

**Solución de alta disponibilidad (HA) y balanceo de carga para el Servicio Web de la
Red de Datos de la Universidad del Cauca**



**Jeimmy Viviana Cuellar Rivera
José Raul Romero Mera**

ANEXO C

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telecomunicaciones
Grupo I+D Nuevas Tecnologías en Telecomunicaciones
Popayán, 2012**

TABLA DE CONTENIDO

LISTA DE FIGURAS	ii
ANEXO C PROYECTOS LIBRES DE ALTA DISPONIBILIDAD Y BALANCEO DE CARGA.....	1
C.1. Proyectos de propósito general.....	1
C.2. Proyectos De Propósito Específico	6
C.3. Proyectos enfocados en Maquinas Virtuales.....	7
BIBLIOGRAFÍA.....	8

LISTA DE FIGURAS

Figura C.1 *Arquitectura del proyecto Linux-HA*.....2

ANEXO C PROYECTOS LIBRES DE ALTA DISPONIBILIDAD Y BALANCEO DE CARGA

Dentro del software libre existe una variedad de herramientas que pueden llegar a ser útiles para la creación de un *cluster* de *fail-over*, pero en su mayoría, es necesario el uso de más de uno de los proyectos existentes para lograr una solución tecnológica completa, estos proyectos se pueden dividir en Proyectos de propósito general y Proyectos de propósito específico y también los enfocados a máquinas virtuales.

C.1. Proyectos de propósito general

Los proyectos de propósito general son aquellos proyectos diseñados para brindar alta disponibilidad y/o balanceo de carga sin importar el tipo de nodo al que le sea aplicado.

A. *Keepalived*: Es un software que realiza una implementación del protocolo VRRP [1]. El protocolo está diseñado (según el RFC3768) para aportar redundancia a nodos de una determinada red mediante la abstracción de una IP virtual, que representará a más de un nodo [2], el objetivo principal de *Keepalived* es añadir una utilidad fuerte y robusta de recuperación ante fallos en el proyecto de Linux Virtual Server [3], está escrito en lenguaje de programación C con verificaciones en múltiples niveles TCP/IP, además *“implementa un marco de trabajo basado en tres familias de verificaciones: Capa3, Capa4 y Capa5/7. Este marco de trabajo le permite al demonio verificar el estado de un conjunto de servidores LVS. Cuando uno de los servidores del conjunto LVS está caído, keepalived le informa al Kernel de Linux por medio de una llamada setsockopt para que remueva la entrada de este servidor de la topología LVS”* [4]. Igualmente cuando el servicio en ese servidor vuelve a estar disponible vuelve a añadir la entrada y este vuelve a estar disponible para dar servicio al exterior. Además *Keepalived* implementa una pila independiente VRRPv2 [1] para manejar el *failover*. [4] *Keepalived* está licenciado bajo la *GNU General Public License (GPL)* [1].

B. *Linux-HA*: Es una solución *software* de alta disponibilidad para servicios críticos que permite que éstos se encuentren disponibles en ejecución de forma ininterrumpida a pesar de los fallos o problemas que puedan surgir durante su funcionamiento, Inicialmente este proyecto de conoció con el nombre *Heartbeat*, con el se hacía referencia al conjunto de herramientas y utilidades de alta disponibilidad con una arquitectura estructurada en capas, formado fundamentalmente por el componente *Heartbeat* en la capa de mensajes o comunicación, un administrador de recursos locales (*Local Resource Manager* o *LRM*) y un administrador de recursos del *clúster* (*Cluster Resource Manager* o *CRM*) en la capa de asignación de recursos, y un agente de recursos (*Resource Agent*) en la capa de recursos, entre otros componente.

En la actualidad, concretamente *“a partir de la versión 2.1.4, se les referencia en forma grupal como Linux-HA (Linux High Availability), mientras que con el nombre Heartbeat se destino para denominar exclusivamente la capa de comunicación o mensajes entre los nodos que forman el clúster. El resto de componentes son ahora proyectos independientes, igualmente necesarios para establecer un cluster de alta disponibilidad”* [5].



Figura C.1 Arquitectura del proyecto Linux-HA.

En la Figura 2.6 se observa los diferentes elementos que hacen parte del proyecto Linux-HA, cuyos elementos son:

- **Pacemaker:** Funciona como parte central del proyecto *Linux-HA* por encima de la capa de comunicación, *Pacemaker* es el módulo *CRM* (gestor de los recursos del *clúster*) y logra la máxima disponibilidad de los servicios del *clúster* mediante la detección y la recuperación de nodo y las fallas de nivel de servicio. Esto se logra mediante la utilización de las capacidades de mensajería y composición proporcionados porque puede hacer uso de *Heartbeat* u *OpenAIS*, puede administrar a grupos de prácticamente cualquier tamaño y viene con un modelo de dependencia de gran alcance para modelar con precisión el entorno [6], en caso de un fallo, los administradores de recursos como el *Pacemaker* de forma automática inician la recuperación y se aseguran de que la solicitud está disponible en una de las máquinas *remaning* en el *clúster*, *Pacemaker* esta registrados bajo la GPLv2 + (versión 2 o posterior de la GPL) y sus cabeceras y las bibliotecas se encuentran bajo la menos restrictiva LGPLv2 + (versión 2 o posterior de la LGPL). La excepción a esto es un pequeño número de archivos necesarios para crear un *plugin OpenAIS / corosync*, los cuales están bajo licencia BSD [6].
- **Heartbeat:** Funciona como capa que comunica los nodos del *clúster* [7], siendo un demonio que proporciona la infraestructura de *clúster* (comunicación y pertenencia) servicios a sus clientes. Esto permite a los clientes saber acerca de la presencia (o la desaparición) de los procesos inter pares de las otras máquinas e intercambiar mensajes fácilmente con ellos permitiendo el intercambio de mensajes para determinar la presencia o la ausencia de pares de procesos entre los nodos [5]. Se encuentra licenciado con licencia GPL.
- **OpenAIS:** OpenAIS Cluster Framework es una implementación *open source* de las *Application Interface Specification (AIS)*. Un conjunto de especificaciones para estandarizar el desarrollo de servicios e interfaces para la alta disponibilidad, desarrolladas por el *Service Availability Forum*. Actualmente, el proyecto implementa las API para mejorar la disponibilidad mediante la reducción de tiempo medio de reparación [8], “la especificación de Interfaz de Aplicación (AIS) es una API de software y políticas que se usan para desarrollar aplicaciones que mantienen el servicio durante fallos. La API consiste en el entorno de trabajo de administración de disponibilidad (AMF) que proporciona recuperación de fallos de

aplicaciones, unirse a agrupamiento (CLM), puntos de comprobación (CKPT), mensajería (MSG) y candados distribuidos (DLOCK)” [9] [10].

“Los principales beneficios de una solución de Cluster HA basado en las normas AIS son la mejora en portabilidad e integración, permite sistemas más escalables, la reducción de costes y reutilización de componentes” [10].

“Esta estandarización puede ser muy beneficiosa no solo para los componentes principales del software o middleware de clustering, si no por el hecho de que el cluster sea capaz de monitorizar un mayor número de servicios y recursos con un API unificada “[10].

El proyecto OpenAIS implementa actualmente los componentes de infraestructura y membresía. Y es utilizado en soluciones completas de *clustering* como *Pacemaker* o *Red Hat Cluster*”[10].

- **Resource Agent:** Agentes de recurso, un extenso conjunto de scripts para la gestión y acceso a los diferentes recursos y servicios que son ofrecidos y mantenidos por el clúster. Los Agentes de recursos han sido manejados como un sub-proyecto de Linux-HA desde su versión 1.0, que coincidió con la liberación del *Heartbeat* 2.99 [11].
 - **Cluster Glue:** El componente *Shoot The Other Node In The Head (STONITH)* ha sido renombrado como *ClusterGlue*, es un conjunto de bibliotecas, herramientas y utilidades adecuadas para la capa de mensajes [12], es un elemento imprescindible en la arquitectura de alta disponibilidad de *Linux-HA* ya que permite la interoperabilidad entre sus diferentes elementos [5].
- C. **Linux Virtual Server (LVS):** Es una solución para gestionar balanceo de carga en sistemas Linux. 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 proporcione buena escalabilidad, confiabilidad y robustez usando tecnología *clustering*, mediante su uso se puede conformar un grupo de servidores reales que de forma transparente para el usuario actúa como si fuera un único y gran servidor [15]. “A este servidor homogéneo se le llama “Virtual Server”, los servidores reales son gestionados por un Director (también denominado *load balancer* o *balanceador de carga*); la parte esencial del director es *IPVS (IP Virtual Server)* que implementa una capa de transporte dentro del Kernel de Linux para distribuir las peticiones de los clientes entre los servidores reales” [4]. *IPVS* es un balanceador de carga que actúa en la capa 4 del modelo OSI ya que usa protocolos como *UDP, TCP Y SCTP* para redistribuir los paquetes a través de la red (Capa de transporte)[13]. La principal idea es proveer de un mecanismo de migración de sockets que se basa en utilizar una máquina servidora a la que se dirigen las peticiones de los usuarios clientes. La interfaz pública (en Internet) de esta máquina normalmente tiene asociada una dirección conocida como *VIP*. La función de esta máquina es direccionar dichas peticiones a otros servidores reales mediante varias técnicas, de este modo los usuarios clientes ven un único servidor. No obstante éste opera con varias máquinas para conceder un servicio único al exterior [14], es actualmente uno de los proyectos más utilizados. [15]
- D. **Corosync Cluster Engine:** Es un proyecto *open source* bajo la licencia *BSD*, derivado del proyecto *Open-AIS*. El objetivo principal del proyecto es desarrollar una solución de *cluster*

completa, certificada por la OSI (*Open Source Initiative*), con soporte para Linux, Solaris, BSD y MacOSX, el proyecto provee cuatro interfaces de programación C. El software se compone de un ejecutable binario que utiliza un modelo de comunicación cliente-servidor, entre las bibliotecas y los motores de servicio, módulos de carga dinámica, llamados motores de servicio, se cargan en el *Corosync Cluster Engine* y utiliza los servicios proporcionados por la API interna del *Corosync Service Engine* [16].

- E. **Ultramonkey:** es un proyecto destinado a crear una red de servicios de alta disponibilidad con balanceo de carga, permite al sistema GNU/Linux proporcionar una solución flexible que puede ser adaptada a una amplia gama de necesidades, desde pequeños *clusters* con dos nodos hasta enormes sistemas sirviendo miles de conexiones por segundo [14] [17].
- F. **Kimberlite:** Es considerado un proyecto único ya que es una completa infraestructura de alta disponibilidad de código abierto que incluye incluso garantía sobre la integridad de los datos. Tiene licencia GPL y es desarrollado por *Mission Critical Linux* [18], actualmente es un proyecto abandonado de muy baja actividad [19].
- G. **LifeKeeper:** Es una alternativa moderna y flexible comparada con los productos que ofrecen soluciones de alta disponibilidad, destaca por su funcionalidad de *failover* (re direccionamiento dinámico en caso de fallos), gran escalabilidad y recuperación automática, así como por su transparencia para los usuarios, su interfaz gráfica basada en java simplifica la instalación, manejo y mantenimiento de *clusters LifeKeeper* [20] además dispone de mecanismos para mantener la integridad de los datos. Una numerosa colección de kits de recuperación para los distintos servicios y/o aplicaciones permiten instalar *LifeKeeper* en cuestión de horas. Es un producto de la empresa *SteelEye* [19]. Como único requisito, el hardware ha de estar equipado con procesadores basados en Intel. *SteelEye* ha firmado alianzas estratégicas con HP, Intel e IBM, cuyo resultado es una serie de soluciones integrales con componentes de hardware certificados [20].
- H. **HP Serviceguard:** Es un software especializado para ofrecer alta disponibilidad a las aplicaciones más críticas ante fallos tanto hardware como software. *HP Serviceguard* monitoriza el estado de cada nodo y responde rápidamente en caso de fallo permitiendo minimizar el tiempo durante el cual no se está ofreciendo el servicio. Está disponible tanto para Linux como para HP-UX [19].
- I. **Open HA Solaris Cluster:** Es el código base de código abierto de *Solaris Clúster* que es una solución de *cluster* de alta disponibilidad (HA) de *Sun Microsystems, Inc.* [21]. Además de tener el código base abierto, Open HA Cluster proporciona herramientas de construcción necesarias para desarrollar, construir y usar el código. proporciona la infraestructura para comunicación y compartición información relacionada, y proporciona una plataforma para la colaboración de la comunidad para la adición de nuevas características de vanguardia a la base de código [21].
- J. **Veritas Cluster Server (VCS):** Es un producto de la empresa Symantec, una solución *clustering* de alta disponibilidad para la industria, es ideal para reducir el tiempo de inactividad planeado y no planeado, al facilitar la consolidación del servidor y administrar eficazmente una amplia gama de aplicaciones en entornos heterogéneos. Debido a que admite hasta 32 clústeres de nodo, Veritas Cluster Server ofrece el poder y la flexibilidad para proteger todo, desde una única instancia de base de datos crítica hasta grandes *clústeres* multi-aplicaciones distribuidos

geográficamente [22] las principales funciones del VCS son: Automatizar la conmutación por error de las aplicaciones dentro de un *datacenter* o en *datacenters* remotos, permitir a los administradores supervisar, administrar y registrar varios clústeres de Veritas en diferentes plataformas desde una sola consola basada en Web, admitir plataformas de sistemas operativos Solaris, HP-UX, AIX, Linux, Windows y VMware con soluciones listas para usar para la mayoría de los proveedores de bases de datos, aplicaciones y almacenamiento y proporciona pruebas de recuperación después de un desastre, sin afectar al entorno principal [23]

- K. Red Hat Cluster Suite:** Es un conjunto integrado de componentes de software que puede ser implementado en una amplia variedad de configuraciones para cubrir las necesidades de rendimiento, alta disponibilidad, balanceo de carga, escalabilidad, compartición de archivos y economía de recursos [24], utiliza como *software* del *cluster* OpenAIS, CMAN, RGManger, Fenced, soporta un máximo de 16 nodos, para almacenamiento compartido soporta SANs, iSCSI y FC, y el uso de sistemas de ficheros compartidos como GFS [10].
- L. Wackamole:** Es una herramienta para hacer un clúster de alta disponibilidad. Gestiona un grupo de direcciones IP virtuales que deben estar disponibles para el mundo exterior en todo momento, y se asegura de que una máquina dentro del grupo escuche cada dirección IP virtual que gestiona Wackamole. Si se descubre que determinadas máquinas dentro de la agrupación no están funcionando, este asegura casi de inmediato que otra de las máquinas adquiera la IP. Wackamole también trabaja hacia el logro de una distribución equilibrada de direcciones IP entre las máquinas del grupo que gestiona [25].
- M. Piranha:** Es una solución para balanceo de carga de Red Hat Enterprise Linux. Ofrece la posibilidad de equilibrar la carga de las solicitudes entrantes de la red IP a través de un grupo de servidores. Se basa en la tecnología de LVS, con importantes mejoras de Red Hat. Piranha es estrictamente una aplicación de software de HA que es el paquete de grupo y la herramienta GUI de administración para el clúster [26].
- N. HAProxy:** Es una solución libre, muy rápida y confiable ofrecida para alta disponibilidad, balanceo de Carga y proxy para aplicaciones basadas en TCP y HTTP. Es especialmente adecuado para los sitios web que se arrastran bajo cargas muy altas, mientras que la persistencia necesitada o procesamiento Capa o Layer7 [27].
- O. PEN Load Balancer:** Es un balanceador de carga para protocolos “simples” basados en TCP como HTTP o SMTP. Permite a varios servidores que aparezcan como una al exterior, detecta automáticamente los servidores que están abajo y distribuye clientes entre los servidores disponibles. Esto le da una alta disponibilidad y un rendimiento escalable [28].
- P. Zen load balancer:** Es una distribución basada en Debian para crear una aplicación para el balanceo de carga para los servicios TCP, UDP y las comunicaciones de datos en línea. Zen LB ofrece una administración avanzada para las interfaces de red, configuración ilimitada de nodos, comprobación avanzada para las granjas TCP, verificación y monitorización para los servidores *back-end*, permite la creación de un *cluster* zen activo/pasivo de alta disponibilidad, gráficos de control, copias de seguridad de configuración, soporte VLAN, configuración en tiempo real de replicación de sincronización de los nodos del clúster, equilibrio de carga HTTP

específica, la persistencia de clientes, de contenedor de SSL, y entre otras funcionalidades [29].

- Q. Vyatta:** El Vyatta Core (VC) es un sistema operativo para red de código abierto galardonado que proporciona soluciones avanzadas de enrutamiento IPv4 e IPv6, cortafuegos *stateful*, IPSec y SSL OpenVPN, prevención de intrusiones, y mucho más. En el sistema Vyatta, el *clustering* puede ser utilizado como un mecanismo de conmutación para proporcionar alta disponibilidad (HA) para servicios de misión crítica. El *cluster* monitorea los nodos que prestan servicios designados (por ejemplo, un túnel IPsec VPN) a una dirección específica. Si el sistema detecta que el nodo ha fallado, o que el enlace con el nodo ha fallado, el sistema migra tanto los servicios y las direcciones IP a un nodo de copia de seguridad, la conmutación por error se admite actualmente entre dos nodos: un nodo primario y un nodo secundario [30].
- R. POUND:** Es un proxy inverso, balanceador de carga y HTTPS *front-end* para el servidor web, Pound fue desarrollado para permitir la distribución de la carga entre varios servidores web y para permitir una cómoda envoltura de SSL aquellos que no lo ofrecen de forma nativa. Pound como proxy inverso pasa las peticiones de los navegadores de los clientes a uno o más servidores *back-end* y como balanceador de carga reparte las solicitudes de los exploradores de los clientes entre varios servidores *back-end*, mientras mantiene la información de sesión; como envoltorio SSL POUND descifra las peticiones HTTPS de los navegadores de los clientes y los pasa como HTTP sin formato a los servidores *back-end*, funciona además como servidor de conmutación por error, si falla el servidor *back-end*, toma nota de los hechos y deja de pasar solicitudes hasta que este se recupere, también funciona como director de solicitud distribuyendo las mismas entre los servidores de acuerdo con la dirección URL solicitada [31].

C.2. Proyectos De Propósito Específico

Es posible hallar múltiples proyectos que buscan alta disponibilidad o alto desempeño para aplicaciones específicas pero cuya arquitectura o funcionalidad resulta inútil o inutilizable para otras aplicaciones.

- A. MySQL Cluster:** Es una versión de alta disponibilidad, alta redundancia de MySQL adaptada para el entorno de computación distribuida, permite *clustering* de bases de datos en memoria en un entorno de no compartición. La arquitectura de no compartición permite que el sistema funcione con hardware barato, y sin ningún requerimiento especial de hardware o software. Tampoco tienen ningún punto único de fallo porque cada componente tiene su propia memoria y disco. Usa el motor de almacenamiento *NDB Cluster* para permitir la ejecución de varios servidores MySQL en un *cluster*. Este motor de almacenamiento está disponible en las distribuciones binarias de MySQL y en los RPMs compatibles con las distribuciones Linux más modernas, Los sistemas operativos en que *MySQL Cluster* está disponible son Linux, Mac OS X, Windows y Solaris. (Algunos usuarios han reportado éxito al ejecutar *MySQL Cluster* en *FreeBSD*, aunque no está soportada oficialmente por MySQL AB.), cabe aclarar que para la comunicación entre los nodos del *cluster* no está cifrada, la única forma de proteger transmisiones dentro de un *MySQL Cluster* es ejecutar su *cluster* en una red protegida. Si se trata de usar *MySQL Cluster* para aplicaciones *Web*, el *cluster* debe residir detrás de un firewall y no en su DMZ o en otro lugar [32] [33].

Está disponible como código abierto y también como edición comercial, la *Community (OSS) Edition* está disponible en software libre / código abierto GNU *General Public License* [34].

- B. Oracle Real Application Clusters (Oracle RAC):** *Real Application Clusters es un software* que permite utilizar un *cluster* de servidores ejecutando múltiples instancias sobre una misma base de datos. Los archivos de base de datos quedan almacenados en discos de forma físicos o lógica, conectados a cada nodo, de modo tal que todas las instancias activas pueden leerlos o escribirlos.

El *software* de RAC maneja el acceso a los datos, de modo tal que los cambios en los datos son coordinados entre las instancias y cada instancia ve imágenes consistentes de la base. El *interconnect del cluster* permite que las instancias se pasen entre ellas información de coordinación e imágenes de los datos.

Esta arquitectura permite que los usuarios y aplicaciones se beneficien de la potencia de procesamiento de múltiples máquinas. La arquitectura RAC también ofrece redundancia; por ejemplo, en el caso de que un nodo quede inutilizado, la aplicación continuará accediendo a los datos vía el resto de las instancias disponibles [35].

A pesar de ser código cerrado se menciona en este apartado ya que la Universidad del Cauca cuenta con bases de datos Oracle y por esto es importante la presentación de la solución para ese ítem en particular.

C.3. Proyectos enfocados en Maquinas Virtuales

Además de los proyectos anteriormente listados los cuales buscan la alta disponibilidad y/o balanceo de carga en nodos “reales”¹ existen soluciones específicas para el uso de maquinas virtuales, ya que las mismas cuentan con características intrínsecas que las hacen diferentes a aquellos nodos establecidos en una maquina real, para esto también existen soluciones que buscan brindar alta disponibilidad y balanceo de carga en estos entornos de tipo especial.

- A. Proxmox:** Es una plataforma de virtualización de código libre desarrollado y mantenido por Proxmox Server Solutions GmbH y financiado por la Internet Foundation Austria. Está dirigido a usuarios de la empresa y diseñado para ser instalado en el hardware de grado empresarial. La distribución de VE Proxmox combina dos tecnologías de máquina virtual, KVM y OpenVZ , así como una interfaz web para gestionar todo [36].

La última versión estable de Proxmox VE está basada en un Debian Lenny de 64 bits. Usa una partición del tipo LVM2 para el disco duro e incluye el Kernel de Proxmox VE, en el cuál están incluidos parches para soportar los sistemas OpenVZ y KVM.

Tiene un sistema de copias de seguridad integrado y la posibilidad de migrar las máquinas virtuales entre distintas máquinas físicas sin la necesidad de pararlas. Esta facilidad y rapidez en la instalación es debida a que Proxmox VE proporciona aplicaciones virtuales prefabricadas y una interfaz, vía web, que permite realizar la gestión de una manera muy sencilla [37].

Proxmox cuenta con licencia GNU *General Public License* [38].

¹ Se refiere a este tipo de nodos como “reales” ya que aunque la mayoría de los que atañen a este proyecto se encuentran virtualizados, en la práctica el nodo para su funcionamiento desconoce esa información.

BIBLIOGRAFÍA

- [1] Keepalived Community, "Keepalived for Linux - Linux High Availability," [keepalived.org](http://www.keepalived.org) .[En línea]. Disponible en: <http://www.keepalived.org>. [Accedido: Dic. 7, 2011].
- [2] A. Borrero, "Firewall en cluster de alta disponibilidad," Proyecto Integrado, IES Gonzalo Nazareno, Sevilla, España, 2011. [En línea]. Disponible en: http://informatica.gonzalonazareno.org/proyectos-ASI/2010-11/proyecto_integrado_arturo_borrero_pdf.pdf. [Accedido: Ene. 10, 2012].
- [3] A. Simon, "Keepalived : haute disponibilité et répartition de charge enfin libérées," [keepalived.org](http://www.keepalived.org), .[En línea]. <http://www.keepalived.org/pdf/asimon-jres-paper.pdf>. [Accedido: Ene. 22, 2012].
- [4] Proyecto Fedora LATAM/Latinoamerica, "Keepalived," proyectofedora.org .[En línea]. Disponible en: <http://proyectofedora.org/wiki/Keepalived>. [Accedido: Dic. 15, 2011].
- [5] E. Villar, "Virtualización de servidores de telefonía IP en GNU/Linux," Trabajo de Titulación, Universidad de Almería, Almería, España, 2010. [En línea]. Disponible en: http://www.adminso.es/images/d/dc/PFC_eugenio.pdf. [Accedido: Ene. 10, 2012].
- [6] Pacemaker Project, "License," clusterlabs.org, .[En línea]. Disponible en: <http://clusterlabs.org/wiki/License>. [Accedido: Ene. 28, 2012].
- [7] Linux-HA "Heartbeat," [linux-ha.org](http://www.linux-ha.org) .[En línea]. Disponible en: <http://www.linux-ha.org/wiki/Heartbeat>. [Accedido: Dic. 7, 2011].
- [8] Freecode, "OpenAIS," freecode.com .[En línea]. Disponible en: <http://freecode.com/projects/openais>. [Accedido: Ene. 28, 2012].
- [9] Imperios, "Cómo instalar openais en Debian, Ubuntu y similares," [imperiosweb.com](http://www.imperiosweb.com), .[En línea]. Disponible en: <http://www.imperiosweb.com/howto/how-to-install/openais.html>. [Accedido: Feb., 2012].
- [10] P. Clavijo, "Soluciones Open Source de Clustering HA," [lintips.com](http://www.lintips.com), [En línea] Disponible en: <http://www.lintips.com/?q=node/120>. [Accedido: Feb., 2012].
- [11] F. Haas "Resource Agents - Linux-HA," [linux-ha.org](http://www.linux-ha.org), Dic. 2009 .[En línea] Disponible en: http://www.linux-ha.org/wiki/Resource_Agents. [Accedido: Feb. 2, 2012].
- [12] F. Haas "Cluster Glue - Linux-HA," [linux-ha.org](http://www.linux-ha.org), Dic. 2009, .[En línea]. Disponible en: http://www.linux-ha.org/wiki/Cluster_Glue. [Accedido: Ene. 20, 2012].
- [13] Rfpp, "Linux Virtual Server: Clusters en Linux - Modulado 3.2," modulado.wordpress.com, Dic. 28, 2008 .[En línea]. Disponible en: <http://modulado.wordpress.com/2008/12/07/linux-virtual-server-clusters-en-linux/>. [Accedido: Ene. 28, 2012].

- [14] J. Cáceres, C. Medina, "IMPLEMENTACIÓN DE UN SERVIDOR WEB APACHE SOBRE UN CLUSTER EN LINUX," Seminario de Título, Universidad Católica del Maule, Talca, Chile, 2007
- [15] "The Linux Virtual Server Project - Linux Server Cluster for Load Balancing," *linuxvirtualserver.org*, .[En línea]. Disponible en: <http://www.linuxvirtualserver.org/> [Accedido: Ene. 28, 2012].
- [16] "Corosync (project) - Everything on Corosync (project) (information, latest news, articles,...), " *spiritus-temporis.com* .[En línea]. Disponible en: <http://www.spiritus-temporis.com/corosync-project/>. [Accedido: Ene. 28, 2012].
- [17] B. Quiroz, "Ultra Monkey," *lugmen.org.ar*, .[En línea]. Disponible en: <http://www.lugmen.org.ar/~boris/linux/Ultra.Monkey.pdf>. [Accedido: Ene. 28, 2012].
- [18] T. Burke, "Kimberlite Cluster Technology," *blu.org*, .[En línea]. Disponible en: <http://www.blu.org/cgi-bin/calendar/2000-oct>. [Accedido: Ene. 28, 2012].
- [19] F. Méndez, "Diseño e implementación de un sistema VoIP de alta disponibilidad y alto rendimiento," Trabajo de Titulación, Universidad de Almería, Almería, España, 2009.
- [20] SUSE, "Alta disponibilidad para SUSE Enterprise Server - SteelEye LifeKeeper for Linux," *suse.de*, .[En línea]. Disponible en: http://www.suse.de/cgi-bin/print_page_www.pl?NPSPath=/webredesign/htdocs/es/business/products/openexchange/steeleye.html. [Accedido: Oct. 5, 2011].
- [21] "Open HA Cluster (Community Group ha-clusters.ohac) – Xwiki," *hub.opensolaris.org*, Oct. 26, 2009 .[En línea]. Disponible en: <http://hub.opensolaris.org/bin/view/Community+Group+ha-clusters/ohac>. [Accedido: Feb. 6, 2012].
- [22] Symantec Corporation, "Hoja de datos: Administración de servidores: Veritas™ Cluster Server: Arquitecturas de recuperación tras desastres con soporte," *nextvision.com*, .[En línea]. Disponible en: http://www.nextvision.com/uploads/com_solucion/Symantec_Cluster_Server.pdf. [Accedido: Feb. 6, 2012].
- [23] Value Added IT Solutions S.A., VAITS, "Veritas Cluster Server," *vaits.com*, 2008 .[En línea]. Disponible en: http://www.vaits.com/sccs/secciones.php?id_titulo=42&PHPSESSID=c0e857e0151e78f30b1d876b22fd4126. [Accedido: Feb. 6, 2012].
- [24] Red Hat, Inc, "1.2. Red Hat Cluster Suite Introduction," *docs.redhat.com* .[En línea]. Disponible en: <http://docs.redhat.com/docs/es->

- ES/Red_Hat_Enterprise_Linux/5/html/Cluster_Suite_Overview/s1-rhcs-intro-CSO.html.
[Accedido: Feb. 6, 2012].
- [25] Center for Networking and Distributed Systems of Johns Hopkins University, “Wackamole,” *backhand.org*, .[En línea]. Disponible en: <http://www.backhand.org/wackamole/>. [Accedido: Feb. 8, 2012].
- [26] “Piranha (Homepage),” *sources.redhat.com* .[En línea]. Disponible en: <http://sources.redhat.com/piranha/>. [Accedido: Sep. 15, 2011].
- [27] HAProxy Community, “The Reliable, High Performance TCP/HTTP Load Balancer,” *haproxy.1wt.eu* .[En línea]. Disponible en: <http://haproxy.1wt.eu/>. [Accedido: Sep. 10, 2011].
- [28] U. Eriksson, “Pen,” *siag.nu* .[En línea]. Disponible en: <http://siag.nu/pen/>. [Accedido: Feb., 2012].
- [29] SofIntel, “ZEN Load Balancer,” *zenloadbalancer.com* .[En línea]. Disponible en: <http://www.zenloadbalancer.com/web/>. [Accedido: Feb. 10, 2012].
- [30] Vyatta, “High Availability REFERENCE GUIDE,” *vyatta.com* .[En línea]. Disponible en: http://www.vyatta.com/downloads/documentation/VC6.3/Vyatta_HA_R6.3_v01.pdf. [Accedido: Dic. 15, 2011].
- [31] APSIS IT Security, “Pound,” *apsis.ch* .[En línea]. Disponible en: <http://www.apsis.ch/pound>. [Accedido: Feb. 20, 2012].
- [32] Oracle and/or its affiliates, “Capítulo 16. MySQL Cluster,” [En línea] Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/ndbcluster.html>. [Accedido: Feb. 6, 2012].
- [33] J. A. Cobas, “MySQL Cluster,” *ecured.cu*. [En línea]. Disponible en: http://www.ecured.cu/index.php/MySQL_Cluster. [Accedido: Feb. 7, 2012].
- [34] Oracle and/or its affiliates “Download MySQL Cluster,” .[En línea]. Disponible en: <http://www.mysql.com/downloads/cluster/>. [Accedido: Feb. 10, 2012].
- [35] F. M. García, “Qué es Oracle Real Application Clusters?,” in *Blog Oracle RAC Notes*, Feb. 20, 2008. [En línea]. Disponible en: <http://oracleracnotes.wordpress.com/2008/02/20/que-es-oracle-real-application-clusters/>. [Accedido: Ago. 7, 2011].
- [36] W. Sanders, “Review: Proxmox Virtual Environment,” *montanalinux.org* .[En línea]. Disponible en: <http://www.montanalinux.org/proxmox-ve-review.html>. [Accedido: Ene. 28, 2012].
- [37] F. Gómez, “Automatización de instalación y configuración de aplicaciones peer-to-peer en un entorno virtualizado basado en modelnet y proxmox,” Proyecto Fin de Carrera,

Universidad Carlos III de Madrid, Leganés, España, 2010. [En línea]. Disponible en: http://e-archivo.uc3m.es/bitstream/10016/11169/1/PFC_Felix_Gomez_Fernandez.pdf. [Accedido: Ene. 28, 2012].

- [38] Proxmox Community, "Open Source," *proxmox.com*. [En línea]. Disponible en: http://pve.proxmox.com/wiki/Open_Source. [Accedido: Ene. 28, 2012].