovation is Freedom - Huawei FusionCloud Desktop - Telefónica Virtual Desktop - VirtualCable UDS Enterprise - VMware Horizon 7 - QVD Qindel Group
Software como Servicio (SaaS)	Hibrido	- Microsoft VDI

Tabla 3: Proveedores de virtualización de escritorio

Descripción de los diferentes tipos de soluciones mencionados anteriormente:

- ❖ Servicio de escritorio remoto: Los servicios de escritorio remoto permiten a un usuario acceder a las aplicaciones y datos almacenados en otro ordenador mediante un acceso por red.
- ❖ Virtualización de aplicaciones: Con la virtualización de aplicaciones el usuario es capaz de ejecutar en su ordenador una aplicación que realmente no se encuentra instalada en su equipo.
- ❖ Infraestructura de escritorio virtual (VDI): Anteriormente hemos explicado en que consiste este tipo de virtualización.
- ❖ Escritorio como servicio: Los servicios DaaS ofrecen un modelo de escritorios virtuales bajo demanda que los usuarios consumen directamente en sus dispositivos fijos y móviles sin importar el sistema operativo<sup>27</sup>, su ubicación o el tipo de red de acceso. Basta con descargar una aplicación para poder acceder a los sistemas corporativos que se hayan definido según el perfil de cada usuario. Es un modelo de pago por uso que se fija según los despliegues realizados y los usuarios concurrentes. El proveedor es el encargado de ofrecer un soporte continuo para asegurar la disponibilidad de los escritorios, ocupándose de la gestión integral de la solución VDI, tanto en su vertiente software como hardware.



- ❖ Software como servicio: El software como servicio (SaaS) permite a los usuarios conectarse a aplicaciones basadas en la nube a través de Internet y usarlas. Un ejemplo común es el correo electrónico. Toda la infraestructura subyacente, el middleware<sup>21</sup>, el software y los datos de las aplicaciones se encuentran en el centro de datos del proveedor. El modelo SaaS es similar al modelo DaaS pero presentan menor grado de automatización y características menos extensas del software.

En esta memoria nos centraremos en conocer las soluciones de virtualización que se han considerado más adecuadas en base a nuestras necesidades. Además, podremos observar cual ha sido su uso e impacto en las diferentes instituciones, centrándonos en las del ámbito educativo. De todas formas, todas las maneras de afrontar el reto de este proyecto quedan reflejadas al final de esta documentación, en el anexo C.

## 4.2. Horizon 7 - VMware

VMware Inc., es una filial de Dell que proporciona software de virtualización. Entre este software se incluye VMware Horizon que es su solución de virtualización de escritorios y aplicaciones. Con esta herramienta pretende optimizar<sup>22</sup> la distribución y gestión de aplicaciones de Windows y Linux o instalaciones en la nube con un control de acceso muy sofisticado, permitiendo a los usuarios el acceso a sus escritorios desde cualquier ubicación.

El producto de VMware Horizon está constituido a su vez de diferentes plataformas: Horizon 7, Horizon Cloud, Horizon Apps y Horizon Flex. El único que ha sido diseñado para permitir distribuir, gestionar y proteger escritorios virtuales (VDI), y aplicaciones a través de una sola plataforma es Horizon 7. La infraestructura de escritorios virtuales que define Horizon 7 puede ser ejecutada de forma local o como un servicio de escritorio alojado en la nube.

Horizon 7 está disponible en tres ediciones: estándar, avanzada y empresa/organización; y se puede elegir hasta dos modelos de licencia:

- ❖ Por usuario designado: Para entornos virtuales donde los empleados necesitan acceso exclusivo a una máquina virtual durante todo el día.
- ❖ Por conexión simultánea: Para entornos virtuales con un número elevado de usuarios que comparten las máquinas por turnos a lo largo del día (por ejemplo, alumnos, trabajadores por turno).

También tenemos que mencionar que VMware proporciona una herramienta de gestión de entornos para Horizon. Esta herramienta conocida como vRealize Operations for Horizon, supervisa el rendimiento<sup>26</sup> de los entornos de Horizon y Citrix<sup>1</sup>. Es decir, mediante una única consola los administradores obtienen parámetros de rendimiento<sup>26</sup> de los escritorios y las aplicaciones e información del uso de Horizon, XenDesktop<sup>18</sup> y XenApp<sup>17</sup>.

Para ampliar información de esta solución recomiendo acudir a las referencias que aparecen reflejadas al final de esta memoria.

### 4.3. VirtualCable – UDS Enterprise

VirtualCable es una compañía española especializada en virtualización, dedicada al desarrollo de software y servicios informáticos. VirtualCable desarrolla y soporta UDS Enterprise que da la posibilidad de administrar de principio a fin el ciclo de vida de los escritorios virtuales.

Universal Desktop Services (UDS) Enterprise es un broker de conexiones multiplataforma para:

- ❖ VDI: Administración y despliegue de escritorios virtuales Windows y Linux.
- ❖ Virtualización de aplicaciones.
- ❖ Consolidación de servicios de escritorio: Servicios en la nube, Moodle, etc.

UDS Enterprise es una solución de infraestructura de escritorios virtuales (VDI) basado en plantillas que puede encontrarse en nuestro centro de datos (solución on-premise) o en la nube, mediante módulos adicionales. Permite desplegar escritorios virtuales Windows y Linux utilizando diferentes sistemas de autenticación (LDAP, SAML, Protocolo RDP,...), y desde los principales hipervisores del mercado (VMware, KVM<sup>19</sup>, Citrix,...).

La arquitectura de una plataforma de servicios de escritorio con UDS Enterprise ha sido diseñada con dos funciones principales según afirma la propia compañía:

- ❖ Definir los recursos virtuales y físicos que ponemos a disposición de los usuarios.
- ❖ Actuar como “guardia de tráfico” redirigiendo, permitiendo o denegando el acceso a los recursos definidos por el administrador.

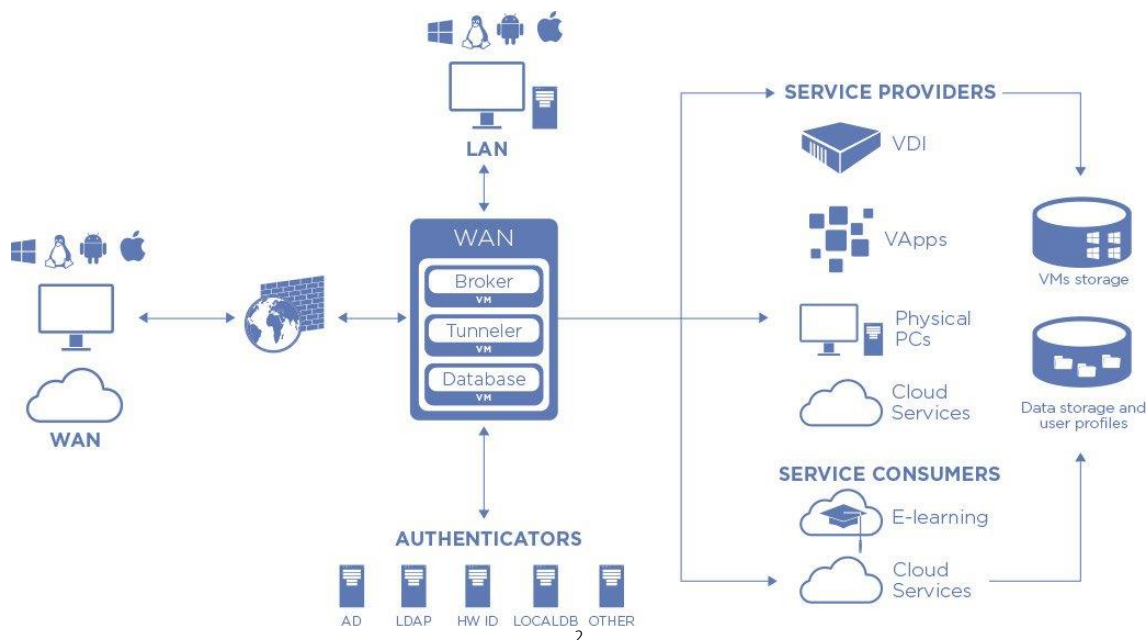


Ilustración 4: Arquitectura UDS Enterprise

<sup>2</sup> [https://www.udsenderprise.com/media/filer\\_public\\_thumbnails/filer\\_public/be/de/bede64be-8602-4fbf-9267-41c8c4d15f51/diagram-intro.jpg\\_\\_1035x562\\_q85\\_crop\\_subsampling-2\\_upscale.jpg](https://www.udsenderprise.com/media/filer_public_thumbnails/filer_public/be/de/bede64be-8602-4fbf-9267-41c8c4d15f51/diagram-intro.jpg__1035x562_q85_crop_subsampling-2_upscale.jpg)

Como se puede observar en la ilustración 4, una arquitectura de escritorios y aplicaciones virtuales con UDS Enterprise está formada por cinco elementos principales:

- ❖ Cliente de conexión: Son los dispositivos que emplean los usuarios para acceder a sus servicios de escritorio remoto (ordenador de sobremesa, thin clients<sup>28</sup>, tabletas,...). Es importante identificar si los usuarios accederán desde una WAN o desde una LAN para configurar adecuadamente la red de acceso a los escritorios y aplicaciones.
- ❖ Servidores UDS: Es el software encargado de mediar entre dispositivos clientes y proveedores de servicio. Se encarga de conectar los diferentes terminales de acceso que pueden utilizar los usuarios (PCs, portátiles, dispositivos móviles, etc.), con los proveedores de servicios definidos por los administradores de la plataforma (escritorio virtuales, aplicaciones virtualizadas, etc.). Gracias a este software, el administrador puede permitir o denegar el acceso a los diferentes servicios de escritorio remoto a un usuario o a un grupo de usuarios.
- ❖ Autenticadores: Controlan el acceso de usuarios a los servicios de escritorio remoto. Los administradores podrán configurar y habilitar uno o varios autenticadores al mismo tiempo en una misma plataforma.
- ❖ Proveedores de servicio:
  - ◆ Plataforma hipervisor: Se encarga de ejecutar las tareas de creación, encendido y eliminación de las máquinas virtuales gestionadas por el broker de conexiones.
  - ◆ Aplicaciones RDS: Es el elemento encargado de proveer las sesiones de aplicaciones que serán administradas por UDS Enterprise.
- ❖ Almacenamiento: Habrá que tener en consideración las necesidades de cada grupo de usuarios para elegir el tipo de almacenamiento que ofrezca las prestaciones más adecuadas, sin influir en el rendimiento<sup>26</sup> de los servicios de escritorio remoto.

Por tanto, una vez hemos conocido la arquitectura de UDS Enterprise, vemos la necesidad de disponer de ciertos recursos para desplegar una plataforma de servicios de escritorio remoto.

Los requerimientos básicos de infraestructura son tres:

- ❖ Plataforma de virtualización: Se encargará de alojar los servidores UDS, los escritorios virtuales y los servidores RDS de aplicaciones. Será indispensable contar con usuario y contraseña del gestor de la plataforma de virtualización con permisos de administrador.
- ❖ Servidor DNS: Es imprescindible para el buen funcionamiento tanto de la plataforma virtual como del entorno UDS que se va a desplegar.
- ❖ Servidor DHCP: Es necesario para poder asignar direcciones IP a los grupos de escritorios virtuales creados por UDS Enterprise.

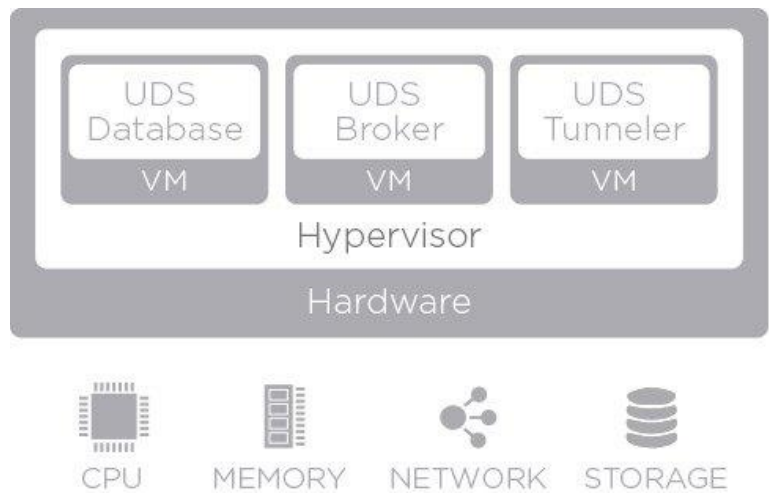


Ilustración 5: Infraestructura UDS Enterprise

VirtualCable comercializa UDS Enterprise mediante un modelo de suscripción, sin licenciamiento, incluyendo soporte y actualizaciones de producto en tramos por número de usuarios. Además, podremos conocer en todo momento el uso que se está haciendo de la plataforma.

<sup>3</sup>[https://www.udsenderprise.com/media/filer\\_public\\_thumbnails/filer\\_public/67/0a/670a827f-9b3c-4e77-914e-5c253c20186a/diagram-primeros-pasos.jpg\\_\\_483x308\\_q85\\_crop\\_subsampling-2\\_upscale.jpg](https://www.udsenderprise.com/media/filer_public_thumbnails/filer_public/67/0a/670a827f-9b3c-4e77-914e-5c253c20186a/diagram-primeros-pasos.jpg__483x308_q85_crop_subsampling-2_upscale.jpg)

## 4.4. Casos de éxito

### 4.4. 1. Introducción

A partir del siguiente apartado se conocerán las diferentes experiencias que han alcanzado ciertas organizaciones en el uso de alguna de las soluciones de virtualización de escritorio comentadas anteriormente, más en concreto aquellas organizaciones del ámbito educativo.

### 4.4.2. Universidad de A Coruña

La Universidad de Coruña es una institución pública que tiene como finalidad esencial la generación, gestión y difusión de la cultura y del conocimiento científico, tecnológico y profesional a través del desarrollo de investigación y de docencia.

La Universidad de A Coruña integra 26.000 usuarios entre estudiantes y empleados de la propia Universidad. El Departamento de TI<sup>29</sup> de la Universidad gestiona un parque de 8.000 dispositivos físicos, con diferentes necesidades en función del tipo de usuario.

Esta Universidad apostó por la solución de VMware Horizon 7. Esta compañía le facilitó diferentes herramientas permitiéndole, según lo que la propia Universidad afirma, realizar una gestión de forma flexible, centralizada y ágil de los diferentes puestos de trabajo independientemente del tipo de usuario. Por ejemplo, con la tecnología de ThinApp y Horizon Workspace, la Universidad ha podido asignar, en función del perfil del usuario, aquellas aplicaciones que necesita en un instante determinado.

La Universidad ha alcanzado ciertos resultados como desplegar hasta 8.000 escritorios virtuales que dan cobertura a todos los dispositivos físicos de la Universidad, o ahorrar en costes de adquisición y renovación de los puestos físicos de los usuarios; todo ello mediante la implantación de puestos de trabajo basados en VMware Horizon, tanto en las aulas de las diferentes facultades y escuelas universitarias, como con otro tipo de usuarios como son los de administración y biblioteca.

### 4.4.3. Universidad de Sevilla

La Universidad de Sevilla es una institución que presta un servicio público de educación superior mediante el estudio, la docencia y la investigación, así como la generación, desarrollo y difusión del conocimiento, teniendo su sede en Sevilla.

Esta Universidad con el objetivo de ofrecer a sus estudiantes entornos informáticos móviles y eficaces, buscaron inicialmente una solución basada en VMware que consistía en el alojamiento centralizado de entornos de escritorio dentro de máquinas virtuales. Pero debido a los costes de esta solución decidieron recurrir a la organización VirtualCable con su software UDS Enterprise, ya que resultaba más rentable. Otro factor que hizo escoger este software fue sus éxitos en otras universidades españolas desplegando soluciones de virtualización.

Debido a la posibilidad de poder elegir los hipervisores, autenticadores y protocolos de conexión, la Universidad alcanzó un resultado muy positivo que cumplía con todas sus expectativas reduciendo los costes (paga anualmente una cuota de suscripción), aumentando la

flexibilidad entre otras cuestiones. Para ello se decantaron por la solución de gestión de virtualización oVirt, haciendo uso de la plataforma KVM<sup>19</sup> para la creación y funcionamiento de las máquinas virtuales que forman su sistema de VDI.

#### 4.4.4. Universidad de Murcia

La Universidad de Murcia (UM) es una universidad pública española, ubicada en la Región de Murcia (España). Sus centros se encuentran distribuidos en cinco campus y tiene más de 34.000 alumnos.

Una de las aspiraciones que tenía la Universidad de Murcia era proporcionar al alumno la posibilidad de tener accesible a través de cualquier dispositivo, en cualquier momento y desde cualquier lugar, una réplica de las aulas informáticas de la Universidad. Para ello realizaron previamente un estudio y evaluación de las distintas soluciones VDI, y se decantaron por la que proporciona la organización VirtualCable con UDS Enterprise. Esto fue posible por la facilidad que proporciona a la hora de integrar escritorios Windows y Linux, la integración del sistema de autenticación que hace uso la propia Universidad (CAS), y la personalización del interfaz de usuario. Viendo los resultados que alcanzaron decidieron ampliar el uso de VDI para desplegar aulas físicas, ofreciendo así a sus alumnos la posibilidad de realizar prácticas de su titulación, cursos, o para seguir una clase impartida en ese aula en ese momento.

Esta Universidad logro grandes ventajas por medio de la implantación de esta solución:

- ❖ Ahorro en equipamiento.
- ❖ Mayor flexibilidad: Los alumnos pueden acceder desde cualquier dispositivo y el personal TI<sup>29</sup> puede gestionar cientos de máquinas desde un punto centralizado.
- ❖ Seguridad.

En este caso, esta universidad decidió hacer uso del hipervisor que proporciona VMware, vSphere<sup>31</sup>.

#### 4.4.5. Universidad Autónoma de Madrid

La Universidad Autónoma de Madrid (UAM) es una universidad pública española ubicada en Madrid. Cuenta con siete facultades (Ciencias, Derecho, Filosofía y Letras, Psicología, Medicina, Ciencias Económicas y Empresariales) dispersas por diversos edificios de la capital española, además de diferentes escuelas universitarias adscritas, alojando aproximadamente hasta 28.286 estudiantes, 2.485 profesores entre otros.

Esta Universidad disfruta desde hace años de las ventajas de la virtualización de escritorios. En 2007 comenzaron a realizar pruebas con esta tecnología implementando un piloto, y un año después decidieron pasarlo a producción.

Su elección fue utilizar el broker de conexiones UDS Enterprise ya que era el único software que les permitía entonces cumplir con sus requerimientos, en el que cabe destacar el querer desplegar escritorios Linux. Otro factor que influyó a la Universidad a escoger esta solución, fue que facilita una sencilla interfaz y da la posibilidad de elegir los hipervisores, autenticadores y protocolos de conexión.

Por medio de la utilización de esta solución han conseguido mayor flexibilidad, ya que los alumnos pueden acceder a los escritorios virtuales desde su propio dispositivo y utilizar las mismas herramientas que tiene en el aula, y los profesores pueden utilizar cualquier aplicación en cualquier aula, aunque no esté instalada. Además el personal de soporte puede desplegar cualquier programa o aplicación en cualquier máquina desde su propio laboratorio reduciendo así las cargas de trabajo. Cabe destacar que han mejorado incluso el tiempo de respuesta pudiendo de esta manera disfrutar, tanto el servicio técnico como todos los usuarios de la Universidad, de una alta disponibilidad de los recursos docentes disponibles en los escritorios virtuales.

## 4.5. Conclusiones

Dadas las circunstancias estudiadas, podemos decir que una de las mejores soluciones a implantar es la proporcionada por la organización VirtualCable, al cumplir con todas nuestras exigencias y haber alcanzado tan buenos resultados en otras universidades del estado español. Sin embargo, en caso de querer hacer uso de la misma, vamos a requerir de una gran inversión económica anual. Esta dependerá del número de usuarios que se suscriban pudiendo llegar a un precio máximo de 20.000 euros. Sobre la misma se puede hacer un descuento al ser en este caso orientada al ámbito educativo de hasta el 33,33%. Es por esa razón, por la que se ha pensado enfocar nuestro estudio en soluciones de software libre que proporcionen una gestión centralizada en una nube privada<sup>24</sup> y que puedan cumplir con los objetivos marcados para este proyecto. Estamos hablando entonces de la plataforma de computación en la nube OpenStack, la cual será explicada con detenimiento en el siguiente apartado.



**5**

---

---

**OPENSTACK**

## 5.1. OpenStack

El proyecto OpenStack se define a sí mismo como una plataforma de Cloud Computing (Anexo A) hecha con software libre para desplegar nubes públicas y privadas, desarrollado con la idea de ser sencillo de implementar, escalable<sup>13</sup> y con muchas prestaciones. OpenStack proporciona una solución de Infraestructura como Servicio (IaaS)<sup>18</sup> a través de un conjunto de servicios interrelacionados.

El proyecto OpenStack se inició en 2010 por la empresa Rackspace Cloud y por la agencia norteamericana, NASA. Actualmente más de 150 empresas se han unido al proyecto, entre las que se encuentran empresas tan importantes como AMD, Intel, Canonical, Red Hat, IBM, Dell, HP, Cisco, etc. OpenStack ha optado por utilizar la licencia de Apache 2.0, una licencia de software libre permisiva y no copyleft<sup>11</sup>. En el caso de OpenStack se ha decidido además por un desarrollo completamente abierto, en que se pueden aceptar opiniones y contribuciones de cualquiera, debiendo hacer uso del lenguaje de programación Python en lo que a la generación del código se refiere.

Por lo tanto, OpenStack es una tecnología que consiste en una serie de proyectos relacionados entre sí en los que participan numerosas empresas (Huawei, AMD, Cisco, Dell, VMware, OVH, Red Hat, etc); para realizar un control de procesamiento, almacenamiento y recursos de red a través de un centro de datos; todos administrados a través de un panel de control que permite a los administradores realizar un seguimiento y control, mientras proveen a los usuarios de los diferentes recursos que requieran a través de una interfaz web.

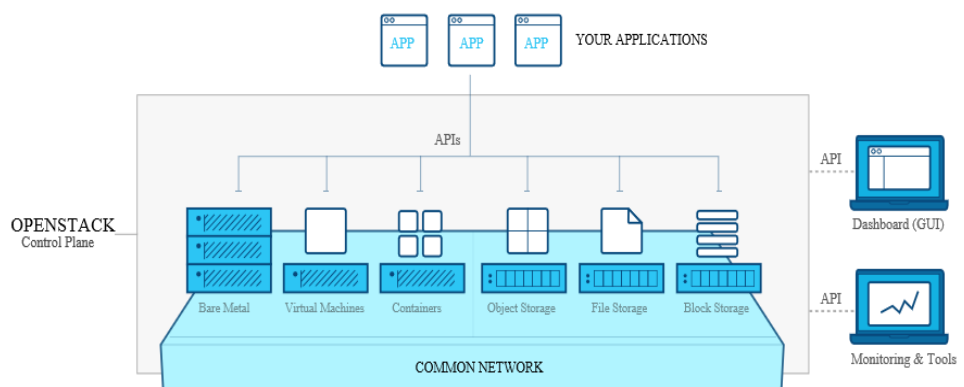


Ilustración 6: Arquitectura OpenStack

El motivo de haber optado por esta solución ha sido porque cumple con todas las necesidades requeridas en el proyecto:

- ❖ Se trata de una solución escalable<sup>13</sup> ya que podemos reutilizar sus componentes en el futuro.
- ❖ Puede tener un precio nulo en licencias, en base a la distribución que se escoja.
- ❖ Se podrá adecuar a la infraestructura de comunicaciones de la Facultad de Informática de San Sebastián (FISS) y al acceso remoto.
- ❖ Soporta el sistema de autenticación LDAP.
- ❖ Ofrece como servicio de Cloud Computing una Infraestructura (IaaS)<sup>18</sup> proporcionando una gestión centralizada.

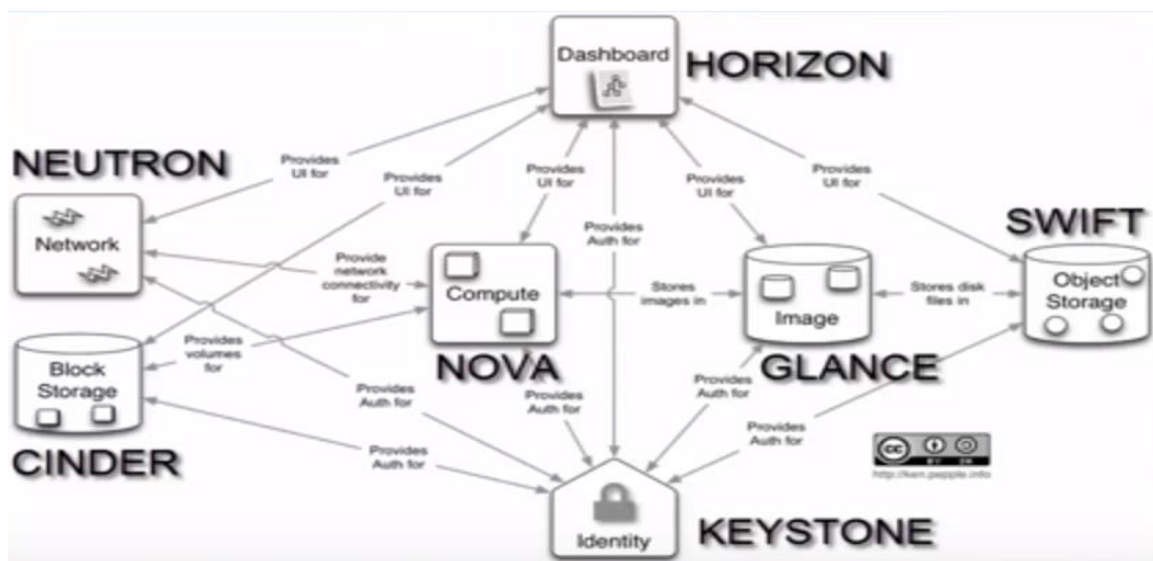
<sup>4</sup> <https://www.openstack.org/software/images/diagram/overview-diagram.svg>

## 5.2. Componentes

OpenStack no es un solo un producto, sino que es la agrupación de varios módulos cada uno de ellos con una función específica y que se pueden instalar de manera separada o conjunta, es decir, en el mismo servidor o en diferentes, según la distribución que se elija siendo algunos de estos componentes obligatorios y otros no.

Por lo tanto, OpenStack es una arquitectura modular que dispone, entre otros, de los siguientes elementos:

- ❖ Nova (Compute): Es el módulo que controla la gestión y ejecución de las máquinas virtuales pudiendo trabajar con muchos hipervisores.
- ❖ Horizon (Dashboard): Es la interfaz gráfica para poder gestionar el acceso, la provisión, etc.
- ❖ Neutron (Networking): Es el módulo que gestiona todo lo relacionado con la red.
- ❖ Keystone (Identity Service): Módulo relacionado con todo lo que tiene que ver con autenticación de usuarios y políticas.
- ❖ Glance (Image Storage): Gestiona todas las imágenes de los sistemas operativos en forma de plantilla listas para su uso.
- ❖ Cinder (Block Storage): Proporciona dispositivos de almacenamiento a nivel de bloque.
- ❖ Swift (Object Storage): Es el que contiene los diferentes objetos. Se entiende por objeto a una entidad única que contiene información como pueden ser por ejemplo una clave.



5

Ilustración 7: Componentes OpenStack

<sup>5</sup><https://virtualizadesdezero.com/wp-content/uploads/2017/03/openstack-conceptual-arch-folsom-compressor.png>

En la mayoría de instalaciones, el proceso que se sigue para lanzar instancias, es decir, lanzar y tener disponible una máquina virtual, es prácticamente el mismo:

- I. El usuario accede al panel web que proporciona el módulo “Horizon” e inicia sesión.
- II. Entra en juego el componente “Keystone” para autenticar al usuario y darle los permisos correspondientes.
- III. Una vez que se ha validado al usuario, este podrá consultar, gracias a “Glance”, la lista de imágenes disponibles ya que se encuentran almacenadas en el módulo “Swift”.
- IV. El usuario escoge una de las imágenes y el componente “Nova” la despliega según las características de cómputo que el usuario necesite.
- V. Finalmente, el componente “Cinder” almacena todo lo relativo a la instancia y el módulo “Neutron” establece la red.

De esta manera, en cuestión de segundos, podremos tener disponible la máquina que deseemos. Quisiera destacar llegados a este punto dos situaciones:

- ❖ Si la plataforma en un instante dado sufre cualquier problema que origina su no disponibilidad, en el momento que se recupere nos encontraremos en la situación anterior al inconveniente. En el caso de las instancias, por ejemplo, bastaría con reiniciarlas.
- ❖ Si deseamos guardar el estado en el que se encuentra una instancia en un momento determinado, OpenStack nos lo proporciona por medio de la creación de instantáneas.

### 5.3. Distribuciones

Como se ha comentado anteriormente, OpenStack es un software libre por lo que cada empresa puede escoger los módulos que desee, haciendo el uso que vea necesario de los mismos, para ofrecerlos finalmente a sus clientes.

Las principales distribuciones de Openstack son las siguientes:

- ❖ VMware Integrated OpenStack

VMware Integrated Openstack es una distribución de OpenStack respaldada por VMware que permite su instalación dentro de su infraestructura vSphere<sup>31</sup>. Es decir, se implementa como clústeres de administración y de procesos en el entorno de vSphere<sup>31</sup>.

- ◆ Clúster de proceso: Controla todas las cargas de trabajo.
- ◆ Clúster de administración: Contiene las máquinas virtuales que componen la implementación de OpenStack. También dispone de los servicios de caché en memoria, cola de mensajes, equilibrio de carga, DHCP y base de datos.

La arquitectura de VMware Integrated OpenStack conecta los recursos de vSphere<sup>31</sup> con los componentes de computación, de red, almacenamiento de bloques, gestión de imágenes, de identificación y de orquestación de OpenStack.

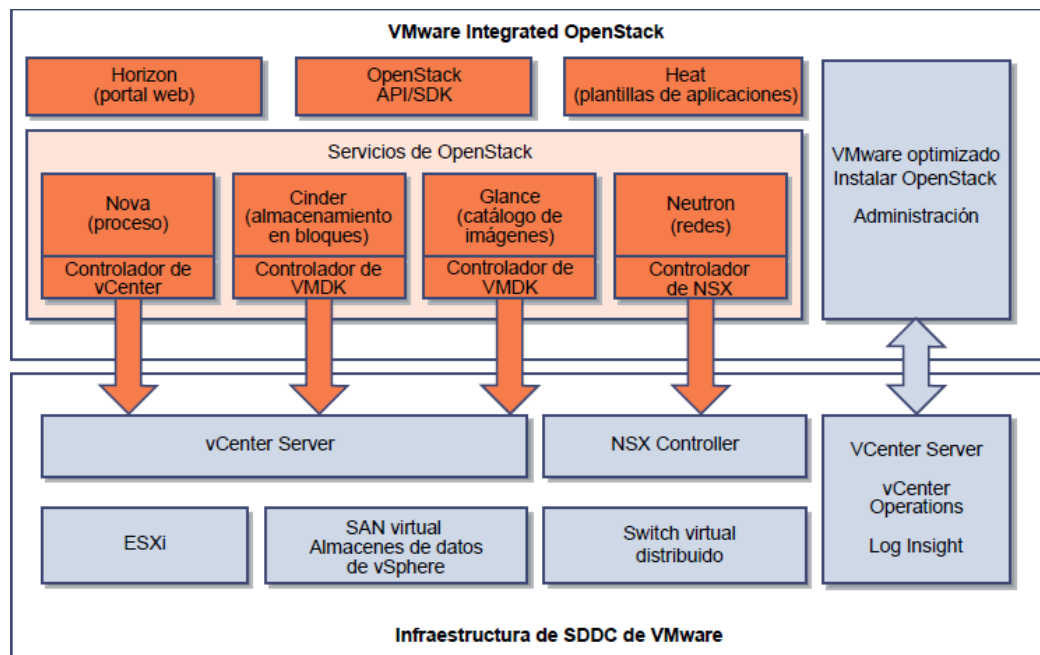


Ilustración 8: Infraestructura VMware Integrated OpenStack

<sup>6</sup><http://pubs.vmware.com/integrated-openstack-2/topic/com.vmware.ICbase/PDF/integrated-openstack-20-install-config-guide.pdf>

## ❖ Red Hat OpenStack Platform

Red Hat, Inc. es una multinacional americana que provee software principalmente a empresas. Red Hat proporciona almacenamiento, plataformas de sistemas operativos, middleware<sup>21</sup>, aplicaciones, productos de administración y servicios de soporte, capacitación y consultoría.

Red Hat, hoy por hoy, es el principal colaborador del proyecto OpenStack y ha diseñado una solución conocida como Red Hat OpenStack Platform. Esta resolución combina la potencia de la distribución comercial de GNU/Linux creada por la propia multinacional (Red Hat Enterprise Linux (RHEL)), con la plataforma Cloud de OpenStack de Red Hat para proporcionar una base escalable<sup>13</sup> y segura a la hora de construir una nube pública o privada. Se trata por tanto de una de las distribuciones OpenStack más maduras y con mejor soporte con el inconveniente que se tendrá que pagar una licencia por uso.

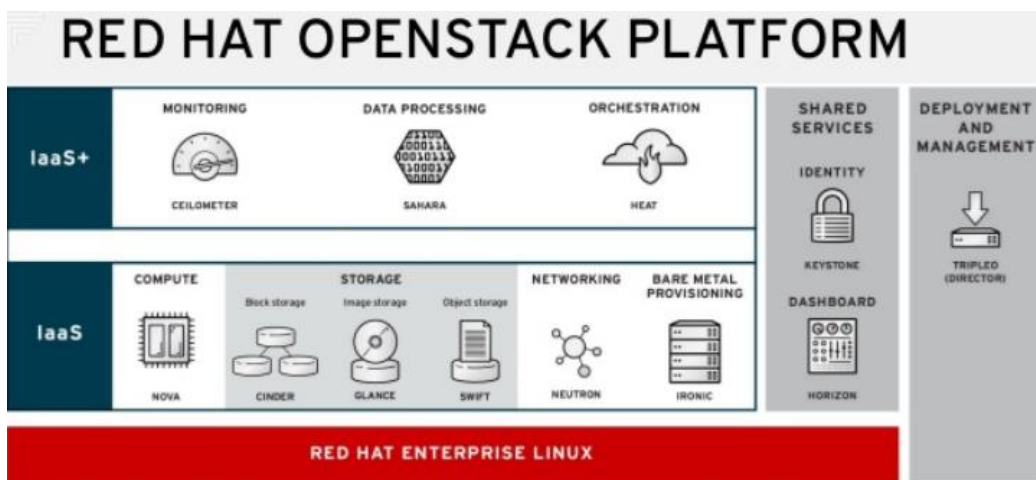


Ilustración 9: Infraestructura Red Hat OpenStack Platform

## ❖ RDO Project

RDO es una distribución mantenida por una comunidad de usuarios cuyo objetivo es facilitar el acceso a OpenStack. Está disponible para montar en CentOS, Fedora y Red Hat.

RDO ha diseñado OpenStack PackStack que permite realizar una instalación más sencilla de la plataforma OpenStack, utilizando para ello el sistema Linux de Red Hat el cual tiene una versión de pruebas de 60 días, o en el sistema CentOS que no tiene ningún requisito de licencia. Ese proyecto proporcionado por RDO, permite montar un entorno de pruebas para testear todo lo que OpenStack es capaz de ofrecerte.

<sup>7</sup> <https://i.ytimg.com/vi/vDLpZtkflzs/maxresdefault.jpg>

## ❖ Mirantis OpenStack

Mirantis Inc. es una empresa de servicios de computación en la nube y es uno de los principales contribuidores al proyecto OpenStack. Mirantis trabaja para construir soluciones en la nube pública, privada y de proveedores de servicios basadas en OpenStack, usando herramientas de código abierto<sup>8</sup> para integrar las capacidades de computación, redes y almacenamiento que definen una infraestructura OpenStack.

La distribución Mirantis OpenStack funciona con varios sistemas operativos, incluidos Red Hat, CentOS y Ubuntu Linux.

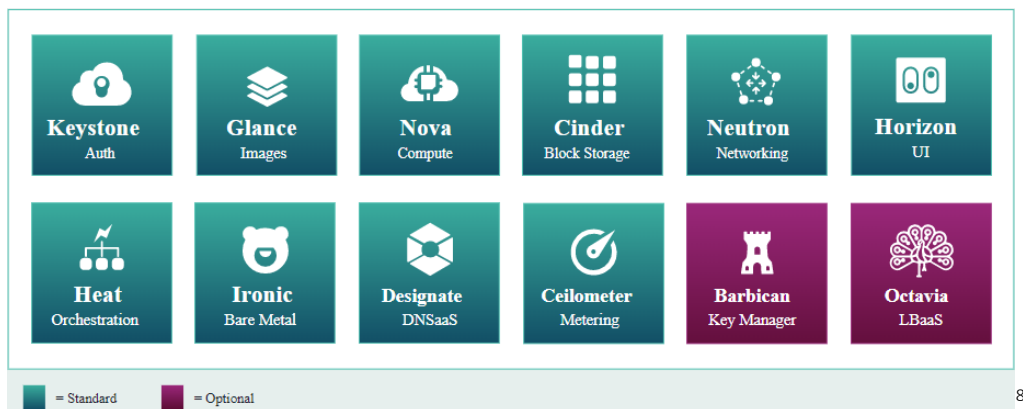


Ilustración 10: Mirantis Inc. OpenStack

## ❖ Ubuntu OpenStack

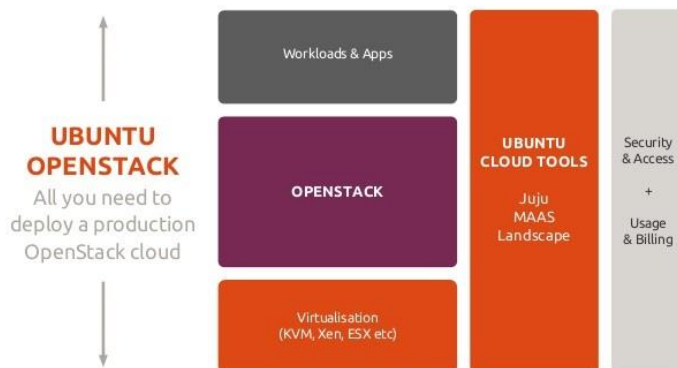
Ubuntu es una distribución del sistema operativo<sup>27</sup> GNU/Linux y que se distribuye como software libre gracias a su patrocinador, Canonical. Ubuntu promueve una de las soluciones de OpenStack más populares.

Para poder llevar a cabo todo ello se basa en un conjunto de herramientas de gestión y automatización:

- ❖ Ubuntu Server: Será el núcleo de Ubuntu OpenStack, incluyendo la última versión de OpenStack, habiendo sido probada, empaquetada y completamente integrada.
- ❖ Landscape: Herramienta que nos permitirá realizar un control de las actualizaciones del sistema y el software, además de monitorizar múltiples parámetros (uso de disco y memoria, la carga del sistema, métricas personalizadas) de los equipos que tenemos bajo control.
- ❖ MAAS: Según Canonical, MAAS apoya el despliegue de infraestructuras como OpenStack, Hadoop, CloudStack entre otras. Los servidores se asignan fácilmente a MAAS mediante una inscripción automática y luego esta herramienta es responsable de todos los aspectos de mantenimiento de los servidores físicos: actualizaciones de firmware, pruebas burn-in, evaluación del rendimiento<sup>26</sup> y otras operaciones similares.
- ❖ JUJU: Permite la gestión de Ubuntu OpenStack desde el navegador. Es un framework<sup>16</sup> de orquestación para el despliegue de servicios de infraestructura en la nube.

<sup>8</sup> <https://www.mirantis.com/json/openstack-services.svg>

Finalmente, la instalación de OpenStack puede llevarse a cabo en un equipo local para conocer las capacidades de esta distribución, o creando nuestro propio entorno en un cluster<sup>7</sup> o en un conjunto de máquinas físicas, entre otras posibles que ofrece Canonical.



9

Ilustración 11: Estructura - Ubuntu OpenStack

#### ❖ OpenStack DevStack

DevStack es un proyecto de código abierto<sup>8</sup> impulsado por la comunidad OpenStack, que proporciona secuencias de comandos y controladores para instalar OpenStack en una sola máquina. Se utiliza interactivamente como un entorno de desarrollo y como base para gran parte de las pruebas funcionales del proyecto OpenStack.

DevStack se basa entonces en una serie de scripts extensibles, utilizados para levantar rápidamente un entorno completo de OpenStack basado en las últimas versiones de todos sus componentes, pudiendo llevar a cabo de esta manera el conjunto de pruebas que deseemos sobre la plataforma OpenStack.

<sup>9</sup> <https://image.slidesharecdn.com/cloudcoredeckfeb2014-141026224112-conversion-gate01/95/ubuntu-cloud-core-deck-feb2014-12-638.jpg?cb=1414390507>



## 5.4. Decisión

Acabamos de ver las distribuciones que son más importantes y usadas pero hay otras que no se han mencionado: HPE Helion OpenStack (HP), Dell EMC Red Hat Cloud Solution, MetaCloud (Cisco), FusionSphere OpenStack (Huawei), IBM Cloud Manager con OpenStack, y más.

Dado el conjunto de distribuciones que tenemos a nuestro alcance, se recomienda revisar las características hardware disponibles (memoria, procesador, etc.) al igual que tener en consideración cual es nuestro objetivo, para escoger el entorno más adecuado. En esta ocasión, debido a que algunas distribuciones hacen uso de licencias y a las limitaciones de hardware disponible, se ha decidido hacer uso de una de las siguientes distribuciones:

- ❖ OpenStack DevStack.
- ❖ Ubuntu OpenStack.
- ❖ OpenStack PackStack con CentOS (RDO Project).

Se tratan de entornos sencillos de implantar, permitiendo realizar un uso completo de la plataforma OpenStack. Además, permite su instalación en equipos personales, es decir, requerirá menos recursos de computación.

Durante este proyecto se ha intentado implantar las dos primeras soluciones (OpenStack DevStack y Ubuntu OpenStack), pero debido a problemas encontrados en la instalación y a las limitaciones del hardware, no se han podido tener disponibles. Esto ha dado lugar a implementar la distribución basada en el sistema operativo<sup>27</sup> CentOS y PackStack con una de las versiones más recientes de OpenStack, Ocata. De esta manera además, se ha podido personalizar la instalación de OpenStack escogiendo que componentes deseo hacer uso, y permitiéndome alterar casi todas las propiedades que puede tener OpenStack, desde contraseñas, redes, hipervisores o tamaños de discos entre otras. Toda esta instalación y configuración queda reflejada al final de esta documentación, en el anexo D.

Una vez hemos sido capaces de realizar correctamente la instalación de la plataforma OpenStack, deberemos de administrarla, es decir, hacer uso de cada uno de sus componentes para alcanzar el objetivo de este proyecto: ser capaces de proporcionar un escritorio virtual. Previamente a esta tarea, se deberá de aprender diferentes conceptos para poder realizar una gestión adecuada, pudiendo recurrir a una de las siguientes formas:

1. Utilizar la documentación oficial de OpenStack. Encontraremos un gran conjunto de documentos de la forma en la que se puede realizar la gestión más adecuada de la plataforma.
2. Llevar a cabo un curso oficial de OpenStack. Esto supone realizar, en la mayoría de los casos, un pago pudiendo alcanzar incluso el certificado oficial que te acredita como administrador de OpenStack (COA).
3. Buscar por páginas no oficiales de OpenStack o por medio de vídeos tutoriales, como poder llevar a cabo una administración básica de la plataforma.

Debido al tiempo que tenemos para la realización del proyecto y al querer realizar unas pruebas básicas sobre OpenStack, se ha decantado por hacer un estudio de la gestión de la plataforma por medio de la tercera solución propuesta sin quitar la posibilidad de recurrir a la documentación oficial de OpenStack si se viera necesario. Todos los aspectos de administración estudiados durante este proyecto quedan reflejados, al igual que sucedía con la instalación de la plataforma, al final de esta documentación, más en concreto, en el anexo E.



# 6

## PRUEBAS REALIZADAS

## 6.1. Introducción

El objetivo que tendrá este apartado será el de asentar las bases sobre lo que la plataforma OpenStack es capaz de ofrecernos en base a los conceptos adquiridos a lo largo de esta documentación. Por esa razón, se realizarán diversas pruebas que se detallarán en los subapartados próximos, debiendo especificar por cada uno de ellos el objetivo que tienen propuesto así como el resultado que se ha alcanzado.

Quisiera destacar, que tanto la instalación de la plataforma que se ha mencionado en el apartado anterior como la realización de estas pruebas, se han llevado a cabo en dos equipos en préstamo: uno cedido por un compañero, otro cedido por la Facultad.

## 6.2. Prueba 1: Instanciación de la imagen y aplicación del protocolo SSH

Por medio de esta prueba se pretende poner en práctica todo lo aprendido para poder realizar una administración básica de la plataforma OpenStack, pudiendo ofrecer a cualquier usuario una instancia a un escritorio virtual.

Lo que se desea es precisamente hacer uso del conjunto de servicios que tiene definido el propio soporte, para poder instanciar una imagen de un sistema operativo<sup>27</sup> en cuestión. Además, dar la posibilidad de acceder a la misma de manera local o externamente por medio del protocolo de comunicaciones SSH. Esto implicará tener que conocer como es el funcionamiento de dicho protocolo en la plataforma OpenStack.

Por tanto la meta que se fija con esta prueba, es que un usuario pueda instanciar una imagen de un sistema operativo<sup>27</sup> en cuestión y acceder a la misma, bien localmente, o bien remotamente por medio del protocolo de comunicaciones SSH. Por esa razón realizaremos una breve explicación de lo que se entiende por protocolo de comunicaciones SSH, para luego poder explicar el procedimiento que se debe de seguir para alcanzar la meta de esta primera prueba.

### 6.2.1. Secure Shell (SSH)

SSH o Secure Shell, es un protocolo de administración remota que permite a los usuarios controlar y modificar sus servidores remotos a través de Internet. El servicio se creó como un reemplazo seguro para el servicio Telnet para garantizar que todas las comunicaciones hacia y desde el servidor remoto sucedan de manera encriptada.

En sistemas Linux o MacOS, los usuarios pueden hacer uso de este protocolo en su servidor remoto directamente desde la ventana del terminal. Por otra parte, en los sistemas Windows los usuarios pueden aprovechar aplicaciones, comúnmente denominados clientes SSH, como es PuTTY.

Por lo tanto, SSH es un protocolo que hace uso del modelo cliente-servidor para permitir la autenticación de dos sistemas remotos garantizando medidas de seguridad por medio del cifrado de datos que pasa entre ellos.

## 6.2.2. Procedimiento

Planteado el objetivo de esta prueba queda resolver como llevarlo a cabo. Todo ello lo explicaremos a continuación, siendo aconsejable recurrir al anexo E, de conceptos de administración:

1º Lo primero que deberemos realizar será la configuración de la red. Para ello nos apoyaremos en el servicio Neutron. Pero, ¿por qué debemos en primer lugar configurar la red de la plataforma? Inicialmente, una vez finalizas con la instalación de OpenStack, no se dispone de ninguna red, ni interna ni externa. También podemos encontrarnos en la situación en donde creamos un nuevo proyecto y no tiene ninguna red interna asociada sobre la misma. En caso de encontrarnos con alguna de estas situaciones o similares, lo más conveniente es asentar la red de la plataforma.

Por ese motivo comentado, se creará en primer lugar una red interna que será la que albergue nuestra instancia dotándola a la misma de una dirección IP privada.

A continuación, se añade una red externa que permitirá acceso desde el exterior a las instancias. Además para alcanzar esa meta, sobre esta red externa deberemos reservar un conjunto de direcciones IP, más conocidas como direcciones IP flotantes, teniendo en consideración que el conjunto que se generen limitaran el número de instancias que podrán acceder y ser reconocidas desde el exterior.

Finalmente, para alcanzar la conexión deseada se deberá hacer uso de un router virtual sobre el que conectaremos por una parte nuestra red interna y por otra nuestra red externa. Este router virtual hará la función de un adaptador de red NAT, permitiendo que la instancia pueda acceder y ser reconocida externamente.

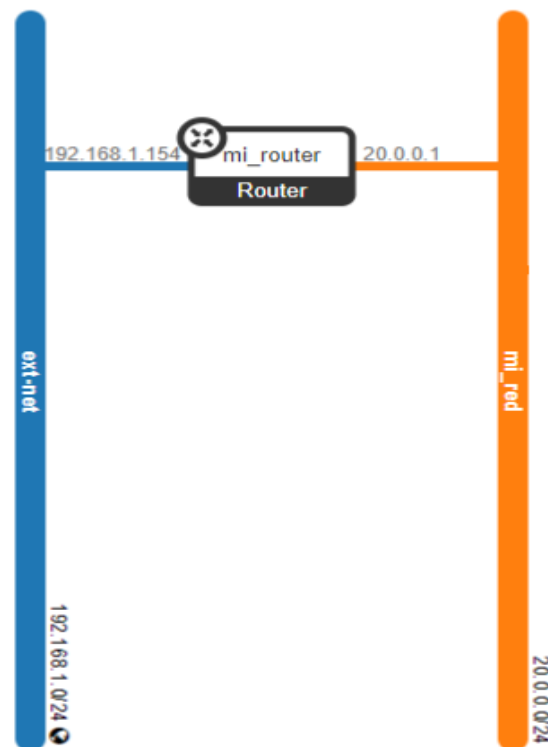


Ilustración 12: Procedimiento SSH - Topología de red

2º Teniendo configurada la red de la plataforma, deberemos subir una imagen en uno de los formatos soportados por OpenStack. Esta imagen representará un sistema operativo<sup>27</sup> en particular.

4º Siendo algo recomendable y de manera opcional, podemos crear un nuevo volumen. Como ya sabemos para ello tenemos el servicio Cinder el cual nos permitirá crear un volumen con una particulares en cuestión. La creación de este volumen nos servirá para guardar todo lo relativo a la instancia que lancemos.

5º Llegados a este punto es donde instanciaremos la imagen dotándola de las características necesarias para poder lanzarse y ser reconocida desde el exterior. Para ello previamente deberemos realizar lo siguiente:

5.1. Como ya se menciona en el anexo E de esta memoria, en donde se adquieren los conocimientos básicos de administración de la plataforma, para que una instancia sea particular de un usuario se requieren de una clave privada y de una clave pública. Las formas que permite la plataforma para la creación de claves es muy diversa. Se puede realizar la creación desde la interfaz web, desde la línea de comandos o incluso importar unas ya disponibles. Por lo tanto, el usuario deberá generar sus propias claves pública y privada según estime oportuno.

5.2. Deberemos configurar el grupo de seguridad. Deberemos habilitar el servicio ICMP para que la instancia sea reconocida desde el exterior y activar el puerto 22 correspondiente al protocolo SSH, permitiendo de esta forma que la instancia pueda ser accedida remotamente por medio de ese protocolo de comunicaciones.

Una vez realizado los pasos comentados, lanzamos la instancia. Para ello en primer lugar seleccionamos los recursos hardware que va a requerir y que son permitidos por la plataforma (sabor); a continuación, escogemos el grupo de seguridad, seleccionamos la clave pública y finalmente la asociamos a la red interna en cuestión. Todo ello sin olvidarnos de dotar a la instancia de un nombre. Cabe mencionar que en la creación de la instancia, si lo deseamos, podemos generar también y asociar automáticamente un volumen para esa instancia en cuestión.

Con todo ello realizado tendremos que disponer de una instancia en funcionamiento con una dirección IP privada disponible. Esto permitirá que la instancia sea reconocida internamente. Lo único que quedaría pendiente sería asociarle un volumen donde almacenaría todo aquello particular de la misma, siempre y cuando no se haya creado en el lanzamiento de la imagen, y una dirección IP flotante. Para esta última, se deberá recurrir al conjunto de direcciones que dispone la red externa y posteriormente, asociar una de ellas a la instancia. De esta forma, ya dispondríamos de un escritorio virtual al cual se puede acceder externamente por medio del protocolo SSH.

6º En este último paso, realizaremos una comprobación para verificar que efectivamente podemos acceder a esa instancia de manera remota por medio del protocolo SSH. Para ello en primer lugar desde nuestro equipo local probamos a reconocer la instancia por medio de la instrucción “ping”<sup>9</sup>.

```
C:\Users\ehu\QEMU>ping 192.168.1.159

Haciendo ping a 192.168.1.159 con 32 bytes de datos:
Respuesta desde 192.168.1.159: bytes=32 tiempo=7ms TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63

Estadísticas de ping para 192.168.1.159:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 7ms, Media = 1ms
```

Ilustración 13: Procedimiento SSH - Resultado reconocimiento de la instancia

En caso de realizarlo correctamente, podríamos hacer uso de un comando en cuestión o de una aplicación en base al sistema que tengamos, para poder acceder por SSH a dicha instancia. Recordemos la necesidad de hacer uso de la clave privada para poder alcanzar esa conexión, pudiéndonos reflejar errores en caso contrario.

```
fedora@instancia-fedora:~
login as: fedora
Authenticating with public key "imported-openssh-key"
Passphrase for key "imported-openssh-key":
[fedora@instancia-fedora ~]$
```

Ilustración 14: Procedimiento SSH - Resultado conexión

Me gustaría destacar, que muchas de las cuestiones anteriormente comentadas solamente pueden ser gestionadas por el administrador de la plataforma. Indistintamente podrán realizarse haciendo uso de la interfaz, gráfica gracias al servicio Horizon, o desde la línea de comandos CLI por medio de los comandos propios de cada servicio o de los proporcionados por OpenStackClient.

En la explicación se ha supuesto que la imagen que se ha subido a la plataforma tiene disponible los paquetes necesarios para poder ser accesible de manera remota por medio del protocolo SSH. En caso de que la imagen no tenga ello disponible, deberá de instalarse y configurarse correctamente. En caso contrario, en ningún caso se alcanzará el resultado deseado.

Para representar y asentar adecuadamente los conocimientos anteriormente descritos, en el anexo F se representa todo el procedimiento descrito para alcanzar una conexión remota a una instancia (máquina virtual), que en ese caso, representa a un sistema Fedora. Toda esta prueba se basará en un conjunto de ilustraciones con los diferentes comandos o acciones que son necesarias para alcanzar el resultado deseado. En ese anexo además se detallarán, las metas logradas y las diferentes dificultades que se han encontrado. `

## 6.3. Prueba 2: Instanciación de la imagen y aplicación de escritorio VNC

En esta ocasión, lo que se desea es tener disponible un escritorio virtual al que se puede acceder remotamente por medio de la aplicación VNC. Para ello, deberemos instanciar en la plataforma OpenStack, una imagen de un sistema operativo<sup>27</sup> en particular que disponga de interfaz gráfica. Esta imagen tendrá que ser accesible internamente y también externamente por medio de la aplicación VNC. Para ello partiremos de la situación alcanzada en la prueba anterior, debiendo conocer esta vez los mecanismos que proporciona la plataforma OpenStack así como la configuración que se requiere, para poder hacer uso de la aplicación VNC. Además, habrá que atender a las limitaciones hardware disponibles en la plataforma a la hora de escoger la imagen del sistema operativo<sup>27</sup> en cuestión.

Al igual que en la prueba anterior, nos basaremos en los conceptos de administración adquiridos en el anexo correspondiente, y describiremos en primer lugar que se entiende por VNC y las soluciones que propone OpenStack, para explicar a continuación, el procedimiento que se requiere para alcanzar el resultado deseado.

### 6.3.1. VNC

VNC corresponde a las siglas en inglés de Virtual Network Computing (en español, Computación Virtual en Red). VNC es un programa de software libre basado en una estructura cliente-servidor que permite observar las acciones del ordenador servidor remotamente a través de un ordenador cliente. VNC no impone restricciones en el sistema operativo<sup>27</sup> del ordenador servidor con respecto al del cliente: es posible compartir la pantalla de una máquina con cualquier sistema operativo<sup>27</sup> que admita VNC conectándose desde otro ordenador o dispositivo que disponga de un cliente VNC portado.

Según lo argumentando anteriormente, vamos a requerir una aplicación que recibe el nombre de VNC Server para el equipo que se desea controlar, y otra aplicación conocida como VNC Viewer para el equipo o dispositivo móvil desde el cual se desea ejercer el control. Entonces, cada una de estas aplicaciones se instalará en el equipo en cuestión. En nuestro caso, habrá que estudiar de qué forma la plataforma OpenStack permite la conexión remota por VNC, mientras que se requerirá tener disponible un servidor VNC en la imagen que posteriormente se importe a la aplicación VNC. Finalmente, en nuestro equipo local se llevará a cabo la instalación del servidor cliente. Para obtener esta aplicación de VNC Viewer se ha recurrido a la empresa RealVNC, más concretamente al siguiente enlace:

- ✓ <https://www.realvnc.com/es/connect/download/viewer/>

De manera de síntesis, la herramienta conocida como VNC es un sistema de poder compartir nuestro equipo de forma gráfica, pudiendo conectarlo remotamente con otro ordenador. Esta herramienta, además, consigue transmitir las órdenes del teclado y el ratón desde un ordenador a otro, y al mismo tiempo poder refrescar la pantalla con todo lo que va ocurriendo. Todo ello es posible gracias al protocolo de comunicación en el que está fundamentado VNC, RFB. Se trata de un conjunto de instrucciones muy simple que está basado en un sistema primitivo gráfico de cliente-servidor, y tiene la capacidad de transferir mensajes de eventos.



### 6.3.2. VNC en OpenStack

Para proporcionar una consola remota o un acceso de escritorio remoto a las máquinas virtuales, OpenStack proporciona dos herramientas: VNC proxy o SPICE HTML5. Estas herramientas pueden ser usadas desde la línea de comandos o a través de la interfaz web, el servicio Horizon.

#### ❖ Consola SPICE

El servicio de computación de OpenStack, comúnmente denominado Nova, admite consolas VNC para invitados. El protocolo VNC es bastante limitado, ya que no admite varios monitores, audio bidireccional, corte y pegado confiable, transmisión de video y más. SPICE es un nuevo protocolo que tiene como objetivo abordar las limitaciones en VNC y proporcionar un buen soporte de escritorio remoto.

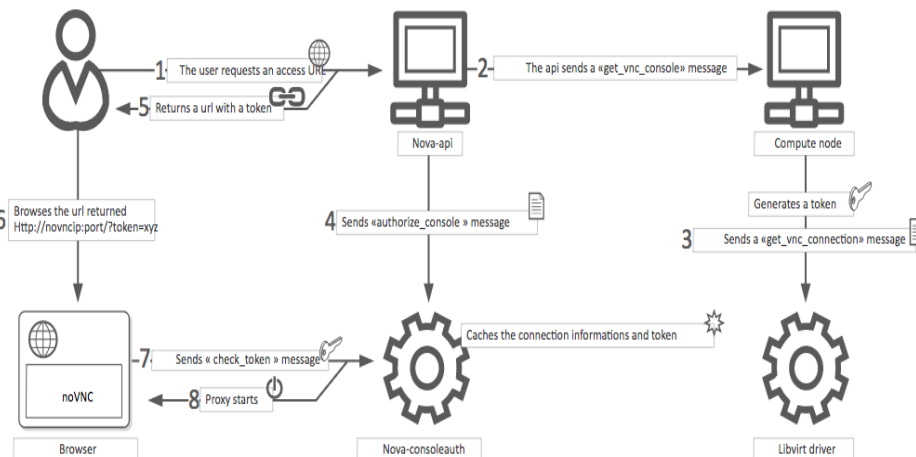
Para obtener acceso a la consola SPICE, VNC debe estar explícitamente inhabilitado.

#### ❖ Proxy de la consola VNC

Se entiende por proxy en una red informática un servidor, programa o dispositivo, que hace de intermediario en las peticiones de recursos que realiza un cliente a otro servidor. Por ejemplo, si una máquina A solicita un recurso a una máquina C, lo hará mediante una petición a B, que a su vez trasladará la petición a C.

El proxy VNC es un componente de OpenStack que permite a los usuarios del servicio de computación acceder a sus instancias a través de clientes VNC. En general el proxy VNC tendrá las siguientes funciones:

- ❖ Establecer un puente entre la red pública donde viven los clientes y la red privada donde viven los servidores VNC.
- ❖ Media la autenticación del usuario (token).
- ❖ Trata de forma transparente los detalles de conexión específicos del hipervisor para proporcionar una experiencia uniforme del cliente.



10

Ilustración 15: Sistema de funcionamiento - OpenStack VNC Proxy

<sup>10</sup> [https://docs.openstack.org/nova/pike/\\_images/SCH\\_5009\\_V00\\_NUAC-VNC\\_OpenStack.png](https://docs.openstack.org/nova/pike/_images/SCH_5009_V00_NUAC-VNC_OpenStack.png)

A la hora de personalizar la consola VNC, se puede usar las diferentes opciones de configuración que se encuentran en el archivo “nova.conf”.

El funcionamiento por tanto se prevé sencillo. Disponemos de una instancia sobre la que tenemos que disponer de un servidor VNC activado. Al encontrarse esta dentro de la plataforma OpenStack, deberemos permitir que se pueda realizar esa conexión con el escritorio remoto impulsando el proxy VNC. Será entonces cuando podremos acceder a esa instancia haciendo uso de VNC Client sin tener por que encontrarnos con ningún problema.

Dadas estas dos alternativas que ofrece OpenStack para hacer uso de la aplicación VNC, habrá que decir por cual nos postulamos. Al encontrarse la consola SPICE en una fase de prueba y de estudio, se recomienda utilizar el proxy VNC.

### 6.3.3. Procedimiento

Mencionemos los pasos que requerimos para alcanzar la meta descrita:

1º Poner en funcionamiento la herramienta de OpenStack que permite la conexión remota por medio de la aplicación VNC. Es recomendable, dado que la consola SPICE aún se encuentra en una fase de prueba y estudio, hacer uso del proxy VNC. Para su puesta en marcha deberemos dirigirnos al fichero de configuración “nova.conf” en donde encontramos diversas alternativas pudiendo destacar las siguientes:

Opciones de configuración (valor predeterminado)	Descripción
key = None	Archivo de clave SSL.
novncproxy_host = 0.0.0.0	Host en el que escuchar las peticiones entrantes.
novncproxy_port = 6080	Puerto en el que escuchar solicitudes entrantes.
record = False	Grabar sesiones en un fichero.
source_is_ipv6 = False	La fuente es ipv6.
ssl_only = False	No permitir conexiones no cifradas.
vnc_port = 5900	Puerto de inicio de VNC.
vnc_port_total = 10000	Puertos totales.
enabled = true	Habilita las funciones relacionadas con VNC.
novncproxy_base_url = <a href="http://127.0.0.1:6080/vnc_auto.html">http://127.0.0.1:6080/vnc_auto.html</a>	Ubicación del proxy de la consola VNC.
vncserver_listen = 127.0.0.1	Dirección IP en el cual la instancia debería escuchar.
vncserver_proxyclient_address = 127.0.0.1	La dirección a la que los clientes proxy deberían conectarse.
xvpncproxy_base_url = <a href="http://127.0.0.1:6081/console">http://127.0.0.1:6081/console</a>	Ubicación del proxy de la consola nova xvp VNC.

Tabla 4: Alternativas archivo "nova.conf"

La configuración que se realice deberá ser lo más adecuada a la situación en la que nos encontremos.

2º Escogemos una imagen con una interfaz gráfica. En esta ocasión y al contrario que en la prueba anterior, en la mayoría de sistemas no se encuentra habilitado el servicio VNC. Por ese motivo, deberemos instalar sobre dicha imagen el conjunto de paquetes que se requieren para poder disponer de un servidor VNC en ese sistema. A continuación, tendremos que configurar la imagen para que pueda lanzar automáticamente ese servicio.

Se recomienda realizar una comprobación de que la imagen efectivamente proporciona ese servicio, y que podemos acceder a la misma como un cliente VNC. Todo ello antes de subir la imagen a la plataforma OpenStack.

3º Teniendo disponible la imagen realizaremos los siguientes pasos:

- 3.1. Teniendo configurada la red por la prueba anterior, subimos el sistema como una nueva imagen en la plataforma OpenStack.
- 3.2. Creamos un nuevo volumen.
- 3.3. Añadimos una nueva regla al grupo de seguridad, habilitando el puerto que corresponde al servicio VNC

<input type="checkbox"/>	Dirección	Tipo Ethernet	Protocolo IP	Rango de puertos	Prefijo de IP Remota	Grupo de Seguridad Remoto	Acciones
<input type="checkbox"/>	Saliente	IPv6	Cualquier	Cualquier	:::0	-	Eliminar Regla
<input type="checkbox"/>	Saliente	IPv4	Cualquier	Cualquier	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	ICMP	Cualquier	0.0.0.0/0	-	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	TCP	22 (SSH)	0.0.0.0/0	-	Eliminar Regla
	<input type="checkbox"/>	Entrante	IPv4	TCP	5900	0.0.0.0/0	Eliminar Regla
<input type="checkbox"/>	Entrante	IPv4	UDP	53	0.0.0.0/0	-	Eliminar Regla

Ilustración 16: Procedimiento VNC - Nueva regla grupo de seguridad

- 3.4. Instanciamos la imagen con todo aquello que se requiere: recursos hardware, grupo de seguridad, claves, red interna a la que está asociada, imagen del sistema que queremos instanciar y el nombre con el que dotar a dicha instancia.
- 3.5. Asociar un volumen a la instancia y una dirección IP flotante.

<input type="checkbox"/>	Nombre de la instancia	Nombre de la imagen	Dirección IP	Sabor	Par de claves	Estado	Zona de Disponibilidad	Tarea	Estado de energía	Tiempo desde su creación	Acciones
<input type="checkbox"/>	Instancia_TinyCore	imagen_tinycore	20.0.0.8 IPs flotantes: 192.168.1.159	m1.tiny	cloudkey	Activo	nova	Ninguno	Ejecutando	2 días	Crear instantánea ▾

Ilustración 17: Procedimiento VNC – Instancia

3.6. Comprobar que la instancia es reconocida desde el exterior, por medio de un “ping”<sup>9</sup>.

```
C:\Users\ehu\QEMU>ping 192.168.1.159

Haciendo ping a 192.168.1.159 con 32 bytes de datos:
Respuesta desde 192.168.1.159: bytes=32 tiempo=7ms TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63

Estadísticas de ping para 192.168.1.159:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 7ms, Media = 1ms
```

Ilustración 18: Procedimiento VNC - Resultado reconocimiento instancia

4º Una vez tenemos disponible la instancia y es reconocida de manera correcta, deberemos de establecer la conexión por VNC. Para ello en primer lugar, creamos una nueva conexión referenciando la dirección IP de nuestra instancia en el cliente VNC de nuestro equipo local.

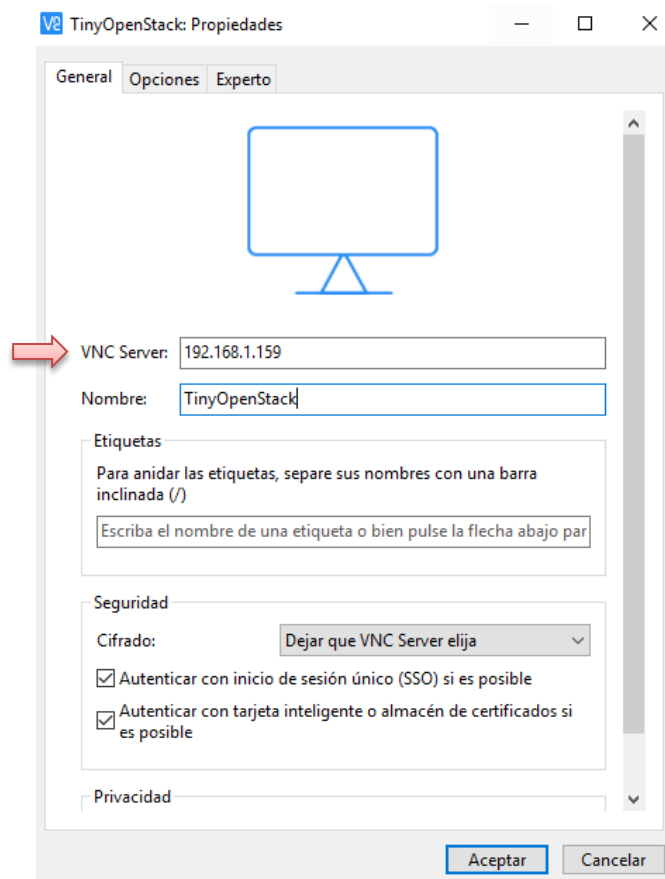


Ilustración 19: Procedimiento VNC - Nueva conexión VNC Client

5º Teniendo ya a nuestra disposición la instancia y definida la conexión deseada desde el cliente VNC, se inicia la sesión para comprobar que efectivamente la instancia es accesible remotamente con la aplicación VNC.



Ilustración 20: Procedimiento VNC - Conexión alcanzada

Mencionar que las ilustraciones visualizadas corresponden al resultado alcanzado en la prueba con un sistema Tiny Core. Por medio de la misma, podemos conocer más de cerca y de una manera más visual todo este procedimiento descrito. Para ello nos tenemos que dirigir al anexo G de esta memoria.



**7**



## OTROS ASPECTOS

## 7.1. LDAP

Como se ha mencionado a lo largo de esta memoria, el sistema de autenticación que hace uso la facultad para la validación de los estudiantes, profesorado entre otros, es LDAP; siendo uno de nuestros objetivos el buscar una plataforma de virtualización de escritorio que lo soporte.

Es por esa razón, que al final de esta documentación tendremos un anexo en donde se describe la forma de integrar la aplicación LDAP en la plataforma OpenStack, así como poder gestionarla por medio del programa JXplorer. Es simplemente una guía que se ha considerado oportuna realizar, pero que no se ha aplicado en este proyecto. La idea reside en hacer uso de ella en un futuro, comprobar su validez, y aplicarla y disponerla en nuestra plataforma OpenStack.

## 7.2. Coste

Por medio del siguiente apartado haremos mención a todo aquello que se debe de atender para poder poner en marcha la plataforma OpenStack en la Facultad de Informática UPV/EHU de San Sebastián/Donostia.

Lo primero que se tendrá que decidir es si la instalación de la plataforma la deseamos de manera separada o conjunta, es decir, disponer de todos los componentes de OpenStack en un mismo servidor o en diferentes. Para ello tendremos que elegir la distribución que más se adecue a las circunstancias.

El disponer de más o menos servidores puede conllevar ventajas así como inconvenientes. Entre las virtudes podemos destacar que podemos gestionar cada uno de los componentes de manera particular, atendiendo a cada una de sus necesidades; sin embargo, el tener que hacer uso de más de un servidor podrá provocar la necesidad de utilizar un mayor conjunto de recursos, todo ello dependiendo a que servicio vayan destinados. Por ejemplo, en el servidor que se establezca el servicio Nova, habrá que almacenar todas las máquinas virtuales que se pretenden proporcionar al conjunto de alumnado, profesorado o cualquier otra persona autorizada de la FISS.

Con todo ello, vemos la necesidad de disponer de un servidor o servidores con grandes capacidades atendiendo al conjunto de usuarios a los que va destinado y a la naturaleza del uso que se le haga. En el caso estudiado, para tener disponible una plataforma básica de pruebas hemos requerido de hasta 6 GB de memoria RAM así como un espacio en disco de 50 GB. Además, el procesador debe disponer de grandes pretensiones. Es por razón, que se si quiere implantar esta solución en la plataforma, la FISS debe proporcionar más recursos de los mencionados anteriormente.

Otro coste que se debe mencionar es el de la formación de las personas que vayan a administrar la plataforma. La mayor dificultad de OpenStack reside en como poder gestionarla de manera adecuada. Se requiere de algún curso de formación debido a que si no, el tiempo invertido en un estudio propio puede superar el tiempo planificado para la utilización e implantación de la plataforma. En este proyecto nos hemos centrado en una administración básica para verificar que OpenStack es una plataforma adecuada para poder proporcionar escritorios virtuales, pero aún quedan grandes aspectos los cuales no han sido resueltos en este Proyecto. Por ejemplo, un curso de formación en “virtualizadesdezero.com”, puede rondarte los 600 o 700 euros. Pero esto sería exclusivamente la preparación para el examen. El examen oficial tiene un coste de hasta 300 euros.



Es recomendable realizar primeramente una prueba sencilla similar a la que estaban realizando los técnicos de la FISS. Para ello podríamos, por ejemplo, seleccionar un conjunto reducido de alumnos. A cada uno de estos usuarios se le proporcionaría una cuenta en la plataforma OpenStack con un rol que limite sus acciones. A continuación, se les formaría para que conocieran la plataforma y pudieran hacerle uso en base a todo lo que les proporcionemos.



# 8

---

---

## CONCLUSIONES Y TRABAJO FUTURO

## 8.1. Valoración del resultado alcanzado

Tras finalizar el proyecto “Propuesta de implantación de escritorios virtuales” se puede afirmar que el resultado ha sido satisfactorio, ya que aunque los objetivos eran más ambiciosos, se ha llegado a una comprobación de que es posible por medio de las diferentes pruebas realizadas.

Justifiquemos la valoración anterior de este proyecto, en los objetivos y competencias alcanzadas:

- ❖ El primero de nuestros objetivos era buscar una solución con coste razonable (incluso nulo) en licencias. Lo hemos alcanzado por medio de la solución proporcionada por la compañía VirtualCable y por el proyecto OpenStack. Son dos soluciones que se distinguen en que una es un software propietario y la otra no, teniendo por lo tanto una diferencia en coste de uso; aunque ello depende de la distribución de OpenStack que queramos hacer uso.
- ❖ Otra meta era que la solución fuese escalable<sup>13</sup> a un gran número de alumnos. Según los casos de éxito estudiados, el software proporcionado por la compañía VirtualCable hubiera podido ser una gran opción. Sin embargo, no profundizamos en ello y vimos en OpenStack la plataforma requerida.
- ❖ La siguiente finalidad era que ofreciera un buen nivel de servicio, teniendo en cuenta la infraestructura de comunicaciones de la Facultad de Informática de San Sebastián (FISS) y el acceso remoto. Según las pruebas realizadas sobre la plataforma OpenStack, este propósito se ha cumplido, aun habiendo encontrado diferentes dificultades e incongruencias en la plataforma. Son inconvenientes relacionados con la falta de recursos o de formación en lo que respecta a entendimiento y comprensión de la plataforma, que han sido correctamente solventados por medio de distintas alternativas. Por otra parte, la solución propuesta por VirtualCable podría cumplir con esta meta a expensas de comprobarlo.
- ❖ La última de las intenciones era encontrar una resolución que soporte el sistema de autenticación en el que se fundamenta la Facultad, LDAP. Aunque no se ha puesto en marcha ni realizado posteriormente su comprobación, la plataforma OpenStack permite este sistema de autenticación. Lo mismo sucedía con el software de la compañía VirtualCable.

## 8.2. Valoración personal

El enfrentarse a las tecnologías requiere un esfuerzo constante de adaptación y de comprensión. En este caso, la búsqueda de soluciones de virtualización de escritorio ha resultado ser una tarea novedosa por todo lo que ella implica, en donde se ha necesitado de paciencia y ayuda de mis directores de proyecto. He aprendido y ampliado una gran cantidad de conceptos desde lo que es la virtualización a los diferentes tipos que existen, entre otros aspectos importantes; así como saber buscar alternativas ante las adversidades. He de reconocer que ha sido en las pruebas en donde más he ampliado mis capacidades, pudiendo enfrentarme a los problemas y conocer más de cerca lo que esconde la plataforma OpenStack.

Teniendo en cuenta los principales objetivos del proyecto, el tiempo invertido realizándolo y los resultados obtenidos, me encuentro bastante satisfecho con el devenir del proyecto.

## 8.3. Lecciones aprendidas

Durante este proyecto se han alcanzado diferentes lecciones que conviene mencionar y tener en cuenta en cualquier proyecto que se vaya a afrontar, tanto en el ámbito universitario como en un futuro dentro de una empresa.

- ❖ No menospreciar los riesgos. A la hora de afrontar un proyecto es importante tener en cuenta todos aquellos riesgos que se puedan identificar aunque la probabilidad sea mínima. Además, se intentará establecer estrategias para minimizarlos y evitar las posibles consecuencias.
- ❖ La formación es una parte importante. Durante este proyecto se tomó inicialmente con calma el tema de la formación, comprendiendo realmente cuales era el propósito de este proyecto. Sin embargo cuando nos decidimos a hacer uso de la plataforma OpenStack, la formación en lo que respecta a la administración no se realizó correctamente. Ello originó numerosas dificultades las cuales tuve que afrontar. Por lo tanto, aprender que en todo proyecto vale más una buena formación para poder alcanzar posteriormente todas y cada una de sus metas.

## 8.4. Posibles líneas de trabajo

Por medio de este apartado damos fin al proyecto, por lo que en el siguiente listado se recogerán las posibles propuestas de mejora. Todo ello con el fin de que en un futuro se dé continuidad al trabajo realizado.

- ❖ Hacer uso de la guía realizada para integrar LDAP sobre la plataforma OpenStack, conociendo de esta manera su funcionamiento.
- ❖ Hacer uso de la versión de pruebas del software de virtualización de escritorio facilitado por la compañía VirtualCable. Sobre él se podrían aplicar todas las pruebas realizadas en OpenStack. De esta forma además, se lograría realizar comparaciones entre esta solución y la proporcionada por el proyecto, pudiendo escoger entre una de ellas para la realización de pruebas más avanzadas.
- ❖ Crear un portal de autogestión en donde definamos las funcionalidades que deban tener cada clase de usuario que identifiquemos dentro de la plataforma. Para ello se podría realizar una primera prueba piloto con el sistema de identificación de OpenStack usando máquinas reales y la red de verdad, para luego, poder aplicarlo con el sistema de autenticación LDAP.

# 9



## BIBLIOGRAFÍA

## 9.1. Glosario

<sup>1</sup>AD: Active Directory (AD) o Directorio Administrativo son los términos que utiliza Microsoft para referirse a su implementación de servicio de directorio en una red distribuida de computadores. Utiliza distintos protocolos como LDAP, DNS, DHCP y Kerberos.

<sup>2</sup>Alojamiento compartido: El alojamiento compartido o alojamiento virtual es una de las modalidades más utilizadas por las empresas dedicadas al negocio del alojamiento web. Dependiendo de los recursos disponibles, permite tener una cantidad variable de dominios<sup>14</sup> y sitios web en una misma máquina.

<sup>3</sup>API: Interfaces de programación de aplicaciones (API: Application Programming Interfaces) es una especificación formal sobre cómo un módulo de un software se comunica o interactúa con otro.

En otras palabras, las API son un conjunto de comandos, funciones y protocolos informáticos que permiten a los desarrolladores crear programas específicos para ciertos sistemas operativos. Las API simplifican en gran medida el trabajo de un creador de programas, ya que no tiene que “escribir” códigos desde cero. Estas permiten al informático usar funciones predefinidas para interactuar con el sistema operativo<sup>27</sup> o con otro programa.

<sup>4</sup>API REST: Las APIs REST son un tipo de arquitecturas de desarrollo que utilizan el protocolo HTTP y son ampliamente utilizadas hoy en día en Internet para la extracción y modificación de datos automáticamente por aplicaciones web y móviles. Las APIs REST también se están convirtiendo en elementos cada vez más comunes en las arquitecturas de sistemas, ya que facilitan la programación de tareas y la conexión sencilla entre diferentes aplicaciones.

<sup>5</sup>BusyBox: En computación, BusyBox es un programa que combina muchas utilidades estándares de Unix en un solo ejecutable pequeño. Es capaz de proveer la mayoría de las utilidades que están especificadas para los sistemas Unix además de muchas de las utilidades que suelen verse en sistema GNU/Linux.

<sup>6</sup>Certificado X.509: En criptografía, X.509 es un estándar para infraestructuras de claves públicas. X.509 especifica, entre otras cosas, las formas estándar para el certificador de claves públicas y un algoritmo de validación de la ruta de certificación.

<sup>7</sup>Cluster: El término cluster se aplica a los conjuntos de computadoras construidos mediante la utilización de componentes de hardware comunes y que se comportan como si fuesen una única computadora.

<sup>8</sup>Código abierto: Es una expresión de la lengua inglesa que pertenece al ámbito de la informática. Se califica como open source a los programas informáticos que permiten el acceso a su código de programación, lo que facilita modificaciones por parte de otros programadores ajenos a los creadores originales del software en cuestión.

Es importante distinguir entre el software open source, que dispone de la mencionada característica de presentar su código abierto, y el software libre (que puede descargarse y distribuirse de manera gratuita). Existe software libre que no brinda acceso al código (y que, por lo tanto, no puede considerarse como open source), y programas open source que se distribuyen de manera comercial o que requieren de una autorización para ser modificados.



<sup>9</sup>Comando “Ping”: Este comando se utiliza para comprobar si una determinada interfaz de red, de nuestra computadora o de otra, se encuentra activa. El “Ping” envía paquetes a la dirección IP o a la dirección local que se le indique, y nos dice cuanto tiempo demoró el paquete en ir y regresar, entre otras informaciones.

<sup>10</sup>Control de acceso informático: El control de acceso informático o control de acceso a sistemas a informáticos, en seguridad informática, consiste en la autenticación, autorización de acceso y auditoría.

<sup>11</sup>Copyleft: Es el término que se utiliza en el ámbito informático (y se aplica de manera análoga a la creación literaria y artística) para designar el tipo de protección jurídica que confieren determinadas licencias que garantizan el derecho de cualquier usuario a utilizar, modificar y redistribuir un programa o sus derivados, siempre que se mantengan estas mismas condiciones de utilización y difusión.

<sup>12</sup>Eficiencia: Uso racional de los recursos con que se cuenta para alcanzar un objetivo predeterminado. A mayor eficiencia menor la cantidad de recursos que se emplearán, logrando mejor optimización<sup>22</sup> y rendimiento<sup>26</sup>.

Por tanto no confundir eficiencia con eficacia. La eficacia es la capacidad de lograr un efecto deseado, esperado o anhelado y en cambio, la eficiencia es la capacidad de lograr ese efecto en cuestión con el mínimo de recursos posibles o en el menor tiempo posible.

<sup>13</sup>Escalable: Se entiende por escalabilidad a la capacidad de adaptación y respuesta de un sistema con respecto a las circunstancias cambiantes.

<sup>14</sup>Dominio: Extensión o dominio de Internet es un nombre único que identifica a un sitio web en Internet.

<sup>15</sup>FLTK: Fast, Light Toolkit es una biblioteca de GUI multiplataforma. Se trata de un software libre licenciado bajo LGPL, que incluye una interfaz de diseño gráfico conocida como FLUID.

<sup>16</sup>Framework: Es una estructura conceptual y tecnológica de soporte definido, normalmente con módulos de software concretos, que puede servir de base para la organización y desarrollo de software. Es decir, es un esquema (un esqueleto, un patrón) para el desarrollo y/o la implementación de una aplicación.

<sup>17</sup>Hash: Las funciones de resumen o hash son algoritmos que consiguen crear a partir de una entrada (ya sea un texto, una contraseña o un archivo, por ejemplo), una salida alfanumérica de longitud normalmente fija que representa un resumen de un archivo o conjunto de archivos. Es decir, es un algoritmo matemático que transforma cualquier bloque arbitrario de datos en una nueva serie de caracteres con una longitud fija. Independientemente de la longitud de los datos de entrada, el valor hash de salida tendrá siempre la misma longitud. Entre sus objetivos esta asegurar la integridad de la información.

<sup>18</sup>IaaS (Infraestructure as a Service): La infraestructura como servicio (IaaS) le permite contar con acceso según demanda a la infraestructura de TI. Esto incluye recursos, como el almacenamiento, las redes y el procesamiento, que usted necesita para la ejecución de sus cargas de trabajo.

<sup>19</sup>Kernel-Based Virtual Machine (KVM): Es un hipervisor tipo 1 que proporciona una solución completa de virtualización para el sistema operativo<sup>27</sup> Linux en arquitecturas x86 que cuenten con las extensiones de virtualización Intel VT/ AMD-V.

<sup>20</sup>Metadatos: El concepto de metadatos se refiere a aquellos datos que describen otros datos, es decir, describen el contenido de los archivos o la información de los mismos, como el contenido, calidad, información y otras circunstancias o atributos.

<sup>21</sup>Middleware: Middleware o lógica de intercambio de información entre aplicaciones es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, o paquetes de programas, redes, hardware y/o sistemas operativos. Por ejemplo, hay una serie de productos de middleware que vinculan a un sistema de base de datos con un servidor web, permitiendo a los usuarios solicitar la información almacenada.

<sup>22</sup>Optimización: Desde el punto de vista informático, la optimización es la búsqueda y el hecho de mejorar el rendimiento<sup>26</sup> de un sistema operativo<sup>27</sup>, programa o dispositivo, a partir de determinados cambios lógicos (software) o físicos (hardware).

<sup>23</sup>OVS: Open vSwitch, abrevado OVS, es un software de código abierto<sup>8</sup>, diseñado para ser utilizado como un switch virtual en entornos de servidores virtualizados. Es el encargado de reenviar el tráfico entre diferentes máquinas virtuales (VMs) en la misma máquina física y también reenviar el tráfico entre las máquinas virtuales y la red física.

<sup>24</sup>Private Cloud/Nube Privada: La nube privada se define como los servicios informáticos que se ofrecen a través de Internet o de una red interna privada solo a algunos usuarios y no al público general.

<sup>25</sup>Puppet: Es una herramienta de gestión de la configuración de código abierto<sup>8</sup>. Su propósito es administrar la configuración de sistemas similares a Unix y a Microsoft Windows de forma declarativa. El usuario describe los recursos del sistema y sus estados utilizando el lenguaje declarativo que proporciona Puppet.

<sup>26</sup>Rendimiento: En informática, medida o cuantificación de la velocidad/resultado con que se realiza una tarea o proceso. En una computadora, su rendimiento no depende sólo del microprocesador como suele pensarse, sino de la suma de sus componentes como la memoria, el bus, los diversos dispositivos, etc; y su software.

<sup>27</sup>Sistema operativo: Es un programa (software) que después de arrancado o iniciado el ordenador se encarga de gestionar todos los recursos del sistema informático, tanto de hardware (partes físicas, disco duro, pantalla, teclado, etc.) como el software (programas e instrucciones); permitiendo así la comunicación entre el usuario y el ordenador.

<sup>28</sup>“Thin Client”: Un cliente liviano o cliente delgado o cliente ligero, es una computadora cliente o un software de cliente en una arquitectura de red cliente-servidor que depende primariamente del servidor central para las tareas de procesamiento, y se enfoca principalmente en transportar la entrada y la salida entre el usuario y el servidor remoto.

<sup>29</sup>TI: Las Tecnologías de Información son el medio por el cuál la información se recoge, administra, almacena, comunica, transforma, visualiza e interpreta. Las TI, además de apoyar los procesos de toma de decisiones, permiten automatizar procesos, monitorear el estado del negocio usando indicadores, aplicar estrategias competitivas, posicionar productos o marcas, identificar nuevas oportunidades de negocio y ganar flexibilidad para operar de manera efectiva.

<sup>30</sup>VLAN: Se trata de un concepto que se emplea en el terreno de la informática para nombrar el desarrollo de redes lógicas vinculadas a una única red de tipo físico. La utilidad de las VLAN radica en la posibilidad de separar aquellos segmentos lógicos que componen una LAN y que no tienen la necesidad de intercambiar información entre sí a través de la red de área local. Puede decirse entonces, que cada una de estas redes agrupa los equipos de un determinado segmento de red.

<sup>31</sup>VMware vSphere: VMware vSphere es una solución para virtualizar centro de datos. Básicamente permite virtualizar sistemas operativos y hardware mediante un servidor anfitrión, comúnmente denominado hipervisor o host, ejecutando una serie de máquinas virtuales que son las que contienen los sistemas operativos instalados. VMware vSphere facilita la administración total de todo nuestro entorno virtual y para ello consta de dos partes: VMware ESXi<sup>32</sup> (ESX en versiones antiguas) y VMware vCenter Server<sup>16</sup>.

<sup>32</sup>VMware ESXi: Es el sistema operativo<sup>27</sup> que permite correr otros sistemas operativos dentro de él. Es decir, es el sistema operativo<sup>27</sup> que lleva el servidor anfitrión o hipervisor, el servidor físico que va a ejecutar las máquinas virtuales.

<sup>33</sup>VMware vCenter Server: Es la herramienta que permite manejar todos nuestros ESXi<sup>32</sup> y nuestras máquinas virtuales de la manera más eficiente<sup>12</sup>.

<sup>34</sup>VPS: Un servidor virtual privado es un método de particionar un servidor físico en varios servidores de tal forma que todo funcione como si se estuviese ejecutando en una única máquina. Cada servidor virtual es capaz de funcionar bajo su propio sistema operativo<sup>27</sup> y además cada servidor puede ser reiniciado de forma independiente.

## 9.2. Referencias

Referencias que se han hecho uso para el concepto de virtualización:

1. <https://azure.microsoft.com/es-es/overview/what-is-virtualization/>
2. <https://www.redhat.com/es/topics/virtualization>
3. <https://es.wikipedia.org/wiki/Virtualizaci%C3%B3n>

Referencias utilizados para el concepto de Escritorio Virtual:

1. <http://www.intercity.cl/que-es-escritorio-virtual/>
2. <http://solutia-it.es/los-escritorios-virtuales-que-son-y-que-beneficios-aportan-a-organizaciones-y-particulares/>
3. [https://es.wikipedia.org/wiki/Virtualizaci%C3%B3n\\_de\\_escritorio](https://es.wikipedia.org/wiki/Virtualizaci%C3%B3n_de_escritorio)
4. [http://escritoriosvirtuales.blogspot.com/2013/05/iv-cuales-son-las-ventajas-y-desventajas\\_9.html](http://escritoriosvirtuales.blogspot.com/2013/05/iv-cuales-son-las-ventajas-y-desventajas_9.html)
5. <http://www.computerworld.es/archive/pros-y-contras-de-los-escritorios-virtuales>

Referencias a la Facultad de Informática de San Sebastián:

1. <https://www.emagister.com/ehu-facultad-informatica-cursos-61453-centrodetalles.htm>
2. [https://www.ehu.eus/documents/340468/1670338/Folleto+del+grado\\_cas](https://www.ehu.eus/documents/340468/1670338/Folleto+del+grado_cas)
3. <https://www.ehu.eus/es/web/informatika-fakultatea/informatika-ingenieritzako-gradua-iig->

En lo que se refiere al concepto aprendidos: de virtualización de aplicaciones:

1. <https://aprendizajeubicuo.wordpress.com/2011/05/19/virtualizacion-de-aplicaciones/> : Virtualización de aplicaciones.
2. [https://es.wikipedia.org/wiki/Cliente\\_liviano](https://es.wikipedia.org/wiki/Cliente_liviano) : Cliente liviano o cliente delgado.
3. <http://www.alegsa.com.ar/Dic/rendimiento.php> : Rendimiento.
4. <http://www.alegsa.com.ar/Dic/optimizar.php> : Optimización.
5. <http://www.alegsa.com.ar/Dic/eficiencia.php> : Eficiencia.
6. <https://virtualizadesdezero.com/que-es-vmware-vsphere/> : VMware vSphere.
7. <https://www.vmware.com/files/es/pdf/VMware-vSphere-Enterprise-Edition-Datasheet.pdf> : VMware vSphere.
8. [https://lac.citrix.com/geo/1\\_es\\_es.html](https://lac.citrix.com/geo/1_es_es.html) : Página oficial de Citrix Systems, Inc.
9. [http://demo.citrixlac.titan.developerplace.com/products/XenDesktop2\\_test.php?tab=overview](http://demo.citrixlac.titan.developerplace.com/products/XenDesktop2_test.php?tab=overview) : Citrix XenDesktop.
10. <http://demo.citrixlac.titan.developerplace.com/products/XenDesktop2.php> : Citrix XenDesktop.
11. <http://www.clusterinformatica.blogspot.com/2011/05/cluster-informatica.html> : Concepto de Cluster.

En lo que respecta al concepto de software como servicio (SaaS):

1. <https://azure.microsoft.com/es-es/overview/what-is-saas/> : Software como servicio (SaaS).
2. <http://www.flexibledesktop.com/iaas-y-daas-que-son/> : Servicio DaaS.
3. <http://searchdatacenter.techtarget.com/es/definicion/Escritorio-como-servicio-DaaS> : Servicio DaaS.

En lo que hace referencia a la solución VDI propuesta por VMware:

1. <https://www.vmware.com/es/solutions/virtualization.html> : Soluciones de virtualización de VMware.
2. <https://www.vmware.com/es/products/horizon.html> : VMware Horizon 7.
3. <https://www.vmware.com/es/products/vrealize-operations-horizon.html> : VMware vRealize Operations for Horizon.

En lo que respecta a la solución VDI de VirtualCable:

1. <http://www.virtualcable.net/quienes-somos/>: Página oficial de VirtualCable.
2. <https://www.udsenderprise.com/es/uds-enterprise/introduccion/>: Página oficial de UDS Enterprise.
3. <https://www.udsenderprise.com/es/blog/2018/02/19/arquitectura-vdi-con-uds-enterprise/> : Arquitectura de UDS Enterprise.

En lo que hace mención a los casos de éxito:

1. <https://www.vmware.com/es/company/customers.html#product=vmware-horizon&industry=education> : Casos de éxito de VMware Horizon en el sector educativo.
2. <https://www.udc.es/sobreUDC/> : Universidad de A Coruña.
3. <https://www.udsenderprise.com/es/uds-enterprise/casos-de-exito/> : Casos de éxito de UDS Enterprise.
4. <https://www.udsenderprise.com/es/blog/2016/10/24/universidad-de-murcia-acceso-aulas-informaticas-24/> : Caso de éxito en la Universidad de Murcia.
5. <http://www.um.es/> : Universidad de Murcia.
6. <https://www.udsenderprise.com/es/blog/2016/11/22/universidad-autonoma-de-madrid-33000-usuarios-vdi-/> : Caso de éxito en la Universidad Autónoma de Madrid.
7. <http://www.uam.es/UAM/Home.htm?language=es> : Universidad Autónoma de Madrid.
8. <https://www.udsenderprise.com/es/blog/2015/03/12/universidad-de-sevilla-escritorios-virtuales-con-o/> : Caso de éxito en la Universidad de Sevilla.
5. <http://www.us.es/> : Universidad de Sevilla.

En lo que respecta a información específica de OpenStack:

1. <https://www.openstack.org/software/>
2. <https://openwebinars.net/blog/que-es-eso-de-openstack-por-que-deberia-conocerlo/>
3. <https://es.wikipedia.org/wiki/OpenStack>
4. <https://virtualizadesdezero.com/que-es-openstack/>

En lo que hace mención a las diferentes compañías que proporcionan una distribución de OpenStack:

1. [https://es.wikipedia.org/wiki/Red\\_Hat](https://es.wikipedia.org/wiki/Red_Hat) : Multinacional RedHat.
2. <https://en.wikipedia.org/wiki/Mirantis> : Compañía Mirantis.
3. [https://es.wikipedia.org/wiki/Oracle\\_Corporation](https://es.wikipedia.org/wiki/Oracle_Corporation) : Organización Oracle.
4. <https://es.wikipedia.org/wiki/Ubuntu> : Ubuntu.

En lo que hace referencia a las distribuciones de OpenStack:

1. <https://virtualizadesdezero.com/distribuciones-openstack/>
2. <http://pubs.vmware.com/integrated-openstack-2/topic/com.vmware.ICbase/PDF/integrated-openstack-20-install-config-guide.pdf> : VMware Integrated Openstack.
3. <https://www.vmware.com/products/openstack.html> : VMware Integrated OpenStack.
4. <https://www.redhat.com/es/technologies/linux-platforms/openstack-platform> : Red Hat OpenStack Platform.
5. <https://www.rdoproject.org/> : RDO Project.
6. <https://www.mirantis.com/software/openstack/> : Mirantis OpenStack.
7. <https://www.oracle.com/es/linux/openstack/index.html> : Oracle OpenStack.
8. <https://www.ubuntu.com/openstack> : Ubuntu OpenStack.
9. <http://www.sctech.es/index.php/ubuntu/> : Ubuntu OpenStack.
10. <http://www.brianlinkletter.com/openstack-on-one-machine/> : OpenStack DevStack.
11. <https://docs.openstack.org/devstack/latest/#all-in-one-single-vm> : OpenStack DevStack.

En lo que hace referencia a la solución de virtualización de escritorio de Microsoft:

1. <https://www.microsoft.com/es-es/cloud-platform/desktop-virtualization> : Solución VDI de Microsoft.
2. <https://blogs.technet.microsoft.com/latam/2017/07/20/arquitectura-de-remote-desktop-services-rds-y-sus-roles-principales/> : RDS.

En lo que hace mención a la organización Citrix:

1. [https://lac.citrix.com/geo/1\\_es\\_es.html](https://lac.citrix.com/geo/1_es_es.html) : Página oficial de Citrix Systems, Inc.
2. [http://demo.citrixlac.titan.developerplace.com/products/XenDesktop2\\_test.php?tab=overview](http://demo.citrixlac.titan.developerplace.com/products/XenDesktop2_test.php?tab=overview) : Citrix XenDesktop.
3. <http://demo.citrixlac.titan.developerplace.com/products/XenDesktop2.php> : Citrix XenDesktop.

En lo que respecta a la compañía Flexible Desktop:

1. <http://www.flexibledesktop.com/> : Página oficial de Flexible Desktop.
- En lo que se refiere a la solución VDI de flexVDI:
2. <https://docs.flexvdi.com/display/V30/Introduction+to+flexVDI> : flexVDI.

En lo que hace mención a la solución VDI propuesta por la compañía Qindel Group:

1. <http://www.qindel.com/grupo-qindel/> : Página oficial de Qindel Group.
2. <http://www.theqvd.com/es/> : Página oficial de QVD.

En lo que respecta a la resolución VDI de la organización OVH:

1. <https://www.ovh.es/cloud/cloud-desktop/> : Página oficial de OVH – Innovation is Freedom

En lo que hace referencia a la solución VDI propuesta por el proveedor Flexible IT:

1. <https://www.flexible.com/flexible-it/> : Página oficial de la compañía Flexible IT.
2. <http://www.flexibledesktop.com/> : Página oficial de Flexible Desktop VDI.

En lo que se refiere a la propuesta de la empresa Nologin:

1. <http://www.nologin.es/es/index> : Página oficial de Nologin.

En lo que hace mención a la solución de Solutia IT:

1. <http://solutia-it.es/> : Página oficial de Solutia IT.
2. <http://evaos.es/>: Página oficial de evaOS.

En lo que respecta a la solución VDI propuesta por la empresa Huawei:

1. <http://www.huawei.com/es/> : Página oficial de Huawei.
2. <http://e.huawei.com/es/solutions/technical/Cloud-Computing/desktop-cloud> : Solución VDI de Huawei.
3. <http://powernet.es/web/blog/virtualizacion-y-almacenamiento/solucion-de-virtualizacion-huawei-fusioncloud/> : Explicación solución Huawei.

En lo que hace referencia a la solución VDI que ofrece Telefónica:

1. <https://www.telefonica.com/es/home> : Página oficial de Telefónica.
2. <http://www.movistar.es/grandes-empresas/soluciones/fichas/virtual-desktop/#> : Solución VDI de telefónica.

Aspectos para conocer la forma de llevar a cabo la administración de la plataforma OpenStack:

1. <http://iesgn.github.io/emergya/curso/u2/> : Curso de Administración de OpenStack.
2. <https://www.dbigcloud.com/cloud-computing/170-openstack-desde-cero-keystone.html> : Administración de OpenStack – Servicio Keystone.
4. <https://www.dbigcloud.com/cloud-computing/171-openstack-desde-cero-glance.html> : Administración de OpenStack – Servicio Glance.
5. <https://www.dbigcloud.com/cloud-computing/173-openstack-desde-cero-nova.html> : Administración de OpenStack – Servicio Nova.
6. <https://searchstorage.techtarget.com/definition/OpenStack-Swift> : Administración de OpenStack – Servicio Swift.
7. <https://translate.google.es/translate?hl=es&sl=en&u=https://www.networkcomputing.com/data-centers/openstack-neutron-components-and-concepts/1828660212&prev=search> : Aspectos importante del servicio Neutron.
8. [https://translate.googleusercontent.com/translate\\_c?depth=1&hl=es&prev=search&rurl=translate.google.es&sl=en&sp=nmt4&u=https://www.networkcomputing.com/data-centers/openstack-neutron-components-and-concepts/1828660212/page/0/1&xid=25657,15700021,15700124,15700149,15700168,15700186,15700201&usg=ALkJrhiwGpUC3Q\\_ripKHZQopBfHwr0iL-w](https://translate.googleusercontent.com/translate_c?depth=1&hl=es&prev=search&rurl=translate.google.es&sl=en&sp=nmt4&u=https://www.networkcomputing.com/data-centers/openstack-neutron-components-and-concepts/1828660212/page/0/1&xid=25657,15700021,15700124,15700149,15700168,15700186,15700201&usg=ALkJrhiwGpUC3Q_ripKHZQopBfHwr0iL-w) : Aspectos importantes del servicio Neutron.
9. [https://docs.oracle.com/cd/E69400\\_01/html/E74914/app-ostckcommands.html](https://docs.oracle.com/cd/E69400_01/html/E74914/app-ostckcommands.html) : OpenStackClient.

Referencia en lo que respecta al estudio realizado sobre LDAP:

1. <http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ldap.html>
2. [https://es.wikipedia.org/wiki/Protocolo\\_Ligero\\_de\\_Acceso\\_a\\_Directorios](https://es.wikipedia.org/wiki/Protocolo_Ligero_de_Acceso_a_Directorios)
3. <https://www.youtube.com/watch?v=di5hZZ8Ty8g>

Reflejemos los diferentes enlaces consultados para establecer la conexión remota por SSH:

1. <https://virtualizadesdezero.com/roles-en-openstack/> : Administración de roles en OpenStack.
2. <https://www.enmimaquinafunciona.com/pregunta/90056/como-reiniciar-servicios-de-openstack-despues-de-reiniciar-cierre-de-sesion> : Reinicio de servicios en OpenStack.
3. <https://ask.openstack.org/en/question/87906/i-get-no-route-to-host-when-i-ssh-my-cirros-instance/> : Conexión remota.
4. <https://rwmj.wordpress.com/2013/08/02/new-in-virt-sysprep-set-root-and-user-passwords/> : Servicio "virt-sysprep".
5. <https://github.com/naturalis/openstack-docs/wiki/Howto:-Creating-and-using-OpenStack-SSH-keypairs-on-Linux-and-OSX> : Uso de claves privadas en el servicio Horizon de OpenStack.
6. <https://stackoverflow.com/questions/20162070/unable-to-ping-to-an-instance-launched-in-openstack> : Problemas ping.
7. <https://docs.openstack.org/horizon/latest/user/configure-access-and-security-for-instances.html> : Conexión SSH con OpenStack.
8. [https://wincp.net/eng/docs/ui\\_puttygen](https://wincp.net/eng/docs/ui_puttygen) : Aplicación PuTTYgen.
9. <https://ask.openstack.org/en/question/53402/unable-to-access-the-nova-instate/> : Clave rechazada.
10. [https://docs.openstack.org/mitaka/user-guide/cli\\_nova\\_configure\\_access\\_security\\_for\\_instances.html](https://docs.openstack.org/mitaka/user-guide/cli_nova_configure_access_security_for_instances.html) : Configuración de grupos de seguridad en OpenStack.
11. <https://www.youtube.com/watch?v=cgY0v6c3Egg> : Configuración de red en OpenStack.
12. <https://www.youtube.com/watch?v=qIKVuTi7TzA> : Configuración de claves y conexión SSH en OpenStack.
13. [http://fpg.x10host.com/VirtualBox/modo\\_adaptador\\_puente.html](http://fpg.x10host.com/VirtualBox/modo_adaptador_puente.html) : Modo adaptador puente.
14. [http://fpg.x10host.com/VirtualBox/modo\\_sloanfitrin.html](http://fpg.x10host.com/VirtualBox/modo_sloanfitrin.html) : Modo solo anfitrión.

Hagamos alusión a todos los enlaces estudiados para comprender el concepto de VNC:

1. <https://es.wikipedia.org/wiki/VNC>
2. <https://www.realvnc.com/es/connect/docs/faq/function.html>
3. <http://www.ordenadores-y-portatiles.com/vnc.html>

Enlaces asociados al estudio realizado sobre Tiny Core Linux:

1. [https://es.wikipedia.org/wiki/Tiny\\_Core\\_Linux](https://es.wikipedia.org/wiki/Tiny_Core_Linux)
2. [https://www.ecured.cu/Tiny\\_Core\\_Linux](https://www.ecured.cu/Tiny_Core_Linux)

Realicemos una observación de los enlaces utilizados para conocer la distribución Kali Linux:

1. <https://blog.desdelinux.net/kali-linux-suite-seguridad-informatica/>
2. <https://computerhoy.com/paso-a-paso/software/que-es-kali-linux-que-puedes-hacer-41671>

Mención a las páginas visitas para comprender el funcionamiento de VNC en OpenStack:

1. <https://docs.openstack.org/nova/pike/admin/remote-console-access.html>



Enlaces asociados al estudio de la herramienta VBoxManage:

1. [http://www.tecgroup.es/tg/documentos/comandos\\_vboxmanage/comandos\\_vboxmanage.pdf](http://www.tecgroup.es/tg/documentos/comandos_vboxmanage/comandos_vboxmanage.pdf)
2. <https://www.virtualbox.org/manual/ch08.html>
3. <http://futuregrid.github.io/manual/os-virtualbox.html>

### 9.3. Vídeos

Por medio de este apartado daremos a conocer el conjunto de vídeos que se han visto necesarios consultar para mejorar el entendimiento y comprensión de cada uno de los conceptos estudiados.

1. [https://www.youtube.com/watch?v=n628Y2NYcXQ&utm\\_source=ActiveCampaign&utm\\_medium=email&utm\\_content=%C2%BFQu%C3%A9+tal+ha+ido+la+instalaci%C3%B3n+de+Openstack+%3F&utm\\_campaign=Ebook+2%C2%BA+d%C3%ADa](https://www.youtube.com/watch?v=n628Y2NYcXQ&utm_source=ActiveCampaign&utm_medium=email&utm_content=%C2%BFQu%C3%A9+tal+ha+ido+la+instalaci%C3%B3n+de+Openstack+%3F&utm_campaign=Ebook+2%C2%BA+d%C3%ADa) : Definición de OpenStack.
2. <https://www.youtube.com/watch?v=WYKwibmL7EO> : Gestión de usuarios y grupos Keystone por la interfaz web.
3. <https://www.youtube.com/watch?v=593Sy7RoEc8> : Gestión de usuarios y grupos Keystone por la línea de comandos.
4. [https://www.youtube.com/watch?v=Q\\_028WYUXGs](https://www.youtube.com/watch?v=Q_028WYUXGs) : Servicio de imágenes Glance.
5. <https://www.youtube.com/watch?v=iSpOCqJgar0> : Servicio de cómputo Nova.
6. <https://www.youtube.com/watch?v=YBLi1udhCRA&list=PL83DWCbd4CqVuPSwIJrVw8a4Ux9A1xhPL&index=8> : Servicio de red Neutron.



*10*

---

---

APÉNDICE

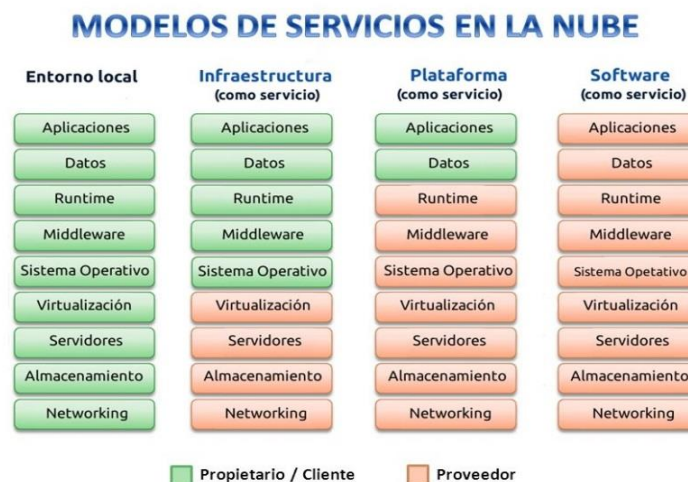
## A. Cloud computing

El término “Cloud Computing” es un término bastante amplio cuyas características esenciales según el Instituto Nacional de Estándares y Tecnología (NIST) de EE.UU. son las siguientes:

- ❖ Servicio disponible de forma automática y a demanda: Un usuario puede comenzar a utilizar un recurso en la nube (una instancia, almacenamiento, etc.) sin ninguna intervención por parte de un operador de la empresa que presta el servicio.
- ❖ Acceso a través de la red: Los recursos están disponibles a través de la red (Internet u otro tipo de red pública o privada), mediante mecanismos estandarizados que permitan el uso de clientes diversos, desde teléfonos a grandes ordenadores.
- ❖ Los recursos son compartidos en general por múltiples clientes, que pueden disponer de ellos a demanda y a los que se les debe garantizar aislamiento y seguridad.
- ❖ Elasticidad: Hay que tener en consideración que los recursos del usuario pueden crecer y decrecer de forma rápida en función de sus necesidades.
- ❖ Pago por uso: La utilización real que hace el usuario de los recursos y no una tarificación por tramos es lo que define el pago de los servicios de cloud.

Los servicios de Cloud Computing se pueden clasificar según el control sobre los recursos contratos o modelos de servicio:

- ❖ Software as a Service (SaaS): Aplicación completa ofrecida como servicio en la nube. Cualquier usuario hace uso de un determinado software de una empresa a través de Internet. Por ejemplo: Servicios de Google, Microsoft Office 365, etc.
- ❖ Plataforma as a Service (PaaS): Aplicación completa para el desarrollo y despliegue de software. Los desarrolladores de software pueden optar por utilizar una plataforma en la nube para sus desarrollos, facilitándoles mucho las tareas de prueba y despliegue. Algunas de las opciones más conocidas son: Google App Engine, Windows Azure, etc.
- ❖ Infraestructura as a Service (IaaS)<sup>18</sup>: Principalmente almacenamiento y capacidades de cómputo (máquinas virtuales) ofrecidos como servicios en la nube. Los administradores de sistemas pueden plantearse montar un servicio de máquina física, una máquina virtual o una instancia en la nube. Los servicios de cloud IaaS<sup>18</sup> más conocidos son: Amazon Web Services, RackSpace Cloud, OpenStack, etc.



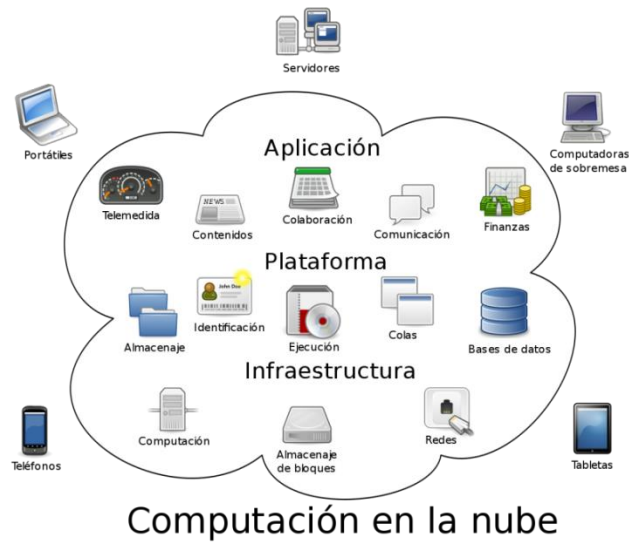
11

Ilustración 21: Modelos de servicio en la nube

<sup>11</sup> <http://slideplayer.es/slide/5421707/17/images/8/MODELOS+DE+SERVICIOS+EN+LA+NUBE.jpg>

Se puede también distinguir el ámbito de uso de Cloud Computing:

- ❖ Público: Una empresa ofrece servicios a terceros, encargándose de toda la gestión del Cloud.
- ❖ Privado: Una organización configura sus propios recursos de forma mucho más flexible en una nube.
- ❖ Híbrido: Algunos servicios se gestionan en la nube privada<sup>24</sup> y otros se transfieren en una pública; o incluso haciendo uso de la elasticidad que ofrece se pasan recursos de la nube privada<sup>24</sup> a la pública en función de las necesidades.



12

Ilustración 22: Cloud Computing

<sup>12</sup>[https://upload.wikimedia.org/wikipedia/commons/thumb/f/ff/Cloud\\_computing-es.svg/300px-Cloud\\_computing-es.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/f/ff/Cloud_computing-es.svg/300px-Cloud_computing-es.svg.png)

## B. Planificación del tiempo

### B.1. Introducción

Por medio de este anexo, se pretende reflejar la planificación que se ha estimado oportuna para este proyecto teniendo en consideración el plazo de 300 horas para la elaboración del trabajo. Las reuniones se llevarán a cabo los viernes teniendo que tener presentes las festividades y otras causas que puedan justificar la no realización de la misma. Además, se ha estimado conveniente dedicar entre 3 y 5 horas diarias, a excepción de los fines de semana, para el progreso del proyecto. Ello puede variar por problemas e inconvenientes que se puedan encontrar. El proyecto se espera tenerlo finalizada a principios del mes de mayo pudiendo de esta manera presentarlo en julio, aunque se prevé que dependiendo de las soluciones que se encuentren y las posibles dificultades, pueda ampliarse ese plazo.

Para el correcto entendimiento de los días de los que se compone el mes, se ha estimado oportuno llevar a cabo la siguiente organización:

- ❖ Si la celda esta sombreada de verde, quiere decir que es una festividad.
- ❖ Si la celda esta sombreada de gris, nos encontramos en un día de fin de semana.
- ❖ Si la celda esta sombreada de color marrón, son fechas importantes del proyecto.
- ❖ En su defecto, son días para la realización de cualquier tarea.

Por lo tanto, es una estimación inicial abierta a cualquier alteración de nuestro proyecto.

## B.2. Estimación de tiempo

### B.2.1. Mes de enero

	Enero 2018																															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
<b>T1 - Gestión</b>																																
T1.1 - Planificación																																
T1.2 - Reunión																																
T1.3 - Seguimiento y Control																																
<b>T2 - Comienzo</b>																																
T2.1 - Formación																																
<b>T3 - Desarrollo</b>																																
T3.1 - Estudio de alternativas																																
T3.2 - Tratamiento de Alternativas																																
T3.2.1 - Implementación																																
T3.2.2 - Simulación y Rendimiento																																
T3.2.2.1 - Pruebas																																
T3.2.2.2 - Estudio																																
<b>T4 - Documentación</b>																																
T4.1 - Memoria																																
T4.2 - Material para la defensa																																

Tabla 5: Mes de enero

B.2.2. Mes de febrero

	Febrero 2018																											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
<b>T1 - Gestión</b>																												
T1.1 - Planificación																												
T1.2 - Reunión																												
T1.3 - Seguimiento y Control																												
<b>T2 - Comienzo</b>																												
T2.1 - Formación																												
<b>T3 - Desarrollo</b>																												
T3.1 - Estudio de alternativas																												
T3.2 - Tratamiento de Alternativas																												
T3.2.1 - Implementación																												
T3.2.2 - Simulación y Rendimiento																												
T3.2.2.1 - Pruebas																												
T3.2.2.2 - Estudio																												
<b>T4 - Documentación</b>																												
T4.1 - Memoria																												
T4.2 - Material para la defensa																												

Tabla 6: Mes de febrero



B.2.3. Mes de marzo

	Marzo 2018																															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
<b>T1 - Gestión</b>																																
T1.1 - Planificación																																
T1.2 - Reunión																																
T1.3 - Seguimiento y Control																																
<b>T2 - Comienzo</b>																																
T2.1 - Formación																																
<b>T3 - Desarrollo</b>																																
T3.1 - Estudio de alternativas																																
T3.2 - Tratamiento de Alternativas																																
T3.2.1 - Implementación																																
T3.2.2 - Simulación y Rendimiento																																
T3.2.2.1 - Pruebas																																
T3.2.2.2 - Estudio																																
<b>T4 - Documentación</b>																																
T4.1 - Memoria																																
T4.2 - Material para la defensa																																

Tabla 7: Mes de marzo

B.2.4. Mes de abril

	Abril 2018																													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
<b>T1 - Gestión</b>																														
T1.1 - Planificación																														
T1.2 - Reunión																														
T1.3 - Seguimiento y Control																														
<b>T2 - Comienzo</b>																														
T2.1 - Formación																														
<b>T3 - Desarrollo</b>																														
T3.1 - Estudio de alternativas																														
T3.2 - Tratamiento de Alternativas																														
T3.2.1 - Implementación																														
T3.2.2 - Simulación y Rendimiento																														
T3.2.2.1 - Pruebas																														
T3.2.2.2 - Estudio																														
<b>T4 - Documentación</b>																														
T4.1 - Memoria																														
T4.2 - Material para la defensa																														

Tabla 8: Mes de abril

B.2.5. Mes de mayo

	Mayo 2018																															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
<b>T1 - Gestión</b>																																
T1.1 - Planificación																																
T1.2 - Reunión																																
T1.3 - Seguimiento y Control																																
<b>T2 - Comienzo</b>																																
T2.1 - Formación																																
<b>T3 - Desarrollo</b>																																
T3.1 - Estudio de alternativas																																
T3.2 - Tratamiento de Alternativas																																
T3.2.1 - Implementación																																
T3.2.2 - Simulación y Rendimiento																																
T3.2.2.1 - Pruebas																																
T3.2.2.2 - Estudio																																
<b>T4 - Documentación</b>																																
T4.1 - Memoria																																
T4.2 - Material para la defensa																																

Tabla 9: Mes de mayo

B.2.6. Mes de junio

	Junio 2018																													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
<b>T1 - Gestión</b>																														
T1.1 - Planificación																														
T1.2 - Reunión																														
T1.3 - Seguimiento y Control																														
<b>T2 - Comienzo</b>																														
T2.1 - Formación																														
<b>T3 - Desarrollo</b>																														
T3.1 - Estudio de alternativas																														
T3.2 - Tratamiento de Alternativas																														
T3.2.1 - Implementación																														
T3.2.2 - Simulación y Rendimiento																														
T3.2.2.1 - Pruebas																														
T3.2.2.2 - Estudio																														
<b>T4 - Documentación</b>																														
T4.1 - Memoria																														
T4.2 - Material para la defensa																														

Tabla 10: Mes de junio

B.2.7. Mes de julio

	Julio 2018																															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
<b>T1 - Gestión</b>																																
T1.1 - Planificación																																
T1.2 - Reunión																																
T1.3 - Seguimiento y Control																																
<b>T2 - Comienzo</b>																																
T2.1 - Formación																																
<b>T3 - Desarrollo</b>																																
T3.1 - Estudio de alternativas																																
T3.2 - Tratamiento de Alternativas																																
T3.2.1 - Implementación																																
T3.2.2 - Simulación y Rendimiento																																
T3.2.2.1 - Pruebas																																
T3.2.2.2 - Estudio																																
<b>T4 - Documentación</b>																																
T4.1 - Memoria																																
T4.2 - Material para la defensa																																

Tabla 11: Mes de julio

## C. Soluciones de virtualización

### C.1. Introducción

Como se ha comentado en el apartado 4 de esta memoria, existen diferentes alternativas a la hora de proporcionar una solución de escritorio virtual. Por medio del siguiente apartado, mencionaremos otras posibles soluciones de virtualización de escritorio que se han estudiado a lo largo de este proyecto. Para ello, distinguiremos entre las organizaciones que presentan soluciones VDI y las empresas que comercializan con éstas.

### C.2. Soluciones VDI

#### C.2.1. XenDesktop - Citrix

Citrix Systems, Inc. es una corporación multinacional fundada en 1989, que suministra tecnologías de virtualización de servidores, conexión en red, software como servicio (SaaS) e informática en la nube.

La tecnología de virtualización de aplicaciones de Citrix pretende aislar las aplicaciones del sistema operativo<sup>27</sup> subyacente y de otras aplicaciones para aumentar la compatibilidad y la manejabilidad. Citrix promueve dos soluciones para la virtualización de escritorio y aplicaciones de forma segura:

- ❖ XenApp
- ❖ XenDesktop

XenApp es un sistema completo para la entrega de aplicaciones virtuales, que ofrece acceso a las aplicaciones, ya sea con o sin conexión, a través de una combinación de alojamiento y transmisión de aplicaciones directamente a los dispositivos de los usuarios. Cuando los usuarios solicitan una aplicación, XenApp determina si su dispositivo es compatible y capaz de ejecutar la aplicación en cuestión. Los requisitos mínimos de un dispositivo de destino son:

- ❖ Un sistema operativo<sup>27</sup> Windows.
- ❖ Un software cliente Citrix adecuado.

Si el dispositivo del usuario cumple con los requisitos mínimos, XenApp inicia la virtualización de la aplicación directamente en un entorno aislado en el dispositivo del usuario. En el caso de que el dispositivo del usuario no pueda ejecutar una aplicación en particular, XenApp inicia la virtualización de sesiones. Mientras que la virtualización de aplicaciones se limita a los sistemas operativos basados en Windows, la virtualización de sesiones a través de XenApp debe permitir que cualquier usuario de cualquier sistema operativo<sup>27</sup> acceda a cualquier aplicación entregada por TI<sup>9</sup>. Como resultado, XenApp habilita los dispositivos de Windows, Mac, Linux, iOS y Android para ejecutar cualquier aplicación mediante la virtualización de sesiones. Tanto en la virtualización de aplicaciones como en la virtualización de sesiones, Citrix pretende que la interacción del usuario con la aplicación sea lo más íntegra posible.

XenApp es por tanto una solución de virtualización de aplicaciones que ayuda a optimizar<sup>22</sup> la productividad con acceso universal a aplicaciones, escritorios y datos virtuales desde cualquier

dispositivo. XenDesktop ofrece la misma funcionalidad que XenApp, más la opción de implementar una solución VDI que sea escalable<sup>13</sup>.

XenDesktop se basa en las siguientes tecnologías:

- ❖ Citrix Receiver: Con esta herramienta Citrix pretende que cualquier PC, Mac, smartphone, tableta entre otros obtengan acceso a los escritorios y a las aplicaciones corporativas.
- ❖ Tecnología Citrix FlexCast: Citrix hace uso de este mecanismo para otorgar una única solución de escritorio virtual para cada tipo de trabajador de la empresa que se encuentre en línea o sin conexión, estandarizado o personalizado.
- ❖ Tecnología Citrix HDX: Con esta tecnología Citrix intenta brindar una experiencia de usuario que se equipara a la experiencia con un PC local.

### C.2.2. flexVDI

Se trata de una solución completa y abierta de VDI que ofrece la libertad de trabajar desde cualquier parte.

Entre sus características destacar que se trata de una solución de virtualización de escritorio basada en código abierto<sup>8</sup> haciendo uso del protocolo SPICE, y utilizando KVM<sup>19</sup> con una serie de modificaciones para mejorar su rendimiento<sup>26</sup> con cargas VDI, y dos sistemas de autenticación: LDAP y AD<sup>1</sup>.

La plataforma flexVDI consiste en un conjunto de componentes hardware y software que iteran entre ellos. Desde el punto de vista del hardware, deben existir:

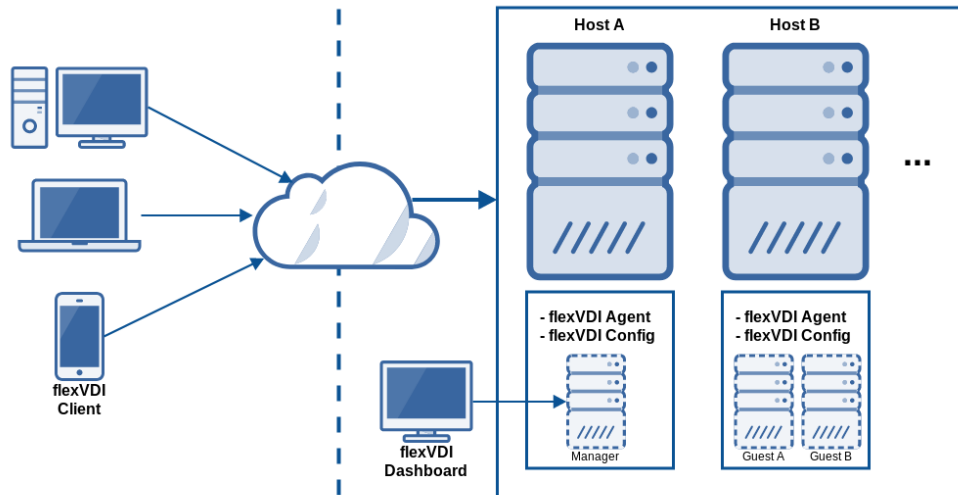
- ❖ Uno o más nodos computacionales, llamados "Hosts". Serán los responsables de proporcionar los recursos de CPU y RAM necesarios para poder hacer uso de los escritorios virtuales.
- ❖ Un espacio de almacenamiento donde alojaremos cada uno de los escritorios virtuales.

Por otra parte, los componentes software básicos serán los siguientes:

- ❖ "flexVDI Manager": Será la representación de la máquina virtual que quiere hacer uso el cliente en un instante determinado y que se encontrará alojada en uno de los "Host".
- ❖ "flexVDI Agent": Un servicio que se ejecuta en cada "Host" que gestiona los recursos locales de la instancia "flexVDI Manager".
- ❖ "flexVDI Config": Herramienta disponible en todos los "Host" que puede ser usada para configurar cada uno de los nodos computacionales o las instancias de "flexVDI Manager".
- ❖ "flexVDI Dashboard": Aplicación que permite a los administradores de la plataforma conectar y administrar una o más instancias "flexVDI Manager".
- ❖ "flexVDI Client": Herramienta de presentación que permite a los usuarios finales acceder a su escritorio virtual.

- ❖ “flexVDI Guests”: Hará mención a la agrupación de todas y cada una de las máquinas virtuales por cada uno de los nodos computacionales. Cada una se encontrará definida con una configuración determinada.

De una manera visual:



13

Ilustración 23: Arquitectura flexVDI

Adicionalmente, flexVDI proporciona los siguientes componentes:

- ❖ “flexVDI Gateway”: Punto de entrada de la plataforma “flexVDI Client”.
- ❖ “flexVDI WebPortal”: Proporciona acceso a los escritorios virtuales desde navegadores web donde no es posible instalar un cliente nativo.
- ❖ “flexVDI Guest Tools”: Conjunto de herramientas que proporcionan funcionalidades adicionales a cada uno de los escritorios virtuales que se haga uso.

Finalmente cabe mencionar que flexVDI tiene un modelo de licenciamiento por uso. Hay dos tipos de licencias:

- ❖ Licencia perpetua: Se paga una única vez por cada usuario.
- ❖ Licencia suscripción: Se paga cada año por cada usuario.

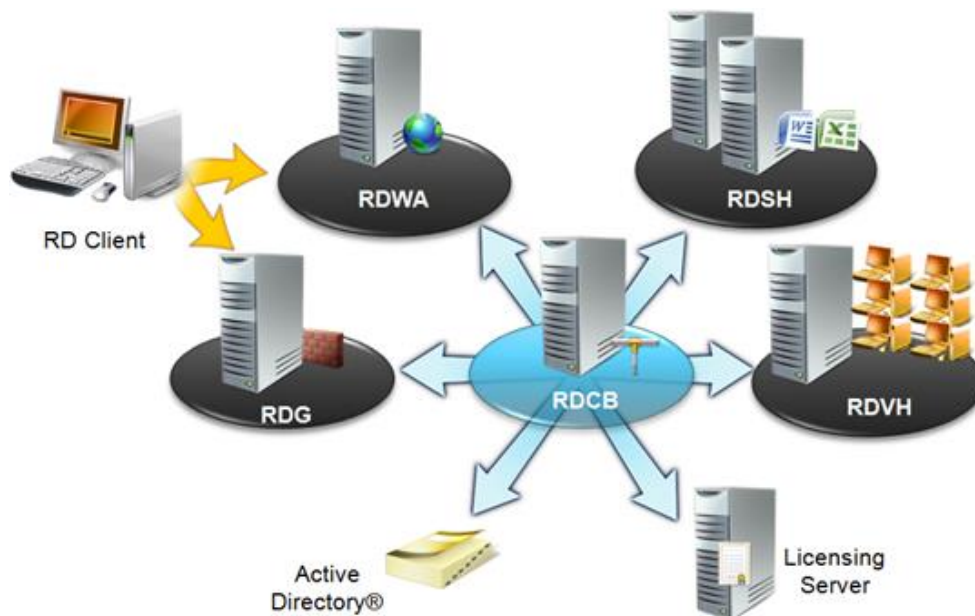


### C.2.3. Microsoft VDI

Microsoft VDI permite ejecutar aplicaciones y escritorios de Windows en cualquier dispositivo. Para ello propone:

- ❖ Posibilidad de elegir entre distintos modelos de implementación para satisfacer todas las posibles necesidades: de forma local o en Azure.
- ❖ Entregar aplicaciones de Windows en cualquier dispositivo: Windows, Mac, iOS, Android.
- ❖ Garantizar la conformidad de los datos.

La columna vertebral de las soluciones de infraestructura de escritorios virtuales (VDI) de Microsoft es el Servicio de Escritorio Remoto (RDS del inglés Remote Desktop Services). RDS es una plataforma para la construcción de soluciones de virtualización para todas las necesidades de los clientes, incluyendo entrega de aplicaciones virtualizadas, acceso seguro móvil y escritorio remoto; y dando la capacidad a los usuarios finales de utilizar sus aplicaciones y escritorios desde la nube.



14

Ilustración 24: Microsoft VDI - Estructura RDS

## C.3. Comercialización VDI

### C.3.1. Qindel Group

Qindel Group es una compañía internacional de consultoría con fuerte especialización en proyectos, entornos Linux y tecnologías de la información. Como la virtualización de escritorios se está convirtiendo cada vez en algo más común, y el mercado se encuentra dominado por VMware y Citrix, Qindel ofrece una alternativa VDI orientada a Linux de nombre QVD. Según Qindel Group: “QVD ofrece la alternativa VDI con mayor densidad de escritorios del mercado, capaz de ofrecer a una gran concentración de usuarios a nivel comercial una infraestructura de escritorios virtuales”.

El objetivo que propone Qindel Group con QVD es proporcionar un método simple para la gestión de entornos de escritorio Linux en toda la empresa, reduciendo los costos de hardware y de licencias y al mismo tiempo facilitando la administración. La forma de ofrecer estos servicios pueden ser on-premise o en la nube.

QVD es un proyecto open-source. Esto significa que se puede tener el control completo sobre el entorno de virtualización y adaptarlo a las necesidades particulares de cada organización. De ahí que QVD proporcione diferentes tipos de soluciones y escritorios tanto para el sector empresarial como educativo.

En lo que respecta a las herramientas y complementos de QVD destacamos los siguientes:

- ❖ Panel de administración web (WAT): Es la herramienta de administración web que permite gestionar de forma centralizada a sus usuarios y escritorios virtuales.
- ❖ Cliente QVD: Permite acceder a uno de sus clientes a su escritorio Linux, Windows, OSX, Android o iOS. Para ello deberá ser instalado en el dispositivo de acceso.

QVD ha sido diseñado para utilizar tecnologías de virtualización como KVM<sup>19</sup> (Máquina Virtual Kernel) y LXC (Contenedores Linux).

En lo que respecta a la Arquitectura QVD tenemos a disposición la siguiente ilustración que se nos muestra.

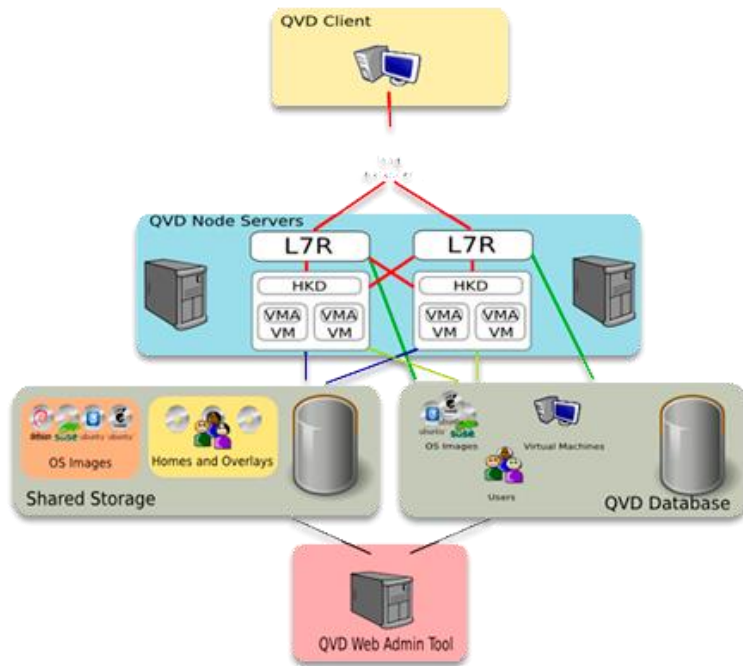


Ilustración 25: Arquitectura Qindel Group –QV<sup>15</sup>D

En esta imagen donde se muestra la típica conectividad de red e interacciones entre los diferentes componentes, cabe destacar:

- ❖ L7R podrá ser uno o más componentes y será el encargado de indicar que máquina virtual desea ser cargada por el usuario en un momento determinado.
- ❖ Las máquinas virtuales son cargadas desde imágenes almacenadas dentro de un ambiente de almacenamiento compartido.
- ❖ HKD será la encargada de arrancar aquellas máquinas virtuales que los usuarios requieran en un preciso instante.

<sup>15</sup> <http://www.theqvd.com/wp-content/uploads/2016/06/Imagen-tecnologia-arquitectura-1.png>

### C.3.2. OVH – Innovation is Freedom

OVH es un proveedor de alojamiento web y telecomunicaciones francés. Ofrece servidores dedicados<sup>6</sup>, alojamiento compartido<sup>2</sup>, registros de dominios<sup>14</sup>, VPS<sup>34</sup> y servicios de Cloud Computing.

OVH ofrece dos alternativas VDI. La primera está orientada para estudiantes, autónomos y PYMES siendo conocido como “Escritorio Virtual en la nube”. La segunda en cambio, está enfocada para grandes empresas y multinacionales. Sobre esta última, conocida como Infraestructura de Escritorio en la nube, será donde centraremos nuestro estudio.

#### C.3.2.1. Cloud Desktop Infraestructure

Cuando una empresa tiene más de 50 equipos informáticos y, por tanto, un sistema que presenta un cierto nivel de complejidad, a esta le suele resultar más conveniente gestionar su propia infraestructura por medio de escritorios virtuales (VDI).

Para simplificar la implementación de semejante infraestructura VDI, OVH propone una resolución de VDI como un servicio basado en VMware Horizon y complementamente alojado en Private Cloud<sup>24</sup>.

Según afirmaciones de la propia compañía, además de poder reproducir de forma idéntica cualquier red local en una infraestructura remota, permite ser más eficaz en la puesta a disposición de nuevos puestos de trabajo, creando sus propias plantillas, incluyendo el sistema operativo<sup>27</sup> y las aplicaciones y permisos de acceso en función de la tipología del usuario y del departamento al que pertenezca.



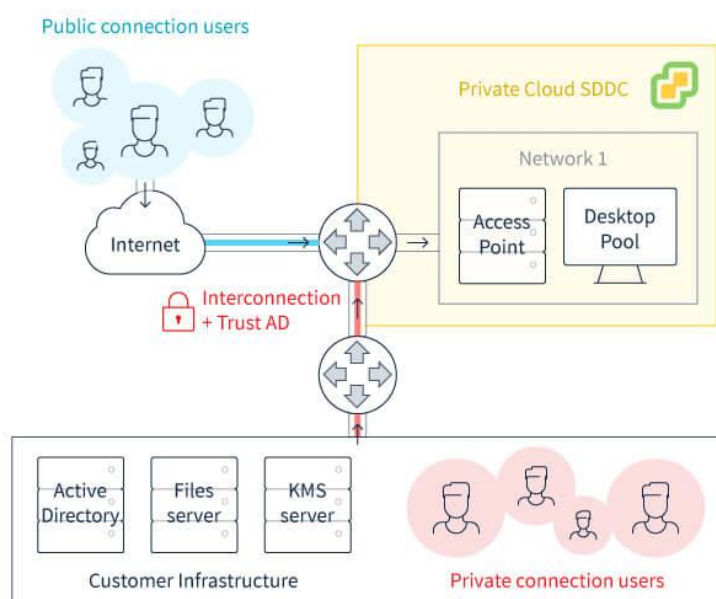
16

Ilustración 26: Cloud Desktop Infraestructure

OVH permite diseñar a la organización la infraestructura que mejor se adapte a sus necesidades. También da la posibilidad de redimensionarla durante todo el ciclo de vida del proyecto. Eso sí, al ser una solución de escritorio virtual en la nube se deberá hacer frente a unos gastos de mantenimiento, de uso, entre otros.

<sup>16</sup> <https://www.ovh.es/images/cloud/cloud-desktop/home/schema-CDI.png>

A continuación, mostramos la arquitectura en la que se basa esta solución de escritorio virtual.



17

Ilustración 27: Arquitectura OVH

Por medio de esta arquitectura, OVH promueve hasta tres configuraciones de conexión:

- ❖ **Conexión pública:** Cada usuario se conecta con sus claves a su escritorio virtual en particular. Dichos escritorios virtuales se encontrarán en una VLAN<sup>30</sup> determinada. Todo ello gestionado por VMware Horizon.
- ❖ **Conexión privada:** En esta configuración, los accesos a los escritorios están totalmente aislados de la red pública. Para llevar a cabo la conexión a los distintos escritorios OVH da dos soluciones: vRack o vRack Connect.
- ❖ **Conexión híbrida:** Tanto la conexión pública como la privada pueden ser complementarias, teniendo unos escritorios en Internet para acceder a ellos y otros aislados para los diferentes servicios de la empresa.

### C.3.3. Nologin

Nologin Consulting es una compañía aragonesa nacida en el año 2000, que asiste a otras empresas de todo el mundo para guardar, gestionar, organizar, proteger y mantener su más preciado activo: la información. Ayudan a las empresas y proveedores de servicios a diseñar, poner en marcha y hacer evolucionar sus infraestructuras tecnológicas a la par que su negocio.

Nologin ofrece una solución de virtualización de escritorios fundamentada en el software de flexVDI. Según la compañía, utilizando esta infraestructura podremos tener un control de nuestra empresa, sin restricciones de lugar o envoltorio físico, reduciendo los costes de propiedad.

<sup>17</sup> <https://www.ovh.es/images/vdi/schema3-V.2.jpg>

### C.3.4. Flexible IT

Flexible IT es una compañía fundada en 2002 que tiene como objetivo ofrecer a sus clientes las soluciones IT más innovadoras y un desarrollo de software personalizado, proporcionando así todos los recursos que una empresa puede necesitar en un instante determinado.

Flexible IT ofrece Flexible Desktop VDI como una alternativa para poder disponer de escritorios virtuales. Flexible Desktop es una solución de escritorios virtuales DaaS (Desktop As A Service) cuyo objetivo es permitir tener disponible todos los aspectos informáticos de una empresa en una nube privada<sup>24</sup>. Se presenta en dos modalidades que se pueden combinar en un mismo entorno:

- ❖ Los escritorios Flexible VDI Premium: Son una solución VDI personalizable, permitiendo la instalación de aplicaciones en el escritorio por parte del usuario.
- ❖ Los escritorios Flexible VDI Professional: Son volátiles (cada VDI se crea de nuevo en cada conexión) y solamente disponen de las aplicaciones configuradas.

Por medio de esta solución Flexible IT pretende proporcionar una herramienta que permite gestionar los escritorios, servidores y recursos por parte de la empresa. A esta tecnología se le conoce como Flexible VDI Manager. También dispone de indicadores de rendimiento<sup>26</sup> de cada escritorio VDI así como un control de acciones para asistir online al usuario.

### C.3.5. Solutia IT

Solutia Innovaworld Technologies, S.L. es una compañía española cuyo objetivo es la distribución, implementación, configuración y mantenimiento de soluciones IT. Es una empresa especializada en el desarrollo de tecnologías aplicadas al sector educativo.

Solutia IT proporciona una solución de virtualización de escritorios de software libre denominada evaOS. Se trata de una infraestructura de escritorios virtuales basada en GNU/Linux cuyo propósito es permitir la publicación y ejecución en un mismo escritorio de aplicaciones Windows y Linux. Existen diferentes modalidades de acceso a evaOS:

- ❖ Con tu propio navegador web puedes acceder a evaOS en dos posibles modos:
  - ◆ Modo escritorio: Te permite acceder al escritorio evaOS donde dispones de las aplicaciones instaladas en el servidor.
  - ◆ Modo portal: Con el modo portal se podrá acceder a las aplicaciones instaladas en el servidor de evaOS.
- ❖ Con el cliente nativo en dos posibles modos:
  - ◆ Modo escritorio: Accedes al escritorio de evaOS donde dispones de las aplicaciones instaladas en el servidor.
  - ◆ Modo portal: En este modo los programas instalados en el servidor de evaOS conviven con los instalados en el propio PC de la organización.

Cabe destacar que el coste de licencias de un usuario evaOS es completamente nulo (coste cero), y dispondrá de diferentes sistemas de autenticación (AD<sup>1</sup>, LDAP y Novell Directory) así como de los principales hipervisores del mercado (Xen, Hyper-V y VMware).

### C.3.6. Huawei

Huawei Technologies Co., Ltd. es una empresa privada multinacional china de alta tecnología especializada en investigación y desarrollo (I+D), producción electrónica y marketing de equipamiento de comunicaciones, además de proveer soluciones de redes personalizadas.

Huawei permite la creación de una infraestructura de escritorio virtual (VDI) por medio del proyecto FusionCloud Desktop.

La solución FusionCloud Desktop de Huawei es una solución de escritorio virtual de extremo a extremo. Esta solución despliega recursos informáticos y de almacenamiento (que incluyen CPU, discos duros y memoria) en un centro de datos en la nube y convierte los recursos físicos en recursos virtuales mediante tecnologías de virtualización. Las máquinas virtuales de diferentes especificaciones se crean mediante recursos virtuales y son provistas a los usuarios según las requieran.

La solución de escritorio FusionCloud de Huawei puede ser aplicable en diversos escenarios, ofreciendo soluciones hechas a medida para cumplir con las necesidades de diferentes industrias. Entre ellas podemos destacar las siguientes:

- ❖ Solución CompactVDI: Orientada a pequeñas empresas.
- ❖ Solución FusionCloud Desktop Appliance: Para empresas medianas y organizaciones con oficinas remotas y sucursales.
- ❖ Solución FusionCloud: Para despliegues comerciales de gran escala.

El uso de la plataforma FusionCloud se suele llevar a cabo por medio de dos herramientas:

- ❖ FusionCube: Es una plataforma convergente de computación, almacenamiento y networking. Incluye los siguientes componentes:
  - ◆ FusionManager: Gestión unificada basada en web, incluyendo monitorización y alarmas, aprovisionamiento y gestión de máquinas virtuales, gestión de usuarios y gestión unificada del hardware.
  - ◆ FusionCompute: Software de virtualización basado en Xen.
  - ◆ FusionStorage: Software de virtualización del almacenamiento.
- ❖ FusionSphere: Es la plataforma en la nube basada en FusionCube. FusionSphere integra todos los componentes de FusionCube y proporciona herramientas integrales de infraestructura en la nube y aplicaciones.

Se utilizará el sistema de autenticación AD<sup>1</sup> para verificar que un usuario pueda acceder a su escritorio virtual determinado.

### C.3.7. Telefónica

Telefónica S.A. es una empresa multinacional española de telecomunicaciones con sede central en Madrid, España.

Telefónica ofrece una solución integral de virtualización del puesto de trabajo en la nube mediante el servicio Virtual Desktop. Con este sistema Telefónica pretende permitir utilizar escritorios y aplicaciones Windows en cualquier dispositivo y cualquier lugar, alcanzando una gestión más eficiente<sup>12</sup>, segura y flexible.

Virtual Desktop es una solución extremo a extremo cuyo objetivo es ofrecer no sólo la capa de infraestructuras sino también la de comunicación. Se trata de un servicio alojado en el centro de datos Tecno-Alcalá y desarrollado con tecnología Citrix.

Telefónica ofrece dos modalidades del servicio:

- ❖ Virtualización de aplicaciones.
- ❖ Virtualización de escritorios: Basada en la entrega al usuario final de un escritorio en la nube con un sistema operativo<sup>27</sup> Windows.



## C.4. Conclusiones

Por medio de este apartado se pretenderá ver de una manera resumida cada una de las soluciones de virtualización de escritorio que hemos estudiado. Por esa razón, distinguiremos entre:

- ❖ Proveedor/Compañía que ofrece o comercializa con una solución de virtualización de escritorio.
- ❖ Si se trata de una solución interna (on-premise) o en la nube.
- ❖ Sistema de autenticación que hace uso.
- ❖ Si se trata de la comercialización de una solución de escritorio virtual proporcionada por otro proveedor.
- ❖ El conjunto de hipervisores que puede utilizar.

Proveedor	Hardware	Sistema de autenticación	Comercialización	Hipervisores
VMware Horizon 7	On premise / Cloud	RSA SercuID, Kerberos, RADIUS, RSA,...	No	VMware vSphere <sup>31</sup> , vSan y NSX
Citrix XenDesktop	On premise	RSA SecureID, Smart Card, RADIUS, Alladin SafeWorld del propio Citrix	No	Citrix Xen Server, Microsoft Hyper-V, Nutanix, Acropolis, VMware vSphere <sup>31</sup>
Microsoft VDI	Cloud	AD <sup>1</sup>	No	RDS
flexVDI	On premise	LDAP, AD <sup>1</sup>	No	KVM <sup>19</sup>
Qindel Group	On premise / Cloud	LDAP	No	KVM <sup>19</sup> , LXC
OVH – Innovation is freedom	Cloud	Trust AD <sup>1</sup> Managed	Si -VMware	
Flexible IT	Cloud		Si -Citrix	
Nologin	On premise		Si - flexVDI	
Huawei	Cloud	AD <sup>1</sup>	Si - Microsoft VDI,	VMware vSphere <sup>31</sup> , Citrix Xen Server
Solutia IT	Cloud	AD <sup>1</sup> , LDAP, Novell Directory	Si - Microsoft VDI, VMware	
VirtualCable	On premise/ Cloud	AD <sup>1</sup> , LDAP, HW, ID, LOCALDB,...	Si - VMware, Citrix,....	KVM <sup>19</sup> entre otros
Telefónica	Cloud	AD <sup>1</sup>	Si - Citrix	

Tabla 12: Resultado proveedores

## D. Manual de instalación de OpenStack PackStack en el Sistema Operativo CentOS

### D.1. Introducción

OpenStack tiene muchas distribuciones y cada día se siguen ampliando. Se ha decidido, al querer montar un entorno de pruebas para testear todo lo que OpenStack es capaz de ofrecernos, y debido a las dificultades encontradas en la implantación de OpenStack DevStack y Ubuntu OpenStack, hacer uso de OpenStack PackStack que es una instalación simplificada del proyecto RDO. RDO como sabemos es una distribución mantenida por una comunidad de usuarios que permite ejecutar OpenStack en sistemas Red Hat y derivados (Fedora y CentOS).

PackStack usa Puppet<sup>25</sup> para desplegar los diversos módulos que componen OpenStack. Se puede hacer uso de un fichero de configuración que permite modificar las propiedades que puede tener OpenStack, desde contraseñas, red, hipervisores, tamaños de disco o incluso instalar más o menos módulos del proyecto OpenStack.

Por tanto el objetivo de este anexo, es reflejar los pasos que tenemos que seguir para poder instalar de manera correcta Openstack utilizando la herramienta PackStack de RDO sobre el sistema operativo<sup>27</sup> CentOS, al estar éste libre de licencia. Esto ni es, ni quiere ser una guía oficial de instalación de OpenStack PackStack. Simplemente quiere ser un manual intuitivo, en la que se represente la manera de poder establecer el sistema operativo<sup>27</sup> y recopilar toda la información necesaria para poder instalar OpenStack.

Para alcanzar este objetivo me he apoyado en los dos próximos vídeos:

- ❖ [https://www.youtube.com/watch?v=ayTCB\\_tvetw&feature=youtu.be](https://www.youtube.com/watch?v=ayTCB_tvetw&feature=youtu.be)
- ❖ <https://www.youtube.com/watch?v=zrMwpZ2yLSk>

Cabe finalmente mencionar, que los datos mostrados son a raíz de la fecha de instalación del producto (marzo 2018), y pueden haber cambiado por motivo de que la frecuencia de actualización de OpenStack es muy alta.

## D.2. Instalación

### D.2.1. Recopilación de información

Para poder hacer uso de esta distribución de OpenStack, tendremos que realizar una instalación de las siguientes aplicaciones:

#### ❖ Aplicación VirtualBox

Es un software de virtualización para arquitecturas x86/amd64. Por medio de esta aplicación es posible instalar sistemas operativos adicionales, conocidos como “sistemas invitados”, dentro de otro sistema operativo<sup>27</sup>, “anfitrión”, cada uno con su propio ambiente virtual. Resulta una aplicación extremadamente útil si estamos analizando sistemas operativos nuevos, versiones Betas inestables, o cuando queremos realizar alguna práctica y no queremos que afecte a nuestro equipo local. Para su correcta descarga se aconseja hacer uso del siguiente enlace: <https://www.virtualbox.org/>.

#### ❖ CentOS Minimal ISO

CentOS (Community Enterprise Operating System) es una distribución de Linux que fue lanzada en marzo de 2004. El proyecto de código abierto<sup>8</sup>, desarrollado y apoyado por una gran comunidad, se basa en los paquetes fuente de Red Hat Enterprise Linux (RHEL). Todas las versiones de CentOS publicadas hasta ahora se apoyan en los lanzamientos equivalentes de RHEL siendo compatible con el mismo.

CentOS es una plataforma corporativa destinada, esencialmente, a su implementación en empresas y organizaciones de gran tamaño. Como ocurre con RHEL, la instalación se realiza a través del gestor gráfico Anaconda. En términos de gestión de paquetes (RPM) y gestión de software (yum), CentOS utiliza los componentes de Red Hat. Lo que no garantiza Red Hat es el correcto funcionamiento de los programas con CentOS al carecer de certificaciones y el apoyo proporcionado por el fabricante.

Se recomienda descargar la última versión de este software por medio de este enlace: [http://isoredirect.centos.org/centos/7/isos/x86\\_64/CentOS-7-x86\\_64-Minimal-1708.iso](http://isoredirect.centos.org/centos/7/isos/x86_64/CentOS-7-x86_64-Minimal-1708.iso) .

#### ❖ Aplicación PuTTY

PuTTY es un cliente SSH, Telnet, rlogin y TCP raw con licencia libre. Disponible originalmente sólo para Windows, ahora también está disponible en varias plataformas Unix, y se está desarrollando la versión para MAC OS clásico y MAC OS X. PuTTY nos va a permitir conectarnos a servidores remotos pudiendo iniciar una sesión de línea de comandos para poder administrarlo.

PuTTY es una aplicación gratuita, de código abierto<sup>8</sup> y portable; proporciona una interfaz sencilla y manejable con multitud de opciones, y se encuentra en constante desarrollo. El nombre PuTTY proviene de las siglas Pu: Port unique y TTY: terminal type. Su traducción al castellano sería: Puerto único de tipo terminal.

Para realizar una correcta descarga se sugiere seguir el siguiente enlace: <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html> .

## D.2.2. Creación de la Máquina Virtual

Una vez que hemos instalado la aplicación VirtualBox, vamos a crear una máquina virtual con la imagen del sistema operativo<sup>27</sup> CentOS que hemos descargado. Para ello seguiremos los siguientes pasos:

- I. Indicar en la aplicación VirtualBox nuestra necesidad de hacer uso de una nueva máquina virtual. Creamos esta nueva máquina a partir de la opción habilitada en la aplicación de VirtualBox.



Ilustración 28: VirtualBox - Creación Nueva Máquina Virtual

- II. Dotamos a nuestra máquina de un nombre. Además, deberemos escoger el tipo de sistema operativo<sup>27</sup> que vamos a utilizar (Linux) y la versión (Red Hat de 64 bits) que corresponde a esa máquina virtual.

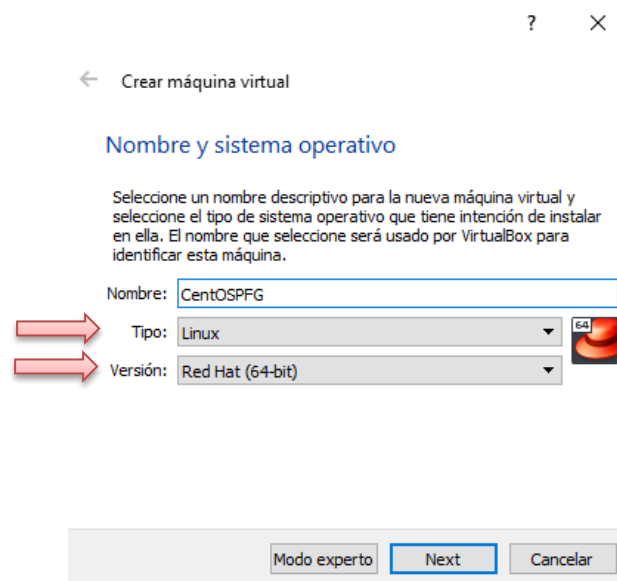


Ilustración 29: VirtualBox - Nombre y Sistema Operativo

- III. Escogemos el espacio de memoria RAM. Para esta instalación se recomienda hacer uso como mínimo de 6GB (6140MB).

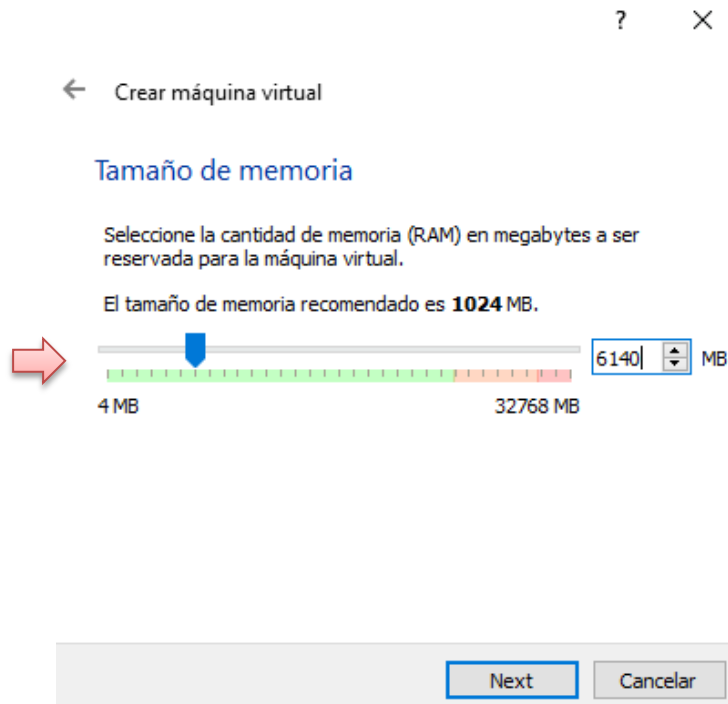


Ilustración 30: VirtualBox - Espacio de memoria (RAM)

Cabe destacar que lo marcado en color verde referencia lo recomendado a utilizar. Es por ello que en mi caso, se ha podido escoger los 6GB de espacio recomendado.

- IV. Agregamos un disco virtual a la nueva máquina con un espacio por defecto de 8 GB que posteriormente podremos cambiar si así lo estimamos oportuno.

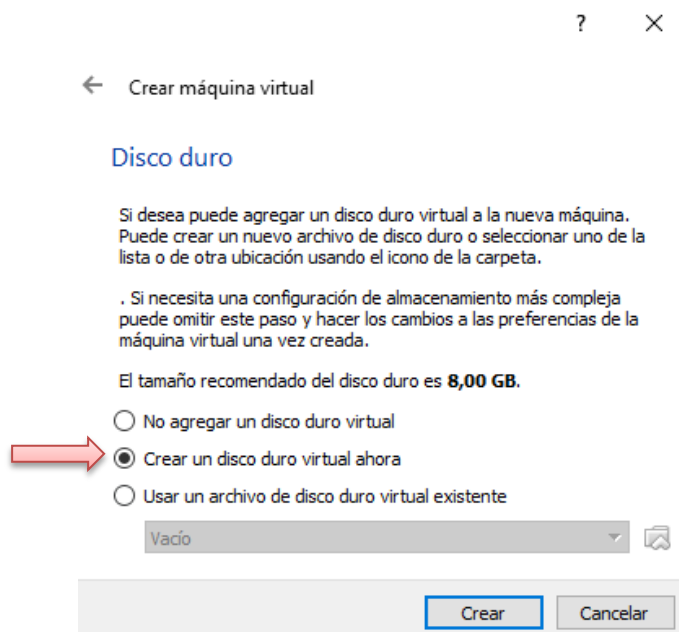


Ilustración 31: VirtualBox - Asignación disco virtual

- V. Seleccionamos el tipo de archivo que se quiere utilizar para el nuevo disco duro como “VDI” (VirtualBox Disk Image).

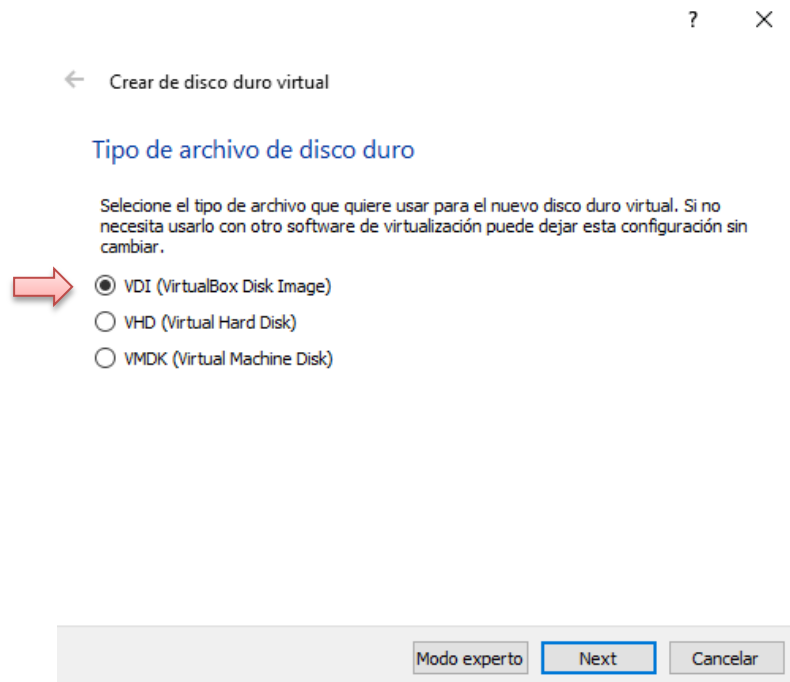


Ilustración 32: VirtualBox - Tipo de archivo de disco duro virtual

- VI. A continuación, determinamos el almacenamiento en la unidad de disco duro virtual como “reservado dinámicamente”.

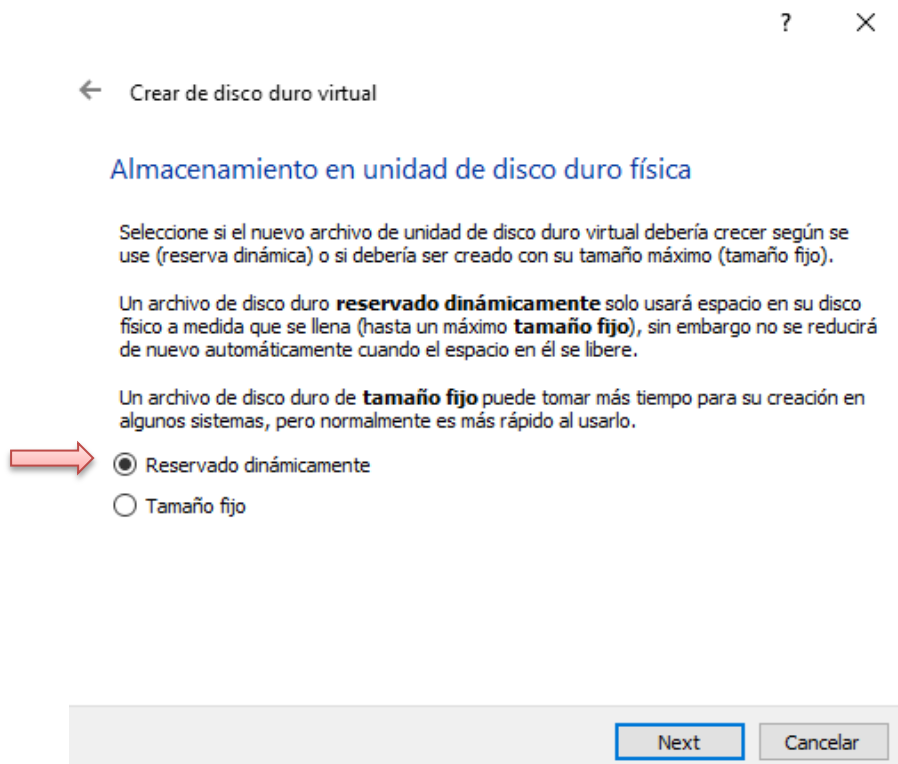


Ilustración 33: VirtualBox - Almacenamiento en unidad de disco duro físico

- VII. Lo siguiente a realizar será seleccionar la ubicación donde almacenaremos nuestra máquina virtual, y el tamaño del disco duro virtual que deseamos asignar (recomendado 50 GB).

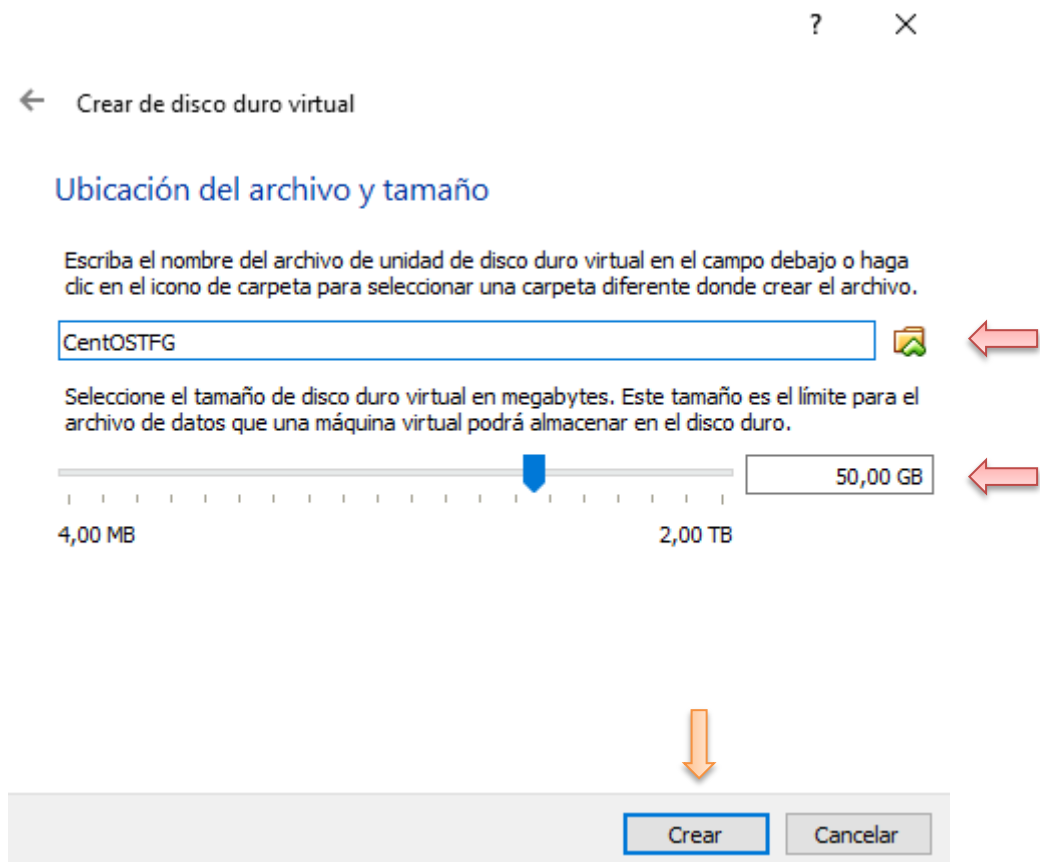


Ilustración 34: VirtualBox - Ubicación del archivo y tamaño

- VIII. Una vez hemos creado la máquina, está aparecerá reflejada en el portal principal de la aplicación VirtualBox con un estado inicial de apagado.

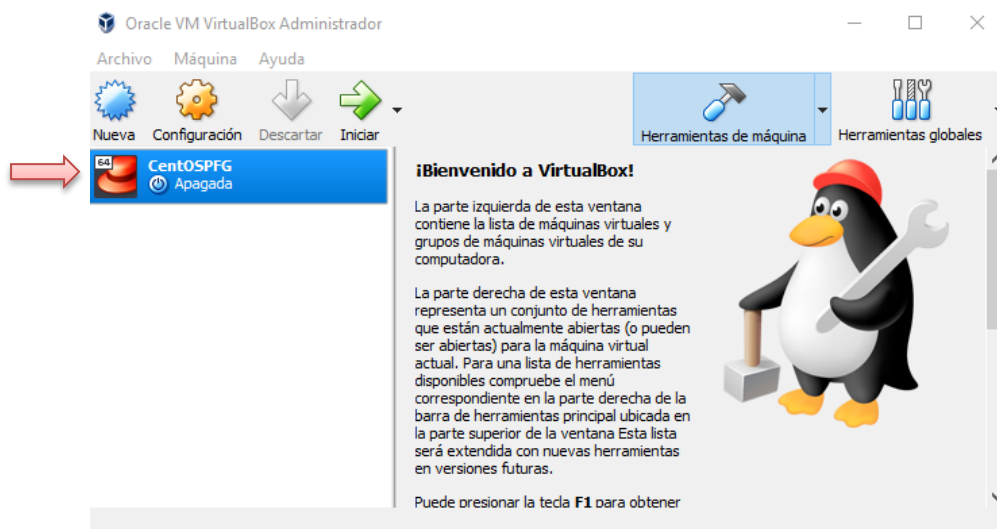


Ilustración 35: VirtualBox - Máquina creada

Una vez hemos visto cómo se realiza la creación de la máquina, y antes de ponerla en funcionamiento, deberemos configurarla por medio de la opción que facilita la aplicación VirtualBox.

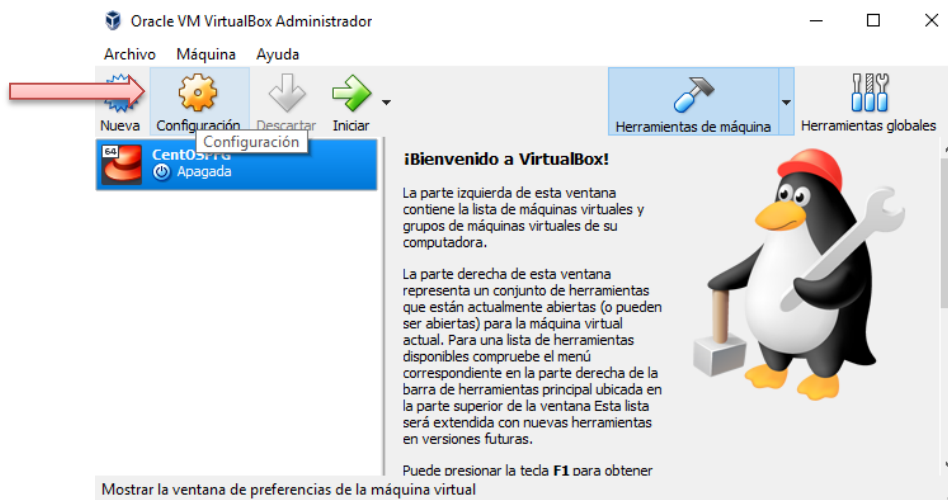


Ilustración 36: VirtualBox - Configuración

Habiendo accedido a la configuración de nuestra máquina virtual, habrá que realizar lo siguiente:

- a) En el apartado de red tenemos que cambiar el adaptador a “puente” o “bridge” para tener una dirección IP de nuestro router. Además, en las opciones avanzadas, tenemos que seleccionar “permitir todo” en la opción de modo promiscuo para evitar un conjunto de problemas que podríamos tener en esta instalación. Con lo cual con esta configuración vamos a ser capaces de poder instalar OpenStack utilizando acceso externo a nuestras instancias.

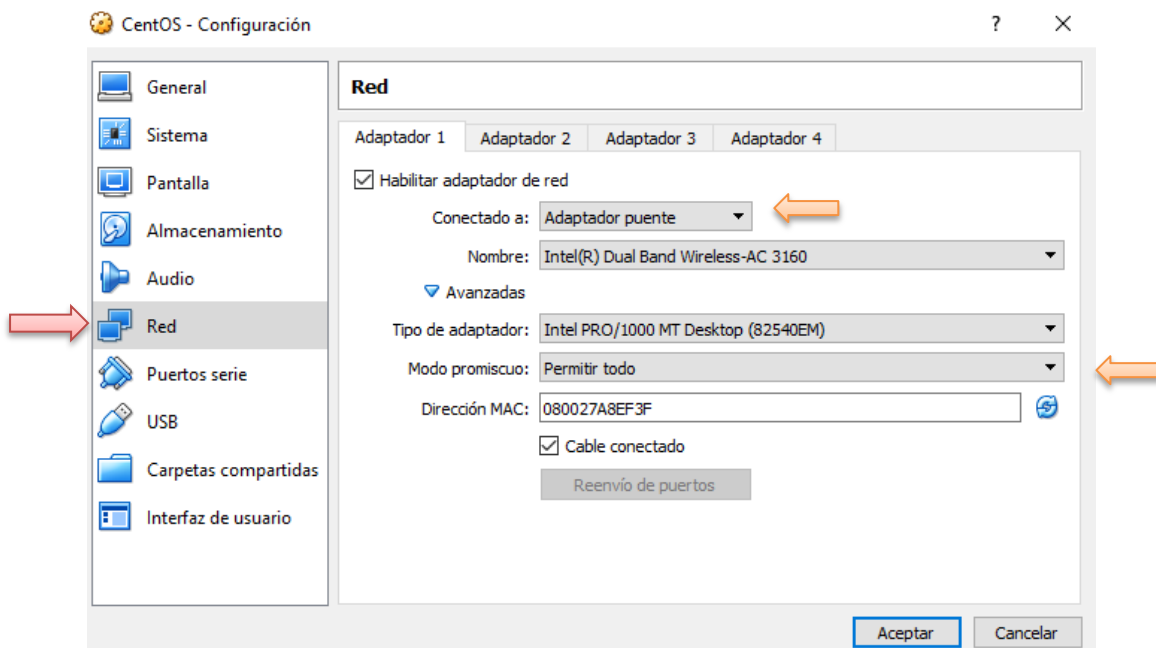


Ilustración 37: VirtualBox - Configuración Red



b) En la parte de almacenamiento seleccionamos en controlador IDE. Finalmente, en la opción de disco seleccionamos nuestra imagen de CentOS, que será el sistema operativo<sup>27</sup> que albergue esa máquina virtual.

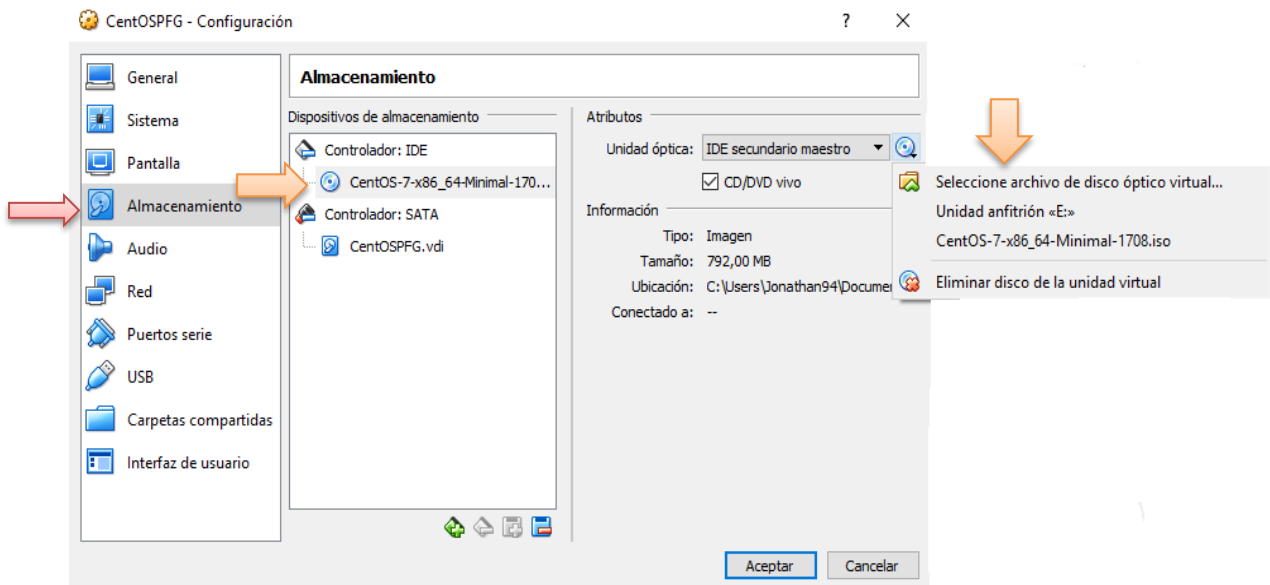


Ilustración 38: VirtualBox - Configuración de almacenamiento

Antes de finalizar con la configuración, se recomienda asegurarse de que el resto de opciones se encuentran en el estado deseado.

Una vez hemos terminado la configuración de la máquina, la arrancamos e iniciamos la instalación de CentOS que vendrá explicada en el subapartado siguiente.

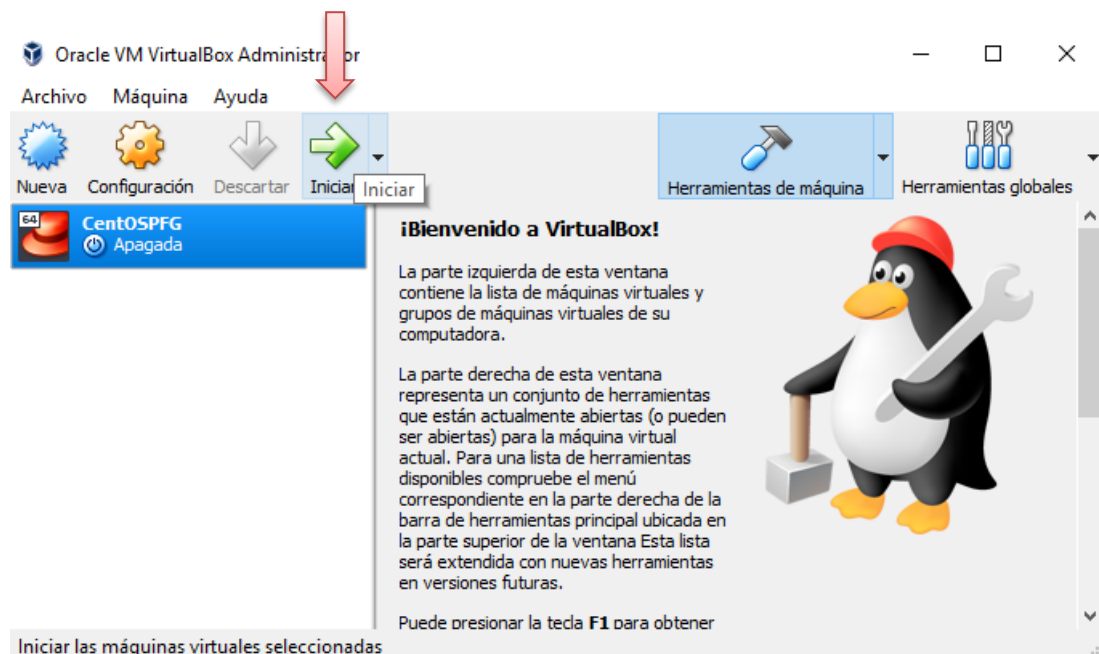


Ilustración 39: VirtualBox - Iniciar máquina virtual

### D.2.3. Instalación del Sistema Operativo CentOS

Por medio de este apartado explicaremos de manera breve la forma de realizar una correcta instalación del sistema operativo<sup>27</sup> CentOS cumpliendo con todos nuestros requerimientos.

1. Seleccionamos el idioma. En nuestro caso será el español.



Ilustración 40: CentOS - Selección idioma

2. Seleccionamos la opción de “destino de la instalación” en donde seleccionaremos y definiremos nuestro particionado.



Ilustración 41: CentOS - Destino de la instalación

2.1. Seleccionamos la opción “Voy a configurar las particiones” y a continuación pulsamos el botón “Listo”.

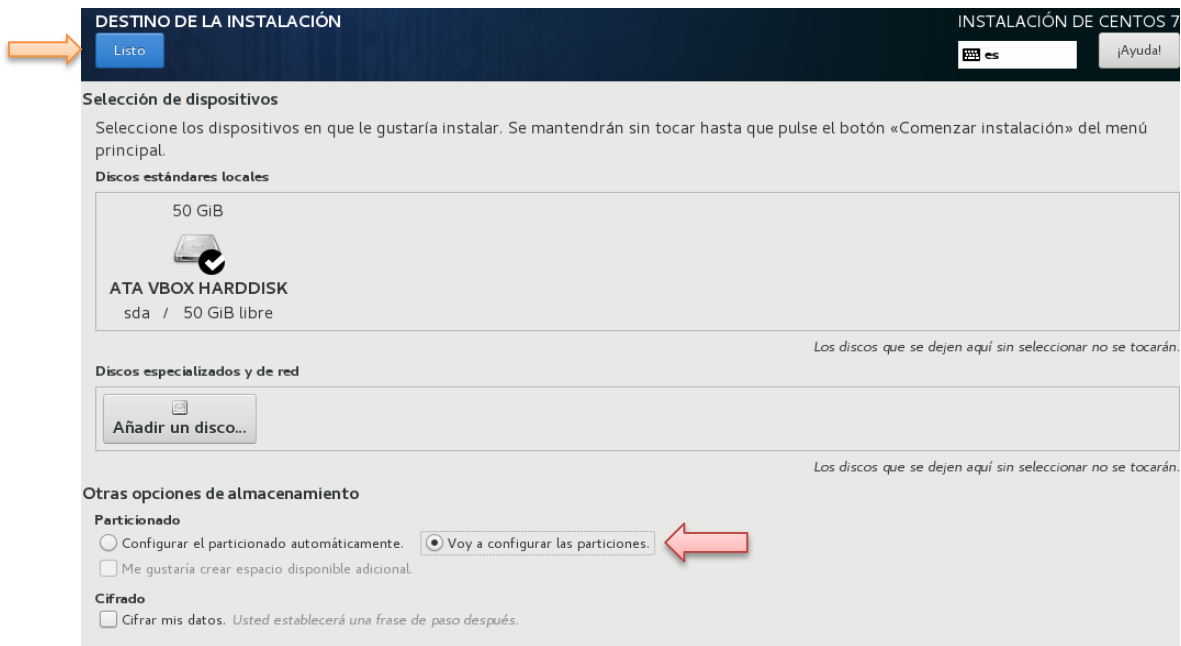


Ilustración 42: CentOS - Opciones de almacenamiento

2.2. Lo siguiente será montar nuestros puntos de montaje. Para ello siempre recurrimos al botón ubicado en la parte inferior izquierda (“+”).

El primer punto de montaje que se agregará será el que corresponde a “/boot” (sistema de arranque) al cual le asignaremos una capacidad de 1024MB.

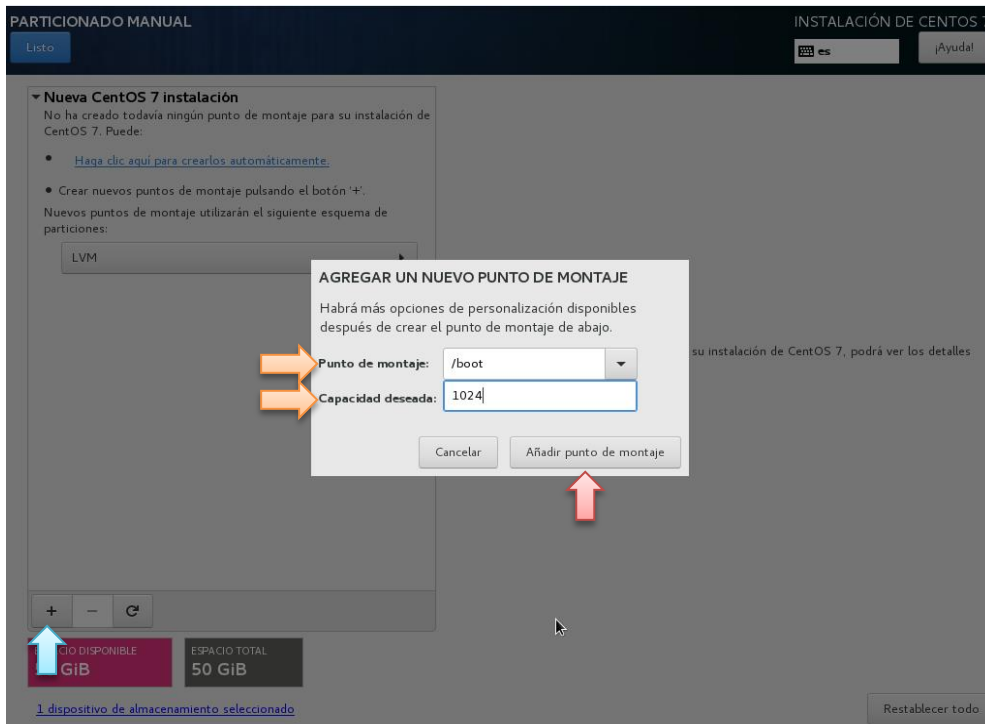


Ilustración 43: CentOS - Punto de montaje “/boot”

2.3. Añadimos un segundo punto de montaje para la “swap” que tendrá una ocupación igual que la memoria asignada a la máquina virtual, es decir, 6140 MB.

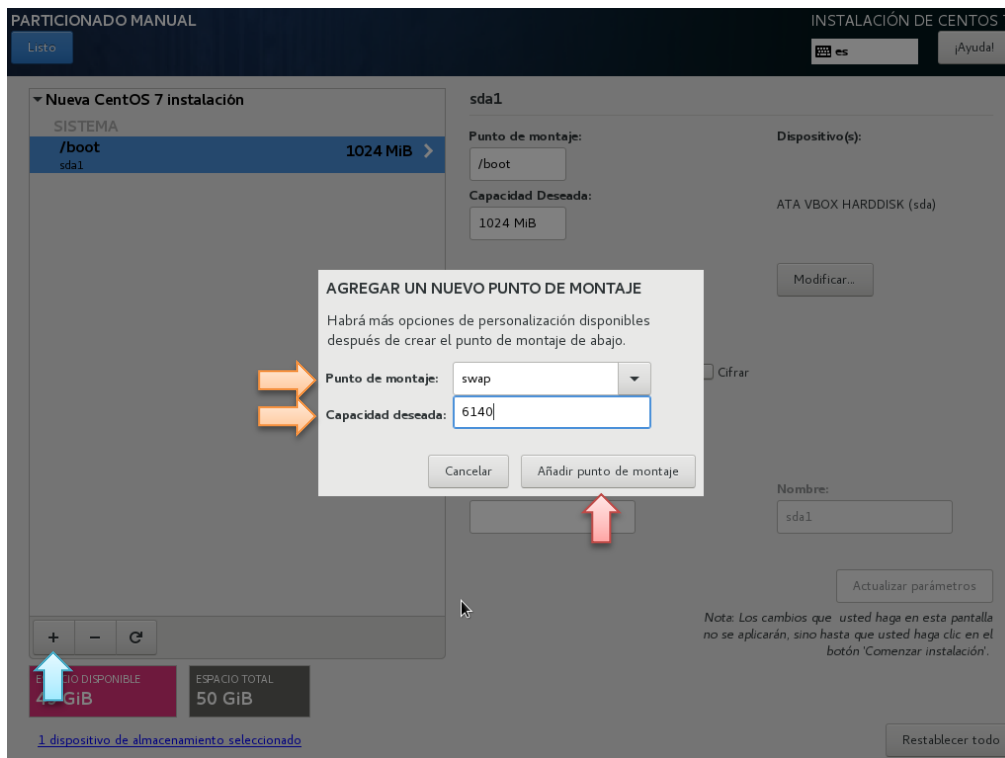


Ilustración 44: CentOS - Punto de montaje "swap"

2.4. Creamos un último punto de montaje conocido como “/”, y no le asignamos capacidad para que pueda montar el resto de componentes disponibles.

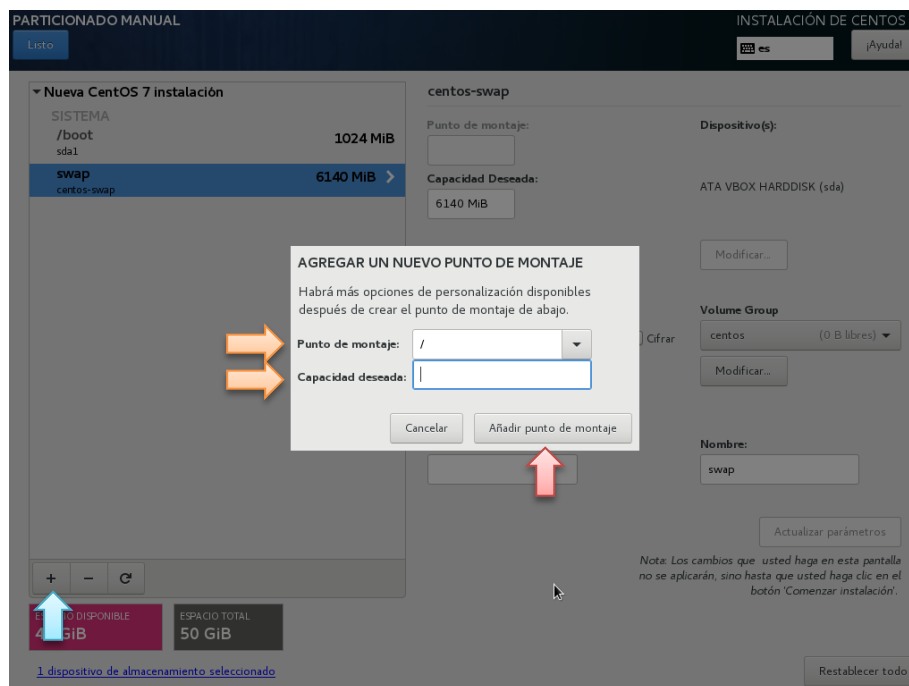


Ilustración 45: Punto de montaje "/"

2.5. Finalizamos pinchando en la opción “Listo” y aceptando los cambios.

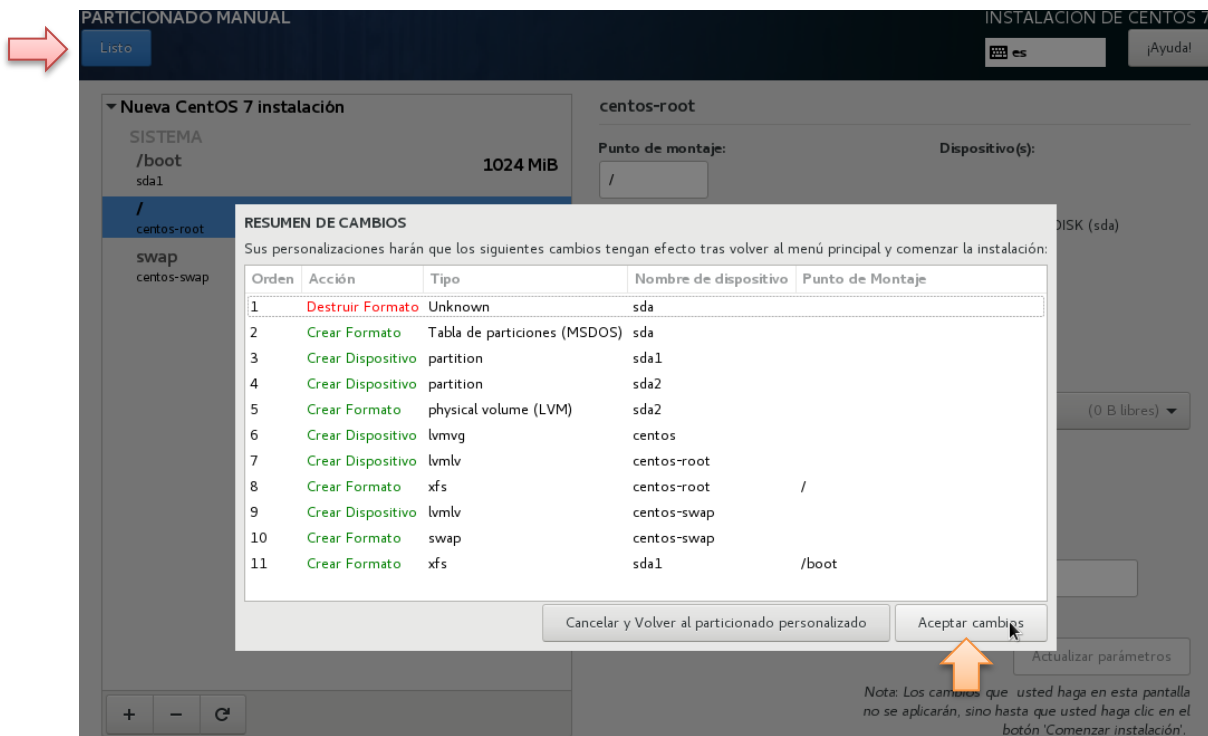


Ilustración 46: CentOS - Aceptación cambios de punto de montaje

3. Nos dirigimos a continuación a la opción KDUMP que será deshabilitada.



Ilustración 47: CentOS - Opción KDUMP

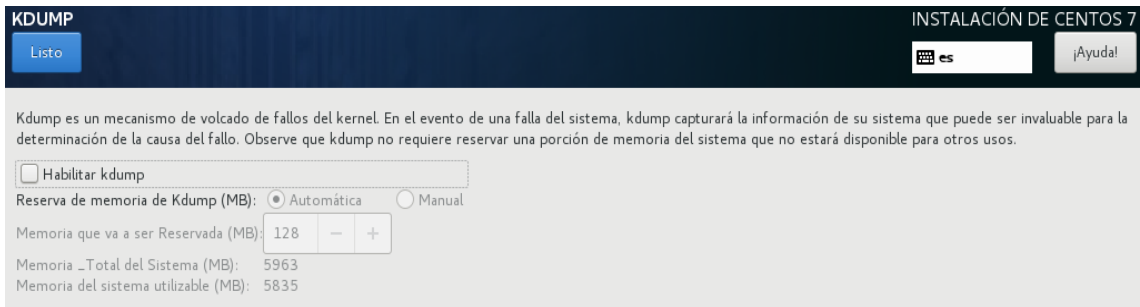


Ilustración 48: CentOS - Deshabilitar KDUMP

4. Seleccionamos la opción “Red & Nombre de Equipo”.



Ilustración 49: CentOS - Seleccionar Red & Nombre de Equipo

4.1. Activamos la opción de Ethernet debiendo de reflejar mayor información que cuando se encuentra deshabilitada.

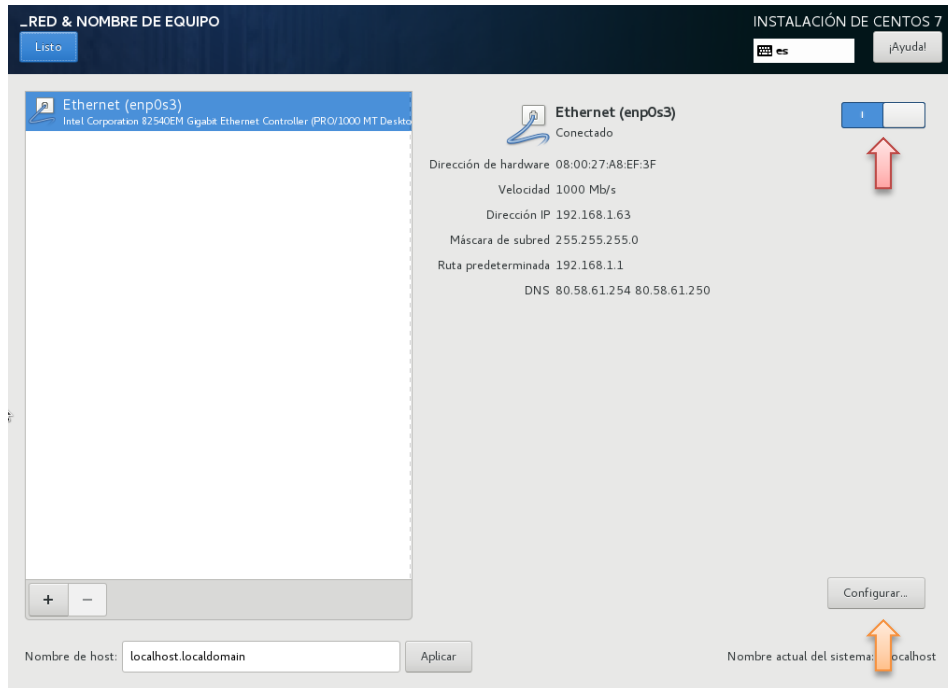


Ilustración 50: CentOS - Activación Ethernet

4.2. Configuramos nuestra red para asignar una dirección IP fija. Para ello nos dirigimos a la opción "Ajustes de IPv4" y seleccionamos el método manual en vez de hacer uso del protocolo DHCP. Rellenamos los datos con la IP, Máscara y Gateway de una red en mi sistema que se encuentre libre y definimos, para este caso, el router como nuestro servidor DNS. Finalizamos guardando todas las alteraciones.

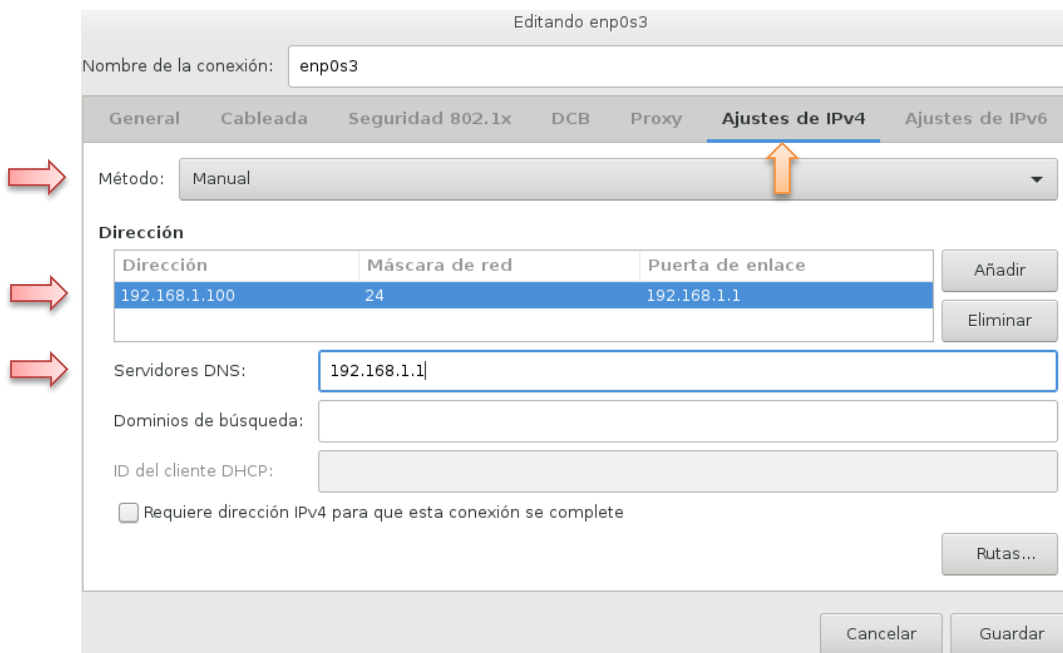


Ilustración 51: CentOS - Ajustes de IPv4

4.2. Observamos todos los cambios realizamos y si deseamos, podemos cambiar incluso el nombre de la máquina (host). En mi caso la he considerado denominar "openstack.localdomain".

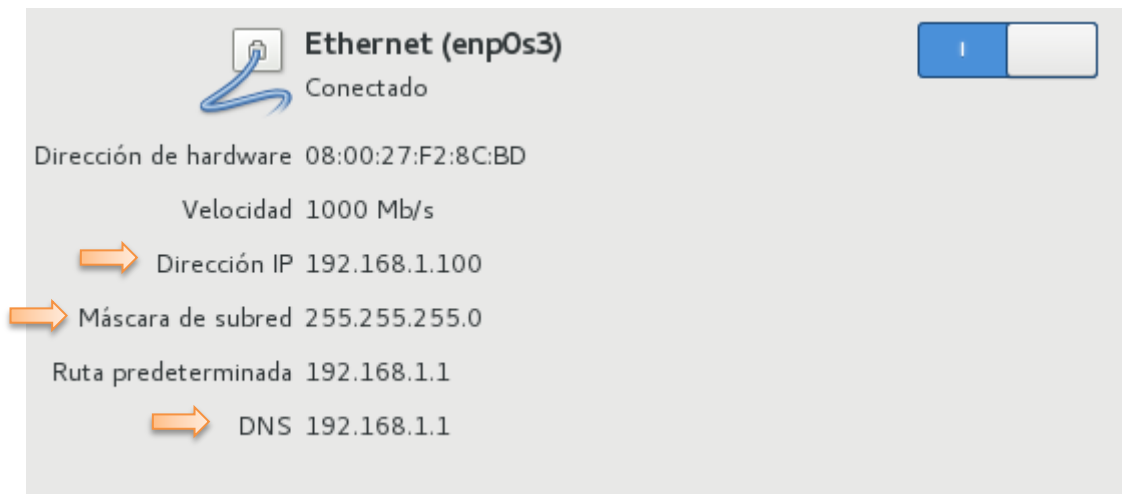


Ilustración 52: CentOS - Cambios tarjeta de red y nombre de máquina

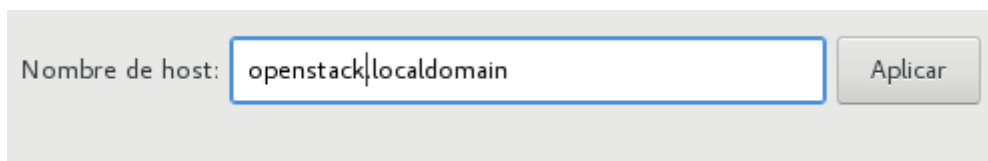


Ilustración 53: CentOS - Nombre máquina

Una vez hemos realizado todos estos cambios en nuestra tarjeta de red, recurrimos a la opción "Listo" para guardarlos.

5. Iniciamos la instalación del sistema operativo<sup>27</sup>.



Ilustración 54: CentOS - Inicio instalación



5.1. Durante la instalación deberemos asignar al usuario administrador (root) una contraseña. Opcionalmente, podemos crear un usuario.



Ilustración 55: CentOS - Contraseña root y creación nuevo usuario

Cuando la instalación finalice, nos pedirá reiniciar. Entonces realizamos este reinicio y nos logueamos con usuario "root".

```
CentOS Linux 7 (Core)
Kernel 3.10.0-693.el7.x86_64 on an x86_64

openstacklocalhost login: root
Password:
[root@openstacklocalhost ~]#
```

Ilustración 56: CentOS - Usuario Root

Para finalizar esta instalación, comprobamos que la dirección IP de nuestra máquina se ha configurado correctamente y que es reconocida, haciendo uso del comando “ping”<sup>9</sup> en la línea de comandos de nuestra máquina local.

```
C:\Users\ehu>ping 192.168.1.100

Haciendo ping a 192.168.1.100 con 32 bytes de datos:
Respuesta desde 192.168.1.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.100: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.1.100:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

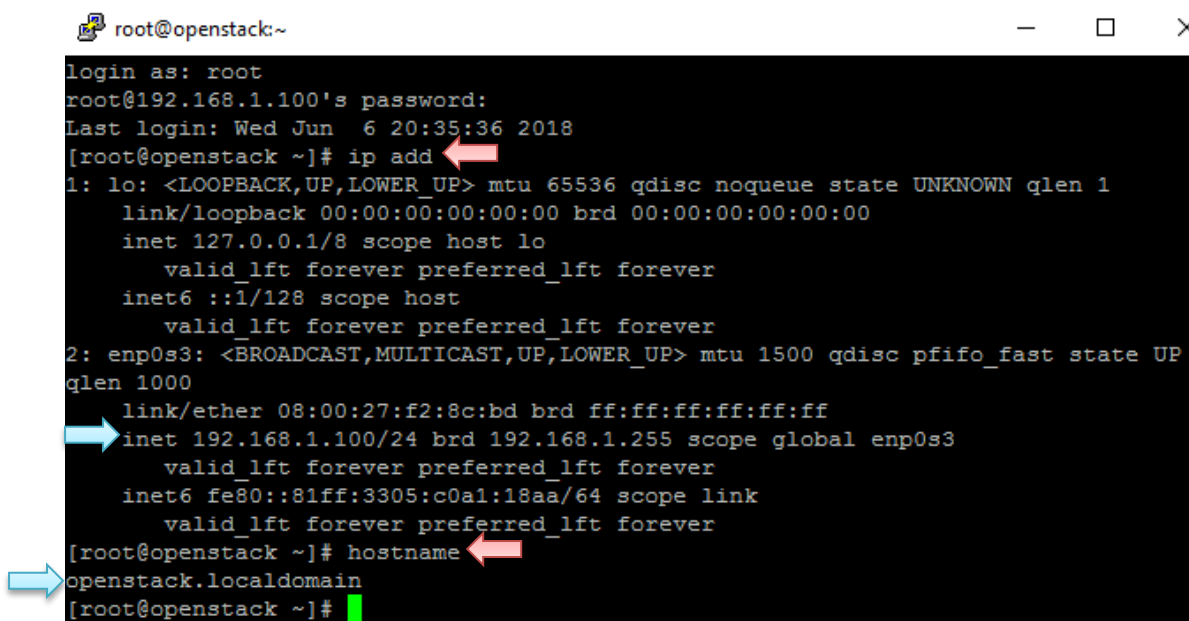
Ilustración 57: CentOS - Comprobación dirección IP

## D.2.4. Configuración del Sistema Operativo CentOS

Accedemos con la aplicación PuTTY a nuestra máquina virtual. Para ello indicamos la dirección y el puerto en cuestión.

Si todo es correcto, iniciamos sesión. Para ello indicamos el usuario (root) y la contraseña que tenga asociada. Si tuviéramos cualquier error, se recomienda además de visitar diferentes blogs, ir al final de esta documentación, donde se refleja una posible solución.

A continuación, necesitamos comprobar que nos encontramos en nuestra máquina virtual. Para ello analizamos si la red y el nombre del equipo son los adecuados mediante los comandos “ip add” y “hostname” respectivamente.



```
root@openstack:~
login as: root
root@192.168.1.100's password:
Last login: Wed Jun  6 20:35:36 2018
[root@openstack ~]# ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
   qlen 1000
    link/ether 08:00:27:f2:8c:bd brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.100/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::81ff:3305:c0a1:18aa/64 scope link
        valid_lft forever preferred_lft forever
[root@openstack ~]# hostname
openstack.localdomain
[root@openstack ~]#
```

Ilustración 58: PuTTY - Comprobación

Una vez hemos concluido correctamente esta comprobación, paramos y deshabilitamos una serie de servicios que pueden provocar diferentes problemas en la instalación de OpenStack: el firewalld y NetworkManager.

```
[root@openstacklocalhost ~]# systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
[root@openstacklocalhost ~]# systemctl stop firewalld
[root@openstacklocalhost ~]# systemctl disable NetworkManager
Removed symlink /etc/systemd/system/multi-user.target.wants/NetworkManager.service.
Removed symlink /etc/systemd/system/dbus-org.freedesktop.NetworkManager.service.
Removed symlink /etc/systemd/system/dbus-org.freedesktop.nm-dispatcher.service.
[root@openstacklocalhost ~]# systemctl stop NetworkManager
```

Ilustración 59: PuTTY - Deshabilitar firewalld y Network Manager

Lo próximo a realizar será deshabilitar la opción “selinux” la cual también puede provocarnos dificultades a la hora de hacer uso de OpenStack.

1. Accedemos al fichero donde se encuentra configurado.

```
[root@openstacklocalhost ~]# vi /etc/selinux/config
```

Ilustración 60: PuTTY - Editar fichero "selinux"

2. Deshabilitamos “selinux”.

```
# 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 protected.
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Ilustración 61: PuTTY - Deshabilitar "selinux"

3. Reiniciamos la máquina con el comando “reboot”.

```
[root@openstacklocalhost ~]# reboot
```

Ilustración 62: PuTTY - Reinicio máquina

Esto originará que PuTTY pierda la conexión establecida con la máquina. Por lo tanto una vez que esta se haya reiniciado esta última, restablecemos la conexión teniendo que volver a indicar el usuario (root) y la contraseña anterior.

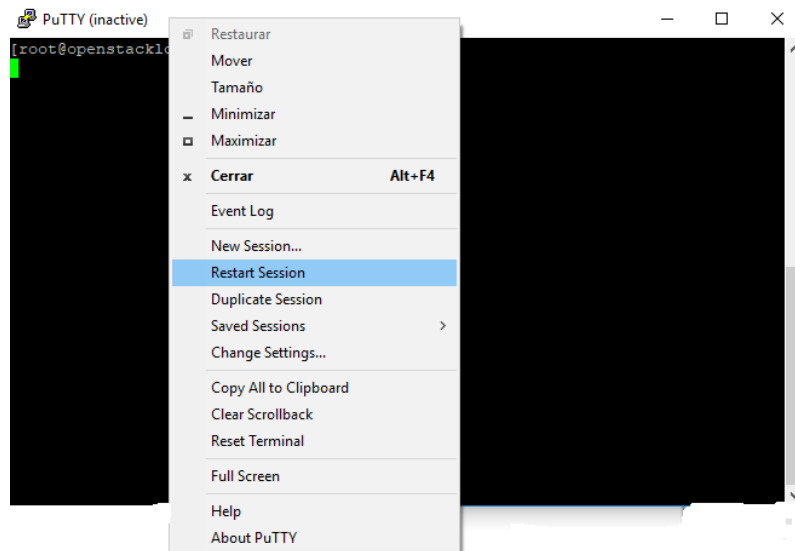


Ilustración 63: PuTTY - Restablecer conexión

4. Cuando hayamos accedido con nuestras credenciales, comprobamos que se ha deshabilitado "selinux".

```
[root@openstacklocalhost ~]# getenforce
Disabled
```

Ilustración 64: PuTTY - Comprobar "selinux" deshabilitado

Una vez hemos alcanzado esta situación, nos encontramos preparados para iniciar la instalación de OpenStack.

## D.2.5. Instalación de OpenStack

La versión de OpenStack que se va a descargar es la de Ocata, que aunque no sea la última, es de las más estables. Para proceder a esta instalación desde los repositorios CentOS, hacemos uso del siguiente comando:

```
[root@openstacklocalhost ~]# yum install -y centos-release-openstack-ocata
```

Ilustración 65: OpenStack - Instalación versión Ocata

Con todo ello accedemos al repositorio para poder descargar Ocata que será posteriormente instalado en nuestro sistema.

Una vez ha sido cargado, procedemos a utilizar el comando “yum update -y” para actualizar nuestro sistema.

```
[root@openstacklocalhost ~]# yum update -y
```

Ilustración 66: OpenStack - Actualizar sistema

La actualización del sistema puede prolongarse un rato, siendo posteriormente recomendable reiniciar nuestro sistema por si se han instalado componentes importantes.

```
[root@openstacklocalhost ~]# reboot
```

Ilustración 67: OpenStack - Reinicio sistema

(Recordemos que se deberá ser restablecida la conexión en la aplicación PuTTY como hemos visto anteriormente)

A continuación, instalamos OpenStack PackStack por medio del siguiente comando.

```
[root@openstacklocalhost ~]# yum install -y openstack-packstack
```

Ilustración 68: OpenStack - Instalación OpenStack PackStack

Con esta instalación habremos obtenido todos los servicios y todos los componentes que necesitamos para desplegar nuestra versión de OpenStack PackStack, aunque en la última de las instalaciones que hagamos deberemos escoger aquellos componentes que vayamos a requerir. Antes volveremos a actualizar nuestro sistema como hemos realizado anteriormente con su posterior reinicio (“yum update -y” y “reboot”).

Por último ejecutaremos el comando de configuración e instalación final de OpenStack PackStack.

```
[root@openstacklocalhost ~]# packstack --allinone --provision-demo=n --os-neutron-ovs-bridge-mappings=extnet:br-ex --os-neutron-ml2-type-drivers=vxlan,flat --os-heat-install=y
```

Ilustración 69: OpenStack - Configuración e Instalación final OpenStack PackStack

En donde:

- ❖ “--allinone --provision-demo=n”: Queremos hacer uso de todos los componentes, ubicándolos en un único nodo de PackStack, sin modo demo.
- ❖ “--os --heat-install=y”: Indicamos nuestra necesidad de descargar el módulo de orquestación que no se encuentra por defecto en OpenStack PackStack.
- ❖ “--os-neutron-ovs-bridge-mappings=extnet:br-ex--os-neutron-ml2-type-drivers=vxlan,flat”: Solicitamos la instalación de un acceso externo para poder acceder a nuestra versión de OpenStack PackStack desde fuera. Esto provoca que podamos acceder a las instancias que vayamos a lanzar. De esta manera también, vamos a poder gestionar nuestras instancias desde la aplicación PuTTY, por ejemplo.

La ejecución de este comando puede tardar entre 30 y 90 minutos aproximadamente, todo dependiendo de las características de nuestro equipo. Cuando finalice tenemos que recibir un mensaje de que la instalación se ha completado correctamente. En caso de existir cualquier error, deberemos buscar de donde puede venir. Se recomienda verificar los requisitos mínimos, asegurarse de que ningún comando te ha dado fallo, y buscar los errores en el navegador.



```
**** Installation completed successfully ****

Additional information:
* A new answerfile was created in: /root/packstack-answers-20180323-114125.txt
* Time synchronization installation was skipped. Please note that unsynchronized time on server instances might be problem for some OpenStack components.
* File /root/keystonerc_admin has been created on OpenStack client host 192.168.1.50. To use the command line tools you need to source the file.
* To access the OpenStack Dashboard browse to http://192.168.1.50/dashboard . Please, find your login credentials stored in the keystonerc_admin in your home directory.
* The installation log file is available at: /var/tmp/packstack/20180323-114125-DMbi2N/openstack-setup.log
* The generated manifests are available at: /var/tmp/packstack/20180323-114125-DMbi2N/manifests
Tiene correo nuevo en /var/spool/mail/root
```

Ilustración 70: OpenStack - Instalación completada satisfactoriamente

En la parte inferior del mensaje de instalación te proporcionan una serie de detalles para acceder a OpenStack, tanto por la interfaz web (Dashboard) como por la línea de comandos (CLI), pero antes de nada vamos a terminar de configurar nuestra instalación.

## D.2.6. Configuración de OpenStack

En primer lugar revisamos lo que ha ocurrido en nuestra red mediante el comando “ip add”.

```
[root@openstacklocalhost ~]# ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
   link/ether 08:00:27:a8:ef:3f brd ff:ff:ff:ff:ff:ff
   inet 192.168.1.50/24 brd 192.168.1.255 scope global enp0s3
       valid_lft forever preferred_lft forever
   inet6 fe80::a00:27ff:fea8:ef3f/64 scope link
       valid_lft forever preferred_lft forever
3: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
   link/ether 26:26:0a:cb:67:90 brd ff:ff:ff:ff:ff:ff
4: br-ex: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN qlen 1000
   link/ether 22:de:98:b8:86:46 brd ff:ff:ff:ff:ff:ff
   inet6 fe80::20de:98ff:feb8:8646/64 scope link
       valid_lft forever preferred_lft forever
5: br-int: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
   link/ether 0e:6d:9b:11:7c:4a brd ff:ff:ff:ff:ff:ff
6: br-tun: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
   link/ether de:b5:ed:7b:1b:4b brd ff:ff:ff:ff:ff:ff
```

Ilustración 71: Configuración - Tarjetas de red

Como bien podemos apreciar tenemos hasta cuatro nuevas tarjetas de red que se han creado durante la instalación de OpenStack. Nuestro objetivo será que la tarjeta “enp0s3” pase los datos a la tarjeta “br-ex”. Esto es por motivo de que si queremos después utilizar las direcciones IP flotantes para acceder a las instancias, por ejemplo por SSH, se requiere realizarlo. Básicamente se fundamenta en montar la tarjeta en un modo específico para luego poder crear la red externa con conexión.

Para llevar a cabo ello deberemos editar los ficheros de configuración de cada una de las tarjetas. Si empezamos por la primera, deberemos acceder al siguiente archivo:

```
[root@openstacklocalhost ssh]# vi /etc/sysconfig/network-scripts/ifcfg-enp0s3
```

Ilustración 72: Configuración - Fichero de configuración tarjeta "enp0s3"

Una vez hemos accedido, borramos todos los datos de los que está constituido y copiamos los siguientes:

```
TYPE="OVSPort"
NAME="enp0s3"
DEVICE="enp0s3"
DEVICETYPE="ovs"
OVS_BRIDGE="br-ex"
ONBOOT="yes"
```

Ilustración 73: Configuración - Datos tarjeta "enp0s3"

Con estos datos lo que promovemos es que la tarjeta “enp0s3” se utilice en modo OVS<sup>23</sup>.

Una vez hemos guardado todos estos cambios realizados sobre la tarjeta “enp0s3”, habrá que modificar el archivo de configuración de la tarjeta “br-ex”.

```
[root@openstacklocalhost ssh]# vi /etc/sysconfig/network-scripts/ifcfg-br-ex
```

Ilustración 74: Configuración - Fichero de configuración tarjeta "br-ex"

Este fichero se encontrará al principio sin ninguna línea escrita, y deberemos reflejar lo siguiente que aparece en la imagen.

```
DEVICE="br-ex"
DEVICETYPE="ovs"
TYPE="OVSBridge"
BOOTPROTO="static"
IPADDR=192.168.1.100
NETMASK=255.255.255.0
GATEWAY=192.168.1.1
PEERDNS="yes"
PEERROUTES="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="no"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_PEERDNS="yes"
IPV6_PEERROUTES="yes"
IPV6_FAILURE_FATAL="no"
DNS1=192.168.1.1
ONBOOT="yes"
```

Ilustración 75: Configuración - Datos tarjeta "br-ex"

Cabe destacar que en cada caso, deberemos rellenar los campos de la dirección IP, la máscara de red, la puerta de enlace y la dirección DNS; con nuestro datos en cuestión.

Finalmente, nos queda reiniciar el servicio de red y comprobar que ahora nuestra tarjeta con dirección IP es la tarjeta “br-ex”.

```
[root@openstacklocalhost ssh]# service network restart
Restarting network (via systemctl): [ OK ]
```

Ilustración 76: Configuración - Reinicio del servicio de red



```

[root@openstack ~]# ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master ovs-system state UP group default qlen 1000
   link/ether 08:00:27:f2:8c:bd brd ff:ff:ff:ff:ff:ff
   inet6 fe80::a00:27ff:fef2:8cbd/64 scope link
       valid_lft forever preferred_lft forever
3: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
   link/ether d2:bb:ab:32:f8:14 brd ff:ff:ff:ff:ff:ff
5: br-int: <BROADCAST,MULTICAST> mtu 1450 qdisc noop state DOWN group default qlen 1000
   link/ether 56:bc:19:84:e3:42 brd ff:ff:ff:ff:ff:ff
6: br-tun: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
   link/ether 4e:14:be:1c:11:44 brd ff:ff:ff:ff:ff:ff
20: br-ex: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN group default qlen 1000
   link/ether 08:00:27:f2:8c:bd brd ff:ff:ff:ff:ff:ff
   inet 192.168.1.100/24 brd 192.168.1.255 scope global br-ex
       valid_lft forever preferred_lft forever
   inet6 fe80::2491:8cff:fe0a:be44/64 scope link
       valid_lft forever preferred_lft forever
21: qbrdbb6e0fa-aa: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1450 qdisc noqueue state UP group default qlen 1000
   link/ether fe:16:3e:2d:0f:fe brd ff:ff:ff:ff:ff:ff
22: qvodb6e0fa-aa@qvodb6e0fa-aa: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1450 qdisc noqueue master ovs-system state UP group default qlen 1000
   link/ether 4a:5a:5f:38:bd:48 brd ff:ff:ff:ff:ff:ff
   inet6 fe80::485a:5fff:fe38:bd48/64 scope link
       valid_lft forever preferred_lft forever
23: qvodb6e0fa-aa@qvodb6e0fa-aa: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1450 qdisc noqueue master qbrdbb6e0fa-aa state UP group default qlen 1000
   link/ether fe:d5:5a:40:05:0b brd ff:ff:ff:ff:ff:ff
   inet6 fe80::fcd5:5aff:fe40:50b/64 scope link
       valid_lft forever preferred_lft forever
24: tapdbb6e0fa-aa: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1450 qdisc pfifo_fast master qbrdbb6e0fa-aa state UNKNOWN group default qlen 1000
   link/ether fe:16:3e:2d:0f:fe brd ff:ff:ff:ff:ff:ff
   inet6 fe80::fc16:3eff:fe2d:ffe/64 scope link
       valid_lft forever preferred_lft forever

```

Ilustración 77: Configuración - Comprobar tarjetas de red

## D.2.7. Acceder a OpenStack

Para acceder a OpenStack podemos hacer uso del servicio Horizon, comúnmente denominado DashBoard, que será nuestro acceso por medio de una interfaz web, o por medio de la línea de comandos CLI. Para ambos tenemos que utilizar la información del usuario (usuario y contraseña) con el que queremos iniciar sesión.

La información del usuario OpenStack se encontrará en la carpeta raíz del usuario que ha instalado la plataforma en el sistema CentOS, en este caso, root.

Inicialmente, una vez hemos finalizado la instalación de OpenStack, tendremos como único usuario el administrador de la plataforma y la propia plataforma te crea una carpeta con toda la información de este usuario en la línea de comandos CLI.

```
cd /root
vi keystone_admin
```

Ilustración 78: Acceso - Fichero Usuarios

Al abrir ese fichero nos encontramos con la siguiente información:

```
unset OS_SERVICE_TOKEN
export OS_USERNAME=admin
export OS_PASSWORD=0ed4c179001f438e
export OS_AUTH_URL=http://192.168.1.100:5000/v3
export PS1='\u@\h \W(keystone_admin)]\$ '

export OS_PROJECT_NAME=admin
export OS_USER_DOMAIN_NAME=Default
export OS_PROJECT_DOMAIN_NAME=Default
export OS_IDENTITY_API_VERSION=3
```

Ilustración 79: Acceso - Usuario y Contraseña Dashboard

Como bien podemos ver en la ilustración anterior, se nos indica el nombre del usuario y su clave de acceso.

Si queremos acceder con este usuario por la línea de comandos, tenemos que hacer uso del comando “source” pasándole todo lo que se encuentra en el fichero correspondiente al usuario en cuestión (usuario, contraseña, proyecto que tiene asociado, etc.). Además, de esta manera seremos capaces de hacer uso de los comandos determinados por OpenStack.

```
[root@openstacklocalhost ~]# source keystone_admin
[root@openstacklocalhost ~(keystone_admin)]# openstack flavor list
+-----+-----+-----+-----+-----+-----+-----+
| ID | Name      | RAM  | Disk | Ephemeral | VCPUs | Is Public |
+-----+-----+-----+-----+-----+-----+-----+
| 1  | ml.tiny   | 512  | 1    | 0          | 1     | True      |
| 2  | ml.small  | 2048 | 20   | 0          | 1     | True      |
| 3  | ml.medium | 4096 | 40   | 0          | 2     | True      |
| 4  | ml.large  | 8192 | 80   | 0          | 4     | True      |
| 5  | ml.xlarge | 16384| 160  | 0          | 8     | True      |
+-----+-----+-----+-----+-----+-----+-----+
```

Ilustración 80: Acceso - CLI y comando OpenStack

En este caso, hemos utilizado el comando “openstack flavor list” para reflejar la lista de “sabores” que tiene disponible el usuario administrador.

Si en cambio, lo que deseamos es acceder a la plataforma OpenStack mediante la interfaz web (Dashboard); reflejaremos en el navegador la dirección IP donde se encuentra alojado nuestra plataforma de OpenStack (en mi caso: 192.168.1.100), e indicaremos el usuario y la contraseña correspondiente.

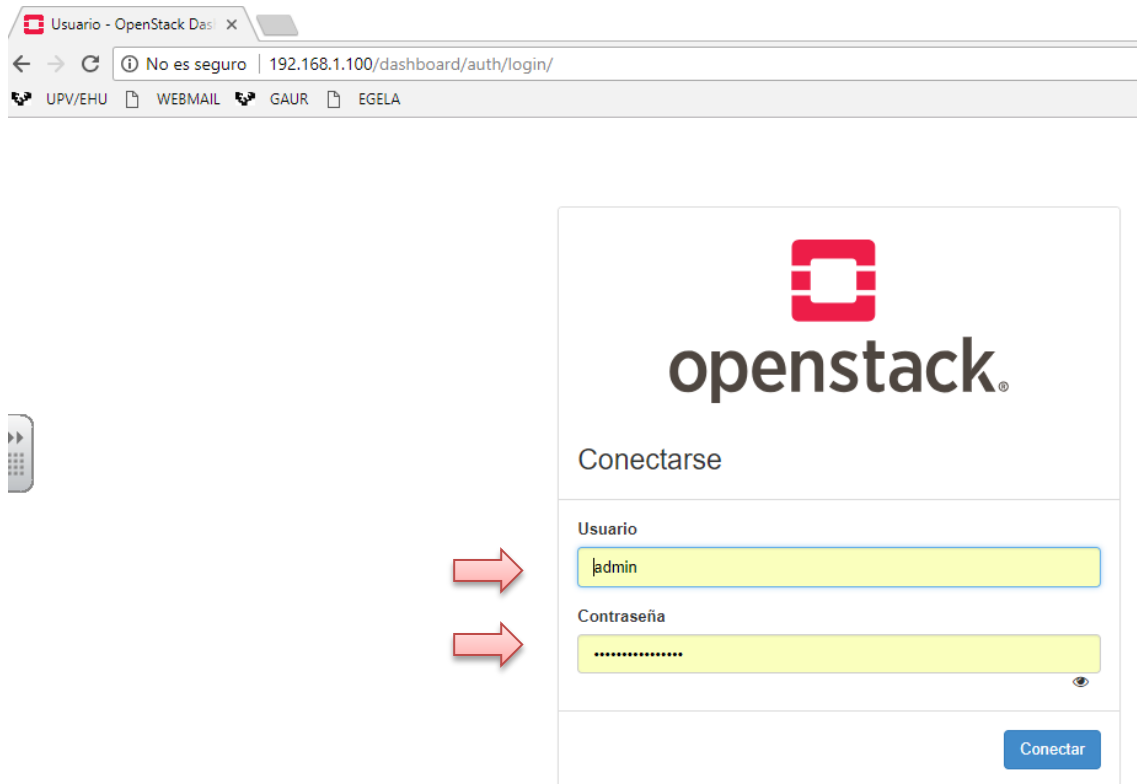


Ilustración 81: Acceso – Dashboard

Con todo ello ya tenemos nuestra instalación de OpenStack concluida. Como se ha podido apreciar, no es una instalación dificultosa pero es importante seguir los pasos previos a la instalación porque si no luego pueden surgir problemas. Lo único que queda pendiente será la tarea de administración, conociendo así además diferentes conceptos de especial importancia dentro de la plataforma.

### D.3. Soporte de ayuda

Durante la instalación de OpenStack podemos encontrarnos con diferentes dificultades pudiendo no alcanzar el resultado deseado si no son tratadas de manera correcta. A continuación, detallo las inconvenientes que yo he encontrado y la solución que he hecho uso.

#### 1. Estado de la máquina

Una vez que hemos realizado la instalación del sistema operativo<sup>27</sup> CentOS, o hemos hecho cualquier configuración sobre el sistema, y deseamos no hacer más uso del sistema; nuestra predilección es apagar la máquina. En este caso, si optamos por esa solución e iniciamos a continuación la máquina, parece que todo lo que hemos realizado lo hemos perdido al solicitarnos realizar de nuevo la instalación del sistema. ¿Pero por qué motivo sucede ello? Porque aún tenemos el sistema de instalación, es decir, el disco de la unidad virtual.

Ante esta situación, cuando iniciemos la máquina en el margen superior tenemos que dirigirnos a la siguiente opción que visualiza la próxima imagen.

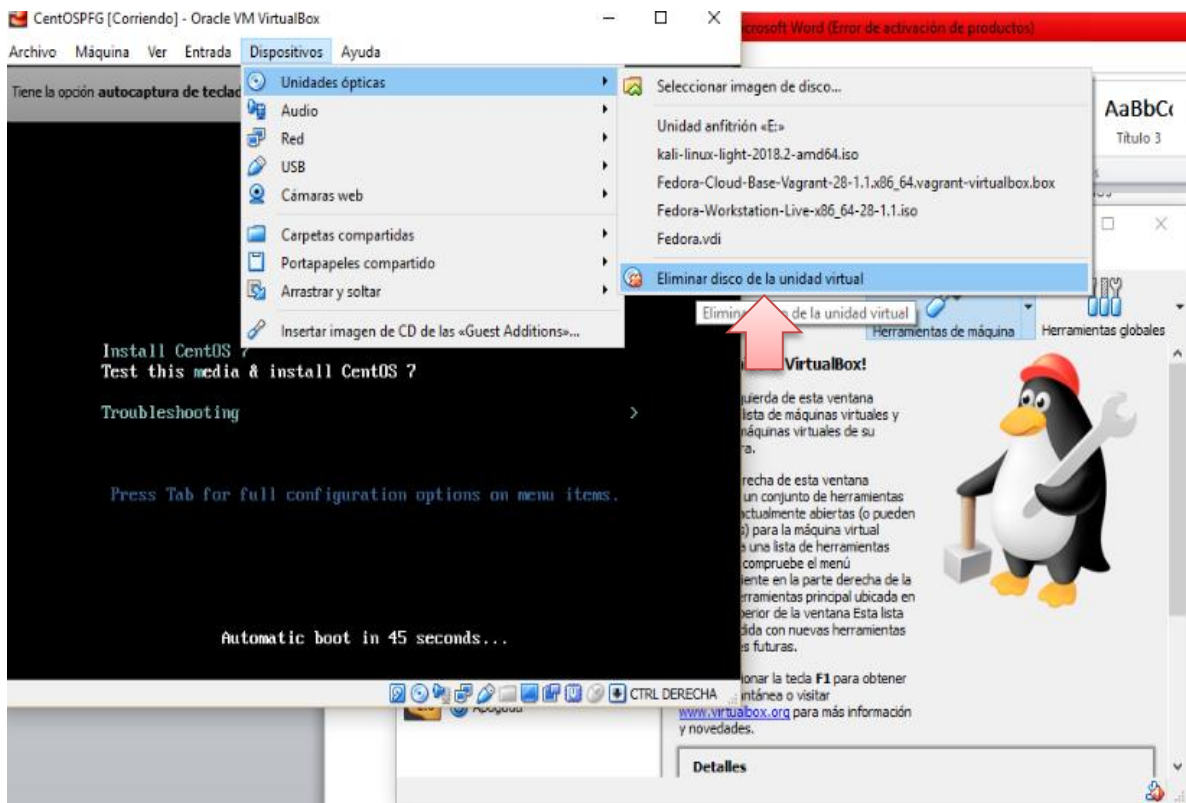


Ilustración 82: Ayuda - Estado de la máquina

Una vez se ha seleccionado esa opción y eliminado el disco de la unidad virtual, apagamos la máquina. Lo siguiente será volver a iniciarla, obteniendo en este caso el resultado deseado.

## 2. Iniciar sesión con la aplicación PuTTY

Una vez que hemos instalado nuestro sistema operativo<sup>27</sup> CentOS y verificado que se ha configurado correctamente la dirección IP de la máquina (comando “ping”<sup>9</sup>); vamos a iniciar sesión con la aplicación PuTTY. En muchos casos, cuando indicamos los parámetros necesarios para realizar una conexión segura (SSH) e iniciamos la sesión, la aplicación PuTTY no nos lo permite porque desconoce esa dirección. Pero, ¿cómo es posible si ya hemos verificado correctamente su configuración?

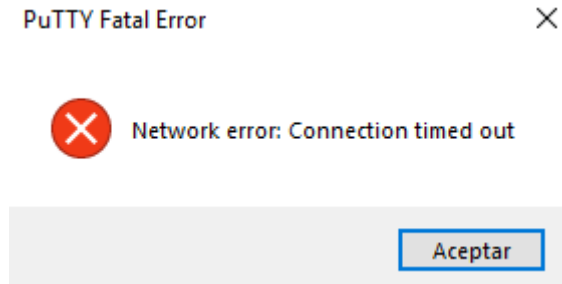


Ilustración 83: Ayuda - Problema PuTTY

Este error se encuentra provocado por los ficheros de configuración SSH de CentOS en la mayoría de los casos. En ambos ficheros se encuentran diferentes aspectos (puerto, etc), donde deberemos eliminar diferentes comentarios en cuestión para poder iniciar la sesión segura requerida.

```
⇒ [root@openstacklocalhost ssh]# pwd
/etc/ssh
[root@openstacklocalhost ssh]# ls
moduli          ssh_host_ecdsa_key      ssh_host_rsa_key
ssh_config      ssh_host_ecdsa_key.pub  ssh_host_rsa_key.pub
sshd_config     ssh_host_ed25519_key    ssh_known_hosts
sshd_config.rpmnew ssh_host_ed25519_key.pub
```

Ilustración 84: Ayuda - Ficheros de configuración SSH

```
⇒ # IdentityFile ~/.ssh/id_ed25519
Port 22
Protocol 2
# Cipher 3des
```

Ilustración 85: Ayuda - Ficheros de configuración ssh\_config

```
⇒ Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
```

Ilustración 86: Ayuda - Ficheros de configuración sshd\_config

Una vez realizados estos cambios, podemos volver a intentar realizar la conexión segura. En caso de continuar el problema, se recomienda indagar información en la web.

### 3. Copia de seguridad

Finalmente, es recomendable realizar copias de seguridad de nuestras máquinas virtuales. Esto es debido a que si realizamos cualquier alteración sobre las características o propiedades de la máquina que provoquen un estado no deseado, o perdemos la propia máquina (entre otros casos que pueden suceder); poder recuperarla en un estado que consideremos adecuado.

En la aplicación VirtualBox las máquinas virtuales quedan almacenadas dentro de una carpeta de nombre "VirtualBox VMs". La copia de seguridad se puede almacenar en un dispositivo externo, en la nube o incluso en alguna otra ubicación dentro de la computadora con la que estamos trabajando.

Otra opción para realizar copias de seguridad es la que te proporciona la propia aplicación VirtualBox por medio de las instantáneas. Este método guarda el estado de la máquina en un momento dado pudiendo ser recuperada. Veamos el proceso a seguir para la creación de una instantánea de nuestra máquina virtual:

1. Seleccionando nuestra máquina nos dirigimos al apartado de "Herramientas de máquina" más en concreto a "Instantáneas".

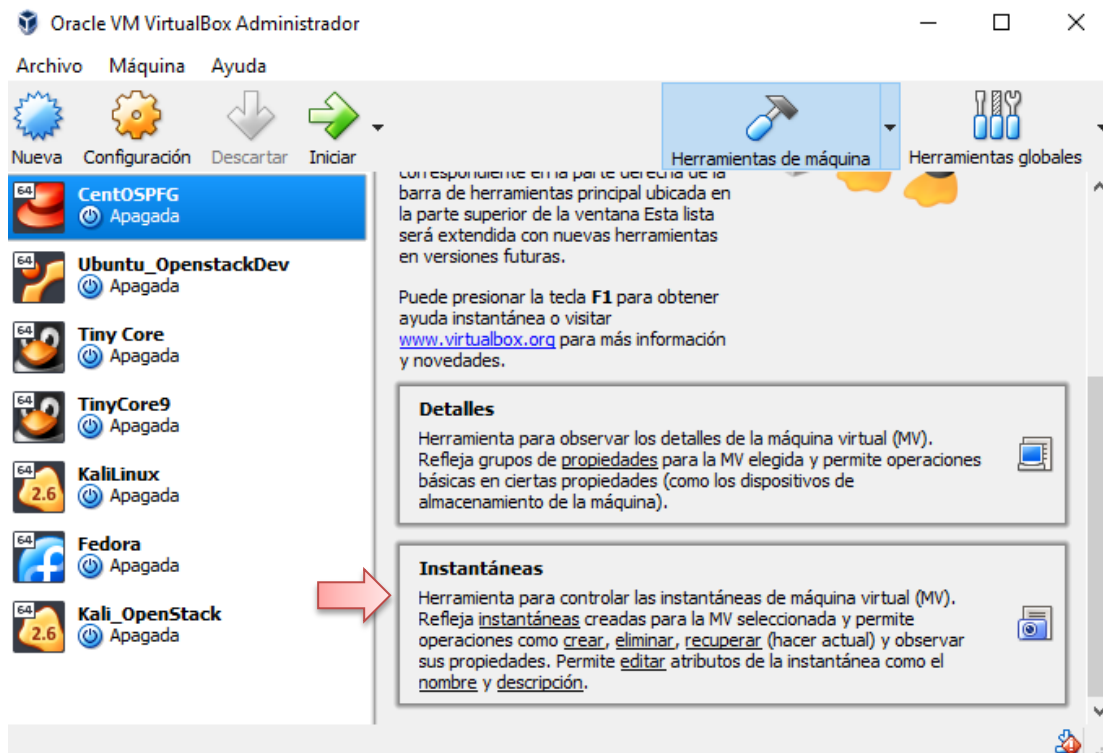


Ilustración 87: Ayuda - Opción instantánea

2. Lo siguiente será “tomar” la instantánea.

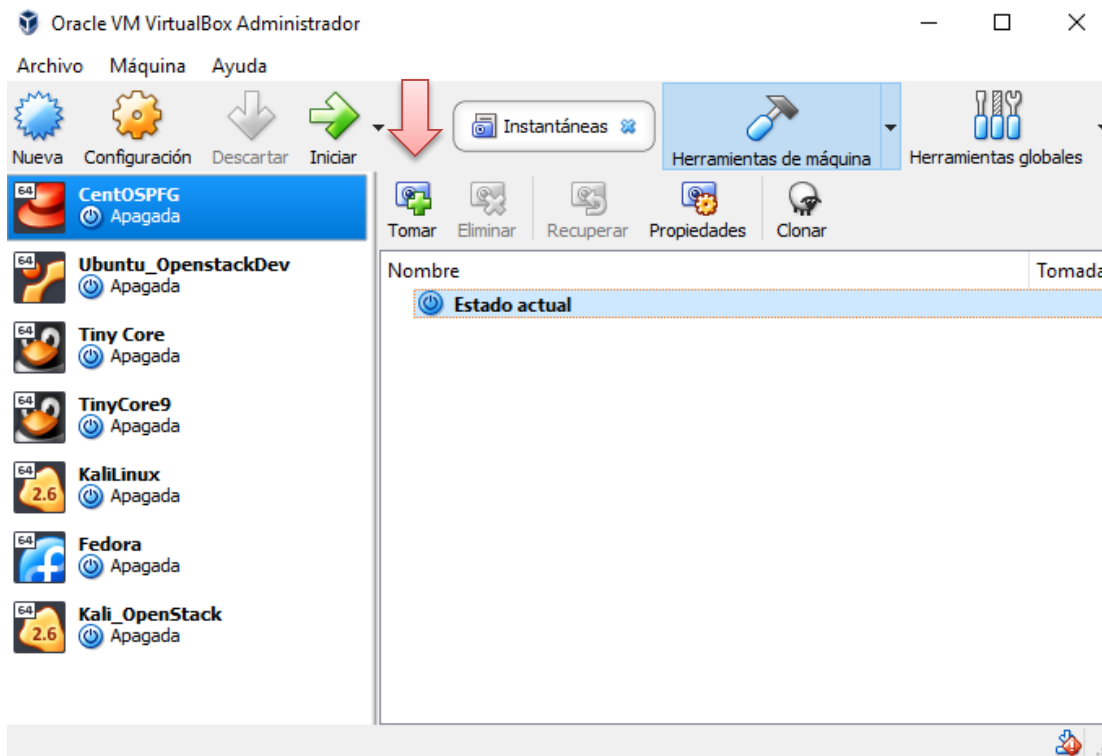


Ilustración 88: Ayuda - Tomar instantáneas

3. Definimos el nombre con el que vamos a dotar a la instantánea y si deseamos, una breve descripción de la misma.

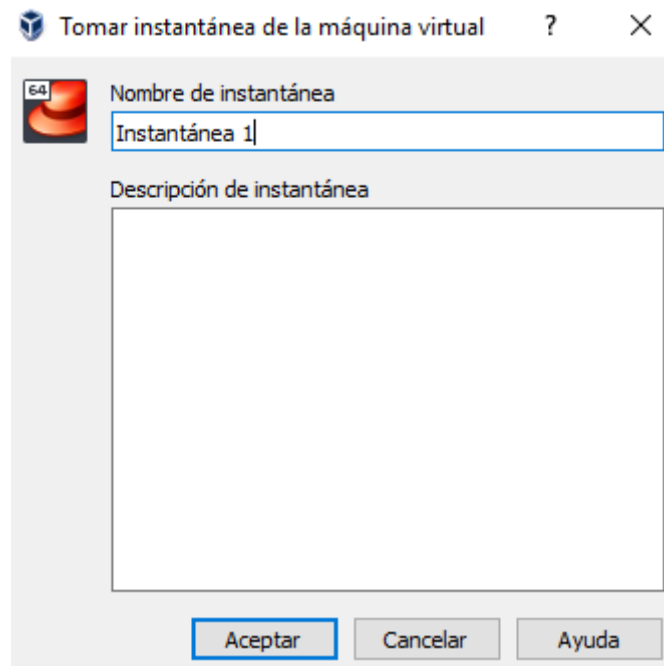


Ilustración 89: Ayuda - Nombre y descripción de instantánea

4. Observamos que se ha creado correctamente la instantánea y las posibilidades que ofrece. Esto es desde poder eliminarla hasta clonarla, pudiendo recuperarla, es decir, establecer esa máquina virtual en ese estado en el que estaba cuando fue creada.

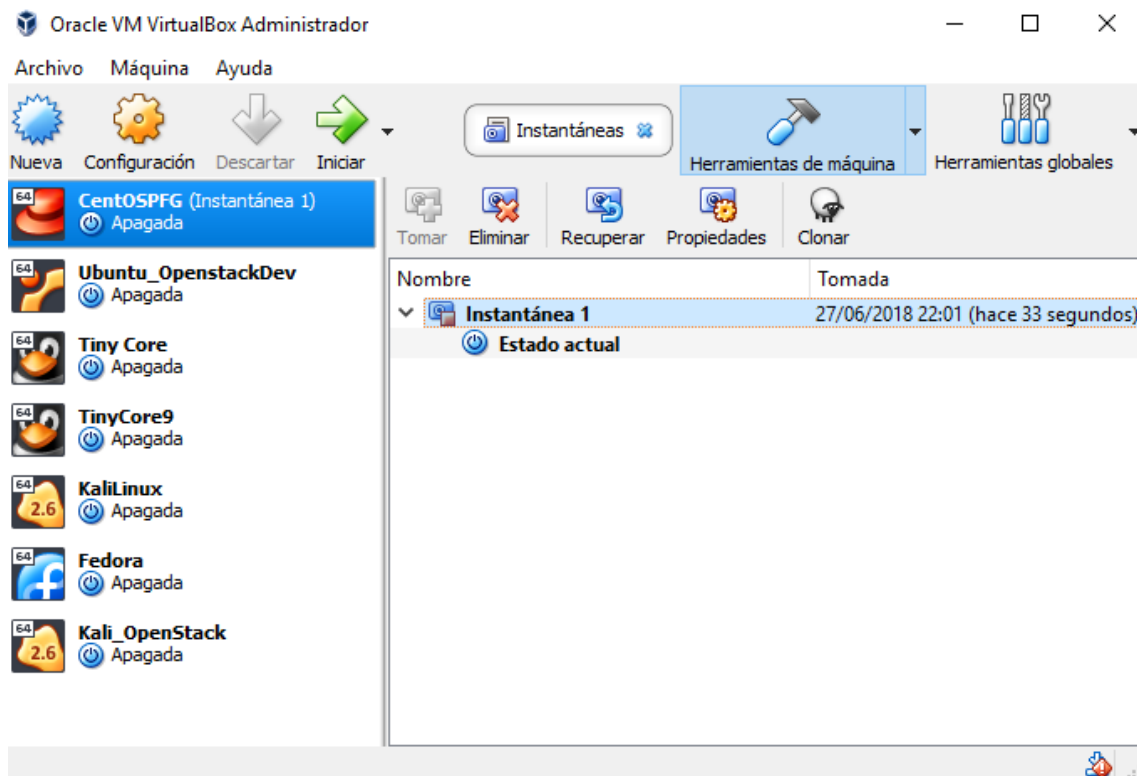


Ilustración 90: Ayuda - Opciones instantáneas



## E. Administración básica de la Plataforma OpenStack

### E.1. Introducción

Antes de mencionar el estudio realizado para poder llevar a cabo una administración básica de la plataforma OpenStack, es recomendable volver a recordar los distintos componentes de los que se constituye la plataforma:

- ❖ Nova (Cómputo): Es el módulo que controla la gestión y ejecución de las máquinas virtuales pudiendo trabajar con muchos hipervisores.
- ❖ Neutron (Red): Instala y gestiona toda la infraestructura de red.
- ❖ Horizon (Dashboard): Interfaz de usuario.
- ❖ Glance (Imágenes): Gestiona las imágenes de una instancia.
- ❖ Cinder (Almacenamiento en bloque): Nodo que controla los bloques de almacenamiento.
- ❖ Keystone: Gestiona los usuarios.
- ❖ Swift: Permite el almacenamiento de objetos.
- ❖ Otros componentes: Heat (Orquestación), Ceilometer, etc.

Por medio de este apartado se pretenderá explicar la estructura de la que está compuesta OpenStack y como poder administrarla de una forma básica. Destacar que la administración se realiza de la misma forma indistintamente de la arquitectura proporcionada por un distribuidor en cuestión (sigue un estándar). Esta gestión se puede llevar a cabo haciendo uso de la interfaz de usuario (servicio Horizon) o por medio de la línea de comandos, comúnmente denominada CLI.

En este apartado hablaremos además de muchos conceptos que es aconsejable mencionar para nuestro buen entendimiento:

- ❖ Imagen: Una imagen de máquina virtual, o de forma abreviada una imagen, es un fichero que contiene un disco virtual con un sistema operativo<sup>27</sup> con la configuración mínima para poder operar adecuadamente dentro de una nube de infraestructura. Existen muchos formatos de imagen, siendo los más conocidos los relacionados con los hipervisores más populares: VMDK para VMware, qcow2 para QEMU/KVM, etc.
- ❖ Instancia: Una instancia es una imagen que se lanza con un conjunto de requisitos determinados. Para lanzar una instancia es necesario una red, una imagen, y un conjunto de recursos como son la CPU, la memoria y el disco entre otros.

Instancia = Plantilla de Sistema Operativo + CPU/RAM + Disco

Por lo tanto, una instancia es un servidor virtual que se ejecuta dentro de la nube de infraestructura y que se basa en una imagen.

- ❖ Flavors (Sabores/Tipo de instancia): Hardware virtual que utilizaremos para poder lanzar una instancia. Define las características de la instancia: cantidad de memoria RAM, tamaño del disco duro del sistema operativo<sup>27</sup>, etc.

- ❖ IP fija: Dirección IP interna de la instancia y que se define en el momento de la creación de la misma. La dirección IP fija se utiliza para la comunicación interna de la instancia y su valor no varía durante la vida de la instancia, de ahí que reciba el nombre de IP fija.
- ❖ IP flotante: Dirección IP que se asigna a una instancia después de crearla y que se utiliza para la comunicación desde el exterior. No todas las instancias tiene que tener una dirección IP flotante.
- ❖ Grupo de seguridad (Cortafuegos): Conjunto de reglas que actúan como cortafuegos a nivel de la instancia.
- ❖ Contenedor: Equivalente a un directorio en un sistema de ficheros. Sirve para organizar objetos, pero tiene una diferencia fundamental con el directorio: no se puede anidar contenedores.
- ❖ Objeto: Elemento que se almacena. Suele estar relacionado con un fichero o archivo, disponiendo además de una serie de metadatos<sup>20</sup>.
- ❖ ACL: Utilizada para gestionar los permisos que otros usuarios pueden tener sobre cada contenedor, no sobre cada objeto.

## E.2. Conceptos de Administración

### E.2.1. Administración con el servicio Horizon

Si accedemos como usuario administrador a la interfaz web proporcionada por OpenStack (Dashboard/ servicio Horizon), tenemos a nuestra disposición una serie de funciones que nos permiten una administración básica de nuestra infraestructura. Cada una de estas funciones podremos tenerla o no en base al rol, al papel, que tengamos dentro de la plataforma OpenStack para un proyecto en cuestión.

#### E.2.1.1. Apartado “Sistema”

Veamos las distintas opciones que podemos encontrar en esta sección:

- ❖ Vista general: Obtenemos una estadística general clasificada por proyectos.
- ❖ Hipervisores: Información sobre los hipervisores que estamos utilizando para la virtualización de las máquinas.
- ❖ Agregados de host: La agregación de hosts divide una zona de disponibilidad en unidades mediante la agrupación de los hosts. En esta pestaña podemos crear nuevos agregados de host y obtener información sobre las zonas de disponibilidad.
- ❖ Instancias: Se accede a la gestión de todas las instancias del proyecto en cuestión.
- ❖ Volúmenes: Se accede a la gestión de todos los volúmenes del proyecto en particular.
- ❖ Sabores: Podemos gestionar los sabores definidos.  
Podemos crear además nuevos sabores teniendo en consideración que, si pretendemos posteriormente cambiar alguna de sus características, deberemos crear uno nuevo. También, dispone de una opción donde podemos definir qué proyectos van a tener acceso a ese sabor que estamos definiendo.
- ❖ Imágenes: Nos permite gestionar las imágenes a las que tienen acceso los usuarios.
- ❖ Redes: Podemos gestionar todas las redes internas de todos los proyectos. Además, tenemos acceso a la red externa que hemos podido definir, pudiendo modificarla si es necesario.
- ❖ Routers: Nos permite gestionar los distintos routers que se han creado en los proyectos. Con un router podemos tener acceso a una instancia de manera externa.
- ❖ Predeterminados: Desde esta opción podemos ver y modificar las cuotas predeterminadas (límites máximos). Por cada proyecto se pueden alterar estas cuotas.
- ❖ Definición de los metadatos<sup>20</sup>: Nos permite importar espacios de nombre y ver información sobre los metadatos<sup>20</sup>.
- ❖ Información del Sistema: Puedes acceder a la siguientes información:
  - ◆ Servicios: Ver la lista de servicios.
  - ◆ Servicios de computación: Ver los servicios de computación instalados.
  - ◆ Servicios de almacenamiento de datos: Ver los servicios de almacenamiento de datos.
  - ◆ Agentes de red: Ver los distintos servicios de red instalados.

## Administrador

### Sistema

Vista general

Hipervisores

Agregados de host

Instancias

Volúmenes

Sabores

Imágenes

### Redes

Routers

IPs flotantes

Predeterminados

Definiciones de los  
metadatos

Información del Sistema

Ilustración 91: Dashboard/Servicio Horizon - Apartado "Sistema"

## E.2.1.2. Apartado "Identity"

En este apartado podemos gestionar los usuarios, grupos, proyectos y roles definidos en el sistema.

- ❖ **Proyectos:** Nos permite gestionar los proyectos definidos en el sistema. Podemos crear nuevos proyectos, administrar los usuarios que forman parte del proyecto, especificar las cuotas que van a afectar a un proyecto determinado, entre otras cuestiones.
- ❖ **Usuarios:** Desde esta opción gestionamos los usuarios del sistema. Podemos editarlos y crear nuevos usuarios.

### Identity

Proyectos

Usuarios

Grupos

Roles

Ilustración 92: Dashboard/Servicio Horizon - Apartado "Identity"

## E.2.2. Administración en la línea de comandos (CLI)

Hasta ahora hemos hablado de un entorno web para gestionar nuestra infraestructura ya que resulta más cómodo al principio para comprender conceptos; pero en situaciones más reales es muy habitual utilizar la línea de comandos o incluso programar con un lenguaje de alto nivel la gestión de la infraestructura, principalmente cuando tenemos que realizar tareas repetitivas o automáticas.

OpenStack es un proyecto modular formado por diferentes componentes independientes, que ofrecen a los usuarios y al resto de elementos de OpenStack una API REST<sup>4</sup> con la cual interactuar. Actualmente la versión oficial de OpenStack incluye unos 15 componentes diferentes, aunque durante esta documentación y este proyecto nos vamos a centrar exclusivamente en los elementos principales.

Nombre	Componente	Función
Nova	OpenStack Compute	Gestión de las instancias.
Glance	OpenStack Image	Gestión de las imágenes.
Keystone	OpenStack Identity	Gestión de usuario, permisos y proyectos.
Cinder	OpenStack Block Storage	Gestión de volúmenes.
Neutron	OpenStack Networking	Gestión de las redes.
Swift	OpenStack Swift	Gestión de objetos.

Tabla 13: OpenStackCLI - Componentes principales

Cada componente de OpenStack tiene su propio cliente en la línea de comandos con todas las funcionalidades, por lo que en un principio tendríamos que instalar los correspondientes a los seis elementos anteriores. Al estar escritas en Python se recomienda instalarlas en un equipo con alguna distribución de GNU/Linux.

Como se ha mencionado en el párrafo anterior, cada componente de OpenStack proporciona su propio cliente de línea de comandos y lamentablemente no siempre la forma de utilizarlo es igual, lo que crea dificultad en los usuarios y provoca errores. Al ir aumentando el número de componentes de OpenStack este problema no hace más que acrecentarse por lo que el proyecto decidió desarrollar un cliente unificado que denominó OpenStackClient (OSC). OpenStackClient (OSC) reúne todo el juego de comandos para las aplicaciones de cálculo, identidad, imagen, almacenamiento de objetos y almacenamiento de bloques en un solo lenguaje con una estructura de comandos uniforme. El nuevo juego de comandos usa “openstack” como comando principal en lugar de los nombres de componentes que se usaron en versiones anteriores de OpenStack.

Por tanto, la gestión en la línea de comandos puede llevarse a cabo haciendo uso de los comandos propios del componente en cuestión o por medio de OpenStackClient (OSC). En todo caso, deberemos iniciar sesión desde la línea de comandos en donde se permitirán realizar más o menos acciones en función del usuario. Para iniciar sesión, por ejemplo, como usuario administrador haríamos uso del comando “source” más el fichero que contenga toda la información relativa a dicho usuario.

```
[root@openstacklocalhost ~]# source keystoneadmin
```

Ilustración 93: Cargar credenciales de administrador

### E.2.3. Conclusiones

Llegados a este punto nos podemos preguntar, ¿administrar un entorno de OpenStack puede ser sencillo? Puede que sí, siempre y cuando no haya problemas y conozcamos el conjunto de comandos a utilizar en el caso de utilizar CLI. Pero, si nos encontramos en un entorno de producción con miles de máquinas virtuales, ¿dejaríamos la administración de dicho entorno a cualquier administrador de sistemas?

Desde mi punto de vista, para administrar OpenStack de forma correcta, hay que conocer el funcionamiento de cada uno de los servicios que componen el entorno, no basta con realizar la instalación con un asistente. Por lo tanto, en el siguiente apartado iremos conociendo en detalle cada uno de los elementos que constituyen esta plataforma y que permiten a OpenStack su funcionamiento.

## E.3. Descripción de los Componentes

### E.3.1. Introducción

Observemos en la siguiente ilustración la arquitectura de los servicios principales de OpenStack.

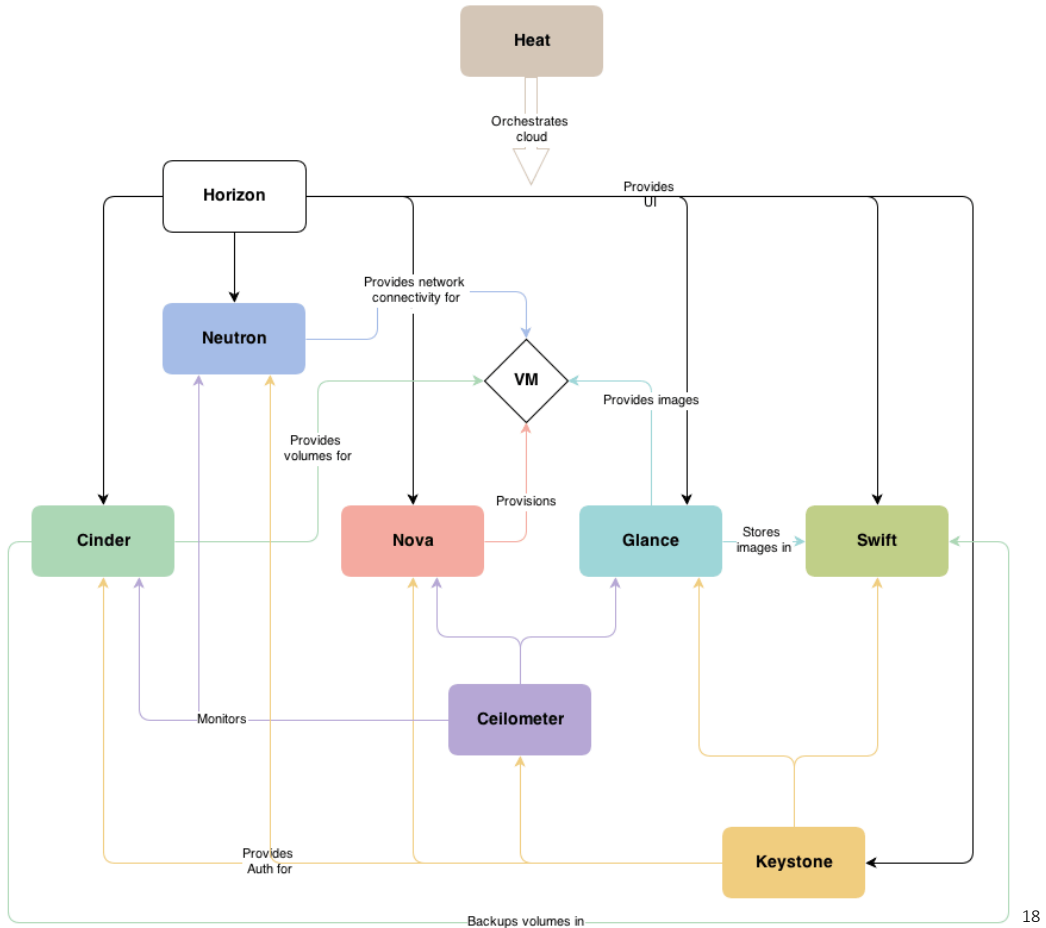


Ilustración 94: Arquitectura de servicio de OpenStack

Por medio del presente apartado nos centraremos en aquellos componentes que se consideran más importantes dentro de la infraestructura que define a OpenStack. Sobre cada uno de estos servicios aprenderemos cuál es su tarea dentro de la plataforma, así como los elementos de los que se encuentra constituido entre otras cuestiones a tener en consideración.

### E.3.2. Componente KeyStone

Keystone es el servicio de identidad que utiliza OpenStack para la autenticación. Keystone nos va a permitir la autenticación y autorización entre los servicios, es decir, es el módulo encargado de gestionar tanto a los usuarios como sus permisos dentro de los proyectos, y se comunica con los otros componentes para decirles si el usuario puede o no ejecutar la acción que está solicitando. Puede ser integrado con el sistema de autenticación de LDAP, además de poder ser usado a través de nombre de usuario y contraseña estándar, sistemas basados en tokens e inicio de sesión, entre otros. Para entender cómo funciona Keystone o Servicio de Identidad debemos tener en consideración los siguientes componentes:

- ❖ Usuario: Pueden ser personas, sistemas o servicios que usen la nube de OpenStack. Keystone será el encargado de validar las peticiones de esté al disponer cada uno de ellos de un “login”.
- ❖ Grupo: Se trata de la agrupación de un conjunto de usuarios que pueden compartir aspectos concretos.
- ❖ Credenciales: Es el dato que va a usar el usuario para su autenticación. Puede ser un nombre de usuario y contraseña, un token, una clave de la API<sup>3</sup>, etc.
- ❖ Autenticación: Es el acto de confirmar la identidad del usuario usando las credenciales provistas. Una vez validada la autenticación se provee de un token para ser usado en las peticiones del usuario. De esta forma, no necesita autenticarse de nuevo.
- ❖ Token: Es un bit de texto usado para acceder a los recursos. Es generado cuando se valida la autenticación. Cada token tiene un alcance el cual nos permite saber a qué recursos puede acceder un usuario.
- ❖ Tenant: Es el contenedor usado para agrupar y aislar los recursos. Estos pueden ser clientes, organizaciones, cuentas o proyectos.
- ❖ Endpoint: Es el direccionamiento IP o URL para acceder a los servicios.
- ❖ Roles: Para especificar o limitar las operaciones de usuario Keystone usa roles, los cuales contienen un grupo de privilegios para determinar el alcance del usuario.

Cabe destacar que un usuario puede pertenecer a más de un proyecto determinado, disponiendo a su vez de diferentes roles por cada proyecto. De igual manera sucede con los grupos. Además, si un grupo o usuario no tienen un rol asignado en un proyecto, no podrán llevar a cabo ninguna acción sobre dicho proyecto. OpenStack dispone de unos roles por defecto: “ResellerAdmin”, “OperatorSwift”, “\_member\_” y “Admin”; aunque siempre podemos definir los nuestros.



### E.3.3. Componente Glance

Glance es el servicio de OpenStack el cual permite registrar, descubrir y recuperar imágenes de máquinas virtuales para ser usadas en nuestro entorno de OpenStack.

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

- ❖ Encolada (Queued): Quiere decir que hemos registrado una imagen nueva, se ha asignado un identificador pero aún no ha sido subida al servicio Glance.
- ❖ Guardando (Saving): La imagen esta subida a Glance.
- ❖ Activa (Active): La imagen está totalmente disponible para ser usada en Glance.
- ❖ Abortada (Killed): Se produce cuando existe un error durante la subida de una imagen y ésta no es legible.
- ❖ Borrada (Deleted): En este estado aún disponemos de la información de la imagen pero no podemos usarla ya que se encuentra eliminada.
- ❖ Pendiente de borrado (Pending\_delete): Estamos a punto de eliminar la imagen pero aún podemos recuperarla.

Al igual que las imágenes, en Glance las tareas también disponen de estados y pueden ser los siguientes:

- ❖ Pendiente (Pending): Al igual que la imagen, la tarea aún no ha sido procesada.
- ❖ Procesado (Processing): En este estado la tarea ha comenzado a ejecutarse pero aún no ha finalizado.
- ❖ Éxito (Success): La tarea ha sido completada de forma correcta.
- ❖ Fallo (Failure): Durante la ejecución de la tarea se ha producido algún error el cual impide que se pueda continuar.

Si añadimos nuevas imágenes, debemos especificar tanto el formato de disco como el formato del contenedor. Estos pueden ser los siguientes:

- ❖ Los formatos de disco pueden ser: raw, vhd, vmdk, vdi, iso, qcow2, aki, asi, ami.
- ❖ En los contenedores dispondremos de los metadatos<sup>20</sup> de las máquinas virtual y estos pueden ser: bare, ovf, aki, ari, ami, ova.

Siempre que no sepamos el tipo de contenedor especificamos “bare”.

### E.3.4. Componente Nova

El componente Nova es el servicio que trabaja con los hipervisores y es el responsable de poner en marcha y gestionar las máquinas virtuales. Nova no es un hipervisor, sino el gestor de recursos de hipervisores tales como Xen, KVM<sup>19</sup>, VMware, etc.

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

- ❖ Servidor de API: Nos va a permitir la comunicación de los hipervisores con nuestro entorno.
- ❖ Colas de mensajes: Se encarga de la comunicación entre componentes.
- ❖ Worker: Es el administrador de instancias que nos va a permitir crear, terminar, reiniciar instancias, añadir o quitar volúmenes; entre otros aspectos.
- ❖ Controlador de red: Con el podremos provisionar recurso de red como por ejemplo la dirección IP, VLAN<sup>30</sup>, router, etc.

#### E.3.4.1. Pares de claves

Puesto que nos encontramos en un entorno de recursos compartidos y es bastante habitual que la misma imagen puede ser instanciada por distintos usuarios, no es razonable que la imagen tenga una contraseña de acceso común que permitiera a cualquiera poder acceder a la instancia de otro usuario en el momento que se crease. La solución más común en Infraestructuras como Servicio (IaaS) <sup>18</sup> es utilizar pares de clave pública/privada para configurar de forma segura el acceso a las instancias.

El funcionamiento es sencillo. El usuario almacena la o las claves públicas que vaya a utilizar en la nube de infraestructura y cuando se crea la instancia se selecciona la clave pública que desea utilizar para acceder a la misma. Durante el proceso de instanciación, el sistema “inyecta” la clave pública en el usuario correspondiente del sistema, de manera que sólo el poseedor de la correspondiente clave privada podrá acceder al mismo.

El sistema nos facilita dos opciones: subir nuestra propia clave pública o bien solicitar al sistema la creación de un par de claves públicas/privadas. La primera opción es más segura y recomendable, ya que nadie tiene acceso ni gestiona en ningún caso a una clave privada; pero la segunda opción es más sencilla y puede ser aceptable para pruebas o entornos donde la seguridad no es un aspecto fundamental.

### E.3.4.2. Grupos de seguridad o Cortafuegos de la instancia

Toda instancia que se crea en OpenStack tiene asociado uno o varios grupos de seguridad, que consisten en un conjunto de reglas que actúan como un cortafuegos. Es importante destacar que este cortafuegos se aplica a nivel de la interfaz de red y por lo tanto las reglas también afectan a las instancias que estén en el mismo segmento de red.

La configuración por defecto más habitual en todos los entornos de OpenStack es que la instancia tiene abierto todo el tráfico de salida, pero cerrado todo el tráfico de entrada. A esta configuración inicial es necesario añadirle la apertura al tráfico entrante de todos los servicios que queramos que estén accesibles desde el exterior, en particular aquellos que nos permitan acceder a ella para su gestión.

### E.3.4.3. Configuración de red

Finalmente cabe mencionar, que cuando se crea una instancia se le asigna por DHCP una dirección IP que es la misma durante toda la vida de la instancia, por lo que paradójicamente recibe el nombre de IP fija. Esta dirección IP permite que la instancia puede conectarse a otros equipos de su propia red o incluso de otras redes internas del cloud; pero si queremos acceder a la instancia desde el exterior es necesario asociar a la instancia lo que se denomina una IP flotante. Esta asignación nos la va a permitir el servicio Nova.

## E.3.5. Componente Cinder

OpenStack Block Storage (Cinder) es un software de código abierto<sup>8</sup> diseñado para crear y administrar un servicio que proporciona almacenamiento persistente de datos para aplicaciones de computación en la nube. Es decir, este servicio proporciona y gestiona dispositivos de bloques conocidos como volúmenes. Entonces nos podemos preguntar, ¿Qué se entiende por volumen en OpenStack?

Un volumen es un dispositivo de bloques que se puede asociar y desasociar de una instancia cuando se desee. Se utiliza para proporcionar almacenamiento permanente o independiente en la vida de una instancia. Todo volumen será gestionado por el componente Cinder y solamente se podrá asociar un volumen por instancia. Esta relación puede llevarse a cabo en el lanzamiento de la imagen o posteriormente.

Cinder también proporciona una interfaz de programación de aplicaciones (API<sup>3</sup>) de autoservicio para permitir a los usuarios solicitar y consumir recursos de almacenamiento sin tener que saber nada sobre su tipo o ubicación.

### E.3.6. Componente Swift

OpenStack Swift, también conocido como OpenStack Object Storage, es un software de código abierto<sup>8</sup> diseñado para administrar el almacenamiento de grandes cantidades de datos. Entre este almacenamiento se puede destacar las copias de seguridad o los archivos de datos no estructurados como documentos, contenido web estático, imágenes, archivos de vídeo, correo electrónico e imágenes de máquinas virtuales.

Por lo tanto, OpenStack Swift almacena datos como objetos binarios en el sistema de archivos subyacente del sistema operativo<sup>27</sup> del servidor. Cada objeto tiene metadatos<sup>20</sup> asociados como parte de los atributos extendidos del archivo.

Este componente puede ser por tanto utilizado por las organizaciones para almacenar gran cantidad de datos de manera eficiente<sup>12</sup> y segura.

### E.3.7. Componente Neutron

El servicio OpenStack Networking (Neutron) es un sistema para la gestión de redes y direcciones IP. Este componente de administración de red en OpenStack es uno de los más complejos debido a la cantidad de posibilidades de configuración que tenemos disponible. Es por esa razón, por la que se recomienda familiarizarse con los conceptos en los que se fundamenta este componente:

- ❖ Red: Red de dominio<sup>14</sup> de capa 2 equivalente a una VLAN<sup>30</sup> en el mundo de las redes físicas. Las redes externas solo pueden ser definidas por el administrador.
- ❖ Subred: Bloque de dirección IPv4 o IPv6 que se asignan a las máquinas virtuales que se conectan a ella. Se pueden asociar varias subredes con una sola red.
- ❖ Router: Dispositivo para conectar redes.
- ❖ Puerto: Un puerto es un punto de conexión para conectar un único dispositivo. En nuestro caso, los puertos virtuales que utilizemos estarán asociados a un switch o a un router.
- ❖ Enrutador: Un enrutador es un dispositivo que puede enrutar el tráfico entre diferentes subredes y redes. Cualquier subred en el mismo enrutador puede comunicarse entre sí sin una tabla de enrutamiento si los grupos de seguridad permiten la conexión.
- ❖ IP Fija: Dirección IP con la que se crea una instancia en una red y que se utiliza para comunicación interna. La dirección IP fija no cambia durante la vida de la instancia.
- ❖ IP Flotante: Dirección IP asociada a una instancia en un momento dado para poder acceder a ella desde fuera. Una IP flotante puede asignarse a otra instancia diferente cuando se estime oportuno.

Por lo tanto, el servicio Neutron nos va a permitir hacer uso de su API<sup>3</sup> a través de plugins y agentes capaces de conectar/desconectar puertos, provisionar redes y subredes, etc. Para expresar todos estos conceptos de mejor forma, echemos un vistazo a la siguiente ilustración, que muestra la conexión que existe entre el servicio de computación (Nova) y los diferentes nodos de red.

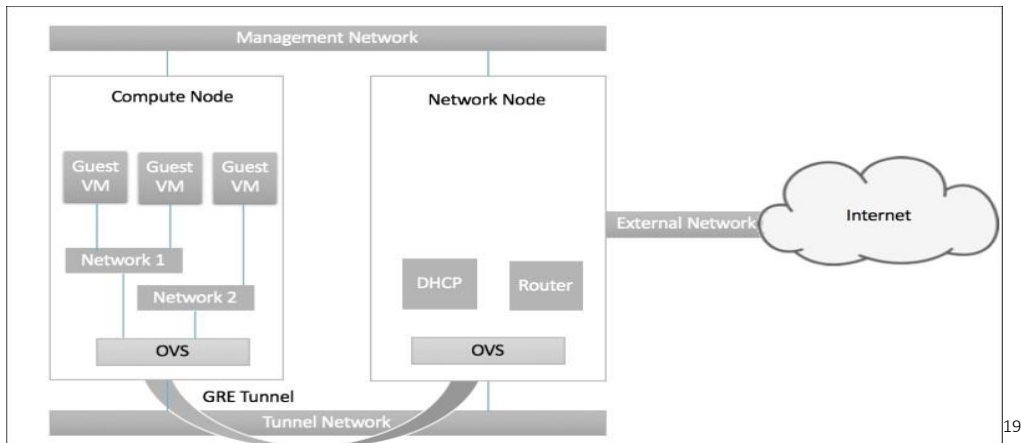


Ilustración 95: Conexión Servicio Nova - Servicio Neutron

Como bien podemos apreciar en la ilustración, ambos servicios se encuentran conectados a través de una red de túneles. Además, podemos afirmar que la tarea principal del servicio Neutron será la de conectar las diferentes redes o las redes a una red externa usando su interfaz externa y el puente OVS<sup>23</sup>. También podrá realizar otras funciones de este nivel, como el balanceo de carga o aspectos de seguridad.

Todo esto comentado se verá más claro a partir de la siguiente ilustración.

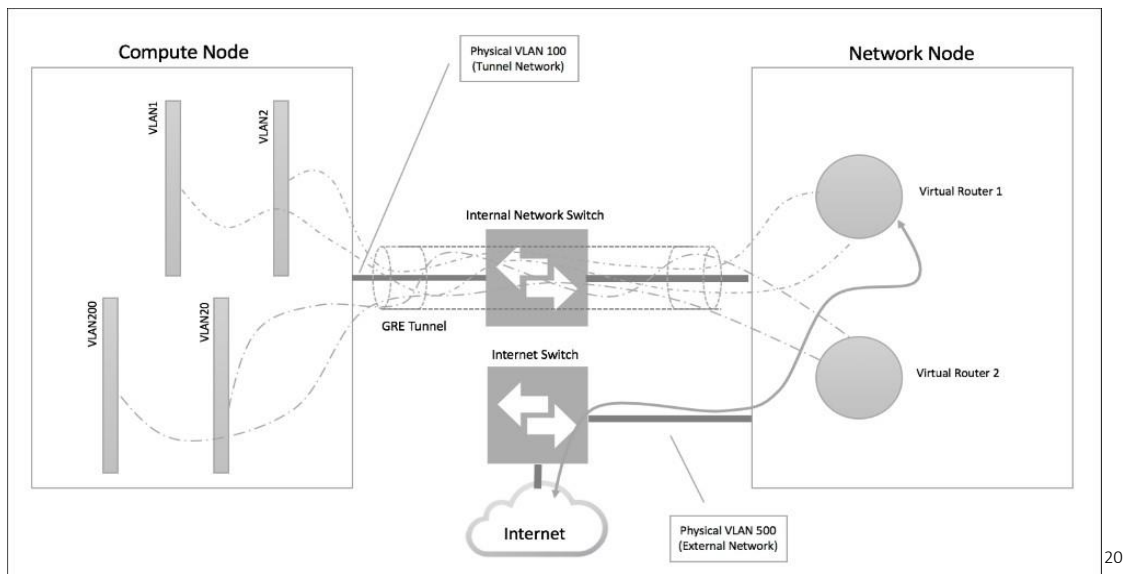


Ilustración 96: Conexión Servicio Nova - Servicio Neutron (2)

Como se puede ver en el diagrama, la red física no tiene absolutamente idea de la existencia de las diferentes VLAN<sup>30</sup>, que ahora se asignarán a diferentes inquilinos o al mismo inquilino para diferentes propósitos. Podemos crear tantas VLAN<sup>30</sup> como sea necesario y la red física no se verá afectada.

<sup>19</sup> <https://www.networkcomputing.com/sites/default/files/resources/nwc/Neutron-2.jpg>

<sup>20</sup> <https://www.networkcomputing.com/sites/default/files/resources/nwc/Neutron-3.jpg>

## E.4. Instrucciones de administración

Llegados a este punto ya conocemos más en profundidad los componentes que se han considerado más importantes dentro de OpenStack. Además, la administración de la plataforma puede llevarse a cabo desde la interfaz web gracias al servicio Horizon, o desde la línea de comandos CLI.

Por medio de este subapartado, se mencionará las instrucciones necesarias que permiten llevar a cabo una administración básica de la plataforma OpenStack. Todo ello realizando una distinción por cada elemento y haciendo o no uso de OpenStackClient.

### E.4.1. Componente KeyStone

Como bien conocemos, existen dos maneras de llevar a cabo la administración de OpenStack. En lo que respecta al uso de la interfaz web para alcanzar este objetivo se recomienda hacer uso del presente enlace:

- ✓ <https://www.youtube.com/watch?v=WYKwibmL7EO>

Si en cambio, queremos realizar esta administración desde la línea de comandos, podemos recurrir a alguna de las siguientes instrucciones.

Comando Keystone	OpenStackClient	Descripción
keystone project-list	openstack project list	Permite ver el número de proyectos que tenemos.
	openstack project create NOMBRE_PROYECTO	Creación de un nuevo proyecto.
	openstack user set -- password	Asignar una contraseña a un usuario en particular.
keystone service-list	openstack service list	Lista los servicios que tenemos disponible.
keystone user-list	openstack user list	Lista el conjunto de usuarios existentes.
keystone tenant-list	openstack tenant list	Lista los "tenants" que tenemos a disposición.
keystone user-create --name NOMBRE_USUARIO --tenant NOMBRE_TENANT --pass CONTRASEÑA	openstack user create NOMBRE_USUARIO -- password CONTRASEÑA	Creación de un usuario.
keystone user-role add --user ID_USUARIO --role ID_ROLE --tenant ID_TENANT	openstack role add NOMBRE_ROL--user NOMBRE_USUARIO --project NOMBRE_PROYECTO	Asignar un rol a un usuario en un proyecto.
	openstack role assignment list - -user NOMBRE_USUARIO -- project NOMBRE_PROYECTO	Conocer el rol de un usuario en un proyecto en cuestión.
keystone role-list	openstack role list	Lista de roles en la plataforma OpenStack.
keystone role-create --name NOMBRE_ROL	openstack role create NOMBRE_ROL	Crear un nuevo rol.

keystone tenant-create --name NOMBRE_TENANT --description DESCRIPCION --enabled true/false	openstack tenant create --name NOMBRE_TENANT	Crear un nuevo contenedor (Tenant).
---	---	-------------------------------------

Tabla 14: KeyStone - Lista de comandos

Cabe destacar, que en muchos de estos comandos hay más opciones de las que se reflejan en las tablas. Para conocer ellas, habrá que ir al manual de uso del comando en cuestión o utilizar la opción “-h”, es decir, por ejemplo:

```
openstack role assignment list -h
```

Si se quiere obtener más información se recomienda visitar alguno de los siguientes enlaces:

- ✓ <https://www.youtube.com/watch?v=593Sy7RoEc8> .
- ✓ <https://www.dbigcloud.com/cloud-computing/170-openstack-desde-cero-keystone.html> .

## E.4.2. Componente Glance

La administración de este servicio, al igual que sucedía con el servicio de autenticación, se puede llevar a cabo desde la interfaz web o por la línea de comandos CLI. En el caso que queramos realizar una administración por medio del servicio Horizon, recomiendo visitar el siguiente enlace:

- ✓ <http://iesgn.github.io/emergya/curso/u3/imagenes> .

Si por el contrario, deseamos llevar a cabo esta administración utilizando la línea de comandos, tendremos que conocer diferentes comandos que detallo a continuación.

Comando Glance	OpenStackClient	Descripción
glance image-list	openstack image list	Lista las imágenes disponibles.
glance image-show NOMBRE_IMAGEN	openstack image show ID_IMAGEN	Ver detalles de la imagen.
glance image-create -- name="NOMBRE_IMAGEN" -- container-format= bare ovf.. --disk-format= raw vhd vmdk... < imagen	openstack image create -- container-format= bare ovf.. --disk-format =raw vhd vmdk.. --public -- file imagen NOMBRE_IMAGEN	Crear una nueva imagen.
glance image-delete 'NOMBRE_IMAGEN'	openstack image delete ID_IMAGEN	Borrar una imagen.
glance image-update [opciones]	openstack image update [opciones]	Modificar una imagen.
glance help		Conocer la lista de comandos disponibles.

Tabla 15: Glance - Lista de comandos

Además de estos comandos existe otro conjunto no mencionado, ya que se ha pretendido destacar aquellos más relevantes.

Si deseamos conocer un ejemplo de uso, recomiendo ver el siguiente vídeo: [https://www.youtube.com/watch?v=Q\\_028WYUXGs](https://www.youtube.com/watch?v=Q_028WYUXGs).

### E.4.3. Servicio Nova

Como bien sabemos, la gestión de este componente podremos realizarlo por medio de la interfaz web o por la línea de comandos.

En lo que respecta a la administración por medio de Dashboard/servicio Horizon, se recomienda analizar los siguientes enlaces:

- ✓ <http://iesgn.github.io/emergya/curso/u3/instancias1> : Creación de una instancia a partir de una imagen.
- ✓ <http://iesgn.github.io/emergya/curso/u3/operaciones>: Operaciones sobre una instancia.
- ✓ <http://iesgn.github.io/emergya/curso/u3/redimension> : Redimensión y reconstrucción de una instancia.
- ✓ <http://iesgn.github.io/emergya/curso/u3/instantaneas> : Instantáneas de una instancia.

Por otra parte tenemos a disposición un conjunto de comandos que nos va a permitir crear la instancia en un primer momento, para luego poder llevar a cabo cualquier de las acciones permitidas sobre la misma. Para ello distinguiremos entre los comandos específicos del servicio y los proporcionados por OpenStackClient.

Centrémonos entonces en los pasos que deberemos seguir para la creación correcta de una instancia.

1. Creamos un nuevo par de claves ssh, modificando los permisos de la clave privada de forma adecuada.

Comando Nova	Comando OpenStackClient	Descripción
<code>nova keypair-add NOMBRE &gt; ~/.ssh/NOMBRE.pem</code>	<code>openstack keypair create NOMBRE &gt; NOMBRE.pem</code>	Creación de un nuevo par de claves de nombre "mis_claves".
<code>chmod 400 ~/.ssh/mi_claves.pem</code>	<code>chmod 400 ~/.ssh/mi_claves.pem</code>	Permisos para que ningún otro usuario pueda leer la clave.

Tabla 16: Nova - Claves SSH



2. Creamos un nuevo grupo de seguridad (cortafuegos) y creamos una regla que nos permita el acceso por ssh.

Comando Nova	Comando OpenStackClient	Descripción
nova secgroup-create NOMBRE DESCRIPCIÓN	openstack security group create NOMBRE	Creación de un nuevo grupo de seguridad.
nova secgroup-add-rule NOMBRE TCP 22 22 RANGOIP	openstack security group rule create --proto TCP --src-ip RANGOIP --dst-port 22 NOMBRE	Permitir acceso por ssh.

Tabla 17: Nova - Grupo Seguridad

3. Obtenemos la información de las imágenes, sabores y redes que tenemos disponibles

Comando Nova	Comando OpenStackClient	Descripción
nova image-list	openstack image list	Listar las imágenes disponibles.
nova flavor-list	openstack flavor list	Listas los sabores que tenemos al alcance.
nova net-list	openstack network list	Listar las redes que tenemos definidas.

Tabla 18: Nova – Información

4. Creamos una nueva instancia. Para ello deberemos elegir la imagen, el sabor, el grupo de seguridad, las claves ssh que hemos creado anteriormente, y la conexión de red.

Comando Nova	Comando OpenStackClient
nova boot --flavor NOMBRE_SABOR --image ID_IMAGEN --security_groups NOMBRE_ SEC_GRUPO --key-name NOMBRE_CLAVE -- nic net-id=NET_ID NOMBRE_INSTANCIA	openstack server create --flavor NOMBRE_SABOR --image NOMBRE_IMAGEN --security-group SEC_NOMBRE_GRUPO -- key-name NOMBRE_CLAVE --nice net- id=NET_ID_PRIVADA NOMBRE_INSTANCIA

Tabla 19: Nova - Creación instancia

5. Reservamos una nueva dirección IP flotante para asignársela a la instancia. De esta manera podremos acceder a ella desde el exterior.

Comando Nova	Comando OpenStackClient	Descripción
nova floating-ip-pool-list	openstack ip floating pool list	Lista el pool de direcciones IP flotantes.
nova floating-ip-create NOMBRE_RED_EXTERNA	openstack ip floating create NOMBRE	Asignar una IP flotante al proyecto.
nova floating-ip-associate NOMBRE_INSTANCIA DIRECCION_IP	openstack ip floating add DIRECCION_IP NOMBRE_INSTANCIA	Asignar la dirección IP flotante a la instancia.

Tabla 20: Nova - IP flotante

6. Finalmente, accedemos a la instancia.

Comando Nova	Comando OpenStackClient
ssh-i ~/.ssh/NOMBRE.pem nombremaquina@dirección_IP	ssh-i NOMBRE.pem nombremaquina@dirección_IP

Tabla 21: Nova - Acceder instancia

Estos comandos son los utilizados por tanto para la creación de una instancia. En lo que se refiere a las instrucciones para realizar el conjunto de acciones sobre las instancias propias del servicio Nova, se detallarán a continuación.

Comando Nova	Descripción
nova help	Conocer todas aquellas operaciones que se puede realizar con el comando proporcionado por el servicio Nova.
nova image-create --poll NOMBRE_INSTANCIA NOMBRE_INSTANTÁNEA	Crear una instantánea de una instancia.
nova image-list	Lista las instantáneas e imágenes disponibles.
nova secgroup-list	Listar los grupos de seguridad.
nova floating-ip-disassociate NOMBRE_INSTANCIA	Desasociar una IP flotante de una instancia.
nova floating-ip-delete X.X.X.X	Liberar una IP flotante.
nova floating-ip-list	Listas las direcciones IP flotantes asignadas al proyecto.
nova delete NOMBRE_INSTANCIA	Terminar una instancia.
nova reboot --hard servidor	Reiniciar una instancia.
nova suspend/resume NOMBRE_INSTANCIA	Suspender/Reiniciar una instancia.
nova pause/unpause NOMBRE_INSTANCIA	Pausar una instancia.
nova keypair-list	Listar las claves ssh.
nova secgroup-list-rules default	Listar las reglas de un grupo de seguridad.

Tabla 22: Nova - Operaciones en las instancias

#### E.4.4. Servicio Cinder

Todo el conjunto de operaciones que podemos llevar a cabo sobre los volúmenes, pueden realizarse haciendo uso de la interfaz web o de la línea de comandos. Si escogemos la primera opción se recomienda visitar el siguiente enlace, donde de una manera visual podremos realizar una correcta administración de este servicio:

- ✓ <http://iesgn.github.io/emergya/curso/u4/operaciones> .

Si en cambio, optamos por utilizar la línea de comandos para realizar la administración deberemos distinguir entre los comandos propios del servicio y los de OpenStackClient. A continuación, reflejaré alguna de las operaciones con los respectivos pasos e instrucciones que debemos seguir y hacer uso respectivamente.

- ❖ Asociación de un volumen a una instancia

Pasos a seguir:

1. Creamos un volumen con un tamaño en GB determinado.
2. Asociamos el volumen anteriormente creado con una instancia en cuestión.
3. Comprobamos que el volumen aparece asociado a la instancia.
4. Opcionalmente, podemos desasociar el volumen de una instancia.

Comando Cinder	Comando OpenStackClient	Operación
nova volume-create TAMAÑOGB --display-name NOMBRE_VOLUMEN	openstack volume create -- size TAMAÑOGB NOMBRE_VOLUMEN	1
nova volume-attach NOMBRE_INSTANCIA ID_VOLUMEN /dev/vdb	openstack server add volume NOMBRE_INSTANCIA NOMBRE_VOLUMEN	2
nova volume_list	openstack volume list	3
nova volume-detach NOMBRE_INSTANCIA ID_VOLUMEN	openstack server remove volume NOMBRE_INSTANCIA NOMBRE_VOLUMEN	4

Tabla 23: Cinder - Asociar volumen a instancia

Cabe destacar que los comandos propios del servicio no son los de Cinder sino lo de Nova.

- ❖ Creación de una instancia con el disco raíz de un volumen

Sigamos los siguientes pasos:

1. Conocemos la lista de sabores e imágenes que tenemos disponibles en nuestro sistema.
2. Creamos un volumen. A este volumen habrá que asignarle un espacio en GB determinados y que contenga una imagen.
3. Nos cercioramos de que OpenStack reconoce al volumen anteriormente creado como arrancable al contener un sistema operativo<sup>27</sup> (bootable=true).
4. Creamos la nueva instancia asignándole el volumen que habíamos creado.

Comando Cinder	Comando OpenStackClient	Operación
nova image-list	openstack image list	1
nova flavor-list	openstack flavor list	1
nova volume-create TAMAÑOGB --image.id ID_IMAGEN --display-name NOMBRE_VOLUMEN	openstack volume create -- size TAMAÑOGB --image ID_IMAGEN NOMBRE_VOLUMEN	2
nova volume-show ID_VOLUMEN	openstack volumen show NOMBRE_VOLUMEN	3
nova boot --flavor NOMBRE_SABOR --boot- volume ID_VOLUMEN -- security-groups NOMBRE_GRUPOSEGURIDAD --key-name NOMBRE_CLAVE -- nic net-id=id_red NOMBRE_INSTANCIA	openstack server create -- flavor NOMBRE_SABOR -- volume NOMBRE_VOLUMEN --security-groups NOMBRE_GRUPOSEGURIDAD --key-name NOMBRE_CLAVE -- nic net-id=id_red NOMBRE_INSTANCIA	4

Tabla 24: Cinder - Creación de instancia con el volumen

Al igual que sucedía con la operación anterior, utilizados comandos propios del servicio Nova.

❖ Creación de una instantánea de un volumen

Al crear una instantánea (snapshot) estoy guardando el contenido del volumen. A partir de una instantánea se pueden crear nuevo volúmenes con el mismo contenido que el original.

Las operaciones que podemos llevar a cabo entonces son las siguientes:

1. Crear una instantánea de un volumen en cuestión.
2. Listar todas las instantáneas que tenemos disponibles.
3. Crear un nuevo volumen a partir de una instantánea.
4. Borrar la instantánea y el volumen creado. Para evitar errores, primero se borra la instantánea y posteriormente el volumen.

Comando OpenStackClient	Operación
openstack snapshot create --name NOMBRE_INSTANTÁNEA NOMBRE_VOLUMEN	1
openstack snapshot list	2
openstack volume create --size TAMAÑOGB --snapshot ID_INSTANTÁNEA	3
openstack snapshot delete NOMBRE_INSTANTÁNEA openstack volume delete NOMBRE_VOLUMEN	4

Tabla 25: Cinder - Instantánea de un volumen

❖ Extender el tamaño de un volumen

Podemos redimensionar el dispositivo de bloque (el volumen). Esta operación sólo se puede realizar si el volumen esta desasociado de una instancia. Para ello seguiremos los siguientes pasos:

1. Redimensionar el tamaño del volumen en cuestión.
2. Listamos los volúmenes disponibles cerciorándonos que se ha redimensionado correctamente.

Comando OpenStackClient	Operación
openstack volume set --size TAMAÑOGB NOMBRE_VOLUMEN	1
openstack volume list	2

Tabla 26: Cinder - Redimensionar volumen

#### ❖ Transferencia de volumen

Esta opción nos permite transferir un volumen de un proyecto a otro. Para ello realizamos lo siguiente:

1. Podemos listar todos los volúmenes que tenemos creados en nuestro sistema y escoger cual queremos transferir.
2. Realizamos la transferencia del volumen en cuestión.
3. El administrador del proyecto al que se desea transferir el volumen, se deberá de loguear y posteriormente ver su listado de volúmenes. Lo más normal es que no lo disponga aún. Entonces, “acepta” la transferencia debiendo indicar el identificador de la transferencia en cuestión y el parámetro de autenticación. A continuación, si vuelve a listar los volúmenes disponibles debería de reflejarse.
4. Si volemos al usuario inicial debería de no disponer del volumen ni de ninguna transferencia pendiente. Ya no es el propietario del volumen.

Comando Cinder	Operación
cinder list	1
cinder transfer-create ID_VOLUMEN	2
cinder list cinder transfer-accept ID_TRANSFERENCIA AUTH_KEY_TRANSFERENCIA cinder list	3
cinder list cinder transfer-list	4

Tabla 27: Cinder - Transferencia de volumen

#### ❖ Copia de seguridad de volúmenes

Este servicio proporciona las herramientas necesarias para crear una copia de seguridad de un volumen. Para realizar esta copia, el volumen debe estar en un estado “Disponible”, es decir, no debe estar asociado a ninguna instancia. Las copias de seguridad se guardan como objetos en el contenedor de objetos Swift.

Para crear una copia de seguridad de un volumen ejecutamos las siguientes instrucciones:

1. Estudiamos los volúmenes que tenemos disponibles y escogemos uno de ellos.
2. Realizamos la copia de seguridad sobre el volumen anteriormente elegido.
3. Nos cercioramos que se ha realizado correctamente la copia de seguridad.
4. Opcionalmente, podemos pedir información sobre la copia de seguridad que se ha realizado.

Comando Cinder	Operación
cinder list	1
cinder backup-create ID_VOLUMEN	2
Cinder backup-list	3
cinder backup-show ID_BACKUP	4

Tabla 28: Cinder - Crear copia de seguridad

Con esta copia de seguridad podemos llevar a cabo, entre otras opciones, la restauración de un nuevo volumen. Para ello seguimos los siguientes pasos:

1. Restauramos a partir de una copia de seguridad un nuevo volumen.
2. Nos cercioramos que se ha creado un nuevo volumen restaurado desde la copia de seguridad.

Comando Cinder	Operación
cinder backup-restore ID_BACKUP	1
cinder list	2

Tabla 29: Cinder - Restauración copia de seguridad

### E.4.5. Servicio Swift

Como bien se ha comentado anteriormente, Swift es el componente de OpenStack que nos permite la gestión de objetos, entendiendo por objeto un elemento que está relacionado con un fichero o archivo, incluyendo además una serie de metadatos<sup>20</sup>, y que se guardan en contenedores.

La administración de este servicio no resulta muy compleja. En lo que respecta a una gestión basada en la interfaz web, se recomienda visitar el siguiente enlace:

- ✓ <http://iesgn.github.io/emergya/curso/u4/swift>

Por otra parte, habrá que tener en consideración los diferentes comandos que se tienen definidos para la administración del servicio Swift por la línea de comandos.

Comando Swift	Comando OpenStackClient	Descripción
swift upload contenedor NOMBRE_CONTENEDOR	openstack container create NOMBRE_CONTENEDOR	Crear un nuevo contenedor.
swift list	openstack container list	Listar los contenedores disponibles.
swift list NOMBRE_CONTENEDOR	openstack object list NOMBRE_CONTENEDOR	Listar los objetos que tiene a disposición un contenedor.
swift upload NOMBRE_CONTENEDOR OBJETO	openstack object create NOMBRE_CONTENEDOR OBJETO	Subir un nuevo objeto a un contenedor.
swift stat NOMBRE_CONTENEDOR [OBJETO]	openstack container show NOMBRE_CONTENEDOR [OBJETO]	Obtener información de un contenedor [o un objeto].
swift download NOMBRE_CONTENEDOR OBJETO	openstack object save NOMBRE_CONTENEDOR OBJETO	Descargar un objeto de un contenedor.
swift delete NOMBRE_CONTENEDOR	openstack container delete NOMBRE_CONTENEDOR	Borrar un contenedor en cuestión.
	openstack object delete NOMBRE_CONTENEDOR OBJETO	Borrar un objeto de un contenedor en cuestión.

https://swift-mm1.mascloud.es/v1.0/ ID_CONTENEDOR/ NOMBRECONTENEDOR/ NOMBRE_OBJETO	https://DIRECCIONIPHORIZON/ v1/ID_CONTENEDOR/ NOMBE_CONTENEDOR/ NOMBRE_OBJETO	Acceder a un objeto determinado.
swift post -r 'r:*' NOMBRE_CONTENEDOR	Openstack container set read_acl='r:*' NOMBRE_CONTENEDOR	Dar permiso de lectura al contenedor para ser accesible la imagen.

Tabla 30: Swift - Lista de comandos

#### E.4.6. Servicio Neutron

Como bien se dijo, el componente Neutron será el servicio encargado de la administración de red. Debido a las diferentes posibilidades de configuración que tiene disponibles, durante este subapartado enunciaremos diferentes situaciones que podemos encontrarnos y como tratarlas.

##### ❖ Creación de la red externa

La red externa es típicamente una red que proporciona conexión con el exterior a las redes internas. Para crear la red externa se recomienda seguir los siguientes pasos:

1. Accedemos al equipo donde hemos instalado OpenStack y cargamos las credenciales de administrador, ya que la red externa sólo la puede crear alguien con este rol.
2. Creamos la red externa por medio de las siguientes instrucciones.
3. Definimos una subred donde aparece la definición lógica de la red y cuyos valores dependerán de la red privada o pública a la que esté conectada.

Comando Neutron	Descripción
neutron net-create NOMBRE_RED_EXTERNA - -router:external --provider:physical_network external --provider:network_type flat	Creación de la red externa
neutron subnet-create NOMBRE_RED_EXTERNA RED --name NOMBRE_SUBRED --allocation-pool start=IP_FLOTANTE_INICIO, end= IP_FLOTANTE_FIN --disable-dhcp --gateway PUERTA_ENLACE	Creación de la subred reservando un segmento para las IP flotantes de las instancias que correrán sobre OpenStack.

Tabla 31: Neutron - Creación red externa

❖ Creación de una nueva red interna asociada a una instancia

Por medio de la siguiente situación se pretenderá crear una red interna, un nuevo router y una instancia conectada a esta nueva red. Será necesario tener instalado el cliente “Python-neutronclient”.

Veamos los pasos que tenemos que seguir:

1. Vemos las redes, subredes y routers que tenemos disponibles.
2. Creamos una nueva red.
3. Asociamos a la nueva red una subred con un direccionamiento determinado.
4. Creamos un nuevo router que conectaremos con la red externa y la subred anteriormente creadas.
5. Creamos una nueva instancia conectada a la nueva red y le asignamos una dirección IP pública.

Comando Neutron	Descripción	Operación
neutron net-list	Lista las redes disponibles.	1
neutron subnet-list	Lista las subredes disponibles.	1
neutron router-list	Lista los routers disponibles.	1
neutron net-create NOMBRE_RED	Creación de una nueva red.	2
neutron subnet-create NOMBRE_RED DIRECCIONAMIENTO --name NOMBRE_SUBRED	Creación de una nueva subred asociándola a una red con un direccionamiento en particular.	3
neutron router-create NOMBRE_ROUTER	Creación de un nuevo router.	4
neutron router-gateway-set NOMBRE_ROUTER NOMBRE_RED	Conectar un router a una red externa en particular.	4
neutron router-interface-add NOMBRE_ROUTER NOMBRE_SUBRED	Conectar un router a la red interna por medio de la subred.	4

Tabla 32: Neutron -Red interna asociada a una instancia

Todo lo que hemos visto hasta ahora corresponde a hacer uso de la línea de comandos para realizar la administración. En cambio, si deseamos llevar a cabo la gestión desde el interfaz web, recomiendo echar un vistazo a los siguientes enlaces:

- ✓ <http://iesgn.github.io/emergya/curso/u5/neutron>
- ✓ <http://iesgn.github.io/emergya/curso/u5/puertos> : Información sobre los puertos.
- ✓ <http://iesgn.github.io/emergya/curso/u5/borrar> : Eliminar infraestructura de red.



## F. Instanciar y conexión remota SSH

### F.1. Introducción

Por medio de este anexo, se pretende reflejar como podemos instanciar una imagen en la plataforma OpenStack, y acceder a la misma remotamente por medio del protocolo de comunicaciones SSH. Es decir, con este anexo explicaremos con mayor precisión los pasos mencionados en el apartado 6, más en concreto en el subapartado 6.2, para que cualquier usuario pueda instanciar una imagen y acceder a ella internamente o remotamente gracias al protocolo SSH. De esta manera alcanzaría el disponer de un escritorio virtual al cual puede acceder desde cualquier dispositivo con capacidad de conexión.

Para poder realizar este caso práctico, nos basaremos en el anexo anterior donde se reflejan los conceptos básicos de administración. Además, recurriremos a un sistema Fedora, debido a que los recursos que requiere para su correcto funcionamiento son soportados por las limitaciones de hardware disponibles.

### F.2. Administración – Fedora

#### F.2.1. Servicio KeyStone

El componente Keystone, como se ha ido mencionando a lo largo de esta memoria, es el servicio que utiliza OpenStack para la autenticación.

Una vez finalizamos la instalación de OpenStack, el único cliente que tenemos disponible, en la mayoría de los casos, es el que corresponde al administrador de la plataforma, comúnmente denominado “admin”. Este usuario tiene a su vez asignado un proyecto del mismo nombre y a su disposición el poder realizar cualquier cosa que desee, desde subir una imagen a lanzar una instancia, por ejemplo.

El objetivo que se tendrá en esta primera prueba, será la creación de un nuevo usuario que tenga un rol determinado dentro de un nuevo proyecto. Por esa razón, hemos utilizado la línea de comandos con las credenciales del administrador, para llevar a cabo todas estas tareas que enumero y explico con mayor precisión a continuación. Además, ello nos servirá para conocer el conjunto de acciones que puede llevar a cabo un usuario normal, no administrador, sobre la plataforma OpenStack.

1. Acceder con las credenciales del administrador.

Lo primero a realizar será iniciar sesión como administradores ya que de esta forma seremos capaces de alcanzar nuestro propósito. Para ello conocemos el fichero que contiene todos los datos relevantes de dicho usuario y hacemos uso del comando “source”.

```
[root@openstacklocalhost ~]# ls
anaconda-ks.cfg keystoneadmin packstack-answers-20180323-114125.txt
[root@openstacklocalhost ~]# source keystoneadmin
```

Ilustración 97: Servicio Keystone - Inicio sesión administrador

2. Listar el conjunto de proyectos, roles y usuarios disponibles.

Es recomendable listar todo el conjunto de proyectos, roles y usuarios que tenemos disponibles al inicio, debido a que de esta manera podemos conocer más de cerca cómo se constituye la plataforma OpenStack.

En lo que respecta a los proyectos disponibles, serán los que se muestren en la siguiente imagen.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack project list
+-----+-----+
| ID | Name |
+-----+-----+
| 6dbfeb3be6d74a588a090768589e7340 | services |
| b509a04d81134b9a9a26b78dca3ab118 | admin |
+-----+-----+
```

Ilustración 98: Servicio Keystone - Proyectos iniciales disponibles

Como bien podemos apreciar, tenemos el proyecto correspondiente al administrador del sistema y otro proyecto denominado “services”, que es de gran importancia debido a que es el que utilizarán los componentes de la plataforma.

En lo que respecta a los usuarios iniciales disponibles, alcanzamos el siguiente resultado.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack user list
+-----+-----+
| ID | Name |
+-----+-----+
| 03d7albfb08444fa8932369151d3441b | gnocchi |
| 511229ce7cbc4d53b53bd0b456fc8eaa | placement |
| 72836d290c8b4530b54bcb5079658c56 | swift |
| 77779f493d6f403c80908bd08987e248 | heat_admin |
| 804be86c38144044956ecl16072c6508d | cinder |
| 840ca3e268904d848bd04a7b746e4a74 | glance |
| 875e183cbec947d39433b0c36e0ec37c | aodh |
| a524cf75d44d4cfl85d7847ac0ddl370 | heat-cfn |
| bb9375e695894aff9b2740ce2b8dc376 | neutron |
| bca9f2a348334569a49a2c329964d9c3 | nova |
| ca2467f9955342c4890f44ffc9flb17a | heat |
| d724bla8bde146e2b0dbd2ec3ca779d5 | ceilometer |
| f98cfce0f59547ed99997961fl16abc2a | admin |
+-----+-----+
```

Ilustración 99: Servicio Keystone - Usuarios iniciales disponibles

Entonces la plataforma OpenStack no solo tiene inicialmente un usuario de nombre “admin”, sino también los correspondientes a los diferentes servicios previamente instalados.

Finalmente, visualicemos los roles por defecto que tiene definido OpenStack.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack role list
+-----+
| ID | Name |
+-----+
| 13d99af6d69e49969b95ab8668587479 | ResellerAdmin |
| 60f1093c38d84b068e96a827bf6358fe | heat_stack_user |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_ |
| e746b48782f7483ea46337470f0760ff | heat_stack_owner |
| ec3d83af1dac436483cb0167fa4b412b | SwiftOperator |
| fbe57b752c794ddabb7bd45b53aed5f8 | admin |
+-----+
```

Ilustración 100: Servicio Keystone - Roles iniciales disponibles

### 3. Creación de usuario.

Como bien se ha comentado anteriormente, se requiere la creación de un nuevo usuario que se realizará por medio de la siguiente instrucción.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack user create openMikel --password openstack
+-----+
| Field | Value |
+-----+
| domain_id | default |
| enabled | True |
| id | cd93403e48cd4a399da62f9398fle0b2 |
| name | openMikel |
| options | {} |
| password_expires_at | None |
+-----+
```

Ilustración 101: Servicio Keystone - Creación de usuario

Existen más opciones que permite el comando, pero en nuestro caso exclusivamente hemos querido dar un nombre y una contraseña a este nuevo usuario, al cual se le asignará posteriormente un identificador único y el dominio<sup>14</sup> por defecto.

### 4. Creación del proyecto.

Para la creación de cualquier proyecto, se utiliza la siguiente instrucción.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack project create openFernandez
+-----+
| Field | Value |
+-----+
| description | |
| domain_id | default |
| enabled | True |
| id | 38a5bba0ec194b65bf1d92263d2f79f8 |
| is_domain | False |
| name | openFernandez |
| parent_id | default |
+-----+
```

Ilustración 102: Servicio Keystone - Creación de proyecto

En este caso, exclusivamente se ha dotado de un nombre al proyecto, aunque tendríamos otras posibilidades añadidas como realizarle una descripción, por ejemplo. De esta forma y al igual que ocurría cuando creabas un usuario, se define un identificador único del proyecto y formará parte del dominio<sup>14</sup> por defecto.

## 5. Creación del rol.

Al igual que hemos visto en los dos casos anteriores, para la creación de un nuevo rol tendremos que hacer uso de un comando en particular que tendrá diferentes opciones, pero que nos hemos centrado en dotar al nuevo rol de un nombre solamente.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack role create openRol
+-----+-----+
| Field      | Value                               |
+-----+-----+
| domain_id  | None                                 |
| id         | 6422bf0085b44990a8da935d97459be8   |
| name       | openRol                              |
+-----+-----+
```

Ilustración 103: Servicio Keystone - Creación de rol

## 6. Comprobación.

Para verificar que el usuario, proyecto y rol se han creado de manera adecuada, realizamos la siguiente comprobación, listando todos los proyectos, roles y usuarios disponibles.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack user list
+-----+-----+
| ID                                               | Name           |
+-----+-----+
| 03d7albfb08444fa8932369151d3441b              | gnocchi        |
| 511229ce7cbc4d53b53bd0b456fc8eaa              | placement      |
| 72836d290c8b4530b54bcb5079658c56              | swift           |
| 77779f493d6f403c80908bd08987e248              | heat_admin     |
| 804be86c38144044956ecl6072c6508d              | cinder          |
| 840ca3e268904d848bd04a7b746e4a74              | glance          |
| 875e183cbec947d39433b0c36e0ec37c              | aodh            |
| a524cf75d44d4cf185d7847ac0dd1370              | heat-cfn        |
| bb9375e695894aff9b2740ce2b8dc376              | neutron         |
| bca9f2a348334569a49a2c329964d9c3              | nova            |
| ca2467f9955342c4890f44ffc9f1b17a              | heat            |
| cd93403e48cd4a399da62f9398fle0b2              | openMikel      |
| d724bla8bde146e2b0dbd2ec3ca779d5              | ceilometer     |
| f98cfce0f59547ed99997961f16abc2a              | admin           |
+-----+-----+

[root@openstacklocalhost ~(keystone_admin)]# openstack project list
+-----+-----+
| ID                                               | Name           |
+-----+-----+
| 38a5bba0ecl94b65bfd92263d2f79f8              | openFernandez  |
| 6dbfeb3be6d74a588a090768589e7340              | services        |
| b509a04d81134b9a9a26b78dca3ab118              | admin           |
+-----+-----+

[root@openstacklocalhost ~(keystone_admin)]# openstack role list
+-----+-----+
| ID                                               | Name           |
+-----+-----+
| 13d99af6d69e49969b95ab8668587479              | ResellerAdmin  |
| 60f1093c38d84b068e96a827bf6358fe              | heat_stack_user|
| 6422bf0085b44990a8da935d97459be8              | openRol        |
| 9fe2ff9ea4384b1894a90878d3e92bab              | _member_       |
| e746b48782f7483ea46337470f0760ff              | heat_stack_owner|
| ec3d83af1dac436483cb0167fa4b412b              | SwiftOperator  |
| fbe57b752c794ddabb7bd45b53aed5f8              | admin           |
+-----+-----+
```

Ilustración 104: Servicio Keystone - Comprobación rol, usuario, proyecto

## 7. Asignar rol a un usuario en un proyecto.

Un usuario que no tiene asignado un rol dentro de un proyecto, es un usuario que no puede realizar ninguna acción sobre dicho proyecto. Entonces, para asignarle un rol a un usuario en un proyecto en cuestión, se hará uso del siguiente comando.

```
[root@openstacklocalhost ~ (keystone_admin)]# openstack role add openRol --user openMikel --project openFernandez
```

Ilustración 105: Servicio Keystone - Asignar rol usuario en proyecto

Como bien podemos apreciar, se indica el rol que se quiere asignar a que usuario y en qué proyecto. Para verificar que se ha realizado correctamente y que ese usuario tiene ahora ese “papel” dentro de ese proyecto, utilizamos una de las siguientes instrucciones.

```
[root@openstacklocalhost ~ (keystone_admin)]# openstack role assignment list --user openMikel --project openFernandez
```

Role	User	Group	Project	Domain	Inherited
6422bf0085b44990a8da935d97459be8	cd93403e48cd4a399da62f9398f1e0b2		38a5bba0ec194b65bfd92263d2f79f8		False

```
[root@openstacklocalhost ~ (keystone_admin)]# openstack role assignment list --user openMikel --project openFernandez --name
```

Role	User	Group	Project	Domain	Inherited
openRol	openMikel@Default		openFernandez@Default		False

Ilustración 106: Servicio Keystone - Comprobación rol asignado

Destacar que el único que formará parte del proyecto será el usuario creado. El usuario “admin”, a no ser que lo especifiquemos, no formará parte de dicho proyecto no pudiendo llevar a cabo ninguna acción.

## 8. Comprobación en el servicio Horizon.

Podemos verificar con nuestra interfaz web que podemos registrarnos como el usuario que hemos creado y así acceder a su plataforma.

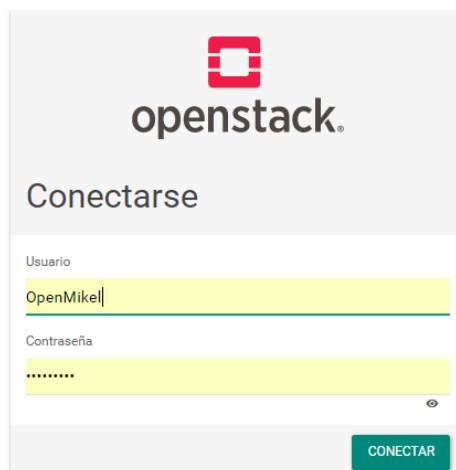


Ilustración 107: Servicio Keystone - Acceso Servicio Horizon

Destacar que las opciones que tendrá a disposición este usuario serán las que figuren en la imagen siguiente, en donde no se tendrá la opción de administrar al no tener el propio rol asignado, pero donde se le permitirán muchas otras cuestiones.

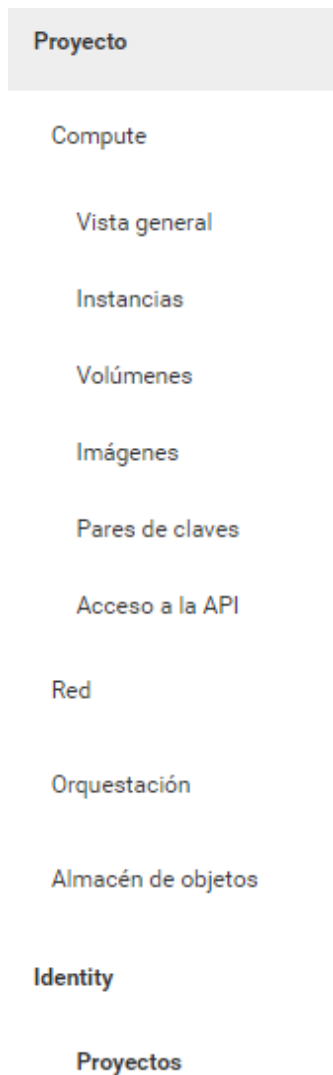


Ilustración 108: Servicio KeyStone - Menú usuario no administrador

Como bien puede apreciarse, en la opción “identity” solo mostrará los proyectos que tiene asignado dicho usuario. En ningún caso conocerá si existen más usuarios, más proyectos o los diferentes roles que existen. Todo ello es gestionado por el administrador del sistema. De todas formas, hay cuestiones que puede realizar y haga replantear a un administrador de esta plataforma, la necesidad o no de que tenga tantas opciones.

## F.2.2. Servicio Neutron

Por medio de este componente, podemos realizar la administración de red de la plataforma OpenStack. Como bien se comentó en los aspectos de administración de la plataforma, cualquier instancia que hagamos uso deberá de conectarse a una red interna. Esta red interna a su vez, deberá conectarse a un router virtual y este a una red externa. Todo ello para que la instancia que se lance puede ser accedida desde el exterior e internamente. El inconveniente inicial que nos encontramos, es precisamente que no disponemos de ninguna red y de ningún router virtual. Esto significa que si utilizásemos cualquier instancia, está en ningún caso se podrá acceder.

Dado el problema encontrado, nuestro objetivo será precisamente crear una red interna donde conectar nuestras instancias, y crear un router virtual y una red externa para que en un instante dado poder conectarnos a esa instancia remotamente. Expliquemos detalladamente como hemos alcanzado esta meta.

1. Comprobar las redes disponibles.

Estudiamos si tenemos disponible alguna red, alguna subred y algún router virtual.

```
[root@openstacklocalhost ~(keystone_admin)]# openstack network list
[root@openstacklocalhost ~(keystone_admin)]# openstack subnet list
[root@openstacklocalhost ~(keystone_admin)]# openstack router list
```

Ilustración 109: Servicio Neutron - Comprobación inicial

Como bien se presagiaba, no tenemos disponible ningún router virtual, ninguna red y ninguna subred.

2. Creación de red interna.

Creemos una nueva red interna con un nombre en cuestión y con una subred asociada. En este caso tendrá el direccionamiento 20.0.0.0/24.

```
[root@openstack ~(keystone_admin)]# openstack network create mi_red
```

Ilustración 110: Servicio Neutron - Creación de red interna

```
[root@openstack ~(keystone_admin)]# neutron subnet-create b7fbc438-8268-41e5-9109-e1448993d738 20.0.0.0/24 --name mi_subred
```

Ilustración 111: Servicio Neutron - Creación subred de red interna

El utilizar ese direccionamiento es indiferente. Se tratan de direcciones virtuales, por lo que por cada red interna que creemos podemos asignarle una dirección en cuestión y será la plataforma la encargada de su gestión. Recordemos hacer un uso correcto del identificador de la red interna a la hora de crear su subred.

### 3. Creación del router virtual.

Para la creación del router virtual utilizamos la siguiente instrucción, donde indicamos el nombre del router virtual.

```
[root@openstack ~(keystone_admin)]# openstack router create mi_router
```

Ilustración 112: Servicio Neutron - Creación de router

### 4. Creación de la red externa.

La red externa es típicamente una red que proporciona conexión con el exterior a las redes internas que se crean sobre la plataforma OpenStack. Lo primero será crear la propia red externa y a continuación, definimos una subred donde aparece la definición lógica de la red y cuyos valores dependerán de la red privada o pública a la que esté conectada.

En este caso y al contrario que en las redes anteriores vamos a hacer uso del servicio Horizon:

a. Nos dirigimos a Administrador → Redes y seleccionamos posteriormente la opción de crear una nueva red.



Ilustración 113: Servicio Neutron - Crear red

b. Se nos abrirá una nueva ventana en donde dotaremos a la red de un nombre, la asignaremos a nuestro proyecto y seleccionamos como proveedor el plano ("flat" en inglés). A continuación, definimos el nombre por el que se reconocerá la red física y habilitamos las opciones "Enable Admin State" y "Red Externa".

A screenshot of the 'Crear red' form in Horizon. The form has a title 'Crear red' and a close button 'x'. It contains several fields and checkboxes. On the left side, four red arrows point to the 'Nombre', 'Proyecto', 'Proveedor Tipo de red', and 'Red física' fields. The 'Nombre' field contains 'ext-net'. The 'Proyecto' dropdown is set to 'admin'. The 'Proveedor Tipo de red' dropdown is set to 'Plano'. The 'Red física' field contains 'extnet'. Below these fields are three checkboxes: 'Enable Admin State' (checked), 'Compartido' (unchecked), and 'Red externa' (checked). On the right side, there is a 'Descripción:' section with text explaining the purpose of the form. At the bottom right, there are 'Cancelar' and 'Enviar' buttons.

Ilustración 114: Servicio Neutron - Red externa



c. Creada la red externa, habrá que asociarle una subred. Para ello realizamos una selección sobre el nombre de la red externa. Esto nos dirigirá a un espacio en donde se describe la red creada. Nuestro objetivo es crear una subred por lo que seleccionamos la opción que recibe el mismo nombre y escogemos la creación de la misma. Esto originará la apertura de una ventana en donde habrá que detallar los datos particulares de la subred externa. Entre ellos destacar que la dirección de red y la puerta de red deben ser equivalentes a la dirección de red de la plataforma.

Crear subred

Subred Detalles de Subred

Nombre de subred  
ext-subnet

Direcciones de red  
192.168.1.0/24

Versión de IP  
IPv4

IP de la puerta de enlace  
192.168.1.1

Deshabilitar puerta de enlace

Crear una subred asociada con la red. Podrá consultar la configuración avanzada haciendo clic en la pestaña "Detalles de la Subred"

Cancelar < Anterior Siguiente >

Ilustración 115: Servicio Neutron - Subred externa (1)

d. Seleccionamos el botón "Siguiente" y en la ventana que se muestra definiremos el segmento que haremos uso para las direcciones IP flotantes, debiendo previamente deshabilitar la opción de DHCP. Se recomienda comprobar que se tienen libres ese conjunto de dirección IP que se asignarán.

Crear subred

Subred Detalles de Subred

Habilitar DHCP

Pools de asignación  
192.168.1.150,192.168.1.160

Servidores DNS

Rutas de host

Especificar atributos adicionales para la subred.

Cancelar < Anterior Crear

Ilustración 116: Servicio Neutron - Subred externa (2)

## 5. Conexión.

El siguiente paso que debemos dar será conectar por una parte el router con la red interna, y por otra, el router con la red externa. Esto lo realizaremos por medio de dos instrucciones que se detallan en las siguientes imágenes.

```
[root@openstack ~(keystone_admin)]# neutron router-gateway-set aldeda0c-7cec-42e9-80dc-5fc1f326460d ext-net
```

Ilustración 117: Servicio Neutron - Conexión router con red externa

```
[root@openstack ~(keystone_admin)]# neutron router-interface-add aldeda0c-7cec-42e9-80dc-5fc1f326460d 792542b5-43cd-4388-9f18-52ce5163fd8e
```

Ilustración 118: Servicio Neutron - Conexión router con red interna

En la primera de las instrucciones se indica el identificador del router con el nombre de la red externa. En cambio, en la segunda, se notifica en primer lugar el identificador del router con el identificador de la subred interna a continuación.

## 6. Comprobación.

Si iniciamos sesión en la interfaz web, ya sea como administrador o como cualquier usuario que tenga un rol dentro del proyecto, y accedemos concretamente al apartado de redes y topología de red, podremos visualizar las redes que tenemos creadas para ese proyecto en cuestión de diferentes formas.

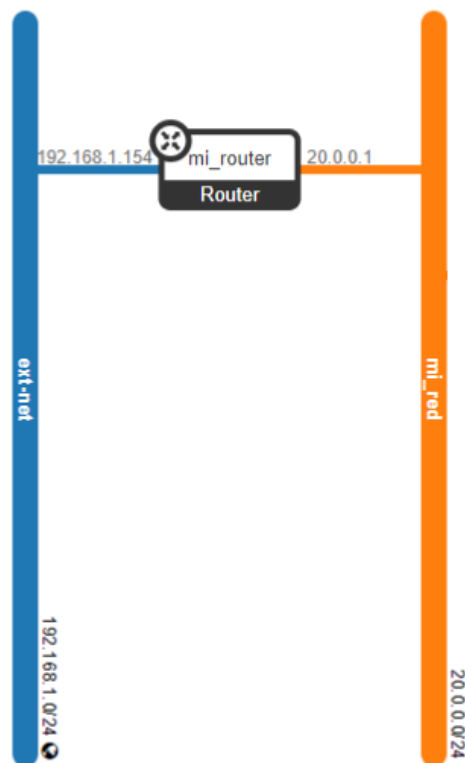


Ilustración 119: Servicio Neutron - Comprobación redes

### F.2.3. Servicio Glance

Glance es el componente de OpenStack el cual nos permite gestionar nuestras imágenes de sistemas operativos. Explicaremos de qué manera lo hemos llevado a cabo.

1. Comprobar las imágenes disponibles.

En primer lugar, comprobamos que conjunto de imágenes tenemos disponibles para hacer uso.

```
[root@openstacklocalhost ~(keystone_admin)]# glance image-list
+-----+-----+
| ID | Name |
+-----+-----+
+-----+-----+
```

Ilustración 120: Servicio Glance - Imágenes disponibles inicialmente

Como bien podemos apreciar, inicialmente no se disponen de ninguna imagen.

2. Descargar la imagen.

A continuación, descargaremos aquella imagen que queramos posteriormente subir a nuestra plataforma OpenStack. Se puede realizar descargándola por la línea de comandos utilizando la instrucción “wget”, o directamente haciendo uso del servicio Horizon de un sistema que tengamos ya disponible.

Para esta prueba se ha pensado en el presente enlace, en donde se disponen de imágenes preparadas específicamente para poder ser utilizadas en la plataforma OpenStack.

- ❖ <https://docs.openstack.org/image-guide/obtain-images.html>

3. Subir imagen.

Lo siguiente a realizar será subir nuestra imagen. Para ello haré uso del servicio Horizon teniendo que indicar los siguientes datos:

- ❖ Nombre: Nombre que asignaremos a la nueva imagen.
- ❖ Descripción: Texto que describa la nueva imagen que vamos a subir.
- ❖ Origen de la imagen: Indicamos desde donde vamos a subir la imagen.
- ❖ Formato: Seleccionamos el formato de la imagen en base a los soportados por la plataforma.
- ❖ Arquitectura: Podemos indicar la arquitectura del sistema operativo<sup>27</sup> de la imagen.
- ❖ Disco mínimo (GB): Podemos indicar como requisito, el tamaño mínimo que tiene que tener una instancia que se cree a partir de la imagen.
- ❖ RAM mínima (MB): Podemos determinar como requisito, el tamaño mínimo de memoria RAM que debe tener una instancia que se cree a partir de la imagen.
- ❖ Público: Con este parámetro indicamos si queremos que la imagen sea pública o no para todos los usuarios.
- ❖ Protegido: La imagen que vamos a crear podrá ser borrada o no por otro usuario.

#### 4. Estados.

Una imagen una vez que es subida pasa por un conjunto de estados. Inicialmente, se encontrará en un estado encolado. Si refrescamos la página web o esperamos unos escasos minutos, la imagen pasará a un estado activo, estando disponible para su posterior uso.



	Owner	Nombre ▲	Tipo	Estado	Visibilidad	Protegido	Disk Format	Tamaño	
<input type="checkbox"/>	> admin	Imagen_Fedora	Imagen	Activo	Público	No	QCOW2	250.00 MB	Launch ▼

Ilustración 121: Servicio Glance - Estado activo

### F.2.4. Servicio Cinder

Este componente de OpenStack permite la gestión de los dispositivos de bloques conocidos como volúmenes, que permiten proporcionar un almacenamiento permanente o independiente en la vida de una instancia.

Nuestro objetivo será crear un volumen para la instancia que posteriormente lanzaremos. Para ello, deberemos seguir los siguientes pasos:

1. Nos dirigimos a “Proyecto-Volumenes” y seleccionamos la opción del margen derecho de nombre “Crear Volumen”.

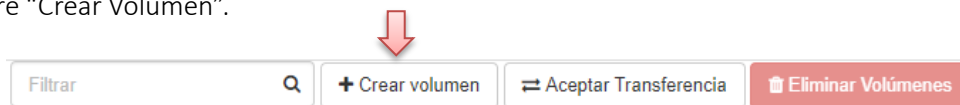


Ilustración 122: Servicio Cinder - Opción "Crear Volumen"

2. Se nos abrirá una ventana en donde deberemos definir el nombre del volumen, su identificador si lo deseamos y el tamaño. Opcionalmente se puede llevar a cabo otras acciones.

## Crear volumen ✕

**Nombre del volumen**

Volumen\_Prueba

**Descripción**

El volumen utilizado para la prueba

**Origen del volumen**

Sin origen, volumen vacío

**Tipo**

iscsi

**Tamaño (GiB) \***

5

**Zona de Disponibilidad**

nova

**Descripción:**

Los volúmenes son dispositivos de bloques que se pueden asociar a instancias.

**Descripción del Tipo de Volumen:**

**iscsi**

No description available.

**Límites del volumen**

Gibibytes total 7 of 1.000 GiB Used

Número de volúmenes 3 of 10 Used

Cancelar Crear volumen

Ilustración 123: Servicio Cinder - Crear volumen

Automáticamente, el volumen se asociará al proyecto en particular sin tener que especificarlo directamente.

2. Comprobación del volumen creado.

Nombre	Descripción	Tamaño	Estado	Tipo	Asociado a	Zona de Disponibilidad	Arrancable	Cifrado	Acciones
Volumen_Prueba	Volumen utilizado para la prueba	5GiB	Disponible	iscsi		nova	no	No	Editar volumen

Ilustración 124: Servicio Cinder - Comprobación volumen creado

## F.2.5. Servicio Nova

El componente Nova de OpenStack es el responsable de poner en marcha y gestionar las máquinas virtuales. En nuestro caso, nos centraremos en instanciar la imagen que hemos subido anteriormente gracias al servicio Glance, pero debiendo de hacer uso de un par de claves y de un grupo de seguridad para asegurarnos el acceso a la instancia.

### F.2.5.1. Claves

Para la creación de claves en OpenStack hay finitas posibilidades. La opción escogida es la creación propia de nuestras claves de seguridad. Estas claves se realizaron en el sistema operativo<sup>27</sup> Kali que tengo disponible como otra máquina virtual, haciendo uso de la siguiente instrucción: `ssh-keygen -t rsa cloudkey`.

Esta instrucción automáticamente originará dos ficheros. Uno de estos archivos representará la clave pública y el otro la clave privada. Llegados a este punto deberemos realizar lo siguiente:

i. Subir la clave pública a la plataforma OpenStack. Para ello nos dirigimos a “Proyecto-Compute-Pares de claves”. Seleccionamos la opción “Importar par de claves”.

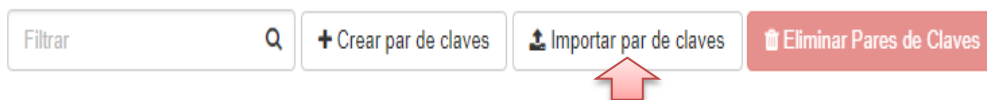


Ilustración 125: Claves - Importar claves

A continuación, dotamos de un nombre a las claves y copiamos el contenido de nuestra clave pública.

### Importar par de claves ✕

**Nombre de par de claves \***

**Clave pública \***

```
ssh-tsa
AAAAB3NzaC1yc2EAAAADAQABAAQCAQC+tn7I
sGuufQF/dlheA6ap1fgmil2JwpUi8BZgk+m3btS00
zOLKmhAk5nPEDGQxRtMtheiQmN3GGYp96GI2
VIVq3AwsD0Zd0P4mdkC/F2k0/Uegauvv8w7H5j
M1IF2+p7wrxjqVUIWW1rX388hCRZxsPouEbripN
pMUn65uRCkg2ms4xX2Na3VX5MTYEZ4AFIET7
K41QRKDBFhJNBvQed032ECdSv323lpOyTzLCZ
NMhiEZkilyz2jX2/P1S2K64d6H5+ZNWJvkQb5NA
+UB80yx8l6LE0tdqFy8stj4J7ADba9864eWrdVVnj
...
```

Los pares de clave son el método utilizado para ingresar en una instancia una vez que se ha lanzado.

Elija un nombre para el par de claves que pueda identificar posteriormente y copie la clave pública ssh en el espacio proporcionado.

Los pares de clave ssh se pueden generar con la instrucción `ssh-keygen`:

```
ssh-keygen -t rsa -f cloud.key
```

Esto genera un par de claves: una clave privada que debe custodiar (`cloud.key`) y una clave pública (`cloud.key.pub`). Copie el contenido del fichero de la clave pública aquí.

Tras lanzar una instancia, se ingresa con la clave privada (el usuario puede cambiar en función de la imagen utilizada):

```
ssh -i cloud.key <username>@<instance_ip>
```

Ilustración 126: Claves – Creación

Finalmente, comprobamos que se ha importado correctamente la clave.

cloudkey

11:9a:b0:17:33:df:ce:4b:ea:36:7a:aa:41:2c:5d:30

Eliminar Par de Clave

Ilustración 127: Claves – Comprobación

ii. En esta prueba se va a intentar establecer la conexión remota a partir de un sistema Windows. La manera que tenemos para conseguir esta conexión por medio del protocolo SSH, es a través de una aplicación específica, haciendo en este caso uso de la aplicación PuTTY. Para poder hacer uso en la misma de la clave privada creada anteriormente en el sistema Kali, deberemos de “generar” una nueva en base a la ya disponible, pero específica de la aplicación. Para ello se hace uso del software “PuTTYgen” en donde cargamos nuestra clave privada (“Load”) y asignándole o no una contraseña (“passphrase”), se define la clave privada específica de la aplicación (“Save private key”).

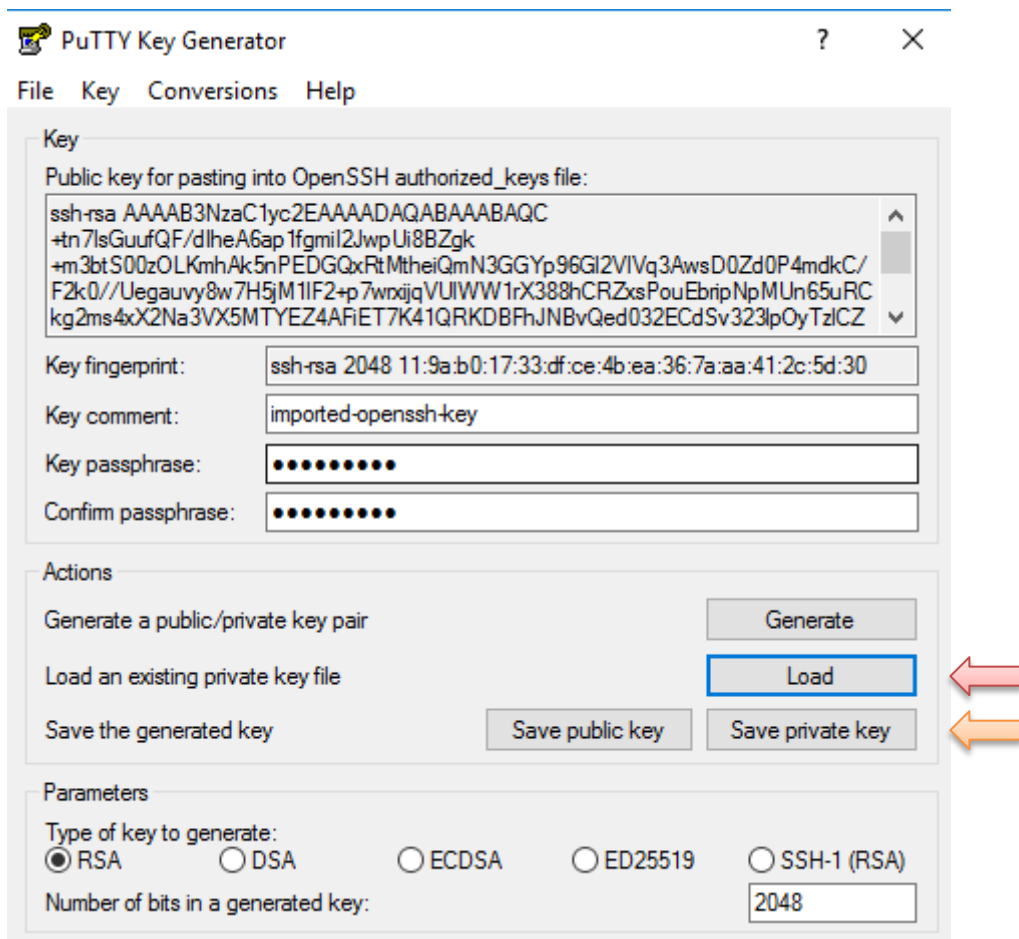


Ilustración 128: Claves – PuTTYgen

Destacar que se ha utilizado el mismo sistema criptográfico, RSA, que cuando se ha generado la clave en el sistema Kali.

### F.2.5.2. Grupo de seguridad

Disponiendo de las claves, deberemos definir un grupo de seguridad que nos permita que la instancia sea accesible y poder conectarnos por medio del protocolo SSH. Por lo tanto, realizaremos los siguientes pasos:

a. Creamos el grupo de seguridad por medio de la siguiente instrucción:

```
nova secgroup-create NOMBRE_GRUPO_SEGURIDAD "DESCRIPCIÓN".
```

b. Definimos las reglas de acceso y posibilidad de conexión SSH. Todo ello gracias a estos dos comandos respectivamente:

```
b.1. nova secgroup-add-rule NOMBRE_GRUPO_SEGURIDAD icmp -1 -1 0.0.0.0/0
```

```
b.2. nova secgroup-add-rule NOMBRE_GRUPO_SEGURIDAD tcp 22 22 0.0.0.0/0
```

### F.2.5.3. Instanciar

Vamos a describir a continuación los pasos que debemos seguir para instanciar una imagen en la plataforma OpenStack de manera adecuada.

1. Nos dirigimos a "Proyecto-Compute-Instancias" y se selecciona la opción "Lanzar instancia".

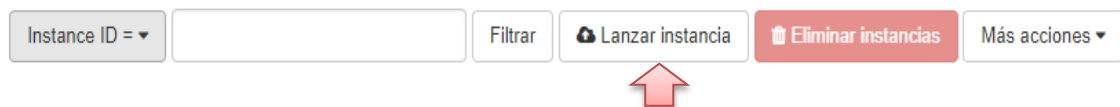


Ilustración 129: Servicio Nova - Opción "Lanzar instancia"



2. Definimos en la ventana emergente en un primer momento el nombre y el número de instancias que vamos a lanzar.

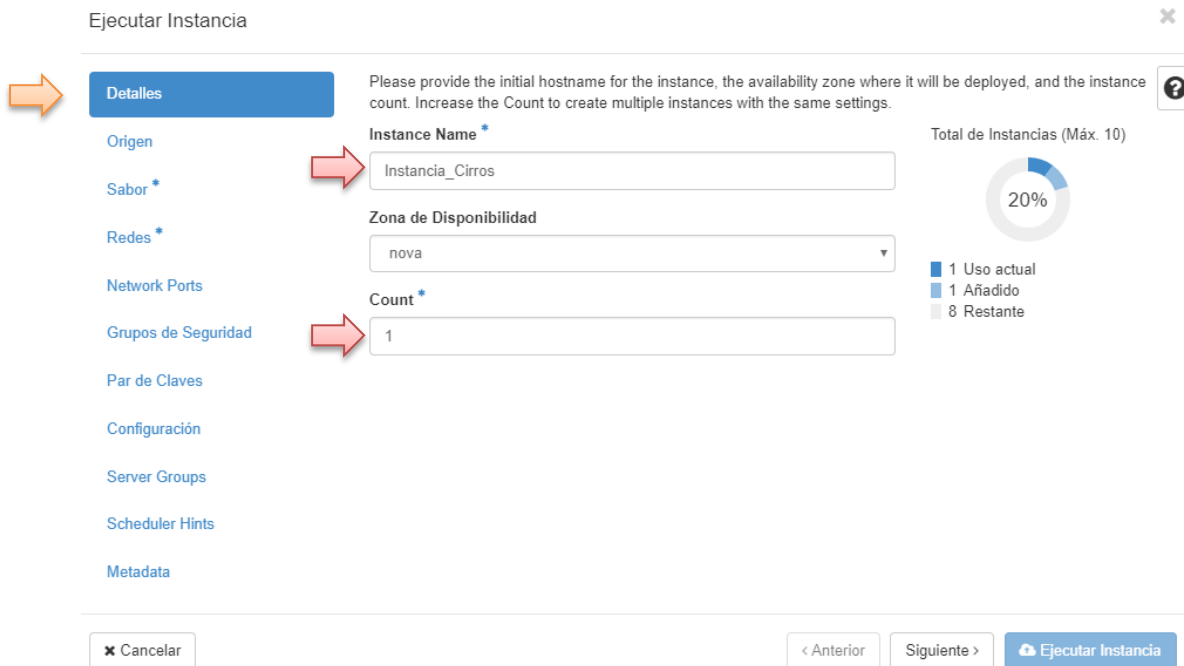


Ilustración 130: Servicio Nova - Detalles instancia

3. Escogemos la imagen que representará la instancia. También deberemos decidir si dotarle de un volumen automáticamente o no. En este caso, se ha optado por no crear un nuevo volumen porque queremos asociárselo posteriormente.

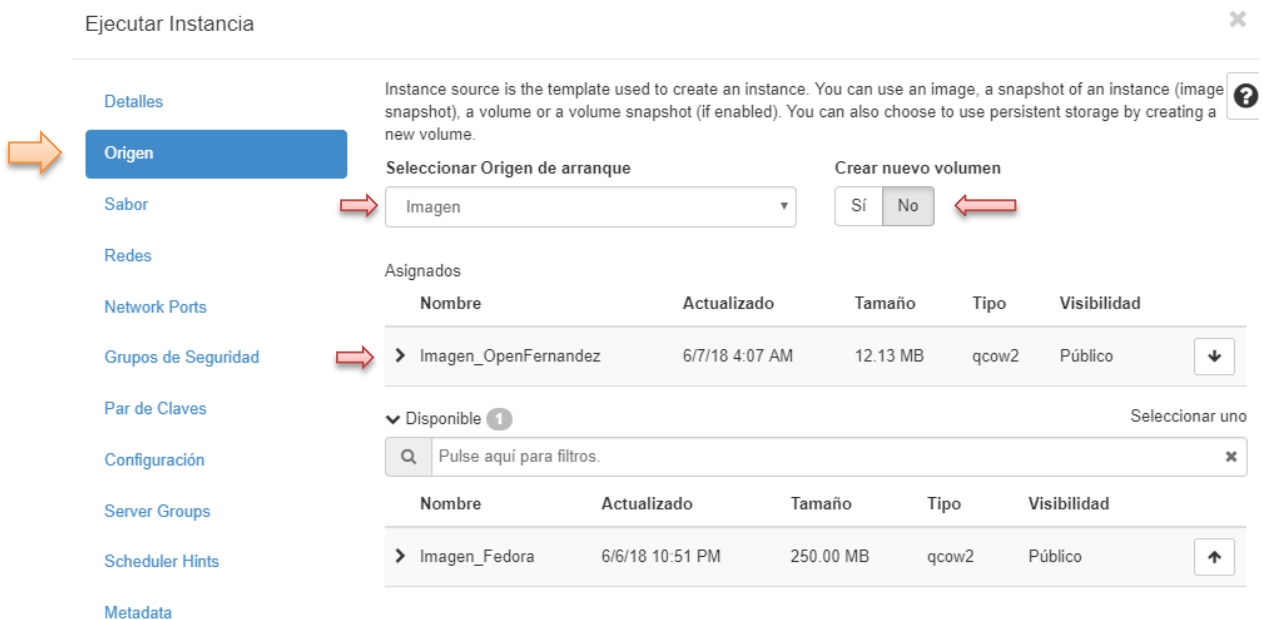


Ilustración 131: Servicio Nova - Origen instancia

4. Seleccionamos los recursos hardware (sabor) que va a requerir la imagen. Habrá que atender tanto a las necesidades de la imagen como de nuestra plataforma.

Ejecutar Instancia ✕

Detalles

Origen

**Sabor**

Redes \*

Network Ports

Grupos de Seguridad

Par de Claves

Configuración

Server Groups

Scheduler Hints

Metadata

Los sabores definen el tamaño que tendrá la instancia respecto a CPU, memoria y almacenamiento. ?

Asignados

Nombre	VCPUS	RAM	Total de Disco	Disco raíz	Disco efímero	Público	
> m1.tiny	1	512 MB	1 GB	1 GB	0 GB	Sí	↓

▼ Disponible 5 Seleccionar uno

Q Pulse aquí para filtros. ✕

Nombre	VCPUS	RAM	Total de Disco	Disco raíz	Disco efímero	Público	
> Sabor_Prueba	1	900 MB	5 GB	5 GB	0 GB	Sí	↑
> m1.small	1	2 GB	20 GB	20 GB	0 GB	Sí	↑
> m1.medium	2	4 GB	40 GB	40 GB	0 GB	Sí	↑
> m1.large	4	8 GB	80 GB	80 GB	0 GB	Sí	↑
> m1.xlarge	8	16 GB	160 GB	160 GB	0 GB	Sí	↑

Ilustración 132: Servicio Nova - Sabor instancia

5. Elegimos la red interna a la que asociamos nuestra instancia.

Ejecutar Instancia ✕

Detalles

Origen

Sabor

**Redes**

Network Ports

Grupos de Seguridad

Par de Claves

Configuración

Server Groups

Scheduler Hints

Metadata

Las Redes proveen los canales de comunicación para las instancias en la nube. ?

▼ Asignados 1 Seleccionar redes de las listadas abajo.

Red	Subredes Asociadas	Compartido	Estado del Administrador	Estado	
↕ 1 > mi_red	mi_subred	No	Arriba	Activo	↓

▼ Disponible 1 Seleccionar al menos una red

Q Pulse aquí para filtros. ✕

Red	Subredes Asociadas	Compartido	Estado del Administrador	Estado	
> ext-net	ext-subnet	No	Arriba	Activo	↑

Ilustración 133: Servicio Nova - Red interna de la instancia

6. Optamos por el grupo de seguridad que represente nuestra instancia. En base a ese grupo y las reglas que tenga definidas, la instancia podrá ser más o menos accesible.

Ejecutar Instancia

Seleccionar el grupo de seguridad para lanzar la instancia.

Asignados 1

Nombre	Descripción
grupo_seguridad	Grupo de seguridad

Disponibles 1 Seleccionar uno o más

Pulse aquí para filtros.

Nombre	Descripción
default	Grupo de seguridad predeterminado

Ilustración 134: Servicio Nova - Grupo de seguridad de la instancia

7. Nos decantamos por la clave de seguridad que nos permitirá en un momento dado acceder a esa instancia por medio del protocolo SSH.

Ejecutar Instancia

A key pair allows you to SSH into your newly created instance. You may select an existing key pair, import a key pair, or generate a new key pair.

+ Crear Par de Claves    ⬇ Importar Par de Claves

Asignados

Mostrando 1 artículo

Nombre	Fingerprint
cloudkey	11:9a:b0:17:33:df:ce:4b:ea:36:7a:aa:41:2c:5d:30

Disponibles 1 Seleccionar uno

Pulse aquí para filtros.

Nombre	Fingerprint
clavePrueba	e9:b5:d9:ac:d0:eb:17:62:ab:f6:cc:fc:2f:aa:46:a8

Cancelar    < Anterior    Siguiente >    Ejecutar Instancia

Ilustración 135: Servicio Nova - Claves instancias

8. Creamos la instancia y comprobamos su correcta disposición. Si es así, asociamos en primer lugar un volumen a la instancia para luego poder definir sobre ella una dirección IP flotante que deberá ser creada. Para todo ello nos dirigimos al margen derecho, más en concreto a la opción “Acciones”.

9. Finalmente, comprobamos que la instancia tenga todo lo requerido para su correcto uso y funcionamiento.

<input type="checkbox"/>	Nombre de la instancia	Nombre de la imagen	Dirección IP	Sabor	Par de claves	Estado	Zona de Disponibilidad	Tarea	Estado de energía	Tiempo desde su creación	Acciones
<input type="checkbox"/>	Instancia_Fedora	Imagen_Fedora	20.0.0.11 IPs flotantes: 192.168.1.154	Sabor_Prueba	cloudkey	Pausada	nova	Ninguno	Pausada	2 horas, 53 minutos	Crear instantánea ▾

Ilustración 136: Servicio Nova - Comprobación instancia

#### F.2.5.4. Comprobación

Para asegurarnos del correcto funcionamiento, en primer lugar se ha llevado a cabo un “ping”<sup>9</sup> entre la máquina local y la instancia. De esta forma comprobamos la accesibilidad de la instancia lanzada.

```
C:\Users\ehu\QEMU>ping 192.168.1.159

Haciendo ping a 192.168.1.159 con 32 bytes de datos:
Respuesta desde 192.168.1.159: bytes=32 tiempo=7ms TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63

Estadísticas de ping para 192.168.1.159:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 7ms, Media = 1ms
```

Ilustración 137: Comprobación SSH – Ping

Asegurado ello, podemos realizar la conexión SSH desde nuestra aplicación PuTTY.

1. Para ello en la opción "Session" establecemos la dirección IP de la instancia.

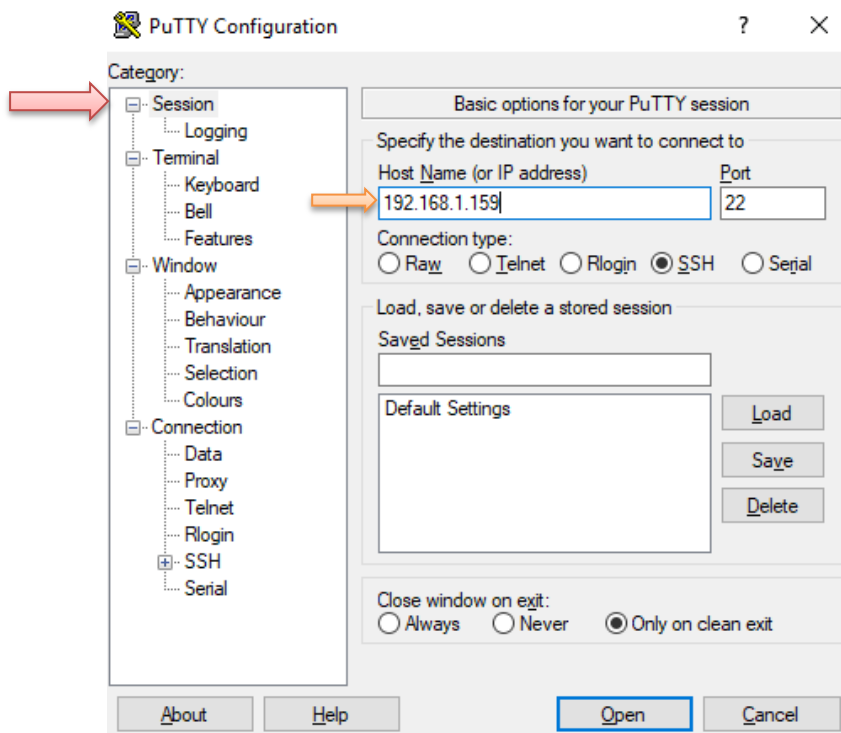


Ilustración 138: Comprobación SSH - Dirección IP en PuTTY

2. A continuación en "Session-Connection-SSH" cargamos nuestra clave privada.

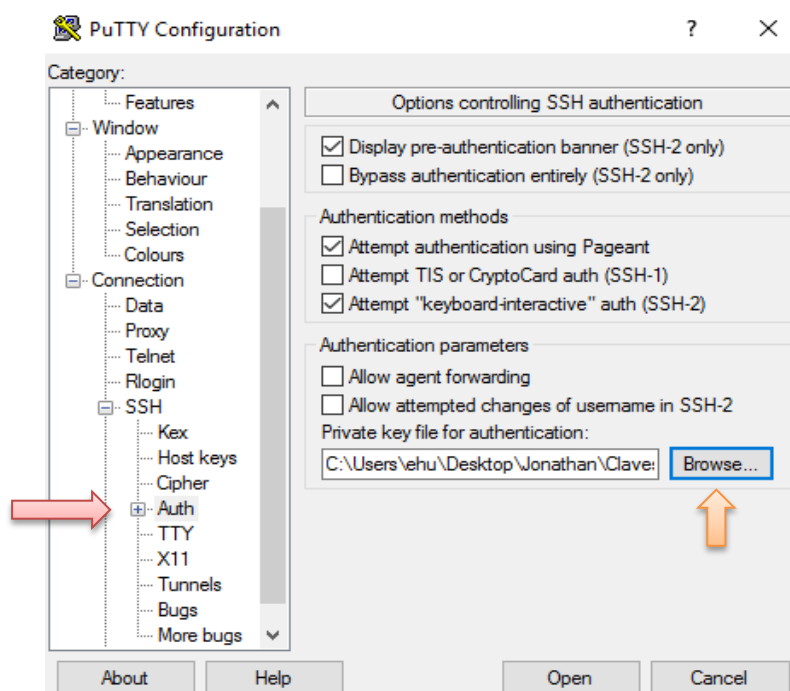


Ilustración 139: Comprobación SSH - Importar clave privada PuTTY

3. Finalmente, iniciamos la sesión como usuario “fedora” y nos solicitará la contraseña de la clave en caso de haberla establecido.

```
fedora@instancia-fedora:~  
login as: fedora  
Authenticating with public key "imported-openssh-key"  
Passphrase for key "imported-openssh-key":  
[fedora@instancia-fedora ~]$
```



Ilustración 140: Comprobación SSH – PuTTY

Como bien podemos apreciar nos encontramos en la instancia desde la aplicación PuTTY al nombrarse en el “PROMPT” el nombre de nuestra instancia además del usuario con el que hemos iniciado sesión.

### F.2.6. Conclusiones

El resultado como bien hemos comprobado ha sido el esperado. Sin embargo, para alcanzarlo ha existido un minucioso estudio porque cualquier detalle es importante. Recomiendo seguir al pie de la letra todos los pasos comentados anteriormente y en caso de que no os funcione volver a repasar lo realizado.

Es un buen resultado ya que con ello hemos comprobado una de las facultades que tiene OpenStack. Con ello podemos asegurarnos que cualquier sistema lanzado desde la plataforma OpenStack puede ser accedido remotamente desde nuestro sistema por medio del protocolo de comunicaciones SSH, proporcionándonos de esa forma un escritorio virtual.

Finalmente me gustaría destacar que esta prueba ha servido además para tener una primera toma de contacto con la plataforma OpenStack y conocer aspectos los cuales cabe mencionar. Por esa razón, en el apartado siguiente se destacarán algunos de ellos.

### F.3. Otras cuestiones

A partir de este apartado se pretende reflejar todos los inconvenientes y aspectos que se recomiendan tener en consideración y que se han encontrado en la realización de esta prueba. Son problemas que en su mayoría requieren una mayor formación en lo que a la administración de la plataforma OpenStack se refiere.

- Roles

Cuando creamos cualquier usuario, le asignamos a este un rol para poder trabajar sobre un proyecto en cuestión. Es decir, por medio de ese rol se determinan el conjunto de acciones que se estiman oportunas que pueda realizar el usuario en un proyecto. Podemos hacer uso de los roles propios de la plataforma OpenStack o podemos crearlos. Sin embargo, no estimo oportuno muchas de las facilidades que ofrecen a un usuario que no es administrador. Me explico mejor con un ejemplo:

- Si nos dirigimos al servicio Horizon con un usuario que no es administrador en uno de sus proyectos, a este se le permite crear una red, un router, eliminar sabores, subir una imagen, etc. ¿Se debe permitir que ese usuario realice ese conjunto de acciones?

Según mi criterio no. El usuario, en el caso de un alumno, lo único que debería realizar es el poder lanzar su instancia y poder hacer así uso de su escritorio virtual, por ejemplo. Por esa razón, se decidió conocer como limitar ello.

- ❖ Por medio de la interfaz web resulto imposible porque no ofrece ninguna posibilidad.
- ❖ La línea de comandos CLI, en cambio, proporciona un fichero en donde se puede restringir lo que pueden realizar los usuarios. Estos archivos, que reciben el nombre de "policy.json", y que existe uno por cada componente de OpenStack, determinan unas reglas que especifican lo que puede hacer un usuario en base al rol que tiene asignado en ese servicio en cuestión. Una vez que se altera cualquier característica de ese fichero, se reinicia el servicio. De esta manera, y siendo siempre administradores, se alcanza el objetivo marcado.

Sin embargo, en el intento de realizarlo sobre el servicio Neutron, me he encontrado con muchas dificultades debiendo incluso de hacer uso de la copia de seguridad que se había realizado, para recuperar el estado anterior de mi plataforma. Por lo tanto, no se ha alcanzado la meta y es una de las cuestiones que quedan pendientes de estudio.

- Carpeta de usuario

Si nos dirigimos a la carpeta raíz de nuestro sistema CentOS, nos encontramos con los siguientes archivos y directorios.

```
[root@openstacklocalhost ~]# ls
anaconda-ks.cfg  keystoneadmin  packstack-answers-20180323-114125.txt
```

Ilustración 141: Inconvenientes SSH - Carpeta raíz

Formulemos la siguiente pregunta, ¿no deberíamos tener disponible otro fichero que represente la información del usuario creado?

En respuesta a la pregunta anteriormente formulada, lo más lógico es pensar que sí. Sin embargo, no aparece reflejada en la carpeta raíz y por lo tanto no somos capaces de acceder como ese usuario desde la línea de comandos. Solo podemos acceder entonces, desde la interfaz web. Por lo cual, a estas alturas, este es otro aspecto que queda pendiente de estudio.

- Creación de redes

Quando se creen redes, internas o externas, o routers virtuales, se deberá reflejar a que proyecto en cuestión pertenece. Esto es así porque en caso de no mencionarlo, se dirigirán y se crearán en el proyecto principal del usuario. Es decir, contemplemos la siguiente imagen:

```
unset OS_SERVICE_TOKEN
export OS_USERNAME=admin
export OS_PASSWORD=4f806856436e4966
export OS_AUTH_URL=http://192.168.1.50:5000/v3
export PS1='\u@\h \W(keystone_admin)]\$ '
export OS_PROJECT_NAME=admin
export OS_USER_DOMAIN_NAME=Default
export OS_PROJECT_DOMAIN_NAME=Default
export OS_IDENTITY_API_VERSION=3
```

Ilustración 142: Inconvenientes SSH - Archivo "admin"

Mencionemos la instrucción utilizada:

```
[root@openstack ~(keystone_admin)]# openstack network create mi_red
```

Ilustración 143: Inconvenientes SSH - Asignación a proyecto por defecto

En este caso, al no indicarnos nada y haber iniciado sesión como administrador, esta red interna será creada en nuestro proyecto por defecto, es decir, "admin". Si en cambio, quisiéramos añadirlo a un proyecto en particular, añadiríamos al final de ese comando lo siguiente:

“--project NOMBRE\_PROYECTO”

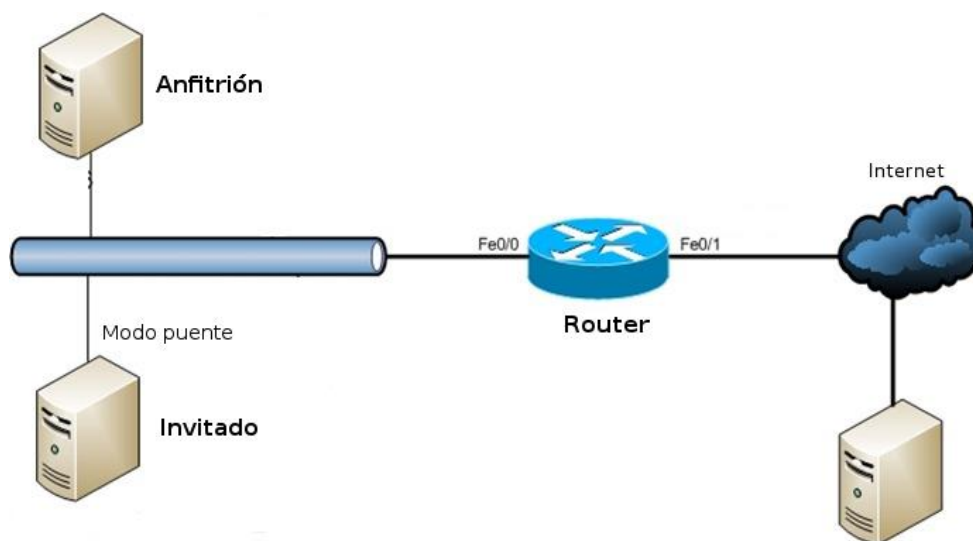
De esta forma se asignaría a ese proyecto en concreto debiendo para ello existir.



Esto mencionado resultado de gran utilidad para comprender un poco más el funcionamiento de la plataforma OpenStack y evitar de esta forma posibles confusiones. Todo lo comentado se debe tener en consideración para otros aspectos además de la configuración de red.

- Red Anfitrión

Actualmente, en base a la instalación que hemos realizado, el modo de red que tiene asociada nuestra máquina virtual CentOS que nos proporciona esa plataforma OpenStack, es un adaptador puente. Este modo simula que la tarjeta virtual está conectada al mismo switch que la tarjeta física del anfitrión, por lo tanto, la máquina virtual se va a comportar como si fuese un equipo más dentro de la misma red física en la que está el equipo anfitrión.



21

Ilustración 144: Inconvenientes SSH - Modo adaptador puente

Conocido el modo de red utilizada para nuestra máquina virtual, es correcto pensar que una vez hemos lanzado una instancia esta podrá tener acceso a la red y ser reconocida por nuestra máquina local. Sin embargo, por causas inexplicables que se asocian en su mayor parte a inconvenientes encontrados en la aplicación VirtualBox con este modo de red, esto no sucede de esta manera. Incluso, la instancia en ningún caso es reconocida por mi equipo local y por lo tanto no puedo acceder a ella.

Entonces, ¿cómo hemos obtenido los resultados anteriormente visualizados? Cambiando el modo de red a solo anfitrión. Este modo se utiliza para crear una red a la que pertenecerá también el equipo anfitrión, algo que no sucede en el modo Red Interna. Esto es posible a que no utiliza la tarjeta física de red del anfitrión, sino que se crea una red virtual que estará conectada al mismo switch virtual al que estarán conectadas las tarjetas de red de las máquinas virtuales con el mismo modo de red. Ello nos permitirá que las instancias sean reconocidas en la máquina local con el inconveniente que no tendrán acceso a la red.

<sup>21</sup> <http://fpg.x10host.com/VirtualBox/AdaptadorPuente.png>

Expliquemos a continuación, los pasos seguidos para lograr este modo:

- a. Administramos una red en modo Solo-anfitrión que se encontraran en la opción “Herramientas globales de VirtualBox” en el margen derecho de la propia aplicación.

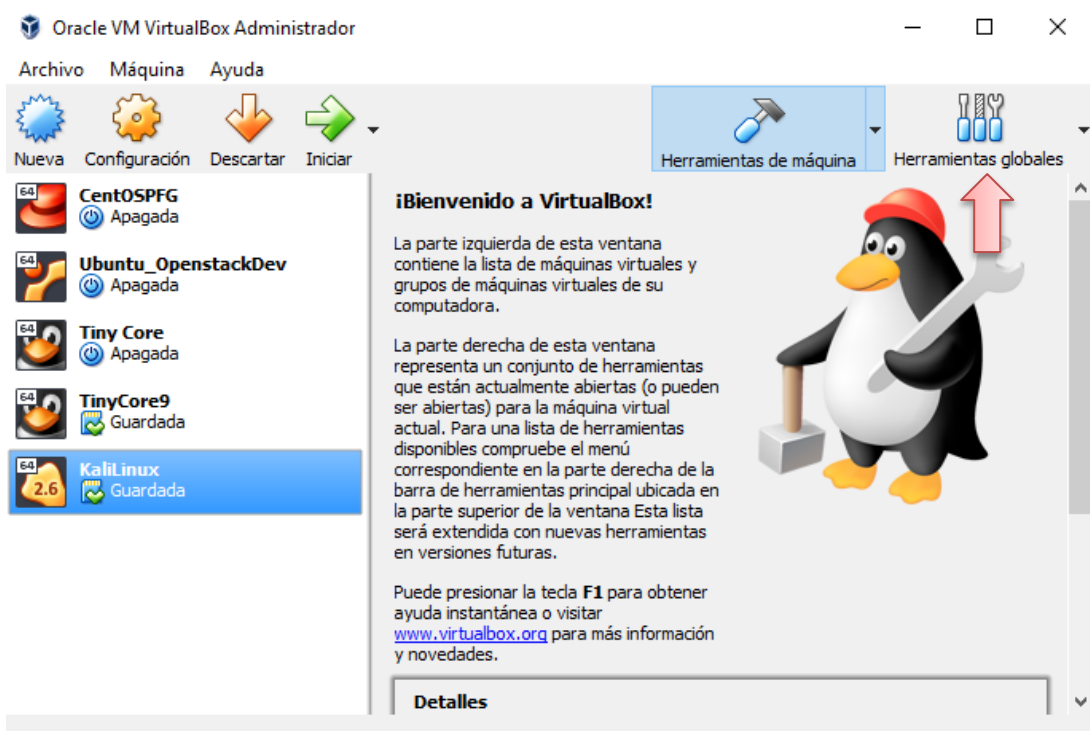


Ilustración 145: Inconvenientes SSH - Administrar red anfitrión (1)



Ilustración 146: Inconvenientes SSH - Administrar red anfitrión (2)

b. Creamos una red de estas características, debiéndose reflejar un resultado similar al siguiente. La dirección IP que se establezca, será la que tendrá nuestro equipo local en ese modo.

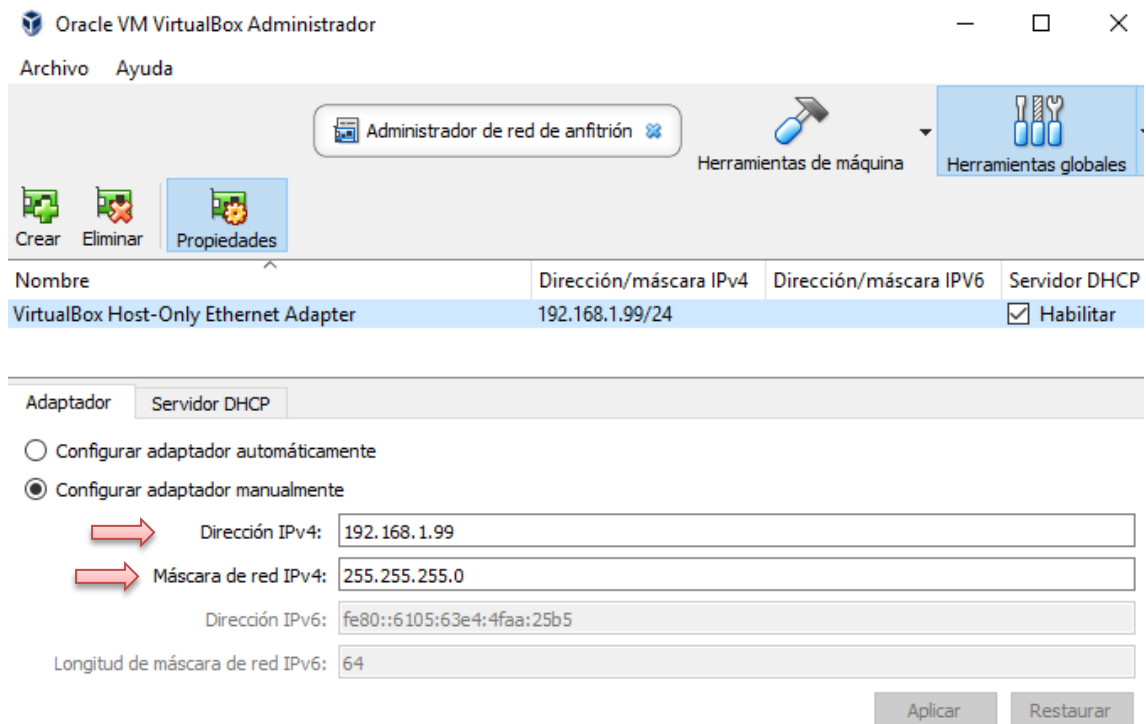


Ilustración 147: Inconvenientes SSH - Creación red anfitrión

c. Nos dirigimos a la configuración de nuestra máquina y en la opción de red establecemos el modo de red a “Solo-anfitrión”.

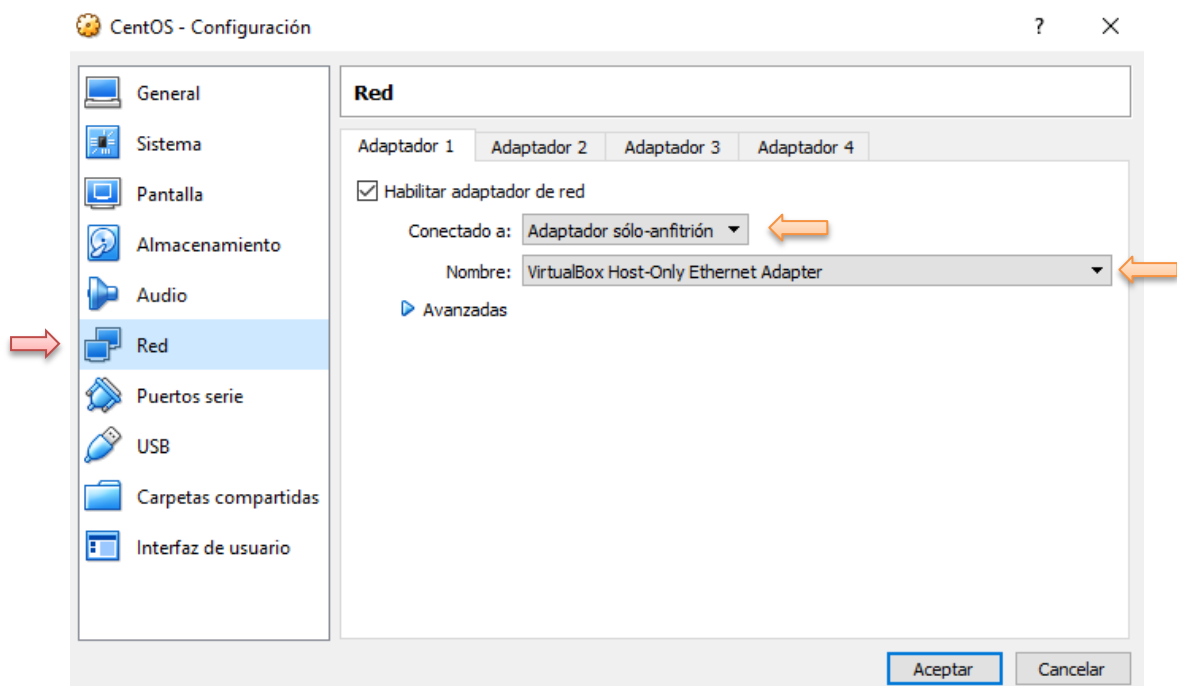


Ilustración 148: Inconvenientes SSH - Modo red solo anfitrión

## G. Instanciación y aplicación de escritorio VNC

### G.1. Introducción

A partir del siguiente anexo, se intentará explicar de una manera más concreta como se puede instanciar una imagen y acceder a ella remotamente por medio de la aplicación VNC. Para lograr esta meta, nos basaremos en el procedimiento descrito en el apartado 6, más en concreto en el apartado 6.2., haciendo uso de un sistema Tiny Core el cual se explicará en el próximo apartado.

El fin por tanto de esta prueba es asentar las bases sobre el funcionamiento de VNC en OpenStack y cómo lograr disponer de un escritorio virtual al cual poder acceder remotamente por medio de dicha aplicación. Esto originará diferentes escenarios sobre los cuales aprenderemos en que consiste VNC y las necesidades que tiene, así como las limitaciones y virtudes de la plataforma.

Todos los logros y dificultades que se vayan encontrando se irán detallando, además de las alternativas que se han hecho uso, de una manera clara y precisa.

### G.2. Conceptos previos

#### G.2.1. Tiny Core Linux

En la primera prueba que se ha llevado a cabo, hemos utilizado, un sistema operativo<sup>27</sup> conocido como Fedora. Se trata de un sistema muy ligero que está pensado para usarse en las pruebas de virtualización, debido a que sus funcionalidades son muy básicas. Sin embargo, en esta segunda prueba que se pretende realizar, se va requerir también de un sistema que necesite de pocos recursos pero que disponga de una interfaz gráfica, cosa que el sistema anteriormente utilizado no dispone. Por lo tanto, deberemos de pensar en un sistema que se adecue a estas necesidades.

Dada la situación en la nos encontramos, se ha pensado en el sistema operativo<sup>27</sup> Tiny Core Linux, o TCL. Es un sistema minimalista centrado en proveer un sistema base con el núcleo Linux, BusyBox<sup>5</sup>, FLTK<sup>15</sup> y algún otro software minimalista. La distribución es especialmente notable por su reducido tamaño, se encuentra bajo la licencia de GNU GPLv2 y dispone de muchas variantes:

- ❖ Tiny Core: Es la opción más recomendada para nuevos usuarios. Incluye una interfaz de usuario gráfica dinámica FLTK<sup>15</sup>/FLWM.
- ❖ Core: También conocida como “Micro Core Linux”, es una variante más pequeña de Tiny Core sin entorno gráfico, pero se pueden añadir extensiones adicionales para crear un sistema con un entorno de escritorio.
- ❖ Core Plus: Es una imagen de la instalación de TCL y no la distribución en sí. Está compuesta de Tiny Core más funcionalidades adicionales.
- ❖ dCore: Es un núcleo hecho desde archivos compatibles con Debian.
- ❖ piCore: Orientado su uso para la Raspberry Pi.

Los requerimientos del sistema para dos de sus variantes más importantes, Tiny Core y Core, son las que se especifican en la siguiente tabla:

Variante	Tipo de Configuración	Mínima	Recomendada
Tiny Core	RAM	46MB	128MB + swap
Core	RAM	28MB	

Tabla 33: Tiny Core – Requisitos

Para realizar una correcta descarga de este sistema se sugiere visitar el siguiente enlace:

- ✓ <http://distro.ibiblio.org/tinycorelinux/downloads.html>

## G.3. OpenStack – VNC

Como se ha mencionado en el apartado 6, más en concreto en el apartado 6.2., OpenStack proporciona dos herramientas para poder conectarnos a una instancia por VNC: la consola SPICE y el proxy VNC. Debido a que la primera de las alternativas se encuentra en una fase prueba y de estudio, se ha visto más conveniente decantarse por el proxy que proporciona la plataforma. Por esa razón, veremos a continuación como se ha dividido esta prueba para poder alcanzar el resultado deseado.

### G.3.1. Tiny Core Linux

Teniendo a nuestra disposición un sistema Tiny Core Linux, lo más común sería subirlo como una nueva imagen a la plataforma OpenStack y a continuación, instanciarla. Una vez hayamos logrado ello, nos dispondríamos a acceder a dicha instancia de manera remota por medio del protocolo SSH, y seguidamente instalar el paquete que proporciona tener disponible en ese sistema el servidor VNC. Finalmente, accederíamos a dicha instancia por medio de la aplicación VNC Viewer en nuestro sistema Windows local. Sin embargo, según se vio en el apartado de “Impedimentos” del anexo anterior, nuestra máquina virtual CentOS que nos proporciona la plataforma está definida con un modo de red solo anfitrión. Esto significa que las instancias pueden ser reconocidas por nuestro sistema, pero no tienen acceso a la red. En relación con nuestro caso actual, no podremos instalar ese paquete que nos va a proporcionar el servidor VNC en el sistema Tiny Core Linux desde la propia plataforma. Ante esta situación, ¿Qué podemos hacer?

Habrá que interactuar con el sistema instalando ese paquete antes de subirla como una nueva imagen de OpenStack. Pero para realizar ello, ¿tendremos que crear una nueva máquina virtual? No tendríamos por qué crearla. Podríamos hacer uso de un emulador que nos proporcione acceso a ese sistema, establecer el paquete y comprobar que funciona. Una vez alcancemos ese resultado, podremos empezar a trabajar con nuestra plataforma OpenStack.

### G.3.2.1. Emulador QEMU

El emulador que vamos a escoger para nuestro sistema Windows va a ser QEMU debido a las facilidades que proporciona. Para su obtención recomiendo visitar el siguiente enlace debiendo escoger aquel que mejor se adecue a nuestro sistema local:


- ✓ <https://qemu.weilnetz.de/w64/>

Una vez realizada la instalación, lo siguiente sería crear una carpeta del mismo nombre que el emulador en la ubicación que mejor convenga por la situación y disponer en ella del sistema Tiny Core Linux. A continuación, lanzamos el sistema Tiny Core de la siguiente forma:

```
C:\Users\ehu\QEMU>"c:\Program Files\qemu\qemu-system-x86_64.exe" -hda linux-tinycore-3.8.2.img -net nic,model=rtl8139 -net user,id=net0,hostfwd=tcp:127.0.0.1:5901-:5900
```

Ilustración 149: QEMU - Lanzar imagen

Esto originará una ventana subyacente donde se cargará el sistema en cuestión. Una vez lo tengamos disponible instalamos el siguiente paquete: v0vncserver para el sistema Tiny Core. Lo siguiente sería configurar el sistema para que cuando sea iniciado se lance automáticamente el servidor VNC. Sin embargo, no se ha encontrado la forma de alcanzar ese resultado y se ha pensado en la creación de un script que haga esa tarea.



```
Terminal
tc@box:~$ ls
launch.sh
tc@box:~$ cat launch.sh
x0vncserver -PasswordFile=/home/tc/.vnc/passwd 2> /dev/null
tc@box:~$ ./launch.sh
```

A red arrow points to the first line of the terminal output, `launch.sh`.

Ilustración 150: QEMU - Lanzar VNC Server Script

Finalmente se comprueba que se puede acceder al sistema desde VNC Viewer. Para ello primero se instala el cliente VNC en el sistema local, a continuación se crea una nueva sesión y finalmente se establece la conexión.

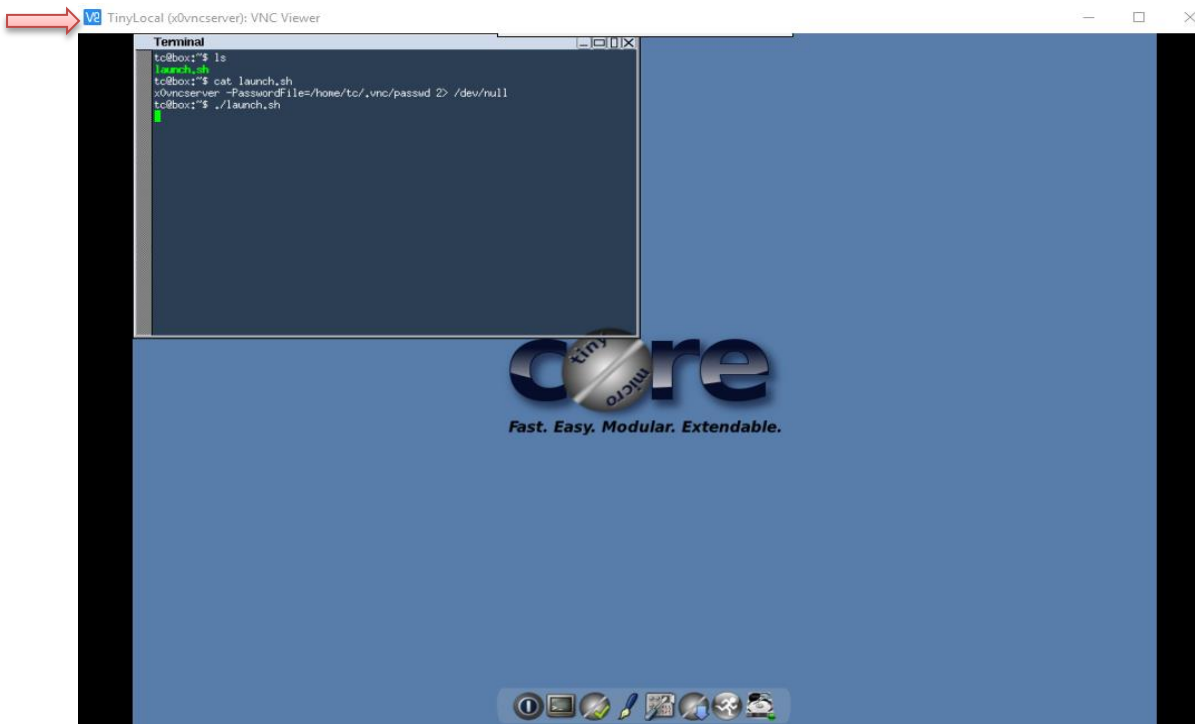


Ilustración 151: QEMU - Comprobación conexión VNC

Como bien podemos observar, hemos sido capaces de acceder al sistema haciendo uso de la aplicación VNC. Para guardar todos los cambios realizados se hace uso del siguiente comando propio del sistema Tiny Core: “filetool.sh -b”.

```
tc@box:~$ filetool.sh -b
Backing up files to /mnt/hda1/tce/mydata.tgz tc@box:~$
```

Ilustración 152: QEMU - Guardar cambios Tiny Core

El inconveniente se va a encontrar entonces es que cuando se suba a la plataforma OpenStack, deberemos acceder a esa instancia y ejecutar el script para disponer del servidor VNC.

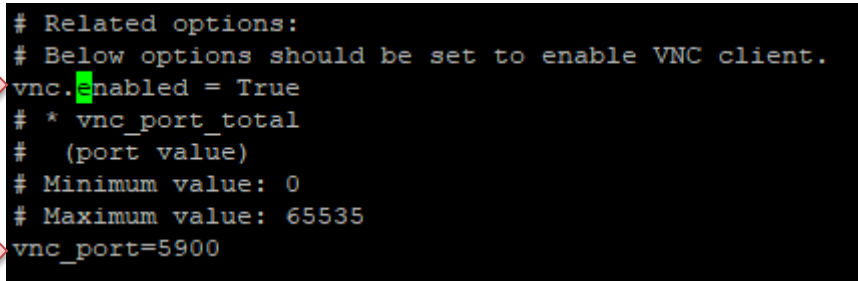
### G.3.2.2. OpenStack

La forma más adecuada que se ha pensado de realizar esta prueba, ha sido dividiéndola en diferentes partes, de tal manera que se vea de una manera clara y ordenada cada uno de los aspectos que hay que considerar para realizar esta prueba correctamente. Habrá muchos aspectos que se han mencionado en documentos anteriores y que se volverán a nombrar y poner en práctica. Todo ello basado en el sistema Tiny Core Linux que hemos mencionado en el apartado anterior.

#### a) Configuración del fichero “nova.conf”

Dentro del fichero que se encuentra ubicado en el directorio de configuración “/etc/nova”, abra que realizar una supervisión y mejora de su contenido, de tal manera que activemos el proxy VNC que proporciona la plataforma OpenStack. Veamos las mejoras realizadas en base al estudio anteriormente realizado:

1. Activamos el proxy VNC y lo definimos en el puerto 5900.

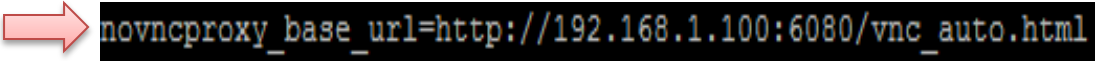


```
# Related options:
# Below options should be set to enable VNC client.
vnc.enabled = True
# * vnc_port_total
# (port value)
# Minimum value: 0
# Maximum value: 65535
vnc_port=5900
```

The screenshot shows a terminal window with a black background and white text. Two red arrows point to the lines 'vnc.enabled = True' and 'vnc\_port=5900'. The text is as follows:

Ilustración 153: Configuración VNC - Activación proxy

2. Comprobamos la ubicación del proxy de la consola VNC, que deberá corresponderse con la de la plataforma OpenStack.

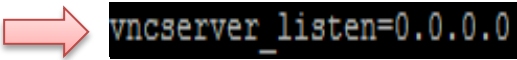


```
novncproxy_base_url=http://192.168.1.100:6080/vnc_auto.html
```

The screenshot shows a terminal window with a black background and white text. A red arrow points to the line 'novncproxy\_base\_url=http://192.168.1.100:6080/vnc\_auto.html'. The text is as follows:

Ilustración 154: Configuración VNC - Ubicación del proxy

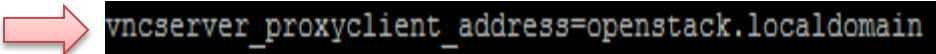
3. Comprobación de la dirección IP en la cual la instancia debería escuchar, y de la dirección a la que los clientes proxy deberían conectarse.



```
vncserver_listen=0.0.0.0
```

The screenshot shows a terminal window with a black background and white text. A red arrow points to the line 'vncserver\_listen=0.0.0.0'. The text is as follows:

Ilustración 155: Configuración VNC - Dirección IP instancia



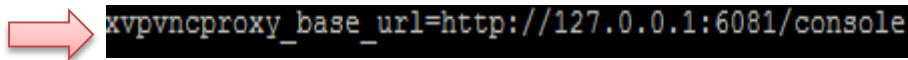
```
vncserver_proxyclient_address=openstack.localdomain
```

The screenshot shows a terminal window with a black background and white text. A red arrow points to the line 'vncserver\_proxyclient\_address=openstack.localdomain'. The text is as follows:

Ilustración 156: Configuración VNC - Dirección IP cliente



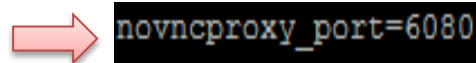
4. Comprobación de la ubicación del proxy de la consola nova xvp VNC.



```
xvpvncproxy_base_url=http://127.0.0.1:6081/console
```

Ilustración 157: Configuración VNC - Ubicación consola nova xvp

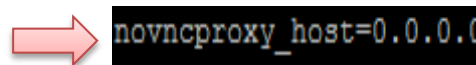
5. Determinar el puerto en el que escuchar las solicitudes entrantes.



```
novncproxy_port=6080
```

Ilustración 158: Configuración VNC - Puerto solicitudes entrantes

6. Determinar el host en el que escuchar las peticiones entrantes.



```
novncproxy_host=0.0.0.0
```

Ilustración 159: Configuración VNC - Host peticiones entrantes

De esta manera tendríamos el proxy VNC habilitado en la plataforma OpenStack. Incidir en que los aspectos mencionados, como las dirección IP, podrán variar dependiendo de la situación en la que nos encontremos.

b) Instancia – Tiny Core Linux

Lo siguiente a realizar será subir la imagen del sistema Tiny Core que hayamos descargado a la plataforma OpenStack, para a continuación instanciarla.

1. Subir imagen.



admin	imagen_tinycore	Imagen	Activo	Público	No	RAW	27.50 MB	Launch
-------	-----------------	--------	--------	---------	----	-----	----------	--------

Ilustración 160: Tiny Core - Subir imagen

En mi caso, se le ha denominado “imagen\_tinycore”, el formato de disco será “raw” y se encuentra tanto visible como en modo protegido.

Como bien podemos apreciar su tamaño es relativamente pequeño para ser un sistema con interfaz gráfica, de ahí que se le considere como un sistema orientado a realizar pruebas.

2. Grupo de seguridad. Las reglas que tenemos definidas en el grupo de seguridad actualmente no recogen la del puerto VNC (5900). Por ese motivo, nos dirigimos a la opción “Red – Grupos de seguridad” y añadimos una nueva regla.

**Agregar regla** ✕

**Regla \***

**Dirección**

**Puerto abierto \***

**Puerto ?**

**Remoto \* ?**

**CIDR ?**

**Descripción:**  
 Las reglas definen el tráfico permitido a las instancias asociadas al grupo de seguridad. Una regla de un grupo de seguridad contiene tres partes principales:  
**Regla:** Puede especificar una plantilla de reglas deseada o usar reglas TCP, UDP e ICMP personalizadas.  
**Puerto abierto/Rango de puertos** Para las reglas de TCP y UDP puede optar por abrir un solo puerto o un rango de ellos. La opción "Rango de puertos" le proporcionará el espacio para especificar tanto el puerto de comienzo como de final del rango. Para las reglas de ICMP por el contrario debe especificar el tipo y código ICMP en los espacios proporcionados.  
**Remoto:** Debe especificar el origen del tráfico a permitir a través de esta regla. Lo puede hacer bien con el formato de un bloque de direcciones IP (CIDR) o especificando un grupo de origen (Grupo de Seguridad). Al seleccionar un grupo de seguridad como origen, se permitirá que cualquier instancia de ese grupo de seguridad pueda acceder a cualquier otra instancia a través de esta regla.

Ilustración 161: Tiny Core - Grupo de seguridad VNC

Comprobamos una vez creada, que disponemos de todas las reglas necesarias para que esa instancia pueda ser reconocida por el sistema local (ICMP), y accedida por medio del protocolo SSH y la aplicación VNC.

<input type="checkbox"/>	Dirección	Tipo Ethernet	Protocolo IP	Rango de puertos	Prefijo de IP Remota	Grupo de Seguridad Remoto	Acciones
<input type="checkbox"/>	Saliente	IPv6	Cualquier	Cualquier	:::0	-	<a href="#">Eliminar Regla</a>
	<input type="checkbox"/>	Saliente	IPv4	Cualquier	0.0.0.0/0	-	<a href="#">Eliminar Regla</a>
	<input type="checkbox"/>	Entrante	IPv4	ICMP	Cualquier	0.0.0.0/0	<a href="#">Eliminar Regla</a>
	<input type="checkbox"/>	Entrante	IPv4	TCP	22 (SSH)	0.0.0.0/0	<a href="#">Eliminar Regla</a>
<input type="checkbox"/>	Entrante	IPv4	TCP	5900	0.0.0.0/0	-	<a href="#">Eliminar Regla</a>
<input type="checkbox"/>	Entrante	IPv4	UDP	53	0.0.0.0/0	-	<a href="#">Eliminar Regla</a>

Ilustración 162: Tiny Core - Grupos de seguridad disponible

3. Instanciar la imagen. Esta instancia se encontrará en la subred interna que se creó en la primera prueba, y por lo tanto se le determinará una dirección IP fija en particular. Además, deberemos asociarle unos recursos hardware, la imagen correspondiente al sistema Tiny Core, el grupo de seguridad y la clave de acceso. A continuación, deberemos asociarle un volumen y una dirección IP flotante para poder ser accesible desde el exterior.

Nombre de la instancia	Nombre de la imagen	Dirección IP	Sabor	Par de claves	Estado	Zona de Disponibilidad	Tarea	Estado de energía	Tiempo desde su creación	Acciones
<input type="checkbox"/> Instancia_TinyCore	imagen_tinycore	20.0.0.8 IPs flotantes: 192.168.1.159	m1.tiny	cloudkey	Activo	nova	Ninguno	Ejecutando	2 días	Crear instantánea ▾

Ilustración 163: Tiny Core - Instanciar imagen

Podemos ver incluso como se refleja esa instancia en nuestra red, por medio de la opción que nos proporciona OpenStack en “Topología de Red”.

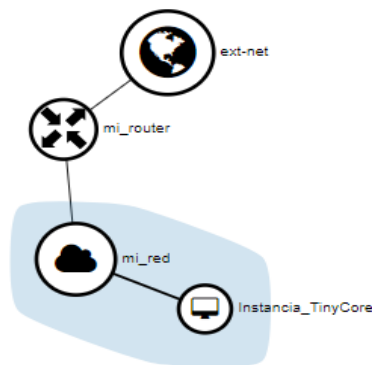


Ilustración 164: Tiny Core - Topología de red

Como bien podemos presenciar, tenemos una red interna en la cual se encuentra nuestra instancia de la imagen correspondiente al sistema Tiny Core, enlazada a su vez por medio de un router virtual a la red externa.

Para comprobar que efectivamente es reconocida por nuestro sistema local hacemos un “ping”<sup>9</sup> desde el mismo a la propia instancia.

```
C:\Users\ehu\QEMU>ping 192.168.1.159

Haciendo ping a 192.168.1.159 con 32 bytes de datos:
Respuesta desde 192.168.1.159: bytes=32 tiempo=7ms TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.1.159: bytes=32 tiempo<1m TTL=63

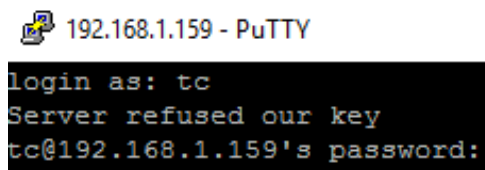
Estadísticas de ping para 192.168.1.159:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 7ms, Media = 1ms
```

Ilustración 165: Tiny Core - Comprobación sistema local

### c) Conexión VNC

Llegados a este punto, debemos comprobar que se establece correctamente la conexión remota por medio de la aplicación VNC. En este caso, la imagen una vez que es instanciada no lanza automáticamente el servidor VNC, lo que generará que tengamos que lanzarlo de manera manual. La interacción con la consola que nos proporciona el servicio Horizon es realmente lenta debido a los recursos que le hemos dotado, por lo que lanzar desde la misma el servidor VNC resulta algo imposible. Por esa razón, nos ubicamos en la aplicación PuTTY en donde nos conectamos como usuario "tc" a la instancia gracias al protocolo de comunicaciones SSH. Como ya se sabe deberemos definir la dirección IP flotante y la clave de acceso de la instancia en cuestión.

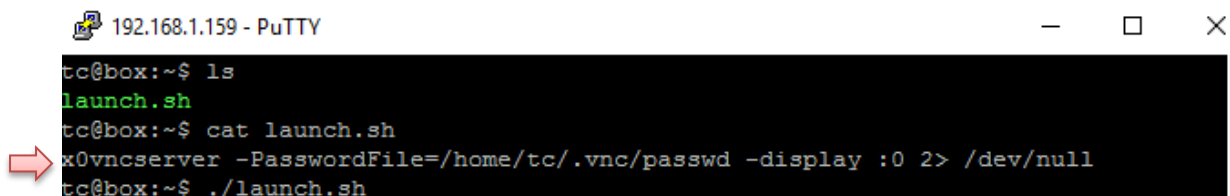
En caso de disponer el sistema de un usuario y una clave, puede rechazarnos la clave privada definida sobre la instancia y pedirnos la contraseña del usuario.



```
192.168.1.159 - PuTTY
login as: tc
Server refused our key
tc@192.168.1.159's password:
```

Ilustración 166: Conexión VNC - PuTTY Tiny Core Linux

Una vez hemos accedido y comprobado que realmente nos encontramos en la instancia, lanzamos el servidor VNC. En este caso, al no disponer de interfaz gráfica en la aplicación PuTTY, alteramos ligeramente el script y lo ejecutamos.



```
192.168.1.159 - PuTTY
tc@box:~$ ls
launch.sh
tc@box:~$ cat launch.sh
x0vncserver -PasswordFile=/home/tc/.vnc/passwd -display :0 2> /dev/null
tc@box:~$ ./launch.sh
```

Ilustración 167: Conexión VNC - Lanzar servidor VNC

Finalmente, nos quedará comprobar que efectivamente es accesible desde el cliente VNC.

1. Creamos una nueva conexión definiendo la dirección IP flotante de nuestra instancia.

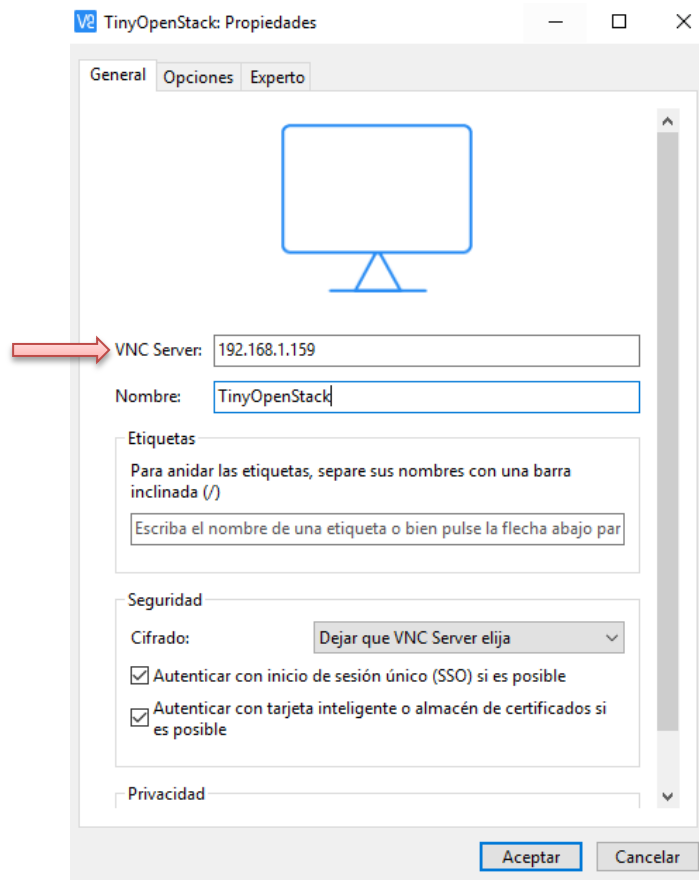


Ilustración 168: Conexión VNC - Nueva conexión VNC Client

2. Seguidamente intentamos establecer la conexión. Nos mostrará un error por motivo de que no disponemos de un certificado SSL para dicha instancia, no estableciéndose la seguridad recomendada. Una vez obviada dicha advertencia introducimos la clave para acceder.

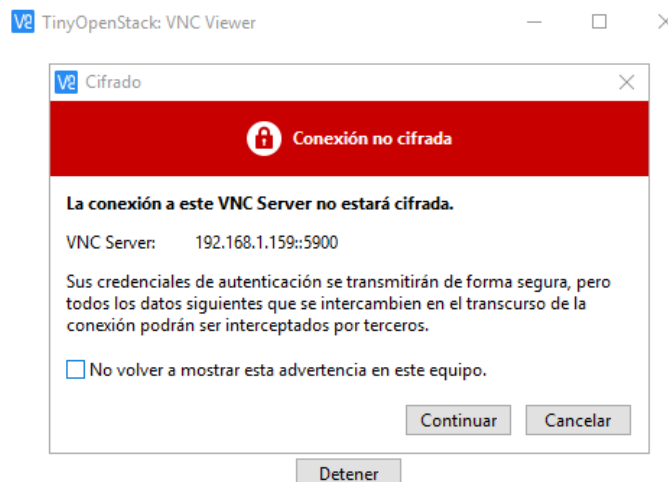


Ilustración 169: Conexión VNC - Acceso VNC Client

3. Una vez la introduzcamos seremos capaces de interactuar con esa instancia desde la aplicación VNC.

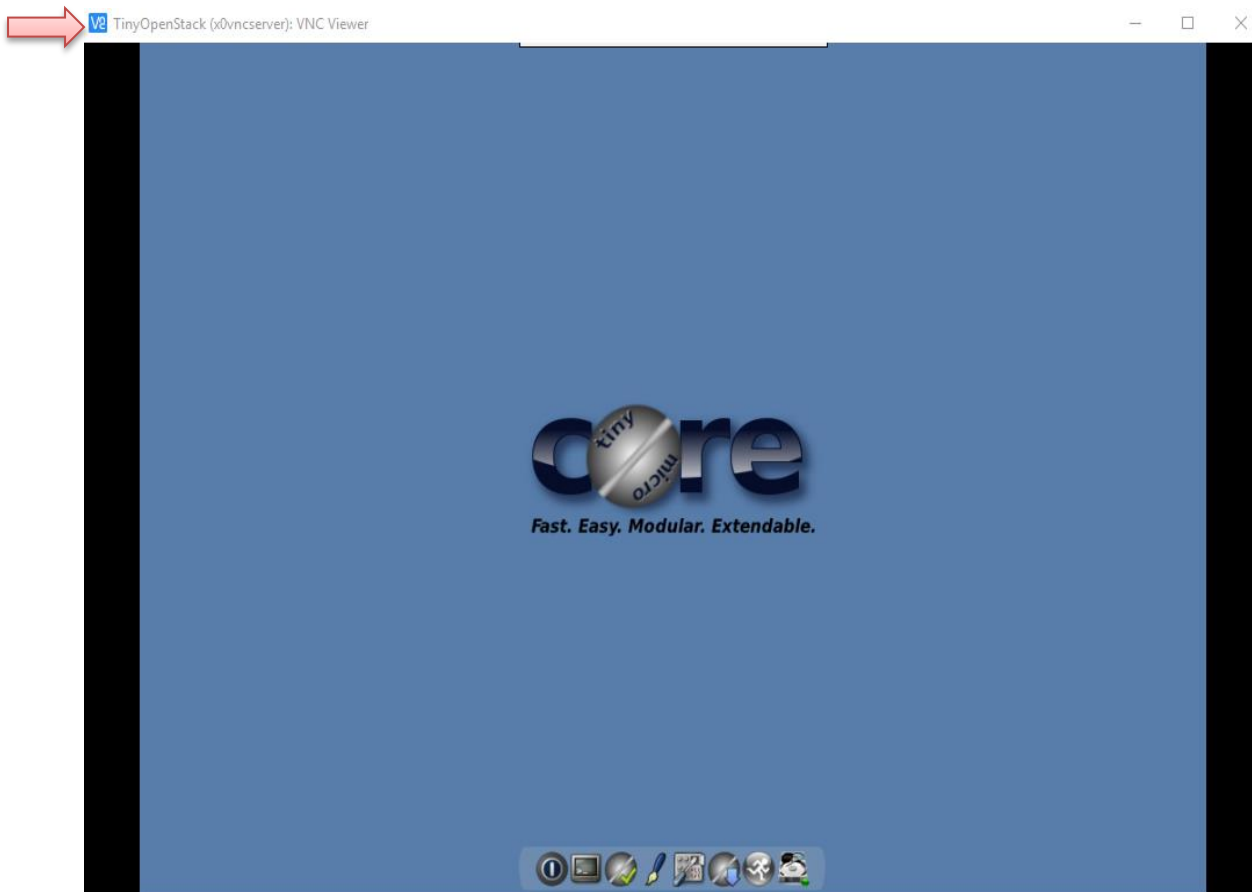


Ilustración 170: Conexión VNC - Conexión alcanzada

### G.3.2. Otras cuestiones

Hemos logrado la conexión VNC a un sistema de pruebas como es Tiny Core. El objetivo siguiente sería alcanzar el mismo resultado con un sistema con mayores facultades. Por esa razón se ha pensado en Kali. Esta distribución de GNU/Linux está basada en Debian y fue diseñada principalmente para la auditoría y seguridad informática. Kali Linux trae preinstalados una gran cantidad de programas relacionados con el tema de seguridad, destacando Nmap para el escáner de puertos, Wireshark como un “sniffer”, John The Ripper como “crackeador” de contraseñas y Metasploit, la gran suite de explotación de vulnerabilidades.

Esta distribución es sencilla de instalar, y aunque no es tan fácil de usar, hay una gran cantidad de información en Internet que facilita nuestro aprendizaje del sistema. Para poder descargarla, recomiendo visitar el siguiente enlace:

- ✓ <https://www.kali.org/downloads/>

Entonces, instalamos dicho sistema en VirtualBox y a continuación le instalamos los paquetes necesarios para poder lanzar un servidor VNC, requiriendo para esto último hacer uso de las siguientes instrucciones con permiso de administrador:

- a) Comando “sudo apt-get install tightvncserver”: Instalar el servidor VNC.
- b) Comando “sudo tightvncserver :1 -geometry 800x600 -depth 24”: Creación del servidor gráfico para que los usuarios con un cliente VNC puedan acceder de forma remota.

Sobre el último comando destacar que “:1” hace referencia al número de display, “geometry 800x600” al tamaño en píxeles, y “depth 24” a la profundidad del color.

Una vez se haga uso de esta instrucción nos solicitará una contraseña para todos aquellos clientes que quieran entrar de forma remota a nuestro servidor, pudiendo definirla o no; y posteriormente se nos pregunta si nos gustaría introducir una contraseña de solo lectura para los usuarios. Sobre esta último sucede lo mismo que la anterior, en donde podemos definirla o no.

Teniendo ya el servidor proporcionando un escritorio remoto, comprobamos si un cliente VNC puede acceder al mismo. Para ello hacemos lo siguiente:

1. Nos dirigimos a la aplicación VNC Viewer y creamos una nueva conexión.

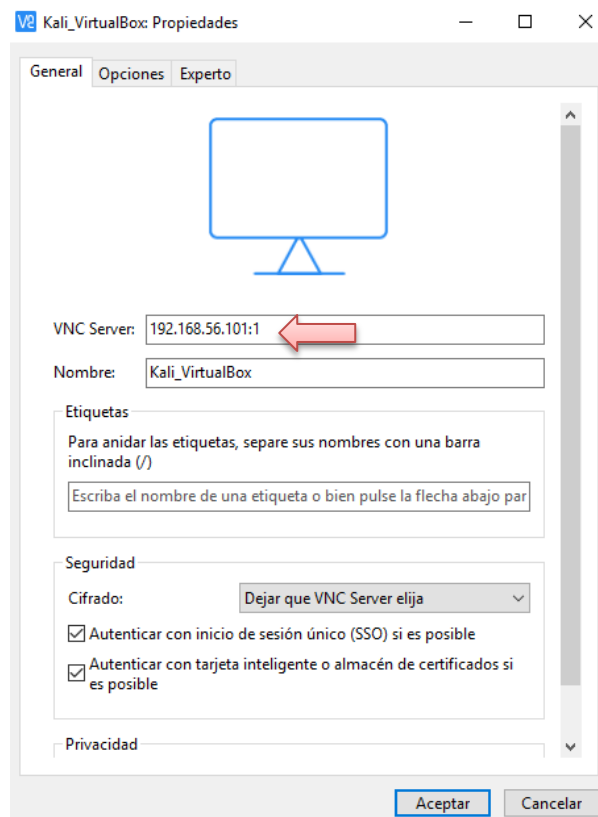


Ilustración 171: VirtualBox Kali - VNC Viewer nueva conexión

Destacar que será la dirección IP de la máquina remota a la que queremos acceder y a continuación el número de display. Recordemos que en nuestro caso es “:1”.

2. Una vez creada esta nueva conexión, aparecerá reflejada en el portal principal de la aplicación, y seleccionándolo de manera continuada dos veces se nos presentará la siguiente ventana.

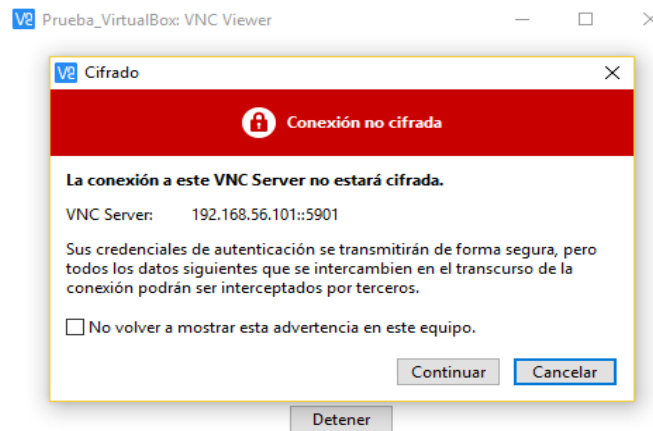


Ilustración 172: VirtualBox Kali - Cifrado VNC Server

Esto nos indica que la conexión con ese servidor no está cifrada, lo cual todos los datos que intercambiamos pueden ser interceptados por terceros.

4. Una vez hemos pulsado en la opción “Continuar”, se nos requiere la contraseña (en el caso de que lo hayamos habilitado) para tener acceso a ese escritorio remoto.

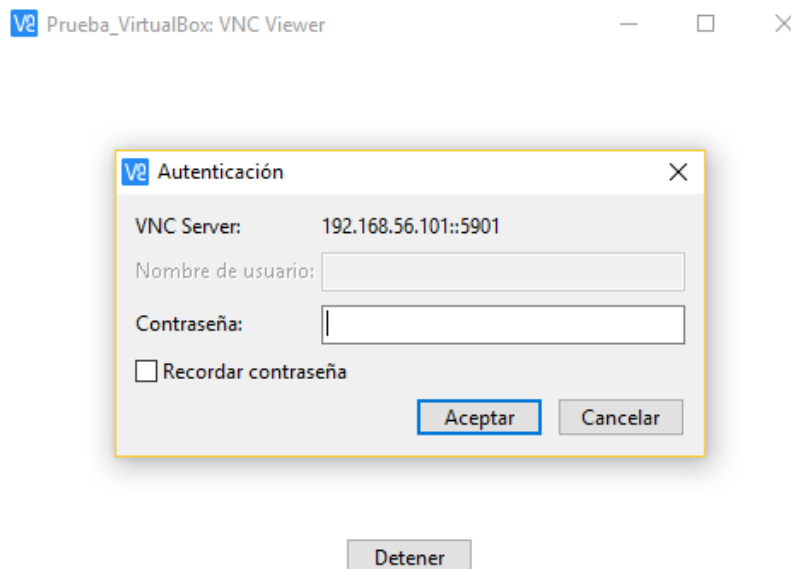


Ilustración 173: VirtualBox - Autenticación VNC Viewer



5. Finalmente, una vez que hemos iniciado sesión de manera correcta, tenemos acceso a esa máquina de forma remota, alcanzado así el resultado deseado.

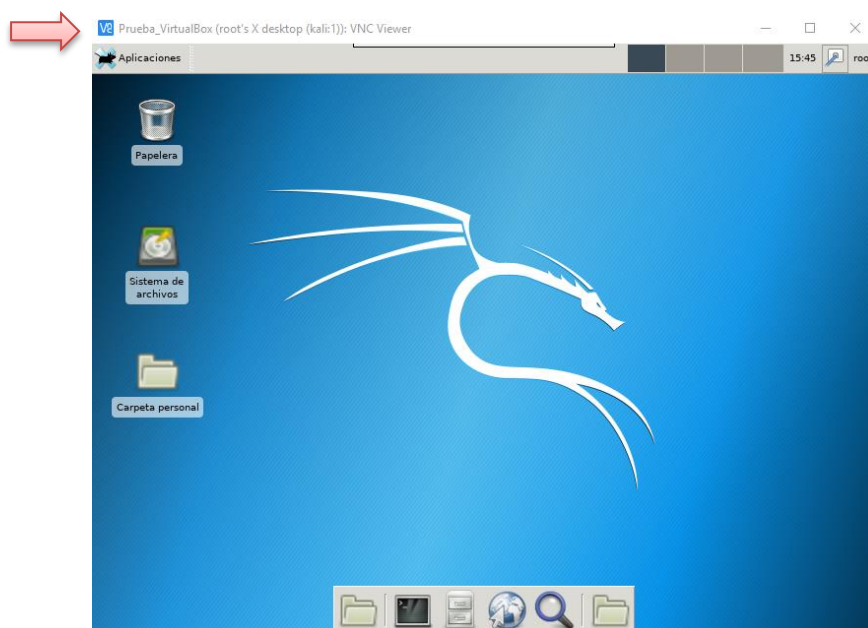


Ilustración 174: VirtualBox - Kali Linux VNC Viewer

Entonces, dado el resultado alcanzado, podemos afirmar que el sistema Kali Linux es capaz de soportar VNC proporcionando un escritorio remoto.

Lo siguiente será subir el sistema como una nueva imagen a la plataforma OpenStack. Sin embargo, esta no soporta los ficheros con formato "vdi". Debemos modificar el formato a "raw". Pero, ¿cómo?. Pues haciendo uso de la aplicación VBoxManage que proporciona la propia aplicación VirtualBox.

VBoxManage se trata de una herramienta para gestionar por la línea de comandos la aplicación VirtualBox y sus máquinas virtuales, en aspectos que no pueden ser realizados por el propio software mediante las opciones que da la vista gráfica. Dispone de un gran conjunto de opciones con las que se puede trabajar y que te pueden ayudar a conseguir información y realizar tareas más complejas. Entre estas instrucciones destacamos la siguiente que deberemos utilizar para alcanzar nuestra meta:

- ❖ VBoxManage clonehd: Copia íntegramente un disco duro a otro perteneciente o no a la misma máquina. Requerirá indicar la dirección donde se encuentra alojado nuestro fichero "vdi", la dirección donde deseamos ubicar el fichero "raw" que definamos, así como el nombre que queremos asignarle, y finalmente el formato.

El resultado de esta operación ha sido el esperado como bien refleja la siguiente imagen.

```
C:\Program Files\Oracle\VirtualBox>VBoxManage clonehd "C:\Users\Jonathan94\VirtualBox VMs\KaliLinux\KaliLinux.vdi" "C:\Users\Jonathan94\VirtualBox VMs\KaliLinux\KaliLinuxRaw.img" --format raw
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
Clone medium created in format 'raw'. UUID: 605ce623-5cde-4772-a27f-f4ce1885e83c
```

Ilustración 175: VBoxManage - Instrucción "clonehd"

Por lo tanto, hemos sido capaces de cambiar el formato al fichero creado en la aplicación VirtualBox. Sin embargo, el archivo generado dispone de un gran tamaño el cual nos hace reflexionar sobre si la plataforma OpenStack, con los recursos que le hemos otorgado, será capaz o no de soportarlo.

Nombre	Fecha de modifica...	Tipo	Tamaño
Logs	10/05/2018 8:38	Carpeta de archivos	
Snapshots	10/05/2018 9:20	Carpeta de archivos	
KaliLinux.vbox	15/05/2018 11:33	VirtualBox Machin...	4 KB
KaliLinux.vbox-prev	15/05/2018 10:39	Archivo VBOX-PREV	4 KB
KaliLinux.vdi	10/05/2018 9:20	Virtual Disk Image	3.371.008 KB
KaliLinuxRaw.img	15/05/2018 10:39	Archivos de imagen	8.388.608 KB



Ilustración 176: VBoxManage - Archivo generado

Teniendo disponible ya el archivo en el formato soportado por la plataforma OpenStack, creamos una nueva imagen con dicho archivo e instanciamos la imagen debiéndole asignar una dirección IP flotante para poder ser accesible desde el exterior. Es entonces, en el momento que creamos la imagen, cuando nos encontramos con un error. Este error está motivado porque los recursos actuales que dispone nuestro sistema, no soportan el tamaño de la imagen.

### G.3.3. Conclusiones

Uno de las metas propuestas en este proyecto era poder instanciar un imagen con un entorno gráfico en la plataforma OpenStack, y poder acceder a ella remotamente por medio de la aplicación VNC. Como hemos podido comprobar, hemos alcanzado esa meta por medio de un sistema con recursos limitados como es Tiny Core Linux. Hemos sabido adecuarnos a la situación en la que la instancia no tiene acceso a la red pudiendo instalar ese paquete para poder lanzar el servidor VNC, todo ello gracias al emulador QEMU. También, por medio de esta prueba hemos conocido el funcionamiento de VNC en la plataforma OpenStack.

Finalmente me gustaría destacar la realización de esta prueba porque hemos ampliado nuestros conocimientos en gran medida, así como nos ha permitido saber reaccionar ante las adversidades, intentando buscar diferentes alternativas que solucionen los problemas encontrados.

## G.4. Dificultades

### G.4.1. VNC en VirtualBox

Cuando habilitamos en la aplicación VirtualBox para una máquina en concreto la pantalla remota, nos puede salir la siguiente advertencia.

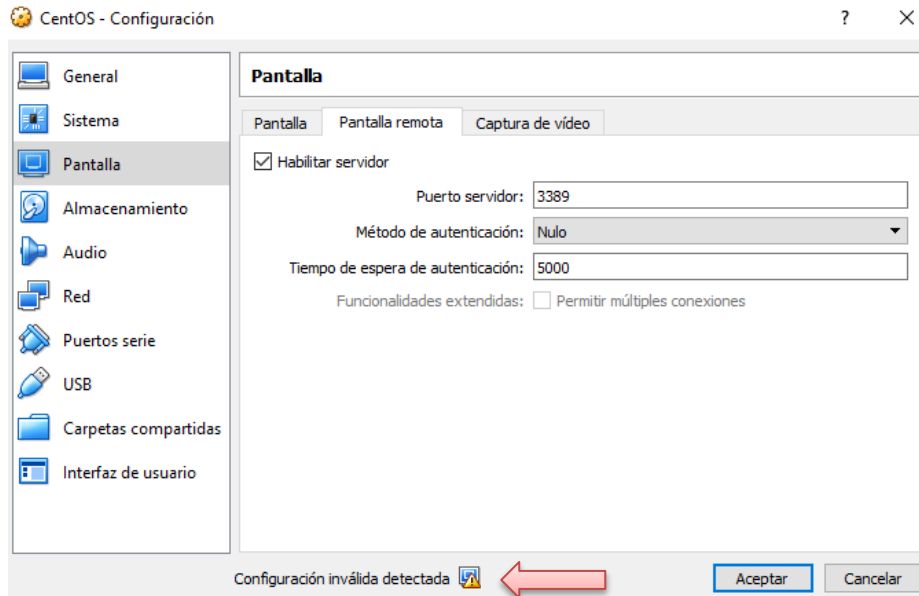


Ilustración 177: Error VirtualBox – Advertencia

Esto viene originado, como bien indica el propio mensaje, por motivo de un paquete en cuestión que requiere la aplicación VirtualBox para poder proporcionar ese servicio. Es decir, por medio de ese paquete podremos extender la funcionalidad ofrecida de base por VirtualBox, proporcionando entre otras cosas, el acceso a un escritorio remoto de una máquina virtual.

Podemos comprobar que esta extensión no está disponible desde la opción “Archivo”, más concretamente en “Preferencias” en el apartado de “Extensiones”.

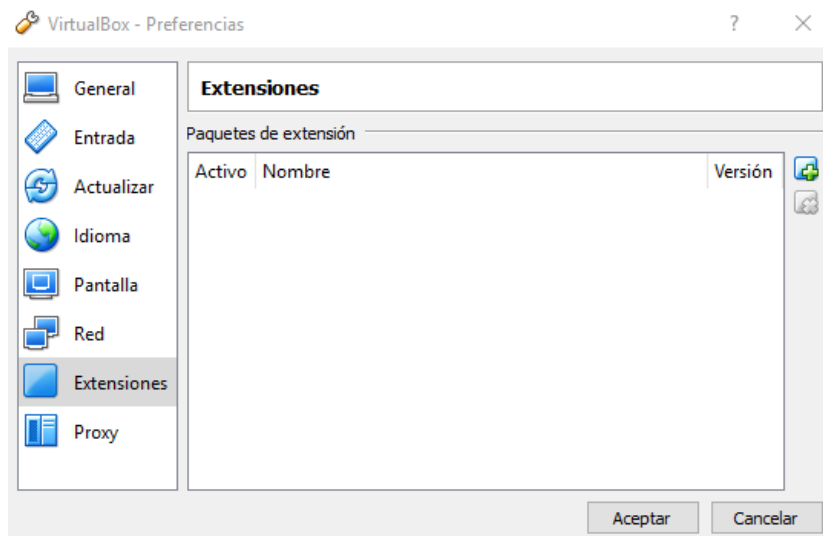


Ilustración 178: Error VirtualBox - Extensión no disponible

Como bien refleja la ilustración no tenemos disponible esa extensión, por lo que deberemos descargarlo e instalarlo en nuestro equipo. Para ello se recomienda visitar la página oficial de la aplicación, en donde si indagamos un poco encontramos la referencia a ese paquete en cuestión:

- ✓ <https://www.virtualbox.org/wiki/Downloads>

Una vez hemos finalizado la instalación, automáticamente esta queda reflejada en VirtualBox como bien podemos apreciar en la siguiente imagen.

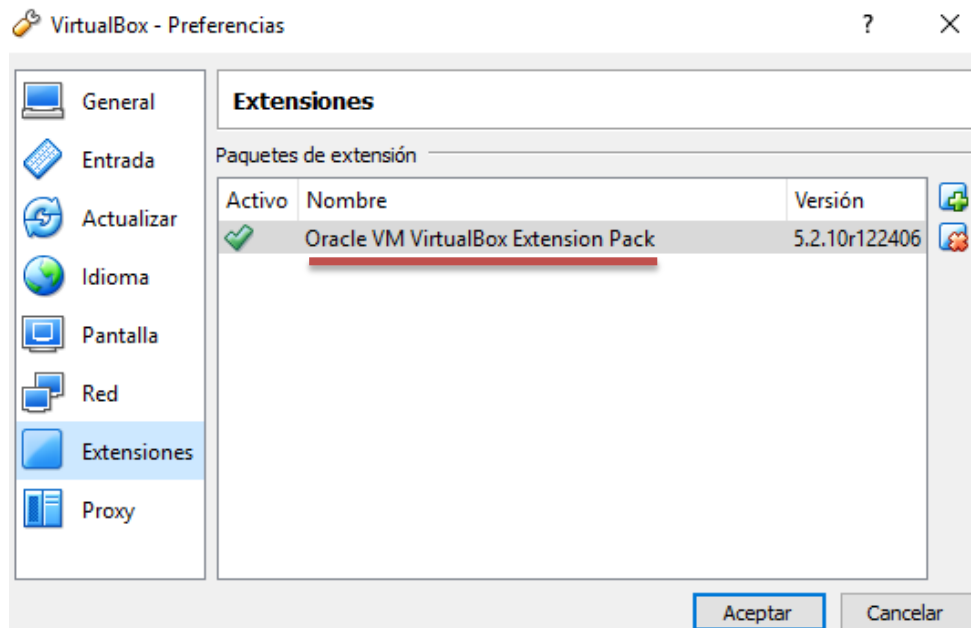


Ilustración 179: Error VirtualBox - Extensión disponible

Además, si volvemos a intentar proporcionar esa opción a la máquina virtual en cuestión, ya no nos debería de aparecer la advertencia.

## G.4.2. OpenStack

Destaquemos por medio de este subapartado, las diferentes complicaciones encontradas en la plataforma OpenStack y que nos ha servido para conocerla más de cerca.

### G.4.2.1. Imagen – Modo protegido

Cuando subimos una imagen podemos definirle si queremos que pueda ser eliminada o no por otro usuario, por medio de la opción “Protegido”.

En el caso de que decidiéramos activar esta opción, ningún usuario, ni el que ha subido la imagen ni el propio administrador, podría eliminarla. OpenStack crea una especie de superusuario que impide esa acción sobre las imágenes. Sí que se podrá borrar en cambio, cuando se le cambien las opciones de configuración a modo no protegido.

### G.4.2.2. Error 500

Este error viene acompañado de un mensaje en donde se nos detalla el problema que ha surgido. Pueden ser diferentes errores debido a que OpenStack engloba muchos de sus errores entorno a ese identificador. En nuestro caso se ha originado cuando queremos instanciar una imagen y esta no resulta posible. Más en concreto, se nos indica que el número de host es insuficiente. Pero, ¿por qué surge este problema? Este problema puede estar relacionado con dos situaciones diferentes:

- ❖ Puede que sea por los recursos que requiere una imagen en concreto para poder ser instanciada, debiendo ser estos superiores a los que se han otorgado en un principio.
- ❖ Otra posibilidad puede ser que los recursos requeridos para la imagen en cuestión para ser instanciada, no pueden ser otorgados por la plataforma porque no los dispone.

Las soluciones que se proponen ante esta adversidad es definir algún sabor con las características hardware que se estimen oportunas, o ampliar las facultades de la plataforma

## H. LDAP

### H.1. Introducción

El objetivo principal de esta documentación es explicar cómo conseguir la autenticación en OpenStack a través de un servidor LDAP. Como primera aproximación, para cumplir con esta meta será necesario disponer de un servidor LDAP y de un host que ejecute OpenStack.

Como servidor LDAP se utilizará el paquete “openldap-servers”. Este será instalado en una máquina CentOS y puede ser virtualizado mediante una máquina virtual en la aplicación VirtualBox. Todo ello podrá realizarse debido a que dicho paquete no requiere de muchos recursos: 1024 MB de RAM, un núcleo de procesador a 2.20 GHz y 8GB de almacenamiento para la instalación. De esta configuración, lo más importante es la configuración correcta de la conexión de red, la cual va a hacer uso de un adaptador puente, debido a que de esa forma seremos capaces de ver al mismo nivel de red, el resto de máquinas virtuales del equipo anfitrión, de las cuales la que más interesa es la que contenga OpenStack. Todos los aspectos de esta instalación serán explicados más adelante.

En cuanto a OpenStack, el sistema requiere de más recursos, cuantos más mejor. Dichos recursos pueden ser optimizados dependiendo del número de máquinas virtuales desplegadas en OpenStack para las pruebas, y dependiendo de los módulos de OpenStack que se estén ejecutando.

Una vez se tienen ambos servidores, el escenario que se plantea es que OpenStack confíe en el servidor LDAP como servicio de autenticación. Una vez esto se consiga, esta conexión tiene que ser asegurada para que pueda defenderse frente a ataques externos y no se robe, por ejemplo, las credenciales de usuario.

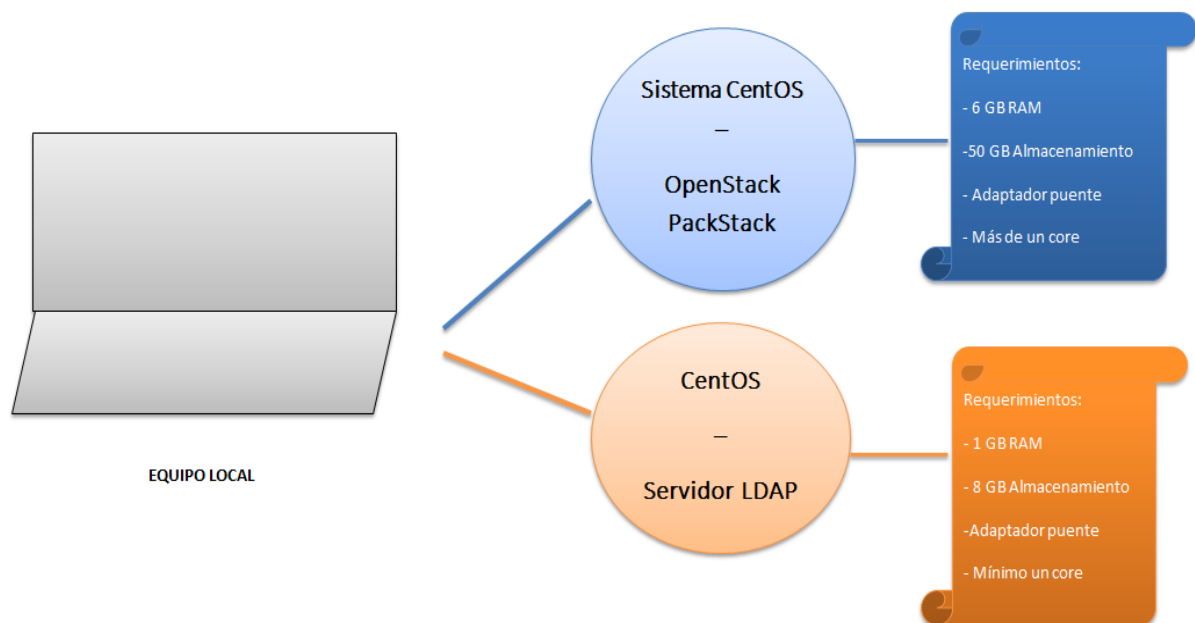


Ilustración 180: Escenario OpenStack – LDAP

## H.2. Conceptos previos

### H.2.1. Aplicación LDAP

LDAP corresponde con las siglas en inglés “Lightweight Directory Access Protocol” (en español Protocolo Ligero/Simplificado de Acceso a Directorios), que hace referencia a un protocolo, a nivel de aplicación, que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Se entiende por directorio el conjunto de objetos con atributos organizados de una manera lógica y jerárquica. El ejemplo más común es el directorio telefónico, que consiste en una serie de nombres (personas u organizaciones) que están ordenados alfabéticamente, con cada nombre teniendo una dirección y un número de teléfono adjunto.

La mayor ventaja de LDAP es que se puede consolidar información para toda una organización dentro de un repositorio central. Por ejemplo, en vez de administrar una lista de usuarios por cada grupo dentro de una organización, se puede usar LDAP como directorio central, accesible desde cualquier parte de la red. Puesto que LDAP soporta la capa de conexión segura (SSL) y la seguridad de la capa de transporte (TLS), los datos confidenciales se puede proteger de los curiosos.

Habitualmente, LDAP almacena la información de autenticación (usuario y contraseña) y es utilizado para autenticarse, aunque como se ha mencionado anteriormente, es posible almacenar otra información como los datos de contacto del usuario, permisos, certificados, etc. Para poblar el directorio LDAP, se puede emplear ficheros LDIF. En estos ficheros se describe la información de forma legible, los atributos y su correspondiente valor. También son útiles para hacer copias de seguridad e importar/exportar los elementos, facilitando enormemente la gestión de la información contenida en el directorio.

A manera de síntesis, LDAP es un protocolo de acceso unificado a un conjunto de información sobre una red.

### H.2.2. Secure Socket Layer (SSL)

Este protocolo proporciona servicios de seguridad extremo a extremo para aplicaciones en Internet independiente del protocolo de aplicación. Esta fundamentando en un PKI jerárquico (Certificados de Autenticación y certificados X.509<sup>6</sup>), y tiene como objetivo el proteger accesos web autenticando al servidor y cifrando los datos.

### H.2.3. Transport Layer Security (TLS)

Al igual que su antecesor, Secure Socket Layer (SSL), proporciona un canal de comunicación seguro. En un principio, es un protocolo que permite la autenticación y privacidad de la información extremo a extremo en una comunicación sobre Internet.

Hoy en día se considera que las implementaciones basadas en SSL son débiles, por lo que suele buscar una solución más robusta haciendo uso de este protocolo de comunicaciones.

## H.2.4. Aplicación JXplorer

JXplorer es un programa que permite gestionar el directorio de LDAP por medio de una interfaz gráfica, a través de peticiones y respuestas del servidor LDAP. Este programa servirá para facilitar la creación de contraseñas, usuarios y otras múltiples tareas que permite LDAP.

Una de las ventajas de este programa es que existe para múltiples Sistemas Operativos, como Windows o Ubuntu. Para su instalación se recomienda acceder al siguiente enlace:

- ✓ <http://jxplorer.org/downloads/users.html>

Se recomienda hacer uso de este programa debido a la sencillez que supone su instalación y manejo frente a otras opciones como “LDAP Administration Tool” o “phpLDAPadmin”.

## H.3. Instalación

### H.3.1. Sistema de autenticación LDAP

Una vez se tiene desplegado el sistema que se quiere controlar, en este caso una nube basada en OpenStack, se procede a la instalación del sistema de autenticación basado en LDAP que complementará al identificador por defecto definido por la plataforma, el servicio Keystone. Para ello se utilizarán los paquetes de OpenLDAP para montar un servidor LDAP sobre una máquina virtual en la aplicación VirtualBox con un sistema operativo<sup>27</sup> como es CentOS.

La instalación del sistema de autenticación LDAP se fundamenta en los siguientes pasos:

1. Crear una nueva máquina virtual en la aplicación VirtualBox que albergará un sistema operativo<sup>27</sup> CentOS.
2. Instalación del sistema operativo<sup>27</sup> CentOS. Se recomienda seguir los pasos descritos en el anexo D, más en concreto, el apartado 2.3. En este caso, a la máquina se le podría dotar con el nombre “LDAP”, por ejemplo, para distinguirla de la anterior.
3. A continuación, se recomienda iniciar sesión con la aplicación PuTTY al servidor LDAP por medio del protocolo SSH. De esta manera, podríamos facilitar el manejo de la máquina virtual.
4. Lo siguiente será instalar el servidor y cliente LDAP en la máquina.
5. Creamos la base de datos y se realiza un cambio de propietario para la misma.
6. Se inicia el dominio<sup>14</sup> del servidor LDAP y se habilita para que se inicie con el sistema durante el arranque.
7. Se genera un hash<sup>17</sup> de una contraseña para ser utilizado como contraseña de administrador. El hash<sup>17</sup> de esta contraseña se utilizará cuando se cree el usuario root de la instalación LDAP.



8. Se genera un fichero LDIF para modificar la información del usuario root en la base de datos. La información que se reflejará en dicho fichero será la siguiente:

```
dn: olcDatabase={0}config,cn=config
changetype: modify
add:olcRootPW
olcRootPW: HASH CONTRASEÑA – PASO 7 ({SSHA})...
```

Tabla 34: Instalación LDAP - Fichero del usuario root

9. Una vez se escribe en el atributo “olcRootPW” el hash<sup>17</sup> de la contraseña que devuelve la instrucción “slappaswd”, se procede a modificar la base de datos.

10. Se importan los esquemas más básicos.

11. Configurar el dominio<sup>14</sup>.

11.1. Obtener un nuevo hash<sup>17</sup> para una nueva contraseña.

11.2. Crear el fichero de configuración del dominio<sup>14</sup>, donde modificaremos el acceso, la base árbol LDAP, el nombre del usuario root del dominio<sup>14</sup> y se añadirá una seguridad mínima.

```
dn: olcDatabase={1}monitor,cn=config
changetype: modify
replace: olcAccess
olcAccess: {0}to * by dn.base=*gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth*
  read by dn.base=*cn=Manager,dc=mydomain,dc=com* read by * noe
dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=mydomain,dc=com
dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcRootDN
olcRootDN: cn=Manager,dc=mydomain,dc=com
olcDatabase={2}hdb,cn=config
changetype: modify
add: olcRootPW
olcRootPW: HASH CONTRASEÑA – PASO 11.1 ({SSHA})...

dn: olcDatabase={2}hdb,cn=config
changetype: modify
add: olcAccess
olcAccess: {0}to attrs=userPassword,shadowLastChange by
  dn="cn=Manager,dc=mydomain,dc=com" write by anonymous auth by self write by * none
olcAccess: {1}to dn.base="" by * read
olcAccess: {2}to * by dn="cn=Manager,dc=mydomain,dc=com" write by * read
```

Tabla 35: Instalación LDAP - Fichero de dominio

Habrà que prestar especial atención al hash<sup>17</sup> de la contraseña anteriormente creada.

12. Cambiamos la configuración del servidor LDAP.

13. Se crea la base del árbol para el dominio<sup>14</sup> LDAP, debiendo de tener dicho fichero la siguiente información implícita.

```
dn: dc=mydomain,dc=com
objectClass: top
objectClass: dcObject
objectclass: organization
o: My Domain
dc: mydomain

dn: cn=Manager,dc=mydomain,dc=com
objectClass: organizationalRole
cn: Manager
description: Directory Manager
```

Tabla 36: Instalación LDAP - Fichero del árbol de dominio

14. Abrimos los puertos del firewall para permitir el servicio LDAP. El protocolo LDAP normalmente hace uso del puerto 389/TCP y el 636/TCP para su versión segura sobre SSL/TLS.

15. Finalmente, podemos hacer uso del programa JXplorer para comprobar la correcta configuración del servidor LDAP.

Mostremos por medio de la siguiente tabla el conjunto de instrucciones que debemos hacer uso en cada paso mencionado anteriormente:

Instrucción	Operación
yum -y install openldap-servers openldap-clients	4
cp /user/share/openldap-servers/DB_CONFIG.example /var/lib/ldap/DB_CONFIG	5
chown ldap. /var/lib/ldap/DB_CONFIG	5
systemctl start slapd	6
systemctl enable slapd	6
slappasswd	7
nano chrootpw.ldif	8
ldapadd -Y EXTERNAL -H ldapi:/// -f chrootpw.ldif	9
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif	10
slappasswd	11.1
nano chdomain.ldif	11.2
ldapadd -Y EXTERNAL -H ldapi:/// -f chdomain.ldif	12
nano basedomain.ldif	13
ldapadd -x -D cn=Manager,dc=mydomain,dc=com -W -f basedomain.ldif	13
firewall-cmd --add-service=ldap --permanent	14
firewall-cmd --reload	14

Tabla 37: Instalación LDAP - Instrucciones

### H.3.2. Seguridad – TLS

Uno de los objetivos es asegurar la comunicación entre el servicio de identificación de la plataforma OpenStack (Keystone) y el servidor LDAP, para poder realizar una autenticación externa segura.

Para cumplir con esta seguridad se hará uso del protocolo TLS. Al hacer uso de una criptografía asimétrica, se deberán generar claves públicas y claves privadas. Además, deberemos compartir los diferentes certificados entre las dos máquinas.

Veamos los pasos requeridos para la creación de un certificado:

1. Lo primero será crear una clave privada.
2. Después se necesita una petición de certificado (CSR).

En la petición del certificado hay diferentes opciones destacando un apartado, “Common Name”, en donde hay que escribir el nombre del dominio<sup>14</sup> del servidor LDAP, ya que es donde va a intentar conectarse OpenStack cuando intente autenticar usuarios en el dominio<sup>14</sup> LDAP.

Instrucción	Operación
<code>openssl genrsa -out localhost.key 8192</code>	1
<code>openssl req -new -key localhost.key -out localhost.csr</code>	2

Tabla 38: Instalación TLS - Creación certificado

Una vez se tiene el certificado, un conjunto de opciones han de ser configuradas tanto en el servidor como en el cliente para habilitar TLS y hacer uso de los certificados.

#### H.3.2.1. Configuración del servidor

Describamos los pasos a seguir para poder realizar una correcta configuración del servidor LDAP.

1. Lo primero será copiar los certificados y las claves a las carpetas adecuadas, estableciéndose además la nueva propiedad de los ficheros.
2. A continuación se crea el fichero LDIF con el que se modificará el servidor LDAP y se activará TLS. Dicho fichero tendrá la siguiente información.

```
dn: cn=config
changetype: modify
add: olcTLSCACertificateFile
olcTLSCACertificateFile: /etc/openldap/certs/ca-bundle.crt
-
replace: olcTLSCertificateFile
olcTLSCertificateFile: /etc/openldap/certs/localhost.crt
-
replace: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/openldap/certs/localhost.key
```

Tabla 39: Instalación TLS - Fichero SSL

3. Lo siguiente a realizar será actualizar la configuración de LDAP.

4. Modificamos ahora las URLs por las que se puede acceder al servidor LDAP, y se añade la opción segura de “ldaps:///”.

Será en la línea 9 donde se añadirá el nuevo protocolo. Es decir, el aspecto que presentará este fichero deberá ser el que se muestra a continuación.

```
# OpenLDAP server configuration
# see 'man slapd' for additional information

# Where the server will run (-h option)
# - ldapi:/// is required for on-the-fly configuration using client tools
# (use SASL with EXTERNAL mechanism for authentication)
# - default: ldapi:/// ldap:///
# - example: ldapi:/// ldap://127.0.0.1/ ldap://10.0.0.1:1389/ ldaps:///
SLAPD_URLS="ldapi:/// ldap:/// ldaps://"

# Any custom options
#SLAPD_OPTIONS=""

# Keytab location for GSSAPI Kerberos authentication
#KRB5_KTNAME="FILE:/etc/openldap/ldap.keytab"
```

Tabla 40: Instalación TLS - Fichero slapd

5. Por último se reinicia el servidor LDAP. No hay que olvidarse de abrir el puerto correspondiente en el firewall.

Todas estas operaciones que se consideran necesarias para la configuración del servidor, quedan descritas por un conjunto de instrucciones que se reflejan en la siguiente tabla:

Instrucción	Operación
cp localhost.key localhost.crt /etc/pki/certs/ca-bundle.crt /etc/openldap/certs	1
chown ldap. /etc/openldap/certs/localhost.key /etc/openldap/certs/localhost.cert /etc/openldap/certs/ca-bundle.crt	1
nano mod_ssl.ldif	2
ldapmodify -Y EXTERNAL -H ldapi:/// -f mod_ssl.ldif	3
nano /etc/sysconfig/slapd	4
systemctl restart slapd firewall-cmd --add -service=ldaps --permanent firewall-cmd --reload	5

Tabla 41: Instalación TLS - Configuración servidor

Para comprobar el correcto funcionamiento de TLS en LDAP, se puede hacer una configuración de un cliente LDAP cualesquiera sin confundir con el verdadero cliente de este trabajo que será el servicio Keystone de OpenStack, siendo este configurado en el apartado siguiente.

### H.3.3. Configuración de OpenStack – LDAP

Para hacer uso de LDAP como sistema autenticador en la plataforma OpenStack, se deberá habilitar el multi-dominio. Este multi-dominio permite mantener separados en dominios<sup>14</sup>, diferentes proyectos, grupos, usuarios y roles.

Suponiendo que ya se disponga de un servidor LDAP configurado adecuadamente, con usuarios y grupos preparados para usarlos en OpenStack, podemos proceder a la configuración.

1. Lo primero que se hace es activar el control de acceso informático<sup>10</sup> correspondiente a LDAP.
2. Se crea el directorio para los diferentes dominios<sup>14</sup>. En este directorio será por tanto donde se encuentren los nuevos dominios<sup>14</sup> que se creen.
3. Lo que se pretende a continuación es modificar el archivo de configuración del servicio Keystone. Para ello hacemos uso del paquete “crudini”.
4. Se habilitan los dominios múltiples en el servicio Horizon. Para ello se añaden un conjunto de líneas al archivo “/etc/openstack-dashboard/local\_settings”, quedando descritas de la siguiente manera:

```
OPENSTACK_API_VERSIONS = {
'identity': 3,
}

# Set this to True if running on multi-domain model. When this is enabled, it
# will require user to enter the Domain name in addition to username for login.
OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = True

# Overrides the default domain used when running on single-domain model
# with Keystone V3. All entities will be created in the default domain.
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = 'Default'
```

Tabla 42: Configuración OpenStack - Fichero "local\_settings"

Hay que asegurarse de que use la versión 3.

5. Se reinicia el servicio para que se apliquen los cambios realizados.

6. Una vez se ha configurado el multi-dominio se procede a crear el dominio<sup>14</sup> Keystone para la integración de LDAP.

7. Se crea un fichero de configuración para el nuevo dominio<sup>14</sup>. Este fichero estará localizado en el directorio que se ha creado para los dominios<sup>14</sup> y dispondrá de la siguiente información.

```
[identity]
driver = ldap

[ldap]
url = ldaps://ldap
suffix = dc=mydomain,dc=com
query_scope = sub
user = cn=Manager,dc=mydomain,dc=com
password = telematica
use_dumb_member = False

user_tree_dn = ou=Users,dc=mydomain,dc=com

user_objectclass = inetOrgPerson

user_id_attribute = cn
#user_id_attribute = uidNumber
user_name_attribute = sn
user_mail_attribute = mail
user_pass_attribute = userPassword
user_enabled_attribute = enabled

group_tree_dn = ou=UserGroups,dc=mydomain,dc=com
group_objectclass = groupOfNames
group_id_attribute = cn
group_name_attribute = ou
group_member_attribute = member
group_desc_attribute = description

user_allow_create = false
user_allow_update = false
user_allow_delete = false
group_allow_create = false
group_allow_update = false
group_allow_delete = false

use_tls = false
tls_cacertfile = /etc/keystone/domains/ca-ldap.crt
```

Tabla 43: Configuración OpenStack - Fichero "keystone\_mydomai.conf"

En este fichero existen bastantes directivas que se deberían de explicar:

- ❖ La dirección [identity] básicamente especifica los drivers que se van a utilizar.
- ❖ En la sección [ldap], se especifica la URL donde dirigirse, debiendo de hacer uso del protocolo seguro de LDAP.
- ❖ Después, se especifica la base del dominio<sup>14</sup> LDAP. También se declara el usuario que se conectará con el servidor LDAP, para administrar las cuentas de usuario y grupos.
- ❖ Por último, se especifica la ruta al archivo que contiene el certificado del servidor LDAP, para así saber en quien confiar durante el intercambio de claves.

8. Se importa el certificado del servidor LDAP almacenándose en la carpeta de dominios<sup>14</sup>.
  9. Al igual que con el cliente LDAP, para poder acceder a la URL “ldaps://ldap”, es necesario declarar el nombre de la máquina en el archivo “hosts” del sistema: “DIRECCIONIP\_MAQUINASERVIDORLDAP ldap”.
  10. Se cambia el propietario del archivo de configuración del nuevo dominio<sup>14</sup>, para otorgárselo al servicio Keystone. Se reinicia el servidor otra vez.
  11. Se le concede al usuario administrador “admin” acceso al dominio<sup>14</sup>.
  12. Se comprueba que todo funciona correctamente intentando obtener una lista con los usuarios en el dominio<sup>14</sup> LDAP.
  13. Lo siguiente a llevar a cabo será configurar el servicio Nova para que utilice el servicio Keystone en la versión 3. Una vez realizado ello, se reinician los servicios del servicio Nova para poder aplicar así los cambios realizados y posteriormente, se reinicia el propio servicio.
  14. También se deberá actualizar la versión del servicio Keystone en el servicio de almacenamiento de bloques (servicio Cinder). Para ello se deberá editar el fichero de configuración de dicho componente, modificando la URL del autenticador y estableciendo su versión a la 3.
- A continuación, se deberán reiniciar todos los servicios del componente Cinder.
15. Finalizamos reiniciando la plataforma.

Todas estas operaciones específicas para la correcta configuración de OpenStack en el uso del servidor LDAP, quedarán descritas por el conjunto de instrucciones que se definen en la tabla siguiente siendo recomendable seguir el orden que figura por cada operación.

Instrucción	Operación
setsebool -P authlogin_nsswitch_use_ldap=on	1
mkdir /etc/keystone/domains/	2
chown keystone /etc/keystone/domains/	2
yum -y install crudini	3
crudini --set /etc/keystone/keystone.conf identity domain_specific_drivers_enabled true	3
crudini --set /etc/keystone/keystone.conf identity domain_config_dir /etc/keystone/domains	3
crudini --set /etc/keystone/keystone.conf identity assignment_driver sql	3
nano /etc/opensack-dashboard/local_settings	4
systemctl restart httpd.service	5
openstack domain create mydomain --description "Dominio LDAP"	6
nano /etc/keystone/domains/keystone.NUEVODOMINIO.conf	7
nano /etc/keystone/domains/NOMBRECERTIFICADO.crt	8
nano /etc/hosts	9
chown keystone /etc/keystone/domains/keystone.MIDDOMINIO.conf	10
systemctl restarts httpd.service	10
openstack role add --domain MIDDOMINIO --user admin admin	11
openstack user list --domain mydomain	12
crudini --set /etc/nova/nova.conf keystone_auth_token_auth_version v3	13
systemctl restart openstack-nova-api.service openstack-nova-cert.service openstack-nova-conductor.service openstack-nova-consoleauth.service openstack-nova-novncproxy.service openstack-nova-scheduler.service	13
systemctl restart openstack-nova-compute.service	13
systemctl restart openstack-cinder-api.service systemctl restart openstack-cinder-backup.service systemctl restart openstack-cinder-scheduler.service systemctl restart openstack-cinder-volume.service	14
systemctl restart httpd.service	15

Tabla 44: Configuración OpenStack - Instrucciones



## H.4. Conclusiones

Por medio de este anexo se pretende orientar y ayudar a conocer como poder disponer del sistema de autenticación LDAP en la plataforma OpenStack. Se realiza una descripción un poco ambigua, la cual conviene ser utilizada para comprobar que efectivamente se alcanza el resultado deseado y que OpenStack realmente puede trabajar con el sistema de autenticación de la Facultad de Informática UPV/EHU de San Sebastián/Donostia. De todas maneras, se debería realizar un estudio más minucioso.

