

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

**Facultad 6**



---

# **Plataforma Integral de Virtualización y Almacenamiento con Tecnologías Libres (PIVA)**

---

**Trabajo de Diploma para optar por el título de  
Ingeniero en Ciencias Informáticas**

**Autores:** Bismark Castilla Monzón

Dany Esquijarosa Bonilla

**Tutor:** Ing. Norge Fajardo Vega

**Co-tutor:** Ing. Manuel Cheong Gómez

---

La Habana, julio de 2011

“Año 53 de la Revolución”



*"Lo que hace crecer el mundo no es el descubrir cómo está hecho, sino el esfuerzo de cada uno por descubrirlo"*

*José Martí*

## DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

\_\_\_\_\_  
Bismark Castilla Monzón  
**Autor**

\_\_\_\_\_  
Dany Esquijarosa Bonilla  
**Autor**

\_\_\_\_\_  
Ing. Norge Fajardo Vega  
**Tutor**

\_\_\_\_\_  
Ing. Manuel Cheong Gómez  
**Co-tutor**

## DATOS DE CONTACTO

### **Ing. Norge Fajardo Vega:**

Graduado en el 2007 de Ingeniero en Ciencias Informáticas de la Universidad de las Ciencias Informáticas (UCI). Especialista general del bloque 8 de la Infraestructura Productiva, donde se desempeña como administrador de la red, desde su graduación. Impartió la asignatura de Práctica Profesional en el primer semestre como profesor de la facultad 4.

[norge@uci.cu](mailto:norge@uci.cu)

### **Ing. Manuel Cheong Gómez:**

Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en el año 2007. Impartió clases durante 2 cursos de Teleinformática en la Facultad 2, en el Departamento de Sistemas Digitales. Hace aproximadamente un año comienza a prestar servicio en el Organismo Central del MIC como especialista principal de Seguridad Informática, donde comienza reorganizando dicha actividad en la entidad. Actualmente cumple la función de Administrador de red principal del MIC, donde ha venido encabezando un proceso de perfeccionamiento tanto en la infraestructura, como en los servicios que se brindan.

[cheong@mic.cu](mailto:cheong@mic.cu)

## AGRADECIMIENTOS

### Bismark:

---

A mi mamá y a mi hermano por ser pacientes y apoyarme en todo momento.

A mi novia por tener esa fe en mí.

A mi abuela y familia.

A la Comunidad de Software Libre.

A la Revolución Cubana que me ha permitido llegar hasta aquí.

### Dany:

---

A la UCI por las oportunidades ofrecidas.

Al Centro UCID, personificada en su jefatura, por la acogida.

A mis compañeros de apartamento por la hospitalidad.

A tío “Anselmito” por su confianza y preocupación. Por compartir su espacio, por ofrecerme su casa y hacerme sentir como en la mía.

A Natasha por sus atenciones.

A Karen y Enrique por compartir sus camas, su alegría.

A tía “Miraida” y Bejerano por sus constantes preocupaciones.

A tío Raúl por su ejemplo, por creer en mí.

A mis amigos (ellos saben quienes son), por su apoyo.

## DEDICATORIA

### Bismark:

A mi mamá y a mi hermano por apoyarme incondicionalmente y ser las dos personas más importante en mi vida.

A mi novia por apoyarme y por ser lo más lindo que me ha pasado.

A mi abuela y familia.

### Dany:

A mi mamá por:

Su personalidad y su conducta.

Su carácter recio, al tiempo que noble y dulce.

Su coherencia, consecuente en cada lugar y en cada momento.

Su sinceridad y su moral.

Su constancia y su autoridad.

Su trabajo y su conducta.

Su amor y sacrificio por la Revolución; como maestra, le ha dado sus mejores años.

Por todo.

## RESUMEN

En el presente trabajo se implementa una Plataforma Integral de Virtualización y Almacenamiento (PIVA) usando tecnologías libres, que permite la migración de la infraestructura tecnológica del Centro de Datos de la UCID, garantizando los niveles de seguridad y fiabilidad requeridos.

Para el desarrollo del sistema se realizó un análisis de las tecnologías y las soluciones libres de virtualización, así como de los sistemas libres de almacenamiento. Se definieron entornos de pruebas y las herramientas a utilizar en las pruebas comparativas.

Como parte de la metodología de análisis, se elaboraron los escenarios prospectivos, con las diferentes tecnologías para determinar la propuesta de solución integral. Se realizó una evaluación técnica a través del “Método de Expertos”, calculando un índice de aceptación de la solución para determinar el posible éxito de la plataforma.

## PALABRAS CLAVE

Virtualización, almacenamiento, software libre, prospectiva tecnológica, escenario prospectivo.

## ÍNDICE DE CONTENIDOS

<b>AGRADECIMIENTOS</b> .....	<b>V</b>
<b>DEDICATORIA</b> .....	<b>VI</b>
<b>RESUMEN</b> .....	<b>VII</b>
<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	<b>4</b>
1.1 INFRAESTRUCTURA TECNOLÓGICA (IT).....	4
1.2 CENTRO DE DATOS (CD).....	4
1.3 VIRTUALIZACIÓN.....	4
1.3.1 <i>Tipos de virtualización</i> .....	5
1.3.2 <i>Ventajas de la virtualización</i> .....	6
1.4 SISTEMAS DE ALMACENAMIENTO DE DATOS.....	7
1.5 ANTECEDENTES, TECNOLOGÍAS Y TENDENCIAS.....	10
1.5.1 <i>Software libre</i> .....	11
1.5.2 <i>Virtualización con tecnologías libres</i> .....	14
1.5.3 <i>Soluciones de virtualización con tecnologías libres</i> .....	15
1.5.4 <i>Sistemas de almacenamiento con tecnologías libres</i> .....	15
1.5.5 <i>Tendencias</i> .....	16
1.6 CONCLUSIONES PARCIALES.....	16
<b>CAPÍTULO 2: MÉTODOS, HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS</b> .....	<b>18</b>
2.1 DESCRIPCIÓN DE LAS TECNOLOGÍAS LIBRES DE VIRTUALIZACIÓN UTILIZADAS.....	18
2.2 DESCRIPCIÓN DE LAS SOLUCIONES DE VIRTUALIZACIÓN CON TECNOLOGÍAS LIBRES UTILIZADAS.....	22
2.3 DESCRIPCIÓN DE LOS SISTEMAS DE ALMACENAMIENTO CON TECNOLOGÍAS LIBRES UTILIZADOS.....	23
2.4 TIPOS DE PRUEBAS EXISTENTES.....	25
2.5 HERRAMIENTAS UTILIZADAS EN LAS PRUEBAS AUTOMÁTICAS.....	26
2.6 ESCENARIOS DE PRUEBA.....	28
2.7 METODOLOGÍA DE ANÁLISIS: LA PROSPECTIVA TECNOLÓGICA.....	35
2.7.1 <i>Análisis estructural</i> .....	37
2.7.2 <i>Estimación y diseño de los escenarios</i> .....	38
2.7.3 <i>Estrategia para lograr el escenario apuesta</i> .....	40
2.8 CONCLUSIONES PARCIALES.....	41
<b>CAPÍTULO 3: IMPLEMENTACIÓN, CONFIGURACIÓN Y EVALUACIÓN DE LA SOLUCIÓN</b> .....	<b>42</b>
3.1 INSTALACIÓN DE LA SOLUCIÓN DE VIRTUALIZACIÓN: PROXMOX.....	42
3.2 CONFIGURACIÓN DE LA SOLUCIÓN DE VIRTUALIZACIÓN: PROXMOX.....	42
3.3 INSTALACIÓN DEL SISTEMA DE ALMACENAMIENTO: NEXENTAS TOR.....	44
3.4 CONFIGURACIÓN DEL SISTEMA DE ALMACENAMIENTO: NEXENTAS TOR.....	45
3.5 EVALUACIÓN TÉCNICA DE LA SOLUCIÓN.....	49
3.6 CONCLUSIONES PARCIALES.....	53
<b>CONCLUSIONES</b> .....	<b>54</b>
<b>RECOMENDACIONES</b> .....	<b>55</b>
<b>BIBLIOGRAFÍA</b> .....	<b>56</b>
<b>ANEXOS</b> .....	<b>59</b>
ANEXO 1:.....	59
ANEXO 2:.....	60
<b>GLOSARIO</b> .....	<b>61</b>

## INTRODUCCIÓN

Actualmente la Infraestructura Tecnológica (IT), define en gran medida el éxito y la eficiencia de las Pequeñas y Medianas Empresas (PYMES), lo cual se traduce muchas veces en un incremento sostenido de la inversión.

Ante esta realidad para un país en desarrollo como Cuba que orienta sus esfuerzos hacia una economía del conocimiento y de los servicios, es decir, una economía que haga un “uso intensivo de la información, la tecnología y el conocimiento en la creación de valor” [1], que intenta informatizar la sociedad, son evidentes la dependencia tecnológica, la inseguridad y los altos costos de la importación de software; es un desafío desvincularse de las pautas establecidas por las corporaciones multinacionales, que ejercen un monopolio absoluto en el sector, teniendo como premisas el acceso abierto y la migración hacia Software Libre.

El elemento más importante hoy de la infraestructura tecnológica de información es el Centro de Datos (CD). En él se alojan los servidores donde se ejecutan las aplicaciones y los procesos que rigen el funcionamiento de cualquier empresa, bajo las normas de temperatura, humedad y suministro eléctrico necesarios. Son esenciales para el tráfico, el procesamiento y el almacenamiento de la información. Deben ser muy confiables y seguros al tiempo que tienen que ser capaces de adaptarse al crecimiento y la reconfiguración.

Para alcanzar la confiabilidad, seguridad, reconfiguración y la adaptación al crecimiento, la nueva tendencia que emerge por encima de las demás, es la unión que se está dando entre la virtualización, la seguridad y el almacenamiento de datos, dentro de una amplia convergencia a Protocolo de Internet (IP, por sus siglas en inglés). Hoy se impone por su propio peso la virtualización del almacenamiento.

La Unidad de Compatibilización, Integración y Desarrollo de Software para la Defensa (UCID), resultado de la integración de las Fuerzas Armadas Revolucionarias (FAR) y la Universidad de las Ciencias Informáticas (UCI), enmarcada dentro de la Infraestructura Productiva (IP), con la misión de desarrollar, integrar y compatibilizar soluciones informáticas para la defensa utilizando software libre, tiene en su CD una plataforma propietaria de virtualización VMware vSphere Hypervisor (ESXi) para particionar y consolidar sistemas en los diferentes entornos. Es una plataforma rentable, altamente escalable, con capacidades avanzadas de gestión de recursos pero propietaria y de licencia costosa, que requiere software con sus respectivas licencias para poder mantener la seguridad y protección. La plataforma de virtualización no se encuentra integrada a un sistema de almacenamiento. Existen soluciones profesionales propietarias de sistemas de almacenamiento que por su costo son imposibles de adquirir.

Ante esta realidad se nos presenta el **Problema científico** siguiente: ¿Cómo alcanzar una soberanía e independencia en la infraestructura tecnológica que integre los avances actuales de virtualización y almacenamiento?; teniendo como **Objeto de Estudio** de la investigación: las tecnologías y soluciones libres de virtualización y los sistemas libres de almacenamiento; y **Campo de acción**: el proceso de virtualización y almacenamiento de los sistemas en los servidores del CD de la UCID.

El **Objetivo General** del presente trabajo es: Implementar con tecnologías libres una plataforma integral de virtualización y almacenamiento en el CD de la UCID. Como **Objetivos Específicos** se tienen:

- Realizar el estudio de las tecnologías y los sistemas existentes.
- Elaborar el análisis y diseño de la propuesta de solución.
- Implementar la propuesta de solución en el entorno de estudio.

Para dar cumplimiento a tales objetivos se plantean las siguientes **Tareas de investigación**:

1. Estudio de las tecnologías y soluciones libres de virtualización, y los sistemas libres de almacenamiento para su posterior instalación y configuración.
2. Selección de las herramientas a utilizar en las pruebas automáticas.
3. Definición de los escenarios de pruebas para las herramientas seleccionadas.
4. Análisis comparativo de las tecnologías de virtualización y los sistemas de almacenamiento.
5. Construcción de los escenarios prospectivos, para integrar las tecnologías de virtualización y los sistemas de almacenamiento.
6. Selección del escenario apuesta para la propuesta de solución integral.
7. Implementación de la Plataforma Integral de Virtualización y Almacenamiento (PIVA).
8. Evaluación técnica de la PIVA mediante el Método de Expertos.

La estrategia de investigación es experimental, realizada a través del estudio y el análisis de las tecnologías de virtualización y los sistemas de almacenamiento, a partir de aquí se elabora el diseño de una plataforma que integre la virtualización y el almacenamiento mediante una adecuada selección de las tecnologías libres que garanticen la confiabilidad del sistema.

Como **métodos de trabajo científico** para el desarrollo de la investigación se utilizaron:

▪ **Métodos Teóricos:**

- **Analítico-Sintético:** permitió analizar y describir las diferentes tecnologías de virtualización y los sistemas de almacenamiento.

▪ **Métodos Lógicos:**

- **Sistémico:** se utiliza en el enfoque de sistema dado al problema de la investigación: los procesos de virtualización y los procesos de almacenamiento se asumen como un todo, que contribuyen a la infraestructura del CD de la UCID, donde la ejecución de los procesos integrados permiten obtener un sistema integral.

▪ **Métodos Empíricos:**

- **Entrevista:** se realizaron entrevistas tanto individuales como colectivas, con el fin de definir el problema a resolver y la estrategia de solución.
- **Opinión de Expertos:** se utilizó para realizar una evaluación técnica de la solución integral, basándonos en los conocimientos, las investigaciones, la experiencia y los estudios bibliográficos de los “expertos”.

El documento está compuesto por tres capítulos, además de la Introducción, las Conclusiones generales, las Recomendaciones, las Referencias, la Bibliografía, el Glosario de términos y varios Anexos con información complementaria para la comprensión del trabajo.

En el **Capítulo 1:** “Fundamentación Teórica”, se ofrece una visión general sobre los Centro de Datos, particularmente sobre los temas de virtualización y sistemas de almacenamiento. Describe los principales conceptos asociados al objeto de investigación y expresa en síntesis, antecedentes, tecnologías y tendencias dentro del Software Libre, a nivel internacional, en Cuba y en la UCI.

En el **Capítulo 2:** “Métodos, herramientas y tecnologías utilizadas”, se seleccionan y caracterizan las tecnologías, soluciones de virtualización y los sistemas de almacenamiento utilizadas. Se describen los escenarios de pruebas y las herramientas a utilizar en las pruebas automáticas. Se construyen los escenarios prospectivos con las tecnologías y los sistemas, como parte de la metodología de análisis para determinar la solución integral.

En el **Capítulo 3:** “Implementación, configuración y evaluación de la solución”, se propone una guía de implementación y configuración de la solución integral para un escenario de prueba definido. Se realizó una evaluación técnica de la solución usando la Opinión de Expertos.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

La información que se ofrece en este capítulo contiene una visión general sobre los CD, particularmente sobre los temas de virtualización y sistemas de almacenamiento. Describe los principales conceptos asociados al objeto de investigación y expresa en síntesis, antecedentes, tecnologías y tendencias dentro del Software Libre, a nivel internacional, en Cuba y en la UCI, que sirvan de base para la propuesta de solución.

### 1.1 Infraestructura tecnológica (IT).

Ante la implementación, desarrollo e investigación de redes y servicios, en los últimos años, las empresas han tenido que reestructurar funciones y presupuesto. Se han creado dependencias, servicios y por consiguiente, inversión en la IT de información. [2]

La IT está asociada en la práctica a los temas de conectividad, soporte técnico, gastos, hardware y software. Puede ser conceptualizada como el conjunto de elementos de hardware (redes, servidores, enlaces de telecomunicaciones), de software (sistemas operativos, bases de datos, herramientas de administración, lenguajes de programación) y de servicios (soporte, salvos, sistemas informáticos), que garantiza el normal funcionamiento y las estrategias de una empresa u organización.

### 1.2 Centro de Datos (CD).

Centro de Datos por su equivalente en inglés Data Center, también se conoce como Centro de Cómputo, Centro de Proceso de Datos (CPD), o Centro de Cálculo, es el elemento más importante de la infraestructura tecnológica.

Un CD es una entidad que tiene como objetivo satisfacer las necesidades de información, de manera veraz y oportuna. Es responsable de centralizar, custodiar y procesar la mayoría de los datos con los que opera una empresa. [3] Atendiendo a las diferentes modelos, y las diferentes utilidades, es el espacio de una empresa que contiene los servidores donde se ejecutan las aplicaciones y los procesos que rigen el funcionamiento de cualquier empresa, bajo las normas de temperatura, humedad y suministro eléctrico necesarios.

### 1.3 Virtualización.

La virtualización puede ser definida –según la Compañía Intel- como “la creación de un sistema computacional lógicamente segmentado, que funciona en una base real. Las plataformas virtuales son vistas por el usuario y funcionan como si fueran computadores físicos. Una vez que son abstractas y

divididas, a partir de bases ocultas y de ellas mismas, las plataformas virtuales son fácilmente transportables y robustas para simplificar la complejidad y, de esta forma, aumentan la seguridad”. [4] Esto implica hacer que un recurso físico, como un servidor, un sistema operativo o un dispositivo de almacenamiento, aparezca como si fuera varios recursos lógicos a la vez, o que varios recursos físicos, como servidores o dispositivos de almacenamiento, aparezcan como un único recurso lógico. Puede ser definida igualmente como la combinación o división de recursos (de computación) para presentar una visión transparente de un determinado entorno. En definitiva, “la virtualización es la clave para la gestión completa de los CD en el futuro y el máximo aprovechamiento de las grandes posibilidades tecnológicas que nos ofrece la industria del hardware: es necesaria para incrementar la eficiencia.” [5]

### 1.3.1 Tipos de virtualización.

Según la clasificación que se realice, existen varios modelos de virtualización.

**Emulación de hardware:** La máquina virtual simula completamente los componentes del hardware. Es variable ya que las máquinas virtuales pueden emular arquitecturas totalmente distintas a las del hardware real. Su utilidad práctica, en diferentes entornos, disminuye debido a la importante pérdida de rendimiento que la emulación provoca en el sistema final. [6]



Figura 1. Emulación de hardware.

**Virtualización completa:** Ejecuta múltiples máquinas virtuales simulando un conjunto de dispositivos de hardwares suficientes para que un sistema operativo pueda ejecutarse sin modificaciones sobre el hardware real. [7]

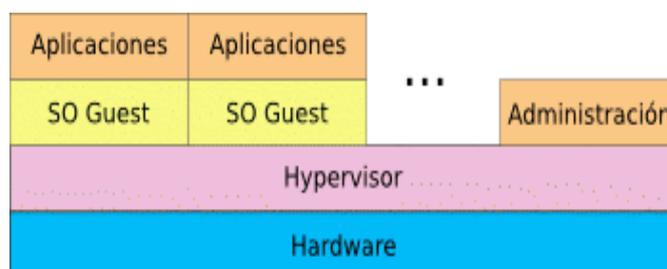


Figura 2. Virtualización completa.

**Paravirtualización:** Puede lograr rendimientos cercanos a la máquina física, pero requiere ciertas modificaciones en el Sistema Operativo de la máquina virtual de modo que coopere con el sistema anfitrión. [8]

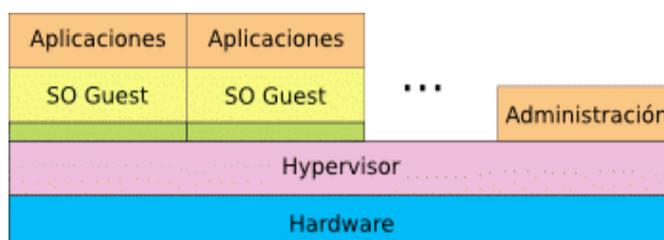


Figura 3. Paravirtualización.

**Virtualización a nivel del Sistema Operativo:** Se ejecuta una única instancia del Sistema Operativo y no se virtualiza el hardware, creando múltiples espacios de ejecución aislados muy similares en concepto a las máquinas virtuales. La pérdida de rendimiento debido a la virtualización es prácticamente nula. [9]

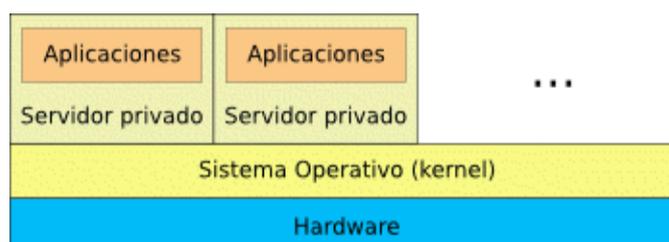


Figura 4. Virtualización a nivel del Sistema Operativo.

**Virtualización del almacenamiento:** Ofrece una vista lógica y abstracta de los dispositivos de almacenamiento físico, permitiendo que los discos de diferentes plataformas de almacenamiento aparezcan como un único recurso de almacenamiento. Los usuarios o aplicaciones accedan al almacenamiento sin saber cómo y dónde se ubica o administra físicamente.[10] Facilita el almacenamiento físico en un entorno para compartir en múltiples servidores de aplicaciones y dispositivos físicos detrás de la capa de virtualización a ser visualizado y administrado como si fuera un gran grupo de almacenamiento sin límites físicos. Este recurso de almacenamiento se puede entonces ubicar, proveer, migrar, duplicar y respaldar sin tener en cuenta su ubicación física. Puede implementarse con productos basados en el host, en el arreglo de almacenamiento o en la estructura. Habilita dos capacidades adicionales claves: la capacidad de ocultar o esconder volúmenes de los servidores que no están autorizados para acceder a esos volúmenes y ofrecer un nivel adicional de seguridad, y por último, la capacidad de modificar y aumentar sobre la marcha para satisfacer las necesidades de servidores individuales.

### 1.3.2 Ventajas de la virtualización.

A pesar de ser la virtualización una de las tecnologías más veteranas, es en los últimos años cuando realmente se ha convertido en una tecnología disruptiva<sup>1</sup>, sobre todo en los CD proporcionando múltiples ventajas, al permitir manejar más trabajo con menos equipos. La principal ventaja se centra en el ahorro de energía, espacio, capacidad de refrigeración, administración y mantenimiento.

*Aislamiento:* Las máquinas virtuales son totalmente independientes. Por tanto un fallo en una aplicación o en una máquina virtual afectará solamente a esa máquina virtual. El resto de máquinas virtuales seguirán funcionando normalmente.

*Seguridad:* Cada máquina tiene un acceso privilegiado independiente. Por tanto, un ataque de seguridad en una máquina virtual sólo afectará a esa máquina.

*Flexibilidad:* Nos permite crear las máquinas virtuales con las características de CPU, memoria, disco que necesitemos, sin necesidad de comprar un nuevo ordenador con esas características concretas. También podemos tener máquinas virtuales con distintos sistemas operativos, ejecutándose dentro de una misma máquina física.

*Agilidad:* La creación de una máquina virtual es un proceso muy rápido. Por tanto, si necesitamos un nuevo servidor lo podemos tener casi inmediatamente, sin necesidad de tener que pasar por el proceso de comprar un nuevo ordenador.

*Portabilidad:* Toda la configuración de una máquina virtual reside en uno o varios ficheros. Esto hace que sea muy fácil replicar o transportar la máquina virtual a otro servidor físico, simplemente copiando y moviendo dichos ficheros que encapsulan la máquina virtual. De esta forma se consigue una recuperación rápida en caso de fallo. Si se dispone de una copia de los ficheros de configuración de la máquina virtual, en caso de desastre, la recuperación será sencilla: bastará con arrancar la máquina virtual con los ficheros de configuración guardados. No será necesario reinstalar, recuperar backups y otros procedimientos largos que se aplican en las máquinas físicas.

“Virtualizar es una respuesta natural y actual a las necesidades que están surgiendo en las empresas y CD. La virtualización permite a las empresas evolucionar de un CD basado en hardware a otro basado en software, en el cual los recursos compartidos se asignan dinámicamente a las aplicaciones que los precisen”. [11]

#### 1.4 Sistemas de almacenamiento de datos.

---

<sup>1</sup> Tecnología disruptiva, es aquella que, siendo significativamente más barata que la actual, o con muchas mayores prestaciones es, con frecuencia, más conveniente de utilizar.

Al final de este año – según expertos - se añadirán “cada año 1.800 exabytes de nuevos datos. Esta masa inimaginable de información se extiende sobre una gran variedad de formatos. Esta explosión de almacenamiento de datos ha causado (o ha sido propiciada por) una explosión en el hardware de almacenamiento de información”. [12] Los bits del mundo residen hoy en una rica colección de exóticos dispositivos. Hoy tenemos diferentes tipos de sistemas de almacenamiento: local, Direct Attached Storage (DAS), Network Attached Storage (NAS), y Storage Area Network (SAN).

**Almacenamiento local:** Se trata del sistema clásico de discos internos del servidor. Es equiparable a Direct Attached Storage (DAS), pero debe diferenciarse porque en este caso son del propio servidor.

**Direct Attached Storage (DAS):** Almacenamiento directamente conectado. Se trata de un disco conectado a un host individual. Aquí la interfaz típica es Small Computer System Interface (SCSI) y cada vez más Serial Attached SCSI (SAS). Sin embargo, es posible cualquier protocolo de transferencia de datos orientado a bloques, tales como Advanced Technology Attachment ATA / ATA with Packet Interface (ATAPI), Canal de Fibra, Internet Small Computer Storage Interface (iSCSI) o Fibre Connection (FICON) / Enterprise Systems Connection (ESCON). [13]

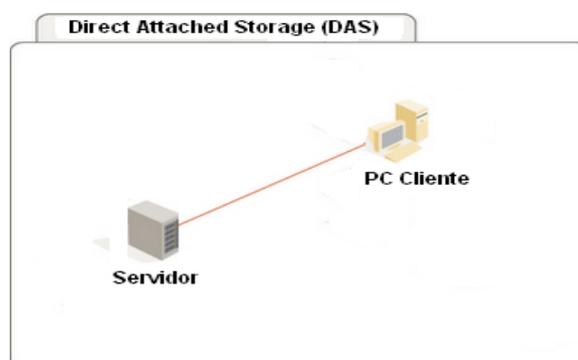


Figura 5. Sistema de almacenamiento DAS.

**Network Attached Storage (NAS):** Almacenamiento conectado directamente a la red. Se trata, básicamente, de un servidor de archivos fácil de administrar. NAS se suele utilizar para añadir capacidad de almacenamiento a una red informática existente sin demasiada carga administrativa. En contraste con DAS, un NAS es siempre un host aparte con su propio sistema operativo, aunque el sistema operativo esté muy personalizado para la función de servidor de archivos. NAS utiliza Ethernet /Internet Protocol para el enlace. Las sobrecargas relacionadas con el uso de redes de comunicaciones basadas en Ethernet estándar impiden la posibilidad de almacenamiento masivo de alta velocidad. Un Storage Area Network (SAN) evita estos inconvenientes. Publica los recursos de almacenamiento usando protocolos basados en ficheros, por ejemplo Common Internet File System (CIFS) o Network File System (NFS), que pueden ser compartidos con más de un equipo. [14]

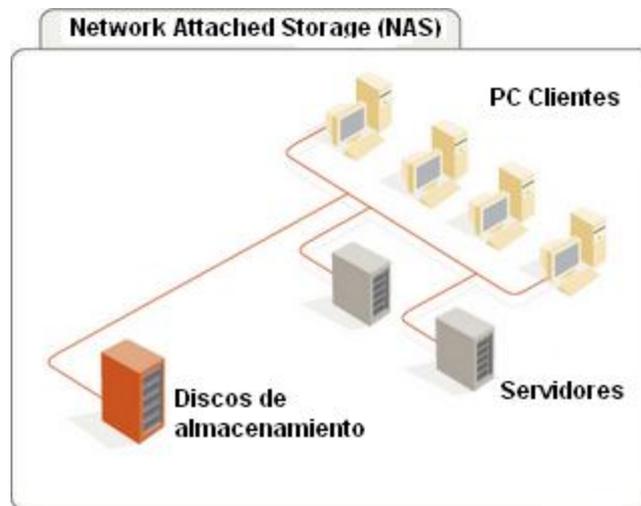


Figura 6. Sistema de almacenamiento NAS.

**Storage Area Network (SAN):** Red de área de almacenamiento. Una red entre los servidores y los recursos de almacenamiento que utilizan independiente de la red de los usuarios. El tráfico de datos en una SAN consta principalmente de transferencia de datos basados en bloques en lugar de ficheros, donde se transfieren bloques individuales en vez de archivos completos. Los protocolos de transporte que se suelen ver aquí son SCSI, Canal de Fibra o iSCSI. [15]

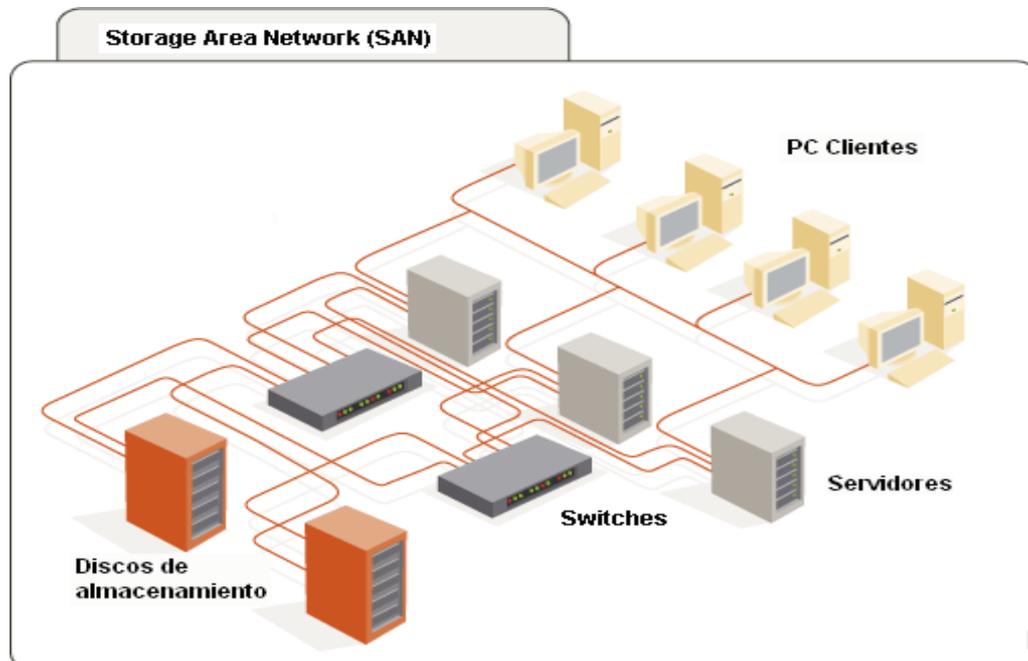


Figura 7. Sistema de almacenamiento SAN.

La red de área de almacenamiento puede ser de dos tipos:

**Red Canal de Fibra** (en inglés Fibre Channel, FC): La red Canal de Fibra es la red física de dispositivos Fibre Channel que emplea Fibre Channel Switches y Directores y el protocolo Fibre Channel Protocol (FCP) para transporte (SCSI-3 serial sobre Fibre Channel).

**Red Internet Protocol (IP):** Emplea la infraestructura del estándar Local Area Network (LAN) con hubs y/o switches Ethernet interconectados. Una SAN IP emplea iSCSI para transporte (SCSI-3 serial sobre IP). El protocolo iSCSI ha sido una tecnología a considerar en el escenario de los sistemas de almacenamiento en red, aunque siempre desde una óptica de segundo plano, superado en fondo y forma por el Canal de Fibra, tanto a nivel de rendimiento, la fiabilidad y la madurez. A este papel secundario ha contribuido en gran parte el escaso número de soluciones iSCSI que han aparecido desde su presentación en 2003.

### **Características SAN IP**

El protocolo iSCSI utiliza Transmission Control Protocol/Internet Protocol (TCP/IP) para sus transferencias de datos. Solamente requiere un simple y una sencilla interfaz Ethernet (o cualquier otra red compatible TCP/IP) para funcionar, a diferencia del “canal de fibra”.

Las pruebas que se han realizado muestran un excelente rendimiento de las soluciones iSCSI SANs, cuando se utilizan enlaces Gigabit Ethernet. Son construidas para minimizar el tiempo de respuesta del medio de transmisión. Permiten que los servidores y dispositivos de almacenamiento se conecten más de una vez a la SAN, de esta forma, se pueden tener *rut*as redundantes que a su vez incrementarían la tolerancia a fallos. La seguridad se garantiza implementando la tecnología de zonificación, que puede llevarse a cabo por hardware, software o ambas; una técnica adicional consiste en hacer que una Logical Unit Number (LUN) sea accesible sólo por una lista predefinida de servidores o nodos.

### **Ventajas SAN IP**

Simplifica la administración y añade flexibilidad. Tiende a maximizar el aprovechamiento del almacenamiento, puesto que varios servidores pueden utilizar el mismo espacio reservado para crecimiento. Un servidor puede acceder a uno o "n" discos y un disco puede ser accedido por más de un servidor, lo que hace que aumente el beneficio o retorno de la inversión. Es compatible con los dispositivos SCSI ya existentes. El rendimiento está directamente relacionado con el tipo de red que se utiliza. La capacidad de se puede extender de manera casi ilimitada y puede alcanzar cientos y hasta miles de terabytes. Permite compartir datos entre varios equipos de la red sin afectar el rendimiento porque el tráfico de SAN está totalmente separado del tráfico de usuario.

## **1.5 Antecedentes, tecnologías y tendencias.**

### **Antecedentes.**

La virtualización no es un tema novedoso, se considera que existe desde hace cuatro o cinco décadas. Por aquel entonces y hasta hace pocos años era aplicada en ámbitos exclusivos, sólo prácticamente para los grandes CD, tanto bancarios como militares y universitarios. Algunos de los usos pioneros de la virtualización incluyen al IBM 7044 (en el que la máquina física era la *M44*, que albergaba varias máquinas lógicas *44X* para los procesos) el Compatible Time Sharing System (CTSS) desarrollado por el Massachusetts Institute of Technology (MIT) en el IBM 7044, y el proyecto *Atlas* de la Manchester University (uno de los primeros supercomputadores del mundo, operativo en 1962), pionero en el uso de paginación bajo demanda y llamadas en modo supervisor. [16]

Actualmente la virtualización está asociada a conceptos como optimización, seguridad, escalabilidad y facilidad de la gestión. Durante algún tiempo, la virtualización de datos se consideró la última pieza del mapa virtual; pero, luego, los profesionales del almacenamiento comenzaron a contemplar la posibilidad de abstraer el almacenamiento lógico del almacenamiento físico. De esta manera, sería posible liberar información y obtener no solo independencia de los datos, sino también independencia virtual.

Si los recursos de procesamiento pueden distribuirse y ubicarse en los puntos más urgentes para necesidades o aplicaciones específicas, se deduce que el almacenamiento debe implementarse para lograr de inmediato que los datos estén disponibles, actualizados y sincronizados en tiempo real con una independencia total de tiempo y distancia. El almacenamiento virtual implica que el uso de datos trasciende el ancho de banda, la coherencia de caché y la latencia geográfica.

El usuario y los datos pueden estar en cualquier punto, y los datos son actuales, consistentes y recientes. De inmediato, el usuario obtiene la información necesaria sin tener que preocuparse en absoluto por el sistema o los datos.

#### **1.5.1 Software libre.**

El software libre es una cuestión de la libertad de los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. Significa que los usuarios de programas tienen las cuatro libertades esenciales.

- La libertad de ejecutar el programa, para cualquier propósito (libertad 0).
- La libertad de estudiar cómo trabaja el programa, y cambiarlo para que haga lo que usted quiera (libertad 1). El acceso al código fuente es una condición necesaria para ello.
- La libertad de redistribuir copias para que pueda ayudar al prójimo (libertad 2).

- La libertad de distribuir copias de sus versiones modificadas a terceros (la 3ª libertad). Si lo hace, puede dar a toda la comunidad una oportunidad de beneficiarse de sus cambios. El acceso al código fuente es una condición necesaria para ello.

Un programa es software libre si los usuarios tienen todas esas libertades. Entonces, debería ser libre de redistribuir copias, tanto con o sin modificaciones, ya sea gratis o cobrando una tarifa por distribución, a cualquiera en cualquier parte. El ser libre de hacer estas cosas significa que no tiene que pedir o pagar el permiso.

La libertad de redistribuir copias debe incluir las formas binarias o ejecutables del programa, así como el código fuente; tanto para las versiones modificadas como para las no lo están. (Distribuir programas en forma de ejecutables es necesario para que los sistemas operativos libres se puedan instalar fácilmente).

Para que la 1ª y 3ª libertad, para realizar cambios y publicar versiones mejoradas, tengan sentido; debe tener acceso al código fuente del programa. Por consiguiente, el acceso al código fuente es una condición necesaria para el software libre. El “código fuente” ofuscado no es código fuente real, y no cuenta como código fuente.

La 1ª libertad incluye la libertad de usar su versión modificada en lugar de la original. Si el programa se entrega con un producto diseñado para ejecutar versiones modificadas de terceros, pero rechaza ejecutar las suyas, una práctica conocida como “arranque seguro” (en la terminología perversa de los que la practican); la 1ª libertad se convierte más en una ficción teórica que en una libertad práctica. Estos binarios no son software libre, incluso si se compilaron desde un código fuente que es libre.

Una manera importante de modificar un programa es fusionando subrutinas y módulos libres disponibles. Si la licencia del programa dice que no puede fusionar un módulo existente con una debida licencia, así como si le requiere ser el titular de los derechos de autor de lo que agregue, entonces la licencia es demasiado restrictiva para calificarla como libre.

La 3ª libertad incluye la libertad de liberar sus versiones modificadas como software libre. Una licencia que requiera que las versiones modificadas no sean libres, no se puede considerar como una licencia libre.

Para que estas libertades puedan ser reales, deben ser irrevocables siempre que usted no cometa ninguna equivocación; si el programador del software tiene el poder de revocar la licencia, o de cambiar retroactivamente sus términos, sin que usted se haya equivocado para justificarlo, el software no es libre.

Sin embargo, ciertos tipos de reglas sobre la manera de distribuir software libre son aceptables, cuando no entran en conflicto con las libertades principales. Por ejemplo, el copyleft es la regla en base a la cual, cuando redistribuye el programa, no puede agregar restricciones para denegar a las demás personas las libertades principales. Esta regla no entra en conflicto con las libertades principales; más bien las protege.

“Software libre” no significa “que no sea comercial”. Un programa libre debe estar disponible para el uso comercial, la programación comercial y la distribución comercial. La programación comercial de software libre ya no es inusual; tal software libre comercial es muy importante. Puede haber pagado dinero para obtener copias de software libre, o puede haber obtenido copias sin costo. Pero sin tener en cuenta cómo obtuvo sus copias, siempre tiene la libertad de copiar y modificar el software, incluso de vender copias.

Las normas del estilo “si pone a disposición su versión de este modo, también debe hacerlo de este otro modo” también pueden ser, bajo la misma condición, admisibles. Las normas que obligan a liberar el código fuente a los usuarios de las versiones que publica también son admisibles. [17]

### **Software libre en Cuba**

El uso del Software Libre es sustentable en Cuba, a partir de las ventajas que tiene con respecto al software propietario<sup>2</sup>.

Desde el punto de vista de la independencia tecnológica, adquiere un carácter político, al no encontrarse a merced de los monopolios del sector, que tienen la capacidad legal de reclamar la no utilización de un Sistema Operativo, basándose en las leyes de la propiedad industrial, provocando una interrupción inmediata del programa de informatización de la sociedad. Representa la alternativa para los países pobres, y es por concepción, propiedad social, si se tiene en cuenta que una vez que comienza a circular rápidamente se encuentra disponible para todos los interesados sin costo alguno o a muy bajo costo. Es desarrollado de forma colectiva y cooperativa, tanto en su creación como en su desarrollo, tanto cuantitativa como cualitativamente mostrando su carácter público y sus objetivos de beneficiar a toda la comunidad.

Su utilización no implica gastos adicionales por concepto de cambio de plataforma de software, por cuanto es operable en el mismo soporte de hardware con que cuenta el país. La adquisición de cualquiera de sus distribuciones puede hacerse de forma gratuita, descargándolas directamente de Internet o en algunos casos a muy bajos precios, se garantiza su explotación con un mínimo de

---

<sup>2</sup> El software propietario, es cualquier programa informático en el que el usuario tiene limitaciones para usarlo, modificarlo o redistribuirlo.

recursos, por cuanto no hay que pagar absolutamente nada por su utilización (no requiere de licencia de uso, las cuales son generalmente muy caras), distribución y/o modificación.

Desde el punto de vista tecnológico y de la seguridad permite su adaptación a los contextos de aplicación, al contar con su código fuente, lo cual garantiza un mayor por ciento de efectividad, además la corrección de sus errores de programación y obtención de las actualizaciones y nuevas versiones.

La migración ordenada hacia plataformas de código abierto en el país, tiene en el Grupo Técnico Nacional para la Migración a Software Libre que preside la UCI, un eslabón fundamental. Las FAR no están ajenas a este proceso de informatización de la sociedad.

### 1.5.2 Virtualización con tecnologías libres.

**Kernel-based Virtual Machine (KVM):** Máquina virtual basada en el núcleo.

Se trata de una solución basada en virtualización completa para GNU/Linux x86 que disponga de un procesador con extensiones (Intel VT o AMD-V). Consiste en un módulo del núcleo (kvm.ko) que provee infraestructura de virtualización y un módulo específico para cada procesador (kvm-intel.ko o kvm-amd.ko). Su característica principal es que permite ejecutar imágenes de GNU/Linux o Windows sin modificar. Cada máquina virtual poseerá su propio hardware virtualizado privado: Tarjeta gráfica, tarjeta de red. [18]

#### **Xen**

Es una máquina virtual de código abierto desarrollada por la Universidad de Cambridge. La meta del diseño es poder ejecutar instancias de sistemas operativos con todas sus características. Proporciona aislamiento seguro, control de recursos, garantías de calidad de servicio y migración de máquinas virtuales en vivo. Los sistemas operativos deben ser modificados explícitamente para correr Xen (aunque manteniendo la compatibilidad con aplicaciones de usuario). [19]

#### **OpenVZ**

Se basa en contenedores para Linux. Crea múltiples recipientes seguros, aislados (también conocido como entornos virtuales) en un único servidor físico, permitiendo una mejor utilización del servidor y garantizar que las aplicaciones no entren en conflicto. Cada contenedor lleva a cabo y ejecuta exactamente como un servidor independiente, un contenedor puede ser reiniciado de forma independiente y tiene acceso de root, usuarios, direcciones IP, memoria, procesos, archivos, aplicaciones, librerías del sistema y archivos de configuración. [20]

### 1.5.3 Soluciones de virtualización con tecnologías libres.

#### **OpenNebula.**

Es un motor de infraestructura virtual que permite el despliegue dinámico y re-ubicación de los servicios virtualizados (grupos de máquinas virtuales interconectados) dentro y entre los sitios. Extiende los beneficios de las plataformas de virtualización que van desde un recurso físico a un conjunto de recursos, la disociación del servidor, no sólo de la infraestructura física, sino también de la ubicación física. [21]

#### **OpenNode.**

Es una plataforma de virtualización y una solución para la gestión de sistemas virtualizados soportando entornos OpenVZ y KVM. Posee un fácil sistema de administración web que permite la administración de maquinas virtuales que operan sobre la plataforma. [22]

#### **Proxmox.**

Es una sencilla plataforma de virtualización, para la implementación de máquinas virtuales utilizando los entornos OpenVZ y KVM. Posee una sencilla interfaz web de administración para la gestión de los sistemas virtualizados. [23]

#### **oVirt node.**

Una imagen pequeña, con bibliotecas y utilidades para administrar máquinas virtuales con servicio de libvirt a través de KVM y permite interactuar con el servidor oVirt. oVirt node se puede ejecutar en modo independiente o en conjunto con el servidor de administración oVirt, un sistema de gestión de máquinas virtuales consola basada en web. oVirt node es una imagen LiveCD que se puede instalar en los servidores físico, o correr como una imagen livecd desde un CD o desde un USB. [24]

### 1.5.4 Sistemas de almacenamiento con tecnologías libres.

#### **OpenFiler.**

Distribución GNU/Linux para el almacenamiento en red. Soporta los protocolos CIFS, NFS, Hypertext Transfer Protocol (HTTP) y File Transfer Protocol (FTP). Tiene soporte iSCSI tanto cliente como servidor. Replicación a nivel de bloques. Clúster de alta disponibilidad. Integración para Network Information Service (NIS), Lightweight Directory Access Protocol (LDAP) y Directorio Activo. Trabaja con cuotas de disco y tiene soporte para Redundant Array of Independent Disks (RAID) 0, 1, 5, 6 y 10. [25]

**FreeNas.**

Es un sistema open source (basado en Licencia BSD) que proporciona servicios de almacenamiento en red, permite convertir un ordenador personal en un soporte de almacenamiento accesible desde la red. Soporta los siguientes protocolos: CIFS (samba), FTP, NFS, Trivial File Transfer Protocol (TFTP), Apple Filing Protocol (AFP), RSYNC, iSCSI (iniciador y de destino) y Universal Plug and Play (UPnP). [26]

**NexentaStor.**

Constituye uno de los mejores sistemas de almacenamiento, sobre la base de un núcleo Opensolaris se constituye un sistema de almacenamiento profesional, con un elevado rendimiento gracias a la gestión inteligente de caches de lectura y escritura. Crea y gestiona volúmenes iSCSI y Fiber Channel (con opción de multipathing). Posibilita la instalación en sistemas de alta disponibilidad y replicación remota. [27]

**1.5.5 Tendencias.**

De las tendencias emergentes, una destaca por encima de las demás: se trata de la convergencia que se está dando entre la virtualización, la seguridad y el almacenamiento de datos. La red IP se está convirtiendo, con gran rapidez, en la plataforma esencial no sólo para distribuir voz, vídeo y datos. Gracias a esto, podemos hablar de tecnologías convergentes que permiten mejorar la productividad de las empresas, y que a su vez tienen la capacidad de ofrecer servicios avanzados para sus clientes. La red es, un elemento esencial a la hora de que las organizaciones desplieguen, gestionen y guarden sus centros de datos en un entorno virtualizado, con la misma seguridad que si el almacenamiento fuese físico.

Con la llegada de la virtualización, que ha traído al mundo de los servidores las ventajas vistas en la consolidación del almacenamiento, se han impuesto nuevas demandas y exigencias en el mundo del almacenamiento, hasta el punto que si no se dispone de una infraestructura de almacenamiento consolidado, gran parte de las ventajas de la virtualización quedan difuminadas. Este ha sido el primer paso al unir en un mismo barco almacenamiento más virtualización, formando un todo. Hoy las soluciones integrales resultan una necesidad y no existen en el mundo del software libre.

**1.6 Conclusiones parciales.**

Hoy es una necesidad migrar hacia tecnologías libres, en busca de independencia y soberanía tecnológica, seguridad y beneficios económicos.

El fenómeno de la convergencia evoluciona en paralelo a las exigencias de mayores tasas de productividad. Si hace unos años comenzaron a converger tecnologías como la voz desde sistemas analógicos a redes IP o el video también analógico al Video IP, desde hace un tiempo se observa cómo un nutrido grupo de tecnologías como son las de seguridad, los sistemas de almacenamiento y las soluciones de virtualización asumen el protagonismo de la convergencia.

El concepto de virtualidad se aplica en su máxima extensión cuando se habla de balancear la carga de trabajo y el aprovechamiento de espacios físicos en servidores y sistemas de almacenamiento en red teniendo en cuenta que los equipos pueden estar ubicados en distintos lugares del mundo. No es posible gestionar eficientemente ubicaciones físicas de sistemas (Virtualización) sin preservar su integridad (Seguridad) asegurando la continuidad del negocio (Almacenamiento de Datos) de las empresas. La correcta gestión de estas tres disciplinas tecnológicas ofrece una capacidad sin precedentes a las empresas.

## CAPÍTULO 2: MÉTODOS, HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS

En este capítulo se seleccionan y caracterizan las tecnologías, soluciones de virtualización y los sistemas de almacenamiento utilizadas. Se describen los escenarios de pruebas y las herramientas a utilizar en las pruebas automáticas. Se construyen los escenarios prospectivos con las tecnologías y los sistemas, como parte de la metodología de análisis para determinar la solución integral.

### 2.1 Descripción de las tecnologías libres de virtualización utilizadas.

**Kernel-based Virtual Machine (KVM):** Máquina virtual basada en el núcleo.

En un escenario de virtualización típico, un componente conocido como hypervisor ofrece una interfaz entre el sistema huésped y su anfitrión. El hypervisor 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 hypervisor en el núcleo, reduciendo así las redundancias y acelerando los tiempos de ejecución. Un controlador de KVM se comunica con el núcleo 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 núcleo. Un pequeño módulo del núcleo 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. En el entorno resultante varios procesos de máquina virtual se ejecutan cerca de otras tareas de espacio de usuario gestionadas directamente por el núcleo. 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.

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. [28]

### Migración de Máquinas Virtuales (MVs) con KVM

Migrar MVs es muy importante para el balanceo de carga y la reducción del tiempo de inactividad durante las actualizaciones. El proceso de migración consiste en mover un huésped de una máquina física a otra.

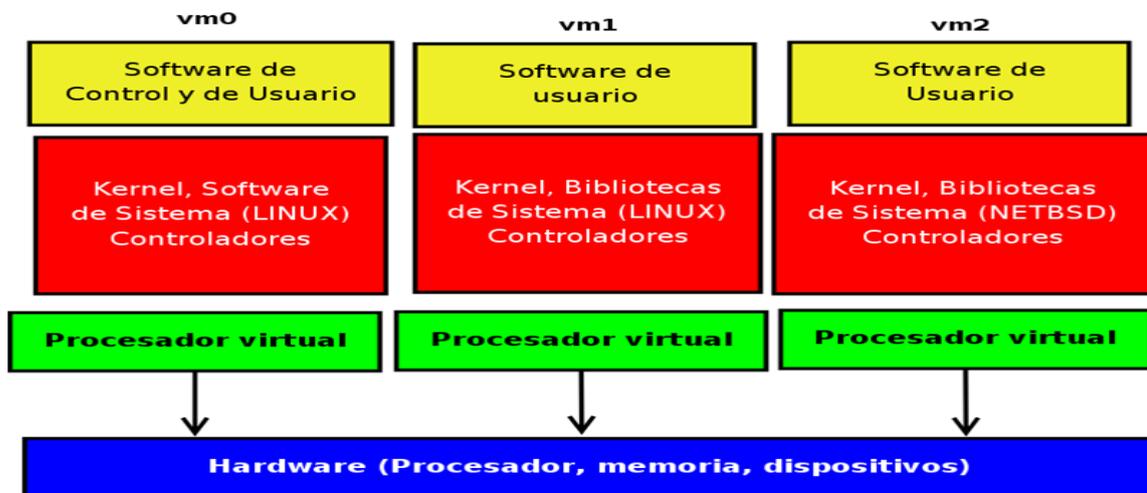


Figura 8. Diagrama de operación de KVM.

La ventaja del método de KVM reside en que los huéspedes no están implicados en la acción migratoria. Además, no se necesitan componentes especiales para migrar a través de un túnel de una sesión Secure Shell (SSH) o comprimir la imagen a migrar. Incluso podemos pasar la imagen a través de un programa antes de su transmisión a la máquina de destino. A menos que se activen funcionalidades específicas de hardware o de un anfitrión, podemos llevar a cabo la migración entre dos máquinas sin mayores problemas. Además podemos migrar tanto huéspedes desactivados como arrancados. La facilidad para la migración se encuentra en Qemu, por lo que no son necesarios cambios en la parte del núcleo para su activación. El dispositivo *state-sync* para la consecución de la migración y el estado de la MV se proporcionan y gestionan en espacio de usuario. [29]

### Ventajas del método de KVM

El método de procedimiento de KVM ofrece varias ventajas. Podemos re-usar todo el software y la infraestructura existentes sin necesidad de aprender nuevos comandos. Por ejemplo, kill y top funcionan normalmente con las tareas huésped del sistema anfitrión.

KVM fue diseñado originalmente para soportar anfitriones x86, centrándose en una virtualización total (ninguna modificación al sistema huésped) sin modificar el núcleo del anfitrión. De cualquier modo, cuando KVM comenzó a ganar desarrolladores y casos de usos interesantes, los desarrolladores comenzaron a trabajar para portar KVM a otras arquitecturas. El soporte de paravirtualización se encuentra también en desarrollo. Si un huésped se puede comunicar con el anfitrión, pueden acelerarse actividades como la de red o Entrada/Salida (E/S) de disco. Además se han propuesto y aceptado modificaciones al sistema anfitrión (Linux) que mejorarán la planificación de tareas y el intercambio de memoria.

KVM funciona en todo tipo de máquinas, servidores, escritorios, portátiles y placas específicas, pudiendo usarse las mismas herramientas de administración e infraestructura que usa Linux. El sistema KVM se integra con el planificador de Linux, la pila de E/S y todos los sistemas de archivos disponibles. Otros beneficios incluyen la migración durante la ejecución y el soporte para máquinas con procesadores NUMA o 4096. Si buscamos una alternativa eficiente para la virtualización que se integre bien con Linux, tenemos que dedicar un tiempo a explorar KVM. [30]

## Xen

Se ha convertido prácticamente en el estándar de virtualización. Dado que emula un ordenador completo, al estilo de VMware, el usuario tiene un abanico de elección de sistemas operativos invitados casi sin restricciones. Permite instalar distintos sistemas operativos sobre el mismo hardware para gestionar eficientemente su potencia computacional. La mayoría de las herramientas de gestión están disponibles, y el sencillo procedimiento de configuración consta sólo de unos pocos pasos. [31]

## Monitorizado por un Hipervisor

Xen introduce un hipervisor que va a contracorriente del núcleo, funcionando directamente sobre el hardware; el hipervisor emplea Xen Virtual Machine Monitor para acceder al hardware. Es él quien destina los recursos a cada una de las máquinas virtuales. Un sistema operativo invitado sólo se comunica con el hipervisor, y es completamente independiente del resto de los invitados.

El hipervisor conecta la máquina virtual a la red y es capaz de comunicarse con el mundo exterior. En el contexto de Xen, al sistema anfitrión con el hipervisor se le llama Dom 0 (o Domain 0, “Dominio 0”), y al sistema invitado Dom U (Domain U).

La única tarea que tiene Dom 0 es contener cada Dom U activo. Aunque en teoría no hay nada que impida emplear Dom 0 para propósitos normales, como hospedar un servidor web, el daño en potencia que un atacante podría causar secuestrando Dom 0 es una buena razón para evitar su empleo. Cualquier usuario con acceso completo a Dom 0 tiene también control total sobre cada Dom U activo. Por esta razón, es importante subcontratar cualquier servicio a un Dom U separado a la hora de usar Xen, y comenzar con un anfitrión fresco que no ejecute ningún servicio (excepto SSH). [32]

## Migración

Una de las opciones especiales de Xen es su capacidad de migrar los servidores virtuales de un anfitrión a otro mientras el servidor está en funcionamiento. Para probarlo, todo lo que necesitamos son dos servidores Xen y una sola máquina virtual.

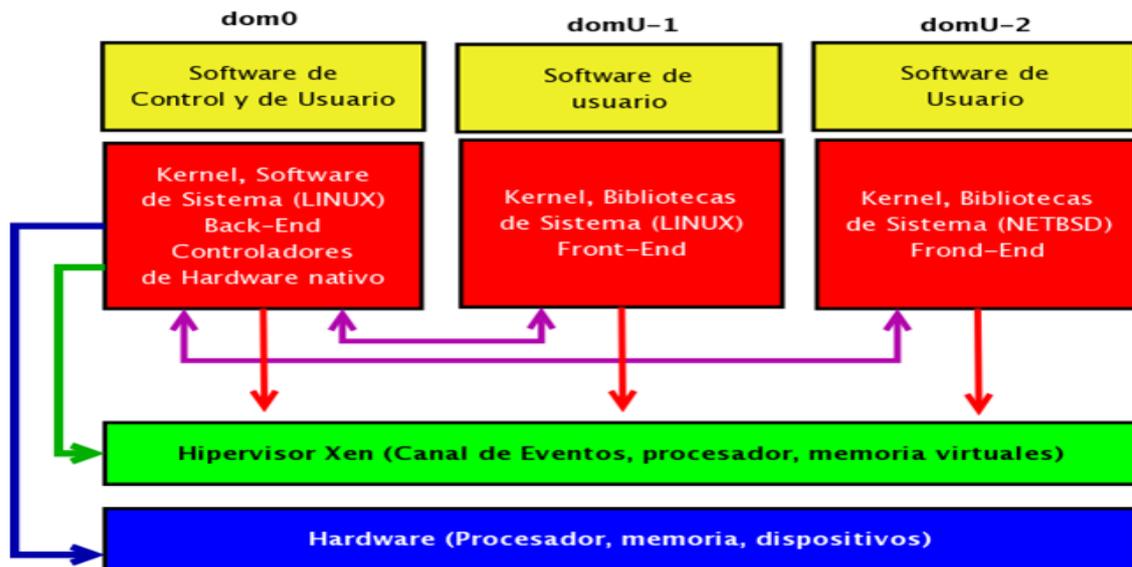


Figura 9. Diagrama de operación de Xen.

En el caso más sencillo podemos usar NFS para los datos compartidos, pero iSCSI o Distributed Replicated Block Device (DRBD) son también perfectamente válidos. Para migrar el sistema virtual de un servidor a otro basta con escribir `xm migrate -live Dom U servidor objetivo`. Las interrupciones del servicio mientras se realiza la mudanza no suelen pasar de un par de décimas de segundo.

Xen es una solución de virtualización muy potente. Al contrario que en otras opciones, como OpenVZ, tiene la ventaja de emular un ordenador completo, de un modo similar a VMware. Por ello, Xen es capaz de virtualizar otros sistemas operativos clientes aparte de Linux. [33]

### OpenVZ

Es considerada una de las mejores opciones a la hora de consolidar servidores Linux sobre anfitriones Linux, distribuidos bajo la licencia GPL. Permite la creación y mantenimiento de servidores virtuales aislados al introducir importantes modificaciones a nivel del núcleo Linux anfitrión, creando una completa infraestructura virtual que parece real para los sistemas hospedados. Dentro de la terminología OpenVZ los servidores virtuales son también a menudo llamados servidores privados virtuales (VPS, por sus siglas en inglés), entornos virtuales o incluso contenedores; en la mayoría de las ocasiones podremos leer que se trata de una solución basada en contenedores.

Incluye un completo conjunto de herramientas para la administración y monitorización de los servidores virtuales que logran hacer que la mayoría de las tareas puedan ser completadas con relativa facilidad.

Es una solución que aprovecha muy bien y explota las ventajas que ofrece aplicar virtualización: mejor utilización de los servidores, evitando conflicto entre servicios. Las modificaciones introducidas en el núcleo introducen las siguientes funcionalidades: [34]

*Virtualización y aislamiento:* Permite que existan múltiples servidores privados virtuales que comparten un mismo núcleo de manera segura. Cada VPS dispone de su propio conjunto de recursos y objetos.

*Administración de recursos:* De especial importancia debido a la existencia finita de recursos, se ocupa de limitar (y en muchos casos garantizar) el acceso a una cantidad determinada de recursos como procesador, memoria, espacio de disco, por servidor privado virtual.

*Checkpointing:* Nos ayuda a llevar a cabo procesos de congelación del estado de los servidores privados virtuales, pudiendo salvarlo en disco, para después si lo deseamos restaurarlo. Esto nos permitirá llevar a cabo migración en caliente de los vps entre diferentes servidores físicos. Esta migración es apreciada por parte del cliente como un retraso en la comunicación más que como tiempo de caída, al tener que migrar también las comunicaciones de red.

## 2.2 Descripción de las soluciones de virtualización con tecnologías libres utilizadas.

### Ovirt

Es una solución visualización y administración de MVs. Para controlarlo podemos usar su interfaz web, conectando al puerto 443, como viene siendo habitual en este tipo de software. Para la virtualización usa KVM (que viene incluido en el núcleo de Linux y requiere que el procesador tenga incluida la virtualización por hardware) y aunque en su web prometen que pronto podrá integrarse también Xen en forma de parche, de momento no tenemos esa posibilidad. Esto ya es una limitación, ya que Xen nos interesa, entre otras cosas, por su capacidad de paravirtualización, es decir, que el sistema operativo huésped o invitado es consciente de que está siendo virtualizado. Este conocimiento tiene como ventaja un incremento de la velocidad. Aunque también es cierto que Ovirt sigue evolucionando.

### OpenNebula

Es un gestor de infraestructuras virtuales distribuido bajo licencia Apache. Esta tecnología permite virtualizar las infraestructuras de almacenamiento y de recursos de cómputo de cualquier centro de datos existente y adaptarla dinámicamente en función de las necesidades de servicio. Combina tecnologías de virtualización, almacenamiento y redes. Permite la construcción de todo tipo de nubes: públicos, híbridos y privados.

Se puede utilizar como herramienta de virtualización para manejar la infraestructura virtual propia, a modo de nube privada, utilizando clúster de ordenadores (Xen, KVM o VMWare).

La arquitectura interna de OpenNebula se divide en tres capas:

*Herramientas:* Esta capa contiene herramientas como una Comand Line Interface (CLI), el planificador, la implementación del API libvirt y la Cloud RESTful interfaces.

*Núcleo:* Conjunto de componentes para controlar y monitorizar máquinas virtuales, redes virtuales, storages y hosts.

*Manejadores:* Conjunto de módulos de tecnologías que pueden añadir o conectar al núcleo, como por ejemplo los servicios cloud y las tecnologías de monitorización.

### **Proxmox**

Es capaz de crear y administrar máquinas virtuales basadas en plantillas, permite realizar virtualización por contenedores mediante OpenVZ o completa con KVM. Incorpora la utilidad DAB (Debian Appliance Builder) para construir plantillas y almacenarlas.

- Tiene una interfaz web de gestión centralizada muy sencilla de utilizar.
- Basado en Debian: se puede administrar por SSH e instalar cualquier paquete compilado para Debian.
- Usuario y contraseña para acceder a todos los nodos y todas las MVs.
- Los discos duros de las MVs pueden ser volúmenes de una partición (LVM) o ficheros como VMWare. LVM tiene mejor rendimiento I/O.
- Soporta varias funcionalidades avanzadas sin adquirir ninguna licencia adicional:
  - Migración en vivo de MVs entre nodos físicos.
  - Clustering de servidores y distribuye carga de trabajo entre ellos.
  - Backups automáticos en caliente de las MVs.
  - Posibilidad de conectar a un NAS/SAN con NFS, iSCSI.

### **Instalación**

La instalación es sencilla dado que se realiza en pocos pasos mediante una interfaz gráfica muy intuitiva. No obstante, por defecto no se pregunta nada del esquema de particionamiento que utilizará. La forma en que se particiona por defecto resulta muy ineficiente, dado que se asigna mucho espacio al directorio raíz y a la swap, espacio que normalmente no se utiliza.

No obstante, como el particionamiento se lleva a cabo utilizando LVM, es posible luego redimensionar esas particiones; sin embargo, es más sencillo escoger el tamaño al momento de la instalación para luego no tener que hacer ningún cambio. Para ello, cuando Proxmox arranca desde el “cdrom” la primera vez se queda esperando con un “prompt” que nos permite pasarle algún parámetro.

### **2.3 Descripción de los sistemas de almacenamiento con tecnologías libres utilizados.**

#### **OpenFiler:**

Es una distribución basada en CentOS, con instalación sencilla (“cdrom” y sencillo asistente). La configuración se realiza desde entorno Web. Es la distribución más utilizada en entornos no

empresariales. Se puede configurar dos máquinas para tener replicación de datos, o incluso alta disponibilidad, pero de forma complicada desde el entorno de líneas de comando.

La autenticación de los directorios de red soporta NIS, LDAP con soporte SMB/CIFS con cifrado de contraseñas, Active Directory (en modo nativo y mixto) y Kerberos 5. Podemos utilizar la autenticación LDAP local, remota, o aprovechar el Active Directory. Podemos compartir mediante Samba hacia la red local. Disponemos además de NFS si lo necesitásemos, y WebDAV para los clientes remotos.

Permite almacenamiento iSCSI, RAID por hardware y LVM (raid por software). Se puede configurar mediante una interfaz web amigable. A base de clic podemos montar RAID 0, 1, 5 y 10 mediante software. Permite establecer cuotas por grupo y usuario individualmente o varios a la vez. Se instala en un servidor físico o virtual.

OpenFiler cubre los aspectos claves del almacenamiento de datos:

*Fiabilidad:* soporta raid software y hardware con utilidades para monitorización y alertas, snapshots de volúmenes y recuperación.

*Disponibilidad:* soporta clúster activo/pasivo de alta disponibilidad, MPIO y replicación a nivel de bloques.

*Rendimiento:* el “kernel” de Linux 2.6 soporta las últimas CPU, hardware de red y almacenamiento.

*Escalabilidad:* escalabilidad del sistema de archivos mayor de 60TB y soporte para hacer crecer el sistema de archivos y los volúmenes en caliente.

## FreeNas

Esta distribución está basada en la licencia BSD. Tiene soporte nativo para Zettabyte File System (ZFS). No tiene posibilidad de configurar alta disponibilidad.

- Su tamaño es prácticamente mínimo: 32 MB, lo cual es insignificante.
- El proceso de instalación es muy sencillo, aunque sólo en inglés.
- Funciona en equipos antiguos con recursos técnicos mínimos.
- La administración del sistema operativo se hace remotamente vía web, incluso, lo puedes hacer desde otra localidad.
- No necesitas tener conectado monitor ni teclado para su operación.
- Compatible con los siguientes protocolos: SMB/CIFS, AFP, NFS, FTP, SCP.
- Servicios extras: cliente de BitTorrent, servidor UPnP, iTunes y Web.
- Compatible con los siguientes sistemas de archivos: NTFS, FAT, UFS, Ext2/3.
- En cuanto al tema de redes, compatible con: 802.11 Wi-Fi, Wake On Lan, LAN.
- Puede ser instalado en un disco duro, memoria flash, o bootearse desde el LiveCD.

## NexentaStor

Sobre la base de un núcleo OpenSolaris, constituye un sistema de almacenamiento profesional. Dispone de dos versiones: Community y Enterprise. Cabinas NAS y SAN. Basado en el sistema de ficheros abierto ZFS. Entre otras dispone de las siguientes características:

- Autoanálisis del sistema, autocorrección de errores.
- Soporte de ilimitados snapshots, almacenamiento ilimitado.
- Gestión de MVs integrada en sistema (realización de copias de seguridad de máquinas de forma instantánea a través del sistema de snapshots).
- Acceso a datos a través de: FTP, CIFS, Web-based Distributed Authoring and Versioning (WebDav), ISCSI, RSYNC, NFS.
- Integración con sistemas de directorio activo (Active Directory) y LDAP.
- Gestión avanzada de errores con sistema de auditoría y notificación al administrador.
- Deduplicación de datos automatizada (ahorro de espacio en ficheros duplicados).
- Elevado rendimiento gracias a la gestión inteligente de caches de lectura y escritura.
- Creación y gestión de volúmenes ISCSI y Canal de Fibra (con opción de multipathing).
- Posibilidad de instalación en sistemas de alta disponibilidad y replicación remota
- Generación automatizada de informes de sistema, posibilidad de análisis avanzado del sistema (Dtrace).
- Sistema abierto: se adapta a la mayor parte del hardware existente en la actualidad (no requiere de sistemas propietarios).

### 2.4 Tipos de pruebas existentes.

#### Prueba de usabilidad

Prueba orientada a factores humanos, estéticos y consistencia en la interfaz de usuario. “Se determina además la calidad de la experiencia de un usuario en la forma en la que éste interactúa con el sistema, se considera la facilidad de uso y el grado de satisfacción del usuario”. [35]

#### Prueba de carga

Se usa para “validar y valorar la aceptabilidad de los límites operacionales de un sistema bajo carga de trabajo variable, mientras el sistema bajo prueba permanece constante. La variación en carga es simular la carga de trabajo promedio y con picos que ocurre dentro de tolerancias operacionales normales”. [36]

### **Prueba de rendimiento o respuesta**

Monitorea “el tiempo en el flujo de ejecución, acceso a datos, en llamada a funciones y sistema para identificar y direccionar los cuellos de botellas y los procesos ineficientes. Compara el rendimiento de un elemento nuevo o desconocido a uno de carga de trabajo de referencia conocido. Determinan los tiempos de respuesta, el espacio que ocupa el módulo en disco o en memoria, el flujo de datos que genera a través de un canal de comunicaciones”. [37]

### **Prueba de estrés**

Su objetivo principal “es identificar y documentar las condiciones bajo las cuales el sistema falla. Verifica que el sistema funciona apropiadamente y sin errores, bajo condiciones de stress”: memoria baja o no disponible, servicios y hardware no disponible, máximo número de clientes conectados o simulados, recursos compartidos no disponible y múltiples usuarios desempeñando la misma transacción con los mismos datos. [38]

### **Prueba de volumen**

Prueba para verificar el manejo de “grandes cantidades de datos, tanto en entrada como en salida, o residente en la base de datos. Puede incluir un procedimiento que indique el uso de consultas que devuelvan todo el contenido de la base de datos, o cuando la cantidad de datos de entrada excede a la cantidad establecida de cada campo”. [39]

### **Prueba de seguridad**

Prueba para “asegurar que los datos o sistemas que son objetos de prueba, son accedidos sólo por los actores que tienen permiso para hacerlo”. [40]

La gran variedad de pruebas que existen están enfocadas a diferentes niveles de una aplicación y en diferentes momentos en la elaboración o explotación de una solución. Por tanto, es necesario saber seleccionar de forma acertada las pruebas más viables para cada entidad, acorde a sus prioridades y objetivos.

## **2.5 Herramientas utilizadas en las pruebas automáticas.**

La automatización de las pruebas, establecen la utilización de herramientas que prueban el sistema sin la interacción constante de las personas, permiten una disminución de tiempo, esfuerzo y gasto de recursos, y se obtiene una mayor cobertura del software a probar. Todo esto implica lograr una selección factible de acuerdo al ambiente de desarrollo de trabajo y a las pruebas de software que se quieran automatizar.

## JMeter

Es una herramienta Java desarrollada dentro del proyecto Jakarta, que permite realizar pruebas de rendimiento y pruebas funcionales sobre aplicaciones web. Permite realizar pruebas web clásicas, pero también realizar test de FTP, JDBC, JNDI, LDAP, SOAP/XML-RPC, y Web Services. Puede ejecutar pruebas de rendimiento distribuidas entre distintos ordenadores. Posibilita activar o desactivar una parte del test, muy útil cuando se está desarrollando un test largo, y se desea deshabilitar ciertas partes iniciales que sean muy pesadas o largas. Posee la variante de generar un caso de prueba a través de una navegación de usuario.

Como herramienta de prueba dispone de varios componentes que facilitan la elaboración de los escenarios de prueba, y permite simular para cada uno de los escenarios miles de usuarios. [41]

“Se caracteriza por sus funcionalidades para realizarle pruebas de estrés, carga y rendimiento a las aplicaciones de manera independiente, al permitir aislar los subsistemas de la aplicación. Además puede ser configurable porque permite definir la cantidad de usuarios que va a simular y las pruebas que se le aplican al marco de trabajo”. [42]

- Se comporta como un proxy local para que a través de él transite toda la información en tiempo real.
- Permite analizar su rendimiento, dadas determinadas condiciones como la navegabilidad en el sistema.
- Numera los errores ocurridos durante la transferencia de datos del servidor al navegador, muchos de estos errores no son visibles para el cliente pero pueden provocar dificultades a corto o largo plazo.
- El muestreo de la información a través de tablas y gráficas facilita la comprensión del flujo de datos y sus características.
- Además brinda información sobre el tiempo de servicio al sistema, el por ciento de rendimiento, el tráfico de información entre el cliente y el servidor tanto los correctos como los incorrectos.

## HardInfo

Este programa ofrece una completa información sobre los componentes de nuestro ordenador, además de realizar pruebas para comprobar el rendimiento de la CPU.

## IPerf

Es un programa cliente-servidor muy sencillo que permite medir la velocidad máxima que alcanzan dos ordenadores conectados en red local.

## IOzone

Es una herramienta destinada a comprobar el rendimiento de un sistema de archivos. La aplicación genera y mide una gran cantidad de operaciones sobre ficheros, entre otras: escritura, re-escritura, lectura, re-lectura, lectura/escritura aleatoria.

### 2.6 Escenarios de prueba.

Se realizaron pruebas de usabilidad, de carga y de rendimiento, a las tecnologías de virtualización y a los sistemas de almacenamiento, en el entorno de prueba definido (**Ver Anexo # 1**).

### Pruebas de usabilidad

Se analiza el equilibrio en la interfaz de usuario, la facilidad de uso y la calidad en la interacción con el usuario mientras ocurren los procesos de instalación y de administración de los sistemas.

Soluciones de virtualización
<b>Ovirt:</b> Su instalación es compleja, presenta un buen sistema de administración que permite la administración centralizada de la plataforma.
<b>OpenNode:</b> Fácil instalación y sencillo de administrar, su interfaz para la administración permite configurarlo de manera eficaz una vez instalado.
<b>OpenNebula:</b> Es un sistema muy complejo de instalar, se requiere amplio conocimiento de la distribución en la cual se instalará porque es necesario conocer cuáles son las dependencias y como instalarlas. Su interfaz de administración es muy buena y una vez instalado es sencillo de administrar.
<b>Proxmox:</b> Es muy sencillo de instalar, su asistente de instalación guía al usuario paso a paso, una vez instalado se puede acceder a su interfaz de administración que da la posibilidad al usuario de administrar todo el sistema de manera fácil.
Sistemas de almacenamientos
<b>Openfiler, Nexenta y Freenas:</b> Poseen un asistente de instalación muy sencillo que guía al usuario paso a paso durante la instalación, sus interfaces de administración una vez instalados es sencilla, el cual permite la administración de los diferentes servicios que poseen.

### Pruebas de carga y rendimiento.

**JMeter**

<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>- Se instala una MV con ubuntu sobre XEN y otra sobre KVM.</li> <li>- Se instala la MV de java.</li> <li>- Se busca la aplicación a someter a las pruebas: redmine.</li> <li>- Se configura el JMeter como servidor proxy.</li> <li>- Se configura el navegador como localhost y se inicia el servidor.</li> <li>- Se inicia la navegación por el redmine.</li> <li>- Se registra todo el tráfico desde el servidor hacia el cliente.</li> <li>- Se guarda los resultados de la navegación.</li> </ul>
<b>Entrada</b>	<ul style="list-style-type: none"> <li>- Se seleccionan la cantidad de usuarios a simular: 250.</li> <li>- Se especifica el tiempo de conexión en cada MV: 1 minuto.</li> </ul>
<b>Procedimientos</b>	<ul style="list-style-type: none"> <li>- Se especifican los “listeners” para registrar los datos acorde a los objetivos y se comienzan las pruebas.</li> </ul>
<b>Poscondiciones</b>	<ul style="list-style-type: none"> <li>- Los “listeners” registran el mismo tráfico previamente realizado con la cantidad de usuarios definidos.</li> </ul>
<b>Salida</b>	<ul style="list-style-type: none"> <li>- En los listener las tablas y las gráficas muestran el tiempo de respuesta del sistema bajo las condiciones configuradas.</li> <li>- Se muestran los componentes enviados desde el servidor así como los errores que se produjeron.</li> <li>- Informa el porcentaje de rendimiento.</li> </ul>
<b>Análisis de resultados</b>	<ul style="list-style-type: none"> <li>- Permite determinar si los tiempos de respuesta son los esperados bajo determinadas condiciones.</li> <li>- Muestra si la aplicación está trabajando con todas sus funcionalidades de forma correcta.</li> <li>- Los resultados en porcentajes muestra si los componentes operan de forma correcta.</li> </ul>

Tabla 1. Descripción del escenario para pruebas de carga y rendimiento con JMeter.

**Análisis de los resultados con KVM.**

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Through...	KB/sec
/redmine/projects/linux-project	500	614	201	1844	89	3843	0.00%	3.9/sec	18.4
/redmine/javascripts/prototype.js	1000	61	44	103	18	1813	0.00%	7.5/sec	1022.2
/redmine/javascripts/effects.js	1000	14	6	13	5	1736	0.00%	7.7/sec	289.5
/redmine/stylesheets/application.css	1000	15	5	11	4	1482	0.00%	7.8/sec	232.8
/redmine/javascripts/application.js	1000	10	3	5	2	1650	0.00%	7.4/sec	40.2
/redmine/javascripts/controls.js	1000	20	6	14	5	1474	0.00%	7.5/sec	253.7
/redmine/stylesheets/jstoolbar.css	1000	14	2	5	1	1710	0.00%	7.5/sec	14.1
/redmine/javascripts/dragdrop.js	1000	15	6	11	4	1476	0.00%	7.9/sec	238.8
/redmine/projects/linux-project/issues/new	500	844	225	2641	26	5440	50.00%	5.5/sec	17.6
/redmine/javascripts/calendar/calendar.js	500	24	12	35	7	1465	0.00%	5.1/sec	244.3
/redmine/javascripts/calendar/lang/calendar-en.js	500	12	3	7	2	1707	0.00%	4.9/sec	17.1
/redmine/javascripts/jstoolbar/jstoolbar.js	750	10	3	6	2	1486	0.00%	7.1/sec	80.1
/redmine/javascripts/calendar/calendar-setup.js	500	7	3	6	2	1396	0.00%	5.0/sec	43.1
/redmine/javascripts/jstoolbar/lang/jstoolbar-en.js	750	7	2	5	1	1465	0.00%	7.0/sec	4.9
/redmine/stylesheets/calendar.css	500	9	3	5	1	1518	0.00%	5.3/sec	25.4
/redmine/stylesheets/scm.css	500	12	3	7	1	1469	0.00%	5.1/sec	30.3
/redmine/javascripts/jstoolbar/textile.js	750	13	3	6	1	1701	0.00%	7.3/sec	29.8
/redmine/issues/4	250	831	495	1991	101	3457	0.00%	3.3/sec	18.5
TOTAL	13000	88	5	91	1	5440	1.92%	94.3/sec	2381.9

Figura 10. Agregación de la máquina virtual 01 en KVM.

Resultados esperados	Resultados obtenidos
Tiempos de respuesta bajo determinadas condiciones.	El tiempo de respuesta promedio obtenido es de 88 ms. Bajo estas condiciones es un resultado positivo.
Trabajo con todas sus funcionalidades de forma correcta.	Según los resultados todas las funcionalidades trabajan correctamente.
Los componentes operan de forma correcta.	El % de errores es muy bajo, se opera de forma correcta. Los 50 % de errores de una aplicación es un problema de latencia.
Cantidad de peticiones / segundos.	La cantidad de peticiones /segundos en KVM fue de <b>94.3</b> .
Kb/s que el servidor procesa.	La cantidad de Kb/s procesado en XEN fue de <b>2381.9</b> .

Tabla 2. Análisis de los resultados de la máquina virtual 01 en KVM.

### Análisis de los resultados con XEN.

Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Through...	KB/sec
/redmine/projects/linux-project	500	624	221	1746	82	3287	0.00%	3.8/sec	17.9
/redmine/javascripts/prototype.js	1000	69	59	131	19	804	0.00%	7.3/sec	994.7
/redmine/javascripts/effects.js	1000	14	8	28	5	205	0.00%	7.5/sec	281.3
/redmine/stylesheets/application.css	1000	13	7	20	4	668	0.00%	7.6/sec	227.6
/redmine/javascripts/application.js	1000	6	4	9	2	689	0.00%	7.1/sec	39.0
/redmine/javascripts/controls.js	1000	15	8	28	5	686	0.00%	7.3/sec	247.6
/redmine/stylesheets/jstoolbar.css	1000	7	3	8	1	758	0.00%	7.2/sec	13.7
/redmine/javascripts/dragdrop.js	1000	12	7	23	5	665	0.00%	7.6/sec	231.5
/redmine/projects/linux-project/issues/new	500	812	239	2476	24	5214	50.00%	5.4/sec	17.2
/redmine/javascripts/calendar/calendar.js	500	27	18	49	8	789	0.00%	5.0/sec	238.6
/redmine/javascripts/calendar/lang/calendar-en.js	500	6	3	11	2	437	0.00%	4.7/sec	16.4
/redmine/javascripts/jstoolbar/jstoolbar.js	750	8	4	10	2	638	0.00%	6.7/sec	74.8
/redmine/javascripts/calendar/calendar-setup.js	500	8	4	10	2	664	0.00%	4.7/sec	40.2
/redmine/javascripts/jstoolbar/lang/jstoolbar-en.js	750	4	3	8	1	136	0.00%	6.6/sec	4.6
/redmine/stylesheets/calendar.css	500	6	3	9	2	267	0.00%	4.9/sec	23.6
/redmine/stylesheets/scm.css	500	4	3	9	2	52	0.00%	4.8/sec	28.6
/redmine/javascripts/jstoolbar/textile.js	750	7	3	8	1	762	0.00%	7.0/sec	28.4
/redmine/issues/4	250	840	497	1959	93	3348	0.00%	3.2/sec	17.7
TOTAL	13000	85	7	99	1	5214	1.92%	91.6/sec	2315.9

Figura 11. Agregación de la máquina virtual 01 en XEN.

Resultados esperados	Resultados obtenidos
Tiempos de respuesta bajo determinadas condiciones.	El tiempo de respuesta promedio obtenido es de 85 ms. Bajo estas condiciones es un resultado positivo.
Trabajo con todas sus funcionalidades de forma correcta.	Según los resultados todas las funcionalidades trabajan correctamente.
Los componentes operan de forma correcta.	El % de errores es muy bajo, se opera de forma correcta. Los 50 % de errores de una aplicación es un problema de latencia.
Cantidad de peticiones / segundos.	La cantidad de peticiones / segundos en KVM fue de 91.6.
Cantidad de Kb /segundos.	La cantidad de Kb/s procesado en XEN fue de 2315.9.

Tabla 3. Análisis de los resultados de la máquina virtual 01 en XEN.

Puede apreciarse en los valores, las mínimas diferencias entre las tecnologías KVM y XEN.

**HardInfo**

<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>- Se instala una máquina virtual con ubuntu sobre XEN.</li> <li>- Se instala una máquina virtual con ubuntu sobre KVM.</li> <li>- Se instala la aplicación HardInfo.</li> </ul>
<b>Entrada</b>	<ul style="list-style-type: none"> <li>- Se seleccionan las pruebas a realizar: CPU Blowfish, CPU CryptoHash, CPU Fibonacci, CPU N-Queens, FPU FFT, FPU Raytracing.</li> </ul>
<b>Procedimientos</b>	<ul style="list-style-type: none"> <li>- Se generan las pruebas.</li> </ul>
<b>Poscondiciones</b>	<ul style="list-style-type: none"> <li>- El sistema realiza los cálculos de los algoritmos definidos en el software.</li> </ul>
<b>Salida</b>	<ul style="list-style-type: none"> <li>- El sistema ofrece los valores de los resultados de los algoritmos definidos en el software</li> </ul>
<b>Análisis de resultados</b>	<ul style="list-style-type: none"> <li>- Permite determinar el rendimiento del procesador mediante los algoritmos definidos.</li> <li>- Exceptuando los resultados de CPU CryptoHash, a menor valor mejor resultado.</li> </ul>

Tabla 4. Descripción del escenario para pruebas de rendimiento con HardInfo.

**Análisis de los resultados con KVM y XEN.**

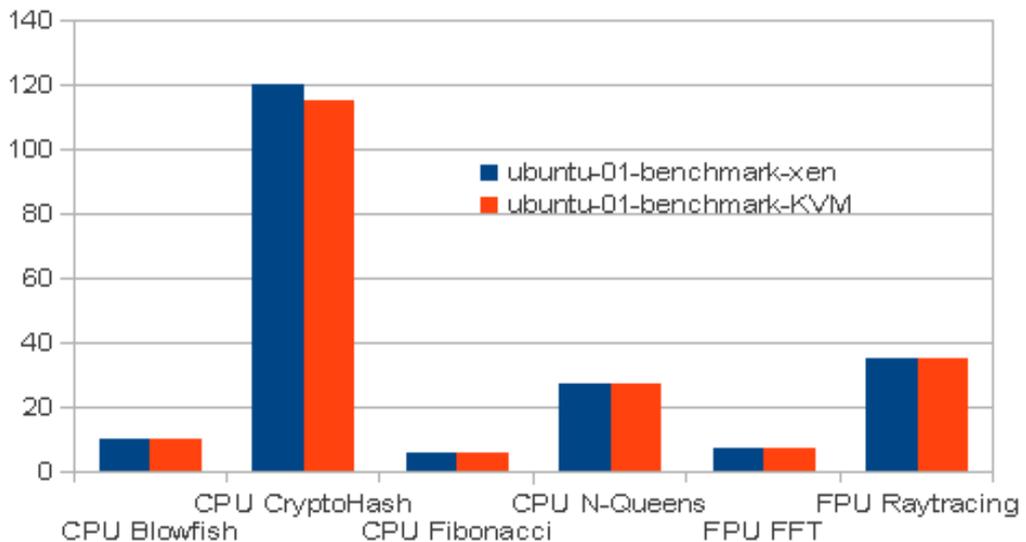


Gráfico 1. Resultados de las pruebas de rendimiento con HardInfo.

En el gráfico podemos apreciar la similitud en el rendimiento entre KVM y Xen, en el cálculo de los algoritmos definidos por HardInfo.

**Pruebas de rendimiento de red.**

**Iperf**

<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>- Se instala una máquina virtual con ubuntu sobre XEN.</li> <li>- Se instala una máquina virtual con ubuntu sobre KVM.</li> <li>- Se instala la herramienta Iperf en las dos máquinas virtuales.</li> </ul>
<b>Entrada</b>	<ul style="list-style-type: none"> <li>- Se especifica un nodo que funcionará como servidor.</li> <li>- Se especifica un nodo que funcionará como cliente.</li> </ul>
<b>Procedimientos</b>	<ul style="list-style-type: none"> <li>- Se ejecuta el comando iperf -s -p 12345 para definir el servidor.</li> <li>- Se ejecuta el comando iperf -c ip_servidor -p 12345 para definir el cliente.</li> <li>- Se inician las pruebas.</li> </ul>
<b>Poscondiciones</b>	<ul style="list-style-type: none"> <li>- Se establece un tráfico entre los nodos cliente y servidor.</li> </ul>
<b>Salida</b>	<ul style="list-style-type: none"> <li>- Se muestran los resultados obtenidos del tráfico generado entre los nodos cliente y servidor.</li> </ul>
<b>Análisis de resultados</b>	<ul style="list-style-type: none"> <li>- Permite determinar el ancho de banda máximo disponible entre los nodos cliente y servidor.</li> <li>- Posibilita detectar si hay congestión en la red.</li> </ul>

Tabla 5. Descripción del escenario para pruebas de rendimiento de red con Iperf.

**Análisis de los resultados en KVM y XEN.**

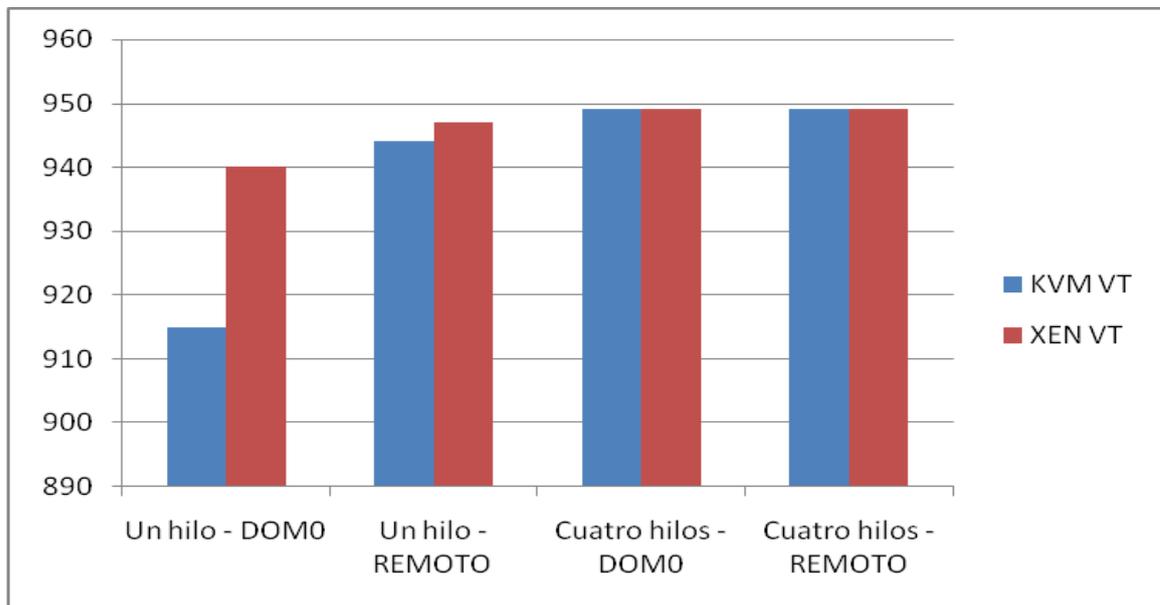


Gráfico 2. Resultados de las pruebas de rendimiento de red con Iperf (Mb/s).

Xen tiene un mejor rendimiento que KVM cuando se transmite un único hilo de información, cuando se transmiten varios hilos simultáneamente el rendimiento es similar.

**IOzone**

<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>- Se instala Freenas, Nexenta y OpenFiler y se configuran para utilizar protocolo ISCSI y NFS.</li> <li>- Se instala una MV con ubuntu sobre XEN y otra sobre KVM.</li> <li>- Se instala iozone en las dos MVs.</li> </ul>
<b>Entrada</b>	<ul style="list-style-type: none"> <li>- Se especifica el tamaño de los ficheros a copiar: 100 Mb.</li> <li>- Se especifica la cantidad de hilos simultáneos: 5 hilos.</li> <li>- Se especifica el tamaño del bloque de los ficheros: 4k.</li> </ul>
<b>Procedimientos</b>	<ul style="list-style-type: none"> <li>- Se ejecuta el comando iozone -R -l 5 -u 5 -r 4k -s 100m en las máquinas virtuales almacenadas localmente en KVM y en XEN.</li> <li>- Se ejecuta el comando iozone -R -l 5 -u 5 -r 4k -s 100m en los sistemas de almacenamiento (Freenas, Nexenta, OpenFiler).</li> </ul>
<b>Poscondiciones</b>	<ul style="list-style-type: none"> <li>- Se establece un tráfico local entre la MV y el disco duro local del virtualizador KVM y otro hacia XEN.</li> <li>- Se establece un tráfico local en los sistemas de almacenamiento (Freenas, Nexenta, OpenFiler).</li> </ul>
<b>Salida</b>	<ul style="list-style-type: none"> <li>- Se muestran en consola la velocidad de lectura, re-lectura, escritura, re-escritura y acceso aleatorio entre la máquina virtual y el disco duro local del virtualizador KVM y en otra consola el de XEN.</li> <li>- Se muestran en consola la velocidad de lectura, re-lectura, escritura, re-escritura y acceso aleatorio en los discos duros locales de los sistemas de almacenamiento (Freenas, Nexenta, OpenFiler).</li> </ul>
<b>Análisis de resultados</b>	<ul style="list-style-type: none"> <li>- Permite determinar la efectividad de los sistemas de almacenamiento con protocolo ISCSI y NFS en las operaciones definidas.</li> <li>- Permite establecer comparaciones entre los diferentes sistemas de almacenamiento (Freenas, Nexenta, OpenFiler).</li> <li>- Permite establecer comparaciones entre XEN y KVM.</li> <li>- Permite conocer el rendimiento integral en el acceso a disco duro tanto en los virtualizadores como en los sistemas de almacenamiento.</li> </ul>

Tabla 6. Descripción del escenario para pruebas de rendimiento con IOzone.

### Análisis de los resultados de KVM y XEN.

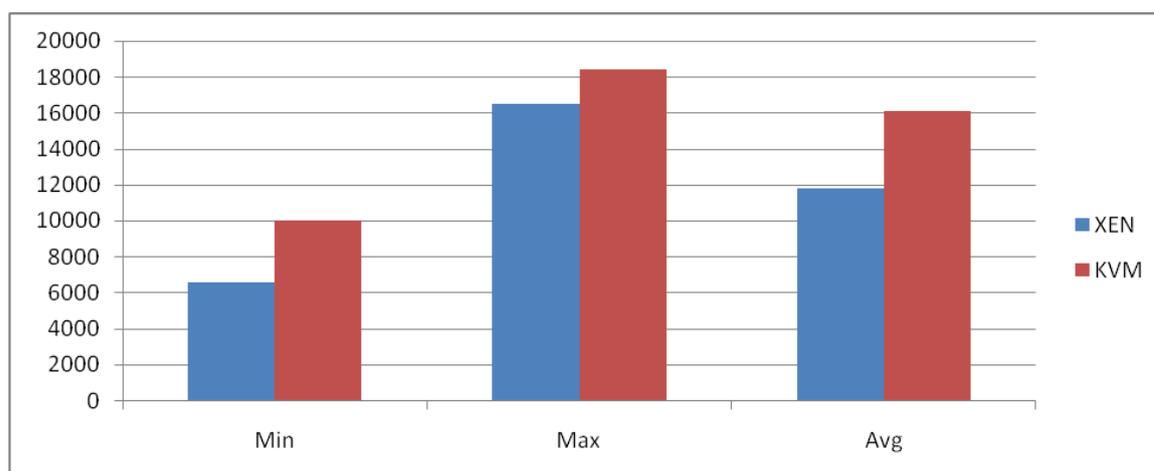


Gráfico 3. Escritura en KVM y XEN de forma local (Kb/s).

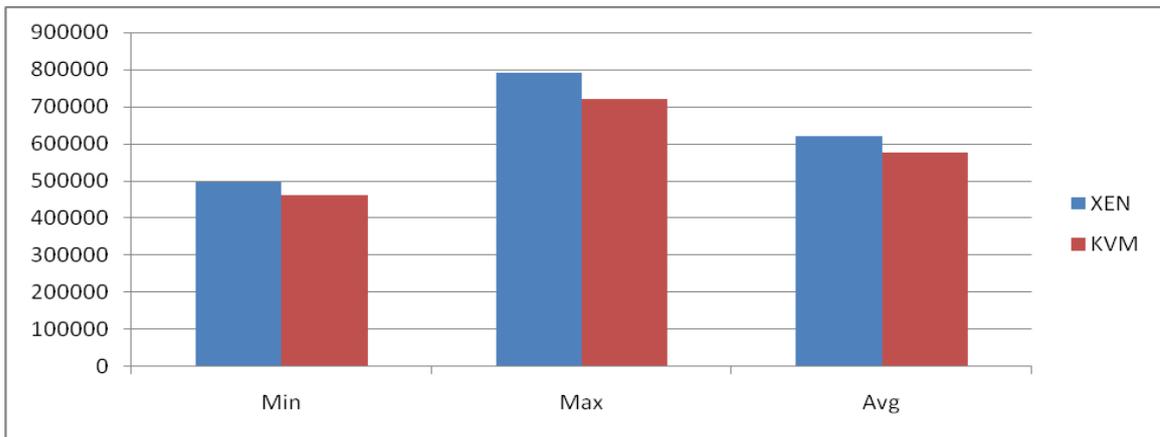


Gráfico 4. Lectura en KVM y XEN de forma local (Kb/s).

Es más eficiente KVM realizando operaciones de escritura mientras XEN es en la lectura, la diferencia no es significativa, pero la escritura es el criterio más importante para nosotros.

**Análisis de los resultados de Freenas, Nexenta y OpenFiler.**

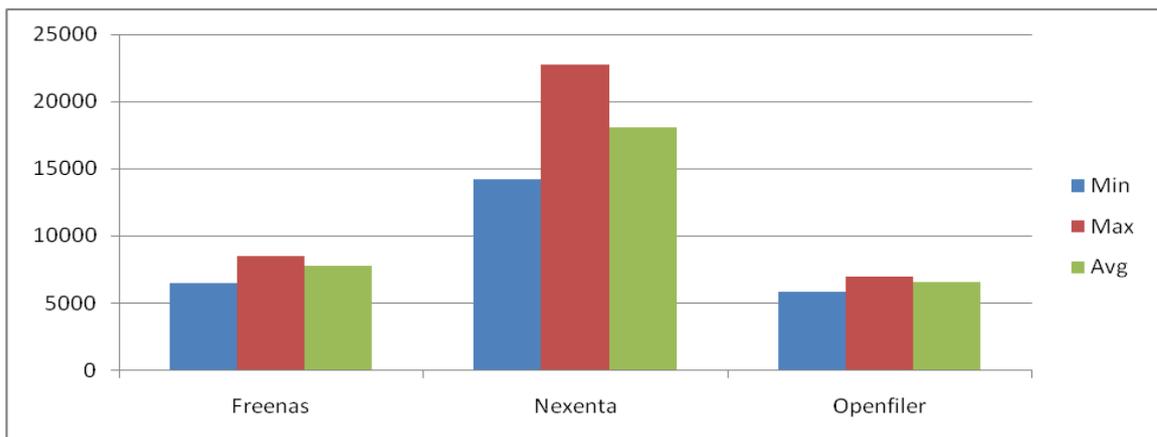


Gráfico 5. Escritura en Freenas, Nexenta y OpenFiler usando ISCSI (Kb/s).

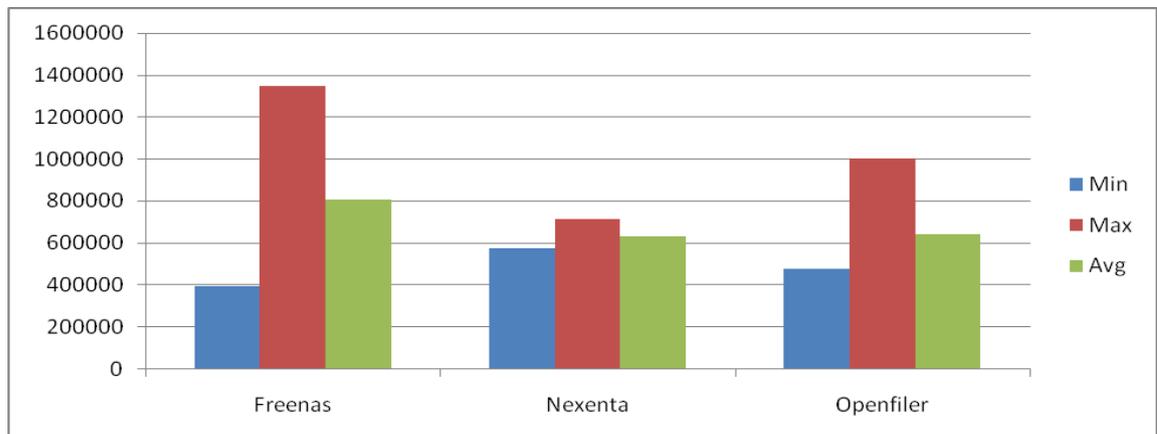


Gráfico 6. Lectura en Freenas, Nexenta y OpenFiler usando ISCSI (Kb/s).

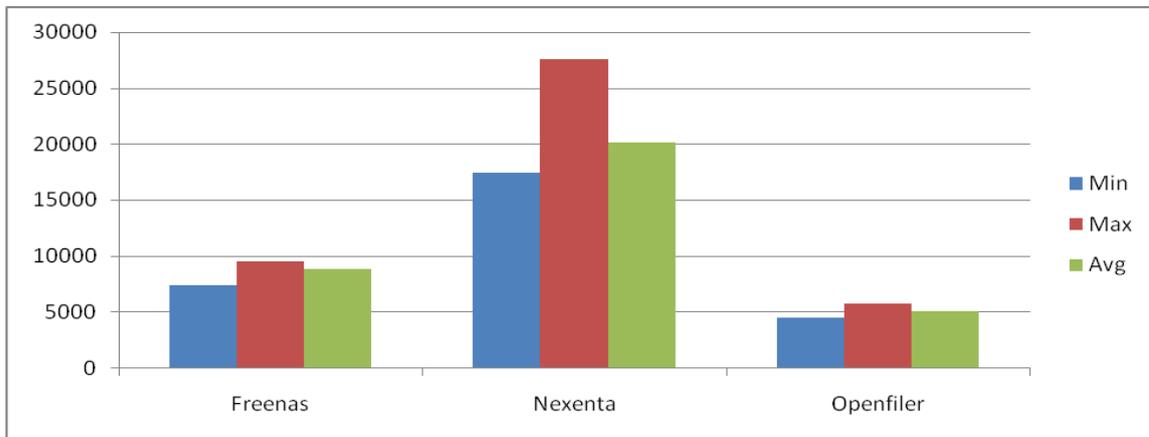


Gráfico 7. Escritura en Freenas, Nexenta y OpenFiler usando NFS (Kb/s).

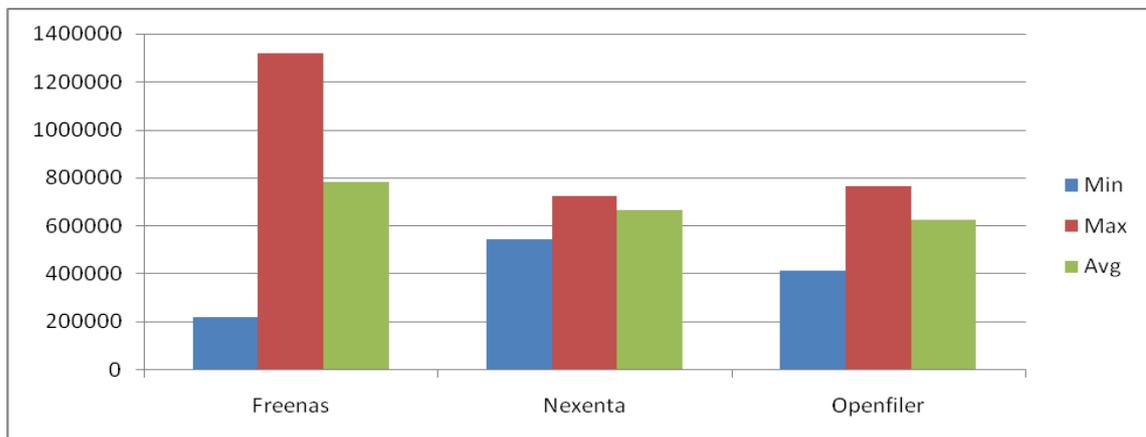


Gráfico 8. Lectura en Freenas, Nexenta y OpenFiler usando NFS (Kb/s).

Usando los protocolos ISCSI y NFS es Nexenta el más eficiente para la escritura, mientras que Freenas lo es en la lectura, en nuestro entorno de trabajo el criterio de escritura es el más importante.

### 2.7 Metodología de análisis: la prospectiva tecnológica.

La Organización para la Cooperación y el Desarrollo Económicos (OCDE) define la prospectiva tecnológica como el “conjunto de tentativas sistemáticas para observar a largo plazo el futuro de la ciencia, la tecnología, la economía y la sociedad con el propósito de identificar las tecnologías emergentes que probablemente produzcan los mayores beneficios económicos o sociales”. [43]

Es un “proceso de anticipación y exploración de la opinión experta proveniente de redes de personas e instituciones del gobierno, la empresa y las universidades en forma estructurada, interactiva, participativa, coordinada y sinérgica para construir visiones estratégicas de la ciencia y la tecnología y su papel en la competitividad y el desarrollo de un país, territorio, sector económico, empresa o institución pública”. [44]

### El modelo prospectivo

El análisis prospectivo tiene como punto de máximo de interés el diseño de un escenario probable y de varios escenarios alternos. El trabajo con los escenarios supone conocer las variables que los integran. Es indispensable realizar una fase previa para identificar los elementos de que constarán los escenarios. Asimismo, como el diseño de escenarios tiene como propósito elegir el más útil y provechoso, la etapa siguiente será la determinación de estrategias encaminadas a alcanzarlo. A lo anterior hay que añadir, en su momento, el conocimiento de los intereses, alianzas y conflictos de los actores implicados.

La prospectiva ha sido experimentada en las condiciones de América Latina y ha dado lugar a adaptaciones y validaciones, que en algunos aspectos lo hacen diferente de las propuestas americanas y europeas, sin traicionar los elementos esenciales.

La siguiente tabla ofrece una visión general del modelo prospectivo con sus resultados y referencias a las técnicas que se pueden emplear en cada caso. [45] Por el número reducido de nuestros actores, no se hace el análisis correspondiente.

Etapas	Resultados	Técnicas
1. Precisión de tendencias, factores de cambio y características del entorno.	Reconocimiento de la situación actual y de las condiciones potenciales del tema que se está estudiando.	- Matriz Dafo - Árbol de Competencias de Marc Giget.
2. Identificación de "variables estratégicas".	Detección de los componentes más importantes y más gobernables del tema.	- Análisis Estructural - Ábaco de Régnier
3. Detección del comportamiento de los actores.	Descripción del poder que manejan y de las posibles jugadas de los actores.	- Juego de Actores
4. Estimación y Diseño de los Escenarios.	Obtención de un escenario probable, de varios escenarios alternos y de un "escenario apuesta".	- Delphi - Ábaco de Régnier - Sistema de Matrices de Impacto Cruzado - Análisis Morfológico - Ejes de Peter Schwartz
5. Estrategias para lograr el escenario apuesta.	Diseño de objetivos, detección de acciones.	- Ábaco de Régnier - Análisis Multicriterios - Árboles de Pertinencia

Tabla 7. Etapas y técnicas más importantes del modelo prospectivo.

“Un escenario es un conjunto formado por la descripción de una situación futura y un camino de acontecimientos que permiten pasar de una situación original o otra futura”. [46]

## Tipos de escenarios

Se pueden distinguir principalmente dos tipos de escenarios: los exploratorios y los de anticipación o normativos. Los primeros parten de tendencias pasadas y presentes y conducen a futuros viables. Los segundos son contruidos a partir de imágenes alternativas de futuro, pueden ser deseables o rechazables. Se conciben de un modo retrospectivo.

Ambos escenarios pueden ser tendenciales o contrastados; según se tomen en cuenta las evoluciones más probables o más extremas.

### 2.7.1 Análisis estructural.

El análisis estructural es una herramienta de estructuración de una reflexión colectiva. Tiene por objetivo, hacer aparecer las principales variables influyente y dependientes y por ello las variables esenciales a la evolución del sistema. Asumiendo la perspectiva de Michel Godet, esta etapa comprende dos subfases:

- Fase 1: Listado de las variables.

Variables internas	Variables externas
<b>Escalabilidad:</b> es la capacidad de crecer y adaptarse a los rediseños o reinstalaciones.	Contribución científica de la solución.
<b>Migración:</b> convertir MVs en físicas (y viceversa). Conversión a través de la red, que requiera poca intervención personal. La migración en vivo es la base de las soluciones al balanceo de carga con MVs.	Necesidad de empleo de la solución.
<b>Fiabilidad:</b> soportar raid software / hardware con utilidades para monitorio y alertas, snapshots de volúmenes y recuperación.	Posibilidades de implementación.
<b>Disponibilidad:</b> tener caminos redundantes para acceso a discos. Menor tiempo de recuperación ante desastres.	Ahorro de costos.
<b>Gestión de carga de trabajo:</b> manejar múltiples MVs o equipos anfitriones físicos y reemplazar instancias en caso de fallo.	Impacto en el área.
<b>Rendimiento:</b> velocidad de acceso a datos. Repartir operaciones de I/O por muchos discos y Caching.	
<b>Monitorización:</b> información detallada y en tiempo real del estado y uso de recursos de cada una de las MVs.	
<b>Fácil administración e instalación:</b> capacidad para planificar instancias virtuales, pasando por el almacenamiento y los recursos de red, hasta los parámetros de configuración para hardware virtual y físico, los usuarios y sus privilegios.	
<b>Aprovisionamiento y despliegue:</b> crear el número requerido de instancias virtuales tan rápido como sea posible, con la mínima configuración manual.	
<b>Funcionalidades:</b> Snapshots, Deduplicación, Replicación, Clúster y Multipathing.	

Tabla 8. Variables internas y externas.

- Fase 2: Identificación de las variables claves.

**Escalabilidad:** es la capacidad de crecer y adaptarse a los rediseños o reinstalaciones.

**Disponibilidad:** tener caminos redundantes para acceso a discos. Menor tiempo de recuperación ante desastres.

**Fiabilidad:** soportar raid software / hardware con utilidades para monitorio y alertas, snapshots de volúmenes y recuperación.

**Rendimiento:** velocidad de acceso a datos. Repartir operaciones de IO por muchos discos y Caching.

**Funcionalidades:** “snapshots”, deduplicación, replicación, clúster y “multipathing”.

### 2.7.2 Estimación y diseño de los escenarios.

El proceso se inicia con la definición de un grupo de seis hipótesis surgidas de los encuentros de trabajo que reflejan la visión de los posibles escenarios. Se tienen en cuenta las variables claves y los objetivos asociados.

H<sub>1</sub>: La integración de KVM con FreeNas asegura escalabilidad en la solución integral.

H<sub>2</sub>: La integración de KVM con Nexenta asegura rendimiento en la solución integral.

H<sub>3</sub>: La integración de KVM con OpenFiler asegura fiabilidad en la solución integral.

H<sub>4</sub>: La integración de Xen con FreeNas asegura despliegue en la solución integral.

H<sub>5</sub>: La integración de Xen con Nexenta asegura funcionalidades en la solución integral.

H<sub>6</sub>: La integración de Xen con OpenFiler asegura disponibilidad en la solución integral.

**Escenarios probables:** por la experiencia adquirida en las tecnologías y los sistemas que lo conforman, son los más rápidos de implementar en el más corto plazo.

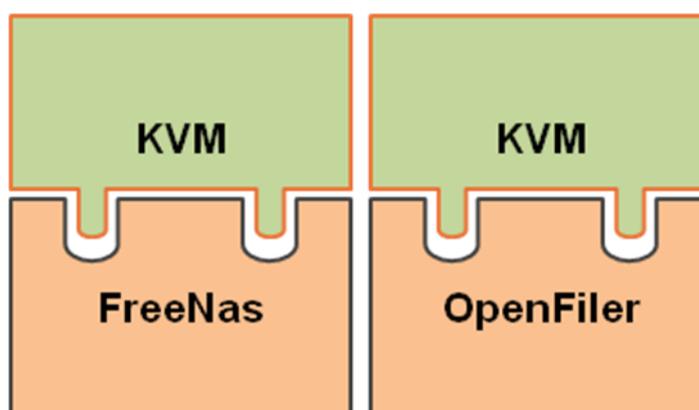


Figura 12. Escenarios probables.

**Escenarios alternos:** pueden implementarse, aún cuando las tecnologías y los sistemas, no hayan sido concebidos a corto plazo, ya sea por su complejidad o por la estrategia de trabajo.

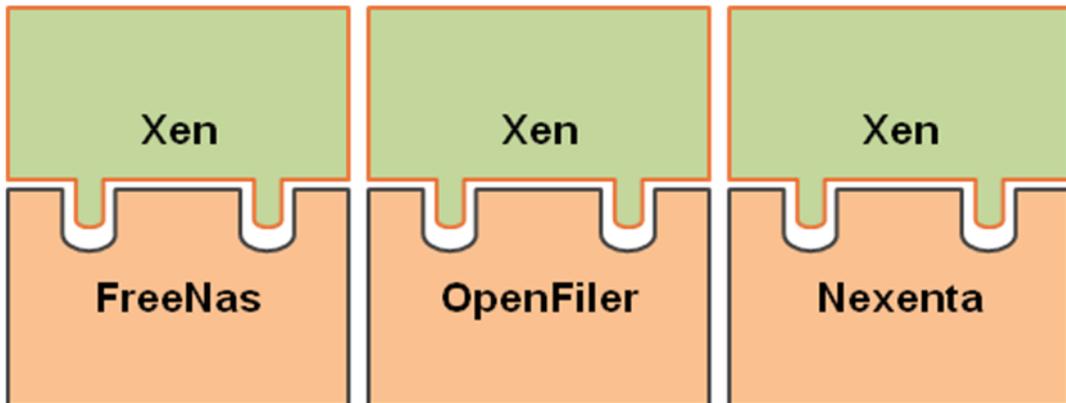


Figura 13. Escenarios alternos.

**Escenario apuesta:** determinará el enfoque que deben seguir las opciones de estrategias de trabajo, teniendo en cuenta los objetivos, los proyectos y el equipo de trabajo. Desglosándose en cada una de las variables estratégicas con la respectiva opción de futuro seleccionada.

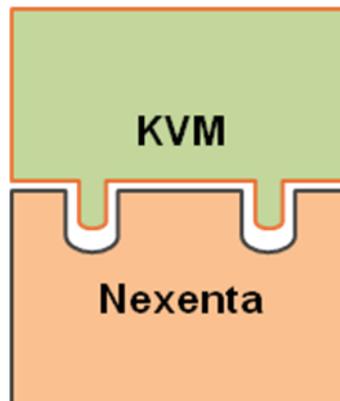


Figura 14. Escenario apuesta.

Usando Proxmox como solución de virtualización con KVM como tecnología base, este escenario nos ofrece el máximo de posibilidades y combinaciones, es una combinación de una solución de virtualización y un sistema de almacenamiento muy extendidos; permitirá encontrar mayor información sobre qué cosas puede hacer y cómo las puede hacer, además de ser más compatible, documentada, de fácil manejo, flexible y con multitud de opciones. Un escenario que sobresalió en las pruebas de rendimiento realizadas. Por su simplicidad de uso y su gran potencia. La posibilidad de utilizar tanto virtualización a nivel de sistema operativo como virtualización completa es una gran característica, así como poder disponer de medios externos y la migración en vivo de las máquinas virtuales. Es cierto que es posible hacer esto manualmente, pero la ventaja es la facilidad para llevarlo a cabo.

Nexenta añade a las características propias de ZFS un conjunto de herramientas de gestión, monitorización y optimización entre las que se encuentran: gestión de ilimitados snapshots y clones, “block level mirroring”, replicación remota a nivel de bloque y fichero, clustering, gestión de LUNs ISCSI (con capacidad de Multipathing), deduplicación de datos, gestión de infraestructuras virtuales (XEN, Hyper-V y Vmware), establecimiento de políticas de mantenimiento de datos, “backups” instantáneos y automatizados. Puede considerarse el mejor escenario para la solución integral teniendo en cuenta las estrategias de trabajo.



Figura 15. Diagrama de la solución.

### 2.7.3 Estrategia para lograr el escenario apuesta.

La estrategia está dirigida a crear las capacidades prospectivas, la superación del personal y la asimilación de las nuevas tecnologías.



Figura 16. Cómo crear las capacidades prospectivas.

## 2.8 Conclusiones parciales.

La prospectiva tecnológica nunca debe ser vista como un ejercicio de profecía o un oráculo infalible, sino como una herramienta de apoyo a la toma de decisiones que puede mejorar sensiblemente el grado de acierto de las mismas. Mejora no debida a la exactitud de sus previsiones si no a proporcionar una referencia de los posibles marcos tecnológicos en el futuro.

Existen diferentes metodologías para la construcción de escenarios dentro de las que se distinguen: el enfoque lógico de Wilson, la previsión humana y social de Medina y el método de Godet.

El escenario apuesta para solución integral ofrece facilidades de trabajo y rendimiento. Las funcionalidades del sistema de almacenamiento, son el complemento para la flexibilidad y las posibilidades de gestión que brinda la solución de virtualización.

## CAPÍTULO 3: IMPLEMENTACIÓN, CONFIGURACIÓN Y EVALUACIÓN DE LA SOLUCIÓN

Se propone una guía de implementación y configuración de la solución integral para un escenario de prueba definido. Se realizó una evaluación técnica de la solución usando la Opinión de Expertos, como método.

### 3.1 Instalación de la solución de virtualización: Proxmox.

1. Se inicia el instalador desde “cdrom” y se acepta la General Public License (GPL).
2. Se selecciona la partición y se selecciona la zona horaria.
3. Se introduce la contraseña y el correo del administrador a utilizar.
4. Se configura la red.
5. Se espera que termine la instalación y se inicia el sistema instalado.

### 3.2 Configuración de la solución de virtualización: Proxmox.

1. Se introduce el usuario y la contraseña del sistema de administración.
2. Se accede a la ventana principal del sistema de administración.
3. Se configura las interfaces de red.



The screenshot shows the Proxmox web interface. The top navigation bar includes 'Home | Logout', 'Proxmox Virtual Environment 1.7', and 'www.proxmox.com'. The left sidebar has three main sections: 'VM Manager' (Virtual Machines, Appliance Templates, ISO Images), 'Configuration' (System, Storage, Backup), and 'Administration' (Server, Logs, Cluster). The main content area is titled 'System Configuration' and has tabs for 'Network', 'DNS', 'Time', 'Admin', and 'Options'. The 'Network' tab is active, showing an 'Interface configuration' table.

Interface	Active	Ports/Slaves	Autostart	IP Address	Subnet Mask	Gateway
eth1	no		<input type="checkbox"/>			
vmbr0	yes	eth0	<input checked="" type="checkbox"/>	10.12.170.94	255.255.255.0	10.12.170.254

4. Se configura un *bonding* para enlaces redundantes.

**VM Manager**

- Virtual Machines
- Appliance Templates
- ISO Images

**Configuration**

- System
- Storage
- Backup

**Administration**

- Server
- Logs
- Cluster

**System Configuration**

Network | DNS | Time | Admin | Options

**Interface configuration**

		Ports/Slaves	Autostart	IP Address	Subnet Mask	Gateway
eth1	no		<input type="checkbox"/>			
vmbr0	yes	eth0	<input checked="" type="checkbox"/>	10.12.170.94	255.255.255.0	10.12.170.254

5. Se selecciona el modo *active-backup* para redundancia si falla la red.

**VM Manager**

- Virtual Machines
- Appliance Templates
- ISO Images

**Configuration**

- System
- Storage
- Backup

**Administration**

- Server
- Logs
- Cluster

**System Configuration**

Network | DNS | Time | Admin | Options

Create bonding device

Bond name:

IP Address:  .  .  .

Subnet Mask:  .  .  .

Autostart:

Slaves:

Mode:

**save**

6. Se configura el *bridge* como elemento principal de la red para que use como *bridge port* el enlace redundante *bond0*.

**VM Manager**

- Virtual Machines
- Appliance Templates
- ISO Images

**Configuration**

- System
- Storage
- Backup

**Administration**

- Server
- Logs
- Cluster

**System Configuration**

Network | DNS | Time | Admin | Options

Configure bridge 'vmbr0'

IP Address:  .  .  .

Subnet Mask:  .  .  .

Gateway:  .  .  .

Autostart:

Bridge Ports:

**save**

7. Se agrega un recurso iSCSI, seleccionando *storage->Add iSCSI Target*, se introduce el IP del sistema de almacenamiento, se selecciona *Scan* y se salvan los cambios.

8. Se muestra la información del recurso de almacenamiento encontrado con *Scan*.

### 3.3 Instalación del sistema de almacenamiento: NexentaStor.

1. Se inicia el instalador desde el "cdrom" y se acepta la licencia EULA.
2. Se selecciona la partición y comienza la instalación hasta que se completa.
3. Se reinicia y carga el sistema instalado. Se introduce el *Machine Signature* en el sitio <http://nexenta.com/register-eval> para recibir el registro.

```

Welcome to ZFS Open Storage Appliance!
-----
Model       : ZFS Storage Appliance (Community Edition)
Software Version : 3.0.4
Release Date  : Sun Oct  3 05:40:40 2010
UUID S/N    : 4230cd0b-89f3-908a-898a-f64bd904ada6

Product registration
-----
Machine Signature : 8HHC1K7JC
Registration Key  : -

Please enter product registration key. To obtain a key, visit
http://www.nexenta.com/register-eval and use the displayed 'Machine
Signature' to register this copy of software.

```

4. Se configura la red, después de haberse registrado correctamente.

```

Welcome to ZFS Open Storage Appliance!
-----
Model       : ZFS Storage Appliance (Community Edition)
Software Version : 3.0.4
Release Date  : Sun Oct  3 05:40:40 2010
UUID S/N    : 4230cd0b-89f3-908a-898a-f64bd904ada6

Product registration
-----
Machine Signature : 8HHC1K7JC
Registration Key  : EVAL-A281690270-8HHC1K7JC-BFHDHL

Thank You!

Configure primary network interface
-----
Interface e1000g0 (PRIMARY): Configured as 192.168.197.100/255.255.0.0 with mtu
1500
Default Gateway           : 192.168.8.2
DNS #1                   :
DNS #2                   :
DNS #3                   :
Reconfigure? (y/n) _

```

5. Se selecciona como acceder al sistema de administración: vía http o https.

```

Gateway IP address : 10.12.170.254
Name Server #1    : 10.12.170.4
Name Server #2    :
Name Server #3    :

Interface e1000g0 (PRIMARY): Configured as 10.12.170.95/255.255.255.0 with mtu 1
500
Default Gateway           : 10.12.170.254
DNS #1                   : 10.12.170.4
DNS #2                   :
DNS #3                   :
Reconfigure? No

Your primary interface is : e1000g0
Web GUI protocol         :
HTTP _ HTTPS

Choose transport protocol for the Web GUI (NMU) interface. Note that you can
always re-run NMC command 'setup appliance init' later to change the setting.
Press 'q' to keep the current setting: HTTP. Navigate with arrow keys (or
hjkl), or Ctrl-C to exit.
NMU interface will be more responsive but unsecured (plain text)

```

### 3.4 Configuración del sistema de almacenamiento: NexentaStor.

1. Se accede al sistema básico de configuración y se siguen los pasos del asistente.

**STEP #1: BASIC CONFIGURATION**

**Host Name**   
 Default appliance's host name. It must begin and end with ASCII letter or digit, and can only contain ASCII letters, digits or hyphens ('-').

**Domain Name**   
 Default appliance's domain name. Domain name is a group of labels delimited by dots ('.'). Labels can only contain ASCII letters, digits, underscores ('\_') and hyphens ('-'). Each label must not have more than 63 characters and must not begin or end with hyphen ('-'). Domain name must contain TLD (top-level domain) part (.com, .net, etc.).

**Time Zone**    
 Geographical time zone for selected continent and country.

**NTP Server**   
 Default NTP server address.

**Keyboard Layout**   
 Default keyboard layout.

**Language**   
 User Interface Language

Copyright 2005-2010 Nexenta Systems, Inc.

2. Se le asigna una contraseña a root, se completa el perfil para el sistema y se salva.
3. Se configura la interfaz de red.
4. Se muestran los discos disponibles.

**STEP #3: DISKS**

**Disks:**

Disk	Device	Type	Size	Volume	Attach	Model
c1t0d0	sd0	disk	8.00 GB	syspool	mpt	VMware, Rev. 1.0
c1t1d0	sd1	disk	8.00 GB		mpt	VMware, Rev. 1.0

Copyright 2005-2010 Nexenta Systems, Inc.

5. Se crean los nuevos volúmenes.

Network
STEP #4: VOLUMES

**iSCSI Initiator**

**Disks**

**Volumes**

Volumes can be created by selecting one or more physical disks connected to the appliance directly or via iSCSI. Volumes can be exported and imported. You may also want to assign a human readable volume description/comment.

**Folders and Shares**

**Review Changes and Exit**

**Volumes:**

No available volumes.

**Operations:**

**Add New Volume**

**Performance considerations**

For mirrored configurations:  
Random read performance scales linearly with the number of disks; write performance scales linearly with the number of mirror sets.  
Sequential read throughput scales linearly with the number of disks; write throughput scales linearly with the number of mirror sets.

For parity (RAID-Z, RAID-Z2) configurations:  
Random read and write performance scales linearly with the number of RAID sets.  
Sequential read and write throughput scales linearly with the number of data (non-parity) disks.

**Caution!** It is NOT recommended to use non-redundant device configuration within a ZFS volume

**Volume Configuration:**

Available Disks	Redundancy Type	Final Volume Configuration
Available physical and logical (virtual) disks. Mounted/Slices/Partitions are not allowed.	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <span style="border: 1px solid #ccc; display: inline-block; width: 100px; height: 1.2em; background-color: #ccc;"></span> </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> <span>Add to pool &gt;&gt;</span> <span>Add to spare &gt;&gt;</span> </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> <span>Add to log &gt;&gt;</span> <span>Add to cache &gt;&gt;</span> </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> <span>Add to selected &gt;&gt;</span> <span>&lt;&lt; Remove selected</span> </div> <div style="display: flex; justify-content: space-between;"> <span>&lt;&lt; Remove all</span> <span>Add or Remove disks and groups</span> </div>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">pool</div> <div style="border: 1px solid #ccc; padding: 5px;">c1t1 d0 : mpt(disk) : 8.00</div> <p style="font-size: 0.7em; margin-top: 5px;">This configuration will be used to create volume.</p>
Available physical and logical (virtual) disks. Mounted/Slices/Partitions are not allowed.	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <span style="border: 1px solid #ccc; display: inline-block; width: 100px; height: 1.2em; background-color: #ccc;"></span> </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> <span>&lt;&lt; Remove selected</span> <span>Add or Remove disks and groups</span> </div> <div style="display: flex; justify-content: space-between;"> <span>&lt;&lt; Remove all</span> <span>Add or Remove disks and groups</span> </div>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"></div> <div style="border: 1px solid #ccc; padding: 5px;"></div> <p style="font-size: 0.7em; margin-top: 5px;">This configuration will be used to create volume.</p>

**Volume Properties:**

**Name**  Volume name must begin with a letter and can only contain alphanumeric characters (a-z, A-Z, 0-9) in addition to the following three special characters: underscore (\_), hyphen (-) and period (.). Volume name has the following restrictions: the beginning sequence c[0-9] is not allowed; the name log is reserved; a name that begins with mirror, raidz, or spare is not allowed because these name are reserved. In addition, volume name must not contain a percent sign (%).

**Description**  Optional volume description. Maximum length is 255 characters.

**Deduplication**  Controls the deduplication option for the volume. If enabled, it will optimize use of duplicate copies of data. Default is off.

**Autoexpand**  Controls automatic pool expansion when the underlying LUN is grown.

**Sync**  Controls synchronous requests (standard - ensure all synchronous requests are written to stable storage; always - every file system transaction will be written and flushed to stable storage by system call return; disabled - synchronous requests are disabled). Default is standard.

**Force creation**  Forces use of (virtual or physical) disks (LUNs) even if they appear to be in use.

**Import Volume**

6. Se configura el sistema de almacenamiento para brindar servicio *iscsi target* y se inicia el servicio.

**Network** | **STEP #5: FOLDERS AND SHARES**

**ISCSI Initiator**

**Disks**

**Volumes**

**Folders and Shares**

At this step you may want to create initial folder structure and share folders via NFS, CIFS, RSYNC, FTP, etc. Consider using the integrated indexing facility to enable probabilistic searching. The indexing facility is integrated with ZFS snapshots allows for searching for documents in snapshots as well.

**Review Changes and Exit**

**Folders:**

<input type="checkbox"/>	Folder	Refer	Used	Avail	CIFS	NFS	FTP	RSYNC	WebDAV	Index	Delete
<input type="checkbox"/>	data/iscsi	31.00 KB	31.00 KB	7.81 GB	<input type="checkbox"/>	<input checked="" type="checkbox"/>					

Results 1 - 1 (all)

**Operations:**

**Create New Folder**

Volume:  (Folder's volume.)

Folder Name:

Description:  (Human-readable description for this folder.)

Record Size:  (Specifies a suggested block size for files in the folder. Default is 128K.)

Log Bias:  (Provide a hint to ZFS about handling of synchronous requests in this dataset. If logbias is set to latency (the default), ZFS will use pool log devices (if configured) to handle the requests at low latency. If logbias is set to throughput, ZFS will not use configured pool log devices. ZFS will instead optimize synchronous operations for global pool throughput and efficient use of resources.)

Deduplication:  (Controls the deduplication option for this dataset. If enabled, it will optimize use of duplicate copies of data. Default is "off".)

Compression:  (Controls the compression algorithm used for this dataset. Default is "off". Setting compression to "on" uses the lzjb compression algorithm. The lzjb compression algorithm is optimized for performance while providing decent data compression. Currently, "gzip" is equivalent to "gzip-6".)

**Network** | **STEP #6: REVIEW CHANGES AND EXIT**

**ISCSI Initiator**

**Disks**

**Volumes**

**Folders and Shares**

**Review Changes and Exit**

You are at final step.

**Network**

Interface	Type	Configuration
e1000g0	single	Configured as 10.12.170.95/255.255.255.0 with mtu 1500

**Disks**

Disk	Device	Type	Size	Volume	Attach	Model
c1t0d0	sd0	disk	8.00 GB	syspool	mpt	VMware, Rev. 1.0
c1t1d0	sd1	disk	8.00 GB	data	mpt	VMware, Rev. 1.0

**Volumes**

Volume	Configuration	Size	Used	Available	Capacity	Dedup Ratio	Status
data	pool: 1	7.94 GB	179.00 KB	7.81 GB	0%	1.00x	ONLINE

**Folders and Shares**

<input type="checkbox"/>	Folder	Refer	Used	Avail	CIFS	NFS	FTP	RSYNC	WebDAV
<input type="checkbox"/>	data/iscsi	31.00 KB	31.00 KB	7.81 GB	-	-	-	-	-

Results 1 - 1 (all)

**Notes:**

- NexentaStor supports system checkpoints - a unique facility designed from bottom up to support live (in-place) upgrades and safe upgrades, with abilities to rollback and/or boot the appliance into any one of the previous checkpoints. At this point it is recommended to create a system checkpoint - a snapshot of the freshly installed and initially configured appliance. For detailed information on the appliance's upgrade/checkpointing functionality see NexentaStor User Guide
- Create a system checkpoint now?
- Optimize appliance's I/O performance by disabling ZFS cache flushing. While providing a considerable performance improvement in certain scenarios (in particular those involving CIFS, NFS or iSCSI) - this settings may be unsafe, in terms of application-level data integrity. It is strongly recommended to use this feature if and only if your storage is NVRAM protected, and the hardware platform is connected to Uninterrupted Power Supply (UPS). Default setting: unchecked (disabled)
- Optimize I/O performance for CIFS/NFS/iSCSI UPS-backed deployments?
- Create periodic scrubbing service (auto-scrub) for system volume weekly, every Sunday at 3am ?

<< Previous Step | Start NMV

7. Se muestra el monitor del sistema.

The screenshot shows the Nexenta system monitor interface. The top navigation bar includes 'About', 'Support', 'Register', 'Help', and 'Login'. The main menu has 'Status', 'Settings', 'Data Management', and 'Analytics'. Below the menu, there are tabs for 'General', 'Storage', and 'Network'. The 'General' tab is active, displaying 'Overall Status Information' on the left sidebar and 'GENERAL STATUS AND DETAILS: NEXENTA' in the main content area. The main content area features a 'CPU and I/O Monitor' section with three gauges: 'CPU Utilization' (0%), 'Network I/O, KB/Sec' (IN and OUT at 0), and 'Disk I/O, KB/Sec' (READ and WRITE at 0). Below the gauges is a 'Seconds between Retrieves' field set to 5 and an 'Update' button. A 'General Appliance Information' table is also visible.

Property	Value
Server Time	Wed Apr 27 02:36:20 2011
Time Zone	America/Havana
Last System Boot	Tue Apr 26 07:22:19 2011
Load Average	0.39, 0.55, 0.34
NMS Version	3.1.0 (r8825)
NMC Version	3.1.0 (r8823)
NMV Version	3.1.0 (r8823)
OS Version	3.0.4
Total Memory	2047MB
Free Memory	981MB

8. Se muestra el target que hay compartido.

The screenshot shows the Nexenta interface for managing 'Zvols'. The top navigation bar includes 'About', 'Support', 'Register', 'Help', 'Welcome Administrator', and 'Logout'. The main menu has 'Status', 'Settings', 'Data Management', and 'Analytics'. Below the menu, there are tabs for 'Data Sets', 'Shares', 'SCSI Target', 'Auto Services', and 'Runners'. The 'SCSI Target' tab is active, displaying 'VIEW ZVOLS' in the main content area. A table lists the shared volumes:

Zvol	Size	Shared	Created
<input checked="" type="checkbox"/> data/san	5.00 GB	No	Tue Apr 26 22:45 2011

Below the table are 'Delete', 'Share', and 'Unshare' buttons. The left sidebar shows 'Zvols' and 'iSCSI' sections with various management options.

### 3.5 Evaluación técnica de la solución.

Se utilizó el método Opinión de Expertos, para realizar la evaluación técnica de la solución integral. Este método permite tomar decisiones para aceptar o rechazar determinada propuesta de acuerdo a criterios definidos y la evaluación dada por los expertos seleccionados. Para llevar a cabo la evaluación se realizaron los siguientes pasos: [47]

**Paso 1:** Se determinan los criterios para la selección de los expertos.

La selección de los expertos se basó en los siguientes criterios:

- Graduado de Ingeniería en Informática, Telecomunicaciones o Automática.
- Experiencia en el trabajo con plataformas de virtualización.
- Experiencia en el trabajo con sistemas de almacenamiento (NAS, SAN).
- Experiencia en el trabajo con plataformas de redes con tecnologías libres.
- Conocimientos sobre virtualización y sistemas de almacenamientos.

**Paso 2:** Se elaboran los criterios utilizados en la evaluación y se agrupan de acuerdo a las características de la propuesta de solución integral.

#### **Grupo 1: Criterios de mérito científico.**

1. Contribución científica de la solución.

#### **Grupo 2: Criterios de creación.**

2. Necesidad de empleo de la solución.
3. Posibilidades de implementación.
4. Ahorro de costos.

#### **Grupo 3: Criterios de flexibilidad.**

5. Escalabilidad.
6. Migración.
7. Fiabilidad.
8. Disponibilidad.

#### **Grupo 4: Criterios de impacto.**

9. Impacto en el área para la cual está destinada la propuesta de solución.
10. Gestión de carga de trabajo.
11. Rendimiento.

#### **Grupo 5: Criterios de usabilidad.**

12. Monitorización.
13. Fácil administración e instalación.
14. Provisiónamiento y despliegue.
15. Funcionalidades.

**Paso 3:** Se le asigna un valor relativo a cada grupo de criterios de acuerdo al porcentaje que representa cada grupo del total y los intereses a evaluar.

Grupo 1: 10 Grupo 2: 15 Grupo 3: 20 Grupo 4: 15 Grupo 5: 40

**Paso 4:** Se seleccionan nueve expertos teniendo en cuenta los criterios del primer paso.

**Paso 5:** Se entrega a los expertos el acceso a la aplicación y una documentación breve sobre la solución, para que se documenten sobre el tema y expresen sus criterios. Los expertos conceden pesos de cero (valor mínimo) a diez (valor máximo) a cada uno de los criterios establecidos y una apreciación cualitativa con una clasificación final del proyecto en alta, media, baja, y fracaso.

**Paso 6:** Después de recibir los valores del peso relativo de cada criterio se construye la tabla correspondiente, donde  $C_{1...15}$  son los criterios a evaluar y  $E_{1...9}$  la evaluación de los expertos.

G	C/E	E <sub>1</sub>	E <sub>2</sub>	E <sub>3</sub>	E <sub>4</sub>	E <sub>5</sub>	E <sub>6</sub>	E <sub>7</sub>	E <sub>8</sub>	E <sub>9</sub>	E <sub>p</sub> = ΣE / 9
10	C <sub>1</sub>	6	7	6	8	7	8	7	8	6	7.6666667
15	C <sub>2</sub>	10	10	9	9	9	8	8	10	10	9.3333333
	C <sub>3</sub>	8	8	7	7	9	9	8	8	10	8.4444444
	C <sub>4</sub>	9	10	9	10	8	9	9	10	8	8.8888889
20	C <sub>5</sub>	10	7	9	8	7	10	8	9	8	8.5555556
	C <sub>6</sub>	7	9	8	9	8	6	7	7	9	7.8888889
	C <sub>7</sub>	10	10	7	9	10	8	10	9	9	8.7777778
	C <sub>8</sub>	7	6	8	5	6	7	7	6	7	7.3333333
15	C <sub>9</sub>	9	8	10	10	8	9	9	7	7	8.5555556
	C <sub>10</sub>	9	7	7	6	9	7	6	8	5	8.3333333
	C <sub>11</sub>	8	8	10	7	9	8	9	10	9	8.6666667
40	C <sub>12</sub>	8	7	8	9	7	6	8	9	8	8.1111111
	C <sub>13</sub>	10	10	9	9	8	8	10	8	9	9.2222222
	C <sub>14</sub>	10	9	9	8	10	9	7	7	8	8.4444444
	C <sub>15</sub>	8	7	7	10	8	9	5	9	8	8.1111111
<b>Total</b>		<b>128</b>	<b>130</b>	<b>122</b>	<b>132</b>	<b>126</b>	<b>123</b>	<b>117</b>	<b>125</b>	<b>134</b>	<b>126.33333</b>

Tabla 9. Pesos otorgados por los expertos a los criterios.

**Paso 7:** Se verifica el trabajo de los expertos, utilizando el coeficiente de concordancia de Kendall y el estadígrafo Chi cuadrado ( $\chi^2$ ). A continuación el procedimiento:

- Sea C el número de criterios que van a evaluarse y E el número de expertos que realizan la evaluación.

- Para cada criterio se determina la  $\Sigma E$  que representa la sumatoria del peso dado por cada experto,  $E_p$  que es la puntuación promedio de los pesos correspondientes a cada criterio.
- Se determina la desviación de la media, que posteriormente se eleva al cuadrado para obtener la dispersión  $S$  por la expresión:  $S = \Sigma (\Sigma E - \Sigma \Sigma E/C)^2$ .

Expertos/Criterios	$\Sigma E$	$\Sigma E/C$	$\Sigma E - \Sigma \Sigma E/C$	$(\Sigma E - \Sigma \Sigma E/C)^2$
C <sub>1</sub>	69	4.6	-6.7999998	46.23999773
C <sub>2</sub>	84	5.6	8.20000017	67.24000273
C <sub>3</sub>	76	5.06666667	0.20000017	0.040000067
C <sub>4</sub>	80	5.33333333	4.20000017	17.6400014
C <sub>5</sub>	77	5.13333333	1.20000017	1.4400004
C <sub>6</sub>	71	4.73333333	-4.7999998	23.0399984
C <sub>7</sub>	79	5.26666667	3.20000017	10.24000107
C <sub>8</sub>	66	4.4	-9.7999998	96.03999673
C <sub>9</sub>	77	5.13333333	1.20000017	1.4400004
C <sub>10</sub>	75	5	-0.7999998	0.639999733
C <sub>11</sub>	78	5.2	2.20000017	4.840000733
C <sub>12</sub>	73	4.86666667	-2.7999998	7.839999067
C <sub>13</sub>	83	5.53333333	7.20000017	51.8400024
C <sub>14</sub>	76	5.06666667	0.20000017	0.040000067
C <sub>15</sub>	73	4.86666666	-2.7999998	7.839999067
$\Sigma \Sigma E/C$		<b>75.7999998</b>		
$S = \Sigma (\Sigma E - \Sigma \Sigma E/C)^2$				<b>336.4</b>

Tabla 10. Cálculo de la Dispersión (S) para hallar la concordancia entre expertos.

- Se calcula el coeficiente de concordancia de Kendall, conociendo la dispersión:

$$W = S / (E^2 (C^3 - C) / 16)$$

- Se calcula el Chi cuadrado real, sabiendo el valor del coeficiente de concordancia de Kendall:

$$X^2 = E * (C - 1) * W$$

S	E <sup>2</sup>	C <sup>3</sup> - C	E <sup>2</sup> (C <sup>3</sup> - C)	E <sup>2</sup> (C <sup>3</sup> - C) / 16	W = S / (E <sup>2</sup> (C <sup>3</sup> - C) / 16)	X <sup>2</sup> = E * (C - 1) * W
336.4	81	3360	272160	17010	0.0197766	<b>2.491852</b>

Tabla 11. Cálculo de concordancia de Kendall.

- Se compara el Chi cuadrado real calculado con el que se obtiene de la Tabla de Distribución Chi Cuadrado (Ver Anexo # 2), se toma  $1 - \alpha = 0.99$  donde  $\alpha = 0.01$  es el error permisible.

Si se cumple que el  $X^2$  real  $< X^2(\alpha, c-1)$  puede decirse que existe concordancia en el trabajo de los expertos.

Según el resultado del cálculo en la tabla anterior: **2.491852 < 30.578**, por tanto, existe concordancia entre los expertos.

**Paso 8:** Posteriormente se identifica el peso (P) relativo de cada criterio y se calcula el Índice de Aceptación (IA) de la solución integral. [48]

- Se calcula el peso de cada criterio (P), conociendo el número de expertos que realizan la evaluación E y la sumatoria de las puntuaciones de cada criterio (C).
- Conociendo el peso de cada criterio y la cantidad de expertos se puede obtener el valor de  $P \times C_p$ , donde  $C_p$ , es el criterio promedio concebido por los expertos en escala de 5.
- Con el valor anterior se calcula el Índice de Aceptación (IA).  **$IA = \Sigma (P \times C_p) / 5$** .

Criterios	$C_p$	P	$P \times C_p$
$C_1$	3.833333333	0.0588	0.2254
$C_2$	4.666666667	0.0761	0.35513333
$C_3$	4.222222222	0.0669	0.28246667
$C_4$	4.444444444	0.0715	0.31777778
$C_5$	4.277777778	0.068	0.29088889
$C_6$	3.944444444	0.0611	0.24100556
$C_7$	4.388888889	0.0703	0.30853889
$C_8$	3.666666667	0.0553	0.20276667
$C_9$	4.277777778	0.068	0.29088889
$C_{10}$	4.166666667	0.0657	0.27375
$C_{11}$	4.333333333	0.0692	0.29986667
$C_{12}$	4.055555556	0.0634	0.25712222
$C_{13}$	4.611111111	0.075	0.34583333
$C_{14}$	4.222222222	0.0669	0.28246667
$C_{15}$	4.055555556	0.0634	0.25712222
$\Sigma (P \times C_p)$			<b>4,23102778</b>
IA			<b>0,84620556</b>

Tabla 12. Calificación de cada criterio e índice de aceptación.

**Paso 9:** Por último se determina la probabilidad de éxito de la propuesta, ubicando el IA calculado anteriormente en rangos que están predefinidos en la Tabla 13, en dependencia de su ubicación será la probabilidad de éxito que tenga la propuesta.

Fracaso seguro: no cumple con un mínimo de variables claves.

Baja probabilidad de éxito: no cumple con las necesidades de integración.

Media probabilidad de éxito: cumple solo con los criterios de flexibilidad.

Alta probabilidad de éxito: cumple con las variables claves de usabilidad e impacto.

<b><math>0.7 &lt; IA</math></b>	Alta probabilidad de éxito
<b><math>0.5 &lt; IA &lt; 0.7</math></b>	Media probabilidad de éxito
<b><math>0.3 &lt; IA &lt; 0.5</math></b>	Baja probabilidad de éxito
<b><math>IA &lt; 0.3</math></b>	Fracaso seguro

Tabla 13. Rangos predefinidos para el IA.

El IA calculado es **0,84620** lo que significa que existe alta probabilidad de éxito con la solución integral.

### 3.6 Conclusiones parciales.

La solución implementada posee un gran aprovisionamiento y despliegue. La administración resulta sencilla y se presenta como una solución con estabilidad y futuro. Nexenta añade a esta solución un ahorro que puede llegar al 70-80% respecto a soluciones propietarias, disponiendo además de características superiores de funcionamiento y flexibilidad, así como de una completa libertad para la migración de todos los datos a otro sistema (por ser open source, y basado en sistemas estandarizados). Todo ello con el soporte directo de Nexenta, así como de la comunidad “open source” alrededor de los proyectos afines.

Según los expertos la solución integral tiene altas probabilidades de éxito, debe contribuir a dar cumplimiento a nuestros objetivos iniciales.

## CONCLUSIONES

Los objetivos y las tareas de investigación trazadas para solucionar el problema planteado, fueron cumplidos. Se estudió, se seleccionó y se realizaron pruebas automáticas de carga y rendimiento a un grupo de tecnologías, soluciones de virtualización y sistemas de almacenamiento, que ya forman parte de las líneas de trabajo de nuestro centro. Aún cuando no están exentas de inconvenientes, sus potencialidades las hacen apropiadas para ser utilizadas, y sean la base para futuros desarrollos. Se ofrecieron los elementos principales para la instalación y configuración de la solución. Además de una evaluación técnica a través de la opinión de expertos, usando como indicadores las variables principales que se identificaron, y se calculó un índice de aceptación para determinar la probabilidad de éxito de la solución integral.

El presente trabajo permite a cualquier especialista implicado en los temas de infraestructura para CD, particularmente en los de virtualización y almacenamiento, adquirir una base y una visión para enfrentar proyectos de este tipo, haciendo especial énfasis en las tecnologías libres, tan útiles y necesarias para un país como Cuba.

## RECOMENDACIONES

Se recomienda usar la Prospectiva Tecnológica como metodología de trabajo para la planeación de proyectos de infraestructura tecnológica, por las posibilidades que ofrece en la planeación a corto, mediano y largo plazo.

Considerar un marco de trabajo específico para proyectos de infraestructura, teniendo en cuenta, que el desarrollo de soluciones de infraestructura tecnológica tiene particularidades que lo hacen diferente al desarrollo de software.

Utilizar herramientas para la realización de pruebas a las soluciones o productos que se obtengan, teniendo en cuenta las ventajas y facilidades que brindan.

Continuar el seguimiento, el desarrollo, estudio y aplicación de las tecnologías libres que aquí se analizan.

Continuar creando las capacidades para la preparación del personal en la utilización de las tecnologías libres en el área del objeto de estudio.

## BIBLIOGRAFÍA

1. ROBLES PEIRO, H. La economía basada en el conocimiento. Las condiciones de los estados mexicanos. *Revista Razón y Palabra*, No 49, p.1. En línea, mayo de 2011:  
<http://www.razonypalabra.org.mx/anteriores/n49/bienal/Mesa%2012/HectorRobles.pdf>
2. MORFÍN OTERO, M.; GONZÁLEZ ROMERO, V.M. Infraestructura de tecnologías de información en universidades mexicanas. Universidad de Guadalajara, Campus Puerto Vallarta, México, p.2. En línea, abril de 2011: [http://www.ateneonline.net/datos/100\\_03\\_Morfin%20Maria\\_Gonzalez\\_Victor.pdf](http://www.ateneonline.net/datos/100_03_Morfin%20Maria_Gonzalez_Victor.pdf)
3. *Curso de Administración de Centro de cómputo*. p.6. En línea, mayo de 2011:  
<http://es.scribd.com/doc/3081866/Administracion-de-Centros-de-Computo>
4. Boletín Virtualización. *Dialogoti*. En línea, junio de 2011:  
<http://www.dialogoti.com/ar/scriptServices/courseToPdf.ashx?courseId=a429243d-5c28-47fc-be35-18db33b5062b>
5. VILLAR FERNÁNDEZ, E.E. *Virtualización de servidores de telefonía IP en GNU/Linux*. p.24. Titulación de Ingeniero en Informática, Universidad de Almería, 2010. En línea, junio de 2011:  
[http://www.adminso.es/images/d/dc/PFC\\_eugenio.pdf](http://www.adminso.es/images/d/dc/PFC_eugenio.pdf)
6. *Ibíd*em, p.28.
7. *Ibíd*em, p.29.
8. *Ibíd*em, p.29.
9. *Ibíd*em, p.29.
10. *Ibíd*em, p.29.
11. *Ibíd*em, p.46.
12. CASAD, Joe; LUKAS, R. Almacenamiento de datos en Linux a examen. *Linux Magazine*, No 63, p.19. En línea, febrero de 2011: [http://www.linux-magazine.es/issue/63/018-019\\_AlmacenamientoLM63.pdf](http://www.linux-magazine.es/issue/63/018-019_AlmacenamientoLM63.pdf)
13. *Ibíd*em.
14. *Ibíd*em.
15. *Ibíd*em.
16. VILLAR FERNÁNDEZ, E.E: Ob. cit., pp.16-17.
17. SITIO OFICIAL DEL PROYECTO GNU. En línea, marzo de 2011:  
<http://www.gnu.org/philosophy/free-sw.es.html>
18. SITIO OFICIAL DE KVM. En línea, marzo de 2011: [http://www.linux-kvm.org/page/Main\\_Page](http://www.linux-kvm.org/page/Main_Page)
19. SITIO OFICIAL DE XEN. En línea, marzo de 2011: <http://www.xensource.com/products/>
20. SITIO OFICIAL DE OPENVZ. En línea, marzo de 2011: [http://wiki.openvz.org/Main\\_Page](http://wiki.openvz.org/Main_Page)

21. SITIO OFICIAL DE OPENNEBULA. En línea, marzo de 2011: <http://opennebula.org/>
22. SITIO OFICIAL DE OPENNODE. En línea, marzo de 2011: <http://opennode.activesys.org/>
23. SITIO OFICIAL DE LA PROXMOX. En línea, marzo de 2011: <http://proxmox.es/>
24. SITIO OFICIAL DEL PROYECTO OVIRT. En línea, marzo de 2011:  
[http://fedoraproject.org/wiki/Features/Ovirt\\_Node](http://fedoraproject.org/wiki/Features/Ovirt_Node)
25. SITIO OFICIAL DE OPENFILER. En línea, marzo de 2011: <http://www.openfiler.com>
26. SITIO OFICIAL DE FREENAS. En línea, marzo de 2011: <http://freenas.org/>
27. SITIO OFICIAL DE NEXENTASTOR. En línea, marzo de 2011: <http://www.nexentastor.org/>
28. SHAH, A. Virtualización basada en el kernel con KVM. *Linux Magazine*, No 36, p.34. En línea, febrero de 2011: [www.linux-magazine.es/issue/36/034-036\\_KVMLM36.pdf](http://www.linux-magazine.es/issue/36/034-036_KVMLM36.pdf)
29. *Ibidem*, p.36.
30. *Ibidem*.
31. LOSCHWITZ, M.; FEILNER, M. Comenzar con la virtualización Xen. *Linux Magazine*, No 39, p.22. En línea, febrero de 2011: [http://www.linux-magazine.es/issue/39/022-025\\_XensacionalLM39.pdf](http://www.linux-magazine.es/issue/39/022-025_XensacionalLM39.pdf)
32. *Ibidem*.
33. *Ibidem*, p.25
34. VILLAR FERNÁNDEZ, E.E: Ob. cit., pp. 80-81.
35. MARTÍNEZ RODRÍGUEZ, C.; MARTÍNEZ MARTÍNEZ, Y. *Suite de pruebas para el Centro para la Informatización de Gestión de Entidades*. p.13. Universidad de las Ciencias Informáticas, 2010.
36. *Ibidem*, p.14
37. *Ibidem*.
38. *Ibidem*, p.15
39. *Ibidem*, p.18
40. *Ibidem*.
41. SÁNCHEZ ALMENARES, L. *Cómo realizar pruebas de carga y estrés en JMeter*. p.24. Universidad de las Ciencias Informáticas, Ciudad Habana, 2008.
42. MARTÍNEZ RODRÍGUEZ, C.; MARTÍNEZ MARTÍNEZ, Y: Ob. cit., p. 35.
43. ESLAVA, Edgar. ¿Qué es prospectiva?, 2010. En línea, junio de 2011:  
<http://www.degerencia.com/articulo/que-es-prospectiva/imp>
44. OLIVÉ ÁLVAREZ, Eunice. La prospectiva tecnológica en el diseño de estrategias. *Primer Taller de Proyección y Prospección en Tecnologías de Información y Comunicación*, 13-15 de octubre de 2008, CITMA. En línea, mayo de 2011: [www.redciencia.info.ve/memorias/ProyProsp/trabajos/I4.ppt](http://www.redciencia.info.ve/memorias/ProyProsp/trabajos/I4.ppt)
45. MOJICA, F.J: Determinismo y construcción del futuro. En línea, mayo de 2011:  
[http://celgyp.org/trabajos/trabajos/trabajos/Determinismo\\_y\\_Construccion\\_del\\_Futuro.pdf](http://celgyp.org/trabajos/trabajos/trabajos/Determinismo_y_Construccion_del_Futuro.pdf)

46. GODET, M.; MONTI, R.; MEUNIER, F.; ROUBELAT, F. La caja de herramientas de la prospectiva estratégica, 2000, p. 17 (fotocopia).
47. HERNÁNDEZ LEÓN, Rolando, *Una Introducción a la Gestión de Proyectos*, (2009). Universidad de las Ciencias Informáticas.
48. SÁNCHEZ TAMAYO, K. *Método para evaluar proyectos informáticos y establecer un orden de prioridad que ayude a la toma de decisiones*. Tesis de maestría, Universidad de las Ciencias Informáticas, La Habana, julio 2010.

## ANEXOS

**Anexo 1:** Característica del hardware y versión del software utilizado en los escenarios de pruebas.

Servidores	RAM	CPU Intel Xeon	Interfaces	HDD	RAID
<b>Lenovo T350</b>	4 Gb	4 (1.6GHz)	2 (Gb Ethernet)	4 (69Gb)	1
<b>Dell 2850</b>	1 Gb	2 (3GHz)	2 (Gb Ethernet)	4 (73Gb)	1

Hardware complementario:

- **Switch Edge Core**, 48 puertos, 1 Gb Ethernet.
- **Cable UTP** Cat5e

Versión del software utilizado en los escenarios de pruebas:

Software	Versión
Proxmox	1.7
Centos	5.6
KVM	0.13
Xen	3.0.3
Hardinfo	0.5.1
lozone	3-392
Nexenta	3.0.4
FreeNas	0.7.2
OpenFiler	2.3
NFS	v3
Jmeter	2.4
Ovirt	1.9.2
OpenNebula	2.0.1
OpenVZ	028stab089.1
Iperf	2.0.5

**Anexo 2:** Tabla de Distribución Chi-cuadrado ( $\chi^2$ ).  $P(X \geq x)$ . Donde “P” es la probabilidad de encontrar un valor mayor o igual que el Chi cuadrado tabulado, y “v” los grados de libertad.

P	0.99	0.95	0.90	0.85	0.80	0.75	0.30	0.20	0.15	0.10	0.05	0.025	0.01	0.005	0.001
v=1	0.000	0.004	0.016	0.036	0.064	0.102	1.074	1.642	2.072	2.706	3.841	5.024	6.635	7.879	10.828
2	0.020	0.103	0.211	0.325	0.446	0.575	2.408	3.219	3.794	4.605	5.991	7.378	9.210	10.597	13.816
3	0.115	0.352	0.584	0.798	1.005	1.213	3.665	4.642	5.317	6.251	7.815	9.348	11.345	12.838	16.266
4	0.297	0.711	1.064	1.366	1.649	1.923	4.878	5.989	6.745	7.779	9.488	11.143	13.277	14.860	18.467
5	0.554	1.145	1.610	1.994	2.343	2.675	6.064	7.289	8.115	9.236	11.070	12.833	15.086	16.750	20.515
6	0.872	1.635	2.204	2.661	3.070	3.455	7.231	8.558	9.446	10.645	12.592	14.449	16.812	18.548	22.458
7	1.239	2.167	2.833	3.358	3.822	4.255	8.383	9.803	10.748	12.017	14.067	16.013	18.475	20.278	24.322
8	1.646	2.733	3.490	4.078	4.594	5.071	9.524	11.030	12.027	13.362	15.507	17.535	20.090	21.955	26.124
9	2.088	3.325	4.168	4.817	5.380	5.899	10.656	12.242	13.288	14.684	16.919	19.023	21.666	23.589	27.877
10	2.558	3.940	4.865	5.570	6.179	6.737	11.781	13.442	14.534	15.987	18.307	20.483	23.209	25.188	29.588
11	3.053	4.575	5.578	6.336	6.989	7.584	12.899	14.631	15.767	17.275	19.675	21.920	24.725	26.757	31.264
12	3.571	5.226	6.304	7.114	7.807	8.438	14.011	15.812	16.989	18.549	21.026	23.337	26.217	28.300	32.909
13	4.107	5.892	7.042	7.901	8.634	9.299	15.119	16.985	18.202	19.812	22.362	24.736	27.688	29.819	34.528
14	4.660	6.571	7.790	8.696	9.467	10.165	16.222	18.151	19.406	21.064	23.685	26.119	29.141	31.319	36.123
15	5.229	7.261	8.547	9.499	10.307	11.037	17.322	19.311	20.603	22.307	24.996	27.488	30.578	32.801	37.697
16	5.812	7.962	9.312	10.309	11.152	11.912	18.418	20.465	21.793	23.542	26.296	28.845	32.000	34.267	39.252
17	6.408	8.672	10.085	11.125	12.002	12.792	19.511	21.615	22.977	24.769	27.587	30.191	33.409	35.718	40.790
18	7.015	9.390	10.865	11.946	12.857	13.675	20.601	22.760	24.155	25.989	28.869	31.526	34.805	37.156	42.312
19	7.633	10.117	11.651	12.773	13.716	14.562	21.689	23.900	25.329	27.204	30.144	32.852	36.191	38.582	43.820
20	8.260	10.851	12.443	13.604	14.578	15.452	22.775	25.038	26.498	28.412	31.410	34.170	37.566	39.997	45.315
21	8.897	11.591	13.240	14.439	15.445	16.344	23.858	26.171	27.662	29.615	32.671	35.479	38.932	41.401	46.797
22	9.542	12.338	14.041	15.279	16.314	17.240	24.939	27.301	28.822	30.813	33.924	36.781	40.289	42.796	48.268
23	10.196	13.091	14.848	16.122	17.187	18.137	26.018	28.429	29.979	32.007	35.172	38.076	41.638	44.181	49.728
24	10.856	13.848	15.659	16.969	18.062	19.037	27.096	29.553	31.132	33.196	36.415	39.364	42.980	45.559	51.179
25	11.524	14.611	16.473	17.818	18.940	19.939	28.172	30.675	32.282	34.382	37.652	40.646	44.314	46.928	52.620
26	12.198	15.379	17.292	18.671	19.820	20.843	29.246	31.795	33.429	35.563	38.885	41.923	45.642	48.290	54.052
27	12.879	16.151	18.114	19.527	20.703	21.749	30.319	32.912	34.574	36.741	40.113	43.195	46.963	49.645	55.476
28	13.565	16.928	18.939	20.386	21.588	22.657	31.391	34.027	35.715	37.916	41.337	44.461	48.278	50.993	56.892
29	14.256	17.708	19.768	21.247	22.475	23.567	32.461	35.139	36.854	39.087	42.557	45.722	49.588	52.336	58.301
30	14.953	18.493	20.599	22.110	23.364	24.478	33.530	36.250	37.990	40.256	43.773	46.979	50.892	53.672	59.703

## GLOSARIO

**Acceso Abierto (Open Access):** es aquel que permite el libre acceso a los recursos digitales derivados de la producción científica o académica sin barreras económicas o restricciones derivadas de los derechos de copyright sobre los mismos.

**Advanced Technology Attachment (ATA):** Interfaz estándar para conectar dispositivos de almacenamiento. Es una evolución del estándar Integrated Drive Electronics (IDE) de Western Digital.

**ATA with Packet Interface (ATAPI):** Es una evolución de ATA que permite conectar dispositivos extraíbles, como unidades ópticas, que necesitan funciones de extracción o verificación de si el soporte está introducido. Básicamente son dispositivos ATA utilizando el protocolo de SCSI en paquetes.

**Apple Filing Protocol (AFP):** Es un protocolo de capa de presentación (según el modelo OSI) que ofrece servicios de archivos. Es uno de los servicios de apoyo a disposición incluida Bloque de mensajes de servidor (SMB), Sistema de archivos de red (NFS), el Protocolo de transferencia de archivos (FTP), y WebDAV.

**Bridge:** Es un dispositivo de interconexión de redes de ordenadores que opera en la capa 2 (nivel de enlace de datos) del modelo OSI. Este interconecta dos segmentos de red (o divide una red en segmentos) haciendo el pasaje de datos de una red hacia otra, con base en la dirección física de destino de cada paquete.

**Bonding:** Es la unión de una o más interfaces de red para obtener redundancia y mayor velocidad en la transmisión de datos.

**Canal de Fibra sobre Ethernet (FCoE, *Fibre Channel over Ethernet*):** Usa Ethernet mejorada como transporte por la red física con el beneficio del canal de fibra estándar, esquivando las limitaciones de TCP/IP, además de que está diseñado para tener el aspecto y actuar como canal de fibra nativo en las capas superiores del software, incluido sistema operativo, aplicaciones y herramientas de gestión.

**CentOS (Community Enterprise Operating System):** Es un clon a nivel binario de la distribución Linux Red Hat Enterprise Linux RHEL, compilado por voluntarios a partir del código fuente liberado por Red Hat.

**Common Internet File System (CIFS):** Sistema común de archivos de Internet, es un Protocolo de red (que pertenece a la capa de aplicación en el modelo OSI) que permite compartir archivos e impresoras (entre otras cosas) entre nodos de una red. Es utilizado principalmente en ordenadores con Microsoft Windows y DOS.

**DOM 0:** Se refiere al primer sistema operativo invitado en XEN, es decir el sistema operativo que ejecuta XEN.

**DOM U:** Se refiere a los sistemas operativos invitado que se ejecutan sobre XEN, es decir las máquinas virtuales que se ejecutan sobre XEN.

**Distributed Replicated Block Device (DRBD):** Se refiere a los dispositivos de bloque concebido como una edificación en bloque, agrupados para formar un clúster de alta disponibilidad (HA). Esto se hace por medio del reflejo completo de un dispositivo de bloque asignado a través de una red.

**Directorio Activo (Active Directory):** Es el término que usa Microsoft para referirse a su implementación de servicio de directorio en una red distribuida de computadores. Utiliza distintos protocolos (principalmente LDAP, DNS, DHCP, Kerberos). Almacena información de una organización en una base de dato central, de forma organizada y accesible.

**Enterprise Systems Connection (ESCON):** Es la marca comercial de IBM para una interfaz óptica serial entre los mainframe de IBM y los dispositivos periféricos tales como unidades de almacenamiento y de respaldo.

**Fibre Connection (FICON):** Es una interfaz desarrollada por IBM siendo la evolución de la interfaz ESCON, esto se produce debido a las limitantes en cuanto a la cantidad de canales que se podían establecer.

**Fibre Channel Protocol (FCP):** Es una tecnología de red utilizada principalmente para redes de almacenamiento, disponible primero a la velocidad de 1 Gbps y posteriormente a 2, 4 y 8 Gbps.

**File Transfer Protocol (FTP):** Protocolo de transferencia de archivos entre sistemas conectados a una red Transmission Control Protocol (TCP), basada en la arquitectura cliente-servidor.

**FreeBSD:** Es un sistema operativo libre para computadoras basado en las CPU de arquitectura Intel, incluyendo procesadores Intel 80386, Intel 80486 (versiones SX y DX), y Pentium.

**Gigabit Ethernet:** También conocida como GigaE, es una ampliación del estándar Ethernet (concretamente la versión 802.3ab y 802.3z del IEEE) que consigue una capacidad de transmisión de 1 gigabit por segundo, correspondientes a unos 1000 megabits por segundo de rendimiento contra unos 100 de Fast Ethernet (también llamado 100BASE-TX).

**GPL:** Licencia Pública General de GNU o General Public License, es una licencia creada por la Free Software Foundation, y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

**Hypertext Transfer Protocol o HTTP:** Protocolo de transferencia de hipertexto, usado en cada transacción de la World Wide Web. HTTP define la sintaxis y la semántica que utilizan los elementos de software de la arquitectura web (clientes, servidores, proxies) para comunicarse.

**Hub (Concentrador):** Es un dispositivo que permite centralizar el cableado de una red y poder ampliarla. Esto significa que dicho dispositivo recibe una señal y repite esta señal emitiéndola por sus diferentes puertos.

**Hypervisor:** Un hipervisor o monitor de máquina virtual es una plataforma que permite aplicar diversas técnicas de control de virtualización para utilizar, al mismo tiempo, diferentes sistemas operativos (sin modificar o modificados en el caso de paravirtualización) en una misma computadora.

**Hardware:** Hardware corresponde a todas las partes tangibles de una computadora: sus componentes eléctricos, electrónicos, electromecánicos y mecánicos; sus cables, gabinetes o cajas, periféricos de todo tipo y cualquier otro elemento físico involucrado.

**Internet Small Computer Storage Interface (iSCSI):** Interfaz de almacenamiento para equipos pequeños en Internet, es un método de conexión de dispositivos de almacenamiento a través de una red mediante TCP/IP. Puede usarse a través de una red de área local (LAN), una red de área extensa (WAN) o Internet.

**Kernel o núcleo:** Es un software que constituye la parte más importante del sistema operativo. Es el principal responsable de facilitar a los distintos programas acceso seguro al hardware de la computadora o en forma más básica, es el encargado de gestionar recursos, a través de servicios de llamada al sistema.

**LUN (Logical Unit Number):** Es una dirección para una unidad de disco duro y por extensión, el disco en sí mismo. El término es originario del protocolo SCSI como una forma de diferenciar unidades de disco individuales dentro de un bus SCSI tal que un arreglo de discos.

**Licencia BSD:** Es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution). Es una licencia de software libre permisiva como la licencia de OpenSSL o la MIT License. Esta licencia tiene menos restricciones en comparación con otras como la GPL estando muy cercana al dominio público. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre.

**LiveCD:** Se denomina a un "cdrom" que contiene software que no depende de ningún sistema operativo para su funcionamiento. Este software puede ser un sistema operativo con todas las herramientas básicas como lo pueden ser , un procesador de textos, procesador de hojas de cálculo, un navegador web, clientes de mensajería instantánea, juegos y demás o bien un solo programa.

**Lightweight Directory Access Protocol (LDAP):** Protocolo Ligero de Acceso a Directorios, es considerado una base de datos (aunque su sistema de almacenamiento puede ser diferente) a la que pueden realizarse consultas. Un directorio es un conjunto de objetos con atributos organizados en una manera lógica y jerárquica.

**Logical Volume Manager (LVM):** Es una implementación de un administrador de volúmenes lógicos para el kernel Linux.

**Máquina Virtual (MV):** Una máquina virtual es un software que emula a una computadora y puede ejecutar programas como si fuese una computadora real.

**Network File System (NFS):** Sistema de archivos en red, es un protocolo de nivel de aplicación, según el Modelo OSI. Es utilizado para sistemas de archivos distribuido en un entorno de red de computadoras de área local.

**Network Information Service (NIS):** Sistema de información de red, es el nombre de un protocolo de servicios de directorios cliente-servidor desarrollado por Sun Microsystems para el envío de datos de configuración en sistemas distribuidos tales como nombres de usuarios y hosts entre computadoras sobre una red.

**Opensolaris:** Fue un sistema operativo libre publicado en 2005 a partir de la versión privativa de Solaris de Sun Microsystems, ahora parte de Oracle Corporation.

**Qemu:** Es un emulador de procesadores basado en la traducción dinámica de binarios (conversión del código binario de la arquitectura fuente en código entendible por la arquitectura huésped). QEMU también tiene capacidades de virtualización dentro de un sistema operativo, ya sea GNU/Linux, Windows, o cualquiera de los sistemas operativos admitidos (es la forma más común de uso).

**Redundant Array of Independent Disks (RAID):** Conjunto redundante de discos independientes, conocido anteriormente como Redundant Array of Inexpensive Disks, (conjunto redundante de discos baratos) hace referencia a un sistema de almacenamiento que usa múltiples discos duros o SSD entre los que distribuyen o replican los datos. Dependiendo de su configuración (a la que suele llamarse “nivel”), los beneficios de un RAID respecto a un único disco son uno o varios de los siguientes: mayor integridad, mayor tolerancia a fallos, mayor rendimiento y mayor capacidad.

**Software:** Es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

**Sistema operativo:** Un sistema operativo es el programa o conjunto de programas que efectúan la gestión de los procesos básicos de un sistema informático, y permite la normal ejecución del resto de las operaciones.

**Small Computer Storage Interface (SCSI):** Es una interfaz estándar para la transferencia de datos entre distintos dispositivos del bus de la computadora.

**Serial Attached SCSI (SAS):** Es una interfaz de transferencia de datos en serie, sucesor del SCSI (Small Computer System Interface) paralelo, aunque sigue utilizando comandos SCSI para interactuar con los dispositivos SAS. Aumenta la velocidad y permite la conexión y desconexión de forma rápida.

**Switch (Conmutador):** Es un dispositivo digital de lógica de interconexión de redes de computadores que opera en la capa de enlace de datos del modelo OSI. Su función es interconectar dos o más segmentos de red, de manera similar a los puentes de red, pasando datos de un segmento a otro de acuerdo con la dirección MAC de destino de las tramas en la red. Los conmutadores se utilizan cuando se desea conectar múltiples redes, fusionándolas en una sola. Al igual que los puentes, dado que funcionan como un filtro en la red, mejoran el rendimiento y la seguridad de las redes de área local.

**Secure Shell (SSH):** Intérprete de órdenes segura, es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, y también puede redirigir el tráfico de X para poder ejecutar programas gráficos si tenemos un Servidor X (en sistemas Unix y Windows) corriendo.

**TCP/IP:** Son las siglas de Protocolo de Control de Transmisión / Protocolo de Internet (en inglés Transmission Control Protocol / Internet Protocol) y es un conjunto de protocolos de red en los que se basa Internet y que permiten la transmisión de datos entre redes de computadoras.

**Trivial File Transfer Protocol (TFTP):** Protocolo de transferencia de archivos trivial, muy simple semejante a una versión básica de FTP. TFTP a menudo se utiliza para transferir pequeños archivos entre ordenadores en una red, como cuando un terminal X Window o cualquier otro cliente ligero que arranca desde un servidor de red.

**Universal Plug and Play (UPnP):** Es una arquitectura de software abierta y distribuida que de forma independiente al fabricante, sistema operativo, lenguaje de programación, etc. permite el intercambio de información y datos a los dispositivos conectados a una red.

**VMware vSphere Hypervisor (ESXi):** Es una plataforma de virtualización a nivel de centro de datos producido por VMware, Inc.. Es el componente de su producto VMware Infraestructura que se encuentra al nivel inferior de la capa de virtualización, el hipervisor, aunque posee herramientas y servicios de gestión autónomos e independientes.

**ZFS:** Es un sistema de archivos desarrollado por Sun Microsystems para su sistema operativo Solaris. El significado original era “Zettabyte File System”, pero ahora es un acrónimo recursivo