Implantación de una plataforma de Cloud Computing

Alumno: Marco Alacot Torres

Titulación: Ingeniería Técnica en Informática de Sistemas

Director del proyecto: Álvaro Álvarez

Noviembre 2011

Índice

Agradecimientos

En primer lugar a mis padres por todo su apoyo durante todos estos años y fundamentalmente porque sin ellos no hubiera sido posible, también a Álvaro Álvarez por todo lo que me ha enseñado y por su paciencia y dedicación durante la realización de este proyecto, infinitas gracias a ambos.

Presentación

El objetivo principal de este proyecto es el estudio del impacto de la virtualización y la computación en la nube, empleando para ello la realización de un proyecto real consistente en la migración de los sistemas que se emplean para la realización de las prácticas de la asignatura ADS/ASO.

La asignatura ADS/ASO realiza un uso intensivo de diferentes sistemas operativos para la realización de las prácticas, actualmente dichos laboratorios poseen una estructura basada en terminales, cada grupo de alumnos está formado por dos terminales, en cada terminal se crea una estructura de máquinas virtuales que forman una red con las máquinas virtuales del segundo terminal de manera que puedan interaccionar entre ellas.

La solución actual requiere el uso de un laboratorio determinado para el desempeño de las prácticas, ya que se ha de realizar un proceso de configuración en los diferentes terminales para realizar las prácticas. El proceso de configuración que ha de realizarse en cada terminal comprende entre otras tareas:

- Instalar hipervisores
- Instalar sistemas operativos
- Crear las subredes de terminales
- Ejecutar scripts de configuración

Esta solución a pesar de ser funcional dista mucho de ser la solución ideal. Lo que se pretende a la finalización de este proyecto es abandonar la infraestructura actual y centralizar la ejecución de las máquinas virtuales en un servidor dedicado, únicamente utilizando los clientes para el acceso a las máquinas virtuales a través de un agente creado para tal efecto, dicho de otra forma, este proyecto consiste en la creación de una plataforma de Cloud Computing de tipo privado que soportará la nueva infraestructura de la asignatura, con esto se pretende obtener una serie de ventajas:

- La asignatura no estará ligada a un laboratorio físico, se puede utilizar cualquiera sin ninguna configuración adicional.
- Es más fácil y económico ampliar el sistema virtualizado que renovar o ampliar todos los PC's del laboratorio.
- Gracias a la tecnología de computación elástica el sistema se adaptará mejor a picos de demanda de procesamiento.
- Los alumnos pueden disponer de la infraestructura del laboratorio en cualquier momento y desde cualquier lugar, pudiendo continuar con las prácticas o estudiar para la asignatura sin estar ligados a la estructura física del laboratorio.
- La infraestructura una vez instalada, es mucho más fácil de administrar, escalar, y el tiempo de respuesta ante errores es menor.

- Sirve igualmente para otros propósitos como dar clase de teoría.

Todas estas ventajas, como podemos suponer fácilmente pueden ser extensibles a cualquier asignatura, especialmente las que utilicen instalaciones personalizadas del sistema operativo, o diversos sistemas operativos diferentes.

1. Introducción

Tradicionalmente, los equipos informáticos y en particular los servidores poseen un elemento físico de hardware, que tiene al sistema operativo como interfaz entre las aplicaciones que se ejecutan en dicha máquina y el hardware.

Habitualmente estos servidores sólo ejecutan un sistema operativo y un conjunto de aplicaciones que son compatibles con este sistema. Uno de los problemas asociados a esta forma de infraestructura es la escasa utilización de los recursos de la máquina, lo que se traduce en un el coste en hardware muy elevado.

En este proyecto analizaremos dos soluciones que buscan optimizar tanto económica, como técnicamente esta metodología tradicional. La primera de ellas es la llamada "**virtualización**", con la virtualización podemos ejecutar múltiples máquinas virtuales dentro de una máquina física, con la consiguiente mejora en el aprovechamiento de la máquina, una vez realizado este análisis procederemos a proyectar una solución para un problema concreto y real, describiendo los agentes involucrados en el mismo así como las decisiones tomadas para la consecución del objetivo.

Actualmente la virtualización se encuentra en auge, esto es debido a que las empresas han adoptado masivamente esta tecnología, debido a una serie de ventajas que mejoran la competitividad de la misma:

- **Consolidación de servidores**: Detener la proliferación de servidores consolidándolos en máquinas virtuales contenidas en un número inferior de servidores muy potentes.

- **Obtener los recursos necesarios con los medios existentes**: Poner rápidamente en marcha aplicaciones y balancear las cargas de trabajo entre los recursos existentes reduciendo en lo posible los sobredimensionamientos.

- **Centros de tolerancia a desastres asequibles**: Utilizar una infraestructura virtual para replicar su centro de datos principal. Replicar en máquinas virtuales los servidores críticos.

- **Alargar la vida de los entornos antiguos**: Ejecutar antiguas aplicaciones que aún necesitan su SO original sobre modernos servidores de alto rendimiento.

- **Entornos de prueba y desarrollo flexibles**: Aprovechamiento de una infraestructura virtual independiente del hardware para probar gran número de entornos sobre un pequeño número de sistemas físicos

- **Virtualización de los puestos de trabajo**: Utilizar todas las ventajas de la virtualización de servidores aplicadas al mundo de los PCs.

Estas ventajas pueden ser cuantificables, según un estudio realizado en 2009 por la empresa norteamericana HP:

- El 60% de los encuestados indica que ha mejorado el tiempo de provisión en más de un 30%

- El 45% presenta una reducción del TCO de servidores superior al 30%,

- 31% asegura haber incrementado más de un 30% la productividad de IT gracias a la virtualización
- 26% presenta un ahorro de energía superior al 30% en el DataCenter
- 85% mejora los tiempos de recuperación ante caídas no planificadas.
- Ratios de uso de recursos de servidor pasan de 5-15% a 60-80% con virtualización

El siguiente paradigma es el llamado “**computación en la nube**”, en este tipo de computación todo lo que puede ofrecer un sistema informático se ofrece como servicio, de modo que los usuarios puedan acceder a los servicios disponibles "en la nube de Internet" sin conocimientos (o, al menos sin ser expertos) en la gestión de los recursos que usan. Según el IEEE Computer society es un paradigma en el que la información se almacena de manera permanente en servidores de Internet y se envía a cachés temporales de cliente, lo que incluye equipos de escritorio, centros de ocio, portátiles, etc.

"**Cloud computing**"es un nuevo modelo de prestación de servicios de negocio y tecnología, que permite al usuario acceder a un catálogo de servicios estandarizados y responder a las necesidades de su negocio, de forma flexible y adaptativa, en caso de demandas no previsibles o de picos de trabajo, pagando únicamente por el consumo efectuado.

El cambio paradigmático que ofrece la computación en nube es que permite aumentar el número de servicios basados en la red. Esto genera beneficios tanto para los proveedores, que pueden ofrecer, de forma más rápida y eficiente, un mayor número de servicios, como

para los usuarios que tienen la posibilidad de acceder a ellos, disfrutando de la 'transparencia' e inmediatez del sistema y de un modelo de pago por consumo.

La computación en la nube consigue aportar estas ventajas, apoyándose sobre una infraestructura tecnológica dinámica que se caracteriza, entre otros factores, por un alto grado de automatización, una rápida movilización de los recursos, una elevada capacidad de adaptación para atender a una demanda variable, así como virtualización avanzada y un precio flexible en función del consumo realizado evitando además el uso fraudulento del software y la piratería.

La computación en nube es un concepto que incorpora el **software como servicio** (SaaS), como en la Web 2.0 y otros conceptos recientes, también conocidos como tendencias tecnológicas, que tienen en común el que confían en Internet para satisfacer las necesidades de cómputo de los usuarios.

Cuando la nube se ofrece al público y se vende en un sistema de prepago lo llamamos **Nube Pública**, ejemplos de este tipo de nube son por ejemplo los Amazon Web Services o la plataforma Azure de Microsoft.

Cuando nos referimos al término **Nube Privada**, estamos haciendo referencia a un tipo de Nube que no está abierta al público, y es utilizada únicamente por la organización en la que se encuentra, no obstante también pueden darse configuraciones híbridas en las que parte de la infraestructura es pública y parte privada.

Figura 1.1: Modelos de Cloud Computing

Este proyecto tiene cómo objetivo último, la implantación de una nube privada enfocada a mejorar los sistemas que se emplean en la asignatura ADS/ASO de las titulaciones de Ingeniería Informática, Ingeniería Técnica en Informática de Sistemas, y nuevos grados de Informática en la Escuela Técnica Superior de Informática de la Universidad Politécnica de Valencia a fin de mejorar la infraestructura actual

En los siguientes apartados se detallará todo el proceso de construcción, que va desde la idea de una nueva infraestructura hasta la creación de la misma, pasando por todo el proceso de análisis y construcción.

1.1. Análisis económico

Hemos introducido a grandes rasgos el impacto económico que tienen las tecnologías de virtualización y cloud computing en el mundo actual, en el presente apartado realizaremos un análisis más minucioso del impacto de estas tecnologías.

1.1.1 Análisis económico: Cloud Computing

La evolución de la nube atraerá nuevos modelos de negocio hasta ahora desconocidos. En su publicación Business Strategy for Cloud Providers, IBM destaca cuatro modelos que surgen como una escisión de los proveedores de Cloud Computing tradicionales:

- **Los proveedores que suministran hardware, software o servicios de cloud profesionales a otros proveedores de cloud computing**. Éstos invierten o compran nuevas tecnologías y llevan a cabo la investigación y las fusiones necesarias para desarrollar nuevas capacidades. Este grupo estará compuesto por grandes empresas y el producto ofrecido será equivalente a una commodity.
- **Los proveedores de outsourcing de TI basado en cloud**. Se trata de alianzas entre empresas de outsourcing y de proveedores de SaaS que ofrecen la infraestructura, los servicios de aplicaciones y la asistencia en la migración a la nube.

- **Los «agregadores de SaaS»**, especialmente atractivos para los nuevos y pequeños proveedores, que reúnen todas las soluciones de SaaS específicas o complementarias para un mismo sector. Su segmento de mercado objetivo son empresas que buscan una solución integral en la nube.

La computación en la nube se perfila como un mercado compuesto por un conjunto relativamente pequeño de grandes proveedores de servicios estandarizados con una amplia base de clientes, junto con una diversidad de pequeñas empresas que ofrecerán servicios diferenciales. En la composición de estos grupos influirán los grandes proveedores ya establecidos, pero también las estrategias que sigan las grandes corporaciones en su adopción de los servicios de la nube.

Estas empresas pueden servir de trampolín para los proveedores de cloud incipientes, puesto que la demanda de una sola de estas empresas puede ser suficiente para cubrir los costes fijos de uno o varios centros de datos. De esta forma, un proveedor pequeño puede ofrecer un servicio especializado y hacerse con la demanda de una gran empresa y, una vez optimizado el servicio, ofrecérselo a otros clientes potenciales.

En la última década, la tecnología informática se ha establecido como pieza angular de la economía y de los mercados, por encima de cualquier otro sector de los negocios. Las grandes empresas tecnológicas se han convertido en los nuevos gigantes de la bolsa, rivalizando con las compañías omnipresentes en los índices, como los bancos y las farmacéuticas. De hecho, Microsoft, Google, Apple, IBM y Oracle se presentan como las cinco principales, capaces de rozar cifras similares a las de las petroleras (Exxon, Chevron, PetroChina, RD Shell y Total)

Estos gigantes, hasta ahora centrados en actividades distintas, están convergiendo y tomando posiciones en el mercado del cloud computing, que se presenta con un enfoque multiproducto y multiproveedor. Las empresas pioneras han estado construyendo las bases del cloud computing, llevando servicios innovadores a empresas y consumidores. Sin embargo, pronto no bastará con ofrecer servicios básicos de cloud y la diferenciación se convertirá en un imperativo. Sólo las más rápidas en llevar los avances al mercado serán las que logren más cuota y mayores márgenes.

1.1.2 Análisis económico: Virtualización

Cómo se ha introducido en la presentación, la virtualización es otra tecnología capital en el desarrollo económico actual ya no sólo del sector informático sino de otros muchos sectores en los que los sistemas TI tienen una importancia capital en el desarrollo de los procesos de negocio.

Según **Vmware**, la implantación de soluciones de virtualización en las organizaciones tiene el siguiente impacto:

“Se reducen los costes operativos y de hardware en aproximadamente un 50%, y los costes energéticos en un 80%, lo que supone un ahorro de más de 2000€ anuales por cada carga de servidor virtualizada, además se reduce el tiempo necesario para el aprovisionamiento de nuevos servidores en hasta un 70%.”

Caso práctico

Para ejemplificar el impacto económico de las tecnologías de virtualización en una organización, podemos suponer el caso de una pequeña empresa de unos 20 empleados, la empresa necesita cinco servidores pequeños (base de datos, almacenamiento, web, correo y pruebas), y se estima que el mantenimiento de estos cinco servidores requiere una dedicación de una persona a tiempo completo (incidencias con los usuarios, reinicios, cambios de piezas, averías, instalación de parches y nuevas versiones de software, etc).

Desde 2009 hasta 2011, la empresa va creciendo, de forma que en 2010 el equipo de ventas necesita duplicar la capacidad de almacenamiento de su servidor y el servidor web corporativo está bastante cargado.

El servidor de correo corporativo empieza a estar bastante ajustado de espacio de almacenamiento. En 2010, debido al fuerte incremento de ventas del año anterior, el servidor de base de datos está también bastante cargado, y sería necesario dotarlo de mayor memoria, CPU y almacenamiento.

En un escenario tradicional, la solución de esta situación pasaría por la compra de cinco servidores, con sus respectivos contratos de mantenimiento y costes de administración. En los siguientes años, sería necesario ampliar los equipos comprados según necesida-des, y en algunos casos, comprar nuevos equipos, lo que conllevaría posiblemente más costes en personal de mantenimiento, que se ha contemplado en la simulación. Dado que hay varios equipos que actualizar, hay más paradas de sistemas, y por tanto pérdi- das de productividad asociadas.

En un escenario de virtualización, el primer año se adquiere un único sistema relativamente potente y escalable, con ciertas capacidades no contempladas en el entorno anterior (serían demasiado caras), como por ejemplo sistemas de discos ampliables en caliente.

En los siguientes años, únicamente hay que ampliar el almacenamiento (aunque es posible aprovechar mucho mejor el existente), y más adelante la potencia total de la máquina añadiendo CPUs y memoria. Por las características de los entornos virtualizados, las ampliaciones instaladas son directamente aprovechadas por todas las máquinas virtuales.

Algunos condicionantes para los desgloses que siguen:

- No se han considerado gastos que son idénticos en ambos, como por ejemplo licencias de sistemas operativos, costes de instalación iniciales, etc.
- Los contratos de mantenimiento de hardware se han especificado como Gold, respuesta insitu en 4 horas, salvo para el servidor de pruebas. Los gastos de dichos contratos se han repartido en 3 años.
- Los precios están en euros y están basados en componentes reales

Los modelos y capacidades detalladas de cada máquina se indican en la siguiente tabla:

Coste de los elementos considerados

Máquina	Modelo	Características
Servidor base de datos	SC1420	1 CPU (máx. 2), 1 GB RAM (máx. 4), 2x80 GB HD mirror
Servidor de almacenamiento	PE830	1 CPU (máx. 1), 512 MB RAM (máx. 4 GB), 2x250 GB HD mirror
Servidor web	SC430	1 CPU (máx. 1), 512 MB RAM (máx. 2 GB), 2x80 GB HD mirror
Servidor correo	SC430	1 CPU (máx. 1), 512 MB RAM (máx. 2 GB), 2x80 GB HD mirror
Servidor pruebas	SC430	1 CPU (máx. 1), 512 MB RAM (máx. 2 GB), 2x80 GB HD mirror
Servidor infraestructura	PE2900	2 CPU (máx. 4), 2 GB RAM (máx. 8 GB), 3x146 GB HD RAID5

Figura 1.1.2.1: Coste de los elementos considerados

Considerando los precios detallados en la anterior tabla, si hacemos un cálculo del coste de las dos infraestructuras tenemos que:

Comparativa de coste

Inversiones	Infraestructura tradicional				Infraestructura virtualizada				
Concepto	**2009**	**2010**	**2011**	**Acumulado**	**2009**	**2010**	**2011**	**Acumulado**	**Ahorro**
Compra servidores	3.905,00	529,00	900,00	5.334,00	2.449,00	-	-	2.499,00	
Actualización discos	-	600,00	200,00	800,00	-	640,00	-	640,00	
Actualización memoria	-	-	400,00	400,00	-	-	700,00	700,00	
Actualización CPU	-	-	440,00	440,00	-	-	360,00	360,00	
Total	3.905,00	1.129,00	1.940,00	6.974,00	2.449,00	640,00	1.060,00	4.194,00	**41,00%**
Gastos									
Concepto	**2009**	**2010**	**2011**	**Acumulado**	**2009**	**2010**	**2011**	**Acumulado**	**Ahorro**
Personal mantenimiento	30.000,00	62.400,00	64.896,00	157.296,00	30.000,00	31.200,00	32.448,00	93.648,00	
Mantenimiento hardware	1.066,67	1.331,67	1.331,67	3.730,01	363,33	363,33	363,33	1.090,00	
Pérdida productividad por actualización hardware	-	3.160,00	1.000,00	4.160,00	-	135,00	500,00	635,00	
Total	31.066,67	66.891,67	67.227,67	165.186,01	30.363,33	31.698,33	33.311,33	95.373,00	**42,00%**

Figura 1.1.2.2: Comparativa de coste entorno virtual vs entorno tradicional

Conclusiones

Del cuadro de financiación mostrado en la Figura 1.4.2.2 se pueden sacar las siguientes conclusiones:

- Es mucho más barato y rentable comprar una máquina potente con escalabili- dad, posibilidad de ampliación y con capacidades avanzadas (conexión de discos en caliente, controladora RAID) que varias máquinas pequeñas con funcionalidades equivalentes.
- La solución al problema de múltiples servicios en la misma máquina en que cuando uno de ellos falla se lleva a todos los demás detrás, se soluciona en el escenario tradicional a base de poner cada servicio en una máquina independiente, y en el segundo escenario mediante la virtualización. Ambas soluciones son equivalentes respecto

al problema original, suponiendo que la virtualización es estable. Hoy día se da esta condición.

- En el escenario virtualizado, al estar las máquinas funcionando sobre hardware virtual, que es independiente del hardware real, las ampliaciones pueden hacerse en caliente en algunos casos (discos duros), y con una simple reconfiguración en otros (CPU y memoria), en todo caso con una interrupción de servicio muy corta. En el escenario tradicional, las ampliaciones de disco en muchas ocasiones provocan tener que restaurar datos desde una copia de seguridad, y las de CPU y memoria una parada corta del sistema (pero más larga que en una máquina virtual).
- Al facilitar las ampliaciones del hardware virtual, esto permite no comprar máquinas adicionales, sino ampliar las que se tienen. Esto se traduce directamente en una contención de los costes administrativos de las máquinas: menos servidores, menos personal.
- Cuando es necesario ampliar la máquina de infraestructura, como se ha comprado con funcionalidades avanzadas (que no se justifican para las máquinas pequeñas), los tiempos de parada son también más cortos, cuando no inexistentes.

La conclusión general es que con técnicas de virtualización de entornos informáticos pueden conseguirse ahorros importantes, tanto en inversión como en costes, que en nuestro escenario imaginario hemos llegado a cuantificar en aproximadamente un 40% en ambos conceptos.

2. Problemas, consideraciones y seguridad

Pese a que en los anteriores apartados hemos enumerado las ventajas, tanto económicas como operativas relativas a la implantación de plataformas de Cloud Computing así como de sistemas de virtualización, en este apartado vamos a mostrar los problemas y obstáculos que se derivan del empleo de estos paradigmas, mostrando especial atención en la adopción de soluciones de nube pública.

2.1. Obstáculos del Cloud Computing

El Cloud Computing también se enfrenta a numerosos obstáculos que pueden frenar su avance, como veremos más adelante, cada obstáculo puede ser emparejado con una oportunidad, además se presentan ideas sobre como superarlos:

2.1.1. Disponibilidad de servicio

Las organizaciones se preocupan por el nivel de disponibilidad que tienen sus servicios, esta preocupación es mayor si externalizamos la infraestructura que soportará dichos servicios, ya que delegamos gran parte del control que tenemos sobre los mismos a un tercero. Irónicamente, los productos SaaS (Software as a Service) existentes han establecido un alto standard en este sentido. Google Search por ejemplo la página de inicio de mucha gente, si los usuarios se encuentran que Google no está disponible podrían pensar que Internet ha caído. Los usuarios esperan una disponibilidad similar de nuevos servicios, lo cual es difícil de hacer.

Así como los grandes proveedores de servicios de Internet utilizan varios proveedores de red, de modo que el fallo de una sola empresa no afecte a todo el servicio, creemos que la única solución plausible para obtener una disponibilidad muy alta pasa por tener múltiples proveedores de Cloud Computing.

La comunidad informática de alta disponibilidad ha seguido por largo tiempo el mantra de "*No single source of failure*" (Ni un sólo foco de fallos), sin embargo, la gestión de un servicio de Cloud Computing por una sola empresa es en realidad un punto único de fallo.

Incluso si la empresa tiene varios centros de datos en diferentes regiones geográficas con diferentes proveedores de red, es posible disponer de una infraestructura común de soft-

ware y sistemas de contabilidad, o incluso la empresa puede quebrar y quedar fuera del negocio.

Los grandes clientes se muestran reacios a migrar hacia la computación en nube, sin una estrategia de continuidad del negocio para este tipo de situaciones.

Otro obstáculo es la existencia de ataques distribuidos de denegación de servicio (DDoS). Los cibercriminales amenazan con cortar los ingresos de los proveedores de SaaS realizando ataques de este tipo, según datos del INTECO (Instituto Nacional de Tecnologías de la Comunicación) es relativamente frecuente extorsionar a dichos proveedores obligándoles a realizar pagos de 10.000€ a 50.000€ para evitar el lanzamiento de un ataque DDoS. Este tipo de ataques suelen utilizar grandes "botnets" que contratan los robots de alquiler en el mercado negro a unos 0,03€ por bot a la semana.

Supongamos que una instancia EC2 puede manejar 500 robots, y se inicia un ataque que genera un extra de 1 GB / segundo de ancho de banda con un total de 500.000 robots a 0.03€ por bot, un ataque le costaría al atacante 15.000 euros pagados por adelantado.

A los precios actuales de AWS (*Amazon Web Services)* , el ataque le costaría a la víctima un extra de 360€ por hora en ancho de banda y un extra de 100€ por hora de la computación. El ataque por lo tanto, tendría que durar 32 horas con el fin de los costos de la potencial víctima superaran a los del atacante, un ataque botnet de este plazo puede ser difícil de sostener, ya que cuanto más tiempo dura el ataque más fácil es descubrirlo y defenderse, del mismo modo los bots no pueden ser inmediatamente re-utilizados para otros ataques contra el mismo proveedor.

Es por ello que el Cloud Computing ha cambiado también el objetivo de los atacantes, pasando de los proveedores de SaaS hacia los proveedores de infraestructura (IaaS).

2.1.2. Lock-in de datos

La interoperabilidad entre plataformas ha mejorado, pero las API's para Cloud Computing en sí siguen siendo esencialmente propietarias, o al menos no han sido objeto de una normalización activa. De este modo, los clientes no puede extraer fácilmente sus datos y programas de un sitio a funcionar en otro.

La preocupación por la dificultad de extracción de los datos de la nube es uno de los principales factores de desconfianza de muchas organizaciones hacia la adopción de Cloud Computing.

El lock-in de clientes puede ser atractivo para los proveedores de Cloud Computing, pero los usuarios son vulnerables a los aumentos de precios, a los problemas de fiabilidad, o incluso a la salida del negocio de los proveedores.

Por ejemplo, un servicio de almacenamiento online llamado "Linkup" cerró el 8 de agosto de 2008 después de dejar sin acceso hasta un 45% de los datos de los clientes. Linkup, a su vez, se había basado en otro servicio de almacenamiento online llamado Nirvanix para almacenar los datos de los clientes, ambas organizaciones se acusaron mutuamente l como

responsables de la pérdida de los datos de los clientes. Mientras tanto, Linkup le comunicó a 20.000 usuarios que el servicio ya no estaba disponible y se les instó a probar otro sitio de almacenamiento.

La solución obvia es la estandarización de las API para que un desarrollador de SaaS pueden desplegar servicios y datos a través de múltiples proveedores de Cloud Computing para evitar que el fallo de una sola empresa se lleve todas las copias de datos de clientes con ella.

2.1.3. Confidencialidad de los datos

Podemos afirmar que no hay ningún obstáculo fundamental para lograr un entorno de cloud computing tan seguro como la gran mayoría de los ambientes tradicionales de TI, y que muchos de los obstáculos se pueden superar de inmediato con un buen entendimiento de tecnologías tales como el almacenamiento cifrado, redes virtuales de área local, y redes de middleboxes (por ejemplo, firewalls, filtros de paquetes).

Un problema relacionado es que muchos países tienen leyes que requieren los proveedores de SaaS para mantener los datos del cliente y el material con derechos de autor dentro de las fronteras nacionales. Del mismo modo, algunas empresas no les gusta la capacidad de un país para acceder a sus datos a través del sistema judicial, por ejemplo, un cliente europeo, podría estar preocupado sobre el uso de SaaS en los Estados Unidos, dada la *Patriot Law* de los EE.UU.

Cloud Computing ofrece SaaS y los proveedores de SaaS los usuarios una mayor libertad para colocar su almacenamiento. Por ejemplo, Amazon S3 proporciona servicios ubicados físicamente en los Estados Unidos y en Europa, permitiendo a los proveedores para mantener los datos en lo que ellos elijan. Con las regiones de AWS, un simple cambio de configuración evita la necesidad de encontrar y negociar con un proveedor de hosting en el extranjero.

2.1.4. Cuellos de botella en transferencias

Las aplicaciones cada vez hacen un uso más intensivo de los datos, si asumimos que las aplicaciones pueden ser localizadas fuera de los límites de la nube, se complica la localización y el transporte. A un coste medio de unos 100€ a 150€ por terabyte transferido, el coste general de la infraestructura puede aumentar sensiblemente haciendo que los costes por transferencia de datos sean un asunto importante.

Los usuarios del cloud y los proveedores tienen que pensar sobre las implicaciones de la localización y el tráfico en todos los niveles del sistema si quieren minimizar costes. Este modo de razonamiento puede ejemplificarse en el desarrollo de Amazon Cloudfront.

Desde el punto de vista del proveedor de servicios, una oportunidad para superar el alto coste de las transferencias a través de Internet es hacer atractivo el hecho de conservar los datos en el cloud, desde el momento en que los datos están en el cloud dejan de ser un cuello de botella y pueden necesitar nuevos servicios que pueden llevar a la compra de ciclos de Cloud Computing. Amazon recientemente comenzó a alojar grandes volúmenes de

datos en S3 de forma gratuita; ya que no hay cargo por transferir datos entre S3 y EC2, estos datasets pueden atraer la compra de ciclos EC2.

Otro ejemplo puede ser un servicio de backup, ya que compañías como Amazon, Google o Microsoft habitualmente envían más datos que reciben, el coste de recibir datos en el cloud debe ser mucho menor, por ejemplo, si semanalmente hay backups completas que se mueven enviando discos físicos, y backups incrementales que se mueven a través de la red, el proveedor de Cloud Computing debería ser capaz de ofrecer una manera económica de backup, nuevos servicios pueden hacerse posibles y pueden resultar en vender mas ciclos de Cloud Computing.

2.1.5. Rendimiento impredecible

La experiencia nos dice que múltiples máquinas virtuales pueden compartir CPU y memoria sorprendentemente bien en Cloud Computing, no obstante la compartición de Entrada/Salida es más problemática. La figura 1.5.1.1(a) muestra la media de ancho de banda para 75 instancias de EC2 corriendo el benchmark de memoria STREAM. La media es de 1355 Mb por segundo, con una desviación típica de sólo 52 Mb/s, menos de un 4% de la media.

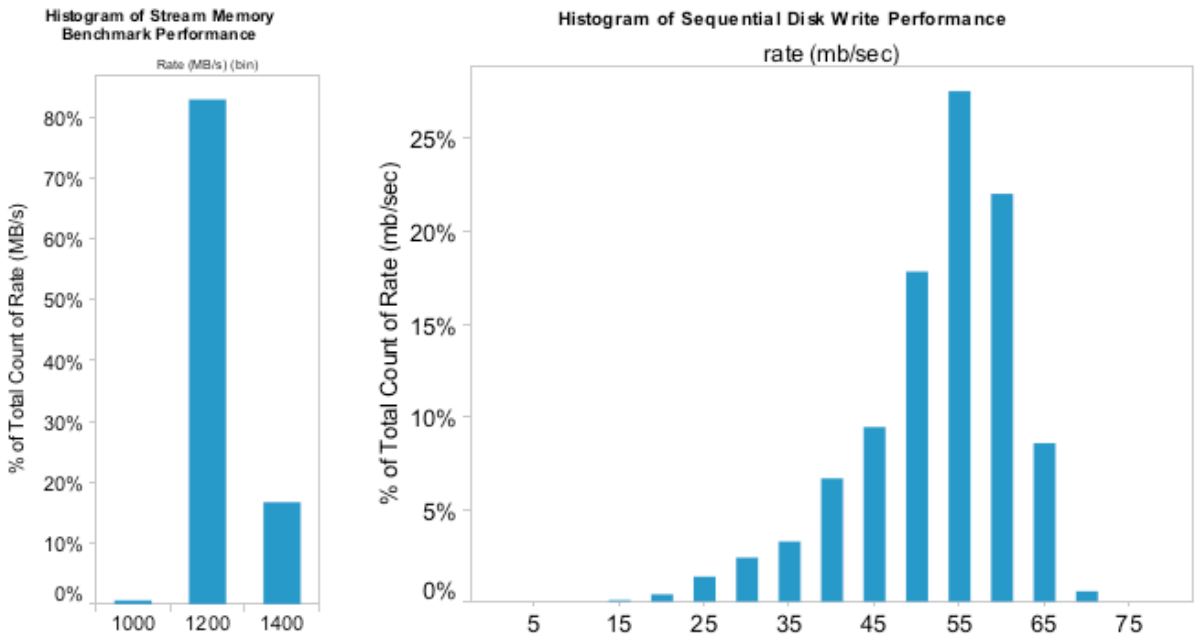

*Figura 2.1.5.1: En la izquierda **(a)** Benchmark de memoria en 75 máquinas virtuales corriendo el banco de pruebas STREAM. A la derecha: **(b)** Rendimiento de disco escribiendo 1 Gb de datos en 75 máquinas virtuales*

La Figura 2.1.5.1(b) muestra la media de ancho de banda de disco para 75 instancias EC2 cada una escribiendo 1GB de ficheros al disco local. La media de escritura de disco está cerca de 55 Mb/s con una desviación típica de un poco mas de 9 Mb/s, más de un 16% de la media. Esto demuestra el problema de interferencia de Entrada/Salida. Una posibilidad para paliar este fenómeno pasa por mejorar arquitecturas y sistemas operativos con el fin de virtualizar las interrupciones y los canales de Entrada/Salida más eficientemente.

Tecnologías como PCI-Express son difícilmente virtualizables, pero no por ello dejan de ser críticas para el cloud. Un motivo para ser optimista es que es que los mainframes y sistemas operativos de IBM superaron estos problemas durante la década de 1980, por tanto tenemos ejemplos de donde aprender.

Otra posibilidad es que la memoria flash reducirá la interferencia de Entrada/salida. La memoria flash está basada en semiconductores que preserva la información aún sin corriente eléctrica de la misma forma que los discos rígidos tradicionales, pero debido a que no tiene partes móviles el acceso es mucho más rápido (microsegundos vs milisegundos) y además consume menos energía. La memoria flash puede soportar muchos mas operaciones de Entrada/salida por segundo, por tanto múltiples máquinas virtuales con trabajos sobre Entrada/Salida en conflicto pueden convivir mucho mejor en el misma máquina física sin la interferencia que hemos visto en el anterior apartado, esta serie de ventajas pueden ser aplicadas de igual forma, tanto para la memoria principal como para el almacenamiento, mejorando el rendimiento sensiblemente y por tanto reduciendo costes hacia los proveedores de Cloud Computing y eventualmente a los usuarios.

2.1.6. Almacenamiento escalable

En anteriores párrafos hemos identificado tres propiedades que combinadas dan al Cloud Computing su aspecto: Utilización de corto plazo (lo que implica escalar hacia abajo una vez los recursos no son necesitados), sin coste por adelantado y “infinita” capacidad bajo demanda. Este tipo de características están claras para requisitos computacionales, pero es menos obvio si nos referimos a recursos de almacenamiento.

Han habido intentos de responder a esta pregunta basadas en la riqueza de las API's de consulta y almacenamiento, las garantías de rendimiento ofrecidas y la complejidad de las estructuras de datos que están directamente soportadas por el sistema de almacenamiento. La oportunidad, que es aún un tema de investigación, es crear un sistema de almacena-

miento que no sólo alcanzará estos requisitos sino que los combinará con las ventajas del Cloud Computing.

2.1.7. Bugs en sistemas distribuidos de gran escala

Uno de los retos complicados de superar en Cloud Computing es eliminar errores en sistemas distribuidos de gran escala. Un hecho común es que esos bugs no puedan ser reproducidos en configuraciones más pequeñas, por tanto el debugging debe ocurrir en sistemas de la escala de los datacenters de producción.

Una posibilidad puede venir de la confianza en las máquinas virtuales dentro del Cloud Computing. Muchos proveedores de SaaS han desarrollado su infraestructura sin utilizar máquinas virtuales, puede ser porque lo hicieron antes de la popularidad de las máquinas virtuales o porque sintieron que no podían soportar el coste en el rendimiento de las máquinas virtuales.

Dado que las máquinas virtuales están de rigor dentro del Utility Computing, ese nivel de virtualización puede hacer posible captura información muy valiosa de una forma que sería imposible sin la utilización de máquinas virtuales.

2.1.8. Licencias de software

Las actuales licencias de software comúnmente restringen las máquinas donde pueden funcionar, los usuarios pagan por el software, y después una tasa de mantenimiento anual. SAP por ejemplo anunció que iba a incrementar su tasa de mantenimiento anual al menos un 22% del precio de compra del software, que es comparable con los precios de Oracle.

Muchos proveedores de Cloud Computing originalmente confiaron en software open source en parte porque el sistema de licencias tradicional para software comercial no es demasiado adecuado para Utility Computing.

La situación actual es que muchas compañias de software están cambiando su sistema de licencias para encajar mejor en el Cloud Computing, por ejemplo Microsoft y Amazon ofrecen licencias de software cuyo pago está basado en el uso que se de de las mismas sobre la plataforma EC2, una licencia de Windows Server y Windows SQL Server corriendo sobre una instancia de EC2 cuestan 0,15€ por hora contra los 0,10€ que cuesta la misma instancia con software open source instalado.

2.2. Cuestiones sobre seguridad

La Cloud Security Alliance se define como una organización internacional sin ánimo de lucro para promover el uso de mejores prácticas para garantizar la seguridad en cloud. En marzo del 2010 publicó un informe «Top Threats to Cloud Computing V1.0» sobre las siete mayores amenazas de la infraestructuras cloud, con el propósito de asistir a las organizaciones en la toma de decisiones y en la adopción de estrategias que incluyan cloud computing. Estas amenazas se actualizan regularmente buscando el consenso de los expertos. A continuación, se resumen las amenazas descritas en este informe.

2.2.1. Mal uso del Cloud Computing

Esta amenaza afecta principalmente a los modelos de servicio IaaS y PaaS y se relaciona con un registro de acceso a estas infraestructuras/plataformas poco restrictivo. Es decir, cualquiera con una tarjeta de crédito válida puede acceder al servicio, con la consecuente proliferación de spammers, creadores de código malicioso y otros criminales que utilizan la nube como centro de operaciones.

Para ejemplificar esta amenaza podemos citar un caso ocurrido en 2009, en el que especialistas de seguridad de Amazon se dieron cuenta de que una botnet Zeus estaba utilizando el

servicio EC2 como plataforma de ataque hacia otros servidores. Los hackers estaba utilizando el servicio RDS (Relational Database Service) de Amazon como un backend alternativo en caso de que perdieran el control hacia el dominio original, lo que resultaría en una pérdida completa del acceso hacia datos financieros obtenidos de los hosts infectados.

Las recomendaciones citadas por la *Cloud Alliance* a fin de evitar estas amenazas son las siguientes:

- Implementar un sistema de registro de acceso más restrictivo
- Coordinar y monitorizar el fraude en tarjetas de crédito
- Monitorizar el trafico de clientes para la detección de posibles actividades ilícitas
- Comprobar las listas negras públicas para identificar si los rangos IP de la infraestructura han entrado en ellas

2.2.2. Interfaces y API poco seguros

Generalmente los proveedores de servicios en la nube ofrecen una serie de interfaces y API (del inglés, Application Programming Interface) para controlar e interactuar con los recursos. De este modo, toda la organización, el control, la provisión y la monitorización de los servicios cloud se realiza a través de estos API o interfaces.

Dado que todo (autenticación, acceso, cifrado de datos, etc.) se realiza a través de estas herramientas, se hace necesario que los interfaces estén diseñados de forma segura, evitando así los problemas de seguridad, tanto los que son intencionados como los que se producen de forma accidental.

Ejemplos de estas malas prácticas pueden ser por ejemplo, el permitir accesos anónimos, reutilización de tokens, autentificaciones sin cifrar, etc.

Las recomendaciones citadas por la *Cloud Alliance* :

- Analizar los problemas de seguridad de las interfaces de los proveedores de servicio
- Asegurarse que la autenticación y los controles de acceso se implementan teniendo en cuenta el cifrado de los datos

2.2.3. Amenaza interna

Como en todos los sistemas de información, la amenaza que suponen los propios usuarios es una de las más importantes, dado que tienen acceso de forma natural a los datos y aplicaciones de la empresa. En un entorno cloud esto no es en absoluto diferente ya que se pueden desencadenar igualmente incidentes de seguridad provocados por empleados descontentos y accidentes por error o desconocimiento.

Además, en muchos casos, es el propio proveedor del servicio el que gestiona las altas y bajas de los usuarios, produciéndose brechas de seguridad cuando el consumidor del servicio no informa al proveedor de las bajas de personal en la empresa.

Como es lógico, estos incidentes repercuten de forma importante en la imagen de la empresa y en los activos que son gestionados.

Los proveedores de servicio deberán proveer a los consumidores del servicio de medios y métodos para el control de las amenazas internas.

Recomendaciones:

- Especificar cláusulas legales y de confidencialidad en los contratos laborales
- Determinar los posibles problemas en los procesos de notificación

2.2.4. Problemas derivados de las tecnologías compartidas

Esta amenaza afecta a los modelos IaaS, ya que en un modelo de Infraestructura como Servicio los componentes físicos (CPU, GPU, etc.) no fueron diseñados específicamente para una arquitectura de aplicaciones compartidas. Se han dado casos en los que los hipervisores de virtualización podían acceder a los recursos físicos del anfitrión provocando, de esta forma, incidentes de seguridad.

Para evitar este tipo de incidentes se recomienda implementar una defensa en profundidad con especial atención a los recursos de computación, almacenamiento y red. Además, se ha de generar una buena estrategia de seguridad que gestione correctamente los recursos para que las actividades de un usuario no puedan interferir en las del resto.

Recomendaciones:

- Diseñar buenas prácticas para la instalación y configuración
- Monitorizar los entornos para detectar cambios no deseados en las configuraciones o la actividad
- Proporcionar autenticación fuerte y control de acceso para el acceso de administración
- Adecuar los acuerdos de nivel de servicio para controlar el parcheado y la corrección de vulnerabilidades

2.2.5. Pérdida o fuga de la información

Existen muchas formas en las que los datos se pueden ver comprometidos. Por ejemplo, el borrado o modificación de datos sin tener una copia de seguridad de los originales, supone una pérdida de datos.

En la nube, aumenta el riesgo de que los datos se vean comprometidos ya que el número de interacciones entre ellos se multiplica debido a la propia arquitectura de la misma. Esto deriva en pérdida de imagen de la compañía, daños económicos y, si se trata de fugas, problemas legales, infracciones de normas, etc.

Ejemplos de estas prácticas pueden ser el mal uso de claves de cifrado y de software, la autenticación y autorización débil, etc.

Recomendaciones:

- Implementar API potentes para el control de acceso
- Proteger el tránsito de datos mediante el cifrado de los mismos
- Analizar la protección de datos tanto en tiempo de diseño como en tiempo de ejecución
- Proporcionar mecanismos potentes para la generación de claves, el almacenamiento y la destrucción de la información
- Definir, por contrato, la destrucción de los datos antes de que los medios de almacenamiento sean eliminados de la infraestructura, así como la política de copias de seguridad

2.2.6 Riesgos por desconocimiento

Uno de los pilares de las infraestructuras cloud es reducir la cantidad de software y hardware que tienen que adquirir y mantener las compañías, para así poder centrarse en el negocio. Esto, si bien repercute en ahorros de costes tanto económicos como operacionales, no puede ser motivo para el deterioro de la seguridad por falta de conocimiento de esta infraestructura.

Para asistir en la toma de decisiones sobre las medidas de seguridad que se han de implantar en un entorno cloud es conveniente conocer, al menos en parte, la información técnica de la plataforma. Datos como con quién se comparte la infraestructura o los intentos de acceso no autorizados pueden resultar muy importantes a la hora de decidir la estrategia de seguridad.

La carencia de información de este tipo puede derivar en brechas de seguridad desconocidas por el afectado.

Recomendaciones:

- Tener acceso a los logs (registros de actividad) de aplicaciones y datos
- Estar al corriente, total o parcialmente, de los detalles de la infraestructura
- Monitorizar y recibir alertas sobre el uso de información crítica

2.2.7. Respuesta a incidentes

La labor del proveedor es básica en las actividades de respuesta ante la ocurrencia de algún incidente de seguridad. Esto incluye la verificación, el análisis del ataque, la contención, la recolección de evidencias, la aplicación de remedios y la restauración del servicio.

La colaboración entre los proveedores y los suscriptores para la detección y reconocimiento de los incidentes es esencial para la seguridad y la privacidad en cloud computing, ya que

la complejidad de los servicios puede dificultar la labor de la detección. Se hace necesario entender y negociar los procedimientos de respuesta a incidentes antes de firmar un contrato de servicio. La localización de los datos es otro aspecto que puede impedir una investigación, por lo que es otro de los puntos que se deben negociar en los contratos.

La solución que se negocie ha de tener la finalidad de mitigar el incidente en un tiempo que limite los daños y que mejore los tiempos de recuperación. Los equipos para la resolución deberían ser mixtos (proveedor y suscriptor) ya que la solución puede involucrar a alguna de las partes de forma individual o a ambas conjuntamente y el incidente puede incluso afectar a otros suscriptores que comparten la infraestructura.

Recomendaciones de seguridad según NIST (National Institute of Standards and Technology)

Área	Recomendación
Gobernanza	Implantar políticas y estándares en la provisión de servicios *cloud*. Establecer mecanismos de auditoría y herramientas para que se sigan las políticas de la organización durante el ciclo de vida.
Cumplimiento	Entender los distintos tipos de leyes y regulaciones y su impacto potencial en los entornos *cloud.* Revisar y valorar las medidas del proveedor con respecto a las necesidades de la organización.
Confianza	Incorporar mecanismos en el contrato que permitan controlar los procesos y controles de privacidad empleados por el proveedor.
Arquitectura	Comprender las tecnologías que sustentan la infraestructura del proveedor para comprender las implicaciones de privacidad y seguridad de los controles técnicos.

Área	Recomendación
Identidad y control de acceso	Asegurar las salvaguardas necesarias para hacer seguras la autenticación, la autorización y las funciones de control de acceso.
Aislamiento de software	Entender la virtualización y otras técnicas de aislamiento que el proveedor emplee y valorar los riesgos implicados
Disponibilidad	Asegurarse que durante una interrupción prolongada del servicio, las operaciones críticas se pueden reanudar inmediatamente y todo el resto de operaciones, en un tiempo prudente.
Respuesta a incidentes	Entender y negociar los contratos de los proveedores así como los procedimientos para la respuesta a incidentes requeridos por la organización.

Tabla 2.2.7.1: Recomendaciones de seguridad según NIST.

2.3. Problemas de la virtualización

Cómo hemos dicho en anteriores apartados, la virtualización comporta grandes beneficios, tanto económicos como operativos, no obstante no es un paradigma perfecto y también presenta problemas, podemos englobar los principales problemas derivados de la virtualización en los siguientes apartados:

- **Facilidad y velocidad de implementación de servidores virtuales:** En entornos virtuales, todo se pude hacer o cambiar extremadamente mas rápido. Por ejemplo, el ciclo de implementación de nuevos servidores virtuales (maquinas virtuales) puede realmente reducirse de días a minutos, o incluso, a segundos. La posibilidad de hacer estos despliegues de una forma mucho mas rápida es, sin lugar a dudas, un benefi-

cio extraordinario para cualquier centro de datos. Pero también existe la necesidad de mitigar el riesgo de errores y de la actividad maliciosa con la misma rapidez. Si las organizaciones carecen de buenas prácticas de planificación, el “factor velocidad” puede exacerbar las debilidades en los procesos de su empresa. Este factor de velocidad en un entorno de cambio continuo, puede resultar inexorablemente en la falta de entendimiento del estado actual de los activos en su centro de datos

- **Consolidaciones de switches de red y servidores en un solo servidor físico**: En una infraestructura física, los servidores y las redes son gestionadas a través de numerosas aplicaciones por separado. En una infraestructura virtual, los servidores y las redes pueden ser gestionados por el mismo software de la capa de virtualización. Los administradores de red están familiarizados con el control de sus switches físicos a través de la aplicación de gestión de red que utilizan para su red física. Por consiguiente, deben adaptarse a las nuevas herramientas de gestión y mejores prácticas en un entorno de red virtual. Estos mismos administradores de red deben actualizar sus conocimientos para operar el software de virtualización con soltura y evitar errores de configuración ya que los riesgos asociados con una mala configuración de red del entorno virtual son muy altos y las consecuencias pueden ser desastrosas para su centro de datos.

- **Encapsulación de las maquinas virtuales:** A diferencia de un servidor físico, una máquina virtual es un conjunto de ficheros que residen físicamente en un almacenamiento compartido. Esta propiedad de encapsulación permite métodos mucho mas sencillos de asegurar lacontinuidad del servicio. Sin embargo, este tipo de encapsulación de la máquina virtual ofrece un nuevo tipo de “robo de datos”. Como la máquina virtual es solo un conjunto de ficheros, un servidor entero puede ser copiado a

un dispositivo USB o copiado durante un proceso de backup no autorizado a un lugar no protegido por su personal de seguridad o administrador de su entorno virtual.

- **Actualizaciones de parches de seguridad:** Es muy fácil despreocuparse de máquinas virtuales que no están en activo, particularmente en entornos de desarrollo. Un ejemplo podría de este posible escenario es el caso de un desarrollador de software que usa algunas máquinas virtuales por algún tiempo y luego las apaga. Un mes mas tarde, este mismo desarrollador vuelve a usar las máquinas virtuales y las enciende sin preocuparse de que estas maquinas virtuales deberían haber sido parcheadas con sus correspondientes parches de seguridad. Asimismo, en un entorno virtual ya no existe la relación one-to-one entre servidor físico y aplicación. Ahora, una máquina virtual puede ejecutarte en cualquier servidor físico y cada servidor físico puede tener una gran variedad de máquinas virtuales. Esta asociación puede cambiar dinámicamente mediante el uso de tecnologías de migraciones en caliente de dichas máquinas virtuales lo que hace más difícil estar al día con todos los cambios en cuanto a los parches de seguridad se refiere.

- **Consolidación de servidores:** La consolidaciones de servidores reduce los gastos de capital (CAPEX) así como los gastos de operaciones (OPEX). Pero también, no es menos cierto, ahora muchos mas servicios que se están ejecutando en estas máquinas virtuales se basan en menos servidores físicos. Por consiguiente, si un servidor físico no es configurado de la forma correcta o es atacado, esto podría afectar a muchos mas servidores virtuales.

3. Background técnico

En este apartado analizaremos los detalles técnicos de las tecnologías de virtualización y el paradigma de cloud computing, así como de los aspectos e implementaciones derivadas de las mismas.

3.1 Virtualización

La virtualización no es un tema nuevo, de hecho ronda desde hace 40 años. Los primeros usos de la virtualización incluyen el IBM 7044, el Sistema de Tiempo Compartido Compatible (CTSS - Compatible Time Sharing System) desarrollado en el Instituto Tecnológico de Massachusetts (MIT - Massachussets Institute of Technology) en el IBM 704. Y el proyecto Atlas de la Universidad de Manchester (uno de los primeros superordenadores del mundo), que fue pionero en el uso de memoria virtual con paginación y llamadas de supervisor.

La tecnología de la virtualización ha supuesto una revolución en los últimos años dentro del mundo de la informática, uno de los objetivos básicos de cualquier empresa, ya sea a me-

dio o a largo plazo, es expandir su actividad hacia nuevos segmentos del mercado y hacia un mayor número de clientes.

No obstante, este propósito implica necesariamente una expansión de las infraestructuras de TI que no siempre es tan eficaz como debería. La tendencia a añadir servidores para ejecutar nuevas aplicaciones puede dar como resultado recursos infrautilizados, mayores costes de gestión, y una reducción en la agilidad y en la fiabilidad de las operaciones.

Para hacer frente a estos inconvenientes, desde unos años las organizaciones están recurriendo a la virtualización como método para asegurar un mejor aprovechamiento de sus recursos y para lograr una reducción de costes.

Como virtualización se entiende a la consolidación de multiples piezas de equipos, como servidores o unidades de cinta, en una sola unidad física, recurriendo generalmente a un software de virtualización.

Tal como describe la compañía VMware, uno de los proveedores más importantes de este tipo de soluciones, se trata de una capa abstracta que separa el hardware físico del sistema operativo para lograr una mayor utilización y flexibilidad de los recursos de TI. O, lo que es lo mismo, la virtualización consiste en utilizar menos hardware para lograr mayores beneficios. Y gracias a esta consolidación de aplicaciones en un menor número de servidores, las organizaciones reducen la proliferación de recursos, simplifican su gestión y mejoran su utilización, aportando más agilidad y fiabilidad a la red.

Tanto es así que los niveles de utilización de servidores del 10% pueden incrementarse hasta el 60%, tal como estiman algunas compañías.

Desde una perspectiva de negocio, hay muchas razones para utilizar virtualización. La mayoría están relacionadas con la consolidación de servidores. Simple, si puedes virtualizar un número de sistemas infrautilizados en un solo servidor, ahorrarás energía, espacio, capacidad de refrigeración y administración ya que tienes menos servidores. Como puede ser difícil determinar el grado de utilización de un servidor, las tecnologías de virtualización soportan la migración en directo. La migración en directo permite que un sistema operativo y sus aplicaciones se muevan a un nuevo servidor para balancear la carga sobre el hardware disponible.

La virtualización también es importante para los desarrolladores. El núcleo Linux ocupa un solo espacio de direcciones, lo que significa que un fallo en el núcleo o en cualquier driver provoca la caída del sistema operativo completo. La virtualización supone que puedes ejecutar varios sistemas operativos, y si uno cae debido a un fallo, el hipervisor y el resto de sistemas operativos continuarán funcionando. Esto puede hacer que depurar el núcleo sea una tarea más parecida a depurar aplicaciones en el espacio del usuario.

3.1.1. Tipos de virtualización

A continuación, analizaremos las técnicas más comunes de virtualización y algunas de sus implementaciones.

3.1.1.1. Virtualización de hardware

Consiste en emular, mediante máquinas virtuales, los componentes de hardware. De esta manera el sistema operativo no se ejecuta sobre el hardware real sino sobre el virtual.

IBM reconoció la importancia de la virtualización en la década de 1960 con el desarrollo del mainframe System/360 Model 67. El Model 67 virtualizó todas las interfaces hardware a través del Monitor de Máquina Virtual (VMM - Virtual Machine Monitor). En los primeros días de la computación, el sistema operativo se llamó supervisor. Con la habilidad de ejecutar sistemas operativos sobre otro sistema operativo, apareció el termino hypervisor (termino acuñado en la década de 1970).

El VMM se ejecutaba directamente sobre el hardware subyacente, permitiendo múltiples máquinas virtuales (VMs). Cada VM podía ejecutar una instancia de su propio sistema operativo privado -- al comienzo este era CMS, o Conversational Monitor System. Las máquinas virtuales han continuado avanzando, y hoy se pueden encontrar ejecutándose en el mainframe System z9. Lo que proporciona compatibilidad hacia atrás, incluso hasta la línea System/360

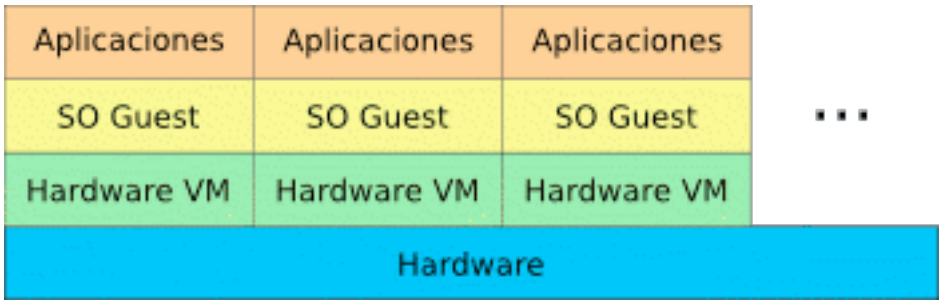

Figura 3.1.1.1.1: Virtualización de hardware

La gran ventaja de este enfoque es que pueden emularse distintas plataformas de hardware (por ejemplo, x86 sobre SPARC). Su principal desventaja es el alto costo de traducción de cada una de las operaciones de las máquinas virtuales a la máquina real, pudiendo obtenerse un rendimiento de 100 a 1000 veces menor.

3.1.1.2. Virtualización de procesador

El lenguaje Java ha seguido el modelo P-code en su máquina virtual. Esto ha permitido la amplia distribución de programas Java sobre incontables arquitecturas simplemente portando la JVM.

Otro de los usos iniciales de la virtualización, en este caso de un procesador simulado, es la máquina de pseudo-código (P-code machine). P-code es un lenguaje máquina que se ejecuta en una máquina virtual en lugar de en hardware real. P-code alcanzó la fama en la década de 1970 en el sistema Pascal de la Universidad de California, San Diego (UCSD), que compilaba programas Pascal en

P-code (o pseudo-código), y luego los ejecutaba en una máquina virtual P-code. Esto permitió que los programas P-code fuesen muy portables y pudiesen ejecutarse en cualquier lugar donde estuviese disponible una máquina virtual P-code.

El mismo concepto se utilizó en la decada de 1960 para el Basic Combined Programming Language (BCPL), un antepasado del lenguaje C. En este caso, un compilador compilaba

código BCPL en un código máquina intermedio llamado O-code. En un segundo paso, el O-code era compilado en el lenguaje nativo de la máquina de destino. Este modelo se utiliza en los compiladores modernos para proporcionar flexibilidad al portar los compiladores hacia nuevas arquitecturas destino (separando el front-end y el back-end por un lenguaje intermedio).

3.1.1.3. Virtualización a nivel del sistema operativo

Este es el otro extremo de la virtualización. En este esquema no se virtualiza el hardware y se ejecuta una única instancia del sistema operativo (kernel). Los distintos procesos perteneciente a cada servidor virtual se ejecutan aislados del resto.

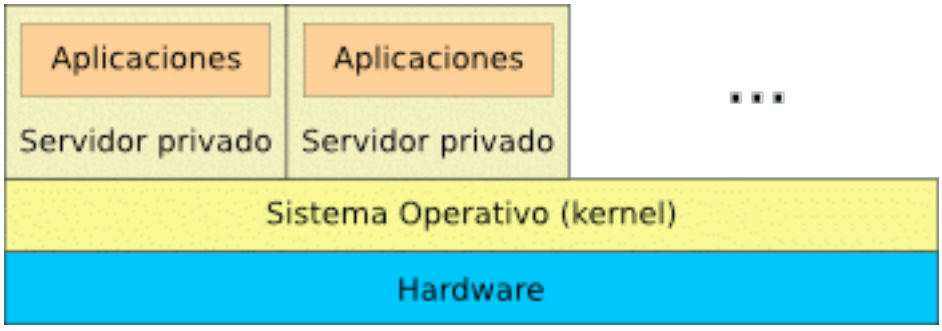

Figura 3.1.1.1.3: Virtualización a nivel de sistema operativo

La ventaja de este enfoque es la separación de los procesos de usuario prácticamente sin pérdida en el rendimiento, pero al compartir todos los servidores virtuales el mismo kernel no pueden obtenerse el resto de las ventajas de la virtualización.

3.1.1.4. Paravirtualización

La paravirtualización consiste en ejecutar sistemas operativos guests sobre otro sistema operativo que actúa como hipervisor (host). Los guests tienen que comunicarse con el hypervisor para lograr la virtualización.

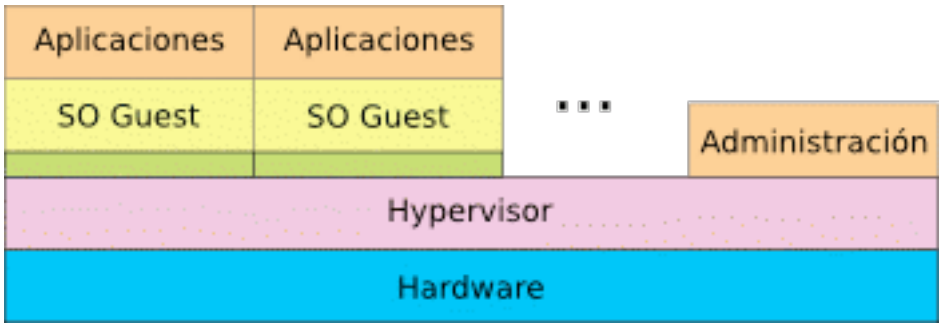

Figura 3.1.1.1.4: Paravirtualización

Las ventajas de este enfoque son un muy buen rendimiento y la posibilidad de ejecutar distintos sistemas operativos como guests. Se obtienen, además, todas las ventajas de la virtualización enunciadas anteriormente. Su desventaja es que los sistemas operativos guests deben ser modificados para funcionar en este esquema.

3.1.1.5. Virtualización completa

La virtualización completa es similar a la paravirtualización pero no requiere que los sistemas operativos guest colaboren con el hypervisor. En plataformas como la x86existen algunos inconvenientes para lograr la virtualización completa, que son solucionados con las últimas tecnologías propuestas por AMD e Intel.

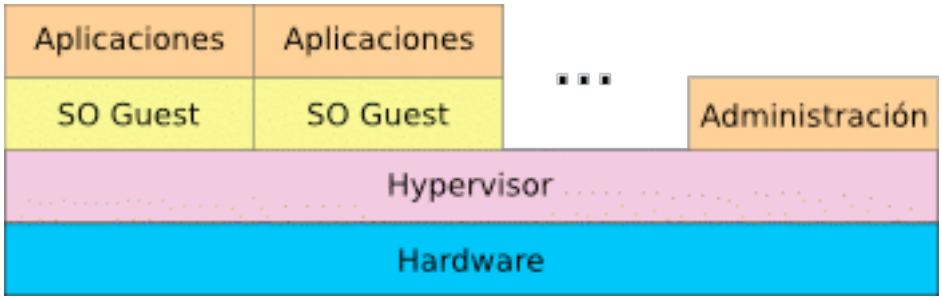

Figura 3.1.1.1.5: Virtualización completa

Este método tiene todas las ventajas de la paravirtualización, con el añadido de que no es necesaria ninguna modificación a los guests. La única restricción es que estos últimos deben soportar la arquitectura de hardware utilizada.

3.1.2. Herramientas de virtualización

En esta sección repasaremos las principales herramientas de virtualización existentes, ya sean opensource o privativas, analizaremos sus detalles técnicos así como sus ventajas sobre el resto de competidores y sus inconvenientes. Las herramientas a analizar son Xen, KVM, VirtualBox y Vmware.

3.1.2.1. Xen

Xen es un hipervisor (de código abierto que permite una mejor utilización de los servidores y la consolidación de los mismos al posibilitar que múltiples imágenes de sistemas operativos se ejecuten simultáneamente en una único servidor físico.

Xen proporciona garantías sobre los recursos a los servidores virtuales para asegurar que los niveles de servicio de cada aplicación se respeten, incluyendo CPU, memoria y entrada/salida. Xen es la infraestructura de virtualización por software más rápida y segura existente, y ha sido adoptada por los principales fabricantes y distribuidores, incluyendo a Intel, AMD, Dell, Hewlett-Packard, IBM, Novell, Red Hat o Sun Microsystems. Xen se distribuye bajo la licencia General Public License de GNU y puede descargarse gratuitamente.

Un servidor virtual es simplemente una instancia de un sistema operativo y su carga de trabajo, ejecutándose bajo el paraguas del hipervisor Xen. En lugar de controlar el hardware directamente, las instancias de sistemas operativos acceden al hardware a través del hipervisor, el cuál además tiene la capacidad de compartir los recursos con otras aplicaciones e instancias de sistemas operativos virtualizadas.

Xen fue creado en el año 2003 en el laboratorio de computación de la Universidad de Cambridge bajo lo que se conoce como el proyecto Xen Hypervisor liderado por Ian Pratt. Algunos de los miembros más destacados del proyecto son Keir Fraser, Steven Hand y Christian Limpach. Este mismo equipo fundó XenSource conjuntamente con Nick Gault y Simon Crosby, que aportaron su experiencia como empresarios en Silicon Valley.

Xen cada vez se usa más en centros de datos con el objetivo de incrementar la utilización de servidores y mejorar el coste total de propiedad (del inglés, Total Cost of Ownership). Xen es ámpliamente utilizado en proveedores de servicios de aplicaciones y compañías de hospedaje porque ofrece un control preciso de los recursos del sistema y permite a los usuarios hospedar más servidores virtuales por máquina física. Xen también se usa en el desarrollo y verificación del funcionamiento de aplicaciones, pues la virtualización permite a los desarrolladores de aplicaciones multihilo hospedar múltiples máquinas virtuales y comprobar su correcto funcionamiento, ahorrando costes en infraestructuras. Más aún, el hardware de pruebas puede ser readaptado instantáneamente para otros usos simplemente instanciando servidores virtuales con las imágenes deseadas. Finalmente, las aplicaciones que han sido verificadas pueden ser puestas en producción directamente desde el entorno de pruebas basado en Xen simplemente migrando la máquina virtual pertinente.

Xen usa de manera óptima las capacidades de virtualización por hardware de los procesadores VT de Intel y los Pacifica de AMD.

La paravirtualización es la clave del éxito de Xen, pues permite obtener un rendimiento drásticamente mayor que los acercamientos alternativos existentes en el mercado. La paravirtualización supone hacer que el sistema operativo del servidor virtual sea consciente de que está siendo virtualizado para permitir una colaboración entre ambas partes que facilite el rendimiento más óptimo. En Linux, BSD y Solaris, los hosts paravirtualizados ven a Xen como una capa de hardware idealizada. De hecho, Xen es simplemente una arquitectura de hardware idealizada del kernel de Linux. Intel ha realizado diversas contribuciones a Xen que han permitido añadir soporte para sus extensiones de arquitectura VT-X Vanderpool. Esta tecnología permite que sistemas operativos sin modificar actúen como hosts dentro de las máquinas virtuales de Xen, siempre y cuando el servidor físico soporte las extensiones VT de Intel o Pacifica de AMD. Para Microsoft Windows y otros hosts que no están al tanto de la existencia de Xen, la capa de virtualización VT de Intel, combinada con la paravirtualización de los controladores de Windows, permiten a Xen conseguir el mismo nivel de rendimiento que los hosts Linux virtualizados.

Debido al pequeño tamaño del código necesario para ejecutar el hipervisor, la sobrecarga en el rendimiento típicamente se encuentra entre el 0,1% y el 3,5% (datos tomados con los benchmarks estándar del mercado). Además, la técnica de paravirtualización le permite a Xen beneficiarse de todos los controladores nativos de Linux y, por lo tanto, soportar una gran cantidad de dispositivos. Los controladores paravirtualizados de Xen se ejecutan fuera del núcleo del hipervisor, donde se implementa una política de compartición de recursos entre las diversas máquinas virtuales, proporcionando así un particionamiento muy eficiente de los recursos de E/S entre los servidores virtuales. Otro benefi

cio de este acercamiento es que los controladores se ejecutan en un nivel de protección más bajo que Xen, manteniendo al hipervisor a salvo de errores en los controladores.

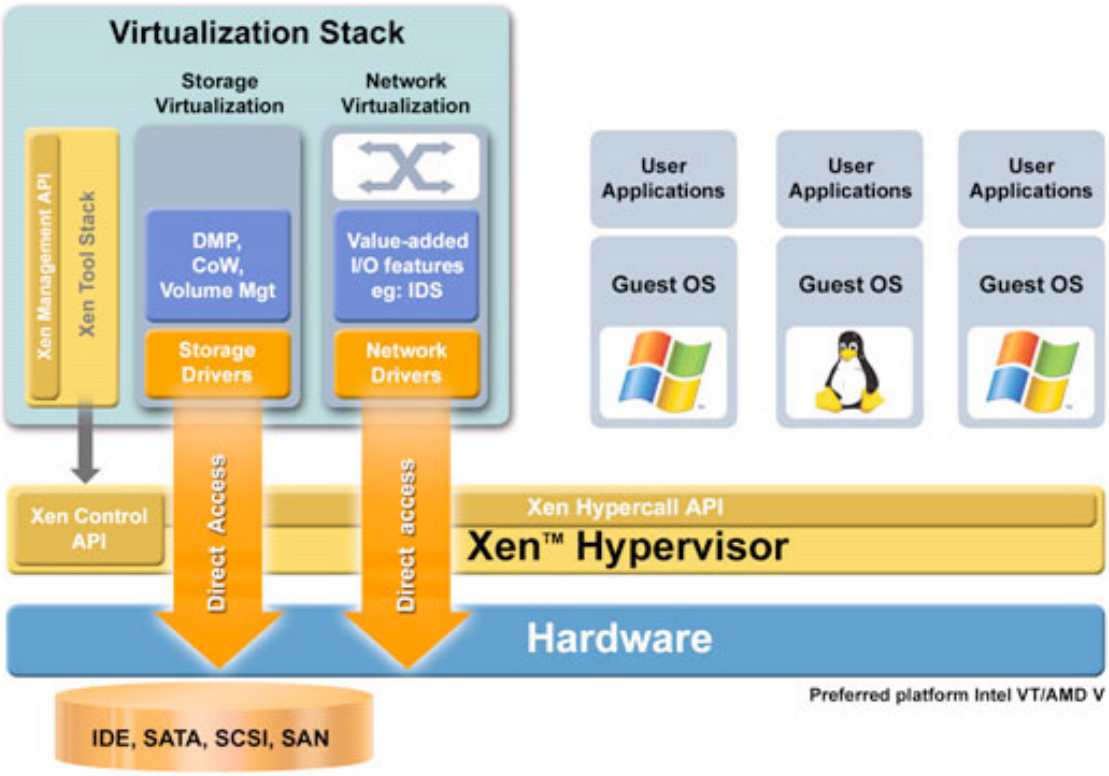

Figura 3.1.2.1.1 Arquitectura de Xen.

En términos de seguridad, Xen soporta un aislamiento absoluto de los recursos entre dominios, lo que significa que tiene el nivel más alto posible de separación y seguridad en un hardware de tipo i386. No es posible, por ejemplo, usar tcpdump en un host virtual para ver el tráfico de los demás hosts virtuales. Además, el código fuente base de Xen es muy pequeño (el núcleo del hipervisor tiene menos de 40.000 líneas), lo que permite realizar auditorías de seguridad en el código más fácilmente. Más importante aún, Xen puede usar las características de seguridad del hardware, como los Trusted Platform Modules (TPM), para construir una capa de monitorización del uso del hardware a través del software.

En el Intel Developer Forum de agosto del 2005, XenSource demostró una solución de hipervisor seguro al integrar Xen con el sistema de detección de intrusos Snort, aplicación de código abierto líder del mercado. De este modo los usuarios reciben una nueva y potente posibilidad para implementar las mismas políticas de seguridad en los servidores virtuales, independientemente del sistema operativo. Más aún, el hipervisor puede incluso asegurar que hosts que no han sido parcheados serán protegidos. Xen puede también impedir que un servidor virtual comprometido se use para atacar a otros servidores virtuales o físicos bloqueando su tráfico.

Las máquinas virtuales de Xen pueden migrarse en caliente entre hosts físicos sin necesidad de detenerlos. Durante este proceso, la memoria de la máquina virtual se copia iterativamente al destino sin parar su ejecución. Una pequeña pausa de entre 60 y 300 milisegundos es necesaria para llevar a cabo la sincronización final antes de que la máquina virtual empiece a ejecutarse en su nuevo destinatario, proporcionando así la apariencia de una migración sin parones. Una tecnología similar se usa para suspender a disco una máquina virtual en ejecución, cambiar a otra máquina virtual y recuperar más tarde la primera máquina virtual.

3.1.2.2. KVM

En un escenario de virtualización típico, un componente conocido como hipervisor ofrece una interfaz entre el sistema huésped y su anfitrión. El hipervisor reside en lo alto del sistema anfitrión, encargándose de la planificación de las tareas y la gestión de la memoria de cada huésped. KVM integra el hipervisor en el kernel, reduciendo así las redundancias y acelerando los tiempos de ejecución.

Un controlador de KVM se comunica con el kernel actuando como interfaz para una máquina virtual en espacio de usuario. La programación de las tareas y la gestión de la memoria son manejadas a través del mismo kernel. Un pequeño módulo del kernel Linux presenta el modo huésped, instala tablas de páginas para él y emula determinadas instrucciones clave.

Las versiones actuales de KVM vienen con una versión modificada del emulador Qemu que gestiona la E/S y opera como una residencia virtual para el sistema huésped. El sistema huésped se ejecuta dentro de Qemu, mientras que Qemu se ejecuta a su vez en espacio de usuario.

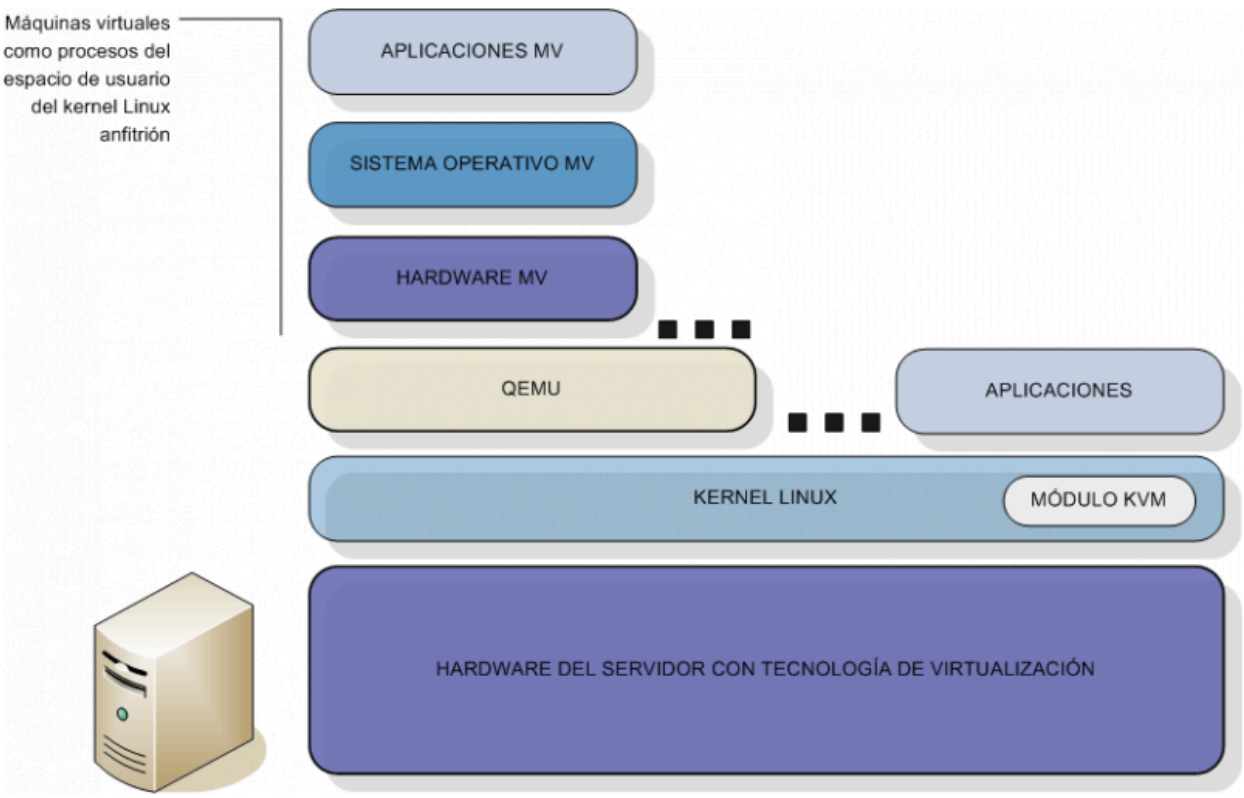

Figura 3.1.2.2.1 Arquitectura de KVM.

El entorno resultante es parecido al escenario representado en la Figura 2.1.2.2.2, en el que varios procesos de máquina virtual se ejecutan cerca de otras tareas de espacio de usuario

gestionadas directamente por el kernel. Cada huésped consta de dos partes: la parte de espacio de usuario (Qemu) y la parte huésped (el huésped en sí mismo). Los procesadores virtuales de dentro de una máquina virtual son simples hilos del proceso del anfitrión.

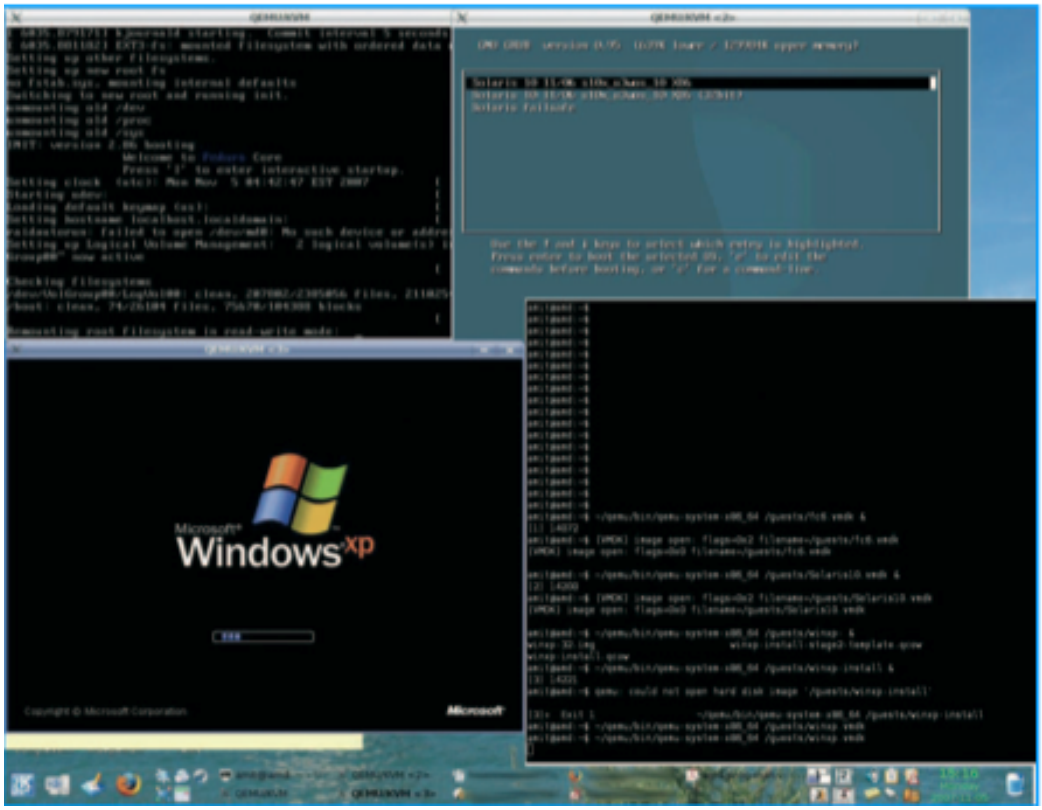

Figura 3.1.2.2.2 Un proceso de máquina virtual se ejecuta cerca de otras tareas en espacio de usuario y es gestionado por el kernel.

Este modelo encaja muy bien en la mentalidad Unix de hacer una cosa y hacerla bien. Lo que hace el módulo KVM es activar el modo huésped y gestionar los accesos virtualizados a los registros. Desde la perspectiva de un usuario, casi no hay diferencia entre ejecutar una máquina virtual Qemu con KVM deshabilitado y hacerlo con KVM habilitado, a excepción claro está del significativo aumento de velocidad.

KVM sigue la filosofía de desarrollo y publicación sobre la que se construye Linux: publicar pronto y a menudo. La última versión estable es parte del kernel Linux 2.6.x, y sus modificaciones aparecen como 2.6.x.y. Las fuentes de KVM se mantienen en un árbol git. Para

obtener la última versión de KVM o el último árbol, podemos dirigirnos a la wiki de KVM (http://kvm.qumranet.com).

3.1.2.3. Vmware

VMware es un sistema de virtualización por software, es similar a su homólogo Virtual PC, aunque existen diferencias entre ambos que afectan a la forma en la que el software interactúa con el sistema físico. El rendimiento del sistema virtual varía dependiendo de las características del sistema físico en el que se ejecute, y de los recursos virtuales (CPU, RAM, etc.) asignados al sistema virtual.

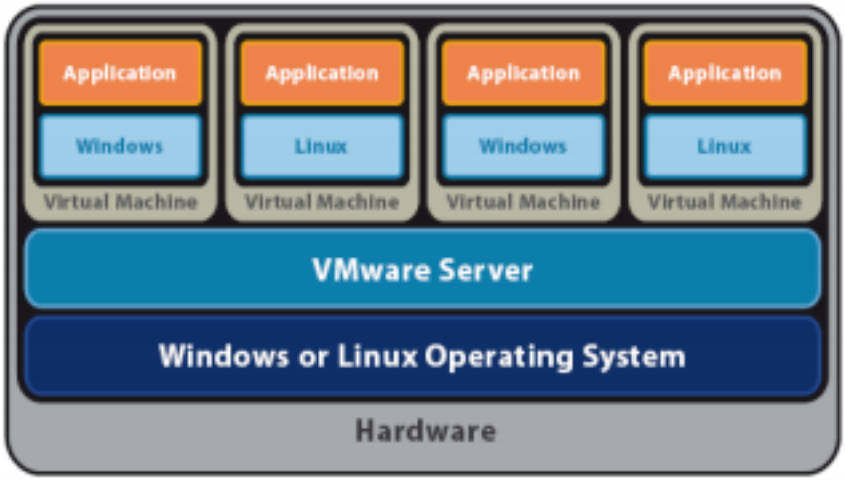

Figura 3.1.2.3.1 Esquema de servidor host con Vmware

Mientras que VirtualPC emula una plataforma x86, Vmware la virtualiza, de forma que la mayor parte de las instrucciones en VMware se ejecutan directamente sobre el hardware físico, mientras que en el caso de Virtual PC se traducen en llamadas al sistema operativo que se ejecuta en el sistema físico.

Dentro del amplio portfolio de Vmware podemos destacar las siguiente soluciones enfocadas al entorno corporativo:

Workstation

Es un producto de pago. Este producto permite crear VM y generar instancias múltiples de snapshots. Es una herramienta ideal para programadores que, de esta forma, pueden tener diversas líneas de desarrollo de una aplicación. Para un perfil de sistemas es también muy útil al permitir generar diferentes vías de investigación en una migración o solución de problemática. Siempre va una versión por delante de VMware Server, disponiendo de algunas funcionalidades que no se incorporan en Server, como la creación de Team, vídeos y mayores capacidades de hardware.

Fusion

Es un producto similar a Workstation y también de pago. Como sistema operativo Host tendrá System OS X. Es una plataforma de virtualización en sistemas macintosh muy robusta. Permite virtualizar hasta Windows Server 2008 x64. Su integración con el sistema operativo es tan grande que podemos tener, por ejemplo, una VM de XP funcionando y ocultar el in-

terfaz de VMware Fusion, de modo que las aplicaciones del XP se integren en el Dock de forma transparente. Su competencia directa es Virtual PC para MAC y Parallels.

Player

Es un producto gratuito. Ejecuta VM ya creadas, sin permitir modificar sustancialmente sus características, a diferencia de VMware Server. Es una herramienta ideal para poder proporcionar una VM a un usuario/cliente sin necesidad de instalarle ningún producto adicional.

Vmware Server

Está versión está pensada para responder ante una demanda mayor que Workstation. Otra diferencia entre VMware Server y Workstation es que se pueden ejecutar de manera concurrente más máquinas virtuales soportando servidores con hasta 32 procesadores y/o 64 GB de memoria, ofreciendo funcionalidad de administración remota, soporta una API avanzada y funcionalidad de scripting y se puede ejecutar en modo headless.

Su uso inicialmente se recomienda para entornos de prueba y pequeños escenarios de producción. Es gratuito, aunque podemos adquirir soporte para problemáticas puntuales o adquirir un mantenimiento continuo del producto si lo utilizamos frecuentemente y queremos tener respaldo garantizado.

Es una aplicación y se puede instalar en sistemas host Windows o Linux. Es un producto perfecto para iniciar la virtualización de una empresa. Una vez consolidada ésta, se puede migrar a un entorno VI de un modo muy simple con VMware Converter.

ESXi

Es una versión especial de Vmware ESX a la que se le ha quitado el 98% de código Red Hat y la Service Console. Se distribuye en una memoria flash ROM integrada en el hardware, de modo que no precisa de instalación, aunque también existe una versión instalable en disco duro.

Simplemente se configuran unos pocos parámetros, como si fuera la BIOS del equipo y se añade al entorno Virtual Center para su gestión como Servidor Stand Alone. Los servidores, por tanto estarán completamente certificados para su correcta ejecución. Ya hay servidores disponibles de HP, Dell, IBM, Fujitsu-Siemens y otros. Desde julio de 2008 es un producto gratuito.

4. Análisis y desarrollo del sistema

En esta sección se va a proceder al análisis de los requisitos específicos del proyecto, se plantearán diferentes soluciones en base a estos requisitos y posteriormente se detallará la arquitectura y la construcción del sistema. Para estructurar estos pasos emplearemos una versión resumida y adaptada de la metodología Metrica V3.

Dividiremos el análisis y desarrollo del sistema en dos grandes bloques, en el primer bloque llamado “Planificación de sistemas de información” analizaremos los requisitos específicos del proyecto, las deficiencias de la situación actual del sistema y las mejoras que se quieren obtener a la consecución del proyecto.

En la segunda fase, llamada “Desarrollo de sistemas de información” plantearemos las posibles soluciones que se pueden adoptar en base a los requisitos planteados en la primera fase, diseñaremos la arquitectura a construir, detallaremos la construcción y así mismo estudiaremos el funcionamiento y las posibles deficiencias del nuevo sistema.

4.1 Planificación de sistemas de información (PSI)

Para la construcción del sistema emplearemos una versión resumida de la metodología **Métrica v3** adaptándola a los requisitos específicos del proyecto.

Métrica es una metodología de planificación, desarrollo y mantenimiento de sistemas de información. Promovida por el Ministerio de Administraciones Públicas del Gobierno de España para la sistematización de actividades del ciclo de vida de los proyectos software en el ámbito de las administraciones públicas. Esta metodología propia está basada en el modelo de procesos del ciclo de vida de desarrollo ISO/IEC 2007 (Information Technology - Software Life Cycle Processes) así como en la norma ISO/IEC 15504 SPICE (Software Process Improvement And Assurance Standards Capability Determination)

Podemos resumir los objetivos de Metrica v3 en los siguientes puntos:

- Proporcionar o definir sistemas de información que ayuden a conseguir los fines de la organización.
- Dotar a la organización de productos software que satisfagan las necesidades de los usuarios.
- Mejorar la productividad de los departamentos de Sistemas y TIC.

- Facilitar la comunicación y entendimiento entre los distintos participantes en la producción de software a lo largo del ciclo de vida del proyecto
- Facilitar la operación, mantenimiento y uso de los productos software obtenidos.

Métrica v3 descompone cada uno de los procesos en actividades, y éstas a su vez en tareas. Para cada tarea se describe su contenido (Principales acciones, productos, técnicas, prácticas y participantes)

Los procesos principales son los siguientes:

- **Planificación de sistemas de información (PSI):** Obtención de un marco de referencia para el desarrollo de SI que responda a los objetivos estratégicos de la organización.
- **Desarrollo de sistemas de información.**
- **Mantenimiento de sistemas de información:** Obtención de una nueva versión de un SI desarrollado con MÉTRICA v. 3n 3 ó 2, a partir de las peticiones de mantenimiento que los usuarios realizan con motivo de un problema detectado en el sistema, o por la necesidad de una mejora del mismo.

En este proyecto no vamos a abarcar todas las fases, ya que el mantenimiento queda fuera del objetivo del mismo, por tanto sólamente se realizarán las dos primeras: Planificación de sistemas de información y Desarrollo de sistemas de información.

Así mismo dividiremos el desarrollo del sistema en los siguientes subprocesos:

- **Estudio de Viabilidad del Sistema (EVS)**: Análisis de un conjunto concreto de necesidades para proponer una solución a corto plazo, que tenga en cuenta restricciones económicas, técnicas, legales y operativas.
- **Análisis del Sistema de Información (ASI)**: Obtención de una especificación detallada del SI que satisfaga las necesidades de información de los usuarios y sirva de base para el posterior diseño del sistema.
- **Diseño del Sistema de Información (DSI)**: Definición de la arquitectura del sistema y del entorno tecnológico que le va a dar soporte, junto con la especificación detallada de los componentes del sistema de información.
- **Construcción del Sistema de Información (CSI)**: Se genera el código de los componentes del SI, se desarrollan todos los procedimientos de operación y seguridad y se elaboran todos los manuales de usuario final y de explotación con el objetivo de asegurar el correcto funcionamiento del sistema para su posterior implantación.
- **Implantación y aceptación del sistema (ASI)**: Entrega y aceptación del sistema en su totalidad, y la realización de todas las actividades necesarias para el paso a producción del mismo.

4.2 Desarrollo de sistemas de información (DSI)

4.2.1 Estudio de viabilidad del sistema

Hemos realizado un primer planteamiento de los requisitos del proyecto en la presentación de esta memoria, vamos a analizar de forma más detallada el conjunto de requisitos y necesidades del proyecto.

La asignatura ADS/ASO realiza un uso intensivo de diferentes sistemas operativos para el desarrollo de las sesiones prácticas, los alumnos se dividen en grupos de 4 personas, cada grupo dispone de 2 terminales a fin de realizar las prácticas.

Estos terminales están configurados de la forma siguiente:

- El primer terminal dispone de 2 máquinas virtuales, una de ellas actúa como servidor Linux y la restante como cliente Windows.
- El segundo terminal igualmente dispone de 2 máquinas virtuales, la primera como cliente Linux y la segunda como servidor Windows.

Estos dos terminales interactúan entre si durante el desarrollo de las sesiones prácticas, es por ello que se necesita de una configuración inicial al principio del curso a fin de poder crear esta infraestructura.

Para esta configuración inicial que tiene como objetivo preparar los terminales e instalar las máquinas virtuales se hace uso de una serie de scripts a fin de automatizar estas tareas. A modo ilustrativo se van a presentar unos cuantos de estos scripts para ejemplificar mejor esta fase del proceso:

El siguiente script realiza la configuración inicial al principio del curso y prepara los discos:

```
# -*- coding: UTF-8 -*-

## Prerequisitos de la instalación de partida:
##     Centos 5
##     Vmware Workstation
##     Cortafuegos Desactivado
##     Partición de máquinas virtuales formateada


## Este programa se ejecuta desde "admon.sh"
## Es el punto de entrada de la configuración automática.




import siempre
siempre.configura ()

#Descomenta la ejecucion de inicio_curso solo al inicio del cuatrimestre
#para la preparacion del laboratorio

#import inicio_curso
#inicio_curso.configura ()

import prepara_disco2
prepara_disco2.configura ()
```

Podemos observar que se utiliza el script “inicio_curso”, este fichero se encarga de preparar la configuración inicial que tienen las máquinas cuando se inicia el curso, borrando las máquinas del curso anterior y verificando la consistencia.

Además de las configuraciones iniciales que se realizan al inicio del curso o del cuatrimestre, siempre que se enciende el anfitrión se deben realizar ciertas configuraciones, estas configuraciones se realizan mediante el siguiente script:

```
def configura ():

    # MONTAMOS SEGUNDO DISCO

    ejecuta_sin_control_error ("umount " + dispositivo_disco2)
    ejecuta_sin_control_error ("mkdir " + dir_disco2)

    ejecuta_con_control_error ("mount " + dispositivo_disco2 + " " +
dir_disco2,
                               "No se ha podido montar " + dispositivo_dis-
co2 + " en " + dir_disco2)


    # PREPARAMOS AL ENRUTADOR

    if es_enrutador:

        ejecuta_con_control_error ("iptables -t nat -A POSTROUTING -o eth0
-j MASQUERADE",
                                   "Problemas configurando iptables")

        ejecuta_con_control_error ("echo 1 > /proc/sys/net/ipv4/ip_for-
ward",
                                   "Problemas configurando forwarding")

        ejecuta_con_control_error ("ifconfig eth0:0 10.1." + str(id_domi-
nio) + ".250 netmask 255.255.255.0",
                                   "Problemas activando la tarjeta virtual
eth0:0")



    # CLIENTE DNS

    f = open ("/etc/resolv.conf", "w")
    f.write ("nameserver 158.42.179.9\n")
    f.close()
```

```
   # CONTROL DE ACCESO


    ejecuta_con_control_error ("authconfig --enablepamaccess --update",
                               "Problemas activando pam_access")


    f = open ("/etc/security/access.conf", "w")

    f.write ("+ : root : ALL\n")
    for t in turnos:
        grupo_unix = t.nombre + "_" + "admon" + str (id_dominio).zfill(2)
        f.write ("+ : " + grupo_unix + " : ALL\n")
    f.write ("- : ALL : ALL\n")
    f.close()


    # integración en el dominio windows 2003 del dsic

    ejecuta_con_control_error ("authconfig --enablekrb5
--krb5kdc=spiderman.dsic.upv.es  --krb5adminserver=spiderman.dsic.upv.es
--krb5realm=DSIC.UPV.ES --update",
                               "Problemas configurando kerberos")

    ejecuta_con_control_error ("cp ldap.conf /etc/ldap.conf",
                               "Problemas copiando ldap.conf")

    ejecuta_con_control_error ('authconfig --enableldap
--ldapserver="ldap://spiderman.dsic.upv.es/,ldap://scar.dsic.upv.es/,ldap:/
/hulk.dsic.upv.es/" --ldapbasedn="dc=dsic,dc=upv,dc=es" --update',
                               "Problemas configurando ldap")


    # activamos la creación de directorios de conexión

    ejecuta_con_control_error ("authconfig --enablemkhomedir --update",
                               "Problemas configurando pam_mkhomedir")

    for d in dirs_conexion:

        if not os.access (d, os.F_OK):
            ejecuta_con_control_error ("mkdir -p " + d,
                                       "Problemas al crear " +  d)
```

```
    # copiamos el script activa-vm a /usr/bin

    ejecuta_con_control_error ("cp activa-mv /usr/bin",
                               "Problemas copiando ldap.conf")


    ejecuta_sin_control_error ("chmod a+rx /usr/bin/activa-vm")


    # desactivamos actualizaciones automáticas

    ejecuta_con_control_error ("chkconfig yum-updatesd off",
                               "Problemas desactivando actualizaciones au-
tomaticas")

    ejecuta_con_control_error ("/etc/init.d/yum-updatesd stop",
                               "Problemas parando actualizaciones automati-
cas")


    # Preparamos repositorio dsic

    ejecuta_con_control_error ("cp CentOS-Base.repo /etc/yum.repos.d",
                               "Problemas copiando CentOS-Base.repo")

    # Permisos vmware-workstation

    ejecuta_con_control_error ("chmod 700 /usr/bin/vmware",
                               "Problemas cambiando permisos a /usr/bin/
vmware")


    # hosts.allow y deny

    f = open ("/etc/hosts.deny", "w")
    f.write ("ALL:ALL\n")
    f.close()

    f = open ("/etc/hosts.allow", "w")
    f.write ("ssh : 158.42.\n")
    f.close()
```

También es necesario por ejemplo realizar la configuración del LDAP, el siguiente fichero contiene la configuración necesaria para que LDAP funcione correctamente:

```
deref never
referrals no
ldap_version 3

binddn cn=usuproxy,ou=especiales,ou=depto,ou=dsic,dc=dsic,dc=upv,dc=es
bindpw quemalaleche

bind_timelimit 1

nss_base_passwd  ou=dsic,dc=dsic,dc=upv,dc=es?sub
nss_base_group ou=admon,ou=labs,ou=dsic,dc=dsic,dc=upv,dc=es?sub
nss_base_group ou=Grupos,dc=dsic,dc=upv,dc=es?sub
nss_map_objectclass posixAccount User
nss_map_objectclass shadowAccount User
nss_map_attribute uid sAMAccountName
nss_map_attribute uniqueMember member
nss_map_attribute homeDirectory msSFUHomeDirectory
nss_map_objectclass posixGroup Group
pam_login_attribute sAMAccountName
pam_filter objectclass=user
pam_password md5
pam_member_attribute member
ssl no
tls_cacertdir /etc/openldap/cacerts

uri ldap://spiderman.dsic.upv.es/ ldap://scar.dsic.upv.es/
ldap://hulk.dsic.upv.es/
base dc=dsic,dc=upv,dc=es
```

Como vemos, el número de configuraciones que se deben realizar a fin de preparar el laboratorio es grande. Estas configuraciones, a pesar de estar bastante automatizadas requieren de un tiempo considerable de preparación al inicio del curso. Además la infraestructura desplegada queda ligada a un laboratorio, y por tanto no es posible utilizar otro laboratorio, lo que resta bastante flexibilidad a la asignatura.

Además, los alumnos no pueden realizar prácticas fuera del horario asignado, ya sea para estudiar o repasar conceptos, ya que no se puede utilizar la infraestructura fuera del laboratorio.

El hecho de que cada grupo de alumnos posea una infraestructura propia, hace que el sistema sea menos tolerante a fallos y que el mantenimiento sea más complicado, siendo difícil responder de forma rápida ante posibles eventos que puedan complicar la realización de las prácticas.

4.2.2 Análisis del sistema de información

Todos los factores anteriormente citados nos sirven para establecer una serie de requisitos que debe satisfacer el sistema de información. Estos requisitos conformarán las características principales de la plataforma a desarrollar:

- **Se debe reducir el tiempo de despliegue de la infraestructura:** Cómo ya hemos dicho, las configuraciones iniciales son bastante costosas, la nueva solución debe minimizar el tiempo de despliegue inicial así como simplificarlo.
- **El proceso de mantenimiento y administración debe simplificarse:** Cuanto más se simplifique el proceso de mantenimiento y administración se tendrán que destinar menos recursos para estos procesos y la asignatura tendrá un coste menor.
- **La infraestructura debe poder se utilizada desde cualquier lugar, eliminando de esta forma la dependencia de un laboratorio concreto:** Este es uno de los principales requisitos, actualmente los alumnos no pueden disponer de los recursos de las prácticas de la asignatura y por tanto no tienen posibilidad de realizar prácticas fuera del horario asignado.
- **Se debe garantizar la seguridad y la independencia entre los diferentes grupos:** Cada grupo debe ser independiente del otro, por tanto no se deben compartir datos

privados entre grupos y debe garantizarse la seguridad a fin de evitar posibles manipulaciones de datos y acceso a información entre diferentes grupos.

- **Debe ser posible el acceso por roles a la infraestructura:** A fin de poder garantizar la seguridad y la independencia de los grupos, podemos hacerlo mediante un acceso por roles y restringiendo el control de las máquinas por parte de los usuarios a nivel de dominios.
- **Se deben emplear soluciones open source a fin de reducir el coste del proyecto:** Puesto que no se dispone de presupuesto para licencias, es fundamental el hecho de encontrar una solución open source que cumpla los requisitos.
- **El sistema debe contemplar soluciones de alta disponibilidad para los puntos críticos de la infraestructura:** Mediante soluciones de alta disponibilidad podemos garantizar una rápida respuesta ante determinados eventos que puedan comprometer el sistema.

Teniendo claros los requisitos de la solución e incidiendo en el hecho de que la solución a adoptar debe ser open source, el siguiente paso es realizar un análisis de las posibles soluciones de Cloud Computing open source a adoptar, esto incluye tanto el hipervisor como la propia plataforma de administración de nuestro Cloud.

4.2.2.1 Análisis de soluciones open source de Cloud Computing

En los últimos años el número de soluciones open source dedicadas a la administración y gestión de entornos de Cloud Computing ha crecido exponencialmente, grandes compañías como Citrix o Red Hat están apostando fuertemente por este tipo de soluciones.

Un licenciamiento gratuito permite reducir los costes de la infraestructura. Pero este hecho debe estar balanceado con los costes de soporte y de desarrollo para personalizar el código de la solución. El licenciamiento en open source normalmente es menos complicado que en soluciones propietarias.

Tradicionalmente el software comercial fue diseñado para entornos estáticos. Por lo tanto, puede ser un pequeño dolor de cabeza como licenciar un entorno dinámico como es la nube.

Adicionalmente, las soluciones open source nos permiten tener un control mayor sobre el las fases de pruebas y evaluación de tecnologías Cloud, por tanto nos permiten crear el entorno ideal para este proyecto.

Las principales soluciones open source actuales son:

- **Eucalyptus**
- **Open Nebula**
- **Open Stack**
- **CloudStack**

Todas ellas soluciones de probada reputación, respaldadas por grandes empresas y/o organizaciones y con un amplio soporte en la comunidad. En los sucesivos apartados detallaremos las características básicas de cada una de ellas.

4.2.2.1.1 Eucalyptus

Eucalyptus es una infraestructura de software de código abierto para implementar Cloud Computing en clusters. Hasta el momento solo sería otra aplicación mas para la creación de clusters sobre los que se ejecutan maquinas virtuales y servicios de almacenamiento, pero Eucalyptus es compatible con la interface de computación en nube de Amazon la EC2, lo que la hace muy llamativa para la realización de pruebas del funcionamiento de EC2 sin necesidad de incurrir en gastos.

También resulta interesante si queremos desplegar una infraestructura local parecida a EC2 y utilizando herramientas creadas para esta; ademas de esta característica Eucalyptus también esta diseñado para soportar otras interfaces cliente. Eucalyptus esta implementado con herramientas básicas Linux y tecnologías de servicios web que lo hacen fácil de instalar y administrar.

Numerosas empresas han empezado a probar Eucalyptus como alternativa a Amazon EC2, las más notables son Rightscale y Elastra. RightScale ha sido uno de los mayores partners de Amazon EC2, y es absolutamente dependiente de ellos. Así que este movimiento va dirigido a no seguir siendo un 'cliente cautivo'.

Eucalyptus implementa nubes de tipo privado e híbrido, de estilo IaaS. La plataforma proporciona una interfaz única que permite al usuario acceso a recursos de infraestructura (máquinas, red y almacenamiento) disponibles en nubes privadas y recursos disponibles externamente en servicios de nube pública.

El software está diseñado con una arquitectura modular y extensible basada en servicios web que permite a Eucalyptus exportar variedad de APIs hacia usuarios vía herramientas cliente. Actualmente, Eucalyptus implementa el API estándar de la industria Amazon Web Services, que permite la interoperabilidad de Eucalyptus con servicios AWS y herramientas.

Eucalyptus proporciona su propio conjunto de herramientas de línea de comandos llamada Euca2ools, que puede utilizarse internamente para interactuar con las intalaciones privadas de Eucalyptus o externamente para material de nubes públicas, incluyendo Amazon EC2.

Eucalyptus incluye las siguientes funciones:

- **Compatibilidad con la API de Amazon Web Services.**
- **Instalación y desarrollo con el útil de gestión de clusters Rock Linux desde código o paquetes DEB y RPM.**
- **Comunicación segura entre los procesos internos vía SOAP y WS-security**
- **Útiles de administración básica**
- **Capacidad de configurar múltiples clústeres de servidores como una sola "cloud"**
- **Soporte para máquinas virtuales Linux y Windows**
- **Direcciones IP elásticas y grupos de seguridad**
- **Gestión de usuarios y grupos**
- **Informes de contabilidad**
- **Políticas programables y configurables**

Eucalyptus funciona de forma similar a la API de gestión de Amazon EC2 pero sobre otro tipo de infraestructura, puede duplicar la funcionalidad de Amazon EC2 a través de sus comandos en línea y su API REST.

Las primeras versiones de Eucalyptus aprovechan un paquete de software de gestión de clusters linux llamado Rocks. En versiones recientes es posible instalarlo directamente sobre máquinas stand-alone, aunque desde Eucalyptus reconocen que a través de Rocks el proceso es más sencillo.

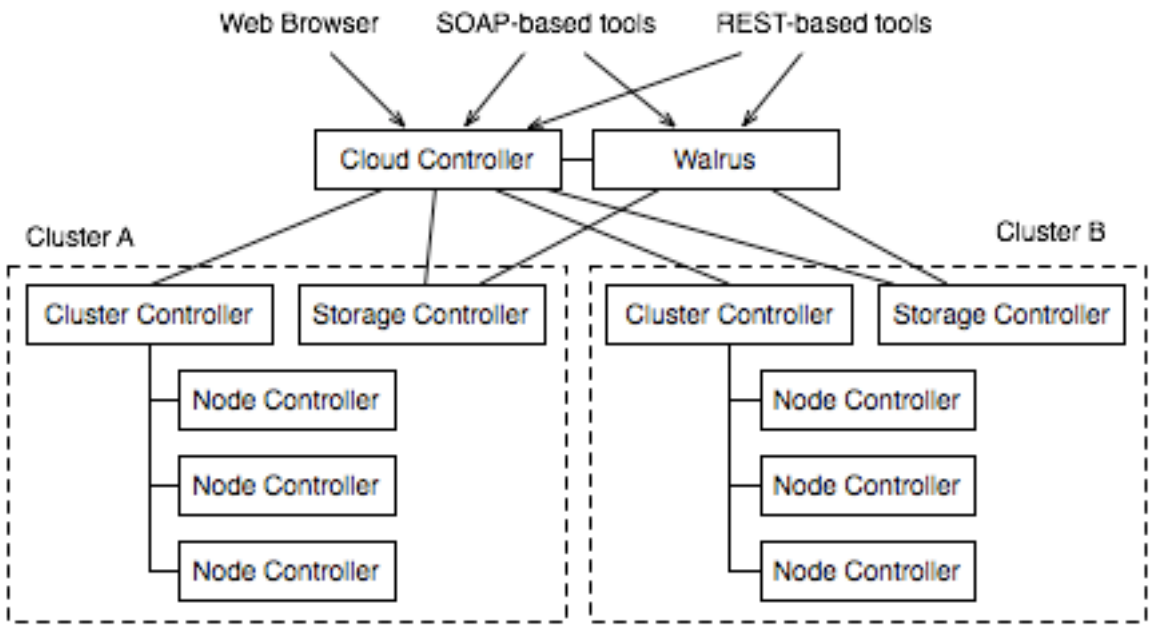

Figura 4.2.2.1.1.1 Arquitectura de Eucalyptus

La nube de plataforma de computación Eucalyptus tiene cuatro componentes de alto nivel:

- **Cloud Controller (CLC)**
- **Cluster Controller (CC)**
- **Walrus, Storage Controller (SC)**
- **Node Controller (NC)**

Cada componente de sistema tiene su propia interfaz web y es implementada como un servicio web stand-alone. Esto tiene dos ventajas principales: Primero, cada servicio Web expone una API bien definida independientemente del idioma (de programación) en la forma de documento WSDL que contiene tanto las operaciones que el servicio puede realizar y las estructuras de datos input/output.

Segundo, Eucalyptus aprovecha funciones Web-service existentes como políticas de seguridad (WSS) para comunicación segura entre componentes y confía en los paquetes de software de servicios web estándar de la industria.

Como tecnología de virtualización se usa Xen en versiones 3.X, pero en principio no hay restricciones sobre utilizar cualquier otra plataforma de virtualización.

Eucalyptus, a pesar de que como hemos visto cumple con la mayor parte de los requisitos de nuestra solución, está orientado a un usuario con grandes conocimientos, la interfaz de administración no está tan desarrollada como otras soluciones, habiendo que realizar gran parte de las tareas de administración a través de la consola Euca2ools.

Por estas razones podemos descartar a Eucalyptus como plataforma para nuestra solución, ya que ni el personal encargado de la administración de la solución, ni los usuarios tienen una formación específica sobre tecnologías de este tipo.

4.2.2.1.2 Open Nebula

OpenNebula es un software open-source que permite construir cualquier tipo de cloud: privado, público e híbrido. Ha sido diseñado para ser integrado con cualquier tipo de red y almacenamiento, para así adaptarse a los centros de datos existentes. Proporciona soporte para distintos hipervisores (Xen, KVM y VMware ESXi).

OpenNebula gestiona el almacenamiento, las redes y las tecnologías de virtualización. Proporciona la posibilidad de desplegar servicios en infraestructuras distribuidas, combinando recursos de centros de datos así como de clouds remotos, de acuerdo con las políticas de despliegue.

OpenNebula emplea en su infraestructura una arquitectura en cluster clásica, con un frontal y con conjunto de nodos donde serán ejecutadas las máquinas virtuales. Al menos, debe haber una red que interconecte todos los nodos con el frontal.

La arquitectura interna de OpenNebula se divide en tres capas:

- **Tools:** Herramientas de gestión desarrolladas empleando las interfaces proporcionadas por el núcleo de OpenNebula
- **Core:** Componentes principales para gestionar las máquinas virtuales, redes virtuales y nodos
- **Drivers:** Proporcionan nuevas tecnologías para la virtualización, el almacenamiento, la monitorización y los servicios de cloud

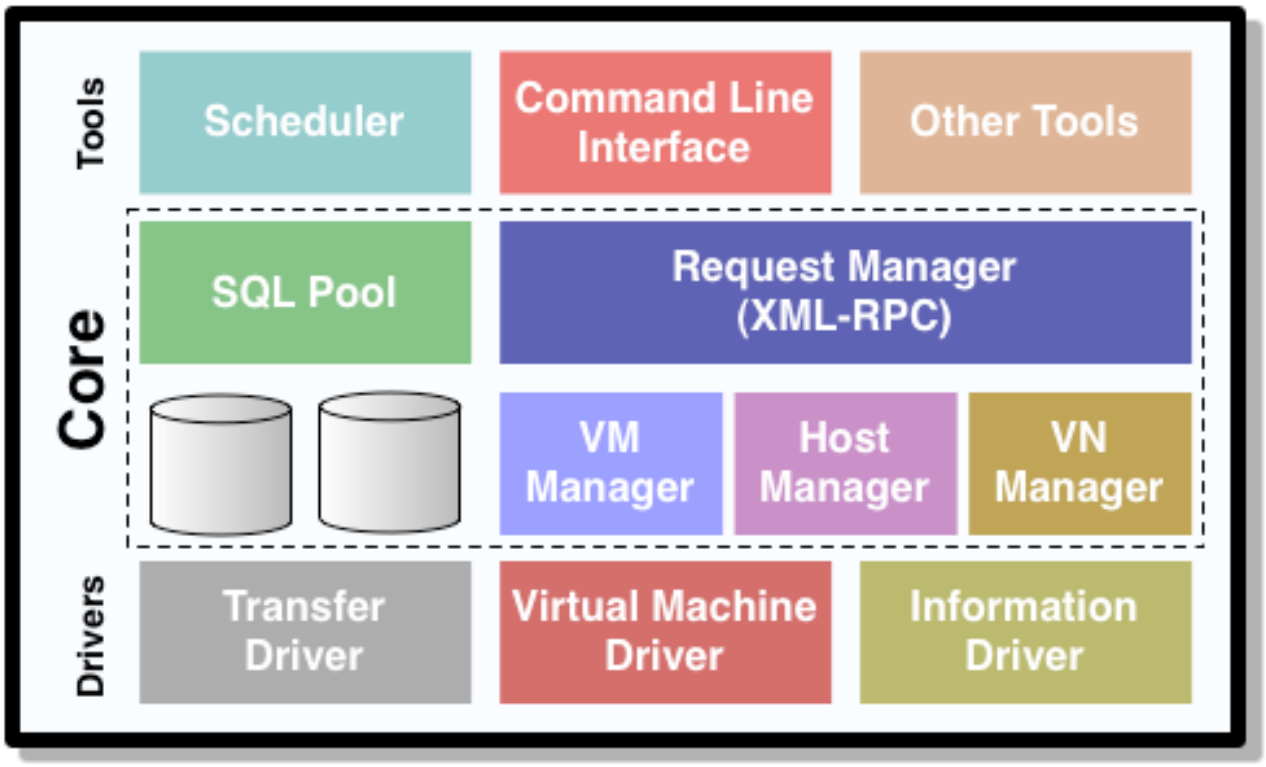

Figura 4.2.2.1.2.1 Componentes de Open Nebula.

OpenNebula emplea en su infraestructura una arquitectura en cluster clásica, con un frontal y con conjunto de nodos donde serán ejecutadas las máquinas virtuales. Al menos, debe haber una red que interconecte todos los nodos con el frontal.

Los componentes de OpenNebula son:

- **Front-end:** Ejecuta OpenNebula y los servicios del cluster
- **Nodos:** Anfitriones que proporcionan los recursos necesarios a las máquinas virtuales, como el software hipervisor
- **Repositorio de imágenes:** Medio que almacena las imágenes virtuales base

- **Demonio OpenNebula:** Gestiona el ciclo de vida de las máquinas virtuales y los subsistemas (red, almacenamiento, hipervisores)
- **Drivers:** Programas empleados por el núcleo de OpenNebula para servir de interfaz para un hipervisor o un sistema de almacenamiento específico

Tipos de usuarios:

- **Oneadmin:** Usuario administrador del cloud privado que gestiona las máquinas virtuales, redes, nodos o usuarios
- **Usuarios no privilegiados:** Gestionan única y exclusivamente sus propios objetos (máquinas virtuales, redes virtuales). Hay que señalar que pueden instanciar aquellas imágenes virtuales del repositorio a las que se les ha establecido el acceso público

Al igual que Eucalyptus, Open Nebula requiere de ciertos conocimientos para una correcta administración, así mismo la interfaz de la plataforma es bastante pobre y tiene poca funcionalidad, habiendo que realizar la mayor parte de tareas de administración a través de la consola, lo cual no tiene porque ser malo, pero en el entorno del proyecto, como hemos dicho, tanto usuarios como administradores no tienen conocimientos específicos sobre herramientas de Cloud Computing, por tanto una mejor interfaz ayudaría a facilitar el uso de la plataforma, es por ello que podemos descartar a Open Nebula como solución.

4.2.2.1.3 Open Stack

El paquete de servicios OpenStack, creado por la fundación con el mismo nombre, es una de las iniciativas de nube abierta que cuenta con más apoyos, entre los que están Dell, Cisco y HP, así como RackSpace. Se trata de intentar echar abajo el dominio de las grandes compañías, como Amazon, que además no facilitan la migración a otras plataformas.

El proyecto OpenStack ha cumplido recientemente un año de vida y sus herramientas ya han sido descargadas 50.000 veces. Se basa en el código abierto y el software libre para proporcionar una serie de servicios como computación en la nube, almacenamiento, redes y gestión del sistema.

OPenStack es uno de los proyectos de nube abierta que más apoyos tiene. Cuenta con el apoyo de la empresa de alojamiento RackSpace, además de con el respaldo de compañías tales como Dell, Cisco y HP.

La intención de los promotores de OpenStack es crear un entorno de cloud computing que se aleje del modelo de grandes compañías como Amazon. En ocasiones no es fácil migrar de un servicio de cloud a otro debido a que la tecnología de cada proveedor no se divulga. OpenStack pretende ser todo lo contrario y por ello ha elegido el modelo de software libre para ello.

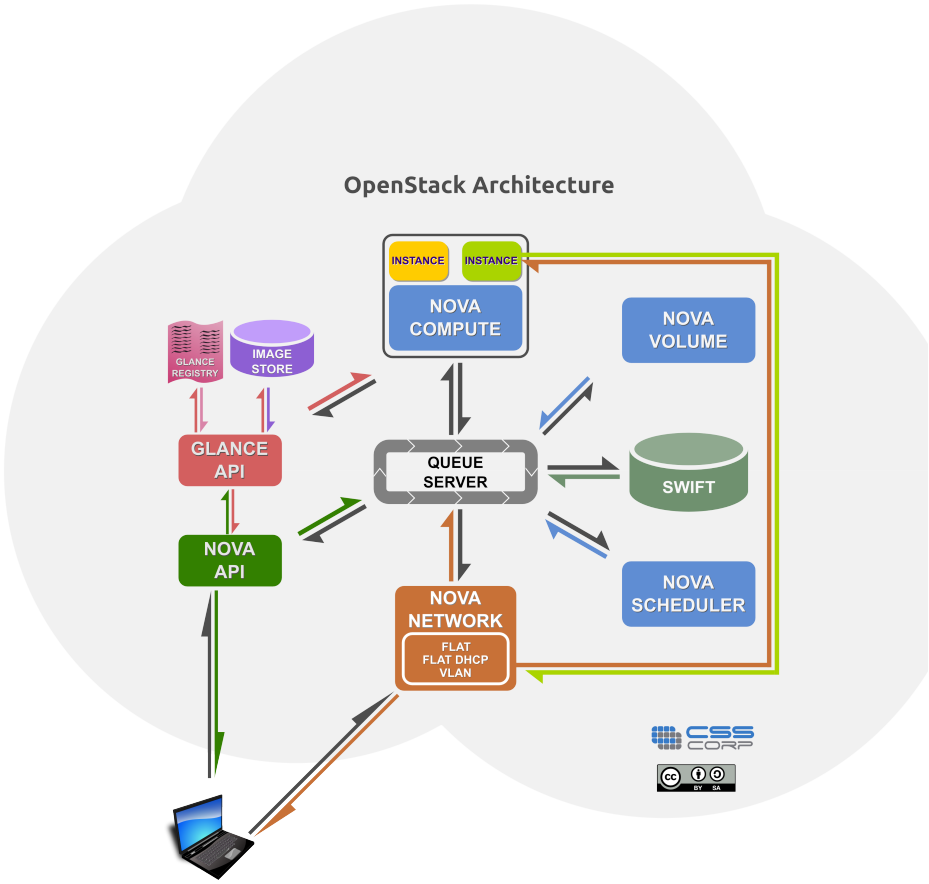

Figura 4.2.2.1.3.1 Arquitectura de Open Stack

Openstack está dividido en tres servicios principales:

1) Infraestructura de procesamiento (Nova)

Nova es el controlador de procesamiento para la nube OpenStack. Todas las actividades necesarias para soportar el ciclo de vida de las instancias de OpenStack son controladas por Nova. Esto hace de Nova una plataforma de administración que administra recursos de procesamiento, red, autorización y necesidades de escalabilidad de la plataforma OpenS-

tack, Nova no provee de ninguna capacidad de virtualización por si misma, en lugar de eso, utiliza las APIs de libvirt para interactuar con los hipervisores soportados.

Nova presenta todas sus capacidades a través de una API de web services que es compatible con la API EC2 de Amazon Web Services.

Funciones y características:

- Control del ciclo de vida de las instancias.
- Control sobre recursos de procesamiento
- Red y autorización
- API basada en REST
- Soporta múltiples hipervisores (Xen, XenServer/XCP, KVM, UML, Vmware Vsphere y Hyper-V)

2) Infraestructura de almacenamiento (Swift)

Swift provee a CloudStack de un almacenamiento distribuido y consistente de objetos virtuales. Es análogo a “Simple Storage Service (S3)” de Amazon Web Services. Swift es capaz de almacenar millones de objetos distribuidos en múltiples nodos. Swift utiliza redundancia por defecto, es extremadamente escalable en términos de tamaño (varios petabytes) y capacidad (número de objetos).

Funciones y características:

- Almacenamiento de un gran número de objetos
- Almacenamiento de objetos de gran tamaño
- Redundancia de datos
- Contenedor de datos para máquinas virtuales y aplicaciones en la nube
- Capacidades para streaming
- Almacenamiento seguro de objetos
- Extrema escalabilidad

3) Infraestructura de imágenes (Glance)

El servicio de imágenes de OpenStack es un sistema de lookup y recuperación de imágenes de máquinas virtuales, puede ser configurado para usar cualquiera de los siguientes backends de almacenamiento:

- OpenStack Object Store para almacenar imágenes
- Almacenamiento directo en S3
- Almacenamiento en S3 con Object Store como intermediario para el acceso a S3.

Funciones y características:

1) Provee de un servicio de imágenes a OpenStack

Pese a que OpenStack tiene un futuro muy prometedor (Grandes empresas están migrando hacia OpenStack) ahora mismo se encuentra en una fase poco madura, por lo que es complicado utilizarla en un entorno de producción, así mismo se echan en falta facilidades de cara al administrador, ya sea en una mejor documentación o mejores interfaces gráficas que faciliten la administración de la plataforma y aumenten sus capacidades. Esta serie de deficiencias hacen que OpenStack quede descartada como solución en nuestro proyecto, pese a que como afirmo, tiene un futuro muy prometedor.

4.2.2.1.4 CloudStack

CloudStack es una arquitectura software open source que permite efectuar el despliegue, la configuración y la gestión de entornos de computación elástica. CloudStack fue desarrollado por Cloud.com y proporciona tres versiones diferentes:

- **CloudStack Community Edition:** Open source, soportado por la comunidad
- **CloudStack Enterprise Edition:** Emplea código open source y código propietario. Fue diseñado para entornos empresariales y se distribuye de forma comercial
- **CloudStack Service Provider Edition:** Emplea código open source y código propietario. Fue diseñado para los proveedores de servicios y se distribuye de forma comercial

CloudStack puede desplegarse en uno o más servidores de gestión de tal forma que se conectarían a una única base de datos MySQL. Opcionalmente, se podrían distribuir las peticiones Web mediante el empleo de gestores de balance de carga. Además, una copia de seguridad de la base de datos del servidor de gestión podría desplegarse empleando la replicación MySQL en un sitio remoto.

La infraestructura de despliegue se basa en la utilización de los siguientes elementos:

1) **Nodos de computación:** Los nodos de computación constituyen el bloque básico para efectuar el escalamiento de la plataforma CloudStack. Se pueden añadir nodos de computación adicionales en cualquier momento para proporcionar mayor capacidad a las máquinas virtuales huésped.
   Los nodos de computación no son visibles para el usuario final y, por tanto, no podrán determinar en qué nodo de computación se ejecutará su máquina virtual.
2) **Pods:** Con los hipervisores KVM, un Pod es una colección de nodos de computación. En la práctica no hay limitación en el número de máquinas que pueden estar asignadas en un Pod.
3) **Zonas de disponibilidad:** Una zona de disponibilidad es una colección de Pods y un almacenamiento secundario que incluirá uno o más switches de capa 3. Las zonas de disponibilidad implican alguna forma de aislamiento físico y redundancia. Son visibles al usuario final. Éste debe seleccionar una zona de disponibilidad para iniciar una máquina virtual.
   Los nodos de computación en la misma zona de disponibilidad son accesibles de forma directa sin la necesidad de atravesar un firewall. Los nodos pertenecientes a diferentes zonas de disponibilidad podrían establecer una comunicación mutua por medio de túneles VPN estáticos.

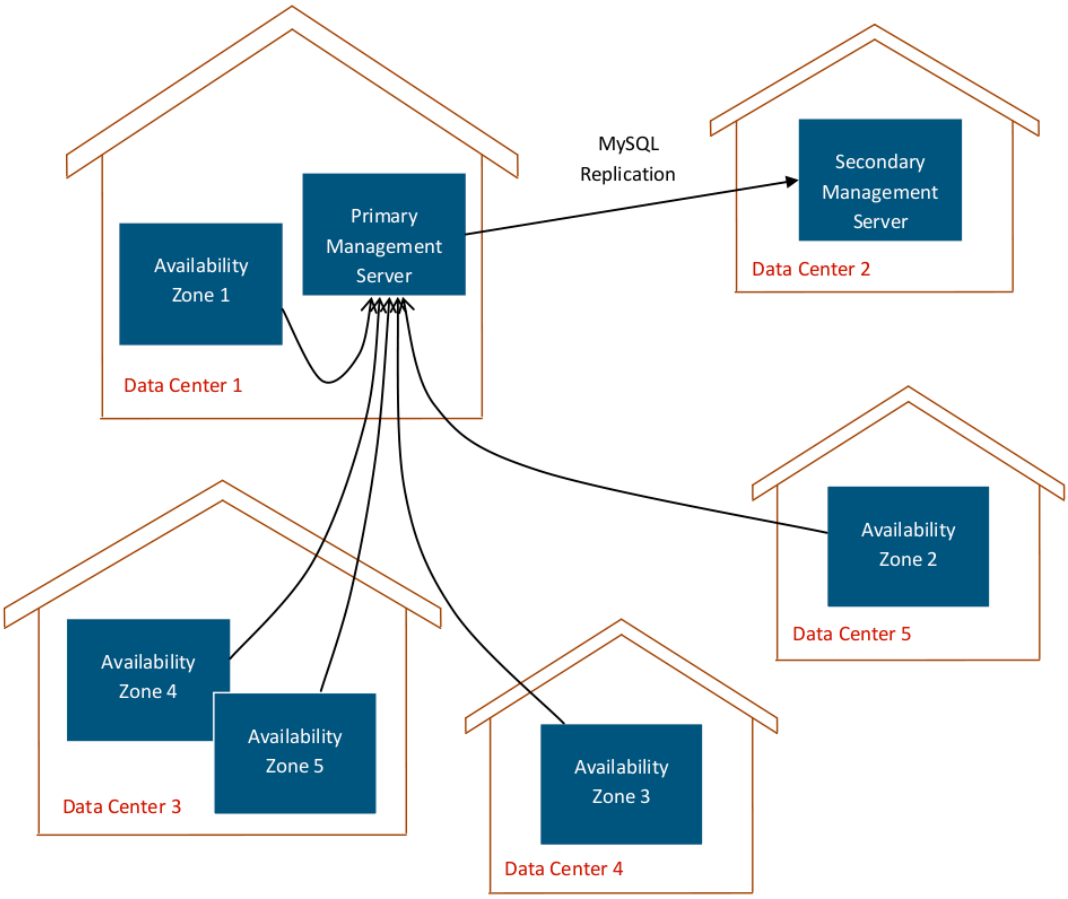

Figura 4.2.2.1.4.1 Ejemplo de infraestructura con CloudStack

El administrador del sistema podrá determinar:

1. Cuántos nodos de computación debe agrupar en un Pod
2. Cuántos servidores primarios de almacenamiento debe ubicar en un Pod

   y la capacidad total de los servidores de almacenamiento
3. Cuántos Pods debe ubicar en una zona de disponibilidad
4. Qué cantidad de almacenamiento secundario debe desplegar en una zona de disponibilidad

Así mismo la plataforma CloudStack proporciona dos tipos de almacenamiento:

- Almacenamiento primario
- Almacenamiento secundario

En el almacenamiento privado puede emplearse iSCSI o NFS. Adicionalmente, podría emplearse el almacenamiento directo. En el almacenamiento secundario siempre se emplea NFS. Hay que señalar que en la plataforma CloudStack, a diferencia de otro software de cloud, todo el almacenamiento es persistente.

Almacenamiento primario

El almacenamiento primario se emplea para almacenar el disco root de las máquinas virtuales, así como los volúmenes adicionales de almacenamiento de datos. El almacenamiento primario (iSCSI o NFS) se registra con el cluster de los nodos de computación. Los volúmenes root se crean de forma automática cuando se crea una máquina virtual. Hay que señalar que también se borran de forma automática cuando una máquina virtual se destruye. Los volúmenes de datos pueden ser creados, conectados y desconectados de forma dinámica a las máquinas virtuales. Los volúmenes de datos no se destruyen cuando se destruye una máquina virtual.

El almacenamiento local es una opción que puede emplearse como forma de almacenamiento primario. Para usarlo en las máquinas virtuales del sistema (como las máquinas virtuales que efectúan la función de router virtual) es necesario establecer *system.vm.use.local.storage* a true en la configuración global de CloudStack.

La plataforma CloudStack permite disponer de múltiples servidores de almacenamiento primario. Una funcionalidad adicional de CloudStack es la posibilidad de definir etiquetas. Una etiqueta es una cadena de texto que se emplea como atributo asociado a un almacenamiento primario, un servicio, o un disco ofertado. Las etiquetas se emplean para identificar los requerimientos de almacenamiento que los servicios ofertados demandan.

Almacenamiento secundario

El almacenamiento secundario se emplea para almacenar plantillas, snapshots de las máquinas virtuales e imágenes ISO. El almacenamiento secundario debe estar localizado en la misma zona de disponibilidad que las máquinas huésped a las que sirve. Debe haber exactamente un dispositivo de almacenamiento secundario por cada zona de disponibilidad.

Plantillas

Una plantilla es una imagen de disco virtual que puede emplearse para instanciar una nueva máquina virtual.

CloudStack distingue dos tipos de plantillas en función de los privilegios de acceso:

- Plantillas públicas: las plantillas públicas están disponibles para todos los usuarios de todas las cuentas. Todos los usuarios pueden crear máquinas virtuales a partir de ellas.
- Plantillas privadas: las plantillas privadas están sólo disponibles para el usuario que las creó. Por defecto, las plantillas subidas a CloudStack son privadas. Los usuarios

pueden crear las máquinas virtuales a partir de su colección de plantillas de la misma forma que crean máquinas a partir de las plantillas públicas.

CloudStack proporciona la posibilidad de subir, publicar o eliminar plantillas:

- **Definir plantillas:** Un usuario puede definir nuevas plantillas que empleará para instanciar las máquinas virtuales. Para utilizar una plantilla, el usuario deberá cargarla en CloudStack especificando, para ello, una URL de la misma forma que si se tratase de una imagen ISO. El protocolo soportado para efectuar la transferencia es HTTP. Una vez indicada la URL, el servidor de gestión de CloudStack efectuará la descarga de la plantilla desde la dirección especificada. Cuando se añade una plantilla, también es necesario especificar el sistema operativo que contiene. Las plantillas son en realidad volúmenes que tienen instalado el sistema operativo huésped y suelen ser archivos de tamaño considerable, por lo que podrían comprimirse mediante gzip para reducir el tiempo de carga en CloudStack. Hay que señalar que en la versión Community Edition las plantillas deben estar en el formato de disco imagen QCOW.
- **Publicar plantillas:** Un usuario puede publicar una plantilla para que esté disponible para otro usuario. En este caso, la plantilla está disponible para ambos usuarios, pero no para el resto.
- **Eliminar plantillas:** Cuando se elimina una plantilla, las máquinas virtuales instanciadas a partir de la misma continúan en ejecución. Sin embargo, no pueden crearse nuevas máquinas virtuales a partir de ella.

Maquinas virtuales

La plataforma CloudStack emplea varios tipos de máquinas virtuales para efectuar las tareas en el cloud, que son:

- **Router virtual**: Un router virtual es una máquina virtual especial que se ejecuta en los nodos de computación. Cada router virtual tiene tres interfaces de red. Su interfaz eth0 sirve como puerta de enlace para las redes virtuales y tiene la dirección IP 10.1.1.1. La interfaz eth1 reside en la red local y se emplea para configurar el router virtual. La interfaz eth2 tiene asignada una dirección IP de la red pública (red que permite el acceso a Internet).

  El router virtual proporciona el servicio DHCP que proveerá direcciones IP a las máquinas virtuales huésped en la red 10.0.0.0/8. El usuario puede reconfigurar manualmente las máquinas virtuales para emplear direcciones IP diferentes.

  El router virtual configura de forma automática NAT para el tráfico saliente de todas las máquinas huésped. El usuario no tiene acceso directo al mismo. Puede efectuar ping y establecer el redireccionamiento de puertos, pero no tiene acceso SSH al mismo. Tampoco existe un mecanismo para que el administrador pueda acceder al router virtual. Sin embargo, puede reinciarlo o detener su funcionamiento.

- **Máquina virtual de usuario**: Son máquinas virtuales convencionales. CloudStack permite iniciar, reiniciar, apagar y eliminar máquinas virtuales. Un usuario sólo puede administrar las máquinas virtuales que pertenecen a su cuenta asociada. Sin embargo, el administrador tiene permisos para gestionar todas las máquinas virtuales de su dominio.

- **Máquinas virtuales vacías:** Los usuarios de CloudStack pueden crear máquinas virtuales vacías, que son máquinas que no tienen asociada una plantilla referente a un sistema operativo. El usuario puede adjuntar una imagen ISO, que es un archivo de

solo lectura perteneciente al tipo ISO/CDROM, e instalar el sistema operativo desde el CD/DVD- ROM, como si te tratase de un ordenador más.

Los usuarios de CloudStack pueden subir sus propias imágenes ISO y montarlas en sus máquinas huésped. Hay que señalar que para subir una imagen ISO es necesario disponer de un servidor WEB, ya que es necesario emplear el protocolo HTTP, y especificar la URL en la que se encuentra almacenada.

Ciclo de vida

La plataforma CloudStack proporciona a los administradores control completo sobre el ciclo de vida de las máquinas virtuales que se están ejecutando en el cloud. Las máquinas virtuales pueden estar en alguno de los siguientes estados o transiciones:

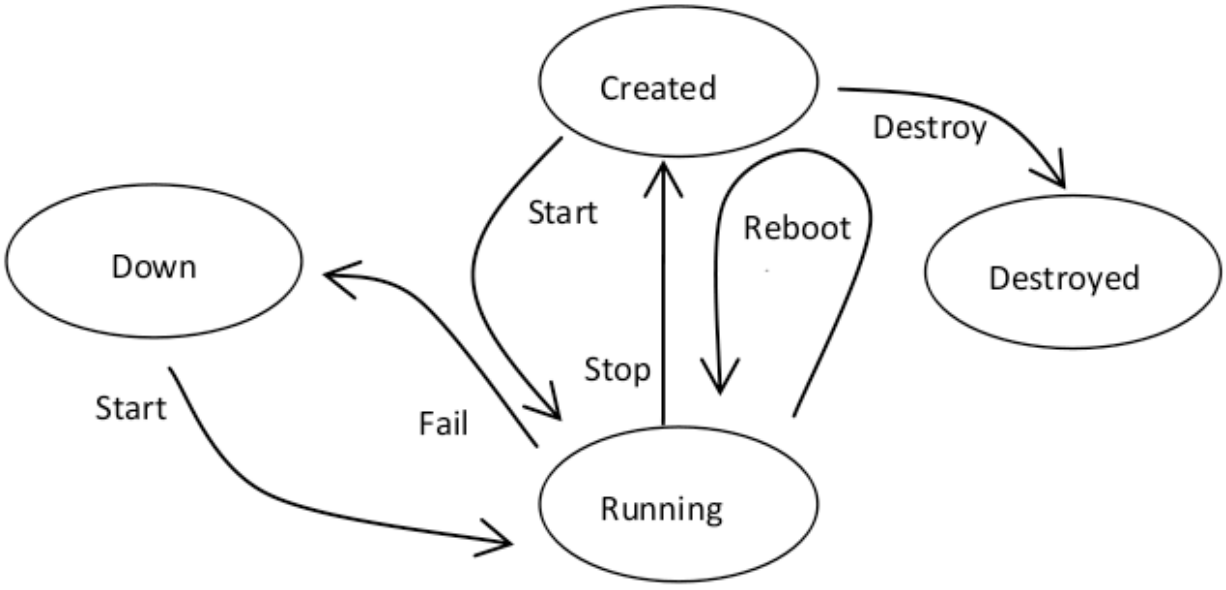

Figura 4.2.2.1.4.2 Ciclo de vida de las máquinas virtuales en CloudStack

4.2.2.1.5. Valoración de las plataformas de Cloud Computing

Fijándonos en los requisitos del proyecto podemos observar que CloudStack es la plataforma que mejor cumple los mismos:

CloudStack posee una potente GUI desde la que se puede gestionar casi cualquier aspecto de la infraestructura, es muy configurable y fácil de utilizar, por lo que la barrera que suponen los conocimientos necesarios para el uso de estas soluciones desaparece en parte con CloudStack, esto nos permite entre otras cosas reducir el tiempo de despliegue de la plataforma y facilitar el proceso de administración y mantenimiento **(Requisitos 1 y 2)**

Es posible el acceso a la plataforma desde virtualmente cualquier dispositivo con acceso a un navegador web y una conexión a Internet **(Requisito 3)**.

Además, se permite organizar la infraestructura en dominios, esta característica unida a que se permite el acceso por roles a la infraestructura desde la GUI, en el cual disponemos de dos roles, uno de “Administrador” y otro de “Usuario”. Nos permite organizar nuestra infraestructura en dominios, que serán los grupos de cada aula.

Por tanto, teniendo que podemos asignar un dominio con su correspondiente administrador para cada grupo cumplimos uno de los principales que la plataforma debe tener; Cada grupo puede ser administrado únicamente por los integrantes del grupo (y el administrador/es root) así mismo, ningún usuario puede administrar grupos ajenos a su dominio, por tanto queda garantizada la seguridad y la independencia entre los grupos **(Requisito 4)**.

Valorando todas estas características tenemos que CloudStack es la plataforma que mejor se adapta a nuestros requisitos y por tanto será la que utilizaremos.

4.2.4. Construcción del sistema de información

Durante los siguientes apartados detallaremos la construcción de nuestra plataforma, desde la preparación del sistema operativo, pasando por la instalación de los diferentes componentes, hasta la puesta en marcha del sistema.

4.2.4.1 Instalación de Cloudstack

Cómo hemos indicado en la descripción de CloudStack, la plataforma está compuesta básicamente por nodos de gestión, nodos de computación y nodos de almacenamiento, los requisitos mínimos para los mismos son los siguientes:

	Descripción	Requisitos mínimos.
Servidor de gestión.	Aloja el software de gestión CloudStack.	CPU 64-bit x86 2 GB de memoria 80 GB de disco Al menos 1 tarjeta de red. (Gigabit Ethernet) 64-bit RHEL/CentOS 5.4+, RHEL6, Fedora 14 o Ubuntu 10.04 LTS Direcciones IP estáticas
Hosts virtualizados	Proporciona todos los recursos de CPU y memoria para las máquinas virtuales.	CPU 64-bit x86 Soporte para virtualización por hardware 4 GB de memoria 30 GB de disco duro local. Al menos 1 tarjeta de red. (Gigabit Ethernet) Direcciones IP estáticas. Citrix XenServer 5.6, RHEL/CentOS 5.6 (64-bit), Fedora 14 (64-bit), RHEL6 (64-bit) o Ubuntu 10.04 LTS (64-bit)
Almacenamiento secundario	Proporciona almacenamiento para las plantillas y snapshots.	Servidor NFS. 100GB de capacidad como mínimo.
Nodo de base de datos		Puede ser dispuesto junto el servidor de gestión. Si se aloja de forma separada, los requisitos son los mismos que los del servidor de gestión.

Cómo vemos los requisitos son bastante elevados, no obstante parece algo lógico ya que el alcance y las posibilidades de la plataforma son muy elevadas.

4.2.4.1.1 Preparación del SO

La plataforma CloudStack en su versión actual (2.2.12) dispone de paquetes binarios de instalación para las siguientes distribuciones de Linux, toda ellas en su versión de 64 bits:

- Red Hat Enterprise Linux 5 / CentOS 5
- Red Hat Enterprise Linux 6.0 / CentOS 6.0
- Red Hat Enterprise Linux 6.1 / CentOS 6.1
- Ubuntu 10.04
- Fedora 14

En el entorno de pruebas de nuestro proyecto vamos a utilizar Ubuntu 10.04, tanto para el servidor de gestión, como para los host y servidores NFS. Primero vamos a instalar y configurar el servidor de gestión, luego añadiremos los nodos así como los servidores NFS que harán de almacenamiento de la solución.

Cloudstack necesita el paquete mysql-server para poder administrar usuarios, contraseñas y demás datos, por tanto realizamos la instalación mediante las siguientes órdenes:

```
sudo -s

apt-get install mysql-server
```

Una vez instalado mysql-server, debemos realizar una pequeña configuración en mysql a fin de evitar en la medida de lo posible futuros bloqueos una vez la plataforma esté en producción:

```
vi /etc/mysql/my.cnf

Y añadiremos las siguientes líneas:

innodb_rollback_on_timeout=1

innodb_lock_wait_timeout=600
```

4.2.4.1.2 Instalación del servidor de gestión

El siguiente paso es modificar el repositorio de Ubuntu para poder obtener los paquetes de instalación:

```
vi /etc/apt/sources.list


Añadimos:

deb http://download.cloud.com/apt/ubuntu/stable/oss ./
```

Actualizamos la lista de paquetes locales:

```
aptitude update
```

```
aptitude install cloud-console-proxy
```

Instalamos el paquete cloud-client:

```
aptitude install cloud-client
```

A continuación modificamos el archivo /etc/hosts para evitar que el loopback se resuelva por medio de ipv6, esto es debido a que CloudStack por el momento no soporta ipv6.

```
Vi /etc/hosts
Eliminar localhost de la linea:
::1 localhost ip6-localhost ip6-loopback
```

Una vez realizado esto tenemos que configurar la base de datos, para ello debemos ejecutar la siguiente órden:

```
cloud-setup-databases cloud:<dbpassword> kvm --deploy-
as=root:<rootpassword>
```

En este momento CloudStack se conectará a la base de datos empleando el usuario cloud. Típicamente se realiza el despliegue como usuario root. <rootpassword> debe ser la contraseña de administración de MySQL (La misma que se definió en la instalación de MySQL.)

Configuramos el servidor de gestión y el console proxy:

```
cloud-setup-management
cloud-setup-console-proxy
```

Durante el proceso de instalación deberemos indicar la ip del servidor de gestión. Llegados a este punto CloudStack se encuentra instalado en el sistema y podemos pasar a la configuración del mismo.

4.2.4.1.3 Configuración del servidor de gestión

En esta sección se indican los pasos necesarios para efectuar la configuración de CloudStack. Para ello, será necesario configurar zonas, pods, almacenamiento y red.

1. **Configuración de la zona**

    1. Acceder al interfaz Web de gestión de CloudStack en: http://management-server-ip-address:8080/client
    2. Autenticarse en CloudStack como administrador, el usuario por defecto es admin y la contraseña es password
    3. Seleccionar “Zones” en “Configuration”
    4. Añadir Zona. Presionar “Add a Zone”
    5. Para añadir una zona es necesario especificar:
        1. **Name:** Nombre de la zona
        2. **DNS 1:** Servidor DNS primario donde se resuelven las direcciones de Internet
        3. **DNS 2:** Servidor DNS secundario donde se resuelven las direcciones de Internet

4. **Internal DNS 1:** Servidor DNS primario a emplear en la red privada que tendrá como misión resolver direcciones de la red interna
5. **Internal DNS 2:** Servidor DNS secundario a emplear en la red privada que tendrá como misión resolver direcciones de la red interna
6. **Guest CIDR:** Rango de direcciones privadas que emplearán las máquinas virtuales en la zona

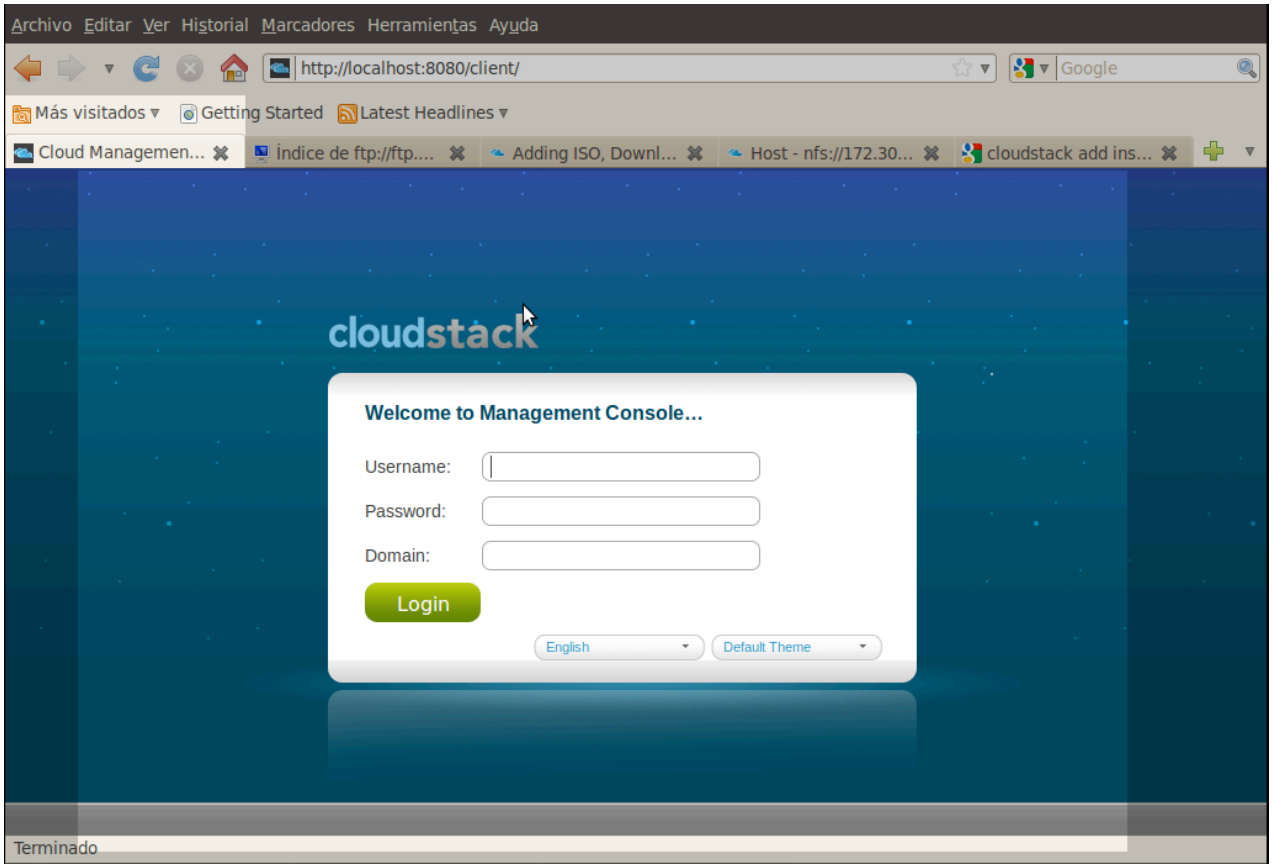

Figura 4.2.4.1.3.1 Consola de login de CloudStack

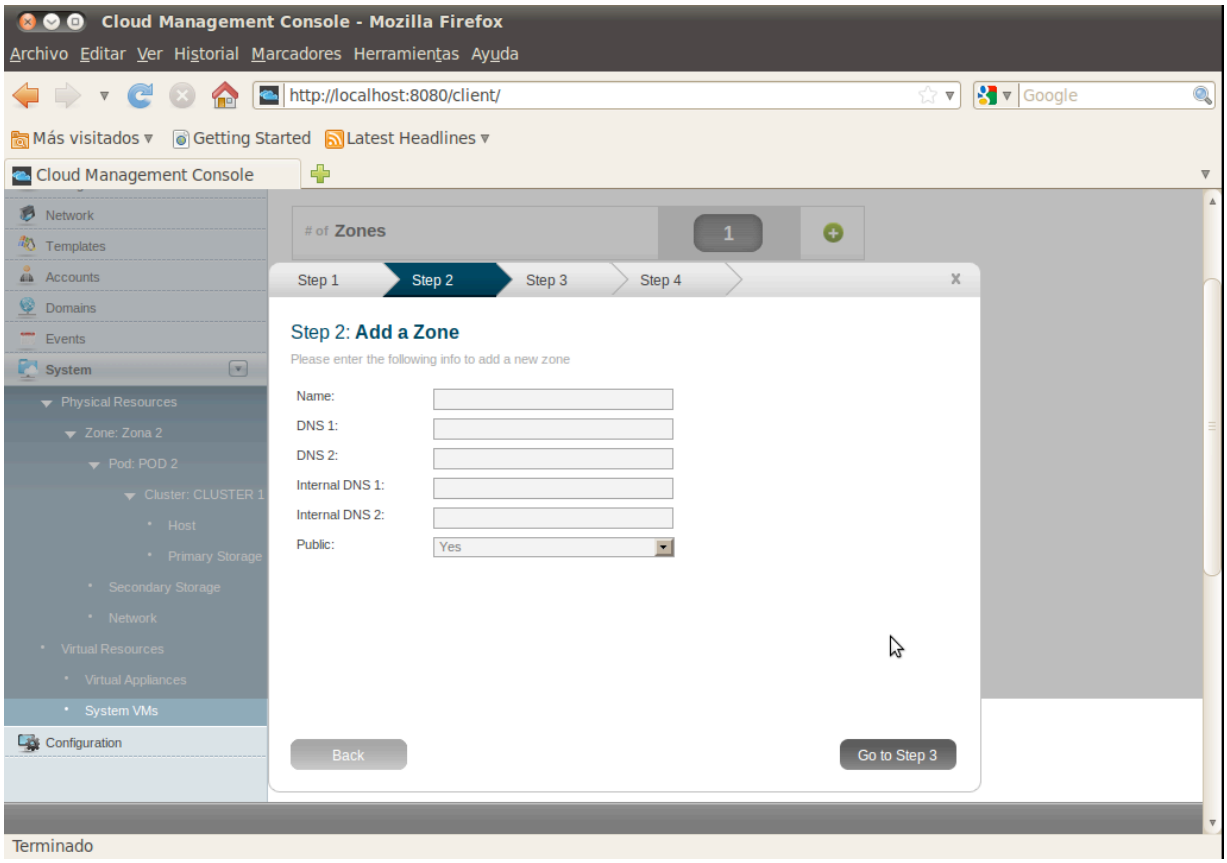

Figura 4.2.4.1.3.2 Formulario para añadir una nueva zona

2. Configurar PODS

1. Acceder al interfaz Web de gestión de CloudStack en: http://management-server-ip-address:8080/client
2. Autenticarse en CloudStack como administrador (El usuario por defecto es admin y la contraseña es password)
3. Seleccionar “Zones” en “Configuration”
4. Seleccionar la zona a la que se le añadirá el pod
5. Presionar “Add Pod”

6. Para añadir un pod es necesario especificar:

   - **Name:** Nombre del pod
   - **Gateway:** Puerta de enlace para el pod
   - **CIDR:** Red a la que pertenece el pod
   - **Private IP Range:** Rango de direcciones IP que emplean las máquinas físicas que forman parte del pod

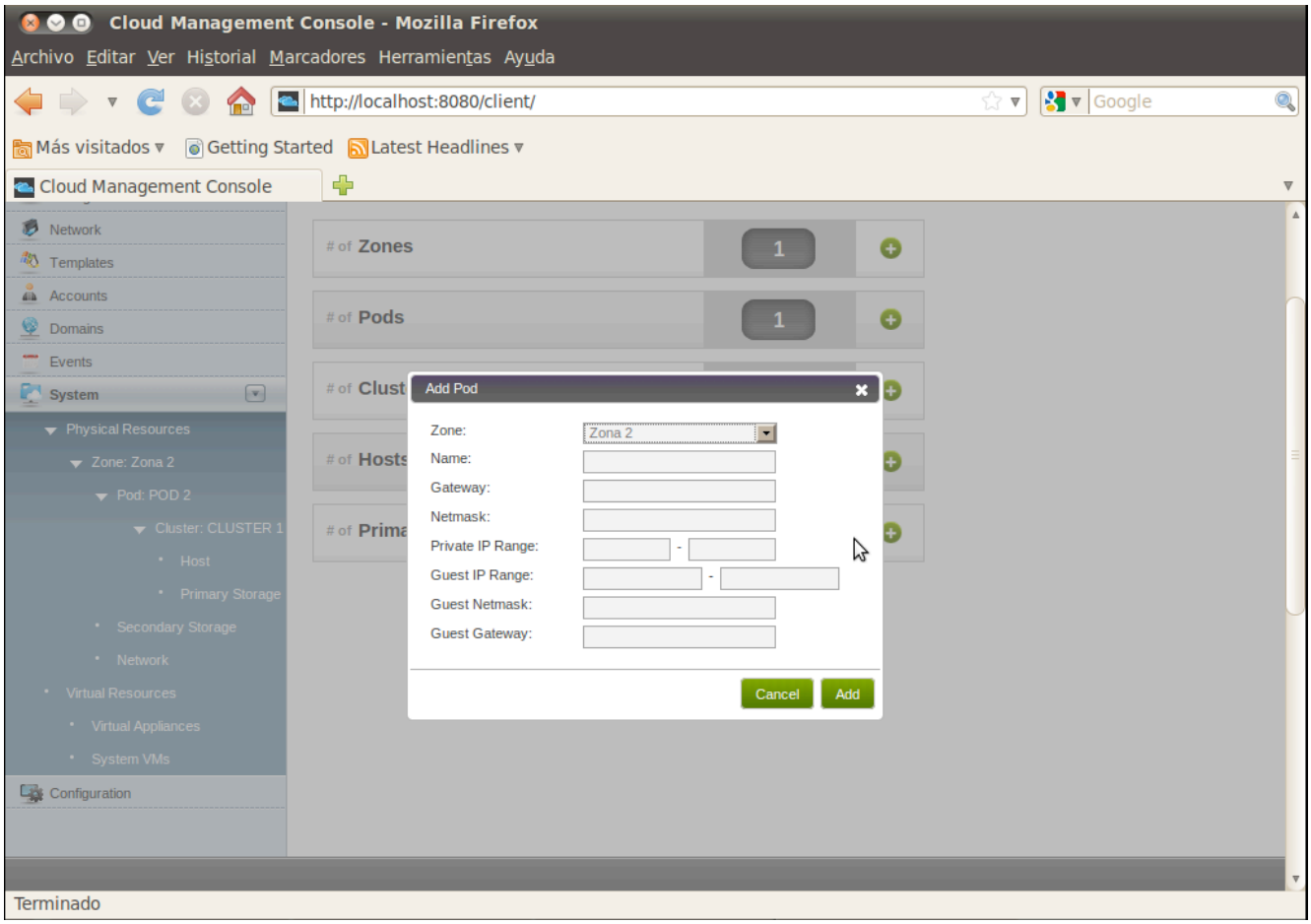

Figura 4.2.4.1.3.3 Formulario para añadir un nuevo pod

3. Configurar almacenamiento primario

1. Acceder al interfaz Web de gestión de CloudStack
2. Autenticarse en CloudStack como administrador
3. Seleccionar “Storage”
4. Seleccionar “Primary Storage”
5. Seleccionar “Add Primary Storage”

Para añadir el almacenamiento es necesario especificar:

- **Availability Zone**: Zona que empleará el almacenamiento primario
- **Pod:** Pod que empleará el almacenamiento primario
- **Name:** Nombre del almacenamiento primario
- **Protocolo:** Protocolo a emplear.
- **Server:** Dirección del servidor de almacenamiento
- **Path:** Ruta al directorio compartido en el servidor de almacenamiento
- **Tags :** (Opcional) Define una etiqueta que puede emplearse como requerimiento para efectuar la instanciación de una máquina virtual

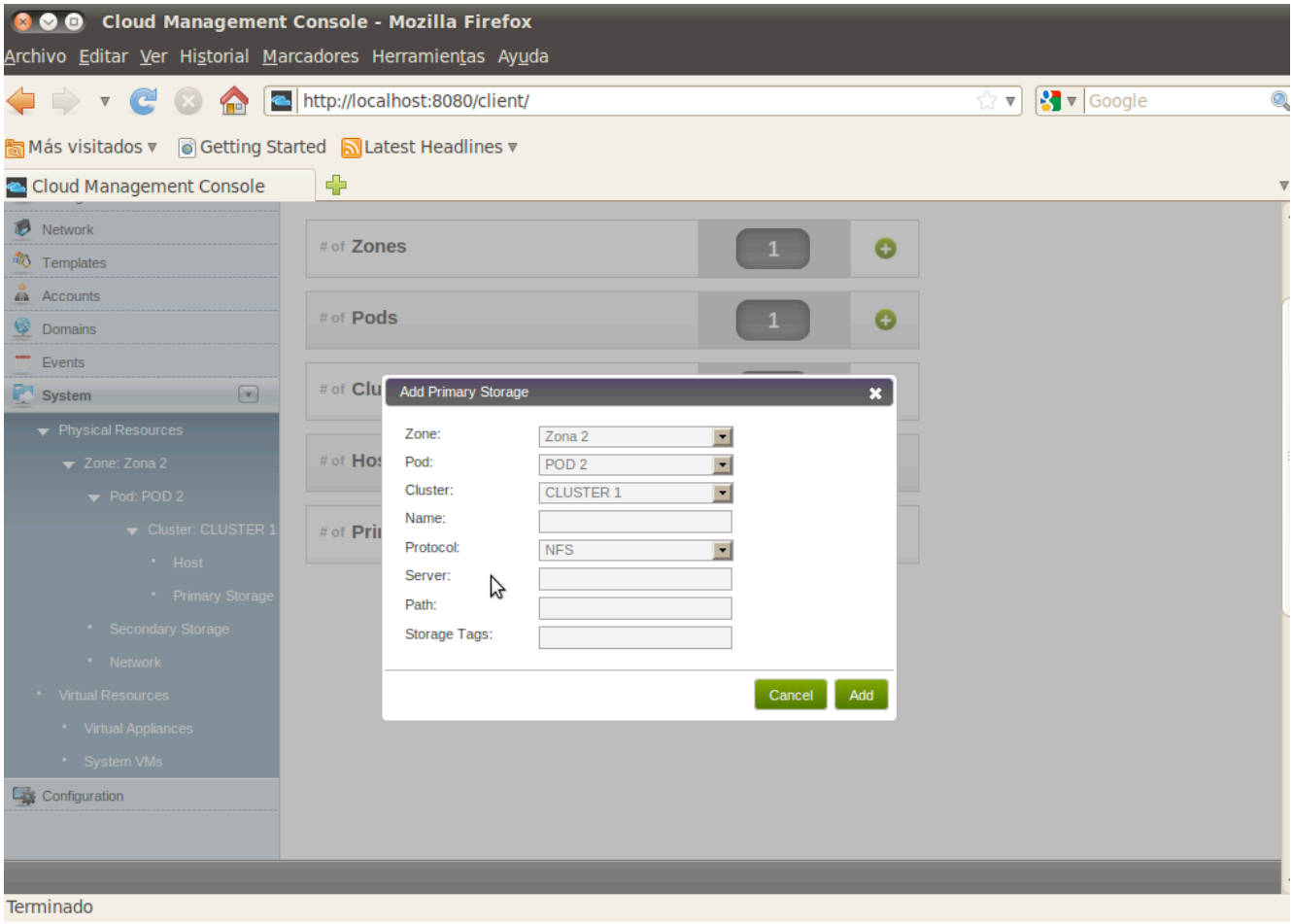

Figura 4.2.4.1.3.4 Formulario para añadir un nuevo almacenamiento primario

4. Configurar el almacenamiento secundario

1) Acceder al interfaz Web de gestión de CloudStack en:

http://management-server-ip-address:8080/client

2)Autenticarse en CloudStack como administrador

3) Seleccionar “Storage”

4) Seleccionar “Secondary Storage”

5) Seleccionar “Add Secondary Storage”, para añadir el almacenamiento es necesario especificar:

- Availability Zone: Zona que empleará el almacenamiento secundario
- NFS Server: Dirección del servidor NFS de almacenamiento secundario
- Path Ruta al directorio compartido en el servidor de almacenamiento

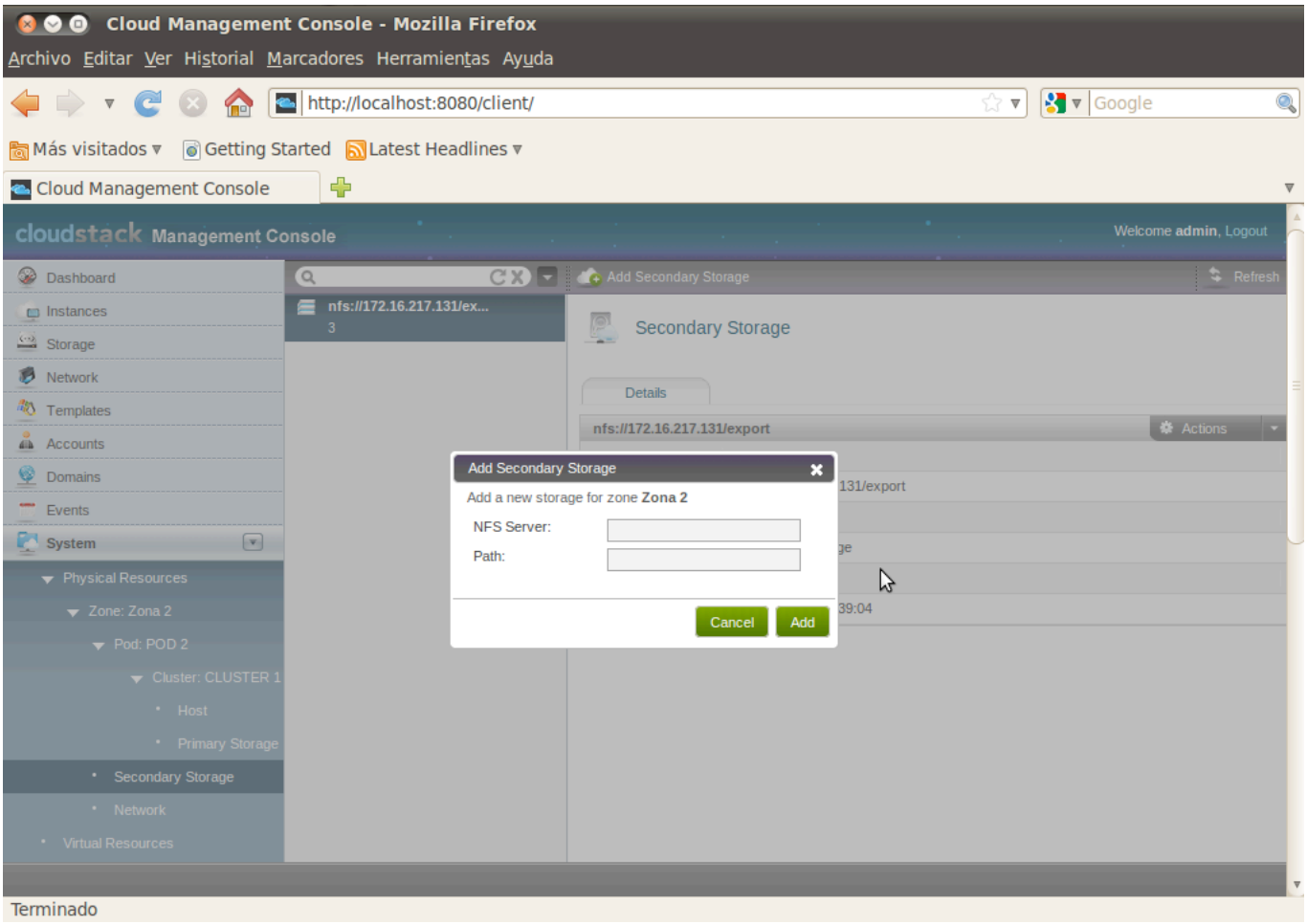

Figura 4.2.4.1.3.5 Formulario para añadir un nuevo almacenamiento secundario

Una vez se ha efectuado la configuración del almacenamiento primario y secundario, CloudStack descargará de forma automática dos plantillas de máquinas virtuales, una plantilla de router virtual y una plantilla de máquina virtual de ejemplo.

Configurar modo de red

Los modos de red soportados son los siguientes:

- **Direct**: Las máquinas virtuales obtienen direcciones IP de la red de la misma forma que si se tratase de máquinas físicas
- **Public**: Las máquinas virtuales hacen uso de un router virtual, que tiene asignada una dirección IP que le permite acceder a Internet, que proporciona el servicio NAT para todos las máquinas huésped de una cuenta de usuario

De forma opcional se puede emplear VLAN para proporcionar aislamiento entre distintas redes.

Ambos modos de red pueden configurarse dentro “Global Settings” en la interfaz de administración de CloudStack, si queremos habilitar el modo “Direct” deberemos modificar los siguientes parámetros de configuración:

- direct.attach.network.externalIpAllocator.enabled true
- network.type vnet

En cambio si lo que queremos es habilitar el modo “Public” el parámetro *direct.attach.network.externalIpAllocator.enabled* debe ser puesto a false.

Acto seguido deberemos reiniciar la plataforma mediante el siguiente comando:

```
service cloud-management restart
```

4.2.4.1.4 Instalación de los nodos de computación

En nuestra solución vamos a emplear KVM como hipervisor, debido a su naturaleza Open Source, sus prestaciones y el amplio soporte del que dispone, así mismo el sistema operativo sobre el que instalaremos KVM y el software de los nodos es Ubuntu 10.04.

Los pasos para la instalación y configuración de un nodo KVM son los siguientes:

1) Obtener permisos de administración:

```
sudo -s
```

2) Preparar el repositorio para encontrar los paquetes de CloudStack:

```
vi /etc/apt/sources.list

Modificaciones a realizar

Añadir:

deb http://download.cloud.com/apt/ubuntu/stable/oss ./
```

3)Actualizar la lista de paquetes locales e instalar Cloud Agent

```
aptitude update

aptitude install cloud-agent
```

2) Modificar archivo /etc/hosts para evitar que el looback se resuelva por medio de ipv6

```
vi /etc/hosts

Descripción

Eliminar localhost de la línea:

::1 localhost ip6-localhost ip6-loopback
```

3) Establecer configuración estática de la red

```
vi /etc/network/interfaces

Configurar la red de forma estática sin emplear network manager
```

4) Configurar nodo

```
cloud-setup-agent

Cloud Agent solicitará la dirección IP del servidor de gestión.

Será necesario seleccionar el pod al que pertenece el nodo.
```

7) Una vez realizados estos pasos deberemos acceder a la interfaz de gestión de CloudStack y comprobar que efectivamente se ha añadido el nuevo nodo y que este se encuentra operativo.

Después de este proceso de configuración la plataforma ya es capaz de crear nuevas instancias, añadir nuevos dominios, gestionar usuarios, etc. Por tanto podemos dar por finalizado el proceso de instalación y configuración.

4.2.5. Implantación y aceptación del sistema de información (IAS)

Una vez construida la plataforma, podemos concluir que se han cumplido todos los requisitos, en la nueva plataforma, el tiempo de despliegue de la infraestructura actual es notablemente inferior a la anterior configuración, así mismo el mantenimiento y la administración se facilitan al disponer de una plataforma centralizada y con una monitorización mucho más avanzada.

Es posible acceder a la infraestructura de la plataforma desde virtualmente cualquier lugar con acceso a una conexión a internet, además este acceso se realiza de una forma fácil y segura, sin complicados procesos de configuración y sin necesidad de instalar software de terceros por tanto la asignatura es totalmente independiente del laboratorio. Además como hemos visto se ha habilitado el acceso por roles, y es posible un control individualizado de todos los grupos, así como un control global por parte del administrador root.

El coste de implantación del plataforma es relativamente bajo, básicamente se reduce a la compra de los servidores (siendo posible la utilización de parte del CPD existente en el DSIC), y a las horas de trabajo del personal dedicado, las características específicas y el coste detallado del sistema escapan del objetivo de este proyecto, ya que dependen en gran medida de decisiones operativas del DSIC.

5. Valoración personal

He aprendido muchísimo durante la realización de este proyecto, añadiendo gran cantidad de conocimientos a los que ya adquirí en la carrera y que sin duda podré aplicar en mi trayectoria profesional.

Los paradigmas del Cloud Computing y la virtualización me parecen muy interesantes y es la razón por la que me decidí por este proyecto, consiguiendo así una formación muy integral ya que en mi titulación me he especializado en Ingeniería del software y este proyecto corresponde al área de sistemas, por lo tanto he podido aprender muchas cosas de ambas áreas.

El hecho de haber concebido el proyecto desde el principio como algo real, me ha dado una muy buena experiencia sobre como funcionan los proyectos en la realidad, sobre que tipos de obstáculos me puedo encontrar, así como maneras de superarlos, ha sido un proyecto largo, pero sin duda ha merecido la pena.

En cuanto a las tecnologías presentadas en este proyecto, mi opinión personal es que el paradigma del Cloud Computing va a ser el eje principal de la mayoría de nuevas tecnologías y aplicaciones que surjan a corto-medio plazo, ya sea basándose en soluciones públicas, privadas o híbridas.

Se superarán los problemas actuales y aumentará la competencia para los proveedores Cloud. Es evidente que las tecnologías de virtualización también mejorarán, reduciendo cada vez más las diferencias de rendimiento entre las soluciones tradicionales y las basadas en sistemas virtualizados, no obstante yo no tildaría a estas soluciones como “revolucionarias” pero sin duda, lo que si afirmo es que están cambiando (y cambiarán mas) la forma de como entendemos los sistemas de información.

Referencias

- Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, Matei Zaharia, *Above the Clouds: A Berkeley View of Cloud Computing*, Electrical Engineering and Computer Sciences University of California at Berkeley, Febrero 2009
- Frank Stienhans, *Cloud Computing @ SAP, Julio 2009*
- Intel Corporation, *Intel Cloud Builder Guide: Cloud Design and Deployment on Intel Platforms*
- John W. Rittinghouse, James F. Ransome, *Cloud Computing: Implementation, Management, and Security*, CRC Press, 2010, ISBN: 978-1-4398-0680-7
- Fundación de la innovación Bankinter, *Cloud Computing: La tercera ola de las Tecnologías de la Información,* 2010
- Mike Culver*, Web scale computing,* Abril 2007
- Cloud Security Alliance (CSA), *Top Threats to Cloud Computing V1.0*, Marzo 2010
- Oracle, *Architectural strategies for Cloud Computing,* Agosto 2009
- Mike Danseglio, *Using Cloud Services to Improve Web Security*, Realtime Publishers
- Instituto Nacional de Tecnologías de la Comunicación (INTECO), *Riesgos y amenazas en Cloud Computing,* Marzo 2011
- Judith Hurwitz, Robin Bloor, Marcia Kaufman, Fern Halper, *Cloud Computing for dummies,* Wiley Publishing, Inc, 2010, ISBN: 978-0-470-48470-8

- Michael Miller, *Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online*, QUE Publishing, Agosto 2008, ISBN: 0-7897-3803-1
- Cloud.com, *CloudStack Installation Guide*, Citrix Systems, 2011.
- Centro Nacional de Referencia de Aplicaciones de las TIC basadas en fuentes abiertas, (CENATIC), *Cloud Computing y Software de Fuentes Abiertas*, Abril 2011.
- Wilmar Arturo Castellanos Morales, *Consideraciones de seguridad y privacidad en cloud computing*, Deloitte & Touche LTDA Febrero de 2011
- Centro de Supercomputación de Galicia (CESGA), *Comparativa de software de gestión cloud,* Mayo 2011
- Centro de Supercomputación de Galicia (CESGA), *Instalación y evaluación de CloudStack,* Mayo 2011
- Centro de Supercomputación de Galicia (CESGA), *Instalación y evaluación de OpenNebula,* Mayo 2011
- Eucalyptus Systems Inc, Eucalyptus Cloud Computing Platform User Guide, 2010
- Eric A. Marks, Bob Lozano, *Executive's Guide to Cloud Computing,* John Wiley & Sons, Inc, 2010, ISBN: 978-0-470-52172-4
- Juan Matias Granda, KVM: *The Kernel Virtual Machine, Virtualizando con GNU/Linux*, 2008
- Hewlett-Packard Development Company, *Ventajas de la virtualización,* 2008
- Jesús M. Doña, Juan E. García, Jesús López, Francisco Pascual, Rubén F. Pascual, *Virtualización de Servidores. Una Solución de Futuro,* 2008, Área de Tecnologías y Siste-

mas de Información. Hospital Universitario Virgen de La Victoria, Campus Universitario de Teatinos

- Vmware, *Understanding Full Virtualization, Paravirtualization, and Hardware Assist*, 2009
- Jorge González Villalonga, *Virtualización de la infraestructura informática: impacto en inversiones y costes de explotación*, 2006, ICAI.
- Josep Ros, *Virtualización Corporativa con VMware*, 2009, Ncora Information Technology S.L