CP-20
PON 44

ALTA DISPONIBILIDAD EN CLUSTER BAJO CENTOS

Raúl Hernández Palacios[1], Waseem M. Haider[2], Víctor Tomás Tomás Mariano[1], Felipe de Jesús Núñez Cardenas[1]

[1]Universidad Autónoma del Estado de Hidalgo - Escuela Superior Huejutla. Corredor Industrial S/N, Parque de Poblamiento, Huejutla de Reyes, Hidalgo. CP. 43000. Tel. 771-7172000, ext. 5880,5881, E-mail: raulhp@ugr.es, rapalacios81@hotmail.es

[2]Centro de Investigación en Tecnologías de la Información y Comunicaciones, Departamento de Arquitectura y Tecnología de Computadores, Universidad de Granada, España. E-mail: mhaider@atc.ugr.es

Abstract--- The availability of design and implementation of a cluster for high-performance, working with parallel file system using PVFS via a RAID on a Linux platform makes the distribution process more reliable and low cost, and that combining the use of a software RAID-10 and PVFS2 more fault tolerant storage system, avoiding the minimum data loss and working through replication within the RAID system. This is intended to cover any expectation regardless of the type of process running as it will be able to adapt to their needs, and to expand the number of nodes in the cluster to obtain scalability, better performance and greater performance platform.

Keywords--- parallel file system, RAID, replication, cluster.

INTRODUCCION

Actualmente las aplicaciones necesitan un procesamiento alto que las computadoras personales actuales no cubren. La computación de alto rendimiento basada en clúster ofrece capacidades de cómputo a gran escala, en la cual involucra desde un sistema de archivos que se encarga de almacenar los datos en los nodos del clúster, hasta la comunicación de los nodos del clúster. La E/S se ha convertido en un aspecto importante en la computación de alto rendimiento ó altas prestaciones.

ROC&C'2012 / CP-20 PONENCIA RECOMENDADA POR EL COMITÉ DE COMPUTACIÓN DEL IEEE SECCIÓN MÉXICO Y PRESENTADA EN LA REUNIÓN INTERNACIONAL DE OTOÑO, ROC&C'2012, ACAPULCO, GRO., DEL 11 AL 15 DE NOVIEMBRE DEL 2012.

En años recientes, las computadoras paralelas, incluyendo clústers en particular, se están incrementando en cantidad de nodos de procesamiento como forma de conseguir mayores prestaciones globales del sistema.

Una de las investigaciones actuales para este proyecto radica en la mejora de las comunicaciones entre los nodos del clúster, empleando protocolos de comunicación más ligeros y redes Gigabit Ethernet o Infiniband, con la finalidad de que las comunicaciones en un sistema clúster no se convierta en el cuello de botella del sistema global. Otra de las investigaciones que surgirán es el de mantener un sistema, además de alto rendimiento, que el mismo ofrezca tolerancia a cualquier fallo, principalmente en los servidores de datos para que las aplicaciones sean siempre disponibles para los usuarios, a nivel de red es posible también dar esta disponibilidad al sistema, primeramente mediante la técnica Channel Bonding *[1]* del núcleo de Linux que permite dar soporte de alta disponibilidad y balanceo de carga en las interfaces de red de cada nodo del sistema clúster, se obtiene de esta manera una redundancia en el sistema global y las aplicaciones, además de dar disponibilidad a nivel de datos mediante DRBD (*Distributed Replicated Block Device)*.

DESARROLLO

DEFINICIÓN DE CLUSTER

Un clúster se puede definir, según lo expresado en *[2]* como "*la agrupación de ordenadores (generalmente computadoras personales) conectadas mediante una red y trabajando en un problema de tamaño considerable que ha sido dividido para ser procesado de forma paralela*".

Otra definición que puede ser considerada lo bastante concreta en *[3]* que define clúster como "*Un conjunto de máquinas unidas por una red de comunicación trabajando por un servicio en conjunto"*, teniendo en cuenta que el término máquina hace relación únicamente a las computadoras personales.

SISTEMA RAID

RAID (Redundant Array of Independent Disk) *[4]*. La idea central en RAID es replicar datos sobre varios discos de manera que los datos no se pierdan si alguno de los discos falla. Existen diversas configuraciones RAID con diferentes características en rendimiento y formas de replicación de datos. RAID 10 es una combinación de los RAID 0 y 1, que proporciona velocidad y tolerancia al fallo simultáneamente. El nivel de RAID 10 fracciona los datos para mejorar el rendimiento, pero también utiliza un conjunto de discos duplicados para conseguir redundancia de datos. Al ser una variedad de RAID híbrida, RAID 10 combina las ventajas de rendimiento de RAID 0 con la redundancia que aporta RAID 1. Sin embargo, la principal desventaja es que requiere un mínimo de cuatro unidades y sólo dos de ellas se utilizan para el almacenamiento de datos. Las unidades se deben añadir en pares cuando se aumenta la capacidad, lo que multiplica por dos los costes de almacenamiento. La elección de esta versión de RAID se ha dado por la capacidad de discos de almacenamiento que tienen los nodos para la plataforma desarrollada.

SISTEMA PVFS2

PVFS *[5]* es un sistema de ficheros paralelo gratuito para Linux que actualmente se encuentra en su segunda versión. Para implementar alta disponibilidad en PVFS2 es necesario utilizar Heartbeat *[6]* que permite verificar el estado de cada uno de los servidores de datos o metadatos del sistema de almacenamiento. Pero esta alternativa requiere que el almacenamiento esté compartido (con una SAN). Para permitir tolerancia a fallos, evitando compartir el almacenamiento, se ha agregado a PVFS2 replicación de datos en el lado de los servidores como se describe en *[7]*. De esta forma se evita el coste de usar una SAN y se pueden aprovechar los discos incluidos en los nodos del clúster. En la implementación actual de replicación un bloque de datos de archivo es almacenado en dos servidores diferentes con su respectiva copia de bloque.

PVFS en su primera versión implementa una alternativa para lograr la tolerancia a fallos, en *[8]* son consideradas las escrituras grandes con RAID5 y escrituras pequeñas con RAID1.

PVFS2 utiliza, como capa de comunicaciones, la librearía BMI *[9]* (Buffered Message Interface) que soporta diversas tecnologías de red, dentro las que se encuentran Infiniband y Mirynet, que permiten alcanzar grandes anchos de banda y baja latencia en las comunicaciones del sistema, pero a un costo elevado, esto debido a la infraestructura física necesaria para mantener todo el sistema de comunicaciones de un clúster. En cambio la tecnología Ethernet, también soportada por BMI, que son las interfaces de red que comercialmente se encuentran a nuestro alcance y a un ancho de banda conveniente de 1000Mbps.

COMUNICACIÓN CON SSH

SSH ó Secure SHell *[10]* es un protocolo que facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor y que permite a los usuarios conectarse a un host remotamente. A diferencia de otros protocolos de comunicación remota tales como FTP o Telnet, SSH encripta la sesión de conexión, haciendo posible la integridad y seguridad de los datos en la comunicación.

REPLICACIÓN A NIVEL DE BLOQUE

DRBD (*Distributed Replicated Block Device) [11]* da soporte para un almacenamiento distribuido a nivel de bloque a plataformas basadas en Linux. Este sistema de almacenamiento es apoyado en la filosofía RAID1, pero a nivel de red, el funcionamiento radica en realizar el almacenamiento en el servidor primario y las réplicas en el servidor secundario a través de la red, en la Fig. 1 se muestra el esquema de funcionamiento de DRBD.

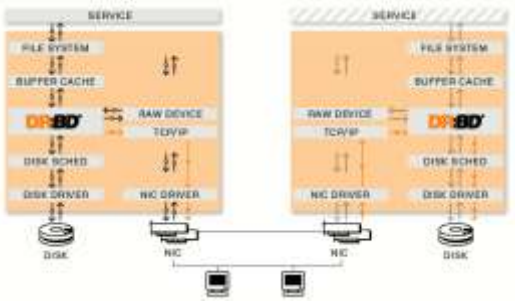

Fig. 1 Replicación de DRBD.

DRBD da soporte a de alta disponibilidad a la tecnología clúster, como se muestra en la figura anterior, DRBD actúa como un módulo que se encarga de almacenar los datos en el disco duro del servidor primario (izquierda) y hacer las réplicas al servidor secundario, actuando el sistema con un modelo primario/secundario o activo/pasivo, por defecto DRBD usa el protocolo TCP/IP para realizar la comunicación, en cuanto al sistema de archivos se puede hacer uso de ext3, ext4 o cualquier sistema de archivos local que soporta Linux, por lo tanto las réplicas a nivel de bloque son transparentes para el usuario.

DESCRIPCIÓN DE LA METODOLOGÍA

El esquema que se muestra en la Fig. 2 consiste en seguir una serie de etapas; la etapa 1 consiste en analizar los requerimientos tanto de hardware y software que se tiene y que se necesite, en la segunda etapa se realiza un diseño global en el cual nos basaremos posteriormente, en la etapa 3 se desarrollara el prototipo mas adecuado a las necesidades en el cual se pondrá en práctica todo lo que se analizó al principio para hacer el prototipo, en la etapa 4 se realizarán las pruebas necesarias para diagnosticar su funcionalidad, y de este punto depende si se concluye o no la última etapa ya que debe de cumplir con todo lo requerido para dar un funcionamiento de calidad o de lo contrario se regresará a la etapa 1, si cumple con los requerimientos planteados se pasa a la etapa 5 que es la entrega final.

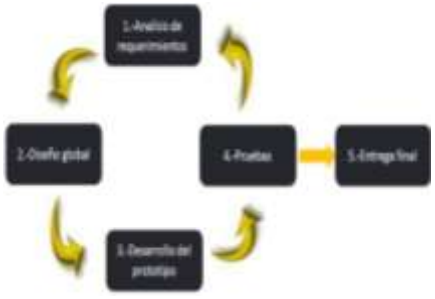

Fig. 2 Esquema grafico sobre la metodologia evolutiva empleada

RESULTADOS EXPERIMENTALES

EVALUACIÓN DE FUNCIONAMIENTO DEL SISTEMA

Para la primera fase, se realizaron las configuraciones necesarias en únicamente dos de los nodos que formarán el clúster, en primer lugar la instalación del sistema operativo Linux CentOS 5.8 con kernel 2.6.18-308.13.1.el5, además de las configuraciones de la conectividad de las tarjetas de red y de la comunicación segura con SSH, para la interconexión de los nodos se usa un Switch con capacidad de soportar Gigabit Ethernet (1000Mbps). De la misma manera se han realizado las configuraciones necesarias de la librería MPI para dar soporte al sistema de ficheros PVFS2.

EXPERIMENTACIÓN CON RAID-10

En la Fig. 3 muestra el panorama de funcionamiento lógico del sistema RAID-10, todas las réplicas del Disco1 y Disco3, se hacen en el Disco2 y Disco4 respectivamente. Esta misma configuración se mantiene en cada uno de los nodos del clúster que tienen el rol de servidor de datos de la configuración de PVFS2.

La experimentación se obtuvo en cada uno de los nodos de clúster para verificar que el sistema RAID que se ha configurado durante la instalación tenga el funcionando adecuado, esta prueba se realizo desde la consola de CentOS. Para ello se utilizo el comando cat /proc/mdstat el cual muestra el estado del RAID, esto se hace solo para verificar que este se muestre como activo.

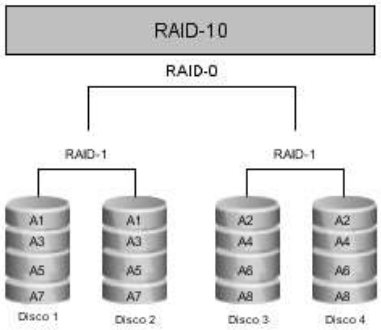

Fig. 3 Funcionamiento lógico de RAID-10

USO DE PVFS2

En PVFS2 mantiene una estructura dónde un nodo es configurado como servidor de datos o servidor de metadatos o ambos roles al mismo tiempo y en el mismo nodo. El servidor de metadatos o metaservers almacenan los metadatos de los archivos, además de los atributos de los archivos creados, la distribución dentro del sistema global. Los nodos clientes del clúster se comunican en primer lugar con el servidor de metadatos para saber la ubicación del archivo que se quiere leer o escribir. Los servidores de datos o dataservers, comúnmente llamados servidores de E/S se encargan de almacenar trozos de los ficheros. Estos trozos se almacenan de una manera siguiendo la técnica Round Robin que se muestra en la Fig. 4. El archivo original se divide en tamaños iguales para hacer la distribución en cada uno de los servidores de E/S.

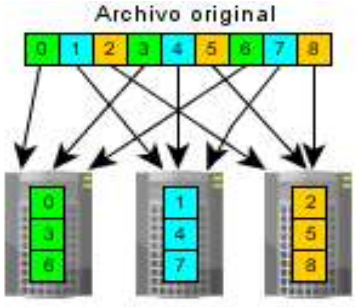

Fig. 4 Esquema de distribución Round Robin.

En la configuración inicial del sistema de alto rendimiento se define como primer prototipo utilizar dos servidores con la configuración de RAID10 en cada nodo, un nodo es configurado como servidor de E/S y servidor de metadatos, y el segundo servidor sólo como E/S. La ventaja del primer servidor es que el servidor de metadatos sólo se tendrá que comunicar con un servidor, con esto se ahorra el tráfico en la red y se evita la congestión en el caso de transferencias grandes de datos. En la Fig. 5 se observa la configuración del sistema global, cada uno de los servidores con su correspondiente configuración RAID de cuatro discos duros, accediendo a este sistema por dos nodos clientes.

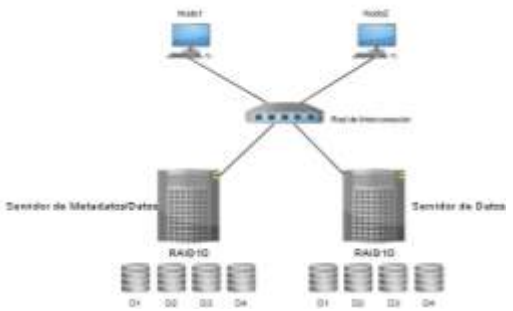

Fig. 5 Esquema de configuración lógica del sistema de almacenamiento.

EXPERIMENTACIÓN CON DRBD

Para hacer uso de este sistema de almacenamiento en Centos 5.8 ha sido necesario instalar DRBD que viene integrado en: drbd84-utils y kmod-drbd84, que refieren la versión 8.4 del sistema, drbd84-utils es necesaria para la administración de los recursos, y kmod-drbd84 se requiere para que el sistema DRBD sea incluido en el Kernel de Linux, además de adquirir las fuentes desde www.drbr.org, para realizar la compilación y posteriormente la configuración necesaria para este desarrollo. Dando como resultado la siguiente configuración:

```
resource r0 {
  on centos1 {
    device      /dev/drbd1;
    disk        /dev/sda1;
    address     192.168.1.20:7789;
    meta-disk   internal;
  }
  on centos2 {
    device      /dev/drbd1;
    disk        /dev/sda1;
    address     192.168.1.30:7789;
    meta-disk   internal;
  }
}
```

La configuración anterior es almacenada en el archivo /etc/drbd.d/r0.res, en /etc/drbd.d es el directorio dónde se almacenan todos los recursos que son compartidos, en este caso el recurso compartido es denominado r0.res que incluye:

- on centos1 y on centos2: se refiere a que son dos los nodos del cluster que serán configurados, centos1 como nodo primario y centos2 como nodo secundario.

- device /dev/drbd1: es el dispositivo lógico que se crea al momento de compilar DRBD, este dispositivo es declarado en los dos nodos.

- disk /dev/sda1: físicamente este el dispositivo donde serán almacenados todos los archivos tanto en el nodo primario como en el nodo secundario existe este dispositivo físico.

- address 192.168.1.20:7789 y 192.168.1.30:7789: son las direcciones IP de los nodos primario y secundario respectivamente, y ambas declaran utilizar el mismo puerto para la comunicación.

- meta-disk internal: de esta manera se le dice al sistema que los metadatos serán almacenados en el mismo disco físico en cada nodo, los metadatos dan información referente a cada archivo como lo hace PVFS2.

DRBD utiliza tres modos o protocolos para la comunicación, que al mismo tiempo deben de mantener la sincronización de los datos en ambos servidores, para que en el caso de que el nodo primario falle, el nodo secundario mantiene tanto el servicio, como la coherencia de los datos. Para esto se utiliza el Protocolo C (*Synchronous replication protocol*) que garantiza la sincronización mediante una confirmación del nodo secundario hasta que las escrituras son realizas en el disco de ambos nodos.

DRBD CON HEARTBEAT

Para conocer el estado de los nodos del clúster, en este trabajo se ha utilizado el software Heartbeat o latido de corazón, su uso ha sido fundamental para conocer el estado de conexión del nodo primario desde el nodo secundario, de tal manera que, al momento de que el nodo secundario no detecta la actividad de servicio del nodo primario, este nodo secundario actúa como nodo primario y da el servicio hasta que el nodo primario tenga nuevamente la actividad para dar el servicio.

CONCLUSIONES Y TRABAJO FUTURO

En esta primera implementación se ha podido comprobar la alta disponibilidad de los datos de dos maneras. En primer lugar, implementando los sistemas RAID en el nivel 10, que, al momento de un fallo de disco los datos estaban disponibles en el disco duro espejo, manteniendo la disponibilidad de la información a nivel local de almacenamiento de cada uno de los servidores de datos. En segundo lugar, mediante el sistema de almacenamiento distribuido DRBD, que ha dado soporte en el fallo del servidor primario con el uso del software Heartbeat. Con la configuración global del sistema se considera una amplia redundancia de los niveles del clúster como: almacenamiento local con las réplicas de datos entre los discos en el mismo nodo y con las réplicas de datos entre los nodos primario y secundario del clúster, consideramos que este nivel de configuración dará soporte necesario para la fiabilidad y disponibilidad de los datos.

Como trabajo futuro se realizará un análisis importante en el nivel de comunicaciones del sistema, debido a que es necesario que la plataforma también mantenga un soporte de fiabilidad y disponibilidad en los enlaces de red que conectan los dos nodos del clúster. Para esto se pretende hacer uso en cada nodo de dos tarjetas de red configuradas con la Técnica Channel Bonding nativa en Linux, de tal manera que se cree un enlace simbólico que agregue los dos canales de comunicación de cada nodo y con esto se logre un ancho de banda teórico de 2000Mbps. Además, se incrementará el número de nodos del clúster y así también se logre una mayor disponibilidad de las aplicaciones y los datos, y un incremento en el rendimiento global del sistema ejecutando las aplicaciones en un menor tiempo posible.

BIBLIOGRAFÍA

[1] *C. University, «Parallel Virtual File System,» 2011. [En línea]. Available: www.pvfs.org.*

[2] *«Heartbeat,» [En línea]. Available: http://www.linux-ha.org/.*

[3] *E. Nieto, R. Hernández, H. Camacho, A. Díaz, M. Anguita Y J. Ortega, «Replicación de Datos en PVFS2 para Conseguir Tolerancia a Fallos,»* XX Jornadas de Paralelismo, *2007.*

[4] *P. Carns, W. I. Ligon, R. Ross y P. Wyckoff, «BMI: a network abstraction layer for parallel I/O,»* Parallel and Distributed Processing Symposium. Proceedings. 19th IEEE International, *2005.*

[5] *D. Barrett, R. Silverman y R. Byrnes, SSH, The Secure Shell: The Definitive Guide. Second Edition, O'Reilly Media, 2005.*

[6] *D. A. Patterson, G. Gibson y R. H. Katz, «A case for redundant arrays of inexpensive disks (RAID),»* ACM SIGMOD international conference on Management of data, *vol. 17, nº ISBN:0-89791-268-3, pp. 109 - 116, 1988.*

[7] *A. F. Diaz, J. Ferreira, J. Ortega, A. Cañas y A. Prieto, «CLIC: Fast Communication on Linux Clusters,»* Second IEEE International Conference on Cluster Computing (CLUSTER'00), *p. 365, 2000.*

[8] *D. H. Milone, A. A. Azar y L. H. Rufiner, «Supercomputadoras basadas en "Clusters" de PCs,»* Revista Ciencia, Docencia y Tecnología, *vol. XIII, nº 25, pp. pp. 173-208, 2002.*

[9] *M. Lauria y M. Pillai, «A high performance redundancy scheme for cluster file systems,»* Proceedings of the 2003 International Conference on, *pp. 216-223, 2003.*

[10] *C. V. Juan Esteban y M. A. Cristian Alejandro,* Implementación de un Servidor Web Apache, *Talca, Chile, 2007.*

CURRICULUM

Raúl Hernández Palacios, Maestro en Ingeniería de Computadores y Redes, egresado de la Universidad de Granada, España en 2007. Actualmente como Profesor Investigador y Perfil Promep de la Escuela Superior de Huejutla de la Universidad Autónoma del Estado de Hidalgo, incorporado a la Licenciatura de Sistemas Computacionales de la misma Universidad. En la actualidad realizando estancia de investigación en el Centro de Investigación en Tecnologías de la Información y Comunicación (CITIC-UGR) de la Universidad de Granada. Interés en área de investigación de Sistemas Distribuidos; sistemas de archivos en red; optimización de comunicaciones eficientes en clúster de computadoras con mecanismos como Multicast, Socket Direct Protocol (SDP), Remote Direct Memory Access (RDMA); alta disponibilidad en sistemas clúster a nivel de red, servidor y almacenamiento (iSCSI).

Waseem M. Haider. Doctor en Ingeniería de Computadores por la Universidad de Granada, España. Actualmente investigador adjunto al Centro de Investigación en Tecnologías de la Información y Comunicación (CITIC-UGR) de la Universidad de Granada. Desarrollo de línea de investigación en optimización de las prestaciones de redes y alternativas de externalización de los protocolos de comunicación en coordinación con el Departamento de Arquitectura de Computadores de la misma Universidad.

Víctor Tomas Tomás Mariano es Maestro en Ciencias Computacionales egresado de la Universidad Autónoma del Estado de Hidalgo (UAEH), en Pachuca de soto, Hidalgo, México en el 2007. Profesor investigador titular, con perfil PROMEP y está incorporado a la Licenciatura en Sistemas Computacionales de la Escuela Superior Huejutla UAEH, en Huejutla de Reyes, Hidalgo, desde el año 2008. Su interés en áreas de investigación incluye: planeación de trayectorias, *maze search problem*, laberintos virtuales 2D y 3D, computación inteligente, Entre las publicaciones más recientes están:
Algoritmo para la Generación de Laberintos de Conexión Múltiple en 2D. Vigesimosegunda Reunión Internacional de Otoño, de Comunicaciones, Computación, Electrónica, Automatización, Robótica y Exposición Industrial ROC&C2011. La Conjunción de las Tecnologías Digitales en las Redes Inteligentes. IEEE Sección México. Acapulco, Guerrero, México. Propuesta para la generación de laberintos ampliados en 2D. Simposio Iberoamericano Multidisciplinario de Ciencias e Ingeniería SIMCI 2011. Zempoala, Hidalgo.

Felipe de Jesús Núñez Cárdenas es Maestro en Ciencias de la Administración con Especialidad en Informática egresado del Centro de posgrado en Administración e Informática A. C. (CPAI). Profesor Investigador Asociado, incorporado a la Licenciatura en Sistemas Computacionales de la Escuela Superior Huejutla UAEH, en Huejutla de Reyes, Hidalgo, desde el año 2011. Su interés en áreas de investigación incluye: Ingeniería de Software, Sistemas de Información, Base de Datos, Minería de Datos, ha publicado:
Portal Integral de Educación Superior en el XIV Congreso Internacional Sobre educación electrónica, virtual y a distancia. TELEEDU 2007 (Bogotá Colombia)