

Universidad Autónoma Metropolitana

Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

Reporte de Proyecto Terminal

Virtualización de servidores de aplicaciones y optimización de servicios para
el Consejo de Salubridad General

Modalidad: Estancia Profesional

Trimestre 2018 Primavera

Dafne Joana Ceja Linares

207200912

al207200912@alumnos.azc.uam.mx

Declaratoria.

Yo, José Ignacio Vega Luna, como asesor declaro que aprobé el contenido del presente Reporte de Proyecto de integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Firma.

Yo, Juan Manuel Saucedo Camacho, como coasesor y/o jefe directo declaro que aprobé el contenido del presente Reporte de Proyecto de integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Firma.

Yo, Dafne Joana Ceja Linares, doy mi autorización a la Coordinación de Servicios de Información de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco, para publicar el presente documento en la Biblioteca Digital, así como en el repositorio Institucional de UAM Azcapotzalco.



Firma.

Contenido

Contenido	3
1. Objetivo General	4
2. Objetivos Específicos	4
3. Justificación.....	4
4. Introducción	5
5. Marco teórico	6
5.1.1. Historia de la Virtualización.....	6
5.1.3. Ventajas de implementar virtualización de servidores.....	6
5.1.4. Desventajas de la virtualización de servidores.....	7
5.2. Software para realizar la virtualización.....	7
5.2.1. Virtualbox.....	7
5.3. Sistema Operativo (SO) implementado.....	7
5.3.1. CentOS.....	7
5.3.2 Servicios instalados.....	8
6. Descripción breve de la solución	10
7. Descripción técnica	11
7.1. Diseño e implementación de servidores virtuales de aplicaciones.....	11
7.1.1. Diseño de la Arquitectura de nuestros servidores	11
7.1.2. Preparación del software que utilizaremos.....	16
7.1.3.Virtualización.....	16
7.1.3Instalación y configuración del SO.....	18
Selección e instalación de las aplicaciones del servidor	24
7.1.4. Instalación y configuración de Herramientas de desarrollo	30
7.2. Optimización de sus servicios para agilizar y minimizar el mantenimiento de estos.....	38
7.2.1. Creación de Shell Scripts para la evaluación de recursos.....	38
7.2.1. Creación de Shell Scripts para monitorizar los servicios instalados en los servidores virtuales.....	42
8.Conclusiones	46
9. Apéndice	47
9.1. Código Shell Script para evaluar recursos.....	47
9.2.Codigo Shell Script para monitorizar los servicios instalados en los servidores virtuales.....	49
10.Bibliografía	54

Virtualización de servidores de aplicaciones y optimización de servicios para el Consejo de Salubridad General

1. Objetivo General

Diseñar e implementar servidores virtuales de aplicaciones y optimizar sus servicios para agilizar y minimizar el mantenimiento y uso de estos.

2. Objetivos Específicos

- Diseñar e implantar la arquitectura de tres servidores virtuales de aplicaciones.
- Caracterizar las funcionalidades de los servidores virtuales de aplicaciones.
- Optimizar el uso de recursos por servidores virtuales para evaluar su desempeño utilizando un shell script.
- Diseñar e implementar un mecanismo de monitorización de servicios instalados en los servidores.
- Establecer un plan de recuperación de un servidor web mediante respaldos periódicos y automáticos de información.

3. Justificación

El CSG siendo un órgano normativo consultivo requiere actualizar su infraestructura tecnológica para cumplir con el plan nacional de desarrollo, por esta razón se realizará el análisis, diseño, implantación y optimización de servidores virtuales. El CSG cuenta con el apoyo de la DGTI, la cual proporcionará direcciones IP homologadas y dominios para cada servidor virtual que se implante en la LAN (*Local Area Network*) del CSG.

En la actualización de infraestructura tecnológica se realizará la consolidación de 3 servidores virtuales de aplicación en un servidor físico para aumentar la eficiencia de los sistemas implantados actualmente. Estos servidores virtuales se usarán de la siguiente manera; el primero sustituirá al servidor web actual y alojará las aplicaciones de usuarios y página web del CSG. El segundo servidor virtual alojará aplicaciones de usuarios y una copia de la página web de producción. Actualmente el CSG suspende el servicio de su sitio web cada que se realizan cambios y actualizaciones en él, por esta razón, en el segundo servidor se realizarán los cambios y pruebas de su página web antes de la puesta en producción sin necesidad de suspender el servicio. El tercer servidor será implementado para ser el respaldo del servidor web de producción. Este servidor será utilizado como plan de contingencia para evitar la pérdida de información que se pueda dar al realizar cambios

en el servidor web principal. Los servidores se deben instalar con los requerimientos técnicos solicitados por la Dirección de Sistemas de Información del CSG.

Con el fin de brindar un mejor servicio se realizará la optimización de los servicios instalados en cada servidor para agilizar y minimizar la monitorización y mantenimiento de los servicios, lo cual se logrará por medio de la implementación de *Shell Scripts* en cada servidor.

4. Introducción

“El Consejo de Salubridad General (CSG) es un órgano colegiado que depende directamente del presidente de la República y tiene el carácter de autoridad sanitaria, con funciones normativas, consultivas y ejecutivas. Las disposiciones que emita serán de carácter general y obligatorias en el país, el CSG tiene la visión de avanzar en concordancia con el Plan Nacional de Desarrollo y consolidar su carácter de autoridad sanitaria, favoreciendo la integración y universalidad del sistema de salud, mediante sus funciones normativas, consultivas y ejecutivas” [1].

Actualmente la infraestructura de comunicaciones del CSG ya no satisface las necesidades crecientes de acceso y procesamiento de información de manera eficiente, debido a que sólo dispone de un servidor en el que se mantiene en línea su página web y 3 servidores virtuales que fueron asignados por la DGTI (Dirección General de Tecnologías de la Información) de la Secretaría de Salud, los cuales no se encuentran en funcionamiento por la falta de capacidad necesaria para el alojamiento de servicios requeridos por la Dirección de SI (Sistemas de Información) del CSG.

Dado lo anterior, el CSG implantará soluciones tecnológicas que apoyen el cumplimiento de los objetivos de su visión mediante el uso de una mejor infraestructura tecnológica centralizada. Usando las recientes tecnologías, se implementarán servidores virtuales que permitan contar con una plataforma de cómputo actualizada y un mecanismo de respaldo de su servidor web. Esta plataforma debe permitir realizar cambios y actualizaciones de aplicaciones antes de que éstas salgan a producción sin generar afectaciones a usuarios que se encuentren usando y accediendo al servidor de producción en línea.

En cada servidor a implantar se diseñarán y realizarán Shell scripts que permitan optimizar los servicios ofrecidos por la plataforma de cómputo del CSG.

5. Marco teórico

5.1. Virtualización:

“La virtualización es un término que se refiere a la abstracción de los recursos de de una computadora llamada Hypervisor o VMM (Virtual Machine Monitor) Crea una capa de la abstracción entre el hardware de la maquina física (host) y el sistema operativo de la máquina virtual (virtual machine, guest). El VMM maneja los recursos de las maquinas físicas subyacentes (designadas por el computador central) de una manera que el usuario pueda crear varias máquinas virtuales presentando a cada una de ellas una interfaz del hardware que sea compatible con el sistema operativo elegido”[2].

5.1.1. Historia de la Virtualización.

“La virtualización se introdujo por primera vez en la década de 1960 por IBM para impulsar la utilización de grandes sistemas (mainframe caros) dividiéndolos en máquinas virtuales separadas lógicas que podían ejecutar múltiples aplicaciones y procesos al mismo tiempo. En los años 1980 y 1990, este modelo de mainframe centralizado y compartido dio paso a un modelo distribuido (la computación cliente-servidor) en el cual muchos servidores independientes x86 de bajo costo eran capaces de ejecutar aplicaciones específicas.

La virtualización se desvaneció como centro de atención durante un tiempo, ahora es una de las últimas tendencias en la industria una vez más, ya que las organizaciones tienen por objeto aumentar la utilización, la flexibilidad y la rentabilidad de sus recursos informáticos. VMware, Citrix, Microsoft, IBM, Red Hat y muchos otros proveedores ofrecen soluciones de virtualización”[2].

5.1.2. Virtualización de servidores.

La Virtualización de servidores se trata de particionar un servidor físico en dos o más servidores virtuales los cuales pueden interactuar de forma independiente con otros dispositivos, aplicaciones y usuarios como si se tratara de un servidor físico independiente, así mismo cada máquina virtual puede ejecutar un diferente SO (Sistema Operativo) y múltiples aplicaciones al mismo tiempo utilizando un solo servidor físico, debido a que cada servidor virtual se encuentra aislado de los demás servidores virtuales, si alguno de estos llega a tener fallas, este no afectará el funcionamiento de los demás servidores virtuales.

5.1.3. Ventajas de implementar virtualización de servidores.

- Reducir costos de mantenimiento de hardware.
- Aumento de eficiencia de utilización de espacio dentro del SITE
- Ejecución de más de un SO como máquinas virtuales en un único servidor físico.
- Portabilidad de máquinas virtuales.
- Carga de trabajo más rápida
- Rendimiento mayor de las aplicaciones

- Mayor disponibilidad,
- Administración centralizada.
- Control de asignación de recursos para cada servidor virtual.

5.1.4. Desventajas de la virtualización de servidores.

- Menor rendimiento al del hardware del servidor físico ya que una máquina virtual corre en una capa intermedia de este.
- No podrás utilizar hardware que no esté soportado por el hipervisor de virtualización.

5.1.5 Servidor de aplicaciones.

Son llamados así aquellos servidores cuya función es proporcionar un entorno de ejecución para un programa de aplicación, este servidor es considerado como el componente de tiempo de ejecución principal en las configuraciones y donde una aplicación es ejecutada. El servidor de aplicaciones trabaja de la mano con el servidor web para brindar una respuesta dinámica y personalizada para cada solicitud de cliente.

5.2. Software para realizar la virtualización.

5.2.1. Virtualbox.

Es un software de código abierto para crear virtualizaciones en arquitecturas x86, AMD64 e Intel 64. Esta aplicación está desarrollada por Oracle Corporation, esta es una aplicación multiplataforma la cual puede ser instalada en diversos SO, como lo son Linux, Windows, MacOS y Solaris.

El SO en el que sea instalado VirtualBox será conocido como host (anfitrión), y será nombrado como guest (huesped) al SO que sea instalado en cada máquina virtual.

5.3. Sistema Operativo (SO) implementado.

5.3.1. CentOS.

“CentOS Linux es una distribución respaldada por la comunidad derivada de fuentes proporcionadas libremente al público por Red Hat para Red Hat Enterprise Linux (RHEL). Como tal, CentOS Linux pretende ser funcionalmente compatible con RHEL. El Proyecto CentOS principalmente cambia los paquetes para eliminar la marca y la obra de arte del proveedor en la etapa inicial. CentOS Linux no tiene costo y se puede redistribuir libremente. Cada versión de CentOS se mantiene durante 10 años (mediante actualizaciones de seguridad: la duración del intervalo de soporte por parte de Red Hat ha variado con el tiempo con respecto a las Fuentes liberadas). Se lanza una nueva versión de CentOS aproximadamente cada 2 años y cada versión de CentOS se actualiza periódicamente (aproximadamente cada 6 meses) para admitir hardware más nuevo. Esto da como resultado un entorno Linux seguro, de bajo mantenimiento, confiable, predecible y reproducible” [4].

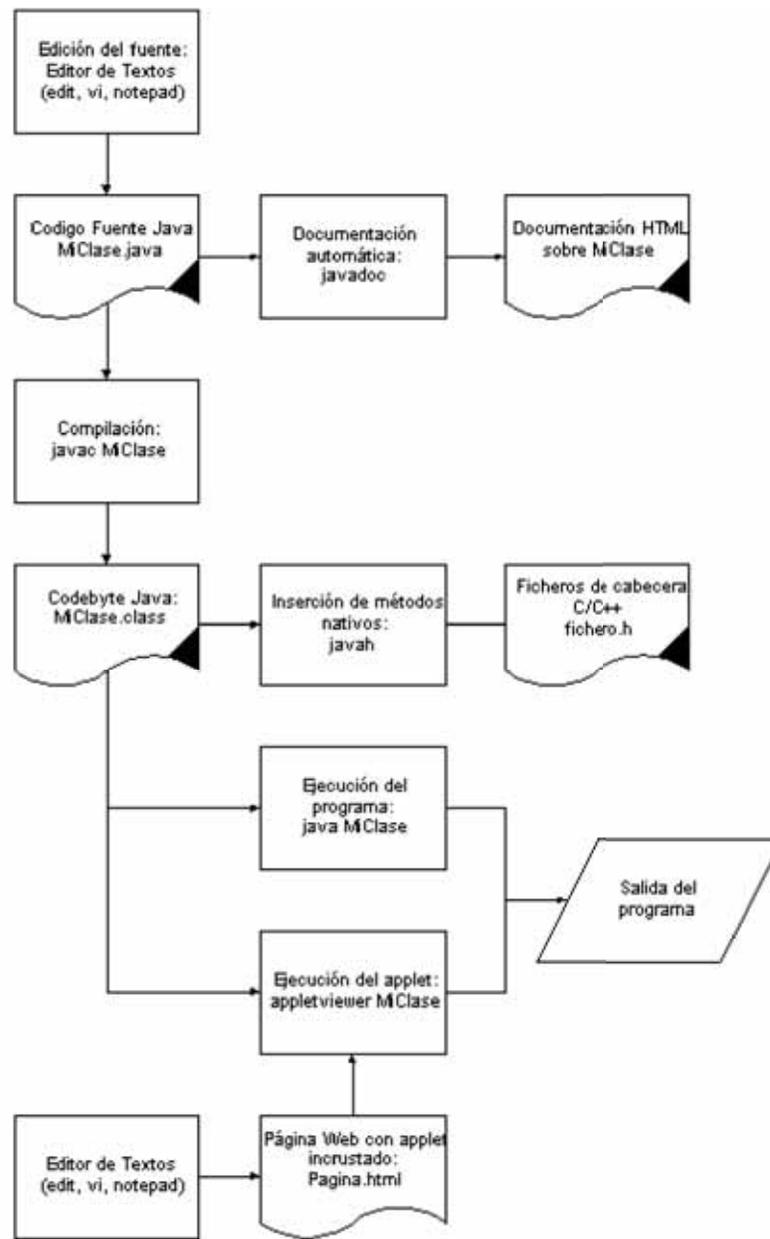
5.3.2 Servicios instalados.

JAVA (JDK)

JDK o Java Development Kit, como su nombre lo dice es un kit o conjunto de herramientas, documentación, utilidades y ejemplos para llevar a cabo el desarrollo de aplicaciones Java. Algunos de los componentes de JDK son:

- JRE o Interprete de tiempo de ejecución.
- Compilador
- Visualizador de Applets
- Depurador
- Generador de documentación

Para desarrollar una aplicación en Java debemos utilizar las herramientas que nos son proporcionadas por JDK en un determinado orden, el cual mostramos en la siguiente imagen.



Utilización del JDK [5].

MYSQL

MySQL MySQL es un sistema de administración de bases de datos (*DBMS*) para bases de datos relacionales. Podemos decir que es una aplicación que nos permite gestionar archivos mejor conocidos como bases de datos.

MySQL fue creado en los lenguajes de programación C y C++ ,lo que hace que esta aplicación se puede adaptar a diferentes entornos de desarrollo.

Dentro de las cosas más destacables de MYSQL está open source de MySQL, la cual nos permite utilizarla de manera gratuita, y con total libertad para modificar y descargar su código fuente, es por esto que MySQL, una de las herramientas más utilizadas por los programadores.

APACHE TOMCAT

El software Apache Tomcat es una implementación de código abierto de las tecnologías Java Servlet, JavaServer Pages, Java Expression Language y Java WebSocket. Las especificaciones Java Servlet, JavaServer Pages, Java Expression Language y Java WebSocket se desarrollan bajo Java Community Process.

El software Apache Tomcat se desarrolla en un entorno abierto y participativo y se publica bajo la versión 2 de Apache License. El proyecto Apache Tomcat pretende ser una colaboración de los mejores desarrolladores de todo el mundo.

El software Apache Tomcat potencia numerosas aplicaciones web de misión crítica a gran escala en una amplia gama de industrias y organizaciones. Algunos de estos usuarios y sus historias se enumeran en la página wiki de PoweredBy [9].

6. Descripción breve de la solución

Realizando el análisis de los servidores virtuales implantados por la DGTI, la Dirección de SI determinó que no cuentan con las especificaciones requeridas para poder ser utilizados, es por ello que se realizó la implantación de 3 servidores virtuales los cuales serán centralizados en un servidor físico, los servidores fueron virtualizados con la aplicación Virtual Box [8], con el fin de obtener compatibilidad y seguridad entre los servidores, estos se implementaron con SO CentOS 6.9 de 64 bits [9], los servidores virtuales se diseñaron con una memoria RAM de 32GB y un disco duro de 3TB el cual fue particionado como se muestra en la tabla 1.0.

	Partition Type	System Type	Size
Boot	Standard Partition	ext4	8 Gb
/	Physical Volumen	ext4	80 Gb
/usr	Physical Volumen	ext4	160 Gb
/tmp	Physical Volumen	ext4	40 Gb
/var	Physical Volumen	ext4	40 Gb
/var/lib	Physical Volumen	ext4	80 Gb
/var/www	Physical Volumen	ext4	80 Gb
/home	Physical Volumen	ext4	80 Gb
/boot	Physical Volumen	ext4	8 Gb
/backup	Physical Volumen	ext4	612 Gb
Swap	Physical Volumen	ext4	32 Gb

Tabla 1. Descripción breve de las particiones creadas.

La arquitectura que se implementó en los servidores fue una arquitectura basada en software libre con una configuración dividida en tres partes:

1. Parte física la cual contiene las particiones de volumen de grupo.
2. Parte lógica que contiene los volúmenes lógicos que están divididos en home, root, usr, swap, tmp, var, var_lib, var_www, var_spool.
3. Parte de aplicación la cual estará dividida en manejador de base de datos, motor virtual, http, ambiente web y capa de presentación.

Al concluir con la implementación del diseño de la arquitectura, se realizó la instalación y configuración del Sistema Operativo CentOS 6.9 de 64 bits, estas se realizaron de forma manual.

Cuando se dio por terminado la instalación y configuración de SO, se inició la instalación y configuración de los servicios requeridos por la Dirección de Sistemas de Información del CSG: Java (JDK) [7], Mysql [8] y TomCat [9].

Para optimizar los servicios instalados en los 3 servidores virtuales y mejorar su administración, se realizó una evaluación en la cual se listaron los servicios que requieren ser monitoreados para contar con un mantenimiento más recurrente, esto se obtuvo al implementar tres shell scripts:

El primer shell script se encarga de obtener la información de los recursos en memoria que utiliza cada servicio instalado en el servidor y la partición en la que se encuentra cada servicio.

El segundo shell script permite al administrador verificar el estatus de todos los servicios instalados en el servidor, este script realizará el reinicio, arranque y paro de cada servicio ya sea por separado o en conjunto.

El tercer shell script realiza la recuperación de un servidor web mediante respaldos periódicos y automáticos, para evitar la pérdida de información en caso de contingencia.

7. Descripción técnica

El proyecto comenzó a partir del diseño de los servidores virtuales de aplicaciones, esto a partir de la determinación de características, parámetros y funcionalidades de los 3 servidores virtuales de aplicaciones. En seguida se muestra paso a paso el proceso para llegar a la implementación de los servidores.

7.1. Diseño e implementación de servidores virtuales de aplicaciones.

Para el diseño e implementación de los servidores se respetaron los siguientes pasos:

Paso 1

7.1.1. Diseño de la Arquitectura de nuestros servidores

Este diseño se realizó en base a los requerimientos y necesidades de la Dirección de Sistemas de Información del CSG.

Cada servidor tiene una arquitectura basada en software libre con una configuración que estará dividida en tres partes:

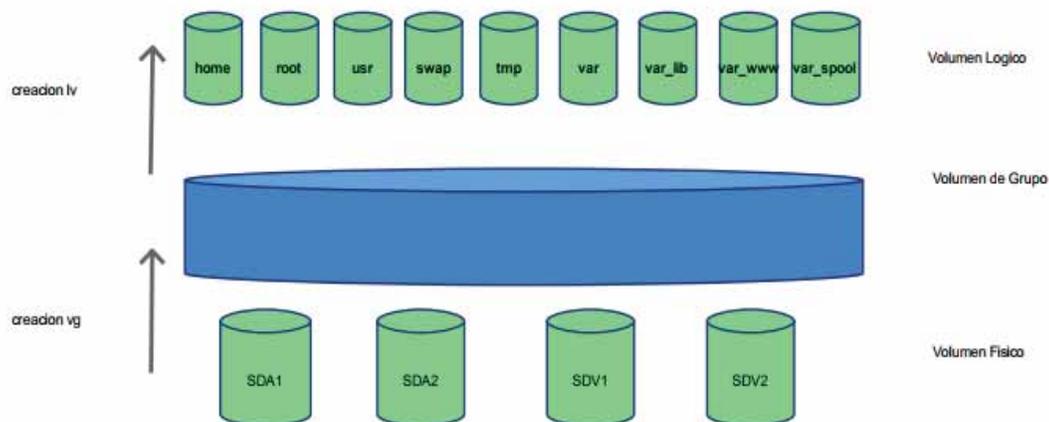


Imagen 1. Volúmenes lógicos y físicos

1. Parte física la cual contiene las particiones de volumen de grupo
2. Parte lógica que contiene los volúmenes lógicos que estarán divididos en:

/home: Almacena las carpetas personales de los usuarios. Cada directorio home de usuario contiene la configuración personalizada de las aplicaciones para ese usuario.

/root: Es la ubicación por defecto recomendada para directorio home de la cuenta de root.

/usr: Es una gran jerarquía dentro del sistema de archivos que contiene datos compartibles y de solo lectura.

/swap: Almacena la información cuando no hay suficiente memoria RAM para almacenar la información que su sistema está procesando.

/tmp: Esta carpeta debe estar disponible para los programas que requieren archivos temporales. Los programas no deben asumir que cualquier archivo o directorio en /tmp se preserven entre invocaciones del programa

/var: Es otra jerarquía que contiene archivos de datos variables. Esto incluye carpetas y archivos de spool, datos administrativos y de registro o archivos transitorios y temporales. La carpeta /var hace posible que /usr sea de solo lectura. Todo lo que una vez estuvo en /usr y que se puede modificar durante el funcionamiento del sistema debe ir a /var.

var_lib: Contiene la información variable del estado.

var_www: Contiene los datos de HTTP.

var_spool: Contiene datos de spool de aplicaciones [6].

3. Parte de aplicación: la cual está dividida en:
 - Manejador de base de datos
 - Motor virtual
 - Http
 - Ambiente web
 - Capa de presentación.

Tomando en cuenta los puntos anteriores pasamos a crear nuestras arquitecturas basadas en volúmenes lógicos, volúmenes físicos y volúmenes de grupo.

Mount Point	Partition	System Type	Size
/boot	Standard Partition	ext4	4GB
/	Physical Volumen	ext4	80GB
/usr	Physical Volumen	ext4	160GB
/tmp	Physical Volumen	ext4	40GB
/var	Physical Volumen	ext4	40GB
/var/lib	Physical Volumen	ext4	80GB
/var/www	Physical Volumen	ext4	80GB
/home	Physical Volumen	ext4	80GB
/backup	Physical Volumen	ext4	612GB
Swap	Physical Volumen	ext4	16GB
/var/spool	Physical Volumen	ext4	32GB

Tabla 2. Tipo de partición y tamaño asignado.

Volúmenes físicos asignados:

Partición	LVM	Name group volume	Asignación de partición	Mount Point	Fyle System Type
/boot	LVM Volumen Group	-	Sda1	/boot	ext4
/usr	LVM Volumen Group	vg_usr	Sda2	/usr	ext4
/	LVM Volumen Group	vg_root	Sda3	/	ext4
Home	LVM Volumen Group	vg_home	Sda4	/home	ext4
/tmp	LVM Volumen Group	vg_tmp	Sad5	/tmp	ext4
/var	LVM Volumen Group	vg_var	Sda6	/var	ext4
/var/lib	LVM Volumen Group	vg_var_lib	Sda7	/var/lib	ext4
/var/www	LVM Volumen Group	vg_var_www	Sda8	/var/www	ext4
Swap	LVM Volumen Group	vg_swap	Sda9	-	swap
/var/spool	LVM Volumen Group	vg_var_spool	Sda10	/var/Spool	ext4
/backup	LVM Volumen Group	Vg_backup	Sda11	/backup	ext4

Tabla 3. Asignación de volumen de grupo y de partición.

Los tamaños asignados a cada partición fueron creados a partir del esquema de particionamiento recomendado para sistemas x786, AMD64, e Intel64 con SO Red Hat citado en seguida:

- Una partición swap (de al menos 256 MB) — las particiones swap sirven para soportar la memoria virtual. En otras palabras, los datos se escriben en una partición swap cuando no hay suficiente memoria RAM para almacenar la información que su sistema está procesando.*

En años anteriores, la cantidad recomendada de espacio swap aumentaba en forma lineal con la cantidad de RAM en el sistema. No obstante, debido a que la cantidad de memoria en sistemas modernos ha aumentado a cientos de GB, ahora se reconoce que la cantidad de espacio swap que el sistema necesita es una función de la carga de trabajo de la memoria que se ejecuta en ese sistema.

El espacio swap suele designarse durante la instalación, aunque puede ser difícil determinar la carga de memoria de un sistema en ese momento. Durante una instalación de kickstart, puede solicitar que la cantidad de espacio swap se establezca de forma automática.

Sin embargo, esta configuración no está calibrada para su sistema, por lo tanto, use la siguiente tabla si requiere una cantidad de espacio de swap más precisa.

<i>Cantidad de RAM en el sistema</i>	<i>Cantidad recomendada de espacio swap</i>
<i>4GB de RAM o menos</i>	<i>Un mínimo de 2GB de espacio swap</i>
<i>De 4GB a 16GB de RAM</i>	<i>Un mínimo de 4GB de espacio swap</i>
<i>De 16GB a 64GB de RAM</i>	<i>Un mínimo de 8GB de espacio swap</i>
<i>De 64GB a 256GB de RAM</i>	<i>Un mínimo de 16GB de espacio swap</i>
<i>De 256GB a 512GB de RAM</i>	<i>Un mínimo de 32 GB de espacio swap</i>

- Una partición /boot (250 MB)*

La partición montada en /boot/ contiene el kernel del sistema operativo (el cual permite a su sistema arrancar Red Hat Enterprise Linux) junto con archivos utilizados durante el proceso de arranque. Para la mayoría de los usuarios, una partición de arranque de 250 MB es suficiente.

*Aquí es importante saber que el gestor de arranque **GRUB** en Red Hat Enterprise Linux 6 soporta únicamente los sistemas de archivos ext2, ext3, y ext4. No puede utilizar ningún otro sistema de archivo para /boot, tales como Btrfs, XFS, o VFAT.*
- Una partición root (3.0 GB - 5.0 GB)*

Aquí es donde se localiza "/" (el directorio raíz). En esta configuración, todos los archivos (excepto aquellos almacenados en /boot) están en la partición raíz.

Una partición de mínimo 3.0 GB le permite hacer una instalación mínima.

Una partición de mínimo 5.0 GB le permite realizar una instalación completa, seleccionando todos los grupos de paquetes.

Cabe destacar que La partición / (o raíz) es la cima de la estructura del directorio. El directorio /root es el directorio principal de la cuenta del usuario-administrador del sistema.

- *Una partición home (al menos de 100 MB)*
Para almacenar datos de forma independiente de los datos del sistema, cree una partición dedicada dentro de un grupo de volumen para el directorio /home. Así podrá actualizar o reinstalar Red Hat Enterprise Linux sin borrar archivos de datos de los usuarios.
- *Una partición para el directorio /var que guarda contenido para una cantidad de aplicaciones que incluyen al servidor web Apache. También se usa para almacenar los paquetes descargados para actualizaciones de forma temporal. Asegure que la partición que contiene al directorio /var tenga suficiente espacio para descargar las actualizaciones pendientes y alojar el resto de los otros contenidos.*
- *Una partición para El directorio /usr el cual guarda la mayoría del contenido del software en un sistema Red Hat Enterprise Linux. Para una instalación con el conjunto estándar de software, asigne al menos 4 GB de espacio. Si usted es un desarrollador de software o planea usar Red Hat Enterprise Linux para aprender y desarrollar sus habilidades en ese sentido, debe considerar al menos doblar ese tamaño.*

Es importante tomar en cuenta que, si /usr está en un sistema de archivos independiente de /, el proceso de arranque se torna mucho más complejo debido a que /usr contiene componentes de arranque críticos. En algunas situaciones (como instalaciones sobre unidades (SCSI), el proceso podría no funcionar.

En la siguiente tabla se muestra el tamaño mínimo de las particiones de nuestros directorios mencionados anteriormente [6].

Directorio	Tamaño mínimo
/	250 MB
/usr	250 MB,
/tmp	50 MB
/var	384 MB
/home	100 MB
/boot	250 MB

En cada partición, el sistema de archivos ext4 fue elegido debido a que cuenta con un mayor soporte para archivos muy grandes, la asignación de espacio de disco es más rápida y eficiente, no limita el número de subdirectorios dentro de un directorio.

Al concluir el diseño de la arquitectura de nuestros servidores, se realizó la instalación y configuración del Sistema Operativo CentOS 6.9 de 64 bits:

Paso 2

7.1.2. Preparación del software que utilizaremos.

- Configuraremos una USB como dispositivo primario de booteo la cual contendrá el iso del sistema operativo a instalar, en nuestro caso será CentOS 6.9 de 64 bits.
- Instalaremos la aplicación VirtualBox en nuestro host el cual tiene SO Windows 10.

Paso 3

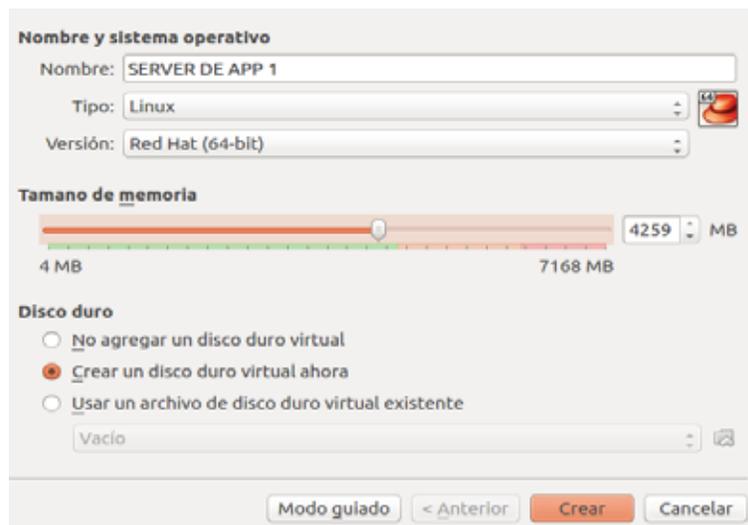
7.1.3. Virtualización.

Para llevar a cabo la virtualización abriremos la aplicación VirtualBox.



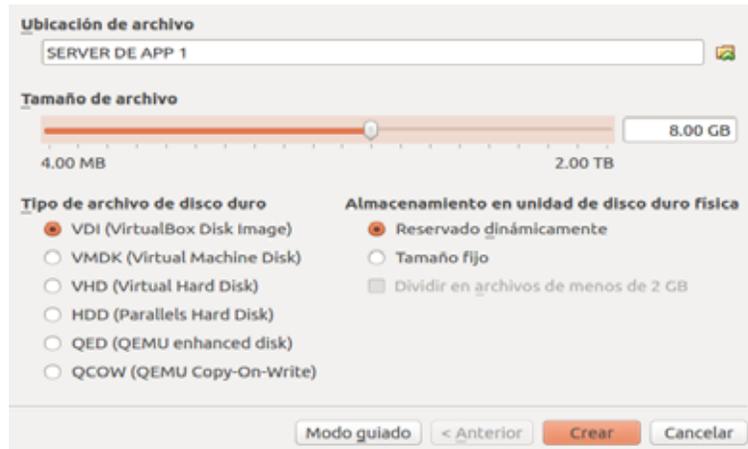
Daremos **click** en la opción NUEVA y se nos abrirá una ventana en la que seleccionaremos Modo experto e indicaremos:

- Nombre
- Tipo, en el cual elegiremos Linux
- Versión, elegiremos RedHat (64 bits)
- Tamaño de memoria (RAM) 32 GB



Al terminar de ingresar los datos que se nos piden, seleccionaremos “Crear un disco duro virtual ahora”, y daremos **click** en Crear.

En seguida se abrirá una segunda ventana, en la que se nos permitirá ajustar el tamaño de archivo de disco duro, en cual será de 3TB, posteriormente seleccionaremos VDI (VirtualBox Disk Image) como tipo de archivo de disco duro para crear una nueva máquina virtual con un disco nuevo.



En almacenamiento en unidad de disco duro física seleccionaremos “reservado dinámicamente”, y daremos **click** en crear.

Hasta este punto la máquina virtual ya ha sido creada, y está lista para iniciar la instalación y configuración del SO elegido.



Paso 3

7.1.3 Instalación y configuración del SO.

Ingresaremos a VirtualBox, seleccionaremos la máquina que creamos en el paso anterior y daremos **click** en INICIAR.

Automáticamente se abrirá una ventana la cual nos pedirá elegir el disco de inicio, ya elegido daremos **click** en iniciar.

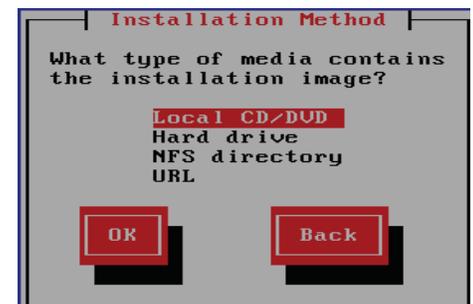
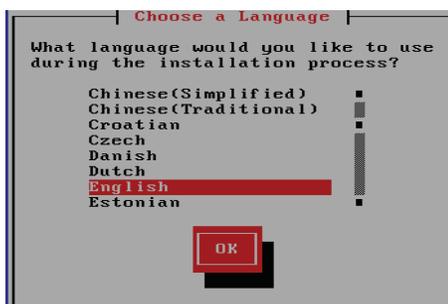


En seguida se nos mostrara una ventana, en la que elegiremos el tipo de instalación, en nuestro caso ya que realizaremos la configuración y preparación del disco duro presionaremos **Tab**, con esto el cursor se posicionara en la parte inferior de la pantalla, ahí agregaremos “**askmethod**” (para seleccionar el método de instalación) y presionaremos **Enter**, con esto iniciaremos la primera configuración del SO.

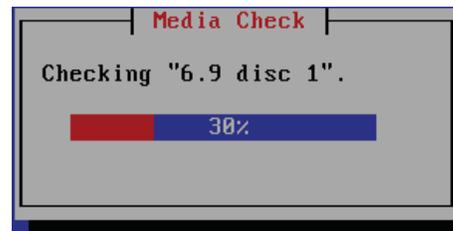
Primera configuración.

En la primera configuración elegiremos:

- Idioma- Español o Ingles en su defecto.
- Tipo de teclado- la -latin1 o el que corresponda a la distribución del teclado.
- Método de instalación- Local CD/DVD



Terminando la configuración inicial realizaremos la prueba para verificar que estén en buenas condiciones los CD's o DVD's de instalación, para evitar errores de instalación.



Al termino del test daremos **click** en continúe, aquí nos aparecerá una pantalla azul con un menú en la parte superior, elegiremos el menú “Dispositivos” y en este seleccionaremos “Unidades óptimas” y daremos **click** en el .iso que estamos instalando, de esta manera reconectaremos el disco y emergerá una pantalla gris la que nos indicará que la reconexión esta lista, al término de esta daremos **click** en OK y seguiremos con la instalación del SO.

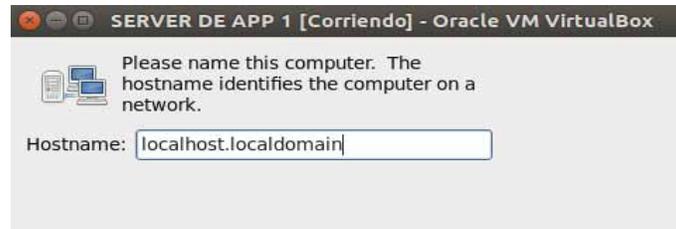
Al concluir la reconexión de nuestro CD ó DVD de instalación, se lanzará la pantalla inicial del SO, aquí indicaremos **NEXT** para iniciar con la configuración.



En la primera pantalla de configuración elegiremos el tipo de disco a utilizar, ya que se realizará todo manualmente elegiremos la opción “Basic Storage Device”, debido a que la maquina es virtual nos aparecerá la siguiente pantalla, en esta elegiremos “No keep any data”.



Ahora se nos pide proporcionar el “hostname” del servidor,



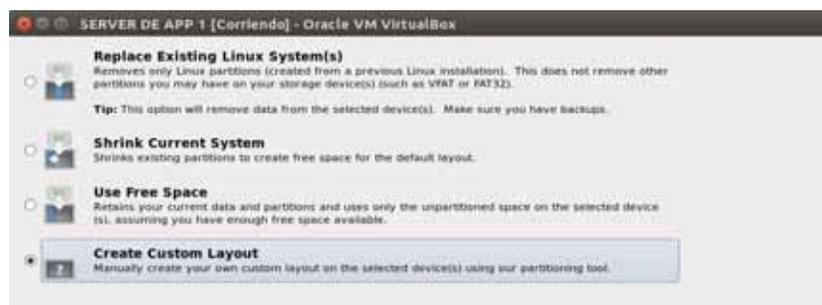
Ya ingresado configuraremos la Red, daremos **click** en Configure Network, en esta ventana elegiremos la interface de Ethernet System eth0 y el protocolo de internet version4 (IPV4), en Settings elegiremos Automatic (DHCP) para que le sea asignada una IP automáticamente.

Teniendo la configuración de la red pasaremos a la siguiente configuración, aquí indicaremos hora, ciudad y posteriormente se nos pedirá indicar la contraseña de root, al terminar de ingresar estos datos pasaremos a la configuración de las particiones.



Configuración y asignación de particiones.

Para configurar las particiones elegiremos la opción Create custom layout, aquí sí el disco es IDE los discos emplean el archivo de dispositivo /dev/hda para el primer disco, /dev/hdb para el segundo, etc. Si el disco es SCSI (el cual es nuestro caso) el archivo de dispositivo es /dev/sda para el primer disco, /dev/sdb para el segundo, etc.



Al elegir crear un diseño personalizado realizaremos las particiones por medio de la creación y configuración de LVM.

Los LVM o Administradores de volúmenes lógicos representan las particiones en almacenamiento físico las cuales pueden agruparse en grupos de volumen. Cada uno de estos grupos de volumen puede dividirse en varios volúmenes lógicos, es por ello que los LVM funcionan como particiones, los cuales pueden abarcar más de un disco duro.

Es en este punto en el que crearemos nuestro sistema de archivos (File system) con administradores de volúmenes lógicos, los cuales quedaran de acuerdo a la arquitectura diseñada en el punto 1, y la cual mostramos en las tablas 2 y 3 mostradas anteriormente.

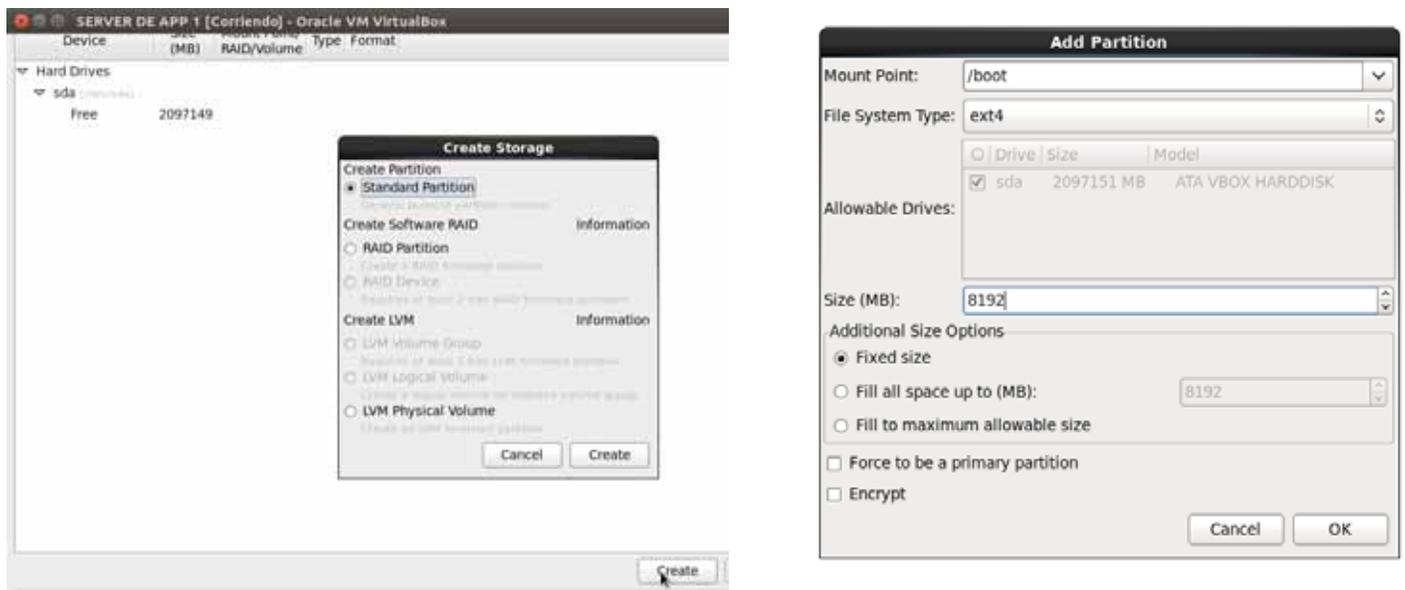
Para crear el File system, iniciaremos creando la partición para nuestro directorio /boot, lo primero que haremos es dar **click** en “create”, al hacerlo emergerá una pantalla en la que elegiremos el tipo de partición, la cual en este caso indicaremos como una partición estándar en la que se montara nuestro directorio, para ello en la pantalla que emergerá al crear la partición estándar ingresaremos:

Como punto de montaje a nuestro directorio /boot

Como tipo de sistema de archivos ext4

y por ultimo Indicaremos tamaño fijo en la opción de tamaño adicional.

Ya ingresados los datos anteriores daremos **click** en **OK** y tendremos nuestra primera partición lista.



Para las particiones que restan crearemos volúmenes físicos, para ellos daremos **click** en “create” nuevamente y aquí elegiremos “LVM Physical Volume” como formato de partición.

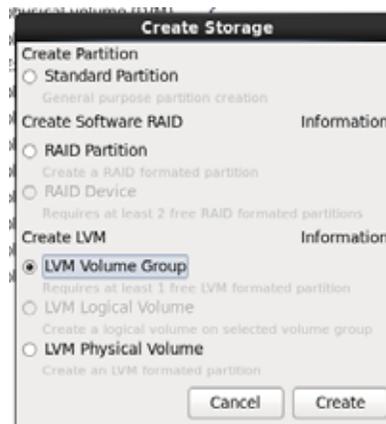
Al dar **click** en crear, emergerá una pantalla mas en la que se nos permitirá elegir el formato y tamaño de cada partición.



Al terminar de crear Nuestros “LVM Physical Volumen”, el Fyle system deberá quedar como se muestra en la siguiente imagen.

Device	(MB)	RAID/Volume	Type	Format
Hard Drives				
▼ sda (cdrom:sda)				
sda1	8190	/boot	ext4	✓
sda2	626688		physical volume (LVM)	✓
sda3	163840		physical volume (LVM)	✓
▼ sda4	1298432		Extended	
sda5	81920		physical volume (LVM)	✓
sda6	81920		physical volume (LVM)	✓
sda7	81920		physical volume (LVM)	✓
sda8	81920		physical volume (LVM)	✓
sda9	40960		physical volume (LVM)	✓
sda10	40960		physical volume (LVM)	✓
sda11	32768		physical volume (LVM)	✓
sda12	16384		physical volume (LVM)	✓
Free	839671			

Ya creados los volúmenes físicos, crearemos los volúmenes de grupo, para ello seleccionaremos nuevamente “create” y aquí elegiremos “LVM Volume Group”.



A cada “LVM Volumen Group” creado se le indicara:

Nombre de volumen de grupo.

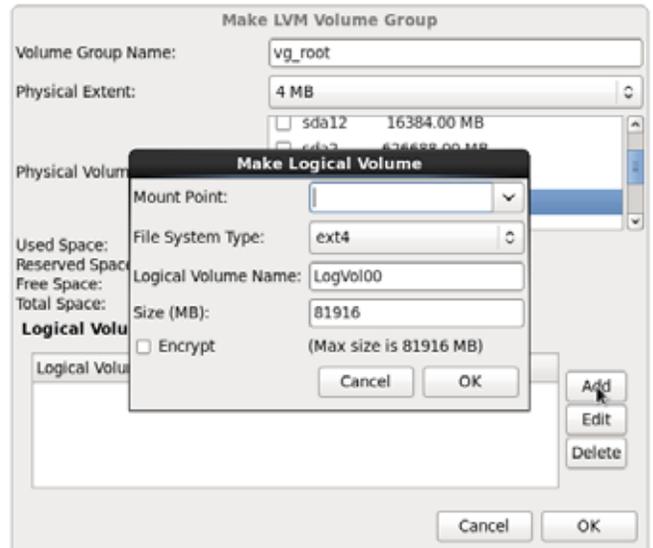
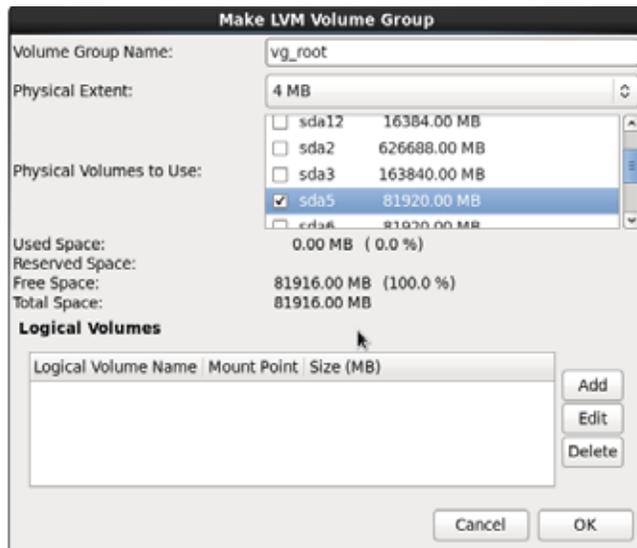
Volumen físico a usar.

Al tener estos dos datos se dará **click** en la pestaña “Add” Y se le añadirá:

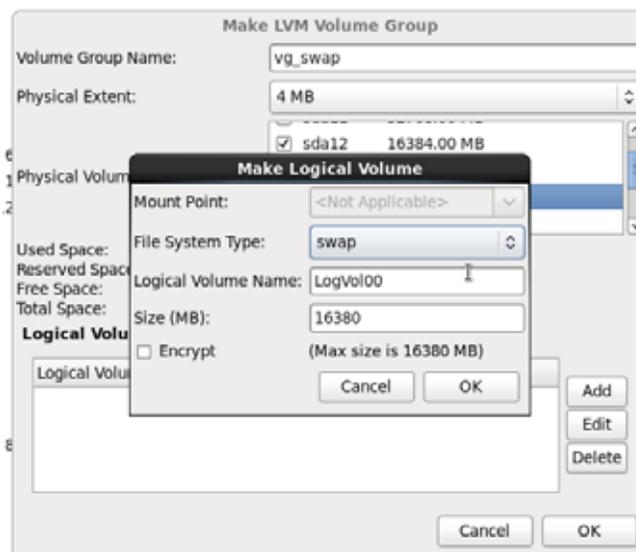
Punto de montaje

Tipo de sistema de archivos, y

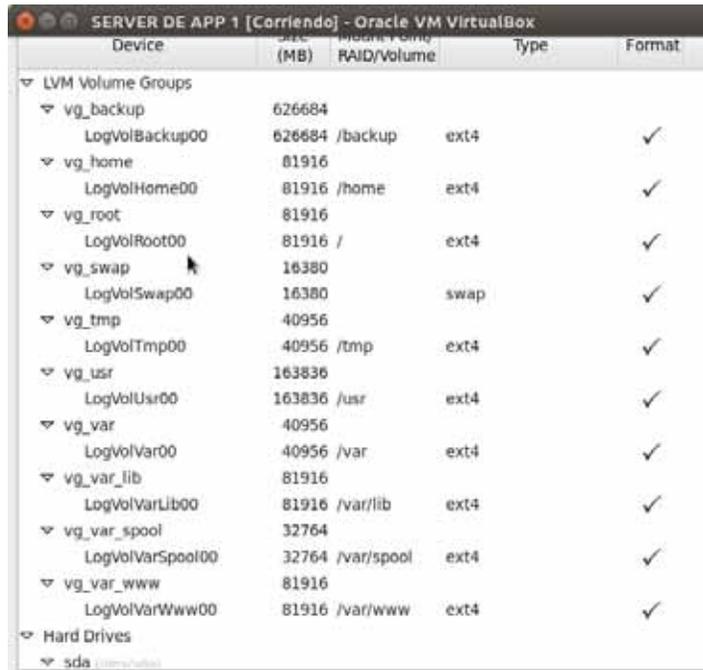
Nombre del volumen lógico



En el caso de /swap seguiremos los pasos anteriores, a excepción del tipo de sistema de archivos, aquí seleccionaremos “swap”.



Terminando de asignar los volúmenes de grupo, hemos terminado de crear las particiones y el file system nos deberá quedar de la siguiente manera.



Device	Size (MB)	RAID/Volume	Type	Format
LVM Volume Groups				
vg_backup	626684			
LogVolBackup00	626684	/backup	ext4	✓
vg_home	81916			
LogVolHome00	81916	/home	ext4	✓
vg_root	81916			
LogVolRoot00	81916	/	ext4	✓
vg_swap	16380			
LogVolSwap00	16380		swap	✓
vg_tmp	40956			
LogVolTmp00	40956	/tmp	ext4	✓
vg_usr	163836			
LogVolUsr00	163836	/usr	ext4	✓
vg_var	40956			
LogVolVar00	40956	/var	ext4	✓
vg_var_lib	81916			
LogVolVarLib00	81916	/var/lib	ext4	✓
vg_var_spool	32764			
LogVolVarSpool00	32764	/var/spool	ext4	✓
vg_var_www	81916			
LogVolVarWww00	81916	/var/www	ext4	✓
Hard Drives				
sda				

Para pasar al siguiente paso solo necesitamos dar **click** en “NEXT”, y nos llevara a elegir donde se instalara el gestor de arranque, se nos mostrara una lista de sistemas operativos, es nuestro caso solo contamos con un SO así que lo elegiremos para instalar el gestor de arranque de arranque.



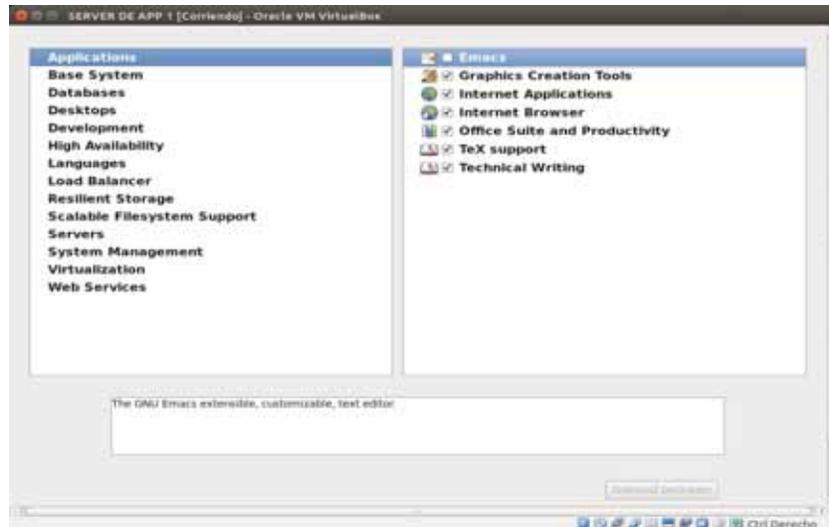
Selección e instalación de las aplicaciones del servidor

Terminando de crear las particiones y de elegir donde instalar el gestor de arranque, seleccionaremos el modo de instalación personalizada para elegir los paquetes que requieren ser instalados para el servidor de aplicaciones.

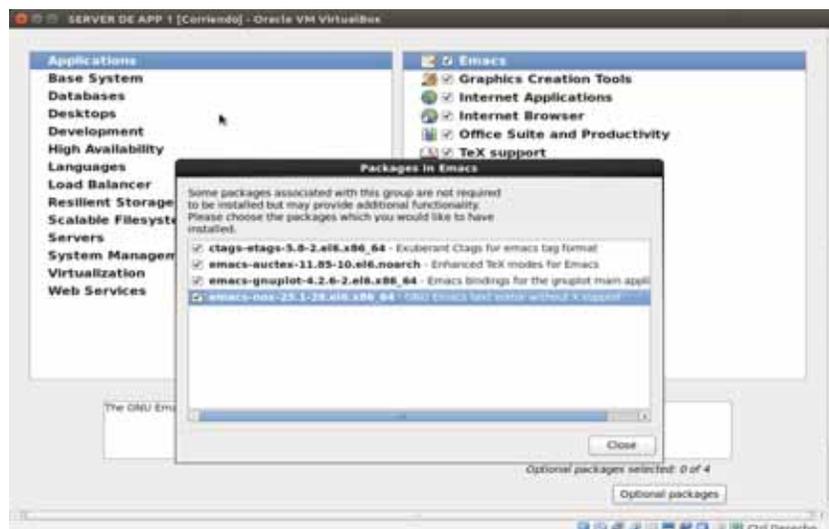


Aquí podemos ver que se cuenta con diferentes grupos de paquetes en los cuales elegiremos los que necesite nuestro servidor, cada grupo de paquetes cuenta con una lista en la que indica los nombres de los paquetes y la función de cada uno de estos.

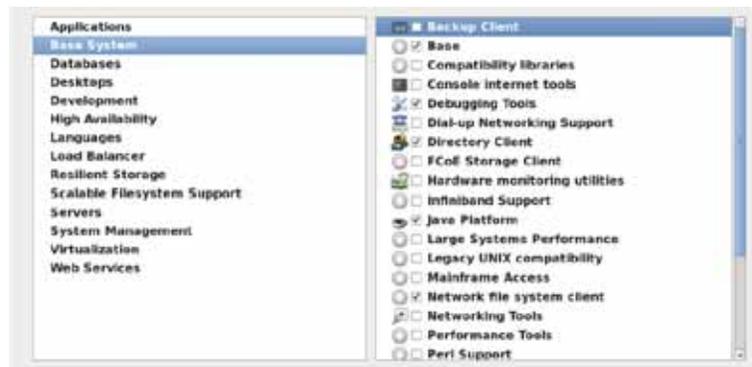
El primer grupo de paquetes que nos aparecerá es Aplicaciones.



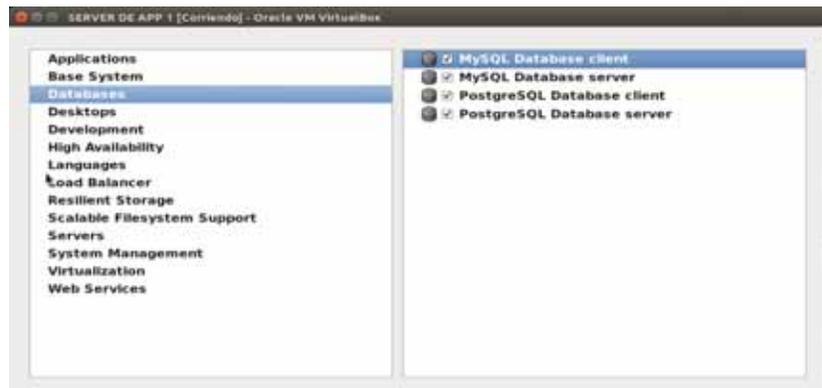
Elegimos todos los paquetes que necesitamos, estos al ser elegidos mostrara el número de paquetes asociados a este grupo y se habilitara la pestaña “Optional packages” al dar **click** sobre esta se mostrara una pantalla en la que se listaran los paquetes asociados que podemos instalar, ya elegidos estos cerramos la pantalla y repetimos el procedimiento con todos los grupos de paquetes que seleccionaremos.



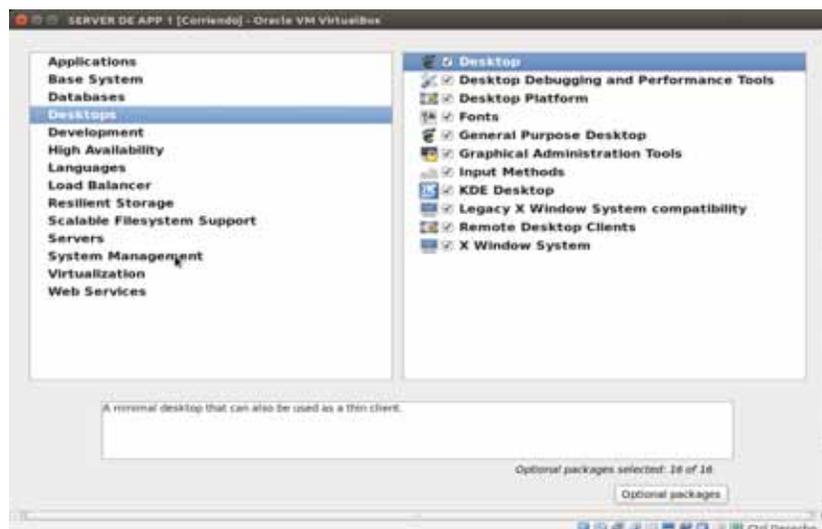
Base System: Elegiremos todos los paquetes que creamos convenientes, tomando en cuenta que hay paquetes que no utilizaremos como Smart card Support, y seleccionarlos para ser instalados solo generaría desperdiciar memoria.



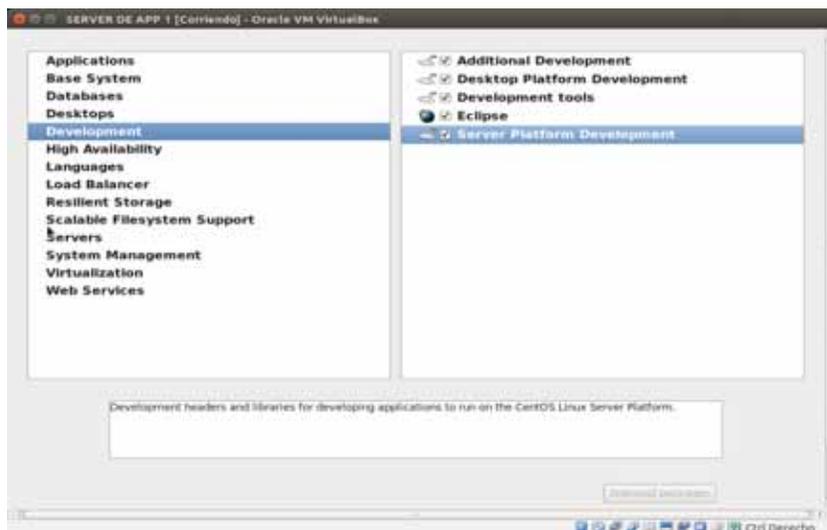
Databases: En esta pestaña elegiremos todos los paquetes ya que estas herramientas son indispensables y nos serán útiles.



Desktop: Al igual que en Databases elegiremos todos los paquetes.



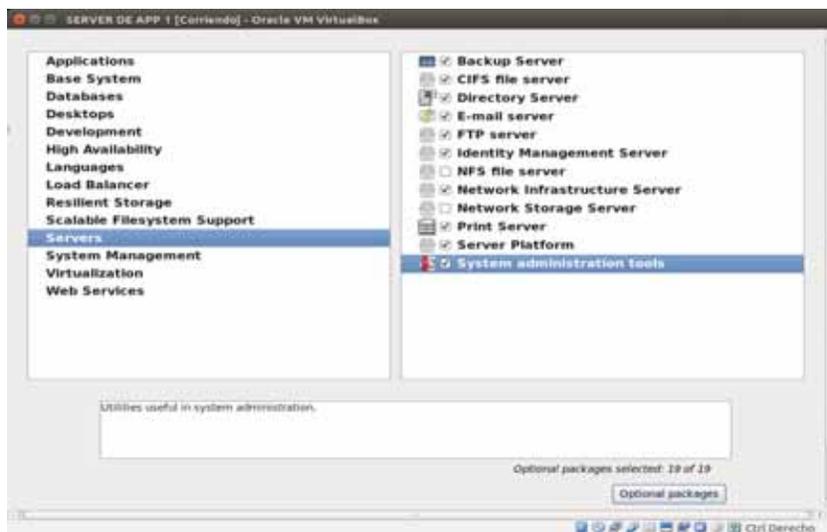
Development y High Availability: Repetimos el paso anterior seleccionando todos los paquetes.



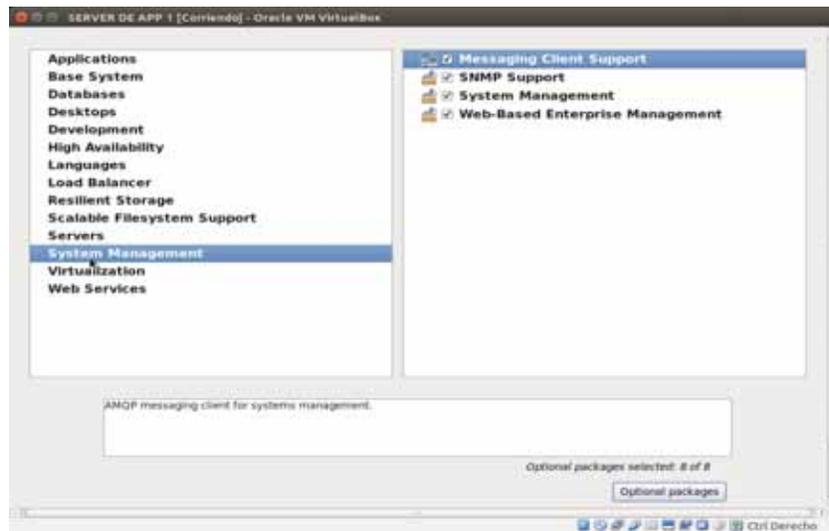
En la parte de Languages elegiremos los lenguajes que deseamos que soporte nuestro servidor, en este caso ya que viene por default Ingles, solo seleccionaremos los paquetes de Spanish Support.

En las secciones de Load Balancer, Resilient Storage y Scalable Filesystem Support elegiremos todos los paquetes disponibles.

En Servers elegiremos los servidores que instalaremos.



En System Management y Web service, seleccionaremos todos los paquetes ya que son paquetes indispensables para el funcionamiento del servidor, omitiremos Virtualización ya que no nos será útil.



Al terminar nuestra selección de paquetes procedemos su instalación, para ello solo requerimos hacer **click** en “Next” de esta manera se revisa la dependencia de estos y se procederá a su instalación.



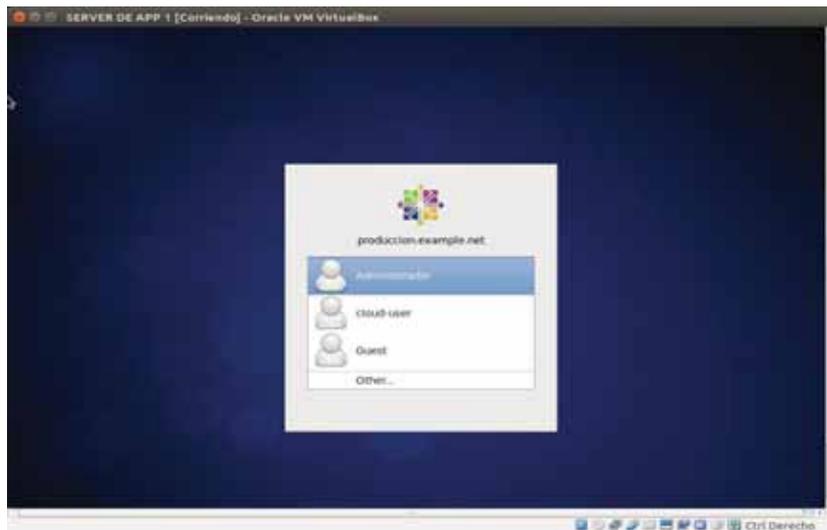
Al terminar la instalación de los paquetes que elegimos, se nos pedirá reiniciar la máquina. Daremos **click** en “reboot” y esperaremos a que el servidor virtual se reinicie.



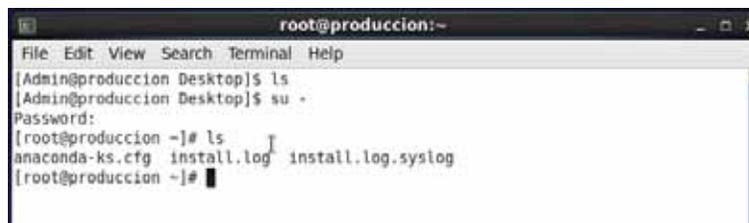
Al iniciar por primera vez el servidor nos dará la bienvenida y nos pedirá leer la licencia y aceptarlos términos para pasar a hacer configuraciones básicas como la creación de usuario, fijar la fecha y hora del sistema.



En la última pestaña, que es “Kdump”, se nos permitirá generar una imagen del Kernel, aquí damos **click** en finish y se nos enviara a la pantalla de inicio de nuestro SO, para acceder solo elegiremos el usuario que creamos e indicamos su contraseña.



Al ingresar se nos mostrara el escritorio de nuestro servidor, abriremos una terminal para verificar lo que tiene instalado nuestro sistema, para ello ingresaremos como usuario root, y posteriormente utilizaremos el comando **ls** para que se nos muestre lo instalado hasta el momento.



En la imagen anterior podemos ver que contamos con anaconda el cual es nuestro programa de instalación estándar.

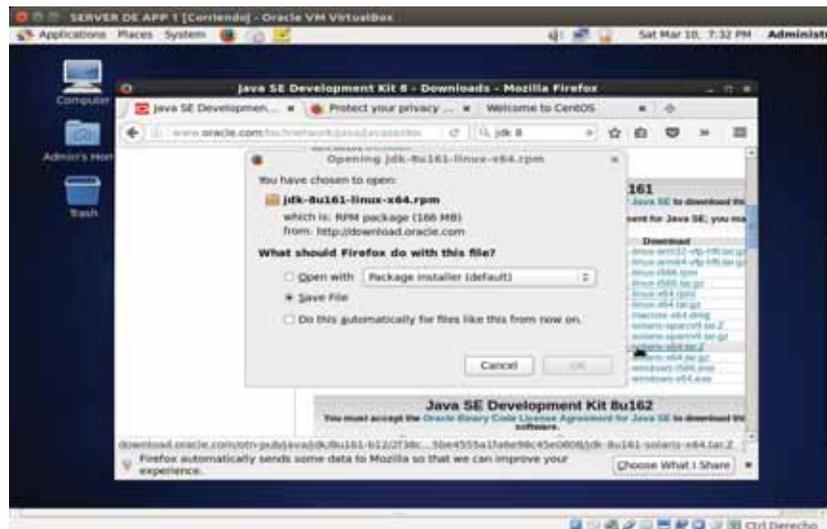
A partir de aquí comenzaremos la instalación y configuración de las herramientas que utilizaremos para desarrollo.

7.1.4. Instalación y configuración de Herramientas de desarrollo

Tenemos diferentes formas para realizar la descarga e instalación de las herramientas, ya sea descargándolas desde la máquina virtual, o en caso de descargar las herramientas en otra máquina solo necesitaran cargarlas en un dispositivo USB y copiarlas a la máquina virtual, y proceder a la instalación por medio de una terminal.

Instalación y configuración de JDK.

Como primer paso tendremos de ingresar a la página de Oracle y descargar la versión más reciente de java para Linux, aquí elegimos el paquete `jdk-8u161linux-x64.rpm`.



Ya teniendo JDK abrimos la terminal y verificamos el contenido de la carpeta, para ello utilizamos el comando `rpm -qa|grep jdk`



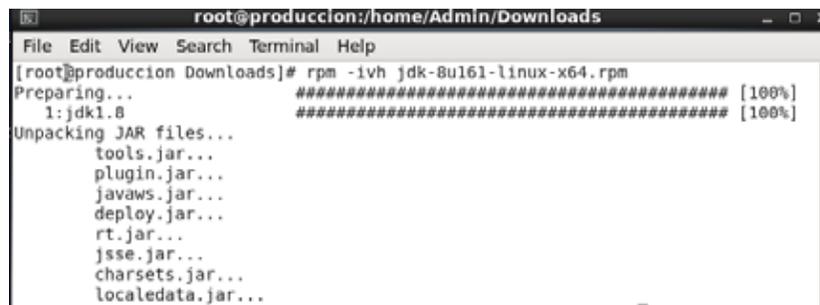
Ingresamos como súper usuario.

su -

Ingresamos al directorio Download en el que se encuentra el paquete jdk que descargamos, ya estando en esta carpeta ingresamos el comando **ll** para listar el contenido y sus parámetros.

```
[Admin@produccion DeskTоп]$ su -
Password:
[root@produccion ~]# cd ~Admin/Downloads/
[root@produccion Downloads]# ll
total 170076
-rw-rw-r--. 1 Admin Admin 174157387 Mar 10 19:37 jdk-8u161-linux-x64.rpm
[root@produccion Downloads]#
```

Instalamos jdk con el comando **rpm -ivh**

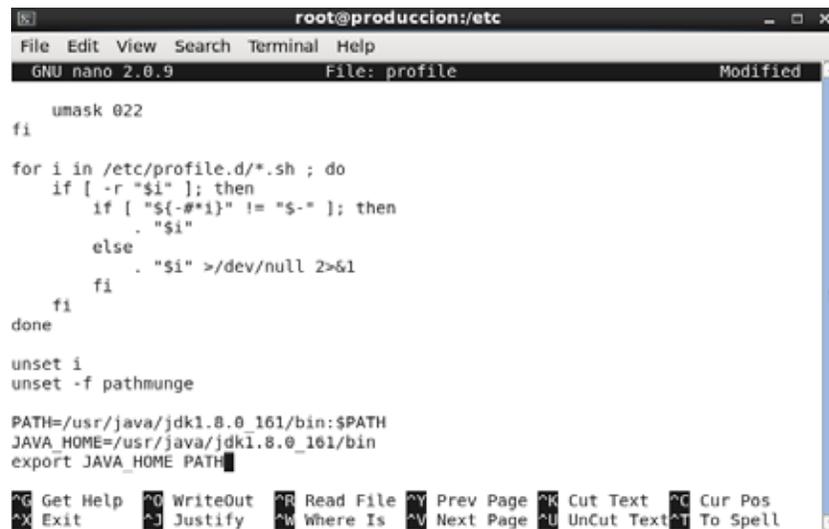


```
root@produccion:/home/Admin/Downloads
File Edit View Search Terminal Help
[root@produccion Downloads]# rpm -ivh jdk-8u161-linux-x64.rpm
Preparing... ##### [100%]
1: jdk1.8 ##### [100%]
Unpacking JAR files...
  tools.jar...
  plugin.jar...
  javaws.jar...
  deploy.jar...
  rt.jar...
  jsse.jar...
  charsets.jar...
  localedata.jar...
```

Verificamos su instalación y el contenido de /usr/java ingresando el comando **ll /usr/java**

```
[root@produccion Downloads]# ll /usr/java
total 4
lrwxrwxrwx. 1 root root 16 Mar 10 20:06 default -> /usr/java/latest
drwxr-xr-x. 9 root root 4096 Mar 10 20:05 jdk1.8.0_161
lrwxrwxrwx. 1 root root 22 Mar 10 20:06 latest -> /usr/java/jdk1.8.0_161
[root@produccion Downloads]#
```

Para terminar con la instalación agregaremos la ruta `/usr/java/jdk1.8.0_161/bin` a la variable PATH en el archivo `/etc/profile`



```
root@produccion:/etc
File Edit View Search Terminal Help
GNU nano 2.0.9 File: profile Modified

    umask 022
fi
for i in /etc/profile.d/*.sh ; do
    if [ -r "$i" ]; then
        if [ "${-#*i}" != "$-" ]; then
            . "$i"
        else
            . "$i" >/dev/null 2>&1
        fi
    fi
done

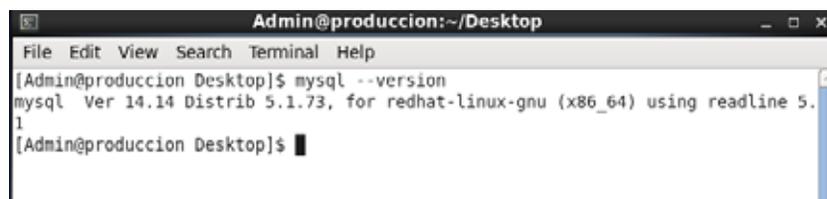
unset i
unset -f pathmunge

PATH=/usr/java/jdk1.8.0_161/bin:$PATH
JAVA_HOME=/usr/java/jdk1.8.0_161/bin
export JAVA_HOME PATH
```

Guardamos los cambios y es en este momento que ya tenemos instalada la herramienta jdk, podemos verificar la versión instalada con el comando `java -version`.

Instalación y configuración de MySQL.

Llegando a este punto recordaremos que MySQL ya está instalado ya que fue uno de los paquetes que se eligieron en la configuración del sistema, revisaremos la versión con la que ya se cuenta y .



```
Admin@produccion:~/Desktop
File Edit View Search Terminal Help
[Admin@produccion Desktop]$ mysql --version
mysql Ver 14.14 Distrib 5.1.73, for redhat-linux-gnu (x86_64) using readline 5.1
[Admin@produccion Desktop]$
```

En la imagen anterior observamos que tenemos instalado MySQL 5.1, por lo que necesitaremos actualizarlo para que se pueda realizar más adelante la instalación de herramientas de desarrollo y se tenga compatibilidad.

Para iniciar la actualización abriremos una terminal e ingresaremos como súper usuario **su -**, descargaremos e instalaremos el repositorio de MySQL Yum, el cual proporciona paquetes RPM para poder instalar la versión más reciente del servidor MySQL. Ingresaremos en la línea de comandos **yum localinstall https://dev.mysql.com/get/mysql57-community-release-el6-11.noarch.rpm**

```
root@produccion:~  
File Edit View Search Terminal Help  
[root@produccion ~]# yum localinstall https://dev.mysql.com/get/mysql57-community-  
release-el6-11.noarch.rpm  
Loaded plugins: aliases, changelog, fastestmirror, kabi, ovl, presto, refresh-  
: packagekit, security, tmprepo, verify, versionlock  
Loading support for CentOS kernel ABI  
Setting up Local Package Process  
mysql57-community-release-el6-11.noarch.rpm | 25 kB 00:00  
Examining /var/tmp/yum-root-EhmlSB/mysql57-community-release-el6-11.noarch.rpm: my  
sql57-community-release-el6-11.noarch  
Marking /var/tmp/yum-root-EhmlSB/mysql57-community-release-el6-11.noarch.rpm to be  
installed  
Loading mirror speeds from cached hostfile  
* base: mirror.cenac.ipn.mx  
* epel: mirror.prgmr.com  
* extras: mirrors.sonic.net  
* updates: mirror.scalabledns.com  
Resolving Dependencies  
--> Running transaction check  
--> Package mysql57-community-release.noarch 0:el6-11 will be installed  
--> Finished Dependency Resolution  
Dependencies Resolved
```

Ya contando con el repositorio Yum instalaremos MySQL 5.7 la cual es la versión más reciente. Ingresaremos **yum install mysql-community-server** en nuestra línea de comandos

```
Complete!  
[root@produccion ~]# yum install mysql-community-server  
Loaded plugins: aliases, changelog, fastestmirror, kabi, ovl, presto, refresh-  
: packagekit, security, tmprepo, verify, versionlock
```

Al instalar MySQL 5.7, también se verifican, reemplazan e instalan más paquetes como se muestra en la siguiente imagen.

```
root@produccion:~  
File Edit View Search Terminal Help  
Verifying : mysql-community-test-5.7.21-1.el6.x86_64 8/16  
Verifying : mysql-community-libs-compat-5.7.21-1.el6.x86_64 9/16  
Verifying : mysql-bench-5.1.73-8.el6_8.x86_64 10/16  
Verifying : mysql-server-5.1.73-8.el6_8.x86_64 11/16  
Verifying : mysql-connector-odbc-5.1.5r1144-7.el6.x86_64 12/16  
Verifying : mysql-devel-5.1.73-8.el6_8.x86_64 13/16  
Verifying : mysql-test-5.1.73-8.el6_8.x86_64 14/16  
Verifying : mysql-5.1.73-8.el6_8.x86_64 15/16  
Verifying : mysql-libs-5.1.73-8.el6_8.x86_64 16/16  
Installed:  
mysql-community-client.x86_64 0:5.7.21-1.el6  
mysql-community-devel.x86_64 0:5.7.21-1.el6  
mysql-community-libs.x86_64 0:5.7.21-1.el6  
mysql-community-libs-compat.x86_64 0:5.7.21-1.el6  
mysql-community-server.x86_64 0:5.7.21-1.el6  
mysql-community-test.x86_64 0:5.7.21-1.el6  
Dependency Installed:  
mysql-community-common.x86_64 0:5.7.21-1.el6 perl-JSON.noarch 0:2.15-5.el6  
Dependency Updated:  
mysql-connector-odbc.x86_64 0:5.3.10-1.el6  
Replaced:  
mysql.x86_64 0:5.1.73-8.el6_8 mysql-bench.x86_64 0:5.1.73-8.el6_8  
mysql-devel.x86_64 0:5.1.73-8.el6_8 mysql-libs.x86_64 0:5.1.73-8.el6_8  
mysql-server.x86_64 0:5.1.73-8.el6_8 mysql-test.x86_64 0:5.1.73-8.el6_8  
Complete!  
[root@produccion ~]#
```

Ahora levantaremos el servicio de MySQL, y empezaremos por inicializar la base de datos MySQL, para ello ingresaremos al directorio /etc., al archivo init.d e indicaremos start. Ya inicializada la base de datos verificaremos el estatus.

```
Complete!
[root@produccion ~]# /etc/init.d/mysql start
Initializing MySQL database: [ OK ]
/usr/bin/mysql_safe: line 517: cd: /root: Not a directory [ OK ]
Starting mysqld:
[root@produccion ~]# service mysqld status
mysqld (pid 8106) is running...
[root@produccion ~]# service mysqld stop
Stopping mysqld: [ OK ]
[root@produccion ~]# service mysqld start
Starting mysqld: [ OK ]
```

En versiones anteriores de MySQL al inicializar la base de datos nos mostraba la contraseña establecida para el usuario root de MySQL y nos pedía hacer la configuración con el script mysql_secure_installation.

En este nuestro caso no nos mostró la contraseña de root así que haremos la recuperación.

Iniciaremos apagando nuestro servicio con:

service mysqld stop

Posteriormente iniciaremos de modo abierto con **-skip-grant-tables**, para que el servidor no use privilegios y nos permita tener acceso a todas las bases de datos y a todos los usuarios de las bases de datos, para proteger un poco a nuestro servidor ingresaremos el comando **--skip-networking** para que trabaje únicamente de forma local.

Tomando en cuenta lo anterior ingresaremos:

mysqld_safe - -skip-grant-tables - -skip-networking &

Ahora ingresaremos como usuario root sin que se nos pida ingresar contraseña.

mysql -u root

Ya estando como root ingresaremos a la base de datos MySQL, y asignaremos la nueva contraseña.

use mysql;

update user set authentication_string=password('newpassword') where user='root';

exit

Ya establecida nuestra nueva contraseña realizaremos la configuración de MySQL, para ello ingresaremos al directorio **/user/bin**, ya estando en **/bin** buscaremos el script **mysql_secure_installation** y lo modificaremos.

```
[root@produccion bin]# mysql_secure_installation
Securing the MySQL server deployment.

Enter password for user root:

The existing password for the user account root has expired. Please set a new password.

New password:
```

Antes de permitirnos modificar se nos pedirá ingresar la contraseña de nuestro usuario root de MySQL, y posteriormente nos indicara contraseña expirada, así que ingresaremos una nueva, ya teniéndola, nos dejará avanzar a la modificación de `mysql_secure_installation`, aquí se nos desplegará un listado de preguntas a los que indicaremos Y, y con esto terminaremos la configuración de MySQL.

```
New password:
Re-enter new password:
The 'validate_password' plugin is installed on the server.
The subsequent steps will run with the existing configuration
of the plugin.
Using existing password for root.

Estimated strength of the password: 25
Change the password for root ? ((Press y|Y for Yes, any other key for No) : y

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y
- Dropping test database...
Success.

- Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

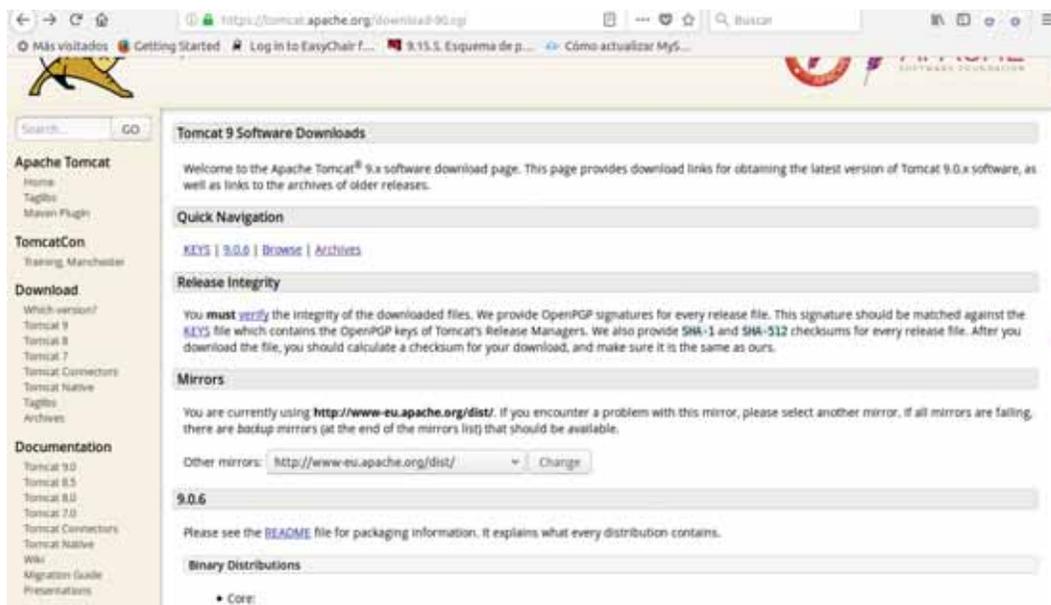
Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
```

Instalación y Configuración del servidor HTTP

Realizaremos la instalación y configuración del Software de código abierto Apache Tomcat, este software es confiable, puede aumentar la eficacia de numerosas aplicaciones web de misión crítica a gran escala.

Iniciamos descargando el software ingresando a la página oficial de Apache Tomcat, seleccionaremos Archivos y descargaremos el paquete:
apache -tomcat-9.0.0.M26-deployer.tar.gz



Verificaremos que ya contamos con el paquete de apache tomcat 9. para ello ingresaremos como súper usuario y posteriormente ingresaremos al directorio que lo contiene, así que ingresaremos a la carpeta de descargas la cual está localizada en Admin, para ellos escribiremos:

cd ~Admin/Downloads

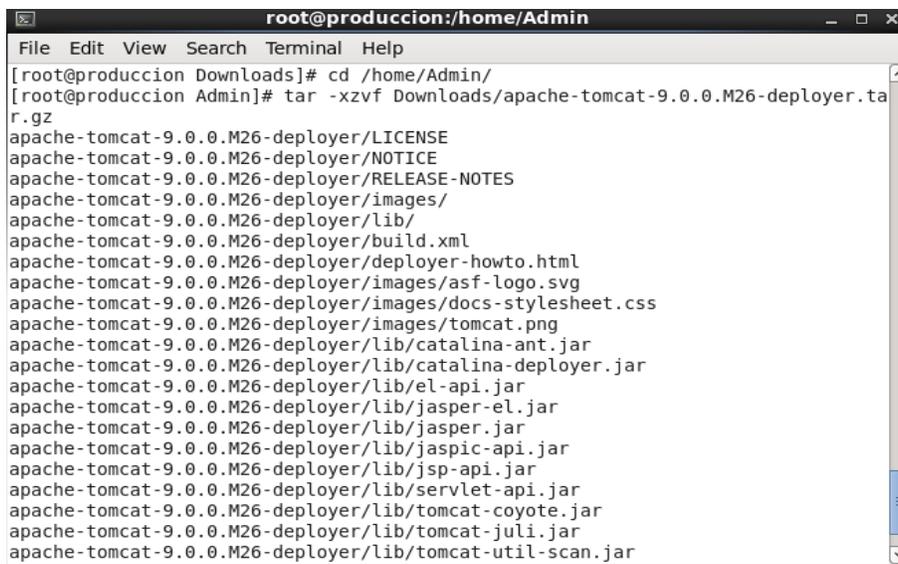
Ya estando en Downloads listaremos los archivos contenidos ingresando el comando **ll**



```
root@produccion:/home/Admin
File Edit View Search Terminal Help
[root@produccion ~]# cd ~Admin/Downloads/
[root@produccion Downloads]# ll
total 172544
-rw-rw-r--. 1 Admin Admin 2525859 Mar 16 23:03 apache-tomcat-9.0.0.M26-deployer.tar.gz
-rw-rw-r--. 1 Admin Admin 174157307 Mar 10 19:37 jdk-8u161-linux-x64.rpm
```

Ya teniendo el paquete de apache tomcat 9, desempacaremos el archivo, para ello nos posicionaremos en /home/Admin y ya estando en Admin utilizaremos el comando **tar -xzf** ingresaremos la carpeta que contiene nuestro paquete y enseguida el nombre del paquete.

Para verificar que se ha desempacado ingresaremos el comando **ll** para listar los archivos contenidos en el directorio.



```
root@produccion:/home/Admin
File Edit View Search Terminal Help
[root@produccion Downloads]# cd /home/Admin/
[root@produccion Admin]# tar -xzf Downloads/apache-tomcat-9.0.0.M26-deployer.tar.gz
apache-tomcat-9.0.0.M26-deployer/LICENSE
apache-tomcat-9.0.0.M26-deployer/NOTICE
apache-tomcat-9.0.0.M26-deployer/RELEASE-NOTES
apache-tomcat-9.0.0.M26-deployer/images/
apache-tomcat-9.0.0.M26-deployer/lib/
apache-tomcat-9.0.0.M26-deployer/build.xml
apache-tomcat-9.0.0.M26-deployer/deployer-howto.html
apache-tomcat-9.0.0.M26-deployer/images/asf-logo.svg
apache-tomcat-9.0.0.M26-deployer/images/docs-stylesheet.css
apache-tomcat-9.0.0.M26-deployer/images/tomcat.png
apache-tomcat-9.0.0.M26-deployer/lib/catalina-ant.jar
apache-tomcat-9.0.0.M26-deployer/lib/catalina-deployer.jar
apache-tomcat-9.0.0.M26-deployer/lib/el-api.jar
apache-tomcat-9.0.0.M26-deployer/lib/jasper-el.jar
apache-tomcat-9.0.0.M26-deployer/lib/jasper.jar
apache-tomcat-9.0.0.M26-deployer/lib/jaspic-api.jar
apache-tomcat-9.0.0.M26-deployer/lib/jsp-api.jar
apache-tomcat-9.0.0.M26-deployer/lib/servlet-api.jar
apache-tomcat-9.0.0.M26-deployer/lib/tomcat-coyote.jar
apache-tomcat-9.0.0.M26-deployer/lib/tomcat-juli.jar
apache-tomcat-9.0.0.M26-deployer/lib/tomcat-util-scan.jar
```

En este punto podemos observar que ya fue desempacado Apache Tomcat y podemos pasar a realizar su instalación.



```
[root@produccion Admin]# ll
total 36
drwxr-xr-x. 4 root root 4096 Mar 16 23:11 apache-tomcat-9.0.0.M26-deployer
drwxr-xr-x. 2 Admin Admin 4096 Mar 10 01:58 Desktop
drwxr-xr-x. 2 Admin Admin 4096 Mar 10 01:58 Documents
drwxr-xr-x. 2 Admin Admin 4096 Mar 16 23:03 Downloads
drwxr-xr-x. 2 Admin Admin 4096 Mar 10 01:58 Music
drwxr-xr-x. 2 Admin Admin 4096 Mar 10 01:58 Pictures
drwxr-xr-x. 2 Admin Admin 4096 Mar 10 01:58 Public
drwxr-xr-x. 2 Admin Admin 4096 Mar 10 01:58 Templates
drwxr-xr-x. 2 Admin Admin 4096 Mar 10 01:58 Videos
[root@produccion Admin]#
```

Ya instalado Apache Tomcat usaremos el comando **chkconfig** para verificar y realizar la actualización de la información de nivel de ejecución para los servicios del sistema, al concluir esto pasaremos a inicializar nuestro servicio con el comando **start**.

```
root@produccion:~  
File Edit View Search Terminal Help  
[root@produccion ~]# chkconfig httpd on  
[root@produccion ~]# service httpd start  
Starting httpd:  
[root@produccion ~]# service httpd status  
httpd (pid 5296) is running...  
[root@produccion ~]# █
```

Ya que hemos inicializado Apache Tomcat, verificaremos cual es el puerto que se ha asignado y estado en que se encuentra, así como la IP que se ha asignado a nuestro localhost.

```
[root@produccion ~]# nmap localhost  
  
Starting Nmap 5.51 ( http://nmap.org ) at 2018-03-19 00:12 CST  
Nmap scan report for localhost (127.0.0.1)  
Host is up (0.000036s latency).  
Other addresses for localhost (not scanned): 127.0.0.1  
Not shown: 993 closed ports  
PORT      STATE SERVICE  
22/tcp    open  ssh  
25/tcp    open  smtp  
80/tcp    open  http  
111/tcp   open  rpcbind  
443/tcp   open  https  
631/tcp   open  ipp  
8443/tcp  open  https-alt
```

Teniendo la información anterior abriremos nuestro navegador e ingresaremos la IP asignada a nuestro localhost, para seguir con las configuraciones de Apache Tomcat. Aquí es necesario contar con el dominio que será asignado a nuestro servidor, de no ser así no se podrá realizar la configuración final.



En este caso no se pudo concluir la configuración por la falta del Dominio, ya que el trámite requiere de un tiempo mayor al tiempo estipulado para la conclusión de la estancia profesional.



7.2. Optimización de sus servicios para agilizar y minimizar el mantenimiento de estos.

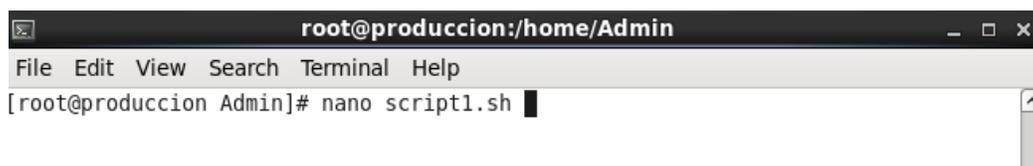
Es este bloque creamos shell scripts para realizar la optimización de los servicios de cada servidor creado, de esta manera agilizamos y minimizamos el mantenimiento de estos.

7.2.1. Creación de Shell Scripts para la evaluación de recursos.

El primer shell script nos servirá para realizar la evaluación del desempeño de los servidores, así como los recursos usados por cada servidor.

Iniciaremos creando el shell script abriendo la terminal e ingresando a la carpeta de nuestro usuario Administrador, ya estando en ella ingresaremos el comando **nano script1.sh**

- **nano** nos indica el editor en el que será creado este script.
- **Script1** será el nombre de nuestro Script
- **.sh** nos indica el tipo de archivo que estamos creando.



Ya estando en el editor de texto iniciaremos nuestro shell script creando un menú.



```
root@produccion:/home/Admin
File Edit View Search Terminal Help
GNU nano 2.0.9 File: script1.sh Modified
#!/bin/bash
clear

while :
do
echo " Seleccione la opción deseada "
echo
echo "1. Visualizar configuración de red"
echo
echo "2. Visualizar uso de disco"
echo
echo "3. Visualizar uso de RAM"
echo
echo "4. Visualizar todas las opciones anteriores"
echo
echo "5. Salir"
echo
echo -n "Indique la opción deseada del [1 al 5]"

```

El comando **echo** nos ayudara a mostrar las opciones de nuestro menú.

Al terminar el menú lo guardaremos y ahora le asignaremos los permisos de lectura, escritura y ejecución, para ellos ingresaremos el siguiente comando **chmod 755** el cual dará permisos al propietario del fichero para leer, escribir y ejecutar el archivo, y a los demás usuarios solo les permitirá leer y ejecutar el archivo.



```
root@produccion:/home/Admin
File Edit View Search Terminal Help
[root@produccion Admin]# nano script1.sh
[root@produccion Admin]# chmod 755 script1.sh
```

Ahora ingresaremos nuevamente a nuestro archivo para seguir construyendo nuestro Script, para ello ingresamos nuevamente el comando **nano** y el nombre del Script e ingresamos el contenido de cada opción, como se muestra en la siguiente imagen.

```
root@produccion:/home/Admin
File Edit View Search Terminal Help
GNU nano 2.0.9 File: script1.sh
ifconfig;
echo
echo "*****"
df;
echo
echo "*****"
echo
free;;
5) echo "seleccionaste salir de script1";

exit 1;;

*) echo "$opc esta es una opción no válida.";

echo "Para continuar, por favor presiona una tecla..";
read foo;;
```

Al terminar de crear nuestro Script guardamos los cambios y lo ejecutamos, para ellos solo ingresaremos ./ y el nombre de nuestro script.

```
root@produccion:/home/Admin
File Edit View Search Terminal Help
[root@produccion Admin]# ./script1.sh
```

Al ejecutar nuestro script se nos muestra lo siguiente:

```
root@produccion:/home/Admin
File Edit View Search Terminal Help
Seleccione la opción deseada
1. Visualizar configuración de red
2. Visualizar uso de disco
3. Visualizar uso de RAM
4. Visualizar todas las opciones anteriores
5. Salir
Indique la opción deseada del [1 al 5]
█
```

Eligiendo la opción 1 visualizamos la configuración de red que tiene nuestro servidor.

```
root@produccion:/home/Admin
File Edit View Search Terminal Help
Indique la opción deseada del [1 al 5]
1
La configuración de red es la siguiente:
*****

eth0      Link encap:Ethernet  HWaddr 08:00:27:D5:25:28
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fed5:2528/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:158700 errors:0 dropped:0 overruns:0 frame:0
          TX packets:48968 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:149439514 (142.5 MiB)  TX bytes:3329180 (3.1 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:4658 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4658 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:251784 (245.8 KiB)  TX bytes:251784 (245.8 KiB)
```

Eligiendo la opción 2 visualizamos el estado del disco duro de nuestro servidor, las particiones con las que cuenta y la cantidad de bits utilizadas y disponibles

```
root@produccion:/home/Admin
File Edit View Search Terminal Help
Indique la opción deseada del [1 al 5]
1
La configuración de red es la siguiente:
*****

eth0      Link encap:Ethernet  HWaddr 08:00:27:D5:25:28
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fed5:2528/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:158700 errors:0 dropped:0 overruns:0 frame:0
          TX packets:48968 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:149439514 (142.5 MiB)  TX bytes:3329180 (3.1 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:4658 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4658 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:251784 (245.8 KiB)  TX bytes:251784 (245.8 KiB)
```

Al elegir la opción 3 se nos muestra la información sobre el uso de la memoria RAM de nuestro servidor.



```
root@produccion:/home/Admin
File Edit View Search Terminal Help

Indique la opción deseada del [1 al 5]
3
El uso de RAM se muestra enseguida:

Mem:          total      used      free      shared  buffers  cached
-/+ buffers/cache: 683596  156360
Swap: 16773116  219956  16553160
Seleccione la opción deseada

1. Visualizar configuración de red
2. Visualizar uso de disco
3. Visualizar uso de RAM
4. Visualizar todas las opciones anteriores
5. salir

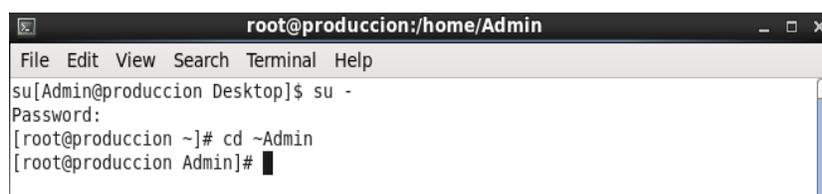
Indique la opción deseada del [1 al 5]
```

Se incluyó en el Script una opción en la que podemos visualizar la información de las 3 anteriores esto con el fin de facilitar la visualización del estatus y configuración de nuestro servidor.

7.2.1. Creación de Shell Scripts para monitorizar los servicios instalados en los servidores virtuales.

Creamos un segundo script el cual nos ayudara a brindar el mantenimiento y verificar el estatus de los servicios de nuestros servidores, así como el reinicio, arranque y apagado de cada uno de ellos.

Al igual que en el script1, este sera creado desde la carpeta del administrador.



```
root@produccion:/home/Admin
File Edit View Search Terminal Help

su[Admin@produccion Desktop]$ su -
Password:
[root@produccion ~]# cd ~Admin
[root@produccion Admin]#
```

Ya estando en la carpeta Admin, iniciamos creando el menú de nuestro script.



```
root@produccion:/home/Admin
File Edit View Search Terminal Help
[root@produccion Admin]# ./Script2.sh

*****
* MENÚ PRINCIPAL *
*****
1) Verificar Status de los servicios
2) Iniciar Servicio
3) Apagar Servicio
4) Reiniciar Servicio
5) Salir

Indica la opcion deseada: █
```

Ya teniendo nuestro menú asignaremos los permisos de ejecución, este script al igual que el script1 solo se podrá leer, escribir y ejecutar por el propietario del fichero, y solo se les permitirá leer y ejecutar el archivo a los demás usuarios.

Al tener asignados los permisos de ejecución crearemos el cuerpo del script, en este caso fue necesario crear submenús y casos.



```
root@produccion:/home/Admin
File Edit View Search Terminal Help
GNU nano 2.0.9 File: Script2.sh

function _submenu1()
{
    echo
    echo "*****"
    echo "1) MYSQL"
    service mysqld status;
    echo ""
    echo "2) APACHE TOMCAT"
    service httpd status;
    echo ""
    echo "3) Open SSH"
    service sshd status;
    echo ""
    echo "4) SERVIDOR DE CORREOS SMTP"
    service sendmail status;
    echo ""
    echo "5) SERVICIO RPCBIND"
    service rpcbind status;
```

Al ejecutar nuestro segundo script ingresando en línea de comandos **./Script2** se nos mostraran las opciones establecidas en nuestro menú , aquí ingresaremos el número de la acción a realizar.

En el caso de querer verificar el status de nuestros servicios elegiremos la opción 1, la cual nos mostrara lo siguiente:

```
Indica la opcion deseada: 1
*****
1) MYSQL
mysqld is stopped
2) APACHE TOMCAT
httpd (pid 2404) is running...
3) Open SSH
openssh-daemon (pid 2288) is running...
4) SERVIDOR DE CORREOS SMTP
sendmail (pid 2341) is running...
sm-client (pid 2351) is running...
5) SERVICIO RPCBIND
rpcbind (pid 1022) is running...
6) Regresar al menú Principal
Indica el número 6 para regresar al menú: █
```

Al elegir la opción 2 para iniciar los servicios se nos enviara a un submenú en el cual podremos elegir el servicio que se desee iniciar:

```
root@produccion:/home/Admin
File Edit View Search Terminal Help
*****
# MENÚ PRINCIPAL #
*****
1) Verificar Status de los servicios
2) Iniciar Servicio
3) Apagar Servicio
4) Reiniciar Servicio
5) Salir
Indica la opcion deseada: 2
1) MYSQL
2) APACHE TOMCAT
3) Open SSH
4) SERVIDOR DE CORREOS SMTP
5) Servicio RPCBIND
6) Regresar al menú Principal
Indica el número de opción a Iniciar: 1
Iniciando servicio...
Starting mysqld:
mysqld (pid 6547) is running...
[ OK ]
```

Al terminar de iniciar el servicio para regresar al menú solo ingresaremos el número 6 y daremos ENTER.

Al elegir la opción 3 en la cual podremos apagar los servicios o la opción 4 la cual reinicia los servicios, al igual que en la opción 2 se nos enviara a un submenú, en el cual también podremos elegir el servicio que se desee apagar o reiniciar:

```
*****
* MENÚ PRINCIPAL *
*****
1) Verificar Status de los servicios
2) Iniciar Servicio
3) Apagar Servicio
4) Reiniciar Servicio
5) Salir
```

Indica la opcion deseada: 3

```
1) MYSQL
2) APACHE TOMCAT
3) Open SSH
4) SERVIDOR DE CORREOS SMTP
5) Servicio RPCBIND
6) TODOS LOS SERVICIOS
7) Regresar al menú Principal
```

Indica el número de opcion a Apagar: █

```
*****
* MENÚ PRINCIPAL *
*****
1) Verificar Status de los servicios
2) Iniciar Servicio
3) Apagar Servicio
4) Reiniciar Servicio
5) Salir
```

Indica la opcion deseada: 4

```
1) MYSQL
2) APACHE TOMCAT
3) Open SSH
4) SERVIDOR DE CORREOS SMTP
5) Servicio RPCBIND
6) TODOS LOS SERVICIOS
7) Regresar al menú Principal
```

Indica el número de opcion a Reiniciar: █

8. Conclusiones

Toda empresa requiere de servidores, los cuales soporten las peticiones y almacenamiento de datos de los usuarios, páginas web, herramientas de desarrollo etc.

Las empresas siempre han implementado servidores físicos los cuales requieren de espacio, y no muchas empresas cuentan con ello, así como de un mantenimiento constante a cada servidor, es por ello que se plantea como una mejor opción la virtualización de servidores, ya que el mantenimiento de los servidores físicos se minimiza a uno solo, el cual puede contener más de 2 servidores virtuales con diferentes Sistemas Operativos.

La virtualización de servidores brinda soluciones a las empresas de hoy en día que desean minimizar costos y espacios en el área de las TI, y mejorar la seguridad de la información que contienen.

La virtualización de servidores no solo tiene ventajas, también encontramos algunas desventajas durante el proceso del proyecto realizado, una de estas fue el rendimiento de los servidores este debido a que siempre se tendrá un rendimiento menor al del servidor nativo.

La implementación de shell scripts en los servidores es bastante útil cuando se tiene a poco personal, estos minimizan el tiempo al verificar el status de los servicios de esta manera se pueden localizar problemas en los servicios, así como ver en qué estado se encuentra la memoria o los directorios de cada servidor.

9. Apéndice

9.1. Código Shell Script para evaluar recursos.

```
#!/bin/bash
clear

while :
do

    echo " Seleccione la opción deseada "
    echo
    echo "1. Visualizar configuración de red"
    echo
    echo "2. Visualizar uso de disco"
    echo
    echo "3. Visualizar uso de RAM"
    echo
    echo "4. Visualizar todas las opciones anteriores"
    echo
    echo "5. Salir"
    echo
    echo -n "Indique la opción deseada del [1 al 5]"
    echo

read opcion
case $opcion in

    1) echo "La configuración de red es la siguiente:";
    echo
    "*****"
    echo
    ifconfig;;

    2) echo "El uso de disco se muestra enseguida:";
    df;;

    3) echo "El uso de RAM se muestra enseguida:";
    echo
    echo
    free;;

    4) echo "En seguida se muestra la información de las opciones anteriores:";
    echo
    echo
```

```
ifconfig;
echo
echo
"*****"
*****"
echo
df;
echo
echo
"*****"
*****"
echo

free;
5) echo "seleccionaste salir de script1";

exit 1;;

*) echo "$opc esta es una opción no válida.";
echo "Para continuar, por favor presiona una tecla...";
read foo;
esac
done
```

9.2.Codigo Shell Script para monitorizar los servicios instalados en los servidores virtuales.

```
#!/bin/bash

#Muestra el menu general
function _menuPrincipal()
{
    echo ""
    echo "*****"
    echo "* MENÚ PRINCIPAL *"
    echo "*****"
    echo "1) Verificar Status de los servicios"
    echo "2) Iniciar Servicio"
    echo "3) Apagar Servicio"
    echo "4) Reiniciar Servicio"
    echo "5) Salir"
    echo
    echo -n "Indica la opcion deseada: "
}
function _submenu1()
{
    echo
    echo "*****"
    echo "1) MYSQL"
    service mysqld status;
    echo ""
    echo "2) APACHE TOMCAT"
    service httpd status;
    echo ""
    echo "3) Open SSH"
    service sshd status;
    echo ""
    echo "4) SERVIDOR DE CORREOS SMTP"
    service sendmail status;
    echo ""
    echo "5) SERVICIO RPCBIND"
    service rpcbind status;
    echo "6) Regresar al menú Principal"
    echo
    echo -n "Indica el número 6 para regresar al menú: "
}
function _submenu2()
{
    echo
    echo "1) MYSQL"
    echo "2) APACHE TOMCAT"
    echo "3) Open SSH"
    echo "4) SERVIDOR DE CORREOS SMTP"
```

```

    echo "5) Servicio RPCBIND"
    echo "6) TODOS LOS SERVICIOS"
    echo "7) Regresar al menú Principal"
    echo
    echo -n "Indica el número de opción a Iniciar: "
}
function _submenu3()
{
    echo
    echo "1) MYSQL"
    echo "2) APACHE TOMCAT"
    echo "3) Open SSH"
    echo "4) SERVIDOR DE CORREOS SMTP"
    echo "5) Servicio RPCBIND"
    echo "6) TODOS LOS SERVICIOS"
    echo "7) Regresar al menú Principal"
    echo
    echo -n "Indica el número de opcion a Apagar: "
}
function _submenu4()
{
    echo
    echo "1) MYSQL"
    echo "2) APACHE TOMCAT"
    echo "3) Open SSH"
    echo "4) SERVIDOR DE CORREOS SMTP"
    echo "5) Servicio RPCBIND"
    echo "6) TODOS LOS SERVICIOS"
    echo "7) Regresar al menú Principal"
    echo
    echo -n "Indica el número de opcion a Reiniciar: "
}
opc=0
until [ $opc -eq 5 ]
do
    case $opc in
        1)
            opc1=0
            until [ $opc1 -eq 6 ]
            do
                case $opc1 in
                    1)
                        echo "menu 1 submenu 1"
                        ;;
                    *)
                        _submenu1
                        ;;
                esac
                read opc1
            done
        *)
            _submenu3
        *)
            _submenu4
    esac
done

```

```

done
  _menuPrincipal
  ;;
2)
opc2=0
until [ $opc2 -eq 7 ]
do
  case $opc2 in
    1)
      echo "Iniciando servicio..."
      service mysqld start
      service mysqld status ;;
    2)
      echo "Iniciando servicio..."
      service httpd start
      service httpd status;;
    3)
      echo "Iniciando servicio..."
      service sshd start
      service sshd status;;
    4)
      echo "Iniciando servicio..."
      service sendmail start
      service sendmail status;;
    5)
      echo "Iniciando servicio..."
      service rpcbind start
      service rpcbind status;;
    6)
      echo "Iniciando servicios..."
      service mysqld start
      service mysqld status;
      service httpd start
      service httpd status;
      service sshd start
      service sshd status;
      service sendmail start
      service sendmail status;
      service rpcbind start
      service rpcbind status;;
    *)
      _submenu2
      ;;
  esac
  read opc2
done
  _menuPrincipal
  ;;

```

3)

```
opc3=0
until [ $opc3 -eq 7 ]
do
  case $opc3 in
    1)
      echo "Apagando servicio..."
      service mysqld stop
      service mysqld status ;;
    2)
      echo "Apagando servicio..."
      service httpd stop
      service httpd status;;
    3)
      echo "Apagando servicio..."
      service sshd stop
      service sshd status;;
    4)
      echo "Apagando servicio..."
      service sendmail stop
      service sendmail status;;
    5)
      echo "Apagando servicio..."
      service rpcbind stop
      service rpcbind status;;
    6)
      echo "Apagando servicios..."
      service mysqld stop
      service mysqld status;
      service httpd stop
      service httpd status;
      service sshd stop
      service sshd status;
      service sendmail stop
      service sendmail status;
      service rpcbind stop
      service rpcbind status;;

    *)
      _submenu3
      ;;
  esac
  read opc3
done
_menuPrincipal
;;
```

4)

```
opc4=0
until [ $opc4 -eq 7 ]
```

```

do
    case $opc4 in
        1)
            echo "Reiniciando servicio..."
            service mysqld restart
            service mysqld status ;;
        2)
            echo "Reiniciando servicio..."
            service httpd restart
            service httpd status;;
        3)
            echo "Reiniciando servicio..."
            service sshd restart
            service sshd status;;
        4)
            echo "Reiniciando servicio..."
            service sendmail restart
            service sendmail status;;
            5)
            echo "Reiniciando servicio..."
            service rpcbind restart
            service rpcbind status;;
        6)
            echo "Apagando servicios..."
            service mysqld stop
            service mysqld status;
            service httpd stop
            service httpd status;
            service sshd stop
            service sshd status;
            service sendmail stop
            service sendmail status;
            service rpcbind stop
            service rpcbind status;;

        *)
            _submenu4
            ;;
    esac
    read opc4
done
_menuPrincipal
;;
*)
_menuPrincipal
;;
esac
read opc
done

```

10. Bibliografía

- [1]2017. [Online]. Available: <http://www.csg.gob.mx/consejo/mision.html>. [Accessed: 15- Dec- 2017].
- [2]"Historia de la Virtualización", *Virtualizacion*, 2018. [Online]. Available: <http://www.virtualizacion.com/virtualizacion/>. [Accessed: 10- Aug- 2018].
- [3]"CentOS Project", Centos.org, 2017. [Online]. Available: <https://www.centos.org/>. [Accessed: 15- Jul- 2017].
- [4]"INFOFIA N°49 - USMP", *Usm.edu.pe*, 2018. [Online]. Available: <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/too.html>. [Accessed: 2- Aug- 2018].
- [5]E. Nemeth, G. Snyder and T. Hein, *La biblia de administración de sistemas Linux*. Madrid: Anaya Multimedia, 2008, pp. 124-126.
- [6]"9.15.5. Esquema de particionamiento recomendado - Red Hat Customer Portal", *Access.redhat.com*, 2018. [Online]. Available: https://access.redhat.com/documentation/es-es/red_hat_enterprise_linux/6/html/installation_guide/s2-diskpartrecommend-x86. [Accessed: 10- Aug- 2018].
- [7]"Java SE Development Kit 8 - Downloads", Oracle.com, 2017. [Online]. <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html?ssSourceSiteId=otnes>. [Accessed: 15- Dec- 2017].
- [8]"MySQL", Mysql.com, 2017. [Online]. Available: <https://www.mysql.com/>. [Accessed: 15- Jul- 2017].
- [9]A. Project, "Apache Tomcat® - Welcome!", Tomcat.apache.org, 2017. [Online]. Available: <http://tomcat.apache.org/>. [Accessed: 17- Jul- 2017].
- [10]"Oracle VM VirtualBox", Virtualbox.org, 2017. [Online]. Available: <https://www.virtualbox.org/>. [Accessed: 17- Jul- 2017].