

UNIVERSIDAD LUTERANA SALVADOREÑA CIENCIAS DEL
HOMBRE Y LA NATURALEZA



Licenciatura en Ciencias de la Computación

Materia: Sistemas Operativos de Redes.

Catedrático: Irwin Guardado

Tema: Perfil de proyecto, "Clúster de Alta disponibilidad"

Integrantes:

Apellidos	Nombres	Participación
Carrillos Gómez	Jennifer Yamileth	(100%-100%)
Crespín Aldana	Katerin Mariela	(100%-100%)
Campos	Irvin Balmore	(100%-100%)

Índice

Resumen.....	4
Palabras clave.....	5
Introducción.....	6
Marco Teórico.....	7
Beneficios de la tecnología clúster	8
¿Qué es un clúster de alta disponibilidad?.....	9
Nodos:.....	9
Servicios Utilizados:.....	9
Heartbeat:.....	10
Apache.....	10
Principales características de Apache2	10
Consiste en instalar, configurar e integrar todos los componentes	10
Materiales.....	11
Instalación de las máquinas virtuales.....	14
.....	16
<i>Asistente de virtual box para clonar máquinas virtuales</i>	16
Instalación de Heratbeat para el clúster de servidores apache y la alta disponibilidad	17
Primero instalamos el servidor web apache2.....	18
A continuación instalamos HEARTBEAT:.....	19
Estos 3 ficheros son:	20
Ahora copiamos el fichero ha.cf	21
Esta configuración será igual en los dos nodos	22
La Ip que colocaremos será la Ip del nodo contrario.	24
Esta configuración será igual en los dos nodos	25
Reiniciamos el servicio de HEARTBEAT en ambos nodos.....	26
Métodos.....	26
Herramientas de configuración y administración de un clúster	27
Conclusión	29
Glosario.....	30
Recomendaciones	31
Bibliografía	33

Resumen

La configuración de un clúster para tener la alta disponibilidad de un servidor web, todo ello con software libre. Para ello se ha utilizado la herramienta de virtualización KVM y el sistema operativo Linux Debían.

El funcionamiento principal consiste en recibir peticiones web por parte de los usuarios, estas peticiones son administradas por los dos nodos principales que reparten la carga a los servidores web. Estos dos nodos principales están en continua comunicación para saber el estado de cada uno de ellos en todo momento y así conseguir la alta disponibilidad.

En la actualidad, debido a la gran demanda de servicios de internet y a la transferencia de la información de todo tipo, es incuestionable que los sistemas informáticos deben funcionar de forma ininterrumpida y sin errores los 365 días del año.

Palabras clave

Balanceo de carga, alta disponibilidad, servicios web, software libre Clúster, Servidor Web, Heartbeat, apache2, ha.cf, haresources, authkeys.

Introducción

El presente documento contiene el perfil de nuestro proyecto “Clúster de alta disponibilidad” a realizarse en la materia de Sistemas Operativos de redes.

Dentro del documento, se presentan las diferentes tecnologías asociadas, una descripción de lo que se pretende implementar así como un estudio básico de factibilidades.

La necesidad de tener siempre un servicio activo, de tal manera que siempre este activo el servicios de ficheros en un servido, tratando de evitar errores tales como que si el servidor se arruine, falle por causa de software o hardware, o cualquier otra falla y no pueda satisfacer la demanda de uno o varios usuarios, nos hace implementar servidores de alta disponibilidad. Al hablar de servidor de alta disponibilidad nos estamos refiriendo a clúster, que permitan tener en funcionamiento de forma permanente el acceso a los ficheros que son demandados por los usuarios.

En este proyecto se han configurado dos servidores virtuales y un cliente, para demostrar que si funciona un clúster y que si hay alta disponibilidad.

La alta disponibilidad hace referencia a que un sistema debe estar funcionando las 24 horas del día los 365 días al año. Para poder alcanzar esta característica es necesario el uso de técnicas de balanceo de carga y de recuperación ante fallos.

Marco Teórico

El término de clúster se aplica en los conjuntos conglomerados de ordenadores unidos entre si normalmente por una red de alta velocidad y que se comportan como si fuesen una única computadora.

La tecnología de clúster ha evolucionado con el apoyo de las actividades que van desde aplicaciones de súper cómputo y software para aplicaciones críticas, servidores web y comercio electrónico, hasta base de datos de altos rendimientos, entre otros usos.

Un clúster es un grupo de múltiples ordenadores unidos mediante una red de alta velocidad, de tal forma que el conjunto es visto como único ordenador, más potente que los comunes de escritorio.

De un clúster se espera que presente combinaciones de los siguientes servicios

1. Alto rendimiento
2. Alta disponibilidad
3. Balanceo de carga
4. Escalabilidad

¿Qué es un clúster?

En términos informáticos, entendemos clúster como conjuntos o conglomerados de computadoras construidos mediante la utilización de componentes de hardware comunes y que se comportan como si fuesen una única computadora.

Existen varios tipos de clúster:

- Alto rendimiento
- Alta disponibilidad
- Escalabilidad
- Balanceo de carga.

En el proyecto hemos elegido la opción de montar un clúster de alta disponibilidad.

Clasificación de los Clúster, según su configuración.

- Homogéneos: Pueden tener todas las mismas configuraciones de hardware y sistema operativo.
- Semi homogéneos: Diferente rendimiento pero con arquitecturas y sistemas operativos similares.
- Heterogéneos: Tienen diferente hardware y sistema operativo, lo que hace más fácil y económica su construcción.

Beneficios de la tecnología clúster

Las aplicaciones paralelas escalables requieren: buen rendimiento, baja latencia, comunicaciones que dispongan de gran ancho de banda, redes escalables y acceso rápido a archivos. Un clúster puede satisfacer estos requisitos usando los recursos que tiene asociados a él; La tecnología clúster permite a las organizaciones incrementar su capacidad de procesamiento usando tecnología estándar, tanto en componentes de hardware como de software que pueden adquirirse a un costo relativamente bajo.

Clasificación de los clústeres:

El término clúster tiene diferentes connotaciones para diferentes grupos de personas. Los tipos de clústeres, establecidos de acuerdo con el uso que se dé y los servicios que ofrecen, determinan el significado del término para el grupo que lo utiliza.

Los clústeres pueden clasificarse según sus características:

- HPCC Clúster: clústeres de alto rendimiento.
- HA o HACC Clúster: clústeres de alta disponibilidad.
- HT o HTCC Clúster: clústeres de alta eficiencia.

Alto rendimiento:

Son clústeres en los cuales se ejecutan tareas que requieren de gran capacidad computacional, grandes cantidades de memoria, o ambos a la vez.

El llevar a cabo estas tareas puede comprometer los recursos del clúster por largos periodos de tiempo.

Alta disponibilidad:

Son clústeres cuyo objetivo de diseño es el de proveer disponibilidad y confiabilidad. Estos clústeres tratan de brindar la máxima disponibilidad de los servicios que ofrecen. La confiabilidad se provee mediante software que detecta fallos y permite recuperarse frente a los mismos, mientras que en hardware se evita tener un único punto de fallos.

Alta eficiencia:

Son clústeres cuyo objetivo de diseño es el ejecutar la mayor cantidad de tareas en el menor tiempo posible. Existe independencia de datos entre las tareas individuales. El retardo entre los nodos del clúster no es considerado un gran problema.

¿Qué es un clúster de alta disponibilidad?

Un clúster de alta disponibilidad es un conjunto de dos o más máquinas que se caracterizan por mantener una serie de servicios compartidos y por estar constantemente motorizándose entre sí.

La construcción de los ordenadores del clúster es más fácil y económica debido a su flexibilidad: pueden tener todas las mismas configuraciones de hardware y sistema operativo (clúster homogéneo), diferente rendimiento pero con arquitecturas y sistemas operativos similares (clúster semihomogéneo), o tener diferente hardware y sistema operativo (clúster heterogéneo), lo que hace más fácil y económica su construcción

Para que un clúster funcione como tal no basta solo con conectar entre si los ordenadores, sino que es necesario promover un sistema de manejo de clúster, el cual se encarga de interactuar con el usuario y los procesos que corren en el para optimizar el funcionamiento. No hay que confundir un clúster de alta disponibilidad con un clúster de alto rendimiento. El segundo es una configuración de equipo diseñado para proporcionar capacidades de cálculo mucho mayor que la proporcionan los equipos individuales.

Nodos:

Es un punto de intersección, conexión o unión de varios elementos que confluyen en el mismo lugar, En redes de computadoras cada una de las máquinas es un nodo, y si la red es Internet, cada servidor constituye también un nodo

Ahora bien, dentro de la informática la palabra nodo puede referirse a conceptos diferentes según el ámbito en el que nos movamos:

- En redes de computadoras cada una de las máquinas es un nodo, y si la red es Internet, cada servidor constituye también un nodo.
- En estructuras de datos dinámicas un nodo es un registro que contiene un dato de interés y al menos un puntero para referenciar (apuntar) a otro nodo. Si la estructura tiene solo un puntero, la única estructura que se puede construir con él es una lista, si el nodo tiene más de un puntero ya se pueden construir estructuras más complejas como árboles o grafos.

Servicios Utilizados:

Como mencionamos anteriormente para realizar Clúster, es necesario proveer un sistema que se encargue de manejar las operaciones entre los equipos ya sean físicos o virtuales.

A continuación listamos algunos de los investigados.

Heartbeat:

- Es un Servicio (Demonio) que proporciona funcionalidad de infraestructura de clúster (comunicación y pertenencia) a sus clientes. Esto permite a los clientes tener conocimiento de la presencia (o desaparición) de los procesos en otras máquinas e intercambiar fácilmente mensajes entre ellos. Para resultar útil a los usuarios el dominio HEARTBEAT necesita emplearse en combinación con un gestor de recursos del clúster (clúster resource manager (CRM)) el cual posee la tarea de iniciar y parar los servicios (Direcciones IP, servidores web...) a los cuales el clúster aportará alta disponibilidad.

Para que sirve: Heartbeat es el Encargado de revisar que cada nodo se halle funcionando. En caso de que un nodo falle migrará los recursos a otro nodo que también se halle ejecutando el servicio heartbeat.

Apache

Apache es un poderoso servidor web es completamente libre, ya que es un software Open Source y con licencia GPL. Una de las ventajas más grandes de Apache, es que es un servidor web multiplataforma, es decir, puede trabajar con diferentes sistemas operativos y mantener su excelente rendimiento.

Principales características de Apache2

Entre las principales características de Apache, se encuentran las siguientes:

- Soporte de seguridad SSL y TLS.
 - Puede realizar autenticación de datos utilizando SGDB.
 - Puede dar soporte a diferentes lenguajes, como Perl, PHP, Python y tcl.
- Apache es un servidor web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos HTTP.
 - **Multiplataforma.**

Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.

Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor

Estrategia de Implementación

Consiste en instalar, configurar e integrar todos los componentes

- Configurar e integrar todos los componentes.

Es necesario realizar las siguientes actividades:

- Diseñar la arquitectura física del clúster.
- Instalar y configurar el sistema operativo GNU/Linux.
- Instalar y configurar el sistema de archivos para clústeres o el administrador de sincronización de datos.
-

Materiales

Requerimientos Básicos para la implementación del clúster

- **3 Computadoras o más con tarjetas de red Ethernet (Pueden ser máquinas virtuales).**
- **1 Red Ethernet.**
- **Acceso a Internet.**
- **Cada computadora debe tener como sistema operativo, Debían.**
- ISO de instalación del sistema operativo Debian9
- Apache2
- Hearbeat
- Virtualbox

Presupuesto

ACTIVIDADES	INSUMOS	VALOR UNITARIO	SUB-TOTAL
RECURSOS HUMANOS	Gastos de pasajes	\$40	\$95.00
	comida	\$55	
	3 Estudiantes Días	Miércoles, Sábado de 1:00pm a 3:30	
TOTAL DEL RUBRO A			\$95.00
RECURSOS MATERIALES	2 computadoras Debian9 GNU/LINUX de 64GB	\$800	\$800
		TOTAL DEL RUBRO B	\$800
TOTAL			\$895

Instalación de las máquinas virtuales

Herramientas necesarias

- ISO de instalación del sistema operativo Debian9
- Apache2
- Heartbert
- Virtual box

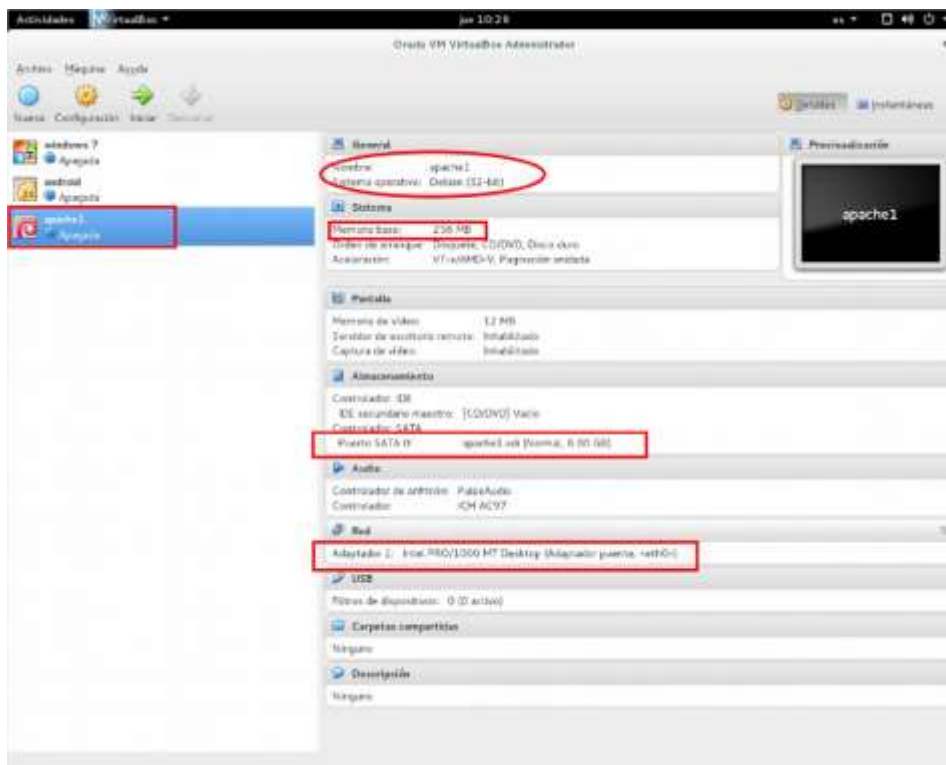
Creación de las máquinas virtuales:

Vamos a montar 2 máquinas virtuales para apache y una máquina virtual para el clúster de alta disponibilidad.

Instalación del sistema operativo en las máquinas virtuales

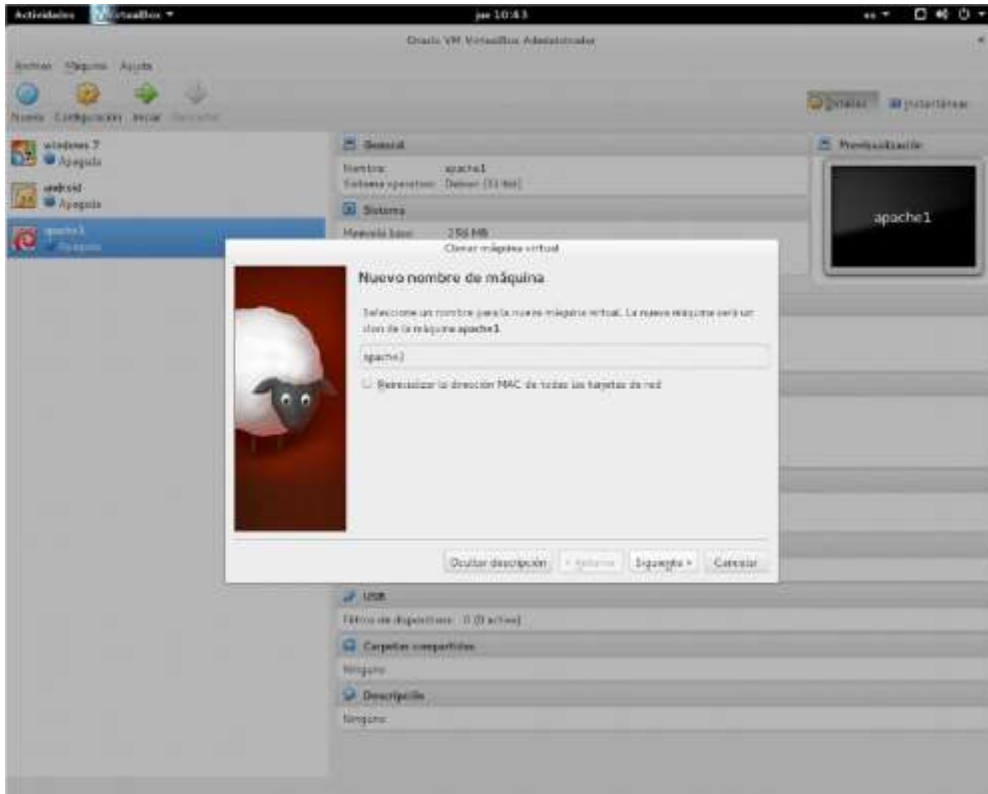
Para la instalación del sistema operativo en las máquinas virtuales utilizaremos la siguiente configuración

Parámetros de instalación, para las máquinas virtuales.



Sistema operativo: Debian9 GNU/Linux de 64 bits

No es necesario hacer la instalación de las otras máquinas virtuales, simplemente se instala una maquina con los recursos mencionados y clonamos el resto.



Asistente de virtual box para clonar máquinas virtuales

Para clonar una máquina virtual hacemos clic derecho sobre la maquina en el administrador de virtual box, seleccionando la opción de clonar y el asistente de virtual box nos pedirá el nombre que le daremos a la nueva máquina virtual.

Para el ejemplo usaremos las ip siguientes

- IP address: 192.168.15.180; Apache document root: /var/www
- IP address: 192.168.15.181; Apache document root: /var/www
- Dirección Virtual IP: 192.168.15.182 (se usara para las peticiones entrantes)

Instalación de Heratbeat para el clúster de servidores apache y la alta disponibilidad

Primero configuramos nuestras IP estáticas en cada máquina.

Recomendación: poner las IP en forma ascendente ya que HEARTBEAT reconoce la IP de la maquina principal de esta misma forma:

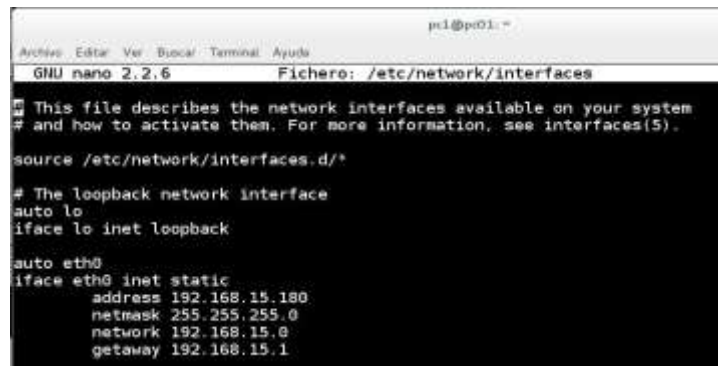
Configuración de interfaces en el archivo `/etc/network/interfaces` en cada maquina

nano /etc/network/interfaces



```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# nano /etc/network/interfaces
```

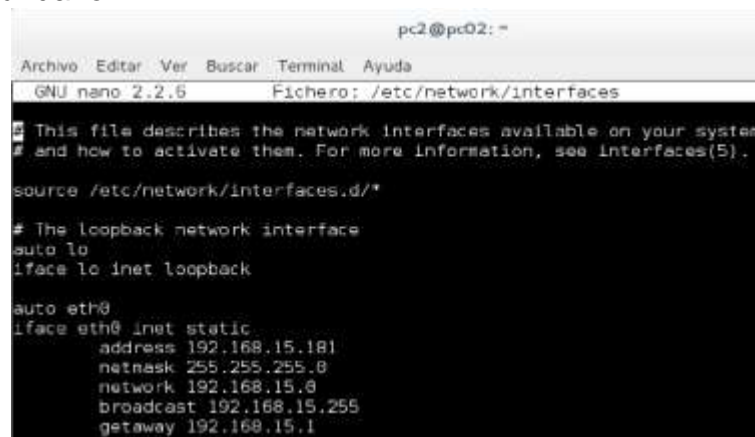
Servicio web principal:



```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.2.6 Fichero: /etc/network/interfaces  
# This file describes the network interfaces available on your system  
# and how to activate them. For more information, see interfaces(5).  
source /etc/network/interfaces.d/*  
  
# The loopback network interface  
auto lo  
iface lo inet loopback  
  
auto eth0  
iface eth0 inet static  
    address 192.168.15.180  
    netmask 255.255.255.0  
    network 192.168.15.0  
    gateway 192.168.15.1
```

Configuraciones del archivo que controla las interfaces de red

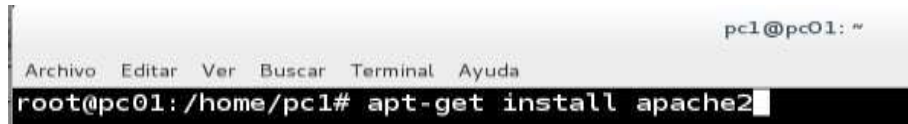
Servicio web secundario:



```
pc2@pc02: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.2.6 Fichero: /etc/network/interfaces  
# This file describes the network interfaces available on your system  
# and how to activate them. For more information, see interfaces(5).  
source /etc/network/interfaces.d/*  
  
# The loopback network interface  
auto lo  
iface lo inet loopback  
  
auto eth0  
iface eth0 inet static  
    address 192.168.15.181  
    netmask 255.255.255.0  
    network 192.168.15.0  
    broadcast 192.168.15.255  
    gateway 192.168.15.1
```


Primero instalamos el servidor web apache2

```
# apt-get install apache2
```

A terminal window screenshot showing the command 'apt-get install apache2' being executed. The terminal title is 'pc1@pc01: ~'. The menu bar includes 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The prompt is 'root@pc01:/home/pc1#' and the command is 'apt-get install apache2' with a cursor at the end.

```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# apt-get install apache2
```

Luego detenemos el

```
servicio # service
```

```
apache2 stop
```

A terminal window screenshot showing the command 'service apache2 stop' being executed. The terminal title is 'pc1@pc01: ~'. The menu bar includes 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The prompt is 'root@pc01:/home/pc1#' and the command is 'service apache2 stop' with a cursor at the end.

```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# service apache2 stop
```

Para evitar que el servicio apache2 se levante al

```
inicio # update-rc.d -f apache2 disable
```

A continuación instalamos HEARTBEAT:

```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# update-rc.d -f apache2 disable
```

apt-get install heartbeat

```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# apt-get install heartbeat
```

Tras la instalación de HEARTBEAT habrá aparecido en la ruta **/usr/share/doc/heartbeat**, la cual contiene 3 ficheros que debemos configurar para que HEARTBEAT funcione.

Estos 3 ficheros son:

Authkeys: Fichero de autenticación entre nodos. **ha.cf:**

Fichero principal de configuración de HEARTBEAT.

Haresources: Fichero donde se indica cuál será el nodo activo. Así como los servicios a gestionar.

Luego de haber instalado HEARTBEAT descomprimos y copiamos los ficheros de configuración de HEARTBEAT en la ruta /etc/ha.d

Descomprimos **ha.cf.gz**

```
#gzip -d /usr/share/doc/heartbeat/ha.cf.gz
```



```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# gzip -d /usr/share/doc/heartbeat/ha.cf.gz
```

Descomprimos **haresources.gz**

```
#gzip -d /usr/share/doc/heartbeat/haresources.gz
```



```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# gzip -d /usr/share/doc/heartbeat/haresources.gz
```

Ahora copiamos el fichero **authkeys**

```
#cp /usr/share/doc/heartbeat/authkeys /etc/ha.d/
```



```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# cp /usr/share/doc/heartbeat/authkeys /etc/ha.d/
```

Ahora copiamos el fichero ha.cf

```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# cp /usr/share/doc/heartbeat/ha.cf /etc/ha.d/
```

```
#cp /usr/share/doc/heartbeat/ha.cf
```

```
/etc/ha.d/ Ahora copiamos el fichero
```

haresource

```
#cp /usr/share/doc/heartbeat/haresource /etc/ha.d/
```

```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# cp /usr/share/doc/heartbeat/haresources /etc/ha.d/
```

authkeys: en este fichero estableceremos la contraseña de autenticación compartida por los dos nodos.

```
#nano /etc/ha.d/authkeys
```

```
pc1@pc01: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@pc01:/home/pc1# nano /etc/ha.d/authkeys  
# sha1 is believed to be the "best", md5 next best.  
#  
# crc adds no security, except from packet corruption.  
# Use only on physically secure networks.  
#  
auth 1  
1 crc  
#2 sha1 HI!  
#3 md5 Hello!
```

Fichero Authkeys

Le damos permisos para que solo pueda ser accedido por el propietario

```
del mismo #chmod 600 /etc/ha.d/authkeys
```

Esta configuración será igual en los dos nodos

ha.cf: este fichero principal de configuración de HEARTBEAT. Aquí descomentaremos algunas líneas que son esenciales para el correcto funcionamiento de HEARTBEAT. #nano

/etc/ha.d/ha.cf

#Tiempo transcurrido entre el envío de cada uno de los latidos (segundos)

keepalive 2

#Tiempo transcurrido hasta declarar al nodo como caído (segundos)

deadtime 10

```
#
#
#   keepalive: how long between heartbeats?
#
keepalive 2
#
#   deadtime: how long-to-declare-host-dead?
#
#       If you set this too low you will get the problematic
#       split-brain (or cluster partition) problem.
#       See the FAQ for how to use warntime to tune deadtime.
#
deadtime 10
#
```

#Tiempo transcurrido para comenzar a levantar los servicios (segundos)

initdead 120

#Puerto para la comunicación UDP

udpport 694

```

#
#       Very first dead time (initdead)
#
#       On some machines/OSes, etc. the network takes a while to come up
#       and start working right after you've been rebooted. As a result
#       we have a separate dead time for when things first come up.
#       It should be at least twice the normal dead time.
#
initdead 120
#
#       What UDP port to use for bcast/ucast communication?
#
udpport 694

```

Esta configuración será igual en los dos nodos

```

#Interfaz utilizado para enviar los mensajes de broadcast
bcast eth0

```

```

#
#
#       What interfaces to broadcast heartbeats over?
#
bcast   eth0           # Linux
#bcast  eth1 eth2      # Linux
#bcast  le0            # Solaris
#bcast  le1 le2        # Solaris
#

```

```

#interfaz y nodo al que enviar los latidos
ucast eth0 h2

```

```

#mcast eth0 225.0.0.1 694 1 0
#
#       Set up a unicast / udp heartbeat medium
#       ucast [dev] [peer-ip-addr]
#
#       [dev]           device to send/rcv heartbeats on
#       [peer-ip-addr] IP address of peer to send packets to
#
ucast eth0 192.168.15.181
#

```

La Ip que colocaremos será la Ip del nodo contrario.

Si el nodo1 tiene Ip 192.168.15.180 entonces la configuración cruzada será 192.168.15.181

Si el nodo2 tiene Ip 192.168.15.181 entonces la configuración cruzada será

192.168.15.180 # si el nodo cae y se recupera, vuelve a su rol original.

```
# See the FAQ for information on how to convert
# from "legacy" to "on" without a flash cut.
# (i.e., using a "rolling upgrade" process)
#
# The default value for auto_failback is "legacy", which
# will issue a warning at startup. So, make sure you put
# an auto_failback directive in your ha.cf file.
# (note: auto_failback can be any boolean or "legacy")
#
auto_failback on
#
```

Esta configuración será igual en los dos

nodos #servidor principal y secundario.

```
#
#watchdog /dev/watchdog
#
# Tell what machines are in the cluster
# node      nodename ...    -- must match uname -n
node      pc01
node      pc02
#
```

Esta configuración será igual en los dos nodos

Haresource: este fichero contiene información de los recursos que deseamos que tenga alta disponibilidad. Al principio de este archivo estableceremos quien será el servidor que predominará. La IP virtual y el servicio que se brindará.

```
#nano /etc/ha.d/haresource
```




```
pc1@pc01: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.6 Fichero: /etc/ha.d/haresources
pc01 IPaddr::192.168.15.182/24/eth0 apache2
#
# This is a list of resources that move from machine to machine as
# nodes go down and come up in the cluster. Do not include
# "administrative" or fixed IP addresses in this file.
#
```

Esta configuración será igual en los dos nodos

Luego modificamos el fichero index.html ubicado en
`/var/www/html/index.html` #nano /var/www/html/index.html

Para tener una prueba gráfica del servicio.



```
pc1@pc01: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.6 Fichero: /var/www/html/index.htm
<!DOCTYPE html!>
<html>
  <head>
    <title> prueba </title>
  </head>
  <body>
    <h1> computadora pc01 </h1>
    <h2> con IP 192.168.15.180 </h2>
  </body>
</html>
```

Ilustración 7: Index, del servidor apache


```
pc2@pc02: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.6 Fichero: /var/www/html/index.html
<!DOCTYPE html!>
<html>
  <head>
    <title> prueba 2 </title>
  </head>
  <body>
    <h1 style="color: red"> Computadora pc02 </h1>
    <h2> Con IP 192.168.15.181 </h2>
  </body>
</html>
```

Reiniciamos el servicio de HEARTBEAT en ambos nodos

```
/etc/init.d/heartbeat stop
/etc/init.d/heartbeat start
```

Luego encendemos las máquinas y levantamos el servicio
service heartbeat start

En ambas máquinas y listo.

Métodos

Software para la implementación de un clúster

En la actualidad hay varios tipos de software para la implementación y administración del clúster. A continuación se muestra un análisis comparativo de las mejores herramientas que se pueden implementar.

	Descripción	Alternativas
Software para clustering	Alta disponibilidad	Pirahna Open Mosix
	Balanceo de carga	Ultra Monkey Pirahna Open Mosix
	Configuración e instalación	FAI SIS System Image System Installer System Configuration
	Monitorización e Instalación	LVSmonn, Sincopt, Fsync, Ghosts, y Pconsole.
	Monitorización	Mon Heartbeat Fake Coda Ganglia

Herramientas de configuración y administración de un clúster

clúster		Ventajas	Desventajas
Alta disponibilidad	Conjunto de dos o más máquinas que se caracterizan por mantener una serie de servicios compartidos y por estar constantemente motorizándose entre sí.	Baja latencia, comunicaciones que dispongan de gran ancho de banda, redes escalables y acceso rápido a archivos. Alto rendimiento Alta disponibilidad Alta eficiencia Escalabilidad	-Rendimiento bajo de la interconexión, no se puede escalar a grandes números de procesadores -Empresas y entidades prefieren seguir utilizando el modelo cliente/servidor tradicional debido al espacio físico o a nuevos problemas que no se daban en la arquitectura tradicional. -Espacio físico para el montaje del clústeres de balanceo de carga.

Balance de carga	<p>El balanceo de carga, comparte el trabajo a realizar entre varios procesos, ordenadores, discos u otros recursos. Está íntimamente ligado a los sistemas de multiprocesamiento, o que hacen uso de más de una unidad de procesamiento para realizar labores útiles.</p>	<p>Envía las solicitudes al nodo con menos carga</p> <p>Si uno de los nodos de Apache se cae, el balanceador de carga se dará cuenta de eso y dirigirá todas las peticiones entrantes al nodo restante</p> <p>La alta disponibilidad en LVS se diseña utilizando software mon, heartbeat, fake y coda, que se utilizan para gestionar la alta disponibilidad y para mantener una gestión segura, la comunicación (heartbeat) entre máquinas</p>	
-------------------------	--	---	--

Conclusión

Los clúster de alta disponibilidad, es un área relativamente nueva en el mercado ya que a través de esta tecnología se pueden realizar innovaciones. A pesar de existir clústeres construidos por grandes compañías y a un alto costo, todavía se pueden construir y configurar clústeres más accesibles en cuanto a su licencia y funcionalidad. Este es el primer paso para su diseño, construcción y configuración, con el fin de satisfacer requerimientos más específicos planteados por los administradores de clústeres.

Por otra parte el software libre ha permitido desarrollar todas las aplicaciones necesarias para construir el clúster web, demostrando su viabilidad y madurez para implementar soluciones que respondan a problemas reales.

Antes de este proyecto no sabíamos que era un servidor de ficheros de alta disponibilidad, ni cómo funcionaba, ahora podemos decir que hemos adquirido suficiente conocimiento como para configurar e instalar todo lo necesario para crearlo

Glosario

Alta disponibilidad:

Es un sistema orientado a ofrecer y garantizar servicios en Alta Disponibilidad, es decir, con un alto grado de fiabilidad y de continuidad operativa.

Heartbeat:

Pulso o "latido" mediante el cual se mantiene la comunicación entre los nodos del clúster. Si el nodo activo no responde al latido, el nodo pasivo toma el control y despliega de inmediato los servicios replicados.

Clúster

Conjunto de computadoras construidas mediante la utilización de Componentes de hardware que se comportan como si fuesen una única computadora.

Apache2

Es el servidor Web HTTP de código abierto

SSH

Es un protocolo que sirve para conectarse remotamente a las máquinas de una red

Recomendaciones

1- Antes de ponerse a querer crear un servidor de alta disponibilidad de ficheros es bueno informarse primero de que es, como funciona y para qué sirve.

2- Decidir si se va configurar en máquinas reales o en máquinas virtuales cada una de estas

Opciones tiene su pro y sus contras.

3- Si se realizara con máquinas virtuales ver si la maquina física puede aceptar virtualización. La computadora en la que se harán las máquinas virtuales deberá estar bien dotada de recursos de hardware capaz de soportar la virtualización y llenar requerimientos tales como suficiente cantidad de memoria RAM, un buen Disco Duro, y

Un microprocesador (CPU) con una buena velocidad, para que a la hora de arrancar las máquinas virtuales no se ponga lenta.

4- Crear un manual paso a paso de instalación del clúster, con capturas de pantalla y otro

Material que sirva para guiar a cualquier usuario que desee crear un servidor de ficheros de alta disponibilidad.

5- Apoyar la creación de un Clúster como servidor de ficheros de alta disponibilidad con la

Creación de un vídeo que muestre los pasos a seguir en la configuración para hacer más fácil la misma.

6- Disponer de todo el hardware necesario para crear y configurar el clúster (servidor de Ficheros de alta disponibilidad),

7- Estudiar todo lo referente a clúster de alta disponibilidad.

8- Preguntar, investigar, cuando aparezca algún error para solucionarlo. Apoyarse con el docente encargado de la materia SOR(Sistemas Operativos de Redes), o con otras personas que tengan experiencia en esta área.

Bibliografía

1- Wikipedia. Título "Clustering (manuales/cluster)"

<http://www.redes-lnuix.com/manuales/cluster/clustering.pdf>

2- Wikipedia. Título "Clúster (informática)"

[https://es.wikipedia.org/wiki/Cl%C3%BAster_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Cl%C3%BAster_(inform%C3%A1tica))

3- Publicado en Informática e Internet Título "Alta disponibilidad que es y cómo se logra"

<https://everac99.wordpress.com/2008/08/19/alta-disponibilidad-que-es-y-como-se-logra/>

4- Publicado por Sergio González "Alta disponibilidad"

http://www.sergio-gonzalez.com/doc/09-conceptos-de-clustering/html/cluster_alta_disponibilidad.html

5- ¿Cómo construir un cluster?

http://www.mcsr.olemiss.edu/bookshelf/articles/how_to_build_a_cluster.html

6- Un documento interesante sobre clusters computacionales sobre pc's:

<http://www.fisica.uson.mx/carlos/LinuxClusters/clustercomputing2000.pdf>

Sobre alta disponibilidad:

<http://www.linuxvirtualserver.org/HighAvailability.html>

7- Author: Falko Timme . Título "High availability loadbalanced and apache cluster"

https://www.howtoforge.com/high_availability_loadbalanced_apache_cluster