Seguridad y Alta Disponibilidad: Clústers HA

Jesús Moreno León

jesus.moreno.edu@juntadeandalucia.es

IES Gonzalo Nazareno
CONSEJERÍA DE EDUCACIÓN

Febrero 2013

Estas diapositivas son una obra derivada del artículo
“Creación de un Clúster de Alta Disponibilidad con software libre”
publicado por Miguel Vidal y José Castro en el número 209
(enero-febrero 2011) de la revista Novática, disponible en:
http://www.ati.es/novatica/2011/209/Nv209-75.pdf

© Jesús Moreno León, Febrero 2013

Algunos derechos reservados.
Este artículo se distribuye bajo la licencia
“Reconocimiento-CompartirIgual 3.0 España" de Creative
Commons, disponible en
http://creativecommons.org/licenses/by-sa/3.0/es/deed.es

Este documento (o uno muy similar)
está disponible en (o enlazado desde)
http://informatica.gonzalonazareno.org

¿Qué es un Clúster de Alta Disponibilidad?

Un **clúster HA** (High Availability) es un sistema orientado a ofrecer y garantizar servicios en Alta Disponibilidad, es decir, con un alto grado de fiabilidad y de continuidad operativa.

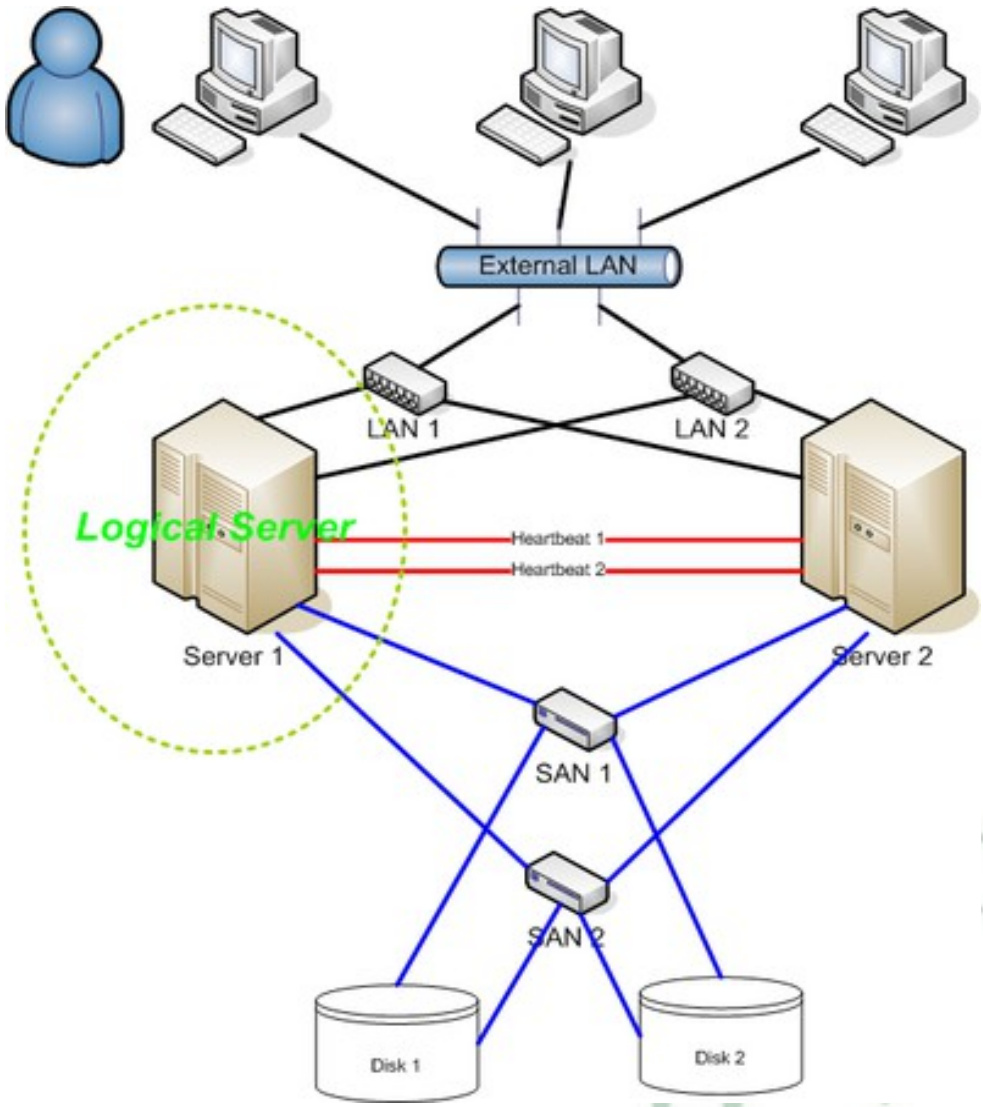
Se basa en **máquinas redundantes** (o nodos) que asumen el servicio cuando algún componente del sistema falla.

Un clúster HA debe ser capaz de detectar cualquier fallo de hardware o de software, reiniciar la aplicación en otro nodo y **mantener el servicio** sin intervención de operador alguno, garantizando la **integridad de los datos** del clúster.

¿Qué es un Clúster de Alta Disponibilidad?

El propósito de los clusters HA es eliminar los Puntos Únicos de Fallo (Single Point of Failure,SPoF), mediante redundancia a todos los niveles:

- Hardware
- Almacenamiento
- Conexiones de red
- ...



Fuente: http://en.wikipedia.org/wiki/High-availability_cluster

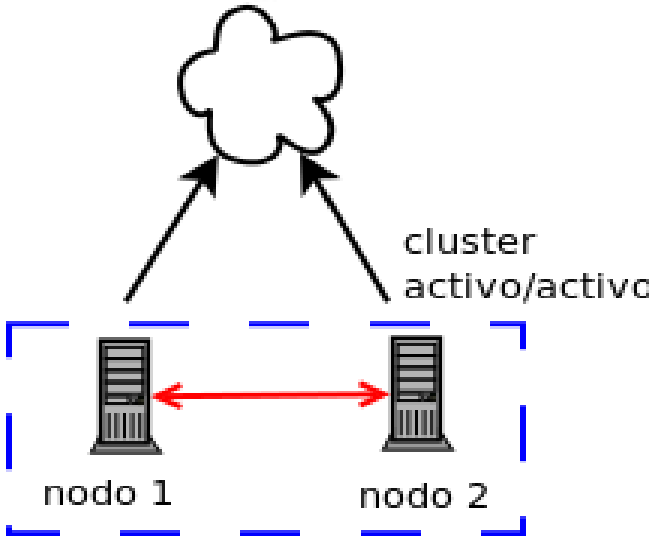
¿Qué es un Clúster de Alta Disponibilidad?

Para mantener la redundancia, la integridad y el control del clúster sobre los recursos es fundamental no realizar acciones sobre los nodos que no puedan ser replicadas, pues podrían crearse **inconsistencias**. Toda modificación de la configuración de los recursos debe hacerse mediante los agentes de recurso del propio clúster.

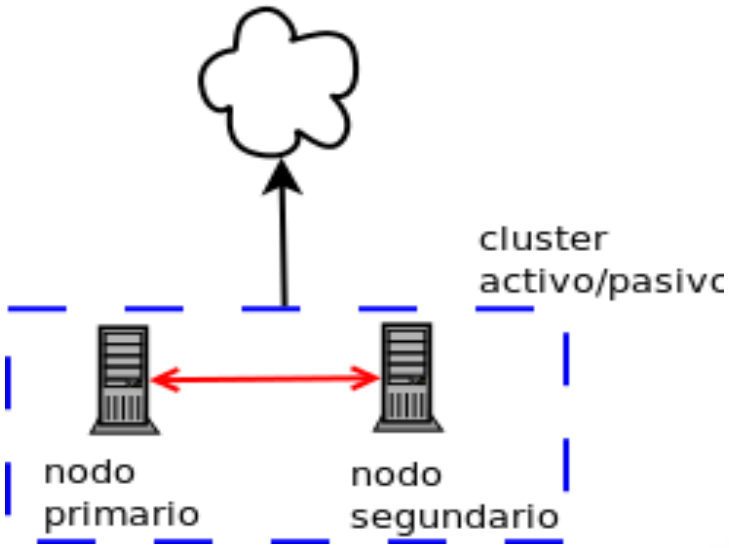
Los Clústers HA son muy utilizados en bases de datos críticas, sistemas de ficheros compartidos en red, aplicaciones de negocio o servicios de clientes como sitios web de comercio electrónico.

Tipos de configuración

El tamaño más habitual de un clúster HA es de dos nodos, ya que es el mínimo exigido para disponer de redundancia. Las dos configuraciones más comunes en los clusters de dos nodos son:



- Aprovecha mucho mejor los recursos físicos.
- Permite repartir mejor la carga entre los nodos.



- Configuración más sencilla

Fuente: http://ral-arturo.blogspot.com.es/2011/04/aproximacion-al-clustering-de-alta.html

Conceptos básicos

- **Failover**: capacidad de recuperarse de un fallo desplegando los servicios en otro nodo. "Clusters HA" = "failover clusters"
- **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.
  - Por lo general, se implementa mediante una conexión de red privada (dedicada) entre los nodos.
  - Debe ofrecer una infraestructura de comunicación del clúster completa y criptográficamente segura.

Conceptos básicos

- **Split-brain**: se produce cuando los enlaces de red que unen a los nodos entre sí caen, pero los nodos siguen operando. Se dice entonces que el clúster se ha "partido". Puede causar corrupción de datos en sistemas de almacenamiento compartido.
- **Quorum**: es un mecanismo para prevenir el split-brain. Se asigna un voto a cada nodo y se le permite operar si obtiene mayoría de votos. Con un clúster de dos nodos, la mayoría son dos votos, por lo que no es posible activar el quorum.

Conceptos básicos

- **STONITH**: "Shoot The Other Node In The Head ". Es un método automático de recuperación para desbloquear un nodo rápidamente, evitando que un administrador deba realizar la operación de forma manual.
  - Utiliza IPMI para el envío de mensajes.
  - El agente de recurso STONISH debe correr de forma obligatoria en el nodo contrario, para evitar que un nodo pueda llegar a "dispararse" a sí mismo.

Linux-HA

Linux-HA es una de las soluciones libres que permite implementar un clúster HA.

Está formado por diversos componentes, entre los que destacan:

- Heartbeat: proporciona los servicios de comunicación y membresía del clúster. **Corosync**.
- CRM (Cluster Resource Manager): se encarga de iniciar o detener los recursos a los que queremos dotar de alta disponibilidad. **Pacemaker**.

Práctica 1

- Sencillo cluster de alta disponibilidad con Linux-HA
  - Configurar un cluster de alta disponibilidad activo/pasivo en el que uno de los dos equipos pueda responder siempre a la dirección IP que se pretende mantener en alta disponibilidad.
  - Esta situación no es totalmente real, ya que lo lógico es que esos equipos estén ofreciendo algún servicio que se quiera mantener en alta disponibilidad (ldap, http, https, etc.)
  - Sin embargo es muy útil para comprender el funcionamiento del software de HA (pacemaker y corosync) sin las configuraciones adicionales necesarias para ofrecer los servicios en HA
  - Guía

Práctica 2

- Replicación de MySQL
  - Instalación de MySQL a partir de los binarios de MySQL
  - Configuración de los nodos
  - Configuración de la replicación MySQL en modo master-slave
  - Configuración de la replicación MySQL en modo master-master
  - Errores comunes
  - Guía

Práctica 3

- Balanceadores: ldirectord vs haproxy
  - Estudio, comparación y configuración de dos soluciones sofware para Linux: ldirectord y haproxy
  - Uso y funcionamiento de los balanceadores
  - Diferencias entre un balanceador proxy (haproxy) y uno que no lo es (ldirectord)
  - Configuraciones y pruebas básicas de ldirectord en modo NAT
  - Configuración y pruebas básicas de haproxy
  - Guía

Práctica 4

- Cluster Apache + PHP + Memcached
  - Aplicaciones de un cluster: failover y balanceo
  - Instalación del servidor Apache con PHP
  - Replicación de ficheros mediante rsync
  - Nociones sobre otros sistemas de replicación más sofisticados
  - Replicación de sesiones con rsync y con memcached
  - Guía