

Unidad 6



Alta disponibilidad

Seguridad y alta disponibilidad





Índice

6.1. Introducción

- 6.1.1. Clústeres de alta disponibilidad
- 6.1.2. Métricas de disponibilidad

6.2. Replicación de servidores

- 6.2.1. Sincronización de directorios
- 6.2.2. Almacenamiento distribuido DRBD
- 6.2.3. Replicación de las bases de datos

6.3. Balanceo de carga

- 6.3.1. Balanceo por DNS
- 6.3.2. Balanceo de conexiones con iptables
- 6.3.3. Software específico de balanceo
- 6.3.4. Balanceo con el servidor web

6.4. Virtualización

- 6.4.1. Ventajas de la virtualización
- 6.4.2. Soluciones de virtualización
- 6.4.3. Configuración de máquinas virtuales
- 6.4.4. Virtualización mediante contenedores
- 6.4.5. Virtualización de redes



Introducción

Cómo diseñar y configurar un sistema informático o sus componentes para garantizar Máxima operatividad posible a través de una alta tolerancia a fallas llamada "Alta Disponibilidad" (High Availability o HA).

Las dos técnicas principales utilizadas en la alta disponibilidad son la redundancia y el equilibrio de carga.

La redundancia puede ser negativa (sistema diseñado teniendo en cuenta la redundancia) o activo (el sistema responde a los errores informando a los servicios). La redundancia en un clúster se logra recreando sus nodos. En función del dato o servicio garantizado, la medición incluirá la sincronización de directorios, el almacenamiento distribuido DRBD, clonación de bases de datos, realización de instantáneas, etc.

El equilibrio de carga se puede realizar mediante el servicio y el control de DNS

Conéctese a través del firewall o mejor usando un programa específico de balanceo, destacando este efecto HAProxy. Existen diferentes algoritmos de balanceo, el más común de los cuales es la inflexión.

Cuando los servidores a equilibrar son servidores web, se pueden utilizar extensiones para que los servidores más populares (Apache, Nginx, IIS) asuman esta tarea sin necesidad de un programa de ecuaciones específico.

El principal problema con los web balancers es la persistencia de las sesiones de los usuarios. Las técnicas para resolver este problema se basan en sesiones persistentes: el mismo cliente siempre debe ser atendido por el mismo servidor web en el clúster.

Al finalizar esta unidad

- + Estudiaremos el concepto de alta disponibilidad
- + Estudiaremos las formas de replicar datos y sistemas
- + Conocer la tecnología de virtualización de equipos
- + Aprenderemos las distintas formas de hacer un balanceo de carga, así como el software llevado a cabo
- + Aprenderemos los distintos conceptos acerca de la alta disponibilidad y de la alta adaptabilidad



6.1.

Introducción

En él se introdujeron conceptos relacionados con la alta disponibilidad. es decir, cómo se diseñan y configuran un sistema informático o sus componentes, esto es para asegurar la máxima operabilidad a través de una alta tolerancia a fallas. Intentamos que los momentos de indisponibilidad del sistema sean los mínimos o nulos posibles, con una duración mínima y sobre todo con una planificación responsable de las tareas de mantenimiento.

Los principales motivos que pueden causar problemas en un sistema informático están relacionados cuando un periodo está inactivo o downtime, son:

- > Fallos en el hardware
- > Fallos en la red
- > Actualizaciones del software o firmware
- > Corrupción de datos
- > Copias de seguridad
- > Ataques externos
- > Ataques internos

Podemos destacar dos técnicas para alcanzar una elevada disponibilidad que serían el balanceo de carga y la redundancia.

En el caso del balanceo de carga, lo que pretende es dividir el trabajo entre todos los componentes del sistema de tal forma que estén equilibrados, con esto se producirá, además, una mejora de la calidad.

En relación con la redundancia, encontramos:

- > **Redundancia pasiva:** se logra incorporando al sistema una elevada cantidad de la capacidad.
- > **Redundancia activa:** el sistema tiene un sistema para percibir determinados fallos y ser capaz de adaptarse a éstos para así poder seguir realizando su acción de forma normal.

6.1.1. Clústeres de alta disponibilidad

Un grupo de alta disponibilidad es un grupo de dos o más dispositivos que se caracterizan por mantener una cadena de servicios compartidos y monitorearse constantemente entre sí.

El clúster de alta disponibilidad no debe confundirse con el clúster de alto rendimiento. La segunda es que la configuración de la computadora está diseñada para proporcionar mucha más potencia de cómputo que la configuración proporcionada por las computadoras individuales (ver por ejemplo el sistema Beowulf Cluster), mientras que el segundo tipo de clúster está diseñado para garantizar el funcionamiento ininterrumpido de algunas aplicaciones.



Estos tipos de bloques se usan comúnmente para balanceadores de carga, servicios de respaldo y conmutación por error. Para configurarse correctamente, todos los servidores deben tener acceso a la misma memoria compartida, de modo que, si uno de ellos falla, una máquina virtual de otro host puede ejecutarse y realizar sus tareas sin interrupción.

Algunos conceptos importantes relacionados con los clústeres de alta disponibilidad son:

- + **Failover:** La conmutación por error o failover es un modo operativo alternativo en el que un componente secundario asume la funcionalidad de un componente del sistema cuando el componente principal deja de estar disponible debido a una falla o bloqueo programado. La conmutación por error es una parte integral de los sistemas de misión crítica.
- + **Switchover:** en caso de que el failover requiera de la actividad humana para que los servicios funcionen, se conocerá como switchover.
- + **Takeover:** mecanismo por el que un nodo adquiere los recursos de un nodo caído para así poder poner en funcionamiento un servicio que ha sido paralizado.
- + **Giveback:** cuando un nodo caído vuelve a incorporarse y vuelve a ser parte del clúster, recuperando de esa forma su acción y el control.
- + **Failback:** se refiere a las acciones requeridas para invertir el clúster a la ubicación inicial una vez que se han resuelto los problemas.

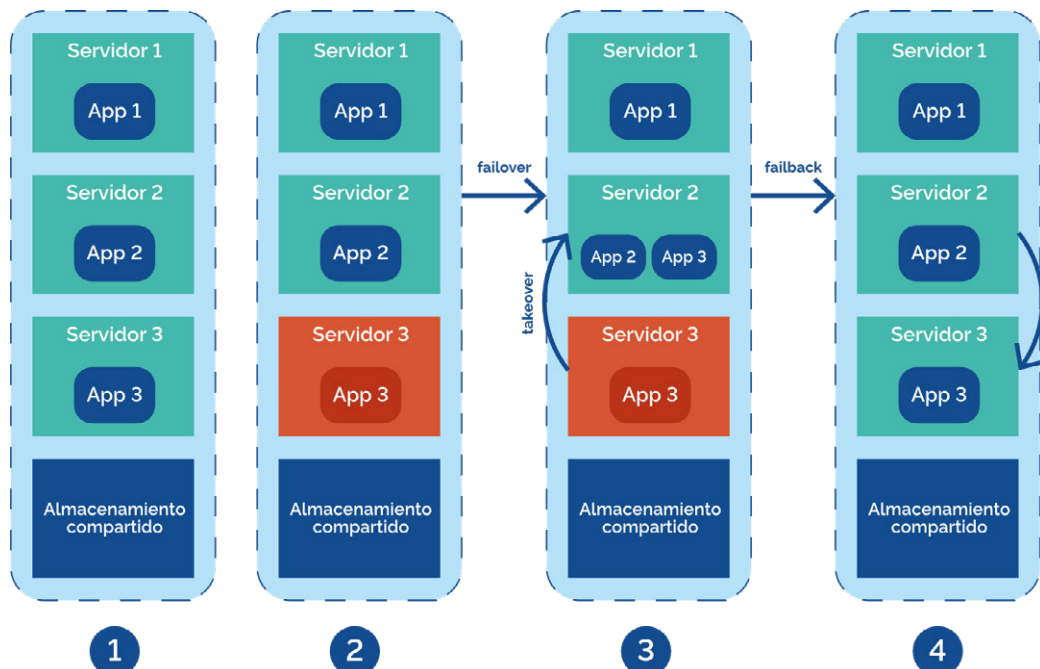


Imagen 1. Explicación gráfica de algunos conceptos relacionados con los clústeres de alta disponibilidad



6.1.2. Métricas de disponibilidad

El objetivo de definir una estructura consistente de métricas para rastrear es poder monitorear, administrar, mejorar e informar regularmente sobre todos nuestros Servicios.

Las medidas deben diseñarse para garantizar que la infraestructura, las redes y las aplicaciones estén configuradas y funcionen correctamente. Para las empresas con infraestructura con máquinas virtuales, contenedores o en la nube, deben aplicar las mismas métricas a estos sistemas.

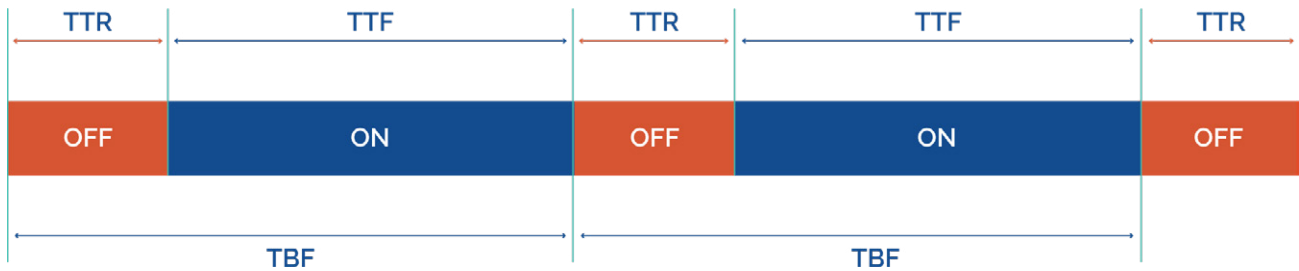


Imagen 2. Representación gráfica de los periodos en que un sistema está o no disponible

Siendo en la imagen anterior:

- + TTR = Tiempo de recuperación
- + TTF = Tiempo de fallo
- + TBF = Tiempo entre fallos
- + MTTR = Tiempo medio de recuperación
- + MTTF = Tiempo medio hasta que se produce un fallo
- + MTBF = Tiempo medio entre fallos



6.2.

Replicación de servidores

La redundancia en los servidores de un clúster se logra mediante la redundancia de sus nodos, proceso por el cual todos actúan como espejos. El resultado de la replicación debería ser que los clientes reciban la misma información o los mismos servicios, independientemente del servidor responsable de su servicio.

Con la llegada de la virtualización, la replicación del servidor se simplifica enormemente, la replicación basada en instantáneas se puede realizar para un servidor en ejecución. Sin embargo, fuera de los grupos de hipervisores, la sincronización suele realizarse exclusivamente en los grupos de hipervisores los datos que desea copiar.

6.2.1. Sincronización de directorios

La sincronización de directorios le permite continuar administrando usuarios y grupos en su implementación local y sincronizar adiciones, eliminaciones y cambios en la nube. Sin embargo, la configuración es un poco complicada y, a veces, puede ser difícil identificar el origen de los problemas. Tenemos los recursos para ayudarlo a identificar y resolver problemas potenciales.

Podemos destacar las características mostradas a continuación:

- > Solo transmiten los bloques que han sido verdaderamente modificados.
- > Podrá comprimir y cifrar la transmisión.
- > Respeta la propiedad, los permisos y los atributos tanto de los ficheros como de los directorios.

Se va a utilizar la herramienta rsync que es *open source* y está incorporada en la mayor parte de las distribuciones de Linux. En la web de esta herramienta podemos ver documentos con gran detalle y ejemplos para el uso de la misma.

Su funcionamiento depende de la arquitectura del servidor cliente. El sistema cliente es el que comienza la sincronización con el sistema servidor usando uno de los dos métodos soportados o bien mediante una conexión de red directa o bien, mediante una sesión SSH.

En el caso del servidor, es necesario tener un demonio rsync que escuche las solicitudes en un puerto, mientras que, en el segundo caso, se necesita un servidor SSH para permitir la ejecución remota de comandos para el usuario que usa rsync.

Una vez que el contacto ha sido creado y aprobado la última versión del protocolo soportad para ambos, se intercambian listas de archivos y directorios y se crean diferentes procesos para realizar la tarea de sincronización, a saber:



- > **Un proceso generador**, que va a estudiar los cambios que se producen en los ficheros.
- > **Un proceso emisor**, que captan la información del generador y la envía a través de una red.
- > **Un proceso receptor**, que percibe los datos del emisor y en su sistema de ficheros los replicará.

Aunque la sincronización siempre la inicia el cliente, esto no significa que el flujo de datos deba ir al servidor: la sincronización puede hacerse en ambos sentidos siempre y cuando el resultado obtenido finalmente presente un directorio igual al directorio de origen.

Es fundamental tener en cuenta que rsync se puede usar para copias de seguridad dentro del mismo equipo.

Además, sus creadores se refieren a ella como "una herramienta rápida y flexible para realizar copiar de un archivo remoto (y local)". Por esta razón, rsync es uno de los comandos más utilizados para crear copias de seguridad.

Las desventajas aparecen del rsync, especialmente cuando hay muchos archivos grandes. Una de las partes más lentas de rsync es que, al sincronizar, los archivos en el directorio local se comparan con los archivos en el servidor, por lo que solo actualiza los diferentes archivos allí. Los archivos similares se descartan directamente. Si hay muchos archivos para sincronizar, esta comparación es demasiado lenta.

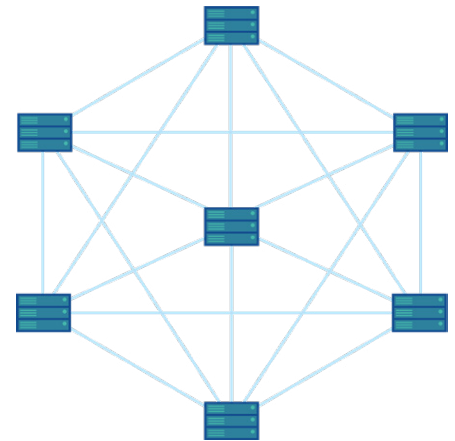


Imagen 3 Con la sincronización P2P todos los nodos de un clúster son, al mismo tiempo, clientes y servidores

6.2.2. Almacenamiento distribuido DRBD

DRBD (abreviatura de Distributed and Redundant Block Devices) es una solución de almacenamiento flexible y de replicación para Linux. Refleja el contenido de dispositivos de bloque como unidades de disco duro, particiones, volúmenes lógicos, etc. entre servidores. Es una copia de los datos en dos dispositivos de almacenamiento, de modo que, si uno falla, se pueden utilizar los datos del otro.

En un primer momento el DRBD se utilizaba en conjuntos de ordenadores de elevada disponibilidad, aunque a partir de la versión número 9 se puede utilizar para incluir soluciones en el almacén y gestión de la nube.

Puede considerarlo como una configuración de red RAID 1 con discos duplicados en el servidor. Sin embargo, funciona de manera muy diferente a yb RAID y también a RAID de red.

Podemos encontrar una serie de diferencias entre DRBD y RAID 1:

- > **Con RAID 1** encontramos una aplicación entrando a dos unidades de manera transparente ya que la aplicación sólo va a detectar un almacenamiento.

- > **Con DRBD** podemos encontrar dos aplicaciones y cada una de ellas va a actuar en un nodo diferente y cada una de las cuales accede a la unidad de almacenamiento concreto.
- > **Con RAID 1**, si la aplicación no funciona correctamente la información de almacen continuarán en el mismo sitio.
- > **Con DRBD**, si la aplicación del nodo no funciona correctamente, a continuación, funcionará otro.

El controlador DRBD ha sido parte del kernel de Linux desde 2009 y puede ejecutarse en el sincrónico o asincrónico.

- > En el modo sincrónico, la aplicación no recibe un mensaje de registro exitoso hasta que esto sea hecho por todos los nodos copiados.
- > En modo de notificación asincrónico, las apuestas se recibieron inmediatamente después de que los datos se registraron localmente, pero aún no se haya producido la propagación a los demás nodos.



6.2.3. Replicación de las bases de datos

Casi ninguna computadora o aplicación web puede funcionar sin un sistema de base de datos, por lo que debe estar familiarizado con los conceptos básicos relacionados con este tema. El sistema básico de gestión de bases de datos de cualquier base de datos es tan importante como el propio conjunto de datos, ya que no se puede gestionar sin él.

La propia base de datos y un sistema de gestión de base de datos o DBMS (también conocido como DBMS, del inglés database management system) constituyen lo que se conoce como sistema de base de datos (en ocasiones, el término base de datos se utiliza simplemente para denominarla).

En general, un DBMS es un software que sigue el modelo de un sistema de base de datos, y por lo tanto es esencial para su configuración, administración y uso. Solo cuando el DBMS está instalado y configurado, los usuarios pueden ingresar y consultar datos.

Los permisos de lectura y escritura, así como las funciones administrativas generales, se establecen a través de interfaces específicas de la aplicación y lenguajes de definición de datos. El lenguaje más conocido es SQL.

Un DBMS consta de varios componentes, todos los cuales contribuyen al funcionamiento normal del programa. Sus componentes básicos incluyen tres componentes: un diccionario de datos, un lenguaje de definición de datos y un lenguaje de procesamiento de datos.

Generalmente, se usan versiones concretas antes de que se lleve a cabo una replicación clásica que es lo habitual y favorecido por dos motivos:

- > Los sistemas están elaborados mediante la replicación de datos en mente.
- > Normalmente suelen incorporar avances de distintos tipos, como por ejemplo el procesamiento de consultas en paralelo.



6.3.

Balaceo de carga

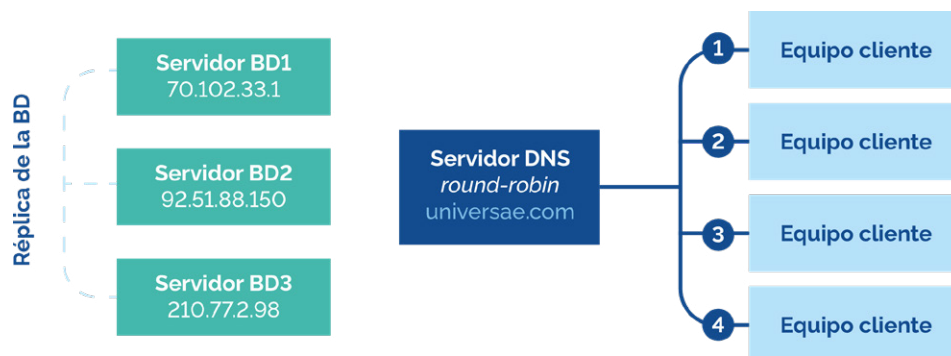
La carga de palabras puede referirse a varias cosas, según el sistema del que estemos hablando. En comparación con los datos, por ejemplo, ya sea por cable o inalámbrico. De forma predeterminada, el equilibrio de carga implica distribuir el tráfico entre los enlaces disponibles y así hacer un mejor uso del ancho de banda disponible.

La disponibilidad está garantizada siempre que el grado de falla de un enlace solo significa que el tráfico se entregará a través del resto de los enlaces. Si se restaura el eslabón perdido o incluso si se aumenta la capacidad con eslabones adicionales, se va a producir una redistribución de la carga de manera uniforme.

6.3.1. Balanceo por DNS

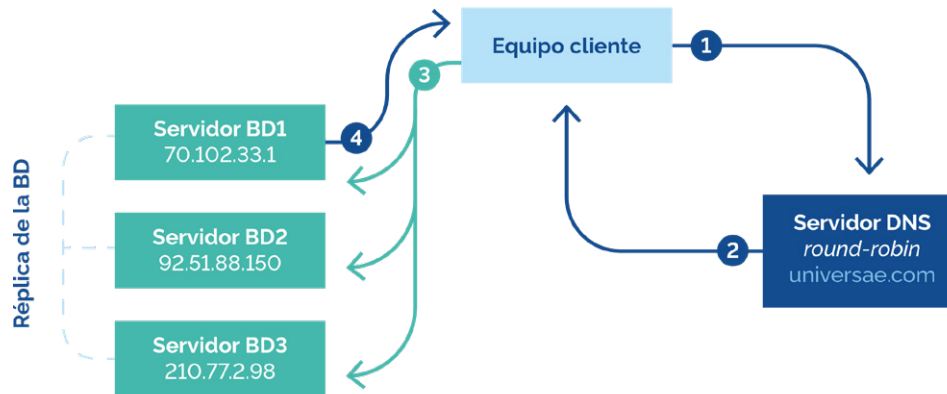
Por ejemplo, el equilibrio de carga de DNS es cuando tiene varios servidores web que brindan el mismo contenido (para redundancia) y desea que la carga se distribuya de manera uniforme entre esos servidores y puede hacerlo mediante el uso de un servidor DNS para proporcionar diferentes direcciones IP a diferente clientela. De esta forma reparte la carga entre sus diferentes servidores.

Para hacer esto con DNS, una técnica (y el único método que admitimos) es especificar varios registros A (direcciones IP) con el mismo nombre, cada uno con una dirección IP diferente. Luego, cuando realice consultas para su dominio, la lista completa de registros se devolverá en el orden modificado cada vez. La mayoría de los clientes prueban la primera dirección IP y pasan a la siguiente si la primera dirección IP no está disponible.



- 1 ¿Cuál es la IP de `bd.universae.com`? → 70.102.33.1
- 2 ¿Cuál es la IP de `bd.universae.com`? → 92.51.88.150
- 3 ¿Cuál es la IP de `bd.universae.com`? → 210.77.2.98
- 4 ¿Cuál es la IP de `bd.universae.com`? → 70.102.33.1

Imagen 4. Balanceo de carga mediante DNS round-robin



- 1 ¿Cuál es la IP de `bd.universae.com`?
- 2 *Response:* Pregunta a los servidores DNS delegados `bd1`, `bd2` y `bd3`.
- 3 *Query:* ¿cuál es la IP de `bd.universae.com`?
- 4 *Response:* (el primer servidor que contesta) → `70.102.33.1`

Imagen 5. Balanceo de carga mediante delegación DNS

El simple funcionamiento del sistema DNS tendrá como resultado que los clientes puedan llevar a cabo el balanceo.

6.3.2. Balanceo de conexiones con iptables

El monitoreo de conexión es una de las características avanzadas de iptables o firewalls con funcionalidad avanzada, ya que viene bajo el nombre de monitoreo de conexión.

Lo que proporciona un simple cortafuegos es una serie de reglas que, dependiendo de las propiedades de las cabeceras de las diferentes capas del modelo TCP/IP, permiten el paso o no del tráfico.

La monitorización de la conexión hace que este control sea más completo y nos permite controlar o limitar una amplia gama de ataques o abusos que podemos recibir de paquetes modificados artificialmente, pero con un simple firewall, las computadoras ya están atrasadas porque el firewall no podrá apagarse ese tráfico.

En el caso de iptables, esto se hace usando un programa llamado Conntrack, que se usa para rastrear conexiones. Este programa se puede usar de forma independiente ya que el paquete Conntrack está presente en la mayoría de las distribuciones y se puede usar en la línea de comandos. También se puede utilizar como módulo de iptables, es decir, una extensión de la base de iptables, ya que permite ampliar su funcionalidad a través de módulos.

Conntrack se puede usar como un todo, con una amplia gama de configuraciones, aunque en este caso solo le mostraremos un subconjunto de las funciones de Conntrack, es decir, sus funciones de gestión de estado y referencia. Los paquetes, ejecutados a través de un módulo de iptables llamado state, son un subconjunto de la funcionalidad proporcionada por Conntrack.

```
iptables -t nat -A PREROUTING -p tcp -i enp0s3
--dport 80 -m state --state NEW -m statistic --mode
nth --every 3 --packet 0 -j DNAT --to-destination
10.0.0.25:80
```

```
iptables -t nat -A PREROUTING -p tcp -i enp0s3
--dport 80 -m state --state NEW -m statistic --mode
nth --every 3 --packet 1 -j DNAT --to-destination
10.0.0.50:80
```

```
iptables -t nat -A PREROUTING -p tcp -i enp0s3
--dport 80 -m state --state NEW -m statistic --mode
nth --every 3 --packet 2 -j DNAT --to-destination
10.0.0.51:80
```

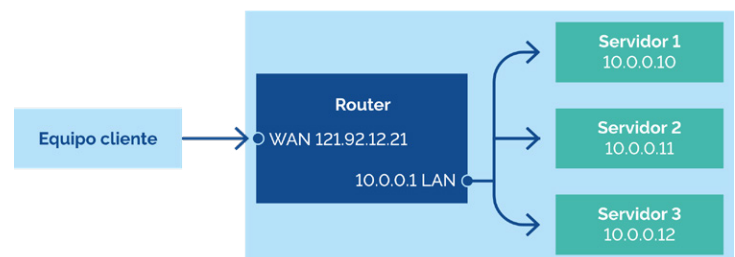


Imagen 6. Clúster de tres servidores



6.3.3. Software específico de balanceo

Linux Virtual Server

Linux Virtual Server (LVS) es una solución de gestión de equilibrio de carga en sistemas Linux. Este es un proyecto de código abierto iniciado por Wensong Zhang en mayo de 1998. El objetivo es desarrollar un servidor Linux de alto rendimiento que brinde buena escalabilidad, confiabilidad y robustez mediante la tecnología de clúster.

Actualmente, el trabajo principal del proyecto LVS es el desarrollo de balanceadores de carga de software IP (IPVS), balanceadores de carga de software a nivel de aplicación y componentes para la gestión de clústeres.

- > **IPVS o balanceo por IP:** es un sistema de IP avanzado de equilibrado de carga por software que está implementado en el núcleo de Linux.
- > **KTCPVS o balanceo a nivel de aplicación:** implementa el equilibrado de carga en la aplicación del núcleo de Linux. A día de hoy está en vías de desarrollo.

Los usuarios pueden utilizar las soluciones LVS para crear un sistema altamente escalable que garantice una alta disponibilidad de los servicios de red, como servicios web, de correo electrónico o de VoIP.

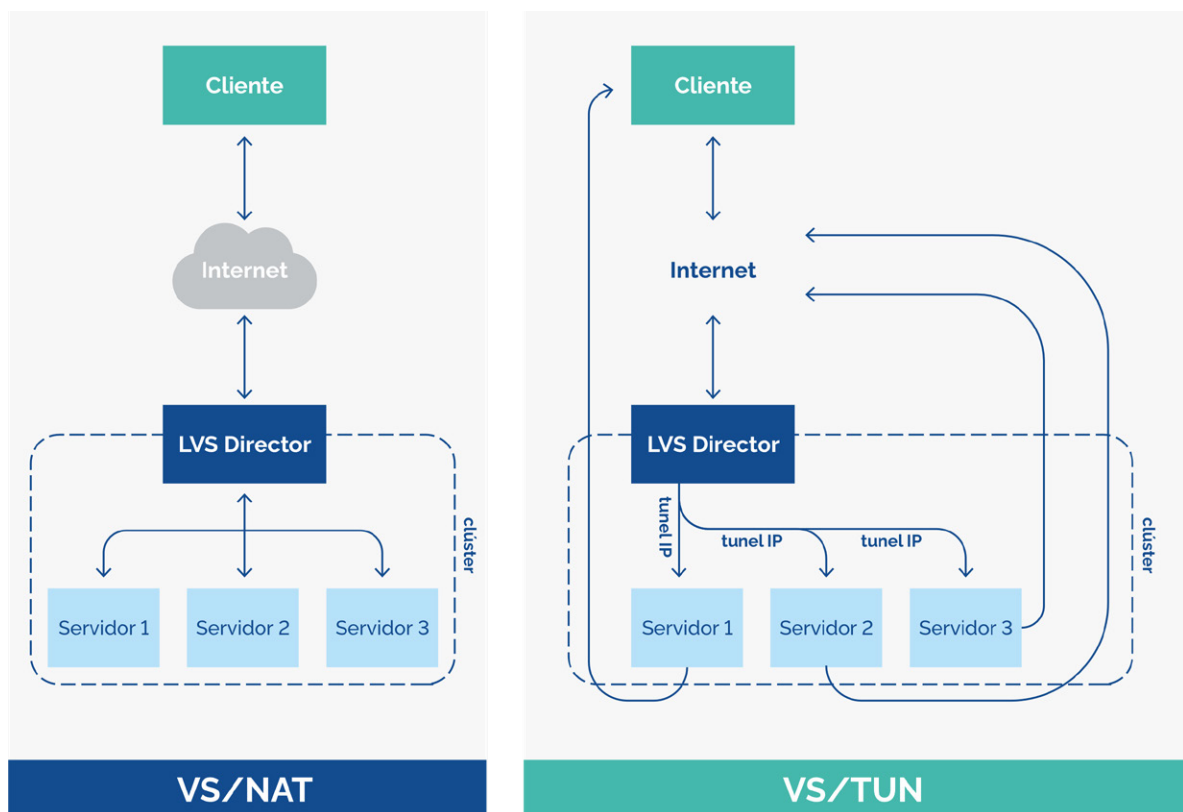


Imagen 7. Los dos modos de funcionamiento de LVS. Las flechas indican el sentido de la comunicación



HAProxy

HAProxy (abreviatura de High Available Proxy) es una popular solución de proxy y equilibrador de carga TCP/HTTP de código abierto para aplicaciones basadas en TCP y HTTP. La forma en que funciona HAProxy hace que la integración en las arquitecturas existentes sea simple y sin riesgos, y brinda suficiente seguridad para no exponer servidores web frágiles a la red. Es rápido, confiable y especialmente adecuado para sitios web de alto tráfico y es utilizado por muchos de los sitios web más visitados del mundo.

Al recopilar una amplia gama de métricas que brindan una visión detallada del rendimiento de HAProxy, HAProxy Monitor hace que el monitoreo de HAProxy de Application Manager sea simple y eficiente.

Podemos destacar cuatro algoritmos:

- > Round-robin
- > Leastconn
- > Source
- > URI

6.3.4. Balanceo con el servidor web

Apache

Apache es un servidor web gratuito, multiplataforma y de código abierto. Este servidor web es uno de los más utilizados en el mundo, con el 43% de los sitios web trabajando actualmente con él.

Este servidor web está desarrollado por Apache Software Foundation y ha estado en funcionamiento desde 1995.

El nombre Apache hace referencia a una tribu de nativos americanos conocida por su gran resistencia y estrategia bélica. Se ha vuelto muy popular entre los programadores debido a su modularidad y actualizaciones constantes de la comunidad.

El servidor Apache se puede encontrar en la mayoría de los servidores del mundo y funciona perfectamente con paneles de control como WePanel, Plesk, VestaCP, etc.



Nginx

Nginx, comúnmente conocido como "engine-ex", es un servidor web de código abierto que, desde su éxito inicial como servidor web, ahora también se usa como proxy inverso, caché HTTP y equilibrador de carga.

Nginx sostiene los algoritmos de balanceo que se expresan a continuación;

- > **Round-robin:** se denomina también algoritmo de los turnos circulares.
- > **Least connections:** sería el equivalente a `mod_lbmethod_bybusiness` en Apache.
- > **Least time:** se parece mucho a Least connections, sin embargo, éste premia al servidor con una respuesta más eficiente.
- > **Generic hash:** para cada una de las solicitudes se va a determinar un valor hash fundamentado en parámetros del sistema.
- > **Random:** selecciona un servidor al azar, aunque considerando el peso de éstos.
- > **IP hash:** es un algoritmo que solo se va a poder usar con HTTP y que es exactamente igual a Generic hash.

IIS

Servidor web de Microsoft, IIS (Internet Information Server), con extensión call (Solicitud de aplicación ROLLing) le permite convertirlo en un balanceador de carga usando diferentes algoritmos de entrega, como los ya mencionados para Apache y Nginx.

Sólo el 9% de los servidores web del mundo son IIS, mientras que el 85 % está representado por Apache y Nginx.

Sticky sessions

Es un proceso en el que un balanceador de carga establece una relación entre un cliente y un servidor de red en particular durante la duración de la sesión (es decir, el tiempo que una dirección IP en particular pasa en una página web). web).

El uso de sesiones persistentes puede ayudar a mejorar la experiencia del usuario y optimizar el uso de los recursos de la red. Para las sesiones persistentes, el balanceador de carga asigna una identidad al usuario, generalmente emitiendo una cookie o rastreando su información de IP. Luego, según el ID de seguimiento, el balanceador de carga puede comenzar a enrutar todas las solicitudes de ese usuario a un servidor específico durante la sesión.

Esto puede ser muy útil, ya que HTTP/S es un protocolo sin estado que no está diseñado teniendo en cuenta la persistencia de la sesión.

Sin embargo, muchas aplicaciones web necesitan proporcionar datos de usuario personalizados (por ejemplo, elementos del carrito de compras o conversaciones de chat) durante una sesión. Sin continuidad de la sesión, la aplicación web tendrá que mantener esta información en varios servidores, lo que puede ser ineficiente, especialmente para redes grandes.



6.4.

Virtualización

La tecnología estrechamente relacionada con la alta disponibilidad es la virtualización,

Permite ejecutar múltiples máquinas virtuales (V TVI) en una misma máquina física, lo que se denomina hipervisor. Esta es una gran solución para usar y reducir los recursos del servidor físico y así poder reducir los costes de hardware, ya que solo hay un servidor físico (potente Pem), que ejecuta el hipervisor

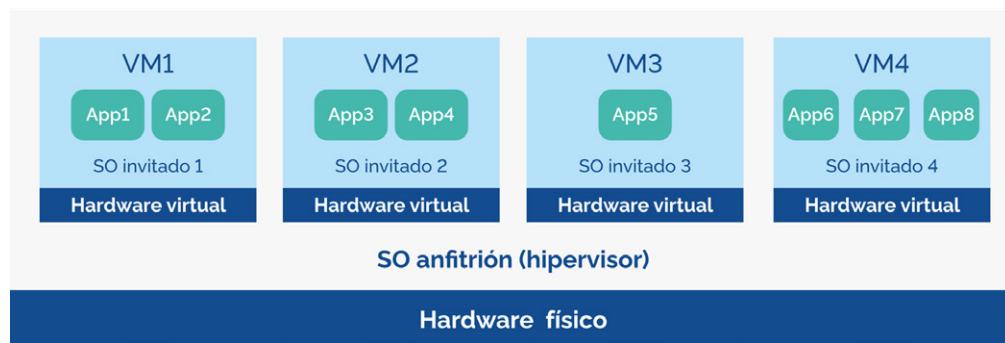


Imagen 8. Las VM residen, junto a su hardware virtual, dentro del hipervisor, que es quien realmente se comunica con el hardware físico

6.4.1. Ventajas de la virtualización

- > Reutilización del hardware
- > Minimizar los costes tanto a nivel económico, como a nivel de espacio
- > Aumento de la velocidad de despliegue
- > Facilidad para generar entornos para estudiar las nuevas herramientas
- > Seguridad y facilidad en la realización de las copias de seguridad
- > Minimizar el tiempo de inactividad
- > Aportación al medio ambiente
- > Migración en caliente de VM entre nodos de un clúster

6.4.2. Soluciones de virtualización

Podemos diferenciar entre tres tipos de hipervisores:

- > **Aplicación hipervisor:** se ejecuta en el sistema operativo host como una aplicación. Más que eso, pero permite la creación e implementación de máquinas virtuales. Se utilizó fundamentalmente para el aislamiento de aplicaciones, desarrollo de software, simulación del entorno de prueba, etc. Algunos ejemplos de este tipo son Oracle VirtualBox y Microsoft.
- > **Sistema operativo hipervisor:** está elaborado para la creación y la ejecución de VM. Además, es importante que sea capaz de establecer un clúster de gran disponibilidad.
- > **Virtualización en la nube o cloud virtualization:** la localización del sistema operativo. Los administradores y toda su infraestructura de hardware se encuentran en servidores remotos. Las plataformas más importantes son Microsoft Azure y Amazon EC2.



6.4.3. Configuración de máquinas virtuales

La configuración de la máquina virtual es la disposición de los recursos asignados a una máquina virtual. Normalmente, los recursos asignados a una máquina virtual (VM) incluyen el procesador, la memoria, el disco, el adaptador de red y la interfaz de usuario específica. La configuración se refiere a cada uno de los elementos específicos que se incluyen y cómo se configuran esos elementos.

Las máquinas virtuales se pueden crear rápida y fácilmente, pero los valores de configuración predeterminados pueden no ser los mejores para el rendimiento de la máquina virtual. Cada máquina virtual solo debe configurarse con los recursos que necesita. Por ejemplo, si una máquina virtual en particular no requiere una unidad de CD o disquete, la eliminación de estas entradas mejorará el rendimiento de la máquina virtual.

Independientemente de la solución elegida, crear una VM va a basarse en definir el hardware que va a estar constituido de los siguientes elementos:

- > CPU
- > Disco duro
- > MEmori
- > Interfaces de red

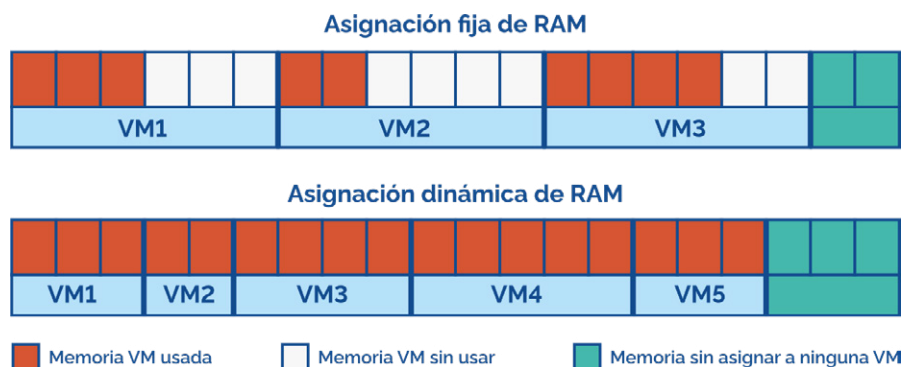


Imagen 9. Un hipervisor con 20GB de RAM y varias VM de 6 GB cada una

La gran parte de los hipervisores pueden optimizar la memoria RAM de su máquina virtual: ellos gestionando siempre la cantidad de memoria que realmente utilizan, en lugar de almacenar toda la memoria de cada máquina virtual en RAM para su uso, solo conservan la cantidad que utilizan, para que puedan ejecutar varias máquinas virtuales al mismo tiempo.

El motivo de esta práctica, lógicamente, es que todas las VM empiezan a abusar de su memoria RAM física finalmente se agotó y me vi obligado a cambiar a la memoria secundaria (swap), esto ralentizará mucho el sistema.



6.4.4. Virtualización mediante contenedores

Algunos hipervisores, como Pmxmox, admiten dos modos de virtualización: máquina virtual (KVM) y contenedor (LXC). Aunque los contenedores también contienen máquinas virtuales, comparten un kernel de sistema operativo con un hipervisor, lo que tiene varias ventajas:

- > Las máquinas virtuales tardan minutos en arrancar mientras que los contenedores lo hacen en segundos.
- > En una actualización del sistema, también se aplicará a los contenedores.
- > Cuando se comparte el núcleo del sistema con el hipervisor anfitrión, solo podrá instalarse los que sean compatibles con el núcleo mencionado.
- > Cuando se comparte el núcleo será fundamental que los contenedores accedan como usuario root.

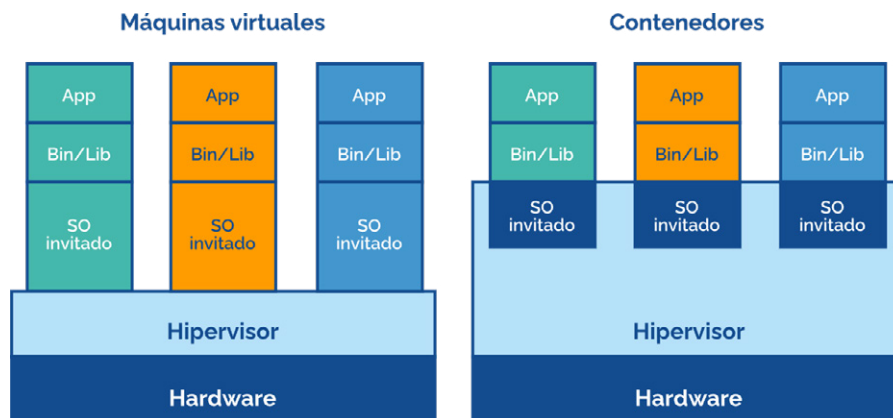


Imagen 10. Diferencia entre máquinas virtuales y contenedores

6.4.5. Virtualización de redes

No solo computadoras virtuales, sino también dispositivos de red virtual. Sin embargo, para ello es necesario recurrir a monitores especializados como Cisco Nexus 1000 V o VNIware CSX. Las cosas mencionadas en el Tema 5, son un caso especial de red virtual. Son todas soluciones complejas y avanzadas, por lo que no profundizaremos en ellas.



 www.universae.com

