ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

"Implementación de un sistema de VoIP de alta disponibilidad basado en Asterisk y Heartbeat"

Mapy Asunción Castillo Palma [(1)], Ana Sofía Rocha Pereira [(2)],
Ing. Gabriel Astudillo, Profesor de la Materia, ESPOL [(3)]
Facultad de Ingeniería Eléctrica y Computación [(1) (2) (3)]
Escuela Superior Politécnica del Litoral (ESPOL) [(1) (2) (3)]
Campus Gustavo Galindo, Km 30.5 vía Perimetral
Apartado 09-01-5863. Guayaquil-Ecuador
mapacast@espol.edu.ec [(1)], ansorope@gmail.com [(2)], gastudillo@espol.edu.ec

Resumen

En este documento se presentaran soluciones de alta disponibilidad basadas en software destinadas a ofrecer alta disponibilidad en servicios y alta disponibilidad en datos en un sistema de VoIP. Se mostrará la instalación de Asterisk con Alta Disponibilidad utilizando herramientas de código abierto con capacidades de recuperación ante desastres. Se analizará la solución, su desarrollo y los resultados obtenidos durante las diferentes pruebas realizadas antes y después de su implementación, se analizará la solución al problema común del tiempo de caída de un sistema de voz y su alcance. Se explicarán las características del sistema: su arquitectura, equipos a usar, archivos a configurar. Al final, luego de obtener los resultados de la implementación del sistema y los resultados de las pruebas se expondrán las conclusiones y se realizaran recomendaciones para la mejora del sistema.

Palabras Claves: *Asterisk, Heartbeat, DRBD, Alta Disponibilidad, Clúster.*

Abstract

This paper will present high-availability solutions based on software designed to provide high availability services and high availability data in a VoIP system. There will be showed the installation of Asterisk with high availability using open source tools with fail-over capabilities. The solution is analyzed, its development and the results obtained during the different tests we made before and after its implementation, we will analyze the solution to the common problem of the fall time of a voice system and its scope. The characteristics of the system will be explained: its architecture, equipment used, files to configure. Finally, after obtaining the results of the implementation of the system and the results of the tests we will expose the conclusions and recommendations for the improvement of the system.

1. Introducción

El servicio de telefonía implantado está dotado de alta disponibilidad en diversos aspectos. Uno de ellos está relacionado con los fallos tipo hardware como puede ser el fallo de la fuente de alimentación de uno de los nodos del sistema, corte del suministro eléctrico, etc. Ante estas situaciones el servicio de telefonía no se verá afectado.

Algunos de los clientes más exigentes que requieren comunicación a tiempo completo, como un Centro de Llamadas o la industria de la Banca, necesitan contar con una estructura física redundante que les provea el menor tiempo de caída, estos usuarios están muy lejos de aceptar una solución que no tenga mecanismos a prueba de fallos o alta disponibilidad. Para esto se necesita una inversión muy alta que muchas veces no está al alcance del presupuesto de una pyme que empieza en el mundo de los negocios.

Eso en la parte física de nuestra red en cuestión, por otro lado, en la estructura lógica de la red, se pueden implementar diversos tipos de soluciones combinando el hardware y software adecuado para satisfacer estas necesidades.

En el caso de una implementación de telefonía IP, usando equipos Linux y con Asterisk como "administrador" de nuestra centralita IP, las soluciones son varias entre ellas: DRBD, Heartbeat, LVS, etc. Pero estos proyectos proveen redundancia en puntos específicos del sistema.

Eso en la parte física de nuestra red en cuestión, por otro lado, en la estructura lógica de la red, se pueden implementar diversos tipos de soluciones combinando el hardware y software adecuado para satisfacer estas necesidades.

2. Metodología

Las soluciones de alta disponibilidad basadas en software que van a ser aquí comentadas están destinadas a ofrecer alta disponibilidad en servicios y alta disponibilidad en datos.

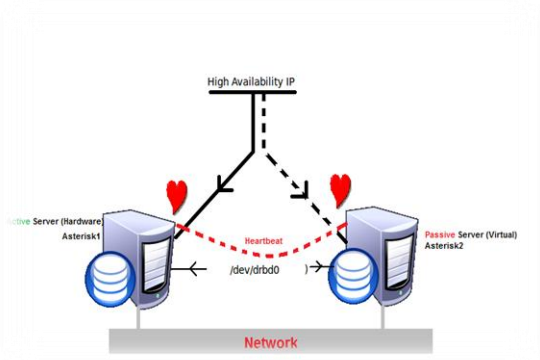

Figura 1 Esquema Alta Disponibilidad

Para cumplir con nuestros objetivos tendremos que hacer trabajar conjuntamente varias piezas, realizaremos la instalación de Asterisk sobre servidor real y un servidor virtual con sistema operativo CENTOS LINUX (Figura 1). Estos servidores deberán cumplir con mínimas especificaciones ya que uno de los logros de este proyecto es que se puede usar una infraestructura simple en cuanto a equipos se refiere.

Adicionalmente dotaremos a los servidores con Heartbeat y Distributed Replicated Block Device para poder establecer canales de comunicación con alta disponibilidad y transparencia para el usuario.

3. Asterisk, Heartbeat y DRBD.

Asterisk es una aplicación de software libre (bajo licencia GPL) de una central telefónica PBX (Private Branch Exchange y Private Automatic Branch Exchange para PABX), este software es una herramienta que nos ayuda a conectar directamente cualquier central telefónica a la red pública de teléfonos por medio de las líneas troncales actúa en Linux, BSD, Windows (emulado) y OS X proporcionando todas las funciones y características que se desarrollan en una PBX; también hace posible la utilización de VoIP en tres protocolos y puede interoperar con equipos de telefonía estándar usando un hardware relativamente sin costo.

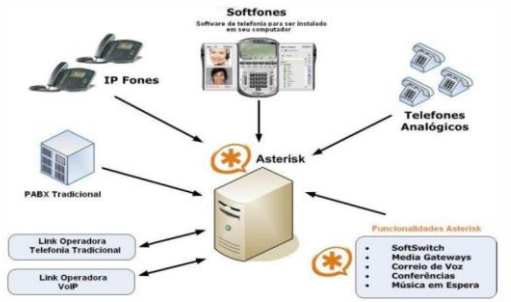

Figura 2 Asterisk

Heartbeat es un proyecto Open Source fundado por Alan Robertson en 1999. El objetivo de este proyecto es proveer software de clustering de alta disponibilidad (HA) para Linux y otras plataformas. Heartbeat es un software que ofrece alta disponibilidad a determinados recursos mediante la creación y mantenimiento de un clúster compuesto por una serie de nodos. Este término no está muy bien definido y puede tener diferentes significados para las personas; según el diccionario informático, un clúster es la unidad de almacenamiento en el disco rígido, muchos de los usuarios Windows estamos relacionados con la perdida de clusters que puede ser

solucionado mediante la ejecución de la utilidad de desfragmentación.

Sin embargo, en un nivel más avanzado de nuestra industria, un clúster puede significar un grupo de computadoras conectadas entre sí de modo que se obtiene más "poder", como por ejemplo se puede lograr ejecutar mas instrucciones por segundo, o se puede lograr una mayor disponibilidad. Los recursos se ejecutan y se mueven entre los distintos nodos ya sea por motivos de fallo o simplemente por motivos de administración.

Las plataformas que soportan Heartbeat son las distribuciones de Linux como: SuSe, Mandriva, Debian, Ubuntu, Red Hat y Gentoo. Otras plataformas incluyen: FreeBSD, OpenBSD, Sun Solaris y Mac OS X.

La segunda versión de Heartbeat ha sido modificada para eliminar ciertas restricciones presentes en la primera versión, las cuales presentaban grandes limitaciones en su funcionamiento. Las restricciones más importantes que limitaban la funcionalidad de la primera versión de Heartbeat son las siguientes:

- Clúster con un tamaño máximo de dos nodos.
- Incapacidad de monitorizar los recursos. Heartbeat-1 no monitoriza los recursos con la finalidad de comprobar si los recursos están operando correctamente por lo que solo se supervisa el estado de funcionamiento del nodo (monitorización hardware) sin tener en cuenta el estado de ejecución de los recursos (monitorización software). De esta manera si el nodo que ofrece el servicio contesta adecuadamente al ping indicando "estoy vivo" pero el recurso, por ejemplo un servicio Web, no se está ejecutando correctamente o incluso ha parado, no se toma ninguna medida de recuperación. Para monitorizar recursos con Heartbeat-1 es necesario hacer uso de aplicaciones de monitorización externas tales como Watchdog.
- Mínima capacidad de expresar la información dependiente. En Heartbeat-1 no es posible crear grupos de recursos que compartan las mismas restricciones, grupos de nodos, etc., lo que repercuten en una pobre flexibilidad a la hora de configurar el clúster.

Distributed Replicated Block Device o DRBD es un software de replicación de dispositivos de bloque (discos duros, particiones, volúmenes, etc.) que permite formar un RAID a través de la red.

Al igual que Heartbeat, DRBD requiere definir los recursos de datos a los cuales se va a dotar de alta redundancia. Más detalladamente, un recurso en DRBD es una colección de términos que hace referencia a todos los aspectos de un dispositivo particular de almacenamiento replicado. Un recurso está formado por: nombre, unidad DRBD, configuración de disco, configuración de red.

En DRBD cualquier recurso tiene un papel, puede ser primario o secundario. Estos papeles hacen referencia a la disponibilidad del almacenamiento. Sin embargo hay otros dos términos que generan mucha confusión con estos: activo y pasivo. Estos últimos hacen referencia a la disponibilidad de una aplicación.

Usualmente en entornos de alta disponibilidad el nodo primario es también el nodo activo, pero esto no tiene porqué ser siempre así.

4. Implementación.

La central telefónica tradicional es reemplazada por un computador que puede variar según las necesidades del cliente; el tamaño de la central dependerá de la concurrencia de llamadas que vaya a tener, pero siempre el dinero gastado será inferior que si compráramos una central telefónica.

4.1. Hardware

Características técnicas mínimas recomendadas para operar Centos:

- Memoria RAM: 192 MB (Mínimo).
- Espacio en Disco Duro : 850 MB (Mínimo) - 2 GB (Recomendado)
- Procesador: Intel Pentium I/II/III/IV/Celeron, AMD K6/II/III, AMD Duron, AMD Athlon/XP/MP ya que funciona sobre las redes IP, que son más baratas que las redes de almacenamiento especiales.

Tabla 1 Caracteristicas de Servidores

Servidor#1		Servidor#2	
Procesador	Dual-Core	Procesador	Core 2 Duo
RAM	1 GB	RAM	1 GB
Disco Duro	15 GB	Disco Duro	15 GB
Tarjeta de Red	10/100 Mbps	Tarjeta de Red	10/100 Mbps

La conexión de los servidores se realizara por medio de un Switch Linksys de las siguientes características:

Tabla 2 Caracteristicas de Switch

Velocidad de transferencia de datos	100 Mbps
Protocolo de interconexión de datos	Ethernet, Fast Ethernet
Interfaces	8 x red - Ethernet 10Base-T/100Base-TX - RJ-45 hembra – 8

4.2. Software

El servidor que será utilizado como centralita telefónica tendrá los siguientes componentes instalados:

Tabla 3 Caracteristicas de Software

Sistema Operativo	Centos Linux 5.4
Software IP PBX	Asterisk Verison 1.6.0.15-1
Dahdi Linux	DAHDI LINUX 2.4.0

Librerías necesarias para que Asterisk funcione correctamente como Clúster High Availability:

- drbd
- kmod-drbd82
- OpenIPMI-libs
- heartbeat-pils
- openhpi
- heartbeat
- heartbeat-stonith

En este proyecto se ha utilizado los Softphones X-Lite y Zoiper simuladores de extensiones SIP.

5. Configuración Heartbeat y DRBD

Para ambos servidores la configuración de Heartbeat debe ser igual, se configuraran solo tres localizados en /etc/ha.d/, estos son ha.cf, authkeys y haresources.

5.1. Configuración ha.cf:

En él se establece la configuración del clúster, como por ejemplo: nodos que lo componen, que interfaz usará cada nodo para comunicarse con el clúster, puertos que usarán para la comunicación, etc.

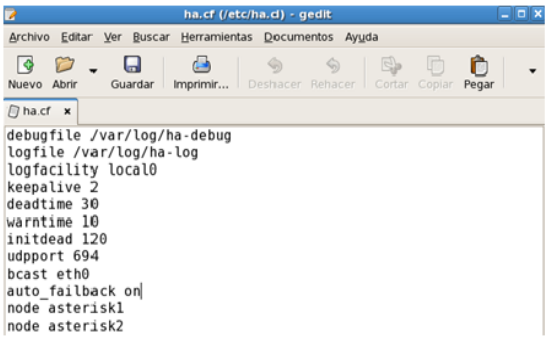

Figura 3 Configuración ha.cf

5.2 Configuración authkeys

Este archivo determina la seguridad del clúster. Para ello se hace uso de claves que deben ser iguales en todos los nodos.

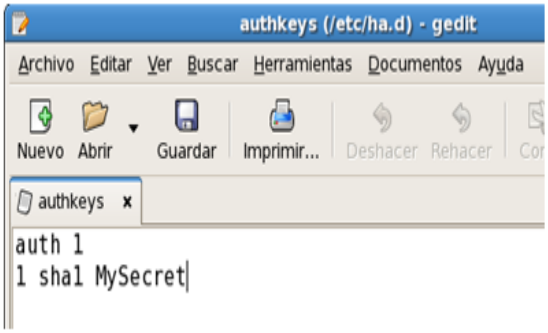

Figura 4 Configuración authkeys

5.3 Configuración haresources

En este fichero se establecen los recursos que el clúster va a gestionar.

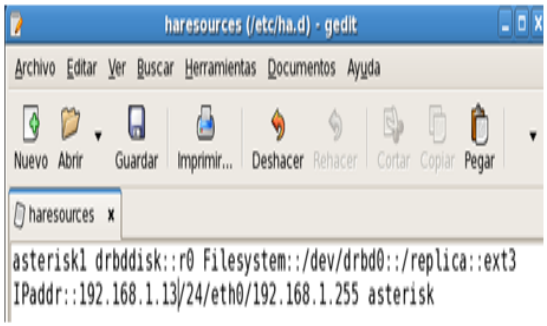

Figura 5 Configuración haresources

5.4 Configuración DRBD

Toda la configuración de DRBD se lleva a cabo en un único archivo de configuración, drbd.conf el cual podemos encontrar en /etc.:

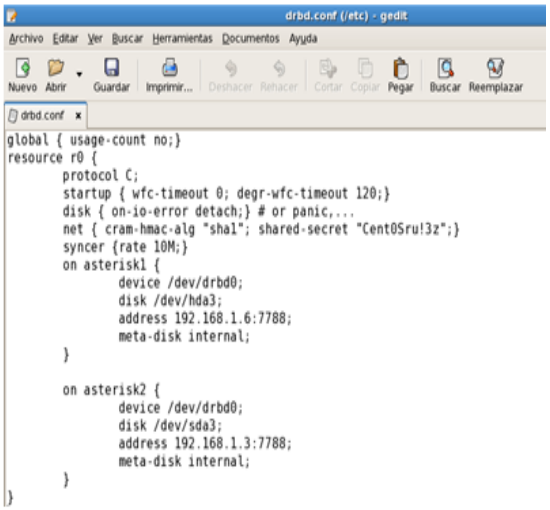

Figura 6 Configuracion drbd.conf

6. Funcionamiento y Pruebas

Los servidores están configurados en un ambiente activo/pasivo; comunicando ambos servidores mediante el envió de paquetes broadcast y unicast a través de la ethernet. Esta configuración consiste en que únicamente uno de los nodos ofrece el servicio (solo lo ofrece el nodo activo de todo el clúster).

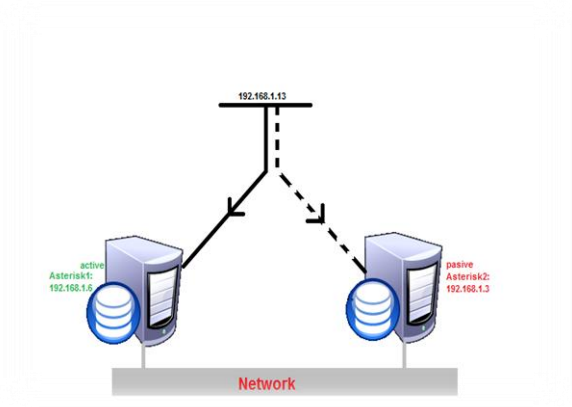

Figura 7 Funcionamiento (activo/pasivo)

Cuando el nodo activo falla, el nodo pasivo determina su caída debido que para de recibir paquetes (heartbeats) del nodo activo. Todos los recursos incluyendo direcciones IP se levantan en el nodo secundario y el servicio inicia. En cuestión de segundos todos los servicios se están ejecutando nuevamente de forma correcta en el nodo pasivo.

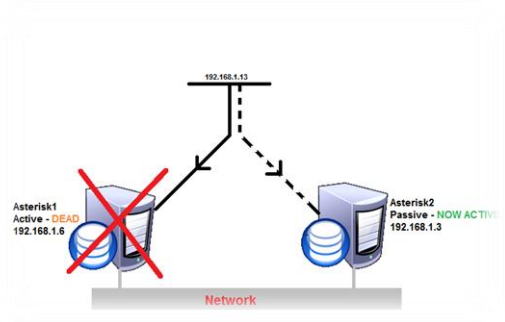

Figura 8 "Activo" deja de funcionar

Dependiendo de la configuración, el servidor caído ahora asumirá el rol de "Pasivo". Si se desea, el clúster puede ser configurado de manera que cuando el nodo ahora Pasivo vuelva a estar en línea, este retome todos los servicios y vuelva asumir el rol de "Activo".

Finalmente, se recogen una serie de pruebas que permiten verificar empíricamente el correcto funcionamiento del balanceo de las conexiones a servicios ofertados por el clúster. Estas mismas pruebas, permiten constatar que se aplica correctamente la persistencia de sesión para el servicio de asterisk.

6.1. Comprobando DRBD y Heartbeat

Arrancamos DRBD tanto en Asterisk1 como en Asterisk2, una vez iniciado comprobamos que DRBD haya iniciado bien:

```
[root@asterisk1 ~]# service drbd start
Starting DRBD resources:    [ s(r0) ].
[root@asterisk1 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.2.6 (api:88/proto:86-88)
GIT-hash: 3e69822d3bb4920a8c1bfdf7d647169eba7d2eb4 build by buildsvn@c5-i386-bui
ld, 2008-10-03 11:42:32
m:res  cs         st               ds                 p  mounted   fstype
0:r0   Connected  Primary/Secondary  UpToDate/UpToDate  C  /replica  ext3
```

Figura 9 Status DRBD (Asterisk1)

```
[root@asterisk2 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.2.6 (api:88/proto:86-88)
GIT-hash: 3e69822d3bb4920a8c1bfdf7d647169eba7d2eb4 build by buildsvn@c5-i386-bui
ld, 2008-10-03 11:42:32
m:res  cs         st               ds                 p  mounted  fstype
0:r0   Connected  Secondary/Primary  UpToDate/UpToDate  C
```

Figura 10 Status DRBD (Asterisk2)

El tiempo que dura el proceso de sincronización dependerá de la velocidad de la red, del tamaño de la partición a replicar así como de la velocidad de transferencia indicada en el archivo de configuración de DRBD.

Arrancamos Heartbeat tanto en Asterisk 1 como en Asterisk2, una vez iniciado comprobamos que Heartbeat haya iniciado bien:

```
[root@asterisk1 ~]# service heartbeat status
heartbeat OK [pid 2062 et al] is running on asterisk1 [asterisk1]...
```

Figura 11 Status Heartbeat (Asterisk1)

```
[root@asterisk2 ~]# service heartbeat status
heartbeat OK [pid 26584 et al] is running on asterisk2 [asterisk2]...
```

Figura 12 Status Heartbeat (Asterisk2)

6.2. Comprobando que se puedan realizar llamadas.

Realizamos una llamada entre las extensiones y comprobamos que las llamadas se registren en el servidor:

```
asterisk1*CLI> sip show channels
Peer           User/ANR        Call ID         Format          Hold     Las
t Message    Expiry
0 active SIP dialogs
  == Using SIP RTP CoS mark 5
    -- Executing [202@internal:1] Dial("SIP/201-00000000", "SIP/202,10") in new
stack
  == Using SIP RTP CoS mark 5
    -- Called 202
    -- SIP/202-00000001 is ringing
```

Figura 13 Sip Show Channels

6.3. Desconexión de suministro eléctrico servidor Asterisk1

Se debe desconectar el suministro de energía al servidor principal activo, lo que provocara el apagado anormal de este. (caída de sistema).

6.4. Resultados:

El servidor secundario notara la ausencia de su homologo e iniciara la secuencia de traspaso por falla, tomando el control del servicio como servidor activo.

Además el servidor principal caído presentara los errores respectivos, por lo cual se iniciaran los procedimientos de recuperación y aseguramiento de la integridad del sistema de archivos, una vez iniciado el servidor deberá retomar su rol de activo, además de informar a su secundario que debe desactivarse.

6.5. Errores:

Existen 2 posibles errores en este caso:

El servidor secundario no detecta la ausencia y no se inicia la secuencia de traspaso por falla, provocando la interrupción del sistema por no haber un servidor que gestione las peticiones y respuestas.

El servidor caído no inicia correctamente debido a una falla mayor irrecuperable (ejemplo: daño físico o lógico), por lo cual no podrá retomar su rol.

7. Conclusiones

1. En una infraestructura donde los fallos son importantes porque se deja de dar un servicio, mantenerlos corriendo en varias maquinas de forma redundante ayuda a evitar estas situaciones.
2. Heartbeat es la mejor herramienta para dar alta disponibilidad a un servicio, en este caso Asterisk; ya que siendo un software libre se adapta a las necesidades del cliente y proporciona un clúster evitando en lo más mínimo la pérdida del servicio de telefonía
3. La instalación y configuración de Heartbeat son procedimientos muy complejos pero necesarios para tener una buena estructura de clustering.
4. El uso de DRBD junto con Heartbeat optimiza en un 100% el desempeño de la solución; puesto que se crea una partición virtual la misma que será compartida entre los servidores que se realice el clustering
5. Con Heartbeat, la adquisición de la IP se produce en menos de un segundo. DRBD se re-configura en casi 30 segundos y luego dependiendo de la plataforma de hardware y la complejidad de las aplicaciones que se ejecuten en Asterisk puede tardar entre 5-15 segundos para que Asterisk se ponga en marcha en el servidor secundario, se sincronizan los archivos de configuración y está listo para procesar las llamadas. Tiempo total del fail-over es entre 15-20 segundos.
6. No toda la información que existe en la Internet acerca de la instalación y configuración es correcta ya que existen varios procedimientos erróneos.

8. Recomendaciones

1. No confiarse en información que se encuentra en la Internet ya la mayoría de procedimientos son erróneos.
2. Es necesario chequear: nombres de host y direcciones IP, ya que son datos importantes para el correcto funcionamiento de DRBD y Heartbeat.
3. Cuando se usa DRBD es posible que se presente un problema conocido como "Split Brain", se deberá consultar [2] para tener conocimiento de cómo solucionarlo contrario la sincronización no se completara.

9. Bibliografía

[1] Telesoft Integrando Technologies, Building Elastix-1.3 High Availability Clusters with Redfone foneBRIDGE2, DRBD and Heartbeat, http://support.red-fone.com/downloads/elastix/Elastix_HA_Cluster.pdf, 23 de Diciembre de 2008

[2] Cancino Marcos, Cluster de Alta Disponibilidad y Bajo Costo Cluster de Comunicaciones Sobre Linux, http://es.scribd.com/doc/36757812/Cluster-de-Bajo-Costo-en-Linux, Octubre 2002

[3] Marcote Gonzalo, HA Asterisk – Alta disponibilidad Asterisk (I). Heartbeat + Drbd, http://www.gonzalomarcote.com/blog/?p=58, Noviembre 2009

[4] Sandiford Bill, Asterisk Redundancy Using Heartbeat, http://taug.ca/files/TAUG-Heartbeat.pdf, 28 de Febrero de 2008