

Nube de almacenamiento distribuida con GlusterFS: Volumen distribuido y replicado

* Sistemas Concurrentes y Distribuidos

Coronado Huamán Ayrton Fabio
Escuela de Ciencia de la Computación
Facultad de Ciencias-UNI
Lima, Perú
acoronadoh@uni.pe

Flores Huamaní Luis Angel
Escuela de Ciencia de la Computación
Facultad de Ciencias-UNI
Lima, Perú
lfloresh@uni.pe

Barrientos Porras Herlees Brayan
Escuela de Ciencia de la Computación
Facultad de Ciencias-UNI
Lima, Perú
herlees.barrientos.p@uni.pe

Abstract—Este documento es el estudio y implementación de una nube de almacenamiento distribuido en Ubuntu Linux 20.04 del tipo volumen distribuido replicado con GlusterFS.

Index Terms—nube, almacenamiento, distribuido, volumen, replicado, GlusterFS, Ubuntu.

I. INTRODUCCION

Vivimos en un mundo donde la información esta creciendo de forma impredecible y almacenarla en un lugar centralizado pueda generar muchos problemas, no solo problemas de gestión de información, también problema de seguridad, logística, baja eficiencia y productividad.

Ante esta problemática el almacenamiento distribuido surge como una solución ideal.

En este artículo tenemos como objetivo estudiar, diseñar e implementar este tipo de sistemas mediante el software gratuito y Open Source GlusterFS, con un servidor desarrollado en Nodejs.

II. ESTADO DEL ARTE

A. Implementación de una Arquitectura Tecnológica basada en Cloud Computing como soporte al portafolio de proyectos profesionales de la EISC

Tiene como objetivo implementar una arquitectura tecnológica utilizando la tecnología de prestación de servicios de Cloud Computing que soporte los proyectos profesionales de la Escuela de Ingeniería de Sistemas y Computación (EISC). El proyecto contiene investigación de los conceptos relacionados a Cloud Computing, contemplando definiciones, historia, modelos, tipos de soluciones y metodologías utilizadas en implementaciones de arquitectura.

Según los autores el proyecto obtuvo el visto bueno de consultores externos, los costos fueran principalmente de recursos humanos, ya que el software empleado era gratuito y la infraestructura fue proveída por una empresa privada.

B. Características de la Nube

- Tercerización de recursos. - El proveedor de la nube asume la responsabilidad para la adquisición de hardware y mantenimiento de este.
- Utilidad informática. - El cliente solicita recursos adicionales, según sea necesario, y de manera similar libera estos recursos cuando no se necesitan de acuerdo a sus demandas de autoservicio.
- La escalabilidad y la autogestión. - Las nubes son como programadores libres que se ocupan de los problemas de escalabilidad, estas nubes ofrecen el cambio de tamaño de una manera automática de los recursos de hardware virtualizados. La escalabilidad y la autogestión es más simple a través de un único dominio administrativo, pero puede surgir ciertos problemas.
- Alta disponibilidad. - De acuerdo a los SLA que se establezcan con el proveedor, se pueden asegurar hasta un 99.95% de disponibilidad.
- Recuperación de desastres. - Esta recuperación ante desastres en la nube son copias de seguridad y restauración estratégicas que implica el almacenamiento y mantenimiento de copias de los registros que se dan en un entorno Cloud como medida de seguridad.

C. Modelos de Despliegue

- Cloud pública. - Plataforma abierta al público en general. Puede ser administrada mediante una empresa, alguna entidad académica o gubernamental, o alguna mezcla de ellas. Las instalaciones se encuentran en las instalaciones del proveedor.
- Cloud privada. - La infraestructura de la nube privada está estructurada para el uso exclusivo de una organización que puede ser utilizada por más de una unidad de negocio. No obstante, la gestión y operación puede estar encargada de la misma organización o algún tercero que se encuentre tanto en los interiores o exteriores de las instalaciones.
- Cloud híbrida. - La infraestructura está conformada por más de un tipo de Cloud. Aprovecha el beneficio que

ambos tipos de Cloud brindan con la finalidad de manejar la carga de trabajo de la organización.

- Cloud comunitaria. - Modelo que consiste en dos o más tipos de Cloud. Utilizada para cubrir necesidades en común de una comunidad de organizaciones en específico. Puede ser gestionada u operada por uno o más empresas tanto en los interiores o exteriores.

III. HERRAMIENTAS

A. GlusterFS

GlusterFS es un sistema de archivos de almacenamiento escalable y distribuido conectado a la red que le permite agrupar recursos de almacenamiento de varias máquinas. A su vez, esto permite manipular diversos dispositivos de almacenamiento que se distribuyen entre muchas computadoras como una unidad única y más potente.

B. NodeJS

Node.js es un entorno en tiempo de ejecución multi-plataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google.

C. Ubuntu 20.04

Ubuntu es un sistema operativo de software libre y código abierto. Es una distribución de Linux basada en Debian. Actualmente corre en computadores de escritorio y servidores. Está orientado al usuario promedio, con un fuerte enfoque en la facilidad de uso y en mejorar la experiencia del usuario. Actualmente la versión 20.04 estable, es la que contiene los paquetes mas actualizados de GlusterFS.

D. Flutter

Flutter es un SDK de código fuente abierto de desarrollo de aplicaciones móviles creado por Google. Suele usarse para desarrollar interfaces de usuario para aplicaciones en Android, iOS y Web.

IV. MODELO DE INVESTIGACIÓN

A. Conceptos de Glusterfs

En Glusterfs se tienen tres conceptos fundamentales:

- Sistema de archivos distribuido.- Es un sistema de archivos en el que los datos se distribuyen entre varios nodos y los usuarios pueden tener acceso a estos datos sin conocer la ubicación real de los archivos. El usuario no experimenta la sensación de acceso remoto.
- Brick.- Un brick (ladrillo) es básicamente cualquier directorio o partición que será compartido y que está asignado a un volumen.
- Volumen.- Un volumen es una colección lógica de Bricks. Las operaciones se basan en los diferentes tipos de volúmenes creados por el usuario.

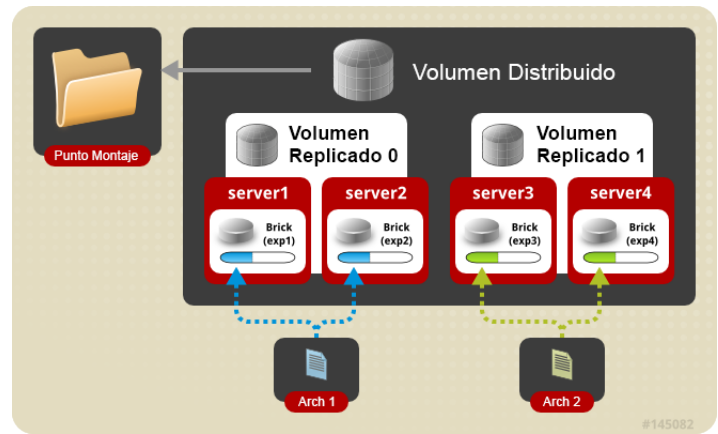


Fig. 1. Volumen distribuido replicado

B. Concurrencia en Nodejs

Node.js funciona con un modelo de evaluación de un único hilo de ejecución, usando entradas y salidas asíncronas las cuales pueden ejecutarse concurrentemente en un número de hasta cientos de miles sin incurrir en costos asociados al cambio de contexto. Este diseño de compartir un único hilo de ejecución entre todas las solicitudes atiende a necesidades de aplicaciones altamente concurrentes, en el que toda operación que realice entradas y salidas debe tener una función callback. Un inconveniente de este enfoque de único hilo de ejecución es que Node.js requiere de módulos adicionales como cluster para escalar la aplicación con el número de núcleos de procesamiento de la máquina en la que se ejecuta.

```
const cluster = require('cluster');
const http = require('http');
const numCPUs = require('os').cpus().length;

if (cluster.isMaster) {
  console.log('Master ${process.pid} is running');

  // Fork workers.
  for (let i = 0; i < numCPUs; i++) {
    cluster.fork();
  }

  cluster.on('exit', (worker, code, signal) => {
    console.log('worker ${worker.process.pid} died');
  });
} else {
  // Los workers pueden compartir cualquier
  // conexión TCP
  // En este caso es un servidor HTTP
  http.createServer((req, res) => {
    res.writeHead(200);
    res.end('hello world\n');
  }).listen(8000);

  console.log('Worker ${process.pid} started');
}
```

```

$ node server.js
Master 3596 is running
Worker 4324 started
Worker 4520 started
Worker 6056 started
Worker 5644 started

```

C. Modelo a desarrollar

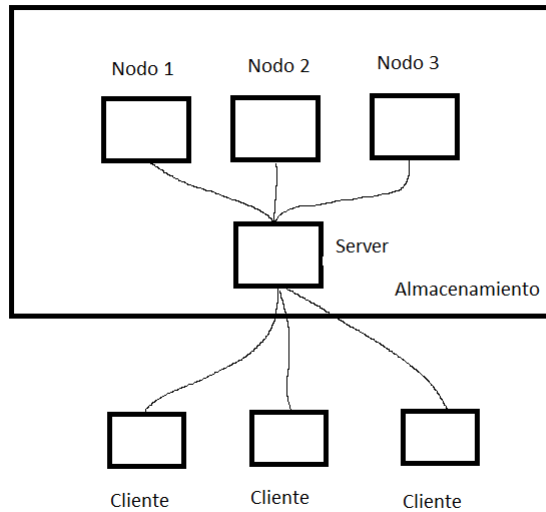


Fig. 2. Implementación

Cada nodo contiene un brick, el servidor tendrá el montaje de almacenamiento.

REFERENCES

REFERENCES

- [1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.
- [2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [4] K. Elissa, "Title of paper if known," unpublished.
- [5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.
- [6] Y. Yoroazu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetism Japan, p. 301, 1982].
- [7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.